Atacama
Large
Millimeter
Array

Interface Control Document

Between
Back End DTS Transmitter Module
And Computing Control Software

ALMA-53.08.00.00-70.35.30.00-B-ICD

Version: B
Status: Released
2008-10-06

Prepared By:	**Organization**	**Date**
A. Baudry C. Langley, R. Marson, P. Metzner	University of Bordeaux NRAO	2008-10-06
IPT Leader Approvals:	**Organization**	**Date**
Chris Langley Fabio Biancat Marchet Brian Glendenning Gianni Raffi	NRAO ESO NRAO ESO	
System Engineering Approvals:	**Organization**	**Date**
Configuration Control Board Approval:	**Organization**	**Date**
JAO Approval:	**Organization**	**Date**

Change Record

Version	Date	Affected Section(s)	Change Request #	Reason/Initiation/Remarks
A	2006-11-01	All		Draft DG = prototype
B	2008-10-06	Many	ALMA-54.01.00.00-007-B-CRE	Added new commands for EEPROM and I2C bus. Changed FR_PAYLOAD commands (debug). Added and modified DG commands for DG= production

Table of Contents

LIST OF FIGURES

1 Description

1.1 Purpose

The purpose of this document is to define the Monitor and Control interface between the Data Transmission System (DTS) Transmitter Module (DTX) and the Computing Monitor and Control software.

1.2 Scope

The document contains lists of monitor and control points available in the DTX and their suggested access rates. These points will allow for the monitor of the state of the DTX for further analysis and identification of any malfunctioning subsystem.

2 Related Documents and Drawings

2.1 Applicable Documents

[AD 01] ALMA-70.35.10.03-001-A-SPE, ALMA Monitor and Control Bus Interface Specification
[AD 02] ALMA Software Glossary, Draft Version, 2003-May-21
[AD 03] ALMA-70.35.05.00-001-B-STD, Generic Monitor and Control Points and Documentation Conventions
[AD 04] ALMA-70.35.10.02-001-A-MAN, ALMA Monitor & Control Bus, AMBSI2 Standard Interface Design Description
[AD 05] BEND-57.02.02.05-001-A-DSN, ALMA Monitor and Control Bus 1 (AMB1) Wiring Description
[AD 06] BEND-53.01.00.00-53.08.00.00-C-ICD, Interface Control Document Between DTS (DG) Assembly and DTS Transmitter (DTX) LRU
[AD 07] ALMA-53.06.00.00-70.35.30.00-B-ICD, Interface Control Document Between Back-End DTS Receiver Module and Computing Control Software
[AD 08] ALMA Memo No. 420, Digital Transmission System Signaling Protocol

2.2 Reference Documents

[RD 01] REFERENCE DOCUMENT FOR 300 PIN 10Gb TRANSPONDER
http://www.300pinmsa.org/html/documents.html

[RD 02] Fujitsu FIM13130/200Wxx Preliminary Specification Rev. C

2.3 Abbreviations and Acronyms

ABM	Antenna Bus Master
ADC	Analog to digital conversion
AMB	ALMA Monitor and control Bus
AMBSI2	ALMA Monitor and control Bus Standard Interface
CAN	Controller Area Network
DG	Digitizer assembly
DRX	DTS Receiver module
DSUB	D-shaped Subminiature connector
DTS	Digital Transmission Subsystem
DTX	DTS Transmitter module
EVLA	Expanded Very Large Array
FOM	Fiber Optic DWDM (Dense Wavelength Division Multiplexing) Multiplexer
FPGA	Field Programmable Gate Array
FR	Formatter
I2C	Inter Integrated Circuit
IFP	Intermediate Frequency Processor module
LO	Local Oscillator
RCA	Relative CAN Address
RNG	Random Number Generator
SPI	Serial Peripheral Interface
TE	Timing Event
TTX	Laser half-transponder unit

See also [AD 02].

2.4 Glossary

Sync Pattern: 10-bit word used to re-establish what is recognized as the first bit of each data frame.

Metaframe Index: 1 bit indicating the first 160-bit frame after an external timing event (20.833 Hz).

Sequence Count: 5-bit counter used to identify 32 consecutive frames. Allows for re-synchronization of the 3-bit (D, C, and B) samples, assuring bits of the original samples remain together.

Parity: 16-bit pattern which is derived by XORing, 16 bits at a time, the 144 bits of frame data in each frame which are not parity bits. This process is repeated at the receiver end of the DTS and the result is compared to the parity broadcast from the transmitter with the original frame.

System Clock: 125 MHz input clock used system-wide.

DG Clock: 250 MHz input clock from the digitizer (DG).

Frame Clock: 62.5 MHz derived clock used to make frame packets.

See also [AD 08].

3 Physical System Interfaces

3.1 Mechanical Interface

See [AD 05].

3.2 Electrical Power Interface

Not applicable.

3.3 Electronic Interface

An AMBSI2 board as described in [AD 01] will be used.

3.3.1 List of connectors

The ALMA Monitor and Control Bus (AMB) connection between the Computing IPT and the BE IPT at each ALMA antenna will be the CAN interface for all modules. The communications processor for the CAN bus will be imbedded in each module. A standard ARINC 600 SGA connector will serve as interface between the AMB and the Data Transmitter Module (DTX) – see

List of Electronic Connectors			
Connector Type BE to Comp	**Connector Reference BE to Comp**	**Function**	**Responsible for delivery**
MIL-DTL-D38999/III	DTX to AMB1	AMB	Back End

3.3.2 Interconnection Data Sheets

See [AD 05].

3.4 Mass/Balance

Not applicable.

3.5 Thermal Interface

Not applicable.

3.6 Gaseous Supply Interface

Not applicable.

4 Software/Control Function Interface

4.1 Overview

The DTS Transmitter Module (DTX), which houses the Digital Formatter (FR) with laser half-transponder units (TTX), Digitizer Assembly (DG), and Monitor Control / Power Supply (MCPS) is controlled and monitored by the Antenna Bus Master (ABM). The M&C interface consists of a single CAN node operating as defined in [AD 01].

Four DTX modules are in each antenna. Each communicates with the CAN bus using an AMBSI2 card that is mounted on the MCPS circuit board. The CAN nodes shall operate at node addresses 0x50, 0x51, 0x52 and 0x53 (hex), and these addresses will correspond to basebands and baseband pairs shown below. The table below summarizes the relationships.

CAN Node	Baseband		Baseband Pair	Allowable CAN Address range
0x50	BB0	BB1	BBpr0	0x140 0000 – 0x143 FFFF
0x51	BB2	BB3	BBpr1	0x144 0000 – 0x147 FFFF
0x52	BB4	BB5	BBpr2	0x148 0000 – 0x14B FFFF
0x53	BB6	BB7	BBpr3	0x14C 0000 – 0x14F FFFF

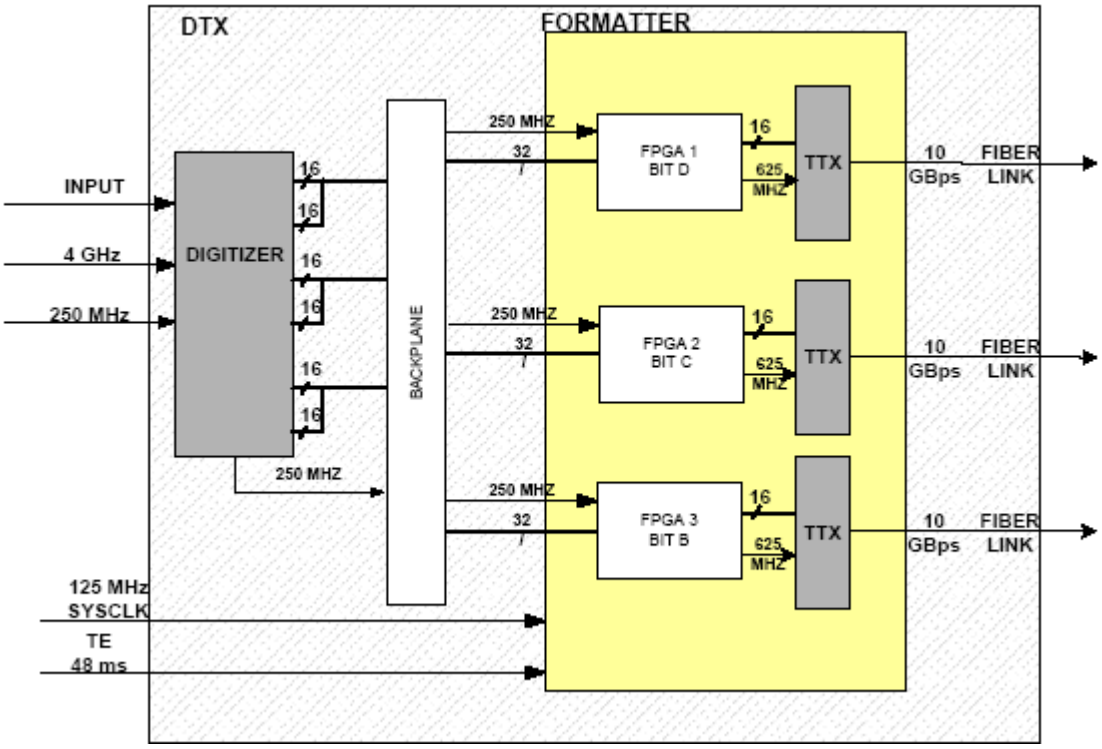
4.1.1 DTX Module Overview

Each DTX will digitize and transmit the two polarizations for each 2 – 4 GHz baseband channel received from the two IF Processor Modules (IFP) in the Analog Rack.

In brief, the ALMA Front End delivers four analog astronomical signals in the 4 - 12 GHz band to the ALMA Back End. These are transmitted by microwave cable to the Analog Rack in two orthogonal polarizations, referred to as P0 and P1. Each 4-12 GHz polarization is defined by two sidebands, S0 and S1. Each IFP accepts the two sidebands of one polarization in the 4 - 12 GHz, and converts it into four conditioned 2 - 4 GHz band limited signals. Each IFP will then send one of these basebands to each DTX, located in the Digital Rack (DR). From one IFP, each DTX will receive one of four 2 - 4 GHz basebands of P0, and from the other IFP, each DTX will receive one of four 2 - 4 GHz basebands of P1. Two 2 – 4 GHz basebands, one of each polarity, constitute a baseband pair. For DTX 0x50, this baseband pair is referred to as BBpr0. Likewise, for DTX 0x51 the pair is BBpr1, for DTX 0x52 the pair is BBpr2, and for 0x53 the pair is referred to as BBpr3.

Each DTX performs the same functions on the received 2 – 4 GHz analog signals. The two polarizations are received into each module, where, on the DG, they are sampled into two separate 3-bit serial data streams. B, C, and D are the bit designators, with D being the sign bit and B being the least significant bit. Each of the two bit streams are then demultiplexed into separate words consisting each of sixteen D, C, and B bits for each polarization. The resulting six16-bit words for each bit level and polarization are then sent at 250 MHz to one of three separate FPGAs on the FR.

The result is that 16 B bits of P0 and 16 B bits of P1 are latched into FPGAB every 250 MHz. Similarly, FPGAC and FPGAD latch in 32 bit words (half of the bits P0, half P1) consisting of C bits and D bits every 250 MHz, respectively. Each FPGA, with the aid of an accompanying demultiplexer integrated into each transponder (TTX), then rate-changes the data and provides the formatting necessary to organize the data so that each of the D, C, and B bit data streams are transmitted at 10 Gb/s throughout the data link. In all, twelve of these data streams are transmitted from each antenna.



Each of the D, C, and B bit streams are formatted in the same manner. Through rate changing, each 32-bit word received into its designated FPGA at 250 MHz is registered until a 128-bit word at 62.5 MHz may be formatted. 32 additional bits of formatting information are added to each 128 bit pattern, resulting in a 160 bit word, or data frame. Refer to [AD 08] for a detailed look at the formatting overhead. In brief, the following formatting items are included in each data frame. Each is used to monitor data integrity, and in some cases re-establish data patterns, in the deformatter at the receiver end of the DTS.

For the terms used in Figure 1, please refer to the Glossary.

