TEXAS INSTRUMENTS
BQ79631-Q1
SLUSEC2A – DECEMBER 2021 – REVISED NOVEMBER 2023

BQ79631-Q1 UIR Sensor with Voltage, Current, and Insulation Resistance Monitoring in EV/BMS HV Automotive Applications

1 Features

- Qualified for automotive applications
- AEC-Q100 Qualified with the following results:
  - Device temperature grade 1: –40°C to +125°C ambient operating temperature range
  - Device HBM ESD classification level 2
  - Device CDM ESD classification level C4B
- Functional Safety-Compliant
  - Developed for functional safety applications
  - Documentation to aid ISO 26262 system design
  - Systematic capability up to ASIL D
  - Hardware capability up to ASIL D
- Differential voltage measurements with integrated filtering
  - Pack+ , Fuse, Link±, Charge± measure
  - Isolation resistance voltage measurement
- Integrated precision current measurement
  - Support low-side shunt resistor
- Insulation resistance monitoring capability
- 8 GPIO inputs as IO, SPI, ADC, and temperature sense
- Isolated differential daisy-chain communication with optional ring architecture
  - Simplifies BJB/BDU system by eliminating safety MCU, CAN transceivers, and multi-wire interface
- UART host interface
- Synchronized current and voltage measurements
- Stackable as well as syncs with other cell and UI monitors 16S (BQ79616-Q1, BQ79656-Q1), 14S (BQ79614-Q1, BQ79654-Q1), 12S (BQ79612-Q1, BQ79652-Q1) and BQ79631
- Built-in SPI master

2 Applications

- Full Electric, Plug-In Hybrid, and Hybrid Vehicles

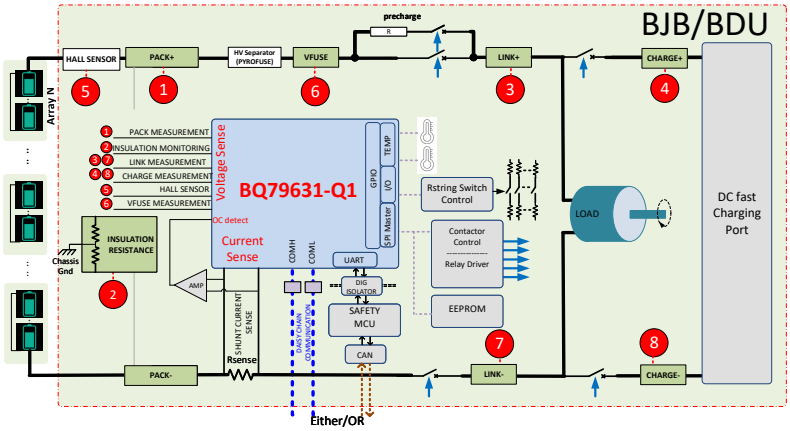
3 Description

BQ79631-Q1 provides high-accuracy measurement capable of measuring divided down voltages from high voltage nodes such as Pack+ (HV Battery Positive terminal), Fuse, Charge (Port), Link (Load) in a battery junction box (BJB) or battery disconnect unit (BDU) system. Key voltage measurements can make use of the integrated digital low-pass filters. The device has a highly accurate integrated current sense ADC capable of measuring current in low-side shunt resistor. The device is capable of measuring the insulation resistance with internal ADC and able to control any switching scheme needed for this measurement. There are eight GPIOs/auxillary inputs that can be used for thermistor measurement, driving relays, measuring voltages, and being a master SPI interface to peripheral SPI devices. The isolated bi-directional daisy-chain ports support both capacitor or transformer based isolation. The device also supports communication over UART.

Device Information

PART NUMBER	PACKAGE(1)	BODY SIZE (NOM)
BQ79631-Q1	HTQFP (64-pin)	10.00 mm x 10.00 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.



Simplified System Diagram

Table of Contents

1 Features....1
2 Applications....1
3 Description....1
4 Revision History....2
5 Description (continued)....3
6 Pin Configuration and Functions....4
7 Specifications....7
7.1 Absolute Maximum Ratings....7
7.2 ESD Ratings....7
7.3 Recommended Operating Conditions....8
7.4 Thermal Information....8
7.5 Electrical Characteristics....8
7.6 Timing Requirements....13
7.7 Typical Characteristics....17
8 Detailed Description....18
8.1 Overview....18
8.2 Functional Block Diagram....19
8.3 Feature Description....20
8.4 Device Functional Modes....82
8.5 Register Maps....91
9 Application and Implementation....160
9.1 Application Information....160
9.2 Typical Applications....160
10 Power Supply Recommendations....178
11 Layout....179
11.1 Layout Guidelines....179
11.2 Layout Example....181
12 Device and Documentation Support....185
12.1 Device Support....185
12.2 Receiving Notification of Documentation Updates....185
12.3 Support Resources....185
12.4 Trademarks....185
12.5 Electrostatic Discharge Caution....185
12.6 Glossary....185
13 Mechanical, Packaging, and Orderable Information....186

4 Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

Changes from Revision * (December 2021) to Revision A (November 2023)	Page
• Changed data sheet status from Restricted to Public	1

5 Description (continued)

Host communication to the BQ79631-Q1 can be connected via the device's dedicated UART interface or through a communication bridge device, BQ79600-Q1. Additionally, an isolated, differential daisy-chain communication interface allows the host to communicate with the other UIR monitors and even cell monitors over a single interface. In the event of a communication line break, the daisy-chain communication interface is configurable to a ring architecture that allows the host to talk to devices at either end of the stack.

6 Pin Configuration and Functions

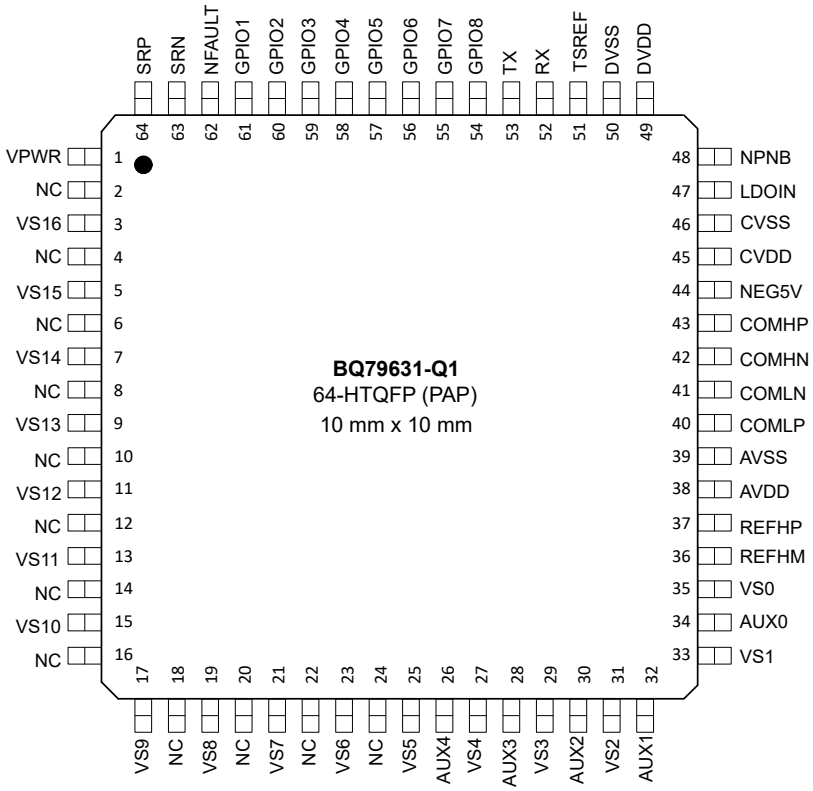


Figure 6-1. PAP Package 64-Pin HTQFP Top View

Table 6-1. Pin Functions

PIN		TYPE	DESCRIPTION
NAME	NO.		
VPWR	1	P	Power supply input
NPNB	48	P	Connect to the base of an external NPN transistor.
LDOIN	47	P	6-V preregulated analog power supply input/sense pin. Connect to the emitter of the external NPN transistor and connect a 0.1-µF decoupling capacitor to AVSS.
AVDD	38	P	5-V regulated output. AVDD supplies the internal analog circuits. Bypass AVDD with a capacitor to AVSS.
AVSS	39	GND	Analog ground. Ground connection for internal analog circuits. Connect DVSS, CVSS, REFHM, and AVSS externally. All ground pins must not be left unconnected.
NEG5V	44	P	Negative 5-V charge pump used for daisy-chain and Main ADC. Connect with a capacitor to AVSS.
DVDD	49	P	1.8-V regulated output. DVDD supplies the internal digital circuits. Bypass DVDD with a capacitor to DVSS.
DVSS	50	GND	Digital ground. Ground connection for internal digital logics. Connect DVSS, CVSS, REFHM, and AVSS externally. All ground pins must not be left unconnected.
CVDD	45	P	5-V daisy-chain communication and I/Os power supply. CVDD supplies the stack daisy-chain communication transceiver circuit and the I/O pins. This power supply also supports an additional 10-mA external load in ACTIVE and SLEEP modes. Bypass CVDD with a capacitor to CVSS.
CVSS	46	GND	Daisy-chain communication ground. Ground connection for internal daisy-chain transceivers. Connect DVSS, CVSS, REFHM, and AVSS externally. All ground pins must not be left unconnected.
TSREF	51	P	5-V bias voltage for thermistor. Connect TSREF to the top of the thermistor resistor divider network to the GPIOs when they are configured for thermistor temperature monitoring. Bypass TSREF with a capacitor to AVSS.
REFHP	37	P	Precision reference output pin. Bypass with a capacitor to REFHM.
REFHM	36	GND	Precision reference ground. Ground connection for the internal precision reference. Connect DVSS, CVSS, REFHM, and AVSS externally. All ground pins must not be left unconnected.
VS16	3	I	Differential voltage sense input.
VS15	5	I	Differential voltage sense input.

Table 6-1. Pin Functions (continued)

PIN		TYPE	DESCRIPTION
NAME	NO.		
VS14	7	I	Differential voltage sense input.
VS13	9	I	Differential voltage sense input.
VS12	11	I	Differential voltage sense input.
VS11	13	I	Differential voltage sense input.
VS10	15	I	Differential voltage sense input.
VS9	17	I	Differential voltage sense input.
VS8	19	I	Differential voltage sense input.
VS7	21	I	Differential voltage sense input.
VS6	23	I	Differential voltage sense input.
VS5	25	I	Differential voltage sense input.
VS4	27	I	Differential voltage sense input.
VS3	29	I	Differential voltage sense input.
VS2	31	I	Differential voltage sense input.
VS1	33	I	Differential voltage sense input.
VS0	35	I	Differential voltage sense input.
NC	2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24	NC	Unused pin, must be tied to VPWR.
SRP	64	I	Current sense input With SRP and SRN connecting across a low-side shunt resistor, the differential SRP-SRN is measured by current sense ADC.
SRN	63	I	Current sense input With SRP and SRN connecting across a low-side shunt resistor, the differential SRP-SRN is measured by current sense ADC.
RX	52	I	UART receiver input. Pullup to CVDD with an external resistor and connect the device RX to the TX output of the host MCU. If unused (for example for stack devices), connect RX to CVDD.
TX	53	O	UART transmitter output. Connect device TX to RX input of the host MCU and shall be pulled up from the host side. If unused (for example for stack devices), leave it floating.
COMHP	43	I/O	Vertical bi-directional communication interface for daisy-chain connection. High-side (north side) differential I/O. Shall connect to the low-side (south side) COMLP and COMLN of the lower adjacent device in the daisy-chain configuration. If unused, connect COMHP and COMHN with a 1-kΩ resistor.
COMHN	42	I/O	
COMLP	40	I/O	Vertical bi-directional communication interface for daisy-chain connection. Low-side (south side) differential I/O. Shall connect to the high-side (north side) COMHP and COMHN of the upper adjacent device in the daisy-chain configuration. If unused, connect COMHP and COMHN with a 1-kΩ resistor.
COMLN	41	I/O	
NFAULT	62	O	Fault indication output. Active low. If used on the base device, pull up NFAULT to CVDD with a pullup resistor and connect NFAULT to host MCU GPIO. If unused, leave it unconnected.
GPIO1	61	I/O	General purpose input/output, configuration options are: • For external thermistor connection, connect thermistor to the pin and pull up to TSREF. Used as input to ADC. • For external DC voltage measurement, configured as input to ADC. • Generic digital input/output. • Use as I/O for SPI master.
GPIO2	60	I/O	
GPIO3	59	I/O	
GPIO4	58	I/O	
GPIO5	57	I/O	
GPIO6	56	I/O	
GPIO7	55	I/O	
GPIO8	54	I/O	
AUX0	34	I	Differential voltage sense input.
AUX1	32	I	Differential voltage sense input.
AUX2	30	I	Differential voltage sense input.
AUX3	28	I	Differential voltage sense input.

Table 6-1. Pin Functions (continued)

PIN		TYPE	DESCRIPTION
NAME	NO.		
AUX4	26	I	Differential voltage sense input.

7 Specifications

7.1 Absolute Maximum Ratings

over operating free-air temperature range (unless otherwise noted)(1)

		MIN	MAX	UNIT
Input Voltage	VPWR, VS* (except VS0), AUX* (except AUX0), NFAULT, to AVSS (2) (3)	–0.3	24	V
	AUX0, VS0 to AVSS	–0.3	5.5	V
	VSn to VSn-1, n = 1 to 16 (2)	–24	24	V
	AUXn to AUXn-1, n = 1 to 4 (3)	–0.3	16	V
	SRP , SRN to AVSS (4)	-0.3	2.1	V
	SRP to SRN (4)	-1.8	1.8	V
	LDOIN to AVSS	–0.3	9	V
	NPNB to AVSS	–0.3	10	V
	AVDD to AVSS	–0.3	5.5	V
	DVDD to DVSS	–0.3	1.98	V
	CVDD to CVSS	–0.3	6	V
	TSREF to AVSS	–0.3	5.5	V
	REFHP to REFHM	–0.3	5.5	V
	NEG5V to AVSS	–5.5	0	V
	TX, RX to AVSS	–0.3	6	V
	COMHP, COMHN, COMLP, COMLN to CVSS	–20	20	V
	COMHP to COMHN, COMLP to COMLN	–5.5	5.5	V
	GPIO* to AVSS	–0.3	5.5	V
I/O current	GPIO*, RX, TX current		10	mA
T_{OTP_PROG}	Device will not start OTP programming above this temperature		55	°C
T_A	Ambient temperature	–40	130	°C
T_J	Junction temperature	–40	150	°C
T_{stg}	Storage temperature	–65	150	°C

(1) Stresses beyond those listed under *Absolute Maximum Rating* may cause permanent damage to the device. These are stress ratings only, which do not imply functional operation of the device at these or any other conditions beyond those indicated under *Recommended Operating Condition*. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.
(2) VS pin voltage has to meet criteria of both VSn to AVSS as well as VSn to VSn-1.
(3) AUX pin voltage has to meet criteria of both AUXn to AVSS as well as AUXn to AUXn-1.
(4) SRP, SRN pin voltage has to meet criteria of both voltage to AVSS as well as SRP to SRN

7.2 ESD Ratings

				VALUE	UNIT
$V_{(ESD)}$	Electrostatic discharge	Human body model (HBM), per ANSI/ESDA/JEDEC JS-001(1)		±2000	V
		Charged device model (CDM), per ANSI/ESDA/JEDEC JS-002 (2)	Corner pins (1,16 ,17, 32, 33, 48, 49, and 64)	±750	
			Other pins	±500	

(1) JEDEC document JEP155 states that 500-V HBM allows safe manufacturing with a standard ESD control process
(2) JEDEC document JEP157 states that 250-V CDM allows safe manufacturing with a standard ESD control process

7.3 Recommended Operating Conditions

over operating free-air temperature range (unless otherwise noted)

		MIN	NOM	MAX	UNIT
V_{VPWR_RANGE}	Total module voltage, full functionality, no OTP programming	9		15	V
$V_{VPWR\_OTP\_RANGE}$	Total module voltage, full functionality, OTP programming allow	11		15	V
V_{INPUT_RANGE}	VS_n - VS_{n-1}, where n = 2 to 16	–1		5	V
	VS0 to AVSS	-0.3		0.3	V
	VS1, VS2 to AVSS	-0.3		5	V
	VSn, to AVSS, where n = 3 to 16	3		5	V
V_{AUX_RANGE}	AUX_n - AUX_{n-1}, where n = 1 to 4	0		5	V
	AUX0 to AVSS	-0.3		0.3	V
	AUX1, AUX2 to AVSS	-0.3		5	V
	AUX3, AUX4 to AVSS	3		5	V
V_{CS_RANGE}	Current sense range, V_{SRP} - V_{SRN}	-100		100	mV
V_{IO_RANGE}	RX, TX, NFAULT	0		CVDD	V
V_{GPIO_RANGE}	$GPIO_n$ input, where n = 1 to 8	0.2		4.8	V
I_{IO}	$GPIO_n$, RX, TX, where n = 1 to 8			5	mA
T_A	Operation temperature	–40		125	°C

7.4 Thermal Information

THERMAL METRIC		BQ79631-Q1 PAP (HTQFP) 64 PINS	UNIT
$R_{\theta JA}$	Junction-to-ambient thermal resistance	21.6	°C/W
$R_{\theta JC(top)}$	Junction-to-case (top) thermal resistance	8.7	°C/W
$R_{\theta JB}$	Junction-to-board thermal resistance	7.9	°C/W
ψ_{JT}	Junction-to-top characterization parameter	0.1	°C/W
ψ_{JB}	Junction-to-board characterization parameter	7.8	°C/W
$R_{\theta JC(bot)}$	Junction-to-case (bottom) thermal resistance	2.1	°C/W

7.5 Electrical Characteristics

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
THERMAL SHUTDOWN						
T_{SHUT}	Thermal shutdown (rising direction)		130	137	152	°C
T_{SHUT_FALL}	Thermal shutdown (falling direction)		112		129	°C
T_{SHUT_HYS}	Thermal shutdown (rising - falling direction)			20		°C
T_{WARN_RANGE}	Thermal warning Threshold (rising direction)		85		115	°C
T_{WARN_HYS}	Thermal warning hysteresis (falling direction)			10		°C
T_{WARN_ACC}	Thermal warning accuracy (+/-)			5		°C
SUPPLY CURRENTS						

7.5 Electrical Characteristics (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
I_{SHDN}	Supply current in SHUTDOWN mode	Sum of both I_{VPWR} and I_{LDOIN}		16	23	µA
$I_{SLP(IDLE)}$	Baseline supply current in SLEEP mode. No fault, no protector comparator	Sum of both I_{VPWR} and I_{LDOIN} T_A = -20℃ to 65℃		120	160	µA
		Sum of both I_{VPWR} and I_{LDOIN} T_A = -40℃ to 125℃			220	µA
$I_{ACT(IDLE)}$	Baseline supply current in ACTIVE mode	Sum of both I_{VPWR} and I_{LDOIN} No fault, no communication, no protector comparator		10.4	11.6	mA
$I_{PROTCOMP}$	Additional supply current when protector comparator is on	Either OV/UV protector is enabled. Other functions are inactive		20	60	µA
I_{ADC}	Additional supply current when CS/ main and aux ADC are enabled	Both CS/Main ADC are on, in continiously mode, Other functions are inactive		1.8	2.4	mA
I_{ADC}	Additional supply current when ADC is enabled	Main or Aux ADC on, and conversion is in progress. Other functions are inactive		0.4	0.6	mA
		2 ADCs on, and conversion is in progress. Other functions are inactive (not applicable if current sense ADC is availbe in this device)		0.6	0.9	mA
I_{VPWR}	Supply current goes into VPWR pin	ACTIVE Mode		150		µA
		SLEEP Mode		25		µA
		SHUTDOWN Mode		5		µA
I_{COMT}	Additional supply current during daisy-chain broadcast read of 128-byte data	Use transformer isolation for daisy-chain interface		10		mA
I_{COMC}	Additional supply current during daisy-chain broadcast read of 128-byte data	Use capacitor or capacitor and choke isolation for daisy-chain interface		10		mA
I_{OW_SINK}	Sink current for open wire test, applies to VS1 to VS16 and AUX1 to AUX 4		380	500	600	µA
I_{OW_SOURCE}	Source current for open wire test, applies to VS0 and AUX0		380	500	600	µA
I_{LEAK_CS}	Leakage current SRP and SRN pin	Main and CS ADC is off			0.2	µA
I_{LEAK}	Leakage current on VS, AUX pins	VS, AUX pins with ADC off.			0.1	µA
V_{SR_OW}	Clamped voltage when I_{OW_SORUCE} is enabled for SRP and SRN				0.9	V
Supplies (LDOIN)						
V_{LDOIN}	LDOIN voltage	No OTP programming	5.9	6	6.1	V
		OTP programming	7.9	8	8.1	V
Supplies (CVDD)						
V_{CVDD}	CVDD output voltage	ACTIVE and SLEEP mode	4.9	5	5.1	V
		SHUTDOWN mode, no external Iload	3.95		6	V
		SHUTDOWN mode, max external Iload = 5mA	3.4		5.5	V
V_{CVDD_LDRG}	CVDD load regulation	ACTIVE/SLEEP mode, max external Iload = 10mA	–30		30	mV
V_{CVDD_OV}	CVDD OV threshold	ACTIVE/SLEEP mode, max external Iload = 10mA	5.3	5.5	5.7	V
V_{CVDD_OVHYS}	CVDD OV Hystersis	ACTIVE/SLEEP mode, max external Iload = 10mA	130	150	170	mV

