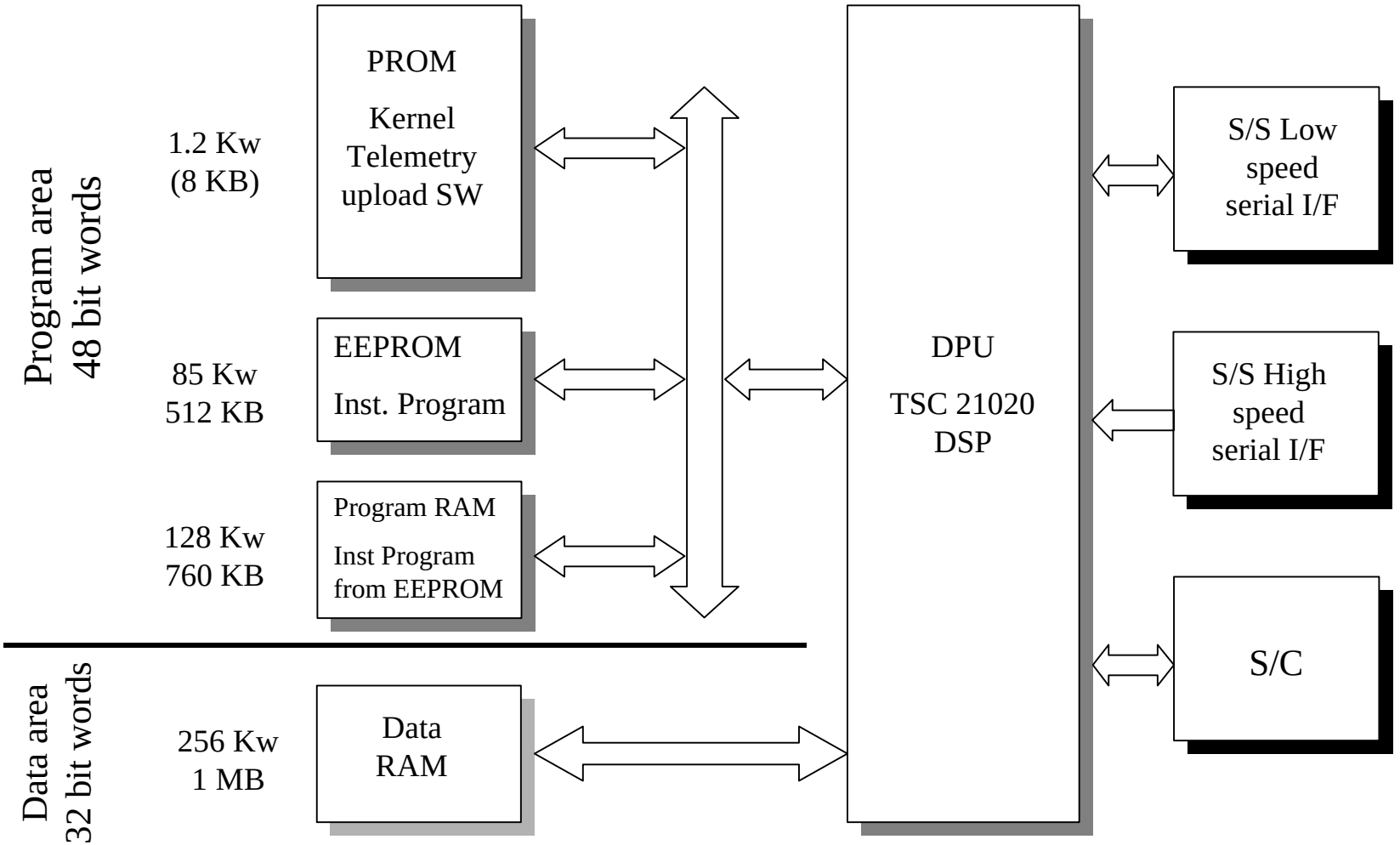
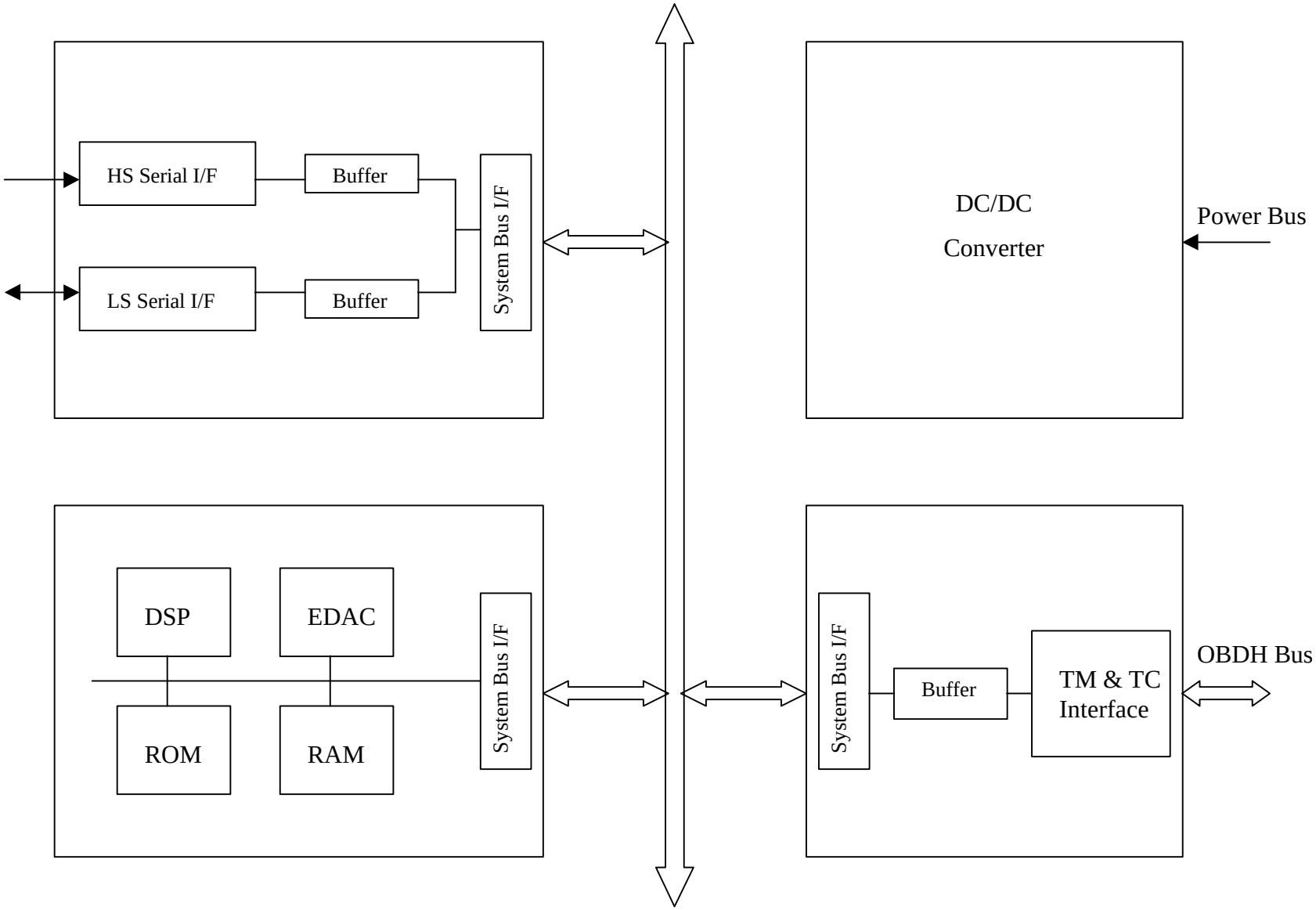
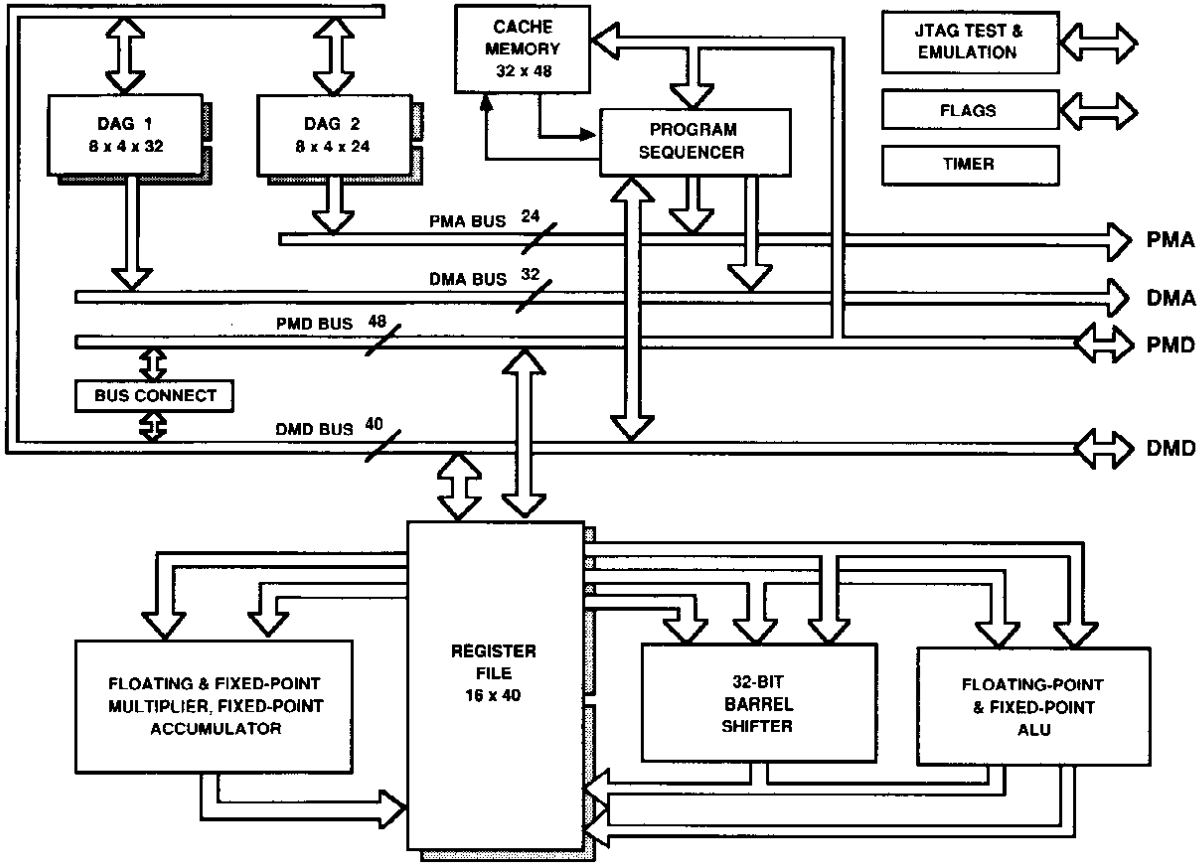
Memory organisation




HS Serial I/F
Buffer
LS Serial I/F
Buffer
System Bus I/F
DC/DC
Converter
Power Bus
DSP
EDAC
ROM
RAM
System Bus I/F
System Bus I/F
Buffer
TM & TC
Interface
OBDH Bus

TSC21020E block diagram



Prog Bus => 48 bit

Data Bus => 32 bit

4 ext. interrupt lines
16 Mw Prog & 4 Gw Data RAM
Internal 32 bit down counter timer

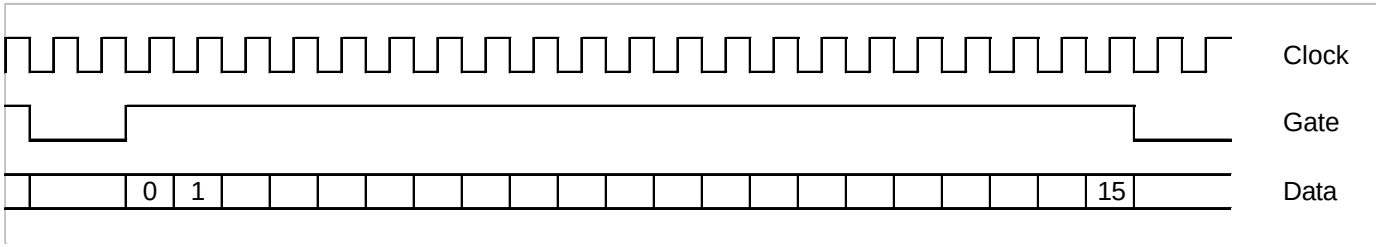
Interfaces

Present design is based on serial interfaces in order to minimize interconnecting wiring and in agreement with requirements with the other two FIRST instruments. From the noise point of view it is required a balanced RS-422 interface.