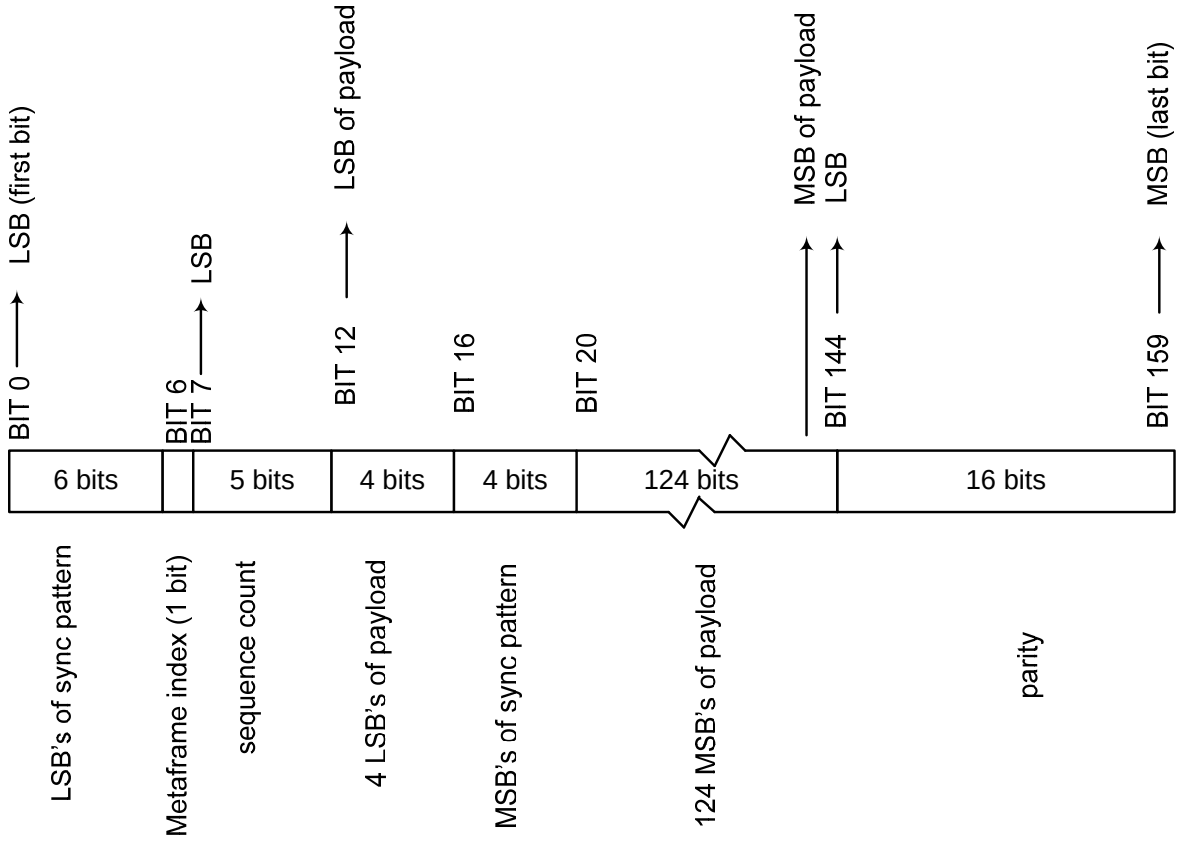
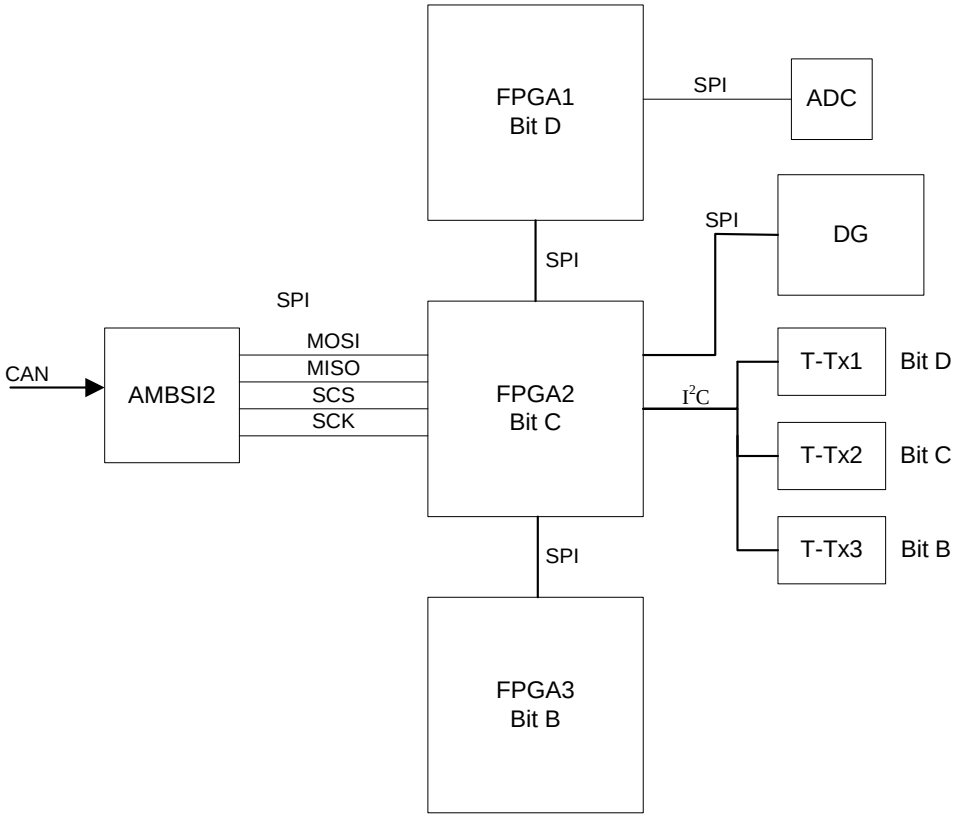


Figure 1: ALMA Memo 420: 160 bit frame organization

Bit 0 LSBs of sync pattern (6 bits): 100101
…
Bit 6 Metaframe index (1 bit): 1=48ms event
Bit 7 Sequence count (5 bits): 0-31
…
Bit 12 4 LSBs of payload (4 bits)
…
Bit 16 MSBs of sync pattern (4 bits): 0110
Bit 20 124 MSBs of payload (124 bits)
…
Bit 144 Parity (16 bits)
…
Bit 160 = Bit 0 of New Packet

4.2 Formatter (FR)

The formatter is a card in the DTX module which talks directly to the AMBSI2. There are three subsystems within the DTX: formatter (FR), digitizer (DG) and the half-transponders (TTX). All communication to the DTX goes from the AMBSI2 to the FR. The FR contains three FPGAs (one each for bit B, C and D). The FPGAs are physically labeled on the board as FPGA1, FPGA2 and FPGA3, which correspond to bits D, C and B respectively.



FPGA2 (bit C) is the only FPGA which is directly connected to the AMBSI2. FPGA2 is used to communicate with the DG, the three TTX modules and the other two FPGAs.

4.2.1 Module Connection Verification

Each DTX should, in normal operation, be paired with a matching DTS receiver module (DRX) located in the correlator room. This correspondence is defined in Table 1 of [AD 07]. When an antenna is moved to a new pad, there must be a corresponding adjustment of the fibers in the patch panel room to ensure that the DTX again communicates with the *same* DRX. The fiber configuration should be verified using the following algorithm:

1. Set the DTS transmitter to transmit a known test pattern (SET_CW_CHn) on a specific baseband channel. No other baseband channel should be transmitting.

2. Verify that this test pattern is received on the expected DRX. It is possible to associate each DRX with a specific baseband and antenna using the CAN node ID.

3. Repeat steps 1 & 2 for each baseband channel and each antenna that was moved.

By virtue of a built-in start-up routine, the DTX is designed to enter normal operating mode upon power-up. However, a settling-in time of 10 seconds after applying power should be observed before any monitored data is considered valid. The assembly receives 48VDC through a 3 position DSUB connector on the faceplate of each module. A switch is located near the connector. The switch should be in the off position before the 48VDC source line is plugged in. No other user intervention is required for the assembly to function properly. No power-down sequencing is required, so the unit may be turned off simply by moving the power switch to the off position.

In addition to the monitor and control points listed in the sections contained in this document, the mandatory monitor points defined in [AD 03] shall be implemented. The system hardware reset signal is not used by the DTX.

4.2.2 Timing Event

The system timing event (TE) enters the formatter card through the center FPGA (Bit C). The center FPGA uses the TE to sync its own internal TE with the system TE. This synchronization is done on the receipt of the tenth system TE. Synchronization will not occur again unless a reset command is issued to the FR.

The internal timing event is passed from FPGA2 to FPGA3 and FPGA1 through hardware lines between the chips. Only FPGA2 (Bit C) monitors the timing event for errors.

When the TE monitor detects that the incoming TE is out of sync with the internally generated TE, it will set an error flag.

To clear the TE error flag, an FR_TE_RESET (or a FR_RESET) command must be issued. If the TE error flag is automatically set by the hardware again, after the reset is issued, then the clock edge used to clock in the TE should be changed (see FR_TE_RESET 4.7.4.3). The clock edge selection is not reset during reset of the chip, but must be maintained anytime the FR_TE_RESET command is sent (i.e. if attempting to reset the TE error registers only, the clock polarity bit must also be sent in the correct state). This register should be read before writing; otherwise, the correct state may not be sent.

4.2.3 Phase Switching

The analog signal that is supplied to the DTX has its phase switched by adjusting the phase of the Local Oscillator (LO). Phase switching is done using the First LO Offset Generator to switch the first LO phase. The primary motivation for phase-switching is to suppress systematic offsets in the signal that are introduced between the receivers and the digitizers. To do this, the analog signal is phase-switched by 180 degrees and removal of this phase switch is done in the DTX.

For ALMA, a 180-degree phase switch can occur every 125 microseconds, and there is a 128-bit sequence (a Walsh function) that defines the switching sequence. This sequence contains zero when the digitized signal should not be changed and one when it should be inverted. The sequence is 128 bits long and hence repeats after every 16 ms. The sequence starts with each timing event and there are exactly three complete passes through this sequence every 48ms.

The phase removal is performed after the DG Assembly in the Formatter within FPGA #1 which is dedicated to the sign bit so that the sign bit is reversed. The FPGA contains a 128 bit register (FR_PHASE_SEQ_A/B). Each bit in this register corresponds to a 125 microsecond interval (125 µs * 128 = 16 ms). When a bit from the register is set to one, the FPGA will invert all D bit samples for a 125 µs period. The inversion begins 8 ms after the TE plus whatever delay is programmed into the FR_PHASE_OFFSET register. Each count in the FR_PHASE_OFFSET register is equivalent to an 8 ns delay.

4.2.4 Random Number Generator

The random number generator (RNG) is used for debug and bit error testing by the correlator. The RNG is activated using the SET_FR_RNG command. It is recommended that all three bits are started simultaneously using the 0xC00A command.

The random number generator is a 32-bit number defined as:

Bit(n) = Bit(n+1) or Bit(n+3) for bits 0 – 31

The bottom 3 bits are carried back to the top:

Bits(32, 33, 34) = Bits(0, 1, 2)

The seed to start the generator is defined in the FPGA for each bit:

Bit D seed: 0xD1D1D1D1AAAAAAAA
Bit C seed: 0x07070707E0E0E0E0
Bit B seed: 0x555555553A3A3A3A

4.2.5 SPI Interface

Details on the internal workings on the ABMSI2 can be found in [AD 01]. FR, DG and TTX commands from the ABM are sent to the AMBSI2 through the CAN bus. The AMBSI2 converts this message to a SPI (Serial Peripheral Interface) bus protocol.

The FR communicates monitor and control data between the AMBSI2 and an FPGA located on the FR using the SPI bus. From the AMBSI2, commands are sent to the FPGA via the same SPI port for FR, DG and TTX components.

Address 0x00000 is reserved for a CAN-to-AMBSI2 serial number request and address 0x18000 is the start of CS2 (which is not used here). The FR has a separate SPI bus for communication with the DG and an I2C bus for communication with the three TTX units (which are actually part of the FR). When the DG or a TTX is addressed, the FR FPGA translates the communication between those modules and the AMBSI2. The FPGA provides that information or control to the appropriate unit and also provides the response to the AMBSI2.

4.2.5.1 ID request

On power-up, the AMBSI2 sends an ID command to all devices on the SPI bus. The FR answers and configures the link using the switch settings on the board (see Section 4.2.9). The address which is reported to the AMBSI2 is actually 1 more than the address used system-wide (address x53 is reported to the AMBSI2 as x54, but read by the system as x53). The AMBSI2 sends:

0xB0	command code indicating ID request
0x00	null byte
0x00	null byte
0x00	null byte

Three null bytes are sent for timing in order to receive 3 bytes from the FR.

The FR will answer the ID command with the codes:

0x50	first ID byte acknowledge
0x5n	n is 00xx and xx are module address (0x50, 0x51, 0x52, 0x53) +1
0x09	where the two least significant bits represent a 2 byte address protocol

The last byte sent tells the AMBSI2 that all Monitor and Control commands sent from the AMBSI2 must have two address bytes transmitted for a complete command.

