ARINC

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS), PART 3 FILE DATA TRANSFER TECHNIQUES

ARINC SPECIFICATION 429P3-18

PUBLISHED: OCTOBER 12, 2001

AN A DOCUMENT
Prepared by
AIRLINES ELECTRONIC ENGINEERING COMMITTEE
Published by
AERONAUTICAL RADIO, INC.
2551 RIVA ROAD, ANNAPOLIS, MARYLAND 21401

ARINC SPECIFICATION 429P3-18©
MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

PART 3

FILE DATA TRANSFER TECHNIQUES

Published: October 12, 2001

Prepared by the Airlines Electronic Engineering Committee

Specification 429P3	Adopted by the Airlines Electronic Engineering Committee:	July 21, 1977
Specification 429P3	Adopted by the Industry:	September 15, 1977

Summary of Document Supplements

Supplement	Adoption Date	Published
Specification 429P3-1	April 11, 1978	June 1, 1978
Specification 429P3-2	December 6, 1978	March 1, 1979
Specification 429P3-3	August 31, 1979	November 1, 1979
Specification 429P3-4	June 17, 1980	August 1, 1980
Specification 429P3-5	March 12, 1981	April 24, 1981
Specification 429P3-6	December 9, 1981	January 22, 1982
Specification 429P3-7	November 4, 1983	January 3, 1984
Specification 429P3-8	November 4, 1984	December 3, 1984
Specification 429P3-9	October 11, 1984	April 30, 1985
Specification 429P3-10	November 7, 1985	November 17, 1986
Specification 429P3-11	June 15, 1988	July 22, 1988
Specification 429P3-12	October 25, 1989	July 1, 1990
Specification 429P3-13	October 8, 1991	December 30, 1991
Specification 429P3-14	November 4, 1992	January 4, 1993
Specification 429P3-15	April 18, 1995	September 1, 1995
Specification 429P3-16	June 24, 1997	June 30, 1997
Specification 429P3-17	March 31, 1999	May 17, 1999
Specification 429P3-18	July 18, 2001	October 12, 2001

A description of the changes introduced by each supplement is included on Goldenrod paper at the end of this document.

FOREWORD

Activities of AERONAUTICAL RADIO, INC. (ARINC)

and the

Purpose of ARINC Reports and Specifications

Aeronautical Radio, Inc. is a corporation in which the United States scheduled airlines are the principal stockholders. Other stockholders include a variety of other air transport companies, aircraft manufacturers and non-U.S. airlines.

Activities of ARINC include the operation of an extensive system of domestic and overseas aeronautical land radio stations, the fulfillment of systems requirements to accomplish ground and airborne compatibility, the allocation and assignment of frequencies to meet those needs, the coordination incident to standard airborne compatibility, the allocation and assignment of frequencies to meet those needs, the coordination incident to standard airborne communications and electronics systems and the exchange of technical information. ARINC sponsors the Airlines Electronic Engineering Committee (AEEC), composed of airline technical personnel. The AEEC formulates standards for electronic equipment and systems for the airlines. The establishment of Equipment Characteristics is a principal function of this Committee.

It is desirable to reference certain general ARINC Specifications or Report which are applicable to more than one type of equipment. These general Specifications and Reports may be considered as supplementary to the Equipment Characteristics in which they are referenced. They are intended to set forth the desires of the airlines pertaining to components and general design, construction and test criteria, in order to insure satisfactory operation and the necessary interchangeability in airline service. The release of a Specification or Equipment Characteristics should not be construed to obligate ARINC or any airline insofar as the purchase of any components or equipment is concerned.

An ARINC Report (Specification or Characteristic) has a twofold purpose, which is:

(1) To indicate to the prospective manufacturers of airline electronic equipment the considered opinion of the airline technical people, coordinated on an industry basis, concerning requisites of new equipment, and

(2) To channel new equipment designs in a direction which can result in the maximum possible standardization of those physical and electrical characteristics which influence interchangeability of equipment without seriously hampering engineering initiative.

ARINC SPECIFICATION 429 PART 3
TABLE OF CONTENTS

ITEM	SUBJECT	PAGE
1.0	INTRODUCTION	1
1.1	Purpose of this Document	1
1.2	Organization of ARINC Specification 429	1
1.3	Development of Data Transfer	1
1.3.1	File Data Transfer Techniques - Basic Philosophy	1
1.3.2	Data Transfer	1
1.3.3	Broadcast Data	1
1.3.4	File Data Transfer	1
1.3.5	Transmission Order	2
1.3.5.1	Data Bit Encoding Logic	2
1.3.6	Bit-Oriented Protocol Determination	2
1.4	Relationship to Other Standards	2
2.0	BIT ORIENTED FILE TRANSFER PROTOCOL	3
2.1 - 2.4	Not used to maintain consistency with previous versions of ARINC Specification 429	3
2.5	Bit-Oriented Communications Protocol	3
2.5.1	Link Data Units (LDU)	4
2.5.2	Link Data Unit (LDU) Size and Word Count	4
2.5.3	System Address Labels (SALs)	4
2.5.4	Bit Rate and Word Timing	4
2.5.5	Word Type	5
2.5.6	Protocol Words	5
2.5.6.1	Protocol Identifier	5
2.5.6.2	Destination Code	5
2.5.6.3	Word Count	5
2.5.7	Request To Send (RTS)	5
2.5.7.1	Clear To Send (CTS)	5
2.5.7.2	Not Clear To Send (NCTS)	5
2.5.7.3	Destination Busy (BUSY)	6
2.5.7.4	No Response to RTS	6
2.5.8	Conflicting RTS Transmissions	6
2.5.8.1	Half Duplex Mode	6
2.5.8.2	Full Duplex Mode	7
2.5.9	Unexpected RTS	7
2.5.10	Start Of Transmission (SOT)	7
2.5.10.1	General Format Identifier (GFI)	7
2.5.10.2	File Sequence Number	7
2.5.10.3	LDU Sequence Number	7
2.5.11	Data	7
2.5.11.1	Full Data Word(s)	7
2.5.11.2	Partial Data Word(s)	8
2.5.12	End of Transmission (EOT)	8
2.5.12.1	CRC Encoding	8
2.5.12.2	CRC Decoding	8
2.5.13	Negative Acknowledgment (NAK)	9
2.5.13.1	Missing SOT Word	9
2.5.13.2	Missing EOT Word	9
2.5.13.3	Parity Errors	9
2.5.13.4	Word Count Errors	9
2.5.13.5	CRC Errors	9
2.5.13.6	Time Out Errors	9
2.5.14	LDU Transfer Acknowledgment (ACK)	9
2.5.14.1	Duplicate LDU	9
2.5.14.2	Auto-Synchronized Files	9
2.5.14.3	Incomplete File timer	10
2.5.15	SYN Word	10
2.5.16	Response to ACK/NAK/SYN	10
2.5.17	Solo Word (Single Word Transfers)	10
2.5.17.1	Test Word and Loop Word	11
2.5.17.2	Optional Solo Word Definitions	11
2.5.18	Optional End-To-End Message Verification	11
2.5.19	Protocol Initialization	11
2.5.19.1	Bit-Oriented Protocol Version	11
2.5.19.1.1	ALOHA	11

TABLE OF CONTENTS

ITEM	SUBJECT	PAGE
2.5.19.1.2	ALOHA Response	12
2.5.19.2	Williamsburg/File Transfer Determination	12
2.6	Windowed Bit-Oriented Communications Protocol	13
3.0	BIT ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL	14
3.1	Bit Oriented Media Access Control (MAC) Protocol	14
3.1.1	Introduction	14
3.1.2	Relationship Between the Version 1 and Version 3 Protocols	14
3.1.3	Protocol Architecture	15
3.1.4	Buffering	15
3.2	Media Access Control (MAC) Sublayer	15
3.2.1	MAC Sublayer Service Specification	15
3.2.2	MA_DATA_request	16
3.2.2.1	Function	16
3.2.2.2	Semantics	16
3.2.2.2.1	destination_MA_address	16
3.2.2.2.2	destination_SAL_address	16
3.2.2.2.3	m_sdu	16
3.2.2.2.4	service_class	16
3.2.2.3	When Generated	17
3.2.2.4	Effect of Receipt	17
3.2.3	MA_DATA.indication	17
3.2.3.1	Function	17
3.2.3.2	Semantics	17
3.2.3.2.1	destination_MA_address	17
3.2.3.2.2	destination_SAL_address	17
3.2.3.2.3	source_MA_address	17
3.2.3.2.4	source_SAL_address	17
3.2.3.2.5	m_sdu	17
3.2.3.2.6	reception_status	17
3.2.3.2.7	service_class	17
3.2.3.3	When Generated	18
3.2.3.4	Effect of Receipt	18
3.2.4	MAC Control functions	18
3.2.4.1	MA_CONTROL.request	18
3.2.4.2	MA_CONTROL.indication	19
3.3	MAC Frame Structures	19
3.3.1	Information Frame Format	19
3.3.1.1	Address fields	19
3.3.1.2	Length/Type Field	19
3.3.1.2.1	Length	19
3.3.1.2.2	Type	20
3.3.1.3	Data Field	20
3.3.1.4	Frame Check Sequence (FCS) Field	20
3.3.1.5	Validation of Information Frame	20
3.3.1.5.1	Invalid Address	20
3.3.1.5.2	Invalid Length/Type	20
3.3.1.5.3	Invalid FCS	20
3.3.2	Command Frame Format	20
3.3.2.1	GFI Field	20
3.3.2.2	Command Type Field	21
3.3.2.3	Data Field	21
3.3.2.4	Frame Check Sequence (FCS) Field	21
3.3.2.5	Validation of Command Frame	21
3.3.2.5.1	Invalid FCS	21
3.4	MAC Transmit/Receive Functions	21
3.4.1	Frame Data Unit (FDU)	21
3.4.2	Frame Data Unit (FDU) Size and Word Count	21
3.4.3	System Address Labels (SALs)	21
3.4.4	Bit Rate and Word Timing	22
3.4.5	Word Type	22
3.4.6	Start of Frame (SOF)	22
3.4.6.1	Information/Command (I/C) Frame Field	22
3.4.6.2	Information SOF word	22

ARINC SPECIFICATION 429 PART 3
TABLE OF CONTENTS

ITEM	SUBJECT	PAGE
3.4.6.2.1	Word Count	22
3.4.6.2.2	Reserved Bits	22
3.4.6.3	Command SOF word	22
3.4.6.3.1	Word Count	22
3.4.6.3.2	Reserved Bits	22
3.4.6.3.3	General Format Identifier Field (GFI)	22
3.4.6.3.4	Command Type Field (CT)	22
3.4.7	Data	23
3.4.7.1	Full Data Word(s)	23
3.4.7.2	Partial Data Word(s)	23
3.4.7.3	SOLO Words	23
3.4.8	End of Frame (EOF)	23
3.4.9	Frame Check Sequence	23
3.4.9.1	32-Bit CRC Encoding	24
3.4.9.2	32-Bit CRC Decoding	24
3.4.10	Incomplete FDU Timer	24
3.4.11	ALOHA	24
3.4.12	Validation of FDUs	25
3.4.12.1	Missing SOF Word	25
3.4.12.2	Missing EOF Word(s)	25
3.4.12.3	Parity Errors	25
3.4.12.4	Word Count Errors	25
3.4.12.5	CRC Errors	25
3.4.13	Inter-FDU Gap Time	26

ATTACHMENTS

1 - 9	Numbers not used in Part 3 of Specification 429	27
10	Variables of Bit-Oriented Protocol	28
11	Bit-Oriented Data File Transfer Word Formats	35
11A	Destination Codes	41
11B	Status Codes	43
11C	ALOHA/ALOHA Response Protocol Word Definition	44
12	Version 1 File Transfer Example	45
12A	Field Mapping Example	46
13	Protocol Determination Procedure Diagrams	47
13A	ALOHA Version Determination Sequence	49
14	System Address Labels	50
15	Deleted by Supplement 16	51
16	Deleted by Supplement 16	52
17	Flow Diagram Used to Determine Character-Oriented vs Bit-Oriented Protocol	53
18	MAC Sublayer Support Diagrams	54
19	Command Frame Data Unit (FDU) Structure and Examples	56
20	Information Frame Data Unit (FDU) Structure and Example	58

APPENDICES

A-E	Letters not used in Part 3 of Specification 429	60
F	Former AIM and File Data Transfer Techniques	61
G	Mathematical Example of CRC Encoding/Decoding	67
H	Interoperability of Bit-Oriented Link Layer Protocol	71
I	SDL Diagrams of the Version 1 Williamsburg Protocol	72
J	Protocol Structure	90
K	Glossary & Acronyms	93

1.0 INTRODUCTION

1.1 Purpose of this Document

This document defines the air transport industry's standards for the transfer of aperiodic digital data between avionics systems elements. Adherence to these standards is desired for all inter-systems communications in which the system line replaceable units are defined as "unit interchangeable" in the relevant ARINC equipment Characteristics. Their use for intra-system communications in systems in which the line replaceable units are defined in the ARINC equipment Characteristics as "system interchangeable" is not essential, although it may be convenient.

1.2 Organization of ARINC Specification 429

c-15

The original release of ARINC Specification 429 was published in its entirety as one document in 1977. Through the years as the Specification grew in content, the physical size grew proportionately. As a result, the effort involved with locating specific data became increasingly difficult. The solution, concurrent with the publication of Supplement 15, was to divide Specification 429 into three parts. Part 1 addresses the physical parameters (wiring, voltage levels, coding, etc.) and label assignments. Part 2 provides the formats of words with discrete bit encoding. Part 3 defines file data transfer protocols.

Parts 1, 2, and 3 are being published separately beginning with the updates provided by Supplement 15. In the future, updates to the individual Parts of ARINC Specification 429 will be accomplished by independent Supplements starting with Supplement 16. Each Part will be updated via Supplement as the need dictates. Therefore the "dash numbers," i.e. -16, -17, etc. may not necessarily be concurrent for all three parts of Specification 429.

The descriptive material for the changes introduced by the Supplement 1-14 are provided in Part 1. Part 3 contains Supplements 12 and above. The new bit-oriented protocol was introduced by Supplement 12. The description of changes introduced by Supplements 15 and later for each Part are contained within the respective Parts of the document.

1.3 Development of File Data Transfer

ARINC Specification 429, "Mark 33 Digital Information Transfer System (DITS)" was adopted by AEEC in July 1977. Specification 429 defined a broadcast data bus. General provisions were made for file data transfer. In October 1989, AEEC updated a file data transfer procedure with a more comprehensive process that will support the transfer of both bit and character-oriented data.

c-16

COMMENTARY

The ACARS character protocol is defined in ARINC Specification 619.

c-15

COMMENTARY

The desire for exchanging binary data via ACARS was instrumental in initiating the development of a more sophisticated file transfer protocol. The fundamental concept was developed at a joint Satellite and ACARS Protocol Working Group meeting held in February 1988 in Williamsburg, Virginia. The new protocol became known popularly as the "Williamsburg Protocol."

1.3.1 File Data Transfer Techniques - Basic Philosophy

This "File Data Transfer Techniques" specification describes a system in which an LRU may generate binary extended length messages "on demand." Data is sent in the form of Link Data Units (LDU) organized in 8-bit octets. System Address Labels (SAL) are used to identify the recipient. Two data bus speeds are supported.

1.3.2 Data Transfer

c-16

The same principles of the physical layer implementation described in Part 1 to ARINC Specification 429, "Functional Description and Word Formats," apply to File Data Transfer. Any avionics system element having information to transmit does so from a designated output port over a single twisted and shielded pair of wires to all other system elements having need of that information. Unlike the simple broadcast protocol that can deliver data to multiple recipients in a single transmission, the File Transfer technique can be used only for point-to-point message delivery.

1.3.3 Broadcast Data

c-15

The same simple "broadcast" transmission technique defined in ARINC Specification 429 Parts 1 and 2 may be supported concurrently with the use of aperiodic File Data Transfer.

1.3.4 File Data Transfer

c-16

When Specification 429 was adopted in 1977, provisions were made for a character-oriented file data transfer protocol. This definition was used as guidance for the character-oriented file transfer protocol descriptions incorporated into many ARINC equipment characteristics. In 1989, a new protocol was developed that expanded the capability of file data protocol to support the transfer of bit-oriented information. The original description of file data transfer was declared obsolete; a copy, as a historical record, is retained in Appendix F. The ACARS character oriented file transfer protocol which was derived from the material in Appendix F is documented in ARINC Specification 619.

c-13

The protocol defined in this document is preferred for new applications. The purpose of this bit-oriented communications protocol is to provide for the transparent transfer of data files using the physical layer data bus defined by Specification 429, Part 1.

c-15

COMMENTARY

The data transparent protocol described in Section 2.5 was developed in order to facilitate the communications of the ACARS Management Unit (MU) and the Satellite Data Unit (SDU). Its viability as a universal protocol was

1.0 INTRODUCTION

1.3.4 File Data Transfer (cont'd)

COMMENTARY (cont'd)

c-15 recognized by the Systems Architecture and Interfaces(SAI) Subcommittee which recommended its inclusion herein as the standard means of file data transfer.

The process for determining the protocol (character-oriented as defined in ARINC Specification 619, or bit-oriented) to be used in the interaction between two units, where this information is not pre-determined is described in Section 2.5.19.

1.3.5 Transmission Order

c-17 The most significant octet of the file and least significant bit (LSB) of each octet should be transmitted first. The label is transmitted ahead of the data in each case. It may be noted that the Label field is encoded in reverse order, i.e., the least significant bit of the word is the most significant bit of the label. This "reversed label" characteristic is a legacy from past systems in which the octal coding of the label field was, apparently, of no significance.

1.3.5.1 Data Bit Encoding Logic

A "HI" state after the beginning of the bit interval returning to a "NULL" state before the end of the same bit interval signifies a logic "one."

A "LO" state after the beginning of the bit interval returning to a "NULL" state before the end of the same bit interval signifies a logic "zero." This is represented graphically in ARINC Specification 429, Part 1 Attachment 7.

1.3.6 Bit-Oriented Protocol Determination

c-16 An LRU will require logic to determine which protocol (character or bit-oriented) and which version to use when prior knowledge is not available. See Section 2.5.19.1 for bit protocol version determination.

1.4 Relationship to Other Standards

This document defines an onboard data link protocol to be used for file data transfer between cooperating Line Replaceable Units (LRU). As an onboard data bus, this standard is often included in AEEC equipment standards (ARINC 700 series) by reference.

c-17 Conversely, this Specification also references other documents. For example, Version 3 of the file transfer protocol, defined in Chapter 3 herein, utilizes many of the principles of the IEEE communications standard 802.3. Appropriately, there are numerous references to that standard. IEEE 802.3, 1998 was current when the definition of Version 3 was completed. Since the IEEE standard may evolve over time, a generic (non-time dated) reference (i.e., IEEE 802.3) is used wherever possible to enable the reference within this document to remain current, to the maximum extent possible, without future Supplements. Exceptions to this practice include references to specific clauses or paragraphs of IEEE 802.3, 1998. These references are not intended to limit the growth or evolution of these provisions, but rather to ensure that the reader is equipped with sufficient information to ensure that the desired section will be located.

Documents referenced in this document include: c-17

IEEE Standard 802.3, 1998 Edition, "Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications"

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.1 This Section number is not used in Part 3 to maintain consistency with previous versions of ARINC Specification 429.

2.2 This Section number is not used in Part 3 to maintain consistency with previous versions of ARINC Specification 429.

2.3 This Section number is not used in Part 3 to maintain consistency with previous versions of ARINC Specification 429.

2.4 This Section number is not used in Part 3 to maintain consistency with previous versions of ARINC Specification 429.

2.5 Bit-Oriented Communications Protocol

c-13 This section describes Version 1 of the bit-oriented (Williamsburg) protocol and message exchange procedures for file data transfer between units desiring to exchange bit-oriented data assembled in data files. This protocol should be used in lieu of the character-oriented file data transfer defined in ARINC Specification 619. All other bus activity remains unchanged. The bit-oriented protocol is designed to accommodate data transfer between sending and receiving units in a form compatible with the Open Systems Interconnect (OSI) model developed by the International Standards Organization (ISO). This document directs itself to an implementation of the Link layer, however, an overview of the first four layers (Physical, Link, Network and Transport) is provided.

Communications will permit the intermixing of bit-oriented file transfer data words (which contain System Address Labels (SALs)) with conventional data words (which contain label codes). If the sink should receive a conventional data word during the process of accepting a bit-oriented file transfer message, the sink should accept the conventional data word and resume processing of the incoming file transfer message.

The process for determining the protocol (character-oriented or bit-oriented) to be used in the interaction between two units, where this information is not pre-determined is described in Section 2.5.19. The definition of the protocol words used to determine the type of protocol is
c-12 contained in Table 11-4 of Attachment 11.

A table illustrating the bit-oriented file transfer word formats is shown in Attachment 11.

The description provided in the following subsections contains references to options which may be exercised and timing values which may be selected for each individual system for which this protocol is chosen. The options are designated with an "O" and a sequence number, e.g., O_5. The timing values are designated with a "T" and a sequence number, e.g., T_2. See Attachment 10 for tables containing standard options, events, applications and timers.

COMMENTARY

> There is no protocol to support negotiation of the parameters, and options such as those defined in Attachment 10. c-12

The data file and associated protocol control information are encoded into 32-bit words and transmitted over the physical interface as described in Part 1 of Specification 429. At the Link layer, data is transferred using a data transparent bit-oriented data file transfer protocol designed to permit the units involved to send and receive information in multiple word frames. It is structured to allow the transmission of any binary data organized into a data file composed of octets. Examples of file transfer and field mapping are given in Attachments 12 and 12A respectively. The bit-oriented protocol will support either full or half duplex operation (O_1). c-16

A. Physical Medium

The physical interface should be as described in Part 1 of Specification 429.

B. Physical Layer

The Physical layer provides the functions necessary to activate, maintain and release the physical link which will carry the bit stream of the communication. The electrical interface, voltage, timing, etc. described in Part 1 of Specification 429 should be used by the interfacing units. Data words will contain 32 bits; bits 1-8 will contain the System Address Label (SAL) and bit 32 will be the parity (odd) bit. c-12

C. Link Layer

The Link layer is responsible for transferring information from one logical network entity to another and for enunciating any errors encountered during transmission. The Link layer provides a highly reliable virtual channel and some flow control mechanisms.

D. Network Layer

It is the responsibility of the Network layer to ensure that data packets are properly routed between any two terminals. The Network layer performs a number of functions. The Network layer expects the Link layer to supply data from correctly received frames. c-13

COMMENTARY

> The Network layer provides for the decoding of information up to the packet level in order to determine which node (unit) the message should be transferred to. To obtain interoperability, this process, though simple in this application, must be reproduced using the same set of rules throughout all the communications networks (and their subnetworks) on-board the aircraft and on the ground. c-14

The bit-oriented data link protocol was designed to operate in a bit-oriented Network layer environment. Specifically, the Data Link Subcommittee expects that ISO 8208 will be selected as the Subnetwork layer protocol for air/ground c-13

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.5 Bit-Oriented Communications Protocol (cont'd)

c-13 subnetworks. There are, however, some applications where the bit-oriented file transfer protocol will be used under other Network layer protocols.

E. Transport Layer

c-12 The Transport layer controls the transportation of data between a source end-system to a destination end-system. It provides "network independent" data delivery between these processing end-systems. It is the highest order of function involved in moving data between systems. It relieves higher layers from any concern with the pure transportation of information between them.

2.5.1 Link Data Units (LDU)

A Link Data Unit (LDU) contains binary encoded octets. The octets may be set to any possible binary value. The LDU may represent raw data, character data, bit-oriented messages, character-oriented messages, or any string of bits desired. The only restriction is that the bits be organized into full 8-bit octets. The interpretation of those bits is not a part of this Link layer protocol. The LDUs are assembled to make up a data file.

c-16 LDUs consist of a set of contiguous ARINC 429 32-bit data words, each containing the System Address Label (see Section 2.5.3) of the sink. The initial data word of each LDU is a Start of Transmission (SOT) as described in Section 2.5.10. The data described above is contained within the data words which follow (See Section 2.5.11). The LDU is concluded with an End of Transmission (EOT) data word (see Section 2.5.12). No data file should exceed 255 LDUs.

Within the context of this document, LDUs correspond to frames and files correspond to packets, as defined in Section 2.5.

2.5.2 Link Data Unit (LDU) Size and Word Count

c-12 The Link Data Unit (LDU) may vary in size from 3 to 255 ARINC 429 words including the SOT and EOT words. When a LDU is organized for transmission, the total number of ARINC 429 words to be sent (word count) is calculated. The word count is the sum of the SOT word, the data words in the LDU and the EOT word.

In order to obtain maximum system efficiency, the data should be encoded into the minimum number of LDUs.

COMMENTARY

c-13 The word count field is 8 bits in length. Thus the maximum number of ARINC 429 words which can be counted in this field is 255. The word count field appears in the RTS and CTS data words. The number of LDUs needed to transfer a specific data file will depend upon the method used to encode the data words.

2.5.3 System Address Labels (SALs)

LDUs are sent point-to-point, even though other systems may be connected and listening to the output of a transmitting system. In order to identify the intended recipient of a transmission, the Label field (bits 1-8) is used to carry a System Address Label (SAL). Each on-board system is assigned a SAL as shown in ARINC Specification 429, Part 1, Attachment 14. When a system sends an LDU to another system, the sending system (the "source") c-13 addresses each ARINC 429 word to the receiving system (the "sink") by setting the Label field to the SAL of the sink. When a system receives any data containing its SAL that is not sent through the established conventions of this protocol, the data received should be ignored.

COMMENTARY

In the data transparent protocol, data files are identified by content rather than by ARINC 429 label. Thus, the label field loses the function of parameter identification available in broadcast communications. c-16

2.5.4 Bit Rate and Word Timing

Data transfer may operate at either high speed or low speed (O_2) as defined in Part 1 of Specification 429. The source should introduce a gap between the end of each ARINC 429 word transmitted and the beginning of the next. The gap should be 4 bit times (minimum). The sink should be capable of receiving the LDU with the minimum word gap of 4 bit times between words. The source should not exceed a maximum average of 64 bit times between data words of an LDU.

COMMENTARY

The maximum average word gap is intended to compel the source to transmit successive data words of an LDU without excessive delay. This provision prevents a source that is transmitting a short message from using the full available LDU transfer time (T_9). The primary value of this provision is realized when assessing a maximum LDU transfer time for short fixed-length LDUs, such as for Automatic Dependence Surveillance (ADS). c-14

If a Williamsburg source device were to synchronously transmit long length or full LDUs over a single ARINC 429 data bus to several sink devices, the source may not be able to transmit the data words for a given LDU at a rate fast enough to satisfy this requirement because of other bus activity. In aircraft operation, given the asynchronous burst mode nature of Williamsburg LDU transmissions, it is extremely unlikely that a Williamsburg source would synchronously begin sending a long length or full LDU to more than two Williamsburg sink devices. Although, a laboratory c-16 condition could be designed to test a Williamsburg transmitter which would likely result in the transmitter's failure to meet the maximum word gap requirement, this test should be disregarded. A failure to meet this requirement will either result in a successful (but slower) LDU transfer, or an LDU retransmission due to an LDU transfer timeout.

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.5.5 Word Type

The Word Type Field occupies bit 31-29 in all bit-oriented LDU words. See Table 11-1A of Attachment 11 for a description of the Word Type field. The Word Type field is used to identify the function of each ARINC 429 data word used by the bit-oriented communication protocol.

2.5.6 Protocol Words

The protocol words are identified with a Word Type field of "100" and are used to control the file transfer process.

2.5.6.1 Protocol Identifier

c-14 The protocol identifier field occupies bits 28-25 of the protocol word and identifies the type of protocol word being transmitted. Table 11-4 of Attachment 11 contains the different protocol words and their formats.

Protocol words with an invalid protocol identifier field should be ignored.

2.5.6.2 Destination Code

c-16 Some protocol words contain a Destination Code. The
c-18 Destination Code field (bits 24-17) typically indicates the
final destination of the LDU (O_{10}). If the LDU is intended
for the use of the system receiving the message, the
c-16 destination code may be set to null (hex 00). However, if
the LDU is a message intended to be passed on to another
on-board system, the Destination Code should indicate the
system to which the message is to be passed. Some
interfaces (e.g., between an ARINC 758 CMU and a multi-
c-18 bearer-system ARINC 761 SDU) use the "Destination"
code to select a specific bearer system to be used for a
downlink message, and to indicate the specific bearer
system used for an uplink message. The Destination Codes
c-16 are assigned according to the applications involved as
shown in Attachment 11A.

c-14 In an OSI environment, the Link layer protocol is not responsible for validating the destination code. It is the responsibility of the higher level entities to detect invalid destination codes and to initiate error logging and recovery.

COMMENTARY

Within the pre-OSI environment, the Destination Code provides Network layer information. In the OSI environment, this field may contain the same information for routing purposes between OSI and non-OSI systems.

2.5.6.3 Word Count

Some protocol words contain a Word Count field. The Word Count field (bits 16-9) reflects the number of ARINC 429 words to be transmitted in the subsequent LDU. The maximum word count value is 255 ARINC 429 words and the minimum word count value is 3 ARINC 429 words. A LDU with the minimum word count value of 3 ARINC 429 words would contain a SOT word, one data word and an EOT word. A LDU with the maximum word count value of 255 ARINC 429 words would contain a SOT word, 253 data words and an EOT word.

2.5.7 Request To Send (RTS)

When an on-board system needs to send a LDU to another on-board system, it will issue a Request To Send (RTS) to that system. The RTS word contains a Destination Code and a Word Count field.

When a system receives a RTS, it should send a response to the source within T_1 milliseconds. The response can be: (1) Clear to Send, (2) Not Clear to Send or (3) Busy.

To be considered valid CTS, NCTS or BUSY data words must have odd parity and contain the same destination code as the corresponding RTS. A valid CTS must also contain the same word count as the RTS.

2.5.7.1 Clear To Send (CTS)

When a system receives a valid RTS and is ready to accept the LDU transfer, it should send a CTS word to the source within T_1 milliseconds. The CTS contains a Destination Code (bits 24-17) and a Word Count field (bits 16-9). The Destination Code in the CTS should contain the same Destination Code as the RTS word (See Section 2.5.6.2). The Word Count field should contain the same Word Count value as the RTS word. If the source receives a CTS containing a different Destination Code or Word Count field value or a Word count field value equal to zero, it should treat it as a valid Not Clear To Send. All of the RTS counters (N_1, N_2, N_3) will be reset after a valid CTS is received.

c-14

2.5.7.2 Not Clear To Send (NCTS)

When a system either receives a valid RTS and is NOT ready to accept the LDU transfer or receives an RTS with an invalid destination or invalid word count, it should send a Not Clear To Send (NCTS) to the source within T_1 milliseconds. See Table 11-4 of Attachment 11 for the format of the NCTS word. The NCTS word should contain the same Destination Code as the RTS word and status code (bits 16-9) as shown in Attachment 11B indicating the reason for the busy response. If the NCTS received does not contain the same Destination Code, then the source should declare the NCTS to be invalid and ignore it. The status codes are for engineering purposes only and should be ignored by the system receiving the NCTS word.

Upon receipt of the NCTS word, the source should wait for T_2 milliseconds before repeating the RTS. The RTS may be repeated T_2 milliseconds after each NCTS until N_1 requests nominally have gone without receiving a valid CTS. The actual number of attempts (N_1) a system should make and the action to be taken when the limit is exceeded depend on the application (A_1). The NCTS counter (N_1) should be reset upon valid (CTS) response to the RTS.

After sending a NCTS, the sink may optionally choose (O_3) to send a CTS with the requested Destination Code and Word Count automatically as soon as it is ready to

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.5.7.2 Not Clear To Send (NCTS) (cont'd)

accept the file transfer, without waiting for another RTS. The source may optionally choose (O_4) to accept such a CTS. Alternatively, the source may ignore the CTS with the requested Destination Code and Word Count and repeat its RTS.

COMMENTARY

If the word count and/or destination fields in the received RTS word are not valid by virtue of illegal or unsupported values, the sink should respond with the NCTS word using an optional status code identifying the condition. Some original implementations simply ignored an invalid RTS word. However, the preference is to respond with a NCTS.

2.5.7.3 Destination Busy (BUSY)

c-14 When a system receives a valid RTS and is not able to accept an LDU within a timely manner, the receiving system may optionally send a BUSY response to the source within T_3 milliseconds. See Table 11-4 of Attachment 11 for the format of the BUSY data word. The BUSY word should contain the same Destination Code as the RTS word and a status code (bits 16-9) as shown in Attachment 11B indicating the reason for the busy response. If the BUSY received does not contain the same Destination Code, then the source should declare the BUSY to be invalid and ignore it. The status codes are for engineering purposes only and should be ignored by the system receiving the BUSY word.

COMMENTARY

A timely manner refers to the interval defined by the NCTS retry sequence.

Upon receipt of the BUSY word, the source should wait for T_4 seconds before repeating the RTS. The RTS may be repeated every T_4 seconds for as many times as the application requires up to a maximum of N_2 . Each new attempt could possibly consist of several RTS transmissions. Note that the busy condition could last for hours, depending on the nature of the application and the buffering capacity of the sink. The action taken following T_4 - N_2 time out depends upon the application (A_2). The application requirements may supersede the value of N_2 defined in Attachment 10. The Busy counter (N_2) should be reset upon valid (CTS) response to the RTS.

After sending a BUSY, the sink may optionally choose (O_3) to send a CTS with the requested Destination Code and Word Count automatically as soon as it is ready to accept the file transfer, without waiting for another RTS. The source may optionally choose (O_4) to accept such a CTS.

Alternatively, the source may ignore the CTS with the requested Destination Code and Word Count and repeat the RTS.

c-16 When expedited file transfers are desired between 2 BOP devices, the sink device should support Option 3 (Send Auto CTS) and the source device should support Option 4 (Accept Auto CTS).

COMMENTARY

If Option 3 (Send Auto CTS) and Option 4 (Accept Auto CTS) cannot be supported between two devices where expedited file transfers are necessary, an alternative approach is to reduce the BUSY Retry Timer (T_4) and increase the BUSY counter (N_2) c-16 accordingly to make the T_4N_2 timeout period identical to the existing T_4N_2 value. The recommended value for low speed is T_4=1.5 seconds and N_2=200.

2.5.7.4 No Response to RTS

If the source receives no response to the RTS within T_5 milliseconds, the request should be repeated. In the absence of any valid response (such as CTS, NCTS, or BUSY), or the absence of an unexpected RTS, the RTS should be repeated every T_5 milliseconds until at least N_3 requests have gone unanswered. Any response other than a valid CTS, NCTS, BUSY, Aloha, or an unexpected RTS should be ignored. The No response counter (N_3) should be reset c-14 upon valid response to the RTS. All the RTS counters (N_1, N_2, N_3) will be reset after a valid CTS is received.

The actual number of attempts a source should make (N_3) before giving up, or taking some different course of action, when the limit is exceeded depends on the application (A_3). The action to be taken is described in Table 10-2 of Attachment 10 or in the applicable equipment characteristic.

2.5.8 Conflicting RTS Transmissions

2.5.8.1 Half Duplex Mode

When operating in half duplex mode, it is possible that two systems might decide to send RTS messages to each other at nearly the same time, causing each system to appear to receive the other's RTS in response to its own RTS. If this occurs, each system should set a random timer to a time in the range of zero to T_6 milliseconds in increments of no more than T_7 milliseconds. If a system receives another c-12 RTS before this timer expires, that system will defer its own need to transmit and will respond to the other system's RTS as defined in the preceding paragraphs. If no RTS is received within the random time, the system should re-transmit the RTS. If a conflicting RTS occurs again, the same procedure will take effect for as many times as it takes for one system to prevail over the other.

If this protocol is used in an environment that has well defined priorities (O_5), one system may be assigned priority over another to resolve RTS conflicts without the random retransmission procedure described above.

COMMENTARY

Typically, a well-defined priority in avionics gives an RTS for uplinks priority over an RTS for downlinks. c-16

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.5.8.2 Full Duplex Mode

When operating in a full duplex mode, both systems must be capable of operating as a source and sink at the same time. If both systems initiate an RTS, both should receive a CTS and both should respond normally to the CTS. Neither system should abort.

A conflicting RTS applies only to a receiving system that has transmitted a CTS (in response to a RTS) and receives another RTS. This RTS is treated simply as a retransmission and the sink should retransmit a CTS.

2.5.9 Unexpected RTS

c-13 It is possible, that after sending a CTS word to a requesting source, that the source does not receive the CTS and re-transmits the RTS to the sink. Alternatively, a source may experience a reset which causes a new RTS to be sent in the middle of an LDU transfer. If, for any reason, the first word received by a sink after having sent a CTS word is an RTS, then the sink should transmit another CTS word. If a sink receives another RTS after having sent a CTS, even in the middle of receiving an LDU, the sink should discard any partial LDU already received and respond as defined in Section 2.5.7.1.

2.5.10 Start Of Transmission (SOT)

When a system receives a valid CTS with a Destination Code and Word Count matching the Destination Code and Word Count of the previous RTS, the system should respond by sending the Start of Transmission (SOT) word within T_{13} milliseconds, immediately followed by the data words which constitute the LDU. See Table 11-6 of Attachment 11 for a description of the SOT word format. The SOT word contains the File Sequence Number in its File Sequence Number field (bits 24-17). It also contains a General Format Identifier (GFI) and a LDU Sequence Number.

c-14

2.5.10.1 General Format Identifier (GFI)

The General Format Identifier (GFI) occupies bits 28-25 in the SOT word. See Table 11-6A of Attachment 11 for a description of the GFI field. The GFI is transparent to the Link layer protocol. It is designated by a higher level entity in the source device, to indicate to a higher level entity in the sink, the format of the data words that follow. It is the responsibility of the higher level entities to detect invalid GFI designations and to initiate error logging and recovery.

COMMENTARY

c-14 Within the pre-OSI environment the GFI provides Network layer function information. In the OSI environment this field may contain the same information for bridging purposes between the OSI and non-OSI world.

c-12 A code of 1111 is used to indicate that an extended GFI of 8 bits will be found in the first data octet (Nibbles 1 and 2) of the first data word in the file.

2.5.10.2 File Sequence Number

The File Sequence Number (bits 24-17) of the SOT word contains an 8-bit number assigned to the file. It is initialized to the hex value 00 and increments by 1 for each new file that is sent over the ARINC 429 link. After reaching hex FF, the File Sequence Number should start over at hex 01, skipping zero. A file consisting of multiple LDUs will have the same File Sequence Number in each of the SOT words of each LDU.

2.5.10.3 LDU Sequence Number

The LDU Sequence Number (bits 16-9) of the SOT word contains an 8-bit number assigned to the LDU. It is initialized to the hex value 00 and increments by 1 for each new LDU of the same file that is sent over the ARINC 429 link. The LDU Sequence Number should be reset to 00 at the beginning of each new file.

2.5.11 Data

Data words immediately follow the SOT word. The octets of the data file are encoded and transmitted in 32-bit data words. There are two basic types of data words: full binary and partial binary. Binary data words may contain one, two, three, four and five semi-octets. A semi-octet (or nibble) is half of an octet, or four bits in length. c-12

Binary data words of five semi-octets are called Full Data words. Binary data words of fewer than five semi-octets are called Partial Binary Data words.

A data file may be sent using any combination of full and partial data words. There are no restrictions regarding the particular type of data within a file. The formats of Data words are illustrated in Attachment 11.

Each binary data file, prior to transmission, should conclude (end) with a complete octet. Any incomplete final octet should be completed with zeros before transmission. Each LDU transmitted, should also end in a complete octet. If, at the end of the transmission, the receiver determines that an odd number of semi-octets has been received, that is, the LDU ends with an incomplete octet, the receiver should send a NAK or assume the upper 4 bits of the partial octet to be zeros, and proceed normally.

2.5.11.1 Full Data Word(s)

A Full Data Word has 20 bits available for data. This space is allocated in five semi-octets. The octets of the data file are divided into two semi-octets and placed sequentially into the data words. The least significant bit of the least significant semi-octet is sent first. c-13

If, in the process of placing the octets into the data words, an octet is split between two different words, the least significant semi-octet goes in the last (n_5) semi-octet of the current data field and the most significant semi-octet follows in the first (n_1) semi-octet of the data field of the next word. See Attachment 11 for data word formats.

