

BS EN 16603-50-13:2014



BSI Standards Publication

Space engineering — Interface and communication protocol for MIL-STD-1553B data bus onboard spacecraft

bsi.

...making excellence a habit.
AVA Documents™
Document Sharing Hub

National foreword

This British Standard is the UK implementation of EN 16603-50-13:2014.

The UK participation in its preparation was entrusted to Technical Committee ACE/68, Space systems and operations.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2014.
Published by BSI Standards Limited 2014

ISBN 978 0 580 84190 3

ICS 49.140

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 30 September 2014.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

English version

Space engineering - Interface and communication protocol for MIL-STD-1553B data bus onboard spacecraft

Ingénierie spatiale - Interface et protocole de
communication pour de bus de données embarqué MIL-
STD 1553B

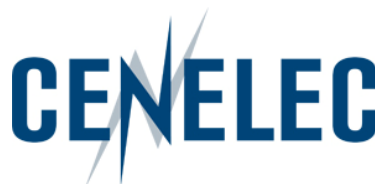
Raumfahrttechnik - Schnittstellen und
Kommunikationsprotokoll für MIL-STD-1553B Datenbusse
in Raumfahrzeugen

This European Standard was approved by CEN on 1 March 2014.

CEN and CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN and CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN and CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CEN and CENELEC members are the national standards bodies and national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.



CEN-CENELEC Management Centre:
Avenue Marnix 17, B-1000 Brussels

Table of contents

Foreword	7
1 Scope.....	8
2 Normative references	10
3 Terms, definitions and abbreviated terms.....	11
3.1 Terms from other standards.....	11
3.2 Terms and definitions to the present standard	11
3.3 Abbreviated terms.....	12
3.4 Conventions.....	13
3.4.1 Bit numbering convention.....	13
3.4.2 Sub-address convention	13
4 Overview.....	14
4.1 Context.....	14
4.2 Approach	14
4.3 Reference architecture	15
4.3.1 Communication devices architecture.....	15
4.3.2 Mapping on CCSDS/SOIS sub-network layer.....	15
4.3.3 Service model	17
4.3.4 1553 bus topology.....	18
4.4 1553 bus scheduling aspects.....	19
4.4.1 Bus profiling and scheduling	19
4.4.2 Bandwidth pre-allocation.....	21
4.4.3 Implementation of the bus profile	23
4.5 Description of services.....	25
4.5.1 Overview.....	25
4.5.2 Time service.....	27
4.5.3 Communication Synchronization service.....	31
4.5.4 Distribution and acquisition: Set and Get Data services	34
4.5.5 Data Block Transfer service	41
4.5.6 Terminal Management services	49

5 Physical Layer requirements	50
5.1 Overview	50
5.2 General.....	52
5.3 Data bus characteristics.....	52
5.4 Terminal characteristics.....	52
5.5 Connectors	53
5.5.1 General	53
5.5.2 Pin allocation for 15-pin.....	54
5.5.3 Pin allocation for remote terminal nominal bus	54
5.6 Transmission method	55
6 Data Link Layer requirements	56
6.1 General.....	56
6.2 Data Words and Messages.....	56
6.2.1 Data word format	56
6.2.2 Messages	58
6.3 Terminal operation.....	59
6.4 Subaddress usage.....	59
6.5 Message retries	59
7 Services definition.....	61
7.1 Time service	61
7.1.1 TimeData primitive	61
7.1.2 TimeSynchronize primitive	62
7.2 Communication Synchronization service	63
7.2.1 CommunicationSynchronize primitive.....	63
7.3 Set Data service	64
7.3.1 SendData primitive.....	64
7.3.2 ReadStatus primitive	66
7.4 Get Data service	67
7.4.1 ReceiveData primitive	67
7.4.2 ReadData primitive.....	69
7.5 Data Block Transfer Service	69
7.5.1 SendData primitive.....	69
8 Protocol specification	71
8.1 Overview	71
8.2 Time service	71
8.2.1 Time Data primitive	71

8.2.2	Time Synchronize primitive	72
8.3	Communication Synchronization service	74
8.3.1	Requirements when the Time Synchronization service is implemented.....	74
8.3.2	Requirements when the Time Synchronization service is not Implemented	75
8.3.3	BC Requirements for Accurate Message Transfer (optional).....	76
8.4	Set Data service	77
8.4.1	BC requirements	77
8.4.2	RT Requirements	77
8.5	Get Data service	77
8.5.1	BC requirements	77
8.5.2	RT requirements	78
8.6	Data Block Transfer service	78
8.6.1	Data Distribution requirements (BC to RT transfer)	78
8.6.2	Data Acquisition requirements (RT to BC transfer).....	83
8.7	Terminal Management services	89
8.7.1	RT monitoring	89
8.7.2	RT Health data word definition	90
8.7.3	Terminal configuration commands	91
8.7.4	Data wrap around	92
9	Test and verification.....	93
9.1	Test specification	93
9.2	Tests traceability.....	93
9.3	Test references.....	93
Annex A	(informative) Tailoring guidelines.....	94
A.1	Scope	94
A.2	Tailoring options and parameters.....	94
A.2.1	Overview	94
A.2.2	Step 1: Function and service selection	94
A.2.3	Step 2: Services configuration.....	94
Annex B	(informative) Unreferenced requirements in MIL-STD-1553B	99
Bibliography		100

Figures

Figure 3-1: Bit numbering convention	13
Figure 4-1: Architecture of typical communication devices with MIL-STD-1553B I/F	16

Figure 4-2: CCSDS/SOIS communication stack architecture.....	16
Figure 4-3: Service model	17
Figure 4-4: 1553 Bus topology.....	18
Figure 4-5: Examples of 1553 bus redundancy scheme	19
Figure 4-6: Process of Bus Profiling	20
Figure 4-7: Example of synchronous access (pre-allocated, populated)	22
Figure 4-8: Examples of asynchronous access (pre-allocated, unpopulated)	22
Figure 4-9: Typical implementation of 1553 messages sequence on BC	23
Figure 4-10: Typical communication frame decomposition on BC	24
Figure 4-11: Services dependencies	26
Figure 4-12: Time Service steps.....	28
Figure 4-13: Time Service	29
Figure 4-14: Time distribution and synchronization.....	30
Figure 4-15: Communication Synchronization scenarios	31
Figure 4-16: Communication Synchronization	32
Figure 4-17: Communication Frame duration adjustment methods.....	33
Figure 4-18: Set Data service	36
Figure 4-19: Get Data service.....	39
Figure 4-20: Data Distribution Transfer, BC to RT.	45
Figure 4-21: Data Distribution timing with Best Effort QoS.....	45
Figure 4-22: Data Distribution timing with Verified Length QoS	46
Figure 4-23: Data Acquisition Transfer, RT to BC.....	47
Figure 4-24: Data Acquisition timing with Best Effort QoS	48
Figure 4-25: Data Acquisition timing with Verified Length QoS	48
Figure 5-1: Bus connectors for separated BCs or RTs	50
Figure 5-2: Bus connectors for integrated BCs or RTs	51
Figure 5-3: Bus connectors for separated BCs or RTs connected to dual buses	51

Tables

Table 5-1: Pin allocation for 15-pin Bus Controller 1553 Bus Connector.....	53
Table 5-2: Pin allocation for 15-pin Remote Terminal 1553 Bus Connector	53
Table 5-3: Pin allocation for 9-pin Bus Controller or Remote Terminal 1553 Bus Connector	54
Table 5-4: Pin allocation for Remote Terminal nominal bus.....	55
Table 6-1: Subaddress allocation	60
Table 7-1: TimeData primitive definition	61
Table 7-2: TimeSynchronize primitive definition.....	62
Table 7-3: CommunicationSynchronize primitive definition.....	63