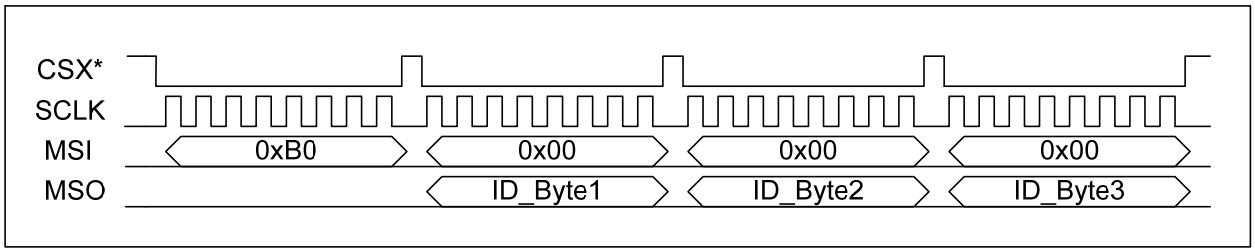


Figure 2: ID Transfer Transaction

The ID command will always be responded to with three bytes only.

4.2.5.2 Monitor Commands

For monitor commands, the AMBSI2 will transmit the code 0xA0. The FR will respond with a single acknowledge byte = 0x59. The AMBSI2 will then transmit two address bytes (high byte, low byte) corresponding to the RCA that has been requested:

i.e. RCA 0x9040 would result in a transmission sequence from the AMBSI2 to the FR of:

0xA0	command code indicating monitor request
0x90	high byte of address (sent after receipt of acknowledge)
0x40	low byte of address

Once the address has been received from the AMBSI2, the FR will transmit a byte count for that request and then transmit the data:

i.e. RCA 0x9040 would result in a response from the FR to the AMBSI2 of:

0x59	monitor request acknowledge (sent immediately)
0x00	1 byte of data to be returned (sent after receipt of address)
0xnn	data requested (1.5 V monitor) (immediately follows address)

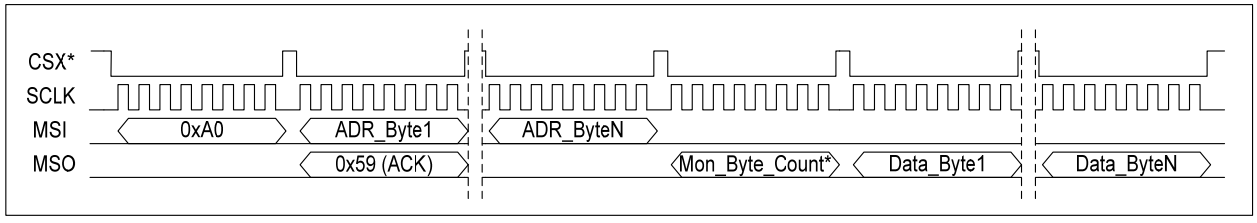


Figure 3: Monitor Transfer Transaction

The FR collects monitor data from the FR and the DG. Monitor data is continually updated into the internal registers of the FPGA where it can then be polled by the AMBSI2. Most FR monitor points are input from 10-bit A/D converters (0-5.0 volts), of which only the upper 8 bits are utilized. The remaining FR monitor points are firmware-related and are generated internally by the FPGA. The DG monitor points are distributed to verify that: a) the 2.5 and 5 VDC used in the DG Assembly are nominal, and b) the temperature inside the DG assembly does not exceed some limits.

4.2.5.3 Control Commands

For control commands, the AMBSI2 will transmit the code 0xAn, where n represents the 4 bits: 1xxx. The highest bit tells the FR that this is a control code. The bottom three bits (xxx) minus one are the number of bytes to follow in this command.

i.e. RCA 0x9030 would result in a transmission sequence from the AMBSI2 to the FR of:

0xA8 command code where LS bits 000 mean 1 control byte to be sent
0x90 high byte of address (sent after receipt of acknowledge)
0x30 low byte of address
0x00 Null byte in this control – the address is the control

The FR will respond to the command with an acknowledgement.

0x59 control command acknowledge

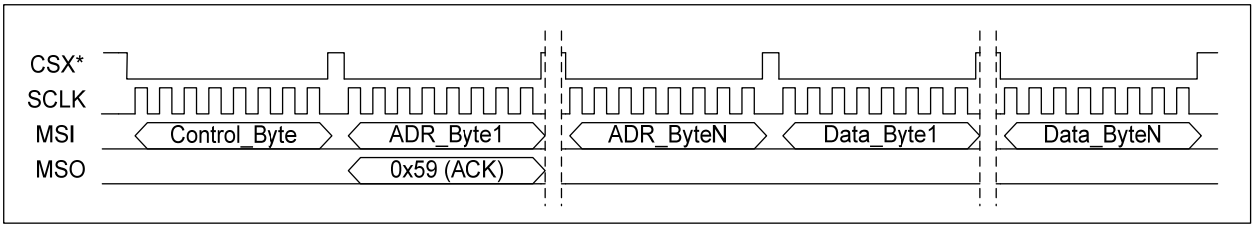


Figure 4: Control Transfer Transaction

For positive verification that the control was carried out, a monitor command can be issued.

4.2.5.4 Bad Address Response

All commands are acknowledged before the full address has been received by the formatter. Once the formatter has received the full address, it determines how many bytes are to be returned (in the case of a monitor point). If a command is received which does not correspond to any of the monitor points in this ICD, the formatter will respond with a single byte with bit 3 set (0x08).

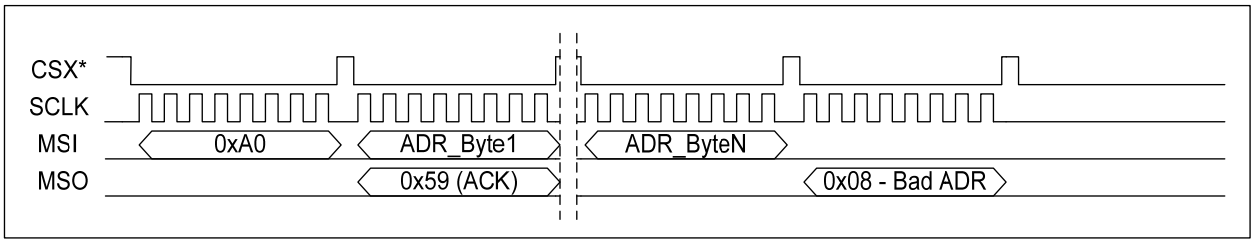


Figure 5: Bad Address Monitor Transfer

This lets the AMBSI2 know that an invalid command was received and the AMBSI2 should not respond to it.

4.2.6 Debug Command Options

Debug commands are designed for the firmware developer or technician to test the system. Commands generally have options to configure the formatter for verification or test. These commands are for test modes only. Options are explained in the FR Control Points in Detail section under the individual command.

4.2.7 RCA Definition

All commands go through the FR card for the three different components of the DTX: FR, DG and TTX. The RCA addressing has been broken up to designate what a command is (monitor or control), which FPGA it goes through, and the specific command. Each RCA is 16 bits:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
m/c	F#	f#	f#	u#	u#	u#	u#								

High Address Byte Low Address Byte

m/c 1 = control, 0 = monitor

f# 001 = CH1 = FPGA1 = bit D
010 = CH2 = FPGA2 = bit C
011 = CH3 = FPGA3 = bit B
100 = All FPGAs (control points only)

u# 0000 = non-specific component
0001 = TTX1 (FPGA2 only)
0010 = TTX2 (FPGA2 only)
0011 = TTX3 (FPGA2 only)
0100 = All transponders (FPGA2 only)
0101 = DG (FPGA 2 only)
0110 = ADC information (FPGA 1 monitor only)

The low address byte is assigned according to command. The same commands to a specific component will have the same low address byte: GET/SET_CW_CHn changes only the FPGA number (f#) to read different channels. Bit 15 is changed according to whether it is a monitor or control. GET_CW_CH1 = 0x1001, SET_CW_CH1 = 0x9001.

Only RCA points specified in this ICD are valid. RCAs cannot be "created" from the above description – the description is given as reference only. If the FR receives a monitor command that is not called out as a specific RCA in this document, the FR will return an error code to the AMBSI2. If the FR receives a control command that is not called out as a specific RCA in this document, the FR will ignore it.

4.2.8 Keep-Alive Signal

A low cost fiber optic input, called the keep-alive signal, is driven by the overpower monitor on the FOM. This signal is driven high if this fiber is broken or if the output of the FOM exceeds the eye-safe level. Otherwise it is driven low. It will shut down the power to the laser diodes.

Loss of the keep-alive signal will trigger the FPGA to shut down the lasers for all three half-transponders. The lasers will not be restarted automatically once the keep-alive signal returns to active (low). Lasers can be restarted through the CAN once the keep-alive signal has gone active. The keep-alive status can be monitored by the CAN through the READ_FR_STATUS command.

The keep-alive signal comes into FPGA 1 as the Overpower signal and is sent to FPGA2 through signal lines between the FPGAs as the Keep Alive signal (see also section 4.2.10).

4.2.9 Switch Settings

The formatter uses the switch S200 to set the module address for the CAN bus. Switch 1 is Module Address bit 0 and switch 2 is Address bit 1. Turning the switch on will make the address bit a 0:

Switch 2	Switch 1	Module Address
ON	ON	x50
ON	OFF	x51
OFF	ON	x52
OFF	OFF	x53

The formatter printed circuit board is the same board used by a different project (EVLA); the software is also the same. Some functions on the boards apply to one project but not the other.

Switch Number	Function	Description
8	Reserved	OFF = normal
7	Reserved	OFF = normal
6	Reserved	OFF = normal
5 SW100	Override Keep Alive	ON = ignore Keep Alive signal, OFF = normal

Switch Number	Function	Description			
5 SW300		OFF = not used - EVLA only			
5 SW200	Laser Control	ON = turn Lasers ON			
		OFF = Laser on/off is controlled by CAN bus			
4	Mode	OFF = normal			
		ON = Control output from switches 3-1			
3, 2, 1	Test Pattern	sw3	sw2	sw1	Formatter Output
		OFF	OFF	OFF	Payload = counting data
		OFF	OFF	ON	Payload = pattern 0xAAAA5555 x 5
		OFF	ON	OFF	Payload = pattern 0x3333 x 10
		OFF	ON	ON	Payload = pattern 0xCCCC x 10
		ON	OFF	OFF	Frame = AAAA5555 x 5
		ON	OFF	ON	Frame = FFFF0000 x 5
		ON	ON	OFF	Frame = 0000FFFF x 5
		ON	ON	ON	Normal frame and payload

The Mode switch will control the data that is fed to the output. If The Mode switch is left in the ON position in the field, the SET_FR_CW_CHi command will not function correctly and data may be only test data. This switch position can be overridden using the FR_RESET_CHi command (see 4.7.4.1).

Frame data switches will change all 160 bits of data to the specified pattern. Payload data refers to the 128 bits of actual data. When payload data selection is made, scramble codes and parity are enabled. When frame data selection is made, scramble codes and parity do not apply.

Frame data and payload data can also be changed through the CAN bus interface (when Mode switch 4 is in the OFF position).

4.2.10 Formatter LEDs

Each bit has two sets of LEDs associated with the electronics for that bit.

Each transponder has eight LEDs associated with status or voltages:

Reference Designation			Color	Function	Normal Operation
419	421	423	red	TxFifoErr	ON
418	420	422	red	TxLockErr	ON
404	410	416	green	+5V	ON
405	411	417	green	+3.3VDC In	ON
403	409	415	green	+3.3V Analog	ON
402	408	414	green	+3.3V Digital	ON
401	407	413	green	-5.2V Analog	ON
400	406	412	green	-5.2V Digital	ON

Each FPGA has eight LEDs associated with the electronics for it:

Reference Designation			Color	Function	Normal Operation
100		300	green	125MHz PLL	ON
	200		green	Laser Bit B	ON (when turned on)
101		301	green	250MHz PLL	ON
	201		green	Laser Bit C	ON (when turned on)
102		302	green	data 8- NA	N/A (EVLA only)
	202		green	Laser Bit D	ON (when turned on)
103		303	green	Test Clock	ON
	203		green	Keep Alive	ON
104			red	Overpower	OFF
	204		red	TTX Alarm	OFF
		304	red	Test Mode	OFF
105	205	305	red	Switch 4 Mode	OFF
	206		red	TE internally generated	OFF
106		306	red	reset	OFF
107	207	307	red	EVLA/ALMA	OFF

4.3 Half-Transponder – Transmitter (TTX)

The formatter board uses the transmitter half of a standard telecom transponder module to convert the electrical signals coming from the FPGAs at 625 MHz and 16 bits into an optical signal at 10 Gbits/s. There are three half-transponders (TTX) per FR; each is individually addressable and configurable. The TTX modules are addressed through the FPGA2 (see 4.2). The FPGA2 translates commands coming from the AMBSI over the SPI bus to I2C (Inter Integrated Circuit) bus. Commands are executed on the rising edge of the reference clock (48 ms).

The half-transponder should work regardless of the power-up sequence. This means that the -5.2V, APS, +3.3V, and +5.0V power supplies may be turned on in any sequence and the transponder should still function correctly. The half-transponder is powered from the FR card and all power supply voltages can be monitored from the FR monitor points.