7.5 Electrical Characteristics (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{CVDD_UV}	CVDD UV threshold	SHUTDOWN mode		3.5		V
		ACTIVE/SLEEP mode, max external Iload = 10mA	4.3	4.45	4.65	V
V_{CVDD_UVHYS}	CVDD UV Hystersis			260		mV
V_{CVDD_ILIMIT}	CVDD current limit	ACTIVE, SLEEP	35	60	85	mA
Supplies (AVDD)						
V_{AVDD}	AVDD output voltage	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	4.85	5	5.21	V
V_{AVDD_OV}	AVDD OV threshold	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	5.25	5.5	5.7	V
V_{AVDD_OVHYS}	AVDD OV Hystersis	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	135	155	165	mV
V_{AVDD_UV}	AVDD UV threshold	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	4.25	4.45	4.6	V
V_{AVDD_UVHYS}	AVDD UV Hystersis	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	235	340	430	mV
V_{AVDD_ILIMIT}	AVDD current limit	$C_{SUPPLIES}$ = 1µF	10	30	50	mA
Supplies (DVDD)						
V_{DVDD}	DVDD output voltage	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	1.65	1.8	1.95	V
V_{DVDD_OV}	DVDD OV threshold	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	1.95	2.1	2.3	V
V_{DVDD_OVHYS}	DVDD OV Hystersis	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	40	65	120	mV
V_{DVDD_UV}	DVDD UV threshold	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	1.623	1.65	1.71	V
V_{DVDD_UVHYS}	DVDD UV Hystersis	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	15	50	73	mV
V_{DVDD_ILIMIT}	DVDD current limit		13	30	53	mA
Supplies (TSREF)						
V_{TSREF}	TSREF output voltage	$C_{SUPPLIES}$ = 1µF, ACTIVE mode	4.975	5	5.025	V
V_{TSREF_LDRG}	TSREF load regulation	I_{load} = 4mA, $C_{SUPPLIES}$ = 1µF, ACTIVE mode	–30		30	mV
V_{TSREF_OV}	TSREF OV threshold	I_{load} = 4mA, $C_{SUPPLIES}$ = 1µF, ACTIVE mode	5.2	5.6	5.8	V
V_{TSREF_OVHYS}	TSREF OV Hystersis	I_{load} = 4mA, $C_{SUPPLIES}$ = 1µF, ACTIVE mode	98	110	120	mV
V_{TSREF_UV}	TSREF UV threshold	I_{load} = 4mA, $C_{SUPPLIES}$ = 1µF, ACTIVE mode	4.0	4.2	4.4	V
V_{TSREF_UVHYS}	TSREF UV Hystersis	I_{load} = 4mA, $C_{SUPPLIES}$ = 1µF, ACTIVE mode	300	350	400	mV
V_{TSREF_ILIMIT}	TSREF current limit	Device in ACTIVE Mode	15	30	52	mA
Negative Charge Pump (NEG5V)						
V_{NEG5V}	NEG5V pin voltage	C_{NEG5V} = 0.1µF	-5.3	-4.6	-4.0	V
V_{NEG5V_UV}	NEG5V UV threshold (rising)	C_{NEG5V} = 0.1µF	-4.1	-3.5	-3.0	V
$V_{NEG5V\_UVRECOV}$	NEG5V UV Recovery	C_{NEG5V} = 0.1µF	-4.3	-3.8	-3.3	V
ADC Resolution						
$ENOB_{MAIN}$	Main ADC Effective number of bits			16		bits
$ENOB_{AUX}$	AUX ADC Effective number of bits			14		bits
V_{LSB_ADC}	Main and AUX ADC Resolution for input voltage measurement			190.73		µV/LSB
$V_{LSB\_CSADC\_AUX}$	AUX CSADC Resolution for (SRP-SRN) measurement			30.52		µV/LSB
$V_{LSB\_MAIN\_DIETEMP1}$	DieTemp1 resolution (Main ADC)	ADC measurement is centered with 0x000 = 0ºC		0.025		°C/LSB

7.5 Electrical Characteristics (continued)

over operating -40°C to 125°C free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$V_{LSB\_AUX\_DIETEMP2}$	DieTemp2 resolution (AUX ADC)	ADC measurement is centered with 0x000 = 0ºC		0.025		°C/LSB
$V_{LSB\_AUX\_VPWR}$	VPWR resolution (AUX ADC)	Applies to VPWR voltage measurement from AUX ADC		3.05		mV/LSB
V_{LSB_GPIO}	GPIO resolution (Main & AUX ADC)			152.59		µV/LSB
V_{LSB_TSREF}	TSREF resolution (Main ADC)			169.54		µV/LSB
V_{LSB_DIAG}	Diagnostic measurements resolution	REFL, VBG2, LPBG5, VCM, AVAO_REF, AVDD_REF, all the HW protector DAC		152.59		µV/LSB
V_{LSB_CS}	Current Sense ADC resolution (24-bit result)	Reading CURRENT_HI/MID/LO registers		14.9		nV/LSB
ADC Accuracy						
I_{VS_DELTA}	VSn to VSn-1 input current delta (when Main ADC is on)	T_A = -20ºC to 65ºC			1.8	µA
		T_A = -40ºC to 105ºC			2	µA
I_{VS}	VSn input current (when Main ADC is on)				12	µA
R_{AUX_INPUT}	AUX pin input impedance (when AUX ADC is on)			16		MΩ
V_{ACC_MAIN}	Total channel accuracy for main ADC VS measurement, LPF_VS[2:0] = 0x03 setting;	2V<V_{IN}<4.5V; T_A=25ºC	-2.2		1.5	mV
		2V<V_{IN}<4.5V; -20ºC<T_A<65ºC	-3.0		2.4	mV
		2V<V_{IN}<4.5V; -40ºC<T_A<105ºC	-3.5		2.6	mV
		2V<V_{IN}<4.5V; -40ºC<T_A<125ºC	-3.5		2.6	mV
		1V<V_{IN}< 5V; -40ºC<T_A<125ºC	-3.7		2.8	mV
		-2V<V_{IN}< 5V; -40ºC<T_A<125ºC	-4.5		3.2	mV
V_{ACC_AUX}	Total channel accuracy for AUX ADC measurement (excluding VPWR and GPIO accuracy);	2V<V_{IN}<4.5V; T_A=25ºC	-7.5		5.4	mV
		2V<V_{IN}<4.5V; -20ºC<T_A<65ºC	-8.0		6.3	mV
		2V<V_{IN}<4.5V; -40ºC<T_A<105ºC	-9.0		6.3	mV
		2V<V_{IN}<4.5V; -40ºC<T_A<125ºC	-9.0		6.5	mV
		1V<V_{IN}< 5V; -40ºC<T_A<125ºC	-9.0		6.6	mV
		0V<V_{IN}< 5V; -40ºC<T_A<125ºC	-9.0		6.6	mV
$V_{(MAIN-AUX)}$	Main - AUX measurement during VIN diagnostic. Same input voltage to both ADC under same T_A;	2V<V_{IN}<4.5V; T_A=25ºC	-7.1		6.1	mV
		2V<V_{IN}<4.5V; -20ºC<T_A<65ºC	-7.8		6.6	mV
		2V<V_{IN}<4.5V; -40ºC<T_A<105ºC	-7.8		6.6	mV
		2V<V_{IN}<4.5V; -40ºC<T_A<125ºC	-7.8		6.7	mV
		1V<V_{IN}< 5V; -40ºC<T_A<125ºC	-7.9		6.9	mV
		0V<V_{IN}< 5V; -40ºC<T_A<125ºC	-7.9		6.9	mV
$V_{ACC\_MAIN\_GPIO\_RATIO}$	Measured GPIO from Main ADC/ measured TSREF from Main ADC;	0.08V<V_{IN}<0.2V, 85ºC<T_A<125ºC	-0.20		0.20	%
		0.2V<V_{IN}<4.6V, -40ºC<T_A<105ºC	-0.20		0.20	%
		4.6V<V_{IN}<4.8V, -40ºC<T_A<-20ºC	-0.30		0.30	%
$V_{ACC\_AUX\_GPIO\_RATIO}$	Measured GPIO from AUX ADC/ measured TSREF from AUX ADC;	0.08V<V_{IN}<0.2V, 85ºC<T_A<125ºC	-0.20		0.20	%
		0.2V<V_{IN}<4.6V, -40ºC<T_A<105ºC	-0.20		0.20	%
		4.6V<V_{IN}<4.8V, -40ºC<T_A<-20ºC	-0.30		0.30	%
$V_{ACC\_MAIN\_GPIO\_ABS}$	Total channel accuracy for GPIO measurement (Main ADC);	0.08V<V_{IN}<0.2V, 85ºC<T_A<125ºC	-4.00		4.00	mV
		0.2V<V_{IN}<4.6V, -40ºC<T_A<105ºC	-5.00		3.00	mV
		4.6V<V_{IN}<4.8V, -40ºC<T_A<-20ºC	-4.00		4.00	mV

7.5 Electrical Characteristics (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$V_{ACC\_AUX\_GPIO\_ABS}$	Accuracy from AUX ADC on GPIO	0.08V<V_{IN}<0.2V, 85ºC<T_A<125ºC	-6.00		6.00	mV
		0.2V<V_{IN}<4.6V, -40ºC<T_A<105ºC	-6.00		6.00	mV
		4.6V<V_{IN}<4.8V, -40ºC<T_A<-20ºC	-6.00		6.00	mV
$V_{ACC\_MAIN\_CS}$	Total channel accuracy for (SRP-SRN) from CSADC AUX measurement	LPF_SR[2:0] = 0x00	-1.1		1.1	mV
$V_{ACC\_AUX\_REFL}$	AUX ADC measurement result		1.092	1.1	1.106	V
$V_{ACC\_AUX\_VBG2}$	AUX ADC measurement result		1.092	1.1	1.106	V
$V_{ACC\_AUX\_VCM}$	AUX ADC measurement result		2.400	2.5	2.550	V
$V_{ACC\_AUX\_AVAO\_REF}$	AUX ADC measurement result		2.400	2.47	2.550	V
$V_{ACC\_AUX\_AVDD\_REF}$	AUX ADC measurement result		2.400	2.47	2.550	V
$V_{ACC\_AUX\_OVDAC}$	AUX ADC measurement result	Setting at 4.475V; T_A = -20ºC to 65ºC	4.450		4.500	V
$V_{ACC\_AUX\_OVDAC}$	AUX ADC measurement result	Setting at 4.475V; T_A = -40ºC to 105ºC	4.445		4.500	V
$V_{ACC\_AUX\_OVDAC}$	AUX ADC measurement result	Setting at 4.475V; T_A = -40ºC to 125ºC	4.445		4.500	V
$V_{ACC\_AUX\_OVDAC}$	AUX ADC measurement result	Setting at 3.8V	3.770		3.825	V
$V_{ACC\_AUX\_OVDAC}$	AUX ADC measurement result	Setting at 3V	2.970		3.030	V
$V_{ACC\_AUX\_UVDAC}$	AUX ADC measurement result	Setting at 3.1V	3.095	3.1	3.150	V
$V_{ACC\_MAIN\_TSREF}$	Main ADC measurement result		4.975	5	5.025	V
$V_{ACC\_MAIN\_DIETEMP}$	Total channel accuracy for Die Temp1 measurement (+/-)			3		℃
$V_{ACC\_AUX\_DIETEMP}$	Total channel accuracy for Die Temp2 measurement (+/-)			6		℃
$I_{SRP\_N\_Diff}$	Differential SRN/SRP input current (CS and main ADC are on)	Apply 100mV differential acrsoss SRP/SRN		1.4		µA
V_{RANGE_CS}	Effective input range of CS ADC		-100		100	mV
V_{NOISE_CS}	CS ADC input referred noise	CS_DS[1:0] = 11, CS ADC in continious mode, short SRP/SRN at pins		0.71		uV_{RMS}
Gain_error_cs_room_uncal	Gain error of CS ADC @25ºC, it could be single temp piont calibrated out	T_A = 25ºC, CS_DS[1:0] = 01, measured at -75mV and 75mV	-0.6		0.6	%
Gain_error_cs_drift1	Gain error of CS ADC drift over temperature, |V_{RANGE_CS}| <100mV	T_A = -20ºC to 85ºC, CS_DS[1:0] = 01, measured at 50mV and 75mV			0.3	%
		T_A = -40ºC to 105ºC, CS_DS[1:0] = 01, measured at 50mV and 75mV			0.3	%
Offset_cs_room_uncal	Input referred offset error of CS ADC @ 25ºC, it could be single temp piont calibrated out	T_A = 25ºC, CS_DS[1:0] = 01, short SRP/SRN at pins	-6		6	µV
Offset_cs_drift	Input referred offset error drift over temperature	T_A = -40ºC to -20ºC, CS_DS[1:0] = 01, short SRP/SRN at pins	-2.5		2.5	µV
		T_A = -20ºC to 105ºC, CS_DS[1:0] = 01, short SRP/SRN at pins	-1.8		1.8	µV
Reference Voltages						
V_{REFH}	REFHP to REFHM voltage		4.975	5	5.025	V
HW Voltage Comparator/Protector (OV/UV)						
$V_{OV\_COMP\_RANGE}$	OV comparator detection threshold setting range (not accuracy)	Step of 25mV	2700		3000	mV
		Step of 25mV	3600		3800	mV
		Step of 25mV	4175		4500	mV

7.5 Electrical Characteristics (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
$V_{OV\_COMP\_HYS}$	OV comparator hysteresis after detection			50		mV
$V_{OV\_COMP\_ACC}$	OV comparator accuracy	T_A = -20ºC to 65ºC	-24		24	mV
		T_A = -40ºC to 105ºC	-28		28	mV
$V_{UV\_COMP\_RANGE}$	UV comparator detection threshold setting range (not accuracy)	Step of 50mV	1200		3100	mV
$V_{UV\_COMP\_HYS}$	UV comparator hysteresis after detection			50		mV
$V_{UV\_COMP\_ACC}$	UV comparator accuracy	T_A = -20ºC to 65ºC	-35		35	mV
		T_A = -40ºC to 105ºC	-50		50	mV
Digital I/Os (TX, RX, GPIO, SPI master)						
V_{OH}	Output as logic level high (TX, GPIO as output)	GPIO is configured as output. I_{OUT} = 1mA	V_{CVDD}-0.3			V
V_{OL}	Output as logic level low (TX, NFAULT, GPIO as output)	GPIO is configured as output. I_{OUT} = 1mA			0.3	V
V_{IH}	Input as logic level high (RX, GPIO as fault input)	GPIO is configured as input. I_{OUT} = 1mA	0.75 x V_{CVDD}			V
V_{IL}	Input as logic level low (RX, GPIO as fault input)	GPIO is configured as input. I_{OUT} = 1mA			0.25 x V_{CVDD}	V
R_{WK_PU}	GPIO weak pull-up resistance		20	37	60	KΩ
R_{WK_PD}	GPIO weak pull-down resistance		20	40	60	KΩ
COML and COMH						
R_{DCTX}	Transmitter output impedance (COML and COMH)			18		Ω
R_{DCCM}	Common mode impedance (COML and COMH)			45		kΩ
V_{DCCM}	Common mode voltage (COML and COMH)		2.21	2.5	2.76	V
V_{COMM_DATA1}	Receiver threshold range (V_{COMP}-V_{COML}) form communication	CODE:0	0.4		1.2	V
V_{COMM_TONE1}	Receiver threshold range (V_{COMP}-V_{COML}) form Tone	CODE:0	0.4		1.2	V

7.6 Timing Requirements

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
POWER STATE TIMING						
$t_{SU(WAKE\_SHUT)}$	Startup from SHUTDOWN to ACTIVE mode	Base device: From the end of WAKE ping to the start of a forwading WAKE tone		6	10	ms
		Stack device: From the end of a recevied WAKE tone to the start of a forwading WAKE tone		6	10	ms
$t_{SU(SLP2ACT)}$	Startup from SLEEP to ACTIVE mode (with SLEEP2ACTIVE ping/tone)	Base device: From the end of SLEEP2ACTIVE ping to the start of the forwarding SLEEP2ACTIVE tone			230	µs
		Stack device: From the end of SLEEP2ACTIVE tone to the start of the forwarding SLEEP2ACTIVE tone			230	µs

7.6 Timing Requirements (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
$t_{SU(WAKE\_SLP)}$	Startup from SLEEP to ACTIVE mode (with WAKE ping/tone)	Base device: From the end of WAKE ping to the start of a forwading WAKE tone			1	ms
		Stack device: From the end of a recevied WAKE tone to the start of a forwading WAKE tone			1	ms
t_{SLP}	From ACTIVE to SLEEP mode	From receiving SLEEP entry condition to enter in SLEEP mode			100	µs
t_{SHTDN}	From ACTIVE to SLHUTDOWN mode	From receiving SHUTDOWN entry condition to enter in SHUTDOWN mode (all LDOs in 10% of their norminal value)		20		ms
t_{RST}	Reset time during ACTIVE mode	CONTROL1[SOFT_RST] = 1 is sent to a completion of the digital reset			1	ms
t_{HWRST}	The time device will be in HW reset, after HW reset ping/tone issued				75	ms
SUPPLIES TIMING						
t_{TSREF_ON}	TSREF ramp up time (10%-90%)	C_{TSREF} = 1µF	6			ms
t_{TSREF_OFF}	TSREF ramp down time (90%-10%)	C_{TSREF} = 1µF			8	ms
PING SIGNAL TIMING						
t_{HLD_WAKE}	WAKE ping low time on RX pin; no external load on CVDD		2		2.5	ms
t_{HLD_SD}	SHUTDOWN ping low time on RX pin; no external load on CVDD		7		10	ms
$t_{UART(StA)}$	SLEEPtoACTIVE ping low time on RX pin		250		300	µs
t_{HLD_HWRST}	HW_RESET ping low time on RX pin		36			ms
COML and COMH (PULSE and TONE TIMING)						
t_{PW_DC}	COMM: Pulse width of data (half bit time) for communiction			250		ns
t_{RECLK_DC}	COMM: data reclocking delay per device from COMH to COML or viceversa			4	5	µs
$t_{COMTONE}$	Time between pulses of comm tones (HFO based). Comm Tones are WAKE, SLEEPtoACTIVE, SHUTDOWN, HWRST tones			11	15	µs
$t_{COMMTONE\_HI}$	The HIGH time of each comms pulse (HFO base)		0.92	1	1.08	µs
$t_{COMMTONE\_LO}$	The LOW time of each comms pulse (HFO base)		0.92	1	1.08	µs
$t_{FLTTONE}$	Time between pulses of FAULT Tone (LFO based). Applies to FAULT Tone and HEARTBEAT			11.5		µs
$t_{FLTTONE\_HI}$	The HIGH time of each pulse of the tone couplete			1		µs
$t_{FLTTONE\_LO}$	The LOW time of each pulse of the tone couplete			1		µs
$n_{WAKEDET}$	Number of pulses to detect as a WAKE tone			60		pulses
n_{WAKE}	Number of pulses to transit for a WAKE tone			90		pulses