Table 7-4: SendData primitive definition with predefined transfers (Type I)	64
Table 7-5: SendData primitive definition with pre-allocated bandwidth (Type II)	65
Table 7-6: ReadStatus primitive definition	66
Table 7-7: ReceiveData primitive definition with predefined transfers (Type I).....	67
Table 7-8: ReceiveData primitive definition with pre-allocated bandwidth (Type II).....	68
Table 7-9: ReadData primitive definition	69
Table 7-10: SendData primitive definition for the Data Block Transfer Service	70
Table 8-1: Time Message data words, CCSDS CUC format.....	72
Table 8-2: Communication Frame Synchronization Message Data Word	75
Table 8-3: Layout of the Distribution Transfer Descriptor (BC to RT, SA 27R).....	78
Table 8-4: Layout of the Distribution Transfer Confirmation (BC to RT, SA 27T)	82
Table 8-5: Layout of the Acquisition Transfer Confirmation, ATC (BC to RT, SA 28R)	85
Table 8-6: Layout of the Acquisition Transfer Request, ATR (RT to BC, SA 28T).....	86
Table 8-7: SA 1T: RT_Health & Monitoring data definition.....	90
Table 8-8: RT_Health data definition	91
Table 8-9: SA 1R: Terminal Configuration definition	92
Table 8-10: Reset Services command definition.....	92
 Table A-1 : Requirements selection.....	 95
Table A-2 : Services configuration.....	96

Foreword

This document (EN 16603-50-13:2014) has been prepared by Technical Committee CEN/CLC/TC 5 “Space”, the secretariat of which is held by DIN.

This standard (EN 16603-50-13:2014) originates from ECSS-E-ST-50-13C.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by March 2015, and conflicting national standards shall be withdrawn at the latest by March 2015.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CEN by the European Commission and the European Free Trade Association.

This document has been developed to cover specifically space systems and has therefore precedence over any EN covering the same scope but with a wider domain of applicability (e.g. : aerospace).

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

1

Scope

Using standard communication protocols for spacecraft communication links can provide interface compatibility between communication devices and components. Thus, it can improve the design and development process as well as integration and test activities at all levels, and provide the potential of reusability across projects.

The aim of this space engineering standard is to define the interface services and to specify their corresponding bus protocol elements for spacecraft using the MIL-STD-1553B data bus. It also aims at defining requirements for harmonisation of physical interface and usage of the MIL-STD-1553B data link layer features.

Another goal of this standard is to facilitate the bus profiling task by proposing a message scheduling scheme to the mission system architects. Such framework helps to homogenise the allocation and control of communication resources across a single project or spacecraft mission.

The scope of this standard is as follows:

- It details the usage of the MIL-STD-1553B.
- It covers the communication protocols, services and functions needed for exchange of information over MIL-STD-1553B data bus.
- It is limited to necessary and sufficient requirements to ensure compatibility for communication through MIL-STD-1553B data bus for communication devices onboard a spacecraft and across projects.
- It covers a wide spectrum of mission needs.
- It does not modify requirements that are under the scope of MIL-STD-1553B.
- It covers recommendation for verification and test of communication devices communicating through a MIL-STD-1553 data bus.

This Standard provides a comprehensive set of requirements for all communication devices and components onboard a spacecraft, which are connected to a single (redundant) data bus according to MIL-STD-1553B.

Although the standard focuses on the specification of single-bus architecture, questions related to multiple-bus-architectures or the use of repeaters for separable busses (for launchers) are also addressed.

This Standard aims at specifying requirements that are technically feasible, correct, consistent and compliant with the needs and overall technological approach and industrial policies of the participating Agencies and Industry.

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

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.

EN reference	Reference in text	Title
EN 16601-00-01	ECSS-S-ST-00-01	ECSS System - Glossary of terms.
	MIL-STD-1553B	Interface Standard for Digital Time Division Command/Response Multiplex Data Bus, Notice 2, 8 th September 1986 Notice 3, 31 st January 1993 Notice 4, 15 th January 1996
	MIL-HDBK-1553A	Military handbook. Multiplex applications handbook, 1 st November 1988

NOTE The technical requirements and their numbering are identical in Notices 2, 3 and 4 of the MIL-STD-1553B standard. Therefore, ECSS-E-ST-50-13 can be used in complement to any of those MIL-STD-1553B standard notices.

3

Terms, definitions and abbreviated terms

3.1 Terms from other standards

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

3.2 Terms and definitions to the present standard

3.2.1 1553 message

data-link layer exchange as defined in clause 3.7 of MIL-STD-1553B

3.2.2 communication device

equipment or component connected on a MIL-STD-1553B bus and providing communication functions as a Bus Controller, Remote Terminal or Bus Monitor

NOTE Communication devices are building blocks of the 1553 reference architecture described in clause 4.3.1.

3.2.3 communication frame (CF)

time interval delimited by two Communication Frame Synchronization messages

NOTE The CF usage is described in clauses 4.4.2.2 and 4.5.3 (informative) and specified in clause 8.3, Communication Synchronization service (normative).

3.2.4 data bus system

bus network and communication devices which forms an autonomous whole capable of performing information exchange

3.2.5 message

See 1553 message, 3.2.1

3.2.6 time synchronization cycle (TSC)

time interval delimited by two Time Synchronization Messages

NOTE The Time Synchronization Message usage is described in the informative clause 4.5.2 and defined by clause 8.2.2, Time Synchronize.

3.3 Abbreviated terms

The following abbreviations are defined and used within this Standard:

Abbreviation	Meaning
ADC	acquisition data counter
AOCS	attitude and orbit control system
API	application programming interface
ATC	acquisition transfer confirmation
ATR	acquisition transfer request
BC	bus controller
BM	bus monitor
CCSDS	Consultative Committee for Space Data Systems
CF	communication frame
CUC	CCSDS unsegmented time code
CW	command word
DDC	distribution data counter
DHS	data handling system
DLL	data link layer
DTD	distribution transfer descriptor
DW	data word
ECSS	European Cooperation for Space Standardization
FDIR	failure detection, isolation and recovery
GND	ground
GNSS	global navigation satellite system
H/W	hardware
ID	identification
I/F	interface
I/O	input/output
ISO	international standards organization
LSB	least significant bit
MSB	most significant bit
N	nominal
N/A	not applicable
NC	not connected
OBC	on board computer
OBT	on board time
PDU	protocol data unit
PUS	packet utilization standard

R	redundant
RD	reference document
RT	remote terminal
SA	sub-address
SDBP	spacecraft data bus protocol
SOIS	spacecraft onboard interface services (from CCSDS)
SYNCH	data bus system synchronization signal
T	data bus termination
TBC	to be confirmed
TBD	to be defined
TBW	to be written
TC	telecommand
TM	telemetry
TM/TC	telemetry/ telecommand
TSC	time synchronization cycle
UML	unified modelling language

3.4 Conventions

3.4.1 Bit numbering convention

The most significant bit of an n-bit field is:

- numbered bit 0 (zero),
- the first bit transmitted, and
- the leftmost bit on a format diagram.

The least significant bit of an n-bit field is:

- numbered bit n-1,
- the last bit transmitted, and
- the rightmost bit on a format diagram.

This convention is illustrated in Figure 3-1.

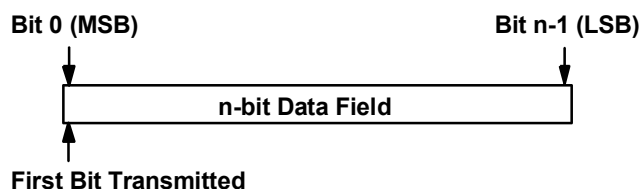


Figure 3-1: Bit numbering convention

3.4.2 Sub-address convention

SA nT stands for Sub-address #n, Transmit, while **SA nR** stands for Sub-address #n, Receive.

4

Overview

4.1 Context

The **MIL-STD-1553B** notice 2 standard is used for many spacecraft onboard data links including commercial and scientific satellites, space exploration vehicles, launchers and in-orbit manned flight applications. Physical and data link layers requirements provided by the **MIL-STD-1553B** allow defining and implementing coherent on-board data bus system capable of exchanging small data messages up to 512 bits. However, no requirement exists within this standard to handle larger data structures or more complex control of data flows in order to support higher level communication and synchronization services.