The TTX communication consists of software monitor points and hardwired control points. Both can be manipulated through the CAN bus. Some of the monitor points can be configured for access via the I2C bus or as an ADC monitor point. For the purposes of this document the user should assume a “Pin Control Mode” of operation: Pin Control mode modules only accept module hardware control commands from the host via hardware pins. A module in this mode works identically to a module that does not have I2C communications capability - it only accepts hardware control commands from hardware pins. However, modules in this mode provide status information via the communications interface. This interface is transparent to the user of the CAN bus and this information is provided as background only.

4.3.1 TTX – Alarm Structure

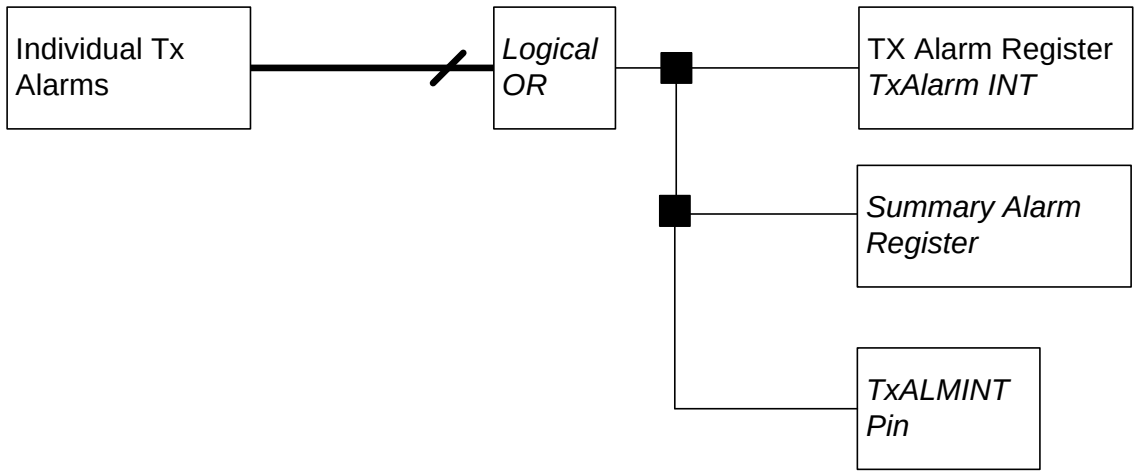


Figure 6: Half-Transponder Alarm Structure (300 pin MSA for 10 and 40GB Transponders)

There is a Transmit Alarm register for each half-transponder. This register is volatile. The FPGA polls the register and does a logical OR with the last read values. This is the 16-bit value read by the AMBSI2. To clear the alarm in the FPGA, the control command TTX_CLR_ALARMS must be sent. The FPGA reads the register whenever the TxALMINT pin goes active or every 48 ms.

Inside the TTX, the alarms are latching. Alarm bits in these registers remain in the alarmed state until a read of this register occurs after the alarm condition has been cleared for greater than or equal to the deactivation time. Note that on power-up, many alarms may be set until the module completes initialization. The FR will read the register but maintain the alarm inside the FPGA until cleared from the CAN bus.

4.3.2 TTX – Hardwired Monitor and Control points

Monitor: The laser power monitor (LsPOWMON), laser temperature monitor (LsTEMPMON) and laser bias monitor (LsBIASMON) values for each half-transponder can be read from the FR analog to digital converters (10 bits) as part of the READ_TEMP monitor points on the FR, or they can be read from the half-transponder itself. Each half-transponder has an internal 24-bit value and can be polled for the information through the I2C bus with the READ_TTX_TEMP RCA.

Control: A master reset will reset the hardwired control point TxReset, and all three half-transponders will be reset simultaneously. Each half-transponder has a TxFIFORES line which will reset the selected internal TTX FIFO. Each TTX can be individually selected to enable the laser.

4.3.3 TTX – I2C Accessed Monitor points

There is an internal state machine running inside the FPGA2 which continuously polls each TTX at the start of each TE (timing event). Internal TTX status and monitor points are read and the data is stored to DPRAM, where it can be read out by the operator through the CAN interface.

All monitor or control points are specified in [RD 01], which details how the high level protocol for the I2C bus can be accessed using the GET_I2C_DATA/ SET_I2C_CMD commands. Familiarity with the reference document is assumed for use of these commands, and the details of that document are not repeated here.

4.3.3.1 Send I2C Command

Every command for the I2C requires a command and length byte. Write commands optionally require data with the command and length. The TTX_I2C_CMD_CHn command is set up to take the command, length and optional data in one eight byte packet that is then translated to the I2C transfer protocol. Checksums and status bytes called out in [RD 01] are inserted automatically in the I2C transfer by the FPGA.

Function	Cmd	Length	Data0	Data1	Data2	Data3	Data4	Data5
Byte #	0	1	2	3	4	5	6	7

Table 1: Send I2C Command

4.3.3.2 Read I2C Command

After a read command is sent (using TTX_I2C_CMD_CHn), the data can be retrieved using the GET_TTX_I2C_DATA command. Data is retrieved in eight-byte packets. The first packet will contain the status and length in the first two bytes. Successive reads will contain data only. The data is returned as:

Table 2: Read I2C Data

Read 1	Status	Length	Data0	Data1	Data2	Data3	Data4	Data5
Read 2	Data6	Data7	Data8	Data9	Data10	Data11	Data12	Data13
Read 3	Data14	Data15	Data16	Data17	Data18	0	0	0
Byte #	0	1	2	3	4	5	6	7

Status is according to the definition in [RD 01]. Only data called out by the length byte is valid. Subsequent reads will read all zero. A TTX_I2C_CMD_CHn command will reset the read counter to the first read. The read command will read any channel.
NOTE: Current version of this code allows only 6 bytes to be returned.

4.4 Digitizer (DG)

The digitizer is the unit inside the DTX module, which takes the analog data from the antenna and turns it into digital signals. The digital signal is then sent to the FR at a rate of 16 bits at 250 MHZ. The digitizer has a control mode for testing which will send out predefined data instead of antenna data. The digitizer monitors its own internal voltages and reports this information to the FR.

The formatter has a separate SPI bus going from FPGA2 to the DG. See [AD 06] for details of the communication between the DG and FR. The formatter polls the DG at the same rate as the timing event (48 ms) for information and stores that information in the FPGA for readout by the AMBSI2.

4.5 Summary of Monitor Points

Monitor data shall be polled by the ALMA bus master according to the protocol specified in [AD 04].

4.5.1 Table 1: Summary of DG Monitor Points

NAME	*Relative CAN Word Address*	*Data Size (bytes)*	*Suggested Interval*	*Timing Event Related ?*
GET_DG_3_3_V	*0x0 25 01*	*1*	*10 sec*	*No*
GET_DG_5_V	*0x0 25 02*	*1*	*10 sec*	*No*
GET_DG_TEMP	*0x0 25 03*	*1*	*10 sec*	*No*
GET_DG_MODE	*0x0 25 04*	*1*	*Debug*	*No*
GET_DG_FW_VER	*0x0 25 05*	*1*	*Startup*	*No*
GET_DG_VH1	*0x0 25 06*	*1*	*Debug*	*No*
GET_DG_VL1	*0x0 25 07*	*1*	*Debug*	*No*
GET_DG_VH2	*0x0 25 08*	*1*	*Debug*	*No*
GET_DG_VL2	*0x0 25 09*	*1*	*Debug*	*No*
GET_DG_SN_MSB	*0x0 25 0A*	*1*	*Startup*	*No*
GET_DG_SN_LSB	*0x0 25 0B*	*1*	*Startup*	*No*
GET_DG_PS_ON_OFF	*0x0 25 0D*	*1*	*Startup*	*No*

4.5.2 Table 2: Summary of FR Monitor Points

NAME	*Relative CAN Word Address*	*Data Size (bytes)*	*Suggested Interval (secs)*	*Timing Event Related?*
GET_FR_1_5_V	*0x0 16 00*	*6*	*300*	*No*
GET_FR_1_8_V	*0x0 16 01*	*6*	*300*	*No*
GET_FR_BOARD_VOLTAGE	*0x0 16 02*	*8*	*300*	*No*
GET_FR_TMP	*0x0 16 03*	*8*	*300*	*No*
GET_FR_LASER_PWR	*0x0 16 04*	*6*	*Debug*	*No*
GET_FR_LASER_BIAS	*0x0 16 05*	*6*	*Debug*	*No*
GET_FR_STATUS	*0x0 20 00*	*3*	*0.048*	*No*
GET_FR_CW_CH1	*0x0 10 01*	*1*	*Debug*	*No*
GET_FR_CW_CH2	*0x0 20 01*	*1*	*Debug*	*No*
GET_FR_CW_CH3	*0x0 30 01*	*1*	*Debug*	*No*
GET_FR_TE_STATUS	*0x0 20 02*	*4*	*0.048*	*Yes*
GET_FR_48_V	*0x0 40 03*	*1*	*Debug*	*No*
GET_FR_FPGA_FW_VER_CH1	*0x0 10 04*	*8*	*Startup*	*No*
GET_FR_FPGA_FW_VER_CH2	*0x0 20 04*	*8*	*Startup*	*No*
GET_FR_FPGA_FW_VER_CH3	*0x0 30 04*	*8*	*Startup*	*No*
GET_FR_PAYLOAD_LO_CH1	*0x0 10 05*	*8*	*Debug*	*Yes*
GET_FR_PAYLOAD_LO_CH2	*0x0 20 05*	*8*	*Debug*	*Yes*
GET_FR_PAYLOAD_LO_CH3	*0x0 30 05*	*8*	*Debug*	*Yes*
GET_FR_PAYLOAD_HI_CH1	*0x0 10 06*	*8*	*Debug*	*Yes*
GET_FR_PAYLOAD_HI_CH2	*0x0 20 06*	*8*	*Debug*	*Yes*
GET_FR_PAYLOAD_HI_CH3	*0x0 30 06*	*8*	*Debug*	*Yes*
GET_FR_PAYLOAD_STATUS	*0x0 20 07*	*1*	*Debug*	*Yes*
GET_FR_PHASE_SEQ_A	*0x0 10 07*	*8*	*Initialize*	*Yes*
GET_FR_PHASE_SEQ_B	*0x0 10 08*	*8*	*Initialize*	*Yes*
GET_FR_PHASE_OFFSET	*0x0 10 09*	*3*	*As needed*	*Yes*

NAME	*Relative CAN Word Address*	*Data Size (bytes)*	*Suggested Interval (secs)*	*Timing Event Related?*
GET_FR_RNG_CH1	*0x0 10 0A*	*1*	*Debug*	*Yes*
GET_FR_RNG_CH2	*0x0 20 0A*	*1*	*Debug*	*Yes*
GET_FR_RNG_CH3	*0x0 30 0A*	*1*	*Debug*	*Yes*
GET_FR_INPUT _TEST _CH1	*0x0 10 0B*	*1*	*Debug*	*No*
GET_FR_INPUT _TEST _CH2	*0x0 20 0B*	*1*	*Debug*	*No*
GET_FR_INPUT _TEST _CH3	*0x0 30 0B*	*1*	*Debug*	*No*
GET_FR_EEPROM_DATA	*0x0 20 0C*	*2*	*Debug*	*No*
GET_FR_SWITCH_CH1	0x0 10 0E	1	Startup	No
GET_FR_SWITCH_CH2	0x0 20 0E	1	Startup	No
GET_FR_SWITCH_CH3	0x0 30 0E	1	Startup	No
GET_FR_LRU_CIN	*0x0 7F FF*	*8*	*Startup*	*No*

4.5.3 Table 3: Summary of TTX Monitor Points

NAME	*Relative CAN Word Address*	*Data Size (bytes)*	*Suggested Interval*	*Timing Event Related?*
GET_TTX_ALARM_STATUS	*0x0 24 01*	*6*	*0.048 secs*	*No*
GET_TTX_LASER_BIAS_CH1	*0x0 21 01*	*3*	*As needed*	*No*
GET_TTX_LASER_BIAS_CH2	*0x0 22 01*	*3*	*As needed*	*No*
GET_TTX_LASER_BIAS_CH3	*0x0 23 01*	*3*	*As needed*	*No*
GET_TTX_LASER_PWR_CH1	*0x0 21 02*	*3*	*As needed*	*No*
GET_TTX_LASER_PWR_CH2	*0x0 22 02*	*3*	*As needed*	*No*
GET_TTX_LASER_PWR_CH3	*0x0 23 02*	*3*	*As needed*	*No*
GET_TTX_LASER_TMP_CH1	*0x0 21 03*	*3*	*As needed*	*No*
GET_TTX_LASER_TMP_CH2	*0x0 22 03*	*3*	*As needed*	*No*
GET_TTX_LASER_TMP_CH3	*0x0 23 03*	*3*	*As needed*	*No*
GET_TTX_LASER_ENABLED	*0x0 24 05*	*1*	*As needed*	*No*
GET_TTX_I2C_DATA	*0x0 24 0E*	*8*	*Debug*	*No*

4.6 Summary of Control Points

Control data shall be transmitted by the ALMA bus master according to the protocol specified in [AD 04].