The present (PACS and HIFI) I/F are:

- One synchronous serial bi-directional bus. This bus is common for all subsystems and handles commands and HK with a speed of ~ 100 - 500 Kbit/s.
- (One) high speed data link(s) based on:
  IEEE Std 1355-1995 (SMCS 332)
  Synchronous serial high speed link.
- Simple monodirectional (S/S to DPU) 3 wire synchronous serial I/F

High speed I/F



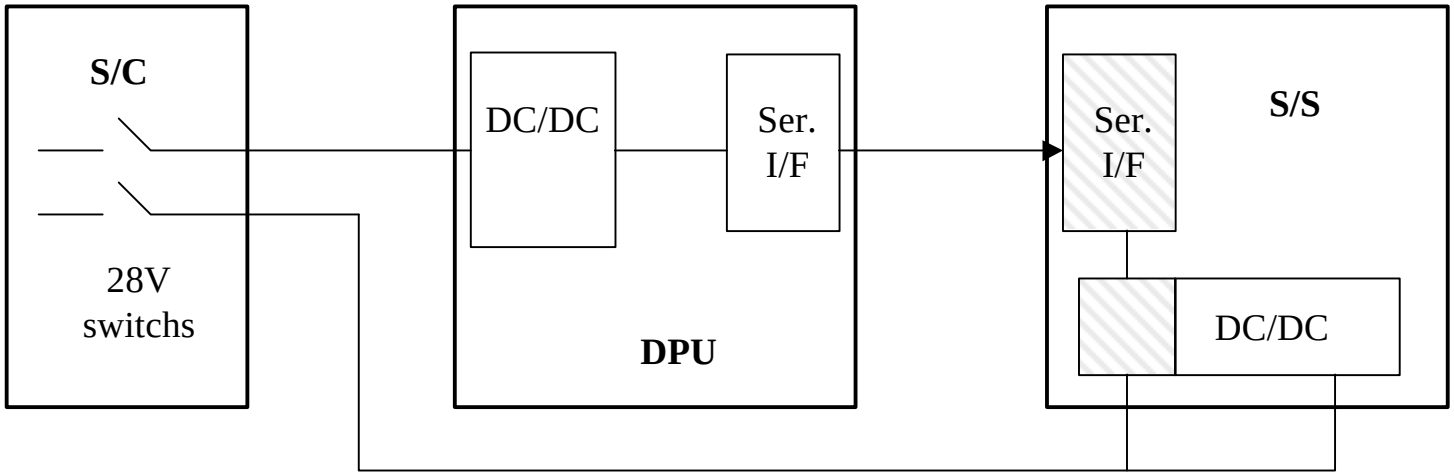
- For HIFI we agreed to receive clock, gate (and data) from S/S (clock speed ~ 2MHz).

- For PACS the fast I/F will be based on IEEE Std 1355-1995 (SMCS 332) Synchronous serial high speed link (**196** pin chip).

Power supply distribution

We assume that each S/S has its own DC/DC converter. The 28 V will be switched directly by the S/C (no relay inside the DPU).

In order to have a controlled switch On/Off sequence, each S/S should have an interface with the DPU powered by a separate section of their converter. The DPU will send the “full” switch On (serial) command.



DPU S/W concept

We plan to use **PC** platform - **Linux** OS whenever possible or Win NT. Programming language will be C.

We are investigating the possibility (HW and SW) to evaluate the EONIC OS. We have a 21020 EZ-LAB Evaluation board which we guess does not support the EONIC OS. In any case we would like to avoid the use of this OS as it will be expensive in terms both of resources and speed.

Revision control will be done using **CVS** repository to enable multiprogrammers activities and easy upgrading (GSE will allow easy uploading of the OBS).

We are evaluating several CASE tools, a good candidate seems to be AxiomSys. http://www.stgcase.com/

which is based for the structured analysis and the architecture modelling on the Yourdon- De Marco method.

S/C simulator

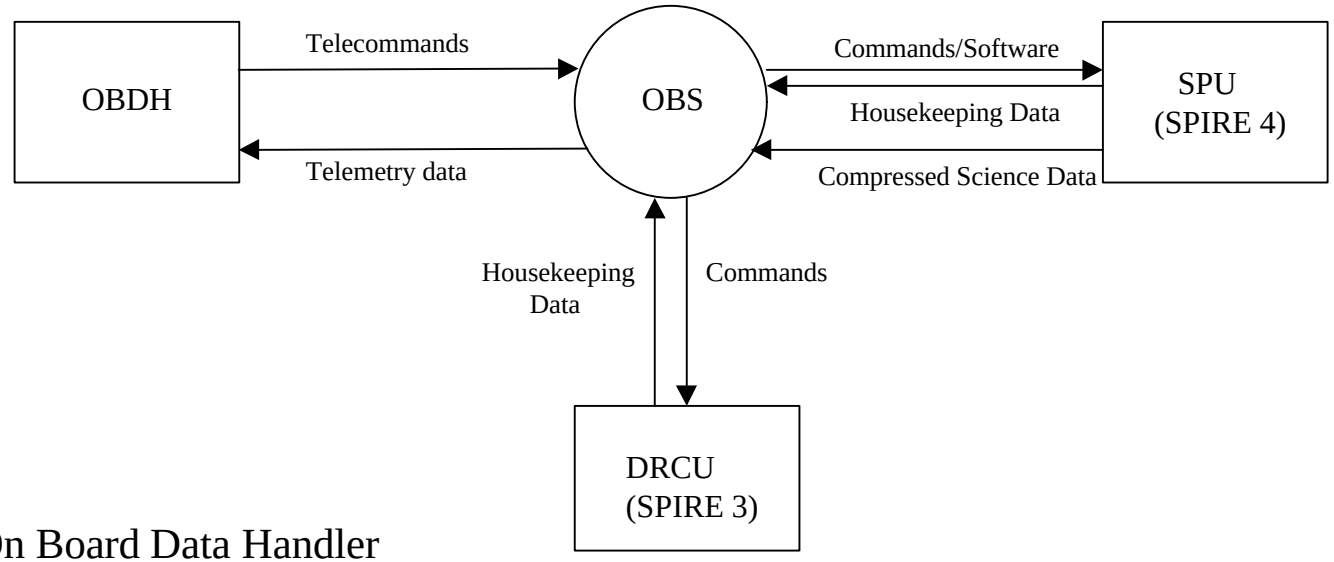
One offer received from CRISA (6000 ECU last year, 20000 ECU one month ago) for a PC-AT board including windows or MS-Dos SW. C driver for custom application included.

Present S/C I/F based on MIL-STD-1553-B. Allows wider supplier choice

Suggestion to EGSE team to ask for other offers.

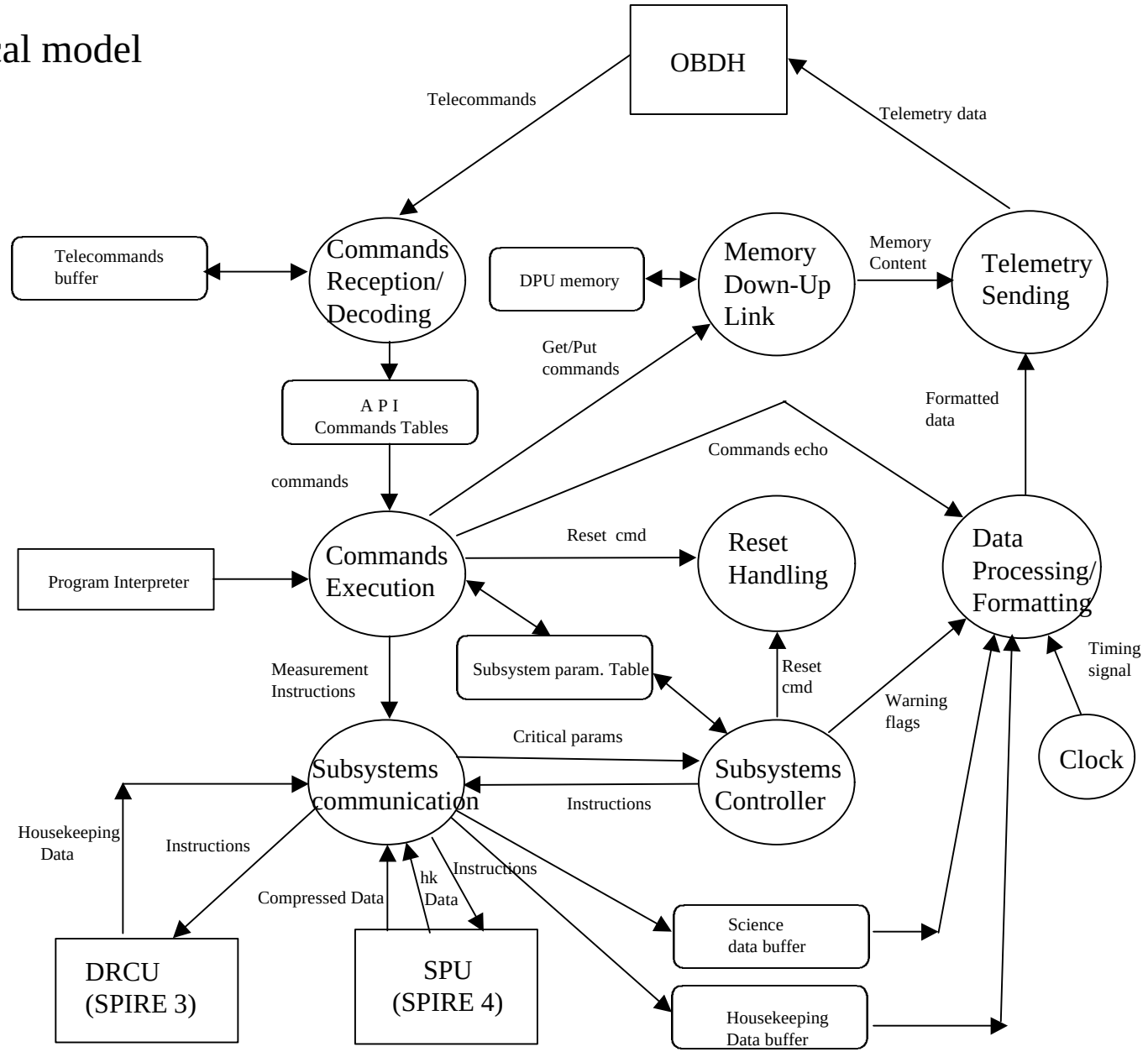
EGSE Planning: we would like to have EGSE in institute Tdelivery - TBD.

SPIRE On Board Software
Context Diagram



OBDH On Board Data Handler
OBS On Board Software
SPU Signal Processing Unit
DRCU Detector Read-out & Control Unit

SPIRE OBS - Logical model



Functional decomposition

Based on the functional description contained in the SPIRE Scientific and Technical Plan, the following main functional components have been individuated for the SPIRE OBS:

1. Commands Reception/Decoding :

- Receives the telecommands from OBDH and stores them sequentially in the Telecommands Buffer (a cyclic buffer necessary to store all the telecommands as soon as they are received, until they are examined for their decoding and execution);
- Reads the Telecommand buffer, executes a first level analysis, and moves the telecommands to the three Commands Tables (one per execution priority: I, P, A), which contain the received commands sequentially, as they were transmitted.

2. Commands Execution :

- Reads from the Commands Tables the commands to be executed, according to their priority;
- Interprets the Commands according to given algorithms;
- If requested, it passes the RESET command to the *Reset Handling* component;
- Updates the Subsystems Parameters Table(s), which is used for the actual execution of the measurement;
- Sends the relevant sequence of digital commands (Measurement Instructions) to the *Subsystem Communication* component;
- If requested, it passes the memory reading/writing commands to the to the *Memory Down-Up link* component.

Functional decomposition

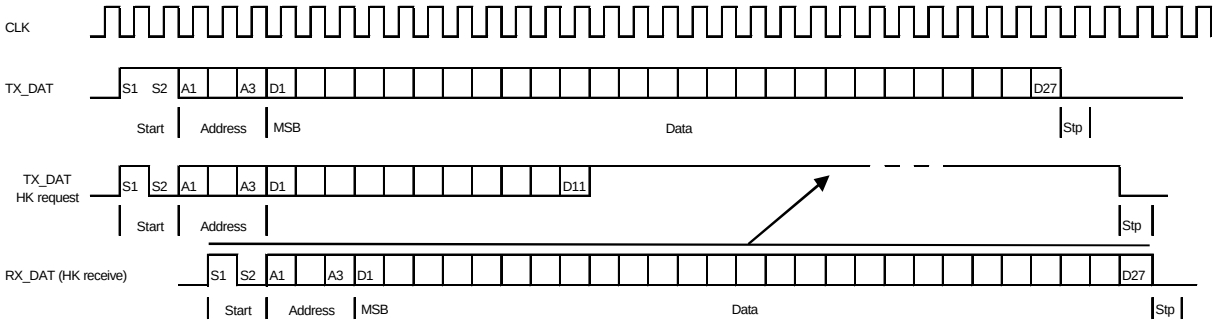
3. Subsystems Communication:

- It interfaces the instrument subsystems, by sending the measurement instructions, and getting the Housekeeping information from all the subsystems and the compressed science data from the SPU;
- Updates the Science Data Buffer and the Housekeeping Data Buffer;
- Sends the subsystems critical parameters to the *Subsystems Controller* component;
- Receives from the *Subsystems Controller* component the instructions generated after the analysis of the critical parameters and sends them to the subsystems (autonomy functions);
- Sends S/W modules to SPU.

