Factory FATP Test Plan for J4xx
Station: Scorpius Characterisation
Module: Scorpius
Build: EVT
Release Date: 16 October 2019

This Document Covers the Following Products:
J417, J418, J420, J421

Revision: EVT_V1.2
Radar: <rdar://problem/47434171> [J4xx] Scorpius Factory ERS

Release Notes

Build type	Version	Date	Notes	Author
P1	1.0	5 April, 2019	Initial Release for P1 Build.	Bhushan Koli
	1.1	16 April, 2019	Taken out **Digital Ping** Test and updated command in **Section 6.2 LPP, Section 6.3.2 ASK and Section 6.3.3 FSK**	Bhushan Koli
	1.2	1 May, 2019	Updated section **6.3.2 ASK** to be same as ASK section in Scorpius Test station.	Bhushan Koli
	1.3	14 May, 2019	Updated the **K postions** to include **nominal position** and taken out **Section 6.4 Critical Vrect Measurement test.**	Bhushan Koli
	1.4	20 May, 2019	Update on use of some commands in **Section 6.3 Open Loop Mode**	Bhushan Koli
P2	1.0	8 July, 2019	Initial Release for P2 Build. Changed to **Close loop Testing with B332 as compared to Open Loop with Ginger.**	Bhushan Koli
EVT	1.1	27 September, 2019	Initial Release for EVT Build. Updated the Limits for Comms packet error.	Bhushan Koli/ Rex
	1.2	16 October 2019	Updated Power Flow limits at 0.1C loading condition	Bhushan Koli

Table of Contents

1. Purpose

This document describes the FATP test plan for the J4xx inductive charging Tx module for P1.

2. Scope

The scope of this document is the Scorpius only module of the J4xx products. It covers FATP tests of the following high level features:

	Scorpius Test
LPP ping and delta calculation	✓
NTC readings	✓
Charging & Efficiency - Open Loop	✓
ASK Comms - Open Loop	✓
FSK Comms - Open Loop	✓

3. References

<rdar://problem/47457563> Scorpius Module Factory Documentation and Requirements
<rdar://problem/35600039> [J3xx] Scorpius Test ERS
<rdar://problem/34249633> [B332] Scorpius Test ERS
<TBD> Radar for Scorpius Factory FW releases

Refer to the relevant radars above for information on firmware releases and download procedures.

4. Glossary & Definitions

Charge Rate	0.1C @ 6.5v Vrect	3C @ 8v Vrect	10C @ 14v Vrect
Loading	40mA ballast	~0.9W	3W

SET DAC vs Boost Voltage Table (Based on P2 Tx Hardware)

SET DAC	1600	1400	1200	1000	800	600	400	200
VBoost	5.87V	7.21V	8.54V	9.88V	11.22V	12.56V	10.1.9V	15.20.1V

Taken from the following: <rdar://problem/36367905> Scorpius FW ERS

4.1. Critical and Frequently Used Commands

Tx Static Factory Mode:

The following command needs to be sent to the Tx to tell it to enter a static, 'factory test' type mode. Without this the Tx will default to a user mode (LPP > Digital Ping > Power negotiation > Closed loop).

A power cycle will reset this so it must be run every time the Tx is powered up.

Reset Tx	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x91, ReportPayload={}"	***Only to initiate Close loop testing***
Driver Ready Mode	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x93, ReportPayload={0;0;0;0}"	
Airplane Mode/BT Enable	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x92, ReportPayload={0x02;0;0;0}"	
Tell Tx to enter Static Mode	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x0A, ReportPayload={0x08; 0; 0; 0}" Only to be used once at very beginning of test sequence or if the unit was reset or needs to be reset.	

The following command needs to be sent to the Tx to enable DebugModes.

Debug Mode 1:

smokey ScorpiusHid --run --test "Set" --args "ReportID=0x01, ReportPayload={0x01}"

```
PowerManager::Instance()->Request(pwrReqNum, "chargepump");
PowerManager::Instance()->Request(pwrReqNum, "boost");
PowerManager::Instance()->Request(pwrReqNum, "tonbak");
PowerManager::Instance()->Request(pwrReqNum, "1v8");
PowerManager::Instance()->Request(pwrReqNum, "lpdet");

//enable channels output before start the timer
master_config->master_timer->ChannelEnable(TIMER_CHANNEL_2, true);
master_config->master_timer->ChannelEnable(TIMER_CHANNEL_3, true);
slave_config->slave_timer->ChannelEnable(TIMER_CHANNEL_2, true);
slave_config->slave_timer->ChannelEnable(TIMER_CHANNEL_3, true);

master_config->master_timer->Start();
slave_config->slave_timer->Start();
bridge_status.bridgeRunning = true;
```

Also, debug mode 1 sets DAC to minimum voltage: 1600

Debug Mode 0:

Smokey ScorpiusHid --run --test "Set" --args "ReportID=0x01, ReportPayload={0x00}"

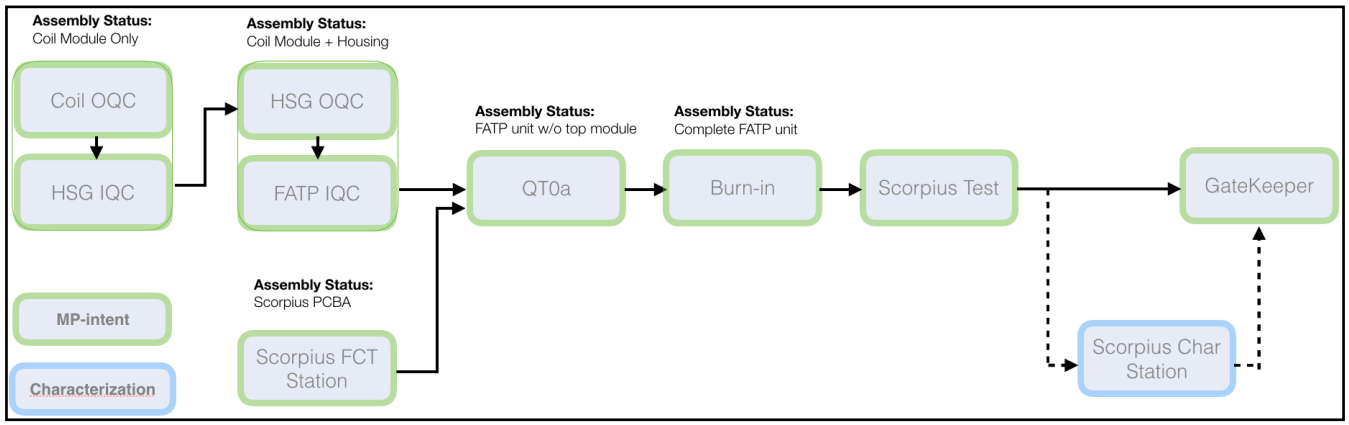
Disables all of the above.

To Reset Scorpius Board on DUT:

```
Socgpio —port 1 —pin 42 —output 0
Socgpio —port 1 —pin 42 —output 1
```

5. Overview

The block diagram below shows the overall end-end test coverage for the inductive Scorpius module. This document covers Scorpius Char Station.



5.1. Summary of Test Coverage

	Kmax	Knom	Kmin
LPP	no load	no load	no load
NTC readings	0.1C, 3C, 10C	0.1C, 3C, 10C	0.1C, 3C, 10C
Close loop			

5.2. Fixture Coupling specs

Throughout this document various tests will have different limits depending on the offset position i.e. coupling. Ensure close attention is paid to the tables shown for the different coupling positions, loads and limits.

