High Accuracy Time Synchronization over SpaceWire Networks

Time Synchronization Protocol

SPWCUC-REP-0003, 29 September 2012, Version 1.1
Sandi Habinc, sandi@gaisler.com
Aeroflex Gaisler AB, Kungsgatan 12, SE-411 19 Göteborg, Sweden, www.gaisler.com

Introduction

This document represents a draft of an ECSS standard which is in development. The document has been written as far as possible according to ECSS drafting rules. The document should not be considered an ECSS standard.

This version of the document encompasses basic time distribution and initialisation/synchronization. It does not completely cover configuration, status and low level synchronization.

Foreword

This Standard is one of the series of ECSS Standards intended to be applied together for the management, engineering and product assurance in space projects and applications. ECSS is a cooperative effort of the European Space Agency, national space agencies and European industry associations for the purpose of developing and maintaining common standards. Requirements in this Standard are defined in terms of what shall be accomplished, rather than in terms of how to organize and perform the necessary work. This allows existing organizational structures and methods to be applied where they are effective, and for the structures and methods to evolve as necessary without rewriting the standards.

This Standard has been prepared by the ECSS-E-ST-50-## Working Group, reviewed by the ECSS Executive Secretariat and approved by the ECSS Technical Authority.

Disclaimer

1 SCOPE

There is a number of communication protocols that can be used in conjunction with the SpaceWire Standard (ECSS-E-ST-50-12), to provide a comprehensive set of services for onboard user applications. To distinguish between the various protocols a protocol identifier is used, as specified in ECSS-E-ST-50-51.

This Standard specifies the Time Distribution Protocol, which is one of these protocols that work over SpaceWire.

The aim of the Time Distribution Protocol is to synchronize time across a SpaceWire network. It does this by an initiator writing a CCSDS Time Code using an RMAP command placed in a SpaceWire packet, transferring it across the SpaceWire network and then extracting the CCSDS Time Code at the target, and by means of SpaceWire time control codes (Time-Codes) used to convey the time instant at which the CCSDS Time Code becomes valid.

This standard may be tailored for the specific characteristic and constrains of a space project in conformance with ECSS-S-ST-00.

2 NORMATIVE REFERENCES

The following normative documents contain provisions which, through reference in this text, constitute provisions of this ECSS Standard. For dated references, subsequent amendments to, or revision of any of these publications do not apply. However, parties to agreements based on this ECSS Standard are encouraged to investigate the possibility of applying the more recent editions of the normative documents indicated below. For undated references, the latest edition of the publication referred to applies.

ECSS-S-ST-00-01	ECSS system – Glossary of terms
ECSS-E-ST-50-12C	Space Engineering - SpaceWire - Links, nodes, routers and networks
ECSS-E-ST-50-51C	Space Engineering - SpaceWire protocol identification
ECSS-E-ST-50-52C	Space Engineering - SpaceWire - Remote memory access protocol
CCSDS 301.0-B-4	Time Code Formats, Blue Book

3 TERMS, DEFINITIONS AND ABBREVIATED TERMS

3.1 Terms defined in other standards

For the purpose of this Standard, the terms and definitions from ECSS-S-ST-00-01, ECSS-E-ST-50-12, and ECSS-E-ST-50-51 apply.

3.1.1 data character [ECSS-E-ST-50-12C]

SpaceWire symbol containing 8-bits of user information, where bit 7 is the most-significant bit and transmitted last

3.1.2 control character [ECSS-E-ST-50-12C]

character that is used to pass control information across a link

3.1.3 control code [ECSS-E-ST-50-12C]

sequence of two **control characters**: … Time-Code (ESC + **data character**) which is used to distribute system time information over a SpaceWire network

3.1.4 Time-Code

code used to distribute system time over a SpaceWire network, which comprises ESC followed by a single **data character** holding six bits of the system time and two reserved bits

3.1.5 byte [ECSS-E-ST-50-51C]