4.6.1 Table 4: Summary of DG Control Points

Name	***Relative CAN Address (hex)***	***Data Size (bytes)***	***Suggested Interval (secs)***	***Timing Event Related?***
SET_DG_PS_ON_OFF	*0x0 A5 0C*	*1*	*As needed*	*No*
SET_DG_TEST_PAT	*0x0 A5 A0*	*1*	*As needed*	*No*
SET_DG_VMAG1	*0x0 A5 C1*	*1*	*As needed*	*No*
SET_DG_VOFF1	*0x0 A5 C2*	*1*	*As needed*	*No*
SET_DG_VMAG2	*0x0 A5 C3*	*1*	*As needed*	*No*
SET_DG_VOFF2	*0x0 A5 C4*	*1*	*As needed*	*No*
SET_DG_250MHZ_DELAY	*0x0 A5 D1*	*1*	*As needed*	*No*

4.6.2 Table 5: Summary of FR Control Points

Name	***Relative CAN Address (hex)***	***Data Size (bytes)***	***Suggested Interval (secs)***	***Timing Event Related?***
FR_RESET_CH1	*0x0 90 00*	*1*	*As needed*	*No*
FR_RESET_CH2	*0x0 A0 00*	*1*	*As needed*	*No*
FR_RESET_CH3	*0x0 B0 00*	*1*	*As needed*	*No*
FR_RESET_ALL	*0x0 C0 00*	*1*	*As needed*	*No*
SET_FR_CW_CH1	*0x0 90 01*	*1*	*Reconfig*	*No*
SET_FR_CW_CH2	*0x0 A0 01*	*1*	*Reconfig*	*No*
SET_FR_CW_CH3	*0x0 B0 01*	*1*	*Reconfig*	*No*

SET_FR_CW_ALL	*0x0 C0 01*	*1*	*Reconfig*	*No*
FR_TE_RESET	*0x0 A0 02*	*1*	*As needed*	*Yes*
SET_FR_48_VOLTS	*0x0 C0 03*	*1*	*As needed*	*No*
FR_RELOAD_FPGA	*0x0 C0 04*	*1*	*As needed*	*No*
SET_FR_PHASE_SEQ_A	*0x0 90 07*	*8*	*As needed*	*Yes*
SET_FR_PHASE_SEQ_B	*0x0 90 08*	*8*	*As needed*	*Yes*
SET_FR_PHASE_OFFSET	*0x0 90 09*	*3*	*As needed*	*Yes*
SET_FR_RNG_CH1	*0x0 90 0A*	*1*	*Reconfig*	*No*
SET_FR_RNG_CH2	*0x0 A0 0A*	*1*	*Reconfig*	*No*
SET_FR_RNG_CH3	*0x0 B0 0A*	*1*	*Reconfig*	*No*
SET_FR_RNG_ALL	*0x0 C0 0A*	*1*	*Reconfig*	*No*
SET_FR_INPUT_TEST_CH1	*0x0 90 0B*	*1*	*Reconfig*	*No*
SET_FR_INPUT_TEST_CH2	*0x0 A0 0B*	*1*	*Reconfig*	*No*
SET_FR_INPUT_TEST_CH3	*0x0 B0 0B*	*1*	*Reconfig*	*No*
SET_FR_INPUT_TEST_ALL	*0x0 C0 0B*	*1*	*Reconfig*	*No*
FR_EEPROM_PROG	*0x0 A0 0C*	*1*	*Reconfig*	*No*
FR_EEPROM_FETCH	*0x0 A0 0D*	*1*	*Reconfig*	*No*
FR_CAPTURE_PAYLOAD	*0x0 C0 0F*	*1*	*As needed*	*No*

4.6.3 Table 6: Summary of TTX Control Points

NAME	***Relative CAN Word Address***	***Data Size (bytes)***	***Suggested Interval***	***Timing Event Related?***
TTX_RESET	*0x0 A4 00*	*1*	*As needed*	*No*
TTX_CLR_ALARMS	*0x0 A4 01*	*1*	*As needed*	*No*
TTX_RESET_FIFO	*0x0 A4 04*	*1*	*As needed*	*No*

NAME	*Relative CAN Word Address*	*Data Size (bytes)*	*Suggested Interval*	*Timing Event Related?*
TTX_LASER_ENABLE	*0x0 A4 05*	*1*	*As needed*	*No*
TTX_I2C_CMD_CH1	*0x0 A1 0E*	*4*	*Debug*	*No*
TTX_I2C_CMD_CH2	*0x0 A2 0E*	*4*	*Debug*	*No*
TTX_I2C_CMD_CH3	*0x0 A3 0E*	*4*	*Debug*	*No*

4.7 Commands in Detail

4.7.1 DG Monitor Point in Detail

4.7.1.1 GET_DG_3_3_V

Relative CAN Address	*0x0 25 01*
Description	*3.3 V voltage reading from the DG*
Suggested Interval	*10 seconds*
Update Rate	*720 msec*
TE Related	*No*
Data	*1 byte, 8 bits:* *Bits 0 – 7: Filtered +3.3 VDC supply.* *The conversion is voltage (V) = byte * 0.021152 V.* *Operating Range: 3.1V – 3.5V*

4.7.1.2 GET_DG_5_V

Relative CAN Address	*0x0 25 02*
Description	*5 V Voltage reading from the DG*
Suggested Interval	*10 seconds*
Update Rate	*720 msec*
TE Related	*No*
Data	*1 byte, 8 bits:* *Bits 0-7: +5 VDC used by filtered voltages.* *The conversion is voltage (V) = byte * 0.032102 V.* *Operating Range: 4.8V – 5.2V*

4.7.1.3 GET_DG_TEMP

Relative CAN Address	*0x0 25 03*
Description	*Temperature from the DG*
Suggested Interval	*10 seconds*
Update Rate	*720 msec*
TE Related	*No*
Data	*1 byte, 8 bits:* *Bits 0-7: Temperature inside the digitizer assembly. The sensor is on a 'service' board close to DGS sub-assemblies and DGD multi-chip boards.* *The conversion is voltage (V) = byte *0.287013 ° C.* *Operating Range: 10 to 40 °C*

4.7.1.4 GET_DG_MODE

Relative CAN Address	*0x0 25 04*
Description	*DG test mode status*
Suggested Interval	*As needed*
Update Rate	*720 msec*
TE Related	*No*
Data	*1 byte, 8 bits:* *Bit 0:* *1 = test mode active* *0 = normal mode* *Bits 1-7: reserved*

4.7.1.5 GET_DG_FW_VER

Relative CAN Address	*0x0 25 05*
Description	*DG Firmware version*
Suggested Interval	*As needed*
Update Rate	*720 msec*
TE Related	*No*
Data	*1 byte, 8 bits:* *Bits 7-4: Major Revision number* *Bits 3-0: Minor Revision number*

4.7.1.6 GET_DG_VH

Relative CAN Address	*0x0 25 06*	*Phase 1 – Reference voltage*
Relative CAN Address	*0x0 25 08*	*Phase 2 – Reference voltage*
Description	*Reference voltage reading from the DG* *VH1 = Phase 1(BB0) Reference voltage High* *VH2 = Phase 2 (BB1) Reference voltage High*	
Suggested Interval	*As needed*	
Update Rate	*720 msec*	
TE Related	*No*	
Data	*1 byte, 8 bits:* *Bits 0 – 7: Reference voltage 0 to 3.3V.* *The conversion is voltage (V) = byte * 0.012891.* *Operating Range: 00 - FF*	

4.7.1.7 GET_DG_VL

Relative CAN Address	*0x0 25 07*	*Phase 1 – Reference voltage*
Relative CAN Address	*0x0 25 09*	*Phase 2 – Reference voltage*
Description	*Reference voltage reading from the DG* *VH1 = Phase 1(BB0) Reference voltage Low* *VH2 = Phase 2 (BB1) Reference voltage Low*	
Suggested Interval	*As needed*	
Update Rate	*720 msec*	
TE Related	*No*	
Data	*1 byte, 8 bits:* *Bits 0 – 7: Reference voltage 0 to 3.3V.* *The conversion is voltage (V) = byte * 0.012891.* *Operating Range: 00 - FF*	

4.7.1.8 GET_DG_SN

Relative CAN Address	*0x0 25 0A*	*SN MSB*
Relative CAN Address	*0x0 25 0B*	*SN LSB*
Description	*Serial number for DG, 16 bits total*	
Suggested Interval	*Startup*	
Update Rate	*720 msec*	
TE Related	*No*	
Data	*1 byte, 8 bits:* *Bits 7 – 0:*	

4.7.1.9 GET_DG_PS_ON_OFF

Relative CAN Address	*0x0 25 0D*
Description	*Read power supply status*
Suggested Interval	*Startup*
Update Rate	*720 msec*
TE Related	*No*
Data	*1 byte, 8 bits :* *Bit 0: 1 = power supply ON, 0 = power supply OFF*

4.7.2 DG Control Points in Detail

4.7.2.1 SET_DG_PS_ON_OFF

Relative CAN Address	*0x0 A5 0C*
Description	*Read digitizer power supply mode*
Suggested Interval	*As needed*
TE Related	*No*
Data	*1 byte, 8 bits :* *Bit 0: 1 = power supply ON, 0 = power supply OFF*

4.7.2.2 SET_DG_TEST_PAT

Relative CAN Address	*0x0 A5 A0*
Description	*Put digitizer into feedback test pattern. When in this mode the digitizer will ignore incoming data and latch the data that is currently in the digitizers buffers for feedback to its own input. The pattern for each bit is 00110011 and will repeat at a 62.5 MHz rate. (See [AD 06])*
Suggested Interval	*As needed*
TE Related	*No*
Data	*1 byte, 8 bits :* *Bit 0: 1 = test mode ON, 0 = test mode OFF*

4.7.2.3 SET_DG_VMAG/VOFF

Relative CAN Address	*0x0 A5 C1*	*Phase 1 – VMAG1*
Relative CAN Address	*0x0 A5 C2*	*Phase 1 – VOFF1*
Relative CAN Address	*0x0 A5 C3*	*Phase 2 – VMAG2*
Relative CAN Address	*0x0 A5 C4*	*Phase 2 – VOFF2*
Description	*Set reference voltage for Phase 1(BB0) or Phase 2(BB1)* *VMAG = Quantum voltage = VH – VL* *VOFF = Average voltage = (VH + VL) / 2.* *(See Sections 4.7.1.6 and 4.7.1.7 and [AD 06])*	
Suggested Interval	*As needed*	
TE Related	*No*	
Data	*1 byte, 8 bits :* *Bits 7- 0:* *0x01 = step +150 µV (fine adjustment)* *0x02 = step +4.13 mV (coarse adjustment)* *0x11 = step -150 µV (fine adjustment)* *0x12 = step -4.13 mV (coarse adjustment)*	

4.7.2.4 SET_DG_250MHZ_DELAY

Relative CAN Address	*0x0 A5 D1*
Description	*Set phase of the 250MHz clock signal for both Phase 1 (BB0) and Phase 2 (BB1) of DG assembly.* *(See [AD 06])*
Suggested Interval	*As needed*
TE Related	*No*
Data	*1 byte, 8 bits :* *Bits 7- 0:* *0x01 = step +0.05 V (fine adjustment)* *0x02 = step +0.25 V (coarse adjustment)* *0x11 = step -0.05 V (fine adjustment)* *0x12 = step -0.25 V (coarse adjustment)* *Note : To adjust the delay of the 250MHz clock, adjust the reverse voltage applied to a varicap diode. The relationship between the capacitance and the reverse voltage is not linear.*

4.7.3 FR Monitor Points in Detail

4.7.3.1 GET_FR_1_5_V

The ADC is connected to FPGA 1; therefore, all ADC monitor points are addressed through FPGA1.