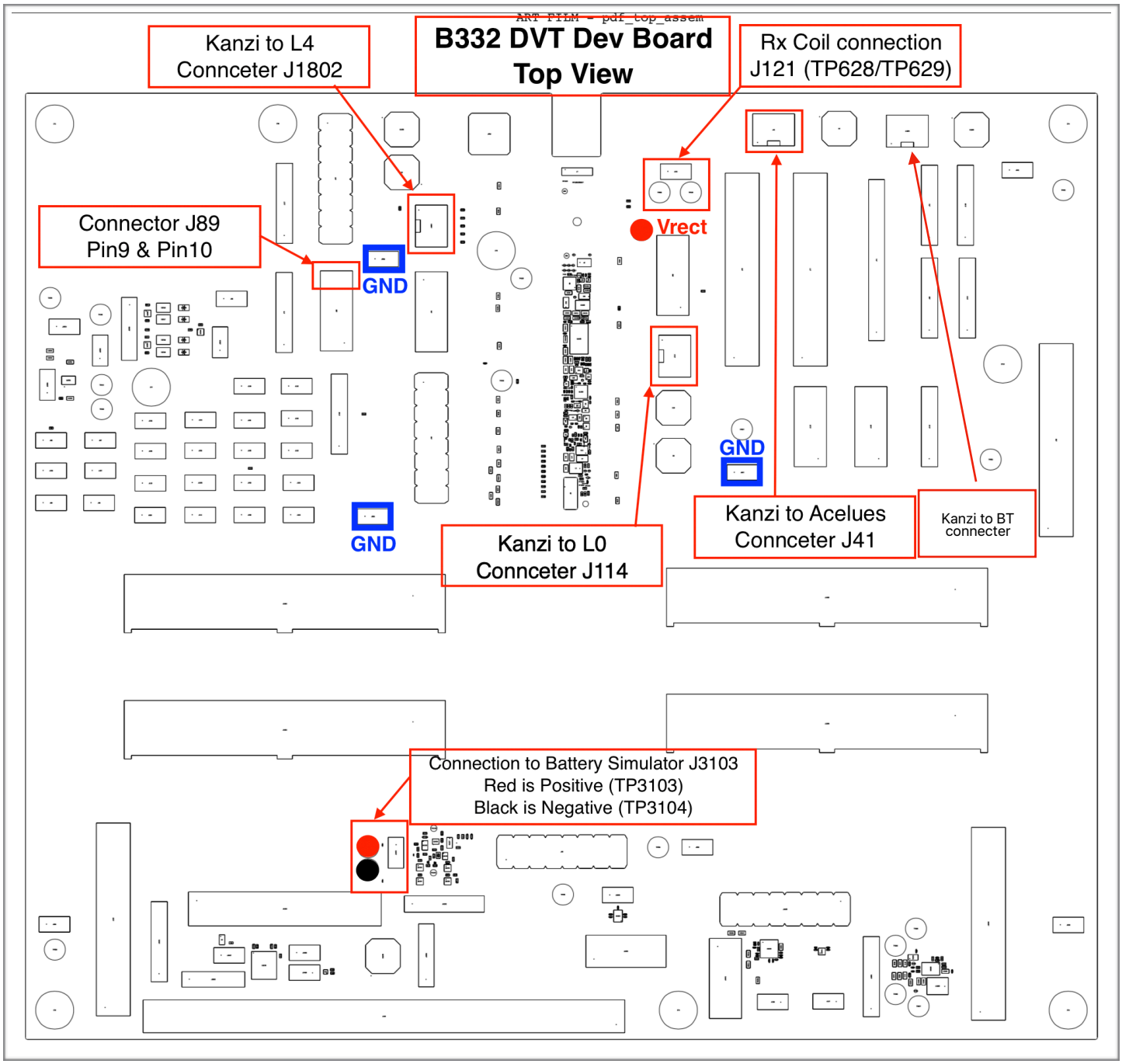
All = all possible positions (MaxK, NomK,MinK)

Physical parameter / InSight Keys Recorded	Position (mm)	K Spec	Measured Results (averaged after 5 readings)	
			J417/J418	J420/J421
KMax	0, 0.83, 0	0.551 - 0.59 +/- 0.015 = 0.536 - 0.605	0.567	0.575
KNom	D1.1, 0.88, L1.1	-		
KMin	D1.5, 0.93, L1.5	0.481 - 0.45 +/- 0.015 = 0.435 - 0.496	0.459	0.460

5.3. B332 Dev Board UART Baud rate

The B332 Dev Board is used to send commands to I2C of Aculeus using UART cable. Follow the steps below to open the B332 Dev Board UART.

1. Connect UART (1MBite baudrate, 1.8V logic) to J89 (Pin9 & Pin 10)
2. Open Terminal
3. nanocom -w 0 -c 1000000,n,8,1
4. Select your Connected UART option.



6. Test Coverage @ Scorpius Char Station

6.1. Tx HW ID Check

Note :-This Test can only be done with Tx FW V413 and above.

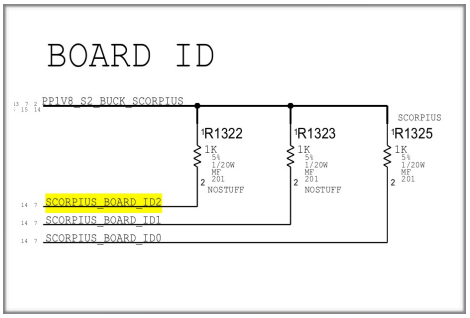
Command to read Tx HW ID:

smokey ScorpiusHid --run --test "Set" --args "ReportID=0x46, ReportPayload={0x05; 0x0; 0x0; 0x0; 0x06; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0}"

smokey ScorpiusHid --run --test "Get" --args "ReportID=0x46"

This replies with
46 BYTE 0 Byte 1 00 where it's 40 + HWID

example:
HW version 2 = 42 => 0x2A



Acceptance criteria:

Product Code	Insight Keys Recorded	HWID Version (Decimal)	HWID Version (Hexadecimal)	Command / Notes
J417/J418	Scorpius_**Tx_HWID**	42	2A	Fail if not equal to 2A
J420/J421		43	2B	Fail if not equal to 2B

6.2. Firmware Version Test

6.2.1. Tx FW Version

Command to read Tx FW version:
smokey ScorpiusHid --run --test "Get" --args "ReportID=0xBB"

This reads back 8 bytes: 0xb0 0x00 0x00 0x00 0x01 0x00 0x03 0x00
- Bootloader Type (byte1&2): 0x00b0
- Bootloader Version (byte3&4): 0x0000
- Main FW Type (byte5&6): 0x0001
- Main FW Version (byte7&8): 0x0003

Command to show the ASCII characters of the git commit hash from which the TX FW was built.
smokey ScorpiusHid --run --test "Get" --args "ReportID=0x73"

This reads back 8 bytes:
Which translates to the ASCII string: c0379514

```
Offset  :       00 01 02 03 04 05 06 07  08 09 0A 0B 0C 0D 0E 0F
-------------------------------------------------------------------
00000000:       63 30 33 37 39 35 31 34
```

6.2.2. Rx FW Version

Command : sys version

```
- Image Versions ------------------------------------------------------------
     Nanoboot [b0]: v0003 -   1572 [   220 free] bytes - HID Development [2]
  Application [01]: v0154 - 489160 [ 33080 free] bytes - HID Development [2]
        BT FW [30]: v0093 - 253920 [270368 free] bytes - HID Development [2]
     Touch FW [20]: v0444 -  62880 [  2656 free] bytes - HID Development [2]
    Touch Cal [c1]: v0000 -      0 [  8192 free] bytes
   Accel Algs [60]: v0010 -   6560 [  1632 free] bytes - HID Development [2]
   Charger FW [50]: v0060 -  52128 [ 46176 free] bytes - HID Development [2]
     Power FW [58]: v0261 -  21088 [ 11680 free] bytes - HID Development [2]
 Power FW OTP [59]: v2020 -  10240 [  2048 free] bytes - None [0]
----------------------------------------------------------------------------
```

6.3. Tx Checking Critical Error

Command to read Tx FW version:
smokey ScorpiusHid --run --test "Get" --args “ReportID=0xE0"
Expected (All OK): 00 00 00 00 00

