XMM's Data-Handling Subsystem

A. Karlsson
XMM Project, ESA Directorate for Scientific Programmes,
ESTEC, Noordwijk, The Netherlands

Scaling the XMM data-handling system to the specific needs of the mission has resulted in a flexible and relatively uncomplicated On-Board Data-Handling (OBDH) Subsystem. The XMM spacecraft is operated in real-time and is fully compliant with the ESA packet data-handling standards. Packets are of variable length and presented to the OBDH when available. The OBDH telemetry/telecommand cycles are asynchronous to the instrument and subsystem cycles. The telemetry rates can easily be reallocated to cope with unforeseen spacecraft configurations, e.g. lost telemetry sources. The absence of requirements for automatic reconfiguration has contributed to the reduced subsystem complexity.

XMM's OBDH Subsystem

The On-Board Data-Handling (OBDH) Subsystem for XMM is implemented in three internally redundant physical units: the Central Data Management Unit (CDMU) and two Remote Terminal Units (RTUs). The CDMU and one RTU are located on the Service Module of the spacecraft. The second RTU is installed on the Focal-Plane Assembly (FPA). In addition to the RTUs and the CDMU, the OBDH includes six Data Bus Units (DBUs), which provide the scientific instruments with a digital interface to the data-handling services.

The data processing in the CDMU is performed by the Central Terminal Unit (CTU) based on a MIL-STD-1750 microprocessor with 256 kwords of RAM. The packet handling functions, i.e. telemetry frame generation and telecommand frame decoding, are implemented in standard ASICs developed under ESA contracts, Virtual Channel Assemblers (VCAs), Virtual Channel Mutiplexers (VCMs) and Packet Telecommand Decoders (PTDs).

The users and all of the OBDH units are interconnected by the ESA OBDH bus, comprising a redundant set of interrogation bus and mono-directional response bus. The OBDH bus can transfer, depending on the command rate, approximately 200 kbps of telemetry data, which is roughly three times the XMM downlink data rate of 69.4 kbps (source-packet level).

The bus activities are governed by the Polling Sequence Table (PST), which has a 4 sec cycle and is subdivided into 184 windows of equal duration. These windows are programmable to be either: telecommand distribution, telemetry acquisition from intelligent bus terminal, telemetry acquisition from RTU, or memory patch or dump of a memory on the bus.

The default Polling Sequence Table on XMM has 36 windows for telecommand distribution, 129 windows for telemetry acquisitions from the instruments, 9 windows for AOCS telemetry, 7 windows for RTU acquisitions, and some spares.

The bus is used for both low-level word-oriented data acquisitions and control, as well as for high-level packet transfers. The low-level protocol is used for data transfers to and from the RTUs, while the high-level protocol is used in the communication with the Packet Terminals.

Packet protocol

Packet Terminals are connected to the OBDH via dedicated Data Bus Units. The interface to the DBU, and thus the OBDH bus, is realised by an ASIC Remote Bus Interface (RBI). In order to have a common interface to all Packet Terminals, this specific RBI was imposed on all instruments and the Attitude and Orbit Control Subsystem (AOCS).

The RBI provides the CDMU with Direct Memory Access (DMA) to the processor memory of the Packet Terminals. In addition, the RBI accommodates registers for communication between the Packet Terminal and the OBDH and a register holding a copy of the onboard time.

