SwissCube Flight Software Architecture

Prepared by:

Benoit Cosandier
Florian George
Ted Choueiri

Checked by:

Approved by:

•
HE-Arc
Le Locle
Switzerland
•
20 August 2007
•

SpAce Center

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

Record of revisions

ISS/REV	Date	Modifications	Created/modified by
1/0	20/08/2007	Initial issue	Benoit Cosandier Florian George

1 ABBREVIATED TERMS AND PREFIXES

1.1 Abbreviated terms

A	Analogical or Ampere
ACK	Acknowledgment
ACKST	Acknowledgement Store
ADC	Attitude Determination & Control
ADCS	Attitude Determination & Control System
ADS	Antenna Deployment System
APID	Application Process ID
AR	Archiving
BAT	Battery
BC	Beacon
BEAC	Beacon
QUALIF	Qualification
CB	Connection Board
CDMS	Command & Data Management System
COM	Communication
COUNT	Counter
CU	Current
D	Digital
DANG	Danger
DI	Disable
EC	Error Code
ED	Enable/Disable
EF	Error Flag
EN	Enable
ENUM	Enumeration
EPS	Electrical Power System
EXT	External
FCT	Function
FRE	Frequency
GYRO	gyroscope
HK	Housekeeping
HKST	Housekeeping Store
HW	hardware
INFO	Information
LR	Last Report
MAGN	Magnetometer
MAX	Maximum
MB	Mother Board
MC	Microcontroller
PBUS	Power Bus
PCB	Printed Circuit Board

PFC	Parameter Format Code
PKT	Packet
PL	Payload
PLST	Payload Store
PTC	Parameter Type Code
PWR	Power
REC	Recovery
RT	Real Time
S3	SwissCube
SC	Spacecraft
SCHE	Scheduler
SCID	Spacecraft ID
SID	Structure ID
SS	Sun Sensor
ST	Status
STORE	Storage
SW	Software
T_OUT	Time Out
TAB	Table
TBC	To Be Checked
TBD	To Be Defined
TLE	Two Line Elements (elements provides by NASA/NORAD)
TP	Temperature
TX	Transmitter
V	Volt
VC	Virtual Channel
VOLT	Voltage
WARN	Warning
WD	Watchdog

1.2 Prefixes

APID_	Application ID
EC_	Error Code
ENUM_	Enumeration
FCT_	Function
HK_	Housekeeping
SC_	Spacecraft
SC_MODE_	Spacecraft Mode
SC_SUBSYSTEM_	Spacecraft Subsystem
SC_VC_	Spacecraft Virtual Channel
SCID_	Spacecraft ID
SID_	Structure ID
STORE_	Storage

2 REFERENCES

2.1 Normative references

[N1]	ECSS-E-70-41A – 30/01/2003
[N2]	S3-B-ICD-1-3-TMTC Packet Definition – 20/08/2007
[N3]	S3-BC-SE-1-2-I2C Internal Protocol – 27/08/2007
[N4]	S3-BC-FS-1-0-User Manual: Functions – 20/08/2007
[N5]	S3-BC-FS-1-0-User Manual: Housekeeping – 20/08/2007
[N6]	S3-BC-SE-1-1-AX.25 Transfer Frames Format – 19/03/2007
[N7]	S3-C-COM-SE-1-1 (Telecommunications System Engineering) – 20/06/2007
[N8]	S3-BC-COM-1-2-Link analysis RF telecom – 03/07/2007
[N9]	S3-B-2-3-Level 5 Beacon specifications – 11/07/2007
[N10]	S3-B-COM-1-6-Beacon message format
[N11]	S3-BC-SE-I2C Internal Protocol – 18/07/2007

3 INTRODUCTION

This document describes the architecture of the SwissCube Flight Software.

First, the main focus is on the data system of the spacecraft with all the internal and external data protocols, the data budget of the RF and main internal data flows.

Then the different modes of operation are explained as well as the transitions between them.

And for the last part, the system-wide definitions like for example the error codes are described.

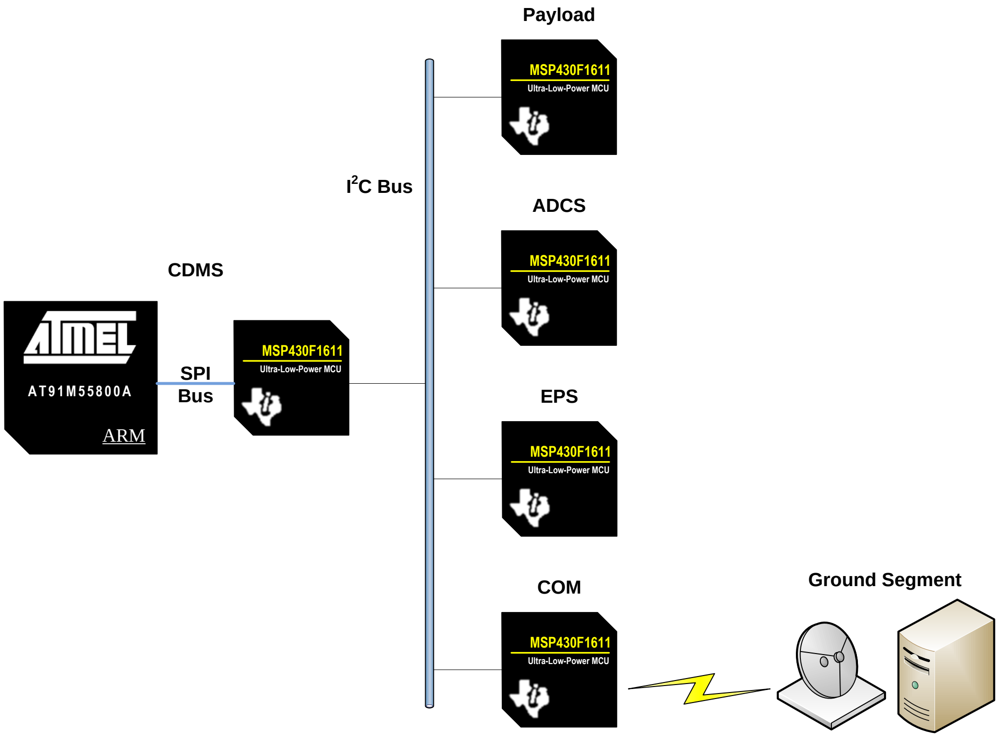
The two others important flight software parts that are the onboard functions and housekeeping parameters for all subsystems are defined in separate documents; respectively [N4] and [N5].

4 SWISSCUBE ONBOARD ARCHITECTURE

To meet the given requirements at best, it was decided that the SwissCube would have a distributed architecture to allow easier development of each subsystems and simplify test procedures as a lot of tests can be done before integration of subsystems. This approach allowed the teams to develop the subsystems independently. As such, each subsystem (COM, PL, ADCS, CDMS and EPS) has its own microcontroller or microprocessor. All the subsystems are linked to a single I2C data bus.