Relative CAN Address	*0x0 16 00*
Description	*1.5 V Voltage monitor point for FPGAs 1, 2, 3.*
Suggested Interval	*300 secs*
Update Rate	*48 msec*
TE Related	*No*
Data	*6 bytes, 48 bits: 3-16 bit* *Byte 0-1 bits 0 – 15: Ch 1 1.5 Volts.* *The conversion is voltage (V) = count * 2.44e-3 V/cnt.* *Operating Range: 1.425V – 1.575V* *Byte 2-3 bits 16-31: Ch 2 1.5 Volts.* *The conversion is voltage (V) = count * 2.44e-3 V/cnt.* *Operating Range: 1.425V – 1.575V* *Byte 4-5 bits 32-47: Ch 3 1.5 Volts.* *The conversion is voltage (V) = count * 2.44e-3 V/cnt.* *Operating Range: 1.425V – 1.575V*

4.7.3.2 GET_FR_1_8_V

Relative CAN Address	*0x0 16 01*
Description	*1.8 V Voltage monitor point for half-transponders 1, 2, 3.*
Suggested Interval	*300 sec*
Update Rate	*48 msec*
TE Related	*No*
Data	*6 bytes, 48 bits: 3-16 bit* *Byte 0-1 bits 0 – 15: Ch 1 1.8 Volts.* *The conversion is voltage (V) = count * 2.44e-3 V/cnt.* *Operating Range: 1.71V – 1.89V* *Byte 2-3 bits 16-31: Ch 2 1.8 Volts.* *The conversion is voltage (V) = count * 2.44e-3 V/cnt.* *Operating Range: 1.71V – 1.89V* *Byte 4-5 bits 32-47: Ch 3 1.8 Volts.* *The conversion is voltage (V) = count * 2.44e-3 V/cnt.* *Operating Range: 1.71V – 1.89V*

4.7.3.3 GET_FR_BOARD_VOLTAGE

Relative CAN Address	*0x0 16 02*
Description	*Voltage monitor point for board voltages.*
Suggested Interval	*300 sec*
Update Rate	*48 msec*
TE Related	*No*
Data	*8 bytes, 64 bits: 3-16 bit* *Bytes 0-1 bits 0 – 15: 3.3 Volts.* *The conversion is voltage (V) = count * 4.6115e-3 V/cnt.* *Operating Range: 3.13V – 3.47V* *Bytes 2-3 bits 16-31: 15.0 Volts.* *The conversion is voltage (V) = count * 24.821e-3 V/cnt.* *Operating Range: 13.5V – 16.0V* *Bytes 4-5 bits 32-47: 5.0 Volts.* *The conversion is voltage (V) = count * 8.3008e-3 V/cnt.* *Operating Range: 4.5V – 5.5V* *Byte 6 bit 48: -5.2 Volts Present* *-5.2 Volts is present if clear (=0)* *Byte 6 bits 49-55: reserved* *Byte 7 bits 56-63: reserved.*

4.7.3.4 GET_FR_TMP

Relative CAN Address	*0x0 16 03*
Description	*Temperature monitor point for FR and TTX.*
Suggested Interval	*300 sec*
Update Rate	*48 msec*
TE Related	*No*
Data	*8 bytes, 64 bits: 4-16 bit* *Bytes 0-1 bits 0 – 15: FR Temp.* *The conversion is temperature = count * 0.244 °C/cnt.* *Operating Range: 2°C - 60°C* *Bytes 2-3 bits 16-31: TTX1 Temp[1].* *The conversion is:* *temperature = (0x400 - count) * 0.0976 °C/cnt.* *Operating Range: ± 1°C* *Bytes 4-5 bits 48-55: TTX2 Temp[1].* *The conversion is:* *temperature = (0x400 – count) * 0.0976 °C/cnt.* *Operating Range: ± 1°C* *Bytes 6-7 bits 56-63: TTX3 Temp[1].* *The conversion is:* *temperature = (0x400 – count) * 0.0976 °C/cnt.* *Operating Range: ±1 °C*

[1]These values are DEBUG values. Temperature values for TTX components can also be read from the TTX RCA as an internal soft value. Those temperatures are more accurate than those read from the ADC on board the FR.

These temperature values read for the half-transponders show a delta from the initial temperature reading on startup. The value at startup should read 0x400 (half the full range). A change of ± 1 degree C indicates an alarm condition – this condition is also indicated in the READ_ALARM_STATUS indicator LsTEMPALM.

4.7.3.5 GET_FR_LASER_PWR

Relative CAN Address	*0x0 16 04*
Description	*Laser power monitor point for TTX.*
Suggested Interval	*300 sec*
Update Rate	*48 msec*
TE Related	*No*
Data	*6 bytes, 48 bits: 3-16 bit* *Bytes 0-1 bits 0 – 15: TTX1 Optical Output Power [1]* *The conversion is power(dBm) = count * 2.44e-3 dBm/cnt* *Operating Range: > 0.25dBm* *Bytes 2-3 bits 16-31: TTX2 Optical Output Power [1]* *The conversion is power(dBm) = count * 2.44e-3 dBm/cnt .* *Operating Range: > 0.25 dBm* *Bytes 4-5 bits 32-47: TTX3 Bias Optical Output Power [1]* *The conversion is power(dBm) = count * 2.44e-3 dBm/cnt .* *Operating Range:> 0.25 dBm*

[1] Laser power output values for TTX components can also be read from the TTX RCA as an internal soft value. The laser power here is in terms of dBm and the actual operating range is specified by the manufacturer to be 50% of the initial value. The initial value is assumed to be 0.5 dBm (0.5V) here. Initial value should be read at startup.

4.7.3.6 GET_FR_LASER_BIAS

Relative CAN Address	*0x0 16 05*
Description	*Laser bias monitor point for TTX.*
Suggested Interval	*300 sec*
Update Rate	*48 msec*
TE Related	*No*
Data	*6 bytes, 48 bits: 3-16 bit* *Bytes 0-1 bits 0 – 15: TTX1 Bias Current[1]* *The conversion is current(A) = count *122e-6 A/cnt.* *Operating Range: 35mA – 105mA* *Bytes 2-3 bits 16-31: TTX2 Bias Current[1].* *The conversion is current(A) = count *122e-6 A/cnt.* *Operating Range: 35mA – 105mA* *Bytes 4-5 bits 48-55: TTX3 Bias Current[1].* *The conversion is current(A) = count *122e-6 A/cnt.* *Operating Range: 35mA – 105mA*

[1]Laser bias output values for TTX components can also be read from the TTX RCA as an internal soft value. The laser bias here is in terms of milliamps. The actual operating range is specified by the manufacturer to be 50% of the initial value. The initial value is assumed to be 70 mA here. Initial value should be read at startup.

4.7.3.7 GET_FR_ STATUS

Relative CAN Address	*0x0 20 00*
Description	*Read the status register.*
Suggested Interval	*0.048 sec*
Update Rate	*Continuous*
TE Related	*No*
Data	*2 bytes, 16 bits* *Byte 1* *Bit 0: OverPower - If set, Keep Alive signal is present* *Bit 1: REFCLK - If set, 48 mS TE present and synced* *Bit 2: If set, 250MHZ PLL in lock Ch1-BitD* *Bit 3: If set, 125MHZ PLL in lock Ch1-BitD* *Bit 4: If set, 250MHZ PLL in lock Ch2-BitC* *Bit 5: If set, 125MHZ PLL in lock Ch2-BitC* *Bit 6: If set, 250MHZ PLL in lock Ch3-BitB* *Bit 7: If set, 125MHZ PLL in lock Ch3-BitB* *Byte 2 Alarm* *Bit 0: Ch1 Bit D laser ON/OFF status (1 = ON)* *Bit 1: Ch2 Bit C laser ON/OFF status (1 = ON)* *Bit 2: Ch3 Bit B laser ON/OFF status (1 = ON)* *Bit 3: Ch1 TTX alarm (0 = alarm active)* *Bit 4: Ch2 TTX alarm (0 = alarm active)* *Bit 5: Ch3 TTX alarm (0 = alarm active)* *Bit 6: TTX Alarm (=0 when any TTX alarm active).* *Bit 7: always 1*

This monitor point indicates the lock status of various loops inside the FR. Each of these bits is latched so that if the loop goes out of lock for a short period of time, before returning to lock, the relevant bit will remain set. To clear the out-of-lock bits, an FR_RESET_CHn command with data = 0x01 must be sent. If this does not clear the bit, the operator must be notified.

The Alarm bits show the current status of the function they are monitoring, i.e. they are not latched. The TTX alarm bit will show an alarm anytime the lasers are turned off. TTX Alarm is a logical OR of bits 3-5 and can be monitored after turning on the lasers for a “ready” status. Once the alarm bit goes inactive, the lasers are transmitting. The individual TTX alarm bits correspond to their associated channel and are the logical OR of all alarms inside of a single TTX unit.

4.7.3.8 GET_FR_CW_Chi

Relative CAN Address	*0x0 10 01*	*CH 1 – bit D*
Relative CAN Address	*0x0 20 01*	*CH 2 – bit C*
Relative CAN Address	*0x0 30 01*	*CH 3 – bit B*
Description	*Control word that can change the output of the data stream. Used for debug and verification.*	
Suggested Interval	*Reconfiguration and debug*	
TE Related	*No*	
Data	*1 byte, 8 bits:* *See SET_FR_CW_CHi*	

4.7.3.9 GET_FR_TE_STATUS

Relative CAN Address	*0x0 20 02*	*CH2 – Bit C*
Description	*Read status of Timing Event registers. This RCA monitors the TE for changes in relative rise time. The code will synchronize the internal TE to an external TE if it is available. If the rising edge of the external TE is close to the rising edge of the DG clock, this monitor point will detect a minimum error (the clock is early) or a maximum error (the clock is late). If this happens, the TE should be clocked in using the opposite edge of clock (Byte 0, Bit 1). The current error is non-latching and will give an indication of whether or not the error is still occurring.*	
Suggested Interval	*0.048 sec*	
TE Related	*Yes*	

Data	*4 bytes:* *Byte 0:* *Bit 0: TE error flag, set when max or min are triggered.* *Bit 1: Clock edge status, 1 = using inverted clock edge to clock in Timing Event* *Bits 2-3: = 0* *Bits 4-7: =1111* *Byte 1:* *Bits 0-3: current error (signed 2's complement)* *Bits 4-7: = 0* *Byte 2:* *Bits 0-3: maximum error (signed 2's complement)* *Bits 4-7: = 0* *Byte 3:* *Bits0-3: minimum error (signed 2's complement)* *Bits 4-7: = 0*

4.7.3.10 GET_FR_48_V

Relative CAN Address	*0x0 40 03*	*All channels*
Description	*Read status of main power to formatter and digitizer*	
Suggested Interval	*Debug*	
TE Related	*No*	
Data	*1 byte:* *bit 0: 1= 48V ON, 0 = 48V OFF* *bit 1: 1 = 125MHZ missing* *bits 2 – 7: Reserved*	

4.7.3.11 GET_FR_FPGA_FW_VERi

Relative CAN Address	*0x0 10 04*	*CH1 – bit D*
Relative CAN Address	*0x0 20 04*	*CH2 – bit C*
Relative CAN Address	*0x0 30 04*	*CH3 – bit B*
Description	*FPGA Firmware Version*	
Suggested Interval	*At startup*	
TE Related	*No*	
Data	*8 bytes:* *Byte 0 – Hex'BE' (Back End)* *Byte 1 – Module type ID (x40 for transmitter)* *Byte 2 – Revision M.m: M=4 bit Major revision; m=4 bit minor revision* *Byte 3 – Day (0-31) day of revision compilation* *Byte 4 – Month (0-12) month of revision compilation* *Byte 5 – Year (00-99) year of revision compilation* *Byte 6 – undefined* *Ch 1 – Bit D – zero* *Ch 2 – Bit C – Formatter serial number (MSB)* *Ch 3 – Bit B – zero* *Byte 7 –* *Ch 1 – Bit D – reserved* *Ch 2 – Bit C – Formatter serial number (LSB)* *Ch 3 – Bit B – reserved*	

4.7.3.12 GET_FR_PHASE_SEQ_A/B

Relative CAN Address	*0x0 10 07*	*A*	*CH1 – Bit D*
Relative CAN Address	*0x0 10 08*	*B*	
Description	*This is a read back of the SET_PHASE_SEQ_1 command. It is used to verify that the DTX has received the previous SET_PHASE_SEQ command. It will return all zeros if the SET_PHASE_SEQ command has never been called. See section 4.2.3 for a description of how this sequence affects the D (sign) bits.*		
Suggested Interval	*Initialization*		
TE Related	*Yes*		
Data	*8 bytes: 128-bit sequence that should correspond to a unique Walsh function* *Byte 1: Most significant byte* *Byte 8: Least significant byte*		
	A	*The first half of any valid Walsh function in the range: 0x0000000000000000 to 0xFFFFFFFFFFFFFFFF. This half will be used first beginning with the least significant bit.*	
	B	*The second half of any valid Walsh function in the range: 0x0000000000000000 to 0xFFFFFFFFFFFFFFFF*	

4.7.3.13 GET_FR_PHASE_OFFSET

Relative CAN Address	*0x0 10 09*	*CH1 – Bit D*
Description	*This is the read back of SET_PHASE_OFFSET. It is the amount by which the phase switching is delayed. The delay of the first phase switching value is a minimum of 8 ms and a maximum of 16.388608 ms with 8 ns of resolution.* *This monitor will return 8ms (or zero) if the SET_PHASE_OFFSET control point has never been called, i.e., after power up.*	
Suggested Interval	*0.048 secs.*	
TE Related	*Yes*	
Data	*3 bytes:* *Byte 1: Most Significant byte* *Byte 3: Least Significant byte* *bits 23 - 20: Unused and always set to zero* *bits 19 – 0: Delay Value*	

