

3 1176 00518 2382

NASA Contractor Report 165947

NASA-CR-165947
19820023450

FUNCTIONAL DESIGN TO SUPPORT CDTI/DABS FLIGHT EXPERIMENTS

Tsuyoshi Goka

Contract NAS1-16802
July 1982

NASA

LIBRARY COPY

JUG 31 1982

LANGLEY RESEARCH CENTER
LIBRARY, NASA
HAMPTON, VIRGINIA



NF01954

NASA Contractor Report 165947

FUNCTIONAL DESIGN TO SUPPORT CDTI/DABS FLIGHT EXPERIMENTS

Tsuyoshi Goka

**Analytical Mechanics Associates, Inc.
Mountain View, California**

**Prepared for
Langley Research Center
under Contract NAS1-16802**



**National Aeronautics and
Space Administration**

**Langley Research Center
Hampton, Virginia 23665**

July 1982

1182-31326#1

This Page Intentionally Left Blank

FOREWORD

This effort for developing a functional design of avionics to support CDTI/DABS flight experiments was supported under NASA Contract No. NAS1-16802 by Langley Research Center, Hampton, VA. The project technical monitor was David Williams of NASA Langley Research Center. Technical discussions with David Williams, James Kelly, Major C. Lee, and David Holmes of Langley Research Center, and John Scardina and Ernest Lucier of FAA Headquarters are gratefully acknowledged.

This Page Intentionally Left Blank

FUNCTIONAL DESIGN TO SUPPORT

CDTI/DABS

FLIGHT EXPERIMENTS

Tsuyoshi Goka

Analytical Mechanics Associates, Inc.
Mountain View, California 94043

SUMMARY

The objectives of this project are to (a) provide a generalized functional design of CDTI avionics using the FAA developed DABS/ATARS ground system as the "traffic sensor", (b) specify software modifications and/or additions to the existing DABS/ATARS ground system to support CDTI avionics, (c) assess the existing avionics of a NASA research aircraft in terms of CDTI applications, and (d) apply the generalized functional design to provide research flight experiment capability.

In this report, DABS Data Link Formats are first specified for CDTI flight experiments. The set of CDTI/DABS Format specifications becomes a vehicle to coordinate the CDTI avionics and ground system designs, and hence, to develop overall system requirements. The report is the first iteration of a system design and development effort to support eventual CDTI flight test experiments.

This Page Intentionally Left Blank

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
II. PRELIMINARY CDTI UPLINK AND DOWNLINK DATA FORMAT SPECIFICATIONS	5
Background Information	5
CDTI Data Format Specifications.	10
CDTI Downlink Data Formats	13
CDTI Uplink Data Formats	19
III. CDTI AVIONICS	31
Background	31
Proposed CDTI Avionics Configuration	36
CDTI Avionics Components	41
Airborne Computer Software Modules for CDTI Applications	59
IV. DABS/ATARS BASED CDTI GROUND SYSTEM	71
Background	71
Modification Philosophy	77
CDTI Modification Requirements	78
Proposed CDTI Ground System Modifications	85
V. SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS	99
Summary	99
Conclusions and Recommendations	101
APPENDIX A: Brief Description of the Discrete Address Beacon System (DABS).	111
APPENDIX B: DABS/ATARS Based CDTI Avionics Interface with the TCV B-737 01 Upgrade System	135
REFERENCES.	157

This Page Intentionally Left Blank

LIST OF FIGURES

Figure	Page
1. CDTI Subsystems Based on DABS/ATARS	1
2. Components Interface for Downlink Operation	12
3. Pilot Request for CDTI Flight Initiation or Termination Format for Comm-B Message Field	14
4. Pilot Request for CDTI Traffic Type Modification for Comm-B Message Field	16
5. Two Basic Post-Flight Data Storage Configurations	18
6. Downlink Information (1st Segment of the ELM)	20
7. Downlink Information (2nd through Last Segment of the ELM).	20
8. General Map and Traffic Identification Information (1st ELM Segment)	21
9. General Map and Traffic Identification Information (2nd through last ELM Segments).	21
10. DABS and Runway Coordinate Systems	23
11. General Structure of Own and Traffic Data Format	25
12. Header Information and Own Aircraft Data Format (1st ELM Segment)	27
13. Traffic Data Format	27
14. AIDS Data Link/ATARS Computer System	32
15. NASA Langley TCV/CDTI Flight Experiment System	33
16. TCV/CDTI Interface Block Diagram and Timing Sequence	34
17. Two Candidate Systems for Retrofitting CDTI Avionics	37
18. CDTI Configuration I - Existing Airborne Computer NOT Modifiable.	38
19. CDTI Configuration II - Existing Airborne Computer Modifiable	40
20. Data Interface for the Mode S Transponder	42
21. Mode S Transponder Digital Section Block Diagram	43
22. SM Interface Timing Diagram for DABS Transactions	45
23. Flow Chart for a Conceptual Operation of the SM Interface Units	47
24. Mode S Transponder Standard Message Interface Concept	48
25. ELM Communications	50
26. ELM Interrogator/Transponder Interface Time-line.	51
27. ELM Interface Timing Diagram	52
28. Conceptual Logic Flow for the ELM Transponder Interface Unit	56
29. Essential CDTI On-Board Computational Flow	61
30. ATARS-Sensor Interface Diagram	72

LIST OF FIGURES (CONT')

31.	Sector Oriented Task Sequencing	73
32.	Computation Flow of Essential Tasks for ATARS Sector Processing	74
33.	Time Windows for ATARS Processing Tasks with Respect to ATARS Sectors	76
34.	Illustration of CDTI Coverage Volume	82
35.	Along Track-Cross Track Geometry	83
36.	Proximity Type Transition Diagram	85
37.	CDTI ELM Data Format Packing	88
38.	ATARS and CDTI Coarse Screen Task	91
39.	Modified Sector Processing	93
40.	Macro Flow Chart for the CDTI Processing	94
41.	ATARS/CDTI Sector Algorithm	96
42.	DABS/CDTI Software Design Validation Facility	105
A.1	DABS Computer Architectural Block Diagram	114
A.2	DABS Sensor Functional Block Diagram	116
A.3	Time Line Depiction of DABS Scheduling	119
A.4	ELM Scheduling	123
A.5	DABS Capacity Plots	125
A.6	Summary of DABS Uplink Formats	128
A.7	Summary of DABS Downlink Formats	129
B.1	TCV 01 Upgrade System Architecture	137
B.2	Upgrade CIU Major Functional Elements	140
B.3	Example Input Timing Sequence	142
B.4	Digital Word Translation	143
B.5	Expected Display System Configuration	144
B.6	Overall CDTI Avionics Interface with the TCV B-737 01 Upgrade System	148
B.7	Schematic Diagram of Input Process	151
B.8	TIU Storage Buffer Organization	152
B.9	Schematic Diagram Dispicting the Output Process from the TIU to CIU	154

LIST OF TABLES

Table		Page
1.	Own Data Bit Specification	6
2.	Position Data Bit Specification.	7
3.	Supplementary Proximate Data Bit Specification	8
4.	Comm-B Data Link Character Sets.	14
5.	Comparison of AIDS and TCV/CDTI	35
6.	Relationship between Uplink/Downlink Timing and SM Interface Timing	46
7.	Definition of ELM Interface Signals	54
8.	Desirable CDTI Display Contents.	58
9.	Threat Parameters.	66
10.	Memory Requirement for On-Board CDTI Implementation	69
11.	Coverage Volume and Traffic Numbers	81
12.	Example of the CDTILST List (On-Board Track File)	95
13.	CDTI System Research Areas	103
A.1	DABS Field Descriptions	130
A.2	DABS Message Types	131
A.3	Projected ATC Services Supported by the DABS Data Link . . .	134

This Page Intentionally Left Blank

List of Acronyms and Abbreviations

AC	Aircraft
ADS	A-Definition Subfield; a subfield of MA
AIDS	Airborne Intelligent Display System
ARINC	Aeronautical Radio, Inc.
ASCII	American Standard for Communication
ATARS	Automatic Traffic Advisory and Resolution Service
ATAS	Automatic Traffic Advisory Service
ATC	Air Traffic Control
ATCRBS	Air Traffic Control Radar Beacon System
BDS	B-Definition Subfield; a subfield of MB
bps	bit/sec
CDS	C-Definition Subfield; a subfield of MC
CDTI	Cockpit Display of Traffic Information
Comm-A	Mode S (DABS) uplink format containing 56 bits of message field, MA. Includes DABS surveillance function bits.
Comm-B	Mode S (DABS) downlink format containing 56 bits of message field, MB. Includes DABS surveillance function bits.
Comm-C	Mode S (DABS) uplink format containing 80 bits of message field, MC. Does not include DABS surveillance function bit.
Comm-D	Mode S (DABS) downlink format containing 80 bits of message field, MD. Does not include DABS surveillance function bit.
CPA	Closest Point of Approach
CPU	Central Processing Unit
CRT	Cathode Ray Tube
CTD	Constant Time Delay
CTP	Constant Time Predictor
DABS	Discrete Address Beacon System
DAS(DDAS)	(Digital) Data Acquisition System
DDS	D-Definition Subfield; a subfield of MD.
DME	Distance Measuring Equipment
EADI	Electronic Attitude Director Indicator
EFR	Electronic Flight Rules
EHSI	Electronic Horizontal situation Indicator (or MFD); Multi-Function Display)

ELM	Extended Length Message
ELMIU	Extended Length Message Interface Unit
FAA	Federal Aviation Administration
FIFO	First In First Out
HDG	Heading (magnetic or true North)
Hz	Hertz (cycle/sec)
IAS	Indicated Airspeed
I/F	Interface
IFR	Instrument Flight Rules
INS	Inertial Navigation System
I/O	Input and/or Output
IPC	Intermittent Positive Control
KBRD	Keyboard
lsb	Least significant bit
MA	56 bits message field contained in a Comm-A format
MB	56 bits message field contained in a Comm-B format
MC	80 bits message field contained in a Comm-C format
MD	80 bits message field contained in a Comm-D format
MLS	Microwave Landing System
MODE A	ATC Radar Transponder
MODE C	Altitude reporting ATC radar transponder
MODE S	DABS format radar transponder
MOTAM	Mission Oriented Terminal Area Model (an ATC simulation facility at NASA/Langley Research Center)
MSL	Mean Sea Level
NASA	National Aeronautics and Space Administration
NCDU	Navigation and Control Display Unit
PD	Packed Discretes
PWI	Proximity (or Pilot) Warning Indicator
RF	Radio Frequency
RNWX	Runway
S/D	(Digital) Serial Data
SM	Standard Message (either MA or MB)
SMIU	Standard Message Interface Unit
SPBP	Split Phase Bi-Polar (bus)
TACAN	Tactical Air Navigation
TAS	True Airspeed
TAU	Closure time defined as range/range rate

TCG	Time Code Generator
TCV	Terminal Configured Vehicle
TIU	Transponder Interface Unit
VFR	Visual Flight Rules
VOR	Very high frequency Omni-directional Range
WFI	Wait For Interrupt
WXR	Weather Radar

I

INTRODUCTION

The joint NASA and FAA Cockpit Display of Traffic Information (CDTI) system research program is at the stage where the emphasis is shifting from a relatively "clean" (idealistic) simulation environment toward the real-world flight simulation environment. This implies that various research tools developed previously must be translated into concrete hardware and software system elements within the realization bounds imposed by currently available support elements.

This report addresses the first step toward the goal of defining CDTI system elements. Specifically, it explores the functional design requirements and specifications for both airborne and ground equipment to support a flight test program. The requirements are based on the FAA developed DABS/ATARS ground system as the CDTI "traffic sensor."

This (traffic) sensor cannot be classified as an airborne sensor in the usual sense such as a body rate sensor or a MLS receiver. In the latter case, the signals are one way, passive, and self-contained. For the DABS/ATARS system, the sensor and the ground system must work in a cooperative and active manner through the DABS communication channel.

The above argument implies that the CDTI system consists of three subsystems: the DABS/ATARS ground system, the DABS communication channel between aircraft and ground, and the airborne CDTI avionics. The RF signals (1030 MHz for uplink and 1090 MHz for downlink), encoding/decoding techniques, and signal length can be considered as the hardware portion of the communication subsystem. The CDTI Data Format specifications constitutes the software portion. Figure 1 shows the CDTI subsystems whose functions are listed below.

- CDTI Ground System

- Downlink Message Processing
- Surveillance/Tracker Algorithm
- General Traffic Sorting and File Update

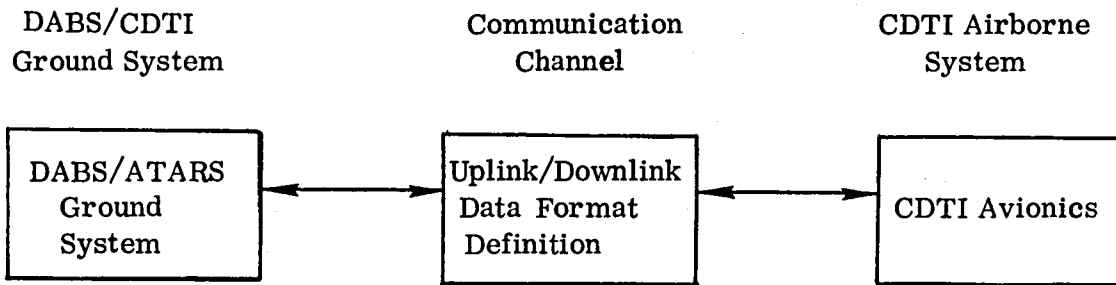


Figure 1. CDTI Subsystems Based on DABS/ATARS

- CDTI Traffic Selection
- Uplink Message Processing and Uplink
- Post-flight Analysis Data Storage
- Communication Channel
 - Pilot CDTI Request or Option Request Format
 - Downlink Information Format
 - Uplink Information Format
- CDTI Airborne System
 - Uplink Message Processing
 - Compensation/Prediction/Coasting
 - Pilot Interface (Keyboard and Display)
 - Downlink Message Processing
 - Post-flight Analysis Data Storage

The point of view taken for the functional design presented in this report is as follows. Because the communication channel links the ground and airborne subsystems together, most of the system requirements and specifications are automatically determined when the communication formats are specified; therefore, the communication channel is specified first (Section II). Then, the airborne subsystem (Section III) and the ground subsystem (Section IV) are specified.

This report is addressed to a diverse audience of three groups of engineers with different backgrounds, viz., the DABS Data Link Format development group, the CDTI avionics development group, and the DABS/ATARS

ground system development group. For this reason, background material is given at the beginning of each chapter. Some of the material is duplicated among the chapters for easier reading. Brief descriptions of the DABS sensor, its signal-in-space characteristics and the DABS messages are collected in Appendix A.

References used for the report are listed at the end. They are grouped according to approximate subject boundaries. Therefore, unless specifically cited, the context should guide the reader to applicable references.

This Page Intentionally Left Blank

II

PRELIMINARY CDTI UPLINK AND DOWNLINK DATA FORMAT SPECIFICATIONS

Background Information

ATARS Data Format Automatic Traffic Advisory and Resolution Service (ATARS), which is an upgraded version of the Intermittent Positive Control concept, is a ground based collision avoidance system. This system obtains the DABS[†] surveillance data, examines the data for proximate or threat situations, computes commands to avoid critical situations, sends (up-linking) appropriate messages (advisories) to the concerned aircraft, and displays the uplinked advisories to the pilot.

The ATARS advisories are uplinked by utilizing 56 bits of MA field contained in DABS Data Format Nos. 20 and 21 (Fig. A.6). Because there are several types of advisories, eight bits out of the available 56 bits are used to specify what types of advisories are contained in the remaining 48 bits. The eight bit subfield is designated as ADS (A-Definition Sub-field), and it serves as an unpacking instruction.

CDTI related ATARS advisories are packed into 24 bits each. Thus, two advisories can be uplinked per transaction. Tables 1 through 3 show the specifications for three applicable advisories - Own Data, Position (or Proximate) Data, and supplementary Proximate Data. Combinations of these and others are uplinked, depending on the aircraft avionics display capability.

System Critique A major functional difference exists between ATARS and CDTI research systems. The main concern of the former system is assessing the threat situation represented by the relative aircraft kinematics (collision avoidance is a relative kinematic problem, albeit a complex one).

[†] The background material on the operational characteristics of DABS ground system and the DABS signal-in-space are given in Appendix A.

TABLE 1

OWN DATA BIT SPECIFICATION [21]

Field	Bits	Interpretation
Own Ground Track Heading	7	2,8125° 1sb* Referenced to magnetic North of DABS site
Own Ground Track Speed	7	10 knot 1sb
Own Ground Track Turn Rate	4	1°/sec 1sb (Two's Complement with Right Positive)
Own ATARS Capability	2	4 Levels possible (only 01 used at present)
Seam Bit	1	Multi (1) or Single (0) ATARS sites can uplink data
Antenna Scan Period	3	1 sec 1sb added to 4 seconds (thus 4 to 11 seconds possible)
Total	24	

* least significant bit

TABLE 2

POSITION DATA BIT SPECIFICATION [21]

Field	Bits	Interpretation
Clock Bearing (CB)	4	1 o'clock (0001) through 12 o'clock (1100)
Fine Bearing (FB)	3	Bearing to target = $[(CB) - 1/2] \times 30^\circ$ $+ (FB) \times 3.75^\circ$
Altitude Zone	2	Bit 1: Equal or above (1) or Below (0) Bit 2: Co-altitude (0 to 500') (1) or Not (600' or more) (0)
Relative Altitude (RA)	3	If Co-altitude: 100' 1sb If not Co-altitude: 200' 1sb beyond 600' (thus 600' to 2000')
Range	6	0.2 nm 1sb
Coarse Heading (CH)	3	N(000), NE(001), thru NW(111)
ATC Control	1	Controlled (1) or Not Controlled (0)
ATARS Equipped	1	ATARS equipped (1) or Not (0)
Most Critical Flag	1	Most critical advisory (1) or Not (0)
Total	24	

TABLE 3

SUPPLEMENTARY PROXIMATE DATA BIT SPECIFICATION [21]

Field	Bits	Interpretation
Track Number	3	0 through 7
Fine Heading (FH)	4	Heading of target = $[(CH) - 1/2] \times 45^\circ$ $+ (FH) \times 2.8125^\circ$ [Note: CH is contained in position data]
Velocity	7	10 knot 1sb
Turn Type of Aircraft	3	Bit 1: Turn (1) or Straight (0) Bit 2: Right (1) or Left (0) Bit 3: Strong (1) or Weak (0)
Vertical Speed of Aircraft	6	200 FPM 1sb (Two's com- plement with positive upward)
Spare	1	Set to Zero
Total	24	

The CDTI research system concerns include relative kinematics as well as navigation and guidance functions. For example, some of the potential CDTI applications are route merging, in-trail following (or station keeping) along a route, and route crossing. The tasks are performed by the pilot relative to a displayed map geometry (air route structure, final approach path, and so on) and relative to surrounding traffic.[†] This implies that information displayed on a CDTI must be as accurate as possible (the exact accuracy requirement is not known at this time). If designed and integrated correctly, a CDTI can serve as a major cockpit instrument from which the pilot can obtain diverse information.

With this background in mind, the following remarks can be made concerning the ATARS advisories and display.

- (a) Own Data does not contain position information. This implies that on-board navigation variables are needed to locate Own aircraft's position with respect to the fixed map geometry.
- (b) Own air derived track angle, ground speed, and turn rate would be more accurate than those derived by the ground system.
- (c) The target ground speed and altitude rate are not contained in the Position Data. (These are contained in Supplementary Position Data).
- (d) Maximum relative range is small, and the resolution (or quantization) accuracy is coarse (up to 12.6 nmi at 0.2 nmi resolution).
- (e) It would be more useful to include the turn rate rather than the turn type.
- (f) Two advisories (48 bits) are required for target data to be complete (even for the ATARS purpose).
- (g) For an encounter situation involving many targets, it is required to uplink multiple Comm-A messages. This is an inherently inefficient use of the available channel capacity.
- (h) The aircraft flight number (or registration number) is not included. This may become a source of confusion for the pilot-ATC controller voice communication.

[†] These examples, by no means, restrict the CDTI applications to the so-called active mode. Even in a passive, monitoring mode, the map information would enhance the pilot situation awareness.

Assumptions for CDTI Applications The ATARS advisories cannot be applied directly for the desired CDTI applications. Because of the shortcomings of the ATARS advisories, a set of CDTI Data Format specifications are now defined which would satisfy the research flight experiment requirements. The proposed specifications depend on the following assumptions:

- (a) The majority of CDTI transactions are carried out using the so-called (Mode S) Extended Length Message (ELM; see Appendix A) capability so that available bit size per transaction would not be a system limitation.
- (b) A CDTI equipped aircraft is, by definition, very sophisticated. Thus, on-board computation capabilities in terms of memory, real time, and display flexibility would not be limiting factors.
- (c) The ground ATARS software can be modified.

The use of the ELM capability can be justified because the CDTI information requires more than 150 bits per scan; this is approximately the cut-off length for efficient use of the ELM*.

CDTI Data Format Specifications

Functions Several CDTI Data Formats need to be specified, depending on the CDTI avionics functions; these include the pilot-to-ground interface and uplink/downlink data transmission. DABS Comm-B protocol is used for the former purpose, and ELM protocol is used for the latter.

The Data Formats are defined for the following functions:

- (a) Pilot request for CDTI data uplink process initiation or termination;
- (b) Pilot request for option selection or modification;
- (c) Downlink of air-derived variables;

* Currently, one other application of the ELM is being developed. It is to transmit digitized weather radar maps. [20] Thus, it is safe to assume that the reliability of the DABS ELM link is established prior to the CDTI flight experiments.

- (d) General map and traffic information; and
- (e) Own and Traffic data.

Item (a) allows the pilot to initiate or terminate the CDTI flight, i.e., it is used to notify the ground system to start or to terminate the CDTI uplink/downlink process[†]. Item (b) is included to give the pilot a limited number of options in the ground traffic selection logic. For example, the pilot may wish to give a high priority to a particular aircraft (even though it may not be a threat) so that the ground system can uplink a full data set. Items (a) and (b) are handled by Comm-B protocol using the pilot request BDS (B-Definition Subfield).

Item (c) is used to downlink the air-derived data which are merged with the ground data. These data are to be stored for post-flight analysis purposes and also to improve the ground tracker accuracy if such a modification is possible. The information is downlinked using the ELM (Comm-D) protocol.

Item (d) is used to uplink general information (not directly dependent on the receiving aircraft), and it contains DABS Sensor Site Identification, its coordinates, and Traffic Identification data. This information is needed to resolve the various coordinate systems and also to create a traffic file in the airborne computer.

Item (e) is used to uplink the Own and Traffic data such as position and velocity variables. This format is mainly intended for describing dynamic variables. Items (d) and (e) must be uplinked together within one complete ELM (16 Comm-C segments).

Figure 2 presents the schematic diagrams of information flow for downlink procedures. For the pilot request function, the pilot's wish is entered into the airborne computer through an input device such as a keyboard. The information is checked, packed, and sent to the SM (Standard Message) Interface Unit at appropriate times. The Mode S transponder

[†] The CDTI uplink and downlink process may be initiated or terminated through automatic logic independent of pilot action.

Pilot Request/Modification

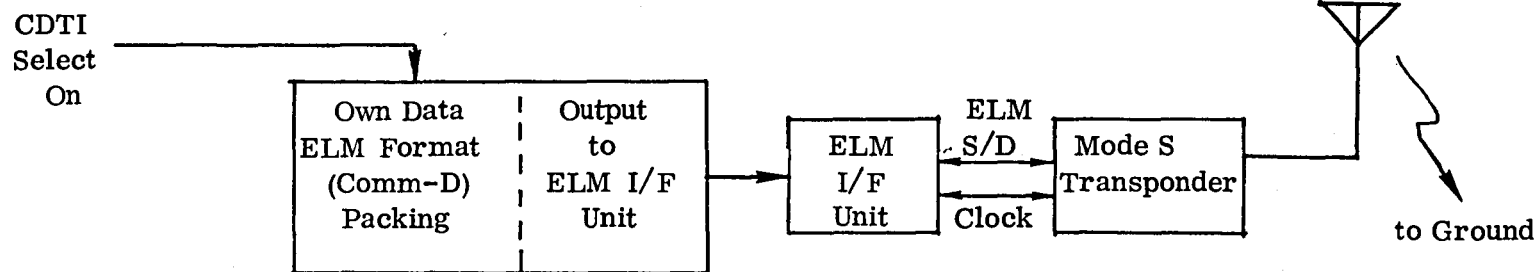
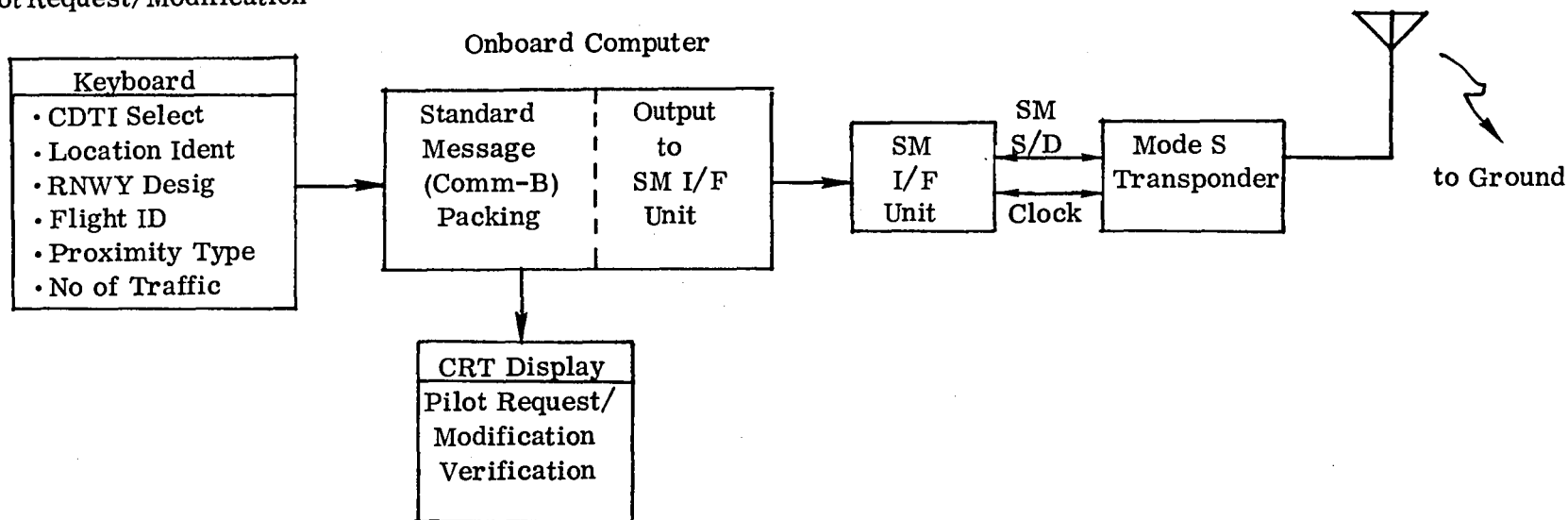


Figure 2. Components Interface for Downlink Operation

then packs the Comm-B message, based on the Interface Unit input and downlinks it to the ground system. For downlinking the airborne variables, the process is semi-automatic. Because the CDTI information was requested by the pilot, the airborne computer selects proper variables, packs them in proper format, and sends them to the ELM Interface Unit. The transponder obtains the ELM data, packs them within the Comm-D format, and downlinks the data to the ground. A more detailed description of this process appears later when the avionics are discussed.

The proposed Data Format specifications are given in the following sections. The downlink formats are discussed first, followed by the uplink formats. It must be emphasized that the associated definitions are based on the assumption that the ground system is modifiable to satisfy the data link requirements and specifications.

CDTI Downlink Data Formats

Format Definition for Pilot Request for CDTI Uplink Initiation or Termination Figure 3 shows the proposed subfield specification for the Comm-B message field (MB) for this function. The various subfields are described as follows:

- A BDS (B-Definition Subfield) designation of (01010000) indicates that this message contains a pilot request.
- A Type Code designation of (001000)* indicates that this message concerns the CDTI flight initiation or termination.
- Bit 15 indicates initiation (B = 1) or termination (B = 0).
- The Location Identifier consists of three five-bit character (truncated ASCII) code fields designating the airport of interest, e.g., SFO, LAX, DEN, etc. Table 4 gives the character definitions. For example,

D	E	N
0 0 1 0 0	0 0 1 0 1	0 1 1 1 0

S	F	O
1 0 0 1 1	0 0 1 1 0	0 1 1 1 1

* The Type Codes of binary 1 through 111 are reserved for pilot weather request applications. [20] As far as is known, these are the only applications planned.

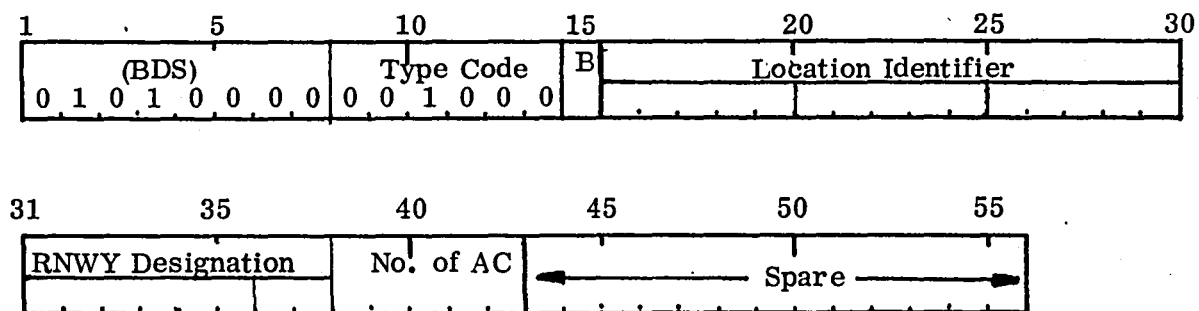


Figure 3. Pilot Request for CDTI Flight Initiation or Termination Format for Comm-B Message Field

TABLE 4

COMM-B DATA LINK CHARACTER SETS [17]

	5-Bit Letter Code					4-Bit Number Code	
	b5	0	0	1	1	b4	0 1
	b4	0	1	0	1		
b3 b2 b1							
0 0 0	Space	H	P	X		0	8
0 0 1	A	I	Q	Y		1	9
0 1 0	B	J	R	Z		2	L
0 1 1	C	K	S	/		3	R
1 0 0	D	L	T	*		4	Space
1 0 1	E	M	U	?		5	/
1 1 0	F	N	V	-		6	C
1 1 1	G	O	W	&		7	.

- The RNWY designation indicates the runway of interest. The field consists of a 6-bit numerical field (binary integer) and a 2-bit field for L, R or C designation. The numerical value is truncated and abbreviated to 10s of degrees. For example,

$\begin{array}{c} 35 \\ \hline 1 \ 0 \ 0 \ 0 \ 1 \ 1 \end{array} \begin{array}{c} \dagger \\ R \end{array}$

implies Runway "35R".

$\begin{array}{c} 17 \\ \hline 0 \ 1 \ 0 \ 0 \ 0 \ 1 \end{array} \begin{array}{c} L \end{array}$

implies Runway "17L".

- The number of the AC field specifies the maximum number of target aircraft to be uplinked. All 0's (0-0) indicates the default value which is determined by the ground system.

Pilot Request for Proximity Type Modification This request format is designed to give the pilot a capability of modifying a target aircraft priority (called proximity type, here). The data set corresponding to a higher priority aircraft (Proximity Type I aircraft) is uplinked every antenna scan, whereas that of a lower priority (Proximity Type II) aircraft is uplinked every other scan. The intention is that the pilot needs to have up-to-the-moment data on a few, more proximate traffic. On the other hand, he needs to be aware of the general presence of lower priority aircraft. The time multiplexing approach is taken to economize the uplink channel loading. Further display refinement (such as not showing full display content for Proximity Type II aircraft) can be made within the airborne computer. Aircraft can always be deleted, but they cannot be added if the data are not stored in the airborne computer.

Figure 4 shows the proposed Data Format. Explanations of the various subfields are given below:

- A BDS designation of (01010000) indicates that this is a pilot request message.
- A Type Code designation of (001001) indicates that the message is concerned with the pilot proximity type modification request.
- Flight ID designates a particular traffic identification. The field consists of 5-bit truncated ASCII codes and three 4-bit BCD (binary coded digit) codes. For example:

\dagger 00 = space (); 01 = L (left); 10 = R (right); 11 = C (center).

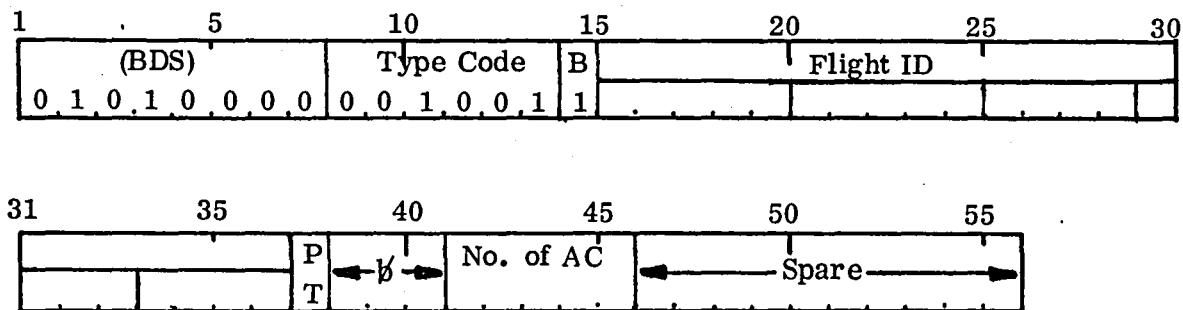
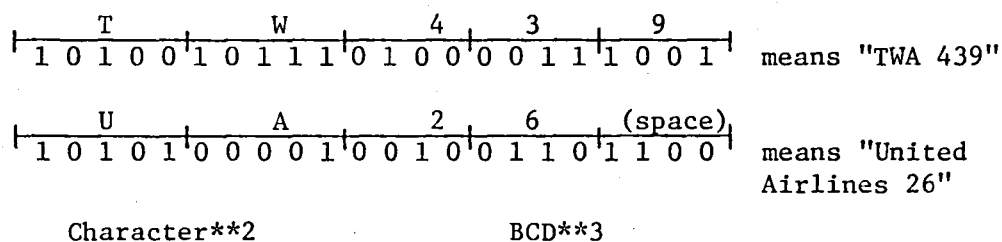


Figure 4. Pilot Request for CDTI Traffic Type Modification for Comm-B Message Field



- The PT designation indicates the traffic proximity type, and it is defined by
 - PT = 1 implies Proximity Type I
 - PT = 0 implies Proximity Type II
- The number of AC subfield is the same as the previous definition. Note that this format is used only when the pilot wishes to override the Proximity Type assigned by the ground system according to some selection and/or ranking criteria. How the PT assignment is done in the ground system is indicated in a later section.

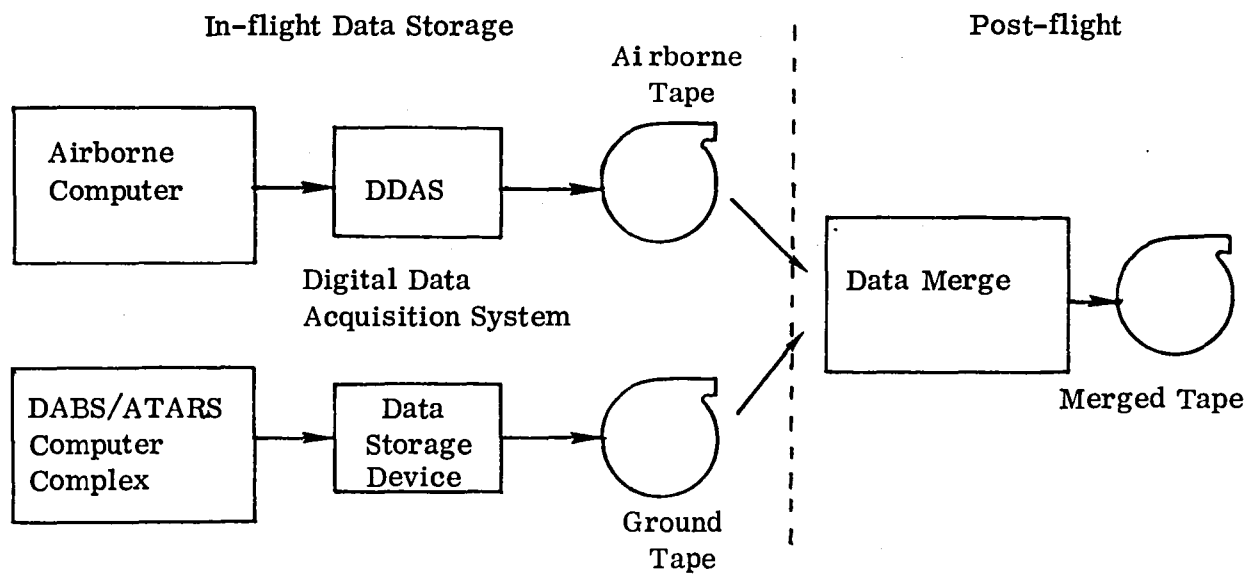
Data Downlinking for Post-flight Analysis For the purpose of post-flight analysis, it is necessary to store a sufficient amount of both ground generated and air-derived data. One way would be to store the DABS ground data (such as raw measurements or tracker outputs) on the ground and the air-derived data in a separate storage facility (either in the aircraft or telemetered to the ground). Then, these two data sets would be merged after the flight. Another way would be to downlink pertinent air-derived data to the ground system, merge these with the ground data, and then store the combined data on the ground.

Figure 5 depicts the two situations - the latter approach is attractive from two view points: (a) There is no need to merge two data sets which may be written in different data formats, with different time periods; and (b) this will off-load the airborne data storage device requirement.

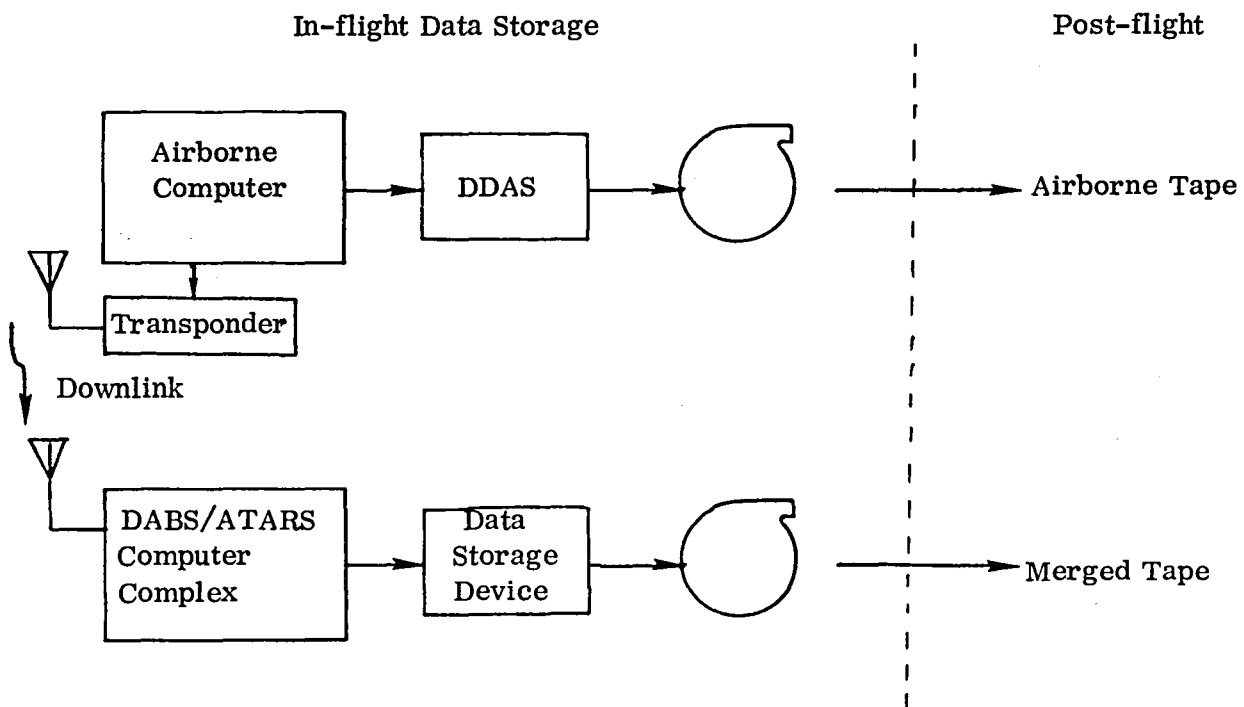
For data downlinking, the air-to-ground ELM, consisting of two Comm-D segments, is utilized. Tentatively, the following variables are selected for downlinking:

- x , y and z position variables;
- time;
- \dot{x} , \dot{y} and \dot{z} velocity variables;
- W_x and W_y wind components;
- V_T , V_I and ψ (true and indicated airspeed and heading angle);
- Number of proximate aircraft in the storage table; and
- PD (packed discrete indicating ownship status).

The position and velocity variables are essentially the outputs of the airborne navigation system which may be based on either VOR/DME, TACAN/DME, DME/DME, INS or MLS. Most probably, the navigation position and velocity outputs are referenced to a rectangular runway coordinate system. Thus, care must be exercised when comparing them with the DABS position report. The time variable may be obtained from a time code generator (TCG) or a clock internal to the computer. Wind vector components may be obtained from an INS or from the on-board navigation/wind estimation system. The true and indicated airspeeds are from an air data system. The heading angle could be from a magnetic compass or a directional gyro. The number of proximate aircraft is the number of target aircraft being stored in the airborne computer (possibly in a large data table form). This number must be the same as the number being uplinked from the ground system. This number will aid as a cross check for the overall status of the communication linkage. The packed discretized are various bits indicating the airborne system status such as autopilot mode, navigation mode, etc. This downlinked data list is not intended to be exhaustive, but these variables are all immediately available from the airborne computers. They are thought to be the minimum required for post-flight analysis purposes.



a. Separate Data Storage Configuration



b. Single Data Storage Configuration

Figure 5. Two Basic Post-flight Data Storage Configurations

Figures 6 and 7 show the downlink bit assignment together with the resolution requirements. Except for TCG, TAS, IAS, HDG, No. of AC, and PD, the variables are coded in the two's complement format. TCG, TAS, IAS, and HDG are coded in the equivalent binary integer (no sign bit) format. Note that because this downlink function utilizes Comm-D, the exact protocol and the interface mechanization must be worked out in detail.

CDTI Uplink Data Formats

Two kinds of data sets are uplinked for the CDTI applications. One contains the general map and traffic identification information which is uplinked once at the CDTI flight initiation time. The other contains the Own and Traffic data; thus, it must be uplinked every antenna scan. The ELM capability is used for uplinking.

General Map and Traffic Data Format The proposed Data Format is presented in Figs. 8 and 9. The ELM may contain up to sixteen Comm-C segments; however, one ELM is sufficient.

The first segment (Fig. 8) consists of map data such as the DABS sensor ID, site ID, and sensor priority status. The second and subsequent segments (Fig. 9) consist of traffic identification information.

The subfield descriptions are given below:

- A CDS designation of (01000100) indicates that this ELM contains general map and traffic information.
- An ME designation of (00) implies no ELM continuation, i.e., up to sixteen Comm-C's are sufficient*.
- The Sensor and Site ID identification numbers for the sensor and site; these are contained in the DABS sensor data set.
- The SPS is the Sensor Priority Status bit (not applicable for a single site system).
- ST signifies the sensor type (enroute = 0, terminal = 1).