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.5.11.1 Full Data Word(s) (cont'd)

If the end of the LDU does not completely fill the last Full Data Word, a Partial Data Word (see Section 2.5.11.2) should be used to complete the LDU transmission.

c-13 In order to obtain maximum system efficiency, Partial Data Words should be used only when they are required to complete the data for an LDU transmission. An LDU should not be encoded as a string of Partial Data Words where 8 or 16 bits of data are encoded for each ARINC 429 word used.

2.5.11.2 Partial Data Word(s)

c-16 A Partial Data Word contains from one to four semi-octets. The number of semi-octets in a partial data word is indicated in bits 28-25 of the word. Only full four-bit semi-octets can be sent; one, two or three bits are invalid. Partial Data Word semi-octets are sent in the same order as the semi-octets in a full data word. The unused semi-octets in the partial data word should be set to binary zeros.

2.5.12 End of Transmission (EOT)

c-14 Each LDU transfer is terminated by an End of Transmission (EOT) word. See Table 11-7 of Attachment 11 for the definition of this word. Table 11-7A of Attachment 11 contains the definition for bit 25 of the EOT word. This bit is used to indicate if the LDU is the final LDU of the file transfer. If the file transfer consists of a single LDU, bit 25 should be set to 1 to indicate that this is the final LDU. If N LDUs are to be sent, then bit 25 of LDU 1 through LDU N-1 should be set to 0 and bit 25 of LDU N should be set to 1. The EOT word contains a Cyclic Redundancy Check or CRC (bits 24-9).

COMMENTARY

c-12 The ARINC 429 data link is a twisted shielded pair of wires which has been demonstrated to exhibit high integrity and unlikely to introduce errors into the data passing through it. Simple parametric data is usually transmitted at a refresh rate high enough to permit recognition and suppression of erroneous data. Since the transfer of data using a file transfer protocol contains no provision for automatic refresh, some applications may require high data integrity to be confirmed by an error checking mechanism. For this reason, each LDU contains a CRC check. The use of the CRC in this case does not imply any inherent lack of integrity of the ARINC 429 link.

2.5.12.1 CRC Encoding

The CRC field is a 16 bit sequence with the most significant bit (MSB) transmitted first. Determination and encoding of the CRC is as follows:

c-12 The k bits of data in the LDU are represented as the coefficients of the polynomial, G(x); where k is the number of data bits in the LDU existing between, but not including, the SOT and EOT words. For example, if the data stream is 101001, k=6 and $G(X)=x^5+x^3+1$.

The CRC calculation is performed over the data octets only of the LDU with any semi-octets zero filled.

There exists a generator polynomial which is of the form,

$$P(x) = x^{16} + x^{12} + x^5 + 1$$

The CRC is then determined as the one's complement of the remainder, R(x), obtained from the modulo 2 division of:

$$\frac{x^{16}G(x) + x^k(x^{15}+x^{14}+x^{13}+...+x^2+x+1)}{P(x)} = Q(x) + \frac{R(x)}{P(x)}$$

Note: The addition of $x^k(x^{15}+x^{14}+x^{13}...+x^2+x+1)$ to $x^{16}G(x)$ (which is equivalent to inverting the first 16 bits of G(x) and appending a bit string of 16 zeroes to the lower order end of G(x)), corresponds to initializing the initial remainder to a value of all "ones." The complementing of R(x), by the transmitter, at the completion of the division ensures that the received, error-free message will result in a unique, non-zero remainder at the receiver.

At the transmitter, the CRC is added to the $x^{16}G(x)$ product, resulting in the message, M(x), of length n where:

$n = k+16$, c-12

and $M(x) = x^{16}G(x) + \overline{R(x)}$

$= x^{16}G(x) + CRC$

2.5.12.2 CRC Decoding

Decoding of the CRC at the receiver is as follows:

At the receiver, the incoming M(x) is multiplied by x^{16}, added to the product,

$x^n(x^{15}+x^{14}+x^{13}+...+x^2+x+1)$
and divided by P(x) as follows:

$$\frac{\overline{x^{16}M(x)+x^n(x^{15}+x^{14}+x^{13}+...+x+1)}}{P(x)} = Qr(x) + \frac{\overline{Rr(x)}}{P(x)}$$

If the transmission of the serial incoming bits plus CRC (i.e., M(x)) is error free, then the remainder, Rr(x) will be 0001110100001111 (coefficients of x^{15} through x^0, respectively). A mathematical example of CRC encoding and decoding can be found in Appendix G.

COMMENTARY

The notation used to describe the CRC is based on the property of cyclic codes that a code vector such as 1000000100001 can be represented by a polynomial $G(x) = x^{12} + x^5 + 1$. The elements of an n element code word are thus the coefficients of a polynomial of order n - 1. In this application, these coefficients can have the value 0 or 1 and all polynomial operations are performed module 2. The polynomial representing the data content (message) of an LDU is generated using the LDU bit which is encoded in bit 9 c-12 of the first data word as the coefficient of the highest order term. A mathematical example of CRC encoding and decoding can be found in Appendix G.

2.5.13 Negative Acknowledgment(NAK)

c-14 If the sink detects any of the errors described in the following subsections, it sends a NAK to the source upon detecting the error or within T_8 milliseconds of receiving the EOT word, whichever occurs first. See Table 11-4 of Attachment 11 for a description of the NAK word format. The NAK word should contain the same File Sequence Number (bits 24-17) as the SOT word and a status code (bits 16-9) as shown in Attachment 11B indicating the reason for the NAK. The File Sequence Number and status code are intended to be used for engineering purposes only and should be ignored by the system receiving the NAK.

c-13 The sink should test for errors to determine if a NAK should be sent. It is not necessary for the sink to be aware of the type of error that occurred, as long as any of the errors listed in the following subsections will elicit a NAK response.

2.5.13.1 Missing SOT Word

c-12 Following reception of a valid CTS word, the source should transmit the SOT word as the first word of the LDU as specified in Section 2.5.10, Start of Transmission (SOT). If the SOT word is not received as the first word of the LDU, the sink should send the NAK response. See also Section 2.5.9.

2.5.13.2 Missing EOT Word

c-13 Following the transmission of the final data word of each LDU, the source should transmit the EOT word as the final word of the LDU as specified in Section 2.5.12, End of Transmission (EOT). If the EOT word is not received as the final word of the LDU within T_9 seconds of the CTS, then the sink should send the NAK response to the source within T_8 milliseconds after the T_9 has expired.

2.5.13.3 Parity Errors

c-12 Bit 32 of each ARINC 429 word should be set to odd parity for the entire word. Upon receipt of a word, the receiving unit should verify that each word was received with odd parity. If any word is received with even parity, the receiving unit should take no action and ignore the word.

COMMENTARY

c-12 When the sink receives a word with bad parity, it cannot be sure of the intended label. The word may not even be a part of the LDU, so by ignoring the offending word there may still be a chance of a successful file transfer. If the offending word was intended to be a part of the LDU, then when the EOT word is received, the actual word count will not match the expected count. The sink will either NAK the source when the EOT word is received, or when it times out waiting for the full expected number of words.

2.5.13.4 Word Count Errors

Upon receipt of the EOT word, the sink should verify that the actual number of words received is the number of words expected, per the RTS and CTS words. If the word counts do not match, the sink should send a NAK response to the source. c-14

2.5.13.5 CRC Errors

Upon receipt of the EOT word, the sink should verify the CRC on the received LDU. If the 16-bit CRC is invalid, the sink should send the NAK response. c-13

2.5.13.6 Time Out Errors

The sink will not time the gaps between the words received, However, if the sink does not receive the complete LDU transfer within T_9 seconds of having sent the CTS, it should send a NAK to the source and discard any partial LDU received.

2.5.14 LDU Transfer Acknowledgment(ACK)

If all words of the LDU transfer are received within T_9 seconds of the CTS, each with odd parity, and the word count and CRC verify, and the LDU is either the next LDU, a duplicate LDU or the first LDU of a new file, then the sink should send an acknowledgment(ACK) to the source within T_8 milliseconds of receiving the EOT word. See Table 11-4 of Attachment 11 for a description of the ACK word format. The ACK word should contain the File Sequence Number (bits 24-17) and LDU sequence number (bits 16-9) to indicate a successful LDU transfer. c-12

2.5.14.1 Duplicate LDU

An LDU is determined to be a duplicate if its File Sequence Number and LDU Sequence Number are not both zero, and it's SOT words are identical to those of the previously received LDU. When a duplicate LDU is detected, the sink should discard the LDU just received and acknowledge per Section 2.5.14. c-16

COMMENTARY

Some implementations look at both SOT and EOT to determine duplicate LDUs.

COMMENTARY

A File Sequence Number of zero along with an LDU Sequence Number of zero should be interpreted as an indication of a reset in the source and the LDU should not be compared to the previous one.

2.5.14.2 Auto-Synchronized Files

c-12

When the File Sequence Number is different from the previous LDU transfer and the LDU Sequence Number is zero, then the sink discards any previously received partial file, and accepts the LDU just received.

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.5.14.2 Auto-Synchronized Files (cont'd)

COMMENTARY

c-12 Some implementations do not support auto-synchronized files.

2.5.14.3 Incomplete File Timer

The Incomplete File (T_{14}) timer is used to insure that a source device may not "lock-up" a sink with an incomplete file.

For multiple-LDU files, the maximum time allowed from transmission of the ACK or NAK for a previous LDU to the reception of the next RTS should be T_{14} minutes. The sink will start the T_{14} timer when it sends a NAK or when it sends an ACK for each LDU of a file except the last LDU.
c-14 The T_{14} timer is stopped each time another valid RTS is received. The T_{14} timer should also be stopped if the sink discards the partial file for other reasons defined in this specification. When T_{14} minutes is exceeded, the sink should send the SYN word within T_8 milliseconds and discard any partial file already received.

For Half Duplex environments (i.e., Option O_1 of Table 10-3) this timer only applies when the device is operating as a sink.

2.5.15 SYN Word

The SYN word is used by the sink to inform the source that it (the sink) has become confused concerning the construction of the file. A typical example is an inappropriate or unexpected File/LDU Sequence Number in
c-13 the SOT word. See Table 11-4 of Attachment 11 for a description of the SYN word format. Upon receipt of the SOT word, the sink should verify the LDU Sequence Number (bits 16-9).

COMMENTARY

If an SOT word of an LDU has the same File Sequence Number and LDU Sequence Numbers as the previous LDU, some implementations will interpret this as an unexpected File/LDU Sequence Number instead of a duplicate LDU.

c-16 If the LDU is not a duplicate and its LDU Sequence Number is not the next in sequence (i.e., is not the next LDU), and is not the first LDU of a new file, then the sink should send the SYN response to the source within T_8 milliseconds of receipt of the SOT word, or within T_8 milliseconds of receipt of the EOT word when necessary to determine a duplicate LDU, and discard any partial file already received.

COMMENTARY

Some implementations look at both SOT and EOT to determine duplicate LDUs.

The SYN word may be sent without timing constraint when
c-13 the received data words cannot be normally processed. There is no acknowledgmentdefined for the source when a SYN word is received. The action taken is specified in c-13
Section 2.5.16.

2.5.16 Response to ACK/NAK/SYN

The source should expect a response to the transmission c-14
within T_{16} of sending the EOT .

COMMENTARY

Timer T_{10} may apply instead of T_{16}, for those devices compliant with version "0000" or "0001" of this protocol (Table 11-4A of Attachment 11). T_{10} is started c-13
after the CTS is received whereas time T_{16} is started after the EOT word is transmitted.

If an ACK is received and the File Sequence Number and LDU Sequence Number match the contents of the SOT word, the source should consider the transfer successful. If an ACK is received and the File Sequence Number and/or the LDU Sequence Number do not match the SOT word, or if a NAK is received or if no response is detected, the source should repeat the entire LDU transmission process, starting with the sending of the RTS word. If the source c-14
detects a SYN during the transmission process, it should retransmit the entire file, if able, beginning with the first LDU. The File Sequence Number contained within the SYN word is provided for testing purposes and is not evaluated by the source. If the source is not able to retransmit or rebuild the entire file, that file is discarded and the source proceeds normally with the first LDU of the next file, when it is ready for transmission.

COMMENTARY

If end-to-end accountability is required, then either upper layer OSI protocols or the application process should provide this capability.

Each new attempt to re-transmit the LDU should start with the necessary RTS transmissions in order to obtain a CTS again. After receiving N_4 consecutive NAK words, or after receiving N_5 consecutive SYN words, the file transfer between the two systems should be considered failed.

2.5.17 Solo Word (Single Word Transfers)

If the data to be sent consists of 1 or 2 octets only, (e.g., a button code from a CDU keyboard) then it is not necessary to obtain a CTS. In this case the data may be sent "in the c-12
blind" using the Solo Word format (O_8). The Solo Word contains a 16-bit data field in bits 24-9 and a 4-bit identifier in bits 28-25 to identify the nature of the data. For example, the I.D. may indicate that the data is a key code from a CDU or a status word. Codes 0000 and 0001 are reserved for the TEST and LOOP words as defined in Section 2.5.17.1. All other codes are available for application use. Solo Words are not acknowledged at the link level. However, they may invoke a Solo Word or data file transfer response as required by the application. Solo Words cannot be interleaved with data file words during a data file transfer. If error detection beyond parity is required, some bits of the data field can be defined as check bits, to be verified by the application.

2.5.17.1 Test Word and Loop Word

The integrity of the ARINC 429 link between two units may be tested. The test is initiated by the source sending the Loop Test Pattern Word (TEST).

The TEST word contains a 16-bit binary pattern (bits 24-9) chosen by the originator. The system receiving the TEST Word should respond by sending a Loop Test Response (LOOP) word containing the same 16-bit pattern (bits 24-9) within T_{11} milliseconds.

COMMENTARY

The preferred reaction to a loop test failure has not been defined.

2.5.17.2 Optional Solo Word Definitions

c-12 Each equipment utilizing this bit-oriented protocol may define solo words as needed. See Table 11-5 of Attachment 11 for the format definition. These solo word definitions should be unambiguous.

COMMENTARY

The same SOLO word ID coding may be usedrepeatedly in different units as long as its meaning remains unambiguous.

2.5.18 Optional End-To-End Message Verification

In some applications, an end-to-end integrity check is desirable in order to validate the correct transfer of a data file from the message source to its final destination. For further information, the reader should refer to the applicable ARINC Equipment Characteristic, e.g., ARINC Characteristic 702, "Flight Management Computer System."

2.5.19 Protocol Initialization

The ALO word should be sent by any system which supports the bit-oriented Link layer protocol just after the system powers-up, or performs a re-initialization for any reason.

c-14 A system which supports the bit-oriented Link layer protocol should first determine if the interfacing device also supports the bit-oriented protocol using the ALO/ALR process described in Section 2.5.19.1. If the system is bilingual and there is no response to the ALO/ALR process, it may also determine if the interfacing device supports the character-oriented protocol as described in Section 2.5.19.2.

c-16 A bilingual system should repeat the processes described in Sections 2.5.19.1 and if applicable 2.5.19.2, until a common protocol version is selected by both systems. Examples of protocol initialization are given in Attachment 13.

COMMENTARY

c-14 In addition to a "power-reset" or a system "re-initialization," a device that supports the bit-oriented protocol may at any time determine the ARINC 429 link status using the ALO/ALR process described in Section 2.5.19.1 and Section 2.5.19.2. c-14

2.5.19.1 Bit-Oriented Protocol Version

The ALO/ALR process is intended to be used when a system needs to determine whether or not an interface supports the bit-oriented protocol. To maintain interoperability, all systems which support the Link layer Bit-Oriented Protocol must be able to respond to the initialization of this process. Attachment 11, Table 11-4, shows the ALO and ALR word formats. c-16

When a system with a bit-oriented Link layer protocol has the need to make this determination, it should construct the ALO word and transmit this word to the device in question.

The system should then wait for a maximum period of time defined by T_{12}. If the device in question has not responded within time T_{12}, the initiating system should initiate another ALO word and again delay up to T_{12}. The initiating system should attempt a maximum of N_6 ALO word operations before declaring the device in question as "Not bit-oriented" or "Not able to respond."

2.5.19.1.1 ALOHA

The first ALOHA word transmitted in a sequence should contain the highest Version Number supported by the source device. If the ALOHA Response contains a version that does not match the ALO version, the source device should take one of the following actions: c-14

a. If the source device is able to adapt to the differences in protocol version, file transfers may proceed using the protocol version identified in the ALR word.

b. If the source device is not able to adapt to the differences in protocol version, the source should again initiate the ALOHA word with the version field set to the highest version supported by the source that is lower than the version indicated in the previous ALR (see examples identified in Attachment 13A).

The ALO/ALR protocol determination process should continue until a common protocol version is found. If none of the protocol versions match, the source should notify the higher level entity of the communications failure and continue the protocol determination process.

When the system only supports bit oriented protocols it should repeat the process described in this section. When the system also supports character oriented protocols then the process defined in 2.5.19.2 should be followed. The protocol determination process will continue until a common protocol is found. See Attachment 17. c-16

The ALOHA word should contain a Subsystem SAL field as shown in Attachment 11, Table 11-4. This field should contain the SAL of the device sending the ALOHA word, with bit 17 as the most significant bit and bit 24 as the least significant bit of the Subsystem SAL.

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.5.19.1.2 ALOHA Response

A device that supports the bit-oriented Link layer protocol should always be able to respond to the receipt of the ALO word. Whenever a device receives the ALO word, it should leave its present task and respond within T_{15} with an ALR word.

c-14 The ALR response should reflect the device's protocol version level by way of the Version Number contained within the ALR word. If the Version Number of the ALO does not match the sink device's protocol Version Number, the sink device should select the next lowest version supported (equal to or lower than the version indicated by the previous ALO word) and indicate this new version in the ALR word.

After the device has responded to the ALO word (with the ALR), the device should take the following action:

If the device was in the sink mode (had already begun receiving any LDU of a file), it should discard any partial file it had received. Since the ALO represents a system "re-initialization" (per Section 2.3), the source may reset the File and LDU sequence numbers.

If the device was in the source mode (had already begun transmitting any LDU or file), it should resend the entire file beginning with the first LDU of the file.

c-16 If Option O_{12} is selected for a particular bit-oriented protocol interface, the device receiving an ALOHA word should use the Subsystem SAL from the ALOHA word as the SAL of the ALOHA Response (ALR) word and for all subsequent bit-oriented protocol transmissions for that interface.

COMMENTARY

c-16 A bit-oriented Link layer protocol device receiving an ALOHA word should take caution if utilizing the Subsystem SAL field of the ALOHA word to determine how to send the ALR reply. Previous supplements of the bit-oriented Link layer protocol defined the Subsystem SAL field as [TBD] bits, and as such, some devices in service may have encoded non-zero bits in this field. A receiving system should therefore implement a reasonableness check to validate that the data bits received in this field represent a SAL associated with a known bit-oriented Link layer protocol device. If a device receives an ALOHA word containing a System SAL of all zeros or an unknown SAL, and the ALOHA word is received on a data bus that has a specific bit-oriented Link layer protocol subsystem associated with it, it is recommended that the ALR word be attempted using the SAL for that specific subsystem, to ensure interoperability.

2.5.19.2 Williamsburg/File Transfer Determination

c-14 In some situations there may be a transition period from a device using the character-oriented file transfer protocol defined in ARINC Specification 619 to the same device using the Williamsburg bit-oriented protocol introduced in Supplement 12. In this situation, it is desirable to have an "automatic" determination sequence which allows the devices to adjust from the character-oriented to bit-oriented protocol. Because of the environment in which these devices are to operate, it is necessary to have a cyclic process to establish the protocol to communicate with. An attempt to establish contact using the bit-oriented ALO-ALR words should be made to the point where the link is considered to be failed or to be established.

If the link has been established, then normal bit-oriented communications can be pursued. If the link is considered failed, then an attempt to obtain a response to the character-oriented RTS-CTS words should be made. The typical repeat sequence for the character-oriented protocol is 3 tries (See Attachment 13). If a character-oriented CTS, NCTS (CTS 0), or BUSY (CTS Q) response is obtained, then the link is considered established and normal character-oriented communications can be pursued. If the character-oriented protocol fails, then the cycle should be re-started using the bit-oriented ALO sequence. There may be a period of inactivity between cyclic attempts of not more than 15 seconds. Attachment 17 contains an example diagram of these determination sequences. c-14

If a protocol has been established and the link fails because of a loss of activity (if defined on that bus), or because of a failure to deliver a message due to a no response, then the protocol determination sequence should be started again. A graphical representation of this is presented in Attachment 17.

COMMENTARY

This determination sequence is only necessary when there is a possibility of having to support both the older character-oriented protocol and the newer bit-oriented protocol on the same ARINC 429 data bus.

COMMENTARY

During the protocol determination process, a bilingual device should recognize protocol words using both the bit-oriented (Williamsburg) and the character-oriented formats. It is recommended that at least the ALOHA word and the character-oriented RTS and NAK words be recognized by the device capable of the automatic protocol determination. This would minimize synchronization problems between the two communicating devices and allow the Link layer method to be established in a timely manner.

COMMENTARY

When an LRU performs the character protocol determination (RTS) it is initiating the sequence of events for a file transfer, but it does not complete the file transfer. Some LRUs will wait forever for the file transfer to be completed unless the source sends a NAK word to terminate the file transfer. Therefore, it is recommended that protocol determination logic which includes character protocol should transmit a character oriented NAK word when a character oriented CTS word is received in response to a character oriented RTS word. c-16

2.0 BIT-ORIENTED FILE TRANSFER PROTOCOL

2.6 Windowed Bit-Oriented Communications Protocol

This section has been deleted.

c-16 Version 2 of the bit-oriented (Williamsburg) Protocol, previously in this section, has been superseded by Version 3 of the bit-oriented (Williamsburg) Protocol defined in Section 3.0 of this document.

3.0 BIT-ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL

3.1 Bit-Oriented Media Access Control (MAC) Protocol

3.1.1 Introduction

This section describes Version 3 of the bit-oriented (Williamsburg) protocol. It is an IEEE 802 compliant MAC protocol for frame oriented data transfer using the Data Link layer.

Within the ISO/OSI Reference Model, the Data Link layer is responsible for the logical connection between at least two entities of the same Local Area Network (LAN). It interfaces to the Network layer, which is responsible for the connection between at least two networks, and the Physical layer, which is responsible for the physical connection between adjacent nodes, i.e. repeaters, bridges or computers. The Data Link layer is further divided into two sublayers.

The Media Access Control (MAC) sublayer, which is a Physical, layer dependent entity. The purpose of the MAC is to provide a standard interface to the entities above it. The MAC isolates the sublayer above it from the details of the Physical layer. The ARINC 429 MAC is defined herein.

c-17 The Logical Link Control (LLC) sublayer is assumed by IEEE/ISO as the second sublayer, which is a physically independent control sublayer.

For further description of the OSI protocol layers please refer to Section 2.5 of this document.

COMMENTARY

Generally, multiple MAC Service Clients (users) interface with the MAC entity (MAC Service Provider). In the IEEE/ISO set of standards the LLC is the most prominent one. Additionally, the bridging function may act as a MAC Service Client as well as a MAC Control function. In the non-OSI world there are a variety of different protocols that may or may not directly interface with the MAC. Here, the most prominent one is Internet Protocol (IP). LLC Service Providers are required for some classes of IEEE 802.3 Local and Metropolitan Area Networks, specifically where the MAC frame contains insufficient information for protocol selection, or when required by higher layer protocols (specifically OSI based protocols such as ATN). This document follows the guidance of IEEE 802.3 where the Length/Type field is used to select between LLC as a MAC Client, and where the Network Layer is selected as the MAC Client. The selection of MAC Client (LLC or non-LLC frame format) is mutually exclusive.

In order to make ARINC 429 compatible with standard data buses adopted by the Institute of Electrical and Electronics Engineers (IEEE) and the International Organization for Standardization (ISO) it is paramount to support their common functionality and interfaces which are constituted by the MAC.

In order to facilitate bridging between this version of the protocol and other IEEE data buses the following subsections are closely modeled after IEEE 802.3 (Ethernet).

Throughout Section 3 of this document the following terms are being used:

MAC Sublayer: This term refers to the abstract definition of a layered communication stack in which the MAC Sublayer is part of the Data Link layer.

MAC Entity: Implementation of the functionality described for the MAC Sublayer

MAC Service Provider: This term can be used interchangeably with MAC entity and is supposed to provide context to the Client/Server nature of the layers of the communication stack.

MAC Service Client: Any entity (implementation) that uses the services of the MAC entity (implementation)

MAC Frame: Logical representation of the information structures exchanged between peer MAC entities. A detailed description of the structures are given in Section 3.3

Frame Data Unit (FDU): An ARINC 429 envelope that contains a MAC frame as well as specific control information. A detailed description of the structures is given in Section 3.4. c-17

3.1.2 Relationship Between Version 1 And Version 3 Protocols

The bit-oriented MAC protocol (Version 3) is derived from the full-handshake (Version 1) bit-oriented protocol (BOP) defined in Section 2.5 of this specification. Version 3 is presented to MAC Service Clients that do not require or desire the more exhaustive Data Link layer transfer validation provided by the Version 1 protocol.

The Version 3 protocol assumes a sufficiently robust ARINC 429 physical layer. All valid frames received by the sink are passed up to the MAC Service Client. The demands for buffering are much greater for Version 3 than for Version 1. In Version 1 the transmitter is responsible for buffering, in Version 3 the receiver is responsible.

The Version 1 ARINC 429 LDU full and partial data words have been retained. New SOF and EOF words have been defined for Version 3 (see Sections 3.4.6 and 3.4.8) which replace the Version 1 SOT and EOT words. The bit-oriented MAC protocol does not use the RTS, CTS, NCTS, BUSY, SYN, ACK and NAK words. Only single Frame Data Unit (FDU) transmissions are supported. Duplicate FDU detection is not performed by the MAC. Any segmentation and reassembly, if necessary, should take place above the ARINC 429 MAC sublayer. The terms Frame Data Unit (FDU) and MAC frame are defined in Sections 3.3.1, 3.3.2, and 3.4.1.

The Version 3 bit-oriented MAC protocol specifies full duplex operation (O_1) to allow simultaneous data transfers in both directions. The Version 1 bit-oriented (Williamsburg) protocol typically operates in half-duplex mode only.

3.1.3 Protocol Architecture

The MAC protocol described herein can support MAC service clients that are part of the Data Link layer such as LLC and can support MAC service clients in higher layer entities such as the Network Layer. A protocol architecture diagram is used to illustrate this relationship (See Figure 18-1 of Attachment 18).

3.1.4 Buffering

The MAC sublayer should provide sufficient buffering to account for expected processing/queuing delays. Some provisions for flow control are provided through the MAC sublayer (see Section 3.2.4). Flow control is primarily the responsibility of the MAC service client or a higher level entity.

COMMENTARY

> Each layer should have sufficient buffering for expected processing/queuing delays. In order to accurately estimate
> c-17 the amount of buffering needed, each layer will need to specify the maximum allowed delays. However, if the buffering capacity is exceeded, new FDUs are discarded.
>
> There is a non-zero time delay between the time when a FDU arrives at the ARINC 429 MAC sublayer and the time when the MAC Service Client processes that same frame. Any flow control, introduced between two systems, is provided by entities above the MAC service entity sending messages to each other. For this reason, the entity responsible for flow control should attempt to anticipate the need to activate flow control in order to accommodate the delays.
>
> The entity responsible for flow control should attempt to have adequate buffering capacity to handle all FDU transfers received from the time it exerts flow control to the time when the source flow control entity stops transmitting. This will only be possible if the maximum allowed delay in each of the various components is specified. These delays include any processing or queuing delays introduced by communications with the MAC Service Client.
>
> Failure to coordinate the requirements (i.e. delays) will make it difficult to consistently provide adequate buffering capacity and may lead to data loss and possible communication failures.

If buffering of incoming frames is done at the MAC sublayer, then the reception status parameter of the MAC primitive, MA_DATA.indication (see Section 3.2.3) should be used to inform the MAC Service Client of a buffering problem for the received frame.

COMMENTARY

> ISO standards documents do not explicitly specify at which (sub) layer buffering is accomplished.

3.2 Media Access Control (MAC) Sublayer

This section describes the various aspects of the MAC sublayer functionality. The MAC Sublayer Service Specification (Section 3.2.1) details the procedures in an abstract way (via service primitives) that provide the communication service to the MAC Service Client. The subsequent section, MAC Frame Structures (Section 3.3), defines the MAC frames that will be presented to and expected from the peer MAC entity. The next section, MAC Transmit/Receive Functions (Section 3.4), is related more to the interface with the particular ARINC 429 transmission and reception process itself and specifies how the FDUs are structured as a sequence of ARINC 429 words.

There are two types of applications that have driven the development of this version of the protocol. One is the need for a bridgeable protocol that can transfer data between an ARINC 429 bus and a non-ARINC 429 based data bus. This type of application would need fast file transfer traffic that is focused upon aircraft-wide topology and would utilize an independent exchange of MAC frame-based information.

The other need is to transfer local, ARINC 429 only,
fast file transfer traffic that is focused upon exchanges c-17
between two closely cooperating systems. In order to allow for the optimized transmission of the two types of information, two specialized MAC frame structures have been defined, the Information frame and the Command frame. The Information frame is intended to be bridgeable to IEEE 802.3 (Ethernet). The Command frame is not bridgeable. Option 13 (O_{13}) in Table 10-3B of Attachment 10 allows for the apriori selection of frame type, based on the applicable equipment interface specification.

From a transmitter's point of view, i.e., the originating MAC Service Client, one parameter within the MA_DATA.request primitive selects which type of MAC frame to create (see Section 3.2.2.2.4).

From the receivers point of view, i.e., the receiving MAC Service Client, the type of frame (Information or Command) received is indicated through a parameter in the MA_DATA.indication primitive, based on the contents of the SOF word (see Sections 3.2.3.2.7 and 3.4.6).

The Version 3 Information frame format facilitates bridging between this version of the ARINC 429 protocol and IEEE data buses. The following subsections are closely modeled after IEEE 802.3 (Ethernet).

The Version 3 Command frame format facilitates the exchange of command/response pairs, which are peer-to-peer only and hence do not require the additional addressing capability provided by the Information frame.

3.2.1 MAC Sublayer Service Specification

This section describes the services that the Media Access Control Sublayer provides to the next higher layer, i.e., to the MAC Service Client. The services

3.0 BIT-ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL

3.2.1 MAC Sublayer Service Specification (cont'd)

are described in an abstract way and do not imply any particular implementation or any exposed interface. There is not necessarily a one-to-one correspondence between the primitives described herein and the implementation.

Four primitives are used to describe this interface. They are:

MA_DATA.request
MA_DATA.indication
MA_CONTROL.request (optional)
MA_CONTROL.indication (optional)

The MAC sublayer is depicted in Attachment 18 Figure 18-2.

These primitives describe the required local information needed to identify the purpose of the incoming or outgoing data transfers. Their services are described in the following subsections. These services are local to each device and do not imply any particular implementation at the service interface.

This section provides all the information needed in order to initiate transfer or process reception of user data or control information.

c-17 3.2.2 MA_DATA.request

The MA_DATA.request primitive is used to describe how to transmit data that is carried by a MAC frame. The service is looked at from the transmitting MAC Service Client point of view.

The requesting MAC Service Client needs to provide sufficient information to the MAC sublayer to enable the MAC sublayer to format a FDU for transmission to a peer MAC Service Client. This information should include the destination, the data and the transmission service (Information or Command).

3.2.2.1 Function

The MA_DATA.request primitive defines the transfer of data from a local MAC Service Client entity to a single peer MAC Service Client entity (or entities in the case of group addresses).

The ARINC 429 MAC protocol provides two classes of service, the Command frame (local non-bridgeable frame) and the Information frame (bridgeable frame).

3.2.2.2 Semantics

The following parameters are provided with this primitive:

MA_DATA.request
(destination_MA_address,
destination_SAL_address
m_sdu,
service_class)

3.2.2.2.1 destination_MA_address

The destination_MA_address parameter should provide either an individual or a group MAC address when an Information frame is to be sent. If an invalid address is being handed over, the primitive should not initiate a FDU transmission. Instead, the MA_DATA.request should be dropped and the layer management should be informed.

The destination_MA_address is not used with a Command frame, but is required for an Information frame.

The format and contents of the destination_MA_address for an Information frame is defined in ARINC Specification 664.

COMMENTARY

At the time this text was written, ARINC Specification 664 was in draft state i.e., Project Paper 664.

3.2.2.2.2 destination_ SAL _address

The destination_SAL_address is required for both Command and Information frames. The destination_SAL_ address field contains the SAL to be used for transmitting the FDU.

The destination_SAL_address field for an Information frame can contain either a unique SAL, a multicast SAL or a bridge SAL. c-17

The destination_SAL_address field for a Command frame should contain a unique SAL.

The format and content of the SAL is defined in ARINC Specification 429, Part 1.

3.2.2.2.3 m_sdu

The m_sdu parameter indicates the MAC service data unit (data content) to be transmitted by the MAC sublayer entity.

If the m_sdu is empty, i.e. the length is zero, the MA_DATA.request primitive should not cause the initiation of a FDU transmission. Instead, the MA_DATA.request should be dropped and the layer management should be informed. The MAC Service Client should not create a MA_DATA.request with an empty m_sdu field.

3.2.2.2.4 service_ class

The service_class parameter indicates a quality of service requested by the MAC Service Client. The parameter indicates whether an Information or Command frame should be constructed. For a Command frame, the GFI field and Command Type field values are also indicated via this parameter (see Sections 3.3.2.1 and 3.3.2.2)

COMMENTARY

Currently, two services have been defined, which are "Transmission of Information Frame" and "Transmission of Command Frame". The determination of how to initiate one or the other is done locally and is a matter of implementation. However, care should be taken in selecting the mechanism to forward that information. More services might be defined at a later stage, which in turn might create a conflict with the implementation.

There is an implication on the destination address field that originates from the determination of which type of frame to use. The Command frame uses only the unique System Address Label (SAL) that has been defined for Version 1 of this protocol. The Information frame, however, needs two addresses in parallel: the IEEE compliant 48-bit destination and source addresses as well as a SAL.

3.2.2.3 When Generated

The MAC Service Client generates the MA_DATA.request primitive. It is a request by a MAC Service Client to the local MAC sublayer to transfer a MAC Service Data Unit (m_sdu) to a peer MAC Service Client entity (or entities). The primitive is generated either as the result of a request from a higher layer entity, or internally from the MAC Service Client itself.

c-17 3.2.2.4 Effect of Receipt

Upon receipt of the MA_DATA.request primitive, the MAC sublayer first creates the MAC frame by appending any MAC-specific fields (See Section 3.3.1 and 3.3.2).

Secondly, the MAC service entity packs the resulting MAC frame into an ARINC 429 "container" consisting of a series of contiguous ARINC 429 32-bit words (See Section 3.4.1) to create a FDU. It transmits the properly formatted FDU to the peer MAC sublayer entity (or entities) by means of the Physical layer services for subsequent transfer to the associated MAC Service Clients.

3.2.3 MA_DATA.indication

The MA_DATA.indication primitive is used to describe, at the destination MAC Service Client, the mechanics to be used in order to receive data that is carried by a FDU. The service is looked at from the receiving MAC Service Client point of view. This primitive is generated by the MAC entity upon reception of a valid FDU, and recreation of the MAC frame.

3.2.3.1 Function

The MA_DATA.indication primitive defines the transfer of data from a remote MAC Service Client entity to a local peer MAC Service Client entity.

3.2.3.2 Semantics

The following parameters are provided with the MA_DATA.indication primitive:

MA_DATA.indication
(destination_MA_address,
destination_SAL_address,
source_MA_address
source_SAL_address
m_sdu,
reception_status,
service_class)

3.2.3.2.1 destination_MA_address

The destination_MA_address is obtained from the Information frame and will contain either an individual or a group MAC entity address. The destination_MA_address parameter is empty when the MA_DATA.Indication is generated in response to receiving a Command frame

3.2.3.2.2 destination_SAL_address

The destination_SAL_address data is obtained from the received FDU for either an Information or Command frame.

3.2.3.2.3 source_MA_ address

The source_MA_address parameter contains the data from the source_MA_address field of an incoming FDU that contains an Information frame. For MAC addressing information refer to ARINC Specification 664. c-17

COMMENTARY

At the time this text was written, ARINC Specification 664 was in draft state i.e., Project Paper 664.

The source_MA_address parameter is empty when the MA_DATA.indication is generated in response to receiving a Command frame.

3.2.3.2.4 source_SAL_ address

The source_SAL_address is generated by the MAC based on the physical port on which the Information or Command frame was received.

3.2.3.2.5 m_sdu

The m_sdu parameter indicates the MAC service data unit as received by the local MAC sublayer entity.

3.2.3.2.6 reception_status

The reception_status parameter is used to pass status information to the MAC Service Client. The content of this parameter is implementation specific.

3.2.3.2.7 service_class

The service_class parameter is used to indicate whether an incoming frame is an Information or Command frame. If it is a Command frame, the GFI and command type (CT) information is also passed to the MAC Service Client.

3.2.3.3 When Generated

The MA_DATA.indication primitive is passed from the MAC sublayer to the MAC Service Client to indicate the arrival of a FDU at the local MAC sublayer entity. The MAC Sublayer first validates the FDU (see Section 3.4.12) and then removes the ARINC 429 "container" (i.e., the ARINC 429 32 bit word protocol headers/footers) and combines the resulting data and parameters into a MA_DATA.indication primitive. The primitive is reported only if the frame is valid (see Sections 3.3.1.5 and 3.3.2.5).

3.2.3.4 Effect of Receipt

The effect of receipt by the MAC Service Client is not defined in this document. Refer to the appropriate ARINC specification for MAC Service Client definitions.

Since buffering capacity is limited, in the event the MAC Service Client can not consume a frame, any subsequent frames may be discarded.

3.2.4 MAC Control functions

c-17

The MAC control function is derived from IEEE 802.3 Clause 31. Support of MAC control provides for real-time control and manipulation of MAC sublayer operations, and is provided as an Option (O_{14}) for the Version 3 protocol. Sections 3.2.4.1 and 3.2.4.2 describe this method of providing flow control for the Version 3 bit-oriented protocol using either the Information frame format for Ethernet bridgeable interfaces, or the Command frame format for non-bridgeable interfaces. Frames destined for the MAC control sublayer (MAC Control frames) are distinguished from frames destined for MAC Service Clients by a unique identifier.

For Information frames, the MAC Control Sublayer receives m_sdu frames and examines the 'Type' field for this unique identifier to determine if it is equal to control value (Pause opcode) of 8808 hex. If it is equal to 8808 hex, the MAC Control sublayer processes the Information frame. Otherwise the Information frame is passed to the MAC Service Client without modification.

For Command frames, the MAC Control Sublayer examines the Command Type parameter. If it is set to 'MAC Control' (and the first two octets of the m_sdu contain the Pause opcode, 8808 hex), the MAC Control sublayer processes the Command frame. Otherwise the Command frame is passed to the MAC Service Client without modification.

In the Version 1 protocol, flow control was provided through the use of BUSY and NCTS word responses to an RTS word. These ARINC 429 specific protocol words are not used in Version 3.

For Version 3, the Ethernet compatible PAUSE function is defined at the MAC Control sublayer. This control sublayer is part of the MAC, physically located just above the MAC sublayer, but below the MAC Service Client. Figure 18-2 of Attachment 18 depicts the usage of interlayer interfaces by the MAC Control sublayer. LRUs that support the option (O_{14}) to implement the MAC control sublayer should support the optional MAC service primitives, MA_CONTROL.request and MA_CONTROL. indication, as illustrated. The PAUSE operation is used to inhibit transmission of data frames from a directly connected full-duplex peer system only, and is therefore not bridgeable.

3.2.4.1 MA_CONTROL.request

The MA_CONTROL.request primitive is generated by the local MAC Control client to send to its peer, via the MAC sublayer, to request inhibiting of MAC frame transmissions from another system for a specified period of time. The MA_CONTROL.request primitive specifies:

a1. the destination_MA_address (for an Information frame) or

a2. Command Type set to MAC Control (for a Command frame)

b. the PAUSE opcode (8808 hex)

c. a Request_operand indicating the length of time for which it wishes to inhibit data frame transmission and

d. the Destination SAL.

The size of a MAC Control frame is 32-bits: a 16-bit (PAUSE opcode), and a 16-bit request operand. c-17

An example of the format of the MAC Control frame using both the Information and Command frame Data Unit (FDU) formats is illustrated in Attachments 19 and 20.

The Pause quanta (units of pause time) inhibits transmission of data frames for a specific period of time. The pause time quanta for Williamsburg Version 3 over high speed ARINC 429 is 5 milliseconds.