Analysis of the implemented on-board data bus systems and communication devices used onboard of spacecraft shows that basic communication and synchronization services, as well as support to the terminal management function can be defined in a common way. At the same time sufficient flexibility remains for tailoring these services to the application specific performance requirements. In this context, the definition of a common communication and synchronization services framework allows for a better capability to reuse communication devices on several future missions as well as providing a common reference for data bus system development and verification.

4.2 Approach

The approach of the ECSS working group for defining this standard aims at identification of layers, services and functions of the typical communication devices with 1553 I/F from the experience gained in various space projects including science, telecommunications, space exploration and space transportation provided by the working group participants. In particular this approach aims at:

- Identifying Reference Architectures (Layers, Services, Functions and Elements of protocol) of typical communication devices with MIL-STD-1553 I/F;
- Characterizing Services, Functions and Elements of Protocol of each Layer within identified Reference Architectures, using concrete project specifications;
- Selecting candidate items within each Layer for standardization, proposing solutions and identifying requirements for RT and BC type MIL-STD-1553 communication devices.

- Define normative requirements rather than recommendations.

NOTE As a lesson learned from many project experiences, this aims at making sure that deviations on a given mission are driven by mission requirements and are tracked at specification time rather than being discovered at integration time which results in delays and generally induce large costs impacts.
- Enabling the development of spacecraft onboard communication services compatible with the CCSDS/SOIS requirements.

As far as possible, the defined communication requirements are extracted from the experience on existing spacecraft specifications.

4.3 Reference architecture

4.3.1 Communication devices architecture

There are three types of communication devices in **MIL-STD-1553B** data bus systems: Bus Controllers (BC), Remote Terminals (RT) and Bus Monitors (BM). BM devices are not subject for any additional requirements within this standard but their usage for integration and test is recommended.

The Figure 4-1 depicts the reference architecture of typical BC and RT type communication devices with 1553 I/F. Only OSI protocol stack related elements of the communication devices are shown in this view.

This can entail units with an existing 1553 I/F design (legacy units), as well as enhanced 1553 communication devices. These units can involve usage of different services, which can be supported by specific driver software.

The communication devices exchange their data through a time-multiplexed wire-harness medium, which topology is further detailed in clause 4.3.4.

4.3.2 Mapping on CCSDS/SOIS sub-network layer

This Standard provides the definition of a set of services that can be mapped on the Sub-network layer of the CCSDS/Spacecraft Onboard Interface Services communication stack as described in Figure 4-2.

The services, as defined in this standard, are directly compatible to the Packet Service, Memory Access Service, Time Distribution Service, Device Discovery Service and Test Service of the SOIS reference model. For some of these services, additional SOIS-specific control information can be needed. The definition of this control information, as well as the capabilities and requirements of the Data Link Convergence Functions, is outside the scope of this standard.