7.6 Timing Requirements (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
$n_{SHDNDET}$	Number of pulses to detect as a SHUTDOWN tone			180		pulses
n_{SHDN}	Number of pulses to transit for a SHUTDOWN tone			270		pulses
$n_{SLPtoACTDET}$	Number of pulses to detect as a SLEEPtoACTIVE tone			20		pulses
$n_{SLPtoACT}$	Number of pulses to transit for a SLEEPtoACTIVE tone			30		pulses
$n_{HWRSTDET}$	Number of pulses to detect as a HW_RESET tone			540		pulses
n_{HWRST}	Number of pulses to transit for a HW_RESET tone			810		pulses
n_{HBDET}	HEARTBEAT: Number of pulses to detect as a valid tone			20		pulses
n_{HB}	HEARTBEAT: Number of pulses to transit for a tone			30		pulses
t_{HB_PERIOD}	HEARTBEAT: Period between HEARTBEAT Burst (from the beginning of a HEARTBEAT to the beginning of the next HEARTBEAT)		360	400	440	ms
$t_{HB\_TIMEOUT}$	HEARTBEAT: Timeout to considered as not receving HEARTBEAT		0.9	1	1.1	s
t_{HB_FAST}	HEARTBEAT: If HEARTBEAT is received within this time, it is considered receving HEARTBEAT too fast			200		ms
$n_{FTONEDET}$	FAULT TONE: Number of pulses to detect as a valid tone			60		pulses
n_{FTONE}	FAULT TONE: Number of pulses to transit for a tone			90		pulses
t_{FTONE_PERIOD}	FAULT TONE: Period between FAULT TONE Burst (from the beginning of a FAULT TONE to the beginning of the next FAULT TONE)			50		ms
$t_{FTS\_LATENCY}$	Fault Tone latency in Stack Device	From time a device receive the tone to the time the same device detects and generate its fault tone		48		µs
$t_{FTB\_LATENCY}$	Fault Tone latency in Base Device	From the time a device receive the tone to the time the same device detects and asserts NFAULT		24		µs
MAIN and AUX ADC TIMING						
t_{SAR_CONV}	Single conversion time (both Main and AUX ADCs)			8		µs
$t_{MAIN\_ADC\_CYCLE}$	Single round robin cycle (Main ADC)			192		µs
$t_{AUX\_ADC\_CYCLE}$	Single round robin cycle (AUX ADC)			192		µs
t_{AFE_SETTLE}	Analog front end (Level shifters) settling time whenever device enter ACTIVE mode from SLEEP or SHUTDOWN			4		ms
t_{CS_SETTLE}	CS ADC settling time			62		µs
$t_{CS\_REFRESH}$	Continious mode refresh rate	CS_DS[1:0] = 11		4.096		ms

7.6 Timing Requirements (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
$t_{CS\_REFRESH}$	Continious mode refresh rate	CS_DS[1:0] = 10		1.024		ms
$t_{CS\_REFRESH}$	Continious mode refresh rate	CS_DS[1:0] = 01		0.512		ms
$t_{CS\_REFRESH}$	Continious mode refresh rate	CS_DS[1:0] = 00		0.256		ms
t_{CS_CONV}	Single conversion time on CS ADC	CS_DS[1:0] = 11		12.350		ms
		CS_DS[1:0] = 10		3.134		ms
		CS_DS[1:0] = 01		1.598		ms
		CS_DS[1:0] = 00		0.83		ms
t_{ADC_ACC}	This includes mux round robin, ADC conversions, and digital filters.		-1.5		1.5	%
HW COMPARATORS/PROTECTORS TIMING						
t_{OV_CYCLE}	OV round robin cycle			8		ms
t_{UV_CYCLE}	UV round robin cycle			8		ms
$t_{OVUV\_BIST\_CYCLE}$	OV and UV BIST cycle		21.8	23	24.2	ms
$t_{PWR\_BIST\_CYCLE}$	Time needed for the power supply BIST to complete after the power BIST go command		10.9	11.5	12.1	ms
$t_{HW\_COMP\_ACC}$	OV,UV comparators timing accuracy		-5		5	%
I/O TIMING (TX, RX, GPIO, NFAULT)						
t_{RISE}	Rise Time	V_{CVDD} > MIN V_{CVDD}, C_{LOAD} = 150pF, GPIO in output mode		12		ns
t_{FALL}	Fall Time (exclude NFAULT)	V_{CVDD} > MIN V_{CVDD}, C_{LOAD} = 150pF, GPIO in output mode		7		ns
t_{FALL_NFAULT}	Fall Time on NFAULT	V_{CVDD} > MIN V_{CVDD}, C_{LOAD} = 150pF, R_{PULLUP} = 10kΩ		100		ns
UART TIMING						
$UART_{BAUD}$	UART TX/RX Baud Rate			1		Mbps
$UART_{ERR\_BAUD(RX)}$	UART RX baud rate error - requirement on the external host		-1		1	%
$UART_{ERR\_BAUD(TX)}$	UART TX baud rate error		-1.5		1.5	%
$t_{UART(CLR)}$	UART Clear low time		15		20	bit period
$t_{UART(RX\_HIGH)}$	After COMM CLEAR, wait this time before sending new frame		1			bit period
OTP NVM TIMING						
t_{CRC_CUST}	Time to complet a single cycle of CRC check on the customer OTP space			175		µs
t_{CRC_FACT}	Time to complet a single cycle of CRC check on the factory OTP space			1.6		ms
SPI MASTER TIMING						
f_{SCLK}	SCLK frequency		450	500	550	kHz
t_{HIGH}, t_{LOW}	SCLK duty cycle			50		%
$t_{SS(HIGH)}$	SS HIGH latency time. Time from register write high to SS pin high			4		µs
$t_{SS(LOW)}$	SS LOW latency time. Time from register write low to SS pin low			4		µs
$t_{SU(MISO)}$	MISO input data setup time - requirement for slave device	MISO stable before SCLK transition	100			ns
$t_{HD(MISO)}$	MISO input dat hold time	MISO stable after SCLK transition		0		ns

7.6 Timing Requirements (continued)

over operating -40℃ to 125℃ free-air temperature range, VVPWR = 9V to 15V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	NOM	MAX	UNIT
OSCILLATOR						
f_{HFO}	High frequency oscillator		31.52	32	32.48	MHz
f_{LFO}	Low frequency oscillator		248.9	262	275.1	kHz

7.7 Typical Characteristics

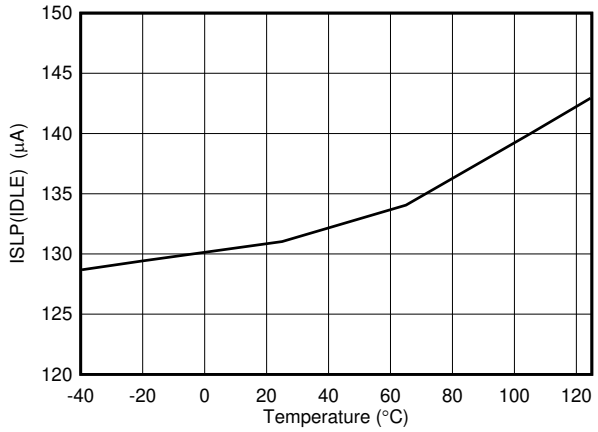


Figure 7-1. ISLP (IDLE) vs Temperature

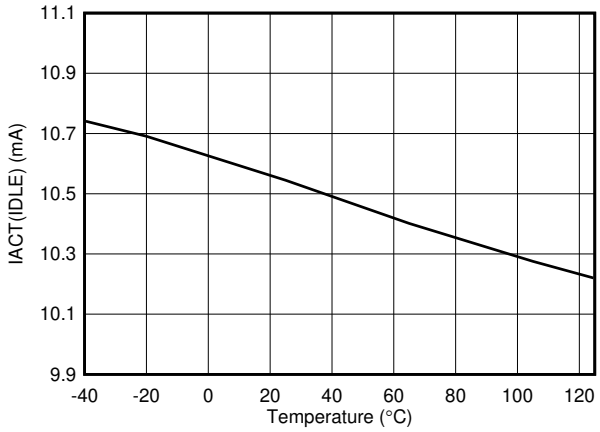


Figure 7-2. IACT (IDLE) vs Temperature

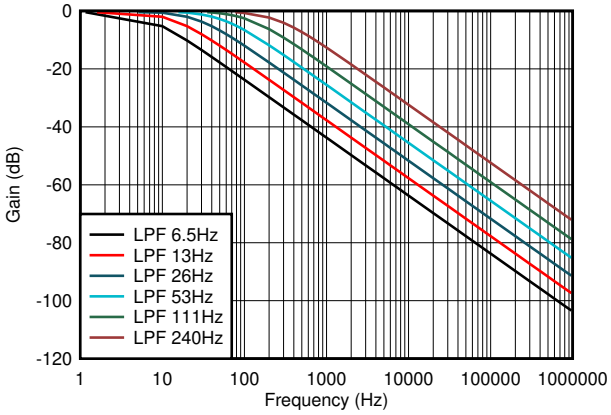


Figure 7-3. Digital Low Pass Filter

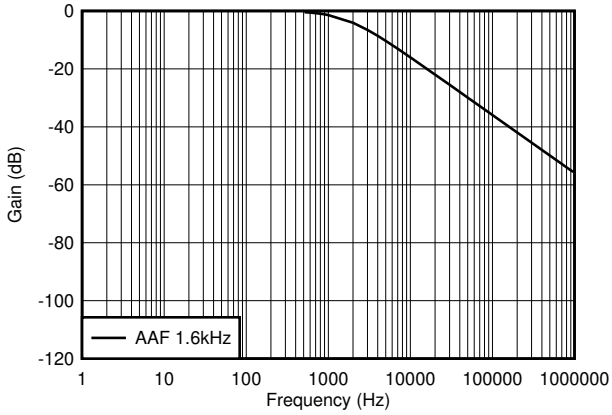


Figure 7-4. MAIN ADC Filter

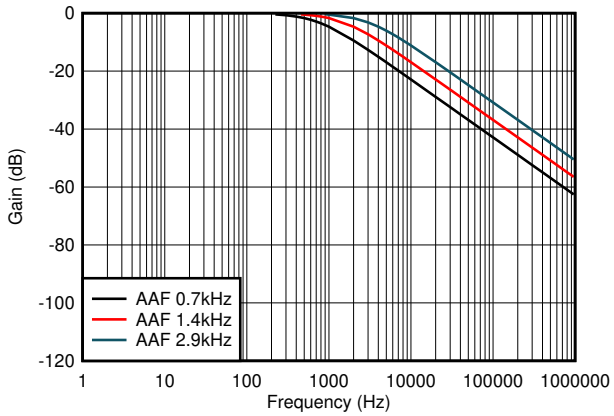


Figure 7-5. AUX ADC Filter

8 Detailed Description

8.1 Overview

The BQ79631-Q1 device is a UIR monitor that measures divided down voltages from various HV nodes. The device also has a precision current measurement that measures voltage across a low-side shunt resistor. The device can be used to measure insulation resistance by utilizing the ADC for voltage measurement and GPIO logic to control switches in this scheme. The device is also capable of measuring temperatures by reading the voltages from thermistors.

The ADCs in the daisy-chained devices can be configured to align the start of voltage measurements and all voltages can be measured within 128 µs. Each VS sensing channel includes a post-ADC digital low-pass filter (LPF) for noise reduction as well as providing moving average measurement results. The device has eight GPIOs, all of which are configurable for thermistor connections. All eight GPIOs can be measured within 1.6 ms.

The eight GPIOs can be configured as general purpose I/O. These I/O pins can be used to drive and receive logic level signals. Additionally, some of these GPIO pins can be configured as master-SPI device in order to control and communicate with external SPI peripheral devices such as an EEPROM and others.

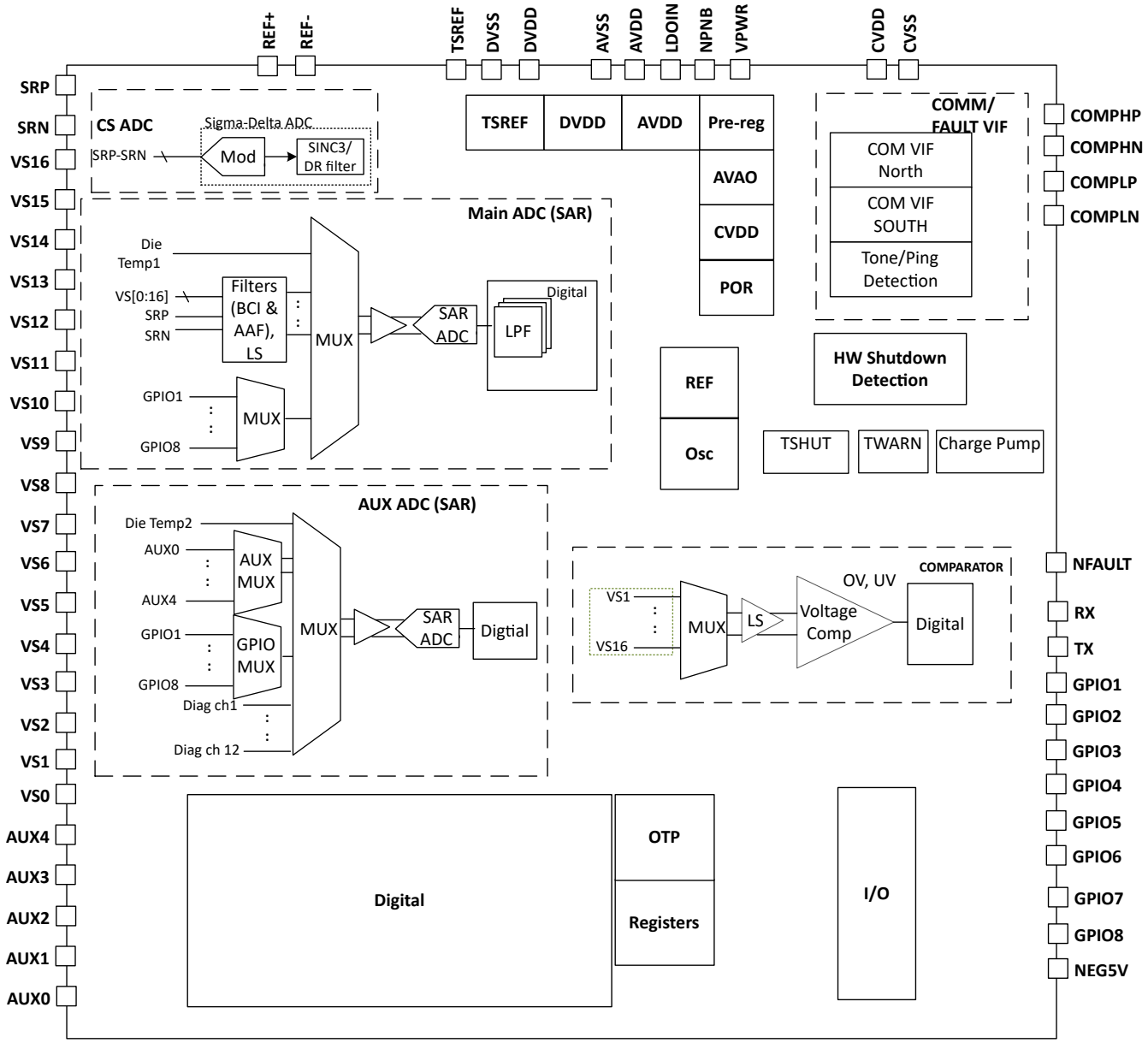
The BQ79631-Q1 communicates over daisy-chain. It has a pair of high (north) and low (south) differential communication ports, requiring only one twisted pair cable. The device supports either capacitive only, capacitive and choke, or transformer isolation. Multiple devices can be connected in a daisy-chain. Communication is reclocked on each daisy-chained device, ensuring communication integrity for long distances. An optional RING connection is supported to reverse the daisy-chain communication direction in case of cable failure. Each device includes a SPI master configured through the GPIOs.

The device includes a hardware OVUV comparator with user configurable thresholds. These can be used for overcurrent detection independent of ADC measurements. This method of overcurrent detection will need an external amplifier between the current sensing inputs and the OVUV comparator input.

The device provides an option to embed fault status information to the communication frame. The device can be configured to trigger an NFAULT pin as an interrupt signal to the system. This provides a way to reduce communication overhead without adding an additional twisted pair cable and isolation for faster fault detection.

The device has SLEEP and SHUTDOWN modes for lower power consumption. All functions work in ACTIVE mode; a hardware comparator for OVUV works in SLEEP mode. While in SHUTDOWN, all active functions are turned off. A HW reset function is available and can be activated by the host MCU. The HW reset provides a POR-like event to the device without actual power supply removal. This provides a reliable, low cost, and recoverable option to improve overall system robustness.

8.2 Functional Block Diagram



8.3 Feature Description

8.3.1 Measurement System

8.3.1.1 Voltage Measurement System

There are two SAR ADCs in the device, a 16-bit Main ADC and a 14-bit AUX ADC; both use a precision reference (REFH) for high-accuracy measurement. Each ADC has its own independent control and can be enabled or disabled separately. The Main ADC is the main measurement for VS voltages and temperature through thermistors connecting to the GPIOs. It also provides TSREF and die temperature measurements. The AUX ADC is mainly used during diagnostic procedures such as providing measurements on internal reference voltages or DAC output of the OVUV comparators. It serves as a redundancy measurement for thermistor temperature input through the GPIOs. The AUX pins are measured by the AUX ADC and this can also provide redundancy when paired with correspondingly numbered VS inputs.

A third ADC, 16-bit sigma-delta current sense ADC (CS ADC), is integrated to the device for dedicated current measurement. It is designed to work with a low-side current sense resistor. The current sense ADC measures the voltage drop across the current sense resistor with a full scale range of VCS_RANGE.

The subsections below provide an overview of the Main and AUX ADCs measurement paths. See Section 9 for the recommended external component connection. See Section 8.3.5.4 for the diagnostic control function and status of this block.

8.3.1.1.1 Main ADC

There are total of 24 inputs (slots) multiplexed to the Main ADC (Figure 8-1). All inputs are measured in round robin fashion (Figure 8-2). Each input takes 8 µs (nominal) to measure and a single round robin cycle completes in 192 µs (nominal). The inputs to the Main ADC are:

- Die temperature 1
- TSREF
- VS1 to VS16 voltages through differential VS_{n-1} to VS_n, where n = 1 to 16
- CSADC_AUX input through differential SRP–SRN pins
- Multiplexed GPIO1 through GPIO8
- Spares (RSVD)

All measurements are reported in 16-bit hexadecimal in 2s complement. Results are reported to the corresponding **_HI* (high-byte) and **_LO* (low-byte) registers. First, convert the hexadecimal results to decimal values. Follow the equations in Table 8-1 to translate the result to µV or °C.

When the Main ADC is enabled, all Main ADC-related result registers shown in Table 8-1 are reset to the default value 0x8000. The measured result is populated to the result registers as the main ADC makes its conversion along the round robin cycle. When MCU reads the **_HI* register, the device will pause the data refresh to the associated **_LO* register until that **_LO* register is read.