Bit-mask for HID report payload

Bit Idx	HID Report Byte	Error Code	InSight Key recorded	Post SFCT	Post QT0	Scorpius Test
0	0x00 00 00 01	WATCHDOG	-	Ignore	Ignore	Ignore
1	0x00 00 00 02	SERIAL_MS	Scorpius_Cal_Critical_Error_Serial_MS	Ignore	**Check**	**Check**
2	0x00 00 00 04	SERIAL_FG	-	Ignore	Ignore	Ignore
3	0x00 00 00 08	DEVNAME_DEF	-	Ignore	Ignore	Ignore
4	0x00 00 00 10	SYSCLK_MSI_FREQ_FAILED	-	Ignore	Ignore	Ignore
5	0x00 00 00 20	SYSCLK_HSI_FREQ_FAILED	-	Ignore	Ignore	Ignore
6	0x00 00 00 40	SYSCLK_LSI_FREQ_FAILED	-	Ignore	Ignore	Ignore
7	0x00 00 00 80	SYSCLK_LSE_FREQ_FAILED	-	Ignore	Ignore	Ignore
8	0x00 00 01 00	ADV_FORCED_RESET	-	Ignore	Ignore	Ignore
9	0x00 00 02 00	OD_CALIBRATION_MISSING	Scorpius_Cal_Critical_Error_OD_CAL_MISSING	Ignore	**Check**	**Check**
10	0x00 00 04 00	OD_CTX_MISSING	Scorpius_Cal_Critical_Error_OD_CTX_MISSING	**Check**	**Check**	**Check**
11	0x00 00 08 00	VCTX_DC_OFFSET_CAL_MISSING	Scorpius_Cal_Critical_Error_VCTX_DC_OFFSET_CAL_MISSING	**Check**	**Check**	**Check**
12	0x00 00 10 00	VCTX_AC_SCALING_CAL_MISSING	Scorpius_Cal_Critical_Error_VCTX_AC_SCALING_CAL_MISSING	**Check**	**Check**	**Check**
13	0x00 00 20 00	DAC_CAL_FAILED	Scorpius_Cal_Critical_Error_DAC_CAL_FAILED	Ignore	**Check**	**Check**
14	0x00 00 40 00	FLASH_ECC_CORRECTION	-	Ignore	Ignore	Ignore
15	0x00 00 80 00	VBOOST_XY_CAL_MISSING	Scorpius_Cal_Critical_Error_VBOOST_XY_CAL_MISSING	**Check**	**Check**	**Check**
16	0x00 01 00 00	IBOOST_XY_CAL_MISSING	Scorpius_Cal_Critical_Error_IBOOSY_XY_CAL_MISSING	**Check**	**Check**	**Check**
17	0x00 02 00 00	NTC_XY_CAL_MISSING	-	Ignore	Ignore	Ignore
18	0x00 04 00 00	VCTX_XY_CAL_MISSING	-	Ignore	Ignore	Ignore

Post SFCT: 0x00019c00
Post QT0: 0x0001be02
Scorpius Test: 0x0001be02

6.4. Low Power Ping (LPP)

Description: Check the frequency and inductance of LPP at minK and maxK coupling and record the deltas from the Free Air calibration done at QT0a.

Failure Mode(s) Captured: Poorly assembled / manufactured coils

Test Setup and Procedure:

Step	Description	Interface	Command / Notes
1	Tell Tx to enter Static Mode	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x0A, ReportPayload={0x08; 0; 0; 0}" Only to be used once at very beginning of test sequence or if the unit was reset or needs to be reset.
2	Set Tx to Debug Mode 0	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x01, ReportPayload={0x00}"
3	With Rx coupled at Nominal position, send 5uS LPP pulse	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x11, ReportPayload={5}" "{5}" is the duration of the pulse in uS.
4	Delay 5mS before proceeding	Fixture	
5	Read output of ping command	Tx HID	smokey ScorpiusHid --run --test "Get" --args "ReportID=0x11
6	Send the following commands to retrieve the raw ADC logs.	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x13, ReportPayload={0; 0}" (Send the following command 07 times) smokey ScorpiusHid --run --test "Get" --args "ReportID=0x13"
7	Repeat steps 2 - 5 x 100 times	Fixture & Tx HID	Save all of the data as a single log file for each unit and upload to InSight. Ping 1: F, L, R, Q, ADC Data1,...., ADC dataN . . Ping 100: L, Q, F, R, ADC Data1,...., ADC dataN
8	Apply limits to the first ping only	Fixture	See table below. Ping 1: Apply limits and apply pass / fail criteria Ping 1-100: See step 6 above.
9	Repeat for both Kmin and Kmax		
10	Read value of Ctx	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x78, ReportPayload={0x018; 0x00; 0x00; 0x00}" //Fixture wait 2 sec smokey ScorpiusHid --run --test "Get" --args "ReportID=0x79" The last two bytes of data return the value of Ctx in 10pF: Byte 5 is LSB and Byte 6 is MSB.

Physical Parameter	InSight Keys Recorded	LL	UL	Unit	Offset Positions	Notes
LPP Frequency	KMax_LPP_Frequency	60.1	76.2	kHz	Kmax	
	KNom_LPP_Frequency	n/a	n/a	n/a	Knom	
	KMin_LPP_Frequency	70	82.2	kHz	Kmin	
	KMax_LPP_Frequency_avg	60.1	76.2	kHz	Kmax	
	KNom_LPP_Frequency_avg	n/a	n/a	n/a	Knom	
	KMin_LPP_Frequency_avg	70	82.2	kHz	Kmin	
LPP Inductance	KMax_LPP_Inductance	22.1	27	µH	Kmax	
	KNom_LPP_Inductance	n/a	n/a	n/a	Knom	
	KMin_LPP_Inductance	18.1	24	µH	Kmin	
	KMax_LPP_Inductance_avg	22.1	27	µH	Kmax	
	KNom_LPP_Inductance_avg	n/a	n/a	n/a	Knom	
	KMin_LPP_Inductance_avg	18.1	24	µH	Kmin	
Q_Res	KMax_LPP_Q_Res	0.1	7	-	Kmax	This limits may change depending upon CPk value from actual data collected.
	KNom_LPP_Q_Res	0.1	7	-	Knom	
	KMin_LPP_Q_Res	0.1	7	-	Kmin	
	KMax_LPP_Q_Res_avg	0.1	7	-	Kmax	This limits may change depending upon CPk value from actual data collected.
	KNom_LPP_Q_Res_avg	0.1	7	-	Knom	
	KMin_LPP_Q_Res_avg	0.1	7	-	Kmin	
Δ Tx Inductance	KMax_LPP_Inductance_delta	2.3	11	µH	Kmax	
	KNom_LPP_Inductance_delta	2.3	11	µH	Knom	
	KMin_LPP_Inductance_delta	2.3	11	µH	Kmin	
	KMax_LPP_Inductance_delta_avg	2.3	11	µH	Kmax	
	KNom_LPP_Inductance_delta_avg	2.3	11	µH	Knom	
	KMin_LPP_Inductance_delta_avg	2.3	11	µH	Kmin	
LPP Frequency STD	LPP_Frequency_STDEV	-	-	-	All	
LPP Inductance STD	LPP_Inductance_STDEV	-	0.4	-	All	Catch Flat line error
Q_Res STD	LPP_Q_Res_STDEV	-	-	-	All	
LPP_repeatability		100	100	-	All	LPP repeatability fails if at least one LPP fails Δf or ΔLtx
LPP_Ctx	LPP_Ctx	188.1	207.9	nF	N/A	Only needs to be recorded once.

Breakdown of response format

Byte	Field	Description
0	Frequency_0	**Frequency (Hz):** This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
1	Frequency_1	
2	Frequency_2	
3	Frequency_3	
4	Inductance_0	**Inductance (H):** This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
5	Inductance_1	
6	Inductance_2	
7	Inductance_3	
8	Impedance_0	**Impedance (ohms):** This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
9	Impedance_1	
10	Impedance_2	
11	Impedance_3	
12	QFactor_0	**Q Factor:** This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
13	QFactor_1	
14	QFactor_2	
15	QFactor_3	
16	FrequencyDelta_0	**FrequencyDelta (Hz):** The difference between the frequency of the ping and the frequency of the calibrated free air ping. This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
17	FrequencyDelta_1	
18	FrequencyDelta_2	
19	FrequencyDelta_3	
20	InductanceDelta_0	**InductanceDelta (H):** The difference between the inductance of the ping and the inductance of the calibrated free air ping. This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
21	InductanceDelta_1	
22	InductanceDelta_2	
23	InductanceDelta_3	
24	ImpedanceDelta_0	**ImpedanceDelta (ohms):** The difference between the impedance of the ping and the impedance of the calibrated free air ping. This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
25	ImpedanceDelta_1	
26	ImpedanceDelta_2	
27	ImpedanceDelta_3	
28	QFactorDelta_0	**QFactorDelta (ohms):** The difference between the Q of the ping and the Q of the calibrated free air ping. This must be interpreted as a 32-bit IEEE754 floating point value where byte 0 is the LSB and byte 3 is the MSB.
29	QFactorDelta_1	
30	QFactorDelta_2	
31	QFactorDelta_3	
32	IsDecayLinear_0	**IsDecayLinear:** 32-bit boolean value that can have value 0 or value 1. 0: The LPP ping response does not decay linearly 1: The LPP ping response decays linearly
33	IsDecayLinear_1	
34	IsDecayLinear_2	
35	IsDecayLinear_3	

6.5. Close Loop Mode

6.5.1. Time to 10C + 10C(Fast Charge) + 3C + 0C + COMs testing

Description: Transferring power at various loads / charge rates using full closed loop control and measuring power and efficiency. Time to reach fast charge should be minimise to maximise time spent at 10C.