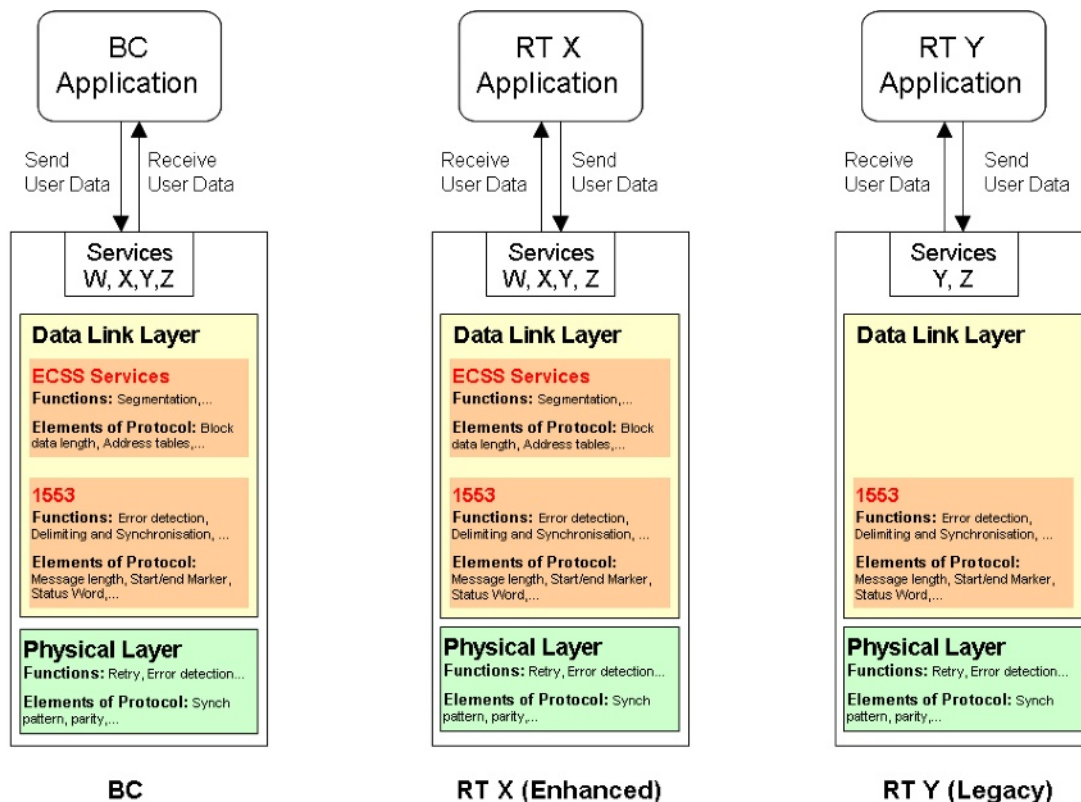


Figure 4-1: Architecture of typical communication devices with MIL-STD-1553B I/F

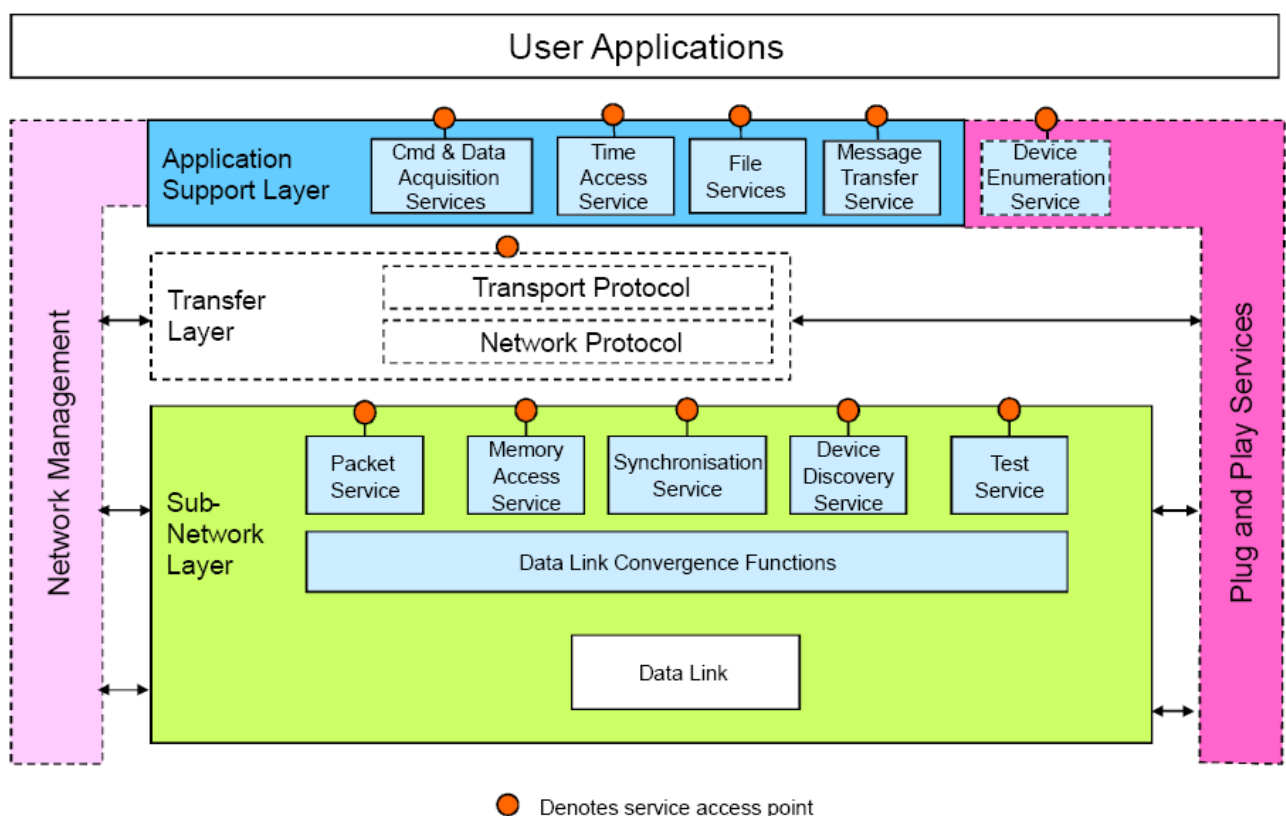


Figure 4-2: CCSDS/SOIS communication stack architecture

4.3.3 Service model

This clause describes a general model which is used for the definition of services in this standard. Figure 4-3 depicts this general service model.

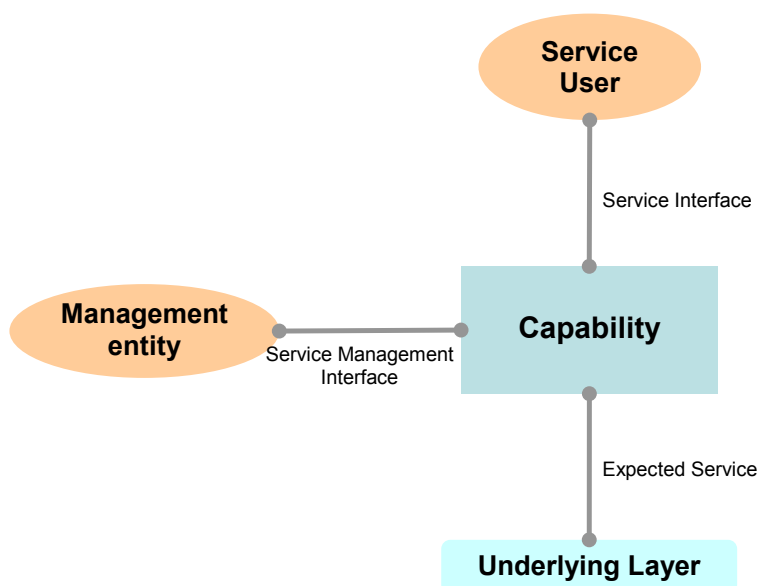


Figure 4-3: Service model

The services defined in this standard describe communication capabilities of service-provider which is provided to service-user at the boundary between service-provider and service-user.. This capability in its turn relies on certain services expected to be provided by the underlying layer. The interactions between the service-user and service provider constitute “abstract interface” at the service boundary. This abstract interface is defined in terms of service primitives which the service-user and the service-provider are allowed to exchange, together with sequencing rules which apply to these exchanges. The UML sequence diagram convention is used throughout this standard to illustrate how sequences of interactions are related in time. The OSI convention and terminology is used to represent service-primitives on those UML sequence diagrams. Services are configured and managed by a service management entity. Service primitives are described through their name (e.g. SendData, ReadStatus), type (e.g. Submit, Deliver or Requestor.Submit, Acceptor.Deliver) and list of parameters (e.g. GroupID, DataLength)

Service primitive is represented by an arrow.

- Arrows located at the left of the vertical line indicate requestor primitives
- Arrows located at the right of the vertical line indicate acceptor primitives
- Arrows directed towards the vertical line indicate submit primitives
- Arrows directed out of the vertical line indicate deliver primitives

Primitives are an abstract, atomic representation of an interaction between service-user and service-provider which are defined in clause 7 of this standard. The primitives and their parameters indicate what information flows between the service-user and the service-provider at their interface. This is not to be confused with an API which is associated with a specific implementation.

There are three types of parameters associated with each definition of a service primitive – Input, Output, and Managed. The CCSDS Draft Recommendation for the encapsulation service [CCSDS-133.1-R-1, Red Book, April 2002], provides the following definition for managed parameters: “In order to conserve bandwidth on the space link, some parameters associated with a Service 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 defined in an abstract sense, and are not intended to imply any particular implementation of a management system”. They allow configuring services according to specific application needs or options relevant to implementation. They are not defined within this standard but examples are provided in this informative clause.

4.3.4 1553 bus topology

This clause presents the bus topology used in the reference architecture (see Figure 4-4).

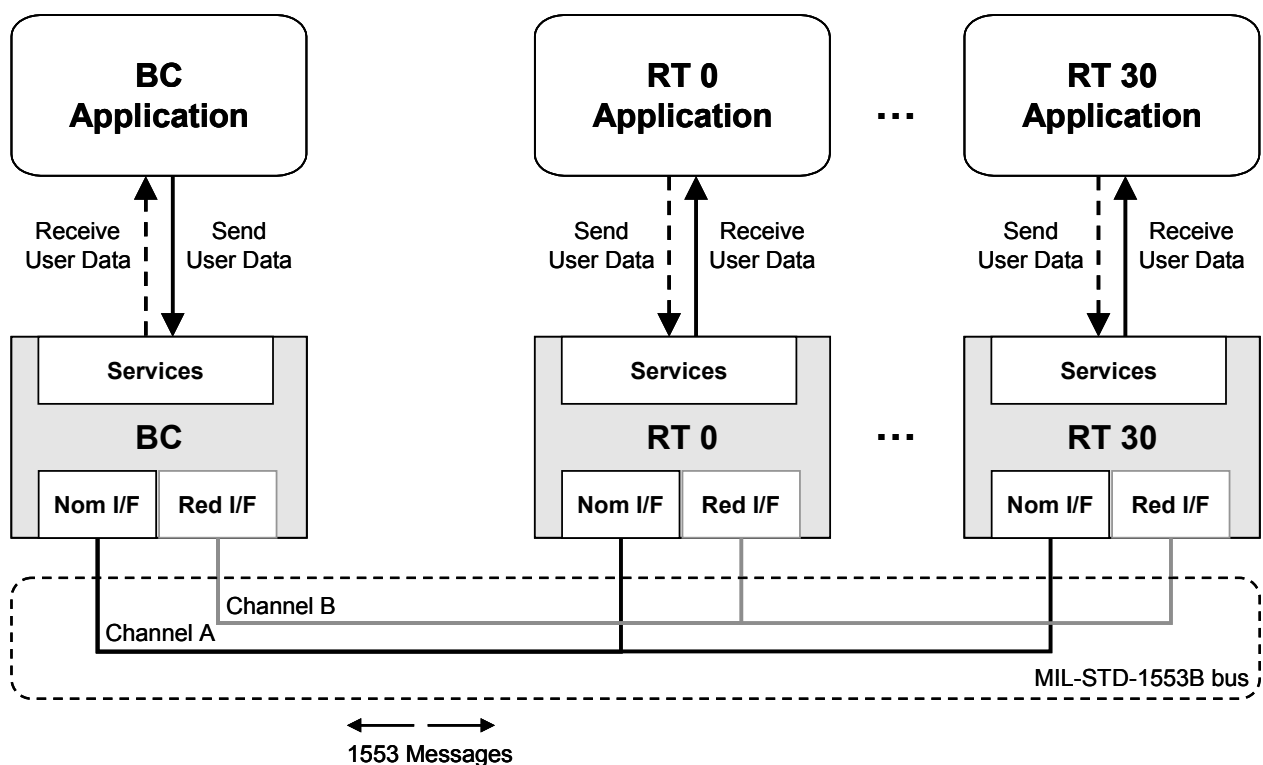


Figure 4-4: 1553 Bus topology

As defined in MIL-STD-1553B, the 1553 network connects the Bus Controller (master of the bus) with up to 31 Remote Terminals (bus slaves).