* In retrospect, this subfield is not needed for CDTI application.

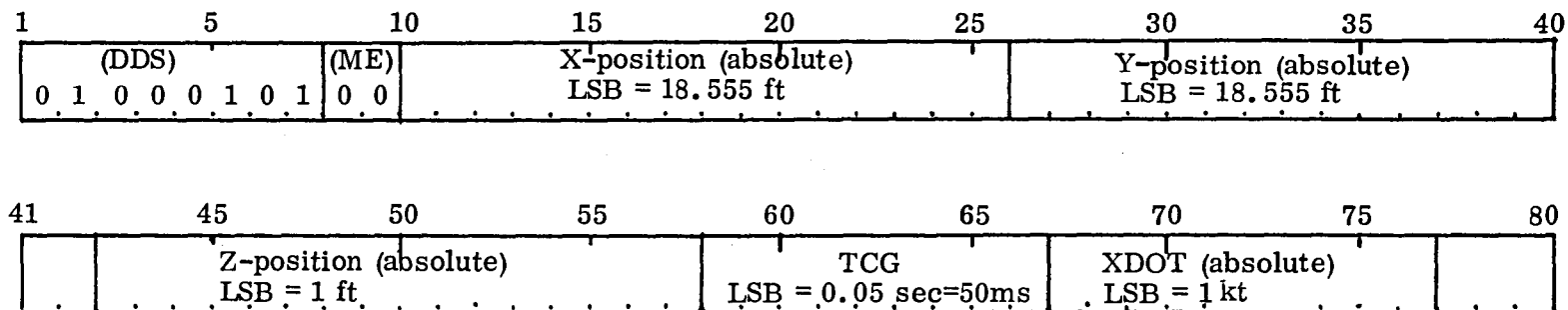


Figure 6. Downlink Information (1st Segment of the ELM)

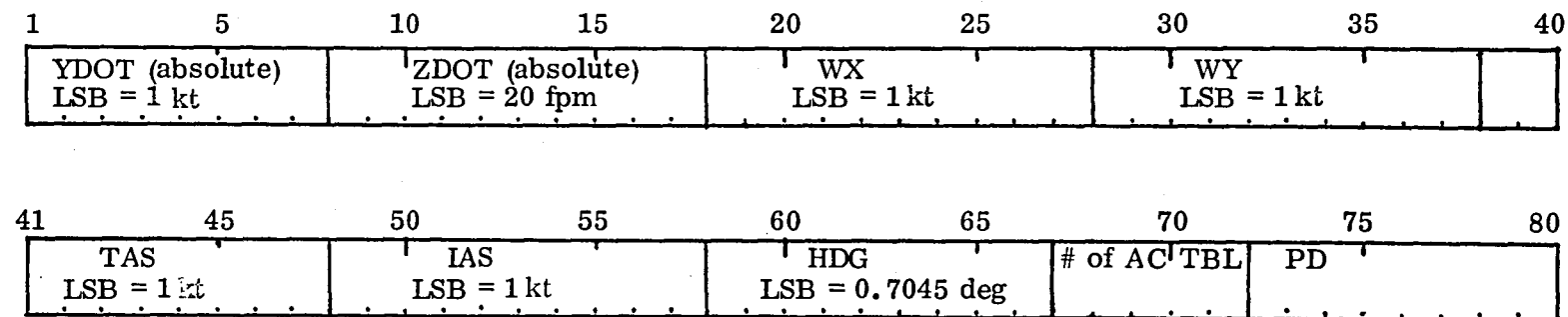


Figure 7. Downlink Information (2nd through Last Segment of the ELM)

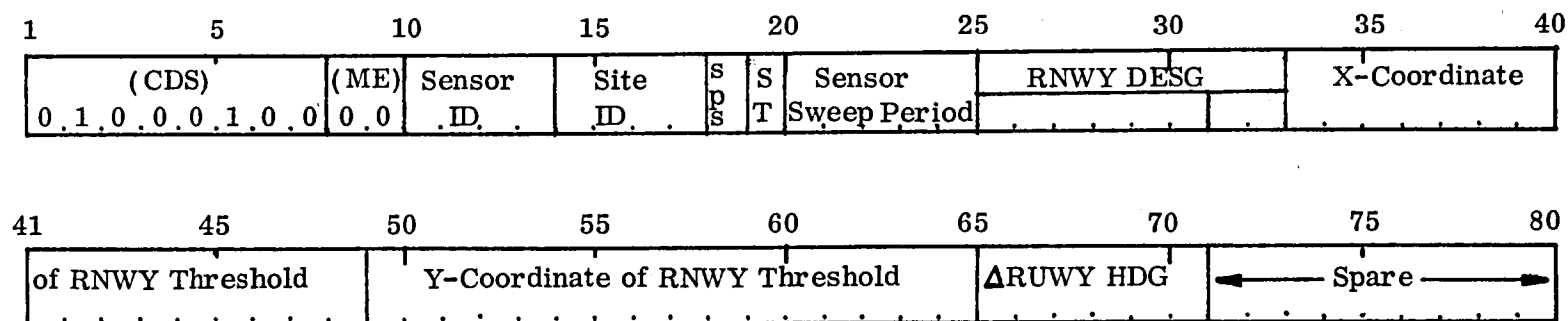


Figure 8. General Map of Traffic Identification Information (1st ELM Segment)

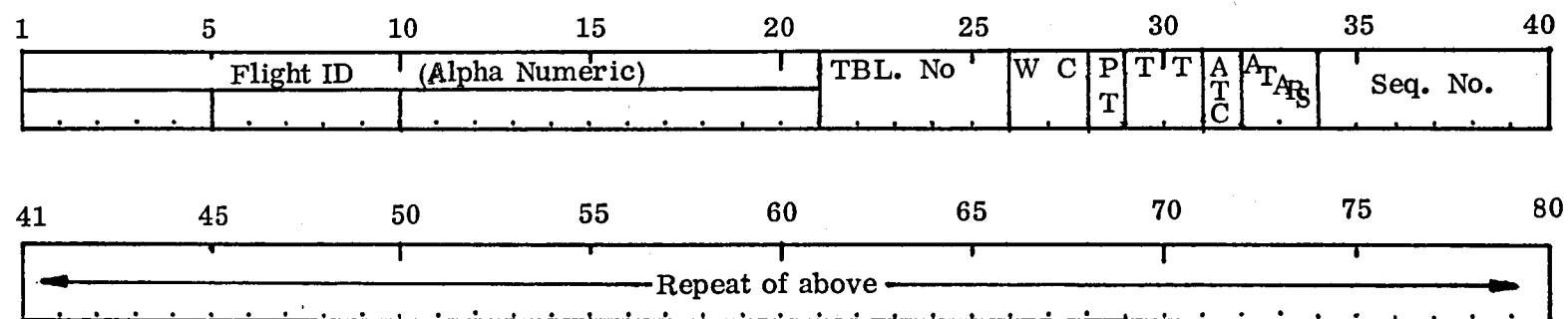


Figure 9. General Map and Traffic Identification Information (2nd through last ELM Segments)

- SSP (Sensor Sweep Period) is defined by the following formulae:
 if $ST = 0$, then $SSP = 10 + N * 0.1$ sec,
 if $ST = 1$, then $SSP = 4.5 + N * 0.05$ sec,
 where N is the content of this subfield.
- A RNWY designation is identical to the previous definition.
 See Fig. 3.
- The x and y coordinates define the runway threshold with respect to the DABS site. These are packed in two's complement with the $LSB = 10$ ft to cover the $(150 \text{ nmi}) \times (150 \text{ nmi})$ area. (These could be expressed in ΔLat and $\Delta Long$ also.)
- The $\Delta RNWY$ HDG subfield supplies the fine runway heading to supplement the truncated portion in the RNWY designation field.

To recover the runway heading, the following computation needs to be performed.

$$\psi_{RNWY} = (RNWY \text{ DESG No}) * 10 + (\Delta RNWY \text{ HDG}) * 0.3226 .$$

For example, if RNWY DESG and $\Delta RNWY$ are given as

$$RNWY \text{ DESG} = \overset{17}{\begin{array}{|c|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ \hline \end{array}},$$

$$\Delta RNWY \text{ HDG} = \overset{9}{\begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline \end{array}}$$

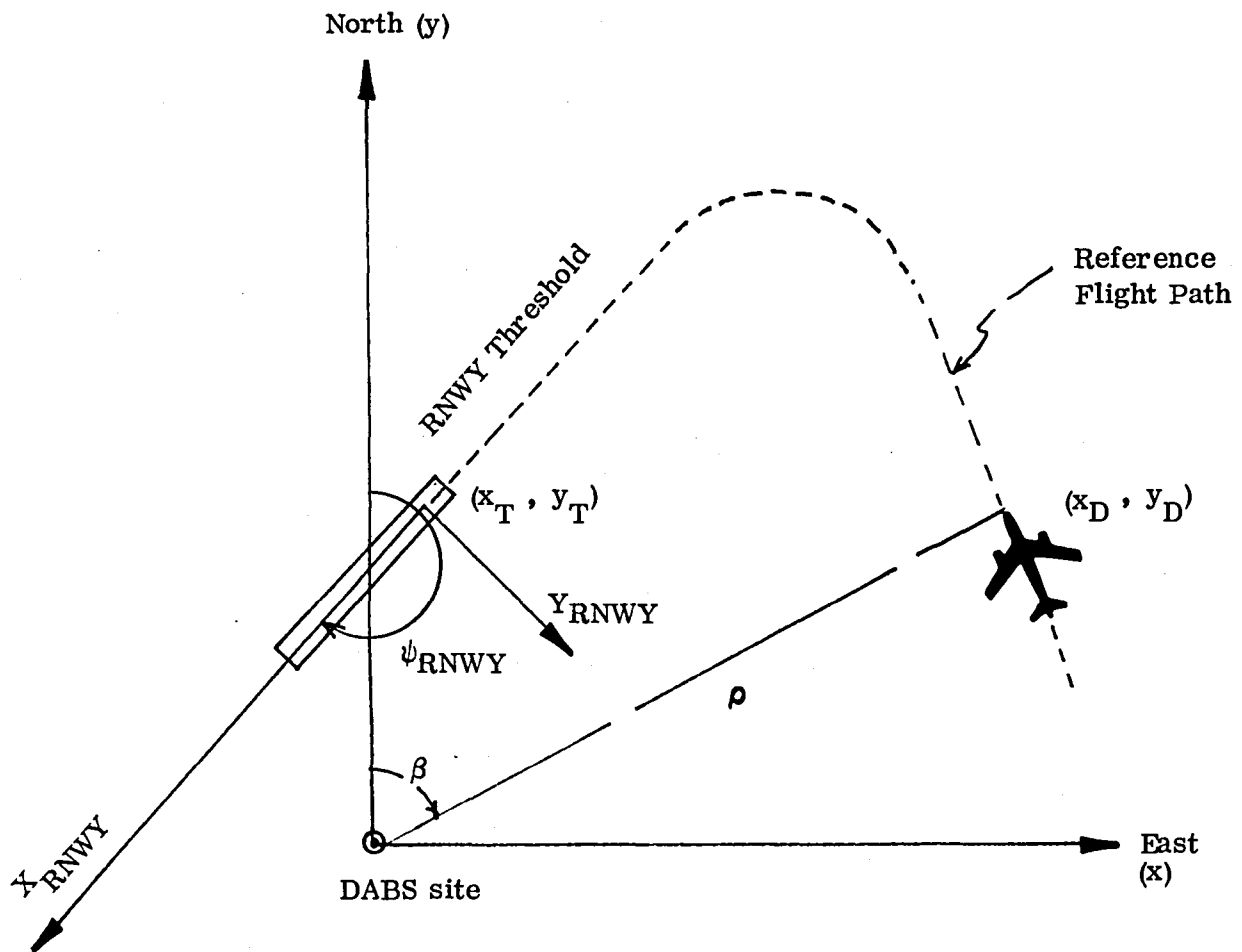
then the runway heading, ψ_{RNWY} is given by

$$\psi_{RNWY} = 17 \times 10 + 9 \times 0.3226 = 172.9 \text{ deg.}$$

The x and y runway threshold coordinates and the runway heading angle can be used to transform the DABS position variables into the extended runway coordinates. To do this, the following computation must be performed. (See Fig. 10.)

$$\begin{bmatrix} x_{RNWY} \\ y_{RNWY} \end{bmatrix} = \begin{bmatrix} \sin \psi_{RNWY} & \cos \psi_{RNWY} \\ -\cos \psi_{RNWY} & \sin \psi_{RNWY} \end{bmatrix} \begin{bmatrix} x_D - x_T \\ y_D - y_T \end{bmatrix} .$$

The variables are defined as follows:



- ρ = DABS range,
 β = DABS bearing,
 (x_D, y_D) = aircraft position wrt DABS site,
 $= (\rho \sin \beta, \rho \cos \beta),$
 (x_T, y_T) = RNWX threshold wrt DABS site,
 ψ_{RNWX} = RNWX heading, and
 X_{RNWX}, Y_{RNWX} = extended runway coordinate system.

Figure 10. DABS and Runway Coordinate Systems

(x_D, y_D) = aircraft position with respect to the DABS site;
 (x_T, y_T) = runway threshold with respect to the DABS site;
 (as given in the Format specification)
 (x_{RNWY}, y_{RNWY}) = aircraft position with respect to the runway
 coordinate system; and
 ψ_{RNWY} = runway heading as given above.

Figure 9 shows the subsequent (second through last) segments of the ELM. Forty bits are used to pack necessary information concerning target characteristics such as flight identification number and weight class. The information for two targets can be packed into one ELM segment. The subfield descriptions are given below.

- The flight ID No. designates the flight number. The field consists of two five-bit truncated ASCII codes followed by an 11-bit binary integer field. (Note that BCD was used for the Pilot Request Format, but an extra bit is needed.)
- The TBL No. designates a uniquely defined Table Number assigned to the flight number. This number is used to uplink traffic information rather than Flight ID No. to save space.
- WC designates the aircraft weight class: heavy (10), light (01), or small (00).
- PT designates the Proximity Type: Type I = 1, and Type II = 0.
- TT designates the transponder type: Mode A (01), Mode C (10) or Mode S (11).
- ATC designates IFR (1) or VFR (0).
- ATARS designates ATARS Service class 0 through 3. Class 3 means CDTI.
- The Seq. No. designates the landing sequence number. All 0's imply that the sequence is not assigned, and all 1's imply that this is a fly-over flight.

Some of the information, notably the Flight ID No. and Seq. No., are not available within the ATARS data structure. Thus, these must be accessed through the ATC computer facility.

Own and Traffic Data Format This Data Format is designed to uplink Own and Traffic data. It is assumed that one complete ELM (16 Com-C's) can be uplinked every antenna scan for the duration of a CDTI flight. The transponder capability is limited to at most one complete ELM.

Figure 11 shows the general ELM structure consisting of Header and Own information, General Traffic Identification information and Traffic data. The first segment contains the Header and Own Data. The next (up to 3) segments are identical to the Data Format given by Fig. 9. They are included so that up to six new CDTI aircraft can be added dynamically in each scan to reflect a changing traffic pattern. The following (up to 3) segments contain the Traffic data for Proximity Type I aircraft. Proximity Type II data are contained in the remaining segments. Therefore, up to 21 ($3 + 9 \times 2$) Target aircraft can be accommodated with this data structure.

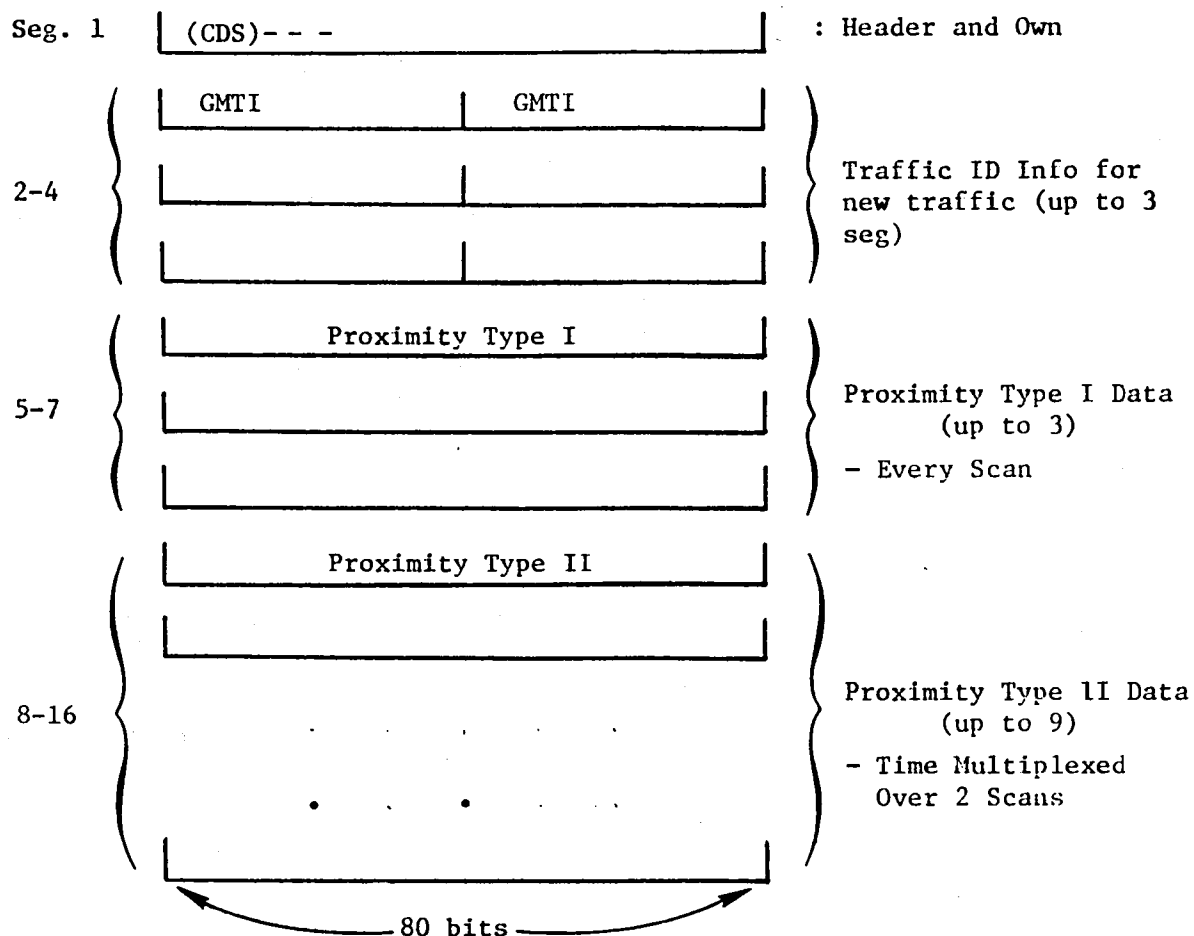


Figure 11. General Structure of Own and Traffic Data Format

Figure 12 shows the Header and Own Data Format contained in the first segment. The subfield descriptions are given below.

- The CDS designation of (01000101) indicates that this ELM contains Own and Traffic data.
- A ME designation of (00) means that this ELM is complete.
- GT specifies the number of segments used for General Traffic information. The maximum is three segments.
- The x and y components are absolute east-north position with respect to the DABS site and are given by the tracker output. At the indicated resolution and bit size, the maximum coverage is a 180 nmi square.
- The z position is the pressure altitude with respect to MSL.
- XDOT and YDOT are east-north velocity components as given by the ground tracker.
- Time is the epoch of the last DABS report. It is a truncated signal and varies in a saw-tooth fashion.
- D1, D2, D3 are discretes indicating the quality of the data.

Figure 13 shows the Data Format for the Traffic Data. Subfield descriptions are as follows.

- TBL No. is the Flight ID No. as defined in Fig. 9.
- PT signifies the Proximity Type (Type I = 1, and Type II = 0).
- Δx , Δy , and Δz are relative east, north and up position components of the Target with respect to Own. They are packed in two's complement form. The coverage volume is $(\pm 33 \text{ nmi}) * (\pm 33 \text{ nmi}) * (\pm 6400 \text{ ft})$.
- XDOT, YDOT and ZDOT are absolute east, north and up Target velocity components.
- ω is the estimated turn rate.
- TIME is defined similar to Own Time.
- D1, D2, and D3 are discretes (DLOUT, SMPR, and CENTR), stored in the State Vector. These discretes indicate the quality of the data.

The absolute position of the Target with respect to the DABS site can be recovered by

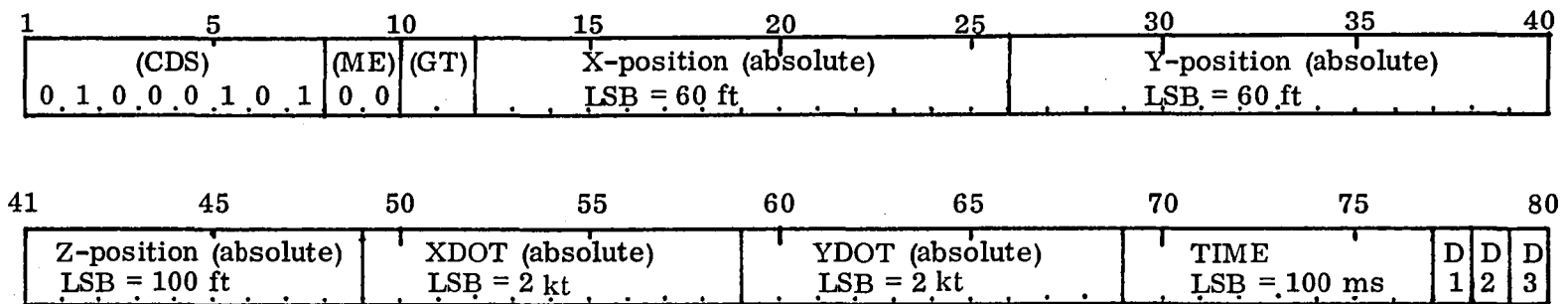


Figure 12. Header Information and Own Aircraft Data Format (1st ELM Segment)

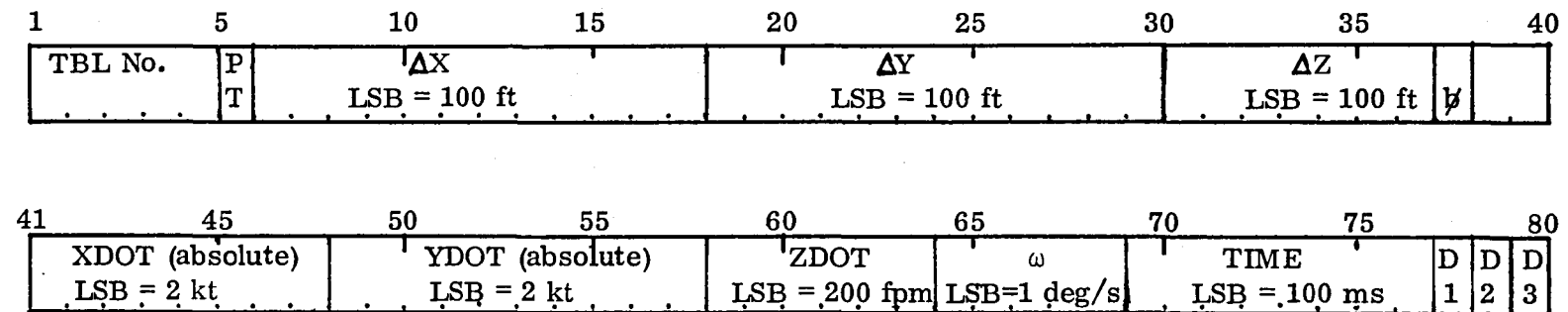


Figure 13. Traffic Data Format

$$\begin{bmatrix} x_T \\ y_T \\ z_T \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} .$$

Here, the first vector representing Own aircraft's position is contained in the first segment. The second vector representing the Target relative position is defined above.

The above relationship can be used to obtain more accurate relative position if own navigation variables are substantially more accurate. To see this, assume

$$\hat{x}_T = x_T + \delta x_T$$

$$\hat{x}_O = x_O + \delta x_O$$

$$\hat{x}_N = x_O + \delta x_N$$

Here, the variables \hat{x}_T , \hat{x}_O and \hat{x}_N are the Target and Own position estimates given by the ground tracker and Own position estimate given by the airborne navigation system. Also, x_T and x_O are the True Target and Own positions. The error terms δx_T , δx_O and δx_N are the corresponding estimation errors. Then, the relative positions are given by

$$\Delta x_G = \hat{x}_T - \hat{x}_O = \Delta x + \delta x_T - \delta x_O ,$$

$$\Delta x_N = \hat{x}_T - \hat{x}_N = \Delta x + \delta x_T - \delta x_N .$$

Here, Δx_G is the ground based tracker relative error, and Δx_N is the airborne navigation system's relative error. If the errors are assumed to be independent, then the rms ground tracker and navigation errors are given by

$$\sigma_{\Delta x_G}^2 = \sigma_{x_T}^2 + \sigma_{x_O}^2 ,$$

and

$$\sigma_{\Delta x_N}^2 = \sigma_{x_T}^2 + \sigma_{x_N}^2 .$$

Thus, $\sigma_{\Delta x_G} > \sigma_{\Delta x_N}$ if $\sigma_{x_O} > \sigma_{x_N}$. This is especially true during the Own maneuver period. A simulation study has shown that in such a situation the ground tracker error can be as high as 1500 ft (1σ) [21].

The resolution accuracy for the variables is made as fine as possible within the bit constraint. There are two reasons for this. One is that researchers can then choose to degrade the resolution for flight experiment purposes. The other is that some of the Target parameters which depend only on the current relative kinematic state variables can be computed in an airborne computer. These parameters include TAU (range/range rate), closest point of approach (CPA), and range and time to CPA. It is advantageous for threat alert applications to make these variables as accurate as possible. The parameters can be computed for Proximity Type I aircraft, and they can be incorporated within the CDTI display symbology.

This Page Intentionally Left Blank

III

CDTI AVIONICS

Background

The main purpose of this chapter is to develop general functional and conceptual design approaches for integrating and retrofitting the CDTI system with representative airborne systems given the uplink and downlink Data Format specifications of the previous section. Two airborne systems of interest are the FAA/NASA T-39 and NASA TCV B-737 systems. The former represents advanced general aviation and business aircraft. The latter represents advanced air-carrier aircraft.

The approaches developed here are meant to be general so that they may be applied to other similar systems. One specific example of such an application is developed and extended. The result which is applicable to the proposed TCV B-737 Phase 01 Upgrade System is discussed in Appendix B.

Existing CDTI-Related Systems Currently, there are two CDTI-related airborne systems which are directly applicable to general CDTI avionics design. One is called Airborne Intelligent Display System (AIDS) developed by the MIT Lincoln Laboratory for the FAA. The other is a system utilizing tape recorded traffic data developed by NASA Langley for CDTI/TCV flight tests.[3] Figure 14 depicts the AIDS computer block diagram. The TCV/CDTI system configuration is shown in Fig. 15, and the interface and timing sequence diagrams are given in Fig. 16.*

In Table 5, merits and demerits for each system are tabulated according to the CDTI research system requirement attributes. The main criticism of AIDS stems from the fact that it is a system design to support ATARS functions but not CDTI functions. The primary criticism of the TCV/CDTI system is that it was not designed based on a real world traffic sensor. Nevertheless, some elements and ideas from both of these designs can be applied to the general CDTI avionics design effort at the conceptual level.

* Clinedist, W. and D. Cordle, "Phase 1 CDTI Software Requirement," NASA Working Memo.

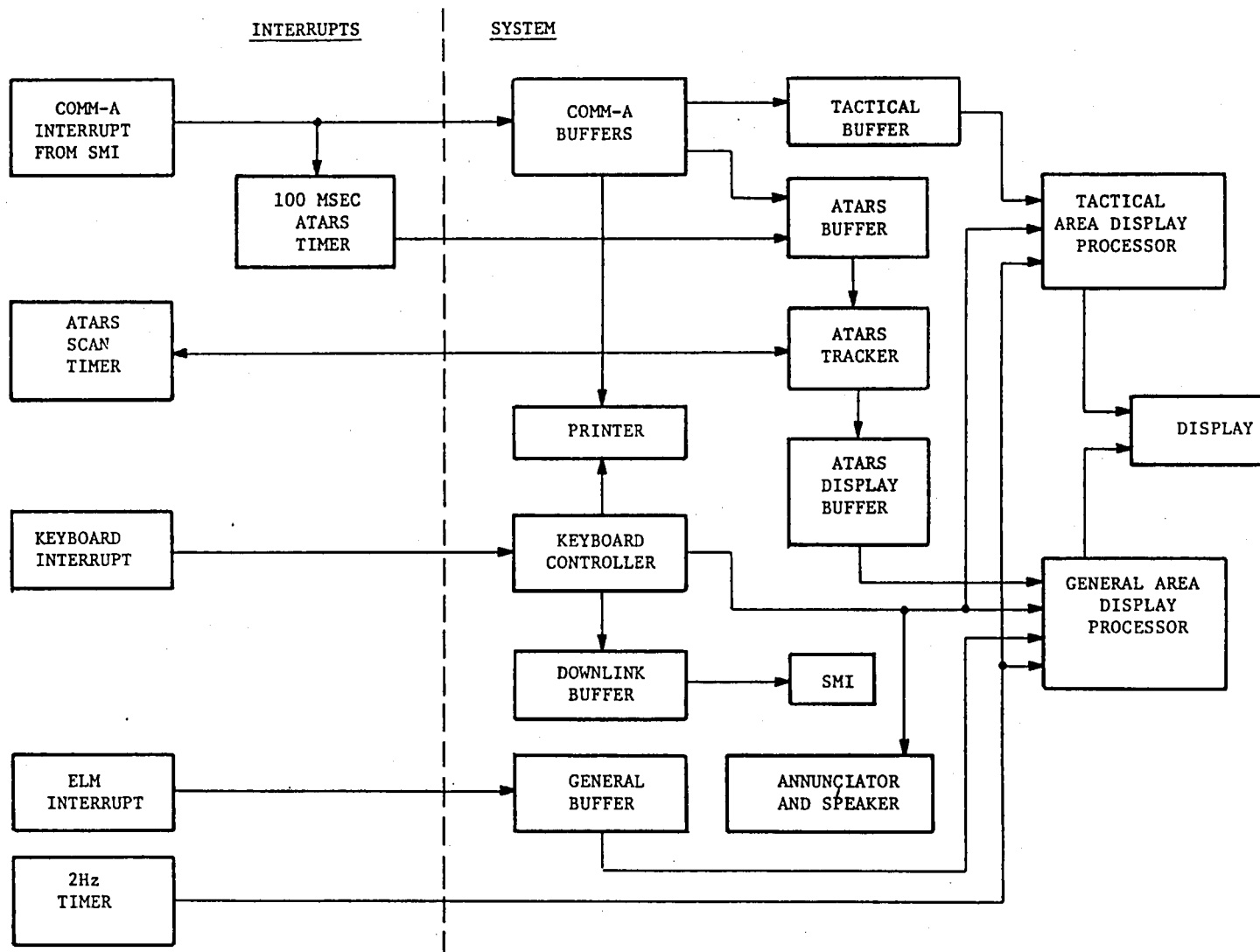


Figure 14. AIDS Data Link/ATARS Computer System [24]

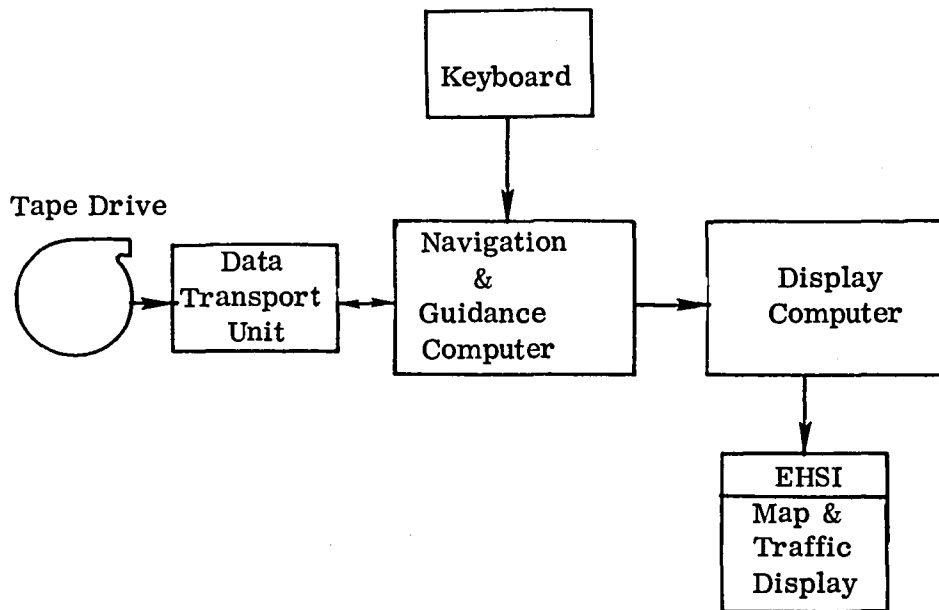


Figure 15. NASA Langley TCV/CDTI Flight Experiment System

Datum 4200
Cassette

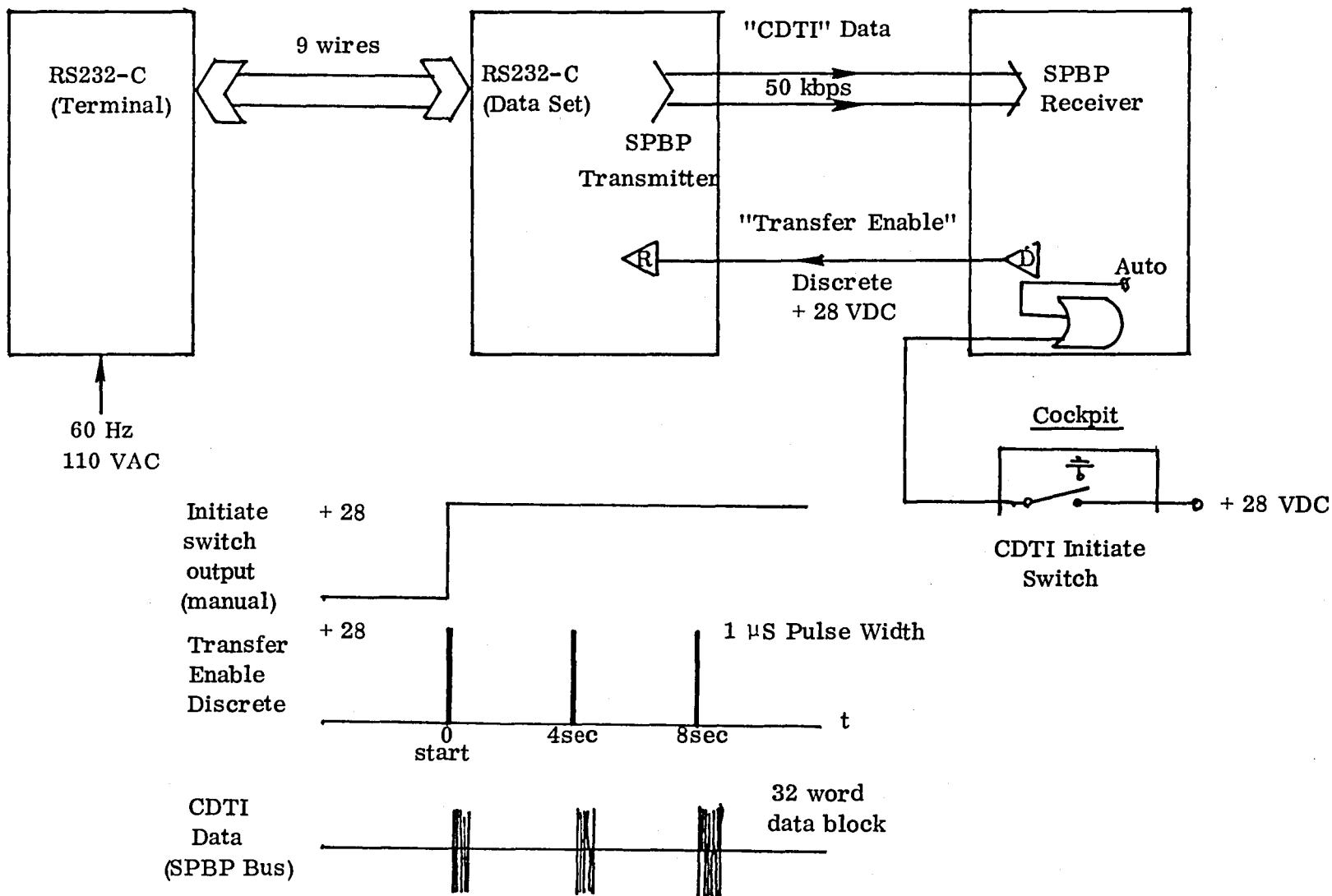


Figure 16. TCV/CDTI Interface Block Diagram and Timing Sequence

TABLE 5

COMPARISON OF AIDS AND TCV/CDTI

	AIDS	TCV/CDTI
Traffic Information Source	<ul style="list-style-type: none"> • DABS/ATARS • Real time • Real world errors 	<ul style="list-style-type: none"> • Magnetic tape cassette • Non-real time • Perfect information
Own Data	<ul style="list-style-type: none"> • Ground system (intermittently present) 	<ul style="list-style-type: none"> • Uses Own navigation
Coverage Area	<ul style="list-style-type: none"> • Small 	<ul style="list-style-type: none"> • Large
Target Data	<ul style="list-style-type: none"> • Variables not adequate • Resolution accuracy very coarse • Targets limited to eight 	<ul style="list-style-type: none"> • Variables adequate • Resolution marginally adequate • Targets limited to ten
Airborne Correction & Signal Processing	<ul style="list-style-type: none"> • Not possible • Data coasting capability 	<ul style="list-style-type: none"> • Possible • No data coasting employed
Display Capability	<ul style="list-style-type: none"> • Very coarse (low resolution) • Small display area • No flight ID No. • No map (reference flight path) display • Pilot display option limited • Pilot display only • No predictor; limited history dots • Proximity & threat situation display 	<ul style="list-style-type: none"> • High resolution • Large display area • Flight ID No. • Map display • Pilot display option limited • Dual (pilot/co-pilot) display • Prediction; history dots • No threat situation display
Pilot Interface	<ul style="list-style-type: none"> • Keyboard • Printer • Voice synthesizer 	<ul style="list-style-type: none"> • Keyboard (NCDU)
Data Storage	<ul style="list-style-type: none"> • Not capable for on-board variables 	<ul style="list-style-type: none"> • Capable for on-board variables

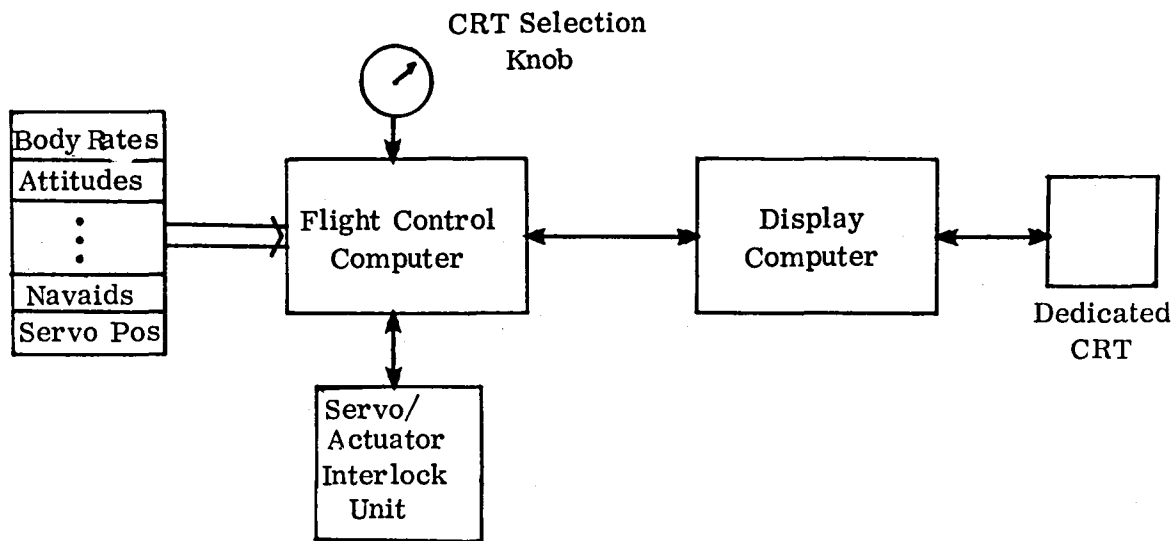
T-39 and TCV Avionics Configurations To provide appropriate initial conditions for the CDTI avionics development effort, two existing avionics configurations are considered as models to which appropriate CDTI components could be retrofit. This approach is advantageous in the sense that existing elements could be used in the resulting airborne system. The systems of interest are resident in both the FAA/NASA T-39 and NASA TCV B-737 aircraft which are slated for upcoming NASA CDTI flight experiments. The system configurations are shown in Fig. 17. (The exact component terminology may differ.)

Compared to the TCV system, the T-39 configuration has somewhat limited capability in the airborne computer (in terms of available memory, real time and input/output functions), the pilot-to-computer interface, and the flexibility of its display. This implies that this class of avionics needs major upgrading in computation and display capabilities. In such a situation, the CDTI functional requirements are met by retrofitting the CDTI avionics in a "parallel" fashion. That is, the additional hardware is attached to function in parallel with the existing capability. This is described next.

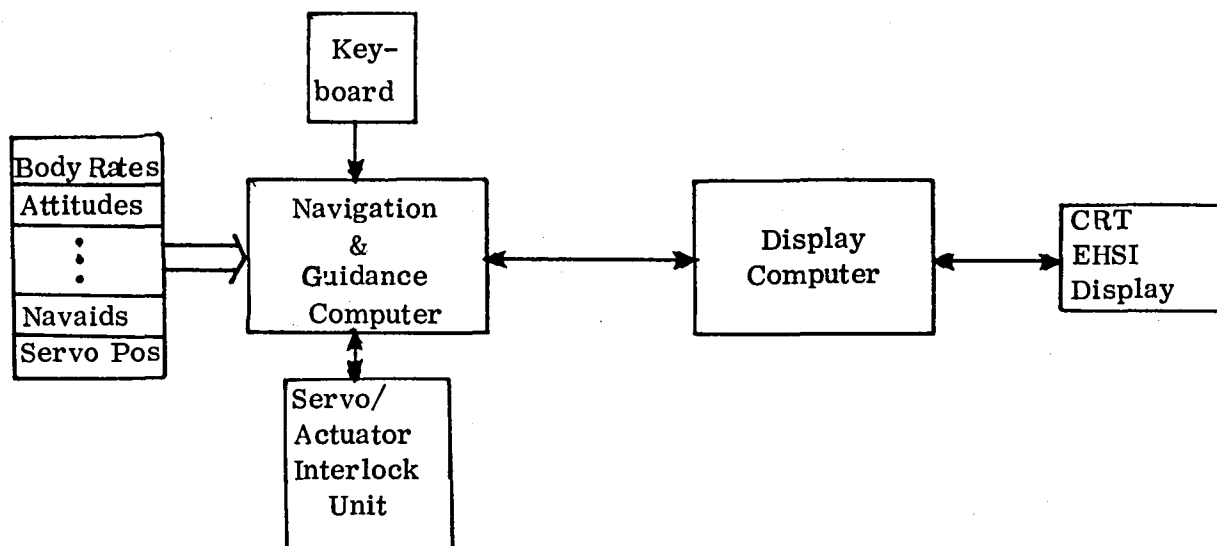
The TCV avionics consist of similar but more capable and flexible components. The computer and display units are configured to support a flexible map display. The pilot interface is achieved through a general purpose keyboard (NCDU) and corresponding software. In such a situation, the necessary hardware requirement is minimal, and the CDTI avionics can be interfaced in a "series" fashion. That is, the additional hardware is added to function at the front-end of existing TCV hardware. Of course, it is understood that the appropriate software modules are coded or modified within the framework of the existing software organization. This series organization is described shortly.