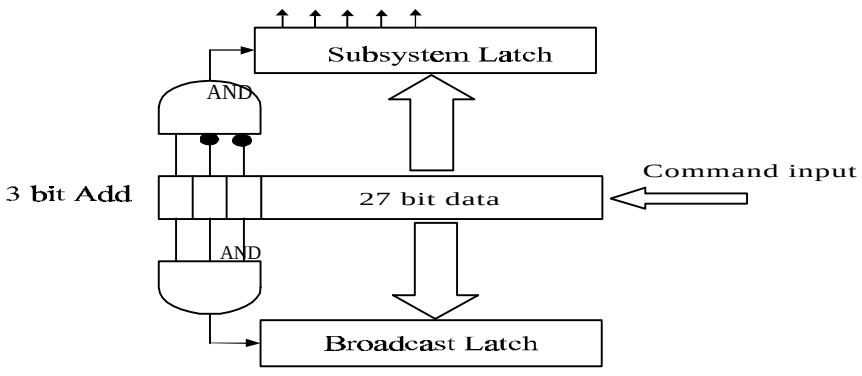
4. Subsystems Controller:

- Receives from the *Subsystems Communication* component the values of the critical parameters;
- Checks the parameters and flags the abnormalities; the warning flags are sent to the *Data Processing/Formatting* component to be included in the Telemetry data;
- In case of autonomous operation, it may take action to switch off parts of the instrument, by sending an appropriate set of instructions to the *Subsystem Communication* component.

All data transactions with the addressed subsystem (addr. In TX_DAT), are initiated by DPU. DPU will send data to all subsystems using one serial data line TX_DAT and can send both commands and HK requests via this line.
Subsystems will send responses via RX_DAT line.
A **command** is made of 2 start bit, followed by 30 bit divided in address and data and 1 stop bit. For HIFI we suggest 3 address bit and 27data bit as shown in figure.
HK request is made of 2 start bit 3 address bit and 11 data bit. After transmission of these bit, the TX_DAT line shall stay high until the corresponding HK response has been received. Then 1 stop bit will follow.
A **HK response** shall consist of 2 start bit, 30 address and data bit and 1 stop bit.
The figure shows the proposed HW protocol. Clock rate ~ 100KHz

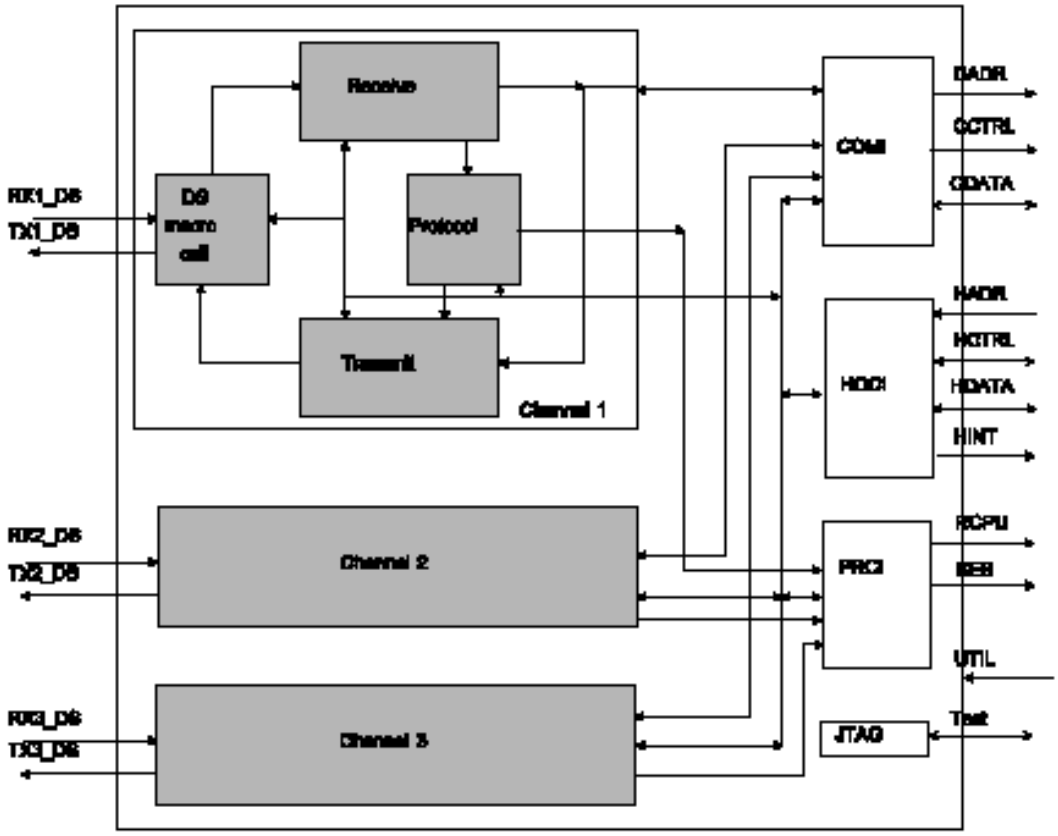


The 3 address lines allow interfacing with 8 different subsystems each with a different formatting of the data bits. One address (i.e. the addr = 7) can be reserved as a broadcast command, with its own formatting of the data field, which carries information “interesting” for all subsystems. The data field of the broadcast command, if necessary, can be further divided in order to address only part of the subsystems to perform a “partial” broadcast command.
Each subsystem should capture at least the serial commands with its own address plus the broadcast command.
In the next figure it is shown the block diagram of the serial command decoding for subsystem at address 4



SMCS 332

Scalable Multi-channel Communication Subsystem



- Scalable and flexible high performance Serial Communications Controller:
  - Three IEEE Std 1355-1995 links (DS-LinkTM): full duplex, up to 200 MBit/s (in each direction) point-to-point links
  - High-level packet oriented data transfer
  - Autonomous command execution
  - Fault tolerance features
- Emphasis put on high-throughput with low CPU interaction; scalable interface allows integration with any CPU type (data bus width of 8, 16 or 32 bits)
- Support for heterogeneous multi-processor systems; see data sheet for ***Mosaic020***

Main features:

- three IEEE 1355 serial communication links
- 0 to 200 MBit/s transfer rate per link
- link disconnect detection and parity check at token level
- checksum generation / check at packet level
- available in various quality grades from Temic/MHS in a 196 pin CQFP package, including radiation tolerant versions (50 Krad, improved SEU performance)

At this stage of the project we can only provide a preliminary list of packet services TBC.
We believe that in the near future some of this services will be dropped.

Service Type	Service Name	Service Scope
1	Telecommand verification	It provides the capability for the verification of telecommand packets.
3	Housekeeping Reporting	It provides for the reporting to the ground of HK info
5	Event reporting	It provides for the reporting to the ground of various events as: failures, anomalies, autonomous actions etc
6	Memory management	It provides the capability for loading, dumping, and checking the contents of on boarb memory.
9	Time management	It provides the capability for the distribution of S/C time reference
11	On board scheduling	It provides the capability to command on board application processes using telecommands pre-loaded on board the S/C
12	On board monitoring	It provides capability to monitor on board parameters with respect to checks defined by the ground
13	Large data transfer	It provides capability to transfer large data units in a controlled manner.
20	Science data transfer	It provides the control of the science data transfer to OBDH
21	Private telecommands	It is used to send private telecommands (i.e. those TC, not covered under other packet services)

- **Instrument commanding**

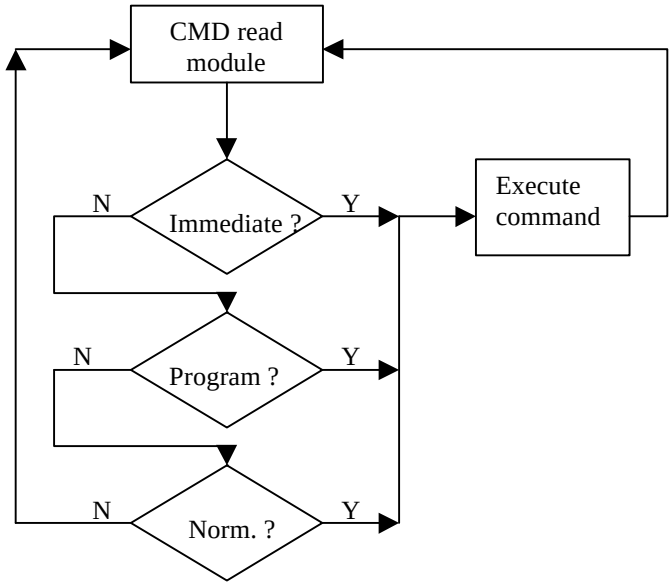
Each instrument command is transmitted as a variable length packet of 16 bit words having the following general structure:

1. a Header describing the Command **function**;
2. the **number** of words to follow;
3. the new values of the **parameters**, if any.

There are two main categories of commands:
- Standard Commands
- Time tagged Commands

The time tagged commands are standard commands to be executed at a specified time. Due to the communications constrain of the FIRST S/C we expect that nearly all commands will be time tagged and the delivery time to the instrument will be managed by the OBDH.

There are 3 types of standard commands, defining the execution priority from high to low, stored by the OBS in different circular buffers.
- Immediate commands **I**
- Program commands **P**
- Normal commands **N**



Generally each command can belong to each of the 3 types.

Immediate commands are executed at the end of the current command execution phase.

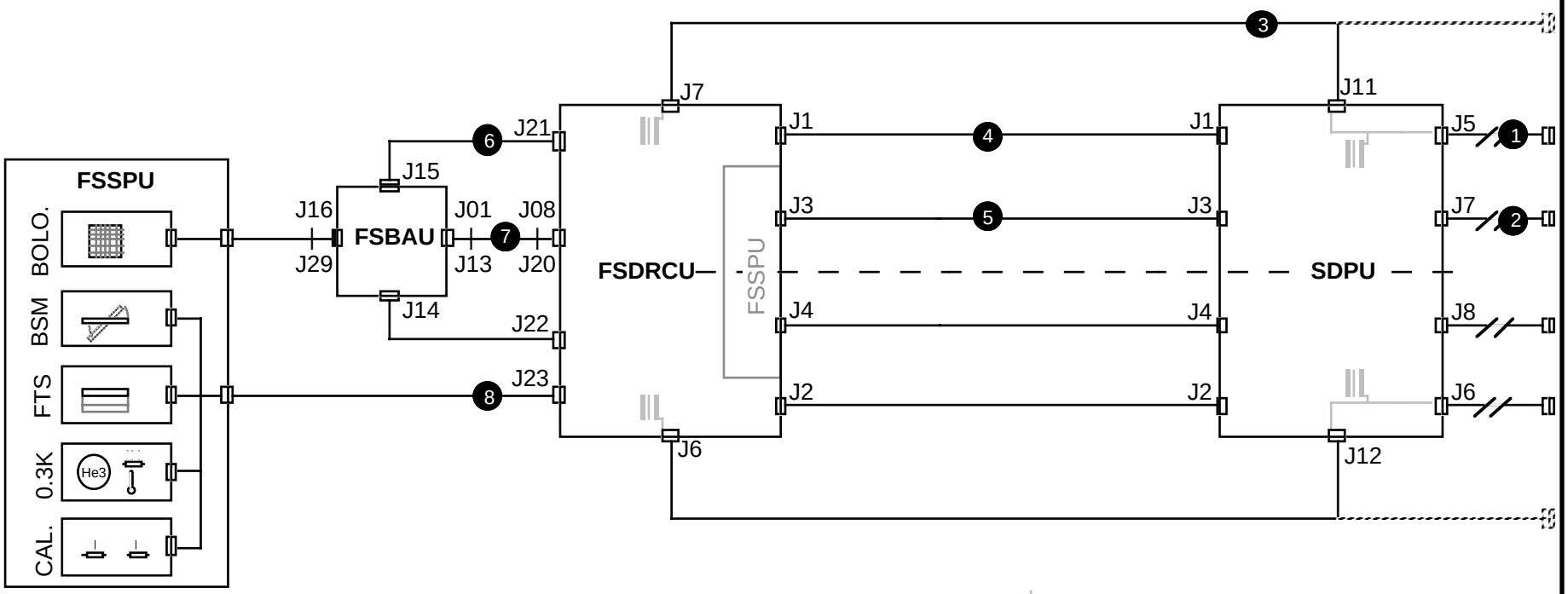
Program commands are commands executed as a sequence. The sequence (i.e. the program) is formed by standard commands plus a few "pseudo instructions" special commands, defining elementary programming language statements (i.e. for loop, if statements, setting of program variables etc). A few of possible "pseudo instructions" are:

RUNP n	Execute program # n
ENDP	End of program
SETRX n, x	Set S/W register n to x
INCRX n	Increment/decrement S/W register n
JUMP …	Various jump based on S/W register count (Jump, Loop, if…..goto)
WAIT	Wait for internal action or fixed time

A program is initiated by a **I** or **N** *RUNP* command and is terminated by the *ENDP* instruction. Depending on the (ring) buffer dimension for programs commands, more than one program may be stored on board and executed at any time with a time tagged RUNP n command.