8-bits where bit 7 is the most-significant bit

3.1.6 word [ECSS-E-ST-50-51C]

multiple **bytes** held in a single memory location

3.2 Terms specific to the present standard

3.2.1 octet [CCSDS 301.0-B-4]

8-bit data field, where bit 0 is the most-significant bit and transmitted first

3.3 Abbreviated terms

The following abbreviations are defined and used within this standard:

Abbreviation	**Meaning**
CCSDS	Consultative Committee for Space Data Systems
ASCII	CCSDS ASCII Calendar Segmented Code (ASCII)
CCS	CCSDS Calendar Segmented Time Code
CDS	CCSDS Day Segmented Time Code
CUC	CCSDS Unsegmented Time Code
LSB	least significant bit
MSB	most significant bit

P-Field	preamble field
RMAP	remote memory access protocol
T-Field	time field
TAI	International Atomic Time

3.4 Conventions

3.4.1 Prefix

In this Standard hexadecimal numbers are written with the prefix 0x, for example 0x34 and 0xDF15.

Binary numbers are written with the prefix 0b, for example 0b01001100 and 0b01.

Decimal numbers have no prefix.

3.4.2 Bit identification

In CCSDS recommendations, the following convention is used to identify each bit in an N-bit field. The first bit in the field to be transmitted (i.e., the most left justified when drawing a figure) is defined to be ‘Bit 0’; the following bit is defined to be ‘Bit 1’ and so on up to ‘Bit N-1’. When the field is used to express a binary value (such as a counter), the Most Significant Bit (MSB) shall be the first transmitted bit of the field, i.e., ‘Bit 0’.

CCSDS thus defines the opposite numbering and transmission order to what is specified for SpaceWire data characters and bytes. To maintain compatibility, the MSB of a CCSDS octet is mapped on the MSB of a SpaceWire byte, which in turn is transmitted as the last bit (also being the MSB) in a SpaceWire data character.

Fields defined by CCSDS are composed of octets. Each such octet is mapped on a SpaceWire byte. The terms octet and byte are used interchangeably in this Standard.

The MSB is drawn to the left in a figure or table, and the Least Significant Bit (LSB) to the right.

A word comprises four octets or bytes in this Standard. Byte ordering is always big-endian, i.e. the most significant byte is located to the left in a word. The most significant byte is addressed with the lowest byte address in a word. The most significant byte is transmitted first.

4 PRINCIPLES

4.1 Purpose

The Time Distribution Protocol has been designed to allow time synchronization across a SpaceWire network, by means of SpaceWire packets and SpaceWire Time-Codes.

4.2 Protocol features

The Time Distribution Protocol provides the capability to transfer CCSDS Time Codes between onboard users of a SpaceWire network. The CCSDS Time Codes may be of variable length or fixed size at the discretion of the user and may be submitted for transmission at variable time intervals, providing a communication service.

The Time Distribution Protocol provides the capability to synchronize nodes in a SpaceWire network by using SpaceWire time control codes (Time-Codes), providing a timing service.

An Initiator is a SpaceWire node distributing CCSDS Time Codes and SpaceWire time-control codes (Time-Code). An Initiator is also an RMAP initiator, capable of transmitting RMAP commands and receiving RMAP replies. There is only one active Initiator in a SpaceWire network during a mission phase.

A Target is a SpaceWire node receiving CCSDS Time Codes and SpaceWire time-control codes (Time-Codes). A Target is also an RMAP target, capable of receiving RMAP commands and transmitting RMAP replies. There can be one or more Targets in a SpaceWire network.

The protocol also provides means for time-stamping of incoming and outgoing SpaceWire time-control codes (Time-Code) in the Target, make this information accessible to an Initiator by means of RMAP accesses.

Note: SpaceWire time-control codes (Time-Code) in this context should be interpreted as the Distributed Interrupts currently being defined for ECSS-E-ST-50-12C Rev.1.

The protocol also provides means for transferring latency correction information from an Initiator to a Target by means of RMAP accesses.