Conversion Factor	*0x00 00 00 – 0x0F FF FF -> 8ms – 16.3886ms (approximately as the actual step size is 8ns per bit).*
Operating Range	*8 – 16.3886 ms*

4.7.3.14 GET_FR_PAYLOAD

Relative CAN Address	*0x0 10 05 / 0x0 10 06*	*CH1 – bit D*
Relative CAN Address	*0x0 20 05 / 0x0 20 06*	*CH2 – bit C*
Relative CAN Address	*0x0 30 05 / 0x0 30 06*	*CH3 – bit B*
Description	*Read internal FIFO payload data*	
Suggested Interval	*Debug*	
TE Related	*Yes*	
Data	*8 bytes* *N005 = read lower 64 bits of FIFO* *N006 = read upper 64 bits of FIFO*	

4.7.3.15 GET_FR_PAYLOAD_STATUS

Relative CAN Address	*0x0 20 07*	*CH2 – bit C*
Description	*Read internal payload FIFO status. In order to get sequential data, the fifo should not be read until the fifo full flag has gone high after sending the FR_CAPTURE_PAYLOAD command.*	
Suggested Interval	*Debug*	
TE Related	*Yes*	
Data	*1 byte:* *Bit 0: Ch 1 fifo empty* *Bit 1: Ch1 fifo full* *Bit 2: Ch2 fifo empty* *Bit 3: Ch2 fifo full* *Bit 4: Ch3 fifo empty* *Bit 5: Ch3 fifo full* *Bits 6-7: not used.*	

4.7.3.16 GET_FR_RNG

Relative CAN Address	*0x0 10 0A*	*CH1 – bit D*
Relative CAN Address	*0x0 20 0A*	*CH2 – bit C*
Relative CAN Address	*0x0 30 0A*	*CH3 – bit B*
Description	*Read RNG mode register setting*	
Suggested Interval	*Debug*	
TE Related	*Yes*	
Data	*1 byte:* *Bit 1-0: RNG mode* *00 = normal mode, RNG OFF* *01 = test mode, RNG ON* *10 =RNG OFF, counting test pattern sent* *11 =RNG ON, re-start (re-seed) every Timing Event* *Bits 2-7: not used*	

4.7.3.17 GET_FR_INPUT_TEST

Relative CAN Address	*0x0 10 0B*	*CH1 – bit D*
Relative CAN Address	*0x0 20 0B*	*CH2 – bit C*
Relative CAN Address	*0x0 30 0B*	*CH3 – bit B*
Description	*Read input test mode register setting.*	
Suggested Interval	*Debug*	
TE Related	*No*	
Data	*1 byte: See SET_FR_INPUT_TEST*	

4.7.3.18 GET_FR_EEPROM_DATA

Relative CAN Address	*0x0 20 0C*	*CH2 – bit C only*
Description	*Read data from the EEPROM. If this command is preceded by a FR_EEPROM_FETCH then the data will be the byte programmed into the EEPROM at the address given in the FR_EEPROM_FETCH command. If the FR_EEPROM_FETCH does not precede this command, the data returned will be the status register of the EEPROM device.*	
Suggested Interval	*Debug*	
TE Related	*No*	
Data	*1 byte: = data programmed in EEPROM or EEPROM status register (see FR_EEPROM_FETCH).*	

4.7.3.19 GET_FR_SWITCH

Relative CAN Address	*0x0 10 0E*	*CH1 – bit D*
Relative CAN Address	*0x0 20 0E*	*CH2 – bit C*
Relative CAN Address	*0x0 30 0E*	*CH3 – bit B*
Description	*Switch setting: show the value of the test switch setting (see Section 4.2.9 for switch setting options)*	
Suggested Interval	*Startup*	
TE Related	*No*	
Data	*1 byte:* *Bits 7-0: 0 = Normal, 1 = Test Mode* *Bit 7 – if this bit is set, the output data to the correlator can be affected. To override this switch setting use the command.*	

4.7.3.20 GET_FR_LRU_CIN

Relative CAN Address	*0x0 7F FF*	*Transmitter*
Description	*Read the Line Replaceable Unit's CIN*	
Suggested Interval	*Startup*	
TE Related	*No*	
Data	*8 bytes:* *Byte 0-3: LRU CIN to the 4th level (53.08.00.00)* *Byte 4-5: LRU Serial Number* *Byte 6: LRURevision 1=A, 2=B etc.* *Byte 7: Reserved*	

4.7.4 FR Control Points in Detail

4.7.4.1 FR_RESET_Chi

Relative CAN Address	*0x0 90 00*	*CH1 – bit D*
Relative CAN Address	*0x0 A0 00*	*CH2 – bit C*
Relative CAN Address	*0x0 B0 00*	*CH3 – bit B*
Relative CAN Address	*0x0 C0 00*	*ALL bits*
Description	*Initiate a software reset of the indicated FPGA. This command will reset all functions inside the FPGA (lasers are turned off when Ch 2 is reset). It should not be issued indiscriminately.*	
Suggested Interval	*As needed*	
TE Related	*No*	
Data	*1 byte* *Bit 0: When equal to 0x1, Reset status bits* *Bits 1-2: Not used.* *Bit 3: Over ride switch position setting* *Bits 4-7: When equal to 0xF, Reset entire chip* *All other values will have no effect.*	

This command has three distinct uses:

1. To reset alarms which have been set, Bit 0 is used to reset the status bits without affecting data. This bit should be set as part of regular maintenance and debugging.
2. To override the switch setting on the FR board. Switches should all be in the inactive state (= '1') when in the field. Switches are set during maintenance and lab debugging. If a switch which controls data were accidentally left in an active state (='0') this bit can be set to disable the FR from using the switch setting.
3. For debug and startup purposes, bits 4-7 can be set to send a "system" reset to the FR. Each FPGA will reset all registers (except those noted in the Control Points in Detail section) to re-configure the FPGA to a defined startup condition. This reset will interrupt data.

4.7.4.2 SET_FR_CW_CHi

Relative CAN Address	*0x0 90 01*	*CH1 – bit D*
Relative CAN Address	*0x0 A0 01*	*CH2 – bit C*
Relative CAN Address	*0x0 B0 01*	*CH3 – bit B*
Relative CAN Address	*0x0 C0 01*	*All channels – all bits*
Description	*Select and activate encoded test patterns. Switches processing from digitizer data to a pre defined pattern. Used for debugging hardware and, as described in section 4.2.1, to ensure the correct connection between a DTS transmitter and a DTS receiver module.*	
Suggested Interval	*Reconfiguration and debug*	
TE Related	*No*	
Data	*1 byte, 8 bits:* *Bits 0-2: change Payload data* *2 1 0 bits* *0 0 0 normal payload from framebuilder* *0 0 1 payload = all zeroes* *0 1 0 payload = all ones* *0 1 1 payload = 0xFFFF0000 x 4* *1 0 0 payload = 0xCCCC x 8* *1 0 1 payload = 0x3333 x 8* *1 1 0 payload = 0xAAAA5555 x 4* *1 1 1 payload = F(cntr) E(cntr) D(cntr) … 1(cntr) 0(cntr) – cntr is a four-bit counter which counts continuously* *Bit 3: parity* *0 = normal, parity enabled* *1 = parity bits disabled (= 0)* *Bits 4-6: change frame data.* *6 5 4 bits* *0 0 0 normal frame* *0 0 1 frame = alternating 0x0000FFFF x 4* *0 1 0 frame = alternating 0xFFFF0000 x 4* *0 1 1 frame = 0xAAAA5555 x 4* *1 0 0 frame = all ones* *1 0 1 frame = all zeros* *1 1 0 frame = 0xCCCC x 8* *1 1 1 frame = 0x3333 x 8* *Bit 7: change scramble code –ALMA, 8 bit select EVLA* *0 = normal – use scramble code* *1 = scramble code = 0*	

The difference between changing the payload data and changing the frame data is where the data gets inserted. Payload data is inserted at the beginning of the stream, while frame data is inserted at the end. If change frame data is active, this will overwrite the payload data (even manufactured payload data.) Parity is inserted after payload data and will be overwritten anytime frame data is not normal.

When bit 3 = 1, the parity bits will be zeroed (this will have no effect if bits 4-6 are set.)

When bit 7 = 1, the scramble code is zeroed.

When bits 0 and 2 are not zero, data representing digitizer data is changed. The rest of the frame will be as shown in Figure 1 (assuming no other control bits are set.)

When bits 5 and 6 are not zero, the entire frame is changed to the data shown above; from Sync bit 0 to checksum bit 159 (see Figure 1). These bits override all other control bits in this command.

If the switches located on the FR board are not all reading '1', the FR board may be set to output test data on startup. FR switches are meant only for test and debug and should not be set before deploying a unit to the field. If a switch has been inadvertently left active, the switch position setting can be overridden with the FR_RESET_Chi (Section 4.7.4.1) command.

4.7.4.3 FR_TE_RESET

Relative CAN Address	*0x0 A0 02*	*CH2 – bit C*
Description	*Reset the TE status registers or change the TE input clock edge or resync the internal TE to the external TE.*	
Suggested Interval	*As needed*	
TE Related	*Yes*	
Data	*1 byte* *Bit 0: Reset TE error counter and latch* *Bit 1: 0=positive edge clock, 1=negative edge clock for clocking in the TE to the formatter* *Bit 2: Resync internal TE to external TE and reset status register* *Bits 2-7: Not used.*	

4.7.4.4 SET_FR_PHASE_SEQ_A/B

Relative CAN Address	*0x0 90 07*	*1A*	*CH1 – Bit D*
Relative CAN Address	*0x0 90 08*	*1B*	
Description	*These two commands define the phase switching sequence (or Walsh function). The sequence is a 128-bit stream and these commands send 64-bit streams corresponding to the top and bottom of the 128 bit value. See section 4.2.3 for a description of how this sequence affects the D (sign) bits.* *The MSB of the first byte sent in stream A will correspond to the start of the switching sequence and the LSB of the last byte sent in stream B will correspond to the end of the sequence.* *To disable all switching a sequence containing only zero should be loaded.*		
Suggested Interval	*Initialization*		
TE Related	*Yes*		
Data	*8 bytes: 128-bit sequence that should correspond to a unique Walsh function* *Byte 1: Most significant byte* *Byte 8: Least significant byte*		
	A		*The first half of any valid Walsh function in the range: 0x0000000000000000 to 0xFFFFFFFFFFFFFFFF*
	B		*The second half of any valid Walsh function in the range: 0x0000000000000000 to 0xFFFFFFFFFFFFFFFF*

4.7.4.5 SET_FR_PHASE_OFFSET

Relative CAN Address	*0x0 90 09*	*CH1 – Bit D*
Description	*This command will set the amount by which the phase switching is delayed. On the TE following the update of this register, the new delay with take effect, delaying the first phase switching value from a minimum of 8 ms and a maximum of 16.388608 ms with 8 ns of resolution.* *Delay from 48 ms = 8 ms + SET_PHASE_OFFSET * 8 ns*	
Suggested Interval	*0.048 secs.*	
TE Related	*Yes*	

Data	*3 bytes:* *Byte 1: Most Significant byte* *Byte 3: Least Significant byte* *bits 23 - 20: Unused and always set to zero* *bits 19 – 0: Delay Value.*

4.7.4.6 SET_FR_48 V

Relative CAN Address	*0x0 C0 03*
Description	*Turn main power to formatter and digitizer off or on*
Suggested Interval	*Debug*
TE Related	*No*
Data	*1 byte:* *bit 0: 1= 48V ON, 0 = 48V OFF* *bits 1 – 7: Reserved*

After issuing the command to turn on main power, no communication should be attempted for at least 5 seconds – allowing the hardware to power up.

4.7.4.7 FR_RELOAD_FPGA

Relative CAN Address	*0x0 C0 04*
Description	*Hardware reset to formatter, causes reload of FPGAs*
Suggested Interval	*Debug*
TE Related	*No*
Data	*1 byte:* *bits 0 – 7: Reserved*

4.7.4.8 FR_CAPTURE_PAYLOAD

Relative CAN Address	*0x0 C0 0F*	*all channels*
Description	*Start capturing payload data into internal FIFO*	
Suggested Interval	*Debug*	
TE Related	*No*	
Data	*1 byte:* *Bit 0:* *0 = data directly from input pins is fed into fifo* *1 = formatted data going to the output is fed into the fifo* *Bits 1 – 7: Reserved*	

This command is used for testing. In order to see internally generated test data (see SET_FR_CW_Chi) in this fifo, the scramble code must be turned off and bit 0 must be equal to 1.