The PAUSE flow control function is defined as optional (O_{14}) for Version 3, since it is not anticipated that all Version 3 interface implementations, such as that of the CMU/VDR, will require flow control at the MAC sublayer. (i.e., VDR Mode A uses the MSK modulation scheme so air/ground throughput is somewhat limited and the need for flow control is not anticipated. For VDL Mode 2, the ARINC 429 MAC Service Client is AVLC/8208, which provides flow control from the CMU to the DSP ground station.)

COMMENTARY

IEEE 802.3, 1998 Annex 31B.2 states: "The pause-time is measured in units of pause quanta, equal to 512 bit times for the particular implementation. The range of possible pause time is 0 to 65535 pause quanta." The bit time for 10 megabit Ethernet is 0.1 us, therefore a pause quanta is 51.2 us for this media.

The bit time for 100 kilobit high speed ARINC 429 is 10 us, therefore the pause quanta for high speed ARINC 429 is 5.12 ms, or approximately 5 ms. The pause time range for ARINC 429 is then from 5 ms to 327 seconds. It is recommended that the pause range be appropriate for the application being supported to prevent adverse effects.

3.0 BIT-ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL

COMMENTARY

In typical operation the PAUSE function can be used in an X-OFF, X-ON operation by setting the pause_time to a large value, then when convenient, sending another PAUSE command with the pause_time = 0 to restart transmission. Sending another MAC Control frame before the Pause value in the previous MAC Control frame times out should cause the sink to replace the current Pause value with the new Pause value, if non-zero, and restart the timer. A Pause value of 0 terminates the Pause and restores normal operation.

3.2.4.2 MA_CONTROL.indication

The MA_CONTROL.indication primitive indicates the status of the local PAUSE operation (i.e., paused, or not paused) to the MAC Service Client.

When the MAC Control sublayer receives a MAC Control frame indicating a Pause condition, it should:

a. Notify the local MAC sublayer to stop sending frames to the peer MAC sublayer. The MAC sublayer should complete a frame transmission in progress.

c-17 b. Notify the local MAC Service Client of the Pause condition.

c. Start timer of duration indicated in the MAC Control frame.

d. Refuse frames from the local MAC Service Client until the timer expires.

When the pause timer expires the MAC Service Client should be notified. Likewise, when another control frame with the pause_time set equal to 0 is received then the MAC Service Client should be notified and normal operation resumed.

COMMENTARY

The design of ARINC 429 ICs frequently contain transmit FIFOs in order to reduce the burden on the microprocessor and therefore do not facilitate the termination of data transmission in mid frame. An implementation that does terminate transmission mid frame will cause Timer T_{17} in the receiver to time out.

3.3 MAC Frame Structures

This section defines in detail the MAC frame_structures for ARINC 429 using MAC procedures. It defines the relative positions of the various components of the MAC frame. It describes the general method for representing station addresses as well as ARINC 429 specific System Address Labels (see Section 3.4.3). Refer to ARINC Specification 664 for the MAC address specification for Information frames.

COMMENTARY

At the time this text was written, ARINC Specification 664 was in draft state i.e., Project Paper 664.

3.3.1 Information Frame Format

The MAC Information frame comprises 5 fields: the destination address field, the source address field, the length/type field, the data field and the frame check sequence field. Of these 5 fields all except the data field are of fixed length. Due to the transmission scheme used no preamble or delimiter fields are required as with other technologies. The frame check sequence field has been put into the EOF (see Section 3.4.8) words. Attachment 20 shows the format of the Information frame (and the FDU structure).

3.3.1.1 Address fields

Each MAC Information frame contains two address fields: the destination and the source address field, they are constructed the same way. The destination field specifies the MAC entity (or entities) for which the FDU is intended. The source address field identifies the MAC entity from which the FDU is initiated. Each address field contains 48 bits (i.e., six octets). For the construction of the MAC address fields see ARINC Specification 664.

COMMENTARY c-17

At the time this text was written, ARINC Specification 664 was in draft state i.e., Project Paper 664.

3.3.1.2 Length/Type Field

The Length/Type Field is defined in IEEE 802.3 Clause 3.2.6. This 2-octet field takes one of two meanings, depending on its numeric value.

a. If the value of this field is less than or equal to 1500 decimal, then the Length/Type field indicates the number of MAC client data octets contained in the subsequent data field of the frame (Length interpretation). In this case the MAC Client is defined to be the LLC service entity, and the LLC header should immediately follow the IEEE 802.3 header.

b. If the value of this field is equal to or greater than 1536 decimal, than the Length/Type field indicates the nature of the MAC Client protocol (Client interpretation).

c. Any other value of this field is considered undefined (i.e., values between 1500 and 1536)

3.3.1.2.1 Length

If the Length/Type field is a Length value, the use of IEEE 802.3 LLC is assumed. The values and uses of the LLC field are beyond the scope of this specification.

Length indicates the total number of octets in the data field of the frame. It does not include the address fields, the length/type field, or the FCS field.

Valid values for length are between 1 and 1500.

3.0 BIT-ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL

3.3.1.2.1 Length (cont'd)

COMMENTARY (cont'd)

The maximum size permitted by IEEE 802.3 is 1500 bytes for the payload of an Ethernet frame (i.e., the m_sdu size of an Ethernet frame).

3.3.1.2.2 Type

Protocols other than LLC may be used and this is supported by using the Length/Type field as an indicator of protocol type. When the Length/Type field is used in this manner then it should contain the protocol type consistent with the protocol encapsulated by the Ethernet frame. Valid values for the type field are defined in ARINC Specification 664.

COMMENTARY

At the time this text was written, ARINC Specification 664 was in draft state i.e., Project Paper 664.

3.3.1.3 Data Field

c-17 The data field can contain up to 2536 (2550-14) octets. However, the number of data field octets should be limited to 1500 in order to allow bridging to Ethernet-based networks.

Full data transparency is provided in the sense that any arbitrary sequence of octet values may appear in the data field.

3.3.1.4 Frame Check Sequence (FCS) Field

A cyclic redundancy check (CRC) sequence is used by the transmit and receive algorithms to generate and check a 32-bit (4-octet) CRC value. This value is computed as a function of the contents of the source address field, destination address field, Length/Type field and data field. The encoding is performed by the generating polynomial as defined in Section 3.4.9. The FCS is calculated by the transmitting MAC Service Entity and appended to the MAC frame. The FCS is recalculated and verified by the receiving MAC Service Entity following reconstruction of the entire frame and prior to providing the MA_DATA.indication to the MAC Service Client.

If Option O_{15} is not selected, the 32 bit CRC will not be generated or evaluated across the interface. If not used for containing the CRC, the FCS field should be set to binary zero to indicate to the receiving MAC service entity that the FCS was not calculated by the transmitting MAC service entity.

COMMENTARY

Not implementing (generating and validating) a Frame Check Sequence in the Information frame may have negative consequences on system integrity. The integrator is cautioned to be aware of applications that utilize the interface in which Option 15 has not been selected.

3.3.1.5 Validation of Information Frame

The receiving MAC sublayer should determine the validity of the incoming Information frame before passing the MAC frame to its local Service Client. The following subsections list the conditions for which the Information frame is said to be invalid. The contents of invalid MAC frames should not be passed to the local MAC Service Client, however reception of an invalid frame, and the reason for declaring it invalid, should be indicated to the MAC Service Client.

3.3.1.5.1 Invalid Address

Address checking should be performed according to ARINC Specification 664.

3.3.1.5.2 Invalid Length/Type

If the value in the Length/Type field is not a valid value as defined in section 3.3.1.2.1 and is not a known Type value then the MAC frame is considered invalid.

If the Length/Type field contains a length value and the number of octets does not match the length then the MAC frame is considered invalid.

3.3.1.5.3 Invalid FCS

The receiving MAC sublayer should verify the 32-bit FCS of the received MAC frame. If the 32-bit FCS is invalid, the MAC frame is also invalid c-17

The value of binary zero in the FCS field is a unique and valid FCS if CRC generation (O_{15}) is not selected. This indicates that the source MAC Service Entity does not calculate the CRC for this field. The receiving system should check that CRC non-generation (O_{15}) has been selected for this port.

3.3.2 Command Frame Format

For a MAC Command frame the following four fields are needed:

a. the GFI field ,

b. the Command Type field,

c. the Data field and

d. the Frame Check Sequence field.

Of these four fields all except the data field are fixed length. In order to retain as much compatibility with ARINC 429 Williamsburg processing as possible, the frame check field has been put into the EOF word (See Section 3.4.8). Attachment 19 shows the format of the Command frame (and FDU) structure.

The Command frame does not contain separate address fields like the Information frame. It relies on the ARINC 429 SAL for addressing. For the definition of the SAL see Section 3.4.3.

3.3.2.1 GFI Field

In order to retain as much compatibility with ARINC 429 Williamsburg Version 1 as possible, the GFI field has been retained in the SOF word. See Section 2.5 for the definition of GFI. If the GFI field is not used, this field is set to binary zeroes.

3.0 BIT-ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL

3.3.2.2 Command Type Field

The Command Type field was created to provide a mechanism for the receiver to differentiate between a Command frame that contains a command message, a control message, or a data message (such as a VDL Mode 2 frame transmitted across a CMU/VDR interface).

3.3.2.3 Data Field

The Data field contains up to 2552 octets. Full data transparency is provided in the sense that any arbitrary sequence of octet values may appear in the data field.

3.3.2.4 Frame Check Sequence (FCS) Field

A Cyclic Redundancy Check (CRC) sequence is used by the transmit and receive algorithms to generate and check a 16-bit (2-octet) CRC value. This value is computed as a function of the contents of the data field. The encoding is performed by the generating polynomial as defined in Section 3.4.9.

3.3.2.5 Validation of Command Frame

c-17 The receiving MAC sublayer should determine the validity of the incoming Command frame before passing the MAC frame to its local Service Client. The following subsection lists the condition for which the Command frame is said to be invalid. The contents of invalid MAC frames should not be passed to the MAC Service Client.

3.3.2.5.1 Invalid FCS

The receiving MAC sublayer should verify the 16-bit FCS of the received MAC frame. If the 16-bit FCS is invalid, the MAC frame is also invalid.

3.4 MAC Transmit/Receive Functions

This section defines the functions to transmit and receive the contents of FDUs on the physical medium. It is also known as the interface to the Physical layer.

3.4.1 Frame Data Unit (FDU)

A Frame Data Unit (FDU) envelops a MAC frame and consists of a set of contiguous ARINC 429 32-bit words, each containing the System Address Label (see Section 3.4.3) of the sink. A FDU can be either an Information frame or a Command frame. The initial data word of each FDU is a Start of Frame (SOF) data word (see Section 3.4.6). The MAC frame (See Section 3.3) is contained within the data words that follow. The FDU is concluded with one or two consecutive End of Frame (EOF) words (see Section 3.4.8). A FDU always consists of no more than one frame.

A Frame Data Unit (FDU) contains binary encoded octets. The octets may be set to any possible binary value. The FDU contains any string of bits desired. The only restriction is that the bits be organized into full 8 bit octets. The interpretation of those bits is not a part of this specification.

3.4.2 Frame Data Unit (FDU) Size and Word Count

The FDU word count is the sum of the SOF word, the data words containing the MAC frame, and the EOF word(s). The Frame Data Unit (FDU) may vary in size from a minimum of three, to a maximum of 1023 ARINC 429 words (including the SOF and EOF words). When a FDU is organized for transmission, the total number of ARINC 429 words to be sent (word count) is calculated.

3.4.3 System Address Labels (SALs) c-17

Each on-board system attached to an ARINC 429 bus that requires aperiodic message capability is assigned a System Address Label (SAL). The listing of SAL assignments can be found in Attachment 11 to ARINC c-18 Specification 429, Part 1. FDUs are sent point-to-point, even though other systems may be connected and listening to the output of a transmitting system. In order to identify the intended recipient of a transmission, the Label field (bits 1-8) is used to carry a System Address Label (SAL).

When a system sends a FDU to another system, the sending system (source) addresses each ARINC 429 word to the receiving system (sink) by setting the Label field to the SAL assigned to the sink.

Apart from existing system specific unique SALs, two universally applicable SALs have been defined: the Multicast SAL (MSAL) and the Bridge SAL (BSAL). These SAL´s are designated for Information frames only and should not be used for Command frames.

For situations where a system will need to communicate with another system, which has no SAL assigned, i.e., is c-17 not attached to an ARINC 429 bus, a Bridge SAL will be used.

COMMENTARY

> The Bridge SAL assumes that no compatibility conflict will arise with other systems. The bridge will accept all FDUs and afterwards selects how to handle them based upon a predefined or, in some cases, learned bridging table.

For situations, where multiple systems will need to be addressed at the same time, a Multicast SAL will be used This SAL can only be used for the transmission of Information frames. In this case, a system needs to examine the MAC destination address contained in the FDU.

COMMENTARY

> Similar to the Bridge SAL, the Multicast SAL assures that no compatibility conflict will arise with other systems. Generally, it addresses all attached systems, local or remote (beyond the bridge). It is the responsibility of each individual system to determine whether or not to accept the FDU being received.

When a system receives any data containing its SAL that is not sent through the established conventions of this protocol, the data received should be ignored.

3.0 BIT-ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL

3.4.4 Bit Rate and Word Timing

Data transfer should operate at the low or high-speed rate as c-18 defined in Part 1 of ARINC Specification *429* depending on the value of Option 2 (see Table 10-3b). The source should introduce a gap between the end of each ARINC 429 word transmitted and the beginning of the next. The gap should be c-17 at least 4 bit times (minimum). The sink should be capable of receiving the FDU with the minimum word gap of 4 bit times between words. The source should not exceed a maximum average of 40 bit times between data words of a FDU.

3.4.5 Word Type

The Word Type field occupies bit 31-29 in all bit-oriented FDU words. See Table 11-1A of Attachment 11 for a description of the Word Type field. The Word Type field is used to identify the function of each ARINC 429 data word used by the bit-oriented communication protocol. There are two new word types for the Version 3 protocol, word types 010 and 011.

3.4.6 Start of Frame (SOF)

When a system wants to transmit either an Information Frame Data Unit or a Command Frame Data Unit, the system should start by sending the Start of Frame (SOF) data word, immediately followed by the data words, which contain the contents of the MAC frame. The format of the c-17 SOF word differs between Information and Command FDUs. See Attachment 11, Tables 11-8 and 11-10 for a description of these two words.

In order to distinguish between Information and Command frames, the SOF word contains an Information/Command Frame field.

3.4.6.1 Information/Command (I/C) Frame Field

The Version 3 protocol provides for two different MAC frame types in order to adapt to the requirements of different applications: an (Ethernet) bridgeable Information frame and a non-bridgeable Command frame. For a description of the Frame Data Units (FDUs) which contain these two types of MAC frames, see Attachments 19 and 20.

To identify the different frame types, bits 20-19 of the SOF word (for both Information and Command FDU) carry the following discriminator:

Bit 20	Bit 19	Definition
0	0	Information frame
0	1	Command frame
1	0	Reserved
1	1	Reserved

3.4.6.2 Information SOF word

The SOF word for the Information Frame Data Unit contains a 10-bit word count, a 2-bit Information/Command (I/C) frame field (as described in Section 3.4.6.1) and an 8-bit Reserved field.

3.4.6.2.1 Word Count

The Word Count field (bits 18-9) of the Information SOF word reflects the number of ARINC 429 words to be transmitted in the Frame Data Unit. For the Information frame, the maximum word count value is 1023 ARINC 429 words. The minimum word count value is 10 ARINC 429 words. An Information FDU with the minimum word count would contain one SOF word, seven data words, and two EOF words. An Information FDU with the maximum word count value would contain one SOF word, 1020 data words and two EOF words.

3.4.6.2.2 Reserved Bits

Bits 28-21 of the SOF word for the Information FDU are reserved (not used). These bits should be set to binary zeroes.

3.4.6.3 Command SOF word

The SOF word for the Command frame Data Unit contains a 10-bit word count, a 2-bit Information/Command (I/C) frame field (described in Section 3.4.6.1), a 2-bit Reserved field, a 2-bit Command Type Field and a 4-bit General Format Identifier (GFI) field.

3.4.6.3.1 Word Count c-17

The Word Count field (bits 18-9) of the Command SOF word reflects the number of ARINC 429 words to be transmitted in the Frame Data Unit. For the Command frame, the maximum word count value is 1023 ARINC 429 words. The minimum word count value is 3 ARINC 429 words. A Command FDU with the minimum word count would contain one SOF word, one data word and one EOF word. A Command FDU with the maximum word count value would contain one SOF word, 1021 data words and one EOF word.

3.4.6.3.2 Reserved Bits

Bits 22-21 of the SOF word for the Command FDU are reserved (not used). These bits should be set to binary zeroes.

3.4.6.3.3 General Format Identifier Field (GFI)

For the Command Frame Data Unit (FDU), a General Format Identifier (GFI) occupies bits 28-25 of the SOF word. It’s function is similar to the General Format Identifier described in Section 2.5.10.1 of the Version 1 bit-oriented protocol. This field is maintained in the Command FDU for backward compatibility to the point-to-point (non-bridgeable) link layer protocol used in Version 1. The GFI field is used in Version 3 as an indicator to the MAC Service Client of the format of data words to follow.

3.4.6.3.4 Command Type Field (CT)

For the Command Frame Data Unit (FDU), a Command Type (CT) field occupies bits 24-23 of the SOF word. The purpose of the CT field is to facilitate the differentiation of FDUs by functional context.

The following values have been assigned:

3.0 BIT-ORIENTED MEDIA ACCESS CONTROL (MAC) PROTOCOL

Bit 24	Bit 23	Meaning
0	0	Command
0	1	Data
1	0	MAC Control
1	1	reserved

COMMENTARY

The contextual differentiation contained in the CT field may be used to implement a flow control mechanism by which FDUs of one context type are blocked while FDUs of other types are not. Such a mechanism might be used in a buffer-limited implementation to control the flow of FDUs containing application data while allowing command or control FDUs to pass freely. The protocols and procedures to implement this mechanism are not part of this specification.

3.4.7 Data

The definition of the data words which contain the MAC frame's contents are identical for both Version 3 (Information or Command) frame types, and are identical with the data words used in Version 1 of the bit-oriented protocol. However, the first six data words of an Information MAC frame data unit contains additional MAC c-17 addressing and length/type information.

Data words immediately follow the SOF word. The octets of the FDU are encoded and transmitted in 32-bit data words. There are two basic types of data words: full binary and partial binary. Binary data words may contain one, two, three, four and five semi-octets. A semi-octet (or nibble) is half of an octet, or four bits in length.

Binary data words of five semi-octets are called Full Data words. Binary data words of fewer than five semi-octets are called Partial Binary Data words.

Each FDU, prior to transmission, should conclude (end) with a complete octet.

3.4.7.1 Full Data Word(s)

A Full Data Word has 20 bits available for data. This space is allocated in five semi-octets. The octets of the data file are divided into two semi-octets and placed sequentially into the data words. The least significant bit of the least significant semi-octet is sent first.

If, in the process of placing the octets into the data words, an octet is split between two different words, the least significant semi-octet goes in the last (n_5) semi-octet of the current data field. The most significant semi-octet follows in the first (n_1) semi-octet of the data field of the next word. See Tables 11-2 and 11-3 of Attachment 11 for data word formats.

If the end of the FDU does not completely fill the last Full Data Word, a Partial Data Word (see Section 3.4.7.2) should be used to complete the FDU transmission.

In order to obtain maximum system efficiency, Partial Data Words should be used only when they are required to complete the data for a FDU transmission. A FDU should not be encoded as a string of Partial Data Words where 8 or 16 bits of data are encoded for each ARINC 429 word used.

3.4.7.2 Partial Data Word(s)

A Partial Data Word contains from one to four semi-octets. The number of semi-octets in a partial data word is indicated in bits 28-25 of the word. Only full four-bit semi-octets can be sent; one, two or three bits are invalid. Partial Data Word semi-octets are sent in the same order as the semi-octets in a full data word. The unused semi-octets in the partial data word should be set to binary zeros.

3.4.7.3 SOLO Words

SOLO words, as defined in the Williamsburg Version 1 protocol in Section 2.5.17, may be used in the Version 3 protocol. However, they should be used only across non-bridgeable interfaces (i.e. in conjunction with Command frames).

3.4.8 End of Frame (EOF)

Each FDU transfer is terminated by one or two End of Frame (EOF) word(s), depending upon the nature of the FDU. The format and number of EOF word(s) differ between Information and Command frames.

For an Information frame the EOF words contain a frame check sequence which consists of a 32-bit Cyclic c-17 Redundancy Check (CRC).

For a Command frame the EOF word is identical to a Version 1 EOT word and contains a frame check sequence which consists of a 16-bit Cyclic Redundancy Check (CRC).

The final FDU bit is always set for consistency with Version 1.

See Attachment 11, Tables 11-9 and 11-11 for the definition of these words.

COMMENTARY

The ARINC 429 Physical layer is a twisted shielded pair of wires which has been demonstrated to exhibit high integrity and unlikely to introduce errors into the data passing through it. Simple parametric data is usually transmitted at a refresh rate high enough to permit recognition and suppression of erroneous data. Since the transfer of data using a file transfer protocol contains no provision for automatic refresh, some applications may require high data integrity to be confirmed by an error checking mechanism. For this reason, each FDU contains a CRC check. The use of the CRC in this case does not imply any inherent lack of integrity of the ARINC 429 link.

3.4.9 Frame Check Sequence

For a Command frame, the Frame Check Sequence field of the EOF word contains a 16-bit CRC as defined in Sections 2.5.12.1 and 2.5.12.2 of the Version 1 bit-oriented protocol. For an Information frame, the Frame Check Sequence field of the EOF words contains a 32-bit CRC, as defined in this section. Both CRC polynomials are referenced in ISO 3309, and the procedure for calculation of the 32-bit CRC is identical

3.4.9 Frame Check Sequence (cont'd)

to that of the 16-bit CRC. The only implementation differences are found in the length and format of the two generator polynomials and length of the CRCs.

The most significant bit (MSB) of the 32-bit CRC sequence for Information frames is transmitted first.

3.4.9.1 32-Bit CRC Encoding

Determination and encoding of the CRC is as follows:

The k bits of information data in the frame are represented as the coefficients of a polynomial, G(x); where k is the number of data bits in the frame existing between, but not including, the SOF and EOF words. For example, if the data stream is 1010001,

$k = 7$ and $G(X) = x^6 + x^4 + 1$

The CRC calculation is performed over the data octets only of the frame with any semi-octets zero filled.

The generator polynomial for the 32-bit CRC is of the form,

c-17

$$P(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

The CRC is then determined as the one's complement of the remainder, R(x), obtained from the modulo 2 division of:

$$\frac{x^{32}G(x) + x^k (x^{31} + x^{30} + x^{29} + x^{28} + \ldots + x^3 + x^2 + x + 1)}{P(x)}$$

$$= Q(x) + \frac{R(x)}{P(x)}$$

Note: The addition of $x^k (x^{31} + x^{30} + x^{29} + x^{28} + \ldots + x^3 + x^2 + x + 1)$ to $x^{32}G(x)$ (which is equivalent to inverting the first 32 bits of G(x) and appending a bit string of 32 zeroes to the lower order end of G(x)), corresponds to initializing the initial remainder to a value of all "ones". The complementing of R(x), by the transmitter, at the completion of the division ensures that the received, error-free message will result in a unique, non-zero remainder at the receiver.

At the transmitter, the CRC is added to the $x^{32}G(x)$ product, resulting in the message, M(x), of length n where:

$$n = k + 32, \text{ and}$$

$$M(x) = x^{32}\, G(x) + \overline{R(x)}$$

$$= x^{32}\, G(x) + CRC$$

3.4.9.2 32-Bit CRC Decoding

Decoding of the CRC at the receiver is as follows:

At the receiver, the incoming M(x) is multiplied by x^{32}, added to the product, $x^n (x^{31} + x^{30} + x^{29} + x^{28} + \ldots + x^3 + x^2 + x + 1)$ and divided by P(x) as follows:

$$\frac{x^{32}M(x) + x^n (x^{31} + x^{30} + x^{29} + x^{28} + \ldots + x^3 + x^2 + x + 1)}{P(x)}$$

$$= Qr(x) + \frac{Rr(x)}{P(x)}$$

If the transmission of the serial incoming bits plus CRC (i.e. M(x)) is error free, then the remainder, Rr(x) will be:

1100 0111 0000 0100 1101 1101 0111 1011 (coefficients of x^{31} through x^0 , respectively).

COMMENTARY

The notation used to describe the CRC is based on the property of cyclic codes that any code vector, such as 1000000100001, can be represented by a polynomial $G(x) = x^{12} + x^5 + 1$. The elements of an n element code word are thus the coefficients of a polynomial of order n - 1. In this application, these coefficients can have the value 0 or 1 and all polynomial operations are performed modulo 2. The polynomial representing the information content of a frame is generated starting with the Frame bit which is encoded in bit 9 of the first ARINC 429 data word (following the SOF word) as the coefficient of the first (highest order) term. c-17

3.4.10 Incomplete FDU Timer

The Incomplete FDU (T_{17}) timer is used to insure that a source device does not "lock-up" a sink with an incomplete frame.

The sink will start the T_{17} timer when it receives a respective SOF word.

In case of an Information frame the T_{17} timer is stopped when both valid EOF words are received.

In case of a Command frame the T_{17} timer is stopped when a single valid EOF word is received.

The T_{17} timer should also be stopped if the sink discards the partial frame for other reasons defined in this specification. When T_{17} is exceeded, the sink should discard any partial frame already received and ignore any data until a proper SOF word is received.

3.4.11 ALOHA

The first ALOHA word transmitted in a sequence should contain the highest Version Number supported by the source device. If the ALOHA Response contains a version that does not match the ALO version, the source device should take one of the following actions:

a. If the source device is able to adapt to the differences in protocol version, file transfers may proceed using the protocol version identified in the ALR word.

b. If the source device is not able to adapt to the differences in protocol version, the source should again initiate the ALOHA word with the version field set to the highest version supported by the source that is lower than the version indicated in the previous ALR (see examples identified in Attachment 13A).

The ALO/ALR protocol determination process should continue until a common protocol version is found. If either a protocol version is found or none of the protocol versions match, the source should notify the local MAC Service Client with a MA_DATA.indication with its reception_status set to notify a reset condition. The parameter in the primitive only has local significance and provides the MAC Service Client with status information.

When the system only supports bit-oriented protocols, it should repeat the process described in this section.

When the system also supports character oriented protocols, then the process defined in Section 2.5.19.2 should be followed. The protocol determination process will continue until a common protocol is found. See Attachment 17.

The ALOHA word should contain a Subsystem SAL field as shown in Attachment 11, Table 11-4. This field should contain the SAL of the device sending the ALOHA word,
c-17 with bit 17 as the most significant bit and bit 24 as the least significant bit of the Subsystem SAL.

A device that supports the bit-oriented Link layer protocol should always be able to respond to the receipt of the ALO word.

Whenever a device receives the ALO word, it should leave its present task and respond within T_{15} with an ALR word.

The ALR response should reflect the device's protocol version level by the way of the Version Number contained within the ALR word. If the Version Number of the ALO does not match the sink device's protocol Version Number, the sink device should select the next lowest version supported (equal to or lower than the version indicated by the previous ALO word) and indicate this new version in the ALR word.

If the device was in the sink mode (had already begun receiving a FDU), it should discard any partial FDU it had received.

If the device was in the source mode (had already begun transmitting a FDU), it should resend the FDU beginning with the first word of the FDU.

Because Option O_{12} is selected, the device receiving an ALOHA word should use the Subsystem SAL from the ALOHA word as the SAL of the ALOHA Response (ALR) word and for all subsequent bit-oriented protocol transmissions for that interface.

3.4.12 Validation of FDUs

The receiving MAC sublayer should determine the validity of the incoming Start of Frame (SOF) word, the MAC frame, and the End of Frame (EOF) words before passing the MAC frame to its local Service Client. The SOF, MAC frame, and EOF are collectively referred to as a Frame Data Unit (FDU). The following subsections list conditions for which the FDU is said to be invalid in addition to the conditions already identified in sections 3.3.1.5 and 3.3.2.4 for invalid frames. The contents of invalid MAC frames should not be passed to the local MAC Service Client.

3.4.12.1 Missing SOF Word

The Information SOF word should be formatted as depicted in Attachment 11, Table 11-10 and should be the first word of the transmitted information FDU. The Command SOF word should be formatted as depicted in Attachment 11, Table 11-8 and should be the first word of the transmitted command FDU. If the SOF word is not received as the first word, the MAC frame is invalid.

3.4.12.2 Missing EOF Word(s)

Two EOF words should follow the transmission of the final data words of an Information MAC frame. They should be formatted as depicted in Attachment 11, Table 11-11. If either or both are missing, or are not formatted as depicted in Attachment 11, Table 11-11, then the MAC frame is invalid.

One EOF word should follow the transmission of the final data words of a Command MAC frame. It should be formatted as depicted in Attachment 11, Table 11-9. c-17
If it is missing, or is not formatted as depicted in Attachment 11, Table 11-9, then the MAC frame is invalid.

3.4.12.3 Parity Errors

Bit 32 of each ARINC 429 word should be set to odd parity for the entire word. Upon receipt of a word, the receiving unit should verify that each word was received with odd parity. If any word is received with even parity, the receiving unit should take no action and ignore the word.

COMMENTARY

When the receiving MAC sublayer receives a word with bad parity, it cannot be sure of the intended label. The word may not even be a part of the FDU, so by ignoring the offending word, there may still be a chance of a successful FDU transfer. If the offending word was intended to be a part of the FDU, then when the EOF word(s) are received, the actual word count will not match the expected word count (and the CRC will probably be invalid). If the word count is incorrect (or if the CRC is invalid), then the MAC frame is also invalid, as per Sections 3.4.12.5 and 3.4.12.4 (or Sections 3.3.1.5.3 and 3.3.2.5.1 for invalid CRC).

3.4.12.4 Word Count Errors

Upon receipt of the final EOF word of the FDU, the receiving MAC sublayer should verify that the actual number of words received is the number of words expected, as per bits 18-9 of the SOF word. If the word counts do not match, the MAC frame is invalid.

3.4.12.5 CRC Errors

Upon receipt of the final EOF word of the FDU, the receiving MAC sublayer should verify the CRC of the received MAC frame. If the CRC is invalid, the MAC frame is also invalid and should be discarded.

c-17 The FDU should contain an integral number of octets in order to pass the FCS.

3.4.13 Inter-FDU Gap Time

An inter-Frame Data Unit gap time, T_{18}, of 10 ms minimum should be implemented between any two successive MAC frame transmissions in order to allow the receiving MAC sublayer sufficient CRC verification time between frames.

Timer T_{18} is closely related to the generation of the CRC field for the Information frame. If a CRC is not to be generated (O_{15} = No), this timer is not necessary and implementation is optional.

ATTACHMENTS 1 - 9

Attachments 1 through 9 are included in ARINC Specification 429 Part 1 and therefore are not used in ARINC Specification 429, Part 3. These attachment numbers are not used to maintain consistency with previous versions of ARINC Specification 429. c-16

ATTACHMENT 10
VARIABLES OF BIT-ORIENTED PROTOCOL

c-16

Table 10-1 BIT-ORIENTED PROTOCOL EVENTS

EVENT	DESCRIPTION OF EVENT	STANDARD VALUE [1]
N_1	MAX NUMBER OF RTS REPEATS FOLLOWING NCTS	5
N_2	MAX NUMBER OF RTS REPEATS FOLLOWING BUSY	20
N_3	MAX NUMBER OF RTS REPEATS FOLLOWING NO RESPONSE	5
N_4	NUMBER OF NAK WORDS RECEIVED BEFORE DECLARING FAILURE OF COMMUNICATION	3
N_5	NUMBER OF SYN WORDS RECEIVED BEFORE DECLARING FAILURE OF COMMUNICATION	3
N_6	MAX NUMBER OF ALO REPEATS FOLLOWING NO RESPONSE	3

Table 10-2 BIT-ORIENTED PROTOCOL APPLICATION SELECTION

APPLICATION	CONDITION	STANDARD ACTIONS
A_1	WHEN T_2 N_1 EXCEEDED	REPORT TO HIGHER LEVEL ENTITY
A_2	WHEN T_4 N_2 EXCEEDED	REPORT TO HIGHER LEVEL ENTITY
A_3	WHEN T_5 N_3 EXCEEDED	REPORT TO HIGHER LEVEL ENTITY

c-16

Table 10-3a BIT-ORIENTED PROTOCOL OPTIONS FOR VERSION 1

OPTION	DESCRIPTION	STANDARD INTERFACE [1]
O_1	Half or Full Duplex Operation	Half Duplex
O_2	High or Low Speed Bus	Low
O_3	Automatic CTS when ready	No [5]
O_4	Accept Auto CTS	No [5]
O_5	Sys Priority to resolve RTS Conflict	Yes
O_6	Reserved	--
O_7	Reserved	--
O_8	Use of SOLO Word	Yes
O_9	Reserved	--
O_{10}	Dest Code in RTS/CTS/NCTS/BUSY used	Yes
O_{11}	Bit-Protocol verification	Yes
O_{12}	Use Subsystem SAL from ALO word	No

ATTACHMENT 10
VARIABLES OF BIT-ORIENTED PROTOCOL

Table 10-3b - BIT-ORIENTED PROTOCOL OPTIONS FOR VERSION 3

Option	Description	Standard Interface [1]	Notes
O_1	Half or Full Duplex Operation	Full Duplex	
O_2	High or Low Speed Bus	High	
O_3	Automatic CTS when ready	N/A	
O_4	Accept Auto CTS	N/A	
O_5	Sys Priority to resolve RTS Conflict	N/A	
O_6	Spare	--	
O_7	Spare	--	
O_8	Use of SOLO Word	Yes	6
O_9	Spare	--	
O_{10}	Destination Code in RTS/CTS/NCTS/BUSY used	N/A	
O_{11}	Bit-Protocol verification	Yes	
O_{12}	Use Subsystem SAL from ALO word	Yes	
O_{13}	Use of Information or Command frames		7
O_{14}	Use of Pause Function		7
O_{15}	Generation (Yes) or Non-generation (No) of 32 bit CRC for Information frame	Yes	8

c-17

NOTES:

[1] The STANDARD VALUE (or STANDARD INTERFACE) should be used as the default value if a specific value is not designated in the applicable equipment specification. For example, the standard interface for option 2 (429 bus speed) defaults to low speed for version 1 systems, unless high speed is specified in equipment specifications. Values shown as N/A indicate that option cannot be used.

[2] For those timers that are not associated with a repeat sequence, it is intended that a working system has minimized actual response times. For example: A system should reply with CTS as soon as possible after reception of RTS. It is not intended that a system take the maximum time T_1 to reply to the RTS on a routine basis.

[3] T_6 and T_7 are used when Option 5 (O_5) is not selected.

[4] Implementation of timer T_{10} is optional. If T_{10} is not used, T_{16} should be used.

[5] 0_3 and 0_4 should be Yes for expedited file transfer. See Section 2.5.7.3.

[6] The SOLO Word is not bridgeable to Ethernet and should only be used in conjunction with Command frame (non-bridgeable) interfaces.

[7] Options 13 and 14 are dependent on the applicable equipment interface specification. For a point to point only interface, such as the CMU/VDR VDL Mode 2 Interface, Option 13 is set to 'Command Frame'. For equipment interfaces that may need to be bridgeable to Ethernet, Option 13 is set to 'Information frame.' If the Pause Function option is selected as 'Yes', then the Pause function will be formatted into a Command FDU if Option 13 is set to 'Command Frame', or into an Information FDU if Option 13 is set to 'Information Frame'.

[8] Selection of Option 15 is dependent on the applicable equipment interface specification.

ATTACHMENT 10
VARIABLES OF BIT-ORIENTED PROTOCOL

Table 10-4 VARIABLES OF LOW SPEED BIT-ORIENTED PROTOCOL - VERSION 1

TIME	DESCRIPTION	MIN VALUE	MAX VALUE	TIMER OR DESIGN GOAL FOR SOURCE OR SINK	NOTES	REFERENCE
T_1	CTS/NCTS Send Time	0 ms	100 ms	Goal for Sink	2	2.5.7
T_2	RTS Repeat Time After Receipt of NCTS	500 ms	700 ms	Timer for Source		2.5.7.2
T_3	Busy Send Time	0 ms	100 ms	Goal for Sink	2	2.5.7.3
T_4	RTS Repeat Time After Receipt of Busy	15 sec	18 sec	Timer for Source		2.5.7.3
T_5	RTS Repeat Time If No Response	500 ms	700 ms	Timer for Source		2.5.7.4
T_6	Time of Random Timer to Resolve RTS Conflicts	50 ms	500 ms	Goal for Source	3	2.5.8.1
T_7	Increment of Time T_6	10 ms	100 ms	Goal for Source	3	2.5.8.1
T_8	ACK/NAK/SYN Send Time	0 ms	200 ms	Goal for Sink	2	2.5.13
c-16 T_9	LDU Timeout Following CTS	2.5 sec	2.7 sec	Timer for Sink		2.5.13.2 2.5.13.6
T_{10}	ACK/NAK Timeout	2.7 sec	3.0 sec	Timer for Source	4	2.5.16
T_{11}	Loop Back Send Time	0 ms	100 ms	Goal for Sink	2	2.5.17.1
T_{12}	ALO Repeat Time if No Response to ALO	200 ms	250 ms	Timer for Source		2.5.19.1
T_{13}	SOT Send Time After Receipt of CTS	0 ms	200 ms	Goal for Source	2	2.5.10
T_{14}	Incomplete File Timeout	2 min	2.2 min	Timer for Sink		2.5.14.3
T_{15}	ALR Send Time	0ms	180 ms	Goal for Sink	2	2.5.19.1.2
T_{16}	ACK/NAK Timeout After EOT	220 ms	330 ms	Timer for Source		2.5.16

ATTACHMENT 10
VARIABLES OF BIT-ORIENTED PROTOCOL

Table 10-5 VARIABLES OF HIGH SPEED BIT-ORIENTED PROTOCOL - VERSION 1

TIME	DESCRIPTION	MIN VALUE	MAX VALUE	TIMER OR DESIGN GOAL FOR SOURCE OR SINK	NOTES	REFERENCE
T_1	CTS/NCTS Send Time	0 ms	100 ms	Goal for Sink	2	2.5.7
T_2	RTS Repeat Time After Receipt of NCTS	100 ms	140 ms	Timer for Source		2.5.7.2
T_3	Busy Send Time	0 ms	100 ms	Goal for Sink	2	2.5.7.3
T_4	RTS Repeat Time After Receipt of Busy	1.0 sec	1.2 sec	Timer for Source		2.5.7.3
T_5	RTS Repeat Time If No Response	150 ms	200 ms	Timer for Source		2.5.7.4
T_6	Time of Random Timer to Resolve RTS Conflicts	50 ms	500 ms	Goal for Source	3	2.5.8.1
T_7	Increment of Time T_6	10 ms	100 ms	Goal for Source	3	2.5.8.1
T_8	ACK/NAK/SYN Send Time	0 ms	200 ms	Goal for Sink	2	2.5.13
T_9	LDU Timeout Following CTS	400 ms	440 ms	Timer for Sink		2.5.13.2 2.5.13.6
T_{10}	ACK/NAK Timeout After CTS	600 ms	660 ms	Timer for Source	4	2.5.16 c-16
T_{11}	Loop Back Send Time	0 ms	100 ms	Goal for Sink	2	2.5.17.1
T_{12}	ALO Repeat Time if No Response to ALO	200 ms	250 ms	Timer for Source		2.5.19.1
T_{13}	SOT Send Time After Receipt of CTS	0 ms	100 ms	Goal for Source	2	2.5.10
T_{14}	Incomplete File Timeout	10 sec	11 sec	Timer for Sink		2.5.14.3
T_{15}	ALR Send Time	0 ms	180 ms	Goal for Sink	2	2.5.19.1.2
T_{16}	ACK/NAK Timeout After EOT	220 ms	330 ms	Timer for Source		2.5.16

ATTACHMENT 10
VARIABLES OF BIT-ORIENTED PROTOCOL

Table 10-6 - VARIABLES OF HIGH SPEED CONNECTIONLESS BIT-ORIENTED PROTOCOL - VERSION 3

TIME	DESCRIPTION	MIN VALUE	MAX VALUE	TIMER OR DESIGN GOAL FOR SOURCE OR SINK	NOTES	REFERENCE
T_1						
T_2						
T_3						
T_4						
T_5						
T_6						
T_7						
T_8						
T_9						
T_{10}						
T_{11}						
T_{12}	ALO Repeat Time if No Response to ALO	200 ms	250 ms	Timer for Source		
T_{13}						
T_{14}						
T_{15}	ALR Send Time	0 ms	180 ms	Goal for Sink		
T_{16}						
$\mathbf{T_{17}}$	Incomplete FDU Timeout	750 ms	1 sec	Timer for Sink		
T_{18}	Inter-FDU Gap Time	10 ms	N/A	Timer for Source	1	

c-17

NOTE:

[1] The minimum value is specified to give the sink time to finish processing an FDU before the next FDU arrives. The designer is encouraged to use the min value of 10 ms in order to make optimal use of the ARINC 429 data bus bandwidth. A maximum value cannot be specified because of the aperiodic nature of the data sent by most applications that use the ARINC 429W file transfer protocol. If the LRU does not have another FDU to transmit then this gap will be very large (seconds, minutes, hours!).