Failure Mode(s) Captured : Time to 10C > then the budget. Closed loop comms not working as intended due to high ASK and FSK packet error rate.

Test Setup and Procedure: DUT needs to be taken to separate discharge station before this test can be run.

Rx battery (Simulator)SOC: 0% (3.4 V)

Steps	Description	Interface	Command	Insight Key Recorder
	Only to initiate Close loop testing			
1	Move to Kmax position, and after active reference			
2	Set battery to 4.1V	Battery Sim		
3	Reset Tx	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x91, ReportPayload={}"	
4	Driver Ready Mode		smokey ScorpiusHid --run --test "Set" --args "ReportID=0x93, ReportPayload={0;0;0;0}"	
5	Clear ASK and FSK counter		smokey ScorpiusHid --run --test "Set" --args "ReportID=0x20, ReportPayload={}" Note: This automatically enables Tx comms logging (FSK Sent and ASK received)	
6	Airplane Mode/BT Enable		smokey ScorpiusHid --run --test "Set" --args "ReportID=0x92, ReportPayload={0x02;0;0;0}"	
7	Enable Highest Tx Power	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x84, ReportPayload={0x03}"	
8	Check if TX is in CloseLoop	Tx HID	smokey ScorpiusHid --run --test "Get" --args "ReportID=0x0A" Reply :- "**0x04** 0x00 0x00 0x00" 0x04 = Close Loop ; 0x03 = LPP	Close_Loop_respond
	Initialise complete, 10C test start			
11	Set battery to 3.5V	Battery Sim		
12	Repeat step 3-6	Tx HID	Reset and start	
	Monitor "InputVoltage" until it is >5V. This is done by continuously sending pmu sensor command	Rx I2C	pmu sensor Note: pmu sensor cycle is around 50mS	
13	Record time (T1)			CL_To_10C_start
14	Monitor "actualChargeCurrent" until it reaches >520mA. This is done by continuously sending pmu sensor command	Rx I2C	pmu sensor Note: pmu sensor cycle is around 50mS	
15	Record time (T2)			CL_At_10C_start
16	Calculate Time_to_10C		T2-T1=Time_to_10C	CL_Time_to_10C
17	wait 1 seconds for V_{Rect} to stabilise	N/A		
18	Read back comms info from Tx side	Tx HID	smokey ScorpiusHid --run --test "Get" --args "ReportID=0x20" Read 12 bytes: Byte 1-4: Number of FSK packets sent Byte 5-8: Number of ASK packets received (including possible bad packets) Byte 9-12: Number of good ASK packets Please refer to section 7.3 for details	CL_FSK_sent@10C CL_ASK_received@10C CL_ASK_Good_received@10C
19	Clear ASK and FSK counter	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x20, ReportPayload={}" Note: This automatically enables Tx comms logging (FSK Sent and ASK received)	

20	Calculate the overall Packet Error Rate		Packet Error = {(FSKsent) - (ASK_Good_received)}/FSKsent	CL_Overall_Packet Error @10C
21	PMU sensor command: Vsense, Isense, NTC1, NTC2, NTC3, Vctx,	Tx HID	smokey ScorpiusHid --run --test "Get" --args “ReportID=0x31"	CL_VBoost@10C CL_IBoost@10C CL_Vctx_IPeak@10C CL_Vctx_RMS@10C CL_Tx_Temp_Sense1@10C CL_Tx_Temp_Sense2@10C CL_Tx_Temp_Sense3@10C
22	PMU sensor command: InputVoltage (Vrect), InputCurrent (Irect), ActualChargeCurrent	Rx I2C	Pmu sensor 10x average of below Command for Irect: scorpius get irect Command for Vrect: scorpius get vrect	CL_Vrect_Aceleus@10C CL_Irect_Aceleus@10C CL_ICharge_B332@10C
23	Calculate power and efficiency		Rx_Loading_Power =Vrect_Aceleus * Irect_Aceleus Tx_Loading_Power=VBoost * IBoost FOD_threshold =Rx_Loading_Power / Tx_Loading_Power	CL_Rx_Loading_Power@10C CL_Tx_Loading_Power@10C CL_FOD_efficiency@10C
24	Monitor charger Irect until it reaches <520mA. This is done by continuously sending pmu sensor command	Rx I2C	pmu sensor Note: pmu sensor cycle is around 50mS	
25	Record time (T3)			CL_At_10C_end
10C test finished, 3C test start				
26	Monitor “actualChargeCurrent” until it reaches <180mA. This is done by continuously sending pmu sensor command	Rx I2C	pmu sensor Note: pmu sensor cycle is around 50mS	Adding on the 20~30mA error margin during charge current change.
27	Record time (T4)			CL_At_3C_start
28	Calculate Time_at_10C and Time_to_3C		Time_at_10C = T3-T2 and Time_from_10C_to_3C=T4-T3	CL_Time_at_10C CL_Time_from_10C_to_3C
29	wait 5 seconds for V_{Rect} to stabilise	N/A		
30	Repeat step 18 to 20			CL_FSK_sent @3C CL_ASK_received @3C CL_Overall_Packet Error@3C CL_ASK_Good_received@3C
30	Repeat step 21 to 23			CL_VBoost@3C CL_IBoost@3C CL_Vctx_IPeak@3C CL_Vctx_RMS@3C CL_Tx_Temp_Sense1@3C CL_Tx_Temp_Sense2@3C CL_Tx_Temp_Sense3@3C CL_Vrect_Aceleus@3C CL_Irect_Aceleus@3C CL_ICharge_B332@3C CL_Rx_Loading_Power@3C CL_Tx_Loading_Power@3C CL_FOD_efficiency@3C
3C test finished, 0.1C test start				
31	Set battery voltage VoC to 3.2V	Battery Sim		
32	Repeat step 3-6	Tx HID	Reset and start	
33	wait 5 seconds for V_{Rect} to stabilise	N/A		
34	Repeat step 17 to 19			CL_FSK_sent @0.1C CL_ASK_received @0.1C CL_Overall_Packet Error @0.1C CL_ASK_Good_received@0.1C

35	Repeat step 20 to 22			CL_VBoost@0.1C CL_IBoost@0.1C CL_Vctx_IPeak@0.1C CL_Vctx_RMS@0.1C CL_Tx_Temp_Sense1@0.1C CL_Tx_Temp_Sense2@0.1C CL_Tx_Temp_Sense3@0.1C CL_Vrect_Aceleus@0.1C CL_Irect_Aceleus@0.1C CL_ICharge_B332@0.1C CL_Rx_Loading_Power@0.1C CL_Tx_Loading_Power@0.1C CL_FOD_efficiency@0.1C
0.1 C test finished, move to different position				
36	Move to Knom position and repeat step 2 to 36			
37	Move to Kmin position and repeat step 2 to 36			