Bus management is strictly based on master-slave relationship between Bus Controller and the Remote Terminals.

The MIL-STD-1553B bus medium is composed of a nominal (Channel A) and a redundant (Channel B) physical interconnection.

Messages can be sent either on channel A or on channel B by the BC, and the RT replies always on the same channel. Here and in the other clauses, unless differently stated, the term **1553 bus** includes nominal and redundant channels.

The management of the redundancy of channels and communication devices in a data bus system is out of the scope of this standard. However elements of this standard to be used by the user application to perform Failure Detection, Isolation, and Recovery (FDIR) tasks are identified within this Overview clause and highlighted in the clause 7.

In Figure 4-5, some examples of commonly used redundancy schemes are presented; alternate bus topologies to handle application specific fault tolerance or layout requirements are considered compliant to this standard as long as they are compliant to **MIL-STD-1553B**.

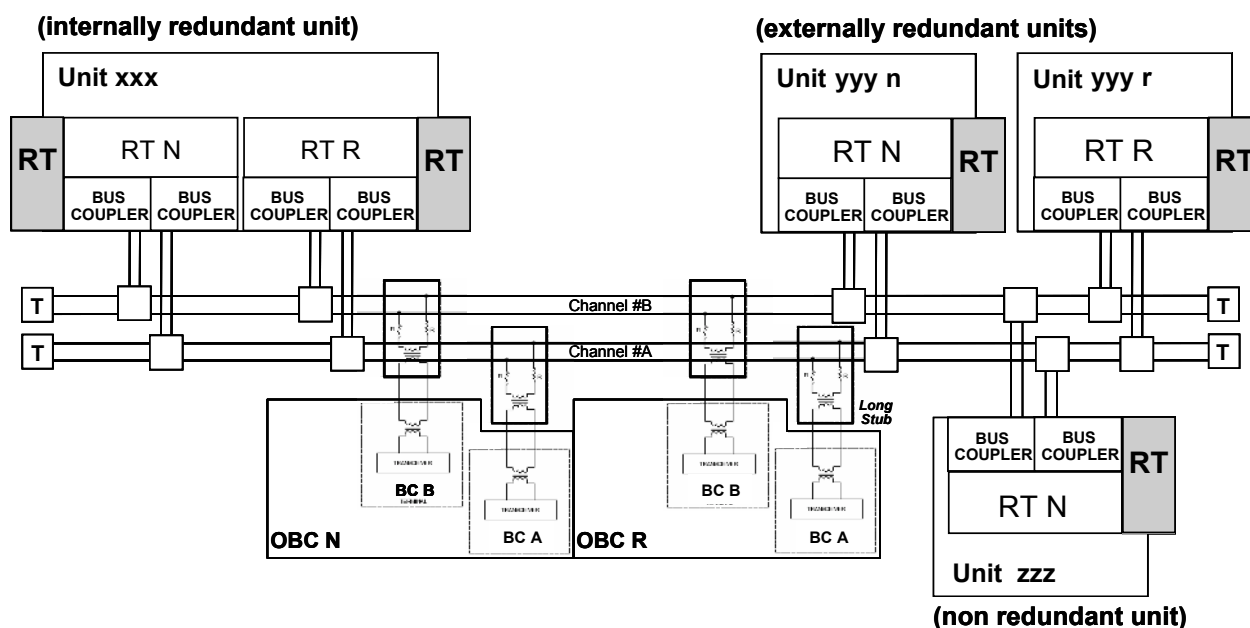


Figure 4-5: Examples of 1553 bus redundancy scheme

4.4 1553 bus scheduling aspects

4.4.1 Bus profiling and scheduling

A **MIL-STD-1553B** Bus is a resource shared between different users (BC and several RT) having different data routing, timing, etc. needs and constraints. Due to the limitations of the medium, it is mandatory to use the data bus in compliance with real-time systems requirements.

In particular, for spacecraft control, the data bus usage needs to be deterministic at any time. Because of that an important aspect in the management of the **MIL-STD-1553B** bus is the optimization of bandwidth allocation to the users in order to satisfy this overall system need. This operation, known as bus profiling, is often performed during the definition of the spacecraft-level system specifications and foresees the allocation of dedicated bandwidth for each activity on the data bus.

The user communication needs differ depending on the type of applications: some users require processing cycles including sequences of acquisition,

processing and commanding, while others require only sporadic command distribution for instance. This has resulted in various definitions for bus profiling terminologies such as *major frame*, *minor frames*, *sub-frames*, *slots* and so on. The following proposed model preserves these user concepts with a common definition of the basic communication elements which are the Time Synchronization Cycle and the Communication Frame.

The bandwidth allocation is commonly organised as Time Synchronization Cycles split into Communication Frames. The frames are synchronous with the on-board time and/or the bus controller software cycles. As result of this, one or more alternative bus profiles (depending on different operational modes/phases) are created and can be applied on the actual system taking advantage of the autonomous or semi-autonomous features commonly supported by the Bus Controller devices.

Profiles can be specified only if users' needs and constraints are represented by homogeneous and comparable attributes. It is possible to describe, build models and verify these needs in terms of requirements as early as possible in the spacecraft system design phase. These requirements put together describe a feasible system and present no inconsistency before the implementation at equipment level starts. As a result, the data bus system can be qualified through verification activities at individual communication devices, resulting in a working communication at the data bus system level.

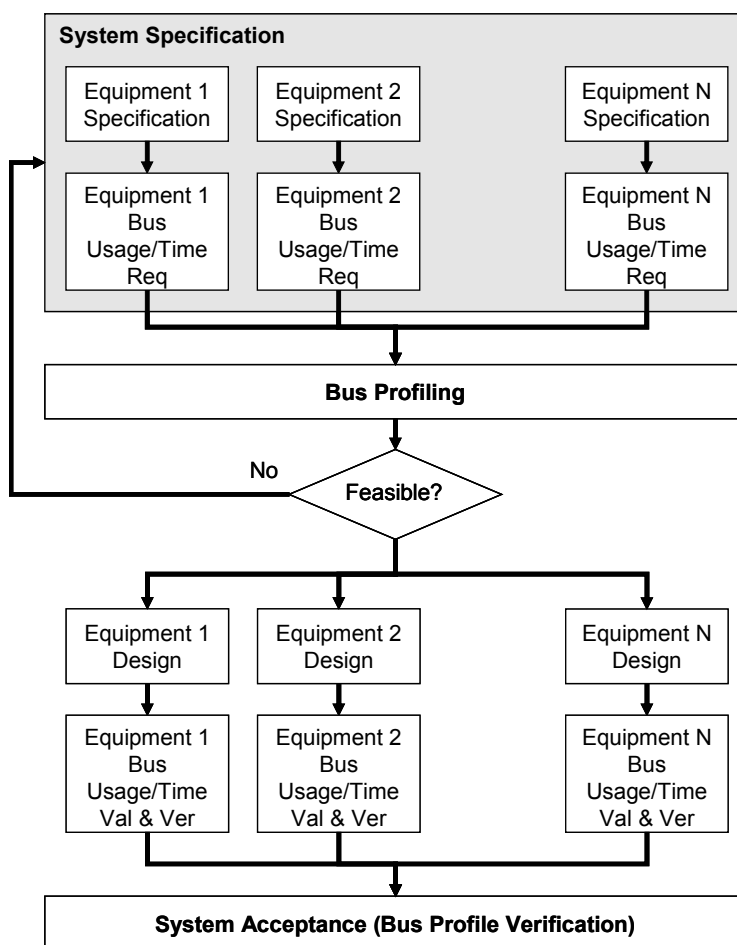


Figure 4-6: Process of Bus Profiling

In Figure 4-6, the bus profiling process is presented within the general process involving the spacecraft system specification definition and the system validation & verification. The term equipment is used to identify communication devices BC and RTs (including commercial-off-the-shelf and legacy products, where the specifications are taken “as is”).