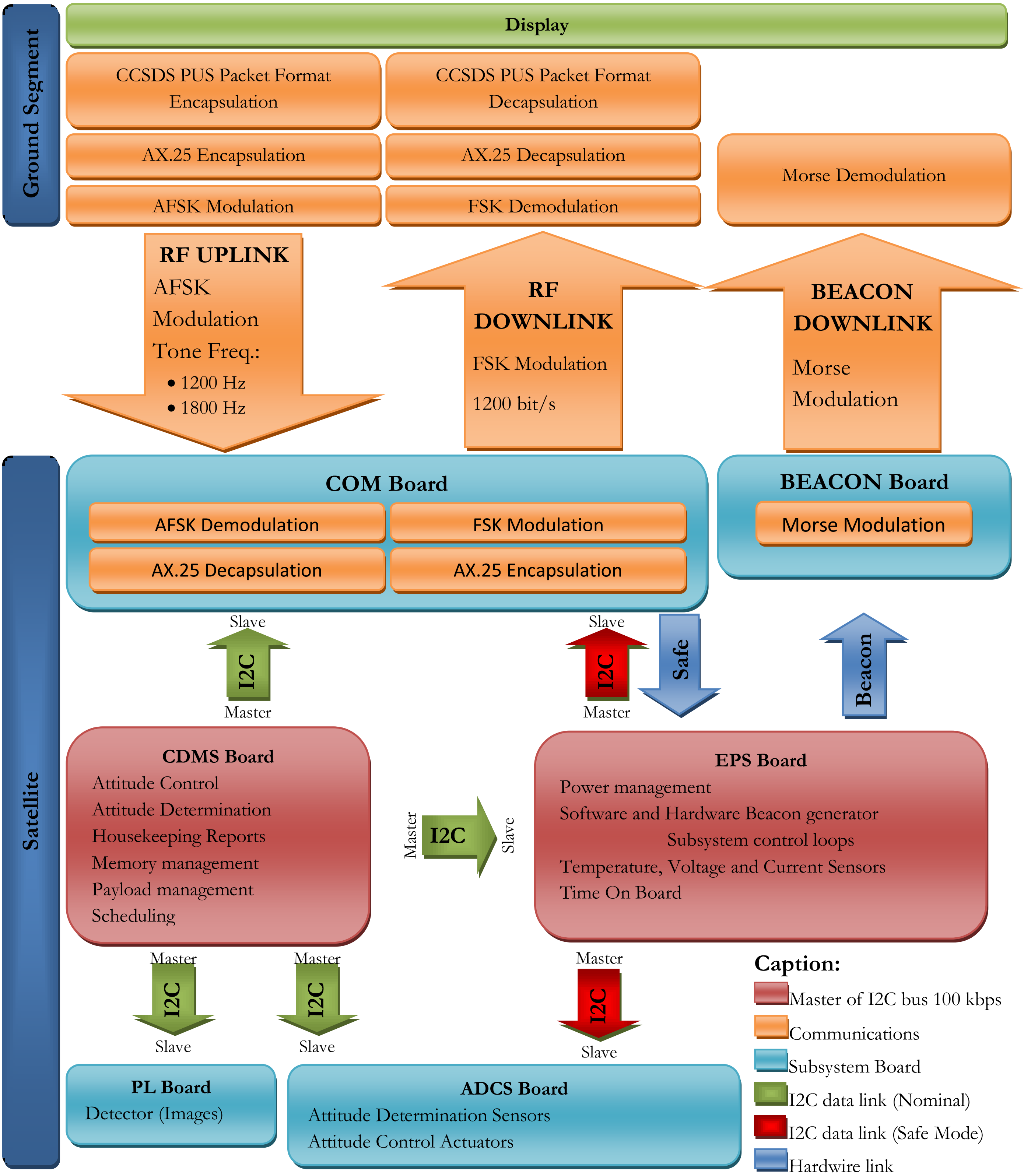
The COM, PL, ADCS, and EPS subsystems all have a MSP430F1611 microcontroller, while the CDMS has an ATMEL ARM AT91M55800A. However, since this microprocessor has no hardware I2C capability, it was decided that it would be linked by an SPI data bus to a MSP430F1611 that would be used as an I2C-SPI bridge. This solution offers two major advantages:

- The MSP430F1611 needs less power than conventional I2C-SPI bridges,
- Commonality of software development for I2C communication,
- This configuration allows the MSP430F1611 to manage all communications for the CDMS board leaving the ARM free to manage the onboard scheduler, housekeeping and other time consuming processes.



5 SWISSCUBE DATA SYSTEM

5.1 Data System Diagram



5.2 Communication Protocols

5.2.1 RF link

The types of modulation used in the different RF links are the most problematic part of the data system. The following subchapters give a brief overview of our choices and problems. More details on these modulations can be found in [N7] and [N8].

5.2.1.1 Beacon

The beacon message is modulated using Morse code, at 437.5 MHz.

The Morse modulation offers several advantages and disadvantages, shown in the table below.

Advantages	Disadvantages
Well known method and technology	Slow bit rate (14 bit/s)
Help from radio-amateur community to locate the Swisscube	
Rapid development of beacon board	
Requires relatively low Eb/No level (6 dB)	

The DPSK modulation was also proposed. It has a higher bit rate and requires less power. However, time of development was critical, and as such the Morse modulation was chosen.

For the beacon messages, see chapter 5.2.2 “Beacon message”.

5.2.1.2 Uplink

The FSK modulation was chosen early on in the project. However a miscommunication led to the development of an AFSK modulation for both uplink and downlink. As the link margin at the end of the RF link budget was sufficient (see [N8]), the AFSK modulation will be kept for the uplink. More details can be found in [N7].

5.2.1.3 Downlink

The link margin for the downlink (see [N8]) was inferior to the required 10 dB, the AFSK for the downlink has to be changed. The COM board is being reviewed to have a better efficiency, and to use the FSK or FFSK modulation.

5.2.2 Beacon message

The BEACON subsystem sends regularly a simple message to earth. It is used to identify the satellite and to send some basic information as to the state of the satellite. It is generated either by the Hardware Beacon Generator or by the Software Beacon Generator, henceforth referred as HBG and SBG.

The details of the beacon message are described in document "S3-B-COM-1-6-Beacon message format" [N10]

5.2.2.1 Basic requirements

The requirements are listed in [N9]. However, here is an overview of some requirements:

- The BEACON board will use at most at input 300mW in peak power, and 50mW in mean power.
- The BEACON board shall transmit at 14bps. This transmission bit rate was a tradeoff between the power consumption for the beacon transmission and the speed at which inexperienced users can decode the Morse message on earth.
- The signal will be modulated in Morse code.

5.2.2.2 Hardware Beacon Message

The Hardware Beacon Generator shall emit the message SWISSCUBE once every 30 seconds. In the absence of a Software Beacon Message (henceforth SBM), the Hardware Beacon Message (henceforth HBM) shall be emitted.

5.2.2.3 Software Beacon Message

The Software Beacon Message shall consist of 3 different parts. Each part will be sent separately, every 30 seconds. Each part will emit the identifier S3, the number of the part, and one or more housekeeping parameters.

5.2.3 AX.25 Protocol

As the project will make use of the radio-amateur network and radio-amateur frequencies, the transfer frames to and from the SwissCube must be compliant with the AX.25 protocol.

After careful consideration, it has been decided that only the UI-Frames (Unnumbered Information Frames) of the protocol will be used. This type of frames allows for telemetry-in-the-blind and command-in-the-blind. The data to be transmitted is first encapsulated in CCSDS PUS packets, and then send in AX.25 transfer frames.

Supplementary fields are present as secondary header and trailer to provide additional capabilities (e.g. virtual channels, frame loss detection, packet sequencing, etc.).

For a complete description these transfer frames, refer to the document “S3-BC-SE-1-1-AX.25 Transfer Frames Format” [N6].

5.2.4 CCSDS PUS Packet format

The packets contained in the AX.25 frames are in the CCSDS/PUS format. This packet format was developed and has been used by the ESA for all its missions for many years. This Standard addresses the utilization of telecommand packets and telemetry source packets for the purposes of remote monitoring and control of subsystems and payloads.

This Standard is structured as follows. Firstly a set of operational concepts is identified, covering all the fundamental requirements for spacecraft monitoring and control during satellite integration, testing and flight operations.

A set of services that satisfy these operational concepts is then defined. The detailed specification of these services includes the structure and contents of the associated service requests (telecommand packets) and service reports (telemetry source packets).

This format is defined in the standard ESCC-E-70-41A [N1].