4.3 Operation

The Initiator and the Target maintain their own time locally, for which the implementation is independent of this standard. The Time Distribution Protocol provides the means for transferring the time of the Initiator to the Targets, and for providing a synchronization point in time.

The time is transferred by means of an RMAP write command carrying a CCSDS Time Code. The synchronization event is signalled by means of transferring a SpaceWire time control code (Time-Code). The transfer of the SpaceWire Time-Code is synchronized with the time maintained by the Initiator.

To distinguish which SpaceWire Time-Code is to be used for synchronization, the value of the SpaceWire Time-Code is transferred from the Initiator to the Target by means of an

RMAP write command prior to the actual transmission of the SpaceWire Time-Code itself.

When there is more than a one Target, the CCSDS Time Code need be transferred to each individual Target separately (unless SpaceWire packet broadcast or multicast can be used). Only one transmission of the SpaceWire Time-Code is however need, although one can imagine systems where different SpaceWire Time-Code values are used for different Targets.

4.4 Services

The Time Distribution Protocol provides users with communication services based on RMAP service primitives and parameters, transferring amongst others CCSDS Time Codes.

The Time Distribution Protocol provides users with timing service based on SpaceWire time-control codes (Time-Codes).

The followings services are defined in this Standard:

- Configuration
- Status
- Command (CCSDS Time Code)
- Datation
- Timing (Initialisation/Synchronization)
- Time-Stamp (of SpaceWire time-control codes (Time-Codes)).
- Latency

4.5 Guide to Clause 5

Clause 5.1 defines the service parameters.
Clause 5.2 specifies the service primitives provided by the protocol.
Clause 5.3 defines the protocol fields used in the packets and registers.
Clause 5.4 specifies the format of the packets and registers used by the protocol.
Clause 5.5 specifies the action of the protocol.

5 REQUIREMENTS

5.1 Service parameters

5.1.1 RMAP services parameters

a. All communication shall be performed by means of services parameters provided by RMAP.

5.1.2 Other service parameters

NOTE 1 The parameters for each service are defined with the service primitives in 5.2.

5.2 Service primitives

5.2.1 RMAP services primitives

a. All communication shall be performed by means of services primitives provided by RMAP.

5.2.2 Configuration Service

a. The service primitives associated with this service shall be the following
   1. TIME_CONFIGURE.request (at Initiator)
   2. TIME_CONFIGURE.indication (at Target)
   3. TIME_CONFIGURE.response (at Target)
   4. TIME_CONFIGURE.confirmation (at Initiator)
b. The service parameters are defined in 5.3.2.

5.2.3 Status Service

a. The service primitives associated with this service shall be the following
   1. TIME_STATUS.request (at Initiator)
   2. TIME_STATUS.indication (at Target)
   3. TIME_STATUS.response (at Target)
   4. TIME_STATUS.confirmation (at Initiator)
b. The service parameters are defined in 5.3.3.

5.2.4 Command Service

a. The service primitives associated with this service shall be the following
   1. TIME_COMMAND.request (at Initiator)
   2. TIME_COMMAND.indication (at Target)
   3. TIME_COMMAND.response (at Target)
   4. TIME_COMMAND.confirmation (at Initiator)
b. The service parameters are defined in 5.3.4.

5.2.5 Datation Service

a. The service primitives associated with this service shall be the following
   1. TIME_DATATION.request (at Initiator)
   2. TIME_DATATION.indication (at Target)

3. TIME_DATATION.response (at Target)
4. TIME_DATATION.confirmation (at Initiator)

b. The service parameters are defined in 5.3.5.

5.2.6 Initialisation/Synchronization Service

a. The service primitives associated with this service shall be the following
   1. TIME_SYNC.request (at Initiator)
   2. TIME_SYNC.indication (at Target)

b. The service parameter is the SpaceWire time-control code (Time-Code).