Normal commands are executed when no other command type is present.

SPIRE ELECTRICAL CONFIGURATION
OPTION C

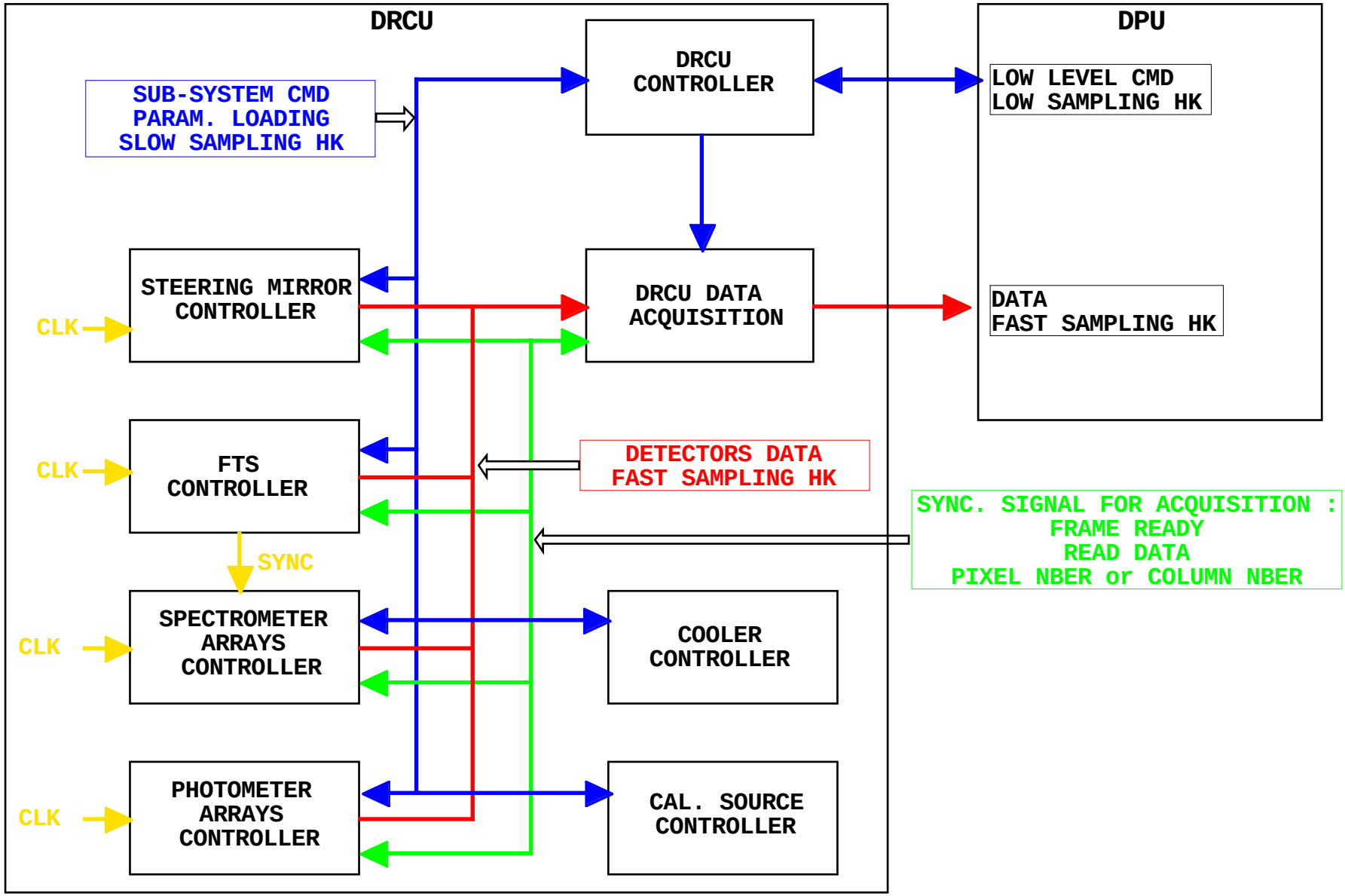


1 PIRE TC/TM
2 PIRE PPWR
3 RCU PPWR
4 RCU LOW LEVEL CMD

SCIENTIFIC DATA INTERFACE
BAU SPWR
BAU SIGNAL
MECHANISM CONTROL + HK

DC/DC CONVERTERS

SPIRE INTERNAL DIGITAL INTERFACES (1)



SPIRE INTERNAL INTERFACES (2)

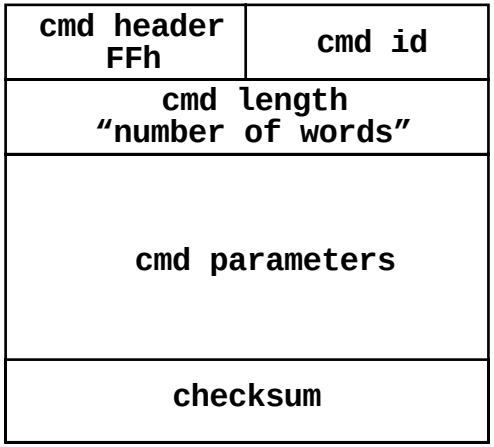
• Low Level Command Interface

➔ Basic command transmission to DRCU (for immediate execution)

- **DRCU sub-systems on / off commands**
- **DRCU sub-systems configuration (by means of parameters)**
- **DRCU software upload / download**
- **Low sampling rate HK (sub-systems status / FPU operating parameters / DRCU S/W status / …)**
- **Command Acknoledge (transmission error report)**

➔ Proposed Protocol :

- **Asynchronous UART based interface**
- **TBD baud rate (value compatible with computer serial port baud rate ≤ 115 kbps)**
- **8 bits of data + 1 stop bit + 1 parity bit**
- **RS 422 compliant electrical standard**
- **Command Structure definition :**

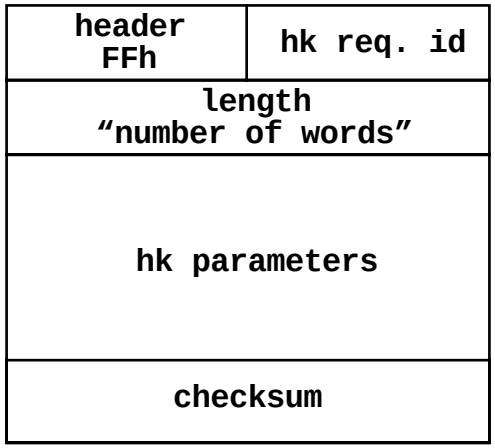


Note : structures are 16-bit wide

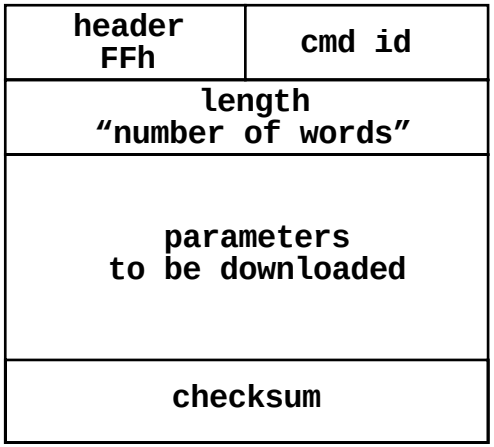
SPIRE INTERNAL INTERFACES (3)

Comment : • **cmd parameters field has a cmd id dependant length (between 0 and TBD max)**
• **header field is implemented for synchronisation purpose**

→ HK structure definition :



→ Response structure definition :



Example : Processor Memory dump

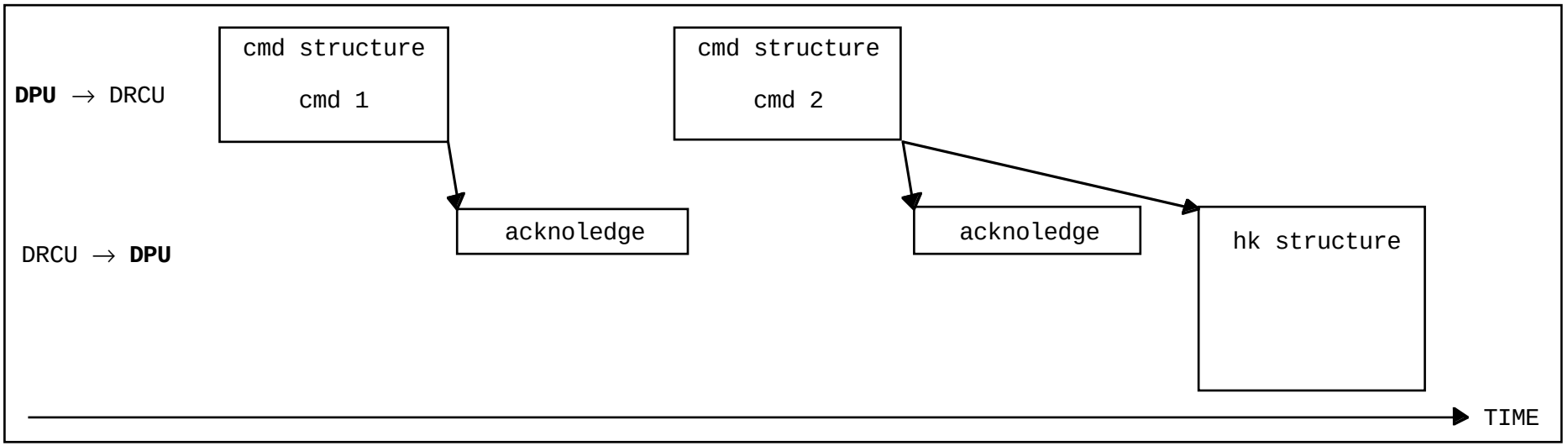
SPIRE INTERNAL INTERFACES (4)

→ **Acknoledge definition :**

ack. code	cmd id

ack. code : **positive acknoledge**
negative acknoledge (report a transmission error : parity or checksum)

Example :



Note : cmd 2 = HK request cmd

SPIRE INTERNAL INTERFACES (5)

• Scientific Data Interface

→ Reduced Detector Data transmission from the DRCU to the DPU
→ Fast Sampling HK
→ Average data rate : 100 kbps (TBC) / Peek data rate : 800 kbps (TBC)

➔ Proposed Protocol

- **Synchronous uni-directionnal serial interface**
- **Contents DATA + CLOCK + STROBE lines**
- **RS 422 compliant electrical standard**
- **20 bit words are transmitted via the DATA line :**

  2 bits for data type (image header, pixel data, ~~Fast~~ HK, …) + 18 bits for data field