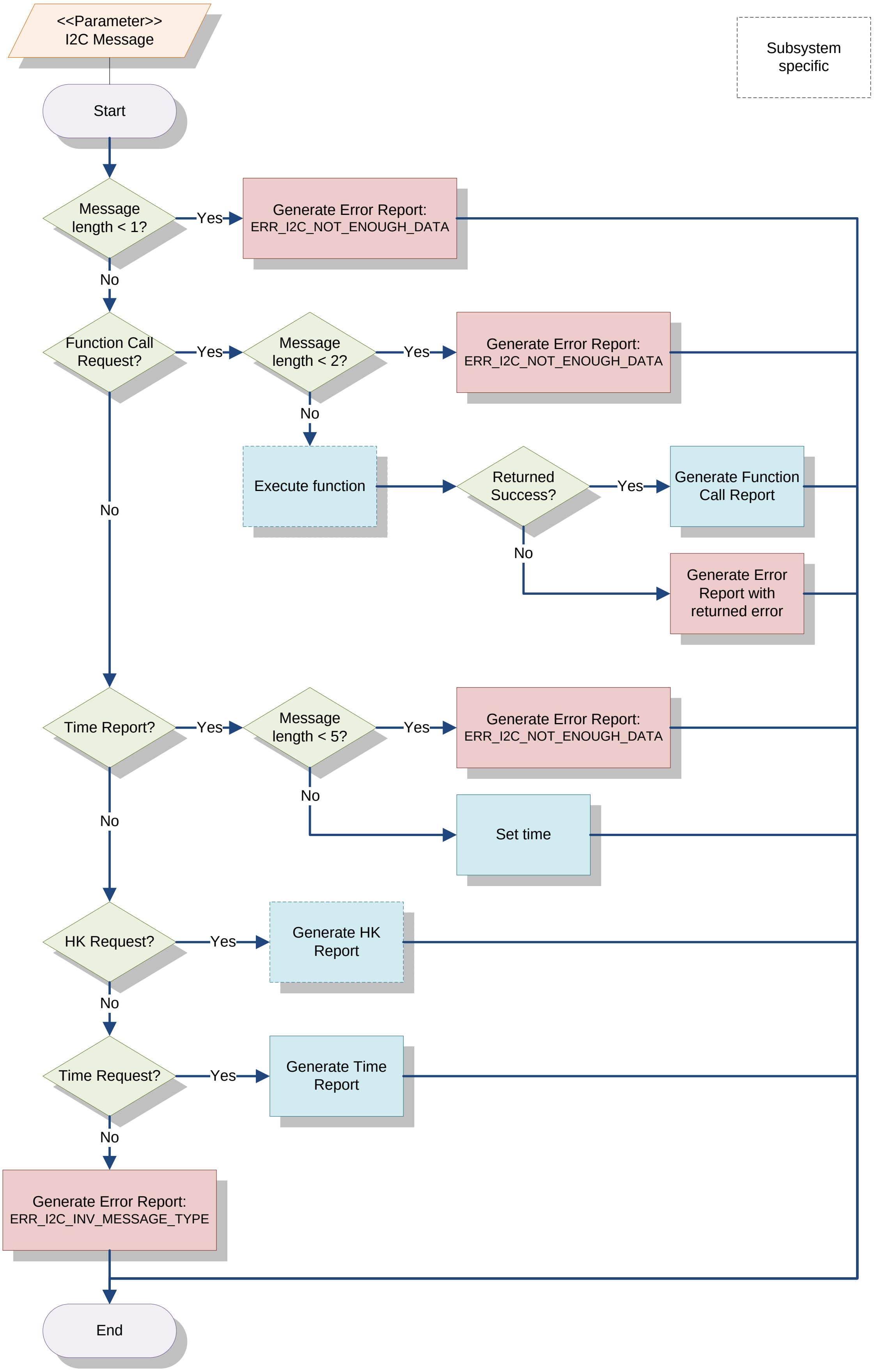
The tailored version for SwissCube is defined in the document “S3-B-ICD-1-3-TMTC Packet Definition” [N2].

5.2.5 I2C Internal protocol

For the onboard communication between the subsystems on the I2C data bus, a simple protocol has been defined. It describes messages used to transfer the housekeeping, call functions on other subsystems and synchronize the time onboard.

This protocol is defined in the document “S3-BC-SE-I2C Internal Protocol” [N11].

The following diagram shows the slave implementation for the processing of the requests defined in this protocol:


<<Parameter>>
I2C Message
Subsystem specific
Start
Message length < 1?
Yes
Generate Error Report:
ERR_I2C_NOT_ENOUGH_DATA
No
Function Call Request?
Yes
Message length < 2?
Yes
Generate Error Report:
ERR_I2C_NOT_ENOUGH_DATA
No
Execute function
Returned Success?
Yes
Generate Function Call Report
No
Generate Error Report with returned error
No
Time Report?
Yes
Message length < 5?
Yes
Generate Error Report:
ERR_I2C_NOT_ENOUGH_DATA
No
Set time
No
HK Request?
Yes
Generate HK Report
No
Time Request?
Yes
Generate Time Report
No
Generate Error Report:
ERR_I2C_INV_MESSAGE_TYPE
End

5.3 Data RF link budget

The SwissCube project is limited by amount of the data that can be transmitted to the ground. This budget was done to control if the SwissCube has enough bandwidth left to manage and download payload images.

5.3.1 RF link information

Bandwidth :				
Raw bandwidth	1200	bits/sec		
Frame AX.25	30	%	29	octets/frame
Error rate	7	%		
Net bandwidth	781.2			
Ground station :				
Min angle of elevation	15	deg		
SwissCube data :				
HK RT Size	200	octets	1600	bits
HK AR Size	200	octets	1600	bits
Payload image size	28200	octets	225600	bits
Payload image size + CCSDS	32296	octets	258368	bits

Raw bandwidth:
The raw bandwidth is 1200bits/sec.

Frame AX.25:
The percentage lost by the AX.25 frame encapsulation.

Error rate:
The error rate is 10^{-5} and margin. With this error rate the RF communication corrupt 2% of AX.25 frame, and 5% of margin for other RF perturbation.

Net bandwidth:
The bandwidth really available for the SwissCube data.

Min angle of elevation:
The minimum angle of elevation of the ground station for successful communication.

HK RT Size:
The housekeeping packets size (real time housekeeping).

HK AR Size:
The housekeeping packets size (archived housekeeping).

Payload image size:
The size of the payload images.

Payload image size + CCSDS:
The size of the payload images including CCSDS packets header.

5.3.2 Budget

	400 km		600 km	
Pass per day	1	2	1	2
Pass duration [min]	4.7	2.3	10.0	5.0
Pass duration [sec]	280	140	600	300
RF initialization [sec]	30	30	30	30
Load Keplerian elements [sec]	5	5	5	5
Load scheduler [sec]	30	30	30	30
Data time available [sec]	215	150	535	470
Total bits available [bits]	167'958	117'180	417'942	367'164
HK RT Frequency [sec]	30	30	30	30
HK RT bandwidth [bits]	14'933	14'933	32'000	32'000
HK RT bandwidth [%]	8.9%	12.7%	7.7%	8.7%
HK AR Frequency [sec]	3600	3600	900	900
HK AR bandwidth [bits]	38400	38400	153600	153600
HK AR bandwidth [%]	22.9%	32.8%	36.8%	41.8%
Acknowledgment RT [bits]	9600	9600	9600	9600
Acknowledgment RT [%]	5.7%	8.2%	2.3%	2.6%
Acknowledgment Store [bits]	9600	9600	9600	9600
Acknowledgment Store [%]	5.7%	8.2%	2.3%	2.6%
Other [bits]	10240	10240	10240	10240
Other [%]	6.1%	8.7%	2.5%	2.8%
Free Bandwidth [bits]	85185	34407	202902	152124
Free Bandwidth [%]	**50.7%**	**29.4%**	**48.5%**	**41.4%**
Number of payload image	**0.33**	**0.13**	**0.79**	**0.59**
Number of days	**3.03**	**7.51**	**1.27**	**1.70**

400 km:

The simulation for a SwissCube launch at 400km. It's the minimum altitude of the launch, and this is the worst case for the RF data budget as the pass duration is the smallest. The simulation is based on the month of July 2008.

600 km:

The simulation for a SwissCube launch at 600km. It's mean altitude for the launch. The simulation is based on the month of July 2008.

Pass per day:

Number of time that the SwissCube ground station sees the SwissCube per day.

Pass duration:

The duration of a pass in minute or second. These values are the mean of passes duration on the simulation for the month of July 2008.

RF initialization:

The time to initialize the RF communication. The SwissCube send a sequence of "01010101…". The ground uses this sequence to synchronize the RF frequency.