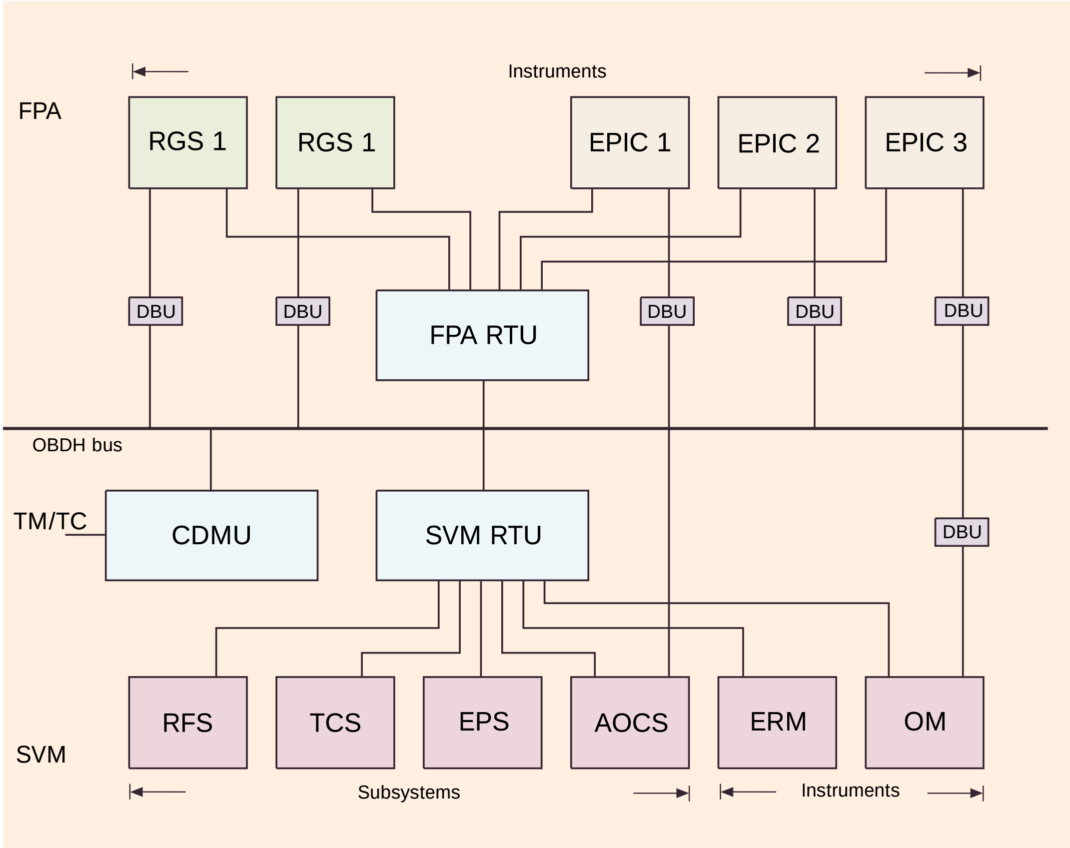


Figure 1. Architecture of XMM's OBDH Subsystem

Packet Terminals receive telecommand packets via DMA in a telecommand buffer in memory. When the OBDH has delivered a telecommand packet, the microprocessor of the Packet Terminal will receive an interrupt as a notification that the telecommand should be processed and removed from the buffer. By setting a flag, the Packet Terminal indicates that it is ready to receive a new telecommand packet.

Whenever the Packet Terminal has a telemetry packet to deliver, it indicates this by setting another flag. This flag is read by the OBDH at each poll of the Packet Terminal, as programmed in the Polling Sequence Table. When the OBDH detects that the flag is set, it will transfer the telemetry packet to a telemetry buffer in the CDMU and downlink it according to its priority.

In order to allow the users to design their scientific instruments without detailed knowledge of the OBDH interface and to fully exploit packet telemetry, the OBDH bus protocol is designed with Packet Terminals being asynchronous to the OBDH data acquisition cycle. This is made possible by datation of the telemetry packets and the direct memory access to the users' memories. The Packet Terminals are free to choose packet production profiles that match the scientific objectives of the instrument, within the constraints of data-rate allocation and the telemetry poll rate.

Autonomy

XMM is required to survive three days of ground-station outage. The autonomy on XMM is decentralised, meaning that all subsystems should manage their own survival and supply essential services to allow other subsystems to survive. Thus the central spacecraft autonomy function, often implemented in the data-handling subsystem, was not required for XMM. Recognising this has allowed a substantial reduction in design and test expenditure by limiting the OBDH autonomy to protection against OBDH internal failures.