5.2.7 Time-stamp Services

5.2.7.1 Time-stamp control and read-out

a. The service primitives associated with this service shall be the following
   1. TIME_STAMP.request (at Initiator)
   2. TIME_STAMP.indication (at Target)
   3. TIME_STAMP.response (at Target)
   4. TIME_STAMP.confirmation (at Initiator)

b. The service parameters are defined in 5.3.65.3.2.

5.2.7.2 Time-Stamp when receiving

a. The service primitives associated with this service shall be the following
   1. TIME_SYNC.request (at Initiator)
   2. TIME_SYNC.indication (at Target)

b. The service parameter is the SpaceWire time-control code (Time-Code).

5.2.7.3 Time-Stamp when sending

a. The service primitives associated with this service shall be the following
   1. TIME_SYNC.request (at Target)
   2. TIME_SYNC.indication (at initiator)

b. The service parameter is the SpaceWire time-control code (Time-Code).

5.2.8 Latency Service

a. The service primitives associated with this service shall be the following
   1. TIME_LATENCY.request (at Initiator)
   2. TIME_LATENCY.indication (at Target)
   3. TIME_LATENCY.response (at Target)
   4. TIME_LATENCY.confirmation (at Initiator)

b. The service parameters are defined in 5.3.7.

5.3 Fields

5.3.1 RMAP fields

5.3.1.1 General

a. All RMAP command and reply fields shall be supported.

5.3.1.2 RMAP Target SpaceWire Address field

a. The Target SpaceWire Address field shall be supported as defined for RMAP.

5.3.1.3 RMAP Target Logical Address field

a. The Target Logical Address field shall be supported as defined for RMAP.

5.3.1.4 RMAP Protocol Identifier field

a. The Protocol Identifier field shall be set to the value 1 (0x01) which is the Protocol Identifier for the Remote Memory Access Protocol.

NOTE 1 The Protocol Identifier will be changed in the future to a unique value to identify the Time Distribution Protocol.

5.3.1.5 RMAP Instruction field

a. All Command Codes of the RMAP Command field shall be supported, except:
    1. Single (non-incrementing) address shall not be supported
    2. Read-Modify-Write incrementing address shall not be supported
    3. Write, incrementing addresses, verify before writing (send reply/no reply) may be supported

5.3.1.6 RMAP Key field

a. The Key field shall be supported as defined for RMAP.

NOTE 1 If not implemented in a target, the default value is 0x00.

5.3.1.7 RMAP Reply Address field

a. The Reply Address field shall be supported as defined for RMAP.

5.3.1.8 RMAP Initiator Logical Address field

a. The Initiator Logical Address field shall be supported as defined for RMAP.

5.3.1.9 RMAP Transaction Identifier field

a. The Transaction Identifier field shall be supported as defined for RMAP.

5.3.1.10 RMAP Extended Address field

a. The Extended Address field shall be supported as defined for RMAP.

NOTE 1 The interpretation of this field is implementation dependent.

5.3.1.11 RMAP Address field

a. The Address field shall be as defined for RMAP.
b. Incrementing addressing shall always be used.
c. The address shall represent a byte.

NOTE 1 The address mapping is defined in 5.4.3.

5.3.1.12 RMAP Data Length field

a. The Data Length field shall be supported as defined for RMAP.

5.3.1.13 RMAP Header CRC field

a. The Header CRC field shall be supported as defined for RMAP.

5.3.1.14 RMAP Data field

a. The contents of the RMAP Data field shall have a one-to-one match to the register or registers, as defined in 5.4.3, which is being accessed by an RMAP command.
b. The RMAP Data field shall be a multiple of 32-bit words (4 bytes).
c. Unused bytes shall be transmitted as 0x00 by the sender and be ignored by the receiver.

5.3.1.15 RMAP Mask field

a. The Mask field defined for RMAP shall not be supported.

5.3.1.16 RMAP Data CRC field

a. The Data CRC field shall be supported as defined for RMAP.

5.3.1.17 RMAP Reply SpaceWire Address field

a. The Reply SpaceWire Address field shall be supported as defined for RMAP.

5.3.1.18 RMAP Status field

a. The Status field shall be supported as defined for RMAP.