Load Keplerian elements:

Time used to load the Keplerian elements. It is for the ADC algorithm.

Load scheduler:

Time used to load the telecommands in the onboard scheduler.

Data time available:

The time available for the rest of the SwissCube data.

Total bits available:

Number of bits available for the SwissCube data.

HK RT Frequency:

The sampling rate for the real time housekeeping reports.

HK RT bandwidth:

The bandwidth used by the real-time housekeeping in bits and percent.

HK AR Frequency:

The sampling time when the housekeeping is archived onboard between passes.

HK AR bandwidth:

The bandwidth used by the housekeeping archive retrieval in bits and percent.

Acknowledgment RT:

The bandwidth used by real-time acknowledgments, the PUS Service 1 "Telecommand verification service", in bits and percent.

Acknowledgment Store:
The acknowledgments stored on the CDMS, in bits and percent.

Other:
Other data. Not defined for the moment, it's a margin.

Free bandwidth:
The bandwidth available for the payload data, in bits and percent.

Number of payload image:
The number of payload images downloadable during one day. To respect the SwissCube requirement, the min value is 0.23 images per day, because the SwissCube must be provide to the ground one image each 4.5 days.

Number of days:
Number of days to download an image. The value must be inferior to 4.5 days.

5.3.3 Conclusion

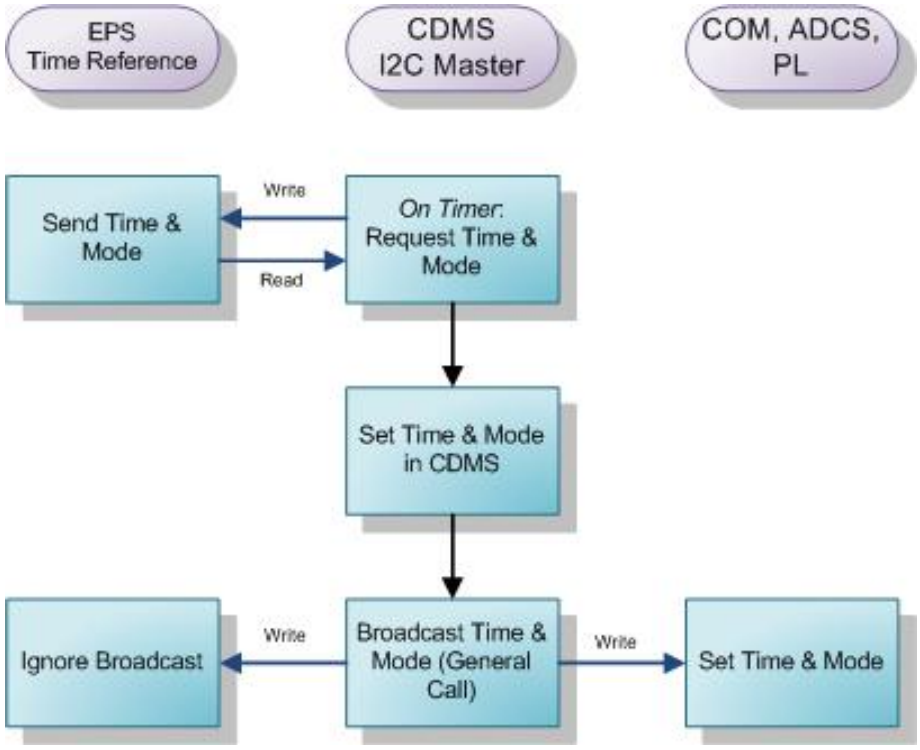
The bandwidth is really a problem for the SwissCube. If the spacecraft is launched at 400km it's possible that we have some problem to respect the payload requirement (one image each 4.5 days). But this case is not very probable. With a spacecraft at 600km we have no problem to respect the requirement.

5.4 Data Flow

This subchapter describes the most important data flows of the data system.

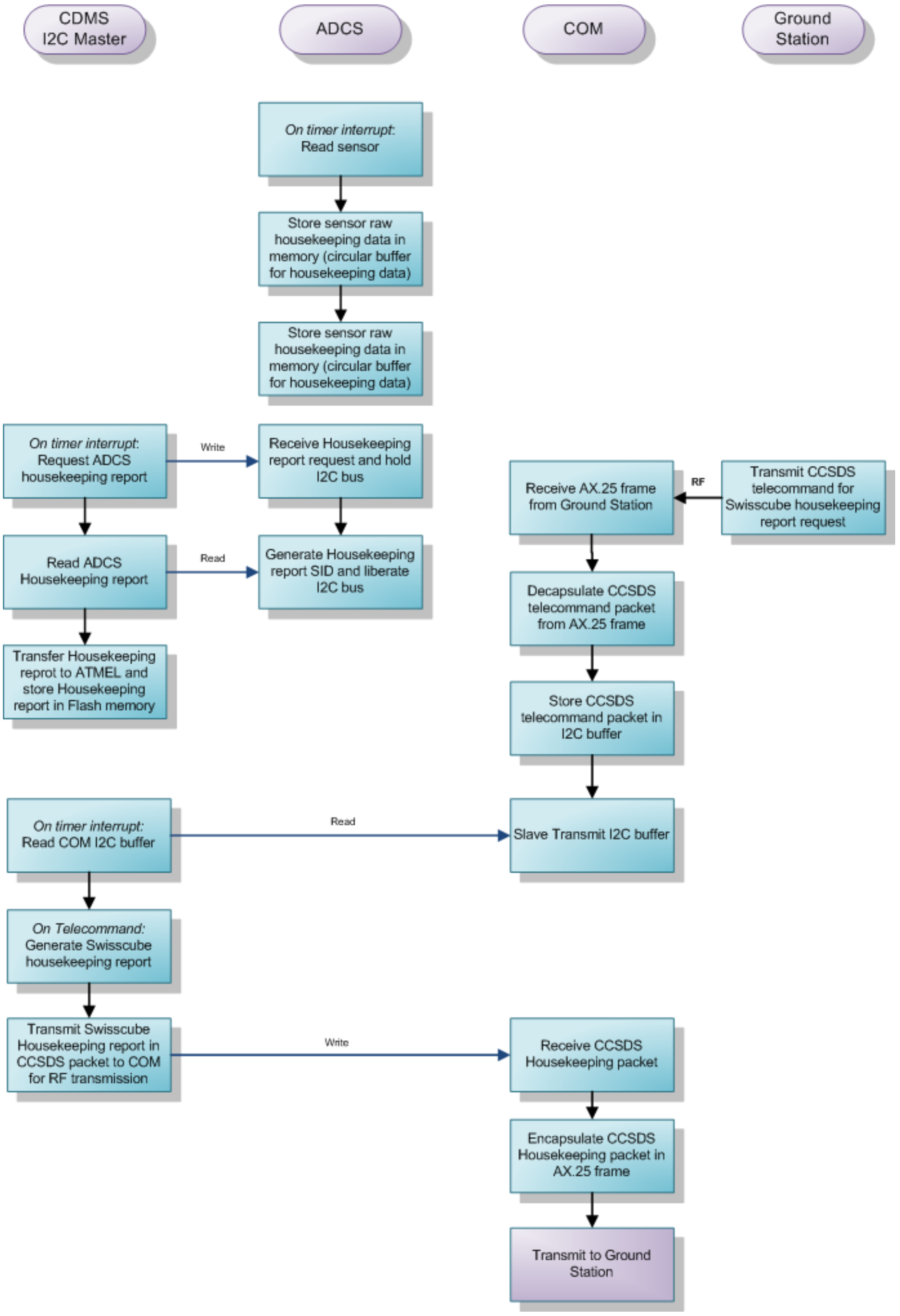
5.4.1 Time On Board

The time reference of the SwissCube is in the EPS microcontroller. The time is updated through the CDMS, as it is the I2C Master, using the messages described in the I2C Internal Protocol [N11]. The figure below shows a diagram of the Time and Mode synchronization in the satellite.



5.4.2 Data flow for a Housekeeping parameter

The figure below is an example of the data flow for the housekeeping parameter of a sensor. For this example, the sensor is on the ADCS board, but the data flow is the same no matter the subsystem.



