

Real-Time access to past measurements using bidirectional communication (TmNS)

Heiko Körtzel¹, Guillermo Martínez Morán², Christian Buechner¹

¹ Airbus Defence and Space GmbH, Manching (Germany)

² Airbus Defence and Space S.A.U., Getafe (Spain)

heiko.koertzel@airbus.com, guillermo.m.martinez@airbus.com, christian.buechner@airbus.com

Abstract:

A fundamental principle of the **Telemetry Network System** (TmNS) approach is to enhance, rather than replace, today's telemetry systems by providing significant improvements in order to revolutionize how flight tests are executed. One of the highlight is the capability to **access in real-time past measurements** (onboard recorded data, while still recording new data) by using the **bidirectional communication of a TmNS**. This paper outlines the approaches taken to implement such a function on an airborne/ground network.

Key words: Telemetry Network System, TmNS, bidirectional communication, record, replay

Introduction

One key element in flight tests is a measurement data recording capability onboard. Here data is collected from several sources like sensors, data busses, data acquisition systems and video systems through suitable interfaces and stored on storage media within the aircraft under test. Beside this data recording capability a data downlink telemetry capability is another key element in classical flight tests and it is obvious that in most cases the sustained data throughput is very different for those both data sinks. For example data recording bandwidths onboard can be hundred times higher (or even more) than data downlink telemetry bandwidths used for flight tests. In classical flight tests the reduced data set to be transmitted over the downlink telemetry channel is predefined pending on the test purpose and achieved by filtering the data onboard before routing them to the telemetry channel and for backup reason also to the onboard recording system.

Within an Airbus Defence and Space internal R&T project called "Interactive Bidirectional Telemetry (IBT)" the establishing of a bidirectional communication should be analyzed as well as several use cases of interest and necessary functions needed in the airborne and ground network. Furthermore the approach should be aligned with the Telemetry Network System (TmNS) featuring selected highlights intended to be reached with TmNS. Among different kind of use cases one out of them is the dynamic change of the data set which is

transmitted over the downlink on request of the flight test engineers. Based on the limitation in bandwidth of the telemetry downlink channel two main reasons for this need are predominant. First, pending on the test purposes or pending on unexpected system behavior during a flight different sets of filtered data could become necessary to be transmitted. Second, due to possible telemetry interruption a retransmitting of data could become necessary rather than wait till access to the onboard storage media can be get (e.g. after aircraft landing).

Guidelines and Requirements

There are several conditions which has to be followed for an implementation of a retransmission function in a flight test instrumentation (FTI) system.

Architectural:

Since all measurements are stored onboard a replay back into the downlink transmission system of a past time slice would be a suitable way to retransmit data. In order to achieve this, switching functions in the airborne instrumentation architecture would become necessary to reroute the data flow. Further the recording system has to provide a controllable and dynamical access to the stored data which shall be compatible to the mentioned switching function. Moreover in any case a continuous onboard recording of the whole data set has to be assured. Another architectural guideline can be the constancy of the telemetry channel. So for a seamless operation between live data and

replayed data the data framing and key figures of the channel should remain identical.

Operational:

Since the need for access of past measurement data may arise unpredictable, the flight test engineers, as the main users of the telemetry data, need a way to change the onboard system into a replay mode on their request. This request has to contain at least start time and end time or duration of the replay and should be given through a suitable user interface.

Safety:

Since the telemetry data are a primary source of information of the state of all aircraft systems and as such a basis for decision even for safety aspects a clear situational awareness shall be present all the time during the execution of a flight test. Therefore a clear indication if the displayed data is a replay content or a live content is needed to all involved persons following the flight test. Furthermore an automatic switch back to a safe state has to be implemented onboard.

Architecture

An Ethernet based onboard architecture including a central manageable network switch allows a flexible data routing implementation. Such a central network switch with at least 6 ports can receive UDP data from a FTI data acquisition system (DAS) on port 1, route them over port 4 and 5 to recording ports on a recorder and over port 6 to a telemetry bridge. While port 2 is used for receiving control commands, port 3 is used to configure the recorder. The recorder itself shall provide 2 Ethernet recording ports to receive UDP data. The first one to receive high bandwidth data for post flight analysis and the second one to receive low bandwidth telemetry data for a later replay. Furthermore the recorder shall provide a general Ethernet port for configuration and control purpose as well as for streaming out recorded data in the replay case. Not mandatory, but for a more simple architecture a IRIG106 Chapter 7 telemetry bridge inside the recorder is desirable (see Fig. 1).