5.3.2 Configuration field

a. The Configuration field shall be a 128-bit composite field.
b. Contents are TBD.

5.3.3 Status field

c. The Status field shall be a 128-bit composite field.
d. Contents are TBD.

5.3.4 Command field

a. The Command field shall be a composite field comprising the Control field, the SpaceWire Time-Code field, and the CCSDS Time Code fields.

5.3.4.1 Control field

a. The Control field shall be an 8-bit composite field comprising the New field and the Initialise field.
b. Unused bits shall be transmitted as 0b by the sender and shall be ignored by the receiver.

5.3.4.1.1 New field

a. New field shall be:
    1. A 1-bit field
    2. Located in bit numbered 7 (MSB of byte)
    3. Specifying a new command when set to 1b
    4. Specifying an old command when set to 0b

5.3.4.1.2 Initialise field

a. Initialise field shall be:
    1. A 1-bit field
    2. Located in bit numbered 6
    3. Specifying initialisation command when set to 1b
    4. Specifying synchronization command when set to 0b

5.3.4.2 Time-Code field

a. The Time-Code field shall be an 8-bit field containing a SpaceWire time-control code (Time-Code).

5.3.4.3 CCSDS Time Code fields

a. The fields defined for the following CCSDS Time Code shall be supported:
   1. CCSDS Unsegmented Time Code (CUC)

   NOTE 1 The CUC format allows many variants, allowing one or two octets for the P-Fields, one to seventeen octets for the T-Field.

b. The fields defined for the following CCSDS Time Codes may be supported:
   1. CCSDS Calendar Segmented Time Code (CCS)
   2. CCSDS Day Segmented Time Code (CDS)

5.3.5 Datation field

a. The Datation field shall be a composite field comprising the CCSDS Time Code fields.

5.3.5.1 CCSDS Time Code fields

a. The fields defined for the following CCSDS Time Code shall be supported:
   1. CCSDS Unsegmented Time Code (CUC)
b. The fields defined for the following CCSDS Time Codes should be supported:
   1. CCSDS Calendar Segmented Time Code (CCS)
   2. CCSDS Day Segmented Time Code (CDS)

NOTE 1 The Datation field needs not match the corresponding fields of the Command field.

5.3.6 Time-Stamp field

a. The Time-Stamp field shall be a composite field comprising the SpaceWire Time-Code field, and two sets of CCSDS Time Code fields.

5.3.6.1 Time-Code field

a. The Time-Code field shall be an 8-bit field containing a SpaceWire time-control code (Time-Code).

NOTE 1 SpaceWire time-control codes (Time-Code) in this context should be interpreted as the Distributed Interrupts currently being defined for ECSS-E-ST-50-12C Rev.1.

5.3.6.2 CCSDS Time Code fields

a. The fields defined for the following CCSDS Time Code shall be supported:
   a. CCSDS Unsegmented Time Code (CUC)
b. The fields defined for the following CCSDS Time Codes should be supported:
   a. CCSDS Calendar Segmented Time Code (CCS)

b. CCSDS Day Segmented Time Code (CDS)

5.3.7 Latency field

a. The Latency field shall be a composite field comprising the CCSDS Time Code fields.

5.3.7.1 CCSDS Time Code fields

a. The fields defined for the following CCSDS Time Code shall be supported:
    a. CCSDS Unsegmented Time Code (CUC)
b. The fields defined for the following CCSDS Time Codes should be supported:
    a. CCSDS Calendar Segmented Time Code (CCS)
    b. CCSDS Day Segmented Time Code (CDS)

5.4 Formats

5.4.1 RMAP commands and replies

a. All communication shall be performed by means of RMAP commands and RMAP replies.

5.4.2 RMAP Data field

a. The Data field of the RMAP commands and RMAP replies shall contain the fields defined in 5.3.2 through 5.3.5.

5.4.3 RMAP address space

1. The Target shall implement an RMAP address space into which all remotely accessible registers shall be mapped.
2. All RMAP addressing shall be performed using the address specified for each register.

