Main Loop 🡨🡪 Panel Control

Interface Messages

**To**: [All Software, Firmware, Hardware Teams]

**From**: [Farid, Giovanni, Chris, Anand, Riddhi]

**Date**: [Date sent]

**Subject**: [Subject of the memo]

Summary

Body

Conclusion

CC: [Send copies to anyone affected by the memo.]

Attachments: [List any attachments to the memo. Only list referred to in the body of the memo.]

Table of Contents

[2 Message type definitions 4](#_Toc442797797)

[3 Panel Functions 6](#_Toc442797798)

[3.1 PANEL\_INIT 6](#_Toc442797799)

[3.1.1 Request (Main Loop 🡪 Panel Control) 6](#_Toc442797800)

[3.1.2 Response (Panel Control 🡪 Main Loop) 7](#_Toc442797801)

[3.2 PANEL\_SET\_TARGET\_DEVICES 9](#_Toc442797802)

[3.2.1 Request (Main Loop 🡪 Panel Control) 9](#_Toc442797803)

[3.2.2 Response (Panel Control 🡪 Main Loop) 10](#_Toc442797804)

[3.3 PANEL\_GET\_TARGET\_DEVICES 11](#_Toc442797805)

[3.3.1 Request (Main Loop 🡪 Panel Control) 11](#_Toc442797806)

[3.3.2 Response (Panel Control 🡪 Main Loop) 12](#_Toc442797807)

[3.4 PANEL\_POLL\_DFE 13](#_Toc442797808)

[3.4.1 Request (Main Loop 🡪 Panel Control) 13](#_Toc442797809)

[3.4.2 Response (Panel Control 🡪 Main Loop) 14](#_Toc442797810)

[3.5 PANEL\_POLL\_TILE 15](#_Toc442797811)

[3.5.1 Request (Main Loop 🡪 Panel Control) 15](#_Toc442797812)

[3.5.2 Response (Panel Control 🡪 Main Loop) 16](#_Toc442797813)

[3.6 PANEL\_BEAM\_LOAD\_TABKE 17](#_Toc442797814)

[3.6.1 Request (Main Loop 🡪 Panel Control) 17](#_Toc442797815)

[3.6.2 Response (Panel Control 🡪 Main Loop) 18](#_Toc442797816)

[3.7 PANEL\_BEAM\_LOAD\_LINE 19](#_Toc442797817)

[3.7.1 Request (Main Loop 🡪 Panel Control) 19](#_Toc442797818)

[3.7.2 Response (Panel Control 🡪 Main Loop) 21](#_Toc442797819)

[3.8 PANEL\_BEAM\_ROTATION 22](#_Toc442797820)

[3.8.1 Request (Main Loop 🡪 Panel Control) 22](#_Toc442797821)

[3.8.2 Response (Panel Control 🡪 Main Loop) 25](#_Toc442797822)

[3.9 PANEL\_OP\_GET\_FREQ 27](#_Toc442797823)

[3.9.1 Request 27](#_Toc442797824)

[3.9.2 Response 28](#_Toc442797825)

[3.10 PANEL\_OP\_SET\_FREQ 29](#_Toc442797826)

[3.10.1 Request 29](#_Toc442797827)

[3.10.2 Response 31](#_Toc442797828)

[3.11 PANEL\_OP\_CHANNEL\_ENABLE 33](#_Toc442797829)

[3.11.1 Request 33](#_Toc442797830)

[3.11.2 Response 34](#_Toc442797831)

[3.12 PANEL\_OP\_SET\_TX\_PWR 35](#_Toc442797832)

[3.12.1 Request 35](#_Toc442797833)

[3.12.2 Response 36](#_Toc442797834)

[3.13 PANEL\_OP\_CFR 37](#_Toc442797835)

[3.13.1 Request 37](#_Toc442797836)

[3.13.2 Response 38](#_Toc442797837)

[3.14 PANEL\_OP\_SET\_RX\_GAIN 39](#_Toc442797838)

[3.14.1 Request 39](#_Toc442797839)

[3.14.2 Response 40](#_Toc442797840)

[3.15 PANEL\_OP\_MAP\_PORT2BEAM 41](#_Toc442797841)

[3.15.1 Request 41](#_Toc442797842)

[3.15.2 Response 45](#_Toc442797843)

[3.16 PANEL\_CAL\_IQ 46](#_Toc442797844)

[3.16.1 Request 46](#_Toc442797845)

[3.16.2 Response 47](#_Toc442797846)

[3.17 PANEL\_CAL\_IF 48](#_Toc442797847)

[3.17.1 Request 48](#_Toc442797848)

[3.17.2 Response 50](#_Toc442797849)

[3.18 PANEL\_CAL\_RF 52](#_Toc442797850)

[3.18.1 Request 52](#_Toc442797851)

[3.18.2 Response 53](#_Toc442797852)

[3.19 PANEL\_CAL\_RF 54](#_Toc442797853)

[3.19.1 Request (Main Loop 🡪 Panel Control) 54](#_Toc442797854)

[3.19.2 Response (Panel Control 🡪 Main Loop) 56](#_Toc442797855)

# Message type definitions

/\*

\*\*

\*\* Message types on Main Loop --> Panel Control Interface

\*\*

\*/

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_INIT\_REQ 0x00

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_POLL\_DFE\_REQ 0x08

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_POLL\_TILE\_REQ 0x09

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_LOAD\_BEAM\_TABLES\_REQ 0x10

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_LOAD\_BEAM\_LINE\_REQ 0x11

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_BEAM\_ROTATION\_REQ 0x12

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_OP\_GET\_FREQ\_REQ 0x20

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_OP\_SET\_FREQ\_REQ 0x21

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_OP\_CH\_ENABLE\_REQ 0x22

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_OP\_SET\_TX\_PWR\_REQ 0x23

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_OP\_CH\_CFR\_REQ 0x24

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_OP\_SET\_RX\_GAIN\_REQ 0x25

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_OP\_MAP\_LAYER\_DATA\_TO\_BEAM\_REQ 0x26

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_IQ\_CAL\_REQ 0x30

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_IF\_CAL\_REQ 0x31

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_BDS\_CAL\_REQ 0x32

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_RFCAL\_REQ 0x33

/\*

\*\*

\*\* Message types on Panel Control --> Main Loop Interface

\*\*

\*/

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_INIT\_RSP 0x80

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_POLL\_DFE\_RSP 0x88

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_POLL\_TILE\_RSP 0x89

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_LOAD\_BEAM\_TABLES\_RSP 0x90