4.4.2 Bandwidth pre-allocation

4.4.2.1 Service access type

Associated with each instance of data transfer are certain latencies, which are consequences of multiplexing data transmissions on a single data bus.

Depending on the desired latency and frequency requested it is foreseen to implement the service instances in two ways:

- Access via pre-allocated bandwidth with populated content

Typically, services requiring low latency or “high” frequency (for example AOCS command and data acquisition, critical command distribution, cyclic housekeeping data acquisition) have the bandwidth pre-allocated using the populated content option: the time position of each 1553 messages in the timeline of the Communication Frame is always reserved to a given service with given attributes (e.g. RT address, maximum data size). In this case latency is defined a priori by the scheduling scheme itself (see example on Figure 4-7).

- Access via pre-allocated bandwidth with unpopulated content

Sporadic/asynchronous or less critical service instances in terms of latency can “share” the pre-allocated but unpopulated remaining bandwidth (for example routing of sporadic telecommands from ground to an onboard application in an RT) unpopulated: access to the bus is performed in one of the pre-allocated time intervals where no specific 1553 message is scheduled for being transferred (unpopulated content). In this case only worst case execution latency is defined by the scheduling scheme; actual bus access latency is defined at run-time.

As shown in Figure 4-8, depending on the previous asynchronous requests, the “service Y request” can be executed in the first or the second unpopulated position (in grey) of Communication Frame #N. The user has knowledge a priori of the upper limit latency (worst case, request served in the second position), but the actual latency can only be measured at run-time depending on when the request is actually scheduled.

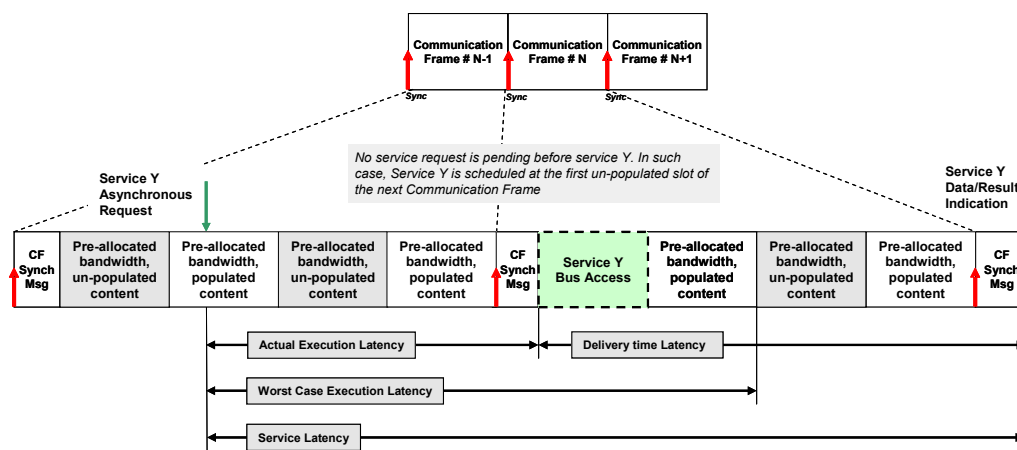


Figure 4-7: Example of synchronous access (pre-allocated, populated)

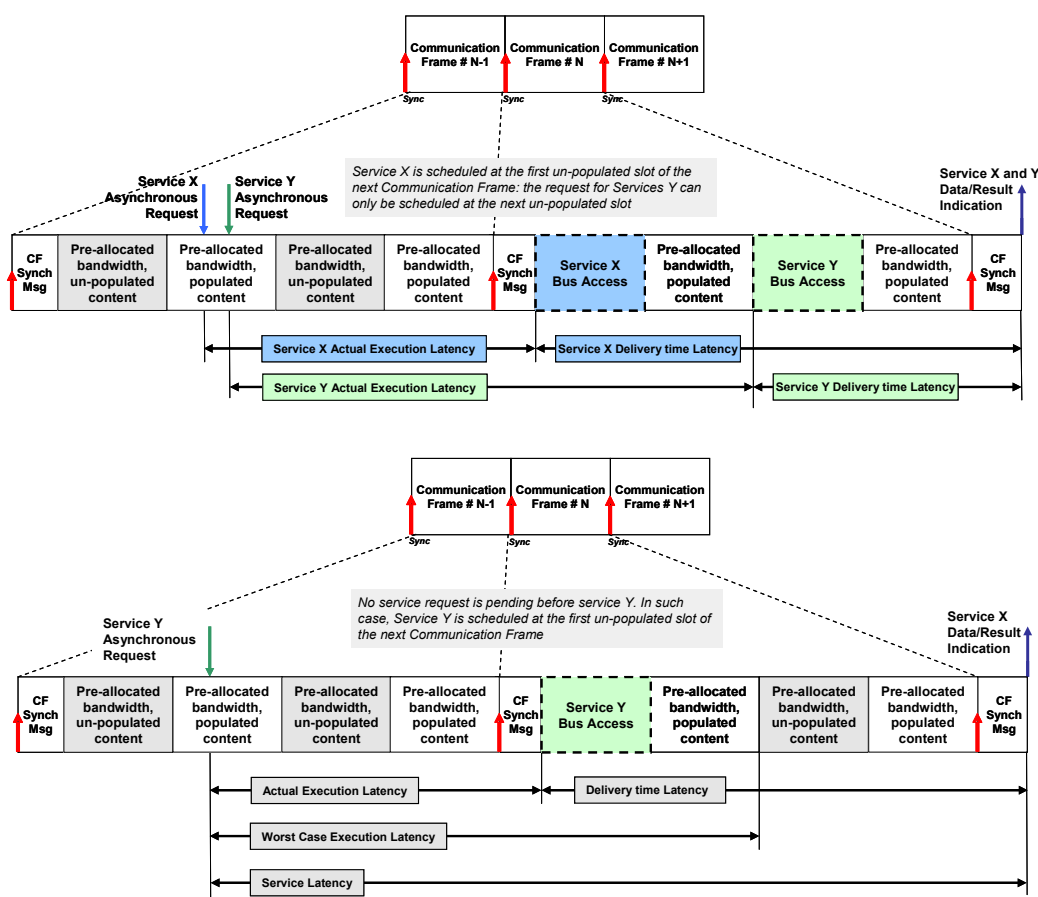


Figure 4-8: Examples of asynchronous access (pre-allocated, unpopulated)

4.4.2.2 Service latencies

In the cases described in 4.4.2.1 the requested latency is based on a priori knowledge by the user of the service(s) made available to it by the Data Link Layer. Thus a user not only has knowledge of the parties with which it can communicate, but also has explicit knowledge of the characteristics of the service it can expect to be provided with each invocation of the service.

Latency can be classified as follows:

- Service Execution Latency: the time between the issuing of the service request and the start of the actual data transfer on the bus;
- Delivery Time Latency: the time between the start of the data transfer on the bus and the time when the user is notified with the result of the service request (e.g. data available, error results);
- Service Latency: the time between the issuing of the service request and the time when the user is notified with the result of the request.

According to the needs of the system designer, different Communication Frames can be adopted during different phases of the execution of the bus data transfers: the only strong requirement is to have the Communication Frame established at runtime, including all data to be transferred, before its execution is scheduled on the bus.

4.4.3 Implementation of the bus profile

Bus profiling benefits from autonomous functionality offered by modern Bus Controller communication devices. Typically, BC devices allow to be programmed setting up a list of messages to be scheduled in sequence or according to a given time delay (optional) known as “send-list” or “message-sequence” (see Figure 4-9).