Limits

Physical parameter	InSight Keys Recorded	LL	UL	Units	K Positions	Notes
CL_Tx_Temp_Sense 1, 2, 3	Kxxx_CL_Tx_Temp1_@xxC Kxxx_CL_Tx_Temp2_@xxC Kxxx_CL_Tx_Temp3_@xxC	18	42	Ea	All	20 - 35degC + error margin, At all loads
CL_VBoost@0.1C	KMax_CL_VBoost@0.1C	5.94	7	V	Kmax	Please refer "In Band Comms Power Packet" Command in section. 7.2
	KNom_CL_VBoost@0.1C	-	-	V	KNom	
	KMin_CL_VBoost@0.1C	5.94	7	V	Kmin	
CL_IBoost@0.1C	KMax_CL_IBoost@0.1C	0.06	0.12	A	Kmax	
	KNom_CL_IBoost@0.1C	-	-	A	KNom	
	KMin_CL_IBoost@0.1C	0.07	0.16	A	Kmin	
CL_Vctx_IPeak@0.1	KMax_CL_Vctx_Ipk@0.1C	0.28	0.55	A	Kmax	
	KNom_CL_Vctx_Ipk@0.1C	-	-	V	KNom	
	KMin_CL_Vctx_Ipk@0.1C	0.4	0.65	A	Kmin	
CL_Vctx_RMS@0.1	KMax_CL_Vctx_RMS@0.1C	0.1	6	V	Kmax	
	KNom_CL_Vctx_RMS@0.1C	-	-	V	KNom	
	KMin_CL_Vctx_RMS@0.1C	3.5	6	V	Kmin	
CL_ICharge_B332@0.1C	KMax_CL_Icharge_Fix@0.1C KNom_CL_Icharge_Fix@0.1C KMin_CL_Icharge_Fix@0.1C	0	20	mA	All	Advised from pencil factory
CL_Vrect_Aceleus@0.1C	KMax_CL_Vrect_Fix@0.1C KNom_CL_Vrect_Fix@0.1C KMin_CL_Vrect_Fix@0.1C	6	7	V		Fixture Cmd: Vrect Target = 6.5v
CL_Irect_Aceleus@0.1C	KMax_CL_Irect_Fix@0.1C KNom_CL_Irect_Fix@0.1C KMin_CL_Irect_Fix@0.1C	0.035	0.045	A		
CL_Rx_Loading_Power@0.1C	KMax_CL_Rx_Loading_power@0.1C KNom_CL_Rx_Loading_power@0.1C KMin_CL_Rx_Loading_power@0.1C	0.21	0.32	W		Vrect * Irect
CL_VBoost@3C	KMax_CL_VBoost@3C	5.94	7.6	V	Kmax	
	KNom_CL_VBoost@3C	-	-	V	KNom	
	KMin_CL_VBoost@3C	5.94	8.5	V	Kmin	
CL_IBoost@3C	KMax_CL_IBoost@3C	0.21	0.44	A	Kmax	
	KNom_CL_IBoost@3C	-	-	A	KNom	

Physical parameter	InSight Keys Recorded	LL	UL	Units	K Positions	Notes
	KMin_CL_IBoost@3C	0.22	0.57	A	Kmin	Please refer "In Band Comms Power Packet" Command in section. 7.2
CL_Vctx_IPeak@3C	KMax_CL_Vctx_Ipk@3C	0.36	0.6	A	Kmax	
	KNom_CL_Vctx_Ipk@3C	-	-	A	KNom	
	KMin_CL_Vctx_Ipk@3C	0.4	0.83	A	Kmin	
CL_Vctx_RMS@3C	KMax_CL_Vctx_RMS@3C	1.5	6	V	Kmax	
	KNom_CL_Vctx_RMS@3C	-	-	V	KNom	
	KMin_CL_Vctx_RMS@3C	4.0.1	7	V	Kmin	
CL_Vrect_Aceleus@3C	KMax_CL_Vrect_Fix@3C KNom_CL_Vrect_Fix@3C KMin_CL_Vrect_Fix@3C	7.5	8.5	V	All	Fixture Cmd: Vrect Target = 8v
CL_ICharge_B332@3C	KMax_CL_Icharge_Fix@3C KNom_CL_Icharge_Fix@3C KMin_CL_Icharge_Fix@3C	140	190	mA		
CL_Irect_Aceleus@3C	KMax_CL_Irect_Fix@3C KNom_CL_Irect_Fix@3C KMin_CL_Irect_Fix@3C	0.08	0.15	A		Fixture Cmd: Irect Target = 113mA
CL_Rx_Loading_Power@3C	KMax_CL_Rx_Loading_power@3C KNom_CL_Rx_Loading_power@3C KMin_CL_Rx_Loading_power@3C	0.3	1	W		Vrect * Irect
CL_VBoost@10C	KMax_CL_VBoost@10C	9.0	12.9	V	Kmax	Please refer "In Band Comms Power Packet" Command in section. 7.2
	KNom_CL_VBoost@10C	-	-	V	KNom	
	KMin_CL_VBoost@10C	9.3	14.5	V	Kmin	
CL_IBoost@10C	KMax_CL_IBoost@10C	0.38	0.64	A	Kmax	
	KNom_CL_IBoost@10C	-	-	A	KNom	
	KMin_CL_IBoost@10C	0.36	0.7	A	Kmin	
CL_Vctx_IPeak@10C	KMax_CL_Vctx_Ipk@10C	0.7	1.1	A	Kmax	
	KNom_CL_Vctx_Ipk@10C	-	-	V	KNom	
	KMin_CL_Vctx_Ipk@10C	1.0	1.4	A	Kmin	
CL_Vctx_RMS@10C	KMax_CL_Vctx_RMS@10C	6	9.6	V	Kmax	
	KNom_CL_Vctx_RMS@10C	-	-	V	KNom	
	KMin_CL_Vctx_RMS@10C	8.5	12	V	Kmin	
CL_ICharge_B332@10C	KMax_CL_Icharge_Fix@10C KNom_CL_Icharge_Fix@10C KMin_CL_Icharge_Fix@10C	520	580	mA	All	550mA nominal, with +/-30mA (advice from pencil factory)
CL_Vrect_Aceleus@10C	KMax_CL_Vrect_Fix@10C KNom_CL_Vrect_Fix@10C KMin_CL_Vrect_Fix@10C	13.5	14.5	V		Fixture Cmd: Vrect Target = 14v
CL_Irect_Aceleus@10C	KMax_CL_Irect_Fix@10C KNom_CL_Irect_Fix@10C KMin_CL_Irect_Fix@10C	0.16	0.28	A		Fixture Cmd: Irect Target = 224mA
CL_Rx_Loading_Power@10C	KMax_CL_Rx_Loading_power@10C KNom_CL_Rx_Loading_power@10C KMin_CL_IRx_Loading_power@10C	2	3.35	W		Vrect * Irect
CL_FOD_threshold@0.1C	KMax_CL_FOD_Efficiency@0.1C	0.19	65	%	Kmax	Rx_Power / (Vboost * IBoost)
CL_FOD_threshold@3C	KMax_CL_FOD_Efficiency@3C	58	70	%		Rx_Power / (Vboost * IBoost)
CL_FOD_threshold@10C	KMax_CL_FOD_Efficiency@10C	56	69	%		Rx_Power / (Vboost * IBoost)
CL_FOD_threshold@0.1C	KNom_CL_FOD_Efficiency@0.1C	n/a	n/a	%		Rx_Power / (Vboost * IBoost)

Physical parameter	InSight Keys Recorded	LL	UL	Units	K Positions	Notes
CL_FOD_threshold@3C	KNom_CL_FOD_Efficiency@3C	n/a	n/a	%	KNom	Rx_Power / (Vboost * IBoost)
CL_FOD_threshold@10C	KNom_CL_FOD_Efficiency@10C	n/a	n/a	%		Rx_Power / (Vboost * IBoost)
CL_FOD_threshold@0.1C	KMin_CL_FOD_Efficiency@0.1C	0.10	55	%	Kmin	Rx_Power / (Vboost * IBoost)
CL_FOD_threshold@3C	KMin_CL_FOD_Efficiency@3C	47	63	%		Rx_Power / (Vboost * IBoost)
CL_FOD_threshold@10C	KMin_CL_FOD_Efficiency@10C	47	62	%		Rx_Power / (Vboost * IBoost)

Comms Test Limits

Parameter / Key Recorded		LL	UL	Units	K Positions	Notes
CL_FSK_sent@10C	Kmin_CL_FSK_sent @10C Knom_CL_FSK_sent @10C Kmax_CL_FSK_sent @10C	-	-	-	All	No limit, data depending on time spent at such load
CL_ASK_received@10C	Kmin_CL_ASK_received @10C Knom_CL_ASK_received @10C Kmax_CL_ASK_received @10C					
CL_ASK_Good_received@10C	Kmin_CL_ASK_Good_received @10C Knom_CL_ASK_Good_received @10C Kmax_CL_ASK_Good_received @10C					
CL_Overall Packet Error@10C	Kmin_CL_Overall_PacketError @10C Knom_CL_Overall_PacketError @10C Kmax_CL_Overall_PacketError @10C	-1	0	-		Packet Error = {(FSKsent) - (ASK_Good_received - 3)}
CL_FSK_sent@3C	Kmin_CL_FSK_sent @3C Knom_CL_FSK_sent @3C Kmax_CL_FSK_sent @3C			-	All	No limit, data depending on time spent at such load
CL_ASK_received@3C	Kmin_CL_ASK_received @3C Knom_CL_ASK_received @3C Kmax_CL_ASK_received @3C					
CL_ASK_Good_received@3C	Kmin_CL_ASK_Good_received @3C Knom_CL_ASK_Good_received @3C Kmax_CL_ASK_Good_received @3C					
CL_Overall_Packet Error@3C	Kmin_CL_Overall_PacketError @3C Knom_CL_Overall_PacketError @3C Kmax_CL_Overall_PacketError @3C	-1	1	-		Packet Error = {(FSKsent) - (ASK_Good_received)}
CL_FSK_sent@0.1C	Kmin_CL_FSK_sent @0.1C Knom_CL_FSK_sent @0.1C Kmax_CL_FSK_sent @0.1C			-	All	No limit, data depending on time spent at such load
CL_ASK_received@0.1C	Kmin_CL_ASK_received @0.1C Knom_CL_ASK_received @0.1C Kmax_CL_ASK_received @0.1C					
CL_ASK_Good_received@0.1C	Kmin_CL_ASK_Good_received @0.1C Knom_CL_ASK_Good_received @0.1C Kmax_CL_ASK_Good_received @0.1C			-		
CL_Overall_Packet Error@0.1C	Kmin_CL_Overall_PacketError @0.1C Knom_CL_Overall_PacketError @0.1C Kmax_CL_Overall_PacketError @0.1C	-1	0	-		Packet Error = {(FSKsent) - (ASK_Good_received - 3)}