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_LOAD\_BEAM\_LINE\_RSP 0x91

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_BEAM\_ROTATION\_RSP 0x92

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_OP\_GET\_FREQ\_RSP 0xA0

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_OP\_SET\_FREQ\_RSP 0xA1

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_OP\_CH\_ENABLE\_RSP 0xA2

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_OP\_SET\_TX\_PWR\_RSP 0xA3

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_OP\_CH\_CFR\_RSP 0xA4

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_OP\_SET\_RX\_GAIN\_RSP 0xA5

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_OP\_MAP\_LAYER\_DATA\_TO\_BEAM\_RSP 0xA6

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_IQ\_CAL\_RSP 0xB0

#define BDS\_MAIN\_LOOP\_TO\_PANEL\_CONTROL\_IF\_CAL\_RSP 0xB1

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_BDS\_CAL\_RSP 0xB2

#define BDS\_PANEL\_CONTROL\_TO\_MAIN\_LOOP\_RFCAL\_RSP 0xB3

R&D NAMING:

'PANEL\_INIT', 0

'PANEL\_SET\_TARGET\_DEVICES', 4

'PANEL\_GET\_TARGET\_DEVICES', 5

% INIT RESERVED 1-7

'PANEL\_POLL\_DFE', 8

'PANEL\_POLL\_TILE', 9

% POLL RESERVED 10-15

'PANEL\_BEAM\_LOAD\_TABLE', 16

'PANEL\_BEAM\_LOAD\_LINE', 17

'PANEL\_BEAM\_ROTATION', 18

% BEAM FORMING RESERVED 19-31

'PANEL\_OP\_GET\_FREQ', 32

'PANEL\_OP\_SET\_FREQ', 33

'PANEL\_OP\_CH\_ENABLE', 34

'PANEL\_OP\_SET\_TX\_PWR', 35

'PANEL\_OP\_CFR', 36

'PANEL\_OP\_SET\_RX\_GAIN', 37

'PANEL\_OP\_MAP\_PORT2BEAM', 38

% OPERATION RESERVED 39-47

'PANEL\_CAL\_IQ', 48

'PANEL\_CAL\_IF', 49

'PANEL\_CAL\_BDS', 50

'PANEL\_CAL\_RF', 51

# Panel Functions

## PANEL\_INIT

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* Call this function during Power UP
* Call selectively when there are issues (mostly in the failure path)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Init Request |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 3 | 1 | DFE Section:  Reserved (4 bits)  CH4 (1 bit)  CH3 (1 bit)  CH2 (1 bit)  CH1 (1 bit) | For these bits:  Set to 0: Don’t Init  Set to 1: Do Init | Digital and analog Clocking sections would get initialized regardless of request |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Variable

Triggers:

* This is sent in response to an Init Request

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | | Default Value | |
| 0 | 1 | Message Type | Type of Message being sent | | Init Response | |
| 1 | 2 | Message Length | Length of Message | | Fill with Number of Bytes | |
| 3 | 1 | DFESuccess (1 bit: LSB): 0: fail 1:Pass |  | |  | |
| 4 | 4 | Board ID | 32 bits value | | Board Identification Number | |
| 8 | 4 | FPGA Version | 32 bits value | | RTL Firmware Version | |
| 12 | 2 | CPRI1:   * Status Bits 15-12 * Speed Bits 11-5 * Reserved 0 Bits 4-3 * SFP Present Bit 2 * Laser OK Bit 1 * CPRI OK Bit 0 |  | | CPRI1 | |
| 14 | 2 | CPRI2:   * Status Bits 15-12 * Speed Bits 11-5 * Reserved 0 Bits 4-3 * SFP Present Bit 2 * Laser OK Bit 1 * CPRI OK Bit 0 |  | | CPRI2 | |
| 16 | 2 | PLLs:   * SYNTHBDS\_Lock Bit 15 * SYNTHMOD\_Lock Bit 14 * SYNTHCAL\_Lock Bit 13 * Reserved 0 Bits 12-10 * AD9523\_PLL2Lock Bit 9 * AD9523\_PLL2FDBK Bit 8 * AD9523\_PLL2REF Bit 7 * AD9523\_PLL1Lock Bit 6 * AD9523\_PLL1REF Bit 5 * AD9523\_OK Bits 4 * Reserved 0 Bit 3-1 * SI5323\_OK Bit 0 |  | | PLLs status | |
| 18 | 1 | Number of RFICs |  | | Valid range is from: 1..96 | |
| Following field (TileResult) is repeated for the number of RFICs | | | | | | |
| 19 | 1 | TileResult (8 bits) x RFIC  Bits 0-1: LSB  RESERVED  Comm: 2 bits  Bits 3-2 (In Binary):  00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL | | For SuccessOp: ‘0’ good ‘1’ fail | |  |

## PANEL\_SET\_TARGET\_DEVICES

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* Call this function to set the target devices

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | | Field | Descriptions | Default Value | |
| 0 | 1 | | Message Type | Type of Message being sent | Init Request | |
| 1 | 2 | | Message Length | Length of Message | Fill with Number of Bytes | |
| 4 | | 4 | RFIC Number Panel 1: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |
| 8 | | 4 | RFIC Number Panel 2: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |
| 12 | | 4 | RFIC Number Panel 3: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |
| 16 | | 4 | RFIC Number Panel 4: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* This is sent in response to the corresponding Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Init Response |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 3 | 1 | Success (1 bit: LSB): 0: fail 1:Pass |  |  |

## PANEL\_GET\_TARGET\_DEVICES

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* Call this function to get the target devices

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Init Request |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* This is sent in response to the corresponding Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Init Response |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 4 | 4 | RFIC Number Panel 1: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |
| 8 | 4 | RFIC Number Panel 2: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |
| 12 | 4 | RFIC Number Panel 3: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |
| 16 | 4 | RFIC Number Panel 4: 24 bits | RFIC ID | Needs 24 bits (out of the 32 bits) to uniquely identify an RFIC within a Panel. |

## PANEL\_POLL\_DFE

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* **Could be called after Init Req/Rsp when the Panel is coming up**
* **Any failure scenarios**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Poll DFE Request |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* This is sent in response to a Poll DFE Request