   NOTE 1 Only the least significant address bits are specified herein, higher order address bits are implementation specific.
3. The RMAP address space shall comprise the following registers:
   1. Configuration
   2. Status
   3. Command
   4. Datation
4. Configuration register shall:
   1. Contain the Configuration field defined in 5.3.2
   2. Be readable and writable
   3. Be addressed at 0x00
5. Status register shall:
   1. Contain the Status field defined in 5.3.3
   2. Be readable and writable
   3. Be address at 0x10
6. Command register shall:
   1. Contain the Command field defined in 5.3.4
   2. Be readable and writable
   3. Be addressed at:
      a. Control field located at 0x020
      b. Time-Code field located at 0x021
      c. CCSDS Time Code field containing:
         1. P-Field Octet 1 located at 0x22
         2. P-Field Octet 2 located at 0x23, if used
         3. T-Field starting at 0x24
         4. T-Field ending at 0x3F

         NOTE 1 The T-Field has a variable length, and ends at a lower address value. The T-Field is left-aligned, towards lower address values.
7. Datation register shall:
   1. Contain the Datation field defined in 5.3.5
   2. Be readable
   3. Be addressed at:
      a. CCSDS Time Code field containing:
         1. Unused byte located at 0x40
         2. Unused byte located at 0x41
         3. P-Field Octet 1 located at 0x42
         4. P-Field Octet 2 located at 0x43, if used
         5. T-Field starting at 0x44
         6. T-Field ending at 0x5F

         NOTE 1 The T-Field has a variable length, and ends at a lower address value. The T-Field is left-aligned, towards lower address values.

8. Time-stamp register shall:
    1. Contain the Time-Stamp field defined in 5.3.6
    2. Be readable and writable
    3. Be addressed at:
        a. Time-Code field located at 0x60
        b. Unused byte located at 0x61
        c. CCSDS Time Code field containing:
            1. P-Field Octet 1 located at 0x62
            2. P-Field Octet 2 located at 0x63, if used
            3. T-Field starting at 0x64
            4. T-Field ending at 0x7F
        d. Unused byte located at 0x80
        e. Unused byte located at 0x81
        f. CCSDS Time Code field containing:
            1. P-Field Octet 1 located at 0x82
            2. P-Field Octet 2 located at 0x83, if used
            3. T-Field starting at 0x84
            4. T-Field ending at 0x9F

NOTE 1 The T-Field has a variable length, and ends at a lower address value. The T-Field is left-aligned, towards lower address values.

NOTE 2 A target need not support all CCSDS Time Codes. By implementing a read-only P-Field in the target, the initiator is informed of the limitation.

9. Latency register shall:
    1. Contain the Latency field defined in 5.3.7
    2. Be readable and writable
    3. Be addressed at:
        a. CCSDS Time Code field containing:
            1. Unused byte located at 0xA0
            2. Unused byte located at 0xA1
            3. P-Field Octet 1 located at 0xA2
            4. P-Field Octet 2 located at 0xA3, if used
            5. T-Field starting at 0xA4
            6. T-Field ending at 0xBF

NOTE 1 The T-Field has a variable length, and ends at a lower address value. The T-Field is left-aligned, towards lower address values.

NOTE 2 A target need not support all CCSDS Time Codes. By implementing a read-only P-Field in the target, the initiator is informed of the limitation.

5.5 Actions

5.5.1 Overview

This standard comprises four communication actions based on RMAP actions:

- Configure target
- Obtain status from target
- Command target (initialisation/synchronization)
- Obtain datation from target

This standard comprises one timing action based on SpaceWire time-control codes (Time-Codes):

- Initialise/Synchronize

5.5.2 Communication actions

a. Communication actions shall be performed by means of actions defined for RMAP commands and RMAP replies.

5.5.2.1 Configure target

a. Configuration of the target shall be made by writing the Configuration field.
b. The configuration of the target shall be obtained by reading the Configuration field.