Table 8-1. Main ADC Measurement Conversion Equations

Main ADC Inputs	Result Registers	Conversion Equations
Die Temperature 1	*DIETEMP1_HI/LO*	Result in °C = $V_{LSB\_MAIN\_DIETEMP1}$ * Result in decimal 0x0000h is centered to 0°C.
TSREF	*TSREF_HI/LO*	Result in µV = V_{LSB_TSREF} * Result in decimal
VS1 to VS16	*VS*_HI/LO*, where * = 1 to 16	Result in µV = V_{LSB_ADC} * Result in decimal
CSADC_AUX	CSAUX_*HI/LO*	Result in µV = V_{LSB_CSAUX} * Result in decimal
GPIO1 to GPIO8	*GPIO*_HI/LO*, where * = 1 to 8	Result in µV = V_{LSB_GPIO} * Result in decimal

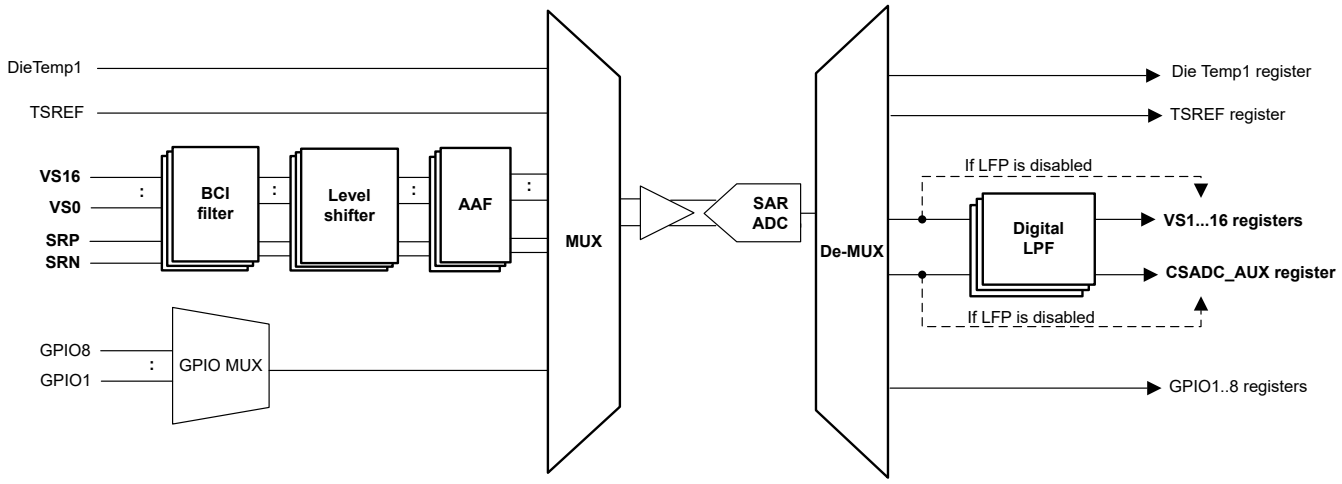


Figure 8-1. Main ADC Measurement Path

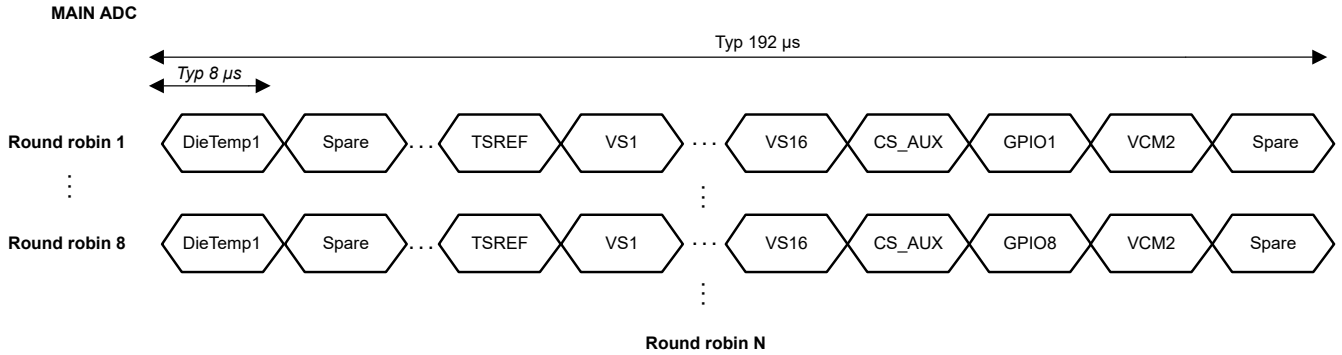


Figure 8-2. Main ADC Round Robin Measurements

8.3.1.1.1.1 VS Voltage Measurements

8.3.1.1.1.1.1 Analog Front End

The VS voltage measurements of the Main ADC are taken from the VS0 through VS16 pins. The device allows a maximum of 16 VS inputs to be measured. All VS measurements are differential measurements between VSn and VSn-1, where n=16..0. The VS0 through VS16 pins are connected to the analog front end which consists of a BCI filter, level shifter, and an anti-aliasing filter (AAF) on each VS input channel. The BCI filter has a cutoff frequency (fcutoff) of 100 kHz and the AAF has fcutoff of 1.6 kHz. This filters out high-frequency noise on the VS input before going to the high-voltage multiplexer and measured by the Main ADC. The level shifter block is turned off to save power in SLEEP and SHUTDOWN modes.

8.3.1.1.1.1.2 VS Channel Measurements

The VS pins are the input channels for VS voltage measurements from the Main ADC measured in the VS1 to VS16 slots of the round robin.

The measurement results are reported in the corresponding *VS*_HI* (high-byte) and *VS*_LO* (low-byte) registers, where * = 1 to 16. If the digital LPFs are disabled, the result registers are reported with the single ADC conversion values; otherwise, the result registers are reported with filtered measurement values.

8.3.1.1.1.1.3 Post-ADC Digital LPF

Each differential VS channel measurement is equipped with a post-ADC LPF. The LPFs have much lower cutoff frequency (f_{cutoff}). There are seven f_{cutoff} options: 6.5 Hz, 13 Hz, 26 Hz, 53 Hz, 111 Hz, 240 Hz, and 600 Hz, configurable through the *ADC_CONF1[LPF_VS2:0]* setting. Once an f_{cutoff} value is selected and the LPFs are enabled by setting *ADC_CTRL1[LPF_VS_EN]* = 1, the same f_{cutoff} setting applies to all VS channel measurements.

The digital LPF is implemented as single-pole filter which responds very similarly as an analog RC circuit. This means the Main ADC will be running in continuous mode for the digital LPFs to produce effective filtered results.

The MCU should take into account the digital filter settling time when there is a step change in the input DC voltage level. Equation below gives a typical estimate of digital filter settling time to hit settling accuracy threshold for a step in VS voltage.

Digital Filter Settling Time ~ [({log10 (Settling Accuracy Threshold [mV] / Voltage Step in Input Voltage [mV])} / {log10(1 - Filter Coefficient)}) - 1] x 0.192 ms

Table 8-2. Fcutoff for Digital LPF

Fcutoff (Hz)	600	240	111	53	26	13	6.5
Filter Coefficient	0.5	0.25	0.125	0.0625	0.03125	0.015625	0.007813

For example: If VS step by 15 mV, and user has to accommodate ~27-ms settling time to within 1 LSB of input step for 26-Hz LPF setting.

When the LPF starts, from disabled to enabled state, it jumps to its first input value and starts the filtering from that point. As compared to starting from 0 V or some mid-level voltage, this implementation allows a fast settling time for Main ADC and LFP is just starting.

8.3.1.1.1.1.4 SRP and SRN Measurements

The SRP and SRN pins are the inputs for current sense measurement from the Main ADC. The intent of this measurement path is to serve as a redundancy current measurement. The SRP/N inputs have the BCI and AAF filters in the front end. This differential current sense measurement path has an option to pass-through a post-ADC digital LPF.

The Main ADC current sense measurement is reported in the *CSAUX_HI* (high-byte) and *CSAUX_LO* (low-byte) registers. If the digital LPF is disabled, the result registers are reported with the single ADC conversion value; otherwise, the result is reported in the filtered measurement value.

8.3.1.1.1.2 Temperature Measurements

8.3.1.1.1.2.1 DieTemp1 Measurement

There are 2 die temperature sensors, DieTemp1 and DieTemp2. The DieTemp1 is routed to the Main ADC and it is also used for the Main ADC gain and offset correction internally. The measurement is reported in the *DIETEMP1_HI* (high-byte) and *DIETEMP1_LO* (low-byte) registers. The 0°C measurement is centered to hex value 0x0000h, so a positive value represents a positive temperature and a negative value represents a negative temperature. The measurement is also capped off to +200°C and –100°C.

8.3.1.1.1.2.2 GPIOs and TSREF Measurements

There are eight GPIOs. All GPIO inputs are available to be used for thermistor connections for temperature measurements and be used as a simple, single-ended, voltage input measurement.

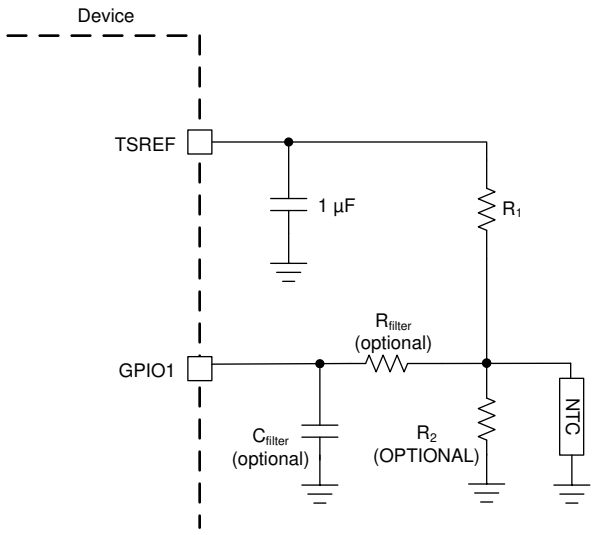


Figure 8-3. Thermistor Connection

Figure 8-3 shows the thermistor circuit when GPIO is enabled for thermistor measurements. MCU ensures TSREF is enabled by setting *CONTROL2[TSREF_EN]* = 1 and settled before taking the measurement value.

The GPIOs are multiplexed to one of the Main ADC MUX inputs. That is, in a single round robin cycle, only one GPIO is measured. To complete all eight GPIO measurements, it takes eight round robin cycles.

To enable the GPIO for ADC measurement, the corresponding *GPIO_CONFn[GPIO*2:0]* (where n = 1 to 4, * = 1 to 8 for the corresponding GPIO) register is configured to ADC input. For example, to enable GPIO1 for ADC measurement only, set *GPIO_CONF1[GPIO12:0]* to ADC input. See Section 8.3.4 for more details. If a GPIO is not configured for any ADC measurement, the device will ignore the corresponding GPIO slot but does not remove the slot from the round robin cycle. See Figure 8-4 for an example when GPIO2 is configured for non-ADC measurement.

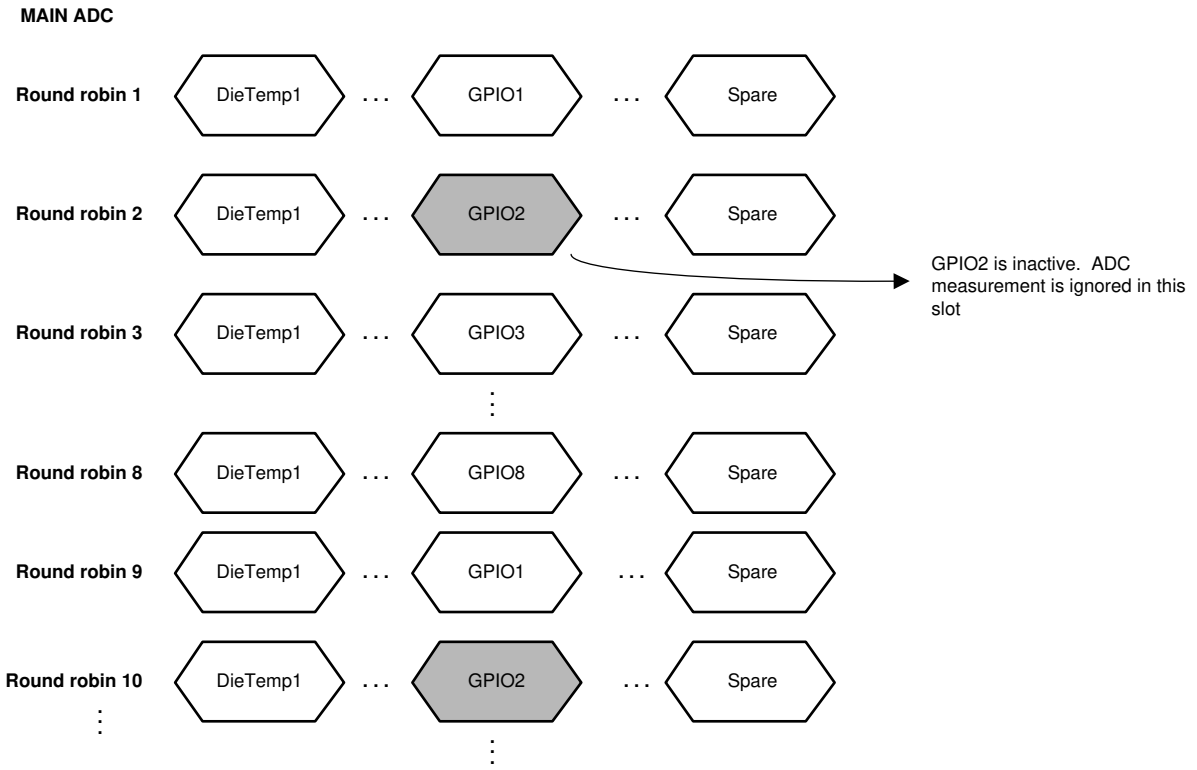


Figure 8-4. GPIO2 Not Configured for ADC Measurement

The measurements are reported in the corresponding *GPIO*_HI* (high-byte) and *GPIO*_LO* (low-byte) registers, where * = 1 to 8. The measurement result is in µV. To achieve better temperature accuracy, the MCU can use a ratiometric measurement by using both TSREF and GPIO measurement with the following formula: (GPIO_ADC/TSREF_ADC) = RNTC/(RNTC + R1), where

- GPIO_ADC = ADC measurement on GPIO
- TSREF_ADC = ADC measurement on TSREF
- RNTC = NTC thermistor resistance
- R1 is the pull-up resistor as shown in Figure 8-3 with the assumption the R2 is not used

For an inactive GPIO channel, the respective *_HI* and *_LO* registers remain with the default value 0x8000.

8.3.1.1.1.3 Main ADC Operation Control

8.3.1.1.1.3.1 Operation Modes and Status

To start the Main ADC, the host MCU sets *ADC_CTRL1[CS_MAIN_GO]* = 1. When the device receives the GO command, it first samples the following settings to determine Main ADC configuration and then operates the Main ADC accordingly. Any change of the settings below requires the MCU to resend another GO command to implement the new settings.

- *ADC_CTRL1[MAIN_MODE1:0]*: three run modes. See Table 8-3 for details.
- *ADC_CTRL1[LPF_VS_EN]*: LPF for VS channels. Set to *ADC_CONF1[LFP_VS2:0]* f_{cutoff} if enabled.
- *ADC_CTRL1[LPF_CSAUX_EN]*: LPF for CSADC_AUX channel. Set to *ADC_CONF1[LFP_CSAUX2:0]* f_{cutoff} if enabled.
- *ADC_CONF2[ADC_DLY5:0]*: Delay the start of the Main ADC. Use to align the ADC start time among the daisy-chained devices.
- *GPIO_CONF1* to *GPIO_CONF4*: Determine the inactive GPIO channel(s) and keep the result registers to default value 0x8000.
- MAIN_ADC_CAL1, MAIN_ADC_CAL2, CS_ADC_CAL1, CS_ADC_CAL2, ADC_CTRL1[CS_DR] registers.

There are two status bits to indicate the Main ADC status:

- *DEV_STAT[MAIN_RUN]*: indicates if the Main ADC is running or not.
- *ADC_STAT1[DRDY_MAIN_ADC]*: set when at least eight round robin cycles have completed indicating all active GPIO channels and all other Main ADC inputs have at least one measurement completed.

Table 8-3. Summary of Main ADC Run Modes

[MAIN_MODE1:0]	Run Mode	Description
0b00	Stop Main ADC	Stop the Main ADC
0b01	8 RR Run (eight round robin cycles)	Main ADC runs for eight round robin cycles then stops. This gives a single measurement on all VS voltages and all GPIO inputs to the system. Filtered measurements are not effective under run mode. For example, use as a quick burst read when MCU is periodically awake during system idle state.
0b10	Continuous Run	Main ADC runs in continuous mode and stops if *[MAIN_MODE1:0]* = 0b00 and a GO is sent. For example, must use this mode if LPF is enabled. Also use in diagnostic operation.

The level shifter is enabled when device enters ACTIVE mode. MCU shall wait for t_{AFE_SETTLE} time before starting the Main ADC whenever the device enters ACTIVE mode.

The Main ADC operates in ACTIVE mode only. If the ADC is running while the device goes into SLEEP, the Main ADC will be “frozen” (that is, ADC is stopped but device still remembers the operational state). When the device returns to ACTIVE mode without any digital reset event, the Main ADC will restart and continues from its “pre-frozen” state. In this condition, the VS voltage measurements are off during the t_{AFE_SETTLE} time because input voltage to the ADC is not settled yet. MCU can ignore these measurements or send a new GO command to restart the Main ADC after t_{AFE_SETTLE}.

8.3.1.1.2 AUX ADC

There are a total of 24 inputs (slots) multiplexed to the AUX ADC (Figure 8-5). All inputs are measured in round robin fashion (Figure 8-6). Each input takes 8 µs (nominal) to measure and a single round robin cycle completes in 192 µs (nominal). The inputs to AUX ADC are:

- Die temperature 2
- Multiplexed differential AUX_{n-1} to AUX_n (AUX1 to AUX4), where n = 1 to 4
- MISC measurements:
  - PWR pin
  - REFL, internal reference
  - VBG2, internal bandgap
  - VCM1, coomon voltage on Main ADC
  - AVAO_REF, always-on block reference
  - AVDD_REF
  - OV DAC from OV protector
  - UV DAC from UV protector
- Multiplexed GPIO1 to GPIO8
- Spares (RSVD)

All measurements are reported in 16-bit hexadecimal in 2s complement. Results are reported to the corresponding **_HI* (high-byte) and **_LO* (low-byte) registers. It first converts the hexadecimal results to decimal values. Follow the equations in Table 8-4 to translate the result to µV or °C.

When the AUX ADC is enabled, all AUX ADC related result registers shown in Table 8-4 are reset to the default value 0x8000. The measured result is populated to the result registers as the AUX ADC makes its conversion along the round robin cycle. When MCU reads the **_HI* register, the device will pause the data refresh to the associated **_LO* register until that **_LO* register is read.

Table 8-4. AUX ADC Measurement Conversion Equations

AUX ADC inputs	Result Registers	Conversion Equations
Die Temperature 2	*DIETEMP2_HI/LO*	Result in °C = $V_{LSB\_AUX\_DIETEMP2}$ * Result in decimal Note: 0x0000h is centered to 0°C.