ATTACHMENT 10
VARIABLES OF BIT-ORIENTED PROTOCOL

Table 10-7 VARIABLES OF LOW SPEED CONNECTIONLESS BIT-ORIENTED PROTOCOL - VERSION 3 c-16

TIME	DESCRIPTION	MIN VALUE	MAX VALUE	TIMER OR DESIGN GOAL FOR SOURCE OR SINK	NOTES	REFERENCE
T_1						
T_2						
T_3						
T_4						
T_5						
T_6						
T_7						
T_8						
T_9						
T_{10}						
T_{11}						
T_{12}	ALO Repeat Time if No Response to ALO	200 ms	250 ms	Timer for Source		
T_{13}						
T_{14}						
T_{15}	ALR Send Time	0 ms	180 ms	Goal for Sink		
T_{16}						
T_{17}	Incomplete FDU Timeout	7.5 seconds	~~1 sec~~	Timer for Sink		
T_{18}	Inter-FDU Gap Time	10 ms	N/A	Timer for Source	1	

c-18

NOTE:

[1] The minimum value is specified to give the sink time to finish processing an FDU before the next FDU arrives. The designer is encouraged to use the min value of 10 ms in order to make optimal use of the ARINC 429 data bus bandwidth. A maximum value cannot be specified because of the aperiodic nature of the data sent by most applications that use the ARINC 429W file transfer protocol. If the LRU does not have another FDU to transmit then this gap will be very large (seconds, minutes, hours!).

ATTACHMENT 10
VARIABLES OF BIT-ORIENTED PROTOCOL

NOTES:

1. The STANDARD VALUE (or STANDARD INTERFACE) should be used as the default value if a specific value is not designated in the applicable equipment specification. For example, the standard interface for option 2 (429 bus c-16 speed) defaults to low speed for version 1 systems, unless high speed is specified in equipment specifications. Values shown as N/A indicate that option cannot be used.

2. For those timers that are not associated with a repeat sequence, it is intended that a working system has minimized actual response times. For example: A system should reply with CTS as soon as possible after reception of RTS. It is not intended that a system take the maximum time T_1 to reply to the RTS on a routine basis.

3. T_6 and T_7 are used when Option 5 (O_5) is not selected.

4. Implementation of timer T_{10} is optional. If T_{10} is not used, T_{16} should be used.

c-16 5. O_3 and O_4 should be Yes for expedited file transfer. See Section 2.5.7.3.

ATTACHMENT 11
BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

NOTE: All reserved fields should be set to binary 0

Table 11-1 GENERAL WORD FORMAT

BIT	32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
[1]	P	Word Type	GFI, Control Data or Word Type Extensions	DATA	SAL

c-16

Table 11-1A WORD TYPE

31	30	29	WORD TYPE
0	0	0	Full Binary Data Word
0	0	1	Partial Binary Data Word
0	1	0	Start of Frame – Version 3
0	1	1	End of Frame – Version 3
1	0	0	Protocol Word
1	0	1	Solo Word
1	1	0	Start Of Transmission – Version 1
1	1	1	End Of Transmission – Version 1

c-16

Table 11-2 FULL DATA WORD

BIT	32	31 30 29	28 27 26 25	24 23 22 21	20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
[2]	P	0 0 0	n5	n4	n3	n2	n1	SAL

Table 11-3 PARTIAL DATA WORD

BIT	32	31 30 29	28 27 26 25	24 23 22 21	20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
[2]	P	0 0 1	1 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	n1	SAL
	P	0 0 1	1 0 0 1	0 0 0 0	0 0 0 0	n2	n1	SAL
[3]	P	0 0 1	1 0 1 0	0 0 0 0	n3	n2	n1	SAL
	P	0 0 1	1 0 1 1	n4	n3	n2	n1	SAL

Table 11-4 PROTOCOL WORD

BIT	32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
RTS	P	1 0 0	0 0 0 1	Destination Code [4]	Word Count [5]		SAL
CTS	P	1 0 0	0 0 1 0	Destination Code .	Word Count .		SAL
NCTS	P	1 0 0	0 0 1 1	Destination Code .	Status Code [6]		SAL
BUSY	P	1 0 0	0 1 0 0	Destination Code .	Status Code .		SAL
NAK	P	1 0 0	0 1 0 1	File Sequence Number	Status Code .		SAL
ACK	P	1 0 0	0 1 1 0	File Sequence Number	LDU Sequence Number		SAL
AL0	P	1 0 0	0 1 1 1	Subsystem SAL [7]	0 0 0 0	Version No. [10]	SAL
ALR	P	1 0 0	1 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0	Version No. [10]	SAL
SYN	P	1 0 0	1 0 0 1	File Sequence Number	Status Code [6]		SAL

c-16

ATTACHMENT 11
BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

Table 11-4A ALO-ALR VERSION NUMBER (See Appendix H)

c-16

12	11	10	9	APPLICABILITY	NOTES
0	0	0	0	Changes to Sec. 2.5 by Supplement 12 of ARINC 429, Part 3	8
0	0	0	1	Changes to Sec. 2.5 through Supplement 16 of ARINC 429, Part 3	8
0	0	1	0	Obsolete, formerly defined in Sec. 2.6 of ARINC 429, Part 3	
0	0	1	1	Section 3.0 of ARINC 429, Part 3	
0	1	0	0	Reserved	
		.		Reserved	
		.		Reserved	
1	1	1	0	Reserved	
1	1	1	1	Reserved	

c-16

Table 11-4B ALO-ALR WINDOW SIZE

This table deleted by Supplement 16.

Table 11-5 SOLO WORD

BIT	32	31	30	29	28	27	26	25	24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
TEST	P	1	0	1	0	0	0	0	16-Bit Test Pattern	SAL
LOOP	P	1	0	1	0	0	0	1	16-Bit Loop Back	SAL
SOLO	P	1	0	1	I. D.				16-Bit Data Field	SAL

Table 11-6 START OF TRANSMISSION

c-16

BIT	32	31	30	29	28 27 26 25	24 23 22 21 20 19 18 17	16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
SOT	P	1	1	0	GFI	File Sequence Number	LDU Sequence Number	SAL

ATTACHMENT 11
BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

Table 11-6A GENERAL FORMAT IDENTIFIER (GFI)

Bit				Description	Notes
28	27	26	25		
0	0	0	0	Reserved	1
0	0	0	1	Reserved	
0	0	1	0	Command-Control Data	
0	0	1	1	General Purpose Bit-Oriented Protocol (GPBOP)	
0	1	0	0	ISO 9577	
0	1	0	1	Reserved	
0	1	1	0	Reserved	
0	1	1	1	Reserved	
1	0	0	0	Reserved	
1	0	0	1	Reserved	
1	0	1	0	Reserved	
1	0	1	1	Reserved	
1	1	0	0	Reserved	
1	1	0	1	Reserved	
1	1	1	0	ACARS VHF Format	
1	1	1	1	Extended GFI	

c-17

NOTE:

[1] For consistency with Version 1, the GFI value (0000) is discouraged. The value 0000 should only be used for local communications between 2 LRUs that will never need to use a different file format on that ARINC 429 bus. The data format for the GFI '0000' is unique to that particular interface and is not defined herein.

Table 11-7 END OF TRANSMISSION

BIT	32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17 16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
EOT		1 1 1	0 0 0 X	LSB Cyclic Redundancy Check (CRC) 9	MSB	SAL

c-14

NOTE: Bit 25 of the EOT is the final LDU bit.

Table 11-7A FINAL LDU BIT

25	DESCRIPTION
0	Not Final LDU
1	Final LDU

ATTACHMENT 11
BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

Table 11-8 COMMAND FRAME SOF

Version 3 Command Frame SOF

BIT	32	31 30 29	28 27 26 25	24 23	22 21	20 19	18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
	P	0 1 0	GFI	CT	Reserved	I/C	ARINC 429 Word Count	U SAL

Where:

P – 32 Bit Parity
GFI – General Format Identifier (GFI) Field
CT – Command Type Field

00	Command Path
01	Data Path
10	MAC Control
11	Reserved

I/C – Information/Command Frame Identifier Field; 00-Information; 01-Command, 10, 11-Reserve
Word Count Field – 10 bits for a Command Frame, Max Command Frame size = 2552 bytes
c-17 U SAL – Unique (standard ARINC 429) System Address Label (SAL) (same as Version 1)

Table 11-9 COMMAND FRAME EOF

Version 3 Command Frame EOF

BIT	32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
	P	0 1 1	0 0 0 1	LSB FCS (16 Bit CRC) MSB	U SAL

Where:

P – 32 Bit Parity
Bit 25 (Command Frame Final Bit) for EOF word always set to 1
FCS – Frame Check Sequence, 16-bit CRC for Command Frame

Table 11-10 INFORMATION FRAME SOF

Version 3 Command Frame SOF

BIT	32	31 30 29	28 27 26 25 24 23 22 21	20 19	18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
	P	0 1 0	Reserved	I/C	ARINC 429 Word Count	M/B/U SAL

Where:

P – 32 Bit Parity
I/C – Information/Command Frame Identifier Field; 00-Information; 01-Command; 10, 11- Reserved
Word Count Field – 10 bits for an Information Frame – Max Information Frame Size = 2550 bytes, Limit for Bridgeability = 1500 bytes
M/B/U SAL – Multicast/Bridge/Unique (standard ARINC 429) System Address Label (SAL)

ATTACHMENT 11
BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

Table 11-11 INFORMATION FRAME EOF

Version 3 Command Frame EOF

BIT	32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
	P	0 1 1	0 0 0 0	FCS (32-Bit CRC) MSB	M/B/U SAL
	P	0 1 1	0 0 0 1	LSB FCS (32-Bit CRC)	M/B/U SAL

c-17

Where:

P – 32 Bit Parity
FCS – Frame Check Sequence, 32-bit CRC for Information Frame
Bit 25 for first half of Information Frame EOF word = 0
Bit 25 for second half of Information Frame EOF word = 1

ATTACHMENT 11
BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

NOTES

[1] All words are transmitted using odd parity (denoted by P in bit 32).

[2] The MSB/LSB determination of the "Data" field for Full and Partial Data words is as follows:

The ordering of octets (or ASCII characters) is from right to left, i.e.:

A) The first (most significant) octet is in bits 16-9 of the first Data word.

B) The second octet is in bits 24-17 of the first Data word ,etc.

The ordering of bits within each octet (or ASCII characters) is:

A) The LSB of the first octet is in bit 9 of the first Data word.

B) The MSB of the first octet is in bit 16 of the first Data word.

C) The LSB of the second octet is in bit 17 of the first Data word, etc.

EXAMPLE: The encoding of "PHX" (using 7-bit ASCII characters with all pad bits set to 0) using one Full Data Word and one Partial Data Word is:

BIT	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8 7 6 5 4 3 2 1
					"X" (Lower ½)				"H"								"P"								
Word	P	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0	0	0	SAL
																	"X" (Upper ½)								
Data Word 2	P	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	SAL

[3] Bits 24 through 31 of Partial Data Words are coded to avoid conflict with old RTS words, defined in Appendix F, which contains the ISO Alphabet control character "DC2" in bits 29 through 23.

[4] Destination Codes are contained in Attachment 11A.

[5] Word Count values of 0000 0000 through 0000 0010 are not used; i.e. values of Word Count may be 3 through 255 (03 through FFh).

c-16

[6] Status Codes are contained in Attachment 11B.

[7] This is the System Address Label of the system which is sending out the ALO.

[8] Versions 0 and 1 are equivalent for the ALOHA (ALO) and ALOHA RESPONSE (ALR) Protocol Words.

[9] The MSB/LSB determination for all Link Layer CRC Computation is:

c-14

Bit 9 of the first Data word is the MSB (coefficient of the highest order term) of the polynomial representing the "message". The LSB (coefficient of the lowest order term) of the polynomial representing the "message" is the most significant bit of the last octet or character (which contains valid non-zero data) which will be found in the final data word preceding the EOT Word.

c-14

[10] Bits 12 through 9 define the version number of the ARINC 429 bit-oriented protocol, as referenced in Table 11-4A.

ATTACHMENT 11A
DESTINATION CODES

DESTINATION	CHARARACTER CODE	BIT CODE [1] 2 2 2 2 2 1 1 1 4 3 2 1 0 9 8 7	NOTES	
CABIN TERMINAL (1-4)	1-4			
USER TERMINAL (5-8)	5-8			
FMC, LEFT SIDE	A	0 1 0 0 0 0 0 1		
FMC, RIGHT SIDE	B	0 1 0 0 0 0 1 0		
CONTROL DISPLAY UNIT	C	0 1 0 0 0 0 1 1		c-15
DFDAU	D	0 1 0 0 0 1 0 0		
CABIN PACKET DATA FUNCTION	E	0 1 0 0 0 1 0 1		c-15
CFDIU	F	0 1 0 0 0 1 1 0		
GROUND STATION	G	0 1 0 0 0 1 1 1		
HF LINK	H	0 1 0 0 1 0 0 0		
EICAS/ECAM/EFIS	I	0 1 0 0 1 0 0 1		
AUTOMATIC DEPENDENT SURVEILLANCE UNIT (ADSU)	J	0 1 0 0 1 0 1 0		c-15
KEYBOARD/DISPLAY	K	0 1 0 0 1 0 1 1		
Unassigned	*L*	0 1 0 0 1 1 0 0		c-18
ACARS MANAGEMENT UNIT (MU)	M	0 1 0 0 1 1 0 1		
FMC, CENTER	N	0 1 0 0 1 1 1 0		
OPTIONAL AUXILIARY TERMINAL	O	0 1 0 0 1 1 1 1		c-16
PRINTER	P	0 1 0 1 0 0 0 0		
SATELLITE DATA UNIT, LEFT SIDE (SDU 1)	Q	0 1 0 1 0 0 0 1	2	c-15
SATELLITE DATA UNIT, RIGHT SIDE (SDU 2)	R	0 1 0 1 0 0 1 0	2	
SATELLITE LINK	S	0 1 0 1 0 0 1 1		
HFDR Left	T	0 1 0 1 0 1 0 0	6	c-18
HFDR Right	*U*	0 1 0 1 0 1 0 1		
VHF LINK (VDR)	V	0 1 0 1 0 1 1 0		
TAWS	W	0 1 0 1 0 1 1 1		
CVR	X	0 1 0 1 1 0 0 0		
Unassigned	Y	0 1 0 1 1 0 0 1		
Unassigned	Z	0 1 0 1 1 0 1 0		c-18
INMARSAT SATELLITE LINK	CONTROL CODE VT	0 0 0 0 1 0 1 1	3, 4	
ICO SATELLITE LINK	CONTROL CODE CR	0 0 0 0 1 1 0 1	3, 4	
GLOBALSTAR SATELLITE LINK	CONTROL CODE SO	0 0 0 0 1 1 1 0	3, 4	

ATTACHMENT 11 A
DESTINATION CODES

NOTES:

c-18 [1] The Destination Code may be a 7-bit ISO Alphabet No. 5 code with no parity, or, alternatively, a binary value. Bit 24 should contain a zero.

[2] Origin/Destination Codes Q and R are used when the SDU transmits/receives onboard messages as an ACARS end system.

[3] These are non-printable control codes (reference ARINC 429 Part 1 Attachment 5).

[4] These codes are used for the specific satellite links indicated; character code "S" is used for any available non-specific satellite link.

c-18

[5] Origin/Destination Codes T and U are used when the HFDR transmits/receives onboard messages as an ACARS End System. The code H is distinguished from codes T and U in that the H is used to designate the HFDR as the downlink medium for a message while the T and U are used to indicate that the message is to be consumed by the DFDR.

[6] Transponder (XPDR) was removed from the DESTINATION column.

ATTACHMENT 11B
STATUS CODES

CODE (HEX)	DESCRIPTION	NAK	NCTS	BUSY	SYN
00	No Information	X	X	X	X
01	User Defined	X	X	X	X
02	User Defined	X	X	X	X
.	.	X	X	X	X
.	.	X	X	X	X
.	.	X	X	X	X
7E	User Defined	X	X	X	X
7F	User Defined	X	X	X	X
80	Missing SOT Word	X			
81	LDU Sequence Number Error	X			X
82	Invalid GFI	X			
83	Missing EOT Word	X			
84	Invalid Destination Code	X	X		
85	CRC Error	X			
86	LDU Time-Out Error	X			
87	Restart Initialization	X			X
88	Word Count Error	X	X		
89	Word Count Error/Input Overrun	X			
8A	Word Count Error/Parity Error	X			
8B	Sink Flow Control		X	X	
8C	Buffer Full		X	X	
8D	Device Off-Line			X	
8E	File Time-Out Error				X
8F	Window with Multiple FSNs				X
90	Missing LDU Control Word	X			
91	Remaining LDUs/Received LDUs Error	X			
92	Window Size Exceeds Sink's Receive Capability	X			
93	Invalid LDU Count in Window	X			
94	Invalid EOT in Window	X			
95	New File with Previous Incomplete				X
96	Reserved				
.	.				
.	.				
FE	Reserved				
FF	Reserved				

c-15

c-15

ATTACHMENT 11C
ALOHA/ALOHA RESPONSE PROTOCOL WORD DEFINITION

Table 11C-1 VERSION 1 ALOHA INITIAL (ALO) PROTOCOL WORD EXAMPLE

32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
P	Protocol	ALO	Subsystem SAL of originator [3]	Reserved [2]	Version Number [1]	SAL
	1 0 0	0 1 1 1		0 0 0 0	0 0 0 1	

[1] See Table 11-4A for version number.

c-16 [2] Reserved fields should be set to binary 0.

[3] This field should contain the SAL of the device sending the ALOHA word, with bit 17 as the most significant bit and bit 24 as the least significant bit of the subsystem SAL.

Table 11C-2 VERSION 1 ALOHA RESPONSE (ALR) PROTOCOL WORD EXAMPLE

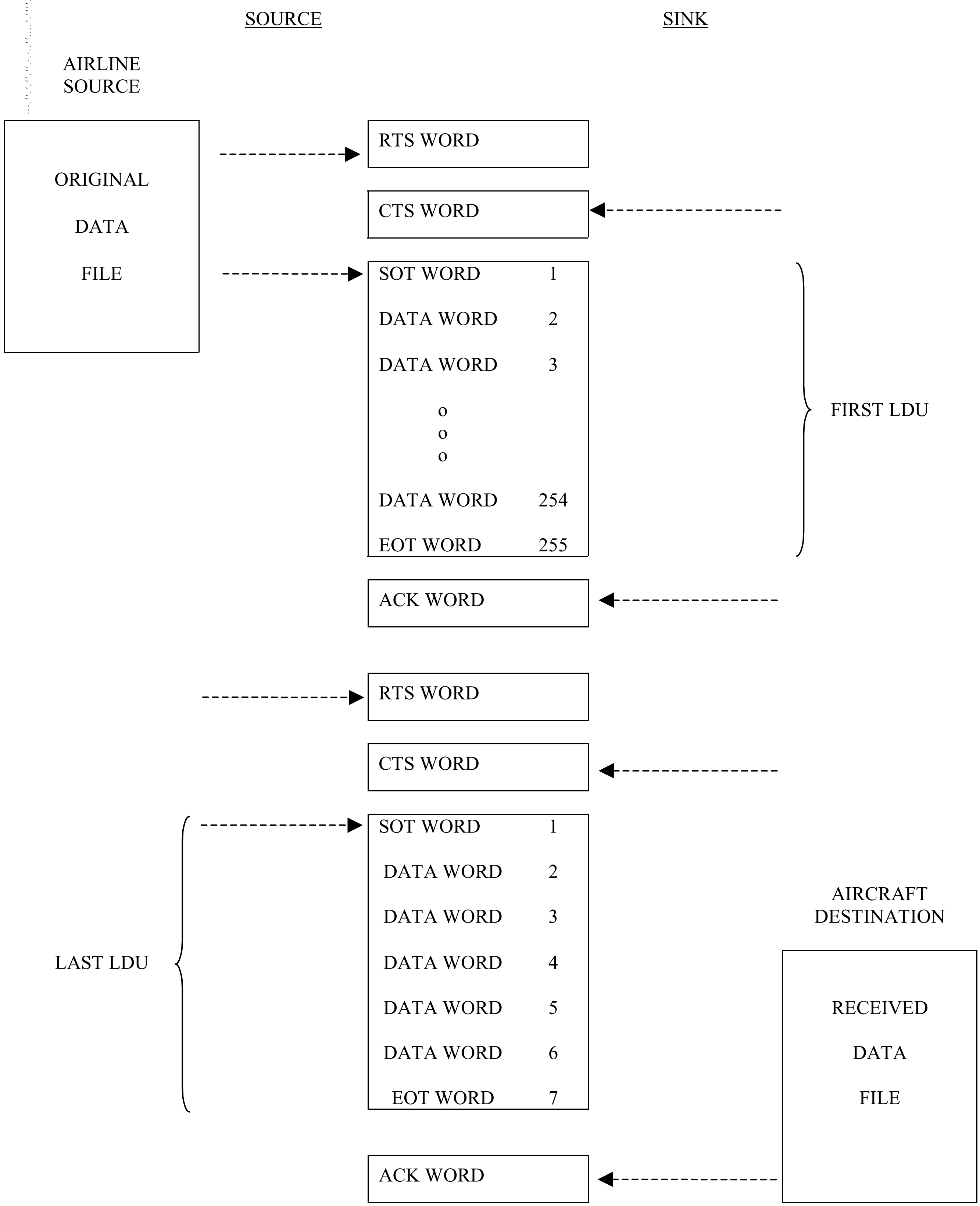
32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
P	Protocol	ALR	Reserved [2]		Ver. Number [1]	SAL
	1 0 0	1 0 0 0	0 0 0 0 0 0 0 0	0 0 0 0	0 0 0 1	

[1] See Table 11-4A for version number.

[2] Reserved fields should be set to binary 0.

NOTE: Table 11-C3 has been moved to Attachment 11, Table 11-4C.

ATTACHMENT 12
VERSION 1 FILE TRANSFER EXAMPLE



ATTACHMENT 12A
FIELD MAPPING EXAMPLE

	B 24	B 23	B 22	B 21	B 20	B 19	B 18	B 17	B 16	B 15	B 14	B 13	B 12	B 11	B 10	B 9	B 8	B 7	B 6	B 5	B 4	B 3	B 2	B 1
CRC	← DATA →																							
LSB MSB	LSB																							MSB

Semi octet

Octet

c-13

FIRST WORD

32	31 30 29	28 27 26 25	24 23 22 21	20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5	4 3 2 1
		n5	n4	↔	↔	↔		
P	0 0 0	B20 B17	B16 B13	B12 n3 B9	B8 n2 B5	B4 n1 B1	SAL	

SECOND WORD

32	31 30 29	28 27 26 25	24 23 22 21	20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
						← n1 →	
P	0 0 1	1 0 0 0	0 0 0 0	0 0 0 0	0 0 0 0	B24 B21	SAL

c-14

EOT WORD

32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
P	1 1 1	0 0 0 *	LSB CRC MSB	SAL

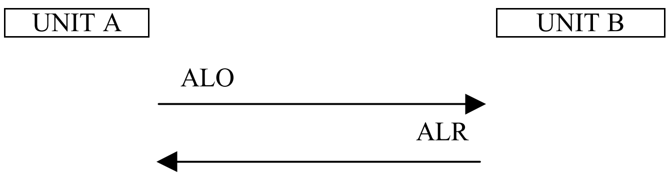
* Final LDU Bit

This example takes 24 bits of data and puts it into two data words. For the CRC computation, the MSB is the bit position that represents the coefficient of the highest order term of the polynomial.

ATTACHMENT 13
PROTOCOL DETERMINATION PROCEDURE DIAGRAMS

Diagram 13-1 PROTOCOL DETERMINATION PROCEDURE DIAGRAM (TWO BILINGUAL UNITS)

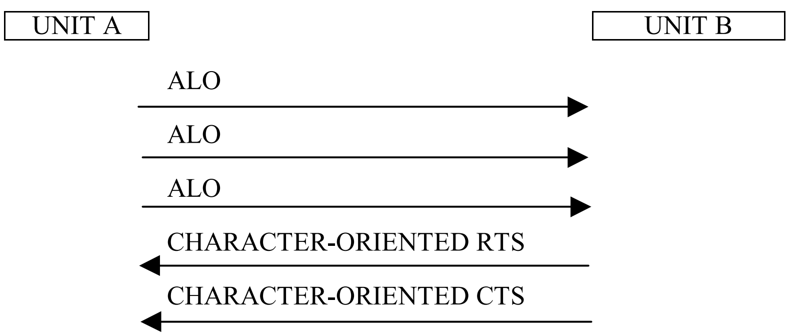
Assume that both Unit A and Unit B are capable of communicating using both ARINC 429 character-oriented file transfer format or bit-oriented file transfer format. Assume Unit A will initiate the exchange to determine the protocol to be used.



BIT-ORIENTED COMMUNICATIONS ESTABLISHED

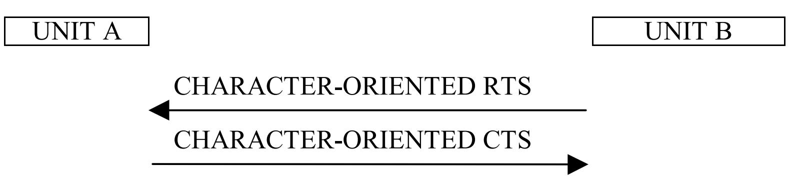
Diagram 13-2 PROTOCOL DETERMINATION PROCEDURE DIAGRAM (ONE BILINGUAL UNIT AND ONE CHARACTER-ONLY UNIT)

Assume that Unit A is capable of communicating using both ARINC 429 character-oriented file transfer format or bit-oriented file transfer format. Assume that Unit B is only capable of communicating using the ARINC 429 character-oriented file transfer format. Assume Unit A will initiate the exchange to determine the protocol to be used.



CHARACTER-ORIENTED COMMUNICATIONS ESTABLISHED

Assume Unit B will initiate the exchange to determine the protocol to be used.

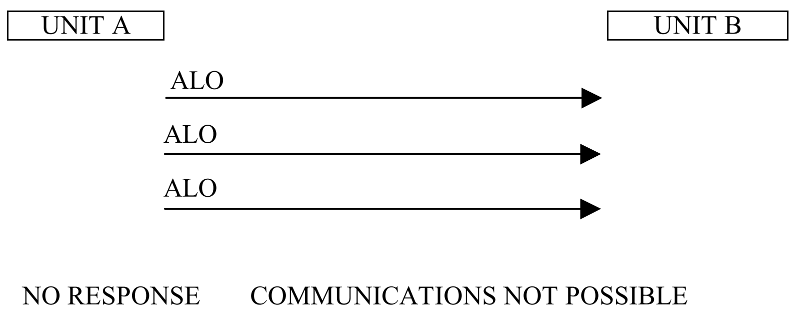


CHARACTER-ORIENTED COMMUNICATIONS ESTABLISHED

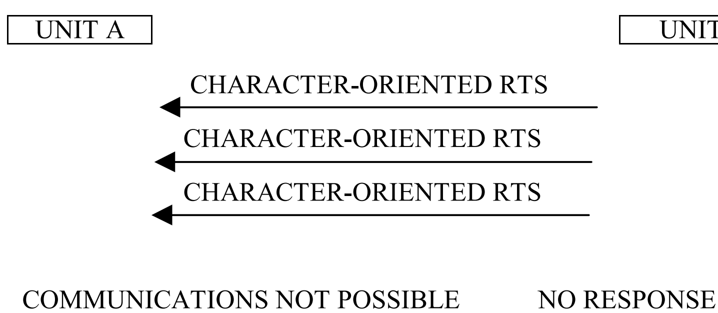
ATTACHMENT 13 (cont'd)
PROTOCOL DETERMINATION PROCEDURE DIAGRAMS

Diagram 13-3 PROTOCOL DETERMINATION PROCEDURE DIAGRAM (TWO CHARACTER-ONLY UNITS)

Assume that Unit A is only capable of communicating using the bit-oriented file transfer format. Assume that Unit B is capable of communicating only using the ARINC 429 character-oriented file transfer format. Assume Unit A will initiate the exchange to determine the protocol to be used. The resulting "No Response" conclusion accurately identifies the inability of the two units to communicate.

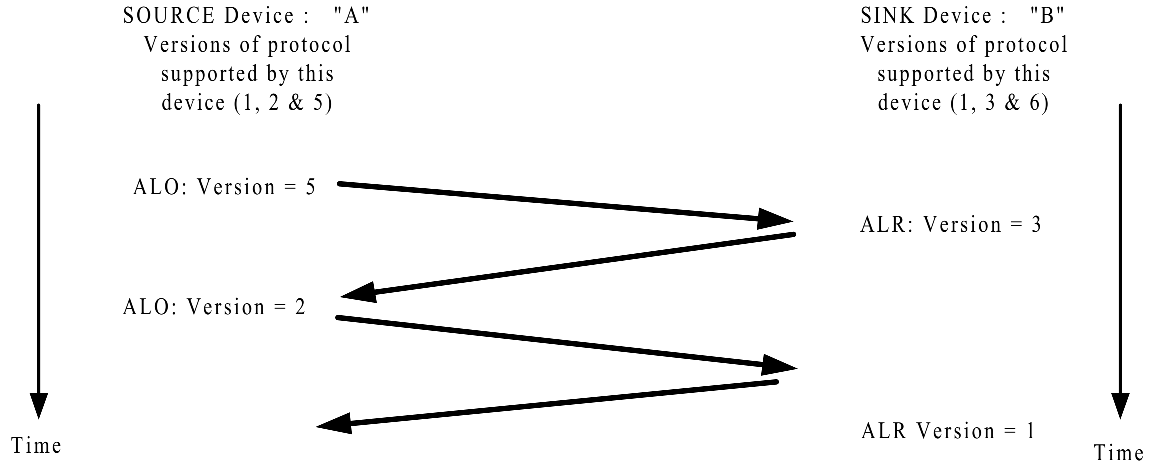


Assume Unit B will initiate the exchange. The resulting “No Response” conclusion accurately identifies the inability of the two units to communicate.

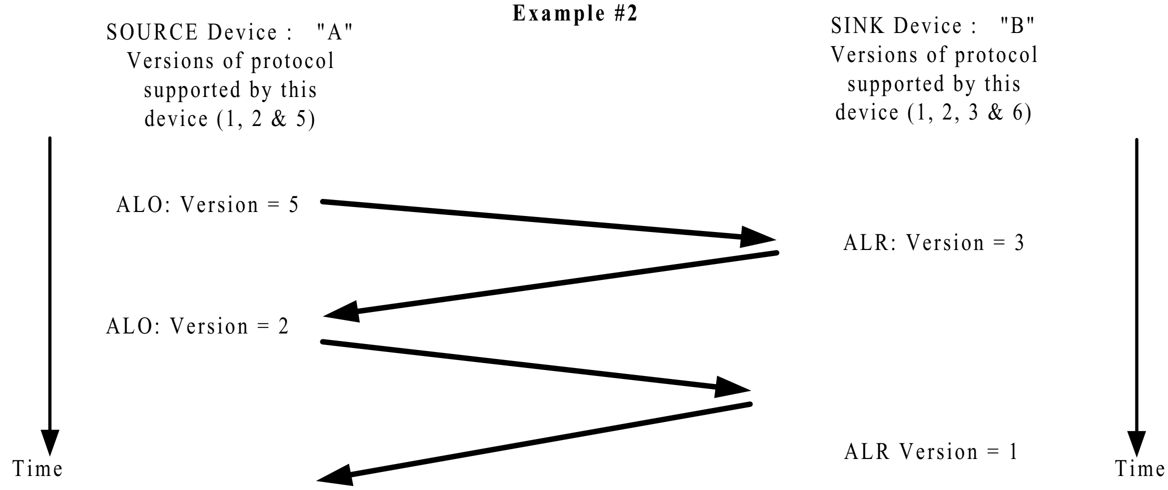


ATTACHMENT 13A
ALOHA VERSION DETERMINATION SEQUENCE

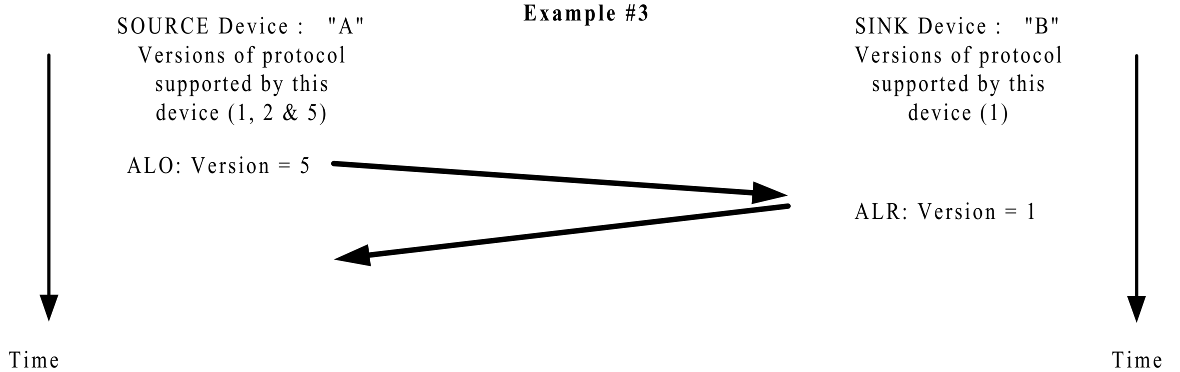
Example #1



Example #2



Example #3



ATTACHMENT 14
SYSTEM ADDRESS LABELS

c-16 The listing of System Address Label (SAL) assignments is contained in Attachment 11 to ARINC Specification 429, Part 1.

ATTACHMENT 15

NOTE: This attachment has been deleted by Supplement 16.

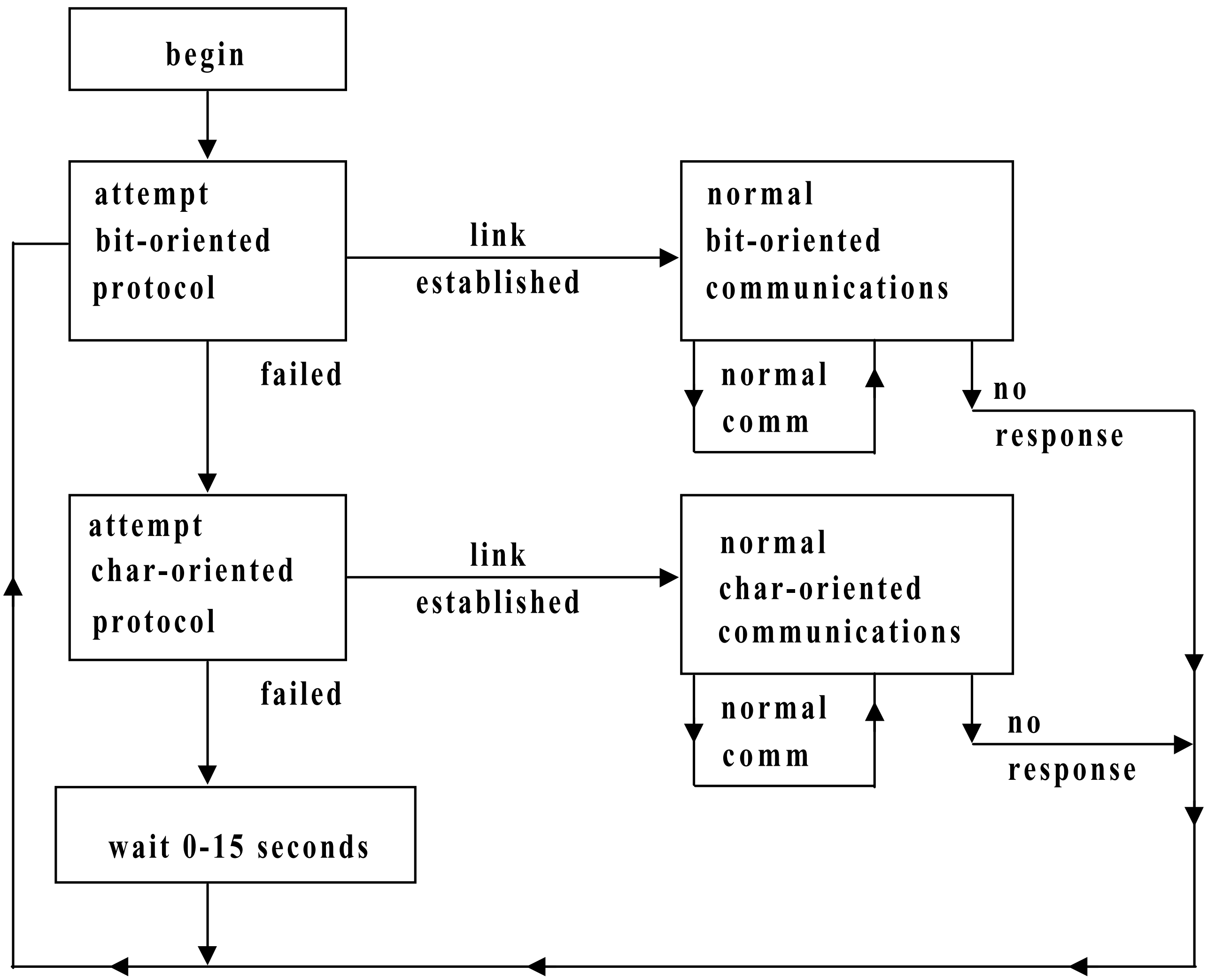
This Attachment number is not used in this Specification to maintain consistency with previous versions of ARINC Specification 429 prior to its separation into 3 parts by Supplement 15 and to avoid confusion among the parts. c-16

ATTACHMENT 16

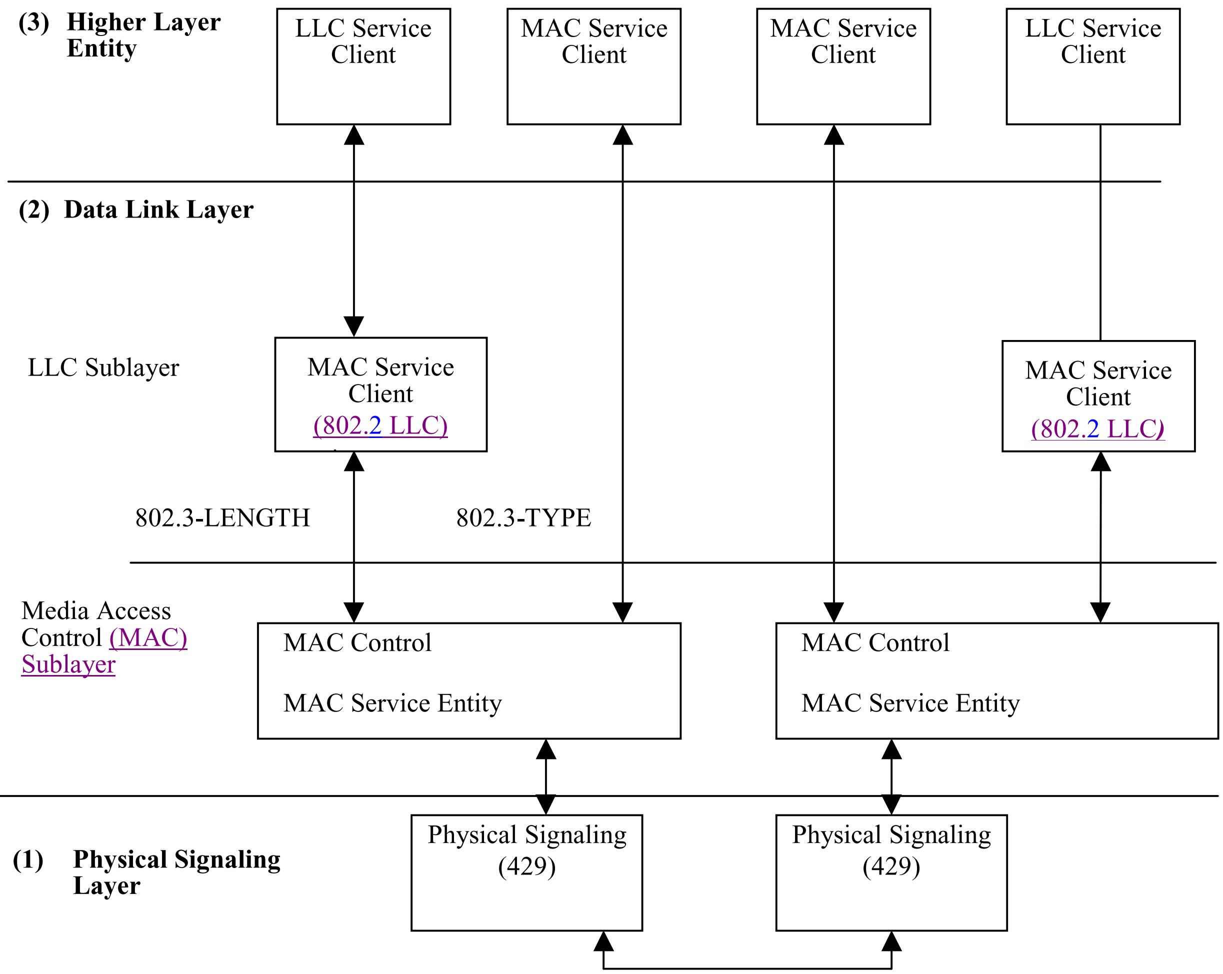
NOTE: This attachment has been deleted by Supplement 16.

c-16 This Attachment number is not used in this Specification to maintain consistency with previous versions of ARINC Specification 429 prior to its separation into 3 parts by Supplement 15 and to avoid confusion among the parts.