Proposed CDTI Avionics Configuration

Parallel Retrofit Configuration Figure 18 shows a CDTI avionics configuration for the parallel retrofit. The new equipment consists of a Mode S transponder and its Control Panel, a digital minicomputer,



(a) Existing T-39 Avionics Configuration



(b) Existing TCV B-737 Avionics Configuration

Figure 17. Two Candidate Systems for Retrofitting CDTI Avionics

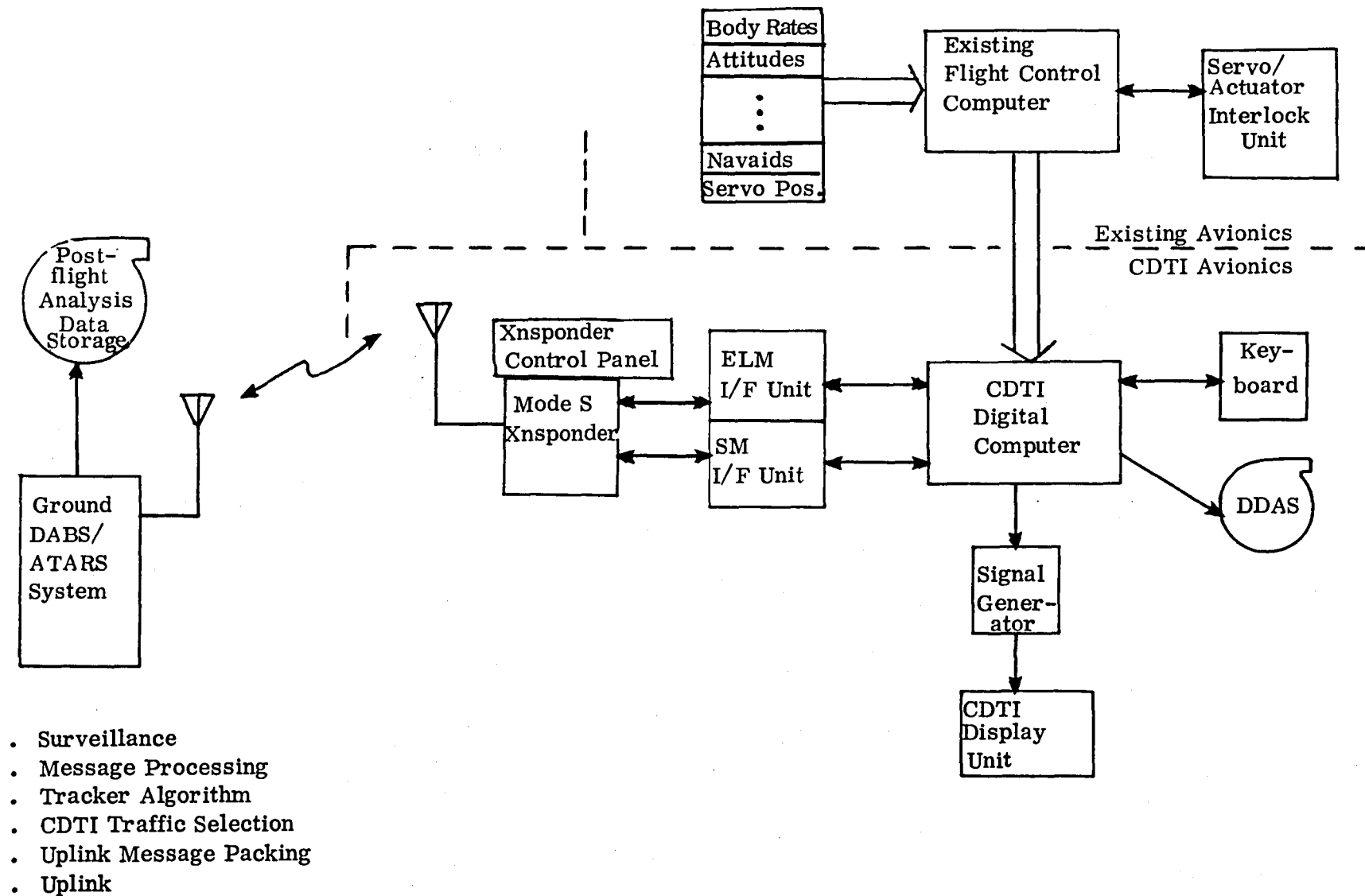


Figure 18. CDTI Configuration I - Existing Airborne Computer Not Modifiable

interface units (including keyboard and CDTI Mode Select Panel, CDTI Display Unit, and a Digital Data Acquisition System. The interface between the existing avionics and the CDTI avionics is minimal; the only requirement here is an input channel from the flight computer to the CDTI computer.

In the AIDS design, the transponder interface (SM Comm-A protocol only), the printer interface, the display symbol generation, and a host of other computations are performed within the AIDS computer (a Cromeco microprocessor with 64K memory). A similar design philosophy (of integrating whole functions into a microprocessor) may be pursued for the CDTI avionics. However, there are certain advantages in keeping the proposed component hardware architecture.

The only conceptual difference between AIDS and the proposed system is the existence of an input channel from the flight (or navigation and guidance) computer to the CDTI computer. The channel is needed to transmit navigation and guidance parameters or variables. These may include position and velocity estimates from the on-board navigation system to compensate for DABS/CDTI ground tracker lag, waypoints and runway data to construct a reference flight path, and roll attitude or yaw rate to generate the Own prediction vector.

Series Retrofit Configuration Figure 19 shows a CDTI avionics configuration with a series retrofit. In this case, the number of required hardware components is minimal. They include a Mode S transponder, the associated Control Panel, and an interface unit between the transponder and the airborne computer.

The proposed configuration is very similar to the TCV/CDTI system where the Tape Drive Unit and the Tape Data Transport Unit are replaced by the aforementioned components. Therefore, the software modules developed for the TCV/CDTI flight experiments may be used with slight modifications.

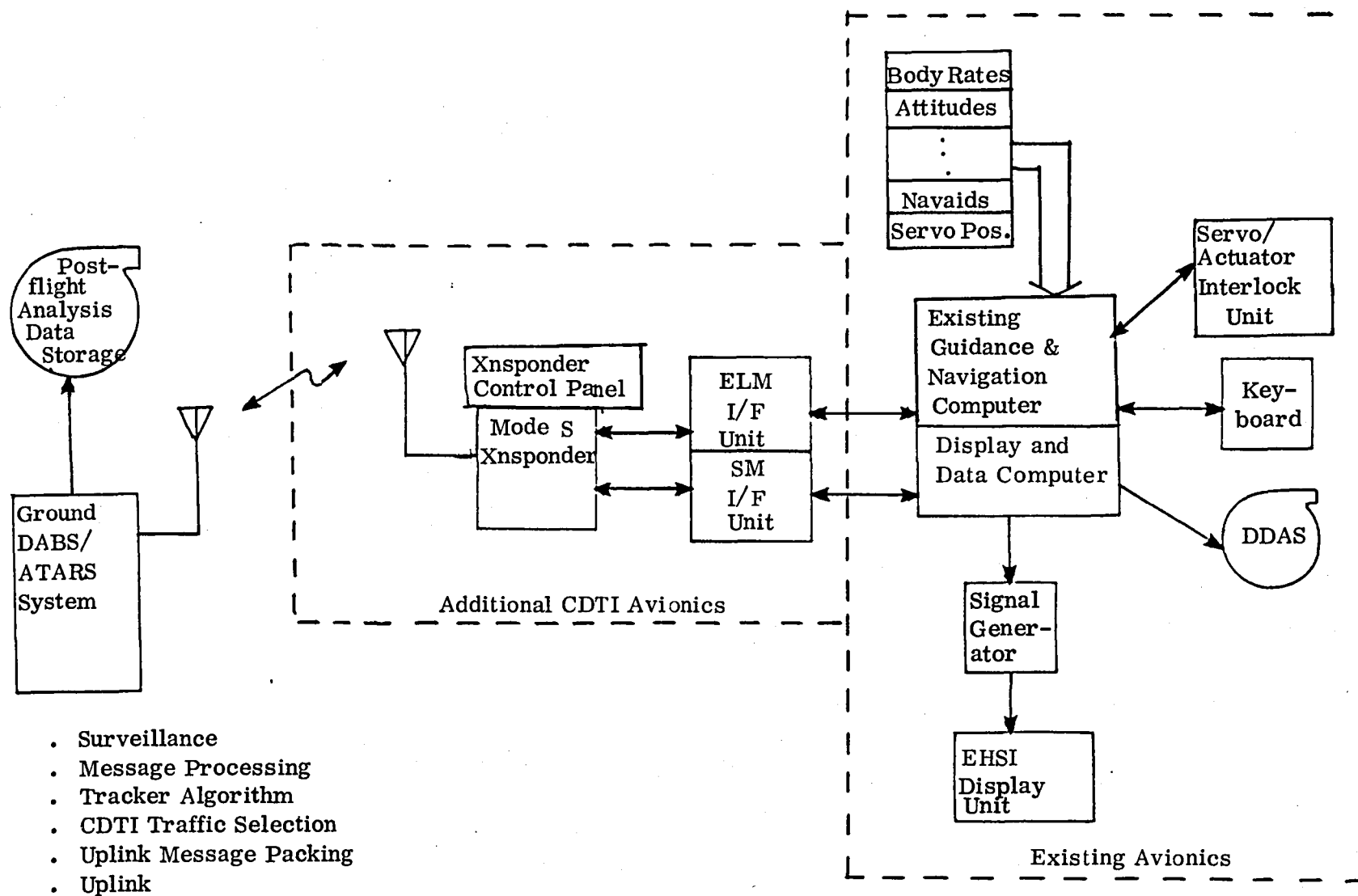


Figure 19. CDTI Configuration II - Existing Airborne Computer Modifiable

CDTI Avionics Components

Some of the design requirements for each component are now discussed. They include both hardware and software aspects. It is emphasized that at this stage, the conclusions reached in the discussion are tentative. This is because some of the required documents providing design detail are not available.

Mode S Transponder and Control Panel For the CDTI application, a Mode S transponder with the Extended Length Message capability is required. This implies that it is capable of using the DABS Comm-B protocol also.[†] Figure 20 shows the overall data interface diagram. Figure 21 presents a block diagram for the digital section of the transponder. Details of the transponder design can be found in the manufacturer's Maintenance Manual [25].

Figure 20 indicates the connection between the transponder and the Control Panel. The transponder Control Panel must be mounted within easy reach of the pilot or co-pilot.

Transponder Interface Unit (TIU) The TIU consists of two parts - the Standard Message (SM) Interface Unit and the Extended Length Message (ELM) Interface Unit. These can be housed in a single box. Because their workings are different, they are discussed separately.

SM Interface Unit (SMIU) The SMIU, which spans between the SMI port of the transponder and the airborne computer, is shown below.



[†] This is the position presented in the DABS National Standard. [9] A transponder with the ELM capability has the Comm-B capability, but not vice versa.

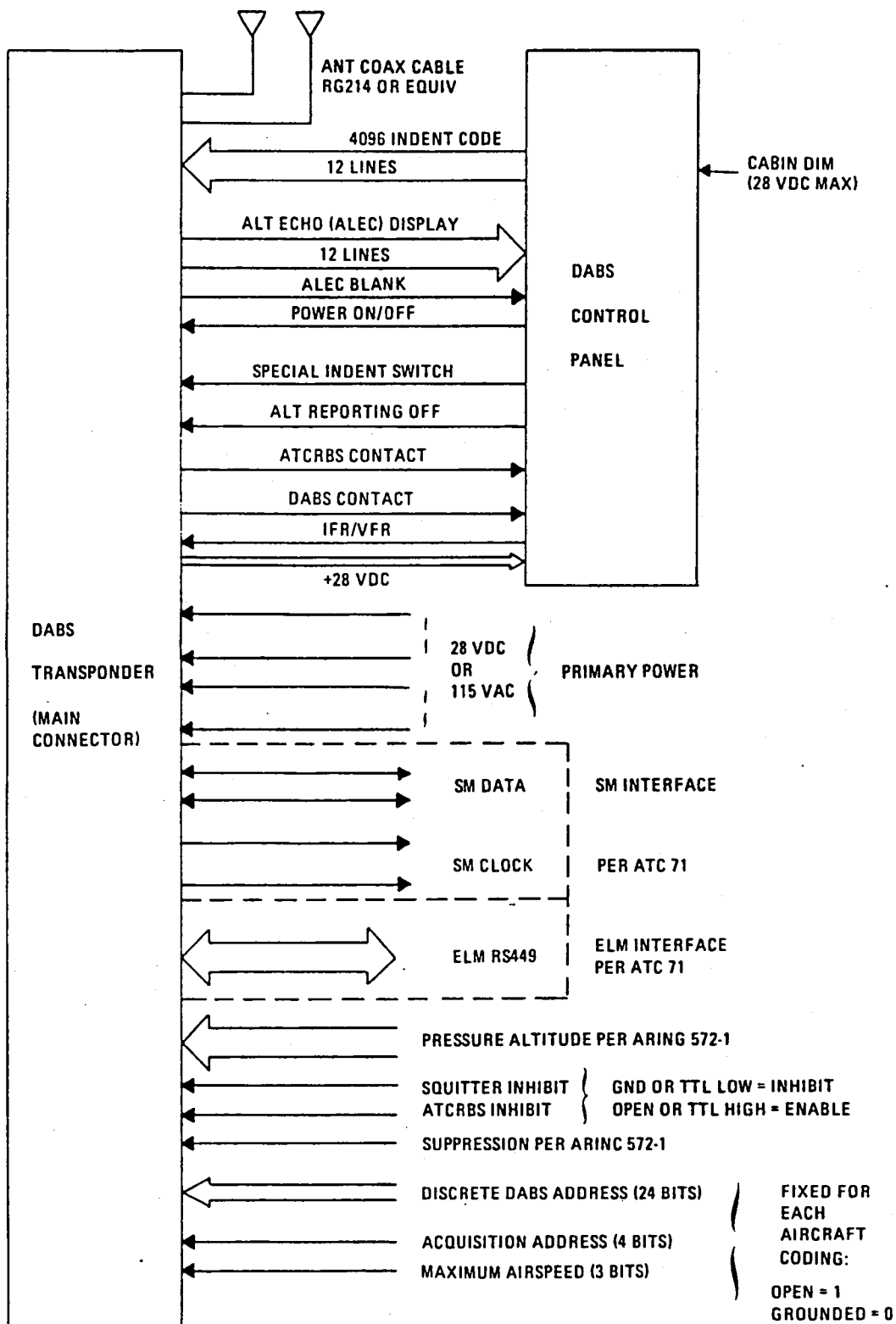


Figure 20. Data Interface for the Mode S Transponder [25]

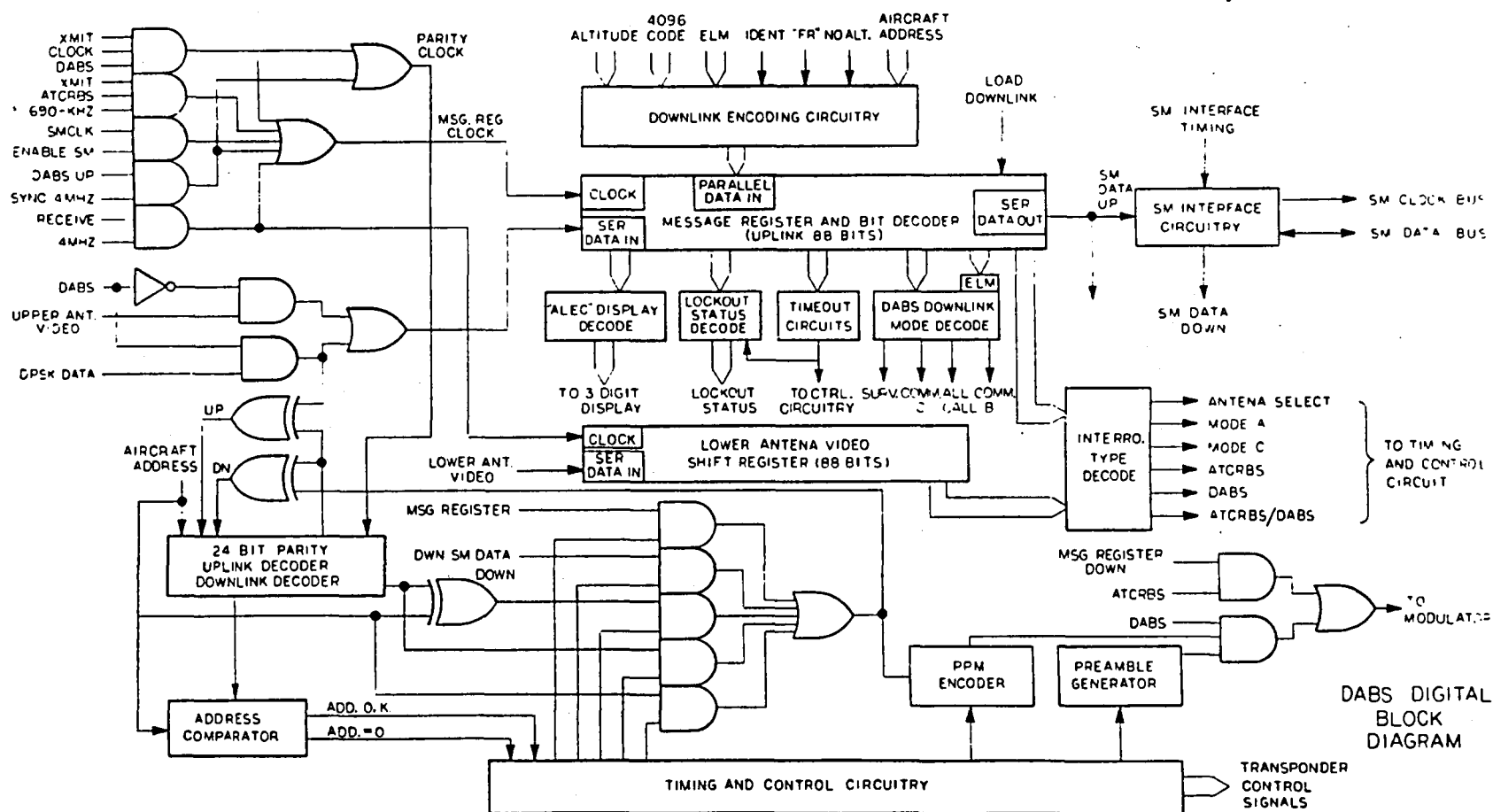


Figure 21. Mode S Transponder Digital Section Block Diagram [25]

The computer acts as the source of Comm-B messages (such as the pilot request for CDTI uplink initiation or proximity type modification (see Chapter II)) and as the destination for Comm-A messages.* The SMIU functions as a temporary data storage device for the inbound and outbound messages. It also takes care of bookkeeping and timekeeping functions.

Transactions between the transponder and SMIU must operate in real time because certain interrogations require almost immediate response with proper reply information. (The fixed transponder delay is 128 μ sec.) The interface timing diagram is given in Fig. 22, and the time line of events is presented in Table 6. The real time operation can be accommodated if messages are constructed by the computer and stored within the interface unit prior to the DABS interrogation/reply cycle. The data line between the transponder and the interface unit is a bi-directional serial digital channel with a capacity of 1 Mbps.

The transactions between the SMIU and the computer need not be in real time. The only requirement is that the interface unit does not introduce unnecessary time delay which may become significant at the DABS scan rate of once per 4.7 sec. The data line between the unit and the computer depends on the computer I/O device. Data could be passed in either a serial or a parallel fashion.

Figure 23 shows the macro flowchart of the interface logic. The SMIU contains four storage buffers, one of which is semi-permanent, read only. The {UPLINK} buffer stores all the Comm-A messages (exclusive of the address/parity bits) over one dwell period. A buffer size of 88 bits x 10 is sufficient. The {DWNLNK} buffer stores Comm-B messages generated by the processor based on the input from the computer. The buffer content should be properly formatted to allow a timely output to the SMI port. A buffer size of 88 bits x 5 is sufficient. The {MASK} buffer contains the masking data to examine the first 32 bits of the

* Comm-A messages are not utilized for supporting the CDTI research flight experiments. The discussions concerning the Comm-A interface are given here on a provisional basis.

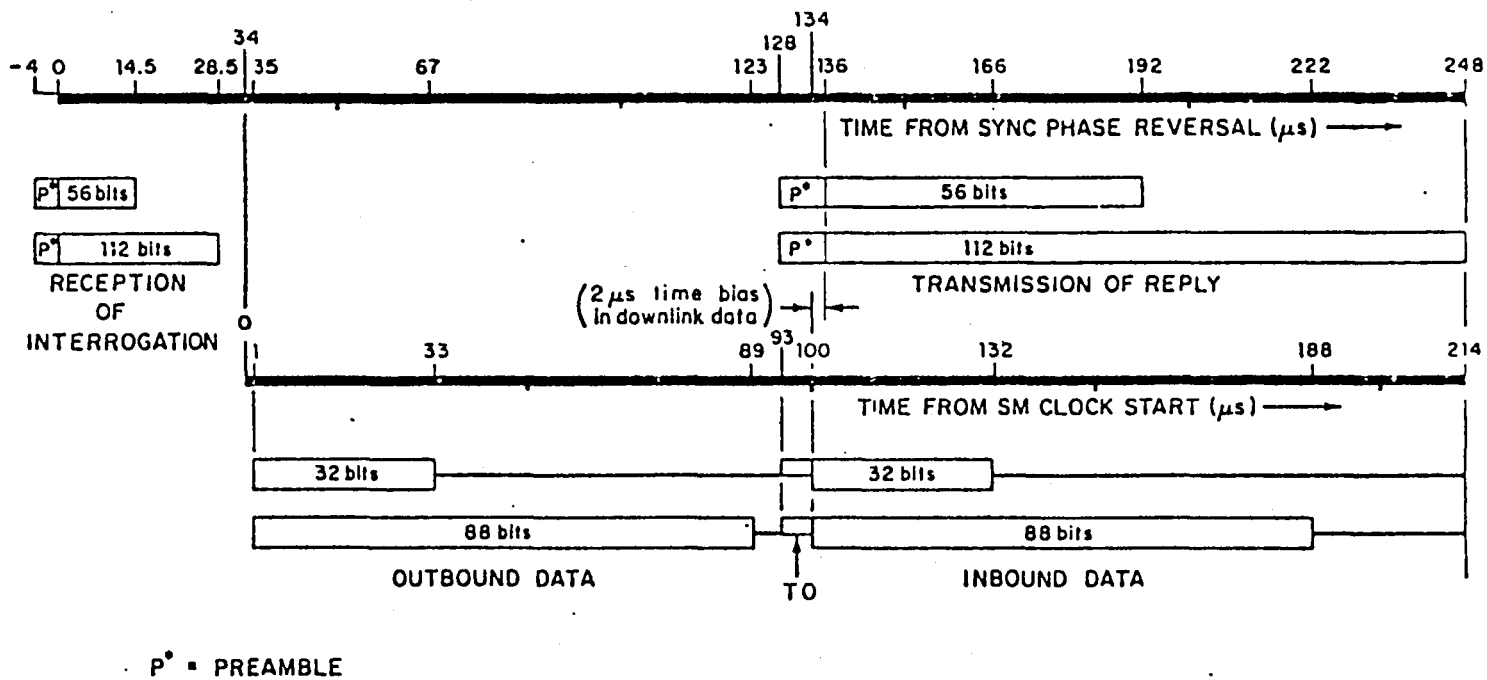


Figure 22. SM Interface Timing Diagram for DABS Transactions [11]

TABLE 6

RELATIONSHIP BETWEEN UPLINK/DOWNLINK TIMING AND
SM INTERFACE TIMING [11]

Event	Time From Sync Phase Reversal (μ sec)	Time From Clock Start (μ sec)
Start, uplink preamble	-4	-
Sync phase reversal or P_4 upstroke	0	-
End, 56-bit uplink	14.5	-
END, 112-bit uplink	28.5	-
Start, SM clock	34	0
Start, SM data or ALL-Call artificial data out	35	1
End, 32-bit data out	67	33
End, 88-bit data or All-Call artificial data out	123	89
Start, TO signal	127	93
Start, downlink preamble	128	94
Start, SM data in	134	100
Start, downlink data	136	102
End, 32-bit data in	166	132
End, 56-bit downlink	192	158
End, 88-bit data in	222	188
End, 112-bit downlink	248	214
Stop, SM clock	248	214

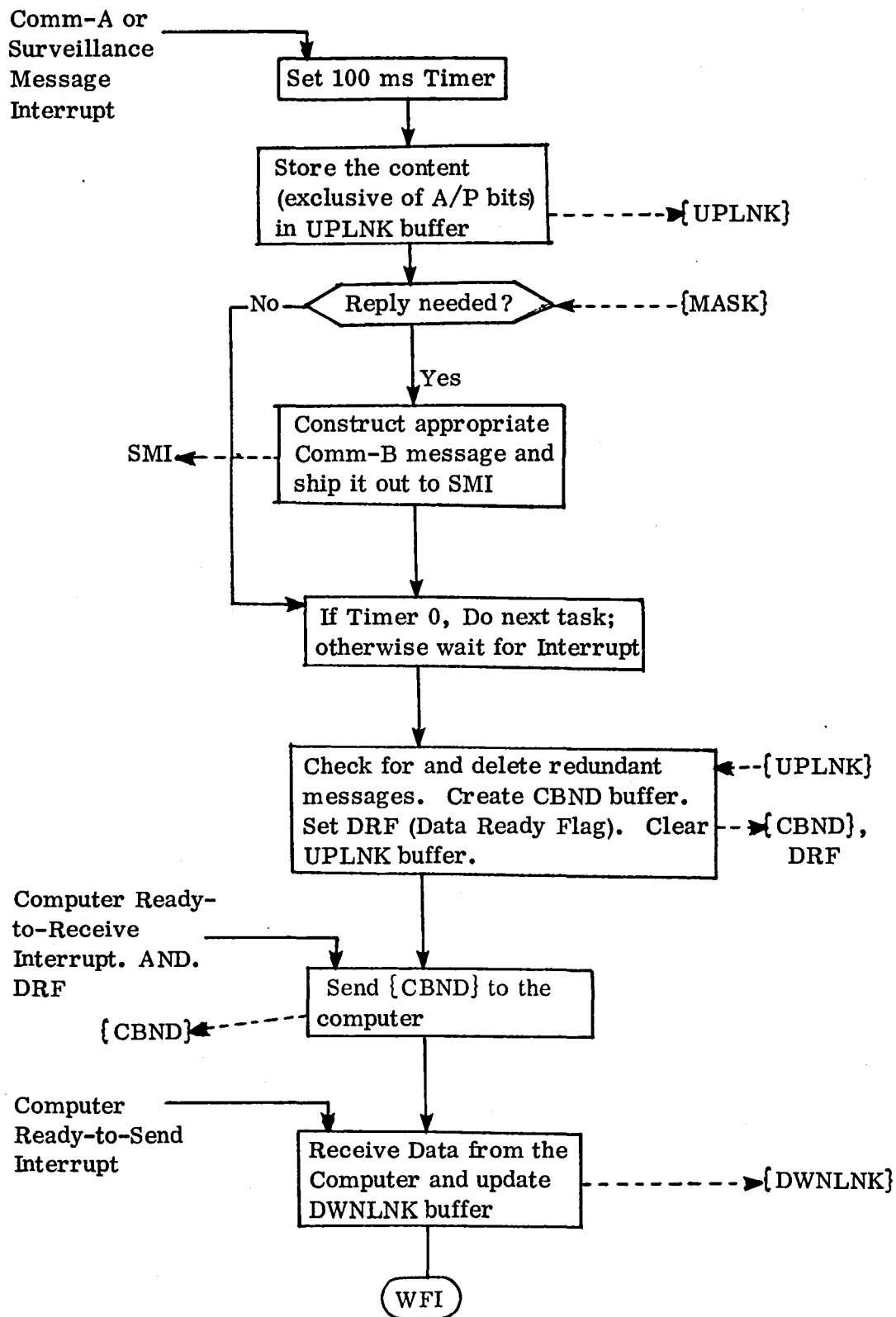


Figure 23. Flow Chart for a Conceptual Operation of the SM Interface Unit

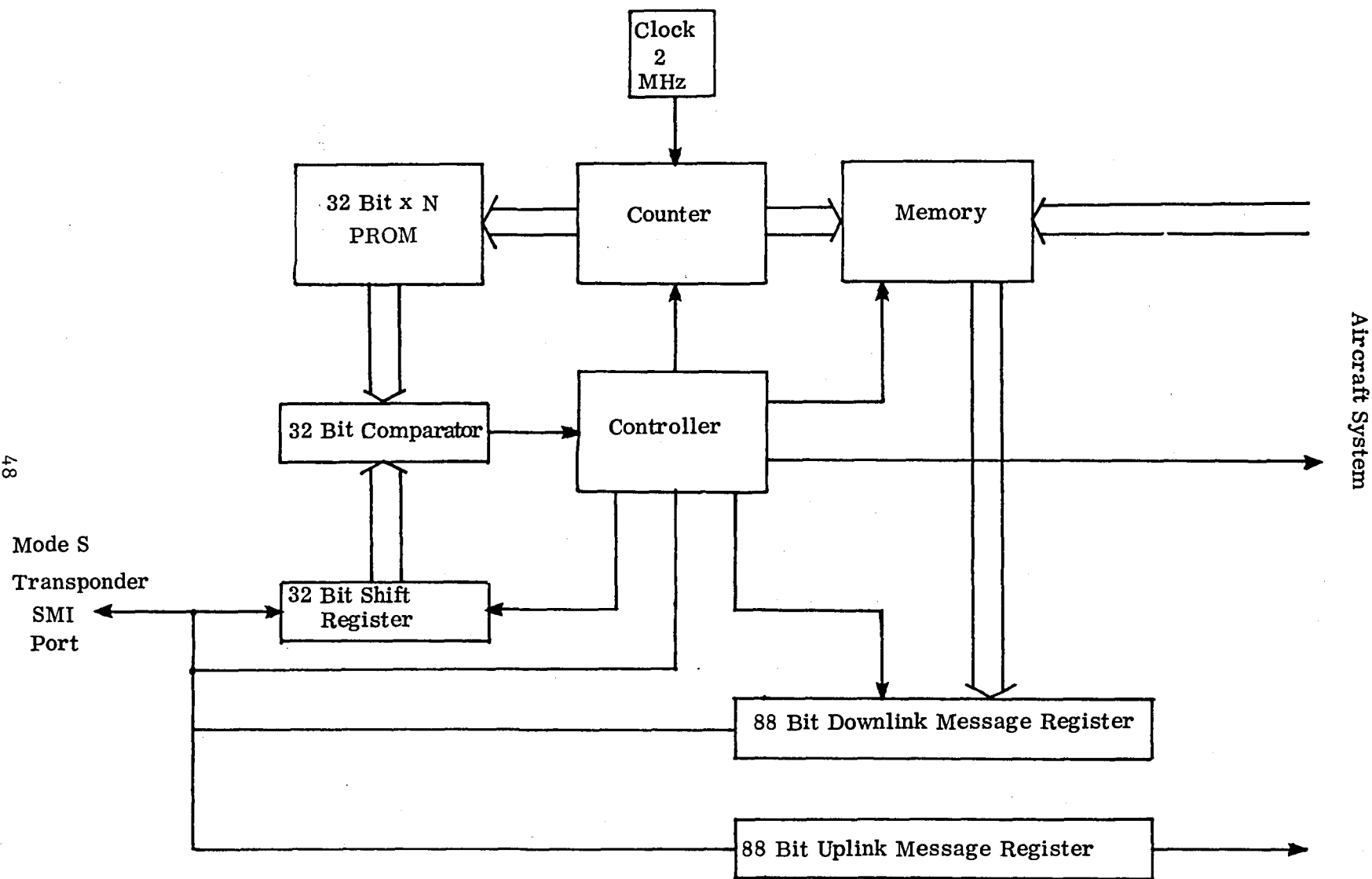


Figure 24. Mode S Transponder Standard Message Interface Concept

incoming messages. The {CBND} buffer stores the computer bound data, and it contains the contents of MA's (the last 56 bits) stored in the {UPLINK} buffer.

The Unit operation is controlled by three interrupts: the Comm-A message, the computer-ready-to-receive, and the ready-to-send. The first is controlled by the transponder, and the other two are controlled by the airborne computer.

A 100 msec timer is used to monitor the real time operation. Because the DABS beam dwell time is typically 25 - 45 msec, the elapsed time of 100 msec since the last Comm-A interrupt indicates that the next interrogation is not due for approximately 4.5 sec. Thus, after the timer has run out, the processor can proceed with the off-line tasks of cleaning the input {UPLNK} buffer and simultaneously stripping and storing 56 MA bits into the output {CBND} buffer. After {CBND} is created, the processor sets the Data Ready Flag (DRF). The computer monitors the DRF; and when the flag is set, the computer sends the computer ready interrupt (when it is ready). Upon receiving the interrupt, the processor sends the contents of the {CBND} buffer, and after completion, the DRF is reset. Following this task, the computer sends the downlink-message-ready interrupt to the processor, if such a message is waiting. Upon receiving the interrupt, the processor opens up the {DWNLNK} buffer and processes it further for suitable formats. The processor then waits for the next Comm-A interrupt.

Figure 24 illustrates a conceptual SMIU design.[†]

Extended-Length Message Interface Unit (ELMIU) Figure 25 depicts the ELM communication block diagram, and Fig. 26 illustrates the time line of transaction based on the manufacturer's description. Transactions between the transponder and aircraft I/O devices are handled by a bi-directional, relatively slow (12.5 Kbps) digital serial channel (RS422) and various control lines which are transponder controlled. Figure 27 shows the associated timing diagram, and Table 7 depicts the various

[†] This particular design is due to M.C. Lee of NASA Langley Research Center.

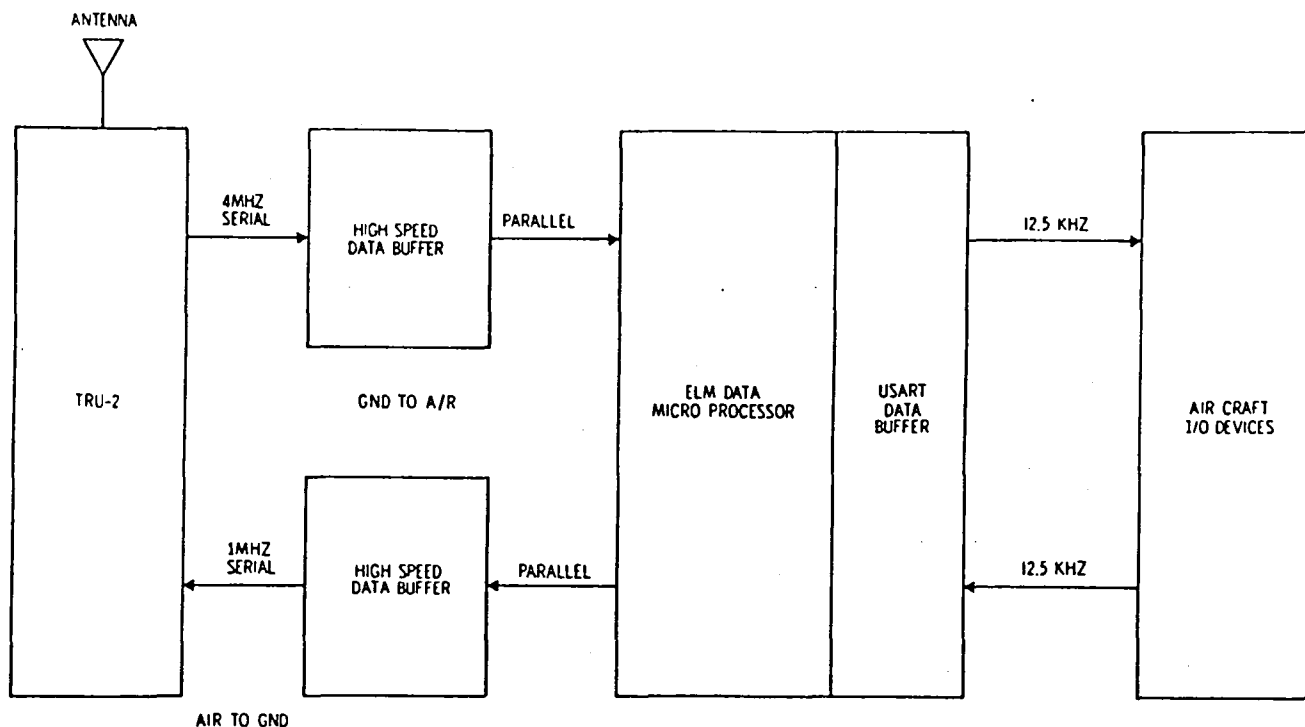


Figure 25. ELM Communications [25]

control line functions. The ELM data (contents of the MC only) are transmitted on a first-in-first-out (FIFO) basis. (The first bit of the first segment is transmitted first, and the last bit of the last segment is last in the burst.) This implies that it takes approximately 105 msec for a full 16 segment ELM (16 x 80 bits) transfer at the stated line speed.

It takes approximately 0.8 - 1.0 msec to uplink a 16 segment ELM when there is no constraint. The time estimate is based on 50 μ sec per segment times 16 segments plus time for two acknowledgement Comm-D's. Therefore, theoretically under the most favorable circumstance, the system should be able to uplink approximately 35 distinct ELM's during a 35 msec dwell period. However, due to the transponder design, the number of uplink ELM is limited to at most one complete ELM per scan. (It is not clear if the transponder can manage multiple ELM's if they are shorter than 16 segments.) The reasoning behind this is as follows: The uplink ELM segments are accumulated in an internal DMA buffer until all the segments are received and the two Comm-D acknowledgements are down-

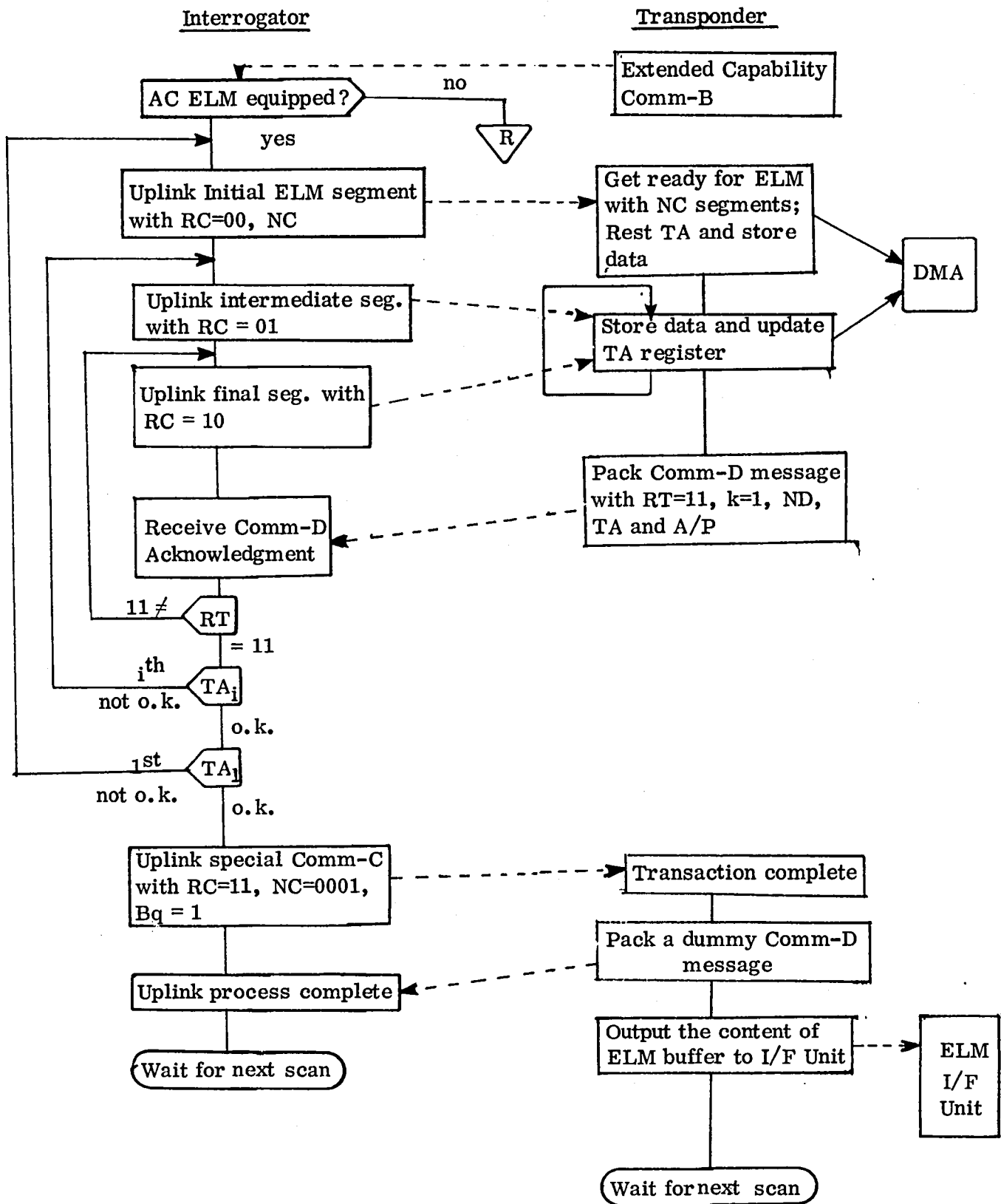


Figure 26. ELM Interrogator/Transponder Interface Time-line

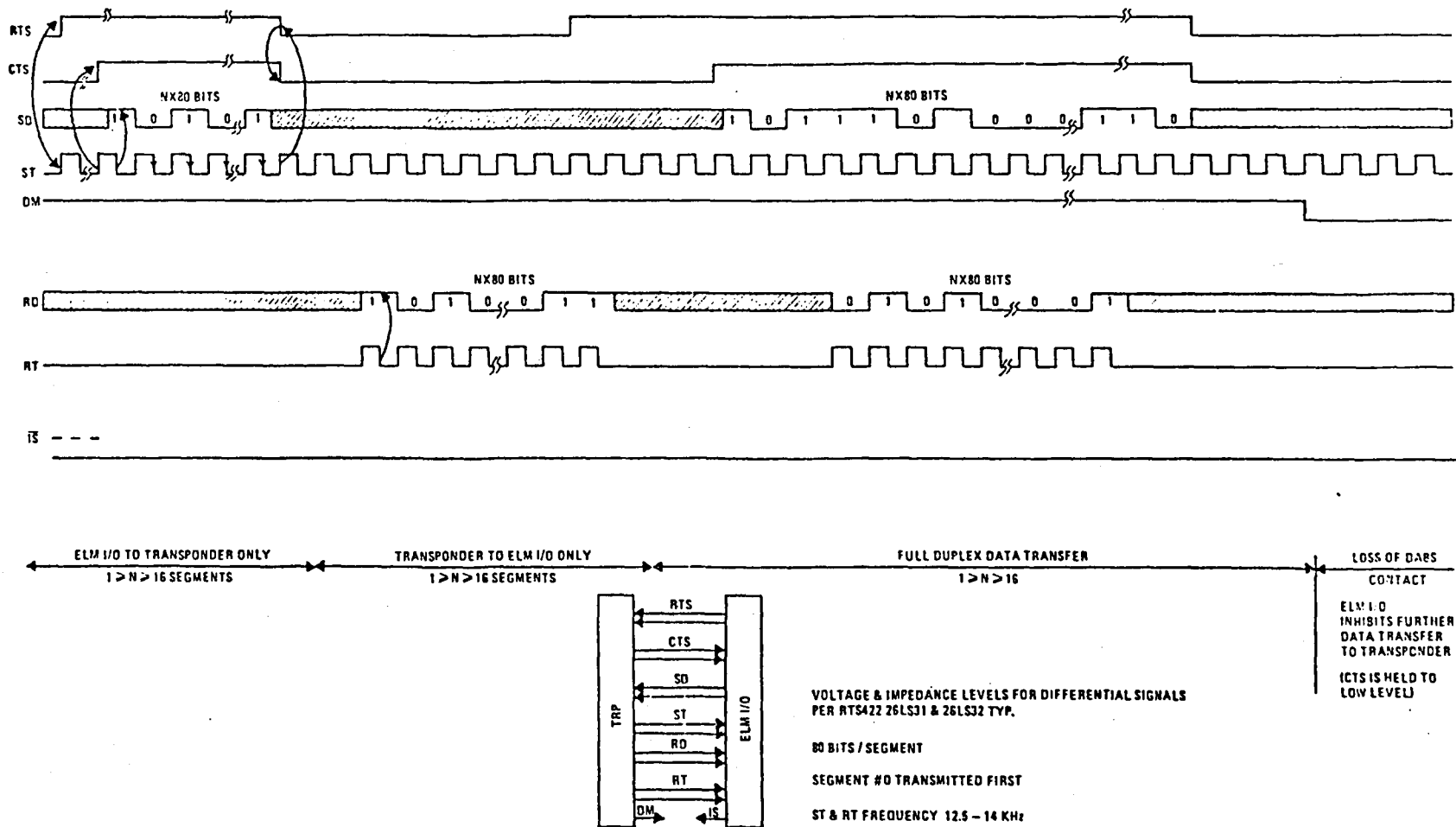
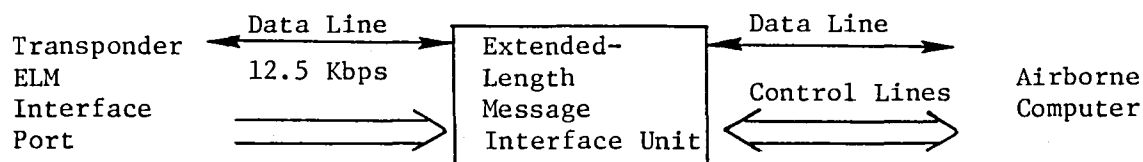


Figure 27. ELM Interface Timing Diagram [25]

linked. This process takes 0.8 - 1.0 msec. Further ELM transactions are inhibited until the content of the DMA buffer is transferred out through a slow 12.5 Kbps data line. This process takes 105 msec for a complete ELM. By the time the transponder is ready to receive another ELM (106 msec has elapsed since the uplink of the initial segment of the last ELM received), the transponder is outside the beam dwell which lasts 35 msec. Thus, no more ELM can be received until the next scan.