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | | Descriptions | | Default Value |
| 0 | 1 | Message Type | | Type of Message being sent | | Poll DFE Response |
| 1 | 2 | Message Length | | Length of Message | | Fill with Number of Bytes |
| 3 | 1 | DFESuccess (1 bit: LSB): 0: fail 1:Pass |  | |  | |
| 4 | 4 | Board ID | |  | |  |
| 8 | 4 | FPGA Ver | |  | |  |
| 12 | 1 | Bit 5: AD9523-1 OK 1: OK, 0: NOK  Bit 3: SI5324 OK 1: OK, 0: NOK  Bit 0: Voltage OK 1: OK, 0: NOK | |  | |  |
| 13 | 1 | Bit6: CPRI2 OK 1: OK, 0: NOK  Bit5: SFP2 Laser OK 1: OK, 0: NOK  Bit4: SFP2 Present 1: OK, 0: NOK  Bit2: CPRI1 OK 1: OK, 0: NOK  Bit1: SFP1 Laser OK 1: OK, 0: NOK  Bit0: SFP1 Present 1: OK, 0: NOK | |  | |  |
| 14 | 4 | TX1 Temperture | | Deg C \* 1000 | |  |
| 18 | 4 | TX2 Temperture | | Deg C \* 1000 | |  |
| 22 | 4 | TX3 Temperture | | Deg C \* 1000 | |  |
| 26 | 4 | TX4 Temperture | | Deg C \* 1000 | |  |
| 30 | 4 | BDS Temperture | | Deg C \* 1000 | |  |
| 34 | 4 | FPGA Temperture | | Deg C \* 1000 | |  |
| 38 | 4 | 28V\_IN\_1 | | In mV | |  |
| 42 | 4 | 28V\_IN\_2 | | In mV | |  |
| 46 | 4 | 12V\_DIG | | In mV | |  |
| 50 | 4 | DFE Current | | In mA | |  |
| 54 | 2 | Panel1 Temperature (DSTR Board) | | Deg C \* 100 | |  |
| 56 | 2 | Panel2 Temperature (DSTR Board) | | Deg C \* 100 | |  |

## PANEL\_POLL\_TILE

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* **Could be called after Init Req/Rsp when the Panel is coming up**
* **Could be called when the Panel is Operational to obtain the Temperature of the Tile (for Auto Calibration)**
* **Any failure scenarios**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Poll Tile Request |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Variable

Triggers:

* This is sent in response to a Poll Tile Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Init Response |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 3 | 1 | Number of RFICs |  | Valid Range: 1..96 |
| Following field is repeated for each RFIC | | | | |
| <4..Variable Length> | 1 | TileResult (8 bits):  Bits 0-1: RESERVED  Comm: 2 bits  Bits 3-2: 00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits (Note: These 4 bits are always set to 0 – meaning they are always OK for this Response, essentially Don’t Care)  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL |  |  |
| <4..Variable Length> | 2 | TilePollResults (2 bytes):  Temperature Value (2bytes): |  |  |

## PANEL\_BEAM\_LOAD\_TABKE

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* Called after Init Req/Rsp is done to fill the tables in the STM
* Could be called at a later time when the Panel is up and Operational to add or update the table in STM
* This could be done in background. It loads it to the STM but doesn’t send it to RFIC. There is a separate command to activate the beams.
* Would be good to load it through UART, for now we can just use CAN bus since its only 23 bytes.
* We would need to call this fun

For the product version - Main Loop reads the text file (which got exported from Matlab) and sends either a select few lines or all of it to Panel control in terms of a Look Up Table. 127 (127 for Phase and 127 for Magnitude) is the Maximum number of lines that each STM can hold.

For field trial version – MATLAB/BDS Controller will send this payload within message.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Load Beam Tables Request |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 3 | 1 | Panel ID | Give the number of panels that are available | Values include 1, 2 or 4 |
| 4 | 1 | RFIC ID |  | Valid range is from: 1..24 |
| 5 | 2x127 | LUTValue:  Phase : 9 bits  Magnitude: 7 bits |  | LSB is 0.2 db  For Mag: 0 is the Max  And 127 is the Min.  For Phase: 0 is 0 degree, LSB is 360/512 degree. |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* Sent in response to the Load Beam Tables Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Load Beam Tables Response |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 3 | 1 | Number of RFICs |  | Valid range is from: 1..96 |
| 4 | 1 | TileResult (8 bits  Bits 0-1: LSB  RESERVED  Comm: 2 bits  Bits 3-2 (In Binary):  00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL | For SuccessOp: ‘0’ good ‘1’ fail |  |

## PANEL\_BEAM\_LOAD\_LINE

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Variable

Triggers:

* Called after Init Req/Rsp is done to fill the tables in the STM
* Could be called at a later time when the Panel is up and Operational to add or update the table in STM
* This could be done in background. It loads it to the STM but doesn’t send it to RFIC. There is a separate command to activate the beams.
* Would be good to load it through UART, for now we can just use CAN bus since its only 23 bytes.
* We would need to call this fun

For the product version - Main Loop reads the text file (which got exported from Matlab) and sends either a select few lines or all of it to Panel control in terms of a Look Up Table. 127 (127 for Phase and 127 for Magnitude) is the Maximum number of lines that each STM can hold.

For field trial version – MATLAB/BDS Controller will send this payload within message.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Load Beam Tables Request |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| ~~3~~ | ~~1~~ | ~~Number of Panels~~ | ~~Give the number of panels that are available~~ | ~~Values include 1, 2 or 4~~ |
| 4 | 1 | Mode: 2 bits  Reserved: 6 bits | Xxxx xx01 : Magnitude  Xxxx xx10 : Phase  Xxxx xx11: Mag & Phase |  |
| 5 | 1 | LUT Position: 7 bits  Reserved: 1 bit | Unsigned Number |  |
| 6 | 1 | Number of RFICs |  | Valid range is from: 1..96 |
| Following field (LUTValue) is repeated per RFIC | | | | |
| <Variable Length> | 2 | LUTValue:  Phase : 9 bits  Magnitude: 7 bits |  | LSB is 0.2 db  For Mag: 0 is the Max  And 127 is the Min.  For Phase: 0 is 0 degree, LSB is 360/512 degree. |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Variable

Triggers:

* Sent in response to the Load Beam Tables Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Load Beam Tables Response |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 3 | 1 | Number of RFICs |  | Valid range is from: 1..96 |
| TileResult is repeated for number of RFICs | | | | |
| <Variable Length> | 1 | TileResult (8 bits) x RFIC  Bits 0-1: LSB  RESERVED  Comm: 2 bits  Bits 3-2 (In Binary):  00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL | For SuccessOp: ‘0’ good ‘1’ fail |  |

## PANEL\_BEAM\_ROTATION

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Variable

Triggers:

Main Loop will need to know a default configuration for the Beam positions. Any new changes coming in from the CLI or the CPRI interface can override this configuration.