Test Parameter /Insight Key Recorded		LL	UL	Units	Notes
Time to 10C	Kmin	0.5	2.5	Sec	
	Kmax	0.5	2.5		
Time at 10C	Kmin	11	14	Sec	
	Kmax	11	14		
Time from 10C to 3C	Kmin	0.1	1	Sec	
	Kmax	0.1	1		

7. Decoding Command Readback

7.1. Breakdown of PMU Sensor Command

RID::SENSORS (0x33) GET report payload				
Sensor ID	**Type**	**Size (Bytes, LSB first)**	**Signed**	**Offset Index (REV5)**
VBoost	Raw	2	No	0
	Converted	2	Yes	2
ISense	Raw	2	No	4
	Converted	2	Yes	6
NTC1	Raw	2	No	8
	Converted	2	Yes	10
NTC2	Raw	2	No	12
	Converted	2	Yes	14
NTC3	Raw	2	No	16
	Converted	2	Yes	18
VCtx	I peak (mA)	4	Yes (float)	20
	Ctx (F)	4	Yes (float)	24
	AC scaling factor	4	Yes (float)	28
	RMS	2	Yes	0.12
VCtx	Min Slope (mV/nS)	2	Yes	0.14
	Max Slope (mV/nS)	2	Yes	0.16
	VCtx Raw Counts Byte 0	2	No	0.18
	...	...	...	...
	VCtx Raw Counts Byte 16	2	No	72

7.2. Breakdown of Power packet via IBC

Protection::RID_POWERCNT (0x31) Get Report					
Byte offset	**Value**	**Comment**	**Byte offset**	**Value**	**Comment**
0	X0	BYTE0 of VCTX peak (int32_t)	14	D0	BYTE0 of NTC1 (int16_t)
1	X1	BYTE1 of VCTX peak (int32_t)	15	D1	BYTE1 of NTC1 (int16_t)
2	X2	BYTE2 of VCTX peak (int32_t)	16	E0	BYTE0 of NTC2 (int16_t)
3	X3	BYTE3 of VCTX peak (int32_t)	17	E1	BYTE1 of NTC2 (int16_t)
4	Y0	BYTE0 of VCTX RMS (int16_t)	18	F0	BYTE0 of NTC3 (int16_t)
5	Y1	BYTE1 of VCTX RMS (int16_t)	19	F1	BYTE1 of NTC3 (int16_t)
6	Z0	BYTE0 of VSNS (int16_t)	20	H	vctx invalid = 0 , vctx valid = 1
7	Z1	BYTE1 of VSNS (int16_t)	21	I	vsns invalid = 0 , vsns valid = 1
8	A0	BYTE0 of ISNS (int16_t)	22	J	isns invalid = 0 , isns valid = 1
9	A1	BYTE1 of ISNS (int16_t)	23	K	vrect invalid = 0 , vrect valid = 1
10	B0	BYTE0 of VRECT (int16_t)	24	L	irect invalid = 0 , irect valid = 1
11	B1	BYTE1 of VRECT (int16_t)	25	M	ntc1 invalid = 0 , ntc1 valid = 1
12	C0	BYTE0 of IRECT (int16_t)	26	N	ntc2 invalid = 0 , ntc2 valid = 1
13	C1	BYTE1 of IRET (int16_t)	27	O	ntc0.1 invalid = 0 , ntc3 valid = 1

7.3. Breakdown of Com Packets

Protection::RID_POWERCNT (0x20) Get Report		
Byte offset	**Value**	**Comment**
0	D	BYTE0 (LSB) of Number of FSK packets sent
1	C	BYTE1 of Number of FSK packets sent
2	B	BYTE2 of Number of FSK packets sent
3	A	BYTE3 (MSB) of Number of FSK packets sent
4	D	BYTE0 (LSB) of Number of ASK packets received (including possible bad packets)
5	C	BYTE1 of Number of ASK packets received
6	B	BYTE2 of Number of ASK packets received
7	A	BYTE3 (MSB) of Number of ASK packets received
8	D	BYTE0 (LSB) of Number of good ASK packets
9	C	BYTE1 of Number of good ASK packets
10	B	BYTE2 of Number of good ASK packets
11	A	BYTE3 (MSB) of Number of good ASK packets

8. Appendix

8.1. Open Loop Mode

Note: Only for reference or FA not for ongoing testing.

8.1.1. Power Flow Tests

Important commands for this section are as follows:

Description	Command
Tell Tx to enter Static Mode	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x0A, ReportPayload={0x08; 0; 0; 0}" Only to be used once at very beginning of test sequence or if the unit was reset or needs to be reset.
Set Tx to DebugMode 1	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x01, ReportPayload={0x01}"
PMU sensor command for following variables: VBoost, IBoost, Vctx, NTC1, NTC2, NTC3	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x33, ReportPayload={0x28; 0x00}" //Fixture wait 2 sec smokey ScorpiusHid --run --test "Get" --args "ReportID=0x33"
Tell Rx to go into static mode	i2c rawwrite charger 10 0 2e 09 02 01
Power packet via in-band comms to retrieve the following variables: VBoost, IBoost, Vrect, Irect, Vctx	smokey ScorpiusHid --run --test "Set" —args "ReportID=0x031, ReportPayload={}" smokey ScorpiusHid --run --test "Get" --args "ReportID=0x031"

Description: Transferring power at various loads / charge rates (0.1C, 3C, 10C) and positions (Kmin, Knom & Kmax) and measuring power and efficiency.

Failure Mode(s) Captured: Efficiency failures potentially resulting in poorly aligned magnetics / coils.

Test Setup and Procedure:

Order of load ramping as follows:

- Adjust bridge phase from 0 - 180 degrees to reach target Vrect at desired load.
- If target Vrect still cannot be achieved with a phase shift of 180 degrees?
- Start increasing VBoost
- VBoost should only be adjusted when phase = 180 degrees.

Physical parameter	InSight Keys Recorded	LL	UL	Units	K Positions	Notes
Tx Frequency Check	Tx_Frequency	125k	130k	Hz	All	Nominal value: 127795Hz, At all loads
PWM Dead Time	PWM Dead Time	50	200	nS		Nominal value: 100nS, At all loads
Tx_Temp_Sense_Raw 1, 2, 3	Kxxx_Tx_Temp1_Raw@xxC Kxxx_Tx_Temp2_Raw@xxC Kxxx_Tx_Temp3_Raw@xxC	2339	1600	Ea		Raw ADC (20 - 35degC + error margin), At all loads
Tx_Temp_Sense 1, 2, 3	Kxxx_Tx_Temp1_@xxC Kxxx_Tx_Temp2_@xxC Kxxx_Tx_Temp3_@xxC	18	42	Ea		20 - 35degC + error margin, At all loads