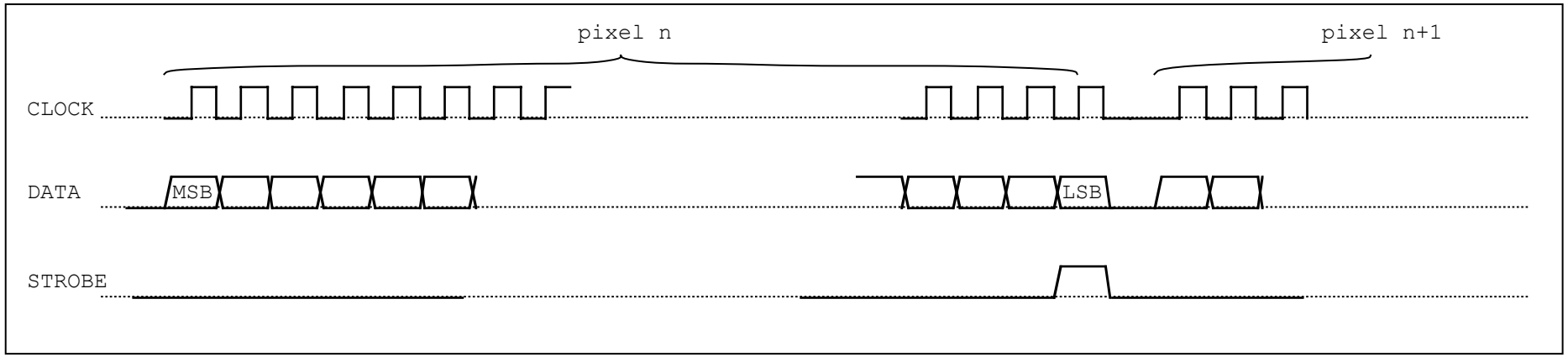
- **3 types of DATA are defined :**

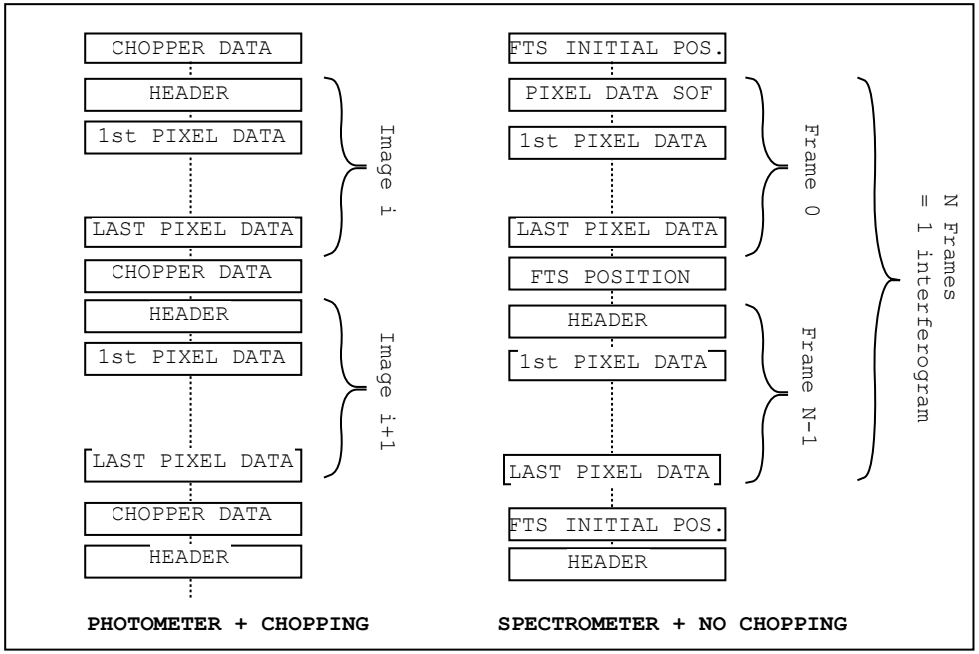
  HEADER (id = 11) : 2 bits for array identification + 1 bit for instrument mode + …
  PIXEL DATA (id = 00) : 1 bit for glitch flag + TBD bits for pixel data
  FAST HK (id = 10) : 2 bits for HK id. (FTS position, BSM Position, …) + 16 bits for HK parameter

A HEADER is sent before the first PIXEL DATA of each image.
FAST HK can be interleaved with PIXEL DATA

SPIRE INTERNAL INTERFACES (6)

→ DATA TIMING





SPIRE - DRCU INTERNAL INTERFACES (1)

DRCU INTERNAL INTERFACES

PHILOSOPHY :

1. **DRCU Controller and sub-systems interface only by means of digital lines**
2. **Sub-system commanding is based on a synchronous serial protocol**
3. ~~**Sub-systems low sampling HK are collected via this command interface**~~
4. **Detector data and fast sampling HK are collected using a parallel bus (see daisy chain description)**
5. **Synchronisation lines link sub-systems when required in term of accuracy (e.g. FTS and Array Controllers)**

• DETECTOR DATA AND FAST SAMPLING HK

→ Parallel data bus
→ DAISY chained strobes enable access to a large number of analog channels (up to 300 for JPL option) while minimising interconnection complexity :

3 SIGNALS (READ+CLOCK+RESET/SELECT) INSTEED OF 10 (ADDRESS + SELECT)

→ A FPGA contains all the needed functions to interface sub-systems and VME CPB interface

→ The present CPB design supports up to 700.000 32-bit transfer per second

→ If required (use of serial A→D converter) the interface FPGA supports both parallel and serial data transfer

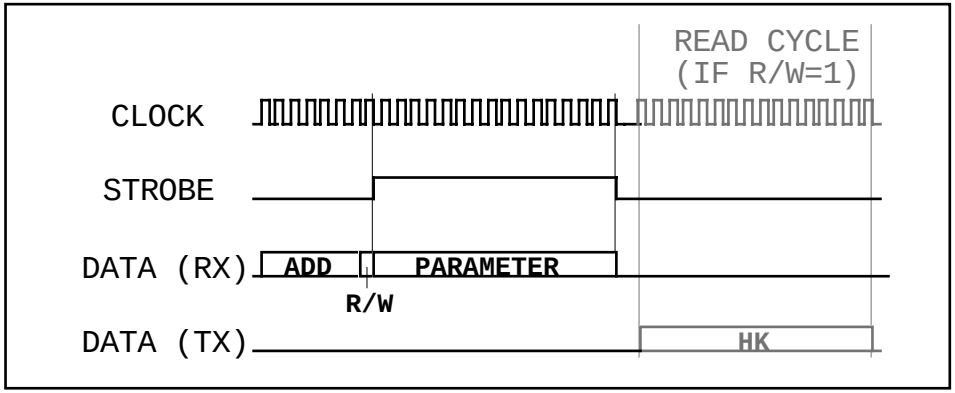
→ Each sub-system is individually selected by one or more reset/select lines

SPIRE - DRCU INTERNAL INTERFACES (2)

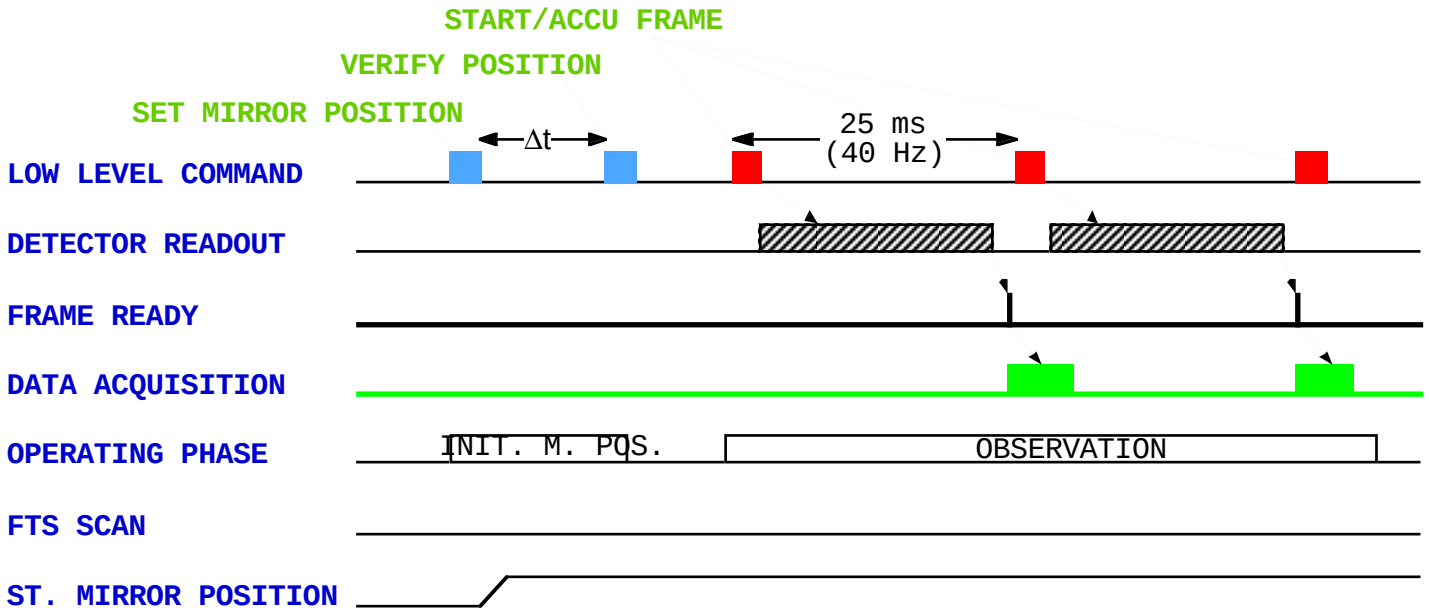
• SUB-SYSTEM PARAMETER LOADING AND SLOW SAMPLING HK

→ **Bidirectionnal serial synchronous line**
→ **Easy to handle by hardware**
→ **7-bit parameter address**
→ **16-bit parameter value (read/write mode)**
→ **Each sub-system is individually selected by one or more "STROBE" lines**

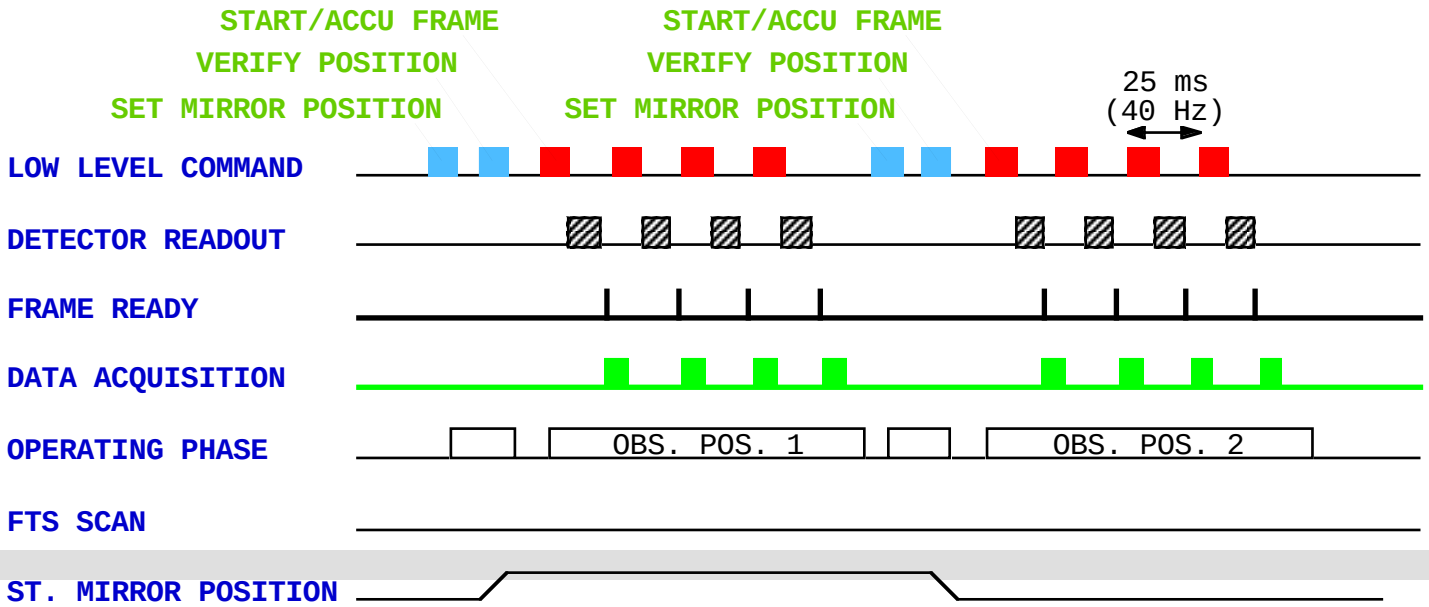
→ **DATA TIMING :**



SPIRE - DRCU INTERNAL INTERFACES - TIMING DIAGRAM (1)