Each index into the Phase and Magnitude table will point to a specific beam location for a specific Channel.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value | |
| 0 | 1 | Message Type | Type of Message being sent | Beam Rotation Request | |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes | |
| 3 | 2 | Channel Number: 14 bits | 4 Values for Tx1..4  4 values for Rx1..4  4 values for Txx3, Txx4, Rxx3, Txx4 (These 4 values are used for Cross Poles) | Following will be the values used:  This is in binary:  Tx1: 0000 0000 0000 0001  Tx2: 0000 0000 0000 0010  Tx3: 0000 0000 0000 0100  Tx4: 0000 0000 0000 1000  Rd: 0000 0000 0001 0000  Rd: 0000 0000 0010 0000  Txx3: 0000 0000 0100 0000  Txx4: 0000 0000 1000 0000  Rx1: 0000 0001 0000 0000  Rx2: 0000 0010 0000 0000  Rx3: 0000 0100 0000 0000  Rx4: 0000 1000 0000 0000  Rxx3: 0001 0000 0000 0000  Rxx4: 0010 0000 0000 0000  Setting all bits to 1 will calibrate everything. We could also bitwise OR multiple channel numbers |
| 5 | 1 | Number of Channels |  | 1..12 |
| Following fields (Position) is repeated per Channel |
| <Variable Length> | 2 | Position: 7 bits x Mag/Phase x Channel  7 bits for Magnitude  1 bit to Select Magnitude  7 bits for Phase  1 bit to Select Phase |  |  |

**Example:**

RotateBeam(‘TX1’, ‘Phase’,30)

RotateBeam(‘RX3’, ‘Magnitude’,4)

RotateBeam({‘TX2’ ‘RX3’}, {‘Magnitude’, 5},{‘Phase’,10} // For Tx2 we want to change Mag to 5 and for RX3 we want to change Phase to 10. If we don’t change, then the previous values get retained.

Main Loop maintains a table of Phase, Magnitude per Channel and will need to update this whenever a rotate beam message is called.

When information for Txx3 (or Rxx3) gets changed, it changes the information for Tx3 (or (rx3) automatically. Main Loop needs to keep track of this. Similarly, for Txx4 (or Rxx4). The inverse is also true, if Tx3 is changed then txx3 automatically gets changed.

RotateBeam({‘TX2’ ‘RX3’}, {‘Magnitude’, ‘Phase’},{5,20},{‘Phase’,10} // In this example, for Tx2 Both Mag and Phase are getting changed. But for RX3 only Phase is getting changed.

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Variable

Triggers:

* Sent in response to the Beam Rotation Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Beam Rotation Response |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
| 3 | 1 | Number of RFICs |  | Valid range is from: 1..96 |
| Following field (TileResult) gets repeated for each RFIC | | | | |
| <4..Variable Length> | 1 | TileResult (8 bits) x RFIC  Bits 0-1: LSB  RESERVED  Comm: 2 bits  Bits 3-2 (In Binary):  00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL | For SuccessOp: ‘0’ good ‘1’ fail |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Phase | | | | | | | |
| POS | RFIC1 | RFIC2 | RFIC3 | … | RFIC24 | … | RFIC96 |
| 0 | 9bits | 9bits | 9bits | … | 9bits | … | 9bits |
| 1 | 9bits | 9bits | 9bits | … | 9bits | … | 9bits |
| 2 | 9bits | 9bits | 9bits | … | 9bits | … | 9bits |
| … | 9bits | 9bits | 9bits | … | 9bits | … | 9bits |
| 125 | 9bits | 9bits | 9bits | … | 9bits | … | 9bits |
| 126 | 9bits | 9bits | 9bits | … | 9bits | … | 9bits |
| 127 | 9bits | 9bits | 9bits | … | 9bits | … | 9bits |
|  |  |  |  |  |  |  |  |
|  | | | | | | | |
|  |  |  |  |  |  |  |  |
| Magnitude | | | | | | | |
|  | | | | | | | |
| POS | RFIC1 | RFIC2 | RFIC3 | … | RFIC24 | … | RFIC96 |
| 0 | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| 1 | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| 2 | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| … | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| … | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| … | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| 125 | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| 126 | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |
| 127 | 7bits | 7bits | 7bits | … | 7bits | … | 7bits |

## PANEL\_OP\_GET\_FREQ

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* This is called to obtain the BDS & IF Frequencies from Panel Control during Init and then these BDS/IF frequencies are used during IF Calibration. Use the TX\_IF Freq for TX\_IF Calibration, RX\_IF Freq for RX\_IF Calibration and BDS Freq for BDS Calibration.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Get Frequency Request |
| 1 | 2 | Data Length | Length of message being sent |  |
| ~~3~~ | ~~1~~ | ~~Band~~ | ~~AWS (Band 1 or Band 3): 0~~  ~~PCS: 1~~  ~~Reserved for future > 1~~ | ~~AWS~~ |
| 4 | 4 | TX\_RF\_Frequency | In Hz |  |

### Response

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* This message is sent in response to Get Frequency Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Get Frequency Response |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | 1 | (DFE)CalcSuccess | Success:1  Failure:0 |  |
| 4 | 4 | BDS\_Frequency | Frequency in Hz |  |
| 8 | 4 | Tx\_IF\_Frequency | Frequency in Hz |  |
| 12 | 4 | Rx\_IF\_Frequency | Frequency in Hz |  |
| 16 | 4 | RX\_RF\_Frequency | Frequency in Hz |  |