5.5.2.2 Obtain status from target

a. The status of the target shall be obtained by reading the Status field.

5.5.2.3 Command target (initialisation/synchronization)

a. Preparation for a timing action in the target shall be made by writing the Command field, such that:
    1. New field specifies that a new command has been received
    2. Initialise field specifies whether Initialisation or Synchronization is due
    3. Time-Code field specifies the value of the received time control code (Time-Code) that qualifies the timing action
    4. CCSDS Time Code field specifies the distributed time at the moment of the timing action.

5.5.2.4 Obtain datation from target

a. The time maintained by the target shall be obtained by reading the Datation field.

5.5.3 Timing action

a. The timing action shall be performed by means of transmission and reception of a SpaceWire time-control codes (Time-Codes).

5.5.3.1 Initialise/Synchronize

a. When a time control code (Time-Code) is received by the target which matches the Time-Code field of the Command register, initialisation or synchronization of the time maintained by the target shall be performed as defined in b hereafter, if:
   1. the New field is set to 1b, and either
   2. the Initialise field is set to 1b, or
   3. the T-Field of the CCSDS Time Code field of the Command register corresponds to the time maintained by the target.

   NOTE 1 The level of correspondence between the T-Field and the time maintained by the target is implementation specific.

b. Initialisation and Synchronization shall:
   1. force the time maintained by the target to be set to the T-Field of the CCSDS Time Code field of the Command register.
   2. force to zero any bits in the time maintained by the target that have a lesser weight than provided in the T-Field of the CCSDS Time Code field of the Command register
   3. force to zero any bits in the time maintained by the target that have a greater weight than provided in the T-Field of the CCSDS Time Code field of the Command register

NOTE 1 Which bits of the T-Field that are actually supported by a target is implementation dependent.

NOTE 2 Bits in the T-Field of the CCSDS Time Code field of the Command register that have a greater weight than in the time maintained by the target can be ignored.

NOTE 3 Bits in the time maintained by the target that have a greater weight than in the T-Field of the CCSDS Time Code field of the Command register are not used.

NOTE 4 Initialisation forces the target to adjust the time it maintains. Synchronization permits the target to ignore adjusting the time it maintains in the case the instant in time of a timing action is outside acceptable tolerances.

c. The New field of the Command register shall be cleared on the reception of a matching time control code (Time-Code).

5.5.4 Time-Stamp

a. Configuration of what Time-Code the target shall time-stamp shall be made by writing the Time-Code field of the Time-Stamp field.
b. When a time control code (Time-Code) is received by the target which matches the Time-Code field of the Time-Stamp register, the time maintained by the target shall be stored in the first CCSDS Time Code field of the Time-Stamp register.
c. When a time control code (Time-Code) is sent by the target which matches the Time-Code field of the Time-Stamp register, the time maintained by the target shall be stored in the second CCSDS Time Code field of the Time-Stamp register.
d. The time-stamp recorded by the target shall be obtained by reading the first CCSDS Time Code field of the Time-Stamp register.
e. The time-stamp recorded by the target shall be obtained by reading the second CCSDS Time Code field of the Time-Stamp field.

NOTE 1 Which bits of the T-Field that are actually supported by a target is implementation dependent.

NOTE 2 The actual procedure for when time control code (Time-Code) are received or transmitted is implementation dependent.

5.5.5 Latency

a. The value by which a target shall correct the time it maintains shall be configured by writing the Latency field.

NOTE 1 Which bits of the T-Field that are actually supported by a target is implementation dependent.

6 ANNEX A (INFORMATIVE) MANAGED PARAMETERS

In order to provide an optimised implementation, some parameters associated with services are handled by management, rather than by inline communications protocol. The managed parameters are those which tend to be static for long periods of time, and whose change generally signifies a major reconfiguration of the service provider associated with a particular mission. Through the use of a management system, management conveys the required information to the service provider.

The managed parameters are listed in Table A-1. These parameters are defined in an abstract sense, and are not intended to imply any particular implementation of a management system.