The implication of the above limitation for the present CDTI application is clear. If the ELM is used for uplinking the CDTI traffic information on a scan-to-scan basis, then no other service can use the ELM protocol. This is because the CDTI application preempts the uplink ELM protocol. The converse is also true, i.e., if another service uses the ELM protocol, then this precludes the channel for the CDTI application for the duration of the other service. A design change may be necessary to remove this limitation. It may require modifications in the ground system, the transponder, or both.[†]

The Interface Unit (schematically shown below) which spans between the transponder and the airborne computer can be less stringent in its design than the SMIU. This is because the real time requirement for the ELM transaction is minimal, and the ELM message is stored in a DMA buffer internal to the transponder. However, the real time requirement for the unit must be met to the extent that the I/O functions are controlled by the transponder through various control lines.



There are essentially two conceptual ways to build an ELMIU. One way is to transmit the bit train as it comes in. This may be done because the speed of the data line is slow compared to other available devices. (For example, a Split Phase Bipolar Transmitter/Receiver (SPBPT/R) which is used to bring in the data from the MLS/DME receiver

[†] This limitation of only one ELM per scan would not impact a CDTI research program, however.

TABLE 7

DEFINITION OF ELM INTERFACE SIGNALS [25]

Signal	Name	Origin	Function
RTS	Request To Send	I/O	An I/O device, with data to be transferred, causes this line to go high.
CTS	Clear To Send	Transponder	This line goes high when the transponder is ready to accept data from the I/O device.
SD	Send Data	I/O	Data segments transferred in a continuous stream from the I/O to the transponder.
ST	Send Timing	Transponder	Send data clock.
DM	DABS Mode	Transponder (internal)	This line is high, enabling the ELM interface, while the system is operating in the DABS mode. A low on this line disables the interface. If, during transfer of data, DM goes low, processing of the data segment in progress is completed.
RD	Receive Data	Transponder	Data segments transferred in a continuous stream from the transponder to an I/O device.
RT	Receive Timing	Transponder	Receive data clock.
$\overline{\text{IS}}$	$\overline{\text{In Service}}$	I/O	This line is always low when an I/O device is connected in the system

(SPBPT/R) which is used to bring in the data from the MLS/DME receiver in the TCV configuration is capable of 50 Kbps speed.) This is predicated by the following:

- (a) the airborne computer can handle the hand-shake operation with respect to the DABS cycle of approximately 4.7 sec. and the computer integration cycle of typically 50 msec, and
- (b) the incoming bit train can be partitioned properly and stored in a computer buffer.

This approach may pose a problem for ELM downlink information because the output buffer of the computer must be ready with the latest data whenever the transponder requests transmittal.

Another approach is for the interface unit to act as an intermediate storage device based on a microprocessor. This approach may be more advantageous in the sense that the airborne computer can control the I/O processes relatively free of the transponder actions. Also, it would be the only way possible if the computer I/O were done in parallel rather than a serial fashion (parallel transfer is generally faster). Figure 28 shows a conceptual macro flowchart indicating the interface processor logic for this approach. It is noted that the interface between the unit and the computer is highly dependent on how and when the computer I/O processor functions.

The logic works basically on four prioritized interrupts. The highest priority interrupt is the RD (receive data) interrupt from the transponder. When this happens, a 150 msec timer is set (150 msec elapsed time is sufficient compared to 105 msec for the data transfer period of 16×80 bits). The incoming bit train is stored in a buffer {INBUF} with a suitable word size. When the timer runs out, the ELMIU waits for the next priority CTS (clear to send) transponder interrupt. When this happens, the downlink ELM message being stored in {OUTBUF} is sent out FIFO as a burst. A 150 msec timer is also used for ascertaining the end of this operation.

The third priority interrupt is from the airborne computer, and it indicates that it is ready to receive the uplink ELM. When this happens,

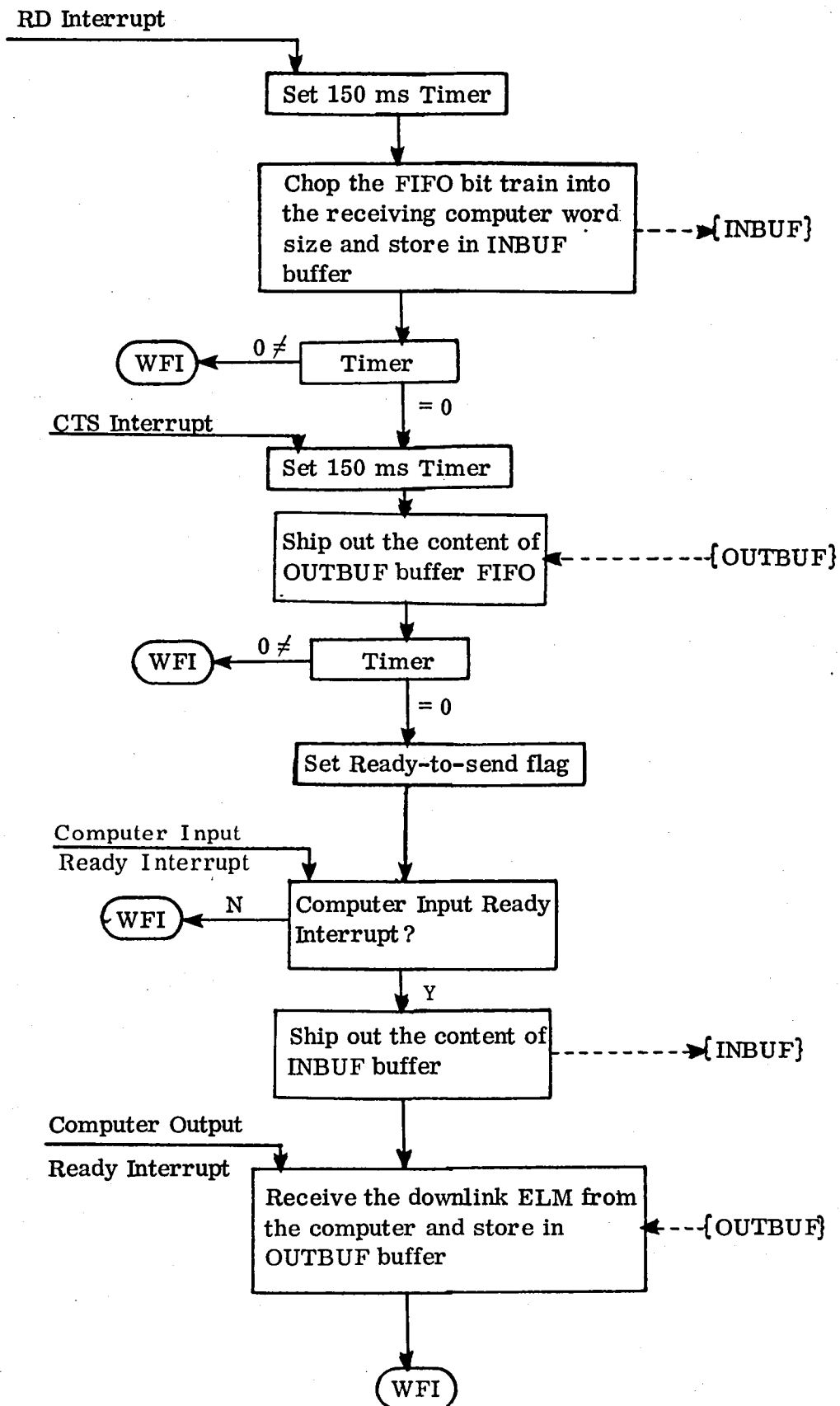


Figure 28. Conceptual Logic Flow for the ELM Transponder Interface Unit

the interface unit transmits the content of {INBUF}. The lowest priority interrupt from the computer indicates that it is ready to send the downlink ELM to the interface unit. When this happens, the interface unit opens a downlink buffer {OUTBUF} to receive the most current message available.

It should be noted that the above concept is based on the assumption that the Mode S transponder can handle at most one pair of uplink and downlink ELM per scan.

Signal Generator and CDTI Display Unit Requirements for the Display Unit (including size, refresh rate, contrast, symbology, and color capability) are dictated mainly by human factors considerations, the majority of which are still in the research stage. As mentioned previously, one of the advantages of a CDTI display, as compared to an ATARS or IPC display, is that the CDTI presents to the pilot not only the surrounding traffic information but also the map information which aids in navigation and guidance functions. If threat parameters (which are computed on-board based on uplinked information) are shown, then the display can serve a triple role: navigation and guidance display (e.g. EHSI), traffic information display, and threat monitor indicator. Table 8 lists desirable information contents of the display, some of which have been identified by CDTI research.

The price for additional display functions is that the associated information storage and handling requirements increase proportionally. In order to reduce these requirements it is necessary to make trade-offs among the items in terms of content priorities and symbology. For example, it may not be necessary to display the listed parameters for each target aircraft. Most probably the full content (including threat parameters, if applicable) would be shown only for two or three Proximity Type I aircraft. Other traffic would be shown only with their horizontal position. The track angle may be shown by the rotation of an aircraft symbol. Some of the map features may be deleted at certain map scales.

With judicious selection of display parameters, clever color coding of symbols, or more concentrated attention and effort by the pilot, a

TABLE 8

DESIRABLE CDTI DISPLAY CONTENTS

Map Display

- Route Structure: Waypoints, IAS and Altitude Window.
- Restricted Area, Mountains, Shoreline, Hazardous Weather
- Runway Geometry, Landing Aids (ILS or MLS)

Own Aircraft

- Position with respect to Map
- Prediction Vector (time or distance)
- Altitude and Ground Speed (alphanumeric)
- Separation Cues (CTD, CTP) and Flight Director
- Landing Sequence Number

Traffic Aircraft

- Position with respect to Map
- Prediction Vector and/or History Dots
- Relative Altitude, Vertical and Ground Speeds (alphanumeric or encoded)
- Flight ID Number, Landing Sequence Number, and Weight class
- ATC Flight Rule and ATARS/CDTI Equipage

Threat Situation Indicators

- TAU, CPA, Time and Range to CPA (computed on-board)
 - Suggested Resolution (or Escape) Maneuver.
-
-

smaller display unit may be workable. However, considering the amount of information to be displayed, it seems that a small weather radar display CRT (such as one being used by AIDS for ATARS) would not take full advantage of the CDTI concept. For research purposes, it is recommended that a larger, high resolution, color capable display be utilized. The airborne computer, signal generator, and display unit combination should be configured to allow rapid modification or implementation of various display formats, preferably some of which are pilot selectable.

Pilot-Computer Interface Unit A keyboard or a CDTI Mode Select Panel is needed to communicate pilot requests to the system. Figure 2 illustrates an example configuration. The destination of a transmitted pilot request may be the ground system (pilot request for CDTI flight initiation, etc.) or the airborne system (target symbol selection, etc.). Because these requests do not require high speed operation, a simple alphanumeric keyboard is sufficient,

Digital Data Acquisition System A digital data acquisition system is needed to store the airborne data for post-flight analysis purposes. Within the proposed CDTI system configuration, a limited amount of ELM downlinked data is stored with the merged ground data. However, due to the downlink capability limitation, the set contains only those variables pertinent to the Own aircraft kinematics (see Chapter II). The pilot related variables (control and actuator inputs), accelerations, and body rates need to be stored in the aircraft (or telemetered to a ground facility independent of DABS). Figure 5.b illustrates a schematic overview of the data storage configuration.

Airborne Computer Software Modules For CDTI Applications

Because the specifics of the available airborne software package into which the required CDTI software modules are to be implemented are not known, the following discussion is very general in nature and addresses only modules which have direct impact for the CDTI applications. It is assumed that the on-board system contains the usual navigation, guidance and control functions. (T-39 and TCV software are capable of these functions).

Figure 29 shows the computational flow of the required CDTI modules at a subroutine level. In actual implementation, these subroutines may reside in separate computers, or they may be imbedded within existing modules. The following discussion applies to Fig. 29.

SM Input Buffer Module This module unpacks and decodes the SM input buffer. It checks to see whether a pilot warning such as Minimum Altitude or Restricted Area* is present. If such is the case, then proper action must be taken by setting warning annunciator flags.

This module also checks to see whether the ground system requests a special reply, such as Flight ID, via Comm-B. In such a case, it requests proper keyboard input from the pilot, and it prepares a data set which would be packed in a later module. This module needs to be processed once per scan (approximately 4.7 sec).

ELM Input Buffer Module This module essentially unpacks and decodes the ELM input buffer and stores the data in a specially designated storage area called a Track File. The Track File contains all the pertinent information about each target aircraft, and it is equivalent to the Central Track Store of the ground system where all the State Vectors are stored.** The computation is performed every antenna scan.

If the input buffer contains an ELM which consists of General Map and Traffic Data formats (CDS = 01000100) per Figs. 8 and 9, then the buffer is unpacked accordingly. Geometric constants, site ID, etc. are stored for future reference, and the corresponding traffic information is entered into the Track File.

If the input buffer contains Own and Traffic Data Formats (CDS = 01000101) per Figs. 9, 12, and 13, then the buffer is unpacked accordingly.

* These advisories are uplinked by the DABS/ATARS ground system via SM protocol. As indicated later, these could be computed on-board independently.

** The details of the ground system are given in the next Chapter.

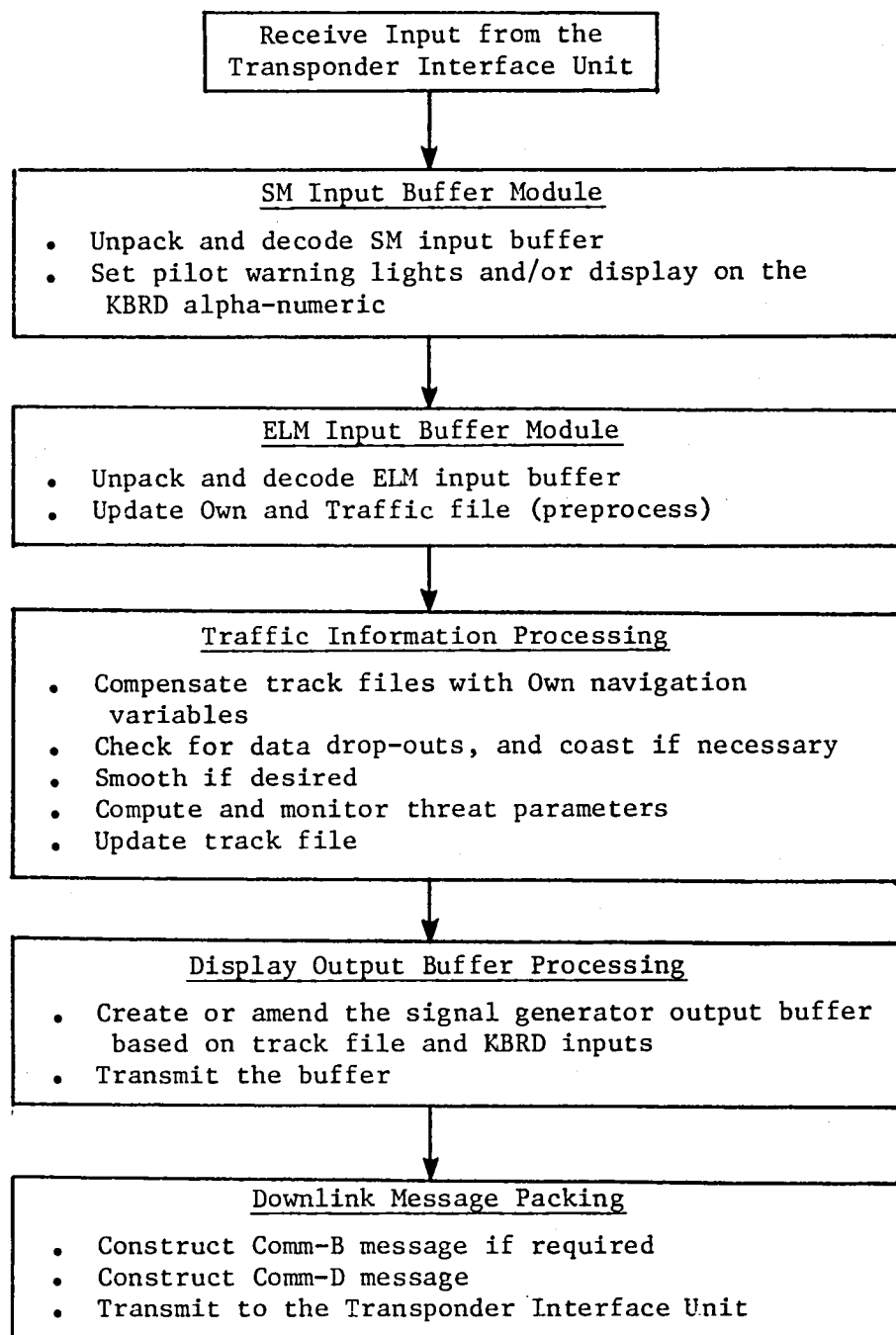


Figure 29. Essential CDTI On-board Computational Flow

Additional new target aircraft are added to the Track File, the variables for Own and Traffic are scaled to be compatible with the airborne variables, and finally the Track File is updated with the latest values.

Traffic Information Processing The bulk of CDTI related airborne processing is done by this module. The processing may involve data coasting (in case of a dropout), time synchronization of target data, coordinate transformation, compensation with respect to the air-derived variables, smoothing (as opposed to filtering), and threat parameter computation. It is understood that the computations are performed within the Track File data structure. After completion of this module, the Track File is assumed to contain necessary data for display processing.

When a data drop-out situation occurs due to a communication problem, all the traffic, including Own, must be coasted for one sample period by dead reckoning (or simple integration of indicated ground speed). The accuracy of such an algorithm for coasting target aircraft is directly proportional to the velocity estimation accuracy as obtained by the ground system. This implies that coasting is limited to one period only[†]. If the communication problem persists, then the CDTI flight data should be re-initialized.

Because the target reports are obtained by the ground system at various time points (the report times are available in the uplink data), it may be advisable to synchronize the target data to a convenient reference time. This is especially true for targets which are one or two ATARS sectors (described later) behind Own, because in this situation the traffic data is based on the DABS report which is lagging more than one full sweep, i.e., 4.7 sec. The report times should be monitored as system integrity indicators. (If a report time is not a monotonically increasing saw tooth function, the associated target data should not be trusted).

The Own and Target positions and velocities are computed in the ground system with respect to the north-east coordinate system centered

[†] This is similar to the logic used in AIDS.

at the DABS site. The information contained in the General Map and Traffic Data Format can be used to translate and rotate the uplinked kinematic variables to the extended runway coordinates used in the airborne navigation system. This was shown in the example associated with Fig. 10. Furthermore, the Target relative position can be improved by referencing the Own position to Own generated navigation quantities.

There is another pay-off for using Own navigation quantities. The threat parameters depend on relative velocity estimates; thus, the ground tracker errors enter twice into these computations - once for Own and once for each Target. The resulting error is therefore greater than necessary. (Follow a similar argument used on p 28-29.)

The relative Target look-angle can be improved by utilizing heading information from the on-board instrumentation. This will enhance the pilot's see-and-be-seen tactics in VFR applications of CDTI.

For a selected Target in a selected CDTI application, a smoothing algorithm may be employed. If the application is in-trial following (station keeping) with a constant time delay (CTD) criterion, then the Target's past position and velocity information are more important than the current. In this case, a smoothing algorithm would yield more accurate information for the task. In the smoothing process, the dynamic error due to the ground tracker must be compensated, and any data drop-outs must be handled properly. Another consideration is that a certain amount of past position and velocity data must be stored in "stacked shift registers". For example, at least 100 sec of past data (22 sample points) would be required for a 100 sec CTD following criterion. If history dots are needed to display target aircraft past positions, the additional memory requirement would not be substantial.

Certain threat parameters may be computed and incorporated within the CDTI display format. The parameters are computed based on the current state estimates, and the relative kinematics are assumed to be rectilinear, i.e., straight lines.* The resulting parameters should

* The same assumption is used for the ATARS logic.

be more accurate than the similar quantities obtained by the ATARS/AIDS algorithms. This is because the data uplinked for the CDTI applications are finer, and the Own quantities are compensated by the on-board navigation system. The following notation is used:

$$\begin{aligned}\Delta x &= x_T - x_O, & \Delta y &= y_T - y_O, \\ \dot{\Delta x} &= \dot{x}_T - \dot{x}_O, & \dot{\Delta y} &= \dot{y}_T - \dot{y}_O,\end{aligned}$$

where x_T , y_T , \dot{x}_T and \dot{y}_T are the Target estimates as uplinked, and x_O , y_O , \dot{x}_O and \dot{y}_O are the Own estimates compensated by the navigation system. They are, of course, referenced to a common coordinate system.

Possible computations then include:

- (a) TAU: This variable is defined as the ratio of current range over range rate ($\rho/\dot{\rho}$), where ρ is given by

$$\rho = (\Delta x^2 + \Delta y^2)^{1/2}.$$

Then,

$$\text{TAU} = (\Delta x^2 + \Delta y^2) / (\Delta x \dot{\Delta x} + \Delta y \dot{\Delta y}).$$

It represents the time left until collision. A TAU value with a magnitude of greater than, say -30 sec, may require a special pilot warning. It is noted that the range must be decreasing in order for a target to be a threat. This implies that the closure rate ($\dot{\rho}$) must be negative; hence, a negative value of TAU indicates a potential threat.

- (b) Closest Point of Approach (CPA): CPA is defined as a point at which a given target aircraft will be closest to Own. The distance between Own and Target at CPA is called the miss distance. If rectilinear motion is assumed, then the distance between Own and Target is given by a function of future time, T as

$$d(T) = [(\Delta x + T \dot{\Delta x})^2 + (\Delta y + T \dot{\Delta y})^2]^{1/2}$$

The time to CPA, T_{CPA} is obtained by minimizing $d(T)$ with respect to T,

$$\begin{aligned}
T_{\text{CPA}} &= (\Delta x \dot{\Delta x} + \Delta y \dot{\Delta y})^2 + (\dot{\Delta x}^2 + \dot{\Delta y}^2) \\
&= -(\Delta x \dot{\Delta x} + \Delta y \dot{\Delta y}) / \Delta V^2,
\end{aligned}$$

where ΔV is the relative speed given by

$$\Delta V = (\dot{\Delta x}^2 + \dot{\Delta y}^2)^{1/2}.$$

The miss distance, m is given by substituting T_{CPA} into $d(T)$.

Thus,

$$m = d(T_{\text{CPA}}) = (\Delta x \dot{\Delta y} - \Delta y \dot{\Delta x}) / \Delta V.$$

CPA for Own and Target are given by

$$\begin{aligned}
\begin{bmatrix} x_{\text{CPA}} \\ y_{\text{CPA}} \end{bmatrix} &= \begin{bmatrix} x_o \\ y_o \end{bmatrix} + T_{\text{CPA}} \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \end{bmatrix}, \\
&\text{Own}
\end{aligned}$$

$$\begin{aligned}
\begin{bmatrix} x_{\text{CPA}} \\ y_{\text{CPA}} \end{bmatrix} &= \begin{bmatrix} x_T \\ y_T \end{bmatrix} + T_{\text{CPA}} \begin{bmatrix} \dot{x}_T \\ \dot{y}_T \end{bmatrix} \\
&\text{Target}
\end{aligned}$$

The distance to CPA for Own is given by

$$D_{\text{CPA}} = T_{\text{CPA}} \sqrt{x_o^2 + y_o^2} = T_{\text{CPA}} V_G$$

Altitude separation at CPA is given by

$$\Delta h_{\text{CPA}} = \Delta h + T_{\text{CPA}} \dot{\Delta h},$$

where Δh and $\dot{\Delta h}$ are relative altitude and vertical rate based on current estimates.

The various threat parameters are summarized in Table 9. The parameters may be displayed to the pilot in tabular form off to the side of the main CDTI display (e.g., upper right hand corner). The CPA for Own and Target may be shown along with the map information as a flashing special symbol, if the situation warrants pilot warning. These are merely possibilities.

TABLE 9
THREAT PARAMETERS

Threat Parameter		Formula	Warning Condition
TAU	Closure time	$(\Delta x^2 + \Delta y^2) / (\Delta x \dot{\Delta x} + \Delta y \dot{\Delta y})^*$	TAU > -30 sec
m	Miss distance	$(\Delta x \dot{\Delta y} - \Delta y \dot{\Delta x}) / \Delta V$	m < 1.2 nmi
T _{CPA}	Time to CPA	$-(\Delta x \dot{\Delta x} + \Delta y \dot{\Delta y}) / \Delta V^2$	T _{CPA} < 30 sec
CPA	Own	$\begin{bmatrix} x_{CPA} \\ y_{CPA} \end{bmatrix} = \begin{bmatrix} x_o \\ y_o \end{bmatrix} + T_{CPA} \begin{bmatrix} \dot{x}_o \\ \dot{y}_o \end{bmatrix}$	_____
	Target	$\begin{bmatrix} x_{CPA} \\ y_{CPA} \end{bmatrix} = \begin{bmatrix} x_T \\ y_T \end{bmatrix} + T_{CPA} \begin{bmatrix} \dot{x}_T \\ \dot{y}_T \end{bmatrix}$	_____
D _{CPA}	Distance to CPA	$T_{CPA} \cdot (\dot{x}_o^2 + \dot{y}_o^2)^{\frac{1}{2}}$	2nmi @ 250 knot
Δh _{CPA}	Relative attitude @ CPA	$\Delta h + T_{CPA} \dot{\Delta h}$	Δh _{CPA} ≤ 500 ft

* $\Delta x = x_T - x_o, \Delta y = y_T - y_o, \Delta h = h_T - h_o$
 $\Delta \dot{x} = \dot{x}_T - \dot{x}_o, \Delta \dot{y} = \dot{y}_T - \dot{y}_o, \Delta \dot{h} = \dot{h}_T - \dot{h}_o$
 $\Delta V = (\Delta \dot{x}^2 + \Delta \dot{y}^2)^{\frac{1}{2}}$

Equivalents of the ATARS Terrain, Obstacle, or Restricted Area Advisories can also be computed here, if they are deleted from the DABS/CDTI uplink information due to channel loading considerations. Interpolation algorithms using tabular values (table look-up procedures) would be suitable. Furthermore, mountain peaks with MSL altitude shown numerically and restricted airspace can be incorporated into a CDTI display format (under a certain map scale) as map elements.

This module also needs to be computed once per scan. However, some time-critical computations, such as updating of threat parameters, may be performed more frequently as the Own navigation information becomes available. However, it is believed that the tracker error would probably cancel any marginal advantage.

Display Output Buffer Processing The main function of this module is to create a buffer which contains necessary instructions and data for the CDTI display signal generator. Inputs are stored in the Track File, the global memory containing map information, and the keyboard or mode select panel buffer containing pilot entries.

First, the keyboard buffer is checked for pilot option entries such as map scale, special symbols, alphanumeric data tag option, history dots, and prediction vectors. Depending on the options, a background picture is drawn containing the map features and target information. Then, the symbology associated with Own is super-imposed on top of the background picture to complete the output buffer. For CDTI purposes, the most useful mode is one in which the Own symbol is fixed with respect to the display area. The background display is in either a track up or heading up orientation. This means that the background picture "slides and rotates" under Own's symbol.

Unless dictated by pilot entries, the background buffer only needs to be updated once per scan because the traffic and map elements remain stationary during the period. The Own aircraft buffer, however, needs to be created at a display refreshment rate which is on the order of 200 msec.

Downlink Message Packing This module takes care of downlink message construction. It consists of two submodules - one for Comm-B message and the other for Comm-D (ELM) messages.

Pilot entries stored in the keyboard or mode select panel buffer are checked for a pilot request message. Also, Comm-B reply request flags (as requested by Comm-A interrogation) are checked. If a downlink message is required, then it is packed per SMIU and CDTI downlink format specifications. These are stored in an SMIU output buffer, and a flag is set for the I/O function. This submodule needs to be performed when the need arises.

The navigation variables (position, velocity, IAS, heading, etc.) for post-flight data analysis are packed and stored in a buffer compatible with ELMIU specifications. This submodule needs to be performed once per scan. To minimize the time delay, the variables should be packed toward the end of a DABS period so that the latest values are available for downlinking.

Some Implementation Notes The bulk of the CDTI computations and processing (as discussed above) needs to be performed only once per antenna scan period of 4.7 sec. This is very much longer than a typical airborne computer integration cycle period of 50 msec. Thus for implementing the above CDTI modules into existing on-board software ("series" configuration), it is advantageous to do so in a background (low priority) computation loop rather than in the high priority foreground (provided that there is sufficient dead time available). A minor portion of the logic, such as servicing the I/O functions, must be performed in foreground.

For a "parallel" configuration where a dedicated CDTI computer is installed, there would be less implementation problems. The real time considerations are the Transponder Interface Unit and input processing from the navigation, guidance, and control computer.

It is roughly estimated that a medium speed, 8K x 16 bit word memory computer (or microprocessor) is required for the task, provided that the actual display generation is performed in a dedicated signal generator. Table 10 shows the estimated memory requirement. These estimates are for the parallel configuration case. The estimates for the series case will depend on the existing software.

TABLE 10

MEMORY REQUIREMENT FOR ON-BOARD CDTI IMPLEMENTATION

Computer Function	Amount (K words)
Track File & Constants	0.6
Pilot Entry Processing	0.5
Uplink Message Processing	1
Traffic Information Processing	2.5
Display Buffer Processing	2
Downlink Message Processing	0.5
Miscellaneous and Spare	0.9
TOTAL	8 K

IV

DABS/ATARS BASED CDTI GROUND SYSTEM

Background

Currently there exists no ground system which can directly support the CDTI requirements as envisioned by the previous two chapters. The concept which comes closest to fulfilling the CDTI requirements is ATARS. The ATARS functional definition has recently been re-evaluated within the FAA. Thus, the status of future ATARS implementation is nebulous at best. Several versions of ATARS have been developed, however, and one version may be modified for NASA/CDTI flight experiment support. A short introductory description of ATARS is given in this chapter.

Automatic Traffic Advisory and Resolution Service (ATARS) As the name implies, ATARS, which is an outgrowth of the Intermittent Positive Control (IPC) concept, is designed as a ground-based collision avoidance and conflict resolution system. This would be accomplished by uplinking messages, called advisories, to concerned aircraft, and the contents would be displayed for proper pilot action. The ATARS advisories, such as Own, Proximity, or Conflict Resolution advisories, are computed based on the DABS sensor reports. Figure 30 shows the ATARS-DABS sensor interface.

In order to minimize the computational delay and to maintain the synchronization with the DABS sensor beam rotation, the ATARS computation is organized on a sector basis. There are sixteen computation sectors, each 22.5 deg wide, which surround the DABS sensor site. All the aircraft within a sector are processed at the same time. Figure 31 presents a block diagram of the sector processing algorithm at a subroutine level. It also shows important times (called critical times) associated with major ATARS subroutines. Figure 32 shows a much more simplified flow chart with the associated completion time windows. Time is measured in terms of the number of sectors after beam passage over the tracked aircraft.

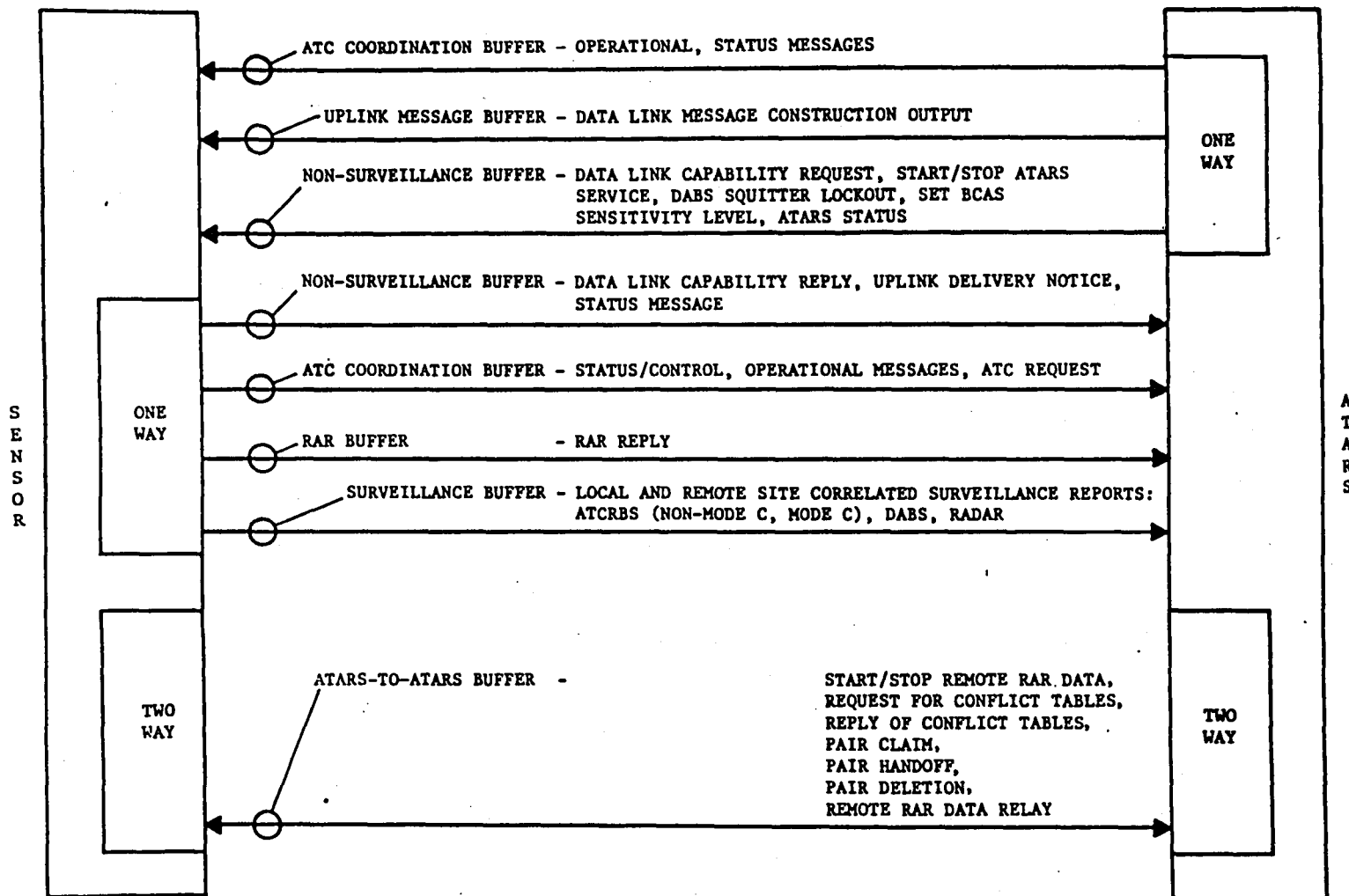
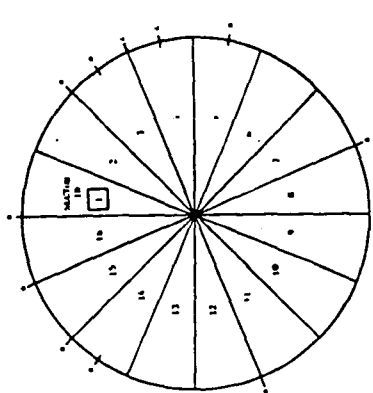


Figure 30. ATARS-Sensor Interface Diagram [13]



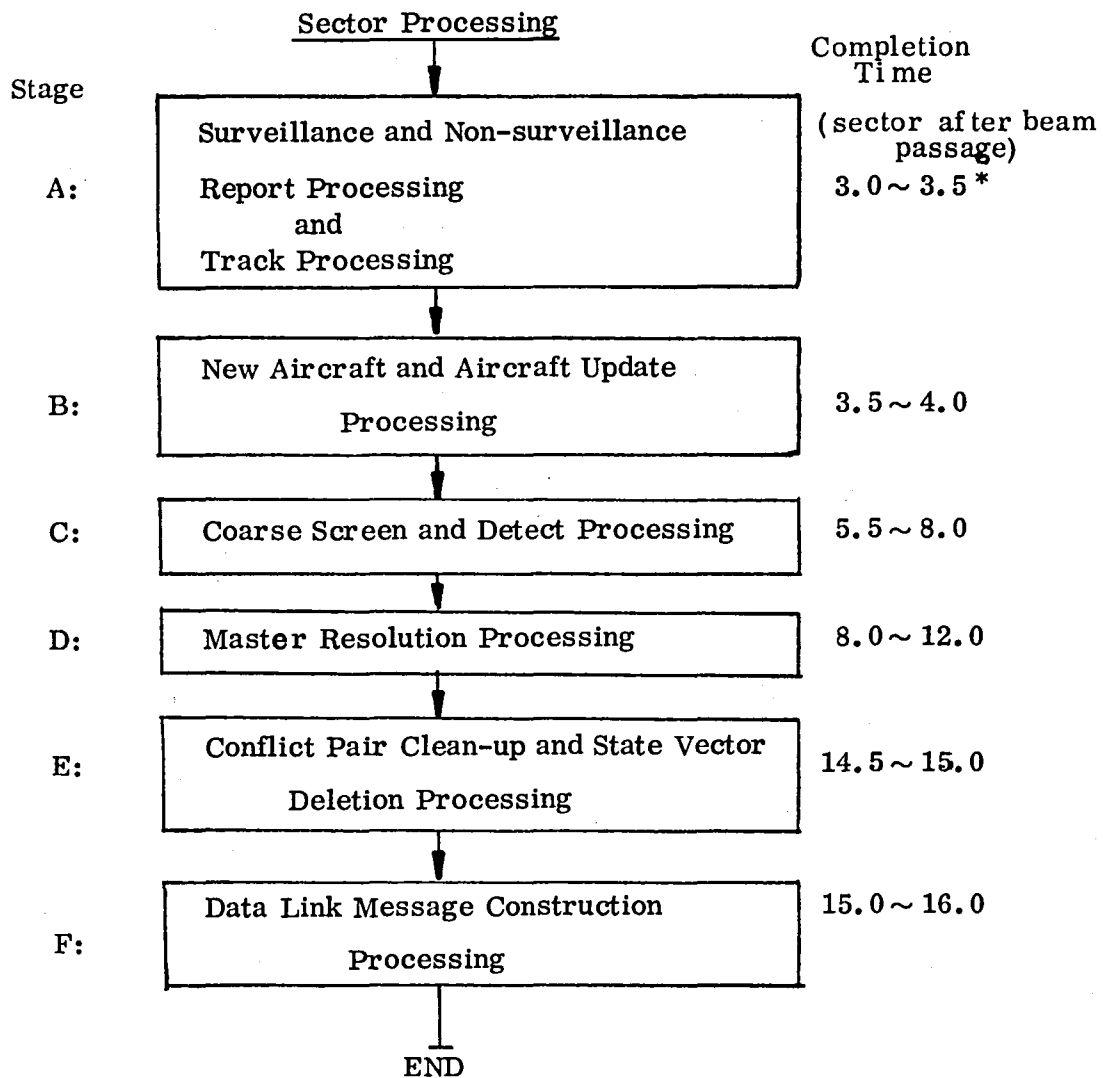
3. REMOTE CRITICAL POINTS IN TASE ANALYSIS. (NOTED IN EXISTING ANALYSIS LINES IN GLANDER) WERE SEVERAL TIMES NOT IN CORRECT POSITION IN 1961.

INDICATES DESIGNATION OF PARTICULAR ATTIVE SECTION

ALL SECTION'S DATA MUST BE PRELIMINARILY REVIEWED BEFORE MOBILE MAY BE EXPEDITED

NO DELAYED PRELIMINARILY FOR SECTION DATA AFTER THIS POINT IS TIME

1	1997
2	1998



* Numbers indicate ATARS antenna sector period. The corresponding real time may be obtained by multiplying by 293 msec.

Figure 32. Computation Flow of Essential Tasks for ATARS Sector Processing

Figure 33 illustrates the overall computation flow management referenced at sector time. It is obvious that numerous computations must be performed simultaneously in a parallel fashion.