## PANEL\_OP\_SET\_FREQ

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* Prereq: Get\_Frequencies is called before calling this Set\_RF\_Frequencies function.
* This is also part of the Init Sequence, but it is called before RF Cal.
* This need not be called after an RF Cal has been done, since RFICs will stay locked after an RF Cal.
* When the Operating Frequency needs to be changed for the Panel, then this can be called. But we will need to first disable the channels before setting the freq and then enable the channels.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Set Frequency Request |
| 1 | 2 | Data Length | Length of message being sent |  |
| 19 | 4 | BDS\_Frequency | Frequency in Hz |  |
| 23 | 1 | Configuration  0: DIG\_CPRI  1: UpDnCnv |  |  |
| 24 | 4 | TX\_IF\_Frequency | Frequency in Hz |  |
| 28 | 4 | RX\_IF\_Frequency | Frequency in Hz |  |

### Response

Size of the Message (Fixed/Variable): Variable

**Triggers:**

* This message is sent in response to Set Frequency Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Set Frequency Response |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | 1 | DFESuccess | Success:1  Failure:0 | N/A |
| 4 | 4 | DFE Result:  ~~Reserved: Bit 0~~  ~~BDS Result: Bit 1~~  ~~IF Mod Result: Bit 2-5~~  ~~Reserved: Bit 6-7~~  Bit7-5: reserved  Bit4: MOD LOCK  Bit3-1: Reserved  Bit0: BDS LOCK | 1: Pass  0: Fail |  |
| 8 | 1 | Number of RFICs |  | Valid Range: 1..96 |
| Tile Success & PLL Lock Result is repeated for each RFIC | | | | |
| <9..Variable Length> | 1 | TileResult (8 bits):  Bits 0-1: RESERVED  Comm: 2 bits  Bits 3-2: 00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits (Note: These 4 bits are always set to 0 – meaning they are always OK for this Response, essentially Don’t Care)  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL |  |  |
| <9..Variable Length> | 2 | PLLLockResults: (same as RF Cal):  FreqCalDone 1 bit (1: Done, 0: Not done)  FreqCalVal 7 bits  DCOffsetDone 1 bit (1: Done, 0: Not done)  DCOffsetSign 1 bit (0: +ve, 1: -ve)  DCOffsetVal 7bits (absolute value) |  |  |

## PANEL\_OP\_CHANNEL\_ENABLE

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* Start or stop the Panel. During a start, this would be the last command called before the panel becomes operational.
* This message Enables/Disables the data path (Tx/Rx) on DFE for each Channel and sets the right switching mode, then Enables/Disables the (Tx/Rx) channel on the all the RFICs on the panel.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Channel Enable Request |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | 2 | Tx1: Bit 0  Tx2: Bit 1  Tx3: Bit 2  Tx4: Bit 3  Reserved: Bits 4-5  Txx3: Bit 6  Txx4: Bit 7  Rx1: Bit 8  Rx2: Bit 9  Rx3: Bit 10  Rx4: Bit 11  Reserved: Bits 12-13  Rxx3: Bit 14  Rxx4: Bit 15 | Note: At the DFE Tx3/4, Txx3/4 (similarly Rx3/4 and Rxx3/4) are treated the same, but it makes a difference for the RFICs.  Frequency in Hz | 0: Disable,  1: Enable |

### Response

Size of the Message (Fixed/Variable): Variable

**Triggers:**

* This message is sent in response to Channel Enable Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Channel Enable Response |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | 1 | DFE Success:  Tx1: Bit 0  Tx2: Bit 1  Tx3: Bit 2  Tx4: Bit 3  Rx1: Bit 4  Rx2: Bit 5  Rx3: Bit 6  Rx4: Bit 7 |  | 0: pass, 1: fail |
| 4 | 1 | Number of RFICs |  | Valid Range: 1..96 |
| TileResult is repeated for Number of RFICs | | | | |
| <5..Variable Length> | 1 | TileResult (8 bits):  Bits 0-1: RESERVED  Comm: 2 bits  Bits 3-2: 00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits (Note: These 4 bits are always set to 0 – meaning they are always OK for this Response, essentially Don’t Care)  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL |  |  |

## PANEL\_OP\_SET\_TX\_PWR

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers/Notes:**

* Best to do this before the Channel is enabled, not after it is enabled.
* Main Loop will need to figure out how to do the power division, power distribution among the various channels/beams, based on the total PA output. Work with Chris on this.
* Set/Change the power (gain) of the beam on Tx Channel based on the Power

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte  Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Set Tx Channel Power Request |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | ~~2~~1 | Channels:  Tx1: Bit 0  Tx2: Bit 1  Tx3: Bit 2  Tx4: Bit 3  Reserved: Bits 4-5  Txx3: Bit 6  Txx4: Bit 7 | Note: At the DFE Tx3/4, Txx3/4 (similarly Rx3/4 and Rxx3/4) are treated the same, but it makes a difference for the RFICs.  Frequency in Hz | 0: Not Selected,  1: Selected |
| 4 | 4 | Power Values:  Power\_Tx1: 1 byte  Power\_Tx2: 1 byte  Power\_Tx3/Txx3: 1 byte  Power\_Tx4/Txx4: 1 byte |  | 0: Nominal Max Pwr,  40: -20dB lower.  (0.5dB steps) |

**Notes:**

* The actual exposed dynamic range can be more for the Power Values, based on lab tests.
* Changing power on Tx3 would change the power on Txx3 AUTOMATICALLY and same for Tx4 and Txx4.
* When changing the power along with the number of beams, care should be taken so as to avoid overloading. Appropriate checks in Software should be present for this.
* To avoid any transient HW behavior, it might be better to turn off the channel (using Channel Enable Request) and then make the Power value changes. Then turn on the channel.

### Response

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* This message is sent in response to Set Tx Channel Power Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Set Tx Channel Power Response |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | 1 | Result:  Tx1: 1 Bit  Tx2: 1 Bit  Tx3/Txx3: 1 Bit  Tx4/Txx4: 1 Bit  Reserved: 4 Bits | Success:0  Failure:1 | N/A |

## PANEL\_OP\_CFR

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* This is the single beam CFR, multi-beam CFR is defined seperately
* Called when the CFR level needs to be changed – Systems team to provide inputs to this
* To disable the CFR on a single beam.
* Controls CFR block (Azcom individual channel)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Channel CFR |
| 1 | 2 | Length of data | Length of data being sent |  |
| 3 | 1 | Channels:  Tx1: Bit 0  Tx2: Bit 1  Tx3: Bit 2  Tx4: Bit 3  Reserved: Bits 4-5  Txx3: Bit 6  Txx4: Bit 7 |  |  |
| 4 | 4 | Thresholds (4 bytes):  Threshold\_Tx1 1 byte  Threshold\_Tx2 1 byte  Threshold\_Tx3/Txx3 1 byte  Threshold\_Tx4/Txx4 1 byte | (Value range is 1- 96) |  |