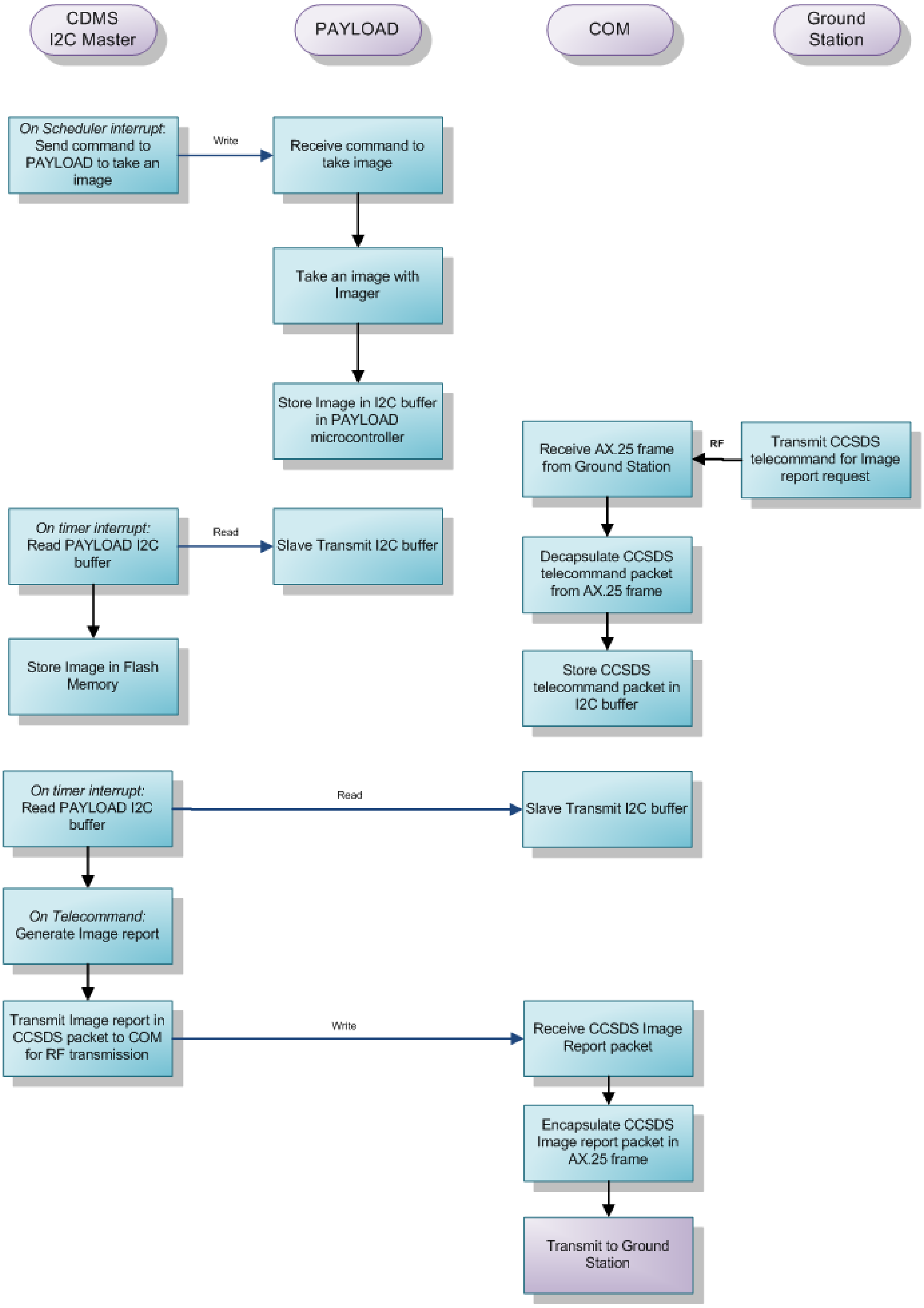
Once the sensor's raw data is read in the ADCS, it is saved in the ADCS microcontroller in a buffer, erasing the old value for the same sensor. Every sensor that can be read by the ADCS microcontroller has a similar storage space in memory.

The CDMS regularly requests housekeeping reports from the subsystems (on a timer interrupt). It initiates communication with the ADCS and requests a housekeeping report. The ADCS microcontroller holds the I2C bus and generates the housekeeping report, in a SID structure.

The CDMS then reads this report, and stores it in the flash memory, again in a circular buffer. It does the same with all the subsystems.

When the Ground Station sends a housekeeping request to the satellite, the COM stores this CCSDS packet in its I2C buffer, and waits for the CDMS to read it (on timer interrupt). The CDMS reads this CCSDS packet and decodes it. As it is destined for the CDMS and to be executed immediately, the CDMS generates the Housekeeping report for the satellite and encapsulates it in a CCSDS Telemetry packet, and sends it to the COM for RF transmission.

5.4.3 Payload Image Data Flow



This data flow is very similar to the data flow of the housekeeping; as such the explanation given above is still valid, with a few modifications.

On a scheduler interrupt (meaning a telecommand from the Ground Station was saved in the scheduler to be executed at a later time), the CDMS sends a command to the Payload to take an Image. The I2C is then released.

Once the image is taken, and saved in the Payload I2C buffer, the rest of the data flow is similar to that of the housekeeping.

6 MODES OF OPERATION

The modes of operation define which subsystems have to be powered on and which spacecraft capabilities are available and are managed by the EPS. Transitions between modes can be done from the ground with telecommands, but can also be triggered by the onboard software in certain situations.

6.1 Enumeration

Modes of operation				
Name	ENUM_SC_Mode			
Description	The spacecraft modes of operation constants.			
Elements	**Name**	SC_MODE_RECOVERY	**Value**	0
	Description	Mode that occurs only when power is extremely low. It stays in this mode until the satellite has enough power to go in safe mode.		
	Name	SC_MODE_SAFE	**Value**	1
	Description	Mode that occurs when the SwissCube start, after a critical failure or when the power is low. The Swisscube needs a telecommand from the Ground Station to exit this state. This mode will also be used during the commissioning phase.		
	Name	SC_MODE_NOMINAL_STANDBY	**Value**	2
	Description	This mode is used when the satellite is in nominal operations but not taking science data or communicating with the Ground Station.		
	Name	SC_MODE_NOMINAL_COM	**Value**	3
	Description	This mode is used when the satellite is communicating with the Ground Station in nominal operations. *Remark: In this mode the Payload's camera must not be used because the SwissCube does not have the sufficient power.*		
Remark				

6.2 Subsystems status

The status of all subsystems for each mode is given in the table below.

Mode		EPS	Beacon	COM Rx	COM Tx	CDMS	ADCS	PL
Safe		🟩	🟩 **SBM**	🟩	🟥	🟥	🟥	🟥
Recovery		🟩	🟥 **SBM to 0**	🟩	🟥	🟥	🟥	🟥
Nominal	Nominal COM	🟩	🟩 **SBM**	🟩	🟩	🟩	🟩	🟩
	Nominal Idle	🟩	🟩 **SBM**	🟩	🟥	🟩	🟩	🟩