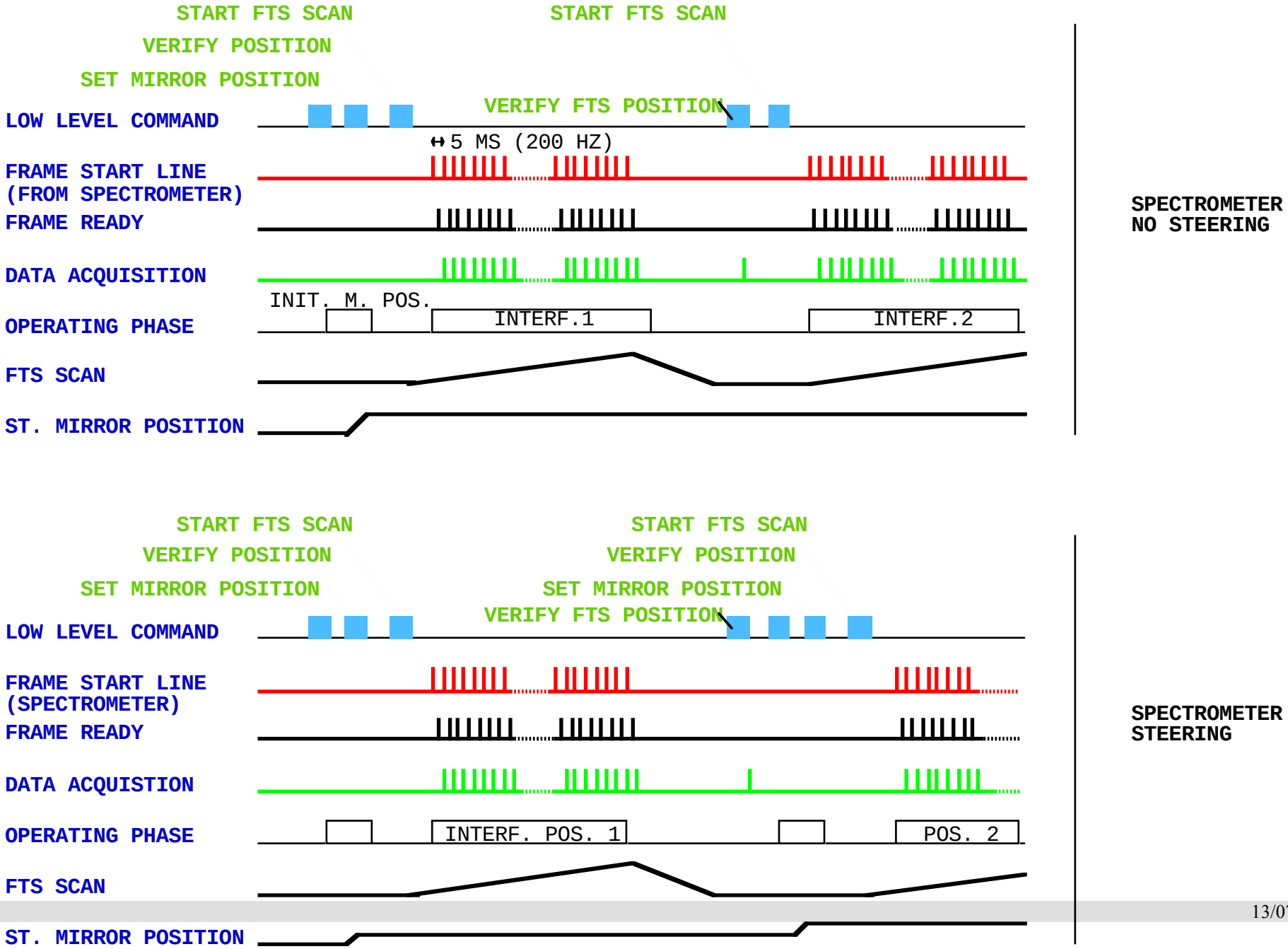


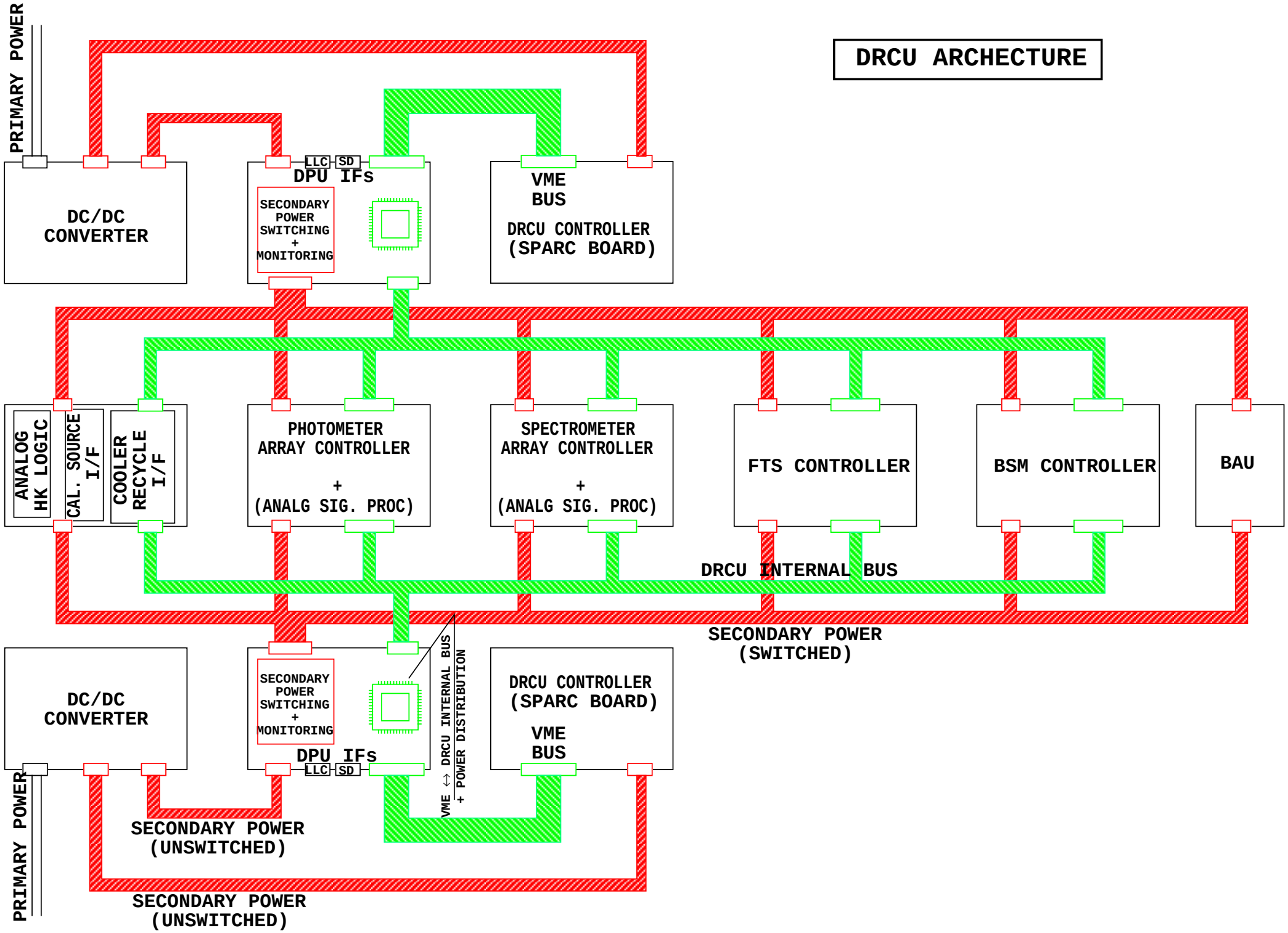
PHOTOMETER
NO CHOPPING



PHOTOMETER
CHOPPING

SPIRE - DRCU INTERNAL INTERFACES - TIMING DIAGRAM (2)




DRCU ARCHECTURE
PRIMARY POWER
DC/DC CONVERTER
LLC
SD
DPU IFs
SECONDARY POWER SWITCHING + MONITORING
VME BUS
DRCU CONTROLLER (SPARC BOARD)
ANALOG HK LOGIC
CAL. SOURCE I/F
COOLER RECYCLE I/F
PHOTOMETER ARRAY CONTROLLER + (ANALG SIG. PROC)
SPECTROMETER ARRAY CONTROLLER + (ANALG SIG. PROC)
FTS CONTROLLER
BSM CONTROLLER
BAU
DRCU INTERNAL BUS
SECONDARY POWER (SWITCHED)
VME ↔ DRCU INTERNAL BUS + POWER DISTRIBUTION
SECONDARY POWER (UNSWITCHED)
SECONDARY POWER (UNSWITCHED)

SPIRE - SIMULATORS (1)

• DRCU SIMULATOR

→ **PURPOSE :**

- **DPU TEST**
- **DPU VERIFICATION (AFTER DELIVERY TO CEA)**
- **AVM (TBC)**

→ **RESPONSIBLE : SWEDEN OBSERVATORY**

DEFINITION :

THIS EQUIPMENT WILL BE IN CHARGE OF SIMULATING THE DRCU DIGITAL FUNCTIONS

THAT MEANS :

- **LOW LEVEL COMMAND I/F IMPLEMENTATION**
- **LOW LEVEL COMMAND PROTOCOL EMULATION**
- **SCIENTIFIC DATA I/F IMPLEMENTATION**
- **+28 V I/F IMPLEMENTATION (DRCU DC/DC PRIMARY POWER + ON/OFF LINE)**
- **LOW LEVEL COMMAND EXECUTION**
- **SIMULATE ANALOG DATA (DETECTOR, TEMPERATURE, …)**
- **FAILURE SIMULATION (FOR DPU AUTONOMY/SAFETY FUNCTIONS TEST)**
  - **LOW LEVEL CMD TRANSMISSION**
  - **ANALOG PARAMETERS OUT OF RANGE (INTO HK DATA)**
  - **SUB-SYSTEMS STATUS**
  - **DRCU S/W REBOOT (TRIGGERED BY WATCHDOG)**

NOTE : DRCU REDUNDANCY WILL BE SIMULATED
SELECTED DETECTORS WILL HAVE NO IMPACT ON DEFINITION (TBC)

SPIRE - SIMULATORS (2)

• INSTRUMENT SIMULATOR

→ PURPOSE :

- **DRCU (+ BAU) INTEGRATION AND TEST**
- **WARM ELECTRONICS (INCLUDING OBSW) INTEGRATION AND TEST**
- **WARM ELECTRONICS VERIFICATION (AFTER DELIVERY TO RAL)**

→ RESPONSABLE : CEA / SAP

DEFINITION :

THIS EQUIPMENT WILL BE IN CHARGE OF SIMULATING THE INSTRUMENT FUNCTIONS LOCATED INSIDE THE VACUUM TANK (+ BAU)

THIS INCLUDES :

- **SIMULATE DETECTOR I/F**
- **SIMULATE DETECTOR DATA**
- **SIMULATE MECHANISMS :**
  - **• FTS**
  - **• STEERING MIRROR**
  - **• SHUTTER**
- **SIMULATE CALIBRATION SOURCES I/F**
- **SIMULATE COOLER I/F**
- **SIMULATE THERMAL PROBE**
- **POSSIBLITY TO SIMULATE FAILURES (MECHANISM & TEMPERATURE)**

NOTE : SELECTED DETECTOR HAVE STRONG IMPACT ON DEFINITION
POSSIBLITY TO USE EXISTING SIMULATORS (E.G. FTS MECHANISM) TO BE INVESTEGATED
THIS EQUIPMENT WILL SUPPORT BOTH DIRECT VACUUM TANK CONNECTION & BAU I/F

SPIRE - SIMULATORS (3)

• LOCAL TEST UNIT

→ PURPOSE :

- **DRCU INTEGRATION & TEST**
- **DRCU VERIFICATION (BEFORE INTEGRATION WITH DPU)**
- **CQM (TBC)**

→ RESPONSIBLE : CEA / SAP

DEFINITION :

THIS EQUIPMENT WILL BE IN CHARGE OF SIMULATING THE DPU INTERFACES WITH THE DRCU AND TO PROVIDE FACILITIES TO GENERATE LOW LEVEL COMMANDS AND TO MONITOR/RECORD SCIENTIFIC DATA & HOUSEKEEPING.
IT WILL BE ALSO IN CHARGE OF SIMULATING DIRECT S/C INTERFACES (E.G. : + 28 V)

THAT MEANS :

- **LOW LEVEL COMMAND INTERFACE IMPLEMENTATION**
- **LOW LEVEL COMMAND PROTOCOL EMULATION**
- **SCIENTIFIC DATA INTERFACE IMPLEMENTATION**
- **SCIENTIFIC DATA REAL TIME MONITORING**
- **SCIENTIFIC DATA RECORDING**
- **HOUSEKEEPING MONOTORING / RECORDING**
- **+28 V I/F IMPLEMENTATION (DRCUY DC/DC PRIMARY POWER + ON/OFF LINE)**
- **FAILURE SIMULATION (LOW LEVEL COMMAND INTERFACE)**

SPIRE - SIMULATORS (4)

• FACTORY SUPPORT EQUIPMENT

PURPOSE :

- **ELECTRONIC BOARDS DEVELOPMENT SUPPORT**
- **ELECTRONIC BOARDS VERIFICATION BEFORE ASSEMBLING INTO SUB-SYSTEMS**

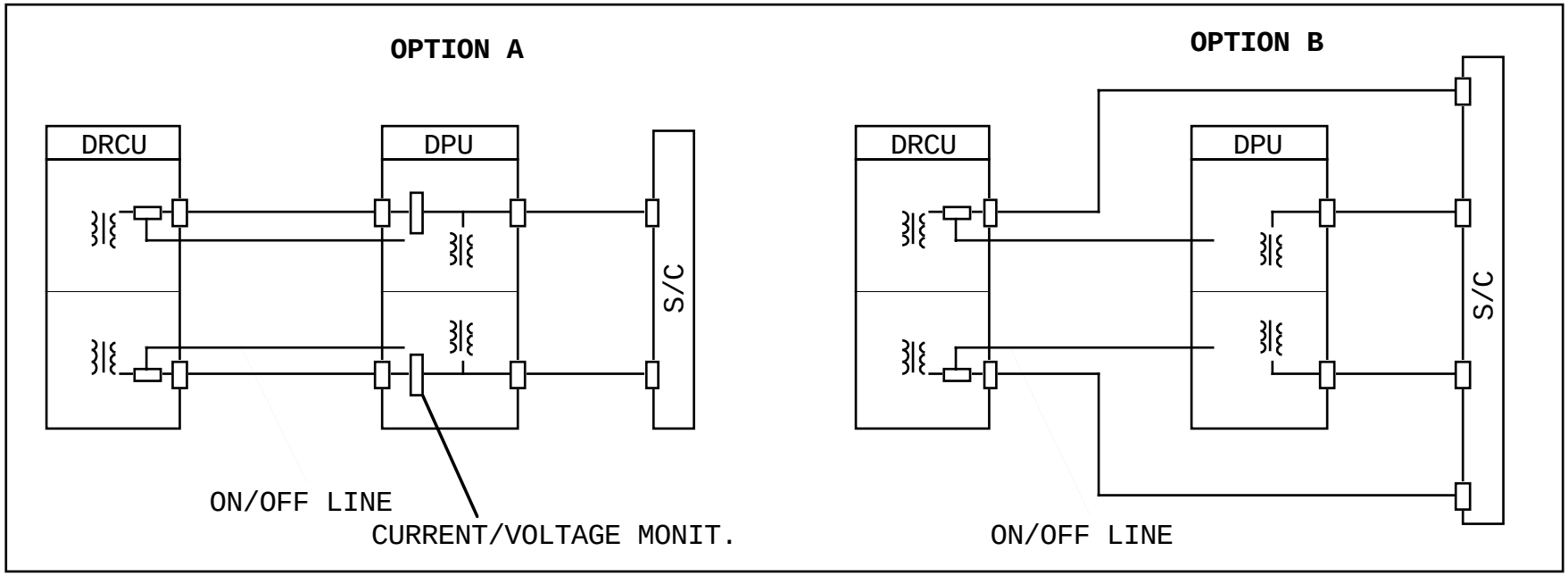
→ RESPONSIBLE : INSTITUTES / MANUFACTERS INVOLVED IN ELECTRONICS DEVELOPMENT

SPIRE - POWER DISTRIBUTION

• THE WARM ELECTRONICS INCLUDES 2 DC/DC CONVERTERS (+ 2 FOR REDUNDANCY CONSIDERATION)