### Response

Size of the Message (Fixed/Variable): Fixed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Channel CFR Response |
| 1 | 2 | Length of data | Length of data being sent |  |
| 3 | 1 | Success (1 byte):  SuccessTx1 bit 0  SuccessTx2 bit 1  SuccessTx3 bit 2  SuccessTx4 bit 3  Reserved bit 4-7 | 0: Success, 1: Fail |  |

## PANEL\_OP\_SET\_RX\_PWR

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers/Notes:**

* Provides attenuation on the Rx side – the term Gain is used to keep it generic.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Set Rx Channel Gain Request |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | ~~2~~1 | Channels:  Rx1: Bit 0  Rx2: Bit 1  Rx3: Bit 2  Rx4: Bit 3  Reserved: Bits 4-5  Rxx3: Bit 6  Rxx4: Bit 7 | Note: At the DFE Rx3/4, Rxx3/4 are treated the same, but it makes a difference for the RFICs. | 0: Not Selected,  1: Selected |
| 4 | 4 | Rx Gain:  Gain\_Tx1: 1 byte  Gain\_Tx2: 1 byte  Gain\_Tx3/Txx3: 1 byte  Gain\_Tx4/Txx4: 1 byte |  | 0: Nominal Max Pwr,  40: -20dB lower.  (0.5dB steps) |

**Notes:**

* The actual exposed dynamic range can be more for the Power Values, based on lab tests.
* Changing power on Tx3 would change the power on Txx3 AUTOMATICALLY and same for Tx4 and Txx4.
* When changing the power along with the number of beams, care should be taken so as to avoid overloading. Appropriate checks in Software should be present for this.
* To avoid any transient HW behavior, it might be better to turn off the channel (using Channel Enable Request) and then make the Power value changes. Then turn on the channel.

### Response

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* This message is sent in response to Set Tx Channel Power Request

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Set Rx channel gain response |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | 1 | Result:  Rx1: 1 Bit  Rx2: 1 Bit  Rx3/Rxx3: 1 Bit  Rx4/Rxx4: 1 Bit  Reserved: 4 Bits | Success:0  Failure:1 | N/A |

## PANEL\_OP\_MAP\_PORT2BEAM

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* This is to map the CPRI channels to our data chain (Tx and Rx)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Map Data Layer To Beam Request |
| 1 | 2 | Data Length | Length of message being sent |  |
| 3 | 64 (divide in 4x16 registers) | TxInput:  Tx1: Bytes 0-15  Bytes 0-3: T11  Byte 0-1 – I  Byte 2-3 – Q  Bytes 4-7: T12  Byte 4-5: I  Byte 6-7: Q  Bytes 8-11: T13  Byte 8-9: I  Byte 10-11: Q  Bytes 12-15: T14  Byte 12-13: I  Byte 14-15: Q  Tx2: Bytes 16-31  Bytes 16-19: T11  Byte 16-17 – I  Byte 18-19 – Q  Bytes 20-23: T12  Byte 20-21 I  Byte 21-23: Q  Bytes 24-27: T13  Byte 24-25: I  Byte 26-27: Q  Bytes 28-31: T14  Byte 28-29: I  Byte 30-31: Q  Tx3/Txx3: Bytes 32-47  Bytes 32-35: T11  Byte 32-33 – I  Byte 34-35 – Q  Bytes 36-39: T12  Byte 36-37 I  Byte 38-39: Q  Bytes 40-43: T13  Byte 40-41: I  Byte 42-43: Q  Bytes 44-47: T14  Byte 44-45: I  Byte 46-47: Q  Tx4/Txx4: Bytes 48-63  Bytes 48-51: T11  Byte 48-49 – I  Byte 50-51 – Q  Bytes 52-55: T12  Byte 52-53 I  Byte 54-55: Q  Bytes 56-59: T13  Byte 56-57: I  Byte 58-59: Q  Bytes 44-47: T14  Byte 60-61: I  Byte 62-63: Q | T11, T12,…are the coefficients.  Tx1, Tx2… are the Output.  Tx\_port1, Tx\_port2… are the CPRI Inputs | N/A |
| 11 | 64 (divide in 4x16 registers) | RxOutput:  RxPort1: Bytes 0-15  Bytes 0-3: R11  Byte 0-1 – I  Byte 2-3 – Q  Bytes 4-7: R12  Byte 4-5: I  Byte 6-7: Q  Bytes 8-11: R13  Byte 8-9: I  Byte 10-11: Q  Bytes 12-15: R14  Byte 12-13: I  Byte 14-15: Q  RxPort2: Bytes 16-31  Bytes 16-19: R21  Byte 16-17 – I  Byte 18-19 – Q  Bytes 20-23: R22  Byte 20-21 I  Byte 21-23: Q  Bytes 24-27: R33  Byte 24-25: I  Byte 26-27: Q  Bytes 28-31: R44  Byte 28-29: I  Byte 30-31: Q  RxPort3: Bytes 32-47  Bytes 32-35: R31  Byte 32-33 – I  Byte 34-35 – Q  Bytes 36-39: R32  Byte 36-37 I  Byte 38-39: Q  Bytes 40-43: R33  Byte 40-41: I  Byte 42-43: Q  Bytes 44-47: R34  Byte 44-45: I  Byte 46-47: Q  RxPort4: Bytes 48-63  Bytes 48-51: R41  Byte 48-49 – I  Byte 50-51 – Q  Bytes 52-55: R42  Byte 52-53 I  Byte 54-55: Q  Bytes 56-59: R43  Byte 56-57: I  Byte 58-59: Q  Bytes 60-63: R44  Byte 60-61: I  Byte 62-63: Q |  | N/A |

Following is how the coefficients are multiplied with the CPRI input ports to:

1. Get the sync channels in the right beam
2. To support all MIMO combinations

Tx1 T11, T12, T13, T14 Tx\_Port1

Tx2 T21, T22, T23, T24 Tx\_Port2

= \*

Tx3 T31, T32, T33, T34 Tx\_Port3

Tx4 T41, T42, T43, T44 Tx\_Port4

* Note: Matrix based mapping on Tx with most flexibility
* Using Ports for CPRI and Tx and Rx are our chain