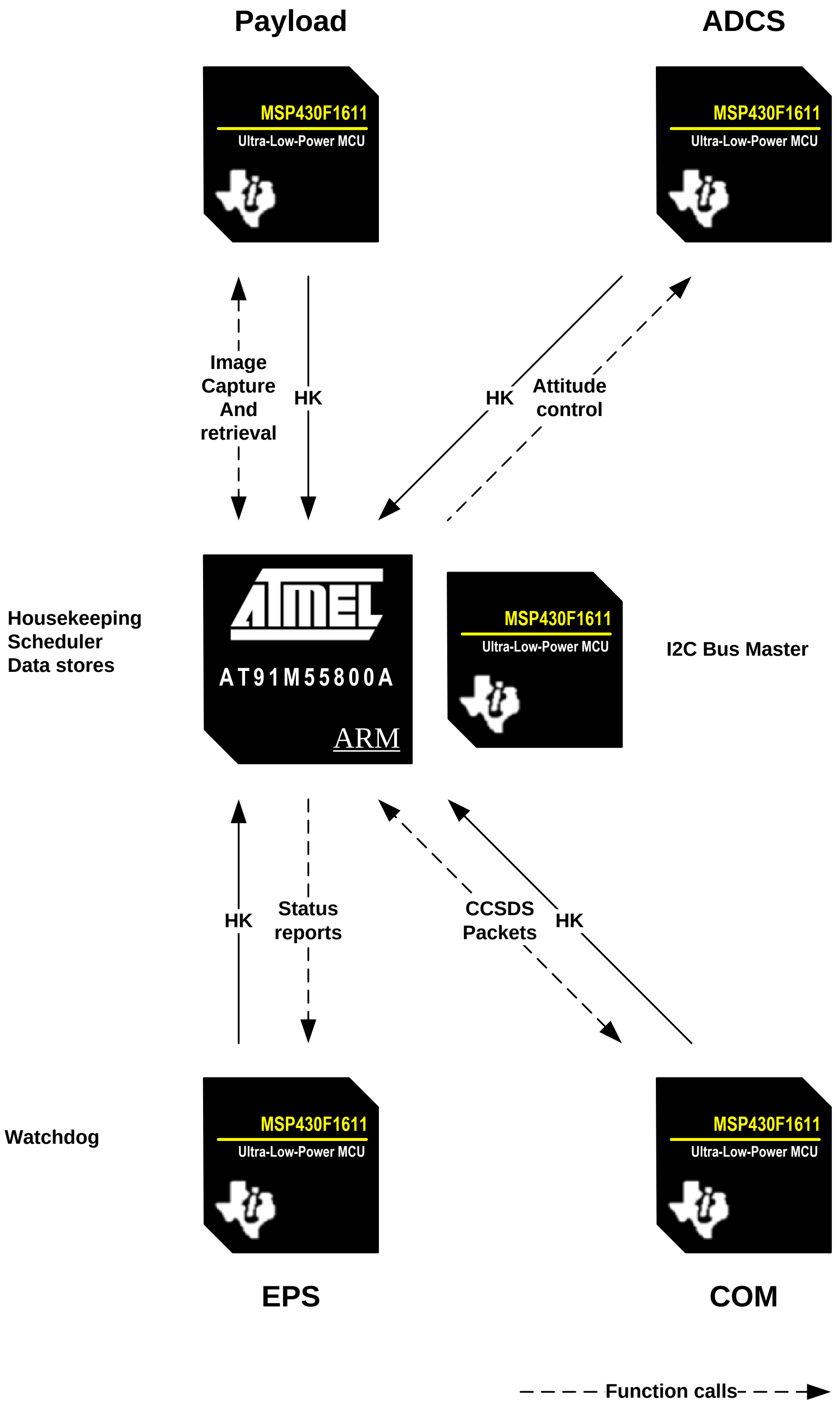
🟩 Active: powered up and executing commands.

🟥 Inactive: not powered up or disabled.

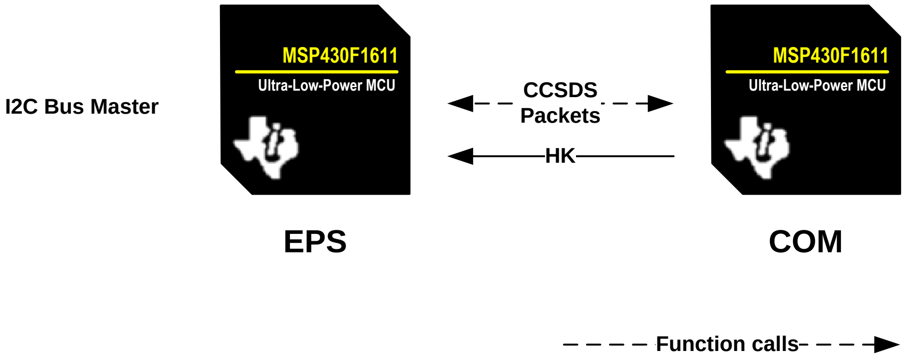
SBM: Software Beacon Message

6.3 Data flows in modes

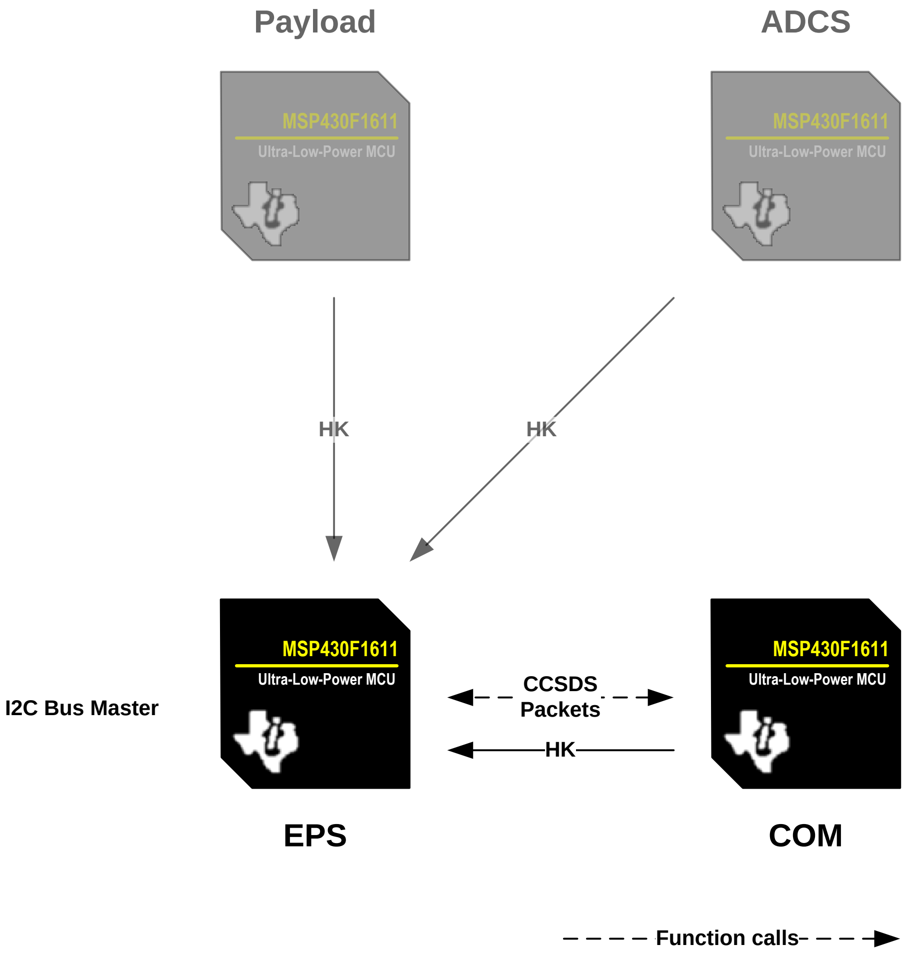
Nominal:



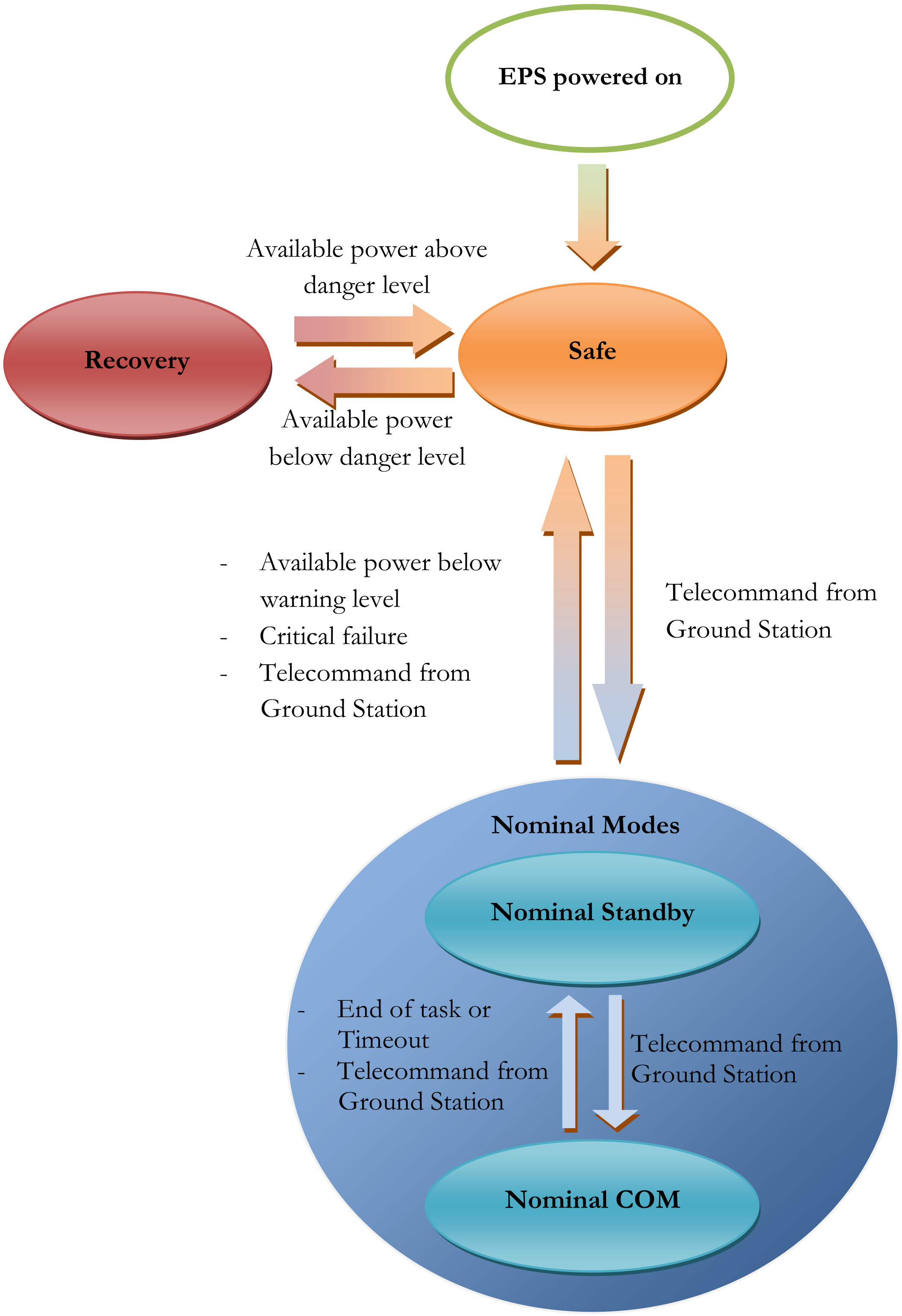
Safe:



Safe during commissioning phase:

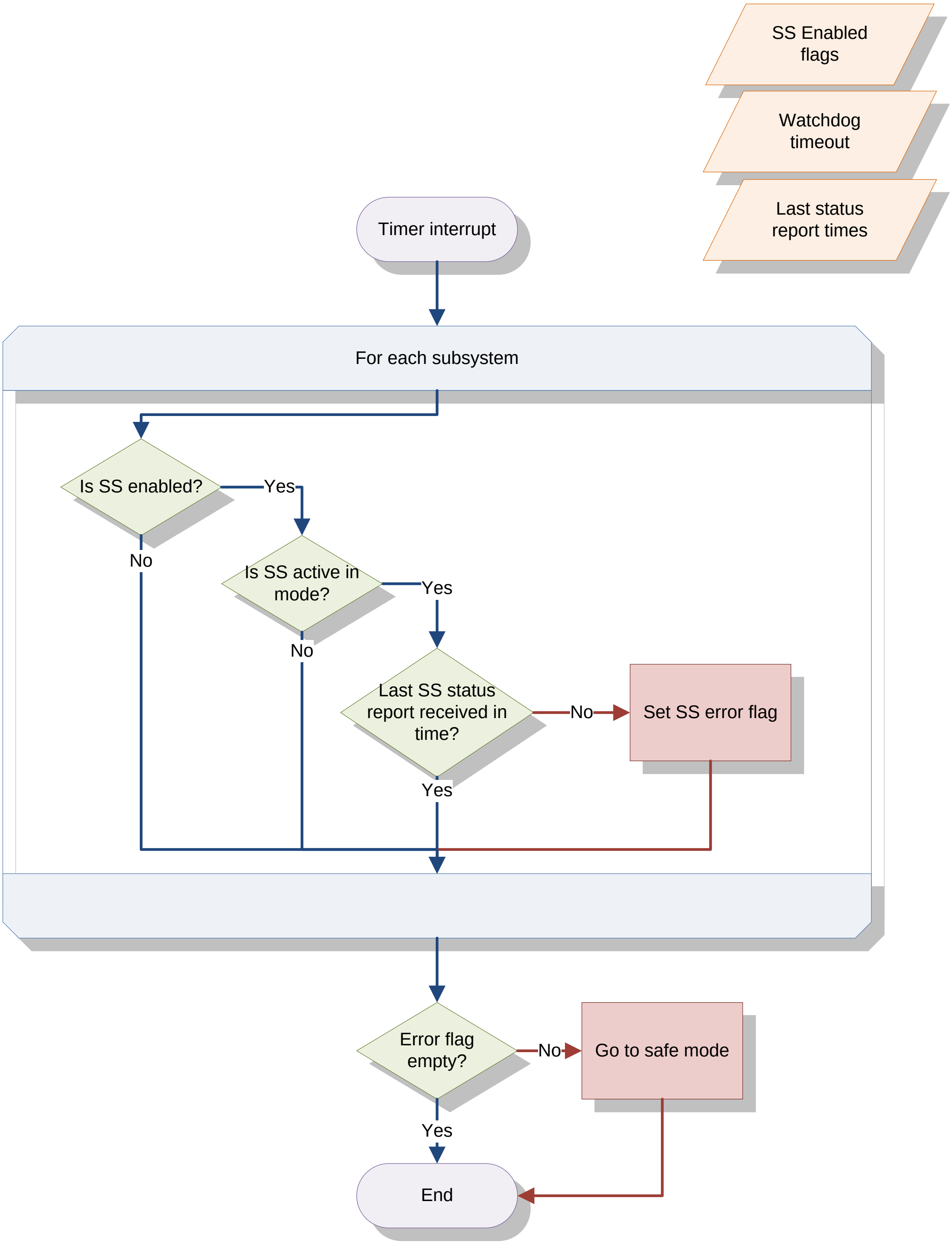


6.4 Transition between modes



6.4.1 Watchdog

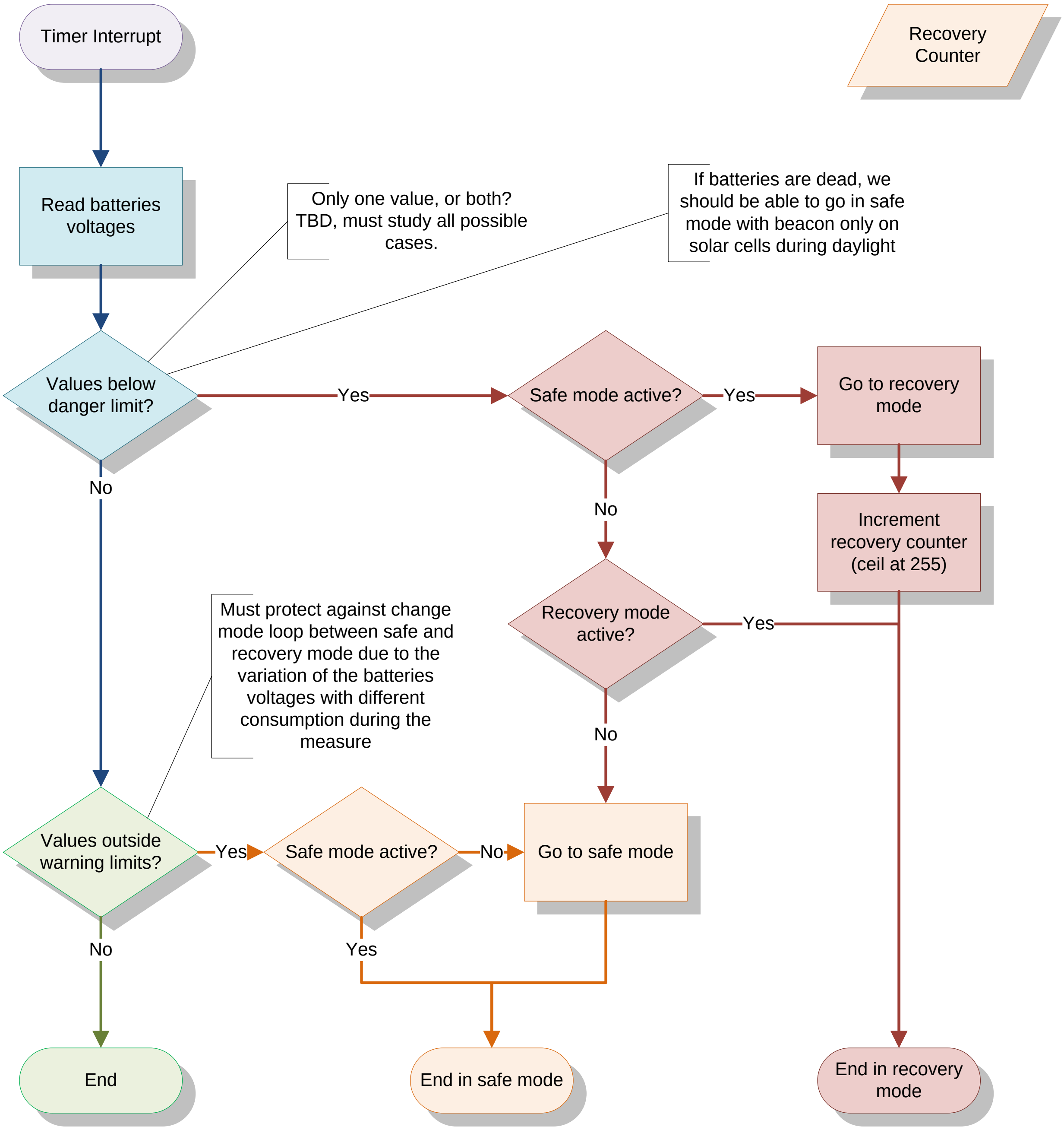
The EPS will automatically detect when a subsystem is not responding if a NACK is encountered on the I2C bus (in nominal modes, the CDMS report the status to the EPS as it is the master of the bus). When a non-responding subsystem is detected, a passage to safe mode is triggered.



6.4.2 Available power check

The EPS will continuously check the available power and trigger a passage in recovery mode if there is not enough left to power the beacon with a good margin. This allows the spacecraft to keep enough power during an eclipse as to not let the batteries be completely empty and have the EPS or COM shut down.

As soon as the available power is high enough, the EPS will automatically trigger a passage in safe mode to get the beacon working again.



7 SPACECRAFT ID

Each spacecraft has a unique ID; we have two flights models and one qualification model.

SwissCube qualification model	
Spacecraft id	0x0103
Name	SCID_S3_QUALIF
Description	SwissCube qualification model.
Remark	

SwissCube flight model 1	
Spacecraft id	0x0101
Name	SCID_S3_FLIGHT_1
Description	SwissCube flight model number 1.
Remark	

SwissCube flight model 2	
Spacecraft id	0x0102
Name	SCID_S3_FLIGHT_2
Description	SwissCube flight model number 2.
Remark	

8 APPLICATION PROCESS ID (APID)

APID of ADCS subsystem			
Application process id	0x0B	**Subsystem**	ADCS
Name	APID_ADCS		
Description	The APID of ADCS subsystem		

APID of CDMS subsystem			
Application process id	0x0A	**Subsystem**	CDMS
Name	APID_CDMS		
Description	The APID of CDMS subsystem		

APID of communication subsystem			
Application process id	0x09	**Subsystem**	COM
Name	APID_COM		
Description	The APID of communication subsystem		

APID of EPS subsystem			
Application process id	0x08	**Subsystem**	EPS
Name	APID_EPS		
Description	The APID of EPS subsystem		

APID of payload subsystem			
Application process id	0x0C	**Subsystem**	Payload
Name	APID_PAYLOAD		
Description	The APID of payload subsystem		

9 ERROR CODES

The tables hereafter describe the error code defined at system-level. Each subsystem functions can define supplementary errors code for their own error reporting.

System-wide errors codes			
Name	ERR_I2C_NACK	**Value**	0xEF
Description	The addressed subsystem on the I2C bus did not acknowledge. It may be not responding or powered down.		
Name	ERR_UNKNOWN_FUNCTION	**Value**	0xF1
Description	An unknown function was requested to be called.		
Name	ERR_MISSING_PARAMETER	**Value**	0xF2
Description	A parameter is missing for the function call.		
Name	ERR_VALUE_OUT_OF_RANGE	**Value**	0xF3
Description	One of the parameter values was out of the acceptable range.		
Name	ERR_INTERNAL_ERROR	**Value**	0xFF
Description	Internal error. Should never happen as subsystem must define their additional error codes instead.		

10 ON-BOARD STORAGE

Acknowledgment storage			
Store id	0x01	**Subsystem**	CDMS
Name	STORE_ACK		
Description	The storage of acknowledgments.		
Content	All acknowledgements when the spacecraft isn't on communication with the ground.		
Bits size	184000 TBC	**Max element**	1000 TBC
Remark	This storage is the first to be empty by the ground.		

Housekeeping archiving storage			
Store id	0x02	**Subsystem**	CDMS
Name	STORE_HK_AR		
Description	The storage of housekeeping archiving.		
Content	The housekeeping archiving.		
Bits size	2480000 TBC	**Max element**	1000 TBC
Remark			

11 SWISSCUBE SUBSYSTEMS ENUMERATION

SwissCube subsystem				
Name	ENUM_SC_SUBSYSTEM			
Description	The SwissCube subsystems.			
Elements	**Name**	SC_SUBSYSTEM_EPS	**Value**	0x01
	Description	EPS subsystem-		
	Name	SC_SUBSYSTEM_COM	**Value**	0x02
	Description	Communication subsystem.		
	Name	SC_SUBSYSTEM_BEACON	**Value**	0x04
	Description	Beacon subsystem.		
	Name	SC_SUBSYSTEM_CDMS	**Value**	0x08
	Description	On-board computer subsystem		
	Name	SC_SUBSYSTEM_ADCS	**Value**	0x10
	Description	Altitude, determination and control subsystem.		
	Name	SC_SUBSYSTEM_PAYLOAD	**Value**	0x20
	Description	Payload subsystem		
	Name	SC_SUBSYSTEM_ALL	**Value**	0xFF
	Description	All subsystems		
Remark	These constants are designed to be combined, each one represent a bit.			

12 VIRTUAL CHANNELS

The use of virtual channels enables the separation of the transmission flow of multiple sources, as seen in the table below.

The virtual channel is coded on 3 bits in the AX.25 transfers frames [N6], this allows for up to eight virtual channels to run on a particular physical data channel.

Virtual channels				
Name	ENUM_SC_VC			
Description	Virtual channels			
Elements	**Name**	SC_VC_ACK_RT	**Value**	0x01
	Description	Acknowledgment real-time virtual channel.		
	Name	SC_VC_ACK_AR	**Value**	0x02
	Description	Acknowledgment archiving virtual channel.		
	Name	SC_VC_PL	**Value**	0x04
	Description	Payload virtual channel.		
	Name	SC_VC_HK_AR	**Value**	0x06
	Description	Housekeeping archiving virtual channel.		
	Name	SC_VC_HK_RT	**Value**	0x07
	Description	Housekeeping real-time virtual channel.		
Remark	To keep a certain compatibility with CCSDS frames, the virtual channel 0 that shall be reserved for time packets is not used here.			