Table 8-4. AUX ADC Measurement Conversion Equations (continued)

AUX ADC inputs	Result Registers	Conversion Equations
Multiplexed AUX1 to AUX4	*AUX_IN_HI/LO*, when AUX MUX is locked to a single channel	Result in µV = V_{LSB_ADC} * Result in decimal
PWR	*AUX_PWR_HI/LO*	Result in µV = $V_{LSB\_AUX\_PWR}$ * Result in decimal
REFL	*AUX_REFL_HI/LO*	Result in µV = $V_{LSB\_AUX\_DIAG}$ * Result in decimal
VCM1	*AUX_VCM1_HI/LO*	
AVAO_REF	*AUX_AVAO_REF_HI/LO*	
AVDD_REF	*AUX_AVDD_REF_HI/LO*	
OV DAC	*AUX_OV_DAC_HI/LO*	
UV_DAC	*AUX_UV_DAC_HI/LO*	
Multiplexed GPIO1 to GPIO8	*AUX_GPIO_HI/LO*	Result in µV = V_{LSB_GPIO} * Result in decimal

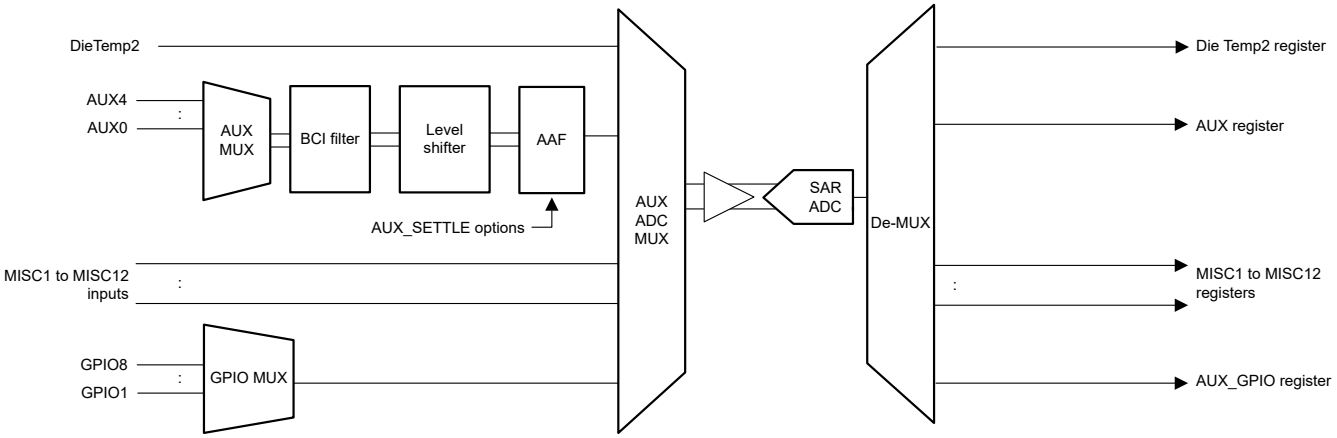


Figure 8-5. AUX ADC Measurement Path

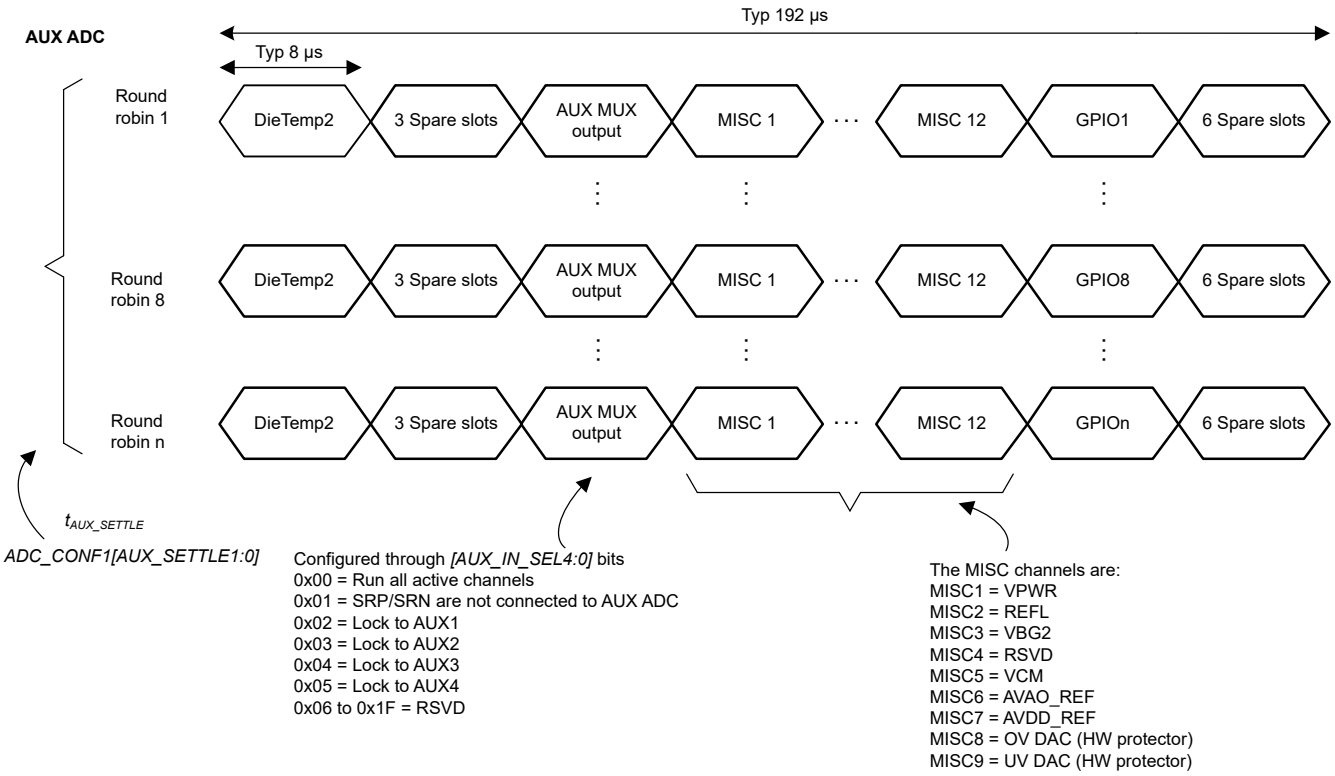


Figure 8-6. AUX ADC Round Robin Measurements

8.3.1.1.2.1 AUX Voltage Measurements

8.3.1.1.2.1.1 AUX Analog Front End

The AUX ADC path serves as a redundancy path to the Main ADC measurement on VS voltage measurements. It also has the front end filters of a BCI filter and an AAF filter in the AUX ADC path. The AUX channel are multiplexed (shown as the AUX MUX in Figure 8-5) to share a single BCI filter and AAF filter. The AUX MUX output after the front end filters is then going into one of the AUX ADC MUX and to the AUX ADC for measurement.

Because the front end filters are shared, the device has to wait for the AAF filter to settle before making any valid AUX channel measurement. The default AAF fcutoff is 1.6 kHz as in the Main ADC path, which translates to additional 4.3-ms settling time to complete a single AUX channel measurement. The device provides three AAF settling time options, 4.3 ms (default), 2.3 ms, and 1.3 ms, configured by the ADC_CONF1[AUX_SETTLE1:0] bits.

8.3.1.1.2.1.2 AUX Channel Measurements

One slot, the AUX MUX output slot, is assigned in the AUX ADC round robin cycle for the AUX channels (differential AUXn–1 – AUXn, where n = 1 to 4) measurement because these channels are multiplexed to a single input to the AUX ADC multiplexer. For a single AUX channel measurement, it takes multiple round robin cycles because the device has to wait for the AAF settling time as well.

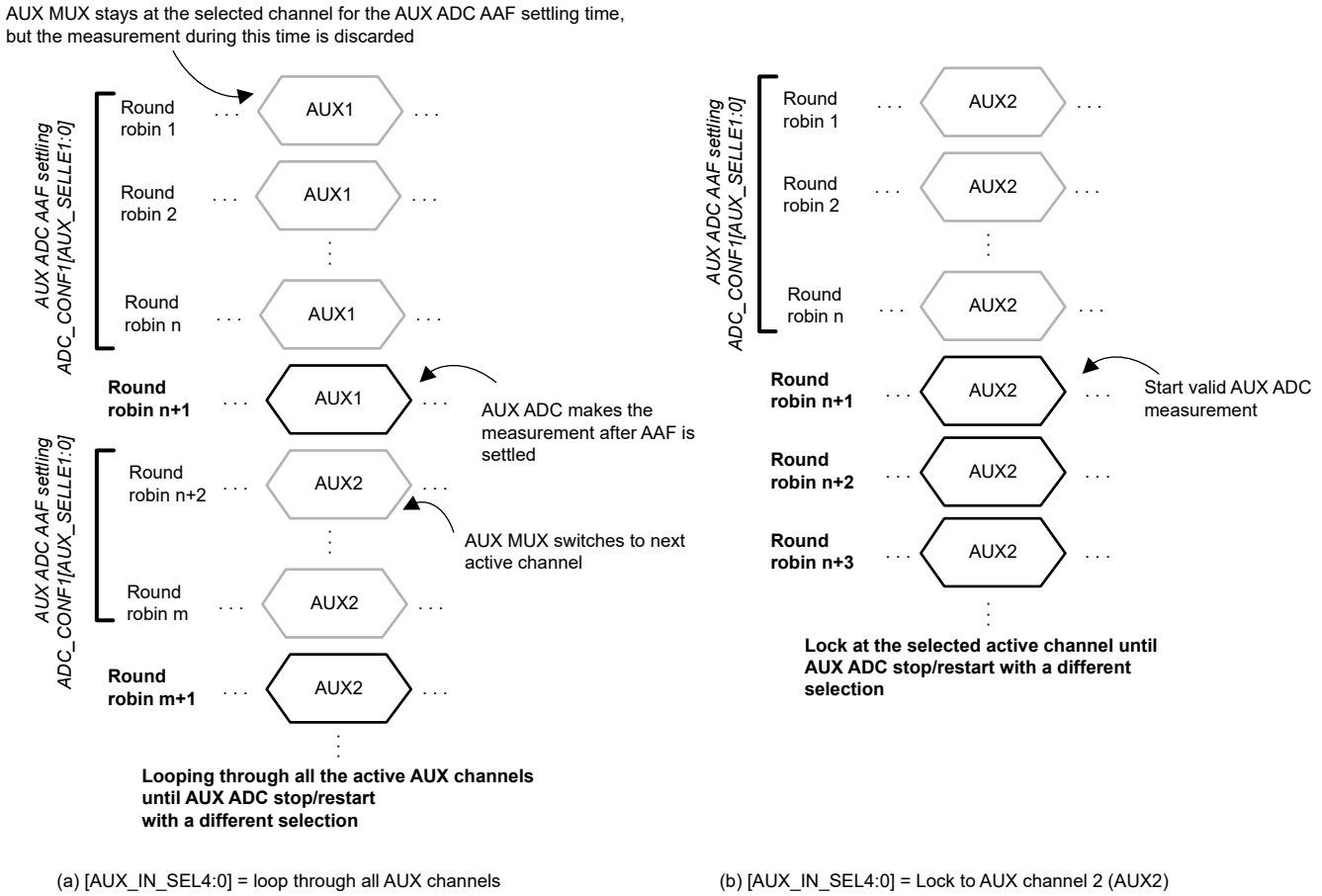
Because of the need to wait for the AAF to settle, the AUX ADC would only measure AUX channels that are active and are selected by the MCU; inactive or unselected channels are skipped.

MCU can control which AUX channels to be measured through the AUX ADC. The *ADC_CTRL2[AUX_IN_SEL4:0]* gives the options to run through all the active AUX channels or to lock to a single AUX channels. Figure 8-5 shows the example of how the AUX slot is implemented with different *[AUX_IN_SEL4:0]* setting.

It is recommend to run AUX ADC in continuous mode and all AUX ADC to measure through all the AUX channels once. This enables the device to reduce the common mode error in AUX ADC measurement. MCU shall perform this procedure before running ADC comparison related diagnostic or locking to a single AUX channel measurement.

There is no post-ADC LPF in the AUX ADC path. When the AUX ADC measurements are used during diagnostics, the AUX channel measurements are compared against the Main ADC prefiltered measurements.

The device makes the AUX channel measurement available to read only when the *[AUX_IN_SEL4:0]* bits are set to lock on a single AUX channel. The measurement is reported in the *AUX_HI* (high-byte) and *AUX_LO* (low-byte) registers. The result registers will be updated after the AAF settling time is passed. When cycling through AUX inputs (not in AUX lock mode) , data from round robin cycles 5 through 16 should be disregarded.



(a) [AUX_IN_SEL4:0] = loop through all AUX channels

(b) [AUX_IN_SEL4:0] = Lock to AUX channel 2 (AUX2)

Figure 8-7. AUX MUX Output Slot with Different *[AUX_IN_SEL4:0]* Setting

8.3.1.1.2.2 AUX Temperature Measurements

8.3.1.1.2.2.1 DieTemp2 Measurement

There are two die temperature sensors, DieTemp1 and DieTemp2. The DieTemp2 is routed to the AUX ADC and is also used for the AUX ADC gain and offset correction internally. The measurement is reported in the *DIETEMP2_HI* (high-byte) and *DIETEMP2_LO* (low-byte) registers. The 0°C measurement is centered to hex value 0x00, so a positive value represents positive temperature and a negative value represents negative temperature. The measurement is also capped off to +200°C and –100°C.

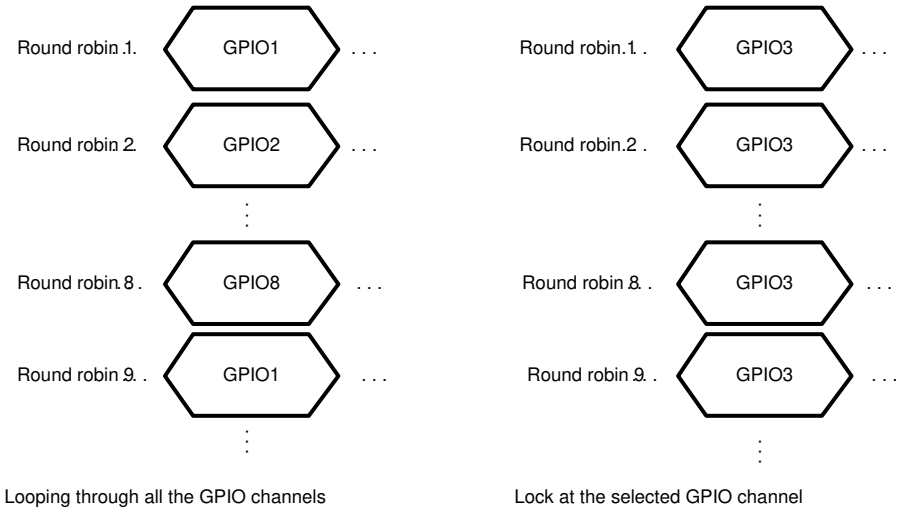
8.3.1.1.2.2.2 AUX GPIO Measurements

The AUX GPIO path is the same as the main GPIO path. All eight GPIOs are multiplexed to a single AUX ADC MUX input. There is only one GPIO slot in the AUX ADC round robin cycle. That is, in a single AUX

ADC round robin cycle, only one GPIO will be measured. To complete all eight GPIO measurements, it takes eight round robin cycles. If GPIO is connected to the thermistor network, the MCU enables TSREF by setting *CONTROL2[TSREF_EN]* = 1 and ensures TSREF is stable before starting the AUX ADC measurement.

When AUX ADC is enabled, the GPIO slot in the 1st round robin cycle is GPIO1, 2nd round robin cycle is GPIO3, and so on. For the AUX ADC to make a measurement on a GPIO, the GPIO must be configured as ADC input in the corresponding *GPIO_CONFn[GPIO*2:0]* bits, where n = 1 to 4, * = 1 to 8 for the respective GPIO channel. See Section 8.3.4 for more details. If the GPIO is inactive for the ADC measurement, the device ignores the corresponding GPIO slot but does not remove the slot from the AUX ADC round robin cycle.

By default, the AUX ADC loops through all GPIO channels and the measurements do not report out to the result registers. However, if MCU locks to a single GPIO channel, the locked GPIO measurement is reported to the *AUX_GPIO*_HI* (high-byte) and *AUX_GPIO*_LO* (low-byte) registers. This channel lock can be set by the *ADC_CTRL3[AUX_GPIO_SEL3:0]* bits. The result registers will report a GPIO measurement if *[AUX_GPIO_SEL3:0]* is locked to single GPIO channel, any other condition will show default value 0x8000.



(a) [AUX_GPIO_SEL3:0] = loop through all GPIO channels **(b) [AUX_GPIO_SEL3:0] = Lock to GPIO3**

Figure 8-8. GPIO Slot with Different *[AUX_GPIO_SEL3:0]* Setting

8.3.1.1.2.3 MISC Measurements

There are 12 MISC measurements listed at the beginning of the AUX ADC section. When the AUX ADC is enabled, these inputs are measured in every round robin cycle. Table 8-4 shows the corresponding result registers.

The DAC inputs of the OVUV protectors reflect the real-time DAC values of the device which shows the OVUV detection or recovery threshold currently in use in the protectors. It is normal to observe a change of the DAC measurements if there are unused channels or if any VS or GPIO channels detect a fault. See Section 8.3.5.4 for the protector DAC measurement configuration.

8.3.1.1.2.4 AUX ADC Operation Control

To start the AUX ADC, the host MCU sets *ADC_CTRL3[AUX_GO]* = 1. When the device receives the GO command, it first samples the following settings to determine the AUX ADC configuration, then operates the AUX ADC accordingly. Any change to the settings below requires the MCU to send another GO command to implement the new settings.

- *ADC_CTRL3[AUX_MODE1:0]*: Four run modes. See Table 8-5 for details.
- *ADC_CTRL2[AUX_IN_SEL4:0]*: Selects which AUX channels are measured by AUX ADC.

- *ADC_CONF1[AUX_SETTLE1:0]*: Configures the AUX ADC AAF settling time.
- *ADC_CTRL3[AUX_GPIO_SEL3:0]*: Selects which GPIO channels are measured by AUX ADC.
- *GPIO_CONF1* to *GPIO_CONF4*: Determines the inactive GPIO channel(s).

There are four status bits to indicate the AUX ADC status:

- *DEV_STAT[AUX_RUN]*: indicates if the AUX ADC is running or not.
- *ADC_STAT1[DRDY_AUX_MISC]*: set when all MISC inputs are measured at least once.
- *ADC_STAT1[DRDY_AUX_IN]*: set when the AUX channels selected by *[AUX_IN_SEL4:0]* are measured at least once.
- *ADC_STAT1[DRDY_AUX_GPIO]*: set when all GPIO channels (active or inactive) have been measured once. Inactive channel measurements will be ignored by the device.

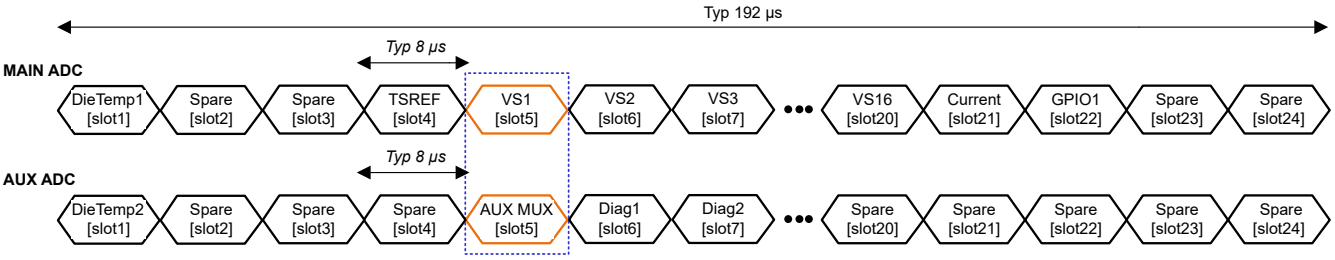
Table 8-5. Summary of AUX ADC Run Modes

[AUX_MODE1:0]	Run Mode	Description
0b00	Stop AUX ADC	Stop the AUX ADC
0b01	Single Run (1 round robin cycle)	AUX ADC runs for one round robin cycle then stops. This gives a single measurement on all MISC inputs. For example, use as a quick burst read for just the MISC inputs without the need to issue a stop command to the AUX ADC.
0b10	Continuous Run	AUX ADC runs in continuous mode and stops if *[AUX_MODE1:0]* = 0b00 and a GO command is sent. For example, must use this mode when ADC diagnostic comparison operation is used. See Section 8.3.5.4 for details.
0b11	8 RR Run (eight round robin cycles)	AUX ADC runs for eight round robin cycles then stops. This gives a single measurement on all active GPIO inputs.