ATTACHMENT 17
FLOW DIAGRAM USED TO DETERMINE CHARACTER-ORIENTED VS BIT-ORIENTED PROTOCOL



ATTACHMENT 18
MAC SUBLAYER SUPPORT DIAGRAMS

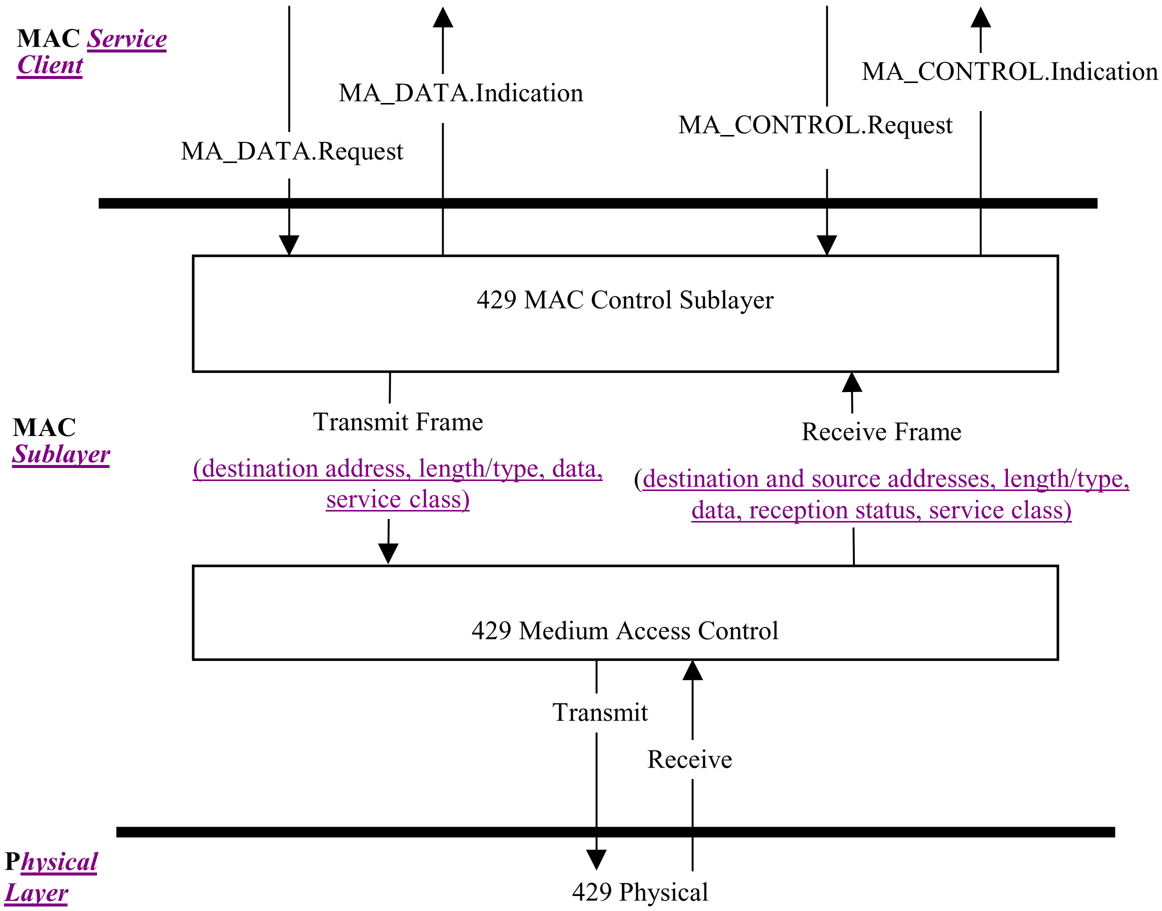


c-17

NOTE: The MAC Control Sublayer is normally a pass through except for MAC Control PDUs that are processed by the MAC Control sublayer entity and are not passed to higher layers. The operation of the MAC Control Sublayer is defined in IEEE 802.3 Clause 31. MAC Control Functions are defined Section 3.2.4 of this Specification.

Figure 18-1 - MAC Sublayer and its Service Clients

ATTACHMENT 18
MAC SUBLAYER SUPPORT DIAGRAMS



c-17

Figure 18-2 - MAC Control Sublayer Support Of Interlayer Service Interfaces

ATTACHMENT 19
COMMAND FRAME DATA UNIT (FDU) STRUCTURE AND EXAMPLES

Table 19-1 V3 Command Frame SOF

BIT	32	31 30 29	28 27 26 25	24 23	22 21	20 19	18 17 16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
	P	0 1 0	GFI	CT	Rsvd	I/C	ARINC 429 Word Count		U SAL

Where:

P – 32 bit Parity
GPI – General Format Identifier (GFI) Field
CT – Command Type Field

00	Command Path
01	Data Path
10	MAC Control
11	Reserved

I/C – Information/Command Frame Identifier Field; 00 – Information; 01 – Command, 10, 11 – Reserved
Word Count Field – 10 bits for a Command Frame, Max Command Frame size = 2552 bytes
U SAL – Unique (standard ARINC 429) System Address Label (SAL) (same as Version 1)

c-17

Table 19-2 V3 Command Frame (Full and Partial) Data Words

BIT	32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17 16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
	P	0 0 0	DATA			U SAL
	P	0 0 0	DATA			U SAL
	P	0 0 0	DATA			U SAL
	P	0 0 0	DATA			U SAL

Table 19-3 V3 Command Frame EOF

BIT	32	31 30 29	28 27 26 25	24 23 22 21 20 19 18 17 16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
		0 1 1	0 0 0 1	LSB FCS (16-Bit CRC)	MSB	U SAL

Where:

Bit 25 (Command Frame Final Bit) for EOF word always set to 1
FCS – Frame Check Sequence, 16-bit CRC for Command Frame

ATTACHMENT 19
COMMAND FRAME DATA UNIT (FDU) STRUCTURE AND EXAMPLES

Table 19-4 Example of Command Frame Data Unit (FDU) Containing a MAC Control Frame Requesting a 5 Pause Quanta (25 millisecond) Delay

BIT	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8 7 6 5 4 3 2 1

V3 Command Frame SOF:

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8 7 6 5 4 3 2 1
	P	0	1	0	X	X	X	X	1	0	0	0	0	1	0	0	0	0	0	0	0	1	0	0	U SAL

V3 Command Frame (Full and Partial) Data Words: c-17

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8 7 6 5 4 3 2 1
	P	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	U SAL
	P	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	U SAL

V3 Command Frame EOF:

	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8 7 6 5 4 3 2 1
	P	0	1	1	0	0	0	1	LSB						FCS (16-Bit CRC)									MSB	U SAL

Where:

GFI Field = X’s as placeholder for actual values used over interface (See Table 11-6A of Attachment 11)
Pause Opcode – 16 bit field = 8808 hex
Request Operand – 16 bit field = Pause Time = 5 Pause Quanta = 0005 hex

ATTACHMENT 20
INFORMATION FRAME DATA UNIT (FDU) STRUCTURE AND EXAMPLE

Table 20-1 Version 3 Information Frame SOF

BIT	32	31 30 29	28 27 26 25 24 23 22 21	20 19	18 17 16 15 14 13 12 11 10 9	8 7 6 5 4 3 2 1
	P	0 1 0	Reserved	I/C	ARINC 429 Word Count	M/B/U SAL

Where:
- P – 32 bit Parity
- I/C – Information/Command Frame Identifier Field; 00-Information; 01-Command; 10, 11-Reserved
- Word Count Field – 10 bits for an Information Frame – Max Information Frame Size = 2550 bytes, Limit for Bridging = 1500 bytes
- M/B/U SAL – Multicast/Bridge/Unique (standard ARINC 429) System Address Label (SAL)

Table 20-2 Version 3 Information Frame (Full and Partial) Data Words

BIT	32	31 30 29	28 27 26 25	24 23 22 21	20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
	P	0 0 0			Destination		LSB	M/B/U SAL
	P	0 0 0			Destination			M/B/U SAL
	P	0 0 0	Source		LSB	MSB	Destination	M/B/U SAL
	P	0 0 0			Source			M/B/U SAL
	P	0 0 0	Length/Type	LSB MSB	Source			M/B/U SAL
	P	0 0 0	Data		MSB	Length/Type		M/B/U SAL
c-17	P	0 0 0			Data			M/B/U SAL
	P	0 0 0			Data			M/B/U SAL
	P	0 0 0			Data			M/B/U SAL

Where:
- P – 32 bit Parity
- Destination – 48 Bit MAC Destination Address
- Source – 48 Bit MAC Source Address
- Length/Type – 16 bit field

Table 20-3 Version 3 Information Frame EOF

BIT	32	31 30 29	28 27 26 25	24 23 22 21	20 19 18 17	16 15 14 13	12 11 10 9	8 7 6 5 4 3 2 1
	P	0 1 1	0 0 0 0		FCS (32-Bit CRC)		MSB	M/B/U SAL
	P	0 1 1	0 0 0 1	LSB	FCS (32-Bit CRC)			M/B/U SAL

Where:
- FCS – Frame Check Sequence, 32-bit CRC for Information Frame
- Bit 25 for first half of Information Frame EOF word = 0
- Bit 25 for second half of Information Frame EOF word = 1

ATTACHMENT 20
INFORMATION FRAME DATA UNIT (FDU) STRUCTURE AND EXAMPLE

Table 20-4: Example of Information Frame Data Unit (FDU) Containing a MAC Control Frame Requesting a 5 Pause Quanta (25 millisecond) Delay

BIT	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8 7 6 5 4 3 2 1

V3 Information Frame SOF:

P	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	U SAL

V3 Information Frame (Full and Partial) Data Words: c-17

P	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	U SAL
P	0	0	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	U SAL
P	0	0	0	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	X	X	X	X	X	X	X	X	U SAL
P	0	0	0	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	U SAL
P	0	0	0	1	0	0	0	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	U SAL
P	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	U SAL
P	0	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	U SAL

V3 Information Frame EOF:

P	0	1	1	0	0	0	0		FCS (32-Bit CRC) MSB	U SAL
P	0	1	1	0	0	0	1	LSB	FCS (32-Bit CRC)	U SAL

Where:

Destination – 48 Bit MAC Destination Address = X's as placeholders (see ARINC Specification 664)
Source – 48 Bit MAC Source Address = Y's as placeholders (see ARINC Specification 664)
Length/Type – 16 bit field = Pause Opcode = 8808 hex
Request Operand – 16 bit field = Pause Time = 5 Pause Quanta = 0005 hex

APPENDICES A - E

c-16 Appendices A through E are included in ARINC Specification 429, Part 1, ARINC Specification 429 and therefore not used in this Part to avoid potential confusion due to duplication. In addition, this approach is used to maintain consistency with previous versions of ARINC Specification 429 when it was published as a whole (through Supplement 14).

APPENDIX F
FORMER AIM AND FILE DATA TRANSFER TECHNIQUES

AEEC Staff Note: See Supplements 4, 5, 6, 7. And 11 of ARINC Specification 429 Part 1 for changes prior to division of ARINC Specification 429 into separate parts. c-18

The information contained in Sections F-2.1.5.2, F-2.1.5.3, F-2.3.1.4 and F-3.2 of this Appendix is no longer applicable to ARINC Specification 429. The contents of Section F-2.3.1.5 provides guidance for character-oriented file transfer protocols reflected in ARINC equipment characteristics. The information is contained herein for reference purposes.

COMMENTARY

Bit-oriented file transfer is the preferred protocol (See Section 2.5 of Part 3 of ARINC Specification 429) for use in new applications. The guidance for character-oriented protocol was removed from the body of the specification to avoid the continuation of its use, but retained herein for those who need to understand the basis for character-oriented protocol already implemented.

F-2.1.5 Sign/Status Matrix

F-2.1.5.2 AIM Data

The order function (first, intermediate, last or control) of AIM or maintenance data should be encoded in bit numbers 30 and 31 of the word as shown in the table below. See Section F-2.3.1.4 of this document for definitions of the terms “Initial Word”, “Control Word”, “Intermediate Word” and “Final Word.”

AIM DATA SIGN/STATUS MATRIX	
BIT 31 30	MEANING
0 0	Intermediate Word
0 1	Initial Word
1 0	Final Word
1 1	Control Word

F-2.1.5.3 Character-Oriented File Transfer

The order and function (first, intermediate, last and control) of text and the sign (Plus/Minus, North/South, etc.) of numeric data transferred by file transfer should be encoded in bits 30 and 31 of each word as shown in the table below.

CHARACTER-ORIENTED FILE TRANSFER STATUS MATRIX		
BIT 31 30	TEXT	DATA
0 0	Intermediate Word	Plus, North, etc.
0 1	Initial Word	Not Defined
1 0	Final Word	Not Defined
1 1	Intermediate Word	Minus, South etc.

Alternatively, the SSM field may be used to describe position in a series of data words (initial, intermediate, final) or word content (control word). Sections F-2.3.1.5.2 through F-2.3.1.5.4 contain the definitions of the terms initial, intermediate and final words.

F-2.3.1 Digital Language

F-2.3.1.4 AIM Data

AIM data (Acknowledgement, ISO Alphabet No. 5 and Maintenance information encoded in dedicated words) should be handled in the manner described in this section.

All three of these applications may involve the transfer of more than 21 bits per “data package”. Source equipment should format such long messages into groups of 32-bit DITS words, each word containing the relevant application label (see ARINC Specification 429, Part 1, Attachment 1) in bits 1 through 8, and a sign/status matrix code in bits 30 and 31.

APPENDIX F
FORMER AIM AND FILE DATA TRANSFER TECHNIQUES

Bit 32 should be encoded to render word parity odd. The first word of each group should contain the sign/status matrix code defined for "initial word" in F-2.1.5.1. It should also contain, in bits 9 through 16, the binary representation of the number of words in the group, except that when this word is the only word to be transmitted, the total number of information bits to be transmitted is 13 or less) bits 9 through 16 should all be binary "zeros". See ARINC Specification 429, Part 1, Attachment 6 for word format.

When the word application label is assigned in ARINC Specification 429, Part 1, Attachment 1 for Acknowledgement Data, bits 17 through 29 of this initial word may be used for information transfer. When the word application label is either of those assigned in ARINC Specification 429, Part 1, Attachment 1 Maintenance Data (ISO Alphabet No. 5), bits 17 through 22 should be binary 'zeros" (spares). When the label is for ISO Alphabet No. 5 Messages, bits 17 through 22 are used for unit addressing. Bit usage is given in the table below.

BIT						FUNCTION
22	21	20	19	18	17	
0	0	0	0	0	0	All Call, All Groups
0	0	X	X	X	X	Group 0, Units 1-15
0	1	0	0	0	0	Group 1, All Call
0	1	X	X	X	X	Group 1, Units 1-15
1	0	0	0	0	0	Group 2, All Call
1	0	X	X	X	X	Group 2, Units 1-15
1	1	0	0	0	0	Group 3, All Call
1	1	X	X	X	X	Group 3, Units 1-15

Example:

1	0	1	0	1	0	Group 2, Unit 10

For ISO Alphabet No. 5 Messages and Maintenance Data bits 23 through 29 should take on the pattern of the IOS Alphabet No. 5 control character "STX".

The second word of the ISO Alphabet No. 5 and Maintenance Data (ISO Alphabet No. 5) application groups is an optional control word containing sign/status matrix code for "control" information for display. When it is used, bits 9 through 13 should contain the binary representation of the line count, bits 14 through 16 should encode the required color, bits 17 and 18 the required intensity, bits 19 and 20 the required character size and bit 21 should indicate whether or not the display is required to flash. See ARINC Specification 429, Part 1, Attachment 6 for the encoding standards. Bits 22 through 29 of the word should be binary "zeros" (spares).

Intermediate words, containing the sign/matrix code for "intermediate word", follow the initial word of the group or the control word, when used. Intermediate words are optional in the sense that they are only transmitted if more words than the initial word and the final word (see below) are needed to accommodate the quantity of information to be transferred. When the word application group label that is assigned in ARINC Specification 429, Part 1, Attachment 1 for Acknowledgement is used. Data bits 9 through 29 of that word are available for information transfer. When the word application label is either of those assigned in ARINC Specification 429, Part 1, Attachment 1 for ISO Alphabet No. 5 data transfer or Maintenance Data (ISO Alphabet No. 5), bits 9 through 29 of each word should be divided into three seven-bit bytes (bits 9 through 15, 16 through 22 and 23 through 29), each of which contains one ISO Alphabet No. 5 character.

Each AIM application group transmission other than single-word transmission (see below) should be terminated with a word containing the sign/status matrix code for "final word" defined in F-2.1.5.1. The data field of this word should be structured similarly to that of the intermediate word. Any unused bit positions in ISO Alphabet No. 5 data transfer or Maintenance Data (ISO Alphabet No. 5) final words resulting from the number of ISO Alphabet No. 5 characters in the message being one or two less than a number wholly divisible by three should be filled with binary "zeros."

APPENDIX F
FORMER AIM AND FILE DATA TRANSFER TECHNIQUES

F-2.3.1.5 File Data Transfer

F-2.3.1.5.1 Command/Response Protocol

File data will consist of both ARINC 429 BNR numeric words and ISO Alphabet No. 5 characters. A file may contain from 1 to 127 records. Each record may contain from 1 to 126 data words.

A record should contain, at the minimum, one of the eight versions of the "initial word" described in F-2.3.1.5.2. Records in which this initial word contains the "Data Follows" code should also contain from 1 to 126 "intermediate words" (data) and a "final word" (error control). The file data transfer protocol is as follows. A transmitter having the data to send to a receiver transmits, on the bus connecting it to that receiver, the "Request to Send" initial word. The receiver responds, on the separate bus provided for return data flow, with the "Clear to Send" reply. The transmitter then sends the "Data Follows: initial word, the "intermediate words" and the "final word". The receiver processes the error control information in the 'final word" and, if no errors are revealed, closes out the transaction by sending the "Data Received OK" word to the transmitter.

If the receiver is not ready to accept data when the transmitter sends its "Request to Send" word, it should so indicate its response (See F-2.3.1.5.2). The transmitter should then wait 200 milliseconds and retransmit the "Request to Send". The transmitter should also repeat a "Request to Send" transmission 50 milliseconds after the initial transmission if no response is obtained from the receiver. If 2 additional attempts also spaced at 50 milliseconds produce no response from the receiver, the transmitter should send the data. This feature is incorporated to enable file transfer (under a degraded mode of operation) in the event of a failure in the receiver-to-transmitter bus.

If the receiver detects a parity error during the transmission, it may request an error-correcting retransmission by sending a "Data Received Not OK" word to the transmitter in which is identified the record in which the error occurred. The transmitter should interrupt the data flow and back up to the start of the record so identified. It should then send a "Data Follows" initial word identifying this record as the starting point of the retransmission and recommence its output of data, continuing through the "final word". The receiver should then close out the transaction as before.

An error detected by processing the error control information in the "final word" should also result in the receiver sending a "Data Received Not OK" word to the transmitter. In the absence of identification of the record in which the error occurred, this word should contain the sequence number of the first record of the file. The transmitter's response should be to retransmit the whole file.

The receiver can signal loss of synchronization to the transmitter at any time bysending the "Synchronization Lost" initial word. On receiving this word, the transmitter should curtail the data flow and back up to the beginning of the file. It should then re-establish that the receiver can accept data by going through the request-to-send routine. Having done this it should send the "Data Follows" initial word, followed by the data and the "final word".

The protocol also allows a transmitter to send the file size information to a receiver without any commitment to send, or request to the receiver to accept, the file itself. The "Header Information" initial word is used for this purpose. Additionally, a "Poll" initial word is defined for use in the system which continuous "hand-shaking" between two terminals is desired. The response to a "Poll" word will be either a "Request" to Send" initial word when the polled terminal does have data to transmit, or another "Poll" word when it does not. An exchange of "Poll" words may be interpreted as the message, "I have nothing for you, do you have anything for me?"

F-2.3.1.5.2 Initial Word Types

The eight initial types are as follows:

Request to Send
Clear to Send
Data Follows
Data Received OK
Data Received Not OK
Synchronization Lose
Header Information
Poll

Bits 1 through 8 of all of those words except the "Poll" word contain the label code identifying the file to be transferred using the protocol. Bits 1 through 8 of the "Poll" word contains binary zeros. Bits 9 through 29 are divided into three seven-bit fields, the contents of which vary with word type as shown in Table A below. Bits 30 and 31 contain the code identifying them as initial words while bit 32 is encoded to render word parity odd.

APPENDIX F
FORMER AIM AND FILE DATA TRANSFER TECHNIQUES

NOTES

1. The amount of data the receiver can accept upon receipt of a "Request to Send" signal is determined by the rate at which data delivery can take place and the amount of time the receiver has available before it must turn its attention to some other function. The receiver will set the count code in bits 9-15 of the "Clear to Send" word to indicate the number of maximum length records it can accept when it determines that the "Request to Send signal originates in a high speed data source. It will set this code to indicate the number of 32-bit words it can accept when it determines that the "Request to Send" originates in a low speed data source, e.g., the ACARS ground-to-air link. The receiver will annunciate the contents of this field (record count or word count) by setting Bit 22 as indicated. It will determine the high or low speed nature of the source by port identification of the source of the "Request to Send" signal, the "Request to Send" word label, the SDI code or some combination of these information items.

2. The record sequence number is the number of that record in a multiple-record file being transmitted.

F-2.3.1.5.3 Intermediate Words

Intermediate words contain the data being transmitted by means of the protocol. Bits 1 through 8 contain the file label. Bits 9 through 29 can accommodate three ISO Alphabet No. 5 characters or one ARINC 429 BNR numeric word without its label. Note that this alpha/numeric data interleaving capability without labels necessitates a prior agreement between transmitter and receiver on data format. Bits 30 and 31 contain the word code or the sign information (only) encoded in the sign/status matrix of BNR numeric data words. Bit 32 is encoded to render the word parity odd.

F-2.3.1.5.4 Final Words

The final word of each record contains error control information. Bits 1 through 8 contain the file label. Bits 9 through 29 contain an error control checksum computed from the states of bits 9 through 29 of all intermediate words of the record. The error control checksum should be generated by the arithmetic addition of the binary values of bits 9 through 29 of all intermediate words and discarding the overflow. Bits 30 and 31 of this word contain the code identifying it as a final word. Bit 32 is encoded to render the word parity odd.

F-2.3.1.5.5 Word Type Encoding

Bits 30 and 31 of each word used in data file transfer should be encoded to indicate word type as follows:

BIT 31	BIT 30	WORD TYPE
0	0	Intermediate Word requiring no sign data or having Plus, North, East, Above Right, or To sign
0	1	Initial Word (all types)
1	0	Final Word
1	1	Intermediate Word having Minus, South, West, Below, Left, or From sign

F-2.3.1.5.6 File Data Formats

As noted in F-2.3.1.5.3, the transmission of file data words without labels necessitates the use of pre-arranged data formats. The need to standardize such formats was examined by the working group. The conclusion was reached that a standard format was desirable for flight management computer flight plan updating and for computer cross-talk, but was not necessary for updating the computer's data base. Manufacturers are invited to submit proposals for a standard flight plan update file and cross-talk bus formats.

F-2.3.1.5.7 File Data Labels

Labels define the application of the file data to be transferred. Such application include FMC program load/update, flight plan load/update, the FMC inter-system cross-talk, etc. There may be a need to assign more than one label to some of those applications if priority override capability is desired.

APPENDIX F
FORMER AIM AND FILE DATA TRANSFER TECHNIQUES

F-3.2 AIM Information Transfer

F-2.3.1.4 describes the techniques to be used for the transfer of Acknowledgement, ISO Alphabet No. 5 and Maintenance (ISO Alphabet No. 5) data by means of the Mark 33 DITS. The motivation for the adoption of this technique was label conservation. Without it, a separate label would have to be assigned to each AIM word application for each source of such data. In it, labels are assigned by word application only, and (where necessary) utilization device input port recognition utilized to identify sources. A special exception to this rule is made for the Airborne Integrated Data System (AIDS), as described in F-2.3.1.4. The technique also accommodates the use of multiple-word DITS messages, as described in Section F-2.3.1.4.

F-3.2.1 Acknowledgement Data

Source equipments responding to requests for acknowledgement of incoming data delivered via a DITS input port should do so in the manner described in F-2.3.1.4. No applications for this system capability have yet been identified and thus no data standards for acknowledgement messages have been established.

F-3.2.2 ISO Alphabet No. 5 Data

Source equipment transmitting ISO Alphabet No. 5 information by means of the Mark 33 DITS should do so in the manner described in F-2.3.1.4. This application, and the use of ISO data for maintenance-related information transfer, will be the most likely to make use of the multiple-word message transmission capability of the system. Receiving equipment should make use of the binary word count and the sign/status matrix codes of the words to ensure that such messages are received in their entirety, with no words having been "lost along the way." Only when this determination has been made, and the parity check for each word shows the data to be error-free, should the message be displayed to the crew or otherwise utilized.

F-3.2.3 Maintenance Data

Source equipment putting out data intended for a maintenance assist system on the aircraft should do so in the manner described in F-2.3.1.4. The Maintenance assist system should use input port recognition to identify data sources. The Maintenance word as described by AIM data handling techniques is limited to IOS Alphabet No. 5 messages.

APPENDIX F
FORMER AIM AND FILE DATA TRANSFER TECHNIQUES

TABLE A - INITIAL WORD TYPES

INITIAL WORD TYPE	FIELD (Bits)	FIELD CONTENTS
Request to Send (Tx-to-Rx)	9-15 16-22 23-29	BNR count number of records to be sent (Max. 127) Binary zeros ISO Alphabet No. 5 Control Character "DC2"
Clear to Send (Rx-to-Tx) See Note 1	9-15 16-21 22 23-29	Binary zeros when receiver is <u>not</u> <u>ready</u> to accept data. BNR count of the number of maximum length of records <u>OR</u> the number of 32-bit words the receiver can accept when it is ready Binary zeros Binary zero when receiver is not ready to receive data <u>and</u> when BNR counts in 9-15 is record count. Binary one when count in bits 9-15 is 32-bit word count. ISO Alphabet Control Characters "DC3"
Data Follows (Tx-to-Rx) See Note 2	9-15 16-22 23-29	BNR count of number of words in record (Max. 126) Record Sequence No. (BNR) ISO Alphabet No. 5 Control Character "STX"
Data Received OK (Rx-to-Tx)	9-15 16-22	BNR count of number of words in record (Max. 126) Record Sequence No. (BNR) ISO Alphabet No. 5 Control Character "ACK"
Data Received Not OK (Rx-to-Tx)	9-15 16-22 23-29	BNR count number of words in record Record sequence no. (BNR) in which error occurred ISO Alphabet No. 5 Control Character "NAK"
Synchronization Lost (Rx-to-Tx)	9-15 16-22 23-29	Binary Zeros Binary Zeros ISO Alphabet No. 5 Control Character "SYN"
Header Information (Tx-to-Rx)	9-15 16-22 23-29	BNR count of number of records in file to be transferred Binary Zeros ISO Alphabet No. 5 Control Character "SOH"
Poll (B-directional)	9-15 16-22 23-29	Binary Zeros Binary Zeros ISO Alphabet No. 5 Control Character "ENQ"

APPENDIX G
MATHEMATICAL EXAMPLE OF CRC ENCODING/DECODING

NOTE: The following example describes the polynomial division procedure for CRC encoding and decoding. Arithmetic operations are modulo 2. Actual software/hardware implementations are expected to vary significantly from this example, since these polynomial divisions are more efficiently simulated by logical operations.

For CRC computations, the MSB is the bit which represents the coefficient of the highest order term of the polynomial. It is not related to the MSB or LSB of each individual octet. Slashes (/) are used to separate octets for readability only, and do not denote division in this example.

The following (arbitrary) 24-bit message is to be transmitted with a CRC encoded:

10100111 / 10000111 / 10101100

(MSB) (LSB)

The mathematical procedure is as follows:
For this message, k = 24, and

$$G(x) = x^{23} + x^{21} + x^{18} + x^{17} + x^{16} + x^{15} + x^{10} + x^{9} + x^{8} + x^{7} + x^{5} + x^{3} + x^{2}.$$

$$x^{16}G(x) = x^{16}(x^{23} + x^{21} + x^{18} + x^{17} + \ldots + x^{7} + x^{5} + x^{3} + x^{2}).$$

$$= x^{39} + x^{37} + x^{34} + x^{33} + \ldots + x^{23} + x^{21} + x^{19} + x^{18}.$$

and

$$x^{k}(x^{15} + x^{14} + x^{13} + x^{12} + x^{11} + \ldots + x^{3} + x^{2} + x + 1)$$

$$= x^{39} + x^{38} + x^{37} + x^{36} + \ldots + x^{27} + x^{26} + x^{25} + x^{24}.$$

At The Transmitter: Using coefficients of the above polynomials, the dividend is calculated as follows:

$x^{16}G(x)$ = 1010 0111 ┆ 1000 0111 ┆ 1010 1100 ┆ 0000 0000 ┆ 0000 0000

and

$x^{k}(x^{15} + x^{14} + \ldots + x^{2} + x + 1)$ = 1111 1111 ┆ 1111 1111 ┆ 0000 0000 ┆ 0000 0000 ┆ 0000 0000

	1010	0111	1000	0111	1010	1100	0000	0000	0000	0000
+	1111	1111	1111	1111	0000	0000	0000	0000	0000	0000
=	0101	1000	0111	1000	1010	1100	0000	0000	0000	0000

Then the Dividend is: 0101 1000 ┆ 0111 1000 ┆ 1010 1100 ┆ 0000 0000 ┆ 0000 0000

and the Divisor, $P(x) = x^{16} + x^{12} + x^{5} + 1$, is: 1 ┆ 0001 0000 ┆ 0010 0001

APPENDIX G
MATHEMATICAL EXAMPLE OF CRC ENCODING/DECODING

P(x) = 1 0001 0000 0010 0001 is the divisor of the dividend below.

(Q(x), the quotient generated by the division process, is not used).

```
                    Q(x)  =   101  1101  1010  1001  1101  1001
         ____________________________________________________________
         0101  1000  0111  1000  1010  1100  0000  0000  0000  0000
          100  0100  0000  1000  01
         ----------------------------
            1  1100  0111  0000  1110
            1  0001  0000  0010  0001
         ----------------------------
               1101  0111  0010  1111  1
               1000  1000  0001  0000  1
               ----------------------------
                101  1111  0011  1111  01
                100  0100  0000  1000  01
                ---------------------------
                  1  1011  0011  0111  0000
                  1  0001  0000  0010  0001
                  --------------------------
                     1010  0011  0101  0101  0
                     1000  1000  0001  0000  1
                     --------------------------
                       10  1011  0100  0001  100
                       10  0010  0000  0100  001
                       -------------------------
                           1001  0100  0101  1010  0
                           1000  1000  0001  0000  1
                           -------------------------
                              1  1100  0100  1010  1

                                   1000  1000  0001  0000  1
                                   -------------------------
                                      1  0001  0010  1101  1000
                                      1  0001  0000  0010  0001
                                         ----------------------
                               R(x) =    0000  0010  1111  1001

                     ____
          CRC  =     R(x)  =   1111  1101  0000  0110
```

Using (synthetic) Polynomial Division:

Note: Since all operations are Modulo 2, addition and subtraction are both equivalent to XOR operations (no carries!)

```
M(x)   =   x^16 G(x) + CRC

       =   1010  0111  1000  0111  1010  1100  0000  0000  0000  0000
       +                                       1111  1101  0000  0110
           ----------------------------------------------------------
       =   1010  0111  1000  0111  1010  1100  1111  1101  0000  0110

   or

M(x)   =   1010  0111  1000  0111  1010  1100  1111  1101  0000  0110
           MSB         (message)         LSB   MSB     (CRC)      LSB
```

M(x) is the transmitted message with CRC.

APPENDIX G
MATHEMATICAL EXAMPLE OF CRC ENCODING/DECODING

At The Receiver:

The dividend to be operated on by P(x) is determined (mathematically) as follows:

$x^{16}M(x) + x^{40}(x^{15} + x^{14} + x^{13} + x^{12} + ... + x^2 + x + 1)$

=	1010	0111	1000	0111	1010	1100	1111	1101	0000	0110	0000	0000	0000	0000
+	1111	1111	1111	1111	0000	0000	0000	0000	0000	0000	0000	0000	0000	0000
=	0101	1000	0111	1000	1010	1100	1111	1101	0000	0110	0000	0000	0000	0000

The above string is the dividend used by the receiver.

(The divisor, 10001000000100001, is the same as that used by the transmitter.)

(The quotient, Q(x), generated by the division process, is not used.)

```
                    Q(x)   =     101   1101   1010   1001   1101   1001    .      .      .      .
0101  1000  0111  1000  1010  1100  1111  1101  0000  0110  0000  0000  0000  0000
 100  0100  0000  1000  01
   1  1100  0111  0000  1110
   1  0001  0000  0010  0001
      1101  0111  0010  1111  1
      1000  1000  0001  0000  1
       101  1111  0011  1111  01
       100  0100  0000  1000  01
         1  1011  0011  0111  0000

                          1001  0110  1110  1101  0
                          1000  1000  0001  0000  1
                             1  1110  1111  1101  1110
                             1  0001  0000  0010  0001
                                1111  1111  1111  1111
```

Note: At this point, if the division is carried through to termination, the final remainder is:

Rr(x) = 0001 1101 0000 1111

APPENDIX G
MATHEMATICAL EXAMPLE OF CRC ENCODING/DECODING

LDU Mapping for 24-bit Example

Because of the transmission order of ARINC 429 32-bit words, the first bit of the first Data Word transmitted after the SAL is the MSB of the message (for CRC computations). Therefore, the actual transmission order of the bit string, M(x), is the reverse of the previous example, when mapped into 32-bit words.

The following represents the mapping of the preceding 24-bit message and CRC into an LDU for transmission:

		MSB		(message)			LSB	MSB	(CRC)		LSB
M(x)	=	1010	0111	1000	0111	1010	1100	1111	1101	0000	0110

SOT Word	P	110	GFI	File No.		LDU No.		SAL	LDU
Full Data Word	P	000	0101	1110	0001	1110	0101	SAL	
Partial Data Word	P	001	1000	0000	0000	0000	0011	SAL	
EOT Word	P	111	0001	0110	0000	1011	1111	SAL	

Legend:

The CRC is calculated over these bits.

These are the CRC bits

APPENDIX H
INTEROPERABILITY OF BIT-ORIENTED LINK LAYER PROTOCOL

H.1 Version Number Designators

The version number of a system is transmitted to the peer system in the ALO and ALR words.

- Version 1 systems are defined to be (Williamsburg) bit-oriented communication protocol systems, as defined in Section 2.0 this Specification.
- Version 2 systems are obsolete and have been deleted from ARINC Specification 429 - Part 3.
- Version 3 systems are defined to provide a standard MAC-based sublayer of the bit-oriented communications-protocol, and are defined in Section 3.0 of ARINC 429 - Part 3. c-16

H.2 Interoperability - Same Version Number

Beginning with Supplement 13 of 429, any version 1 system should interoperate with any other version 1 system (i.e., the version number should be independent of supplement number implementation.) For example, a version 1 429-13 system should interoperate with a version 1 429-14 system.

Similarly, any version 3 system should interoperate with any other version 3 system (independent of supplement number).

In general, higher supplement numbers of the same version number clarify requirements.

H.3 Interoperability - Different Version Numbers

c-17

Version 1 and Version 3 can coexist on the same physical bus.