Since all autonomous functions are implemented in the various subsystems without relying on services from the data-handling subsystem, it was permissible to cease data-handling services on the detection of an internal OBDH anomaly. As a result, whenever the OBDH detects an anomaly, e.g. an uncorrectable error in memory, it will terminate execution and enter a halt mode. The users of the OBDH services will detect that the OBDH has entered the halt mode by, for example, an accumulation of telemetry packets. The user has to take appropriate measures to operate without OBDH services until they are resumed.

Before entering the halted mode, the OBDH generates an error report describing the reason for the mode transition and forwards it to the telemetry system. Since the telemetry system is implemented in hardware, it does not require

the processor to be running to output the error report to the radio-frequency subsystem for down-linking to the ground. To improve the likelihood of an error message being seen by the ground, all error reports are stored on-board in protected memory, which is not affected by a resetting of the CDMU. However, with an internal CDMU error it cannot be guaranteed that this error report is generated, down-linked or stored.

To recover from OBDH halted mode, the ground activates prime or redundant CDMUs, and in either case it is possible to retrieve the error report from the protected memory.

Packet formats

The XMM spacecraft implements the ESA packet data-handling standards, with the following restrictions: (i) segmentation of source packets is not supported, and (ii) the lengths of telemetry packets and telecommand packets have been limited to 518 octets and 248 octets, respectively. It should be noted that 'grouping' is allowed for the science data telemetry packets.

Telemetry and telecomand packet types and subtypes, allowed on XMM, are specified in the Packet Structure Definition, derived from a draft version of the Packet Utilisation Standard. It defines a reduced set of packets tailored to the needs of the average ESA spacecraft, and XMM specifically.

The Packet Structure Definition identifies the packet types and subtypes as derived from the operations concept and the on-board functions required. Furthermore, it specifies the format of the packets, i.e. packet data field headers, identifiers and data records, and parameter formats including, for example, the format of the time and error control words. The following telemetry packet types are defined:

TM(1,x)	Periodic Telemetry Reports
TM(3,x)	Telecommand Verification Reports
TM(4,x)	Non-Periodic Telemetry Reports
TM(5,x)	Task Maintenance Reports
TM(6,x)	Memory Management Reports
TM(7,x)	Time-Tag Buffer Reports
TM(8,x)	On-Board Monitoring Reports
TM(9,x)	Telemetry Management Reports
TM(10,x)	On-Board Time Management Reports
TM(15,x)	Science Telemetry Reports

The following telecommand packet types are defined:

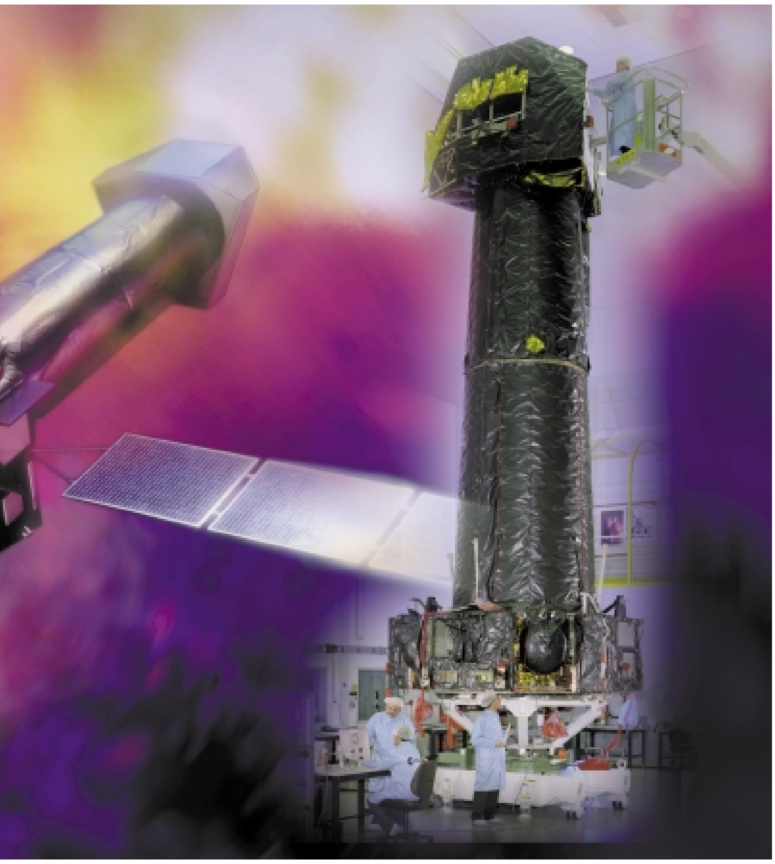
TC(2,x)	Device Commanding
TC(5,x)	Task Management
TC(6,x)	Memory Maintenance
TC(7,x)	Time-Tag Commanding
TC(8,x)	On-Board Monitoring
TC(9,x)	Telemetry Management
TC(10,x)	On-Board Time Management
TC(11,x)	Retrieval Stored Non-Periodic Packets
TC(13,x)	Test Telecommands