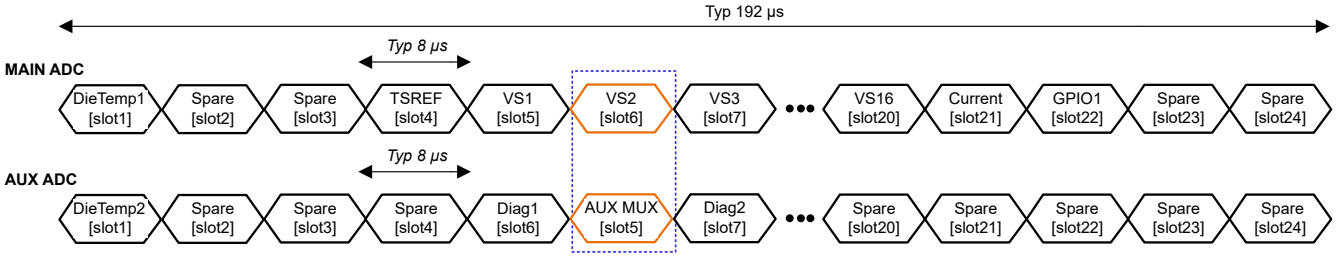
The AUX ADC operates in ACTIVE mode only. If the ADC is running while the device goes into SLEEP mode, the AUX ADC will be “freezed”; that is, the ADC stops but the device still remembers the operational state. When the device returns to ACTIVE mode without any digital reset event, the AUX ADC will restart and continue from its “prefreeze” state.

8.3.1.1.3 Synchronization Between MAIN and AUX ADC Measurements

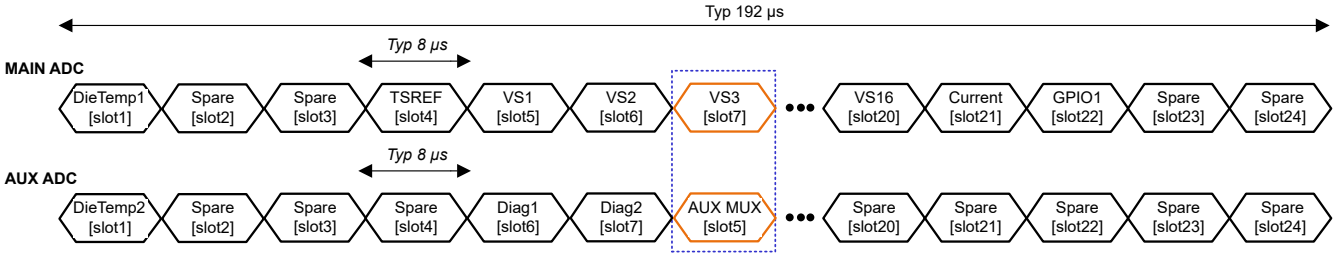
The device aligns AUX time slot number 5 with the target VS channel slot on MAIN input. DieTemp2 starts without any delay, and AUX MUX slot #5 moves dynamically accordingly to match the selected MAIN input and the remaining AUX ADC slots adjust accordingly. This ensures that there is no time skew between MAIN input and AUX input ADCs sampling. This feature helps improve the ASIL-D accuracy significantly.



a) AUX_IN_SEL = 00h – Running all active channels. Ch1 conversion.



b) AUX_IN_SEL = 00h – Running all active channels. Ch2 conversion.



c) AUX_IN_SEL = 04h – Lock to AUX 3. Ch3 conversion.

Figure 8-9. Synchronization Between MAIN and AUX ADC Sampling

8.3.1.2 Current Sense ADC

The CS ADC is a high accuracy delta-sigma ADC with a SINC3 filter, dedicated for current sensing. It is used a divided down precision reference. The same precision reference is also used by the Main and AUX ADCs. The CS ADC block measures current by directly sensing the differential voltage across a sense resistor connecting between SRP and SRN pins. Bi-directional current can be measured. The CS ADC supports only low side sense resistor. The full-scale ADC input range is V_{CS_RANGE}. See the Electrical Characteristics table for the recommended range. If current sense ADC input is larger than Full Scale input voltage/V_{CS_RANGE}, CURRENT_HI/MID/LOW would be clamped around 75-mV output reading. To verify, user could read CSAUX_HI/LO.

The decimation ratio (DR) directly correlates to how quickly a conversion result is available to be read from the ADC. Lower DR corresponds to faster conversion time and lower effective number of bits (ENOB). The DR setting is controlled by ADC_CTRL1[CS_DR1:0]. The CS ADC shares the same start and mode control bits as the Main ADC located in ADC_CTRL1 register. Both the Main and CS ADCs stop together. Such design is to allow better voltage and current measurement alignment.

The measurement is reported in 24-bit hexadecimal in 2s complement. Results are reported to the corresponding CURRENT_HI (high-byte), CURRENT_MID (mid-byte) and CURRENT_LO (low-byte) registers. It first converts the hexadecimal results to decimal values. Convert the result to µV, where result in µV = V_{LSB_CS} result in decimal.

After receiving the GO command, the CS ADC starts its first conversion after t_{CS_SETTLE}. Since the CS ADC is using a SINC3 filter, the first conversion takes t_{CS_CONV} time to complete, but any subsequent conversion takes t_{CS_CONV}/3 time to complete. If MCU needs to catch every current measurement conversion, GPIO1 has an option to toggle low every time a CS ADC conversion is completed, the pin returns high when MCU read CURRENT_HI register. This signal can be used as an interrupt to the MCU to avoid missing a conversion. This function is enabled by setting GPIO_CONF2[CS_RDY_EN] = 1. The CSADC is measuring the input throughout its conversion time.

Auxiliary measurement of the voltage across the shunt resistor can be done by the SAR ADC in the MAIN ADC path. The CSADC and the Auxiliary measurement path are shown in Figure 8-10 and Figure 8-11. Details pertaining to the Auxiliary measurement are available in the Main ADC measurement section.

Over current detection scheme can be implemented using the OVUV comparators in this device. Refer to Section 9.2.1.2.5 for the implementation details.

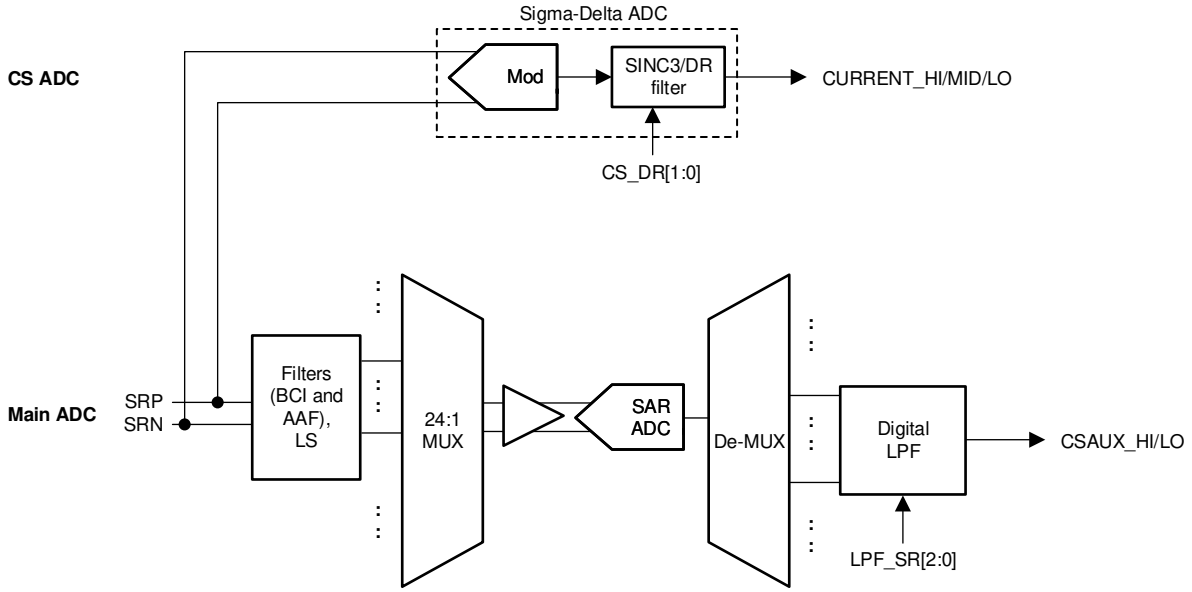


Figure 8-10. CS ADC

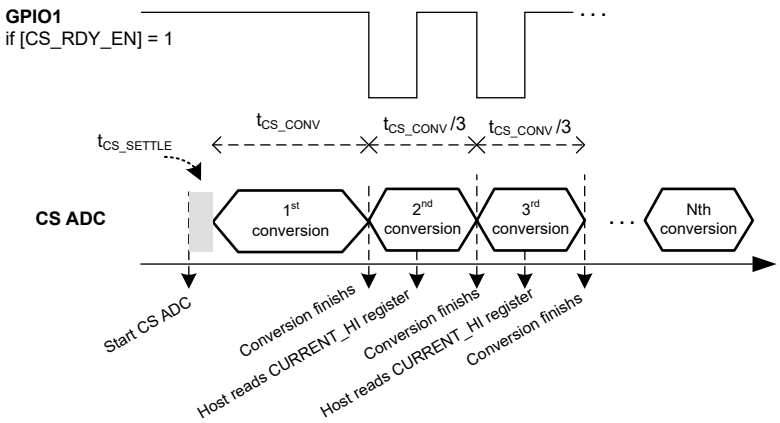


Figure 8-11. CS ADC Measurement

8.3.2 OVUV Detection

The device integrates VS OV and UV protectors with programmable thresholds. OV and UV comparators are useful for flagging an over voltage or undervoltage on particular pins. This feature can be used to implement Over Current detect feature by monitoring an external amplifier that amplifies the voltage across a current

shunt resistor. These protectors are independent of the ADC functionality or the ADC measurements path. The OVUV protectors can operate in ACTIVE or SLEEP mode. The subsections below provide an overview of the protectors. See Section 8.3.5.4 for diagnostic control function and status of this block

A set window comparator provides voltage monitoring for all VS channels. This comparator function is entirely separate from the ADC function and as such, even if the ADC function fails, the analog comparators still flag the crossing of the overvoltage (OV) and undervoltage (UV) comparator thresholds. The programmed thresholds are translated through DACs to the comparators.

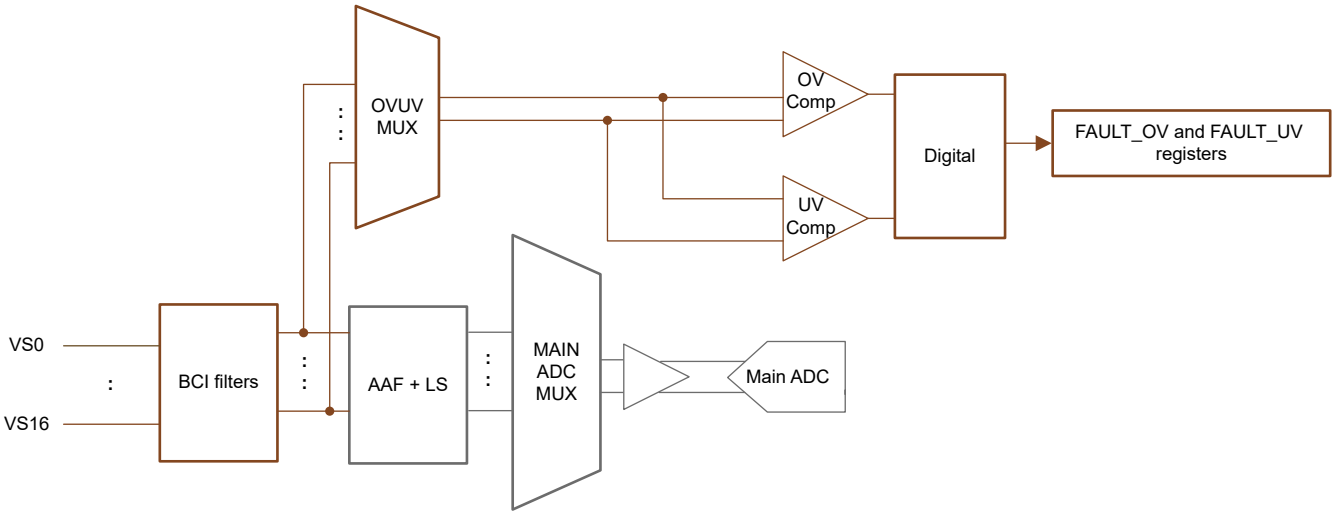


Figure 8-12. OV and UV Protectors

The OV and UV thresholds set by *OV_THRESH* and *UV_THRESH* registers are the same for all VS channels.

The *UV_DISABLE1* and *UV_DISABLE2* registers setting disable any individual channel for UV detection.

Otherwise, the OV protector detects an OV fault on a particular channel if the VS channel voltage is greater than the OV_THRESH setting. The UV protector detects a UV fault on a particular channel if the VS channel voltage is less than the UV_THRESH setting.

8.3.2.1 OVUV Operation Modes

The OV and UV protectors have several operation modes controlled by *OVUV_CTRL[OVUV_MODE1:0]* and is summarized in Table 8-6. To start the OVUV protectors, MCU sets *OVUV_CTRL[OVUV_GO]* = 1.

Table 8-6. OVUV Protector Operation Modes

[OVUV_MODE1:0]	Operation Mode	Description
0b00	Stop OV and UV protectors	Stop OV and UV protectors
0b01	Round robin run	The OV and UV protectors are looping through all VS inputs. The UV protector detects *UV_THRESH*.
0b10	OV and UV BIST run (diagnostic use, see Section 8.3.5.4 for details)	A BIST (built-in self-test) cycle on the OV and UV comparators and the detection paths. VS ADC measurement from the Main ADC and the OV and UV detections through the OVUV protectors are not available during this run. MCU shall stop ADC measurement when performing OVUV BIST.
0b11	Single channel run (diagnostic use, see Section 8.3.5.4 for details)	Use for checking the OV and UV DACs. The OV and UV comparator is locked to a single VS input channel in this mode. Channel is locked by *OVUV_CTRL[OVUV_LOCK3:0]*.

If OVUV BIST run is in progress, but MCU start ADC, the ADC result registers will be held at 0x8000. ADC measurements will resume once OVUV BIST is completed and after t_{AFE_SETTLE} time pass.

If ADC is running, but MCU start OVUV BIST, the ADC result registers will be held at its last measurement. ADC measurement update resumes once OVUV BIST is completed and after t_{AFE_SETTLE} time pass

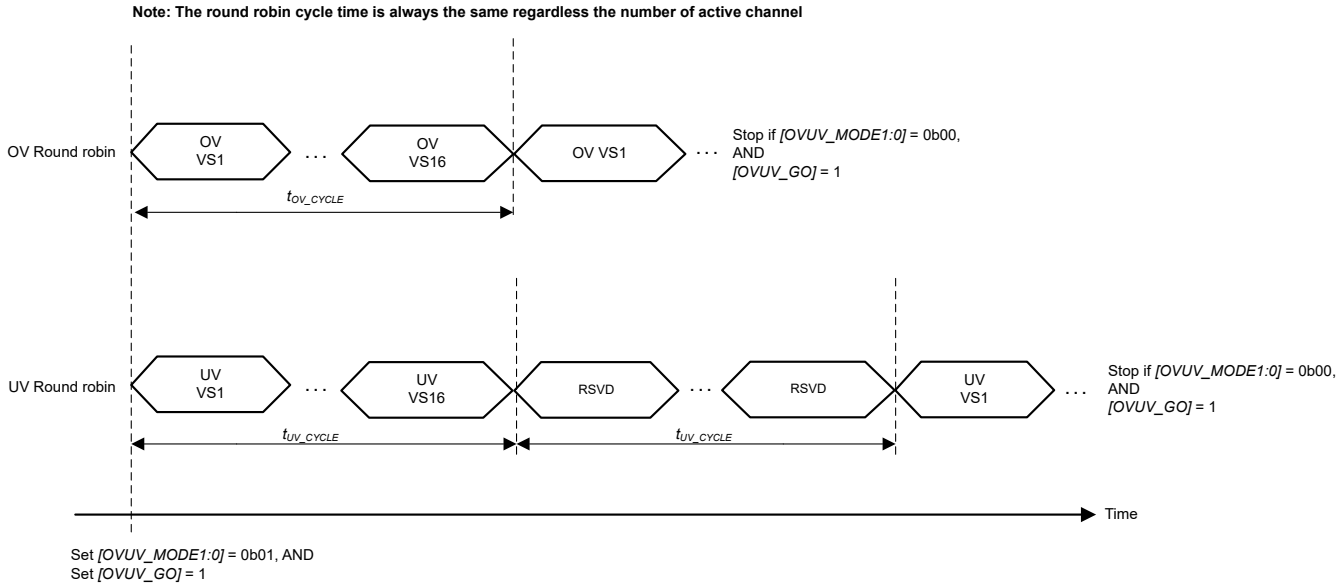


Figure 8-13. OV and UV Round Robin Mode

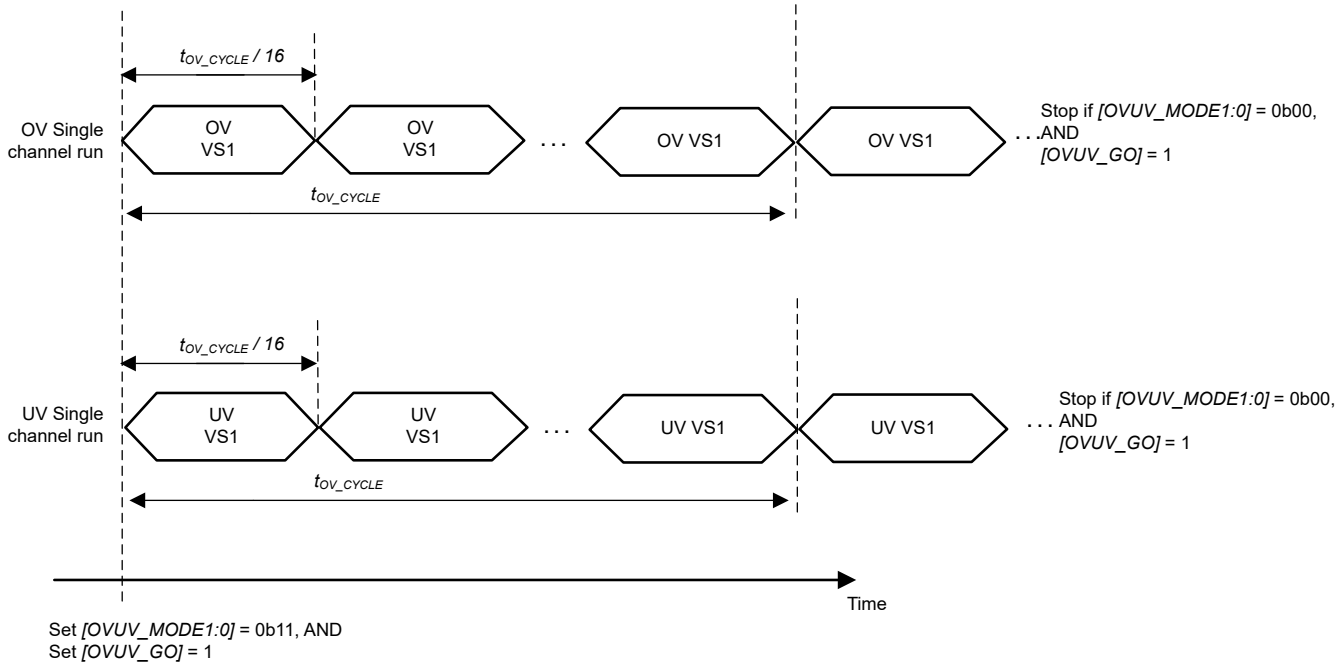


Figure 8-14. OV and UV Single Channel Run, Locked to VS1

8.3.2.2 OVUV Control and Status

8.3.2.2.1 OVUV Control

To start the OV and UV protectors, MCU sets *OVUV_CTRL[OVUV_GO]* = 1. When the device receives the GO command, it samples the following register settings and then starts the OVUV protectors accordingly. Any change of the settings below requires the MCU to resend another GO command to implement the new settings.