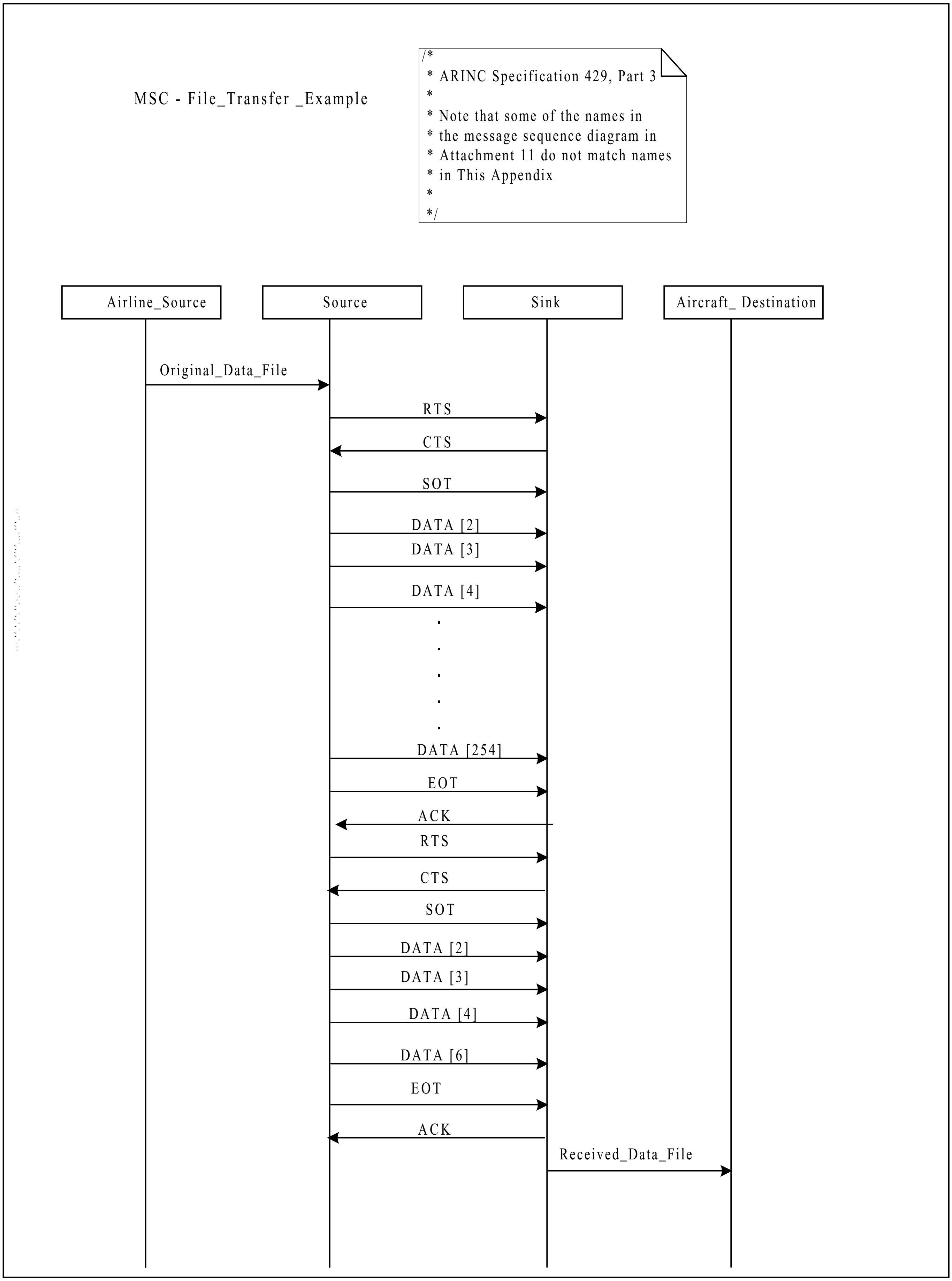
H.4 Bit-Oriented Link Layer GFIs for Standard Network Service

When a system implements a standard (e.g. ISO) network service, which resides directly above the 429 Version 1 (Williamsburg), then the ISO GFI bit-coding of "4h" (for ISO 9577) as specified in ARINC 429-14 should be used. c-16

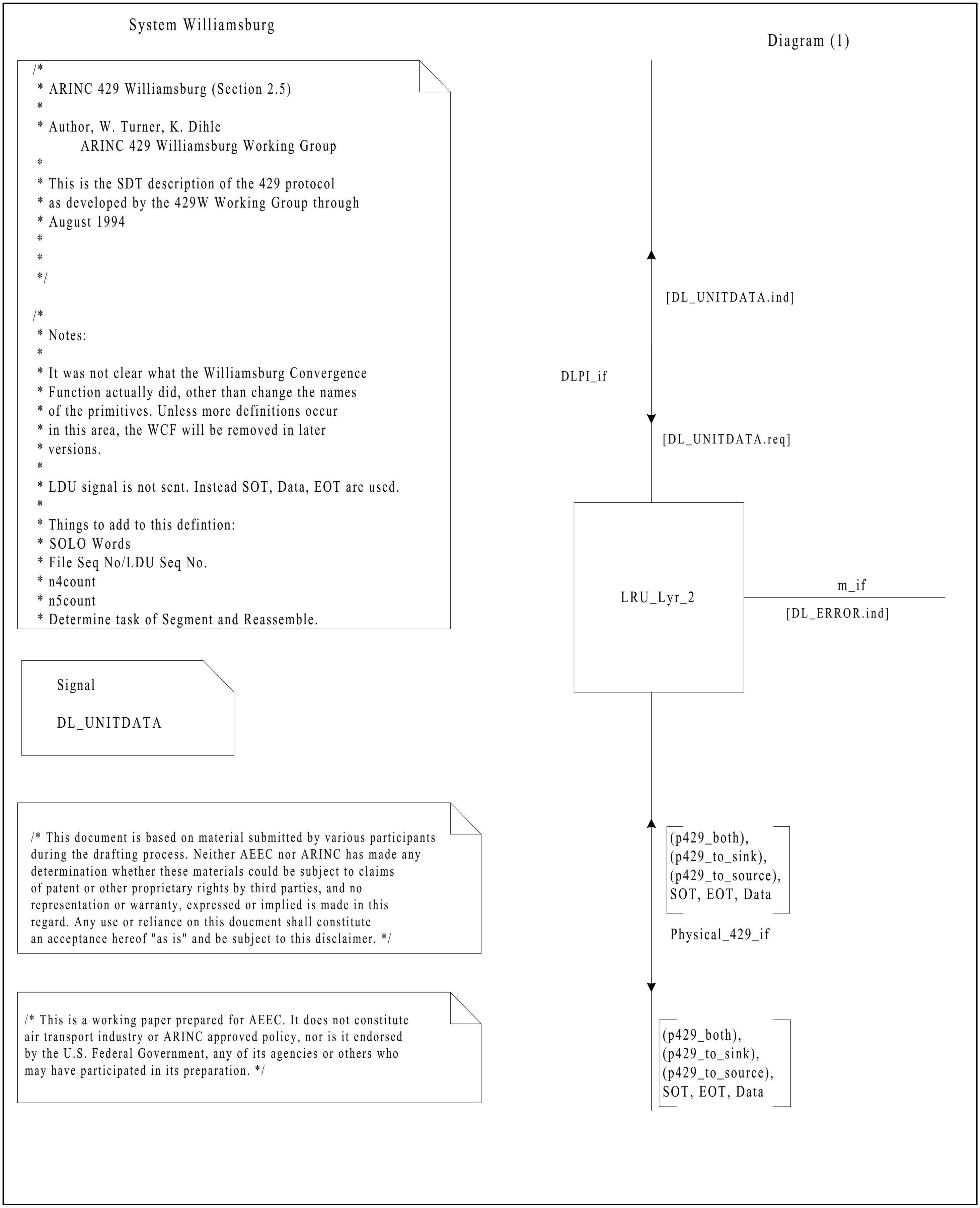
ISO GFI bit-codings originally specified in ARINC 429-13 were "1h" for ISO 8208 and "4h" for ISO 8473. These GFI bit-codings are not compatible with the new GFI "4h" designator and should not be used as standard network service identifiers.

APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL

c-15



APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



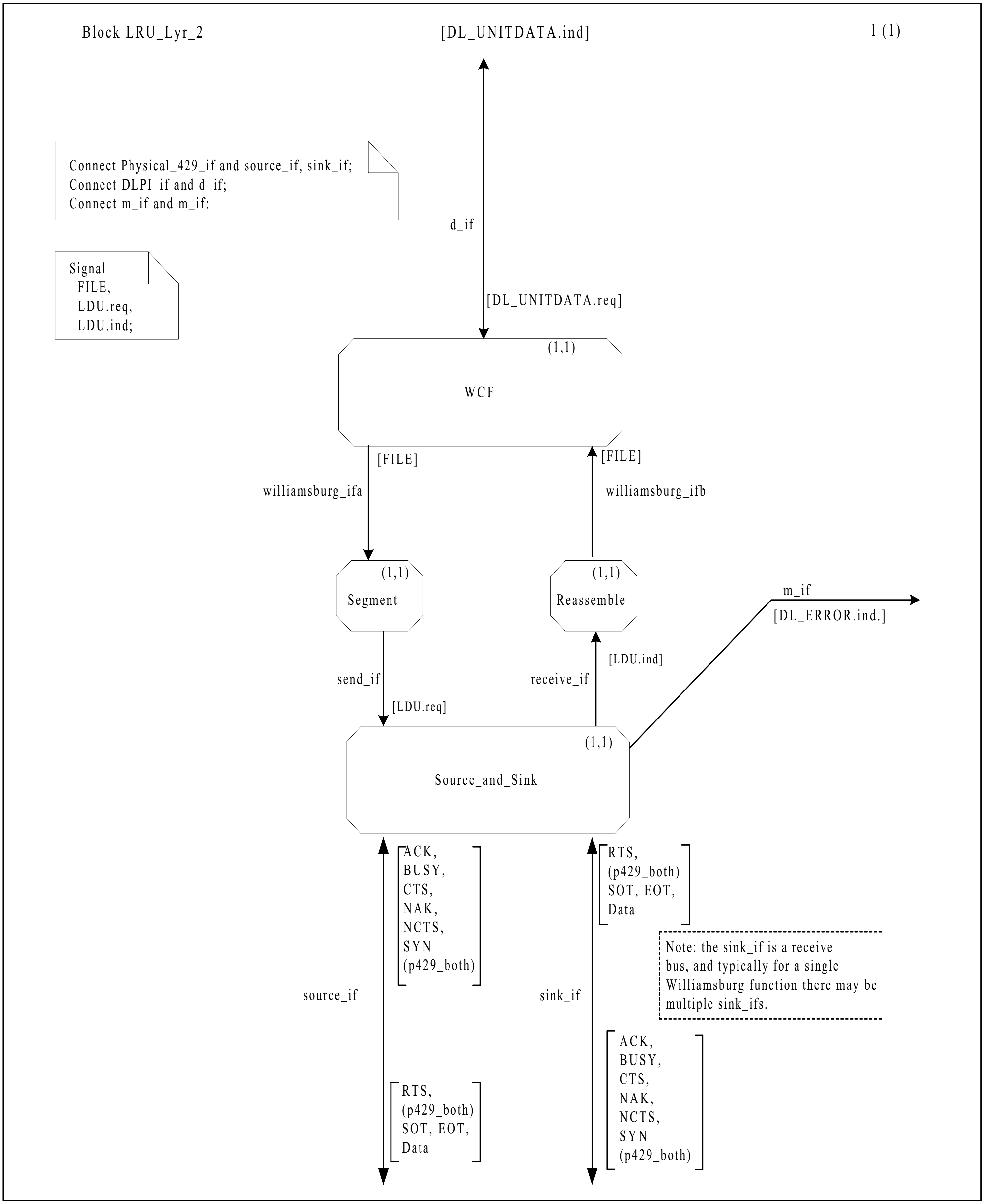
c-15

APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL

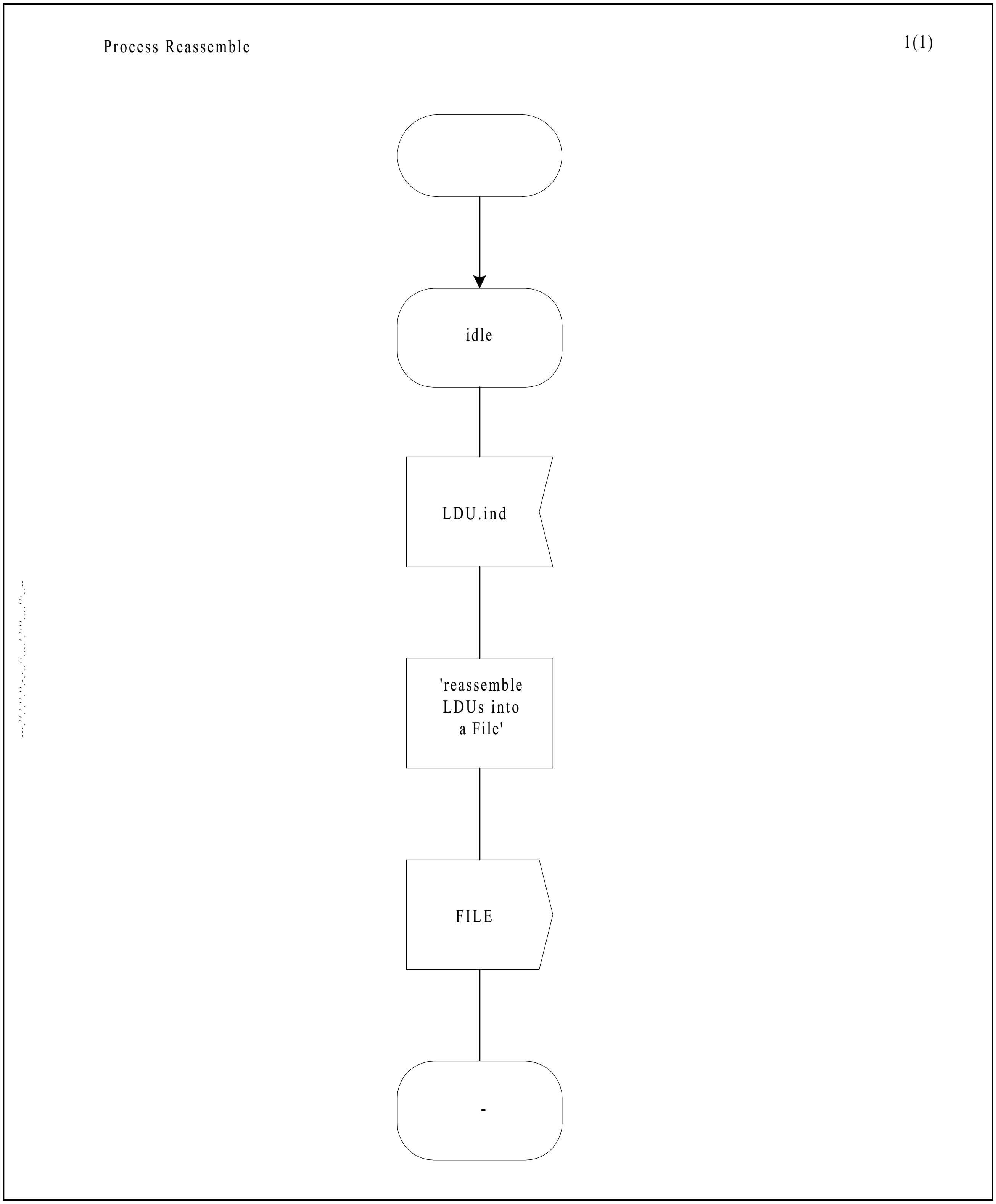
The following definitions are used:

```
    Signal

    DL_UNITDATA.req,
    DL_UNITDATA.ind,
    DL_ERROR.ind,

    /* Wilmsbrg*/
    /* -----*/

ACK,        /*2.5.1.4     */
ALO,        /*2.5.19.1.1  */
ALR,        /*2.5.19.1.2  */
BUSY,       /*2.5.7.3     */
CTS,        /*2.5.7.1     */
Data,       /*2.5.11      */
EOT,        /*2.5.12      */
NAK,        /*2.5.13      */
NCTS,       /*2.5.7.2     */
RTS,        /*2.5.7       */
SOT,        /*2.5.10      */
SYN,        /*2.5.15      */

/*LOOP,     2.5.17.1      */
/*SOLO,     2.5.17.2      */
/*TEST,     2.5.17.2      */
/*LCW                     */

Signalist p429_both =
    ALO<ALR/* LOOP,SOLO,TEST */;

Signalist p429_to_sink =
    RTS;

Signalist p429_to_source =
    ACK,BUSY,CTS,NAK,NCTS,SYN;
```

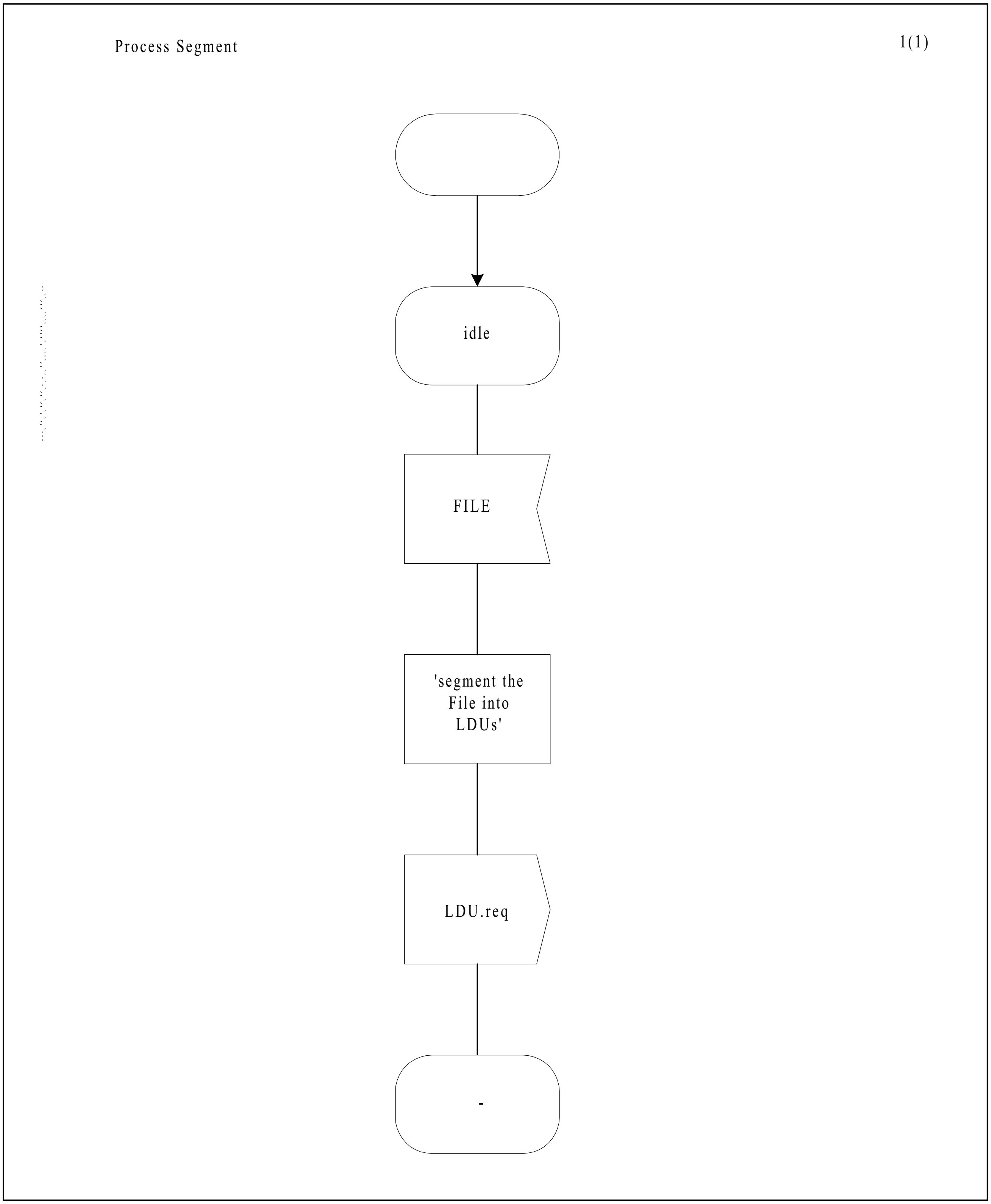
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL

Process Source_and_Sink

1 (12)

c-15

```
/*
 * Bit-Oriented Protocol Events
 * Part 3, Table 10-1
 */
Synonym N1 integer =  5;  /*Number of RTS repeats from NCTS */
Synonym N2 integer = 20;  /*Number of RTS repeats after BUSY */
Synonym N3 integer =  5;  /*Number of RTS repeats after no resp */
Synonym N4 integer =  3;  /*Number of NAKs rcvd before NO COMM */
Synonym N5 integer =  3;  /*Number of SYN before NO COMM */
Synonym N6 integer =  3;  /*Number of ALO after no response*/
/*
 * Selected Options
 * Table 10-3
 *
 * Option 1 Half Duplex
 * Option 2 Low Speed Bus Rate
 * Option 3 Automatic CTS when Ready = No
 * Option 4 Accept Auto CTS = No
 * Option 5 System Priority to resolve RTS Conflict = No
 * Option 8 Use Solo Word = Yes
 * Option 10 Dest Code in RTS/CTS/NCTS/BUSY used = Yes
 * Option 11 Bit-Oriented Protocol Vertification = Yes
 */
```

c-15

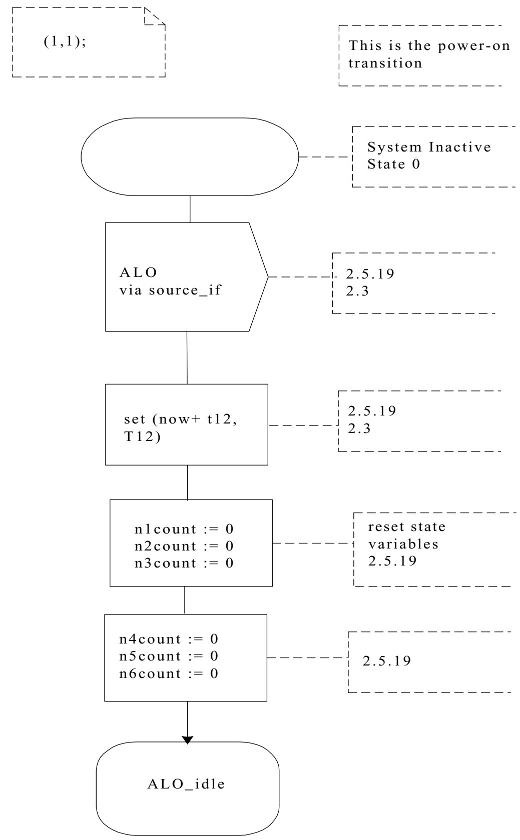
```
/*
 * Variables for Low Speed Bit-Oriented Protocol
   Part 3, Table 10-4
*/

/* Timer for Source */

Synonym t2 duration = 500; /* RTS repeat time after receipt of NCTS */
Synonym t4 duration = 15000; /* RTS repeat time after receipt of BUSY */
Synonym t5 duration = 500; /* RTS repeat time  if no response*/
Synonym t12 duration = 200; /* ALO repeat time  if no response to ALO*/
Synonym t16 duration = 220; /* ACK/NAK Timeout after EOT*/
```

c-15

```
/* Timer for Sink  */

Synonym t9 duration = 2500; /* LDU timeout following CTS */
Synonym t14 duration = 120000; /* Incomplete File Timeout */
/*
 * Variables for High Speed Bit-Oriented Protocol
 * Part 3, Table 10-5
*/

/* Timer for Source */

Synonym t2 duration = 100; /* RTS repeat time after receipt of NCTS */
Synonym t4 duration = 1000; /* RTS repeat time after receipt of BUSY */
Synonym t5 duration = 150; /* RTS repeat time  if no response*/
Synonym t12 duration = 200; /* ALO repeat time  if no response to ALO*/
Synonym t16 duration = 220; /* ACK/NAK Timeout after EOT*/
```

c-15

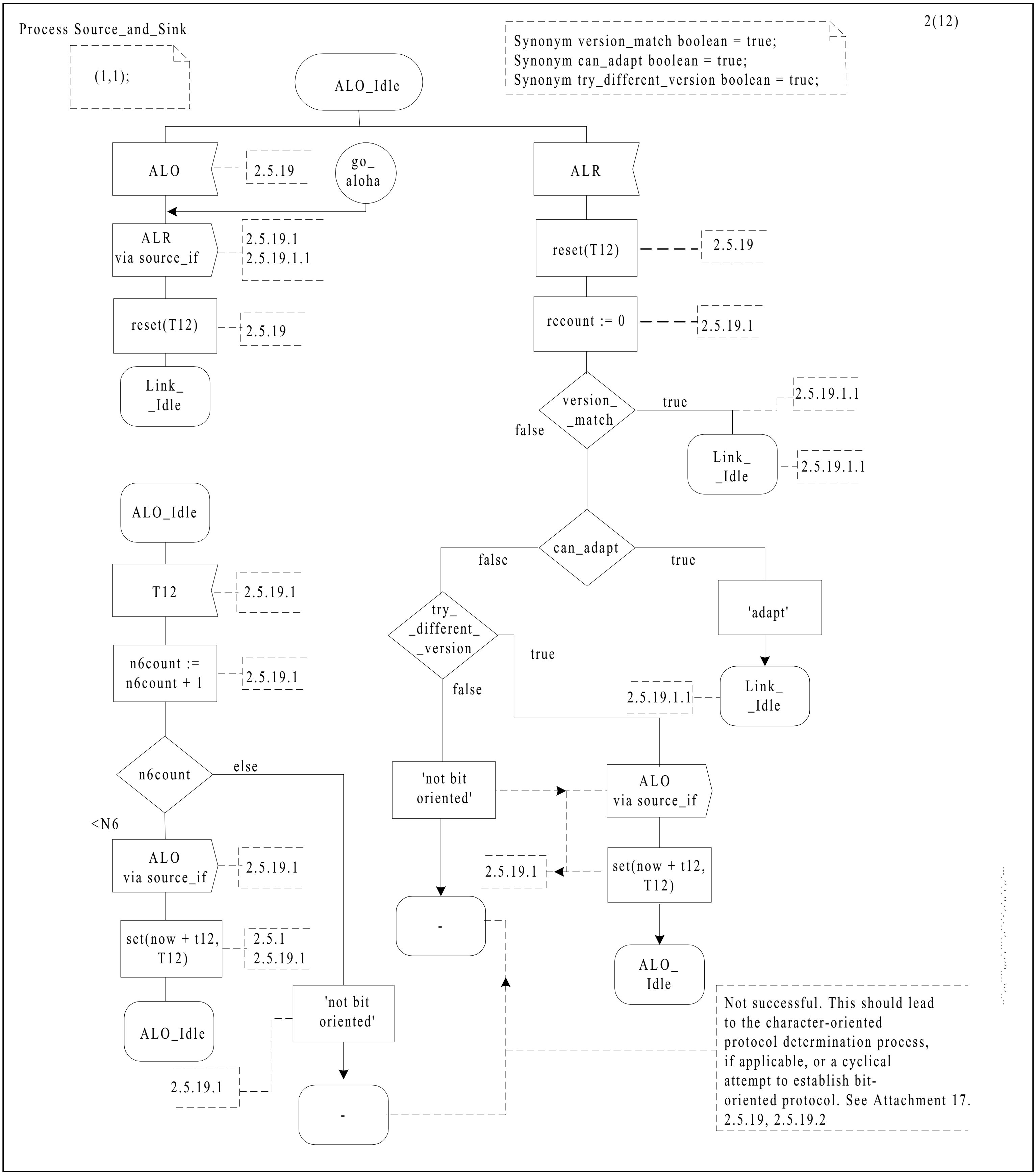
```
/* Timer for Sink */

/* Synonym t9 duration = 400  /* LDU timeout following CTS */
/* Synonym t14 duration = 10000' /* Incomplete File Timeout */
```



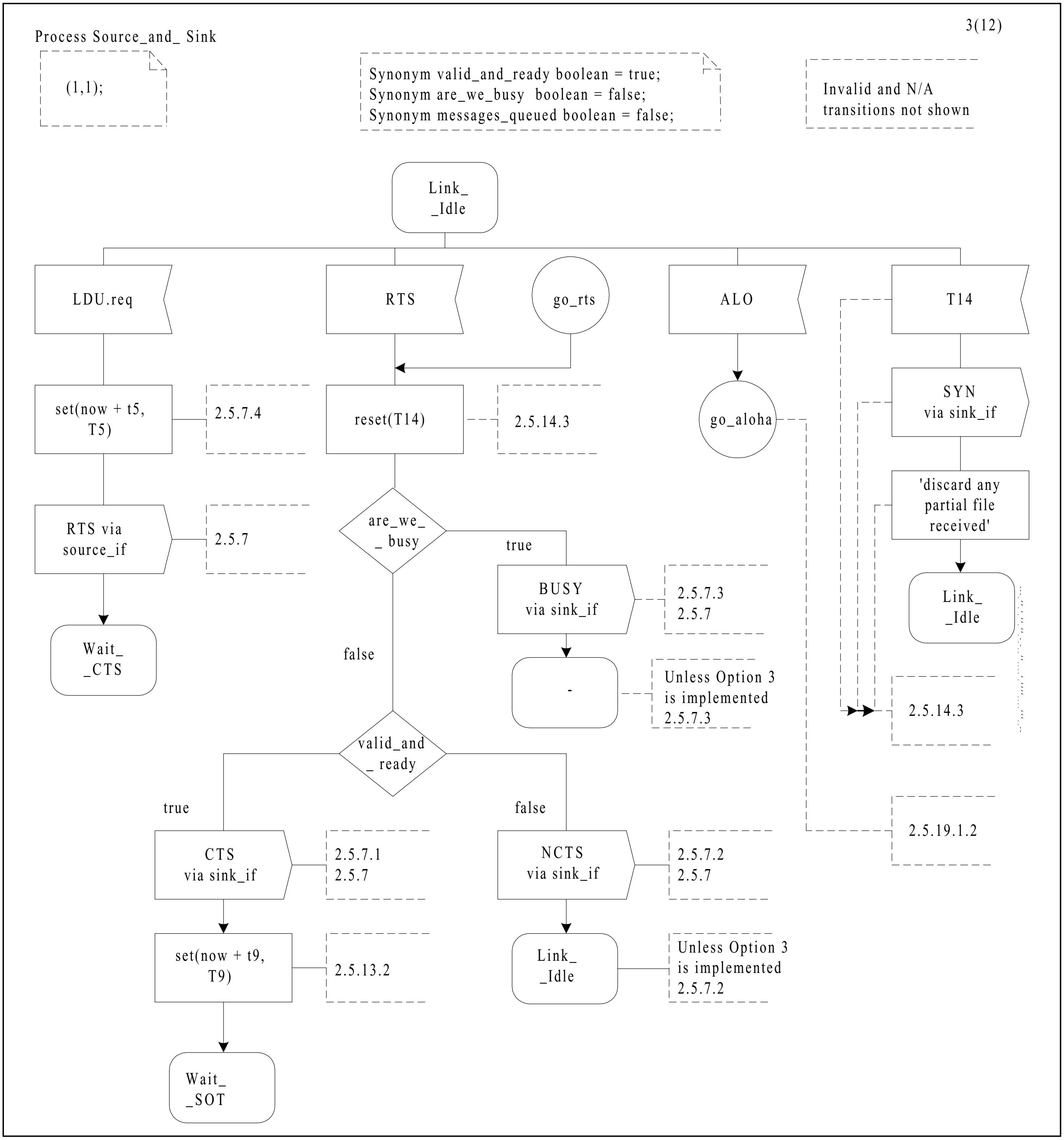

```
Dc:
n1count,
n2count,
n3count
n4count,
n5count
n6count integer;

Timer
T2,T4,T5,T9,T12,T14,T16;
```

```
/* This Source_and_Sink state
machine is replicated for each
SAL pair. This state
machine asumes that SAL
assignments have occurred; i.e.
- the state machine
only makes use of the pre-
determined SAL pair.*/
```

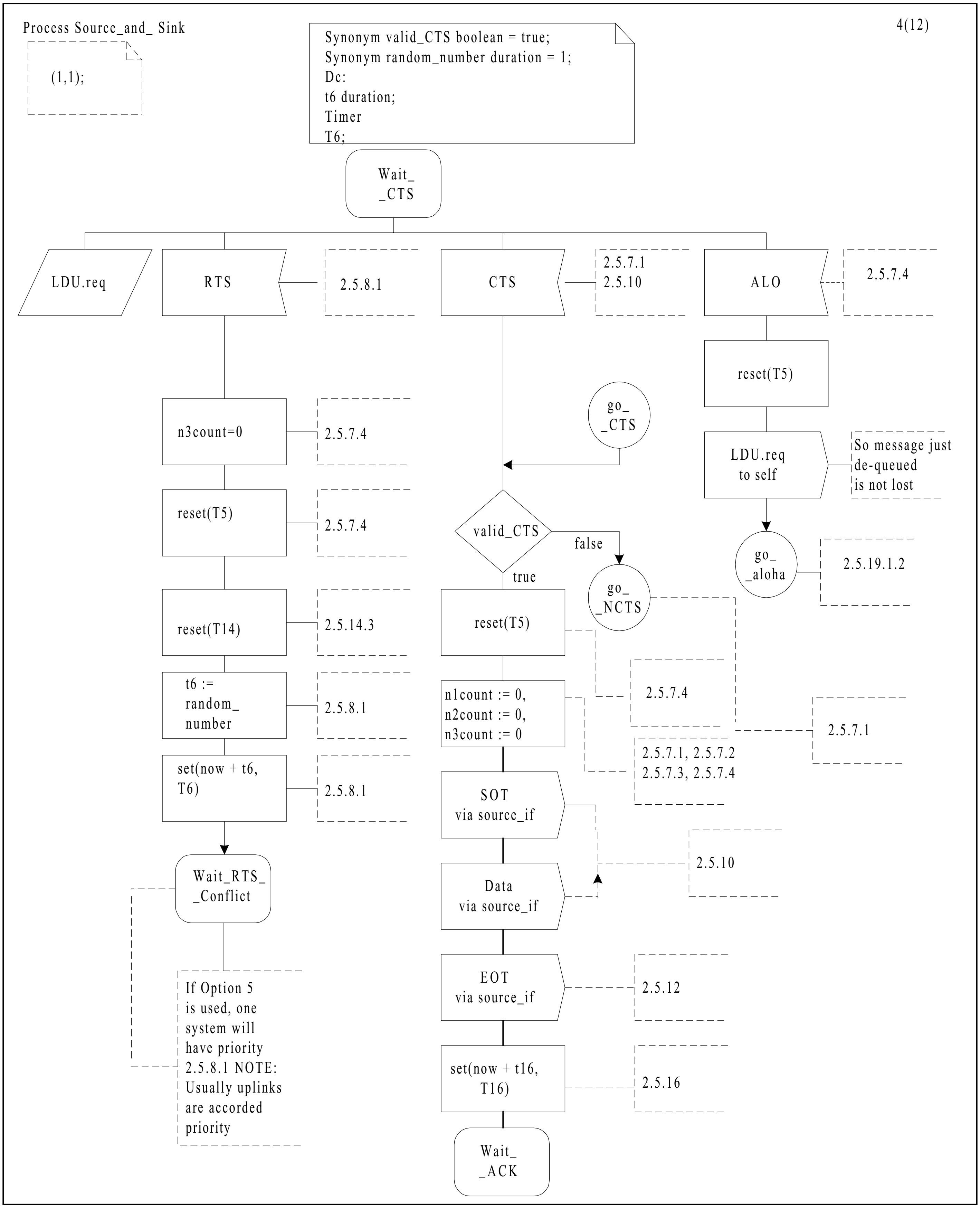
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



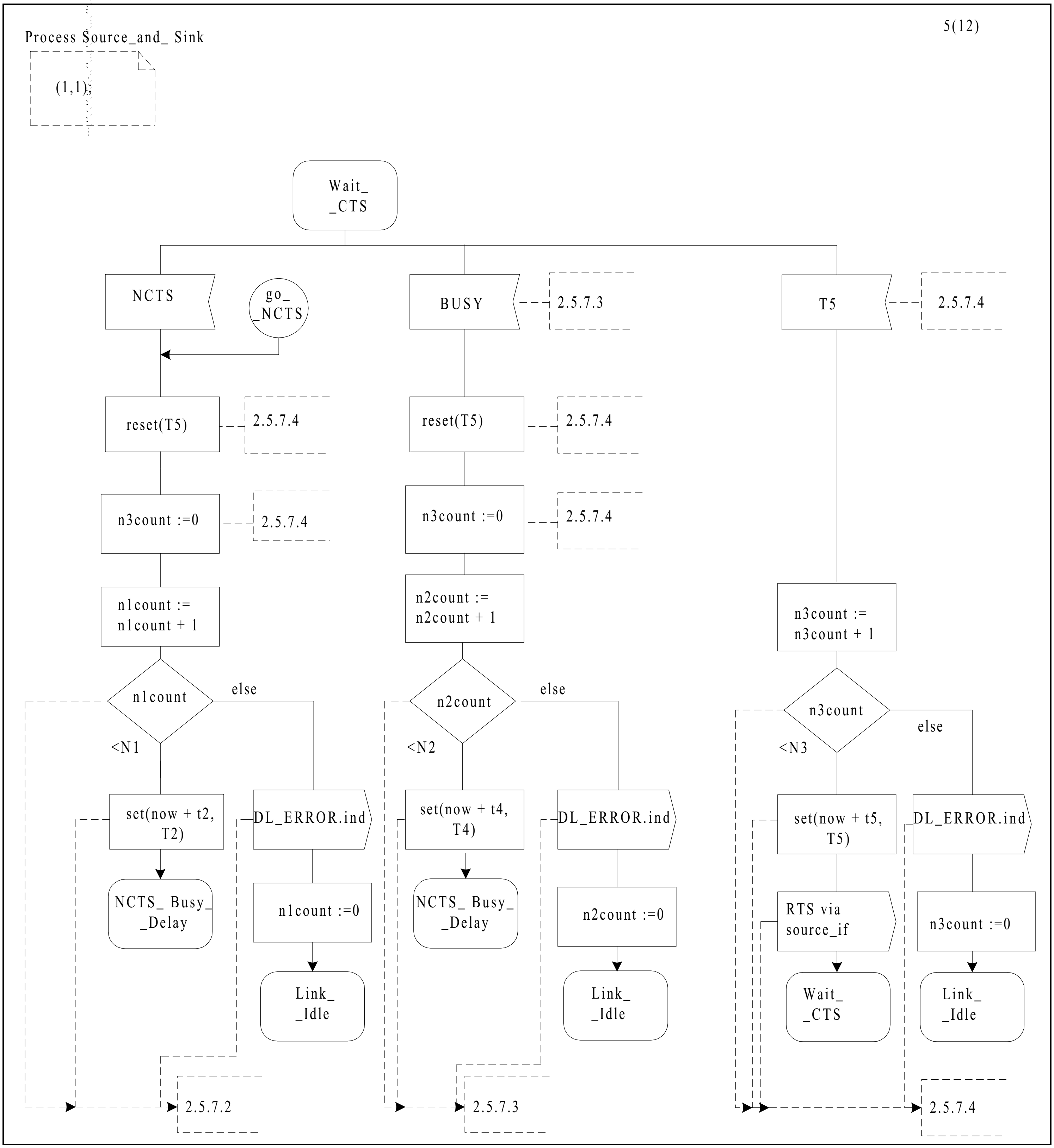
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



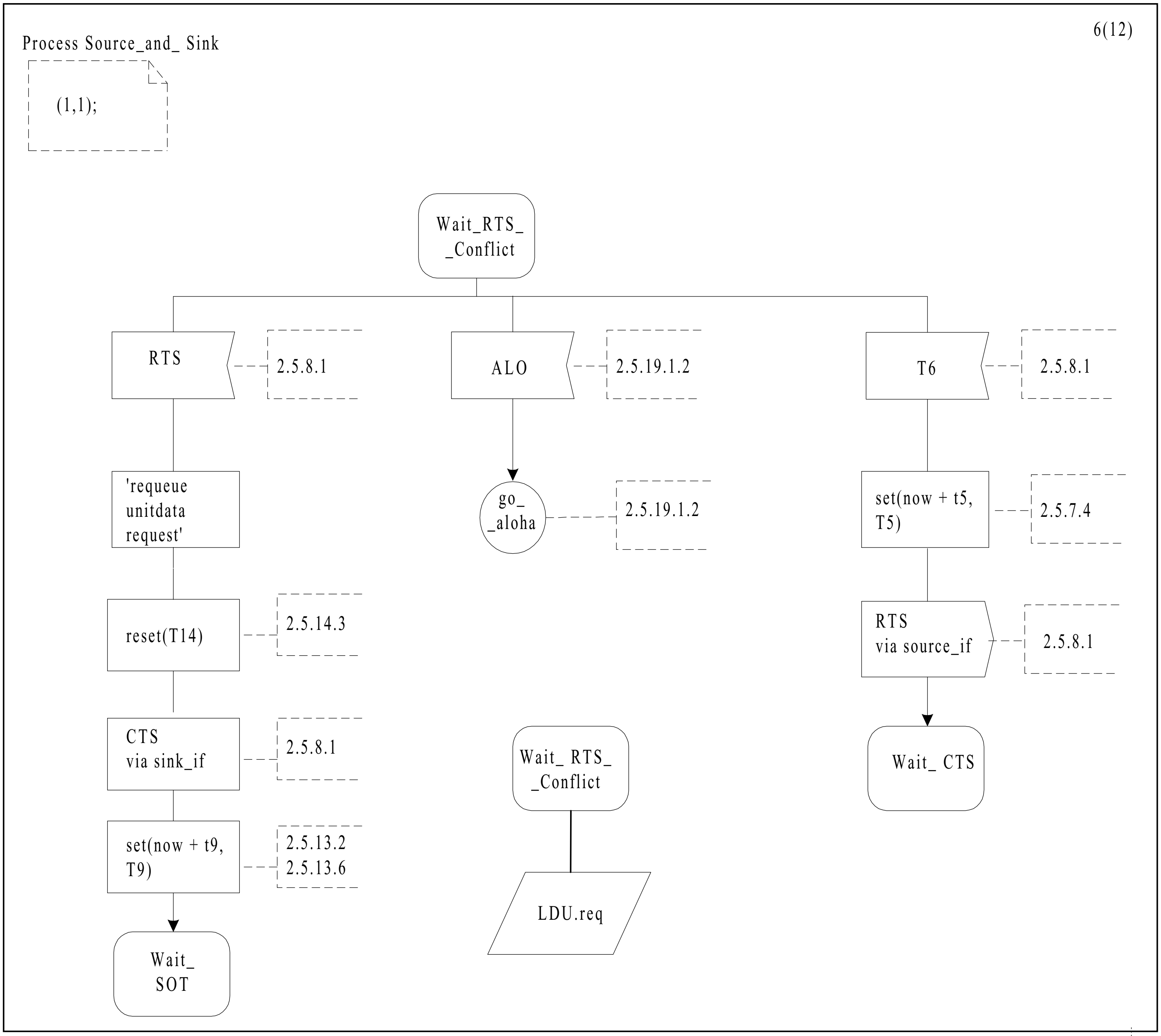
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