Functional descriptions of the major processing modules of ATARS shown in Fig. 32 are as follows:

- A. Surveillance and non-surveillance reports from the DABS sensor are processed. The reports include Mode A (radar only), Mode C (ATCRBS) and Mode S (DABS) reports as well as any downlinked messages addressed to ATARS. The "raw" position data are fed into a tracker algorithm (similar to TABG filter) to update the aircraft position and velocity estimates. Pertinent aircraft information is stored in an area called "State Vector". The State Vectors are organized within the Central Track Store memory structure for easy access.
- B. Aircraft positions are updated or added (in case of a new aircraft) to either X-list or EX-list* depending on their speed and altitude. The State Vectors are rethreaded along the lists. Special care is taken for aircraft within the hub area (nominally within 5 nmi of the sensor site), because they may move across sector lines rapidly.
- C. The updated X- and EX-lists are examined for potential conflict situations. An aircraft pair which may pose a conflict is entered into the Potential Pair lists. This is done by the Coarse Screen Task. Potential Pair lists are further examined, and conflict determinations are made. These are entered in PWILST (Proximity Warning Indicator) lists. Also Terrain/Obstacle/Restricted Area situations are entered into PWILST lists. This is the Detect Task.
- D. PWILST lists and the RAR (Resolution Advisory Registers) are examined for uplink message pre-processing. The advisories from adjacent ATARS sites are examined for possible inclusion.

* The X- and EX-lists are ordered on the X coordinates of the aircraft with the DABS site as the center post. The X-list includes all aircraft whose altitudes are below a threshold altitude (nominally 10,000 ft AGL) and whose ground speeds are below a limit value (nominally 240 kt). All other aircraft are on the EX-list. The X- and EX-lists are used to index the State Vectors of all aircraft within the ATARS coverage. This indexing (called "threading") must be updated (or rethreaded) every scan because of the ever changing relative aircraft positions. This threading concept (a special indexing procedure) is used throughout the ATARS data structure, including the state vectors and PWILST lists, to facilitate the retrieval of necessary data from the central track store in an orderly fashion (i.e., with a minimum search time).

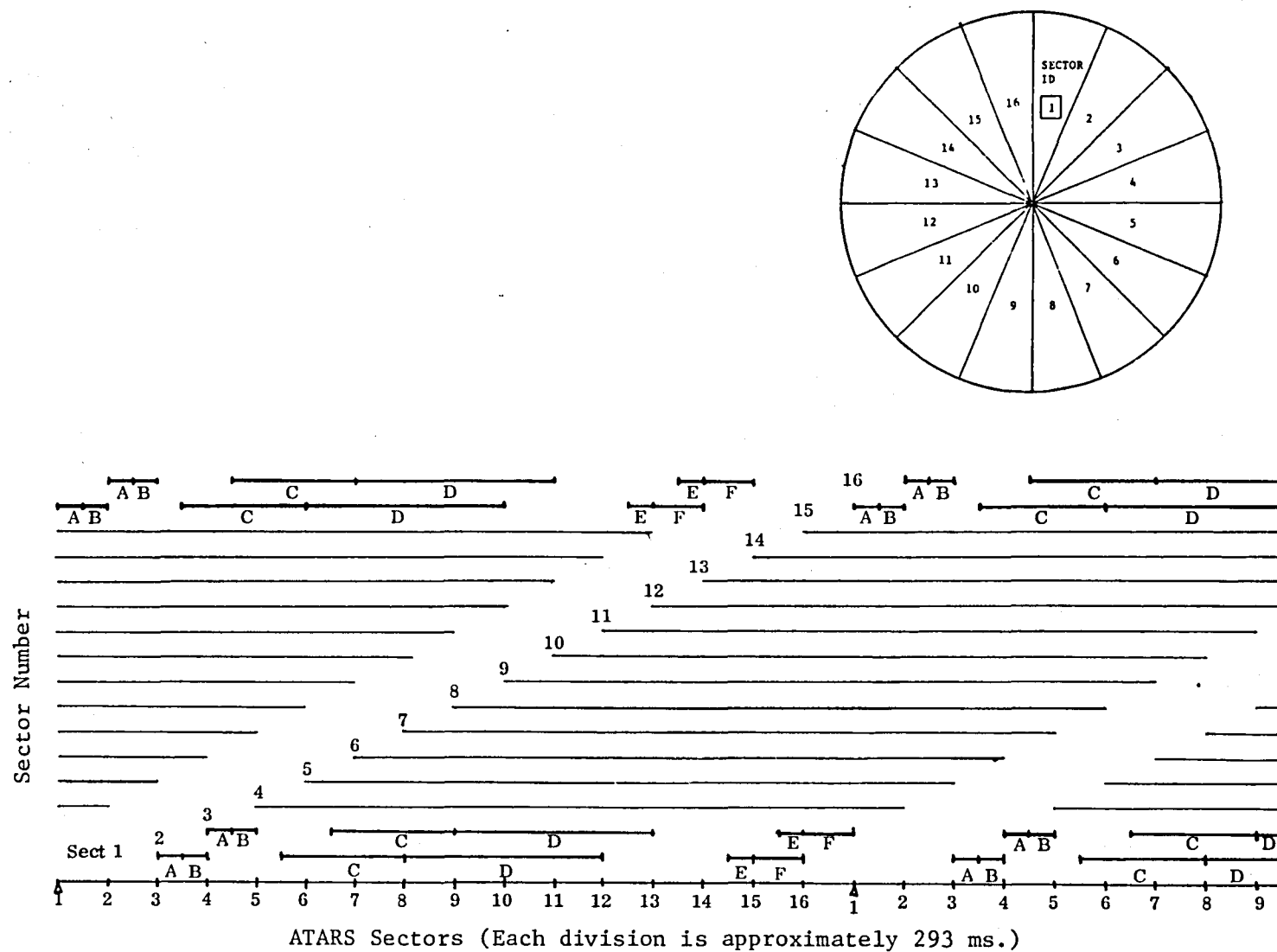


Figure 33. Time Windows for ATARS Processing Tasks with Respect to ATARS Sectors

- E. Outdated advisory registers and State Vectors associated with aircraft no longer within the ATARS coverage are removed.
- F. The PWILST list for each aircraft is examined and prioritized to include up to eight entries. The appropriate advisories are constructed from the data contained in the PWILST lists according to the ATARS uplink format definitions. The uplink priority of each message is assigned and stored in a buffer to be transmitted to the DABS sensor. This is done in the Data Link Message Construction Processing Task.

Among these modules, the ones which may be directly applicable for CDTI support are A, B, E and F. It should be understood that these are possibilities based on available documentation. The actual implementation of the required modifications will depend on the available resources in terms of ground software support, various time elements such as design, coding, debugging and validation periods and memory and real time considerations. However, the current design may point to a general direction for undertaking the complex design task.

Modification Philosophy

As can be seen in the introductory material, the ATARS software is based on a highly complex computational structure with strict adherence to the critical real-time constraints. Any attempt to modify a program of this magnitude and complexity must be pursued with utmost care. This dictates that the number of CDTI modifications be limited to an absolute minimum.

There are two different approaches for accomplishing the required CDTI functions:

Approach 1 - Retain only those ATARS modules which are directly applicable and add to them a specially designed CDTI software package. The resulting software may not support ATARS functions, because timing may be altered.

Approach 2 - Retain the ATARS software in its entirety and incorporate the CDTI support software in a parallel fashion so that both are supported. The very impact of such an approach is that many ATARS modules and their data structure would be duplicated.

As usual, a compromise solution exists between the two extreme approaches. In the following section, CDTI requirements are delineated and discussed. Then, selected modification details are proposed at a subroutine level.

CDTI Modification Requirements

CDTI Data Formats For CDTI support purposes, the DABS Comm-B, Comm-C and Comm-D data formats are primarily utilized. The detailed format specifications are previously given in Chapter II. Pilot request formats of BDS equal to 01010000 and Type Codes equal to 001000 and 001001 are directed to ATARS. Also directed to the ATARS is the downlink ELM with the DDS designation of 01000101 which contains the downlink information. Messages containing the above subfield designations must be unpacked and interpreted accordingly.

CDTI uplink transactions are carried out through ELM protocol. Proper interface for transferring ELM messages to the DABS sensor must be added if the capability does not currently exist. It is noted that some of the required data, notably the flight ID and landing sequence number, are not available within the ATARS data structure. This means that they may have to be brought into the ATARS software from an ATC computer complex. The CDTI ELM uplink formats are given in Figs. 8, 9, 12 and 13.

Data Storage For post-flight analyses purposes, the downlinked variables contained in DDS equal to 010000101 (Figs. 6 and 7) need to be merged with the ground-derived variables, and the combined data set is then stored in a magnetic tape or a similar device. The merging and storage should be done with minimum time delay, preferably at a regular

time interval synchronized at the DABS antenna rotation period. This will minimize the amount of intermediate tape processing such as interpolation between different time points.

Tracker Algorithm (This item is more of a remark than a requirement.) Simulation studies [18,20] have shown that the tracker algorithm currently implemented in the ATARS software may pose certain problems for CDTI applications. This stems from the fact that the velocity error (notably in the course angle) due to the tracker lag becomes very large when an aircraft undergoes a turning maneuver.

It does not seem as though the tracking accuracy can be improved in the near future. One reason is that currently there is no easily implementable tracking algorithm based only on the range and bearing measurements with substantially better tracking performance. An algorithm which reduces the tracker lag error to negligible magnitude, uses the downlinked heading as well as the DABS sensor measurements. [18] Implementing this or a similar algorithm is not relevant at the present time when most of the current traffic population is either Mode C or Mode A transponder equipped. However, for CDTI flight test research, the algorithm may be implemented to be used with a "cooperative" aircraft which has the Mode S Comm-B capability.

As discussed in the avionics section, the best strategy to improve the ground tracker signal is to compliment the Own data with estimates obtained by the on-board navigation system. This approach will at least attenuate the effect of Own tracking error in the computed relative kinematic variables. Also for certain CDTI applications, an on-board smoothing algorithm may be devised to process the uplinked traffic data.

CDTI Traffic Selection Criteria To select surrounding traffic for CDTI display is not a trivial task, especially in a high density area. In fact, it is an on-going research topic. The traffic selection depends on the coverage volume, the appropriate number of aircraft to be displayed to the pilot, and the uplink channel capacity. The number of displayed

aircraft is dependent upon the CDTI function. The proximity coverage implemented in the ATARS is determined dynamically by

$$\text{relative range} \leq (\hat{V}_O + \hat{V}_T) * 30 \text{ sec},$$

where \hat{V}_O and \hat{V}_T are estimated Own and Target ground speeds. The number of aircraft is limited to eight, mainly because of DABS communication channel loading considerations. A study by Boeing contains a more comprehensive analysis of the appropriate number [2]. Table 11 from Ref. 2 shows the recommended coverage volume and number of displayed aircraft according to the specific CDTI function. The Boeing study does not indicate how to reduce the traffic numbers to the values shown, if there are more aircraft within the given volume.

For general CDTI applications, the required CDTI coverage volume is obtained by taking the "least common denominator" of the volumes defined in the Boeing study (except for the Severe Weather Avoidance function). The rationale is that the CDTI functions listed in Table 11 are not mutually exclusive. For example, the pilot could use the CDTI display for an in-trail spacing control task at the same time he is monitoring the surrounding traffic. General traffic monitoring coverage is illustrated in Fig. 34. The along-track and cross-track distances are computed according to the geometry shown in Fig. 35. The relative altitude limits of ± 6000 ft are based on the 3:1 rule, i.e., 3 nmi DME change for a 1000 ft altitude change.

The resulting CDTI coverage area represents a fairly large area covering 600 nmi^2 . This means that there could be up to 180 aircraft within the coverage (at a projected peak density of 0.3 aircraft per square mile). At a more moderate density of 0.05 aircraft per square mile, there would be up to 30 aircraft. Obviously, a reasonable and preferably simple selection procedure is needed. If too many aircraft are selected, then the DABS communication channel would be taxed heavily, and the display would become so cluttered and busy that it would be rendered useless to the pilot. If, on the other hand, too few aircraft are selected, then the main CDTI functions of being a surrounding traffic monitor and a blunder detection device would be compromised.

TABLE 11

COVERAGE VOLUME AND TRAFFIC NUMBERS [2]

Potential CDTI Function	Coverage Volume	Traffic No.
<u>Traffic Monitor Roles</u>		
1. General Traffic Monitoring	$\begin{pmatrix} +20 \\ -10 \end{pmatrix} \times (\pm 10) \times (\pm 2500)$ **	6 - 20
2. Longitudinal Separation of Arrivals	$(+7) \times (\pm 3)$	1
3. Independent Parallel Approaches	$(+10) \times (+5) \times (\pm 1000)$	2
<u>Active Control Roles</u>		
4. Arrival Merging	$(\pm 10) \times (\pm 10) \times (\pm 1000)$	2
5. Arrival In-Trail Spacing Control	$(+15) \times (\pm 5) \times (\pm 5000)$	1 - 2
6. Enroute Passing and Crossing	$(\pm 10) \times (\pm 10) \times (\pm 5000)$	1
7. Severe Weather Avoidance	$(+100) \times (\pm 50) \times (\pm 5000)$	6

$$** \begin{pmatrix} +\Delta L \\ -\Delta L \end{pmatrix} \times \begin{pmatrix} +\Delta C \\ -\Delta C \end{pmatrix} \times (\pm \Delta h)$$

$+\Delta L$ (nmi) = ahead of Own; $-\Delta L$ (nmi) = behind Own

$\pm \Delta C$ (nmi) = lateral cross track

$\pm \Delta h$ (ft) = above or below relative altitude

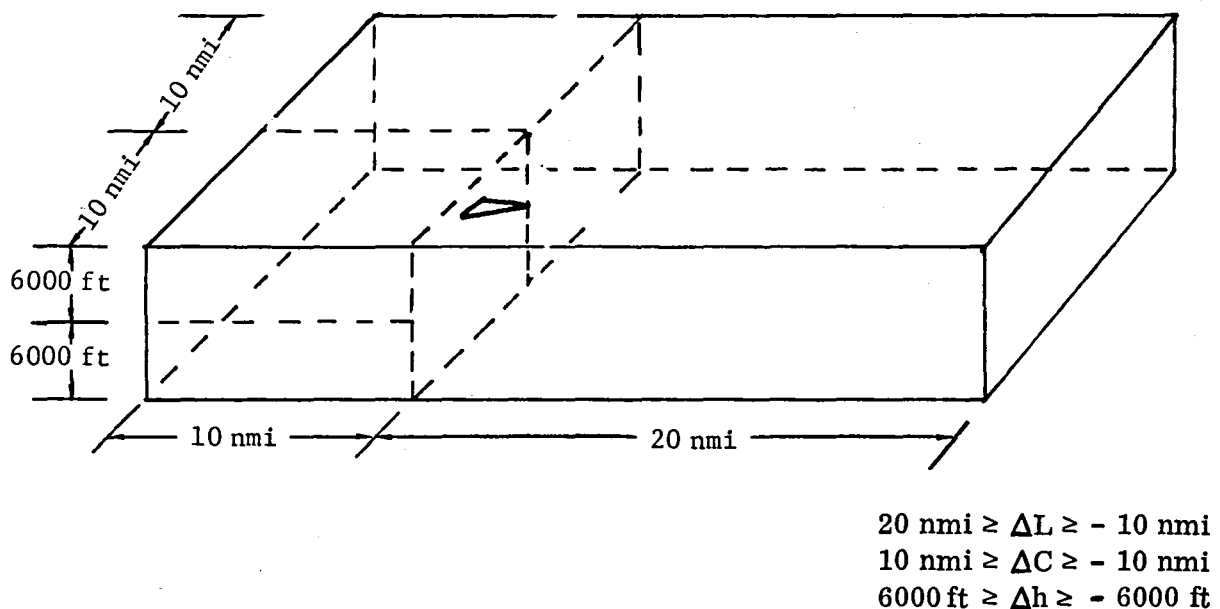
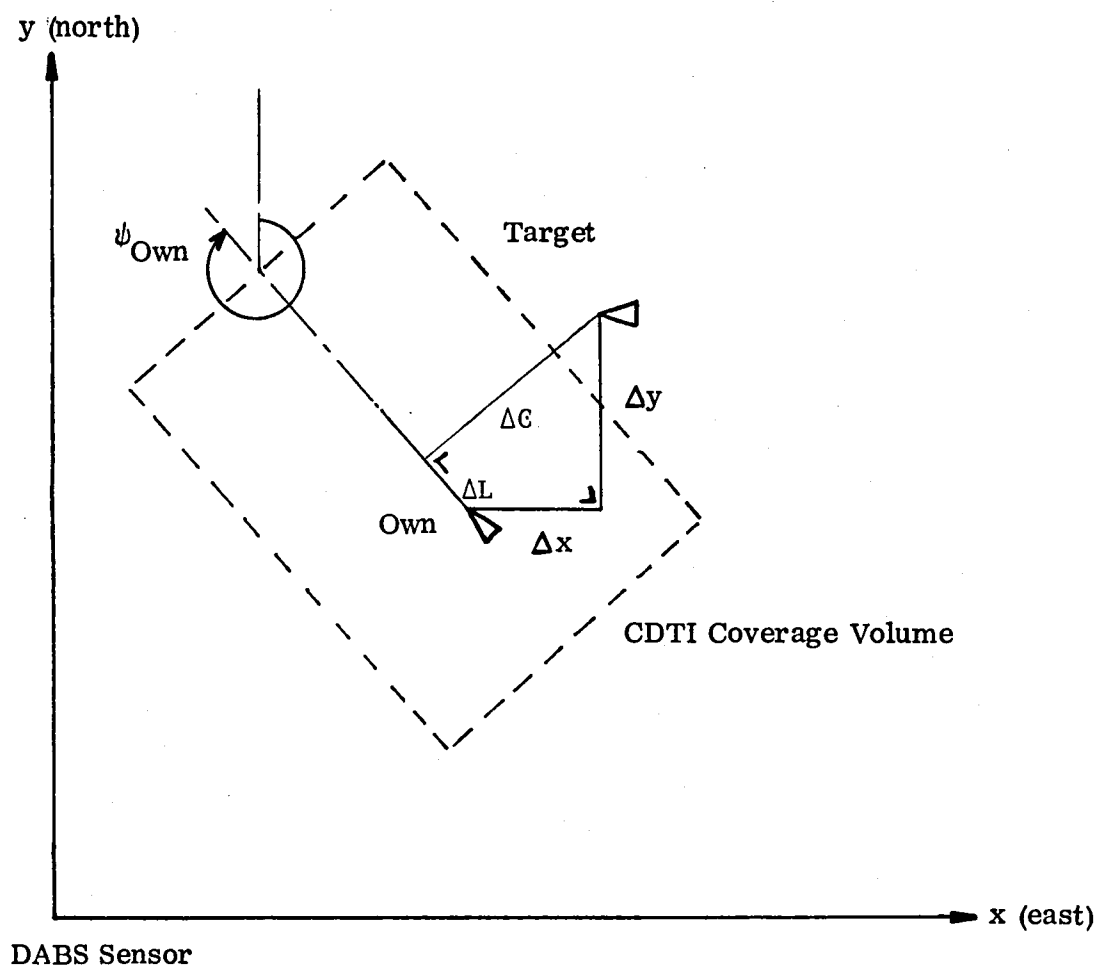


Figure 34. Illustration of CDTI Coverage Volume

The Boeing study results (Table 11) indicate that the number of pertinent aircraft to be displayed would be one or two except for the traffic monitoring function; in this case, up to 20 aircraft would be displayed. Therefore, one solution is to divide the traffic into two groups according to a priority assignment. This idea leads to the concepts of Proximity Type assignment and time multiplexing data uplink procedure. A few (up to 3) high priority aircraft would be shown with their full data set every scan, and many (up to 18) low priority aircraft would be shown with very limited data sets (position indication only) every other scan. This concept was incorporated into the CDTI uplink Data Format specifications in Chapter II. Accordingly, these specifications satisfy the following three requirements without taxing the DABS communication channel.

- a. It can handle up to 21 target aircraft with one uplink ELM message;
- b. A few critical targets can be displayed with full data sets that would minimize the display clutter; and
- c. The pilot's traffic monitoring function is satisfied because he would be made aware of many surrounding aircraft.



$$\left. \begin{array}{l} \Delta x = x_T - x_o \\ \Delta y = y_T - y_o \end{array} \right\} \begin{array}{l} \text{With respect to} \\ \text{DABS North-East} \\ \text{Coordinate System} \end{array}$$

$$\left. \begin{array}{l} \Delta L \\ \Delta C \end{array} \right\} \begin{array}{l} \text{With respect to Own aircraft} \\ \text{angle, } \psi_{\text{own}} \end{array}$$

$$\begin{bmatrix} \Delta L \\ \Delta C \end{bmatrix} = \begin{bmatrix} \sin \psi_{\text{own}} & \cos \psi_{\text{own}} \\ \cos \psi_{\text{own}} & -\sin \psi_{\text{own}} \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}$$

Figure 35. Along Track-Cross Track Geometry

If the surrounding traffic within the CDTI coverage volume remains the same throughout the Own flight duration, then the selection procedure would be very simple. However, some aircraft will fly into the volume; others may fly away. A high speed aircraft may "pop in and out" in a short duration of time. Possibilities are numerous, and it is likely that the selected CDTI traffic will change from scan to scan, especially when considering a somewhat large coverage volume and the uplink limitation of the number of CDTI aircraft. Therefore, given the coverage volume and the maximum number of aircraft whose data can be uplinked, two problems remain to be solved in order to complete the traffic selection procedure. The first problem is to limit the number of selected target aircraft to within the maximum, if there are more aircraft in the coverage volume than the uplink channel can transmit. The second problem is to assign the proximity type to each selected aircraft.

The ATARS selects a maximum of eight targets which are implicitly prioritized depending on the threat status by means of several criteria. In particular, the proximate aircraft are ranked according to the weighted relative range (the altitude is weighted five times the horizontal component). A similar ranking procedure may be applicable for the CDTI traffic selection procedure.

The CDTI traffic selection procedure has not been defined in detail (beyond the scope at this stage); however, it is recommended that certain underlying rules are followed. Some of these are:

- a. A proximity type assignment by the pilot overrides the ground software assignment;
- b. The number of aircraft transitioning from proximity Type II to I and vice versa is limited to one per scan; and
- c. The number of new proximity Type II aircraft per scan is limited to six.

Rules (b) and (c) are included for display stability purposes, (i.e., to prevent an aircraft to be selected, dropped and selected again during a short interval of time). Figure 36 presents a schematic diagram for the traffic selection and proximity type assignment procedure transitioning

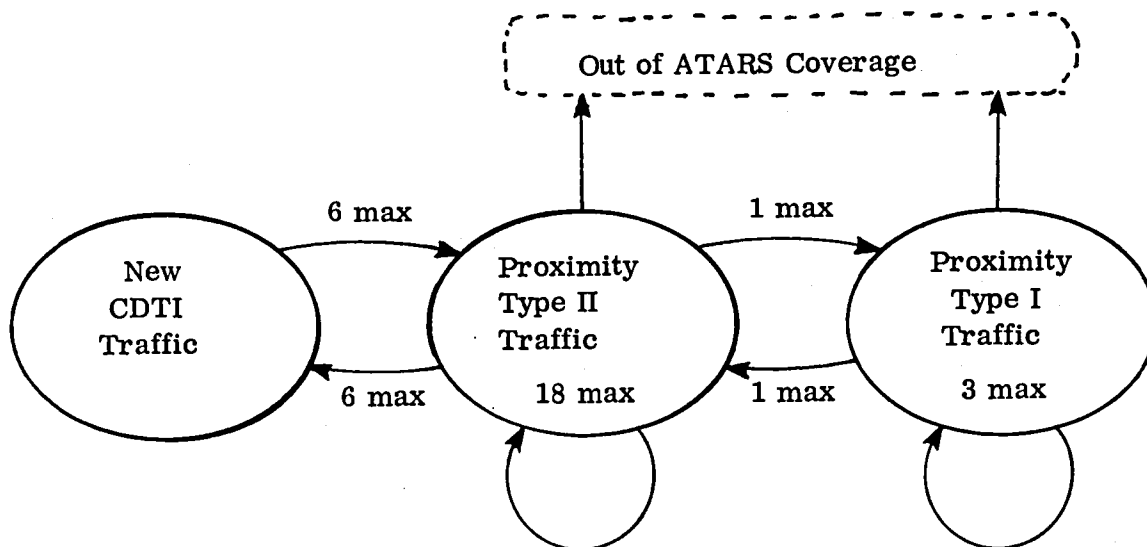


Figure 36. Proximity Type Transition Diagram

from one scan period to the next. It illustrates conceptually how new CDTI traffic is added to the uplink list or deleted from the list, and how a Type II aircraft is upgraded to Type I and vice versa.

Proposed CDTI Ground Software Modifications

The following discussion is directed toward further development and refinement of CDTI support software imbedded within the ATARS software structure. Undoubtedly, the DABS/CDTI software design task will undergo several iterations among concerned groups of researchers and designers; therefore, the discussion contained here must be considered merely as a first step in the iteration cycle.

Currently, ATARS is designed to serve three user classes (Classes 0, 1 and 2) depending on the on-board avionics and display capability. The purposes behind the service class designation are two-fold - to serve a wide user spectrum and to minimize the DABS communication channel loading. The ATARS class designation is stored in the State Vector (SVECT.ACLASS in Pseudo Code E notation) based on the downlinked information in the extended capability subfield (bit 45 through 60 of the Comm-B format).

To extend the service provided to CDTI equipped aircraft, a fourth service class (Class 3) may be created. The CDTI capability information could be downlinked from the aircraft in two ways:

- a. Use the ACS (Avionics Capability) or ECS (Extended Capability) subfield of the ATARS directed Comm-B; or
- b. Use the Pilot Request Comm-B (BDS = 01010000) as defined previously (Fig. 3).

The advantage of (a) is that the message is recognized by the ATARS software, because it is an ATARS directed Comm-B message. The second approach requires a more than trivial modification in the Non-Surveillance Report Processing Task. This latter approach can transmit more information, such as runway designation.

The CDTI service designation (SVECT.ACLASS = 3) could be used to branch out from the ATARS modules to implement the CDTI support functions.

In the following paragraphs, two approaches are discussed for modifying the ATARS software to support the CDTI research flight experiments. The first approach (Modification I) is relatively simple and straight forward but does not contain the desired CDTI traffic selection logic. Instead, it depends on the ATARS proximate traffic selection logic. The second approach (Modification II) is more extensive as a result of incorporating the desired CDTI traffic selection logic.

Modification I One of the simplest ways to support the CDTI flight experiments in a rudimentary manner would be as follows:

- a. Change the ATARS coarse screen parameters to approximately satisfy the CDTI coverage volume specification, and
- b. Pack the CDTI ELM Data Format according to the specifications given earlier in Chapter II.

The ATARS coarse screen parameters need to be changed to cover a somewhat larger volume. The current ATARS coverage is limited to a 12.6 nmi radius area and a relative altitude of ± 2000 ft. It may also be advisable to modify the threat monitor parameters to inhibit generating the resolution advisories. This approach was taken to support the FAA CDTI flight experiments.[†] The basic ATARS advisory Data Formats were used with only slight modifications.

The ATARS Advisories are packed in the Data Link Message Construction Task based on the values stored in the (threaded) PWILST lists. Therefore, this routine needs to be modified in order to service the CDTI equipped aircraft. A schematic diagram incorporating the modification is presented as Fig. 27. Test logic is provided which checks for SVECT.ACLASS for each aircraft. If it is less than three (indicating the aircraft is ATARS equipped), then appropriate ATARS Advisories are packed and stored in the DABS uplink buffer. If it is equal to three (indicating that the aircraft is CDTI equipped), then the CDTI information is packed according to ELM Data Format specifications. The initial ELM provides the General Map and Traffic Flight ID Information, and the ELM in the subsequent antenna scan supplies Own and Target data. The packed ELM messages are stored in a buffer for the DABS Comm-C uplink transaction.

Some of the possible implications of these modifications are:

- a. The State Vector corresponding to the target aircraft in the PWILST lists are assumed accessible;
- b. The traffic selection criteria is similar to that of ATARS;
- c. The number of traffic aircraft is limited to eight;
- d. The pilot target selection option may not be easily implemented;

[†] John Fabry, the FAA Technical Center, Atlantic City, NJ

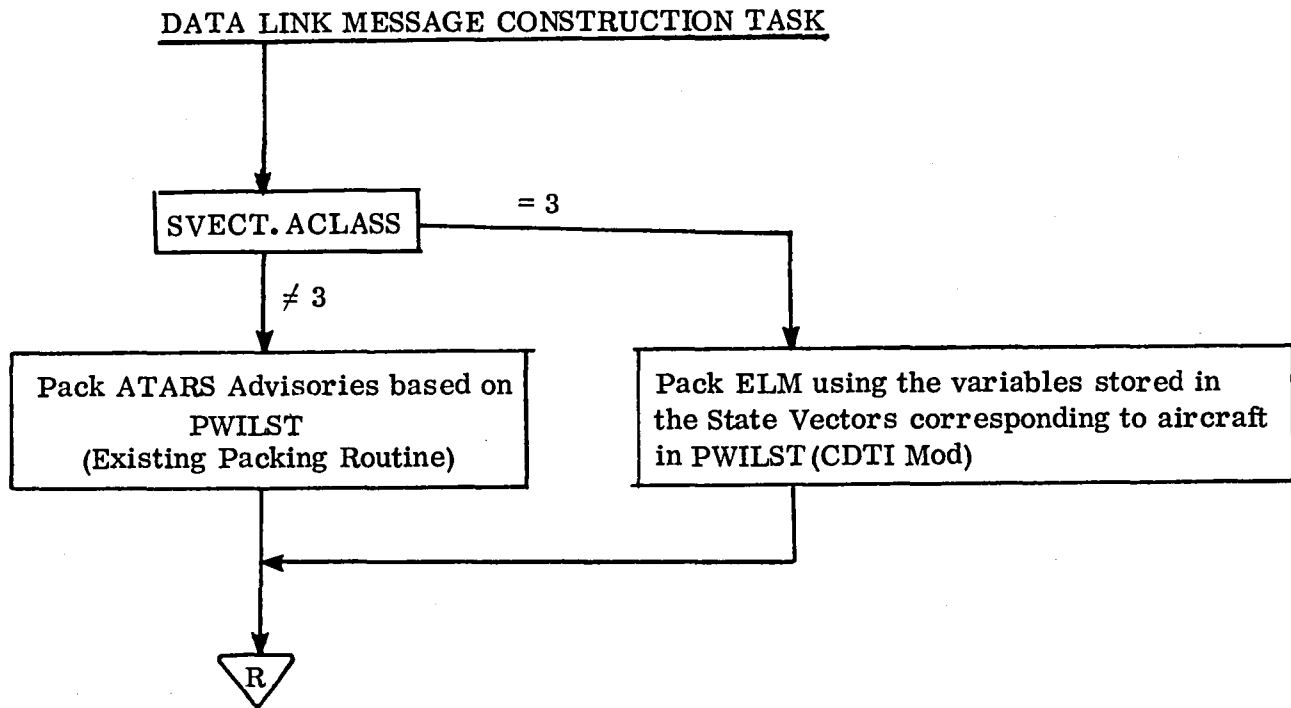


Figure 37. CDTI ELM Data Format Packing

- e. ATARS advisories are lost to the CDTI class; and
- f. CDTI ELM Data Format packing is assumed to require no more real time than the ATARS advisories.

Some remarks are in order:

1. The current advisory packing routine obtains the data from the PWILST, but not from the State Vector. The data stored within the PWILST data structure are in integer form, i.e., they are rounded-up according to the ATARS resolution accuracy requirement. Therefore, higher resolution accuracy and different variables required for CDTI support imply that these values must be accessed directly from the State Vector.
2. Because the ELM uplink is scheduled by the DABS sensor at the beginning of the DABS period on an available time basis, (see Appendix A) it is necessary to complete the ELM packing prior to the SM (Comm-B) packing.
3. The CDTI equipped aircraft may not be able to receive the ATARS advisory due to channel overloading; however, the ATARS advisories (except for Conflict Resolution) can be computed on-board, given the specified CDTI ELM data format.

4. The ATARS proximity volume can be adjusted by means of several coarse screen parameters. The implication of the enlarged search volume, in terms of real time requirements, is not known at this time.
5. The effect of ignoring ATARS advisories by CDTI equipped aircraft is not known. The ATARS software is designed to monitor the status (uplinked, received, confirmed, etc.) of each advisory through the RA Register.

Modification II The next level of CDTI modification involves a major design, development and programming effort. The modification details are made to satisfy the CDTI ground system requirements delineated previously, and they include a CDTI traffic selection criterion, proximity type assignment, and the data format specifications. The CDTI functions are implemented in parallel in such a way that the existing ATARS services are not compromised for the ATARS equipped aircraft.

After the ATARS surveillance and aircraft update functions, a coarse screen filter is invoked to test each (subject) aircraft in the sector list against all other (object) aircraft in the X-list for possible conflict situations. The possible conflict pairs are entered in a list called the Potential Pair List. This is the Coarse Screen Task. Then the Detect Task examines the Potential Pair Lists for a proximity, threat, or resolution situation; it then creates a PWILST list which is threaded for easy retrieval by the Data Link Message Construction Task. The threaded PWILST lists are examined for each aircraft by the Data Link Message Construction Task. If the lists contain more than eight threat pairs, then they are limited to eight by a ranking procedure, and appropriate advisories are packed according to the service class.

From the above summary, a reasonable ATARS subroutine (or Task) for modification is the Coarse Screen Task. The computational flow for the ATARS and CDTI aircraft processing would essentially depart from this routine.

If the remaining ATARS functions are to remain intact, it is important not to skip the ATARS Coarse Screen Task for CDTI equipped aircraft (SVECT.ACLASS = 3). There are two reasons for this: (1) For some aircraft,

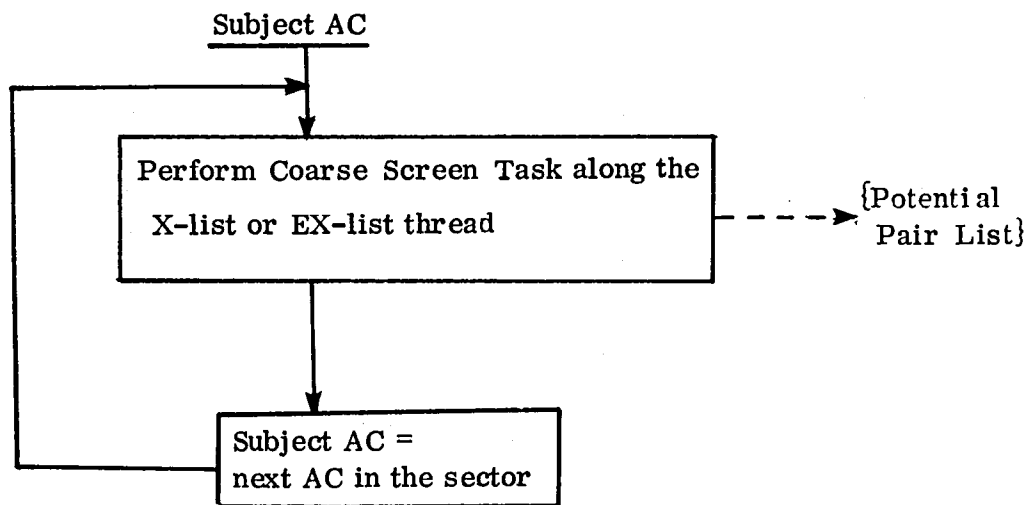
the coarse screen search direction is one way (positive X-list direction); and (2) The subject* aircraft must be included in the Potential Pair List so that the object*aircrafts' PWILST lists contains the subject aircraft. This implies that the CDTI Coarse Screen Task must be implemented in series with the ATARS Coarse Screen Task.

Figure 38 shows the schematic diagrams for the existing and modified Coarse Screen Task. After the ATARS Coarse Screen Task processing is completed, the subject aircraft service class (SVEVCT.ACLASS) is checked. If it is less than three, then the next aircraft in the sector is examined.

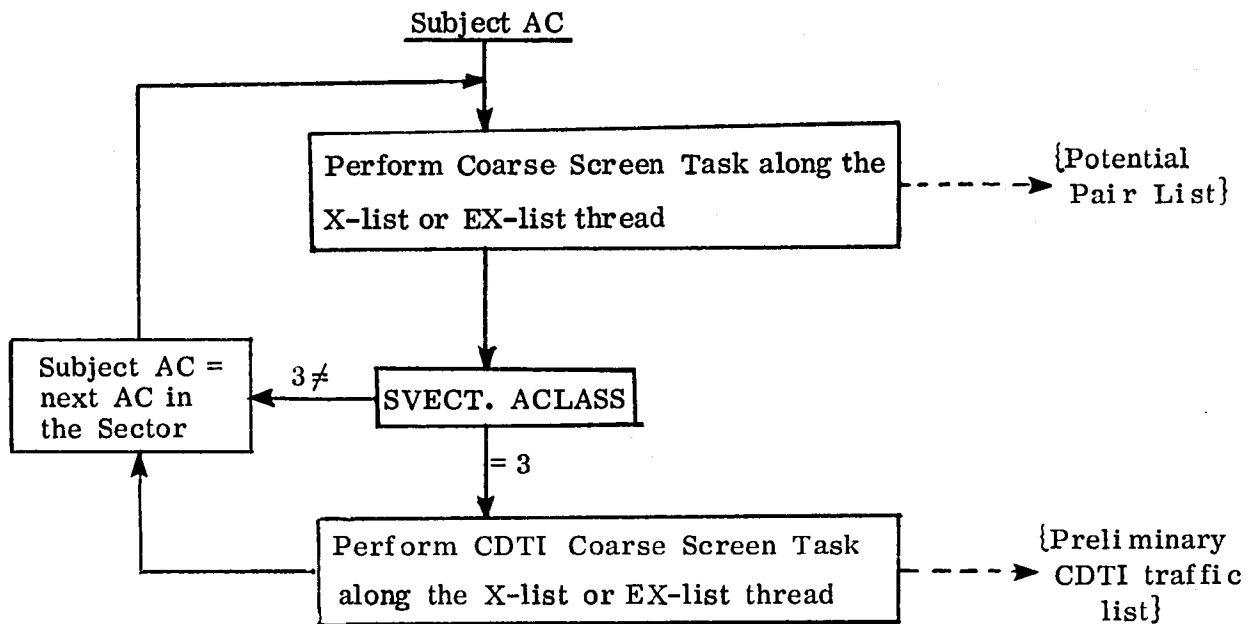
If SVECT.ACLASS equals three, then the subject aircraft undergoes the CDTI Coarse Screen processing. The processing consists of testing the subject aircraft against all other aircraft in the X-list or EX-list for possible inclusion in the CDTI traffic selection process. Possible candidates are included in a list called the Preliminary CDTI Traffic list. Therefore, for each CDTI equipped aircraft there is a corresponding Preliminary CDTI Traffic list. Thus, a pointer needs to be allocated within the State Vector data structure for indicating which Preliminary CDTI Traffic list corresponds to the subject aircraft. The PWILST pointer (SVECT.PWPTR) may be used for this purpose, provided that sufficient care is taken not to change it in the subject ATARS tasks. The CDTI Coarse Screen logic should be kept as simple as possible so that the real time requirement is small. The ATARS Coarse Screen Task and CDTI Coarse Screen Task together must be completed within 2.5 sector periods (approximately 730 msec). Tradeoffs between real time and memory size and between this task and subsequent CDTI tasks must be examined to satisfy the real time requirement.

After both ATARS and CDTI Coarse Screen Tasks are completed, the Executive Program must direct the sector algorithm into a parallel mode. One branch proceeds along the original ATARS sector algorithm conducting the Detect Task, Traffic Advisory Task, etc. The other branch proceeds

* The subject aircraft is a particular aircraft in the sector against which other object aircraft are tested. The object aircraft can belong to other sectors.



(a) Existing ATARS Coarse Screen



(b) With the CDTI modification

Figure 38. ATARS and CDTI Coarse Screen Task

with further CDTI processing. Figure 39 shows a schematic diagram of the parallel processing. Because the only real time requirement for the CDTI processing is that the uplink message be constructed by the fifteenth sector time, the CDTI branch can be processed in background (or dead) time over a 2.05 sec interval. This assumes that the ATARS software is configured for such a computational mode and that the State Vectors can be accessed from the background level.* If such a mode is not available, then the executive must invoke a separate computer for processing the remaining CDTI module software. Functions of the remaining CDTI processing are to implement the transition logic depicted in Fig. 36 and to construct the uplink ELM according to the CDTI specifications given previously.

It may be convenient to organize the remaining CDTI processing into three subroutines as shown in the schematic diagram in Fig. 40. The first subroutine processes the Preliminary CDTI Traffic list. The second subroutine finalizes the selected traffic by updating a list called CDTILST. The third subroutine packs the uplink ELM and stores the message in a buffer which is, in turn, sent to the DABS sensor for the actual uplinking. Each of these subroutines is now discussed from the conceptual point of view.

Preliminary CDTI Traffic List Processing Routine The CDTI Coarse Screen Task creates a Preliminary CDTI Traffic list which contains all the candidate aircraft to be included in the current scan. Due to time considerations, the search procedure used in the Coarse Screen Task is not complete; that is, the list contains aircraft which may not be in the coverage volume. The subroutine, therefore, examines each aircraft in the preliminary list in terms of the CDTI coverage volume. Then it selects up to six new aircraft to be added to the finalized CDTILST list. This list is depicted in Table 12. It is noted that the new aircraft are assigned Proximity Type II automatically. The inputs to this routine are

* It is doubtful that such a mode exists within the existing software structure, because ATARS modules are tagged at critical time windows, i.e., they must be performed in a foreground structure where a strict time control can be maintained.

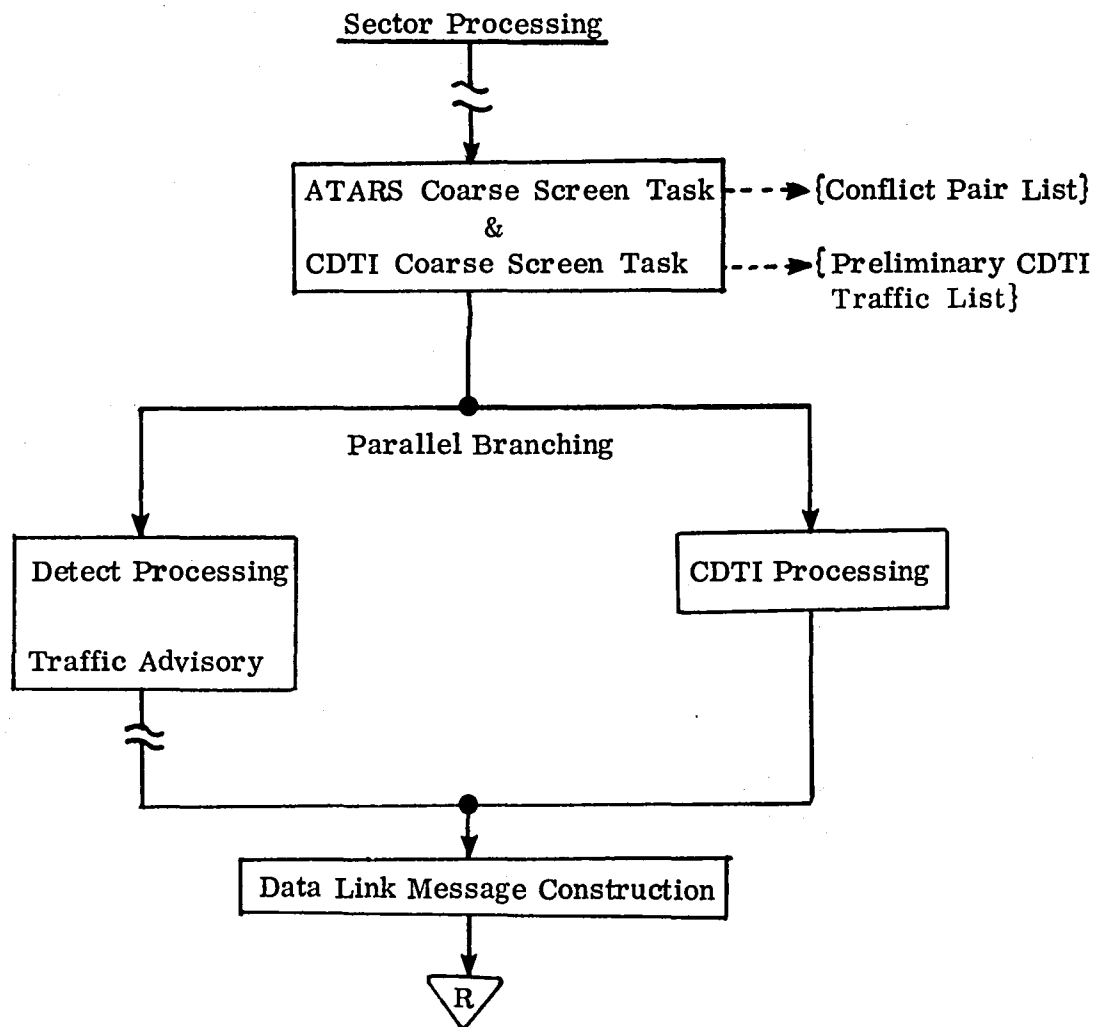


Figure 39. Modified Sector Processing

{Preliminary CDTI Traffic List}, {State Vector}, and {CDTILST}_{OLD}. The output is a list of new aircraft, and it may be stored in the {Preliminary CDTI Traffic List}.