4.7.4.9 SET_FR_RNG

Relative CAN Address	*0x0 90 0A*	*CH1 – bit D*
Relative CAN Address	*0x0 A0 0A*	*CH2 – bit C*
Relative CAN Address	*0x0 B0 0A*	*CH3 – bit B*
Relative CAN Address	*0x0 C0 0A*	*All channels – All bits*
Description	*Set RNG mode register.*	
Suggested Interval	*Debug*	
TE Related	*No*	
Data	*1 byte:* *Bit 1-0: RNG mode* *00 = normal mode, RNG OFF* *01 = test mode, RNG ON* *10 =RNG OFF, counting test pattern sent* *11 =RNG ON, re-start (re-seed) every Timing Event* *Bits 2-7: not used*	

4.7.4.10 SET_FR_INPUT_TEST

Relative CAN Address	*0x0 90 0B*	*CH1 – bit D*
Relative CAN Address	*0x0 A0 0B*	*CH2 – bit C*
Relative CAN Address	*0x0 B0 0B*	*CH3 – bit B*
Relative CAN Address	*0x0 C0 0B*	*All channels – All bits*
Description	*Set input test mode register.*	
Suggested Interval	*Debug*	
TE Related	*No*	
Data	*1 byte:* *Bits 3-0: input clock edge selection* *1 = clock BB0 at falling edge, BB1 at rising edge* *2 = clock BB0 at rising edge, BB1 at falling edge* *3 = clock both BBs at falling edge* *All other combinations = normal mode, both BBs clocked in on the rising edge of the 250 MHz clock* *Bits 4-7: test mode* *1 = all input data equals 0 for 500 µs after TE* *2 = sample 0 of BB0 and BB1 equals 0 for 1 frame after TE* *3 = all input data equals 1* *4 = all input data equals 0* *All other combinations = normal mode*	

4.7.4.11 FR_EEPROM_PROG

Relative CAN Address	*0x0 A0 0C*	*CH2 – Bit C*
Description	*Program one byte into the EEPROM at the given address. Valid user addresses for EEPROM are 0 to 0x2FFF*	
Suggested Interval	*Reconfigure*	
TE Related	*No*	
Data	*5 bytes:* *Byte 1: Most significant address byte (0 - 0x2F)* *Byte 2: Least significant address byte (0 – 0xFF)* *Byte 3: Data byte to be programmed into EEPROM* *Byte 4: 00 only* *Byte 5: 00 only* *If the last two bytes are not set to 00, the EEPROM will not program. Status should be checked before programming a byte to ensure the EEPROM is ready. The EEPROM may take as long as 10 ms to program a byte. Wait 10 ms between each program command.*	

4.7.4.12 FR_EEPROM_FETCH

Relative CAN Address	*0x0 A0 0D*	*CH2 – Bit C*
Description	*Read one byte from EEPROM at the given address. Valid user addresses for EEPROM are 0 to 0x3FFF. The byte read is available using the GET_FR_EEPROM_DATA command.*	
Suggested Interval	*Reconfigure*	
TE Related	*No*	
Data	*2 bytes:* *Byte 1: Most significant address byte (0 - 0x3F)* *Byte 2: Least significant address byte (0 – 0xFF)* *A read of the EEPROM takes approximately 80 µs. Data will not be available to the GET_FR_EEPROM_DATA command until after this time interval.*	

4.7.5 TTX Monitor Points in Detail

4.7.5.1 GET_TTX_ALARM_STATUS

Relative CAN Address	*0x0 24 01*
Description	*Transmit Alarm bit fields for all three TTX units. These alarms are latching. Alarm bits in this register remain in the alarmed state until a read of this register occurs after the alarm condition has been cleared for greater than or equal to the deactivation time, therefore the alarm may have to be cleared more than once for a single alarm.*
Suggested Interval	*0.048 seconds*
Update Rate	*0.048 seconds*
TE Related	*No*
Data	*6 Bytes (48 bits): 3 channels with 2 bytes each*

Data	*Bit*	*Name*	*Condition*
Byte 0: TTX1 – D (bits 0-7) Byte 2: TTX2 – C (bits 16-23) Byte 4: TTX3 – B (bits 32-39)	*0*	*EOLALM (Laser end of life alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	1	*ModTEMPALM (Modulator Temperature Alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	2-4	*Reserved*	*All bits 1*
	5	*LsWAVALM (Laser Wavelength Alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	6-7	*Reserved*	*All bits 1*
Byte 1: TTX1 – D (bits 8-15) Byte 3: TTX2 – C (bits 24-31) Byte 5: TTX3 – B (bits 40-47)	*0*	*TxALM INT (Tx Summary alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	1	*LsBIASALM (Laser bias current alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	2	*LsTEMPALM (Laser temperature alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	3	*TxLOCKERR (Loss of TxPLL lock indicator)*	*0 for alarm since last read, 1 for no alarm since last read*
	4	*Reserved*	*Mirror bit 7*
	5	*LsPOWALM (Laser power alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	6	*ModBIASALM (Modulator bias alarm)*	*0 for alarm since last read, 1 for no alarm since last read*
	7	*TxFIFOERR (Mux FIFO error indicator)*	*0 for alarm since last read, 1 for no alarm since last read*

	Byte 1								Byte 0							
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Func CH1	TxF	MdB	LsP	R	TxL	LsT	LsB	TxA	T	R	LsW	R	R	R	MdT	EOL
	Byte 3								**Byte 2**							
Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Func CH2	TxF	MdB	LsP	R	TxL	LsT	LsB	TxA	T	R	LsW	R	R	R	MdT	EOL
	Byte 5								**Byte 4**							
Bit	47	46	45	44	42	42	41	40	39	38	37	36	35	34	33	32
Func CH3	TxF	MdB	LsP	R	TxL	LsT	LsB	TxA	T	R	LsW	R	R	R	MdT	EOL

Internal to the transponder, these alarms are latching. Alarm bits in this register remain in the alarmed state until a read of the internal register occurs after the alarm condition has been cleared for greater than or equal to the deactivation time. (The read of the internal register is automatically done by the FPGA, but the alarm can only be cleared from the FPGA register by a TTX_CLR_ALARMS command.) Note that on power-up, many alarms may be set until the module completes initialization. Therefore, the host should read this register as part of module initialization to clear those alarms. Alarms may be triggered when lasers are turned off; therefore, the alarm should be cleared after turning on the lasers and then checked.

4.7.5.2 GET_TTX_LASER_BIAS_CHi

Relative CAN Address	*0x0 21 01*	*CH1 – Bit D*
Relative CAN Address	*0x0 22 01*	*CH2 – Bit C*
Relative CAN Address	*0x0 23 01*	*CH3 – Bit B*
Description	*The magnitude of the laser bias power setting current.*	
Suggested Interval	*As required*	
Update Rate	*0.048 seconds*	
TE Related	*No*	
Data	*3 Bytes: MSByte first, 24 bits twos complement*	
Conversion Factors	*1 μA / count*	
Operating Range	*Current value < 150% of initial value (initial value + 50%)**	

*Related alarm value LsBIASALM from READ_ALARM_STATUS will trigger at this value.
Normal operating range is 70 mA. Initial value must be read at startup.

4.7.5.3 GET_TTX_LASER_PWR_CHi

Relative CAN Address	*0x0 21 02*	*CH1 – Bit D*
Relative CAN Address	*0x0 22 02*	*CH2 – Bit C*
Relative CAN Address	*0x0 23 02*	*CH3 – Bit B*
Description	*Laser output power. This is a software equivalent to the LsPOWMON pin in the FR.*	
Suggested Interval	*As required*	
Update Rate	*0.048 seconds*	
TE Related	*No*	
Data	*3 Bytes: MSByte first, 24 bits twos complement*	
Conversion Factors	*1 μW / count*	
Operating Range	*Current value > 50 % of initial value **	

*Related alarm value LsPOWALM from READ_ALARM_STATUS will trigger at this value. Initial value must be read after starting the lasers.

4.7.5.4 GET_TTX_LASER_TMP_CHi

Relative CAN Address	*0x0 21 03*	*CH1 – Bit D*
Relative CAN Address	*0x0 22 03*	*CH2 – Bit C*
Relative CAN Address	*0x0 23 03*	*CH3 – Bit B*
Description	*Temperature of the laser. This is an alternate of the READ_TEMP command. The value returned is a relative error to the factory determined correct laser operating temperature.*	
Suggested Interval	*As required*	
Update Rate	*0.048 seconds*	
TE Related	*No*	
Data	*3 Bytes: MSByte first, 24 bits twos complement*	
Conversion Factors	*1 m℃ / count*	
Operating Range	*±1℃ from initial value – this will be determined at startup.*	

4.7.5.5 GET_TTX_LASER_ENABLED

Relative CAN Address	*0x0 24 05*
Description	*Read the status of the half-transponder lasers*
Suggested Interval	*As required*
Update Rate	*When written by user command TTX_LS_ENA*
TE Related	*No*
Data	*1 byte* *Bit 0: 1 = TTX1 laser ON* *Bit 1: 1 = TTX2 laser ON* *Bit 2: 1 = TTX3 laser ON* *Bits 3-7: reserved*

4.7.5.6 GET_TTX_I2C_DATA

Relative CAN Address	*0x0 24 0E*	*Any channel/bit*
Description	*Read the response from a SET_I2C_CMD control point. See Section 4.3.3.2*	
Suggested Interval	*Debug*	
TE Related	*No*	
Data	*8 Bytes:* *Byte 0: Status** *Byte 1: Length* *Bytes 2-7: dependant on SET_I2C_CMD sent*	
Conversion Factors	*See[RD 01].*	

*Status conversion uses some of the MSA 300 Status Bit definition as defined in [RD 01] and some custom defined bits:

Bits (3-0) = same as MSA 300 definition
Bit 4 = CPN (Bit 7 of the MSA 300)
Bit 5 = No response received from transponder
Bit 6 = Checksum error from transponder
Bit 7 = I2C bus is busy

Bits 5-7 are for engineering verification only. They are documented here as a reference and are not for user applications.

4.7.6 TTX Control Points

4.7.6.1 TTX_RESET

Relative CAN Address	*0x0 A4 00*
Description	*Reset all three half-transponders*
Suggested Interval	*As required*
TE Related	*No*
Data	*1 byte* *Bits 0-7: reserved*

4.7.6.2 TTX_CLR_ALARMS

Relative CAN Address	*0x0 A4 01*
Description	*Clear all TTX alarm bits – these will be activated again if the alarm condition still exists. Invalid alarms may occur when the transponder laser is turned off.*
Suggested Interval	*As required*
TE Related	*No*
Data	*1 byte* *Bits 0-7: reserved*

4.7.6.3 TTX_RESET_FIFO

Relative CAN Address	*0x0 A4 04*
Description	*Command the selected half-transponder to reset the internal FIFO.*
Suggested Interval	*As required*
TE Related	*No*
Data	*1 byte* *Bit 0: 1 = Reset TTX1 internal FIFO* *Bit 1: 1 = Reset TTX2 internal FIFO* *Bit 2: 1 = Reset TTX3 internal FIFO* *Bits 3-7: reserved*

4.7.6.4 TTX_LASER_ENABLE

Relative CAN Address	*0x0 A4 05*
Description	*Enable or disable the selected half-transponder lasers*
Suggested Interval	*As required*
TE Related	*No*
Data	*1 byte* *Bit 0: 1/Enable, 0/Disable TTX1 (Bit D) laser* *Bit 1: 1/Enable, 0/Disable TTX2(Bit C) laser* *Bit 2: 1/Enable, 0/Disable TTX3(Bit B) laser* *Bits 3-7: reserved*

4.7.6.5 TTX_I2C_CMD

Relative CAN Address	*0x0 A1 0E*	*CH1 – bit D*
Relative CAN Address	*0x0 A2 0E*	*CH2 – bit C*
Relative CAN Address	*0x0 A3 0E*	*CH3 – bit B*
Description	*Send an I2C command. See Section 4.3.3.1*	
Suggested Interval	*As Required*	
TE Related	*No*	
Data	*8 Bytes:* *Byte 0: Command* *Byte 1: Length* *Bytes 2-7: dependent on SET_I2C_CMD sent*	

Send commands must address a particular TTX. Only one TTX at a time can be communicated with. All communication goes through FPGA 2. Once a response has been received from a TTX it is stored in FPGA2 and can be retrieved using the GET_TTX_I2C_DATA command.

5 Safety Interface

Provisions are made to shut down all the lasers on all the TTX's on the antenna in the event that the signal on the output of the optical multiplexers exceeds the eye-safe level. The output of the multiplexer is continually monitored. As long as it is below the safe level, a signal (the 'keep-alive') is sent to each DTX. If this signal is lost, the lasers are stopped immediately. An FPGA also monitors the keep-alive and holds the shut-down even if the level becomes safe again.

Many of the TTX control points are either safety limits or operating points. Changing these indiscriminately could upset normal operation or precipitate damage to the hardware. For this reason, they should only be changed by, or with the express authority of, a qualified engineer.