- **1 (+1) IS LOCATED IN THE DPU**
- **1 (+1) IS LOCATED IN THE DRCU**

→ AT LEAST TWO POSSIBLE INTERCONNECTION OPTIONS BETWEEN CONVERTERS AND S/C



<u>OPTION A</u>
PRO : DPU MONITORS DRCU PRIMARY POWER
CON : NUMBER OF CONNECTOR (LESS RELIABLE)

SPIRE - POWER DISTRIBUTION

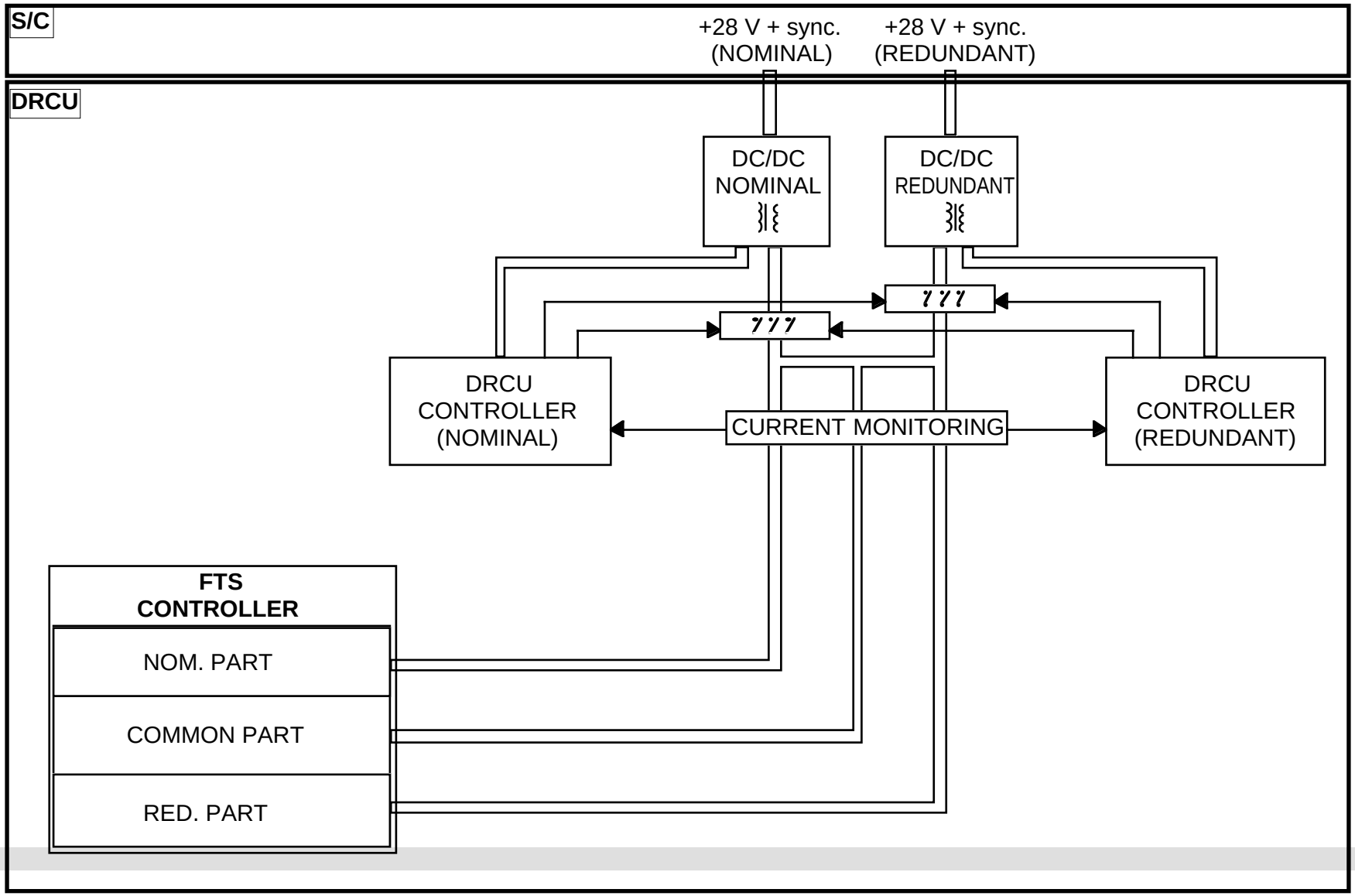
OPTION B

PRO : **NUMBER OF CONNECTOR (MORE RELIABLE)**
DPU SIMPLIFICATION

CON : **NO DRCU PRIMARY POWER MONITORING**

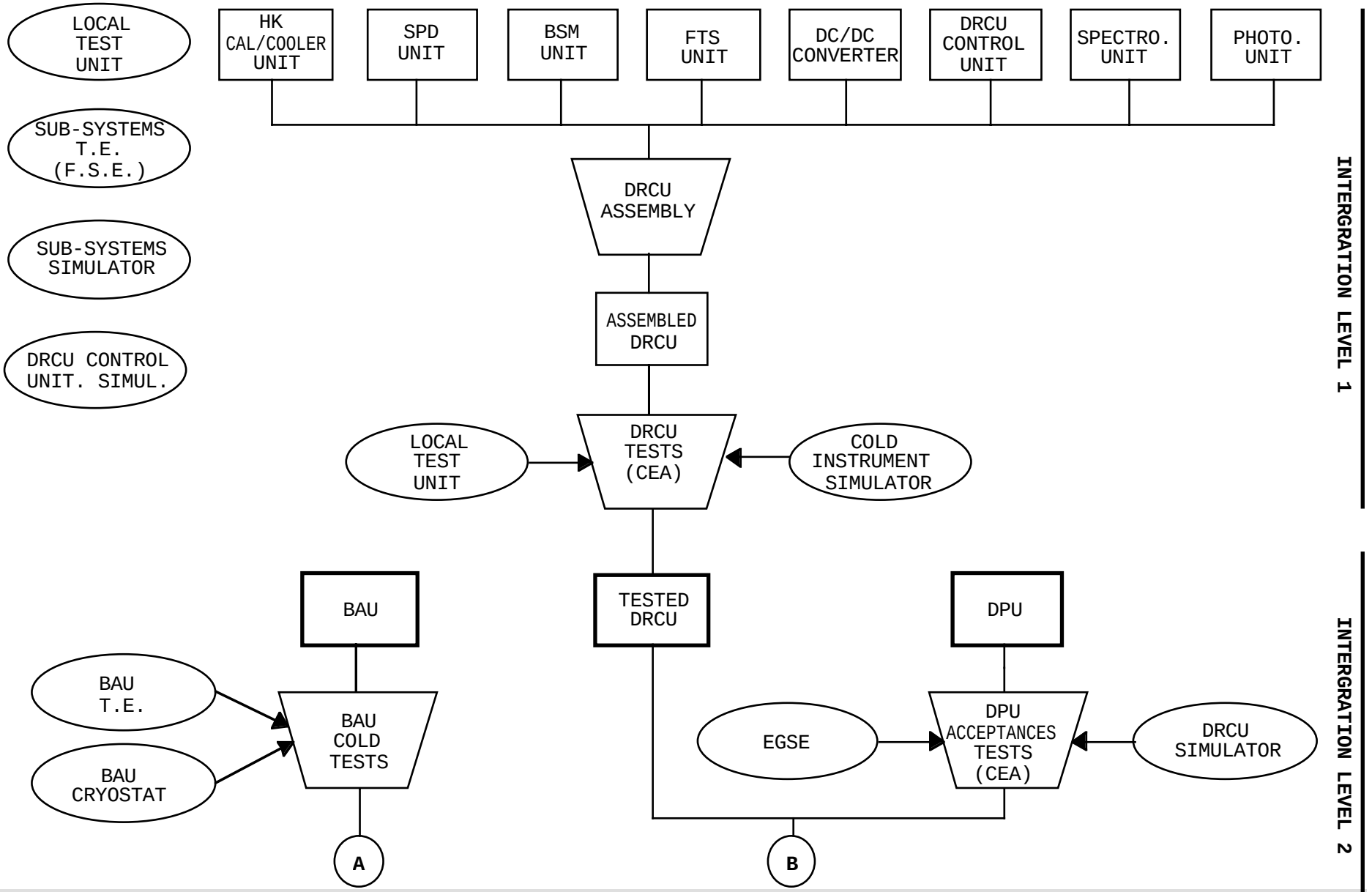
SPIRE - POWER DISTRIBUTION

DRCU SECONDARY POWER DISTRIBUTION
FTS EXAMPLE



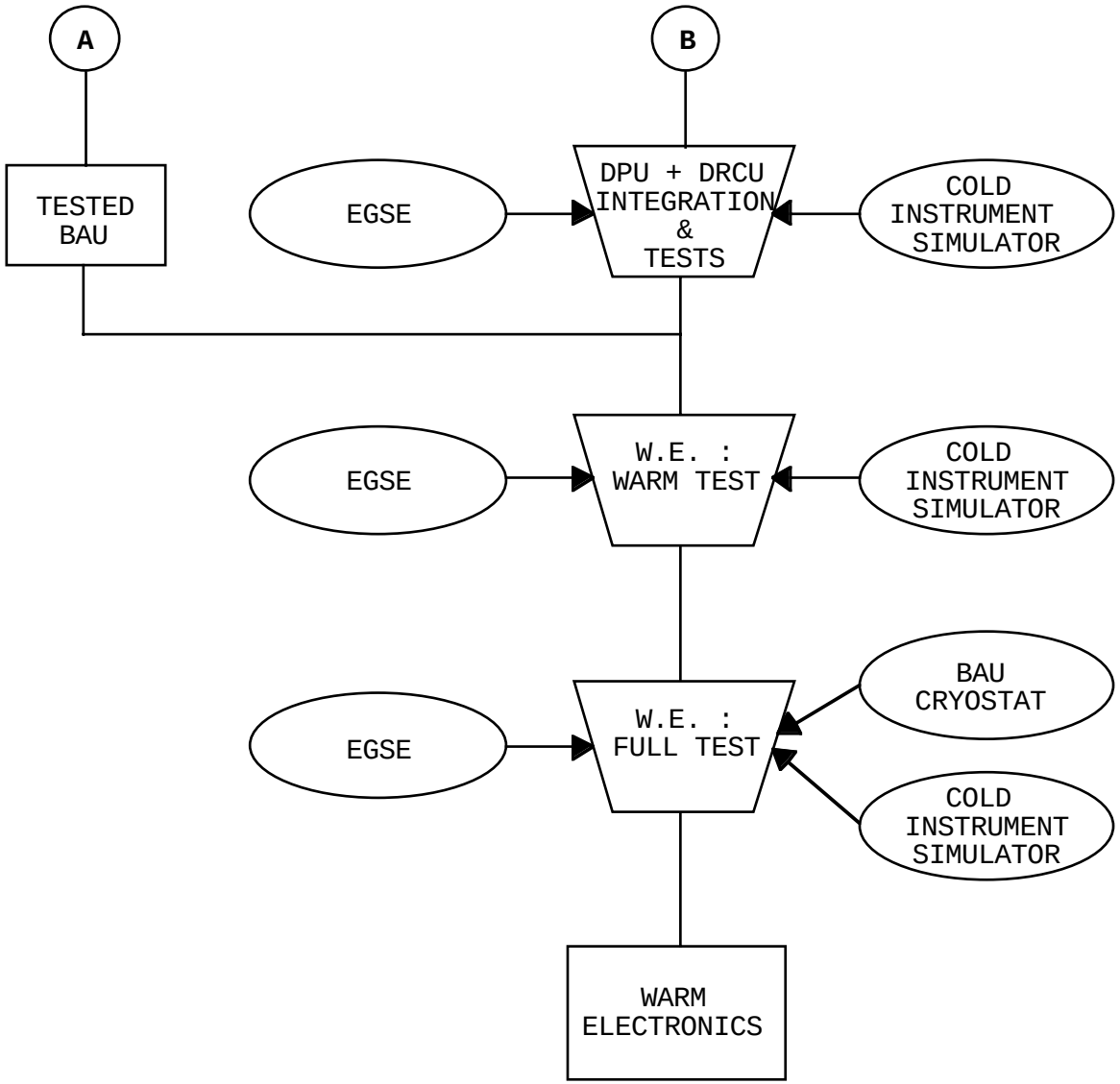
SPIRE - WARM ELECTRONICS AIV PLAN (1)