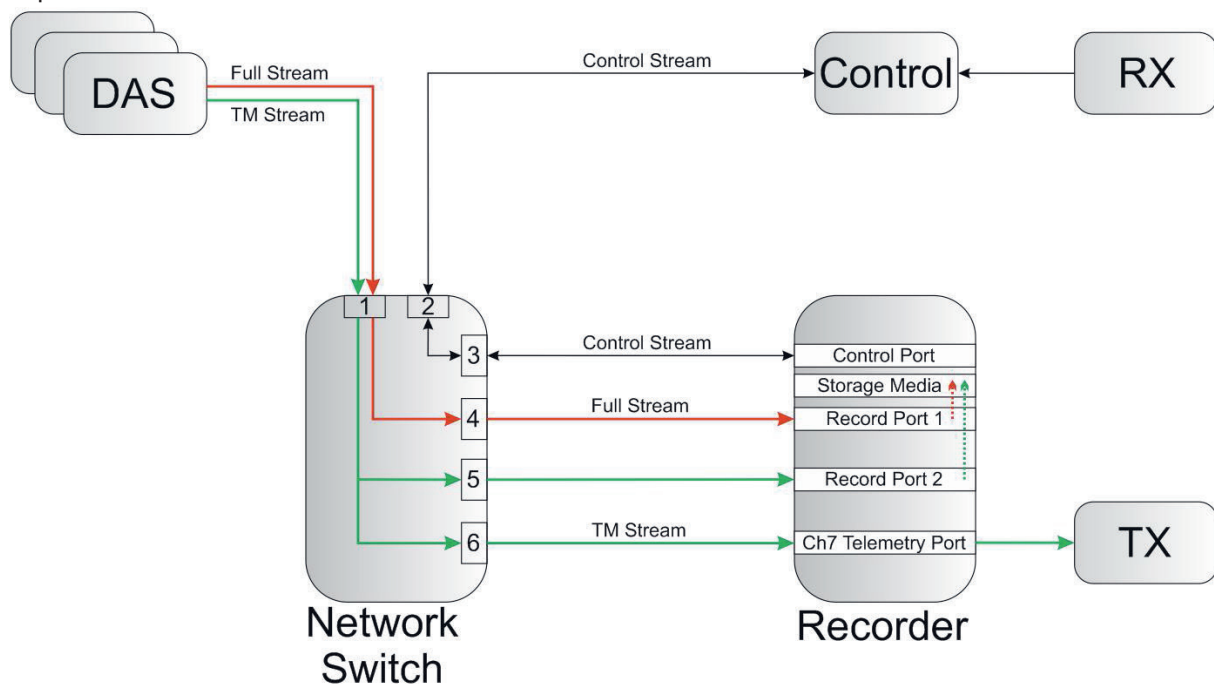


Fig. 1. Architecture with data flow in case of live data telemetry

An uplink RX system receives control commands. Those commands are analyzed and validated by a control system which also initialize on request a reconfiguration of the switch and the recorder. After such a reconfiguration for a data replay, the switch

stops the routing of live telemetry data from port 1 to port 6 and routes instead replay data from port 3 to port 6. While the recorder is reconfigured to replay data of the previously recorded telemetry data stream over its control port to the port 3 of the switch (see Fig. 2).