- *OV_THRESH* register: Sets the OV threshold for all VS channels

- *UV_THRESH* register: Sets the UV threshold for all VS channels
- *OVUV_CTRL[OVUV_MODE1:0]*: OVUV operation mode selection
- *UV_DISABLE1* and *UV_DISABLE2* registers: Determines the inactive VS channel(s) and ignores the detection result accordingly.

The OVUV protectors can also operate in SLEEP mode. MCU first starts the protector in ACTIVE mode, then puts the device in SLEEP mode. The OVUV protectors will continue the operation until the MCU commands to stop or if the device shuts down.

8.3.2.2.2 OVUV Status

The *DEV_STAT[OVUV_RUN]* = 1 indicates the OVUV protectors are running. The OV detection result is reflected in the *FAULT_OV1* and *FAULT_OV2* registers; the UV detection result is reflected in the *FAULT_UV1* and *FAULT_UV2* registers.

8.3.3 Power Supplies

The device generates all required supplies for its operation from the supply connected to the PWR pin. The following subsections provide an overview of each internal supply block. See Section 9 for recommended component connection. See Section 8.3.5.4 for diagnostic control and fault detection on the power supplies block.

8.3.3.1 AVAO_REF and AVDD_REF

The AVAO_REF block (analog voltage always on) is powered from the PWR pin. It powers the always-on low- current circuits that are required for all power modes. This block also generates a preregulated reference, AVAO_REF. The AVAO_REF voltage passes through a load switch controlled by the SHUTDOWN mode. The reference voltage after the load switch is AVDD_REF.

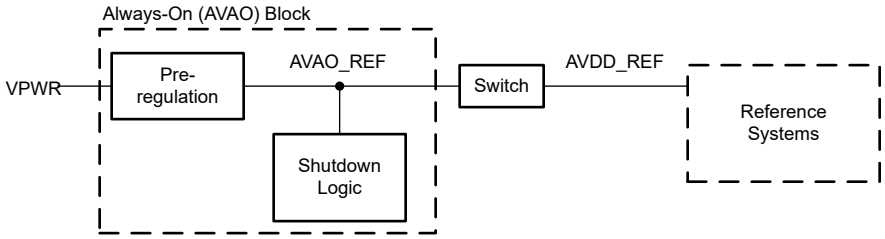


Figure 8-15. AVAO Block

8.3.3.2 LDOIN

From VPWR, the device generates a 6-V regulated voltage (nominal) on the LDOIN pin through the internal linear regulator and an external NPN transistor. The NPNB pin controls the external NPN transistor of the regulator. The LDOIN output is the preregulated input to the rest of the internal low-dropout regulators (LDOs). During OTP (One-Time Programmable) memory programming, the LDOIN pin will be regulated to 8 V (nominal) to supply the programming voltage internally to the OTP programming. The LDOIN is turned off only during HW reset or a POR event.

8.3.3.3 AVDD

The AVDD LDO is the supply for the analog circuits. It takes the input voltage from LDOIN and generates a nominal 5 V. It will not be used to power any external circuit. This LDO is powered down in SHUTDOWN mode, during HW reset, or a POR event.

8.3.3.4 DVDD

The DVDD LDO is the supply for the digital circuits. It takes the input voltage from LDOIN and generates a nominal 1.8 V. It will not be used to power any external circuit. This LDO is powered down in SHUTDOWN mode, during HW reset, or a POR event.

8.3.3.5 CVDD and NEG5V

The CVDD LDO is the supply for the daisy-chain interface (or vertical interface, VIF) and the I/O pins (RX, TX, NFAULT, and GPIOs). It takes the input voltage from LDOIN and generates a nominal 5 V. Besides providing power for internal usage, this LDO can support an extra 10-mA external load in ACTIVE and SLEEP mode.

External loading support is limited under restricted condition when device is in SHUTDOWN. A max of 5-mA external load is allowed in SHUTDOWN mode only if device can be waken up by WAKE ping. For example, when device is used as standalone device or when place in the base position in the daisy-chain. When device is place in the stack, in which a WAKE tone is required to wake up the device, CVDD cannot have any external loading when device is in SHUTDOWN.

There is a –5-V charge pump used for the daisy-chain driver and Main ADC blocks. The NEG5V pin has a –4.6-V output (nominal). It will be in a low-power burst mode when the device is in SLEEP or SHUTDOWN mode.

8.3.3.6 TSREF

The TSREF is a 5-V buffered reference that can bias the external thermistor circuits, allowing the ADCs to measure temperature. This reference is measurable by the Main ADC. Both TSREF and GPIO measured by the Main ADC give a ratiometric measurement for best temperature measurement.

The TSREF is capable of supplying up to ITSREF_ILMIT. The TSREF is off by default and can be enabled or disabled through the *CONTROL2[TSREF_EN]* bit. The startup time of TSREF is determined by the external capacitance. The MCU ensures TSREF is stable before making any GPIO measurement. After enabling TSREF LDO, user shall wait 380 µs before sending the next command.

8.3.4 GPIO Configuration

The device has eight GPIOs. Each GPIO can be programmed to be one of the configurations below through the *GPIO_CONF1* to *GPIO_CONF4* registers.

Table 8-7. GPIO Configuration

GPIO	DISABLE	INPUT		OUTPUT		WEAK PULL-UP/DOWN		SPECIAL	
	High-Z	Digital	ADC Only	High	Low	ADC & weak pull-up	ADC & weak pull-down	SPI Master *[SPI_EN]* = 1	Fault Input *[FAULT_IN_ EN]* = 1
GPIO1	√	√	√	√	√	√	√		
GPIO2	√	√	√	√	√	√	√		
GPIO3	√	√	√	√	√	√	√		
GPIO4	√	√	√	√	√	√	√	√ (SS)	
GPIO5	√	√	√	√	√	√	√	√ (MISO)	
GPIO6	√	√	√	√	√	√	√	√ (MOSI)	
GPIO7	√	√	√	√	√	√	√	√ (SCLK)	
GPIO8	√	√	√	√	√	√	√		√ (input, active low)

Table 8-8. GPIO Configuration

GPIO Configuration		Description
DISABLE	High-Z	This is the default GPIO configuration at reset if OTP is not programmed
INPUT	Digital	When GPIO is configured as Digital Input, the device detects the input voltage level to determine a 1 or 0 with respect to its V_{IL} and V_{IH} levels. The result is shown in the *GPIO_STAT* register.
	ADC only	The GPIO is configured to be measurable by the ADC (both main and AUX ADCs) only. Example: use this selection to measurement voltage on GPIO.
OUTPUT	High	The GPIO is configured as digital output high (internally pull up to CVDD). The logic state is also shown in the *GPIO_STAT* register.
	Low	The GPIO is configured as digital output low. The logic state is also shown in the *GPIO_STAT* register.

Table 8-8. GPIO Configuration (continued)

GPIO Configuration		Description
WEAK PULL-UP/DOWN	ADC and Weak Pull-up	The GPIO is pull up internally and is configured to measured by the ADC (both main and AUX ADCs)
	ADC and Weak Pull-down	The GPIO is pull down internally and is configured to measured by the ADC (both main and AUX ADCs)
SPECIAL	SPI Master	When *GPIO_CONF1[SPI_EN]* = 1, GPIO4 to GPIO7 are taken over as the SPI master communication lines. This configuration has higher priority over any of the INPUT/OUTPUT configurations on GPIO4 to GPIO7.
	Fault Input	When *GPIO_CONF1[FAULT_IN_EN]* = 1, GPIO8 is taken over as an input that if the GPIO was asserted (active low), will set *FAULT_SYS[GPIO]* = 1 and assert NFAULT (if enabled).

8.3.5 Communication, OTP, Diagnostic Control

8.3.5.1 Communication

The device can operate as a standalone device in a multidrop configuration (*DEV_CONF[MULTIDROP_EN]* = 1) or as a base/stack device in a daisy-chain configuration (*DEV_CONF[MULTIDROP_EN]* = 0). In multidrop configuration, the daisy-chain communication is disabled and the host communicates only with a single device through UART interface. This document will focus on the daisy-chain communication.

In daisy-chain configuration, each device is identified by a 6-bit device address; hence, up to 64 devices can be connected in the daisy-chain. In this configuration, a device is either defined to a base (interface with host through UART) or a stack (interface through the daisy-chain ports COMH/COML to the base device). The base description in this document assumes the use of BQ79631-Q1 as base device. If a communication extender (also known as bridge device) is used as a base, user must refer to the bridge device's data sheet for details.

8.3.5.1.1 Serial Interface

The device has a serial interface which uses UART protocol as the physical layer to communicate between base device and host. The communication is specified in a proprietary frame structure.

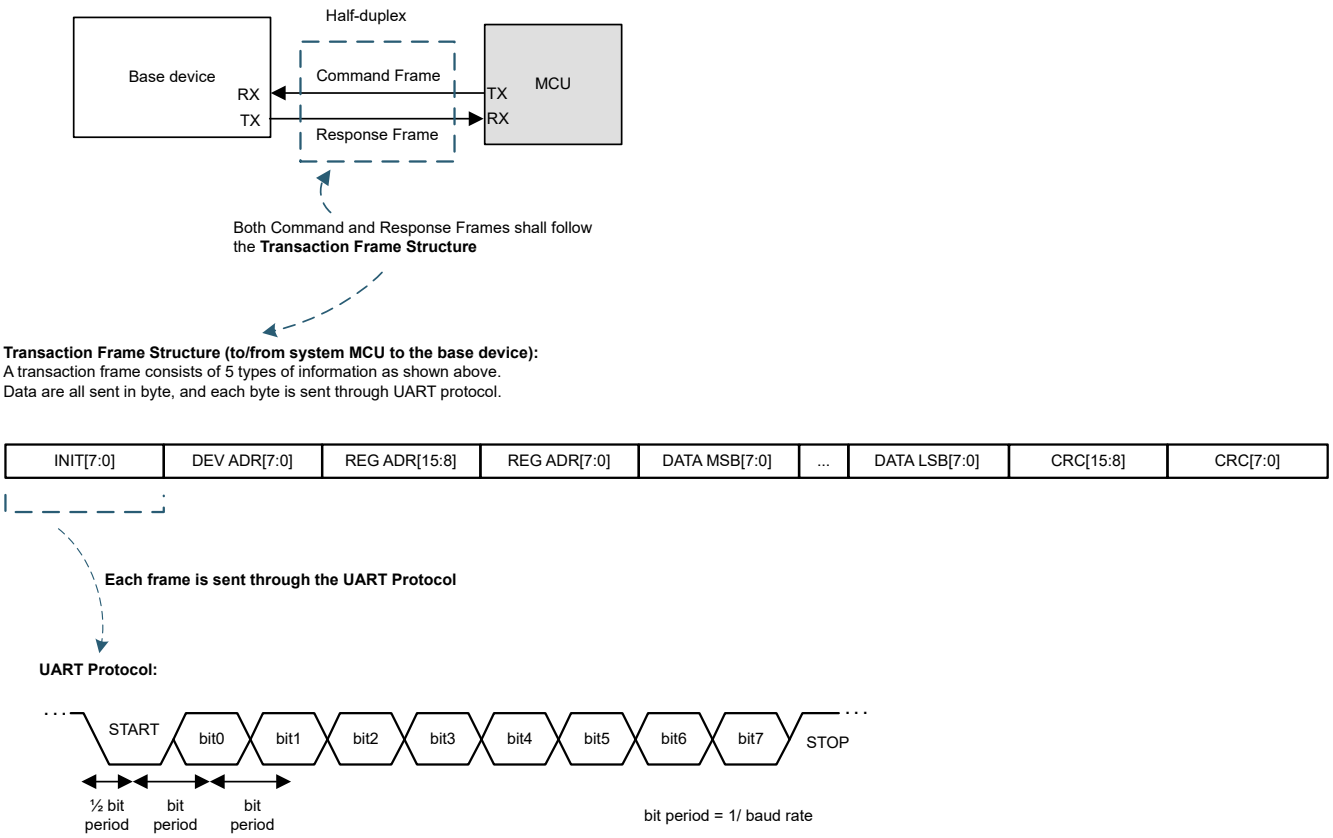


Figure 8-16. UART Communication to Host

8.3.5.1.1.1 UART Physical Layer

The UART interface follows the standard serial protocol of 8-N-1, where it sends information as a START bit, followed by eight data bits, and then one STOP bit. The STOP bit indicates the end of the byte. If a byte is received that does not have the STOP bit set, the *FAULT_COMM1[STOP_DET]* bit is set, indicating there may be a baud rate issue between the host and the device. The device supports 1-Mbps baud rate. Additionally, during development, a slower baud rate is needed to debug the communication, an optional 250-kbps baud rate can be enabled under communication debug mode.

The UART sends data on the TX pin and receives data on the RX pin. When idle, the TX and RX pins are high. The UART interface requires that RX is pulled up to CVDD through a resistor on the base device. The RX is pulled up on the device side. Do not leave RX unconnected. Ensure RX is connected directly to CVDD for stack devices.

The TX pin is disabled in stack devices, but must be pulled high through a resistor on the host side on base device to prevent triggering an invalid communications frame when the communication cable is not attached, or during power-off or SHUTDOWN state when TX is high impedance. TX is always pulled to CVDD internally while in ACTIVE or SLEEP mode, whether enabled or disabled. Leave TX unconnected if not used in stack devices.

The UART interface is strictly a half-duplex interface. While transmitting, any attempted communication on RX is ignored. The only exception is COMM CLEAR signal on RX pin, which immediately terminates the communication. See Section 8.3.5.1.1.1.3 for details.

Using two STOP bits in UART:

The device can be set up with two stop bits (*DEV_CONF[TWO_STOP_EN]* = 1), the UART response frame transmits from device to host will always return with two STOP bits as shown below. Host is not required to send the command frame to the device with two STOP bits. The device is able to receive one or more stop bits with or without this function enabled.

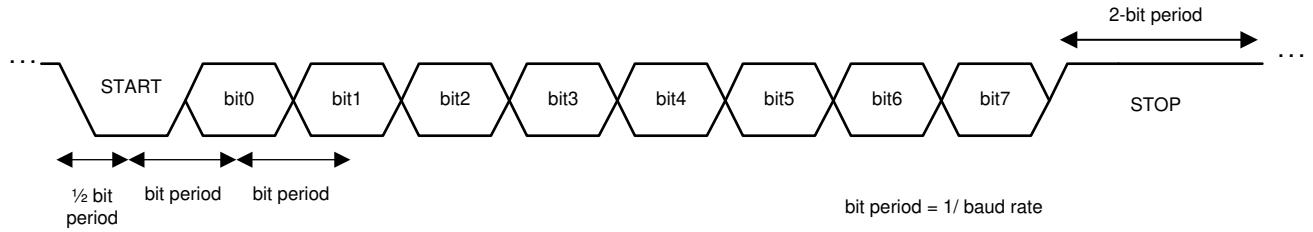


Figure 8-17. UART Response Frame with Two STOP Bits

Potential use of the two stop bits may be to:

- The host to gain extra time to process the data before receiving next data frame.
- The clock tolerance between device and host might cause the data detection out of sync. Having two STOP bits allows re-synchronization of the communication; hence, improving communication robustness.

Although UART is only used by the base device, if the *[TWO_STOP_EN]* = 1, the stack devices also set the *[TWO_STOP_EN]* = 1 even though UART is not used in stacks. It is because the stack devices will use the bit setting to determine the proper gap applying between two communication frames.

8.3.5.1.1.1.1 UART Transmitter

The transmitter is configured to wait a specified number of bit periods after the last bit reception before starting transmissions using the *TX_HOLD_OFF* register. This provides time for the host to switch the bus direction at the end of its transmission. The UART transmitter is disabled by default in the stack devices.

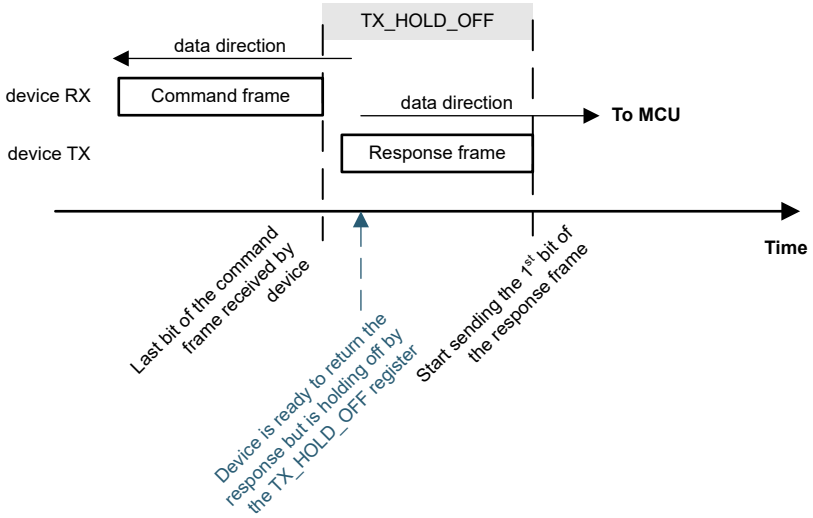


Figure 8-18. UART TX_HOLD_OFF

8.3.5.1.1.1.2 UART Receiver

While the device is transmitting data on TX, RX is ignored except when receiving a COMM CLEAR. To avoid collisions during data transmission up the daisy-chain interface, the host must wait until all bytes of a communication transmission are received from the device before attempting additional communication to the device. If the host starts a transmitting without waiting to receive the preceding transaction's response, the communication is not considered reliable and the host must send a COMM CLEAR to restore normal communications to the base device.

8.3.5.1.1.1.3 COMM CLEAR

A COMM CLEAR is sent on the RX pin of the base device. It does not send to the stack devices. RX cannot be disabled and a COMM CLEAR can be sent at any time regardless of the TX status. Ensure that the COMM CLEAR does not exceed the maximum value of $t_{UART(CLR)}$ bit periods, as this may result in recognition of other communication pings.

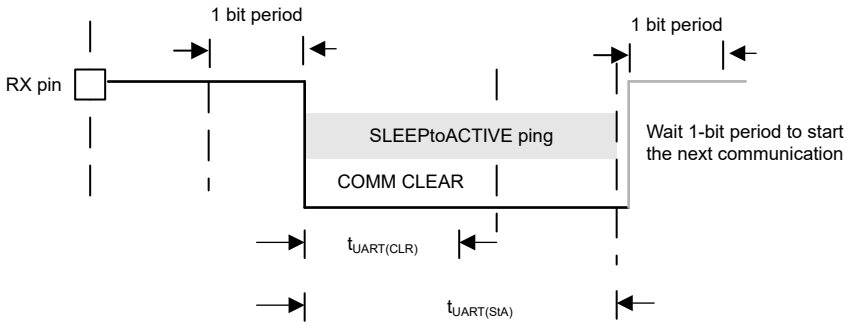


Figure 8-19. UART COMM CLEAR

Use the COMM CLEAR command to clear the receiver and instruct the UART engine to look for a new start of frame. The next byte following the COMM CLEAR is always considered a start-of-frame byte. When detected, a COMM CLEAR sets the *FAULT_COMM1[COMMCLR_DET]* flag. The host must wait at least $t_{UART(RXMIN)}$ after the COMM CLEAR to start sending a new frame. It should be noted that in addition to the *[COMMCLR_DET]* flag, the *FAULT_COMM1[STOP_DET]* flag is also set because the COMM CLEAR timing violates the typical byte timing and the STOP bit is seen as 0.

A SLEEPtoACTIVE ping/tone also clears the UART receiver. This ping/tone sets the *[COMMCLR_DET]* flag when transiting from SLEEP to ACTIVE mode. If this ping/tone is sent during ACTIVE mode, the *[COMMCLR_DET]* and *[STOP_DET]* flags are set.