FM AIV PLAN (1)



SPIRE - WARM ELECTRONICS AIV PLAN (1)

FM AIV PLAN (2)



INTERGRATION LEVEL 2

LEVEL 3

SPIRE - HANDLING OF AUTONOMY/SAFETY (1)

This subject is covered by the following documents :

- **"Autonomous Decision Mode" - L.Vigroux - 22/06/99**
- **"Answer to the questions of K.King about autonomous mode requirements" - L.Vigroux - 24/06/99**

→ In order to simplify the autonomous mode it is proposed to share the related functions in the DRCU and DPU
→ The basic concept is based on a hierarchical monitoring of critical parameters (e.g. supply current) of sub-systems :

<u>DRCU to sub-systems :</u>

Each sub-system control unit should be able :

- **to be switched off/on by the DRCU controller**
- **to detect internal errors and send error codes to the DRCU controller**

The DRCU should be able

- **to switch on/off each system**
- **to monitor the power supply of each sub-system**
- **to compare these value with thresholds**
- **to handle error by assigning them a severity grade**
- **to communicate to the DPU the error code and the severity grade**
- **to set up a status (VETO) active or inactive for each sub-system which enable or disable commands addressed to it**

SPIRE - HANDLING OF AUTONOMY/SAFETY (2)

DRCU controller health and safety :

- **A watchdog resets the processor in case of software/hardware failure. In this case the sub-system power supply are switched off.**
- **Check before execution the validity of a command by performing a checksum**
- **Verify that individual commands have been executed**
- **In case of unrecoverable memory error the wrong memory contents and its address are transmitted to the DPU**

DPU to DRCU :

The DPU should be able to :

- **to switch on/off the DRCU power supply**
- **to monitor the DRCU power supply**
- **to compare this value with thresholds**
- **to handle error code by assigning a severity grade**

In case of failure the SPIRE instrument is switched in ON mode.

→ The proposal avoids this monitoring to be mode dependant by decentralizing the monitoring as close as possible to each sub-system.

S/C to DPU/DRCU :

The S/C should be able to switch off SPIRE in case of unexpected power consumption

July 19-20 Agenda

1. Action review.

2. Outcomes of PDR 1 and possible impact on WE.

3. WE Requirements
   We need to get together a plan for providing a set of requirements that can be reviewed at the PDR. This includes
   a. WE requirements
   b. DPU requirements (including OBSW)
   c. DRCU requirements (including OBSW)

4. WE architecture and internal I/F
   (DPU-DRCU, DRCU-WE subsystems I/F philosophy).

   The framework is the Option C (DPU+DRCU) which has been agreed by the SPIRE Steering Group.

   This topics includes discussions about:

   a. the respective roles of the DPU and the DRCU regarding both h/w and s/w aspects e.g.:
      i. Handling of autonomy/safety.
      ii. Redundancy (subsystems and I/F).
      iii. Power supply internal distribution.
      iv. Command distribution/handling.
      v. Data handling (science and housekeeping).

   b. FPU subsystems requirements and I/F:
      i. Detector readout system.
      ii. FTS.
      iii. Steering mirror.
      iv. Shutter.
      v. Calibration sources.
      vi. Cooler.
      vii. Thermometry.
      viii. H/K.

5. External Interfacing issues
   a. Main Power Supply (DC/DC specification).
   b. T/C interface
   c. TLM interface
   d. Grounding scheme.
   e. Other interfaces (e.g. operational interaction with S/C)

July 19-20 Agenda (cont.)

6. S/W development:
   a. at IFSI.
   b. at SAp.

7. Support Equipment
   a. Test equipment:
      i. EGSE.
      ii. Local Test unit.
      iii. Factory Support Equipment.

   b. Simulators:
      i. Definition of the various simulators and their I/F.
      ii. Worksharing.

8. Development Plan:
   a. Worksharing.
   b. Models definition.
   c. Resources.
   d. Development schedule.
   e. AIV flow and schedule

9. Quality requirements.

10. PDR preparation:
    a. What is expected for the PDR.
    b. Worksharing.
    c. Short term schedule.

11. AOBs

12. End of meeting.

Pending Actions

A - From WEG meetings

- **WEG meeting #2 - Saclay - May 6, 1999**

Action 1	JLA	asap	Open	Circulate the preliminary draft of the WE Requirement Document.

- **WEG meeting #1 - Saclay - March 24-25, 1999**

Action 1	KK		Open	to issue a note stating both objectives and expected inputs for the Sept. 99 PDR.
Action 5	System team		Open	to perform a risk analysis.
Action 6	KK		Open	to boost the SPIRE QA activity.
Action 7	RAL/GD	asap	Open	Issue the SPIRE Quality Requirements
Action 9	SAp/FL	04/99	Open	Produce the first issue of the Warm Electronics Product Assurance Plan
Action 10	SAp/FL	05/99	Open	Produce the first issue of the Warm Electronics Quality requirements.
Action 12	WE Inst.	06/99	Open	to produce their Product Assurance Plans.
Action 14	System Team		Open	to produce redundancy requirements at system level.

Pending Actions

B - From other meetings

- **Warm electronics & S/W working group splinter meeting (SPIRE Consortium Meeting - RAL - Dec. 1-2, 1998)**

Action 2	BMS	20/01/99	Open	To respond the essential input request list.
Action 3	KJK	15/12/99	Open	To provide a Development Plan containing AIV information as well..

- **SAp/IFSI meeting - IFSI - Feb. 16, 1999**

Action 2	IFSI & SAp	15/03/99	Open	Comment the preliminary draft of the SPIRE Development Plan.

- **CWG #4 - Feb. 10, 1999**

Action 3	JLA	20/05/99	Open	SPIRE to confirm allocation of responsibilities for SPU h/w & s/w implementation.
Action 5	IRC (*)	06/05/99	Open	Generate and Co-ordinate "requirements" on instrument commanding & verification.
Action 7	RC/JLA	06/05/99	Open	Define OBSW related milestones and activities till end 99.
Action 8	KK	06/05/99	Open	Provide comments to Appendix 1 of Mission Operation Scenario.

- **CWG #3 - Feb. 3, 1999**

Action 3	KK	19/05/99	Open	Provide plans for ILTs, indicating required deliveries (S/C simulator, CCE,...)
Action 8	KK	30/04/99	Open	Comment on PACS RTA requirements used for SCOS testing.
Action 9	KK	05/03/99	Open	To supply estimates of manpower available for RTA related activities.

- **CWG #1 #2 - March 3, 1999**

Action 1	RO	25/05/99	Open	Submit list of needed common parts.

- **CWG #1 #2 - July 2, 1999**

Action 2	All teams	01/11/99	Open	to define any need of special timing signals, in particular their accuracy, from the OBDH.
Action 1	All teams	15/09/99	Open	to update their part list.

Products & Tests vs. Models (1)

		Resp.	EM	AVM	CQM	PFM	FS	Comments
WE UNITs								
DPU								
	Processing Unit							
	CPU Board	IFSI	1	1	2	2	(2/3)	FS shared with other Inst.
	Memory	IFSI	1	1	2	2	(2/3)	
	S/W	IFSI	V0	V1	V2	V flight	V flight	
	Component Grade		Std	Std	Ext	Qual	Qual	
	Power Supply	IFSI	1	1	2	2	(2/3)	
	Mechanics							
	DPU Box	IFSI	1	1	1	1	(1/2)	FS shared with other Inst.
	Connectors	IFSI	X	X	X	X	X	
DRCU								
	Processing Unit							
	CPU Board	SAp	1	1	2	2	1	MMS SPARC Based Board
	Memory	SAp	xMo	xMo	2*(xMo)	2*(xMo)	xMo	8 to 12Mb on CPU Board
	S/W	SAp	V0?	V1?	V2?	V flight	V flight	
	Component Grade		Std	Std	Ext	Qual	Qual	
	Subsystem							
	Detector Readout	SAp	1	-	1	1	1	
	FTS Control	LAS	1	-	2	2	1?	
	Cryo Control	SAp	1	-	2	2	1?	
	Steering Mirror Control	SAp	1	-	2	2	1?	
	Calib. Source Control	SAp	1	-	2	2	1?	
	H/K readout	SAp	1	-	2	2	1?	
	Shutter Control	SAp	1	-	2	2	1?	
	Component Grade		Std	-	Mil	Qual	Qual	
	Power Supply	SAp	1	1	2	2	1	
	Mechanics							
	DRCU Box	SAp	1	-	1	1	1	
	Connectors	SAp	X	X	X	X	X	
BAU								
	Electronics							
	Buffers	SAp	1		1	1	1	
	Temp. Regulation	SAp	1		1	1	1	
	Component Grade		Std		Mil	Qual	Qual	
	Mechanics							
	BAU Box	SAp	1	2	2	2	1	
	Connectors	SAp	X	X	X	X	X	
R/F Filters								
	Filters	SAp	1		1	1	1	
	Component Grade		Std		Mil	Qual	Qual	
	Mechanics							
	R/F Filter Box	SAp	1	2	2	2	1	
	Connectors	SAp	X	X	X	X	X	
Harness								
	DPU to SPU	SAp	1	1	2	2	1	
	DRCU to BAU	SAp	1	-	2	2	1	

Products & Tests vs. Models (2)

		Resp.	EM	AVM	CQM	PFM	FS	Comments
Simulators								
FPU Simulator								Developed by CEA/SIG
	Electronics	SAp	1			1		
	Mechanics	SAp	1			1		
DRCU Simulator								
	H/W							
	Station	SO	2	1				
	Elect. Interface	SO	2	1				
	S/W							
	Simulation S/W	SO	V0	V1	V2			
Test Facilities								
EGSE								
	Station	Can.	4					
	Elect. Interface	Can.	4					
Local Test Unit								Developed by CEA/SIG
	Station		2					
	Elect. Interface		2					
Test Facility S/W								
	OBDH Interface Emulation	SIG	1					Local Test Unit
	RTA Common	RAL	1					
	RTA Specific	RAL	1					
	QLA	RAL	1					
	Telecom. Generation Tool	RAL	1					
Tests								
	EMC							
	Thermal Vacuum							
	Vibration							

Milestones

Milestones	Resp.	Date	Comments
EM			
DPU delivery to SAp	IFSI		
DRCU Simulator delivery to SAp	SO		
FPU simulator availability	SAp		
EM ready for use	SAp		
AVM			
EGSE availability	Can.		
DPU delivery to SAp	IFSI		
Delivery to RAL	SAp		
Delivery to ESA	RAL	04/2003	
CQM			
DPU delivery to SAp	IFSI		AVM unit (redundancy?)
Delivery to RAL	SAp		
Delivery to ESA	RAL	04/2003	
PFM			
DPU delivery to SAp	IFSI		
Delivery to RAL	SAp		
Delivery to ESA	RAL	07/2004	
FS			
DPU delivery to SAp	IFSI		
Delivery to RAL	SAp		
Delivery to ESA	RAL		

Development Plans

- **Each Institute is requested to provide a development plan.**

- **Proposed Template** (use Word and MSProject for the schedule)**:**

1. Introduction.
1.1 Purpose of the document.
1.2 Scope of the project.
1.3 Assumptions.
1.4 Document Overview.
2. Subsystem Description
2.1 Overall design.
Overall description of the products concerned
2.2 Interfaces.
Overall description of the interfaces with the other subsystems (electrical, mechanics,...)
3. Development.
3.1 Model Definition.
Description of the products to be produced and delivered for each model
3.1.1 EM.
3.1.2 AVM
3.1.3 CQM.
3.1.4 PFM.
3.1.5 FS.
3.2 AIV.
3.2.1 AIV Procedures.
Description of AIV steps using a diagram providing indication on both the steps, test means and responsibilities (who is carrying out the task).
3.2.2 Test Equipments.
Description of the test to be performed and the necessary equipment. Per subsystem: test performed, location, needed facilities, availability (available, to be built, to be purchased,...)
3.3 Product Tree.
Complete product tree including the test equipment and simulators.
3.4 Worksharing.
Description of the workpackages and who is responsible.
3.5 Schedule.
3.5.1 Driving Milestones.
Description of the Main Milestone driving the realisation of the subsystem including the availability of external products or test equipments.
3.5.2 Development Schedule.
Complete Schedule (Gant & PERT) including responsibilities for each task and the estimated resources (manpower).