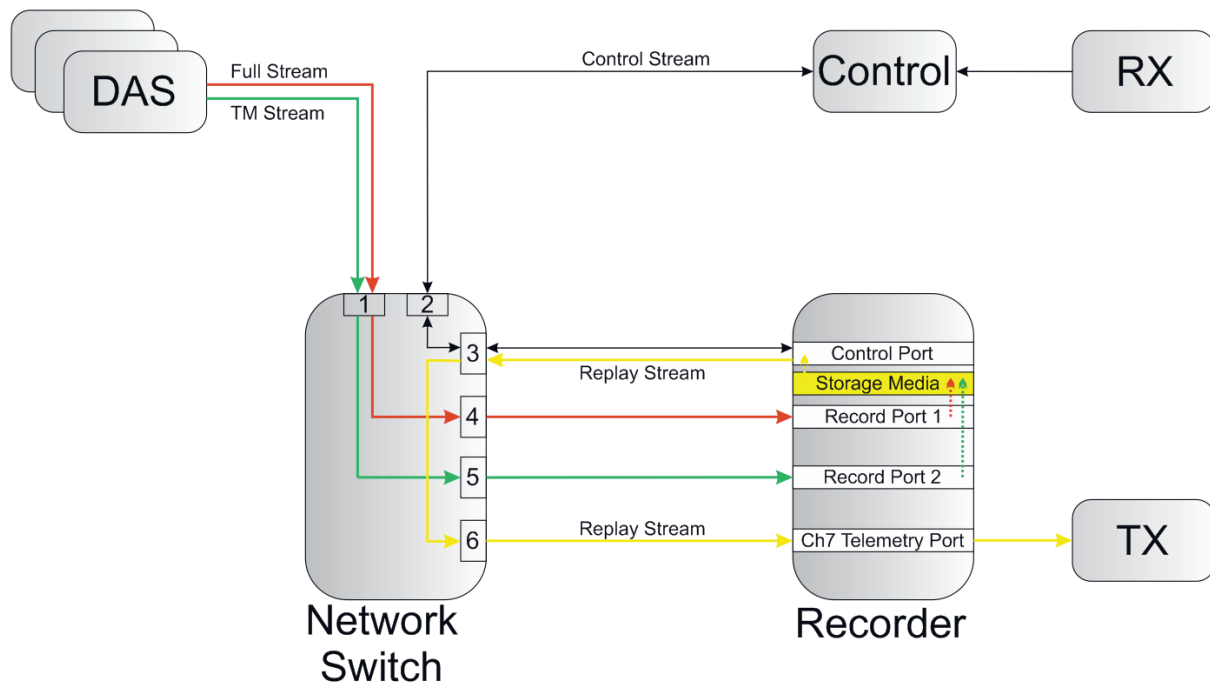


Fig. 2. Architecture with data flow in case of replay data telemetry

Implementation

Within the setup of the Airbus Defence and Space internal R&T project, the DAS generates a MPEG-TS video Ethernet stream, that is recorded and streamed according to the configuration of the network switch to a Remote Control Application, that is capable to decode the video Ethernet stream provided by the DAS or the recorder. The implementation of this R&T project can be divided into a hardware part and into a software part.

On the one hand the hardware part within the FTI architecture with dedicated connections to control the network switch and recorder. On the other hand the software part, which is part of the control unit to receive and translate data coming from a remote control application (see Fig. 2 – RX and TX block)

Hardware:

For this R&T project following hardware was used:

Control Unit

The control unit is a standalone device equipped with an Ethernet TCP/IP interface provided by i.e. a microcontroller, PC, etc. The control unit is connected to the network switch directly and to the recorder over the network switch (see Fig. 2.).

Remote Control Device

As Remote Control Device a standard PC is used, running the Remote Control Application on it (Fig. 2 – RX and TX block).

Network Switch

A NSW-12GT-1 twelve port Gigabit airborne network switch provided by Teletronics Technology Corporation (TTC) was used [1].

Data Acquisition System

The DAS system consist of an ACRA KAM-500 chassis equipped with a KAD/BCU/140 Ethernet backplane controller (IENA/iNET-X) and a KAD/VID/106 video module, provided by Curtiss-Wright [2]. The captured video is distributed by the Ethernet backplane controller via TCP/IP encapsulating iNET-X and MPEG-TS frames.

Recorder

As data recorder a Zodiac Data Systems MDR-8shRC equipped with an Ethernet module (METH2) and data storage provided by ZODIAC AEROSPACE was used. [3]

Software:

The software itself is also sub-divided in two parts. First the remote control application, that is sending control commands to the control unit and decoding the Ethernet telemetry data stream. Second the application running on the control unit, which receives control commands from the remote control application and configures the network switch and recorder according to the received remote control data. As programming language Python 3.x with QT framework was used.

Control Unit Application

The control unit application is a stand-alone application that can be executed without a graphical user interface and no user interaction. The software architecture can be splitted in three parts. The first part is a TCP/IP server awaiting data from the remote control application. If data is received it is processed according to the command (start/stop replay). The second part controls the network switch configuration. To configure the network switch, a TCP/IP Telnet connection between the switch and control unit is established.