CDTILST Update Routine This routine examines each aircraft contained in the previous {CDTILST}_{OLD} and deletes traffic which are no longer within the ATARS coverage or within the CDTI coverage volume. The new aircraft list, as obtained by the preceding subroutine, is added. Then the Proximity Types are assigned or updated according to a pilot request or a rank test. The new aircraft are assigned track numbers, and the uplink flip-flops are reset for the Proximity Type II aircraft for time multiplexing

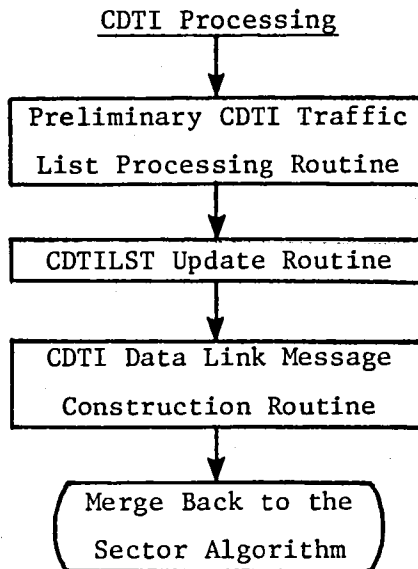


Figure 40. Macro Flow Chart for CDTI Processing

purposes. The subroutine inputs are {Preliminary CDTI Traffic List}, {CDTILST}_{OLD}, and {CDTILST}_{NEW} list which contains all the information necessary for packing the uplinked CDTI ELM.

CDTI Data Link Message Construction Routine The function of this routine is to pack the CDTI ELM according to the Data Format specifications. If the uplink is the initial one, then the General Map and Traffic Flight ID Data Formats are packed. If it is not the initial uplink, then Own Data, Traffic Flight ID (if a new aircraft is present), and Traffic Data are packed within a 16-segment ELM. The packed message is stored in a DABS uplink ELM buffer. The inputs to this routine are {CDTILST}_{NEW} and various map parameters and flight ID numbers. The source for map parameters and flight ID's is not known at this time. The information does not seem to be available within the ATARS data structure.

The above three subroutines must be completed within seven sector periods (approximately 2.06 sec) so that the uplink CDTI ELM buffer is ready prior to the ATARS SM buffer.

An overall subroutine level flow chart, including what was referred to as CDTI Modification II, is shown in Fig. 41. The ATARS modules

TABLE 12

EXAMPLE OF THE CDTILST LIST (ON-BOARD TRACK FILE)

Existing CDTI Traffic	OID	Seq. No.	ACID-I	#PTI	ACID-I	#PTI	ACID-N	#New	OVF	UPLNKG	UPLNKT	Data				
	ACID	Seq. No.	TR#	NAC	PAC	PT	PT ⁻	PR	✓	W	TT	ATC	ATAS	RNK	RNK ⁻	Data
Candidate New	ACID	Seq. No.	TR#	NAC	PAC	PT	PT ⁻	PR	✓	W	TT	ATC	ATAS	RNK	RNK ⁻	Data

21 slots

6 slots

21 slots

6 slots

95

OID = Own aircraft (AC) identity (ID)
 Seq. No. = Landing Sequence Number (0 = fly-over;
 111..1 = not assigned)
 ACID-I = ACID of First PI-I aircraft
 #PTI = No. of PT-I ac (must be less than or
 equal to 3)
 ACID-II = ACID of First PT-II aircraft
 #PTI = No. of PT-II ac (must be less than or
 equal to 30, (#PR=II) = 30 - 2 * (#PTI))
 ACID-N = ACID of first New ac
 #New = Number of New ac
 OVF = Overflow indicator, i.e., No. of new
 ac too large
 UPLNKG = General Information uplink flag
 UPLNKT = Traffic Information uplink flag
 Data = Own aircraft track data
 ACID = Target ac ID
 TR# = Unique track number assigned to this
 ac for the duration

NAC = ACID of next lower ranking ac in the
 thread
 PAC = ACID of next higher ranking ac in the
 thread + or -
 PT = Assigned proximity type this scan } indicates
 PT⁻ = Proximity type last scan } flip-flop
 PR = Pilot request flag
 ✓ = Check mark used by the CDTI-Coarse-
 Screen
 W = Weight class, i.e., heavy, light or small
 TT = Transponder type (Mode - A, C or S)
 ATC = IFR or VFR
 ATAS = ATAS service class
 RNK = Rank assignment variable this scan
 RNK⁻ = Rank assignment variable last scan

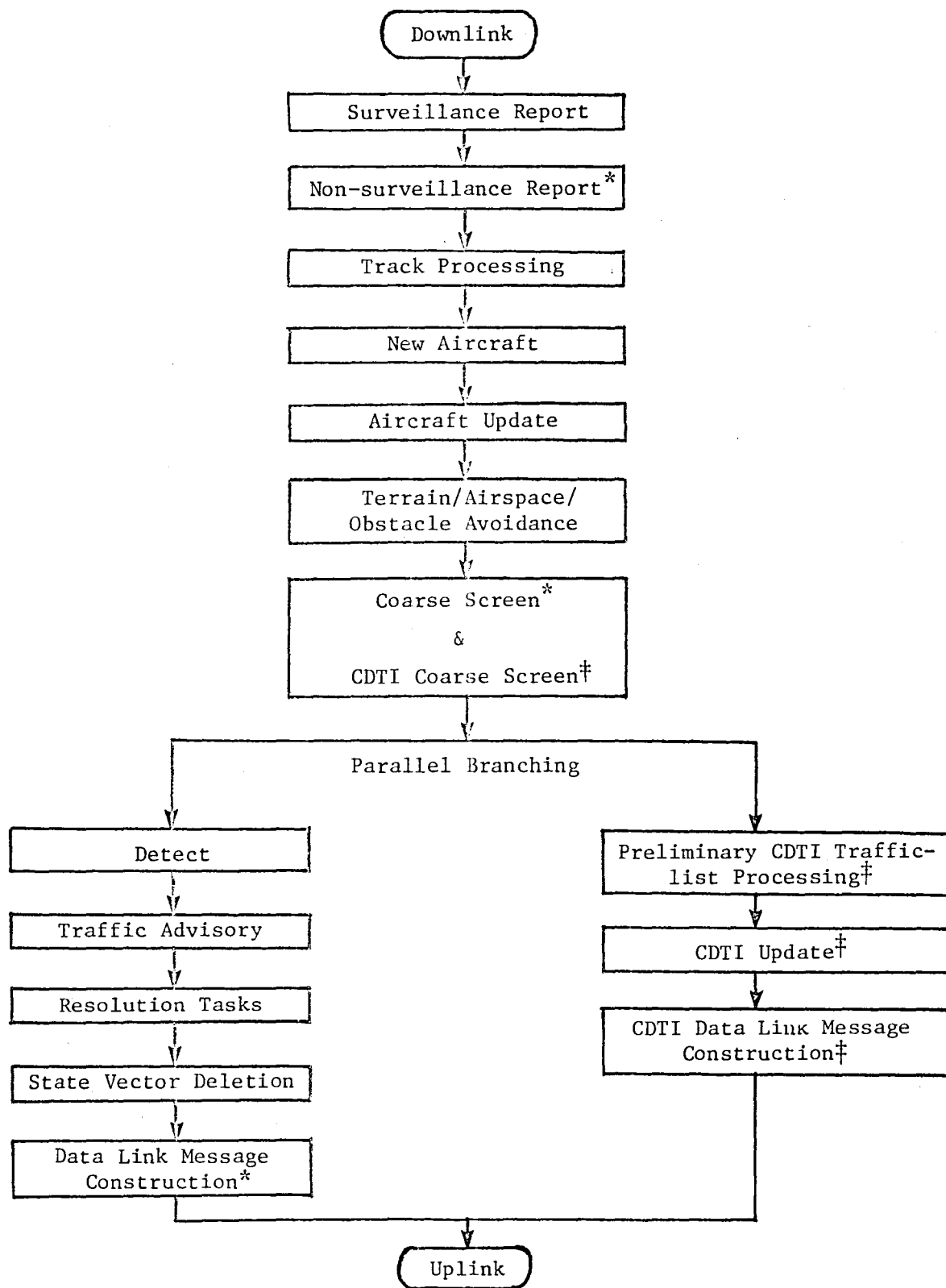


Figure 41. ATARS/CDTI Sector Algorithm

which need modifications are marked with an *. The additional CDTI modules are marked with a ‡. Modifications which have not been discussed previously are indicated. These modifications are regarded to be non-major.

The ATARS Non-Surveillance Report Processing Task needs to be modified to accept the CDTI Pilot Request Comm-B (BDS = 01010000, Type Codes = 001000 and 001001) format. It must be modified to decode the Extended Capability or Avionics Capability subfields. Depending on the designation, this task must also set the service class designator, SVECT.ACLASS, equal to 3.

The ATARS Data Link Message Construction Task needs to be modified in such a way that ATARS Advisories are not uplinked to subject aircraft belonging to the CDTI class. This is to prevent DABS channel overloading.

This Page Intentionally Left Blank

SUMMARY, CONCLUSIONS, AND RECOMMENDATIONS

The objectives of this project have been (a) to provide a generalized functional design of DABS/ATARS based CDTI avionics, (b) to specify software modifications and/or additions to the existing DABS/ATARS ground system, (c) to assess the existing avionics and on-board software of a NASA research aircraft in terms of CDTI applications, and (d) to apply the generalized functional design for supporting CDTI research flight experiments.

In order for DABS/ATARS based CDTI avionics to function properly, requirements and specifications for the avionics and the ground system must be interfaced in a systematic way. In this report, DABS Data Link Formats were first specified for supporting CDTI flight experiments. The resulting set of CDTI Format specifications was then used as a vehicle to coordinate the CDTI avionics and ground system requirements.

Summary

CDTI/DABS DATA Formats - The Data Format used for uplinking ATARS Advisories was found inadequate for supporting CDTI flight experiments because (1) certain variables are not included in the data set, and (2) the resolution accuracy of those variables which are included is coarse. To overcome the bit length constraint of Comm-A protocol, a decision was made to utilize the DABS ELM (Comm-C) protocol for the CDTI uplink transaction. It was assumed that at least one complete ELM pair (one uplink and one downlink) as well as one Comm-A/B pair (for surveillance and communication) were available in each and every antenna beam dwell period of approximately 35 msec.

Pilot request formats for CDTI flight initiation/termination and option selection were specified utilizing the Comm-B protocol. A data format utilizing the Comm-D protocol was specified for downlinking air-derived variables for post-flight analysis purposes.

The CDTI uplink formats include the general map parameters, traffic flight ID information, Own and Traffic data. A combination of these data types is packed within a DABS ELM (up to 16 Comm-C's) and is uplinked during each antenna scan. The variable types and the corresponding resolution accuracies are chosen to satisfy the CDTI display content and symbology requirements. As a corollary, the format allows the airborne computation of various threat parameters.

By incorporating a time multiplexing uplink concept, the proposed uplink format can accommodate up to 21 surrounding aircraft; this number is thought to be adequate for the most stringent applications.

CDTI Avionics - Two CDTI related avionic systems were examined. One was the Airborne Intelligent Display System (AIDS) developed by MIT/Lincoln Laboratory to support the ATARS Advisory display. The other, the TCV/CDTI system, was developed by NASA Langley to support CDTI flight experiments using tape recorded simulated traffic. An essential shortcoming of the AIDS system is that it is designed to support ATARS functions but not other functions deemed necessary for CDTI research. The counterpoint for the TCV/CDTI system is that it is not based on an actual traffic sensor. Nevertheless, the ideas and concepts developed for both systems can be applied to a generalized CDTI avionics design.

Two existing digital avionics systems were examined from the point of view of "retrofitting" CDTI avionics components. The current TCV avionics represents a class whose computational, pilot interface I/O function and display capabilities are flexible. The T-39 avionics represents a class whose capabilities are somewhat limited. A "series retrofit" configuration with on-board computer software modification is appropriate for the TCV class. In this case, the additional hardware components are minimal. A "parallel retrofit" configuration with a separate computer installation is appropriate for the T-39 class. The additional hardware components are maximal for this configuration.

Functional requirements for the CDTI hardware components and software modules are identified and described to the extent known within the

scope of this project. The proposed avionics architectures are configured so that the commonality of system elements are maintained as much as possible.

The CDTI software modules are specified to the subroutine level. A scheme to compensate the ground based variables by the air-derived navigation variables is given. Additionally, formulae for computing various threat parameters are derived. Therefore, except for the (Conflict) Resolution Advisory, the ATARS functions can be duplicated in the air.

A rough estimate of required memory was made assuming that all the CDTI software modules need to be implemented.

DABS/ATARS Based Ground System - The ATARS software was examined using available documents from the point of view of supporting the CDTI flight experiments. Based on the CDTI Data Formats and avionics, the functional requirements for the ground system are given including traffic selection and proximity type assignment.

Essentially two modification approaches are discussed. The modifications are based on extending the current ATARS service class to include a CDTI equipped aircraft. One approach represents a relatively simple modification where the majority of ATARS modules are kept intact. This is similar to what was done in order to support the FAA/CDTI flight test experiments. The only conceptual departure is that the DABS ELM (rather than SM) protocol is utilized to uplink the expanded CDTI traffic data.

The other approach requires more extensive modifications and additions. The proposed modifications or additions are discussed at the subroutine level; further development of the details is beyond the scope of this project.

Conclusions and Recommendations

The main product of this report is a set of functional requirements and specifications for a CDTI system based on the DABS/ATARS ground system

as the traffic sensor. The system is partitioned into three logical subsystems - the DABS/ATARS ground system, the CDTI airborne system, and the Mode S communication link between the ground and airborne systems. The requirements and specifications for each system are defined so that the whole (integrated) system would be capable of supporting the most extensive CDTI flight Research Experiments.

All the necessary system elements are discussed from the functional point of view; therefore, this document provides the preliminary design for an integrated CDTI system. However, due to the complexity of the system and the limitation in scope of this project, there remain many additional details to be developed prior to construction of a satisfactory flight test system.

From the system design view-point, there are four fundamental CDTI research questions, some of which are being resolved currently. These are all inter-related. Table 13 lists the problem areas and the associated issues. The main difficulty is the information transfer limitations caused by the uplink capacity limits, the possible large number of target aircraft, and the data quantization accuracy restrictions. Further developmental efforts are discussed for each subsystem.

Data Link It was assumed that the ELM protocol would be available every antenna scan period for the traffic information uplink. The tacit assumption is that the ATARS software can interface with the DABS uplink ELM buffer. This capability does not exist currently. This assumption may not be realized in a real, dense operational environment for two reasons. One is that the (quoted) capability of at least one ELM per scan is not a guaranteed fact. The other is that the transponder can receive at most one complete ELM per scan, which precludes other ATC services from using the ELM channel.

The above implies that: (a) the transponder and/or ground ELM processors can be redesigned; (b) the CDTI uplink is available intermittently (not continuously); or (c) either the SM protocol alone or a combination of SM and ELM protocols are utilized. It would be necessary to resolve the

TABLE 13

CDTI SYSTEM RESEARCH AREAS

-
-
- Traffic Selection Procedure
 - Pilot requirements for CDTI applications;
 - Uplink limitations (SM vs. ELM);
 - Threat, coverage volume, or function based;
 - CDTI modes: active vs. passive.

 - Accuracy Requirements
 - Surveillance accuracy;
 - Data quantization accuracy;
 - Uplink limitation;
 - Ground tracker algorithm and/or airborne smoothing and compensation algorithm.

 - Update Frequency
 - Continuous (every scan) or intermitten uplink;
 - Uplink limitation (SM vs. ELM).

 - Display Design
 - Pilot requirements;
 - Symbology development;
 - Self-contained vs. integration with MFD (map display);
 - CAS parameters.
-
-

ELM capability and limitation of the operating ground system and transponder combination. Such an experiment is planned in a near future by the FAA with respect to uplinking a digitized weather radar map.

CDTI Avionics There are several items which require further development. These are:

- (a) Exact input/output functions (interrupts and control line specifications) among the transponder, TIU and CIU (or the airborne computer);
- (b) Exact pilot-to-system interface procedures including the CDTI menu page for the NCDU;
- (c) The detailed on-board software modules including the track-file organization, airborne compensation and smoothing algorithms, and CAS parameter computations; and
- (d) Development of the CDTI display symbology including the CAS parameters.

These are outlined at the functional and conceptual level in Chapter III and Appendix B. These over-all component and functional descriptions will remain valid so that only the details need to be provided by further effort.

DABS/ATARS Ground System Detailed subroutines and modules need to be developed following the proposed modifications of Chapter IV. The major task here is the development of a reasonable traffic selection procedure. The real time requirement of each module must be carefully assessed so that the entire sector process timing is not affected. Additionally, the ELM interface capability between the ATARS software and the DABS sensor needs to be added.

Simulation Capability In order to facilitate further development of a DABS/ATARS based CDTI flight system, it may be advisable to build a simulation facility. It must include rudimentary elements of the expected flight system. They include a realistic traffic generator, pertinent DABS/ATARS software modules, ATC functions, a Data Link element, and a piloted aircraft simulator. For this purpose, it may be advisable to utilize the NASA Langley MOTAM simulation facility with an active piloted simulator properly interfaced. A schematic diagram of this facility is shown in Fig. 42.

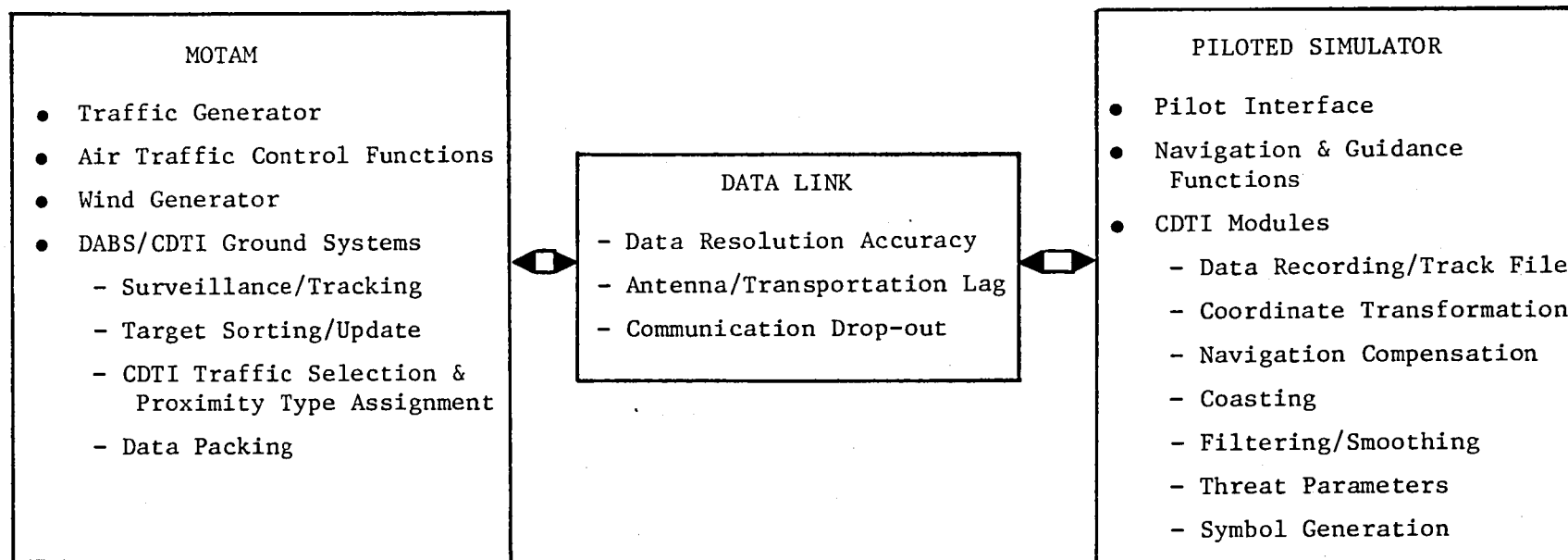


Figure 42. DABS/CDTI Software Design Validation Facility

Referring to Fig. 42, the MOTAM facility provides realistic near terminal traffic generation, Air Traffic Control functions, wind generation and DABS/CDTI ground system modules. The ground system consists of a surveillance/tracking function with projected DABS error characteristics, a target sorting/update algorithm, CDTI traffic selection and Proximity Type assignment, and data packing. The Data Link element consists of data resolution accuracy, antenna/transport lag and communication dropout characteristics. The piloted simulator element consists of existing simulator implements including a navigation algorithm with typical navaid errors plus CDTI modules. The CDTI modules include the data decoding/track file update function, coordinate transformation from DABS to navigation reference, Own navigation variable compensation, a coasting algorithm in case of data dropout, an airborne filtering/smoothing algorithm for target data depending on applications, threat parameter computations, and CDTI display symbol generation.

The design task for each software submodule may be pursued through use of a small scale simulation, possibly utilizing recorded traffic data. This will yield an added design tool for the researcher because his objective requires use of specialized software. However, a full scale MOTAM/XTCV/CDTI simulation facility offers several advantages:

- (a) It can be used as a final validation facility for a CDTI software design prior to actual implementation;
- (b) It can be used as a more realistic pilot training station;
- (c) It can check the potential validity of a particular flight experiment; and
- (d) It can be used to study the utility of the CDTI concept against a future ground-based ATC automation concept.

Item (a) implies that actual implementation of the airborne and ground software should be delayed as much as possible. The final design should be obtained elsewhere before the DABS/ATARS software modifications/additions are attempted. Item (c) implies that if a particular CDTI application concept is not acceptable to a pilot in the simulation environment, then it is also not likely to be acceptable in flight. Therefore, the facility can be used to filter CDTI application concepts to those

acceptable to pilots without flight test. Item (d) is self-explanatory. If ground-based ATC automation becomes a reality, it is doubtful that pilots would fly into the terminal area without a CDTI to monitor the overall operation. In such a situation, a CDTI would function as a Collision Avoidance System in addition to providing navigation, guidance and traffic situation monitoring.

Recent Developments Since the onset of this effort (Sept. 1981), there has occurred two major modifications to the FAA ATARS and BCAS (Beacon based Collision Avoidance System) programs. In place of these two systems, concepts called, ATAS and TCAS are being developed.[†] Because these two concepts relate to CDTI mechanization and the effort documented in this report, it is appropriate to comment on these systems.

The proposed Automatic Traffic Advisory Service (ATAS) is a reduced form of ATARS. Four major ATARS elements are deleted to form ATAS. These are the conflict resolution logic, the coordination logic between the ground system and the airborne collision avoidance system, the coordination logic among the neighboring sites, and the Resolution Advisory Register logic.

The ATAS advisories are planned to be uplinked by utilizing the SM protocol (no ELM provision). Other ATARS functions such as the tracker algorithm, traffic file sorting and management, and the coarse screen filter are presumably left intact. A draft report on the ATAS logic is planned by July, 1982.

Because of the reduced functions, the ATAS software requires much less computations. Only seven ground computers are needed compared to twelve for ATARS. The real time requirement may also be reduced. The ATAS software may be a better medium for implementing the required CDTI modifications. The possibility needs to be researched further when the documentation becomes available.

There are essentially three Threat Alert and Collision Avoidance Systems (TCAS) being developed by the FAA. These are the TCAS I, "minimum"

[†] National Airspace System Plan, FAA Document, December 1981

TCAS II, and "enhanced" TCAS II. The TCAS I may be a very low powered active device with a limited surveillance capability. The surveillance (and/or resolution) data are provided by a TCAS II equipped aircraft or possibly by the ATAS. Therefore, TCAS I does not seem to be suitable as a CDTI Traffic sensor.

The minimum TCAS II, which is basically the active BCAS developed by MIT Lincoln Laboratory, provides surveillance data in the range and altitude axes but not in the bearing axis. Therefore, its applicability as a CDTI traffic sensor is very limited.

Enhanced TCAS II provides a bearing measurement in addition to the range and altitude data. Its design is based on the so-called full BCAS concept; however, the passive mode, utilizing the ground surveillance interrogations and the transponder replies, is deleted. In order to provide the bearing measurement and to reduce fruits and synchronous garble, it employs a scanning beam interrogation process via a directional antenna and several levels of whisper/shout.

The range measurement error of this device is reported to be approximately ± 100 ft (1 σ). The bearing error is ± 8 deg (1 σ) for a 45-90 deg sector width system and ± 1 deg (1 σ) for a 22.5 deg sector width system. These accuracies have yet to be confirmed by actual data. The altitude error is essentially dominated by the 100 ft encoder quantization. The surveillance radius is less than 25 nmi. The less accurate system can provide a PWI type or an AIDS type horizontal display. Most probably such a system would not incorporate horizontal collision avoidance logic. As in the minimum TCAS II, the collision avoidance logic depends on the estimated range and range rate, and the resolution maneuver would be carried out in the vertical axis. The angular accuracy may be sufficient to provide an "o'clock bearing" of the intruder, so that the VFR application is enhanced. The more accurate system may provide a full range of CDTI applications. The current plan calls for the vertical collision avoidance logic (Model A) and for the horizontal logic (Model B) at a later date.

Both of these enhanced TCAS II devices may be applied as a CDTI traffic sensor. The operating characteristics need to be studied.

These include the surveillance capability, the maximum number of target aircraft in the TCAS internal file, and descriptions of the input/output interface with external devices.

Close to the DABS sensor, the DABS/ATAS ground system would provide more accurate surveillance data than the enhanced TCAS II. However, the system implementation for DABS/ATAS would be more complex. Furthermore, the TCAS would be functional regardless of the ground system (provided that its operation in a high density area is proven).

This Page Intentionally Left Blank

APPENDIX A

BRIEF DESCRIPTION

OF THE

DISCRETE ADDRESS BEACON SYSTEM (DABS)

Introduction

This appendix presents brief descriptions of the DABS sensor, its signal-in-space characteristics and the DABS messages. Also included are descriptions of the projected ATC services supported by the DABS data link. The pertinent operating characteristics of the DABS (or Mode S) transponder are given in the main text. The material presented here is based on Refs. 6-11.

The Federal Aviation Administration (FAA) is developing the Discrete Address Beacon System (DABS), in response to known deficiencies in the quality of surveillance obtainable from the existing Air Traffic Control Radar Beacon System (ATCRBS). DABS is a combined surveillance and ground-air-ground data link system capable of providing the aircraft surveillance and communications necessary to support ATC automation in the dense traffic environments expected in the future.

DABS is comprised of the sensors, transponders, and signals-in-space which form the link between them. DABS uses the same interrogation and reply frequencies as ATCRBS, and the signal formats have been chosen to permit substantial commonality in hardware. This degree of compatibility permits common-channel inter-operation with the current ATCRBS. Thus, DABS interrogators (sensors) will provide surveillance of ATCRBS-equipped aircraft, and DABS transponders will reply to ATCRBS interrogators.

An innovative feature of DABS is the DABS Address. Each DABS aircraft transponder will be permanently assigned a unique 24-bit DABS Address code, called its Discrete Address. This unique code provides a

fundamental difference between DABS and ATCRBS in the manner of addressing aircraft, or selecting which aircraft will respond to an interrogation.

In ATCRBS, the selection is spatial, i.e., aircraft within the main-beam of the interrogator respond. As the sensor beam sweeps around, all angles are interrogated, and all aircraft within line-of-sight of the antenna respond. In DABS, selection of which aircraft is to respond to an interrogation is accomplished by including the aircraft's address code in the interrogation. Each such interrogation is thus directed to a particular aircraft.

Two major advantages result from the use of the discrete address for surveillance. First, an interrogator is now able to limit its interrogation to only those targets for which it has surveillance responsibility, rather than to interrogate all targets continuously within line-of-sight. This prevents surveillance system saturation caused by all transponders responding to all interrogators within line-of-sight. Secondly, appropriate timing of interrogations ensures that the responses from aircraft do not overlap. This eliminates the mutual interference which is caused by the overlapping of replies from closely spaced aircraft (so-called synchronous garble).

In addition to the improved surveillance capability, the use of the discrete address in interrogations and replies permits the inclusion of messages to or from a particular aircraft. This provides a ground-air and air-ground digital data link to individual tracked aircraft.

The DABS Sensor

Sensor Types The basic types of DABS sensors are the Terminal DABS and the Enroute DABS. Although alike in hardware and software, the DABS sites will differ in antenna configurations, scan periods and radar digitizer interfaces.

The Terminal DABS sensors will have a single face antenna (beacon and search antennas aligned in azimuth) rotating at the rate of 12-15 rpm (4-5 second scan period). The range of these sensors will be nominally 60 - 70 nmi.

The Enroute DABS sensor will use a back-to-back antenna. The front face will consist of a beacon and search antenna; the back face, which will be aligned 180° to the front face, will consist of a beacon antenna only. The back-to-back antenna will rotate at a rate of 6 rpm. This will result in a scan period of 5 seconds for beacon data and 10 seconds for search data. The range of an Enroute DABS sensor, with its slower antenna rotational rate, extends to 200 nmi.

The following discussion is mostly concerned with the terminal sensors.

System Architecture To perform its functions, DABS incorporates a distributed computer architecture involving over 30 computers, several global memory devices, global and ensemble data buses, and memory couplers. (See Figure A.1.)

DABS computers are grouped into ensembles with up to four computers in each ensemble. These computers are connected to an ensemble data bus through which they communicate with the rest of the system. Each DABS computer consists of two central processors, voting logic for the central processors, and 8,192 bits of local memory.

The application of redundancy at the module level supports the high reliability requirements of DABS. Common backup (as standby units) is provided on-line for each module type such that failure/recovery can be accomplished at the local level without major perturbation to the remainder of the system. All communication between computers is through global memory such that each computer with its tasks becomes an independent subsystem. If a computer fails (as indicated by the voting logic), its tasks can be switched automatically to another computer with minimum interference with the rest of the system.

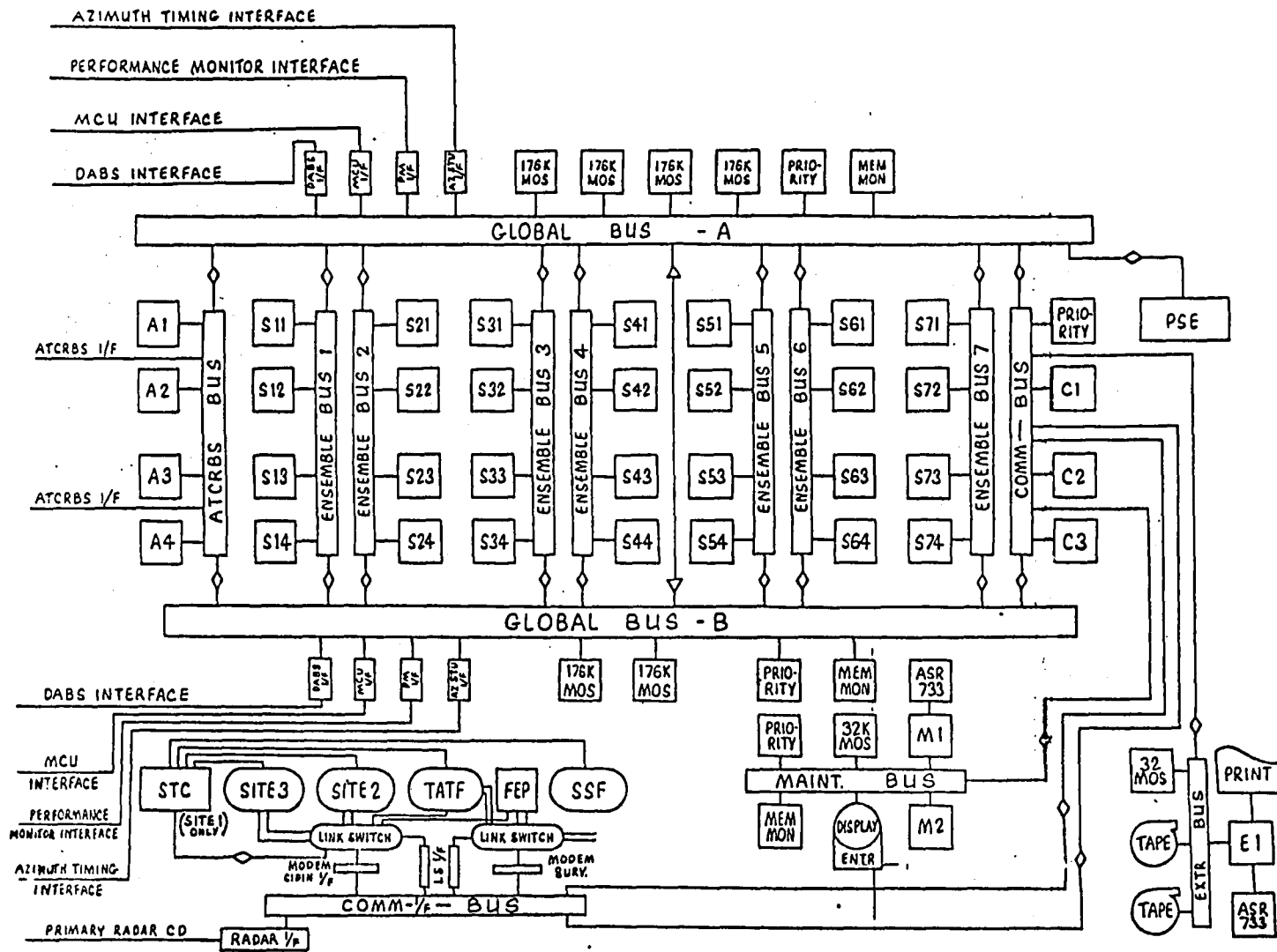


Figure A.1. DABS Computer Architectural Block Diagram

Functional Architecture The functional architecture of the sensor is illustrated in Fig. A.2. Most sensor functions are conveniently categorized according to the time scale on which they operate, as follows:

- (a) Those which involve the generation and processing of signals, and therefore operate on a microsecond time scale:
 - 1. modulator/transmitter;
 - 2. multichannel receiver; and
 - 3. DABS and ATCRBS reply processors.
- (b) Those which involve channel transactions, and operate at a millisecond time scale commensurate with the dwell time of the interrogator antenna on a target:
 - 1. Channel management; and
 - 2. ATCRBS reply correlation.
- (c) Those which are paced by the antenna scan time, and therefore operate on a time scale the order of a second:
 - 1. Surveillance processing;
 - 2. Data-link processing;
 - 3. Network management;
 - 4. Performance monitoring; and
 - 5. ATARS.

The transmitter-modulator control generates all waveforms and RF signals in the ATCRBS and DABS modes for transmission through the antenna. The multi-channel receiver provides the path from the antenna to the processors for the DABS and ATCRBS aircraft replies.

The channel management function determines the nature and timing of each event taking place on the RF channel. Channel management controls both the ATCRBS and DABS activities of the sensor in accordance with a site adaptable input table.

The ATCRBS processor accepts video inputs from the receiver and provides ATCRBS target replies. These target replies consist of range, azimuth, one of 4,096 beacon identity codes, altitude codes, ATCRBS con-

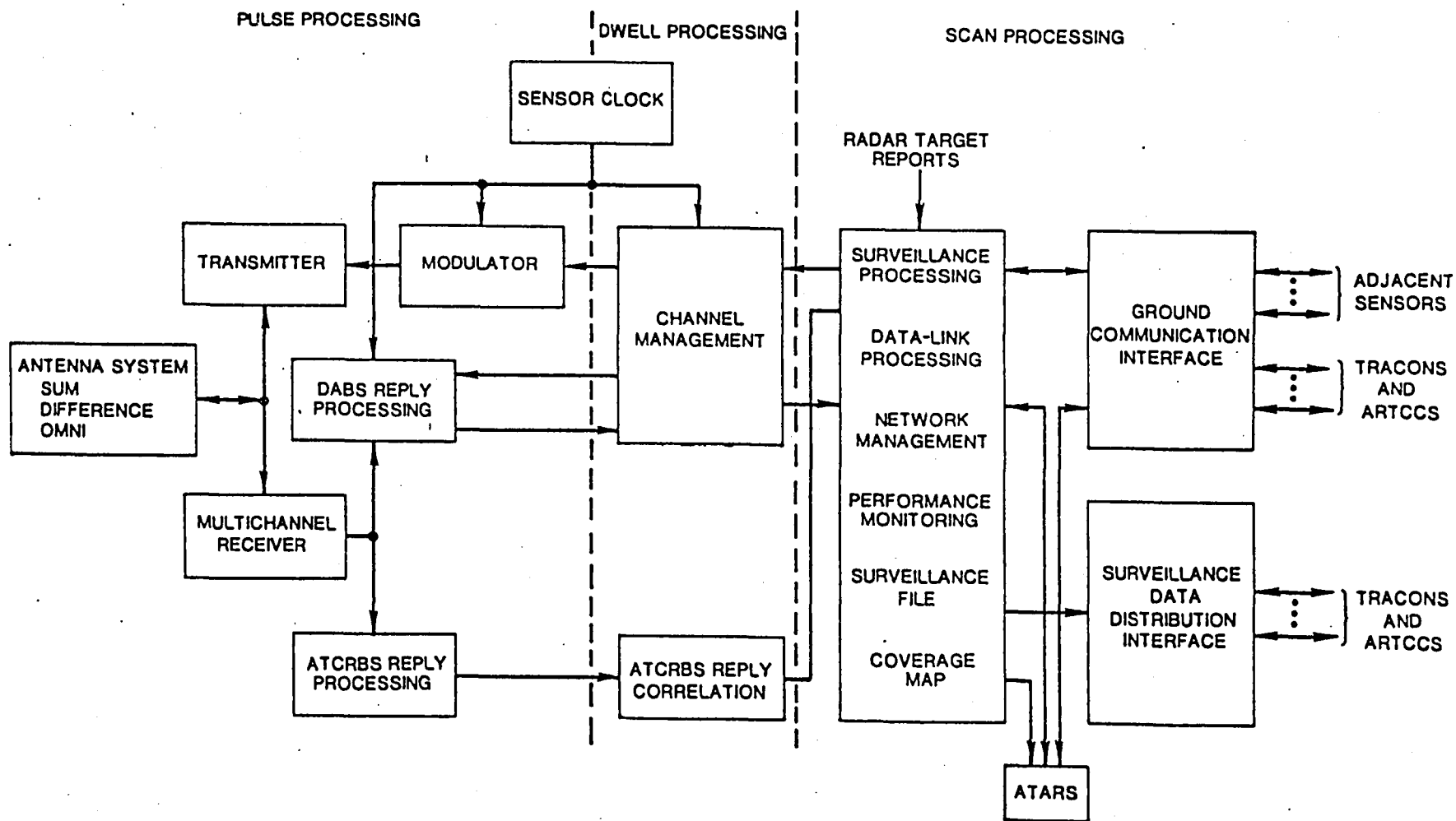


Figure A.2. DABS Sensor Functional Block Diagram [7]

fidence, monopulse average, and time. The ATCRBS reply-to-reply correlation function is performed by a software algorithm and outputs target reports.

DABS target reports consist of an estimate for range and azimuth, and the information bits that have been transmitted as part of the reply. Using error flags and error-correcting codes, the DABS processor will give an indication whenever a reply has been received unsatisfactorily. The unsatisfactory reply condition is relayed to the channel management function so that another DABS interrogation can be scheduled for that particular aircraft during the same beam dwell.

The surveillance file processing function maintains target files on all ATCRBS and DABS aircraft within the sensor's coverage volume. Its principal functions are to:

1. Predict next-scan position of DABS aircraft for interrogation scheduling.
2. Edit and correct ATCRBS target reports based upon data from previous scans.
3. Perform track initiation.
4. Accomplish target-to-track correlation.
5. Perform radar/beacon correlation of target reports from a co-located radar.
6. Disseminate composite ATCRBS/DABS radar surveillance data to ATC users.

The surveillance processing function performs DABS and ATCRBS scan-to-scan correlation. Beacon reports are correlated with digitized primary radar reports. These reports are transmitted to ATC facilities as "radar reinforced" beacon reports. Radar substitution reports, in beacon format, are transmitted to ATC for those radar reports correlating with beacon tracks. Radar reports which do not correlate with either a beacon report or beacon track are classified as "radar only" reports. Radar only scan-to-scan correlation is also performed by surveillance processing depending upon the type of primary radar digitizer interfaced with DABS. Scan-to-scan radar correlation is performed for the moving target detector (MTD) and sensor receiver and processor-1 (SRAP-1), although not for the common

digitizer (CD). Uncorrelated CD radar reports are transmitted to ATC facilities as uncorrelated radar only reports.

The data link processor provides the message link between the ATC facilities and the sensor. Downlink messages are passed through the data link processor and forwarded to the designated ground-based user. Uplink messages, sent from the ATC facilities, are formatted and listed in the data link file with their priority and appropriate tags to indicate message type.

Scheduling and Time Frame Management The DABS sensor employs a rotating directional beam. All surveillance or communication transactions with an aircraft must therefore be performed within the time that the beam is passing over the aircraft. This time is called the beam dwell and equals the ratio of the beam width to the angular velocity, this is typically 25-45 msec for a terminal sensor.

To service both DABS and ATCRBS transponders, the sensor operates in two modes: ATCRBS mode and DABS mode. The RF channel is time-shared (over a dwell period) between the ATCRBS/DABS all-call modes and the DABS mode. That is, a dwell time is subdivided into several ATCRBS and DABS periods as depicted by the top sketch in Fig. A.3. The ATCRBS period is used for ATCRBS surveillance interrogations and also for acquiring new DABS equipped aircraft which are not yet on the sensor list. The latter is accomplished through a special DABS all-call interrogation. The ATCRBS interrogations are scheduled at the beginning of an ATCRBS period whose total duration is determined by the number of ATCRBS aircraft present and the required "listening time".