### Response

Size of the Message (Fixed/Variable): Fixed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | Map Data Layer To Beam Response |
| 1 | 2 | Data Length | Length of message being sent |  |
| 19 | 1 | Success (1 byte):  SuccessTx1 bit 0  SuccessTx2 bit 1  SuccessTx3 bit 2  SuccessTx4 bit 3  SuccessRx1 bit 4  SuccessRx2 bit 5  SuccessRx3 bit 6  SuccessRx4 bit 7 | Success:0  Failure:1 | N/A |

## PANEL\_CAL\_IQ

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* After Init Req/Rsp
* Whenever the IF signal quality degrades
* Suppress the DC and the IF image

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | IQ Calibration request |
| 1 | 2 | Length of data | Length of data being sent |  |
| 3 | 1 | Channels:  Tx1: Bit 0  Tx2: Bit 1  Tx3: Bit 2  Tx4: Bit 3  Reserved: Bits 4-5  Txx3: Bit 6  Txx4: Bit 7 | 0: Not selected, 1: Selected |  |

### Response

Size of the Message (Fixed/Variable): Variable

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | IQ calibration Response |
| 1 | 2 | Length of data | Length of data being sent |  |
| 3 | 1 | Channels (1 byte):  TX1 bit0  TX2 bit1  TX3/Txx3 bit2  TX4/Txx4 bit3  Reserved bits 4-7 | 0: Not selected, 1: Selected |  |
| 4 | 144 | Results (144 bytes)  CommonAmp (18bits on 4 bytes)  DiffAmp (18bits on 4 bytes)  CommonPhase (16 bits on 4 bytes)  DiffPhase (16 bits on 4 bytes)  IDC (16 bits on 4 bytes)  QDC (16 bits on 4 bytes)  Delay (32 bits on 4 bytes)  Power (32 Bits on 4 bytes)  MSE (32 Bits on 4 bytes) | Repeats for each channel (order 1 to 4 |  |

## PANEL\_CAL\_IF

### Request

Size of the Message (Fixed/Variable): Fixed

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | | Field | Descriptions | | Default Value | | |
| 0 | 1 | | Message Type | Type of message being sent | | IF calibration request | | |
| 1 | 2 | | Length of data | Length of data being sent | |  | | |
| 19 | | 2 | Channel Number: 14 bits | | 4 Values for Tx1..4  4 values for Rx1..4  4 values for Txx3, Txx4, Rxx3, Txx4 (These 4 values are used for Cross Poles) | | Following will be the values used:  This is in binary:  Tx1: 0000 0000 0000 0001  Tx2: 0000 0000 0000 0010  Tx3: 0000 0000 0000 0100  Tx4: 0000 0000 0000 1000  Rsvd: 0000 0000 0001 0000  Rsvd: 0000 0000 0010 0000  Txx3: 0000 0000 0100 0000  Txx4: 0000 0000 1000 0000  Rx1: 0000 0001 0000 0000  Rx2: 0000 0010 0000 0000  Rx3: 0000 0100 0000 0000  Rx4: 0000 1000 0000 0000  Rxx3: 0001 0000 0000 0000  Rxx4: 0010 0000 0000 0000  Setting all bits to 1 will calibrate everything. We could also bitwise OR multiple channel numbers |
| 9 | 1 | | Phase\_Magnitude | 00, 01, 10, 11 (Phase MSB, Mag LSB) | |  | | |
| 10 | 4 | | Tx\_IF\_Frequency |  | |  | | |
| 14 | 4 | | Rx\_IF\_Frequency |  | |  | | |

### Response

Size of the Message (Fixed/Variable): Variable

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | IF calibration response |
| 1 | 2 | Length of data | Length of data being sent |  |
| 3 | 2 | DFE Success  Tx Phase:  Preparation: 1 bit  SetChannel: 1 bit  MeasureTx: 1 bit  Reserve: 1bit  Rx Phase:  Preparation: 1 bit  SetChannel: 1 bit  MeasureTx: 1 bit  MeasureRx: 1 bit  Tx Amp:  Preparation: 1 bit  SetChannel: 1 bit  MeasureTx: 1 bit  Reserve:  Rx Amp:  Preparation: 1 bit  SetChannel: 1 bit  MeasureTx: 1 bit  MeasureRx: 1 bit | 0: Success, 1: Fail |  |
| 4 | 1 | TileResult (8 bits) x RFIC  Bits 0-1: LSB  RESERVED  Comm: 2 bits  Bits 3-2 (In Binary):  00: Good  01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL |  |  |
|  |  | Results   |  |  |  |  | | --- | --- | --- | --- | | TX1 | 1)MAG |  | RFIC 1 | | Value | 7 bits | | 2)PHASE |  |  | | Value | 9 bits | | TX2 | 3)MAG |  |  | | Value | 7 bits | | 4)PHASE |  |  | | Value | 9 bits | | TX3 | 5)MAG |  |  | | Value | 7 bits | | 6)PHASE |  |  | | Value | 9 bits | | TX4 | 7)MAG |  |  | | Value | 7 bits | | 8)PHASE |  |  | | Value | 9 bits | | RX1 | 9)MAG |  |  | | Value | 7 bits | | 10)PHASE |  |  | | Value | 9 bits | | RX2 | 11)MAG |  |  | | Value | 7 bits | | 12)PHASE |  |  | | Value | 9 bits | | RX3 | 13) MAG |  |  | | Value | 7 bits | | 14)PHASE |  |  | | Value | 9 bits | | RX4 | 15)MAG |  |  | | Value | 7 bits | | 16)PHASE |  |  | | Value | 9 bits | |  |  |

## PANEL\_CAL\_BDS

### Request

Size of the Message (Fixed/Variable): Fixed

**Triggers:**

* On a QPanel this is not needed. If there are more than 1 Q panel, then we will need to initiate this under the following scenarios:
* As part of the Calibration after Init Req/Response
* Also as and when needed when the Panel is UP and running – will need inputs from Systems team
* Before proceeding with this BDS Calibration message, the panel should be explicitly stopped by calling the Channel\_Enable (DISABLE).
* When the Panel is Up and running, and then if we initiate BDS\_CAL and it is successful, then must call the Set\_Frequency API to lock the PLLs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | BDS Calibration Request |
| 1 | 2 | Length of data | Length of data being sent |  |
| 3 | 1 | MBRDs (1 byte):  MBRD1 bit0  MBRD2 bit1  MBRD3 bit2  MBRD4 bit3  Reserved bits 4-7 | 0: Not selected, 1: Selected |  |
| 4 | 4 | Frequency | In Hz |  |