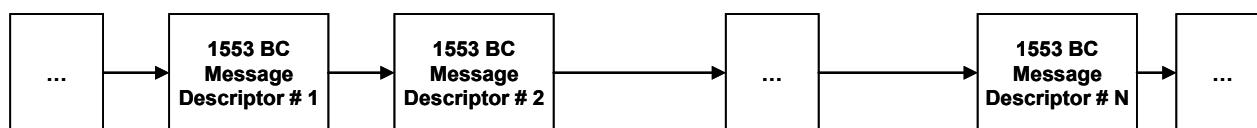


Figure 4-9: Typical implementation of 1553 messages sequence on BC

The host computer housing the BC device is in charge only to prepare the data to be delivered, to trigger the start of the sequence and to retrieve the returned data at the end of the sequence (typically notified by an interrupt) or handling error conditions. The BC device autonomously performs the low level operations of sending and receiving the elementary 1553 messages with respect to the programmed sequence (see Figure 4-10). In this context a bus profile can be efficiently implemented by programming the message sequence in the bus controller at initialization time and managed at run-time by setting up and retrieving data to be transferred on the bus.

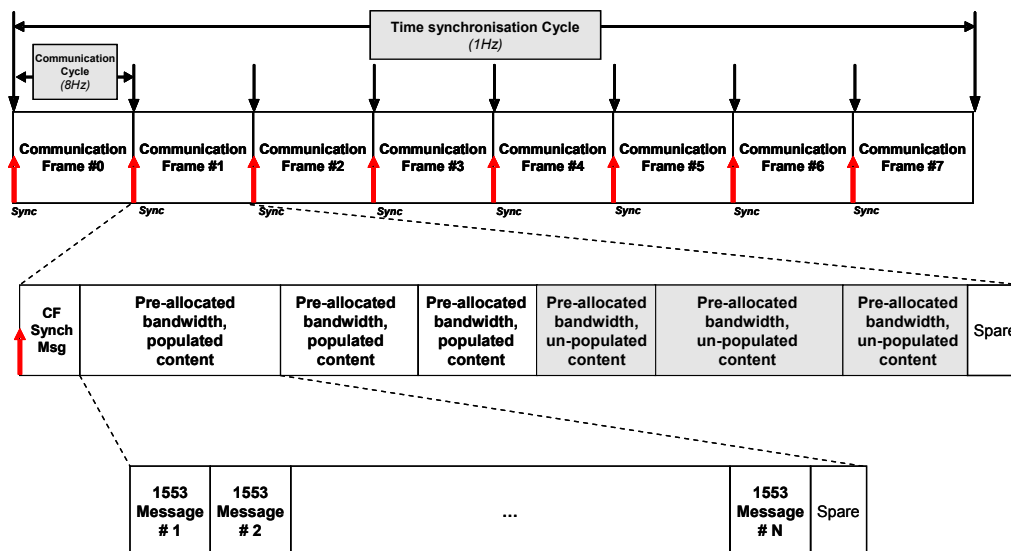


Figure 4-10: Typical communication frame decomposition on BC

If different operative modes or mission phases are foreseen, alternative profiles can be implemented by different and independent message sequences to be programmed and started on the BC according to the mission needs. As a basic principle, however, it is not recommended to modify a bus profile ad hoc in an adaptive, demand-driven way. As a result, data throughput and worst-case latencies can be guaranteed for each individual user, which is connected to the data bus - irrespective of the nominal or non-nominal behaviour of any other remote terminal.

4.5 Description of services

4.5.1 Overview

This clause provides an overview of the services defined in this standard within the data-link layer.

These services are:

- **Time** service which supports the distribution of time information across a 1553 data bus.
- **Communication Synchronization** service which supports time multiplexing of data bus messages in a deterministic way.
- **Set Data** and **Get Data** services which support non-confirmed data transfers of restricted length with a simple protocol without handshake.
- **Data Block Transfer** service which supports confirmed transfers of data blocks on request of the sender with handshaking supported by a protocol.
- **Terminal Management** services which provide standard functions and structured data in support to the implementation of project specific terminal management.

NOTE This clause provides informal description of the services. The reader is referred to clause 7. of this standard (normative requirements chapter) for the detailed specification of the above defined services, as well as for the definition of message formats.

The services are not fully independent. The “Communication Synchronization” service is a mandatory general service which aims at guaranteeing real-time properties and which is used by all other services. The “Terminal Management” service relies on the “Get/Set service”, which in turn relies on the “Communication Synchronization” service. “Time”, “Get/Set” and, “Data Block Transfer” services are independent from each other but all rely on the “Communication Synchronization” service. The dependency graph is depicted in Figure 4-11.

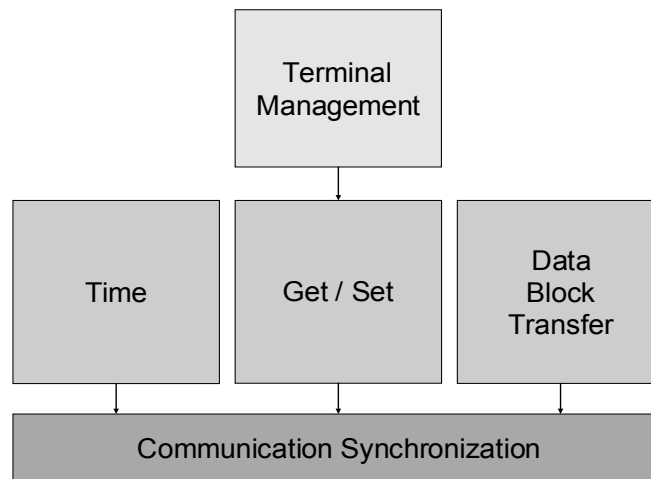


Figure 4-11: Services dependencies

The services primitives provide reporting on the service completion with respect to the service capabilities rather than a detailed view of the 1553 bus activity. No detailed view on the 1553 message transfer is provided which can be viewed as a restriction for failure investigation. However, experience has demonstrated that direct analysis of communication errors for FDIR purpose is typically what has failed in many projects because of the additional SW involved and its impact on performance generating problems in real-time operations. Therefore, this standard considers that specific details for offline FDIR analysis can be acquired through specific FDIR services which can be project specific or be specified in another standard.

4.5.2 Time service

The Time service supports distribution of reference time across a 1553 data bus from a central On-Board Time Master located at the BC side of the bus to one or more On-Board Time Slaves located on the RT's sides. The distribution is done as a periodic process where two service primitives – TimeData and TimeSynchronize used in steps, as shown in Figure 4-12, are continuously repeated. The two service primitives can also be used separately in case one of them is not selected for use in a service-user (i.e. an application.)

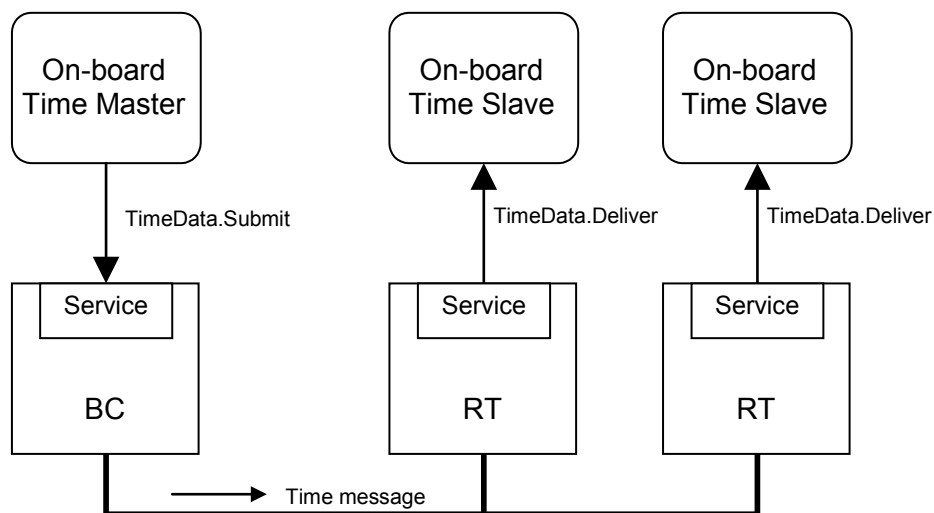
- Step 1:

A TimeData.Submit primitive is invoked by the On-Board Time Master to request service-provider at BC to distribute Time Message on a bus to all On-Board Time Slaves. This request contains the time that is valid now or at the next Time Synchronization Message in case if TimeSynchronize primitive is also used. The request results in a Time Message being sent on the data bus in broadcast mode or separately to all RTs, which deliver the received time to their local On-Board Time Slaves by invoking TimeData.Deliver primitive. Each On-Board Time Slave saves the received time for use in the next step. The time format is normalized by this standard.