Physical parameter	InSight Keys Recorded	LL	UL	Units	K Positions	Notes
VBoost @ 0.1C	KMax_VBoost@0.1C	5.94	7	V	Kmax	See "PMU Sensor Command"
	KNom_VBoost@0.1C	-	-	V	KNom	
	KMin_VBoost@0.1C	5.94	7	V	Kmin	
IBoost @ 0.1C	KMax_IBoost@0.1C	0.08	0.12	A	Kmax	
	KNom_IBoost@0.1C	-	-	A	KNom	
	KMin_IBoost@0.1C	0.07	0.16	A	Kmin	
Vctx_IPeak @ 0.1	KMax_Vctx_Ipk@0.1C	0.28	0.55	A	Kmax	
	KNom_Vctx_Ipk@0.1C	-	-	V	KNom	
	KMin_Vctx_Ipk@0.1C	0.4	0.65	A	Kmin	
Vctx_RMS @ 0.1	KMax_Vctx_RMS@0.1C	0.1	6	V	Kmax	
	KNom_Vctx_RMS@0.1C	-	-	V	KNom	
	KMin_Vctx_RMS@0.1C	0.1.5	6	V	Kmin	
Vrect_Fix @ 0.1C	KMax_Vrect_Fix@0.1C KNom_Vrect_Fix@0.1C KMin_Vrect_Fix@0.1C	6	7	V	All	Fixture Cmd: Vrect Target = 6.5v
Irect_Fix @ 0.1C	KMax_Irect_Fix@0.1C KNom_Irect_Fix@0.1C KMin_Irect_Fix@0.1C	0.035	0.045	A		Iktara ballast load = 40mA. No fixture load required.
VBoost @ 3C	KMax_VBoost@3C	5.94	7.6	V	Kmax	See "PMU Sensor Command"
	KNom_VBoost@3C	-	-	V	KNom	
	KMin_VBoost@3C	5.94	8.5	V	Kmin	
IBoost @ 3C	KMax_IBoost@3C	0.21	0.44	A	Kmax	
	KNom_IBoost@3C	-	-	A	KNom	
	KMin_IBoost@3C	0.22	0.57	A	Kmin	
Vctx_IPeak @ 3C	KMax_Vctx_Ipk@3C	0.0.16	0.6	A	Kmax	
	KNom_Vctx_Ipk@3C	-	-	A	KNom	
	KMin_Vctx_Ipk@3C	0.4	0.83	A	Kmin	
Vctx_RMS @ 3C	KMax_Vctx_RMS@3C	0.1.5	6	V	Kmax	
	KNom_Vctx_RMS@3C	-	-	V	KNom	
	KMin_Vctx_RMS@3C	4.0.1	7	V	Kmin	
Vrect_Fix @ 3C	KMax_Vrect_Fix@3C KNom_Vrect_Fix@3C KMin_Vrect_Fix@3C	7.5	8.5	V	All	Fixture Cmd: Vrect Target = 8v
Irect_Fix @ 3C	KMax_Irect_Fix@3C KNom_Irect_Fix@3C KMin_Irect_Fix@3C	0.08	0.15	A		Fixture Cmd: Irect Target = 113mA
Rx_Loading_Power @ 0.1C	KMax_Rx_Loading_power@0.1C KNom_Rx_Loading_power@0.1C KMin_IRx_Loading_power@0.1C	0.8	1	W		Vrect * Irect
VBoost @ 10C	KMax_VBoost@10C	9.0	12.9	V	Kmax	
	KNom_VBoost@10C	-	-	V	KNom	

Physical parameter	InSight Keys Recorded	LL	UL	Units	K Positions	Notes
	KMin_VBoost@10C	9.3	14.5	V	Kmin	See "PMU Sensor Command"
IBoost @ 10C	KMax_IBoost@10C	0.38	0.64	A	Kmax	
	KNom_IBoost@10C	-	-	A	KNom	
	KMin_IBoost@10C	0.36	0.7	A	Kmin	
Vctx_IPeak @ 10C	KMax_Vctx_Ipk@10C	0.7	1.1	A	Kmax	
	KNom_Vctx_Ipk@10C	-	-	V	KNom	
	KMin_Vctx_Ipk@10C	1.0	1.4	A	Kmin	
Vctx_RMS @ 10C	KMax_Vctx_RMS@10C	6	9.6	V	Kmax	
	KNom_Vctx_RMS@10C	-	-	V	KNom	
	KMin_Vctx_RMS@10C	8.5	12	V	Kmin	
Vrect_Fix @ 10C	KMax_Vrect_Fix@10C KNom_Vrect_Fix@10C KMin_Vrect_Fix@10C	13.5	14.5	V	All	Fixture Cmd: Vrect Target = 14v
Irect_Fix @ 10C	KMax_Irect_Fix@10C KNom_Irect_Fix@10C KMin_Irect_Fix@10C	0.217	0.231	A		Fixture Cmd: Irect Target = 224mA
Rx_Loading_Power @ 10C	KMax_Rx_Loading_power@10C KNom_Rx_Loading_power@10C KMin_IRx_Loading_power@10C	2.9	3.35	W		Vrect * Irect
FOD_threshold @ 0.1C	KMax_FOD_Efficiency@0.1C	0.19	65	%	Kmax	Rx_Power / (Vboost * IBoost)
FOD_threshold @ 3C	KMax_FOD_Efficiency@3C	54.3	65.5	%		Rx_Power / (Vboost * IBoost)
FOD_threshold @ 10C	KMax_FOD_Efficiency@10C	54.5	65.7	%		Rx_Power / (Vboost * IBoost)
FOD_threshold @ 0.1C	KNom_FOD_Efficiency@0.1C	n/a	n/a	%	KNom	Rx_Power / (Vboost * IBoost)
FOD_threshold @ 3C	KNom_FOD_Efficiency@3C	n/a	n/a	%		Rx_Power / (Vboost * IBoost)
FOD_threshold @ 10C	KNom_FOD_Efficiency@10C	n/a	n/a	%		Rx_Power / (Vboost * IBoost)
FOD_threshold @ 0.1C	KMin_FOD_Efficiency@0.1C	0.10	55	%	Kmin	Rx_Power / (Vboost * IBoost)
FOD_threshold @ 3C	KMin_FOD_Efficiency@3C	44	59.5	%		Rx_Power / (Vboost * IBoost)
FOD_threshold @ 10C	KMin_FOD_Efficiency@10C	46	60	%		Rx_Power / (Vboost * IBoost)

8.1.2. ASK

Description: Decoding fixed number of packets check the overall packet error rate and integrity.
Failure Mode Captured:
Test Procedure:

Step	Description	Interface	Command / Notes
1	Tell Tx to enter Static Mode	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x0A, ReportPayload={0x08; 0; 0; 0}"
2	Set Tx to DebugMode 0 (Confirm Bridge is turned off)	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x01, ReportPayload={0x00}"
3	Delay 100ms	Fixture	Wait 100ms
4	Set Tx to DebugMode1	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x01, ReportPayload={0x01}"
5	Lock the Rx I2C for Aculeus	Rx	i2c lock charger
6	Tell Rx to go into static mode	Rx	i2c rawwrite charger 10 0 2e 09 02 01
7	Choose Comm1	Rx	i2c rawwrite charger 0F 0 2d 05 00 01
8	Tell Tx to capture fixed number of ASK packets and what the timeout period should be. i.e. 10 packets, 2seconds	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x1F, ReportPayload={0x00; 0x0A; 0x02}"
9	Tell Rx to send 10 ASK packets with fixed data	Rx	i2c rawwrite charger 0F 01 07 33 55 0A
10	Wait 3 second for RX to send packets before reading buffer	Fixture	Wait 3 second
11	Read back data that was captured from the Tx. *Each packet is 8 bytes. See Note further below for breakdown of each packet.	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x21, ReportPayload={0; 0; 0x80; 0}" smokey ScorpiusHid --run --test "Get" --args "ReportID=0x21" Command to read first 128 bytes
12	Record the results as per the table below.		Each ASK packet contains 8 bytes which are broken down as follows: Byte 1: Packet Number Byte 2: Error state (bit 7-4); preamble number (bit 3-0) Breakdown of error states as follows: 0000 = ASK_OK 0001 = ASK_INVALID_START_BIT 0010 = ASK_INVALID_STOP_BIT 0011 = ASK_PARITY_ERROR 0100 = ASK_INVALID_BIT 0101 = ASK_CHECKSUM_MISMATCH Byte 3 - 4: Packet Interval Time Byte 5 - 8: Header, 2 Data, 1 Checksum
13	Repeat for all the following loads and positions		Kmax: 0.1C, 3C, 10C Knom: 0.1C, 3C, 10C Kmin: 0.1C, 3C, 10C
14	UnLock the Rx I2C for Aculeus	Rx	i2c unlock charger