The DABS period is used exclusively for transacting with the DABS transponders. Scheduling of DABS interrogations and replies operates under the following principal ground rules:

- (a) DABS interrogations are addressed only to aircraft within the antenna beam.
- (b) Channel time is allocated to each DABS interrogation and reply based upon a prediction of aircraft range.

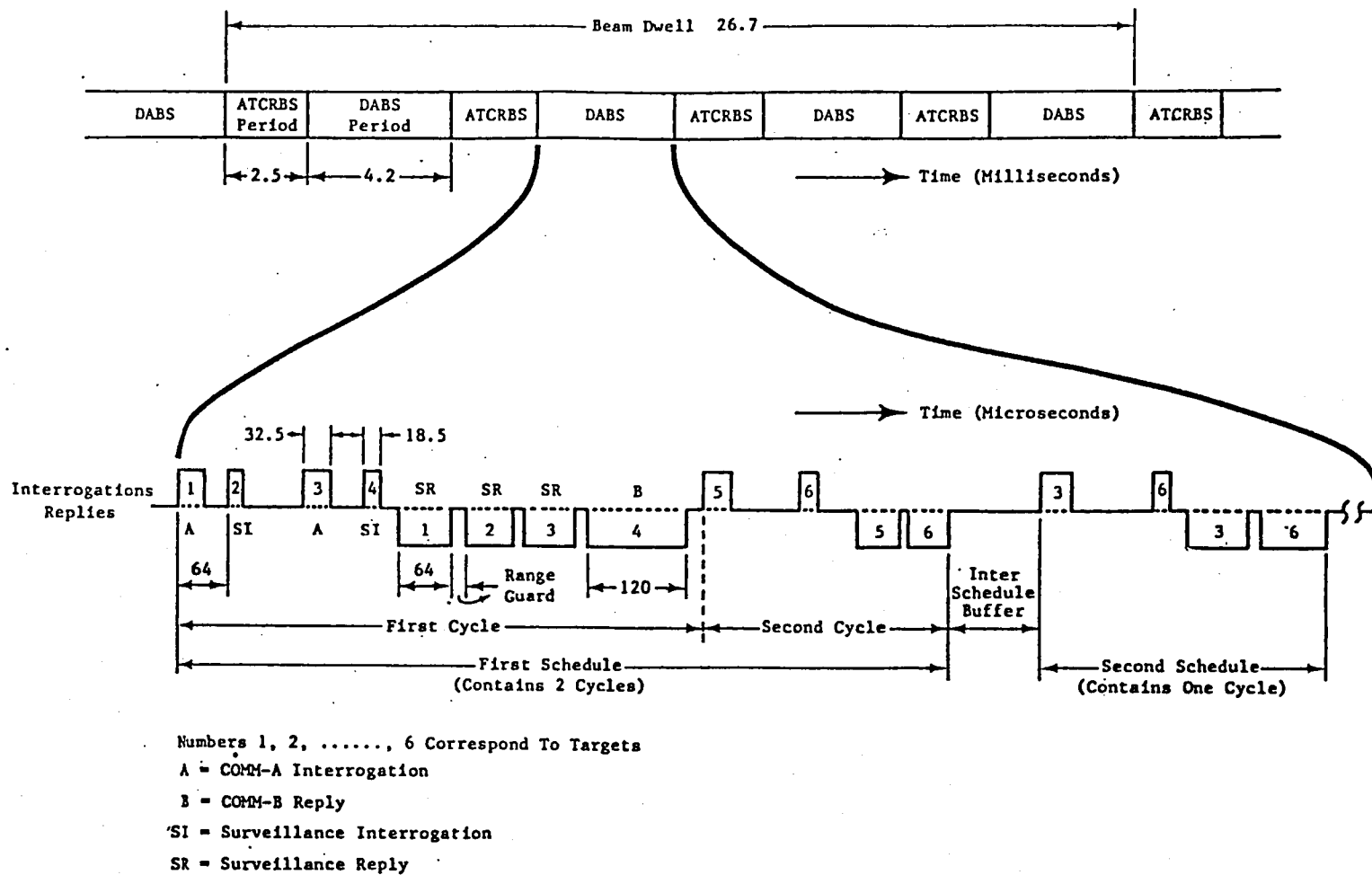


Figure A.3. Time Line Depiction of DABS Scheduling [9]

- (c) DABS surveillance and data-link procedures may require more than one interrogation to each aircraft. Therefore, the sensor is able to reinterrogate an aircraft while it remains in the beam.

The sensor maintains an active target list, comprising those DABS targets that are within the antenna beam, and makes repeated passes through this list, scheduling discretely-addressed DABS interrogations and replies on a nonconflicting basis. This is referred to as the DABS roll-call.

The roll-call scheduling begins with the first (longest range) target on the list, scheduling an interrogation at the assigned start time of the schedule. Next, the expected reply arrival time is computed, and a suitable listening period is provided. Subsequent targets are scheduled by placing their reply listening periods in sequence and computing the corresponding interrogation times. A cycle is completed when the next interrogation, if so scheduled, would overlap the first reply. This interrogation is deferred to start a new cycle. This can be explained more easily by using an example.

Figure A.3 shows an example of DABS scheduling over one DABS period. The furthest aircraft in the active target list (#1 in the figure) is scheduled first. In this case a Comm-A* is scheduled on the uplink eliciting a surveillance reply (RDLY + TDLY) microseconds later. Here, RDLY is the total (round trip) propagation delay which is known to DABS since the target range is known, and TDLY is a fixed transponder delay. (New targets unknown to the sensors are acquired in the DABS all-call mode. The acquisition provides the sensor with the aircraft discrete address and position. The position of a known aircraft is updated each scan.) An interrogation and reply pair thus scheduled forms a transaction. The next transaction is scheduled for target #2 which is next in the range order. The interrogation is so scheduled that the reply to it will arrive immediately after the first reply, separated only by a range-guard to provide for the uncertainty in range due to the movement of the aircraft. Targets with smaller ranges are scheduled successively

* Comm-A, Comm-D etc are explained in a later section of this appendix.

in a similar manner until the next interrogation, if scheduled, would overlap the first reply. This group of interrogations and replies comprises a DABS cycle.

A next cycle of transactions similar to the first one is then created and the process continued until all targets are exhausted. Such a set of transactions is called a schedule. In Figure A.3, the first DABS schedule contains two cycles of four and two transactions respectively, meaning there are six DABS aircraft in the beam dwell.

A target is scheduled only once per schedule, since the current message must be successfully delivered before delivery of the next message is attempted for the same aircraft. If a message is not delivered in the current schedule (known by the absence of a valid reply) it simply gets repeated in the next schedule. If, after completing one schedule there are more messages to be delivered (whether new ones or repeats), another schedule is formed. Consecutive schedules are separated by an interschedule buffer. More schedules are formed until all messages are delivered or until the available time for DABS scheduling in this period is exhausted.

As explained in a later section, DABS provides a pure data communication link function devoid of the surveillance function. This is called the Extended Length Message (ELM) capability. A ground-to-air ELM is composed of up to sixteen so-called Comm-C segments, which individually do not require downlink acknowledgement. (Such is the case for a surveillance interrogation). An uplink ELM transaction consists of a series of (uplink) Comm-C segments followed by two (downlink) Comm-D segments for acknowledgement. Therefore, an ELM transaction requires special scheduling consideration.

Except for the final Comm-C/Comm-D transaction, all other Comm-C's are scheduled in one or more "precursors" to the main schedules formed from the standard Comm-A/Comm-B/surveillance transactions. They are scheduled within the time left over from scheduling the higher priority

Comm-A/Comm-B and surveillance transactions, although they are inserted at the beginning of the period rather than at the end. Figure A.4 shows an example of scheduling an ELM to an aircraft. In this example the non-final Comm-C segments to the aircraft are shown scheduled in two precursors in two different DABS periods. The final Comm-C/Comm-D is shown scheduled in a following period and is part of a standard schedule. Depending upon the time available, the eligible periods may even belong to different scan times.

Sensor Capacity The capacity of a DABS sensor is most generally defined as the number of aircraft to which a sensor can provide surveillance and data-link service. With this broad definition, capacity depends not only on the sensor operating characteristics, but also on the number of interrogations needed for each aircraft and the azimuth distribution (or bunching) of aircraft in range of the sensor.

The design specification requires the sensor to be capable of processing the following:

1. A total of 400 aircraft.
2. A minimum short term peak capacity of 12 aircraft in a 1.0° azimuth wedge for up to four continuous wedges.
3. A peak capacity of 50 aircraft uniformly distributed in an 11.25° sector for not more than eight consecutive sectors.

A numerical estimate of the DABS sensor capacity in a typical operating environment was obtained based on analysis and simulation of the DABS interrogation scheduling algorithm. The estimate is given by the following formula:

$$n \approx 18.5 \left[T - 360 \frac{N_a}{\theta} (2R/c + t_a) \right]$$

where,

- n = number of transactions per degree,
- R = operating range,
- T = interrogator antenna scan period,
- θ = interrogator antenna beamwidth,

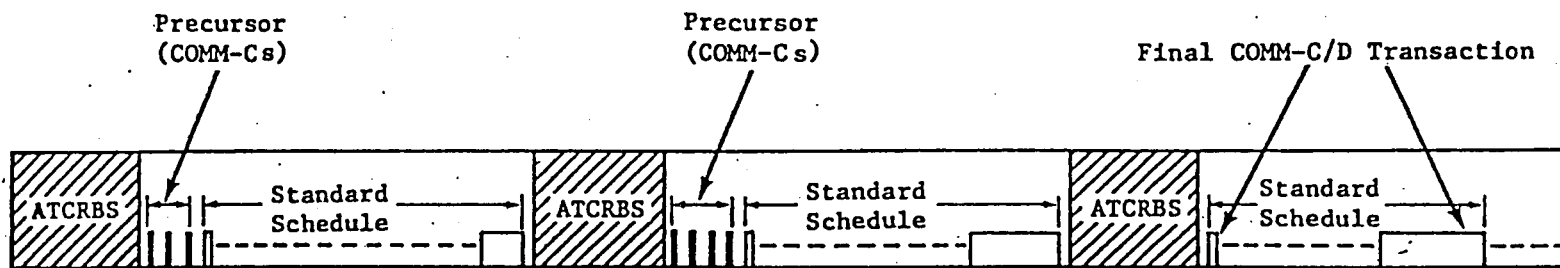


Figure A.4. ELM Scheduling [9]

N_a = number of ATCRBS interrogations per beamwidth,
 t_a = ATCRBS listening period,
 c = speed of light.

Figure A.5 presents plots of capacity vs. interrogator antenna beamwidth for various values of operating range. Typical values are used for scan time (4 seconds) and ATCRBS interrogations per beamwidth (four). Except on the longest (200 nmi) range, the ATCRBS listening interval was set at 2 ms to allow time for ATCRBS replies from distant targets (outside the operating range) to ring out before the beginning of the DABS periods.

The very large capacity of the DABS sensor is evident from these curves. For anticipated interrogator antenna beamwidths ($2.4^\circ - 4^\circ$) and operating range, the channel can accommodate more than 40 calls per degree, a number fully sufficient to accommodate expected sensor loading, including effects of azimuth bunching and multiple interrogations to each aircraft.

In the CDTI flight experiments conducted by the FAA Technical Center it was found that up to seven (7) Comm-A messages per dwell period can be received by the DABS aircraft. However, this figure reflects a low density environment. A more realistic number in a typical operating environment is one Comm-A/B pair and one complete (i.e., sixteen segments) ELM per dwell period.

Surveillance Accuracy DABS provides accurate surveillance data in range and bearing measurements for both the ATCRBS and DABS aircraft. The improvement in accuracy is obtained by essentially two design innovations. One is the mono-pulse technique for the ATCRBS mode and the other is the discrete addressing capability for the DABS mode. The surveillance data are obtained approximately every 4.8 sec for a terminal sensor and 5 sec for an enroute sensor. The surveillance reliability is expected to be better than 95%.

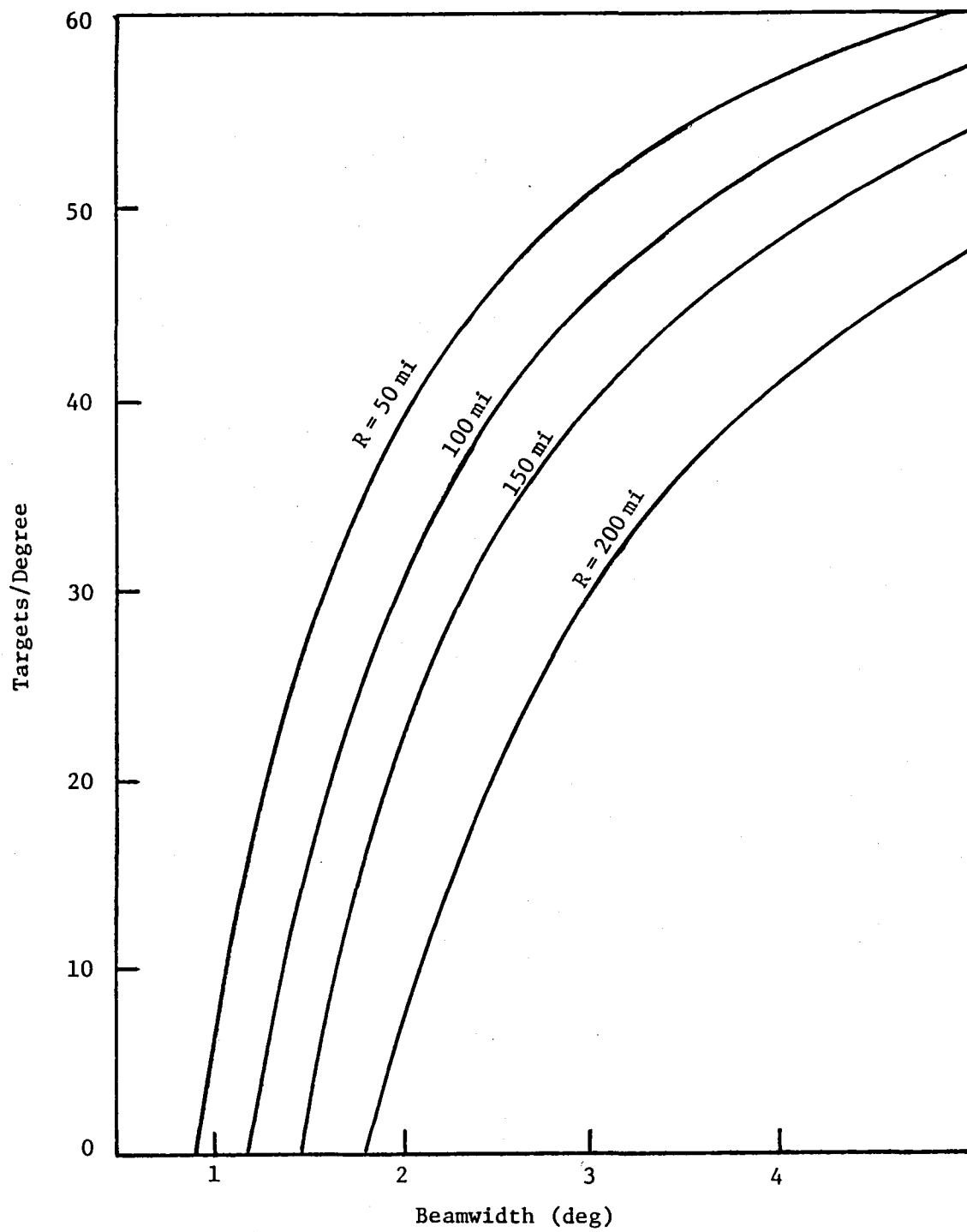


Figure A.5. DABS Capacity Plots [7]

The surveillance errors for both the DABS and ATCRBS modes are required not to exceed the following specification values:

Range: ± 150 ft (bias) ± 50 ft (jitter)
 Azimuth: ± 0.1 deg (jitter).

Preliminary accuracy validation data were taken at the FAA Technical Center using engineering DABS ground sensors and a NIKE tracking system. The preliminary results seem to indicate that the design specifications are exceeded. The following table summarizes the preliminary rms error findings:

Mode	Range	Azimuth
ATCRBS	$61 + 1.48 \cdot r$ ft (bias) [†] ± 60 ft (jitter)	± 0.04 deg (mean bias) ± 0.034 deg (jitter)
DABS	$-66 + 0.83 \cdot r$ ft (bias) [†] ± 22 ft (jitter)	± 0.02 deg (mean bias) [‡] ± 0.037 deg (jitter)

[†] The range bias is range (r) dependent, where r is expressed in nmi.

[‡] The DABS azimuth bias seems to be elevation dependent. The best mathematical model is given by the expression $1.58 - 1.5876 \text{ Secant}(El)$.

The effect of the bias values are not significant for the tracker algorithm design purposes. Various researchers in the past have used the following values as the DABS surveillance accuracy (jitter standard deviation) characteristics.

Scan Rate = 4.8 seconds
 Blip Scan = 0.95
 Sigma Azimuth = 0.40° = 0.000698 radians
 Sigma Range = 30 ft. = 0.00493 nm.

Because the final report of the FAA DABS sensor validation effort is not yet available, the error values presented are preliminary. More detailed descriptions can be found in Ref. 8 and 18.

DABS Messages and Data Formats

DABS Signals DABS provides both ground-to-air and air-to-ground data link capability. These capabilities are referred to as the uplink and downlink functions, respectively. The principal characteristics of the DABS RF signals are as follows:

Uplink

Frequency: 1030 MHz
Modulation: Differential Phase-Shift Keying (DPSK)
Data Rate: 4 Mbps

Downlink

Frequency: 1090 MHz
Modulation: Pulse Position Modulation (PPM)
Data Rate: 1 Mbps

The DABS messages are bit streams containing either 56 or 112 bits. The bit structures are specified according to the so-called DABS Data Format. The DABS uplink Data Format, downlink Data Format and subfield descriptions are given in Figs. A.6, A.7 and Table A.1, respectively.

Basic DABS Messages DABS contains provision for several message types to meet different data transfer needs. Table A.2 describes the message types of interest here. The data link function always requires the 112 bit format. If no data is to be transferred, the short (56 bit) messages are used.

The standard uplink message is called Comm-A and the standard downlink message is called Comm-B. Both provide for 56 data bits and include surveillance information as part of the 112 bit message. This means that a Comm-A message can send 56 data bits to an aircraft and at the same time provide the surveillance interrogation necessary to elicit a surveillance response from the aircraft. The aircraft usually responds with a surveillance reply; however, it can also respond with a Comm-B if air-to-ground data transmission is required in addition to the necessary surveillance response.

Format No.	UF	
0	(0 0000)	(AQ:1) (BI:26) (AP:24) Short Special Surveillance
1	(0 0001)	-----27----- (AP:24)
2	(0 0010)	(EP:8) -----19----- (AP:24) Short Synchr. Surveillance
3	(0 0011)	-----27----- (AP:24)
4	(0 0100)	(PC:3) (RR:5) (DI:3) (SD:16) (AP:24) Surveillance, Altitude
5	(0 0101)	(PC:3) (RR:5) (DI:3) (SD:16) (AP:24) Surveillance, Identity
6	(0 0110)	-----27----- (AP:24) Ground-Air Coordination
7	(0 0111)	-----27----- (AP:24)
8	(0 1000)	-----27----- (AP:24)
9	(0 1001)	-----27----- (AP:24)
10	(0 1010)	-----27----- (AP:24)
11	(0 1011)	(PR:4) (II:4) (--19 one's--) (AP:24) DABS-Only All-Call
12	(0 1100)	-----27----- (AP:24)
13	(0 1101)	-----27----- (AP:24)
14	(0 1110)	-----27----- (AP:24)
15	(0 1111)	-----27----- (AP:24)
16	(1 0000)	(AQ:1) (BI:26) (MU:56) (AP:24) . Long Special Surveillance
17	(1 0001)	-----83----- (AP:24)
18	(1 0010)	(EP:8) -----19----- (MS:56) (AP:24) . Long Synchr. Surveillance
19	(1 0011)	-----83----- (AP:24)
20	(1 0100)	(PC:3) (RR:5) (DI:3) (SD:16) (MA:56) (AP:24) . Comm-A, Altitude
21	(1 0101)	(PC:3) (RR:5) (DI:3) (SD:16) (MA:56) (AP:24) . Comm-A, Identity
22	(1 0110)	-----83----- (AP:24) . Ground-Air Coordination
23	(1 0111)	-----83----- (AP:24)
24	(11) (RC:2) (NC:4)	(HC:80) (AP:24) . Comm-C (ELM)

- Notes: (1) (XX:M) denotes a field designated "XX" which is assigned M bits,
 (2) ---N--- denotes free coding space with N available bits,
 (3) UF (Uplink Format) codes 24 through 31 are reserved for Comm-C transmissions. The leading bits of these codes are always "11"; the remaining bits vary with the content of the RC and NC fields.

Figure A.6. Summary of DABS Uplink Formats [7]

Format

No.	DF	
0	(0 0000)	(AQ:1) (BR:13) (AC:13) (AP:24) Short Special Surveillance
1	(0 0001)	-----27----- (AP:24)
2	(0 0010)	(EP:8) -----6----- (AC:13) (AP:24) Short Synchr. Surveillance
3	(0 0011)	-----27----- (AP:24)
4	(0 0100)	(FS:3) (DR:5) (UM:6) (AC:13) (AP:24) Surveillance, Altitude
5	(0 0101)	(FS:3) (DR:5) (UM:6) (ID:13) (AP:24) Surveillance, Identity
6	(0 0110)	-----27----- (AP:24) Not Used
7	(0 0111)	-----27----- (AP:24)
8	(0 1000)	-----27----- (AP:24)
9	(0 1001)	-----27----- (AP:24)
10	(0 1010)	-----27----- (AP:24)
11	(0 1011)	(CA:3) (AA:24) (PI:24) All-Call Reply
12	(0 1100)	-----27----- (AP:24)
13	(0 1101)	-----27----- (AP:24)
14	(0 1110)	-----27----- (AP:24)
15	(0 1111)	-----27----- (AP:24)
16	(1 0000)	(AQ:1) (BR:13) (AC:13) (SC:56) (AP:24) . . Long Special Surveillance
17	(1 0001)	-----83----- (AP:24)
18	(1 0010)	(EP:8) -----6----- (AC:13) (MT:56) (AP:24) . . Long Synchr. Surveillance
19	(1 0011)	-----83----- (AP:24)
20	(1 0100)	(FS:3) (DR:5) (UM:6) (AC:13) (MB:56) (AP:24) . . Comm-B, Altitude
21	(1 0101)	(FS:3) (DR:5) (UM:6) (ID:13) (MB:56) (AP:24) . . Comm-B, Identity
22	(1 0110)	-----83----- (AP:24) . . Not Used
23	(1 0111)	-----83----- (AP:24)
24	(11) --1--	(KE:1) (ND:4) (MD:80) (AP:24) . . Comm-D (ELM)

Notes: (1) (XX:M) denotes a field designated "XX" which is assigned M bits,
(2) ---N--- denotes free coding space with N available bits,
(3) DF (Downlink Format) codes 24 through 31 are reserved for Comm-D transmissions. The leading bits of these codes are always "11"; the remaining bits vary with the content of the KE and ND fields.

Figure A.7. Summary of DABS Downlink Formats [7]

TABLE A.1

DABS FIELD DESCRIPTIONS [7]

<u>Code</u>	<u>Field Name</u>	<u>Downlink (D)/Uplink (U) Meaning</u>
AA	Address Announced	D Aircraft identification in All- Call reply
AC	Altitude Code	D equivalent to aircraft Mode-C code
AP	Address/Parity	U/D error detection field
AQ	Acquisition	U/D part of BCAS protocol
BR	BCAS Reply Data	D special data for BCAS
CA	Capability	D aircraft report of system capability
DF	Downlink Format	D downlink descriptor
DI	Data Identification	U describes content of SD field
DR	Downlink Request	D aircraft requests permission to send data
EP	Epoch	U/D synchro-DABS time indicator
FS	Flight Status	D aircraft's situation report
ID	Identification	D equivalent to ATCRBS identity number
II	Interrogator Identification	U site number for multisite features
KE	Control, ELM	D part of Extended Length Message protocol
MA	Message, Comm-A	U message to aircraft
MB	Message, Comm-B	D message from aircraft
MC	Message, Comm-C	U long message segment to aircraft
MD	Message, Comm-D	D long message segment from aircraft
MS	Message, Synchro-DABS	D uplink synchro DABS message
MT	Message, Synchro-DABS	D downlink synchro DABS message
MU	Message, Uplink	U message without surveillance function
MS	Message, Synchro-DABS	D uplink synchro DABS message
MT	Message, Synchro-DABS	D downlink synchro DABS message
MU	Message, Uplink	U message without surveillance function (BCAS etc.)
NC	Number, C-segment	U part of ELM protocol
MD	Number, D-segment	D part of ELM protocol
PC	Protocol	U operating commands for the transponder
PI	Parity/Interr. Identifier	D reports source of interrogation
PR	Probability of Reply	U used in stochastic acquisition mode
RC	Reply Control	U part of ELM protocol
RR	Reply Request	U commands details of reply
SC	Special Communication	D BCAS report
SD	Special Designator	U control codes to transponder
UF	Uplink Format	U format descriptor
UM	Utility Message	D short message from aircraft

TABLE A.2

DABS MESSAGE TYPES [9]

TYPE	MESSAGE LENGTH (BITS)	INCLUDES SURVEILLANCE?	NUMBER OF DATA BITS	TRANSMISSION TIME* (MICROSECONDS)
UPLINK				
SURVEILLANCE INTERROGATION	56	YES	0**	18.5
COMM-A	112	YES	56	32.5
COMM-C	112	NO	80	32.5
DOWNLINK				
SURVEILLANCE REPLY	56	YES	0	64
COMM-B	112	YES	56	120
COMM-D	112	NO	80	120

*Uplink transmission on 1030 MHz frequency at 4 Megabits per second

Downlink transmission is on 1090 MHz frequency at 1 Megabits per second

**Current system allows a few bits in surveillance transactions for minimum data link function

Both the surveillance and the Comm-A interrogations require and elicit a downlink response. If such a response is not immediately received by the sensor, the uplink message is repeated.

More than 56 bits (the number of data bits in a single Comm-A or Comm-B) can be transacted in a single scan by utilizing a series of Comm-A and/or Comm-Bs. Surveillance, Comm-A and Comm-B messages are assigned high priority scheduling, which means that they are always scheduled first, before attempting to transmit other messages. Data that is significantly longer or that does not require urgent delivery is transmitted via Comm-C's and Comm-D's, through a protocol called Extended Length Messages (ELM).

Extended Length Messages (ELMs) ELM messages are composed of Comm-C (uplink) and Comm-D (downlink) segments. It should be noted that a Comm-C and a Comm-A are identical as far as their RF carrier characteristics are concerned. Both are uplink transmissions and are 112 bits long. They therefore occupy the same amount of time on the RF channel. Similarly, the Comm-B and the Comm-D are identical in their RF carrier frequency characteristics. Neither the Comm-C nor the Comm-D can, however, perform the surveillance functions.

An uplink ELM message consists of several Comm-C's (up to a maximum of 16) and two Comm-D's for bookkeeping. Individual Comm-C segments of an uplink ELM do not require individual acknowledgements (i.e., a response from the transponder). Each Comm-C allows the transfer of 80 data bits. Thus, up to a total of 1280 bits can be transferred in one ELM. The individual Comm-C segments can be sent in any number of subgroups over several DABS periods or several scans if necessary. Since the individual Comm-C segments do not require acknowledgements, they can be transmitted every 50 microseconds. 50 microseconds is the minimum separation between two interrogations. In the absence of any other constraints, this would presumably be the transmission rate used for Comm-C's.

ELM transmissions are inherently more efficient than Comm-A transmissions when transfers of data longer than about 150 bits are involved and potential delivery delays of up to a few scans are acceptable.

This means, that for the same amount of data required to be transferred, an ELM always requires less RF activity, as long as the data stream is at least 150 bits long. The downlink ELM protocol is similar to the uplink protocol, except that most of the transmission is downlink.

The uplink and downlink ELM protocols offer the potential of delivering significant amounts of data in a flexible manner. The major advantage of the ELM is that the transfer of the same amount of data through Comm-As and Comm-Bs require individual acknowledgements and provide for fewer data bits even though they occupy the same amount of channel time as Comm-Cs and Comm-Ds. The price paid for ELM is that the delivery may be completed over a much longer time span (e.g., several radar scans)*.

Projected ATC Services Supported by the DABS Data Link

The DABS data link will be the vehicle for providing many services which will contribute to the safety of aircraft, increase capacity of airports, increase controller/pilot productivity and facilitate procedures for maximum energy conservation. Possible services are given in Table A.3.

* Due to the transponder design, the upper bound for the uplink ELM is one complete ELM (i.e., 16 segments of COMM-Cs) per scan.

Table A.3

PROJECTED ATC SERVICES SUPPORTED BY THE
DABS DATA LINK [9]

Automated Separation Assurance

- . Automatic Traffic Advisory and Resolution Service (ATARS)*
- . Automated Minimum Safe Altitude Warning to Pilots

Automation of ATC Services

- . Confirmation of Clearances for Routing, Departure, Altitude Assignment, Holding and Approach
- . Voice Frequency Assignments for ATC Handoff Automation
- . Advanced Metering and Spacing
- . Automated Clearances
- . Flight Plan Revisions

Other Services

- . Weather
 - Severe Weather Advisories
 - Weather Information for Pilot-Requested Site
 - Digitized Weather Map
 - . Enhanced Terminal Information Service
 - Routine Terminal Information Including Runway In Use and Local Weather
 - Environmental Updates and Alerts
 - . Wind Profile Generation through Downlinked Air-Data
-
-

* The current FAA plan calls for the deletion of the Resolution (Collision Avoidance) part of ATARS. The remaining parts form the core of ATAS (Automatic Traffic Advisory Service).

APPENDIX B

DABS/ATARS BASED CDTI AVIONICS INTERFACE WITH THE TCV B-737

01 UPGRADE SYSTEM

Introduction

This appendix delineates the requirements for interfacing DABS/ATARS-based CDTI avionics with the proposed TCV B-737 01 Upgrade System. These requirements are preliminary in nature. Two limiting factors existing at this time are: (a) the detailed specifications for the 01 Upgrade are yet to be finalized; and (b) the requirement specification and design effort is always an iterative process. The appendix should be considered a first step toward an integrated CDTI flight system and should serve as a focal point for further development effort by various responsible individuals.

The CDTI avionics functional design contained in this appendix is based on the CDTI Data Formats specified in Chapter II and the DABS/ATARS ground system modification indicated in Chapter IV. Because the Upgrade System is planned to be very flexible in terms of input/output, display, data storage and computation capabilities, the "series retrofit" configuration approach, as developed in Chapter III, can be applied directly. The required hardware components are minimal and include an ELM-capable Mode S transponder, its associated Transponder Control Panel, and a Transponder Interface Unit.

In the following section, main features of the Upgrade System are discussed to the extent known at this time. Also, aspects which may directly concern the CDTI applications are noted along the way. Then, the required hardware aspects (notably TIU design) are discussed. The functional description of the airborne software modules are treated in Chapter III. The readers are referred to that section.

It should be understood that the proposed CDTI avionics configuration is meant to support research flight experiments. Therefore, system flexibility is stressed more when there are alternatives.

Major Features of TCV 01 Upgrade System

Major features of the Upgrade System which have direct bearing on the CDTI avionics interface design are briefly discussed here as background material. Even though the final specifications are yet to be made, the "big picture" is expected to remain unaltered.

System Architecture Figure B.1 shows a schematic diagram of the proposed 01 Upgrade System architecture. It consists of two Norden 11/70 M airborne computers, a versatile Computer Interface Unit (CIU), dual (pilot/copilot) display units, a pilot-to-system interface unit, a Control Mode Panel (CMP) and Data Translation Unit, various monitoring terminals for research stations, and several data storage devices including a Data Acquisition System (DAS).

The display subsystem consists of a programmable display generator which supports EADI and EHSI CRT display units in the cockpit and presumably identical repeater units for research stations. In addition, the display generator is provided with an output port for a less capable CRT unit.

The floppy disks are used for loading and unloading computer programs. A digital tape recording device is used for engineering system monitor and check purposes. The DAS is used as the main data storage device for post-flight data analysis purposes.

The data interface to and from the airborne computers is accomplished using Direct Memory Access (DMA) devices provided by the computers. The DMA data rate is 500 K word/sec at a word length of 16 bits. In the proposed architecture, the major portions of the sensor, actuator, computer and DAS I/O bookkeeping and timekeeping functions are transferred to the

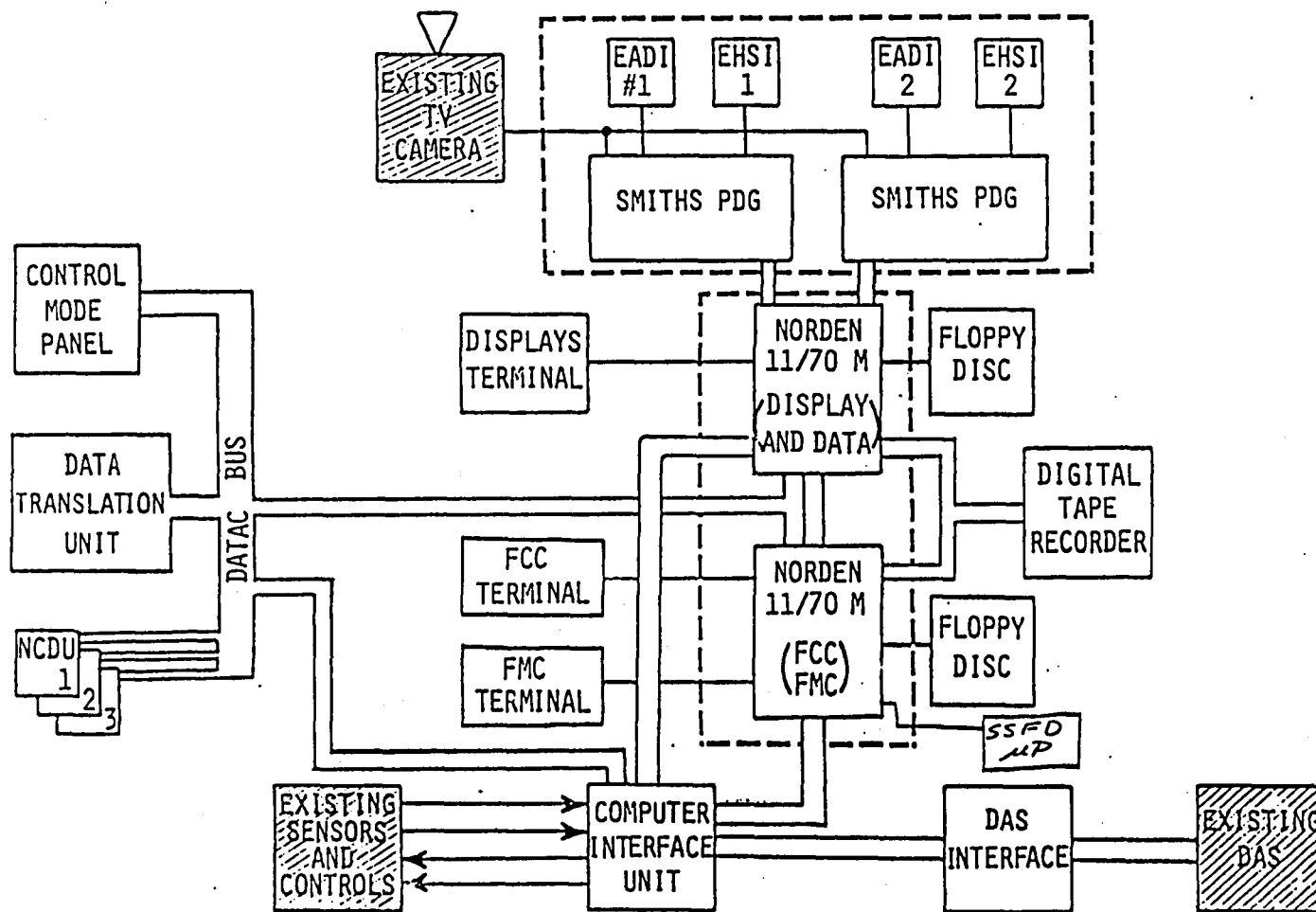


Figure B.1. TCV 01 Upgrade System Architecture

CIU. This frees the computer's CPU capability for main algorithmic computation purposes.

Airborne Computers The two Norden 11/70 M computers are 100% airborne versions of the DEC PDP 11/70 M. The memory capacity is 128 K at a 16 bit word length. One computer performs the flight control and flight management functions (FCC/FMC), and the other computer services the display and data processing functions (DDC). The two computers are interconnected via high-speed, bi-directional DMA, minimizing the computational duplicity between the two computers. Both computers are provided with identical inputs from the CIU via DMA.

The main programming language is a higher order language called HAL-S. The possibility and extent of using the more common FORTRAN capability remains to be seen. It is indicated that object codes from both sources could be linked.

The software structure for each computer is not specified in enough detail at this time. The unspecified information includes the exact partitioning of functional and computational separation between the two computers, the variables to be transferred from one computer to the other, and the global memory allocations. It is planned that the Airborne Executive would have a multi-priority background level structure; the highest priority level is presumably cycled at the minor tick (or cycle time of 10 msec interval).

Note 1: It is recommended that the CDTI software modules be implemented (see Chapter III) within the Display and Data Computer. This way the processed CDTI data need not be transferred from the FC/FM computer to the DDC. The CDTI software modules are expected to occupy less than 4 K words of memory. (This is not absolutely necessary because the inputs from the CIU are common to both computers, i.e., the CDTI uplink data are available in both computers.)

Note 2: The exact partitioning of display data processing between the DDC and the programmable symbol generator is not known.

Note 3: The multi-layered computer executive structure seems to be very flexible, and this will support the CDTI software requirements. As noted in Chapter III, most of the CDTI modules only need to be cycled at approximately a 4.7 sec interval. Thus, the modules can be accessed at that background level. Exceptions are uplink message decoding and downlink message packing modules. Care should be taken when foreground variables are accessed from a background level. The foreground variables should be transferred simultaneously to disjoint memory locations which are dedicated for the background level. This approach will allow the synchronization of the uplinked data with, for example, Own navigation variables.

Navigation and Control Display Unit (NCDU) The NCDU functions in the Upgrade System are presumably the same as the current TCV configuration. The NCDU is used essentially as an interface device between the pilot and the system for such functions as STAR selection, designated runway selection, and waypoint window modification. This is accomplished by providing "menu pages" on the CRT for pilot prompting for keyboard entry or mode selection.

Note: It is anticipated that a CDTI menu page needs to be added. The pilot actions are directed toward the airborne system or the ground system. In the latter case, based on the keyboard entry or mode selection, the DDC must construct and pack appropriate COMM-B messages which would be downlinked to the ground system.

Computer Interface Unit (CIU)* The heart of the Upgrade System is the versatile CIU which interfaces the airborne computers with the aircraft sensors and control actuators. Microprocessor based design allows the CIU to carry the major burden of I/O functions so that the computers can operate relatively independent of handshaking tasks with the aircraft sensors. Characteristics which are pertinent to the CDTI applications are given below.

Figure B.2 depicts the major functional elements of the CIU. It supports three types of input or output data: packed discretes, analog

* For detailed description of the preliminary CIU design requirements, the readers are referred to an internal working memo, "01 TCV B-737 Upgrade Computer Interface Unit Requirements" by C. Meissner of NASA/LRC.

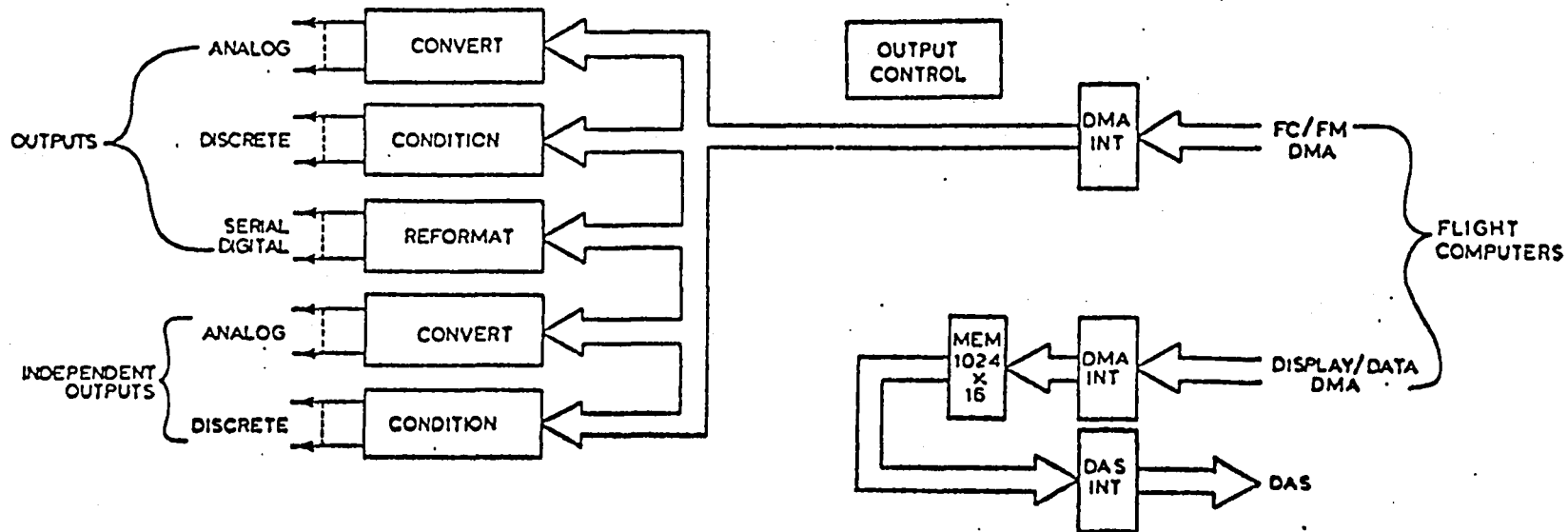
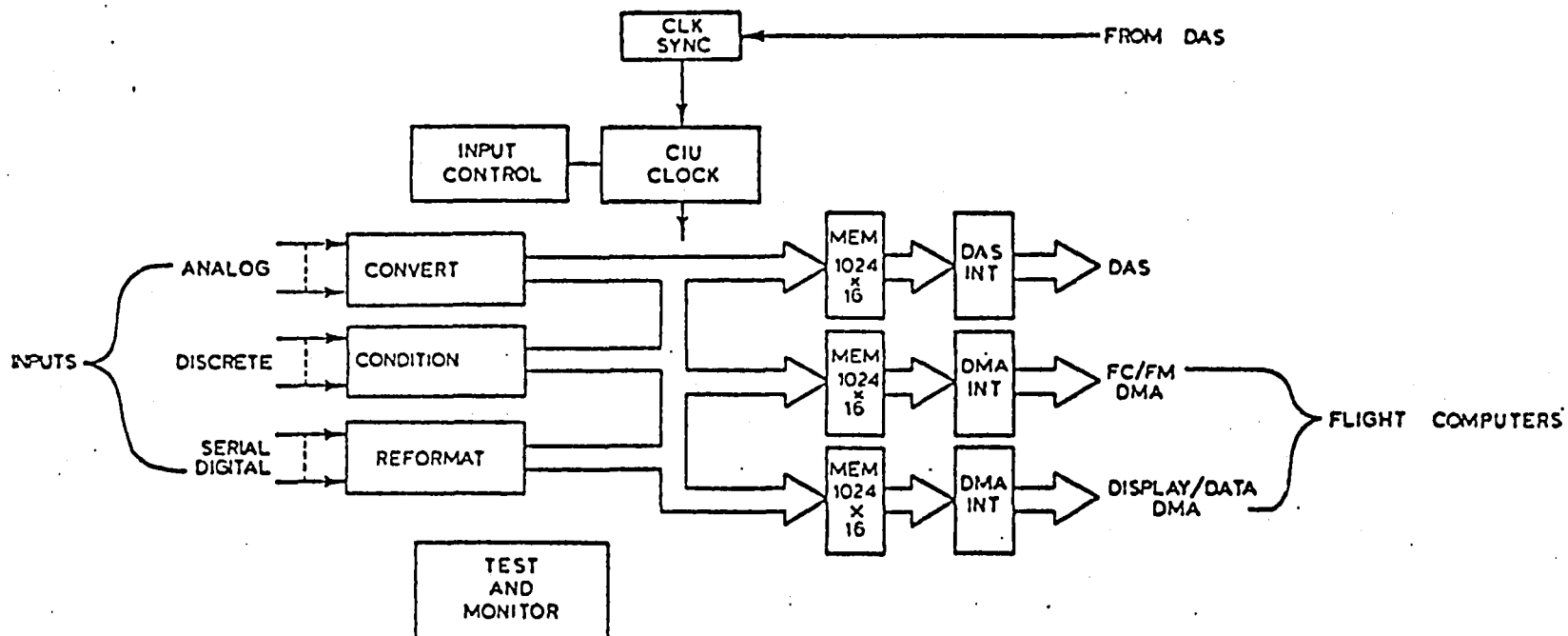


Figure B.2. Upgrade CIU Major Functional Elements

data and serial digital data. After proper processing or conditioning the sensor inputs are stored in holding memory locations (1024 * 16 bit) internal to the CIU. The memory contents are then transferred via DMA at the prompting of a 10 msec minor tick or a 50 msec major tick. See Fig. B.3 for the timing sequence. The output process is done in the reverse order.