The values of these parameters are defined for a specific mission implementation.

Table A-1: Managed parameters

Managed parameters	**Allowed value**
CCSDS Time Code formats	CCS, CDS, CUC
CCSDS Time Code resolution	As per CCSDS 301.0-B-4

7 ANNEX B (INFORMATIVE) METHODS FOR TIMING ACCURACY

7.1 Introduction

This document standardizes the protocol for time distribution over SpaceWire, but it does not provide any implementation details.

In this annex some aspects of maintaining highly accurate timing synchronization over SpaceWire networks are therefore briefly discussed.

In [1] aspects of jitter and latency affecting time control codes (Time-Code) in a SpaceWire network are presented, and are further discussed in 7.3 and 7.4.

In [2] a concept of clock state correction and clock rate correction is introduced. This standard provides a means for the former (Timing action is defined in 5.5.3 and is discussed further in 7.2). The latter is implementation dependent and is discussed in 7.5.

7.2 Time correction

Clock state correction as discussed in [2] is supported by this standard by means of the Timing action, where new time information is conveyed from an initiator to a target and the time information is subsequently qualified by means of a time control code (Time-Code).

7.3 Latency measurement and correction

Due to the natural latency of transferring time control codes (Time-Codes) in a SpaceWire network, the clock state correction accuracy discussed in 7.2 will be limited by an offset difference between the initiator and the target (or even between targets). This standard provides a support for measuring the latency for time control code (Time-Codes) transfer in a SpaceWire network.

This standard utilizes the new Distributed Interrupts defined in [3] which use are distributed using similar methods as time-control codes and can therefore be used a means for measuring the propagation delay of the latter. Each target can be configured by the initiator to perform a time-stamp when ever a Distributed Interrupt with a specified value has been sent or received. The time-stamping is done with the time that is maintained by the target.

The initiator performs similar time-stamping it its end and then uses the time-stamps in both ends to calculate the latency or (propagation delay) in either direction. The calculated latency can the be written by the initiator to a specific target register which can be used for correcting the time maintained in the target.

The actual methods for how to send and receive Distributed Interrupts, how to perform the measurements and how to realise the correction are left to the implementers.

7.4 Jitter mitigation

Statistical methods and regulation techniques can be used to mitigate the jitter seen on time control codes (Time-Codes) in a SpaceWire network, as discussed in [1]. Jitter and drift mitigation discussed in 7.5 could be combined.

7.5 Drift mitigation

Clock rate correction as discussed in [2] is not precluded by this standard.

Clock rate correction can be performed based on periodically received time control code (Time-Code) in by target. The mean interval between received time control codes (Time-Codes) could be measured with the local time maintained by the target as reference, and any long term variation could be fed back to the generation of the local time.

7.6 References

[1] *High Accuracy Time Synchronization over SpaceWire Networks - Problem formulation: Jitter and drift of Time-Codes in SpaceWire networks*, Sandi Habinc, SPWCUC-REP-0002, 25th June 2012, Version 1.2, Aeroflex Gaisler AB

[2] *Integration of Internal and External Clock Synchronization by the Combination of Clock-State and Clock-Rate Correction in Fault-Tolerant Distributed Systems*, Hermann Kopetz, Astrit Ademaj, Alexander Hanzlik, Proceedings of the 25th IEEE International Real-Time Systems Symposium (RTSS 2004), 1052-8725/04, 2004

[3] *Space engineering - SpaceWire – Links, nodes, switches and networks,* ECSS-E-ST-50-12C Rev.1 Draft 1

8 BIBLIOGRAPHY

ECSS-ST-S-00	ECSS system - Description, implementation and general requirements
ISO 885-1:1998	Information Technology—8-Bit Single-Byte Coded Graphic Character Sets—Part 1: Latin Alphabet No. 1. International Standard, ISO/IEC 8859-1:1998. Geneva: ISO, 1998.
http://www.spacewire.esa.int	SpaceWire website