Afterwards, in case of configuring the switch to route replay data from the recorder, the dedicated IP routing table for the specific port is changed. Now replay data provided by the recorder, available at switch port 3 is routed to port 6 and transmitted by the IRIG106 Ch7 telemetry module within the recorder (see Fig.2). If a stop command is processed by the control unit, the network switch is reconfigured by TCP/IP telnet commands to route telemetry data coming from the DAS to the IRIG106 Ch7 telemetry module again (see Fig.1). Within the network switch only the routing table for the involved switch ports are modified, that only specific multicast IP Ethernet streams are blocked or not blocked for involved ports.

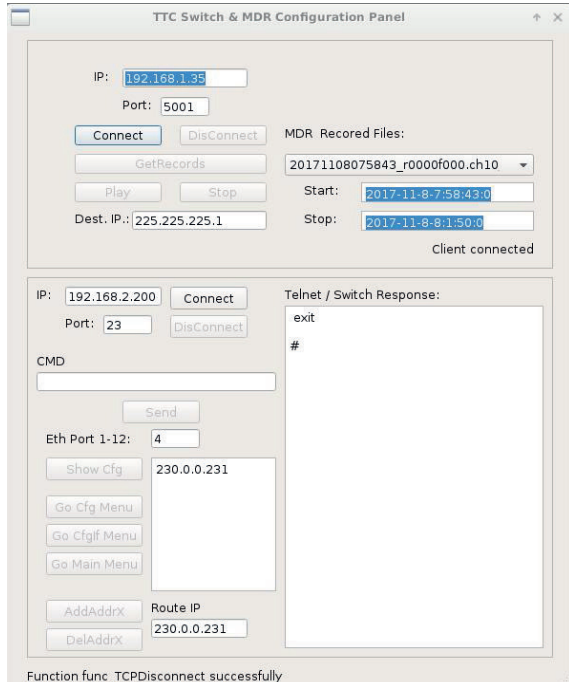


Fig. 3. GUI Control Unit Application

Just after the network switch configuration, the third part of the control unit application is initiated and directs the recorder. The recorder provides three different control modes:

- IRIG-106 remote commands (serial connection)

- Simple Network Management Protocol (SNMP) commands (by TCP/IP)
- UDP remote access (by TCP/IP)

The control unit application uses the UDP remote access for recorder communication.

In the first step the control unit application queries the name and start/stop time of all recordings from the recorder. Afterwards the recording file is selected that matches the start/stop time for replay data (start/stop time is retrieved from the remote control application).

bits	15 (msb)	8	7	(lsb) 0
bytes	+0		+1	
+0	max_udp_size (uint16)			
+2	tmats_interval (by default = 0) (uint16)			
+4	op_mode (uint8)		speed_mode (uint8)	
+6	speed (uint16)			
+8	reserved1 (0) (uint16)			
+10	reserved2 (0) (uint16)			
+12	reserved3 (0) (uint16)			
+14	reserved4 (0) (uint16)			
+16	start_year (uint16)			
+18	start_month (uint8)		start_day (uint8)	
+20	start_hour (uint8)		start_minute (uint8)	
+22	start_second (uint8)		start_10ms (uint8)	
+24	stop_year (uint16)			
+26	stop_month (uint8)		stop_day (uint8)	
+28	stop_hour (uint8)		stop_minute (uint8)	
+30	stop_second (uint8)		stop_10ms (uint8)	
+32	{ file_name (512 bytes)			
...				
+542	}			
+544	mcb		nof_subcomponents (uint32)	
+546	lcb			
+0	{			
+2			destination_ip (by default = 11.59.31.255) (uint32)	
+4	mcb		destination_port (default = 2000) (uint32)	
+6	lcb			
+8	mcb		nof_channel_ids (uint32)	
+10	lcb			
+0	{			
+2			channel_id (def = -1 for no filter) (uint16)	
+2			reserved5 (0) (uint16)	
	}}			

Fig. 4. Recorder UDP Start Replay command [4]

Afterwards a specific UDP control frame (see Fig. 4) is send to initiate replay data from the recorder. The main settings in this control frame are filename, start-/stop time for replaying and multicast destination IP address and port. From now on replay data is streamed by the recorder in IRIG-106 chapter 10 format, encapsulated in a UDP Ethernet frame. If replay data streaming has stopped a command is received from the remote application and the IP routing table of the network switch is changed to route data coming from the DAS to the telemetry gateway again.