The CIU and the computer memory are location tagged (not necessarily in the same sequential order); i.e., the particular location of the memory identifies the input variable. The correspondence between the CIU memory locations and the computer memory can be programmed.

The CIU can accommodate a variety of ARINC standard serial digital buses for input and output purposes. The serial bit streams are parity checked. Eight (8) bits are used to assign the label for each serial bus; thus the maximum number of words (or labels) per serial bus is 256. Figure B.4 shows the double word (32 bit) structure of serial digital transmission data in the computer as well as in the CIU memory. Accordingly, 22 bits out of the available 32 bits can be used to "pack" input data. Bit 32 is the parity bit for inputting and transmitting or the inhibit bit for outputting. Bit 31 is the complement (or reverse bit) of Bit 30 (sign bit). Bit 8 through 1 contains the label code. It is noted that the least significant bit (bit 1) is transmitted first. This contrasts with the FIFO transmission of the Mode S transponder output ports.

The current plan calls for accommodating three (3) Split Phase Bipolar (SPBP) buses, two of which are to be used for the MLS/DME receiver and Transponder Data System inputs. It is understood that more buses could be added depending on the applications.

Note: The CDTI avionics consisting of a Mode S transponder, its Control Panel, and the Transponder Interface Unit (TIU) will be interfaced to the CIU through digital serial buses. These will most probably be a pair of SPBP buses. Functional design of the TIU is given later.

Display System The O1 Upgrade System contains a versatile pilot display system. Figure B.5 shows a schematic diagram of the expected

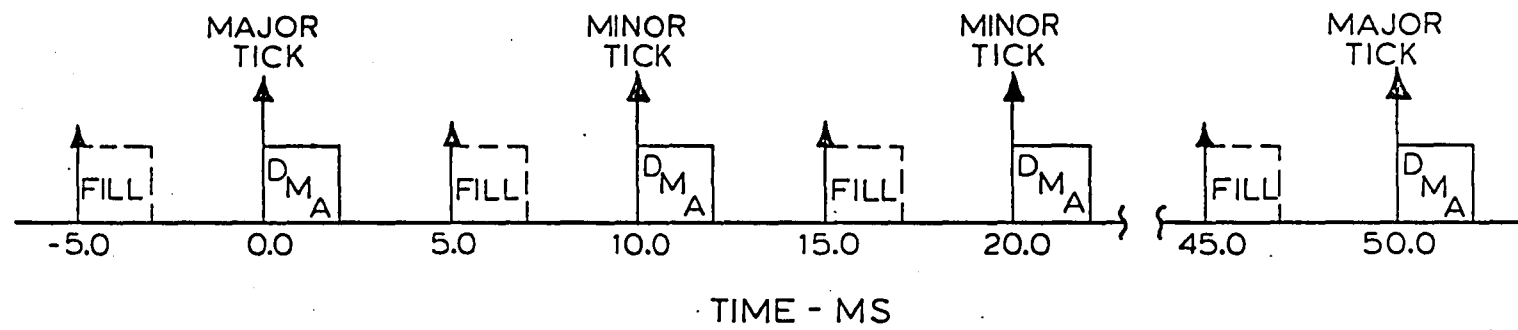


Figure B.3. Example Input Timing Sequence

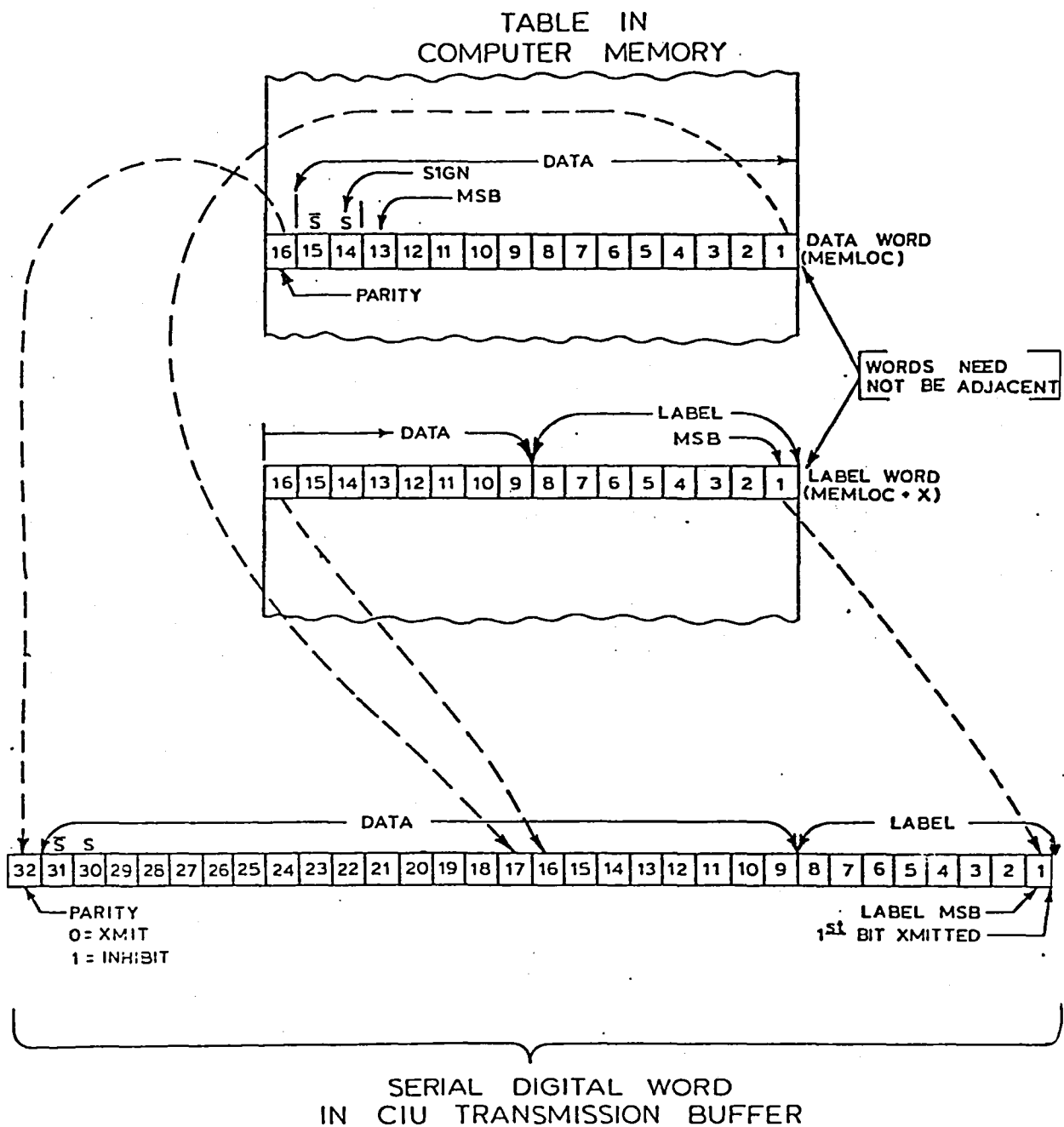


Figure B.4. Digital Word Translation

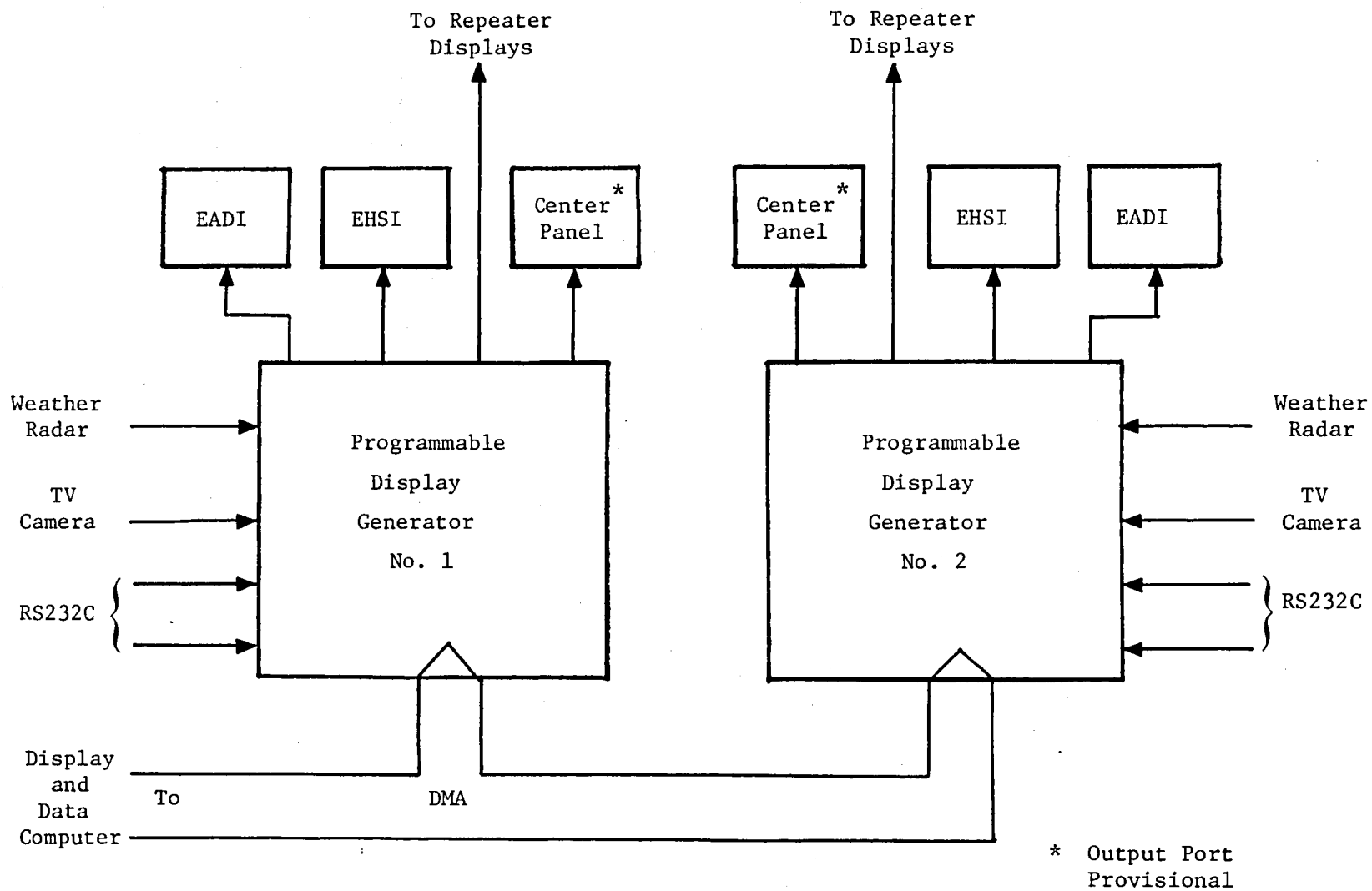


Figure B.5 Expected Display System Configuration

system configuration. The major components are Smith Industry's Programmable Display Generators (PDG), full hybrid (meaning vector stroke and raster scan) color CRT units (6.25" * 6.25" usable display area) for EHSI and EADI functions, and a provisional output port for supporting a less capable (most probably stroke only) CRT unit (Center Panel). Repeater displays are provided for the research/monitor stations.

The PDG interfaces with the Norden 11/70 M DDC via DMA. It also interfaces with a weather radar and a forward looking TV camera located at the front of the aircraft fuselage. Some of the major features and capabilities of the display generator are:

1. It supports full hybrid CRT units (stroke outline and raster fill);
2. Its special stroke characters are stored in 4 K * 8 bits of firmware; and
3. It contains 32 K * 16 bits of programmable RAM, and it uses assembly language programming.

The software partitioning and allocation between the PDG and the DDC computer is yet to be defined. The approach philosophy seems to be that routine computations which do not depend on the DDC input dynamically are performed in the PDG. For complex symbology which dynamically depends on sensor input data, the associated computations are performed in the DDC computer; and the vector positions are transmitted to the PDG.

Note 1: The Center Panel CRT display unit is planned for a "cluster-of-instruments" display for flight management purposes. A simpler PWI or AIDS/ATARS type CDTI may be displayed using this unit.

Note 2: It is not known if the PDG can support a smaller less capable CRT (such as one used for the ATARS/AIDS configuration).

Note 3: For the VFR applications of the CDTI concept, either the Center Panel or one of the repeater EHSI units may be installed in the forward cockpit. If the repeater unit is used, then the symbology could be identical to one of the dual units. Software development is necessary if the Center Panel is used. One of the advantages of the simultaneous deployment of CDTI in the forward

and aft cockpits is that a single research flight can support both VFR and EFR (Electronic Flight Rules) CDTI experiments.

Note 4: It is not clear how the computational load for generating the CDTI traffic symbols is distributed between the PDG and the DDC computer. It seems that the traffic symbols should be superimposed on the map features as background picture elements, since the traffic data update rate is rather slow. The number of displayed traffic aircraft (which may depend on the map scale), aircraft data tag selection, and aircraft symbol selection should remain under the DDC software control, i.e., under the pilot control.

Data Acquisition System (DAS) The Upgrade System contains at least three data storage devices -- floppy disks, digital tape recorders, and the DAS (PMC tape). The present plan calls for the floppy disks to be used for loading or unloading computer programs and possibly as an auxiliary memory. The tape recorders are to be used for engineering check and monitor purposes. The DAS is for post-flight data analysis purposes.

The DAS capability should be exploited for the CDTI flight experiments. Three types of data, PADS frame, sensor information and computer generated data are stored every 50 msec on major ticks. The computer generated data are transferred to the DAS via the DAS Computer Information Interface (CII). The CIU/CII has a maximum capacity of $1024 * 16$ bit words (currently 224 words are planned). Presumably, the physical location within the DAS computer buffer identifies the variable; i.e., the DAS tape can be read according to the sequential order of variable appearance.

Note: Care should be taken when storing the CDTI related data on the DAS tape. A maximum of $105 * 16$ bit words are generated every 4.7 sec for the CDTI application. The estimate is based on five Comm-A messages (56 bits each), $16 * 80$ ELM bits and approximately $20 * 16$ bits of computer generated threat parameters. It is not wise to allocate 105 storage locations every 50 msec, since they do not change for 4.7 sec or 94 50 msec intervals. With time multiplexing usage of a storage location (i.e., variables are rotated every 50 msec), four (4) 16-bit locations can accommodate 5760 bits of storage over a 4.5 sec interval ($4 \text{ locations} * 16 \text{ bits} * 20 \text{ Hz} * 4.5 \text{ sec}$). This should be more than sufficient to store the CDTI related variables. The multiplexing logic must be implemented in a flight computer.

CDTI Avionics Interface

Major CDTI avionics interface elements are discussed following the organization of Chapter III. The series retrofit configuration is appropriate for the Upgrade System; therefore, the additional airborne component requirements are an ELM capable Mode S transponder, the Transponder Control Panel and the Transponder Interface Unit.

A schematic diagram for the series retrofit configuration is shown in Fig. B.6. The data transfer to and from the Transponder Interface Unit and the Computer Interface Unit can be accomplished via a pair of SPBP buses with the line speed of 50 Kbps. The parity bit (Bit 32 of the double words buffer) and the sign reversal bit (Bit 31 = complement of Bit 30) are the only communication test bits for this bus. Therefore, the SPBP bus should be secure with respect to the communication transaction.

The rest of the retrofit requirements are satisfied by DDC software additions or modifications. These include decoding and unpacking, Track File management, smoothing/compensation of the uplinked data, threat parameter computation, CDTI symbol generation, and packing of downlink message modules. The descriptions of the required modules are discussed in Chapter III at a subroutine level.

Discussions of a Transponder Interface Unit design are given next.

Transponder Interface Unit (TIU) The TIU which spans between the Mode S transponder and the Computer Interface Unit must be designed and fabricated. Its functions are as follows:

- (1) Receive, verify (or clean-up) and store uplink data from the transponder;
- (2) Output the stored uplink data to the CIU;
- (3) Receive and store downlink data from the CIU; and
- (4) Output the stored downlink data to the transponder.

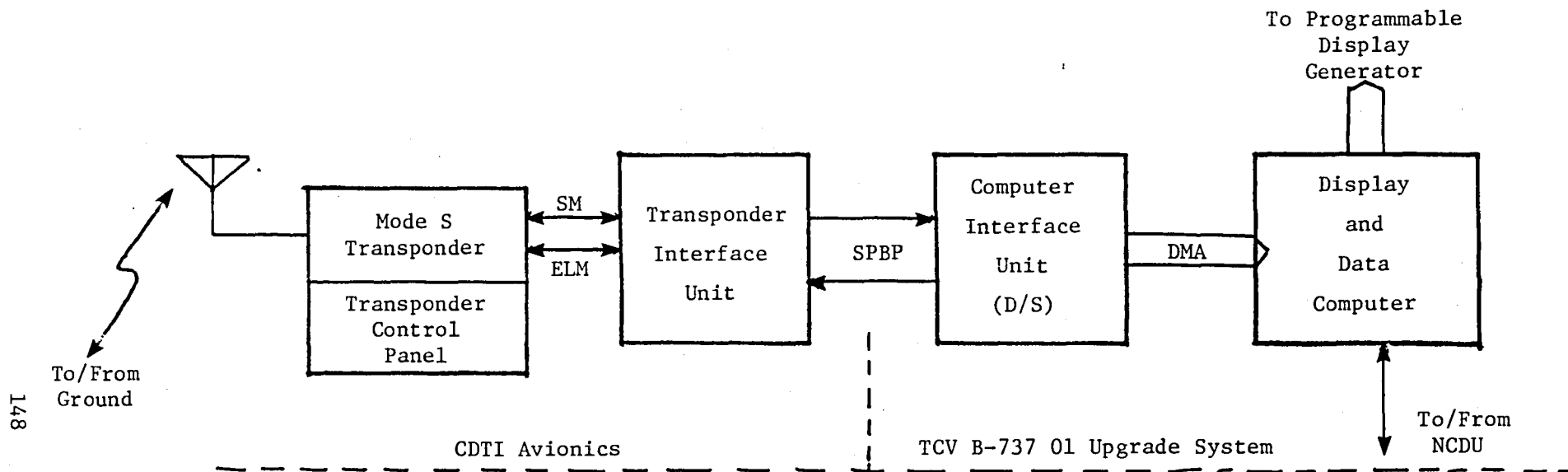


Figure B.6. Overall CDTI Avionics Interface with the TCV B-737 01 Upgrade System

In Chapter III, general specifications and a preliminary microprocessor-based design for a TIU are given at the functional level. The front end (i.e., the transponder side) of the TIU remains essentially the same. The back end (i.e., the airborne computer side) can be specified further, given the Upgrade CIU functional specifications. The following operational characteristics are assumed.

- (a) The data transfer between the TIU and the CIU are performed via a pair of SPBP digital serial buses (one for inputting and the other for outputting) since such provisions are already made in the Upgrade System. The line speed is 50 Kbps or approximately 750 μ sec per label (32 bit double words) including the spacing. The serial data bus can support up to 256 labels.
- (b) The CIU memory (and hence the corresponding computer memory) can be updated every 50 msec at major ticks.
- (c) Twenty-two (22) bits per label can be used to pack serial data. Bit 32 is used for odd parity check; Bit 31 is always the complement of Bit 30; and the last eight bits designate the label code.
- (d) The Mode S transponder is assumed to receive a maximum of five Comm-A messages and one complete (16-segment) ELM during one antenna dwell period. In addition it is assumed to downlink a maximum of three Comm-B messages and one two-segment ELM during the same dwell period.
- (e) It is estimated that all the uplink messages are stored in a TIU storage buffer ready to be transferred to the CIU within 300 msec of the first uplink message interrupt. The estimate includes the message reception time (less than the 35 msec beam dwell period), message processing and dead time (150 msec budget), and the transportation time from the transponder to the TIU (110 msec).

Note 1: It takes four labels to transport one Comm-A (88 bits) or one ELM segment (80 bits).

Note 2: Previously the Standard Message (SM) and ELM interfaces were treated somewhat independently at the back end. However, with the CIU specifications in mind, it may be more advantageous to store both Comm-A and Comm-C messages in the same buffer. In this way, only one SPBP is needed for each direction; i.e., this eliminates the need for one pair of SPBP buses for Comm-A and B and another for Comm-C and D messages.

Note 3: As mentioned previously, it is not certain at this stage, what the maximum number of Comm-A message uplinked per scan is.

The uncertainty comes because it is not known what functions other than the CDTI applications are supported by the DABS ground system. (Comm-A protocol is not used for the CDTI per se). Five Comm-A messages seem to be a reasonable provisional number.

Note 4: If the total transportation delay between the initial uplink interrupt time and the airborne computer processing initiation time is to be less than 0.5 sec, then the data transfer from the TIU to the CIU should be completed in less than 200 msec. The transfer time from the CIU to the flight computers is negligible. The transportation delay for the downlink process does not seem very critical.

Figure B.7 shows a schematic diagram depicting an overall input process from the transponder to the Display and Data Computer. The following discussion applies to the figure.

The Standard Message (SM) processor of the TIU receives the uplinked Comm-A messages (exclusive of the 24-bit address and parity code) via the transponder's SMI port. All the incoming Comm-A messages are accumulated for the dwell duration. Any duplicate messages are deleted and stored in the Comm-A portion of a TIU storage buffer (21 * 88 bits).

The ELM processor of the transponder accumulates the uplinked ELM segments. When the uplink is completed, the transponder transfers the message content (i.e., MC part of Comm-Cs) to the TIU on a FIFO basis through the ELM port. The resulting bit stream consists of the number of segments * 80 bits of MC. The ELM processor of the TIU subdivides the incoming bit stream into segments of 80 bits each, i.e., the original MC segments. The processor attaches eight bits in front of the 80 bit segments to form 88 bit segments each. The eight bits consist of two bits for ELM designation, four bits for segment number assignment, and two spare bits. The resulting number of segments * 88 bits of data are stored in the ELM portion of the TIU storage buffer.

The storage buffer size is 21 * 88 bits (five Comm-As plus sixteen modified ELM segments). The first five locations of the buffer are reserved for the Comm-As, and the rest are for the ELM. If no uplink message is present, then the corresponding buffer locations should be null. The resulting TIU storage buffer organization is shown in Fig. B.8.

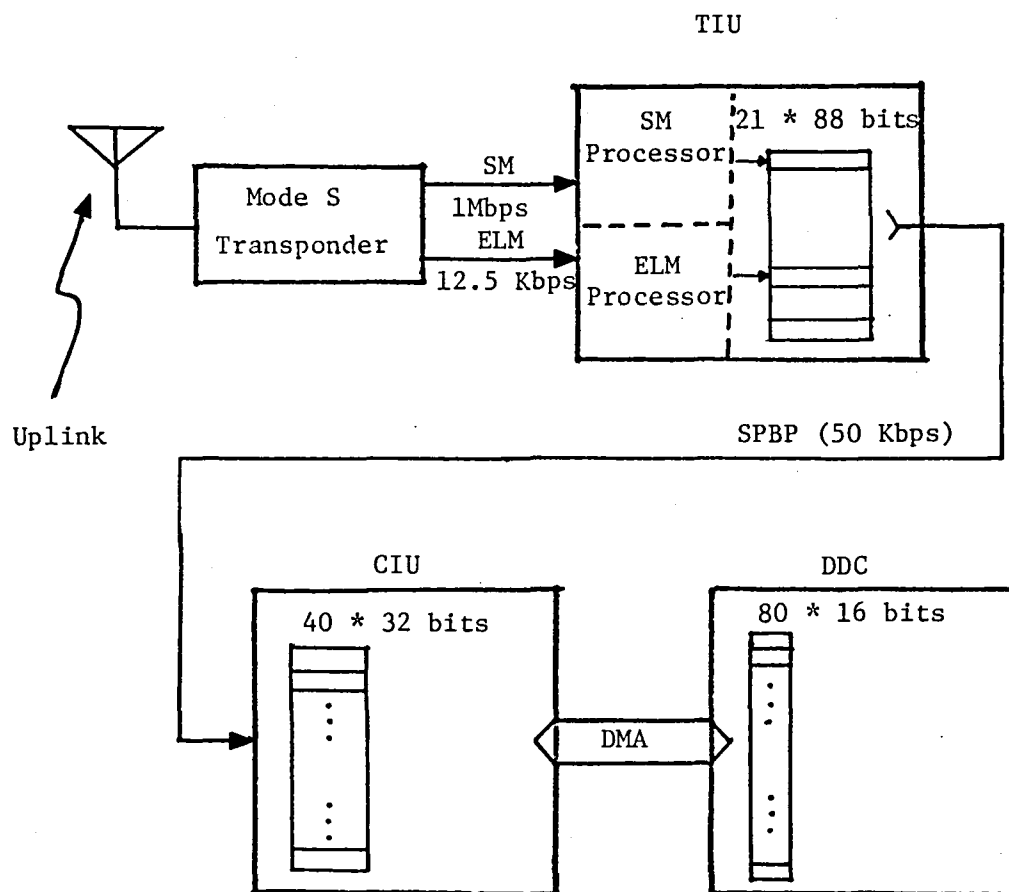
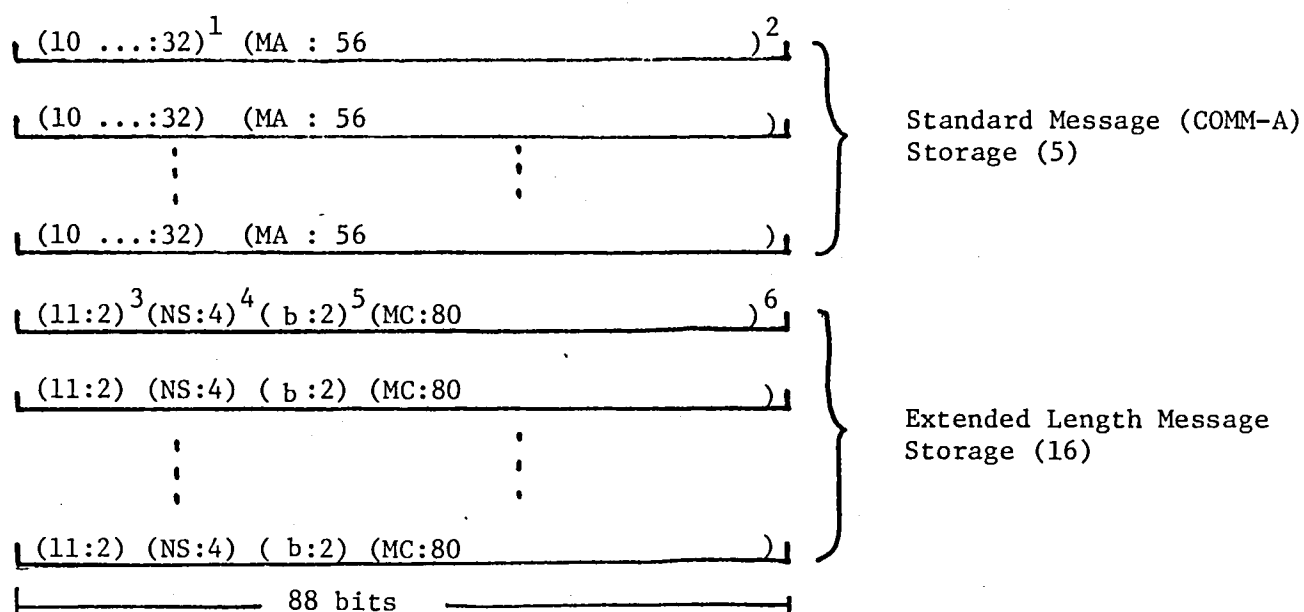


Figure B.7. Schematic Diagram of Input Process



(yy ... y:N) \equiv Bit pattern or subfield designation followed by assigned number of bits.

- (1) COMM-A messages always begin with (10) pattern
- (2) (MA:56) = 56 bits MA message field
- (3) (11) bit pattern indicates this is ELM
- (4) (NS:4) = 4 bit segment number designation (reconstructed in the ELM processor)
- (5) (b:2) = two spare bits
- (6) (MC:80) = 80 bits MC message field.

Figure B.8. TIU Storage Buffer Organization

It is noted that the Comm-A messages (DABS Data Format No. 20 and 21; of Fig. A.6) begin with a 10-bit pattern. Therefore, if the modified ELM segments begin with a 11-bit pattern, then there would be no confusion for decoding purpose.

As is depicted in Fig. B.7, upon an interrupt from the CIU, the contents of the storage buffer are transferred to the CIU via the SPBP bus. This transfer is done over 2 or 3 stages (i.e., over 100 ~ 150 msec) as follows. The first ten non-zero locations of the twenty-one storage locations are transferred. Each 88-bit buffer location is broken into four 22-bit length segments. Each 22-bit segment is packed according to the digital serial data format (see Figs. B.4 and B.9). That is, the parity and reverse sign bits are added in front, and the eight-bit label code (with consecutive label numbers) is added at the tail end to form a 32-bit serial data word which is sent to the CIU via the SPBP bus. This process is repeated until all the non-zero buffer locations are transferred.

When the $40 * 32$ bits of data are received and stored in the CIU storage memory, the contents are transmitted via DMA to the Display and Data Computer Memory per the CIU specification.

The downlink process from the DDC to the transponder should be done in the reverse order. The process should be initiated by the DDC (i.e., construction and transmission of downlink messages) approximately 3.5 sec after the first reception of the uplink messages from the CIU. This will allow for reasonable transportation lag.

Note 5: $40 * 32$ bits (or $80 * 16$ bits) of CIU memory location needs to be allocated for uplink message processing. Hence, $80 * 16$ bits of DMA locations are needed in the DDC. The same number of locations are needed for the downlink process.

Note 6: It is estimated that the uplink message transfer would be completed within two input periods most of the time, since Comm-A messages are not always present every antenna scan. Therefore, the transportation time from the CIU to the DDC is approximately 100 msec. The downlink processing should be done in one 50 msec interval, since it involves at most three Comm-Bs and one two-segment ELM per scan.

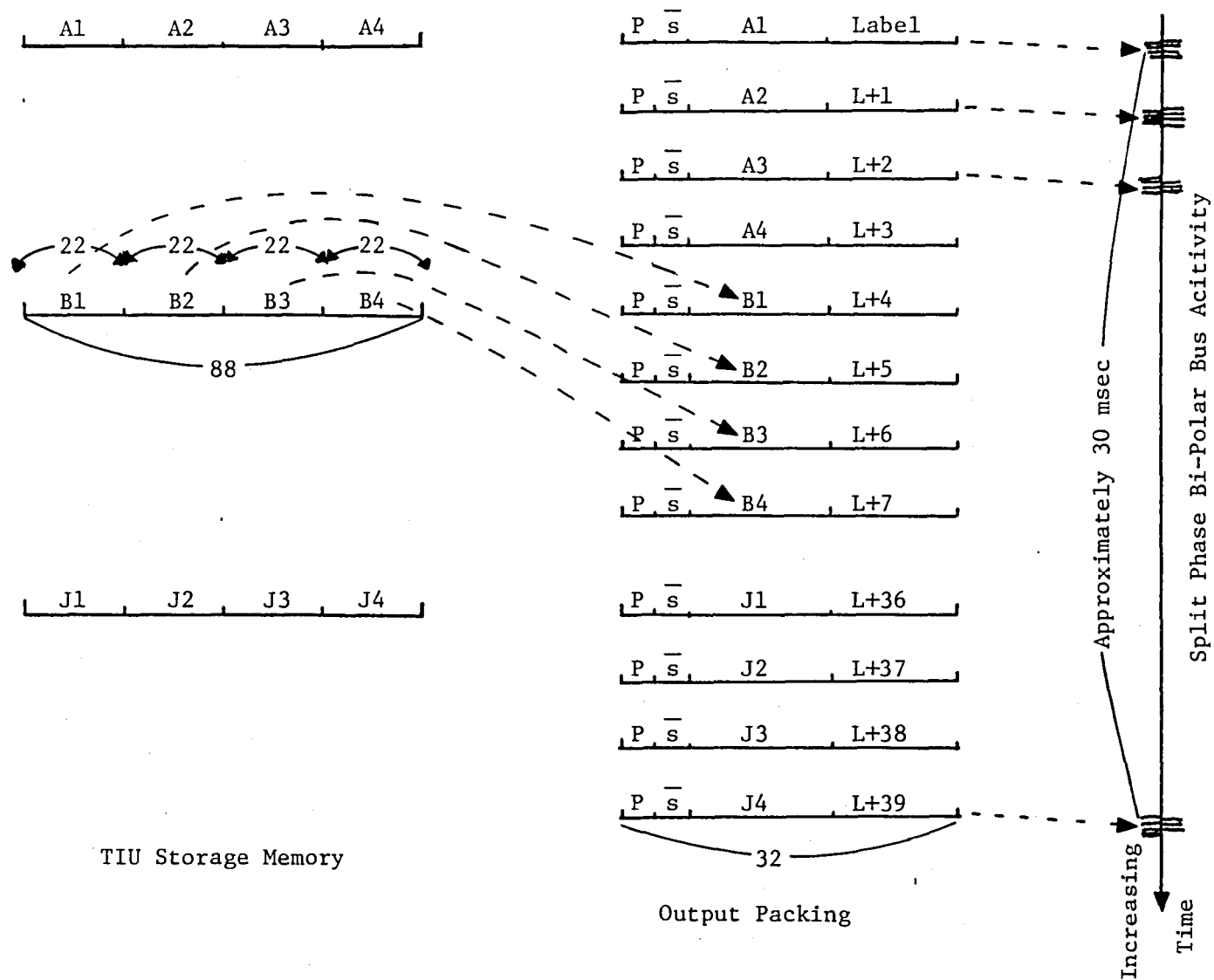


Figure B.9. Schematic Diagram Dipping the Output Process from the TIU to CIU

Note 7; The exact hand-shake operations (i.e.; definitions and functions of the necessary interrupts and/or control lines among the Mode S transponder, TIU and CIU) need to be worked out in detail.

Note 8: It is recommended that consecutive memory locations be assigned in the DDC so that Do-loops in steps of eight (It takes $8 * 16$ bits for a Comm-A or one segment of ELM.) can be used for decoding and unpacking purposes. Furthermore, the memory locations should be reset to zero afterwards for easier monitoring. The CDTI buffer locations should be monitored every 50 msec. If non-zero elements are present, then the locations contain new uplink data.

Note 9; The design task would be simpler if more labels are used. However, it should be pointed out that the corresponding memory locations would be idle for most of the 4.7 sec interval (except for 50 to 100 msec of active transactions). On the other hand, if a larger transportation delay is permissible, the required data transfer can be accomplished over several 50 msec (major tick) intervals with fewer memory allocations.

Note 10: It is estimated that the total time lag (from the Mode S surveillance report to the cockpit display) is of the order of 1 - 2 sec. This does not include the delay within the DABS/ATARS ground processor.

This Page Intentionally Left Blank

REFERENCES

CDTI

1. Anon, "Joint NASA/FAA Program - Cockpit Display of Traffic Information (CDTI)," Federal Aviation Administration and National Aeronautics and Space Administration, Feb. 1980.
2. Anon, "Cockpit Displayed Traffic Information Study," Report No. D642968, Boeing Commercial Airplane Company, Seattle, WA, Sept. 1977.
3. Abbott, T.S., et al, "Flight Investigation of Cockpit Displayed Traffic Information Utilizing Coded Symbolology in an Advanced Operational Environment," NASA TP 1684, July 1980.

T-39 Avionics

4. Anon, "Software Documentation: T-39 CCD and CCU," Bendix Corp.

DABS

5. Anon, "DABS-Foundation for Future Air Traffic Control," FAA Publication, FAA, Washington, D.C.
6. Anon, "Proposed U.S. National Aviation Standard for the Discrete Address Beacon System (DABS)," Federal Register, FAA, March 1978.
7. Orland, V.A. and P.R. Drouilhet, "Discrete Address Beacon System Functional Description," Report No. FAA-RD-80-41, MIT/Lincoln Laboratory, Lexington, MA, April 1980.
8. Holtz, M., et al, "Discrete Address Beacon System (DABS) Baseline Test and Evaluation," Report No. FAA-RD-80-36, FAA, Washington, D.C., April 1980.
9. Mundra, A.D., "DABS Data Link Capability Utilization: Interim Report," MTR-80 W301, Mitre Corp., McLean, VA, Nov. 1980.
10. Diendonno, J.E. and R.W. Lautenschlager, "Discrete Address Beacon System/Automated Traffic Advisory and Resolution Service/Air Traffic Control Operational System Description," Report No. FAA-RD-80-42, Mitre Corp., McLean, VA, April 1980.
11. Welch, J.D., and P.H. Robeck, "Proposed Technical Characteristics for the Discrete Address Beacon System (DABS)," Report No. FAA-RD-77-143 (ATC-71), MIT/Lincoln Laboratory, Lexington, MA, Sept. 1977.

ATARS

12. Scardina, J.A., "Automatic Traffic Advisory and Resolution Service (ATARS). The Design of a Ground-Based Solution to Midair Collisions," WESCON Paper No. 7/2, Los Angeles, CA, Sept. 1980.
13. Lentz, R.H., et al, "Automatic Traffic Advisory and Resolution Service (ATARS) Multi-Site Algorithms," Report No. FAA-RD-80-3, Rev. 1, Mitre Corp., McLean, VA, Oct. 1980.

14. Lentz, R.H., et al, "Automatic Traffic Advisory and Resolution Service (ATARS) Algorithms Including Resolution Advisory-Register Logic," Report No. FAA-RD-81-45, Rev. 2, Mitre Corp., McLean, VA, June 1981.

DABS/ATARS Data Format

15. Anon, "U.S. National Aviation Standard for the Automatic Traffic Advisory and Resolution Service - Draft," Federal Aviation Administration, Washington, D.C., March 1987.
16. Lucier, E., "Uplink Information Provided by ATARS," SRDS Report No. RD-80-11-LR, FAA, Washington, D.C., April 1980.
17. Leeper, J.L., and R.S. Kennedy, "DABS Data Link Applications Formats (Rev. 1)," ATC Working Paper No. 42 WP-5083, MIT/Lincoln Laboratory, Lexington, MA, Feb. 1980.

ATARS Tracker Algorithm

18. Goka, T. and J.A. Sorensen, "CDTI Sensor Model Development and Filter Investigation," Report No. AMA 81-10, Analytical Mechanics Associates, Mt. View, CA, Aug. 1981.
19. Norris, D.O., and L.E. Snyder, "Analysis of the Automatic Traffic Advisory Resolution Service (ATARS) Tracking and Turn Detection Function," Interim Report, Avionics Engineering Center, Ohio University, Athens, OH, June 1981.
20. Steinbacker, J.G., "Results of Threshold Alpha-Beta-Gamma (TABG) Tracking Simulation," Report No. MTR-79W00330, Mitre Corp., McLean, VA, Oct. 1979.

AIDS, Hardware Interface, Transponder Manual

21. Gertz, J.L., "Initial Design and Experimental Implementation of the Traffic Advisory Service of ATARS," Report No. FAA-RD-80-117 (ATC-101), MIT/Lincoln Laboratory, Lexington, MA, Nov. 1980.
22. Anderson, J.C., and J.L. Leeper, "The DABS Data Link Airborne Intelligent Display Operator's Manual," Report No. FAA-RD-80-106 (ATC-100), MIT/Lincoln Laboratory, Lexington, MA, Sept. 1980.
23. Colby, G.V., et al, "Provisional Data Link Interface Standard for the DABS Transponder," Report No. FAA-RD-74-64 (ATC-34), MIT/Lincoln Laboratory, Lexington, MA, April 1974.
24. Robeck, P.H., "Present the Rationale for Including a 'Real Time' Interface in the MOPS for the DABS Transponder," Report Memo to RTCA SC-142, MIT/Lincoln Laboratory, Lexington, MA, Sept. 1979.
25. Anon, "TRU-2 DABS/ATC Transponder System," Maintenance Manual I.B. 1171A, Avionics Division, Bendix Corp., Fort Lauderdale, FL, Dec. 1980.

1. Report No. CR-165947		2. Government Accession No.		3. Recipient's Catalog No.	
4. Title and Subtitle FUNCTIONAL DESIGN TO SUPPORT CDTI/DABS FLIGHT EXPERIMENTS				5. Report Date July 1982	
				6. Performing Organization Code	
7. Author(s) Tsuyoshi Goka				8. Performing Organization Report No.	
9. Performing Organization Name and Address Analytical Mechanics Associates, Inc. 2483 Old Middlefield Way, Suite 210 Mountain View, CA 94043				10. Work Unit No.	
				11. Contract or Grant No. NAS1-16802	
				13. Type of Report and Period Covered Contractor Report	
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546				14. Sponsoring Agency Code	
15. Supplementary Notes Langley Technical Monitor: David Williams Final Report					
16. Abstract The objectives of this project are to (a) provide a generalized functional design of CDTI avionics using the FAA developed DABS/ATARS ground system as the "traffic sensor", (b) specify software modifications and/or additions to the existing DABS/ATARS ground system to support CDTI avionics, (c) assess the existing avionics of a NASA research aircraft in terms of CDTI applications, and (d) apply the generalized functional design to provide research flight experiment capability. In this report, DABS Data Link Formats are first specified for CDTI flight experiments. The set of CDTI/DABS Format specifications becomes a vehicle to coordinate the CDTI avionics and ground system designs, and hence, to develop overall system requirements. The report is the first iteration of a system design and development effort to support eventual CDTI flight test experiments.					
17. Key Words (Suggested by Author(s)) CDTI avionics and display DABS, Mode S Sensor			18. Distribution Statement Unclassified - Unlimited		
19. Security Classif. (of this report) Unclassified		20. Security Classif. (of this page) Unclassified		21. No. of Pages 158	
				22. Price	

End of Document