Parameters	InSight Keys Recorded	LL	UL	Units	Limits Reference / Justification
Minimum Preamble Bits	KMax_ASK_Min_Preambles@ __C KNom_ASK_Min_Preambles@ __C KMin_ASK_Min_Preambles@ __C	4	11	Bits	11 bit preamble and minimum of 4 required for successful ASK.
Below Expected Preamble Bits	KMax_ASK_Below_Expected_Preambles@ __C KNom_ASK_Below_Expected_Preambles@ __C KMin_ASK_Below_Expected_Preambles@ __C	n/a	n/a		No pass / fail criteria. Data only
Error State - Invalid Start Bit	KMax_ASK_ErrorState_Invalid_Start_Bit@ __C KNom_ASK_ErrorState_Invalid_Start_Bit@ __C KMin_ASK_ErrorState_Invalid_Start_Bit@ __C	0	0		Expectation = 100% Packet Integrity
Error State - Invalid Stop Bit	KMax_ASK_ErrorState_Invalid_Stop_Bit@ __C KNom_ASK_ErrorState_Invalid_Stop_Bit@ __C KMin_ASK_ErrorState_Invalid_Stop_Bit@ __C	0	0		Expectation = 100% Packet Integrity
Error State - Parity Error	KMax_ASK_ErrorState_Parity_Error@ __C KNom_ASK_ErrorState_Parity_Error@ __C KMin_ASK_ErrorState_Parity_Error@ __C	0	0		Expectation = 100% Packet Integrity
Error State - Invalid Bit	KMax_ASK_ErrorState_Invalid_Bit@ __C KNom_ASK_ErrorState_Invalid_Bit@ __C KMin_ASK_ErrorState_Invalid_Bit@ __C	0	0		Expectation = 100% Packet Integrity
Error State -Checksum Mismatch	KMax_ASK_ErrorState_Checksum_Mismatch@ __C KNom_ASK_ErrorState_Checksum_Mismatch@ __C KMin_ASK_ErrorState_Checksum_Mismatch@ __C	0	0		Expectation = 100% Packet Integrity
ASK Packet Header	KMax_ASK_Packet_Header@ __C KNom_ASK_Packet_Header@ __C KMin_ASK_Packet_Header@ __C	0.12	0.12	Int	Expected Header = 0x20 (Hex)
ASK Packet Data	KMax_ASK_Packet_Data@ __C KNom_ASK_Packet_Data@ __C KMin_ASK_Packet_Data@ __C	21811	21811	Int	Expected Packet Data = 0x55 0x0.10.1 (Hex)
ASK Packet Checksum	KMax_ASK_Packet_Checksum@ __C KNom_ASK_Packet_Checksum@ __C KMin_ASK_Packet_Checksum@ __C	70	70	Int	Expected Checksum = 0x46 (Hex)
Max Packet Interval	KMax_ASK_Max_Packet_Interval@ __C KNom_ASK_Max_Packet_Interval@ __C KMin_ASK_Max_Packet_Interval@ __C	90.1	100.1	ms	Nominal 100ms packet interval
Min Packet Interval	KMax_ASK_Min_Packet_Interval@ __C KNom_ASK_Min_Packet_Interval@ __C KMin_ASK_Min_Packet_Interval@ __C	90.1	100.1	ms	Nominal 100ms packet interval
Number of good packets	KMax_ASK_Number of Good Packets@ __C KNom_ASK_Number of Good Packets@ __C KMin_ASK_Number of Good Packets@ __C	10	10		Expectation that all 10 packets are decoded.

8.1.3.FSK

Description: Decoding certain number of packets for fixed time to check the overall packet error rate and integrity.
Failure Mode Captured:
Test Procedure: Open loop mode

Step	Description	Interface	Command / Notes
1	Tell Tx to go to Factory Support Mode	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x0A, ReportPayload={0x08; 0; 0; 0}"
2	Set Tx to DebugMode1	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x01, ReportPayload={0x01}"
3	Lock the Rx I2C for Aculeus	RX	i2c lock charger
4	Clear Rx Buffer	Rx	i2c rawwrite charger 0F 01 0A 00 00 00
5	Tell Tx to send 10 FSK packets with fixed data.	Tx HID	smokey ScorpiusHid --run --test "Set" --args "ReportID=0x25, ReportPayload={0x**64**; 0x00; 0x0A; 0x00; 0x00; 0x2E; 0x55; 0xAA}" Time interval between FSK packets is **100**ms.
6	Wait Period		Allow 2 seconds after sending above command before issuing Rx read commands below. ***This is very important as you may miss 1~2 packets if step 6 is performed too quickly.***
7	Read back data that was captured from the Rx. *Each packet is 8 bytes. See Note further below for breakdown of each packet.	Rx	i2c rawwrite charger 12 C8 i2c rawread 201
8	Record the results as per the table below.		Each FSK packet contains 8 bytes which are broken down as follows: Byte 1: Packet Number Byte 2: Error state 0000 = FSK_OK 0001 = FSK_INVALID_START_BIT 0010 = FSK_INVALID_STOP_BIT 0011 = FSK_PARITY_ERROR 0100 = FSK_INVALID_BIT 0101 = FSK_CHECKSUM_MISMATCH 0110 = FSK_INVALID_PREAMBLE Byte 3 - 4: Packet Interval Time Byte 5 - 8: Header, 2 Data, 1 Checksum
9	Repeat for all the following loads and positions		Kmax: 0.1C, 3C, 10C Knom: 0.1C, 3C, 10C Kmin: 0.1C, 3C, 10C
10	UnLock the Rx I2C for Aculeus	Rx	i2c unlock charger

Parameters	InSight Keys Recorded	LL	UL	Units	Limits Reference / Justification
Error State - Invalid Start Bit	KMax_FSK_ErrorState_Invalid_Start_Bit@ __C KNom_FSK_ErrorState_Invalid_Start_Bit@ __C KMin_FSK_ErrorState_Invalid_Start_Bit@ __C	0	0		Expectation = 100% Packet Integrity
Error State - Invalid Stop Bit	KMax_FSK_ErrorState_Invalid_Stop_Bit@ __C KNom_FSK_ErrorState_Invalid_Stop_Bit@ __C KMin_FSK_ErrorState_Invalid_Stop_Bit@ __C	0	0		Expectation = 100% Packet Integrity
Error State - Parity Error	KMax_FSK_ErrorState_Parity_Error@ __C KNom_FSK_ErrorState_Parity_Error@ __C KMin_FSK_ErrorState_Parity_Error@ __C	0	0		Expectation = 100% Packet Integrity
Error State - Invalid Bit	KMax_FSK_ErrorState_Invalid_Bit@ __C KNom_FSK_ErrorState_Invalid_Bit@ __C KMin_FSK_ErrorState_Invalid_Bit@ __C	0	0		Expectation = 100% Packet Integrity
Error State -Checksum Mismatch	KMax_FSK_ErrorState_Checksum_Mismatch@ __C KNom_FSK_ErrorState_Checksum_Mismatch@ __C KMin_FSK_ErrorState_Checksum_Mismatch@ __C	0	0		Expectation = 100% Packet Integrity
Error State - Invalid Preamble	KMax_FSK_ErrorState_Invalid_Preambles@ __C KNom_FSK_ErrorState_Invalid_Preambles@ __C KMin_FSK_ErrorState_Invalid_Preambles@ __C	n/a	n/a		No pass / fail criteria. Data only
FSK Packet Header	KMax_FSK_Packet_Header@ __C KNom_FSK_Packet_Header@ __C KMin_FSK_Packet_Header@ __C	46	46	Int	Expected Header = 0x2E (Hex)
FSK Packet Data	KMax_FSK_Packet_Data@ __C KNom_FSK_Packet_Data@ __C KMin_FSK_Packet_Data@ __C	21930	21930	Int	Expected Packet Data = 0x55 0xAA (Hex)
FSK Packet Checksum	KMax_FSK_Packet_Checksum@ __C KNom_FSK_Packet_Checksum@ __C KMin_FSK_Packet_Checksum@ __C	209	209	Int	Expected Checksum = 0xD1 (Hex)
Max Packet Interval	KMax_FSK_Max_Packet_Interval@ __C KNom_FSK_Max_Packet_Interval@ __C KMin_FSK_Max_Packet_Interval@ __C	90	110	ms	Nominal 100ms packet interval
Min Packet Interval	KMax_FSK_Min_Packet_Interval@ __C KNom_FSK_Min_Packet_Interval@ __C KMin_FSK_Min_Packet_Interval@ __C	90	110	ms	Nominal 100ms packet interval
Number of good packets	KMax_FSK_Number of Good Packets@ __C KNom_FSK_Number of Good Packets@ __C KMin_FSK_Number of Good Packets@ __C	10	10		Expectation that all 10 packets are decoded.