Obviously, the users need to specify the packet formats corresponding to their own needs. This is done via packet definitions produced by the users based on the types and formats provided in the Packet Structure Definition.

Telemetry system

The main scientific instruments on board XMM have a data-generation rate proportional to the strength of the observed X-ray target. This will result in time-variable demands from the users on the data-handling services. Responding to this inherent user need, it was decided to implement packets with variable length.

In having variable length packets, the telemetry service to the Packet Terminals is specified by: (i) the number of polls per bus cycle, where polls are evenly distributed over the bus cycle



with a certain tolerance, and (ii) the maximum amount of data that may be transferred from a Packet Terminal during a bus cycle. The former is specified in the Polling Sequence Table, and the latter in the Bit Rate Allocation Table. Both tables are programmable from the ground.

To allow the Packet Terminals to produce short telemetry packets, they are polled more frequently than strictly necessary. If all Packet Terminals deliver maximum-size packets at all polls, the total data rate would be 142 kbps, while an average of only 65.6 kbps from the Packet Terminals can be accommodated on the down link. It is therefore essential that the output data rate from the Packet Terminals be controlled.

The control is performed after each telemetry packet read-out. The CDMU verifies how much data it has received from the Packet Terminal during the current bus cycle and compares it against the telemetry threshold of the Packet Terminal. Whenever the threshold has been exceeded, the OBDH will cease acquisition of any further packet from that particular Packet Terminal until the end of the current bus cycle. Note that the read packet is accepted for transfer by the CDMU. The maximum excess data delivered by a Packet Terminal will be a fraction of a telemetry packet.

At the beginning of the next bus cycle, the CDMU will calculate the new threshold by subtracting the excess data of the previous bus cycle from the nominal telemetry allocation.

However, if the Packet Terminal provides less than its allocation in one bus cycle, it will not be credited for the loss of bandwidth, and the next threshold will be set to the nominal value.

The telemetry is down-linked using two virtual channels, one for housekeeping data (VC0), and one for science data (VC7). There are two software-controlled buffers on VC0, one for high-priority data, i.e. periodic housekeeping data and time-correlation packets, and one for normal-priority data. On VC7, there is a single software buffer for science data.

Whenever the packet terminals do not fully utilise their down-link rates, Idle Telemetry Frames will be generated automatically, maintaining a fixed bit rate on the down link.

Conclusion

The benefits of the XMM data-handling architecture and its packet data-handling approach are demonstrated by:

– very smooth integration of the different data-handling systems onboard the spacecraft, and
– the fact that, late in the programme, two Visual Monitoring Cameras could be integrated without any changes to the onboard software.

Acknowledgements

The successful development of XMM's Data-Handling Subsystem is an achievement of several individuals and companies, contributing in various ways. The contributing companies are Laben SpA (I) for the CDMU and RTUs, Alcatel Espacio (E) for the DBUs, and Dornier (D) for the subsystem engineering. Without underestimating the efforts of any individual, I wish to acknowledge the valuable work performed by K. Günthner (Dornier), G. Aranci and G. Rosani (Laben) who worked towards the common goal of developing a reliable data-handling subsystem to meet XMM's requirements.