Remote Control Application

The Remote Control application (see Fig. 2 – combination between RX and TX block) provides a graphical user interface (GUI), with all parameters that are necessary to start data replay of specific recorded data. The main functionality of this application is to establish a TCP/IP Ethernet connection to the server provided by the Control Unit application. The Remote Control application encapsulated a start/stop command combined with start/stop time for the data replay and transmits this data to the Control Unit where this data is extracted and starting or stopping replay data is initiated. Another functionality of the application is detecting UDP Ethernet data provided by the Recorder or DAS. If data replay has finished, a stop command is transmitted to the control unit application to initiate reconfiguration of the network switch to route the DAS TM (telemetry) data stream to the recorder IRIG106 Ch7 telemetry gateway module again.

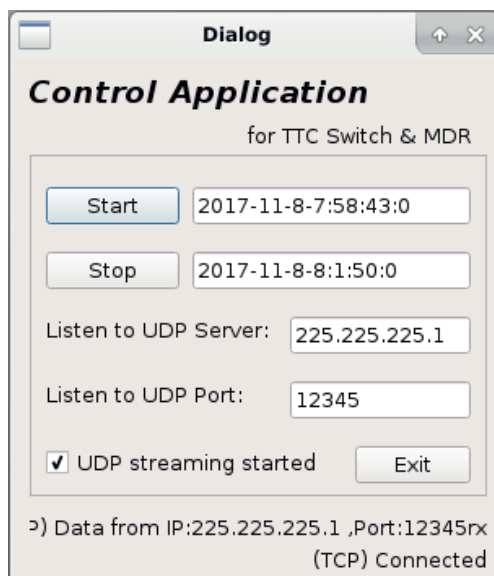


Fig. 5. GUI Remote Control Application

To decode a telemetry Ethernet data stream, the application must handle different Ethernet frame formats encapsulated in each other. For example a video Ethernet data stream provided by DAS can consist of following encapsulation:

ETH/IP (UDP (IENA/iNET-X (MPEG-TS)))

If this data is replayed by the recorder the encapsulation of this stream will increase to following framework:

ETH/IP (UDP (IRIG-106 Ch10 (IENA/iNET-X (MPEG-TS))))

For evaluation purposes the Remote Control Application running on a dedicated PC is connected to the network switch directly to pass the IRIG106 Ch7 telemetry gateway module

and to extract encapsulated data without the need to decode IRIG-106 Ch7 header.

Summary

This first implementation of a real-time access function to past measurements shows a seamless operation between transmitting live data and replay of stored data. All architectural and operational guidelines were implemented and the final implementation of safety guidelines should be easily feasible with the Control Unit Application. Furthermore this implementation can be considered as a practical use of some features emphasized by the TmNS Standard as followed [5]:

“The IP network foundation of the TmNS brings with it features including routing, Quality of Service (QoS), and congestion control. The following list highlights some of the capabilities that IP networking brings to telemetry.

- *Addition of Bidirectional Communications to Telemetry: bidirectional communications is one of the most fundamental enhancements provided by the TmNS. It enables the following capabilities.*

- o *Real-Time Access to Test Article (TA) Measurements: Provides a mechanism for real-time access to current and past measurements on the TA both directly from the sensors and from the recorder(s).*

- o *PCM Backfill: Provides the ability to retrieve measurements from the TAs in near real time that were dropped in the Serial Streaming Telemetry feed (PCM dropouts).”*

References

- [1] Teletronics Technology Corporation, NSW-12GT-1 Manual, Revision D, 8/29/16
- [2] Curtiss-Wright, ACRA KAM-500 Databook, HW/BK/0002, 12/04/2017
- [3] Zodiac Data Systems GmbH, Technical Specification HEIM DATaRec 4 MDR-8s/MDR-8sf, Doc. 56024102

Zodiac Data Systems GmbH, Technical Specification METH2, METH2A, Doc. TS_METH2_METH2A_160419_HD5620041R018 35.fm
- [4] Zodiac Data Systems GmbH, DATaRec® 4 Series, Programming Manual, Version 1.41, 4/4/2017
- [5] Telemetry Standards, RCC Standard 106-17 Chapter 21, July 2017