- Step 2:

When the next Time Synchronization Event occurs, the On-Board Time Master invokes Time Synchronize.Submit primitive. The request results in a Time Synchronization Message in the form of a Synchronize mode command being sent on the data bus in broadcast mode to all RTs or in the form of a separate signal on a dedicated line. The service-provider at RT invokes the TimeSynchronize.Deliver primitive to notify its service-user - local On-Board Time Slave - on occurrence of Time Synchronization Event. An On-Board Time Slave can then set its local copy of on-board time directly at the event or perform other algorithms to ensure that its local copy of onboard time is synchronous with the reference time.

Step 1:



Step 2:

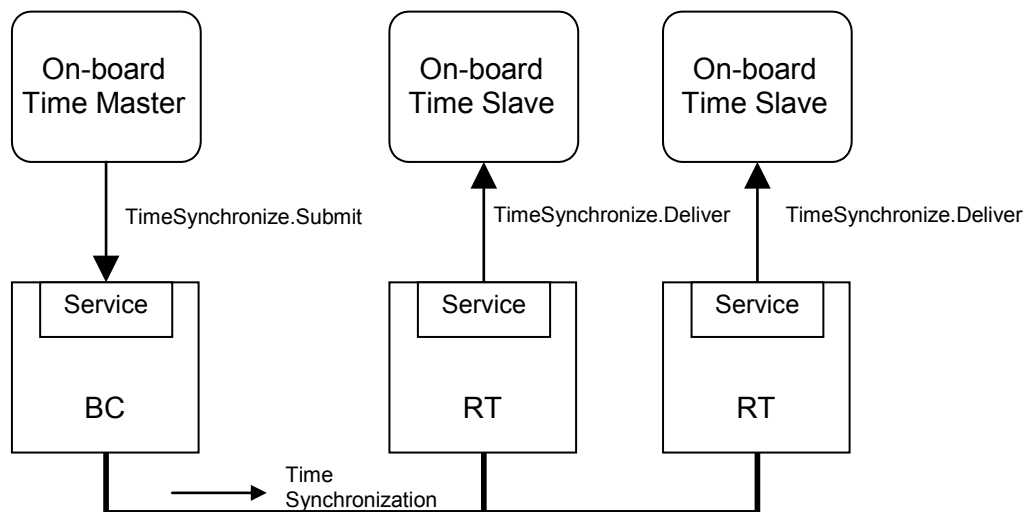


Figure 4-12: Time Service steps

In case of usage of the TimeSynchronize primitive, the process is started by the BC first sending a Time Message followed by the first Time Synchronization Message. As RTs can be powered at any point in time the RT applications are able to tolerate any order of appearance of two messages.

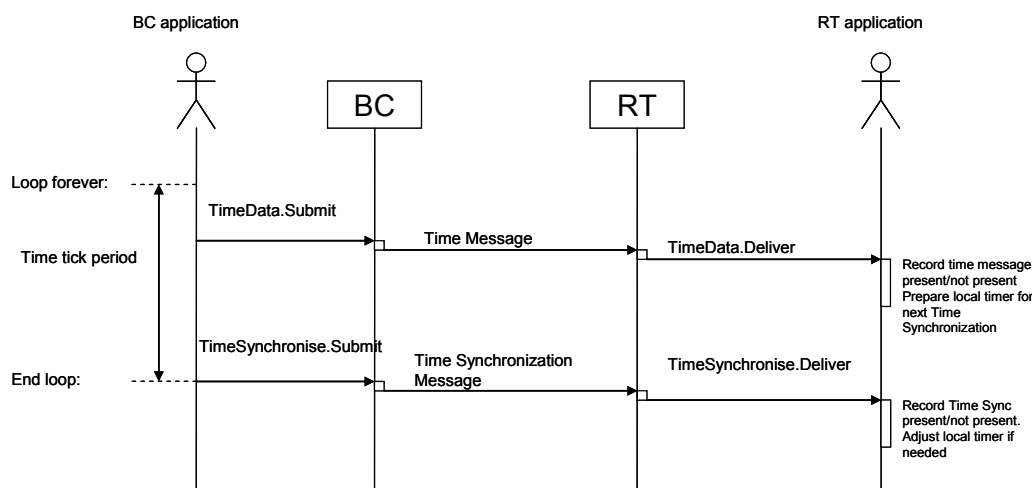


Figure 4-13: Time Service

The occurrence of a `TimeSynchronise.Submit` primitive is controlled by the On-Board Time Master. It is generated with a frequency that is naturally occurring in the time counter and that is extensively used in the system. For the 1553 data bus reference system described in this document a 1 second Time Tick Period is used.

As the On-Board Time Master can be both free-running and being synchronized to, for instance, a GNSS receiver there can be occasions where the OBT Master performs a synchronization operation. Two methods are allowed:

1. The OBT Master adjusts its frequency by speeding up or slowing down such that it within a reasonable time ends up in phase with the GNSS system. Typically the adjustment is made until the second ticks coincide and thereafter any coarse OBT setting can take place to keep the spacecraft exactly in phase with the GNSS system.
2. The OBT Master locks its frequency to the GNSS clock frequency but keeps running with a fixed offset relative to the GNSS system. This offset is then added to the Master OBT time when the time is sent on the 1553 bus. With this approach the subsecond part of the distributed time is typically not be zero when the Master OBT is in full sync.

The two primitives of this service can also be used separately. For applications requiring very accurate time synchronization, the Time Synchronization Message can be sent using a dedicated line (i.e. not being a 1553 Message). For applications requiring very coarse synchronization, usage of the `TimeSynchronise` primitive can be completely omitted. The accuracy also affects the resolution used in the Time Message, using the full resolution only makes sense if the Time Synchronization Message is distributed via a separate line.

For applications not requiring time distribution, such as simple cyclic control applications, the `TimeData` primitive can be omitted.

To allow RT application to correlate the two service primitives the BC has to provide some time for the RT to process a Time Message before the Time Synchronization Message is sent. The BC cannot send a new Time Message too

soon after the Time Synchronization Message to allow the RT to process the Time Synchronization Message without having its time value changed. In Figure 4-14, these relations are shown as setup and hold times relative to the Time Synchronization Message.

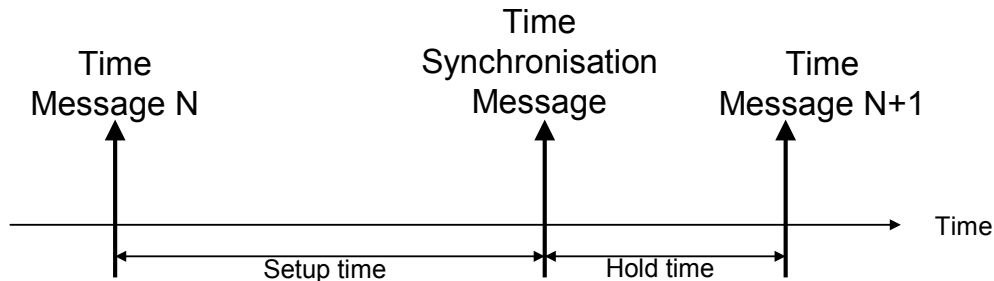


Figure 4-14: Time distribution and synchronization

An RT application that detects a missing Time Message or a missing Time Synchronization Message can report the condition using the RT Health monitoring function if it considers that the missing message can cause the RT to be unsynchronized.

Primitives:

*** TimeData**

This primitive is used by On-Board Time Master on the BC to distribute Time Message to all On-Board Time Slaves on the RT's.

*** TimeSynchronize**

This primitive is used by On-Board Time Master on the BC to distribute Time Synchronization Message to all On-Board Time Slaves on the RT's