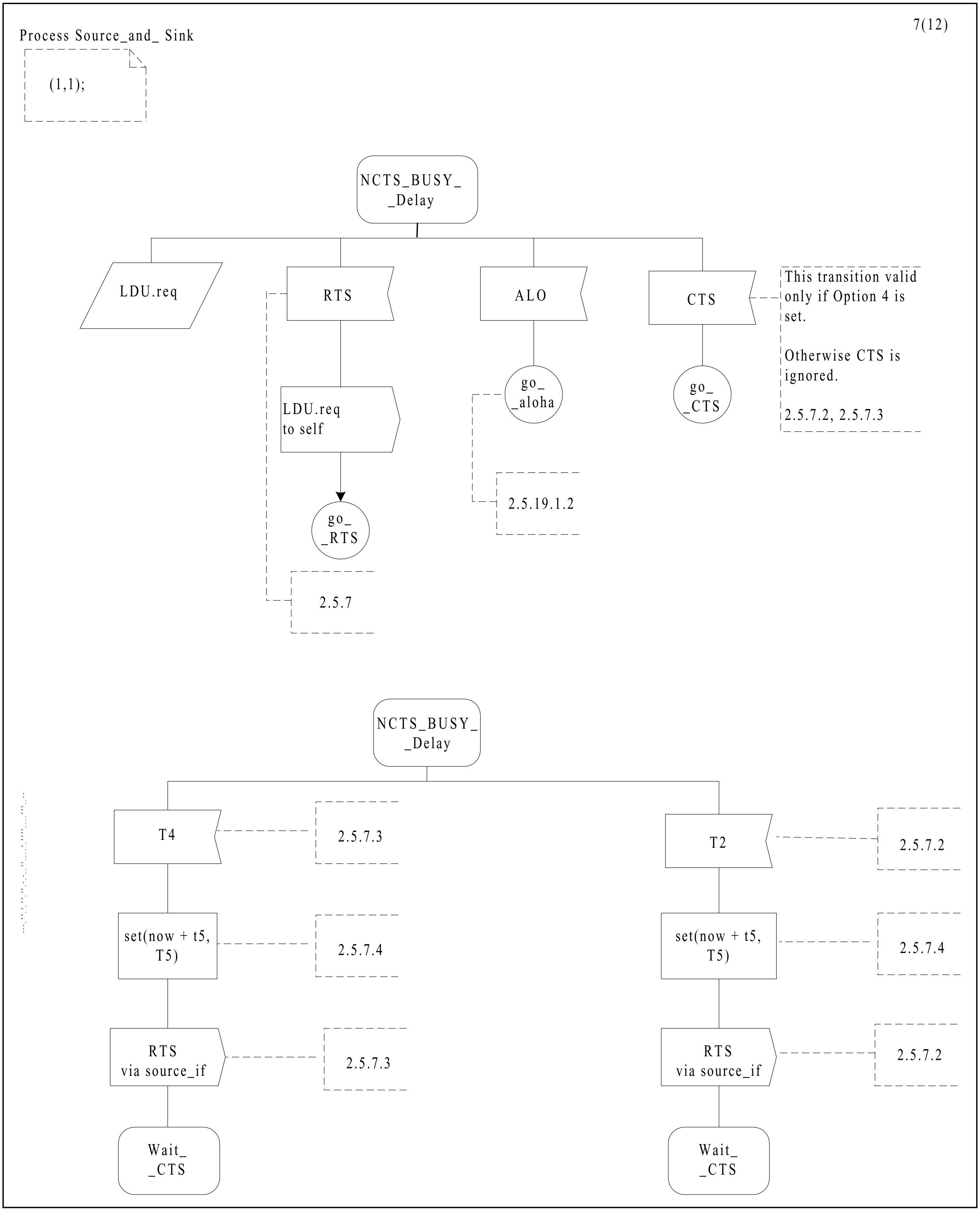
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



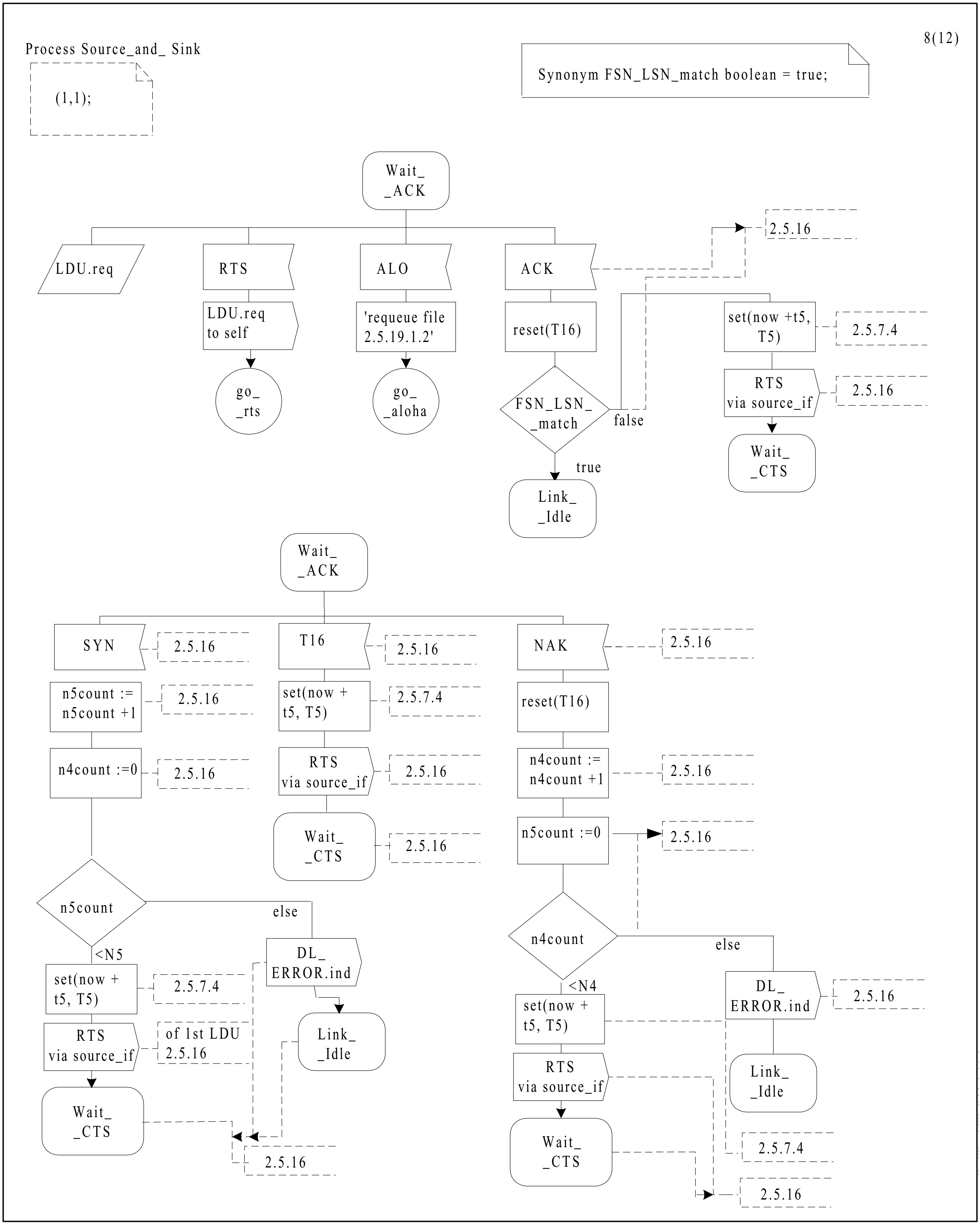
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



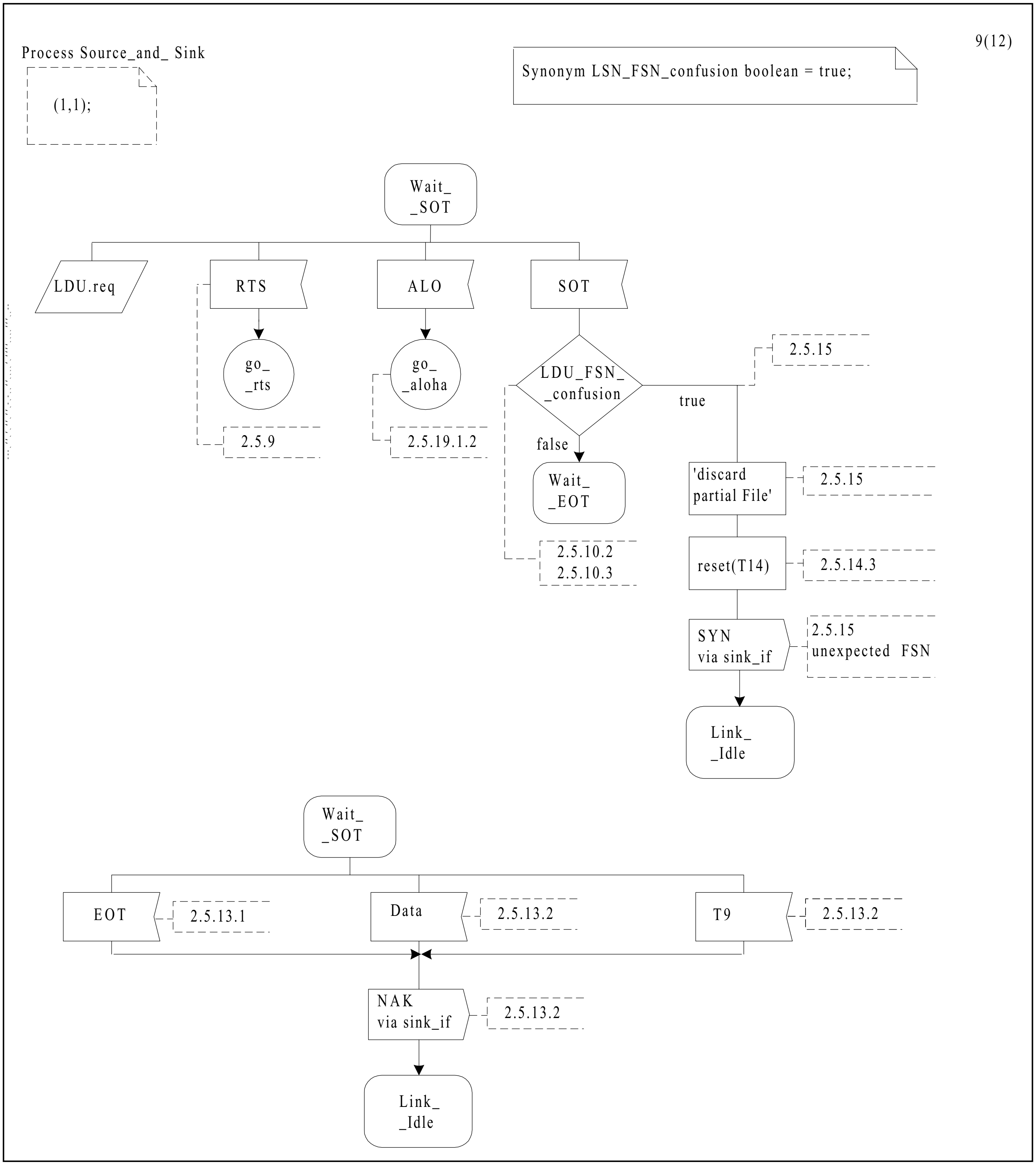
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



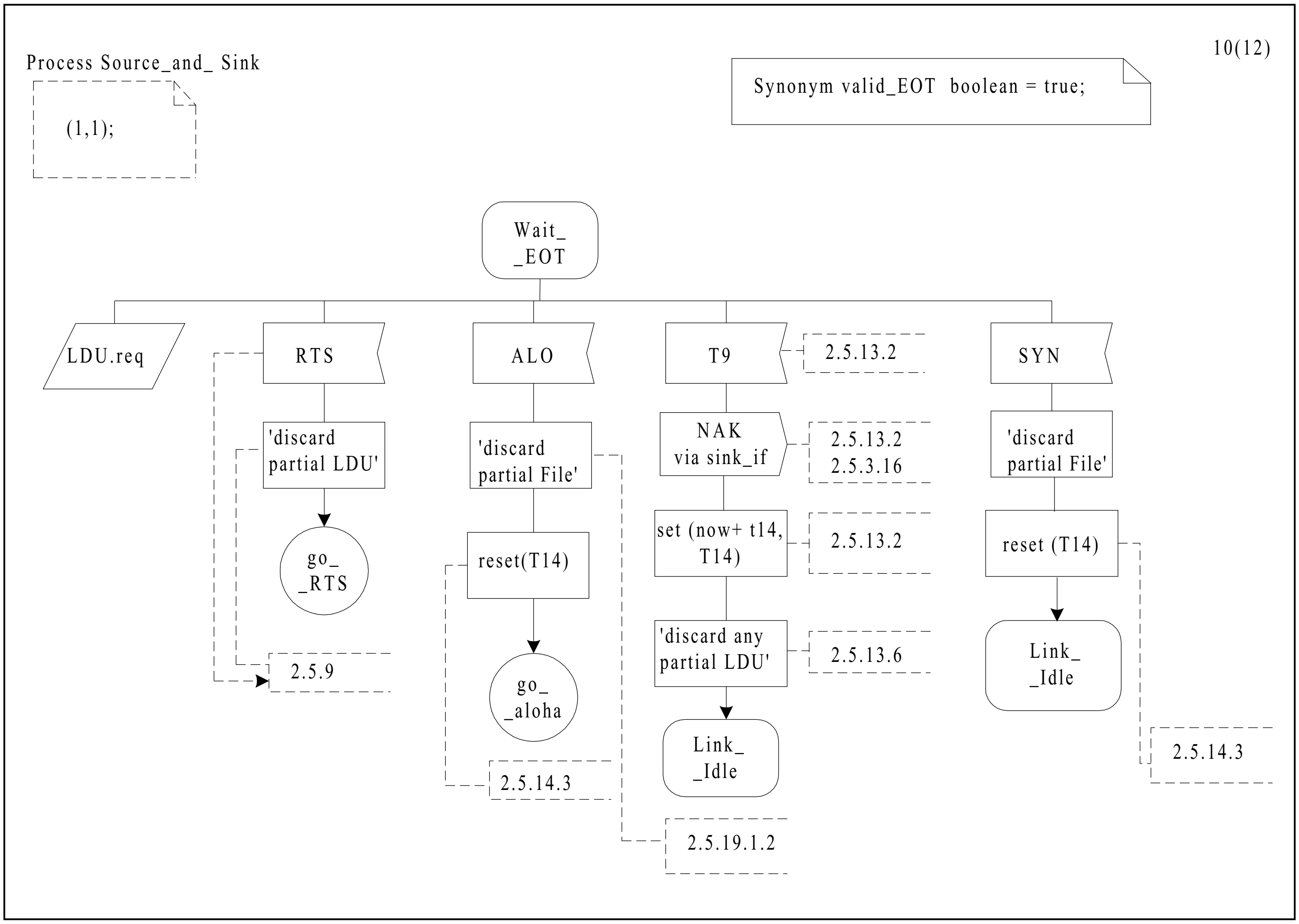
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



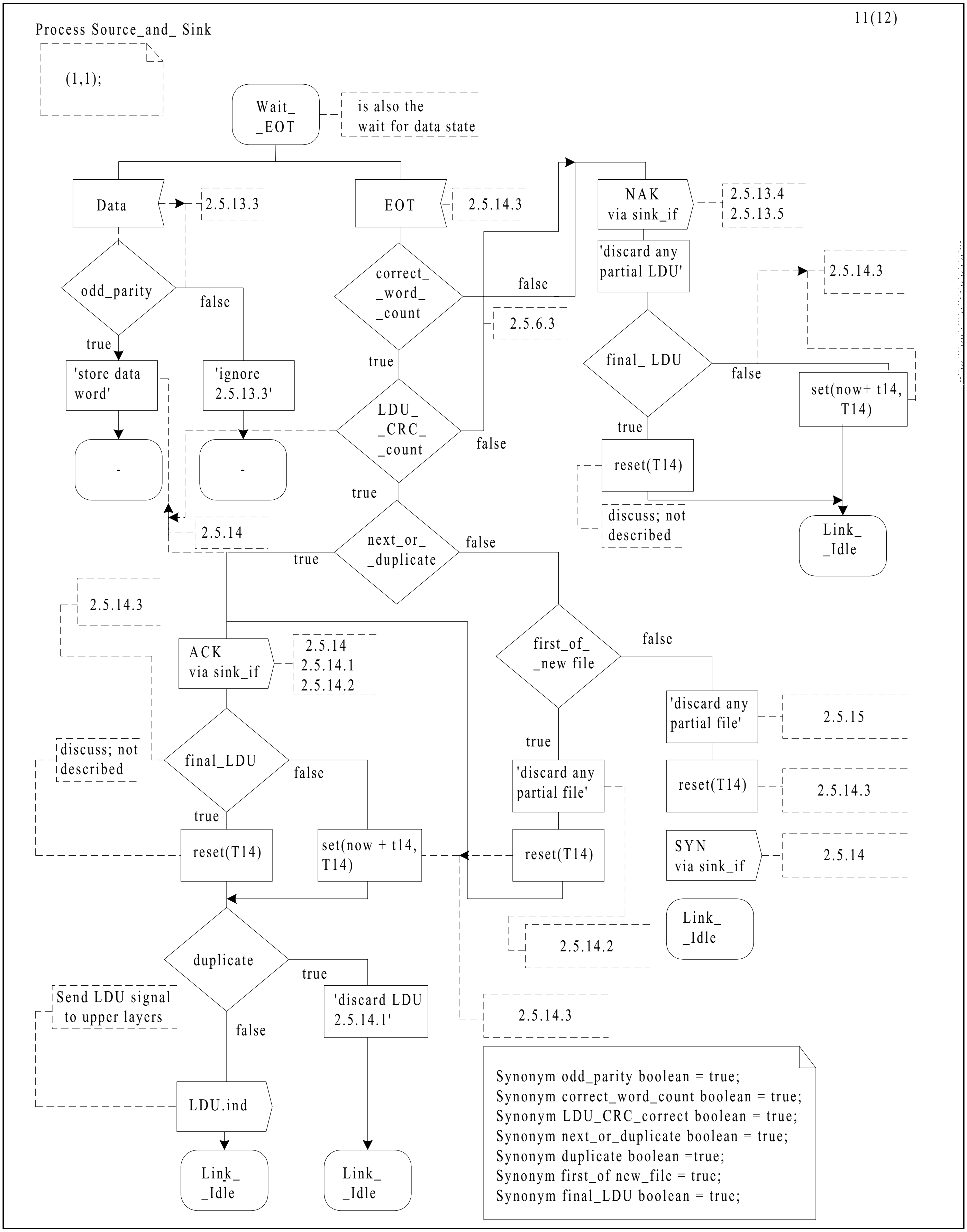
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



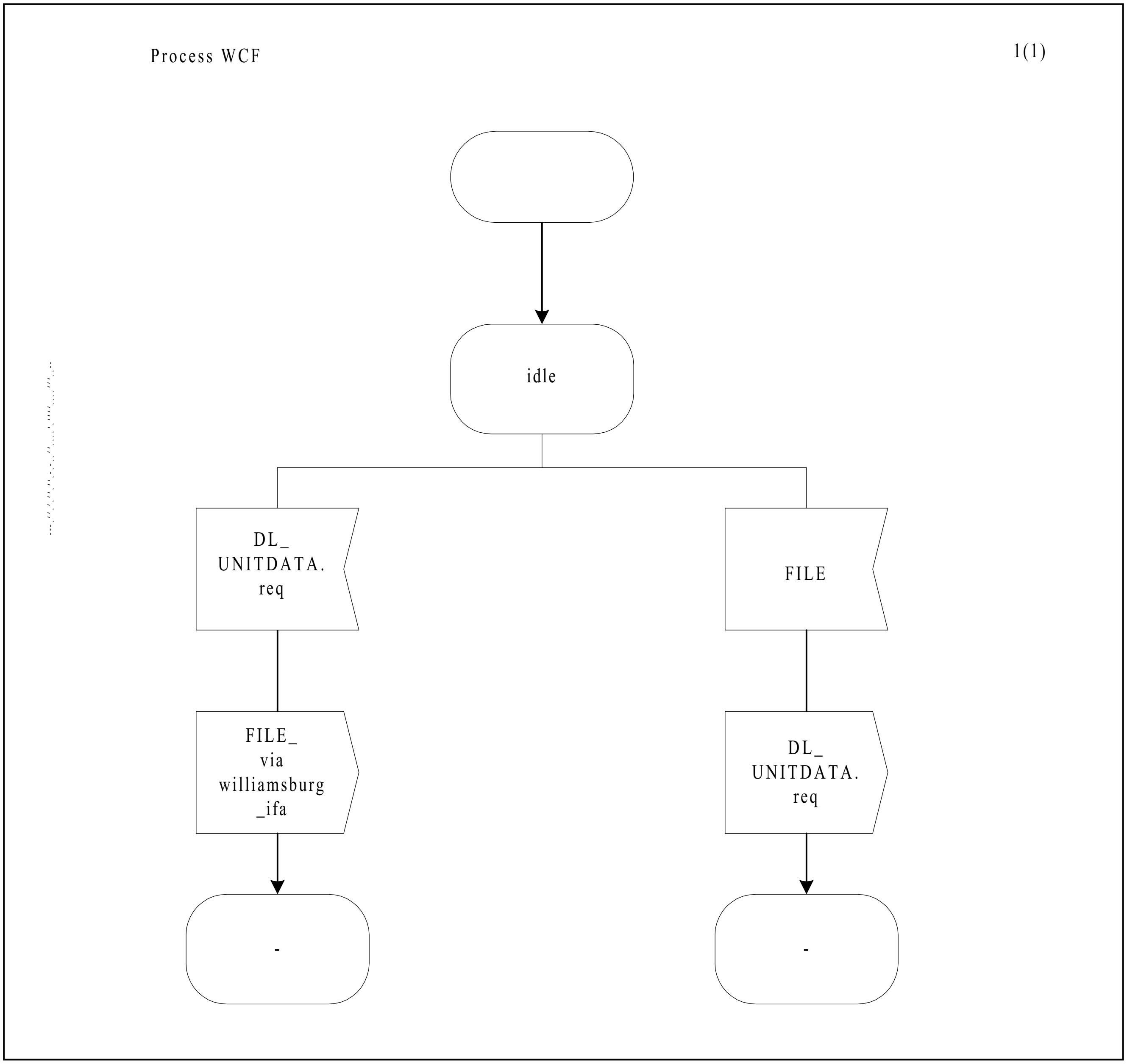
APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



APPENDIX I
SDL DIAGRAMS OF THE VERSION 1 WILLIAMSBURG PROTOCOL



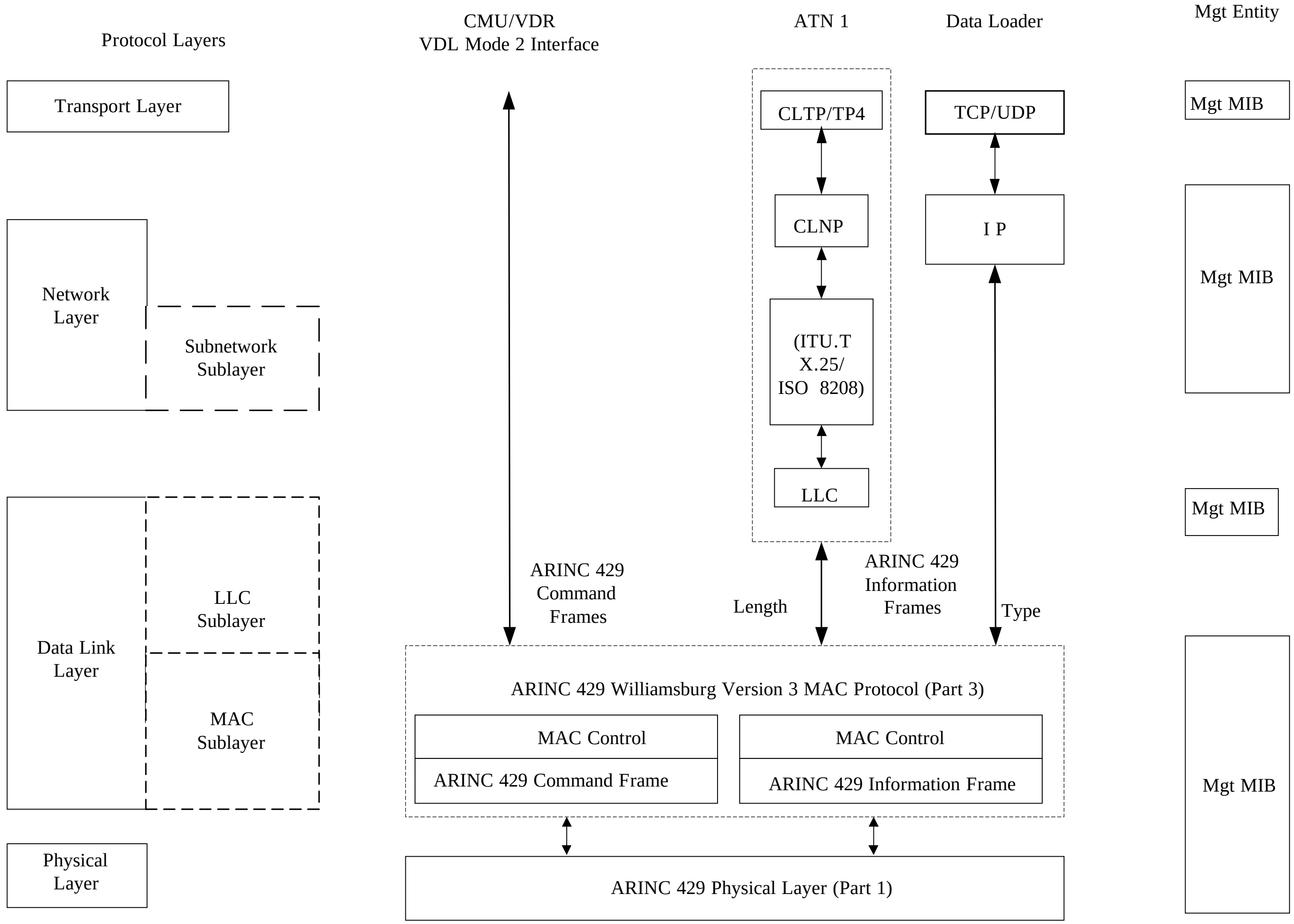


Figure J1-1 ARINC 429 Williamsburg Version 3 Protocol Layer Diagram

APPENDIX J
PROTOCOL STRUCTURE

Physical Layer - The physical layer is a high speed ARINC 429 data bus.

Link Layer – It is responsible for the exchange of data between points (nodes) connected to one network. One network can be constructed either by a bus (A429, A629) or by a set of buses and point-to-point links that can be interconnected by bridges (A646).

MAC Sublayer – The scope of this document is the "lower" part of the Link Layer. The ARINC 429 (Williamsburg) Version 3 protocol specifies the functionality of the MAC Service Provider that provides two types of MAC Frames, a bridgeable Information frame based upon the frame structure specified in IEEE 802.3 (Ethernet) and a non-bridgeable Command frame for point-to-point transfers of data of any kind, e.g. link management information.

LLC Sublayer – One potential MAC Service Client is the Logical Link Control Sublayer, which may be used to provide a logical interface between LLC entities. All IEEE 802 specifications share the notion of this "upper" portion of the Link Layer, which enables it to provide a common set of services. It may be used to provide a logical interface between peer entities. Three classes of service have been defined which can be connectionless-unacknowledged, connectionless-acknowledged or connection-oriented.

Network Layer – Other potential MAC Service Clients are either Internet (summarized by TCP/IP), specific (e.g. VDL Mode 2) or further standardized or non-standardized protocols. The Network layer provides a home for specifications of protocols that support the communication across network boundaries. In the example shown, these are, e.g., IP or CLNP.

Subnetwork Sublayer – In the scope of this document this sublayer is defined as an X.25 subset (profile). It may be required to provide services to the Network Layer, and interfaces with the LLC Sublayer. In the ISO network definition, these are the Subnetwork Access Protocols (SNAP) and provide services to the Subnetwork Independent Convergence Protocol (SNICP) and the Subnetwork Dependent Convergence Protocol (SNDCP). These services include flow control, c-17
error recovery and segmentation/reassembly.

It should be noted, however, that there is another (totally separate) SNAP definition by IEEE that resides on top of LLC Class 1 (connectionless-unacknowledged service) and acts as a Link layer client multiplexer similar to the 802.3 TYPE interpretation.

Transport Layer – This layer provides services for the exchange of information between communication applications, such as the Trivial File Transfer Protocol (TFTP) in the Internet suite of protocols which in this example is used by the Data Loader application or Simple Network Management Protocol (SNMP) for network management.

Typical Applications
This section focuses on the scope of this document, the various characteristics of the ARINC 429 Version 3 specification.

A429 Information Frame – As specified in section 3.3.1 this frame provides for a bridgeable frame format that is used by various applications. In this example, Data Loader (via TCP/IP stack) or ATN 1 (via OSI stack) could take advantage of this format.

A429 Command Frame – As specified in section 3.3.2 this frame provides for a non-bridgeable point-to-point frame format that may be used by the CMU/VDR VDL Mode 2 Interface.

ÍEEE 802.3 – IEEE used to assume that LLC resides above any of the 802.n (with n bigger than 2) specifications. IEEE 802.3 has always been in "competition" with the Ethernet specification, developed by Xerox, Digital and Intel. However, Xerox has shifted authority about the Ethernet specification to IEEE, which incorporated it into the current (1998) edition of the 802.3 specification. The main difference between the two specifications was the interpretation of a field that is now called "LENGTH/TYPE".

802.3-LENGTH – The "LENGTH" interpretation assumes that LLC is the MAC Service Client. If the value in this field is less than or equal to 1500 decimal, a "LENGTH" interpretation is specified.

APPENDIX J
PROTOCOL STRUCTURE

802.3-TYPE – The "TYPE" interpretation assumes that a different protocol is the MAC Service Client. If the value in this field is bigger than or equal to 1536 decimal, the value specifies the respective protocol. Assignments are documented in an Internet Request for Comment (RFC).

MAC Control – This functionality has been introduced recently and currently provides for a PAUSE function only. It enables temporary suppression of any data transmission when sent to the peer MAC entity (reverse MAC Control commands can still be returned, though). Both types of frames, Information and Command, support this function in different ways.

c-17

Network Management – Each layer of the communication stack is required to maintain a Management Information Base (MIB) which consists of parameters and behavioral characteristics of that layer and may be retrieved by the Network Management entity. The contents of the respective MIB are specified in the related protocol layer specification whereas the MIB structure as well as the Network Management Protocol and functionality is specified in a separate specification.

APPENDIX K
GLOSSARY & ACRONYMS

ACK	Acknowledge
ADS	Automatic Dependent Surveillance
AIDS	Airborne Integrated Data System
ALR	ALOHA Response
BOP	Bit-Oriented Protocol
BSAL	Bridge System Address Label
CRC	Cyclic Redundancy Check
CT	Command Type
CTS	Clear to Send
DITS	Mark 33 Digital Information Transfer System
EOF	End of Field
EOT	End of Transmission
FDU	Frame Data Unit
GFI	General Format Identifier
I/C	Information/Command
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ISO	International Standard Organization
LAN	Local Area Network
LDU	Link Data Units
LLC	Logical Link Control
LOOP	Loop Test Response
LRU	Line Replaceable Units
LSB	Least Significant Bit
MAC	Media Access Control
MSAL	Multicast System Address Label
MSB	Most Significant Bit
MU	Management Unit
NCTS	Not Clear to Send
OSI	Open Systems Interconnect
RTS	Request to Send
SAI	Systems Architecture and Interfaces Subcommittee
SAL	System Address Labels
SOF	Start of Field
SOT	Start of Transmission
TEST	Loop Test Pattern Word
VDL	VHF Data Link

NOTE: Due to the large number of changes Created by this Supplement, it is NOT available separately to update 429-11.

AERONAUTICAL RADIO, INC.
2551 Riva Road
Annapolis, Maryland 21401 – 7645 USA

SUPPLEMENT 12

TO

ARINC SPECIFICATION 429©

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

Published: July 1, 1990

Prepared by the Airlines Electronic Engineering Committee

Adopted by the Airlines Electronic Engineering Committee: October 25, 1989

A. PURPOSE OF THIS SUPPLEMENT

This Supplement introduces the Williamsburg bit-oriented file data transfer protocol which supports the transfer of binary and character data. The previous AIM and character-oriented file data transfer protocol sections are moved to Appendix 6. The Sign Status Matrix (SSM) information is revised and reorganized. In addition, this Supplement introduces new label assignments and equipment identification codes.

B. ORGANIZATION OF THIS SUPPLEMENT

The first part of this document, printed on goldenrod paper contains descriptions of changes introduced into this Specification by this Supplement and where appropriate extracts from the original text for comparison purposes. The second part consists of replacement white pages for the Specification, modified to reflect these changes. The modified and added material on each replacement page is identified with "c-12" symbols in the margins. Existing copies of ARINC Specification 429 may be updated by simply inserting the replacement white pages they replace. The goldenrod pages are inserted inside the rear cover of the Specification.

Copies of the Specification bearing the number 429-12 already contain this Supplement and thus do not require revision by the reader.

C. CHANGES TO ARINC SPECIFICATION 429 INTRODUCED BY THIS SUPPLEMENT

This section presents a complete tabulation of the changes and additions to the Specification introduced by this Supplement. Each change or addition is entitled by the section number and the title currently employed in the Specification or by the section name and title that will be employed when the Supplement is eventually incorporated. In each case there is included a brief description of the addition or change and, for other than very minor revision, any text originally contained in the Specification is reproduced for reference.

2.1.3 Information Identifier

This section contains editorial corrections to comply with changes introduced in Supplement 11.

2.1.5 Sign/Status Matrix

This section was revised and reorganized. The changes include moving the AIM and file transfer SSM definitions to Appendix 6, adding failure reporting to the discrete word truth table (Section 2.1.5.3) and moving the description of status priorities to Section 2.1.5.

2.3.1 Digital Language

The contents of Sections 2.3.1.4 through 2.3.1.5.7 were moved to Appendix 6. The AIM Data and File Data transfer section headings were retained for reference purposes. Section 2.3.1.5, File Data Transfer, provides the reason for moving the original file transfer protocol and introduces the Williamsburg protocol.

2.5 Bit-Oriented Communications Protocol

This new section was added to describe a bit-oriented data transfer protocol. The new protocol was developed to accommodate the interface of the ACARS Management Unit (MU) and the Satellite Data Unit (SDU).

3.2 AIM Information Transfer

The information previously contained in this section is no longer applicable to ARINC Specification 429. For reference purposes, the section header is retained and the original contents of this section are located in Appendix 6.

ATTACHMENT 1 – LABEL CODES

The following labels have been given new assignments:

002 115	013 0B8	016 0B8	046 10A	046 10B	047 10A
047 10B	107 0BB	110 0BB	112 0BB	114 0BB	114 10A
114 10B	127 10A	127 10B	130 035	130 10A	130 10B
131 035	132 035	133 10A	133 10B	134 10A	134 10B
137 10A	137 10B	155 10A	155 10B	156 10A	156 10B
157 10A	157 10B	160 10A	160 10B	161 10A	161 10B
201 115	203 035	203 10A	203 10B	205 10A	205 10B
211 10A	211 10B	220 116	221 116	222 115	222 116
223 116	224 116	226 035	230 116	234 039	234 040
235 039	235 040	236 039	236 040	237 039	237 040
244 10A	244 10B	256 114	257 114	260 10A	260 10B
260 114	261 10A	261 10B	261 114	262 10A	262 10B
262 114	263 10A	263 10B	263 114	264 10A	264 10B
264 114	265 004	265 038	265 10A	265 10B	265 114
267 10A	267 10B	270 10A	270 10B	270 114	270 115
271 10A	271 10B	271 114	272 002	272 10A	272 10B
272 114	273 10A	273 10B	273 114	274 10A	274 10B
274 114	275 10A	275 10B	275 114	276 114	277 018
300 10A	300 10B	300 TBD	301 10A	301 10B	302 10A
302 10B	303 10A	303 10B	304 10A	304 10B	305 10A
305 10B	306 10D	310 114	311 114	312 114	313 114
316 10A	316 10B	320 035	321 10A	321 10B	322 10A
322 10B	323 10A	323 10B	324 10A	324 10B	325 10A
325 10B	326 10A	326 10B	327 10A	327 10B	330 10A
330 10B	331 10A	331 10B	335 10A	335 10B	336 002
336 10A	336 10B	337 002	337 002	337 10A	337 10B
341 10A	341 10B	342 10A	342 10B	343 10A	343 10B
344 10A	344 10B	345 10A	345 10B	346 10A	346 10B
347 10A	347 10B	350 10A	350 10B	350 114	350 115
351 10A	351 10B	351 114	352 10A	352 10B	352 114
353 10A	353 10B	353 114	354 10A	354 10B	357 035
360 10A	360 10B	360 TBD	361 10A	361 10B	362 10B
362 10B	362 115	363 10A	363 10B	365 TBD	372 10A
372 10B	373 10A	373 10B	374 10A	374 10B	374 TBD
375 10A	375 10B	375 TBD			

Revised label 130 035 from "Traffic Advisory Range" to "Intruder Range".

Revised label 131 035 from "Traffic Advisory Altitude" to "Intruder Altitude".

Revised label 132 035 from "Traffic Advisory Bearing" to "Intruder Bearing".

Removed label 130 030 Traffic Advisory Range.

Removed label 131 030 Traffic Advisory Altitude.

Removed label 132 030 Traffic Advisory Bearing Estimate.

Removed label 270 030 Transponder Discrete.

Removed label 347 030 Sector Control.

Removed 347 035 Antenna Control.

ATTACHMENT 1 – EQUIPMENT CODES

The following codes have been given new assignments:

113, 114, 115, 116, 117, 118, 119, 11A, 123, 124, 125, 126, 127, 128, 129, 15A, 15B, 15C, 15D, 15E, 16A, 16B, 16C, 16D, 16E, 17A, 17B, 17C, 18A, 18B, 18C, 18D, 18E, 18F.

ATTACHMENT 2 – DATA STANDARDS

Tables 1, 2 updated to reflect changes to Attachment 1.

Binary Data notes 6, 7 and 8 added.

Discrete Data Standards entered for new labels:

272 002	271 018	272 018	273 018	275 018	276 018
277 018	274 018	270 035	271 035	273 035	274 035
275 035	013 0B8	016 0B8	161 10A	161 10B	350 114
351 114	352 114	353 114	270 115	350 115	

ATTACHMENT 6 – GENERAL WORD FORMATS AND ENCODING EXAMPLES

Add format for TCAS Intruder Range label 130.

Add format for TCAS Intruder Altitude label 131.

Add format for TCAS Intruder Bearing label 132.

Add format for Transponder Altitude/TCAS Own A/C Altitude label 203.

Removed 730 ASAS Sector Control Word example.

Removed 730 TCAS Traffic Advisory Range Word example.

Removed 730 TCAS Traffic Advisory Bearing Estimate Word example.

ATTACHMENT 9B – GENERAL AVIATION WORD EXAMPLES

Add new Company Name Identifier.

ATTACHMENT 10 – VARIABLES OF BIT-ORIENTED PROTOCOL

Add new Attachment.

ATTTACHMENT 11 – BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

Add new Attachment.

ATTACHMENT 11A - DESTINATION CODES

Add new Attachment.

ATTACHMENT 11B - STATUS CODES

Add new Attachment.

ATTACHMENT 11C - ALOHA/ALOHA RESPONSE PROTOCOL WORDS

Add new Attachment.

ATTACHMENT 12 - FILE TRANSFER EXAMPLE

Add new Attachment.

ATTACHMENT 12A - FIELD MAPPING EXAMPLE

Add new Attachment.

ATTACHMENT 13 - PROTOCOL DETERMINATION PROCEDURE DIAGRAMS

Add new Attachment.

ATTACHMENT 14 – SYSTEM ADDRESS LABELS

Add new Attachment.

ATTACHMENT 15 - LINK LAYER CRD DATA EXAMPLE

Add new Attachment.

APPENDIX 6 - FORMER MAINTENANCE, AIM AND FILE TRANSFER TECHNIQUES

Add new Appendix.

APPENDIX 7 – MATHEMATICAL EXAMPLE OF CRC ENCODING/DECODING

Add new Appendix.