### Response

Size of the Message (Fixed/Variable): Variable

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of message being sent | BDS Calibration response |
| 1 | 2 | Length of data | Length of data being sent |  |
| 3 | 1 | MBRD1 bit0  MBRD2 bit1  MBRD3 bit2  MBRD4 bit3  Reserved bits 4-7 | 0: Not selected, 1: Selected |  |
| 4 | 4 | Results (32 bytes):  Amp (18 bits)  Phase (16 bits) | Repeats for each MBRD (order 1 to 4) |  |

## PANEL\_CAL\_RF

### Request (Main Loop 🡪 Panel Control)

Size of the Message (Fixed/Variable): Fixed

Triggers:

* RF Call for all the RFICs will be called during Initialization
* This could also be called selectively when there are issues (some failure path)
* Also could be called during Autonomous Calibration

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | Panel Control RF Cal Request |
| 1 | 2 | Message Length | Length of Message | Fill with Number of Bytes |
|  | 1 | RF CAL Mode | Simultaneous or Sequential | Default will be Sequential.  Note: Simultaneous Option needs to be tested in the lab to find the best way to use it. Even when it is supported, the RFICs will be grouped together to minimize interference. Main Loop will need to group the RFICs appropriately. **For the** **Field Trial we will go with Sequential Mode.** |
| 19 | 2 | Channel Number: 14 bits | 4 Values for Tx1..4  4 values for Rx1..4  4 values for Txx3, Txx4, Rxx3, Txx4 (These 4 values are used for Cross Poles) | Following will be the values used:  This is in binary:  Tx1: 0000 0000 0000 0001  Tx2: 0000 0000 0000 0010  Tx3: 0000 0000 0000 0100  Tx4: 0000 0000 0000 1000  Rsvd: 0000 0000 0001 0000  Rsvd: 0000 0000 0010 0000  Txx3: 0000 0000 0100 0000  Txx4: 0000 0000 1000 0000  Rx1: 0000 0001 0000 0000  Rx2: 0000 0010 0000 0000  Rx3: 0000 0100 0000 0000  Rx4: 0000 1000 0000 0000  Rxx3: 0001 0000 0000 0000  Rxx4: 0010 0000 0000 0000  Setting all bits to 1 will calibrate everything. We could also bitwise OR multiple channel numbers |
| 21 | 1 | Magnitude\_Phase: 2 bits |  | First bit Magnitude  Second bit Phase  ~~Different possible values in Binary:~~  00  01  10  11 |
| 22 | 4 | RF Frequency TX | This is sent in Hz | Use 4 bytes |
| 26 | 4 | RF Frequency RX | This is sent in Hz | Use 4 bytes |

### Response (Panel Control 🡪 Main Loop)

Size of the Message (Fixed/Variable): Variable

Triggers:

* This message is sent in response to an RF Cal Request Message

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Byte Offset | Byte Size | Field | Descriptions | Default Value |
| 0 | 1 | Message Type | Type of Message being sent | RF Cal Response |
| 1 | 2 | Message Length | Length of Message | Variable Length depending on the content |
| 2 | 1 | DFE Success: 1 bit |  | 0: Success  1: Failure |
| 3 | 1 | Number of RFICs |  | Valid range is from: 1..96 |
| Following field (TileResult) is repeated for number of RFICs  Following is repeated for each RFIC (Will be the same as the request, for eg. 1..96 or a subset of this). The order of the response will be the same as the Request, i.e. if RFIC 10 was sent first in the Request then Response will start with RFIC 10. | | | | |
| 4..<Variable Length> | 1 | TileResult (8 bits)  Bits 0-1: LSB  RESERVED  Comm: 2 bits  Bits 3-2 (In Binary):  00: Good  ` 01: RFIC Fail  10: SPI Fail  11: CAN Fail  SuccessOp: 4 bits  Bits 4: OutofRange  Bits 5: PLL  Bits 6: PRECAL  Bits 7: RFCAL | For SuccessOp: ‘0’ good ‘1’ fail |  |
|  | | | | |
|  | | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 1 | 1)PreCalTx | | Done | 1 bit | | | Sign | 1 bit | | | Value | 6 bit | | | 1 | TX1 | 2)MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 3)PHASE | Done | 1 bit | | |  |  |  | Value | 9 bits | | 1 | TX2 | 4)MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 5)PHASE | Done | 1 bit | | | Value | 9 bits | | | 1 | TX3/TXx3 | 6)MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 7)PHASE | Done | 1 bit | | | Value | 9 bits | | | 1 | TX4/TXx4 | 8)MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 9)PHASE | Done | 1 bit | | | Value | 9 bits | | | 2 | 10)Ra  FCAL PLL | | PllLock | 1 bit | | | FreqCalDone | 1 bits | | | FreqCalValue | 7 bit | | | 2 | 11) PLL Assist | | FreqCalDone | 1 bit | | | FreqCalValue | 7 bits | | | Lock | 1 bit | | | 1 | 12) PreCalRx | | Done | 1 bit | | | Sign | 1 bit | | | Value | 6 bit | | | 4 | 13) PLL | | FreqCalDone | 1 bit | | | FreqCalValue  PllLock | 7 bit  1 bit | | | DCOffsetDone | 1 bit | | | DCOffsetSign | 1 bit | | | DCOffsetValue | 7 bit | | | 1 | RX1 | 14)MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 15)PHASE | Done | 1 bit | | | Value | 9 bits | | | 1 | RX2 | 16)MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 17)PHASE | Done | 1 bit | | | Value | 9 bits | | | 1 | RX3/RXx3 | 18) MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 19)PHASE | Done | 1 bit | | | Value | 9 bits | | | 1 | RX4/Rxx4 | 20)MAG | Done | 1 bit | | | Value | 7 bits | | | 2 | 21)PHASE | Done | 1 bit | | | Value | 9 bits | | | Done bit is set to 1, when successful, 0 when something is wrong.  These 33 bytes are repeated for each RFIC, hence this message is of variable size (depending on the number of RFICs that are acted upon in the Request).  **NOTE: From an RFIC point of view, each of these tasks, PreCalTx, Tx1, etc. will be done for all the RFICs together and if we need these grouped per RFIC, Panel Control will need to organize it that way.** | |

# 