AERONAUTICAL RADIO, INC.
2551 Riva Road
Annapolis, Maryland 21401 – 7645 USA

SUPPLEMENT 13

TO

ARINC SPECIFICATION 429©

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

Published: December 30, 1991

Prepared by the Airlines Electronic Engineering Committee

Adopted by the Airlines Electronic Engineering Committee: October 8, 1991

A. PURPOSE OF THIS SUPPLEMENT

This Supplement introduces changes made to the Williamsburg protocol as a result of its initial implementation. This protocol supports the transfer of binary and character data. In addition, this Supplement introduces new label assignments and equipment identification codes.

B. ORGANIZATION OF THIS SUPPLEMENT

The first part of this document, printed on goldenrod paper contains descriptions of changes introduced into this Specification by this Supplement. The second part consists of replacement white pages for the Specification, modified to reflect the changes. The modified and added material on each page is identified by a c-13 in the margins. Existing copies of ARINC Specification 429 may be updated by simply inserting the replacement white pages where necessary and destroying the pages they replace. The goldenrod pages are inserted inside the rear cover of the Specification.

C. CHANGES TO ARINC SPECIFICATION 429 INTRODUCED BY THIS SUPPLEMENT

This section presents a complete tabulation of the changes and additions to the Specification introduced by this Supplement. Each change or addition is defined by the section number and the title currently employed in the Specification or by the section name and title that will be employed when the Supplement is eventually incorporated. In each case a brief description of the change or addition is included.

AEEC STAFF NOTE: THESE CHANGES APPLY TO ARINC 429, PART 3 ONLY.

2.3.1.5 File Data Transfer

An editorial change, correction to section numbering.

2.3.1.5.1 Bit-Oriented Protocol Determination

New Section added to describe ALO/ALR protocol process to be used when a bilingual Link Layer protocol system needs to determine necessary bit-oriented interfaces.

2.5 Bit-Oriented communications Protocol

Included term "Williamsburg" parenthetically since this terminology well-known in industry. Added commentary to explain non-negotiation or parameters in this protocol.

D. Corrected Network Layer definition.

2.5.2 Link Data Unit (LDU) Size and Word Count

Added second paragraph to text, since it is a requirement, and removed second paragraph from commentary.

2.5.4 Bit Rate and Word Timing

Corrected the commentary to change the more ambiguous term "message" to LDU.

2.5.5.3 Destination Code

An editorial change was made.

2.5.6 Response to RTS

The last sentence in the second paragraph was reworded and moved to a more appropriate section, 2.5.6.2.

2.5.6.1 Clear to Send (CTS)

In the second to last sentence, the word "valid" was added to clarify the Not clear to send condition. The last sentence was added to clarify the resetting of RTS counters.

2.5.6.2 Not Clear to Send (NCTS)

The first paragraph was updated to include the information deleted from Section 2.5.6 and to clarify the validity requirements. The second paragraph was updated to describe that and NCTS counter would be reset upon a valid CTS response. The last sentence in the third paragraph was deleted and it's content expanded in the following commentary of that section.

2.5.6.3 Destination Busy

The second paragraph of this section was updated to indicate that a BUSY counter should be reset with a valid CTS response to RTS.

2.5.7 No Response to RTS

The first paragraph of this section was updated to describe proper response to RTS.

2.5.9 Unexpected RTS

This section was updated to include editorial changes and a description of the correct responses to RTS. The last sentence was deleted as redundant to Section in 2.5.13.1 and in conflict with other possible responses.

2.5.11 Data

The fourth paragraph of this section was updated to describe the proper ending of an LDU transmission, and to include the optional NAK response for receipt of an incomplete octet.

2.5.11.3 Character Data Words

In the last paragraph, the "note" designator was removed and the text clarified for the transfer of characters with a parity bit.

2.5.13 Negative Acknowledgement (NAK)

This section was updated to clarify conditions for sending the NAK word.

2.5.13.1 Missing SOT word

Text was corrected to refer to "reception" instead of "transmission" of a valid SOT word. Also, incorrect text referring to the NAK response timing was deleted.

2.5.13.2 LDU Sequence Number Error

The original text was omitted. Sections 2.5.13.1 – 2.5.13.7 were renumbered.

2.5.13.3 Parity Errors

A commentary section was added to describe the procedures for receiving words with bad parity.

2.5.13.4 Word Count Errors

This section was updated to clarify the NAK response time for word count errors.

2.5.13.5 CRC Errors

This section was updated to clarify the NAK response time for CRC errors.

2.5.13.6 Time Out Errors

This section was renumbered.

2.5.13.7 Restart Initialization

This section was omitted due to potential conflicts with the ALO/ALR procedures.

2.5.14 LDU Transfer Acknowledgement (ACK)

Text was revised to include LDU conditions for sink acknowledgement transmission.

2.5.14.1 Duplicate LDU

This section was added to describe duplicate LDU occurrences.

2.5.14.2 Auto-Synchronized Files

This section was added to describe the method of handling auto-synchronized files.

2.5.15 SYN Word

New text was added to describe SYN response times for non-consecutive LDU Sequence numbers. The last paragraph was incorrect and deleted.

2.5.16 Response to ACK/NAK/SYN

New text was added to describe actions when NAK and SYN are detected during a transmission.

2.5.19 ALO Response

A new section was added and updated to describe ALO responses.

ATTACHMENT 10 – VARIABLES OF BIT ORIENTED PROTOCOL

Tables 10-1 and 10-3 were updated to include events N_5, N^6, and time T_{12}. Options 0_7 and 0_{12} in Table 10-4 were changed to spares for consistency with corresponding text updates.

ATTACHMENT 11C – ALOHA/ALOHA RESPONSE PROTOCOL WORD DEFINITION

Table 11C-3 was added to clarify protocol version number assignments, and is referenced by "note 1". "Note 2" was added to describe the GFI field of the ALOHA word.

ATTACHMENT 12A – FIELD MAPPING EXAMPLE

B_k was changed to B_{24} in the data word map, "nibble" was changed to "semi-octet", and semi-octet arrow lengths were shortened to correspond to the proper four and eight-bit lengths.

APPENDIX 7 – MATHEMATICAL EXAMPLE OF CRC ENDODING/DECODING

Format (alignment) changes were made in the polynomial divisions, "(X)" was corrected to "Q(x)", and the transmission order for the LDU Mapping of the 24-bit example was deleted to avoid possible misinterpretation.

SUPPLEMENT 14

TO

ARINC SPECIFICATION 429©

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

Published: January 4, 1993

Prepared by the Airlines Electronic Engineering Committee

Adopted by the Airlines Electronic Engineering Committee: November 4, 1992

A. PURPOSE OF THIS SUPPLEMENT

This Supplement introduces changes made to increase the efficiency of data transfer across an ARINC 429 high speed bit-oriented link. This protocol supports the transfer of binary and character data.

B. ORGANIZATION OF THIS SUPPLEMENT

The first part of this document, printed on goldenrod paper, contains descriptions of changes introduced into this Specification by this Supplement. The second part consists of replacement white pages for the Specification, modified to reflect the changes. The modified and added material on each page is identified by a c-14 in the margins. Existing copies of ARINC 429 may be updated by simply inserting the replacement white pages where necessary and destroying the pages they replace. The goldenrod pages are inserted inside the rear cover of the Specification.

C. CHANGES TO ARINC SPECIFICATION 429 INTRODUCED BY THIS SUPPLEMENT

This section presents a complete tabulation of the changes and additions to the Specification introduced by this Supplement. Each change and addition is defined by the section number and the title currently employed in the Specification or by the section name and title that will be employed when the Supplement is eventually incorporated. In each case a brief description of the change or addition is included.

AEEC STAFF NOTE: THESE CHANGES APPLY TO ARINC 429, PART 3 ONLY.

2.3.1.5 File Data Transfer

An editorial change was needed to reference new section.

2.3.1.5.1 Bit-Oriented Protocol Determination

This section was expanded to include determination of different version numbers of the bit-oriented protocol, and was moved to Section 2.5.19.

2.5 Bit-Oriented Communication Protocol

An editorial change references a new section number.

2.5.4 Bit Rate and Word Timing

A maximum word gap of 64 bit-times, (averaged over the LDU transmission) was added to eliminate excessive delay in source transmission time.

Note: Sections 2.5.5 through 2.7 have been renumbered and reordered for consistency.

2.5.5 Word type

The basic definition of "word type" was corrected to include bits 31-29 in all bit-oriented words of an LDU.

2.5.6 Protocol Words

This section was added to specifically define the word type for protocol words.

2.5.6.1 Protocol Identifier

This section was added to clarify the definition of bits 28-25 for protocol words and to specify the relevant addition for error conditions.

2.5.6.2 Destination Code

This section was updated, and a commentary added, to clarify the role of the link layer protocol for upward compatibility with changing network functionality. The requirement for Destination code validation is not a link layer function.

2.5.6.3 Word Count

This section was renumbered.

2.5.7 Request to Send (RTS)

This section was previously titled "Response to TS", and has been renumbered. The title was changed for consistency, and an introductory paragraph added to clarify the basic RTS function.

2.5.7.1 Clear to Send (CTS)

This section was renumbered.

2.5.7.2 Not Clear to Send (NCTS)

This section was renumbered.

2.5.7.3 Destination Busy

This section was renumbered, and an introductory replacement paragraph inserted to clarify the "optional" BUSY response, which may be used when a system cannot accept a transmission by the source in a "timely manner". New commentary equates a "timely manner" to the shorter retry sequence of the NCTS series.

2.5.7.4 No Response to RTS

This section was renumbered, and the ALOHA word was included in the logic for error determination.

2.5.10 Start of Transmission (SOT)

Timer T_{13} was added as a requirement on the source to begin transmission of an LDU within a specified interval after receipt of the CTS word from the sink.

2.5.10.1 General Format Identifier (GFI)

This section was updated, and commentary added to clarify the role of the GFI in pre-OSI as well as OSI environments. Validation of the GFI code is required by a high level entity (network layer) in both environments to determine the format of the data words to follow. GFI validation is not necessarily a link layer function.

2.5.11 Data

All references to Character Data word formats were deleted.

2.5.11.3 Character Data Words

This section was deleted. The Character Data Word format was removed from Supplement 14, as the format is incompatible with those for Full and Partial Data word formats. Currently, both binary and character data are transmitted in octets defined by the other two data word formats. The special character data format is not required.

2.5.12.1 CRC Encoding

References to character data words were deleted. The text for equation: M9x) = x^{16}G(x) + R(x) was corrected by moving the "bar" from G(x) to R(x).

2.5.13 Negative Acknowledgement (NAK)

NAK word interpretation was changed to remove constraint on source for specific order of file sequencing (i.e. Allows source to restart file with new FSN if necessary).

2.5.14.1 Duplicate LDU

This first paragraph was rewritten to clarify.

2.5.14.3 Incomplete File Timer

This section was added to allow the sink to discard a partial file of multiple LDUs when the T_{14} timeout between LDU transmissions is exceeded. It ensures that a source device cannot "lock-up" a sink.

2.5.15 SYN Word

The LDU sequence anomalies which generate a SYN response by the sink were clarified.

2.5.16 Response to ACK/NAK/SYN

The T_{16} timer was introduced to replace T_{10} and T_8. Also, the action taken by the source upon receipt of a SYN word was updated, which relaxes requirements to maintain a specific File Sequence ordering by the source.

2.5.19 Protocol Initialization

2.5.19.1 Bit-Oriented Protocol Version,

2.5.19.2 ALOHA Response, and

2.5.19.3 Character-429 Determination

This section has been added to replace and expand on the definition of the process to determine the link layer protocol version supported by an interfacing system. These sections replace three sections from Supplement 13.

2.3.1.5.1 Bit-Oriented Protocol Determination

2.5.19 ALO Response, and

2.5.20 Bit Protocol Verification

2.6 Windowed Bit-Oriented Protocol

This is a completely new section which contains the system description of the new LLC2-like bit-oriented link layer protocol for 429. It is based on Section 2.5, "Bit-Oriented Communications Protocol", with expanded text as specified to allow for more efficient use of the 429 high (or low) speed data bus through "windowing". The definition includes provision for a Link Control Word prior to each LDU.

ATTACHMENT 1 – EQUIPMENT CODES

New Equipment Code Identifiers were added.

ATTACHEMENT 6 – WORD FORMATS AND ENCODING EXAMPLES

Example added for label 171.

ATTACHMENT 10 – VARIABLES OF BIT ORIENTED PROTOCOL

Table 10-1 was updated to include a standard value for N_7, the maximum number of LDUs in a window (see Section 2.6 "Windowed Bit-Oriented Protocol").

Table 10-3 deleted Option 6 (O_6) for NAK Send Time, and deleted Option 9 (O_9) for the Character Data Word, both of which are no longer used.

Table 10-4 was revised to include columns for low speed maximum and minimum values. These values were established for timers and as response time design goals for incoming transmissions. Timers T_{13} through T_{16} were added.

Table 10-5 was added to include a definition of high speed maximum and minimum values for timers and response time design goals. The format is the same as the revised Table 10-4. Timer T_{10} is not used in the high speed protocol.

Table 10-6 was added to include notes to Tables 10-1 through 10-5.

ATTACHMENT 11 – BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

Table 11-1A added "spares" for the deleted Character Data Formats and corrected "Protocol Data Word" to read "Protocol Word".

Table 11-4 updated definitions for bits 9 through 24 of the ALO and ALR words, and added the LCW (LDU Control Word) format definition.

Table 11-4A was added as a partial replacement for ATTACHMENT 11C and Table 11-4B was added to define the new window definitions for the Windowed Bit-Oriented protocol in Section 2.6.

Table 11-6A was revised, changing the former GFI bit pattern (0001) for ISO 8208 to "unassigned". The bit pattern (0100) for ISO 8473 was changed to a more

ATTACHMENT 11 – BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS (cont'd)

generic ISO 9577 definition. The bit pattern 1110 (hex"E") is now defined as "ACARS VHF Format".The "NOTES" in ATTACHMENT 11 have been renumbered to correspond to the new table definitions.

ATTACHMENT 11C – ALOHA/ALOHA RESPONSE PROTOCOL WORD DEFINITION

This Attachment has been deleted. This information has been moved to Tables 11-4, 11-4A, and 11-4B.

ATTACHMENT 13A – ALOHA VERSION DETERMINATION SEQUENCE

This Attachment was added to support the ALOHA version determination sequence called out in Section 2.5.19.1.1.

ATTACHMENT 14 – SYSTEM ADDERESS LABELS

New System Address Labels (SAL) were added.

ATTACHMENT 16 – SEQUENCE OF PROTOCOL AND DATA WORDS IN WINDOW TRANSFER

This Attachment was added to illustrate the window transfers for new Section 2.6.

ATTACHMENT 17 – FLOW DIAGRAM USED TO DETERMINE CHARACTER-ORIENTED VS BIT-ORIENTED PROTOCOL

This Attachment was added to illustrate the logic flow that determines whether a character-oriented or bit-oriented link layer protocol interface is to be used.

SUPPLEMENT 15

TO

ARINC SPECIFICATION 429©

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

PART 3

FILE DATA TRANSFER TECHNIQUES

Published: August 31, 1995

Prepared by the Airlines Electronic Engineering Committee

Adopted by the Airlines Electronic Engineering Committee: April 18, 1995

A. PURPOSE OF THIS SUPPLEMENT

This portion of Supplement 15 provides corrections and additions to the file transfer provisions of ARINC Specification 429. The reader should note that the organization of ARINC 429 has been described in Section B below.

Appendix C was added to assist designers in establishing connectivity between LRUs designed to different versions of Specification 429. Appendix D comprises the Specification and Description Language (SDL) diagrams that reflect the intent of the textual material. The SDL diagrams have not been fully proofed, and remain advisory in nature. Therefore, the text material has precedence over the SDL diagrams. When the SDL diagrams have been validated, they will be moved to an Attachment.

B. ORGANIZATION OF THIS SUPPLEMENT

The portion of this document, printed on goldenrod paper, contains descriptions of changes introduced into this Specification by this Supplement. In the text, printed on white paper, the modified and added material on each page is identified by a c-15 in the margins. In view of the document reorganization, existing copies of ARINC 429 cannot be updated.

This Supplement is the first in which ARINC Specification 429 is divided into three parts. This part, Part 3, contains the definition of the protocols used for file data transfer. Typically, file data transfer is non-periodic in nature.

The fundamental physical layer descriptions of the wire, voltage levels and coding of data are contained in Part 1. Part 1 also contains the listing of data word labels assigned for the transmission of broadcast periodic data.

Part 2 contains a tabulation of the ever-increasing list of Discrete data words used to provide status information.

C. CHANGES TO ARINC SPECIFICATION 429 INTRODUCED BY THIS SUPPLEMENT

This section presents a complete tabulation of the changes and additions to the Specification introduced by this Supplement. Each change and addition is defined by the section number and the title currently employed in the Specification or by the section name and title that will be employed when the Supplement is eventually incorporated. In each case, a brief description of the change or addition is included. A tabulation of sections is included with this supplement to enable the reader to correlate the previous section assignments with the new Part 3 Supplement 15 section number assignments.

The following changes affect only ARINC Specification 429-15, Part 3, File Data Transfer Techniques. Refer to Parts 1 and 2 for changes impacting the broadcast provisions of ARINC Specification 429.

ATTACHMENT 10 - VARIABLES OF BIT-ORIENTED PROTOCOL

Revised Notes 1 and 4.

Table 10-3 BIT-ORIENTED PROTOCOL OPTIONS - Added Option 0_{12}.

Table 10-5 VARIABLES OF HIGH SPEED BIT-ORIENTED PROTOCOL - Revised Time T_{10} min and max values.

ATTACHMENT 11 - BIT-ORIENTED DATA FILE TRANSFER WORD FORMATS

Table 11-6A GENERAL FORMAT IDENTIFIER (GFI) - Revised “Reserved ISO 9577” to “ISO 9577”

ATTACHMENT 11A - DESTINATION CODES

Added Cabin Packet Data Function. Corrected Ground Station bit encoding.

ATTACHMENT 11B - STATUS CODES

Revised description of Code 86. Added entries for Code 8E through 95.

ATTACHMENT 14 - SYSTEM ADDRESS LABELS

The following labels were added:

170	DFDAU (Mandatory Load Function)
266	Cabin Video System (Airshow)
334	Cabin Telecommunications Unit (CTU)
340	HF Data Radio/Data Unit #1
344	HF Data Radio/Data Unit #2

The following labels were revised:

175	HGA HPA
176	Spare
177	LGA HPA

APPENDIX 8 - INTEROPERABILITY OF BIT-ORIENTED LINK LAYER PROTOCOL

Appendix added.

APPENDIX 9 - SDL DIAGRAMS OF THE WILLIAMSBURG PROTOCOL

Appendix added.

ARINC Specification 429 is now available in three separate parts: Part 1 "Functional Description and Word Formats", Part 2 "Digital Information Transfer System Standards" and Part 3 "File Data Transfer Techniques." The changes are described in Supplements printed on goldenrod colored paper. The following pages provided a record of the section numbering of the text now included in Part 3.

Old Number (Supp 14)	New Number (Supp 15)	Section Title
1.1	1.1	Purpose of this Document
1.2	1.2	Relationship to ARINC Specification 419 changed to 429
1.3	1.3	"Mark 33 Digital Information Transfer System"_ - Basic Philosophy **changed to: "File Data Transfer Techniques"**
1.3.1	1.3.1	Number Data Transfer **changed to: Data Transfer**
1.3.2	1.3.2	ISO Alphabet No. 4 Data Transfer **changed to: Broadcast Data**
1.3.3	1.3.3	Graphic Data Transfer
2.3.1.5	2.1	File Data Transfer
2.3.1.5.1	2.3	Bit-Oriented Protocol Determination
2.3.2	2.2	Transmission Order
2.5	Chapter 3.0	Bit-Oriented Communications Protocol **changed to: Bit Oriented File Transfer Protocol**
2.5	3.1	Bit-Oriented File Transfer Protocol
2.5.1	3.2	Link Data Units (LDU)
2.5.2	3.3	Link Data Unit (LDU) Size and Word Count
2.5.3	3.4	System Address Labels (SAL)
2.5.4	3.5	Bit Rate and Word Timing
2.5.5	3.6	Word Type
2.5.6	3.7	Protocol Words
2.5.6.1	3.7.1	Protocol Identifier
2.5.6.2	3.7.2	Destination Code
2.5.6.3	3.7.3	Word Count
2.5.7	3.8	Request To Send (RTS)
2.5.7.1	3.8.1	Clear To Send (CTS)
2.5.7.2	3.8.2	Not Clear To Send (NCTS)
2.5.7.3	3.8.3	Destination Busy (BUSY)
2.5.7.4	3.8.4	No Response to RTS
2.5.8	3.9	Conflicting RTS Transmissions
2.5.8.1	3.9.1	Half Duplex Mode
2.5.8.2	3.9.2	Full Duplex Mode
2.5.9	3.10	Unexpected RTS
2.5.10	3.11	Start of Transmission (SOT)

Old Number (Supp 14)	New Number (Supp 15)	Section Title
2.5.10.1	3.11.1	General Format Identifier (GFI)
2.5.10.2	3.11.2	File Sequence Number
2.5.10.3	3.11.2	LDU Sequence Number
2.5.11	3.12	Data
2.5.11.1	3.12.1	Full Data Word(s)
2.5.11.2	3.12.2	Partial Data Word(s)
2.5.12	3.13	End of Transmission (EOT)
2.5.12.1	3.13.1	CRC Encoding
2.5.12.2	3.13.2	CRC Decoding
2.5.13	3.14	Negative Acknowledgement (NAK)
2.5.13.1	3.14.1	Missing SOT Word
2.5.13.2	3.14.2	Missing EOT Word
2.5.13.3	3.14.3	Parity Errors
2.5.13.4	3.14.4	Word Count Errors
2.5.13.5	3.14.5	CRC Errors
2.5.13.6	3.14.6	Time Out Errors
2.5.14	3.15	LDU Transfer Acknowledgement (ACK)
2.5.14.1	3.15.1	Duplicate LDU
2.5.14.2	3.15.2	Auto-Synchronized Files
2.5.14.3	3.15.3	Incomplete File Time
2.5.15	3.16	SYN Word
2.5.16	3.17	Response to ACK/NAK/SN
2.5.17	3.18	Solo Word
2.5.17.1	3.18.1	Test Word and Loop Word
2.5.17.2	3.18.2	Optional Solo Word Definitions
2.5.18	3.19	Optional End-to-End Message Verification
2.5.19	3.20	Protocol Initialization
2.5.19.1	3.20.1	Bit-Oriented Protocol Version
2.5.19.1.1	3.20.1.1	ALOHA
2.5.19.1.2	3.20.1.2	ALOHA Response
2.5.19.2	3.20.2	Williamsburg/File Transfer Determination
2.6	Chapter 4.0	Window Bit-Oriented Protocol **change to:** **Window Bit-Oriented File Transfer Protocol**
2.6	4.1	Windowed Bit-Oriented Protocol **change to:** **Windowed Bit-Oriented Communications Protocol**
2.6.1	4.2	Window Size
2.6.2	4.3	Window Definition

Old Number (Supp 14)	New Number (Supp 15)	Section Title
2.6.3	4.4	Protocol Word Deltas
2.6.3.1	4.4.1	Request To Send (RTS)
2.6.3.1.1	4.4.1.1	Unexpected Request To Send (RTS)
2.6.3.2	4.4.2	Clear To Send (CTS)
2.6.3.3	4.4.3	Not Clear To Send (NCTS)
2.6.3.4	4.4.4	BUSY
2.6.3.5	4.4.5	LDU Control Word (LCW)
2.6.3.6	4.4.6	Start Of Transmission (SOT)
2.6.3.7	4.4.7	Negative Acknowledgement (NAK)
2.6.3.7.1	4.4.7.1	Missing LDU Control Word
2.6.3.7.2	4.4.7.2	Missing SOT Word
2.6.3.7.3	4.4.7.3	Missing EOT Word
2.6.3.7.4	4.4.7.4	Parity Errors
2.6.3.7.5	4.4.7.5	Word Count Errors
2.6.3.7.6	4.4.7.6	CRC Errors
2.6.3.7.7	4.4.7.7	LDU Timeout Errors
2.6.3.8	4.4.8	Window Transfer Acknowledgement (ACK)
2.6.3.8.1	4.4.8.1	Determination of End of Window
2.6.3.8.2	4.4.8.2	Incomplete File Timer
2.6.4	4.5	Bit Rate and Word Timing
2.6.5	4.6	Response to ACK/NAK/SYN
2.6.6	4.7	Protocol Initialization Deltas
2.6.6.1	4.7.1	ALOHA
2.6.6.2	4.7.2	ALOHA Response
ATT 10 Table 10-1 Table 10-2 Table 10-3 Table 10-4 Table 10-5	ATT 1 Table 1-1 Table 1-2 Table 1-3 Table 1-4 Table 1-5	Variables of Bit-Oriented Protocol Bit-Oriented Protocol Events Bit-Oriented Protocol Application Selection Bit-Oriented Protocol Options Variables of Low Speed Bit-Oriented Protocol Variables of High Speed Bit-Oriented Protocol
ATT 11 Table 11-1 Table 11-1A Table 11-2 Table 11-2 Table 11-4 Table11-4A Table 11-4B Table 11-5 Table 11-6 Table 11-6A Table 11-7 Table 11-7A	ATT 2 Table 2-1 Table 2-1A Table 2-2 Table 2-3 Table 2-4 Table 2-4A Table 2-4B Table 2-5 Table 2-6 Table 2-6A Table 2-7 Table 2-7A	Bit-Oriented Data File Transfer Word Formats General Word Format Word Type Full Data Word Partial Data Word Protocol Word ALO-ALR Version Number ALO-ALR Window Size Solo Word Start of Transmission General Format Identifier (GFI) End of Transmission Final LDU Bit
ATT 11A	ATT 3	Destination Codes

Old Number (Supp 14)	New Number (Supp 15)	Section Title
ATT 11B	ATT 4	Status Codes
ATT 11C Table 11C-1 Table 11C-2 Table 11C-3	ATT 5 Table 5-1 Table 5-2 Table 5-3	ALOHA/ALOHA Response Protocol Word Definition ALOHA Initial (ALO) Protocol Word ALOHA Response (ALR) Protocol Word Version Number for ALO/ALR Protocol Words
ATT 12	ATT 6	File Transfer Example
ATT 12A	ATT 7	Field Mapping Exchange
ATT 13 Diagram 13-1 Diagram 13-2 Diagram 13-3	ATT 8 Diagram 8-1 Diagram 8-2 Diagram 8-3	Protocol Determination procedure Diagrams Protocol Determination Procedure Diagram (Two Bilingual Units) Protocol Determination Procedure Diagram (One Bilingual Unit and One Character-Only Unit)
ATT 13A	ATT 9	ALOHA Version Determination Sequence
ATT 15	ATT 10	Link Layer CRC Data Example
ATT 16	ATT 11	Sequence of Protocol and Data Words in Window Transfer
ATT 17	ATT 12	Flow Diagram Used to Determine Character-Oriented vs Bit-Oriented Protocol
APPENDIX 6 A6-2.1.5 A6-2.1.5.2 A6-2.1.5.3 A6-2.3.1 A6-2.3.1.4 A6-2.3.1.5 A6-2.3.1.5.1 A6-2.3.1.5.2 A6-2.3.1.5.3 A6-2.3.1.5.4 A6-2.3.1.5.5 A6-2.3.1.5.6 A6-2.3.1.5.7 A6-3.2 A6-3.2.1 A6-3.2.2 A6-3.2.3	APPENDIX A A2.1.5 A2.1.5.2 A2.1.5.3 A2.3.1 A2.3.1.4 A2.3.1.5 A2.3.1.5.1 A2.3.1.5.2 A2.3.1.5.3 A2.3.1.5.4 A2.3.1.5.5 A2.3.1.5.6 A2.3.1.5.7 A-3.2 A3.2.1 A3.2.2 A3.2.3	Former AIM and File Data Transfer Techniques Sign/Status Matrix AIM Data Character-Oriented File Transfer Digital Language AIM Data File Data Transfer Command/Response Protocol Initial Word Types Intermediate Words Final Words Word Type Encoding File Data Formats File Data Labels AIM Information Transfer Acknowledgement Data ISO Alphabet No. 5 Data Maintenance Data
APPENDIX 7	APPENDIX B	Mathematical Example of CRC Encoding/Decoding
APPENDIX 8	APPENDIX C	Interoperability of Bit-oriented Link Layer Protocol
APPENDIX 9	APPENDIX D	SDL Diagrams of Williamsburg Protocol

SUPPLEMENT 16

TO

ARINC SPECIFICATION 429©

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

PART 3

FILE DATA TRANSFER TECHNIQUES

Published: June 30, 1997

Prepared by the Airlines Electronic Engineering Committee

Adopted by the Airlines Electronic Engineering Committee: June 24, 1997

A. PURPOSE OF THIS SUPPLEMENT

This Supplement reorganizes Part 3 to be consistent with previous published versions of ARINC Specification 429. It also restores several paragraphs missing from Supplement 15.

The technical changes include clarification of the Version 1 (Williamsburg) protocol, deletion of the Version 2 protocol, and creation of the Version 3 protocol. The definition of the Version 3 protocol will be completed in a future Supplement.

B. ORGANIZATION OF THIS SUPPLEMENT

The first part of this document, printed on goldrod colored paper, contains descriptions of the changes introduced into this Specification by this Supplement. The second part, printed on white paper, contains the changes made to the specification. The modified and added material on each page is identified by a c-16 in the margins. In view of the document reorganization, ARINC Specification 429, Part 3, is reprinted in its entirety as ARINC Specification 429-16, Part 3.

C. CHANGES TO ARINC SPECIFICATION 429, INTRODUCED BY THIS SUPPLEMENT

This section presents a complete tabulation of the changes and additions to the Specification introduced by this Supplement. Each change and addition is defined by the Section number and the title that will be employed when the Supplement is eventually incorporated. In each case, a brief description of the change or addition is included.

1.0 Introduction

This section contains a reorganization of the material previously in Section 1.0 and 2.0 of ARINC Specification 419P3-15 with the following exceptions.

The section on Graphic Data Transfer (formerly 1.3.4) is deleted, Sections 1.3, 1.3.1, 1.3.2, and 1.3.4 include minor changes clarifying the background for file data transfer, and Section 1.3.6 on Bit-Oriented Protocol Determination (formerly 1.3.7) was revised to refer to Section 2.5.19.

2.0 Bit-Oriented File Transfer Protocol

Section number 2.1 – 2.4 have been inserted as placeholders to re-establish section numbering consistency with ARINC Specification 429-14 and its predecessors.

Section 2.5 and subsections contain the material previously published in Section 3.0 of ARINC Specification 429P3-15, as modified below.

2.5 Bit-Oriented Communications Protocol

References to Attachments 12 and 12A updated.

2.5.1 Link Data Units (LDU)

The definition of LDU is clarified.

2.5.3 System Address labels (SALs)

Commentary on use of SALs clarified.

2.5.4 Bit Rate and Word Timing

Commentary on use of word gap criteria clarified.

2.5.6.2 Destination Code

Introduction to section added, and use of Destination Code clarified.

2.5.6.3 Word Count

Introduction to section added.

2.5.7.3 Destination Busy (BUSY)

The use of Option 3 (Send Auto CTS) and Option 4 (Accept Auto CTS) is clarified.

2.5.81 Half Duplex Mode

This section restores text missing from the published version of Part 3, Supplement 15.

2.5.11.2 Partial Data Word

Location of the length of a partial data word is clarified.

2.5.14.1 Duplicate LDU

The definition of a duplicate LDU is clarified.

2.5.15 SYN Word

The definition of a duplicate LDU is clarified.

2.5.19 Protocol Initialization

The protocol version determination is clarified.

2.5.19.1 Bit-Oriented Protocol Version

The protocol version determination is clarified.

2.5.19.1.1 ALOHA

The protocol version determination is clarified.

2.5.19.1.2 ALOHA Response

This section restores text missing from the published version of Part 3, Supplement 15.

2.5.19.2 Williamsburg/File Transfer Determination

This section restores text missing from the published version of Part 3, Supplement 15, and commentary is added on use of a NAK in the protocol determination logic.

2.6 Windowed Bit-Oriented Communications Protocol

Section 2.6 and subsections have been deleted. Section 2.6 contained the definition of Version 2 of the Williamsburg protocol. Version 2 of the Williamsburg protocol has been superseded by Version 3. Section 2.6 and subsections contained the material previously published in Section 4.0 of ARINC Specification 429P3-15.

3.0 Bit-Oriented Media Access Control (MAC)

An introduction to the Bit-Oriented Media Access Control (Williamsburg Version 3) protocol is added.

ATTACHMENTS 1-17

Attachment numbers 1-9 have been inserted as placeholders to re-establish section numbering consistency with ARINC Specification 429-14 and its predecessors.

Attachments 10-17 contain material published in Attachments 1-12 of ARINC Specification 429P3-15, as modified below.

ATTACHMENT 10 - VAIABLES OF BIT-ORIENTED PROTOCOL

Table 10-3 is replaced with Table 10-3A, containing options for Version 1. Tables 10-3B, 10-6 and 10-7 are added as placeholders for Version 3 Williamsburg. Variables for the Version 2 protocol in Tables 10-1 and 10-3A are deleted.

ATTACHMENT 11 - BIT-ORITENTED DATA FILE TRANSFER WORD FORMATS

The general word format in Table 11-1 is clarified.

The LCW protocol word format in Table 11-4 is modified.

Table 11-4 is modified to add the Service Class Identifier to the LCW format.

Table 11-4B is deleted as part of the Version 2 protocol.

Table 11-4A is modified to add the version number for Version 3, and delete references to Version 2.

ATTACHMENT 11A - DESTINATION CODES

The destination Code N for FMC, Center, is added.

ATTACHMENT 11C - ALOHA/ALOHA RESPONSE PROTOCOL WORD DEFINTION

The ALOHA and ALOHA Response protocol word definitions are revised to be consistent with other changes made to the protocol, and the titles of the tables modified to indicate they are examples.

ATTACHMENT 12 - VERSION 1 FILE TRANSFER EXAMPLE

The title is changed to indicate Version 1.

ATTACHMENT 12A - FIELD MAPPING EXAMPLE

Attachment 12A is replaced with an updated example

ATTACHMENT 15 - LINK LAYER CRC DATA EXAMPALE

This section is deleted as part of the Version 2 protocol.

ATTACHMENT 16 - SEQUENCE OF PROTOCOL AND DATA WORDS IN WINDOWN TRANSFER

This section is deleted as part of the Version 2 protocol.

APPENDICES A-K

Appendix numbers A-E have been inserted as placeholders to re-establish section numbering consistency with ARINC Specification 429-14 and its predecessor.

Appendices F-J contain the material published in Appendices A-D of ARINC 429P3-15.

Appendix H was revised to reflect the deletion of the Version 2 protocol and creation of the Version 3 protocol.

SUPPLEMENT 17

TO

ARINC SPECIFICATION 429©

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

PART 3

FILE DATA TRANSFER TECHNIQUES

Published: May 31, 1999

Prepared by the Airlines Electronic Engineering Committee

Adopted by the Airlines Electronic Engineering Committee: March 31, 1999

A. PURPOSE OF THIS SUPPLEMENT

This Supplement introduces the definition of a new bit-oriented file data transfer protocol. The protocol is designed to be consistent with the IEEE-802 Media Access Control (MAC) protocol definition. Version 3 fills the role intended for Version 2 of the Williamsburg protocol by providing a high throughput avionics file data transfer interface. Version 2 was deleted by Supplement 16. Version 3 is intended to be capable of being bridged to other common data bus protocols, most significantly, Ethernet.

B. ORGANIZATION OF THIS SUPPLEMENT

Changes introduced by Supplement 17 were deemed sufficiently significant to issue an entirely new publication of Specification 429 Part 3. There is no standalone Supplement.

This part, printed on goldenrod-colored paper, contains a list of descriptions of changes introduced into this Specification by this Supplement 17.

In the body of the document, the changes (.i.e., the modified and added material) introduced by Supplement 17 are identified by c-17 change bars in the margins.

C. CHANGES TO ARINC SPECIFICATION 429, PART 3 INTRODUCED BY THIS SUPPLEMENT

This section presents a complete tabulation of the changes and additions to the Specification introduced by this Supplement. In the text below, the Section number and title of each affected Section, Attachment or Appendix is listed, followed by a brief description of the change or addition.

1.3.5 Transmission Order

Transmission order of bits was clarified.

1.4 Relationship to Other Standards

A new section was added. It discusses the relationship of this document to other AEEC documents and to other industry documents.

3.0 Bit-Oriented Media Access Control Protocol

The definition of the bit-oriented Media Access Control (Williamsburg Version 3) protocol was added, replacing introductory text inserted by Supplement 16 as a placeholder.

ATTACHMENT 10 - VARIABLES OF BIT-ORIENTED PROTOCOL

Table 10-3B, containing options for Version 3, was added.

Table 10-6, containing timer values for the ARINC 429 high-speed Version 3 bus, was added.

Table 10-7, containing a placeholder for low speed bus timers associated with Version 3 protocol was deleted because the low speed implementation is not recommended.

ATTACHMENT 11 - BIT ORIENTED DATA FILE TRANSFER WORD FORMATS

Table 11-1A was updated to add Version 3 SOF and EOF words.

Table 11-8 was added defining the command frame SOF.

Table 11-9 was added defining the command frame EOF.

Table 11-10 was added defining the information frame SOF.

Table 11-11 was added defining the information frame EOF.

ATTACHMENT 18 – MAC SUBLAYER SUPPORT DIAGRAMS

New Attachment added.

ATTACHMENT 19 – COMMAND FRAME DATA UNIT (FDU) STRUCTURE AND EXAMPLES

New Attachment added.

ATTACHMENT 20 – INFORMATION FRAME DATA UNIT (FDU) STRUCTURE AND EXAMPLES

New Attachment added.

APPENDIX 8 - INTEROPERABILITY OF BIT-ORIENTED LINK LAYER PROTOCOL

Appendix 8 is updated to discuss interoperability between Version 1 and Version 3.

APPENDIX 10 - ARINC 429 WILLIAMSBURG PROTOCOL LAYER DIAGRAM

A new Appendix was added providing a general overview of the protocol structure over different communication stacks.

SUPPLEMENT 18

TO

ARINC SPECIFICATION 429©

MARK 33 DIGITAL INFORMATION TRANSFER SYSTEM (DITS)

PART 3

FILE DATA TRANSFER TECHNIQUES

Published: October 12, 2001

Prepared by the Airlines Electronic Engineering Committee

Adopted by the Airlines Electronic Engineering Committee: July 18, 2001

A. PURPOSE OF THIS SUPPLEMENT

This Supplement introduces the assignment of 3 new satellite links, HFDR Right, TAWS, and CVR into the Destination Code table.

A Table was added to define the Variables of Low Speed Connectionless Bit-Oriented Protocol.

Typographical errors were corrected in the text.

B. ORGANIZATION OF THIS SUPPLEMENT

The first part of this document printed on golden-rod paper contains descriptions of changes introduced into this Specification by this Supplement.

The changes introduced by Supplement 18 have been identified using change bars and are labeled in the margin by a "c-18" indicator.

C. CHANGES TO ARINC SPECIFICATION 429, PART 3 INTRODUCED BY THIS SUPPLEMENT

This section presents a complete tabulation of the changes and additions to the Specification to be introduced by this Supplement. Each change or addition is identified by the section number and the title that will be employed for that section when the Supplement is eventually incorporated. In each case a brief description of the change or addition is included.

3.4.3 System Address Labels (SAL)

Corrected the reference to the table of SAL assignments (from Attachment 14 to Attachment 11) in ARINC Specification 429 Part 1.

3.4.4 Bit Rate and Word Timing

Provision was added to specify that the Williamsburg version 3 protocol may be operated at low speed.

ATTACHMENT 10 – VARIABLES TO BIT-ORIENTED PROTOCOL

Added new Table 10-7 to support low speed operation of Williamsburg protocol at low speed. Later modified the value of the variables.

ATTACHMENT 11A – DESTINATION CODES

The assignment of 'T' for the transponder was deleted. Six new entries, HFDR Right, TAWS, CVR Inmarsat, ICO, and Globalstar satellite link identifiers were added as destination codes. The format and content of the table was aligned with the corresponding Table 3-1 of Attachment 3 to ARINC Specification 619 to improve consistency.

APPENDIX A – J

These appendices were formerly identified as Appendix 1 – 10. During the regeneration of Specification Description Language (SDL) diagrams in Appendix I, references to Section 1.3.7 were revised to Section 2.5.19.

APPENDIX K

New Appendix added.