COMM CLEAR sent during daisy-chain communication:

When a read command is sent, but the response has not yet completely returned to the host, if a COMM CLEAR is received in the base device at this condition, the device response is discarded. In addition, the stack devices do not see the COMM CLEAR and continue to send their responses which are forwarded to the host, resulting in host receiving unexpected response frames. Hence, host should avoid this condition by waiting until all responses are received from the stack before sending a COMM CLEAR.

If the above condition occurs, the base device low-level communication debug register *DEBUG_UART_RR_TR[TR_WAIT]* (indicating device is waiting to transmit response) or *DEBUG_UART_RR_TR[TR_SOF]* (indicating a COMM CLEAR is received while device is transmitting data) bits can be set depending on the timing in receiving the COMM CLEAR signal.

When using the multidrop configuration, a COMM CLEAR signal must be used before every frame to ensure consistent communication.

8.3.5.1.1.2 Command and Response Protocol

The host initiates every transaction between the host and device. The device never transmits data without first receiving a command frame from the host. A command frame is a communication frame sent from host to the device; a response frame is a response (to a read command) from device to host. After a command frame is transmitted, the host must wait for all expected responses to be returned (or a timeout in case of error) before initiating a new command frame. The commands supported by the device are listed in Table 8-9.

Table 8-9. Commands

Command	Description
Single Device Read	To read a register(s) from a single device (base or stack)
Single Device Write	To write a register(s) to a single device (base or stack)
Stack Read	To read a register(s) from the stack devices only. The device must be configured as a stack device with *COMM_CTRL[STACK_DEV]* = 1 to respond to Stack Read commands..
Stack Write	To write a register(s) for only the stack devices. The device must be configured as a stack device with *COMM_CTRL[STACK_DEV]* = 1 to respond to Stack Write commands.
Broadcast Read	To read a register(s) for all of the devices in the daisy-chain, including base device.
Broadcast Write	To write a register(s) for all of the devices in the daisy-chain, including base device.

Table 8-9. Commands (continued)

Command	Description
Broadcast Write Reverse Direction	To send a broadcast write in the reverse direction set by *CONTROL1[DIR_SEL]* bit. This command is intended to be used for switching the communication direction with the RING interface.

8.3.5.1.1.2.1 Transaction Frame Structure

The protocol layer is made up of transaction frames. There are two basic types of transaction frames: command frames (transactions from host) and response frames (transactions from device). The transaction frames are made up of the following five field types:

- Frame initialization (INIT, 1-byte)
- Device address (DEV ADR, 1-byte)
- Register address (REG ADR, 2-byte)
- Data (DATA, various byte length)
- Cyclic redundancy check (CRC, 2-byte)

8.3.5.1.1.2.1.1 Frame Initialization Byte

The frame initialization byte is used in both command and response frames. It is always the first byte of the frame. The frame initialization byte performs two functions. First, it defines the frame as either a command frame (host) or a response frame (device). Second, it defines the length of the frame that follows after the frame initialization byte. This provides the receiver an exact number of bytes to expect for a complete command or response.

Table 8-10. Command Frame Initialization Byte Definition

		Command Frame		Response Frame	
	Bit	**Bit Name**	**Description**	**Bit Name**	**Description**
INIT	7	FRAME_TYPE	1 = Define Command Frame	FRAME_TYPE	0 = Defines Response Frame
	6	REQ_TYPE	000 = Single Device Read 001 = Single Device Write 010 = Stack Read 011 = Stack Write 100 = Broadcast Read 101 = Broadcast Write 110 = Broadcast Write Reverse 111 = RSVD(1)	RESPONSE_BYTE	Number of the data bytes 0x00 = 1 byte 0x01 = 2 bytes : 0x7F = 128 bytes
	5				
	4				
	3	RSVD	Reserved. This bit is ignored		
	2	DATA_SIZE	Number of data bytes of the command frame, excluding device address, register address or CRC 000 = 1 byte 001 = 2 bytes : 111 = 8 bytes		
	1				
	0				

(1) No function to this selection, however, selecting this setting will set the *DEBUG_COMMH[RC_IERR]* or *DEBUG_COMMH[RC_IERR]* flag depends on which daisy-chain interface receives the command frame.

8.3.5.1.1.2.1.2 Device Address Byte

The device address byte identifies the device targeted by the single device read/write command. This byte is omitted for broadcast, stack, and broadcast reverse direction command frames. All response frames contain the device address byte. In single device read/write commands, the device that contains a matching value in the *DIR0_ADDR* (used for communication direction with *CONTROL1[DIR_SEL]* = 0) or in *DIR1_ADDR* (used for communication direction with *CONTROL1[DIR_SEL]* = 1) responds to the command. If multiple devices have matching values, all of those devices will respond and cause collision.

Table 8-11. Device Address Byte Definition

		Command Frame		Response Frame	
	Bit	Bit Name	Description	Bit Name	Description
DEV ADR	7	RSVD	Should always write 0	RSVD	Should always write 0
	6	RSVD	Should always write 0	RSVD	Should always write 0
	5 to 0	Device Address	Set the device address range from 0x00 to 0x3F	Device Address	Set the device address range from 0x00 to 0x3F

8.3.5.1.1.2.1.3 Register Address Bytes

Register addresses are two bytes in length. Any write command to an invalid register address is ignored. Any read from an invalid register returns a 0x00 response. This is true for command frames sent to an individual register with invalid address, or as part of command sent to multiple registers with invalid addresses. When read/write addresses a block of registers with only some invalid addresses, the valid addresses respond as normal, while the invalid addresses respond as previously described.

Table 8-12. Register Address Byte Definition

		Command Frame		Response Frame	
	Bit	Bit Name	Description	Bit Name	Description
REG_ADR	7 to 0	Register Address (MSB)	Target or beginning of the register address	Register Address (MSB)	Target or beginning of the register address
	7 to 0	Register Address (LSB)	Target or beginning of the register address	Register Address (LSB)	Target or beginning of the register address

8.3.5.1.1.2.1.4 Data Bytes

The number of data bytes and the relevant information they convey is determined by the type of command frame sent and the target register specified in that command frame. When part of a command frame, the data bytes contain the values to be written to the registers. When part of a response frame, the data bytes contain the values returned from the registers.

Table 8-13. Data Bytes Definition

		Command Frame		Response Frame	
	Bit	Bit Name	Description	Bit Name	Description
DATA	7	Data Byte[0]	For Write command: Data value to be written to the register(s) is specified in the REG_ADR frame For Read command: Specify the number of bytes need to be returned by the read command. 0x00 = 1 byte 0x01 = 2 bytes : 0x7F = 128 bytes	Data Byte[0]	Data value return from the register(s) is specified in the REG_ADR frame
	6				
	5				
	4				
	3				
	2				
	1				
	0				
	…	……	……	……	……
	7	Data Byte [n]	For Write command: Data value to be written to the register(s) is specified in the REG_ADR frame	Data Byte [n]	Data value return from the register(s) is specified in the REG_ADR frame
	6				
	5				
	4				
	3				
	2				
	1				
	0				

8.3.5.1.1.2.1.5 CRC Bytes

The device uses a CRC (cyclic redundancy check) to protect data integrity during transmission. The CRC represents the remainder of a process analogous to polynomial long division, where the frame being checked is divided by the generator. The CRC appended to the frame is the remainder. Because of this process, when the device receives a frame, the CRC calculated by the receiver across the entire frame including the transmitted CRC will be zero, indicating a correct transmission and reception. A non-zero result indicates a communication error. Specifically, the device uses the CRC-16-IBM polynomial ($x^{16} + x^{15} + x^2 + 1$) with 0xFFFF initialization.

The CRC value is checked as the first step after receiving the communication frame. If the CRC is incorrect, the entire frame is discarded and not processed. Any additional frame errors are not checked and any errors are not indicated other than CRC error. The bytes are still transferred up or down the stack, thus every device that processed the frame will indicate a CRC error. This results in multiple devices indicating CRC faults on the same communication frame.

8.3.5.1.1.2.1.6 Calculating Frame CRC Value

The CRC calculation by the transmitter is in bit-stream order across the entire transmission frame (except for the CRC). When determining bit-stream order for implementing the CRC algorithm, it is important to note that protocol bytes transmit serially, least-significant bit first. Figure 8-20 illustrates the bit-stream order concept.

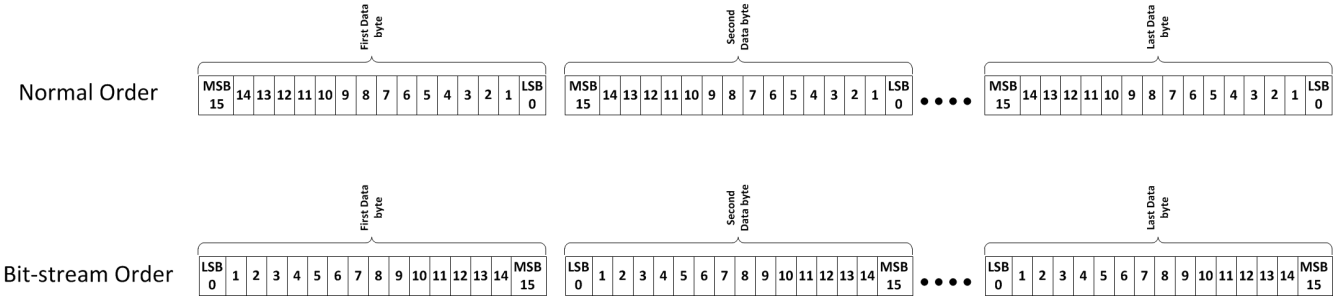


Figure 8-20. Bit-Stream Order Explanation

The CRC (0x0000) is appended to the end of the bit-stream. This bit-stream is then initialized by XOR'ing with 0xFFFF to catch any leading 0 errors. This new bit-stream is then divided by the polynomial (0xC002) until only the 2-byte CRC remains. During this process, the most significant 17 bits of the bit stream are XOR'd with the polynomial. The leading zeroes of the result are removed and that result is XOR'd with the polynomial once again. The process is repeated until only the 2-byte CRC remains. For example:

Example 1: CRC Calculation Using Polynomial Division

```
Command Frame = 0x80 00 02 0F 0B (0b1000 0000 0000 0000 0000 0010 0000 1111 0000 1011)
Command Frame in bit stream order = 0x01 00 40 F0 D0 (0b0000 0001 0000 0000 0100 0000 1111 0000
1101 0000)
After Initialization (XOR with 0xFFFF) = 0b1111 1110 1111 1111 0100 0000 1111 0000 1101 0000
1111 1110 1111 1111 0100 0000 1111 0000 1101 0000 0000 0000 0000 0000 #append 0x0000 for CRC
1100 0000 0000 0010 1 #XOR with polynomial
0011 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0000 0000 0000
11 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0000 0000 0000 #delete leading zeros from
previous result
11 0000 0000 0000 101 #XOR with polynomial
00 1110 1111 1101 0110 0000 1111 0000 1101 0000
……
……
……
1100 0110 0000 0001 0000 0000
1100 0000 0000 0010 1 #XOR with polynomial
0000 0110 0000 0011 1000 0000
110 0000 0011 1000 0000
110 0000 0000 0001 01 #XOR with polynomial
000 0000 0011 1001 0100
0000 0011 1001 0100 #CRC result in bit stream order
1100 0000 0010 1001 #final CRC result in normal order
CRC final 0xC029
```

8.3.5.1.1.2.1.7 Verifying Frame CRC

There are several methods for checking the CRC of a frame. One method is to simply calculate the CRC for the transmitted command except the last two bytes (CRC bytes) using the method described in the previous section, and then compare that result with the transmitted CRC bytes. A more simple option is to run the entire transmission through the CRC algorithm. If the CRC is correct, the result is 0000. In this case, the initial zero padding of the bit-stream with 16 zeroes is not necessary. Using the previous result and running through the algorithm produces the following results:

Example 1: CRC Verification Using Polynomial Division:

```
Command Frame = 0x80 00 02 0F 0B (0b1000 0000 0000 0000 0000 0010 0000 1111 0000 1011)
CRC to Check = 0xC029
Command Frame w/ CRC in bit stream order = 0x80 00 02 0F 0B C0 29 (0b1000 0000 0000 0000 0000 0010
0000 1111 0000 1011 0000 0011 1001 0100)
After Initialization (XOR with 0xFFFF) = 0b0 1111 1110 1111 1111 0100 0000 1111 0000 1101 0000 0000
0011 1001 0100
1111 1110 1111 1111 0100 0000 1111 0000 1101 0000 0000 0011 1001 010 #delete leading zeros from
previous result
1100 0000 0000 0010 1 #XOR with polynomial
0011 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0011 1001 0100
11 1110 1111 1101 1100 0000 1111 0000 1101 0000 0000 0011 1001 0100 #delete leading zeros from
previous result
11 0000 0000 0000 101 #XOR with polynomial
00 1110 1111 1101 0110 0000 1111 0000 1101 0000 0000 0011 1001 0100
……
……
……
1100 0110 0000 0010 1001 0100
1100 0000 0000 0010 1 #XOR with polynomial
0000 0110 0000 0000 0001 0100
1 1000 0000 0000 0101 00
1 1000 0000 0000 0101 #XOR with polynomial
0 0000 0000 0000 0000 00
0x0000 #verfiy that CRC checks out valid
```

Note

The result of ‘0b0000 0000 0000 0000’ for the CRC indicates a successful check.

8.3.5.1.1.2.2 Transaction Frame Examples

Transaction frames are created using the frame structure discussed in the previous sections. This section outlines how the command and response frames are passing through the daisy-chain. The CRC values in the examples are correct and can be used to verify the customer CRC algorithm. The CRC is verified by the device with every received command frame and the command is not executed unless the CRC is valid.

8.3.5.1.1.2.2.1 Single Device Read/Write

Single Device Read:

Device address must be set up before using this command. A single device read generates a response frame whose length depends on the requested number of register bytes read. The command frame send by host must contain the register address to start at (address field) and the number of bytes to return (number of registers to read). The DATA_SIZE field in the initialization byte for the single device read command is always 0b000.

The command frame travels to all devices in the daisy-chain, but only the device that matches the command frame’s device address field will respond to the single device read command. The corresponding device will respond with returned data request by the single device read, following the response frame format.

Single Device Write:

Device address must be set up before using this command. A write command for a single device enables the customer to update up to eight consecutive registers with one command. The single device write command frame must contain the register address to start at (address field) and the data bytes to write to the registers.

The DATA_SIZE field in the initialization byte for the single device write command is the number of registers to update.

The command frame travels to all devices in the daisy-chain, but only the device that matches the command frame's device address field will execute the single device write command.

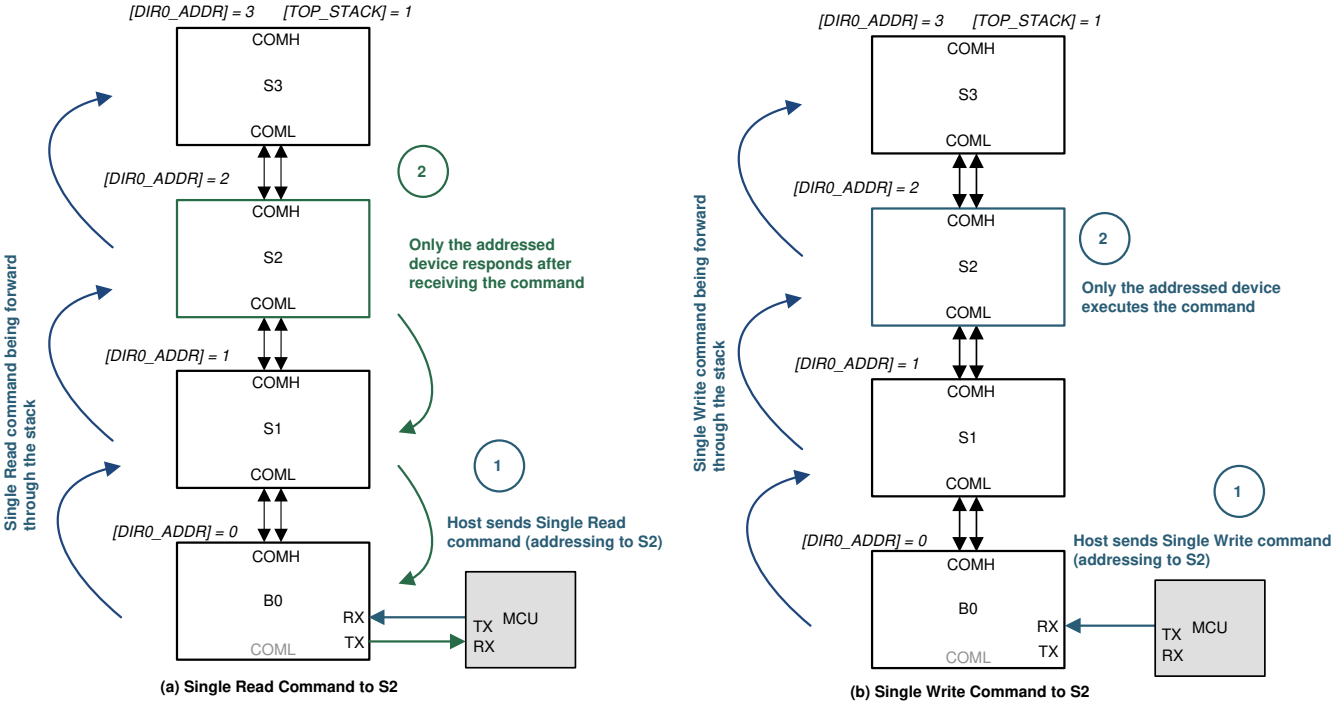


Figure 8-21. Single Device Read/Write

Table 8-14. Single Device Read/Write

	Single Read Command Sent by Host		Single Write Command Sent by Host	
Example	**Read 16 VS Voltages from S2**		**Write OTP Unlock Code to OTP_PROG_UNLOCK1A to 1D Registers**	
Frame Field	**Data**	**Comments**	**Data**	**Comments**
Initialization Byte	0x80	Always 0x80 FRAME_TYPE = 1 REQ_TYPE = 0b000 = Single Read DATA_SIZE = 0b000	0x93	0x90 for 1 byte data read, 0x91 for 2 bytes data read, 0x92 for 3 bytes data read, and so on. For this example: FRAME_TYPE = 1 REQ_TYPE = 0b001= Single Write DATA_SIZE = 0b11 = 4 bytes
Device Address	0x02	Device address 0x02 (S2) in this example	0x02	Device address 0x02 (B0) in this example
Register Address	0x0568	Start address of the register block to read (address of VS16_HI in this example)	0x0300	Start address of the register block to write (address of OTP_PROG_UNLOCK1A in this example)
Data	0x1F	Instruct the target device to return 32 bytes of data (that is, from address 0x0568 to 0x0587), assuming each VSn_HI = 0x80, VSn_LO = 0x00, where n = 1 to 16.	0x02B7 78BC	The unlock value to OTP_PROG_UNLOCK1A to OTP_PROG_UNLOCK1D
CRC	0x5A6F		0xB8AE	

8.3.5.1.1.2.2.2 Stack Read/Write

Stack Read:

The device address, *COMM_CTRL[STACK_DEV]* bit and *[TOP_STACK]* bit must be configured before using this command. A stack read command generates a number of response frames depending on the number of devices in the stack (that is, device with *COMM_CTRL[STACK_DEV]* = 1), whose length depends on the requested number of register bytes read. The stack read command frame must contain the register address to start at (address field) and the number of bytes to return (number of registers to read). The DATA_SIZE field in the initialization byte for the read command is always 0b000.

The command frame travels to all devices in the daisy-chain, but only the device with *COMM_CTRL[STACK_DEV]* = 1 will respond. During the response, the device with *COMM_CTRL[TOP_STACK]* = 1 will return the response frame first. Each device (address N) in the stack waits until the device above (address N+1) responds before appending its response frame. The CRC is validated while receiving the responses. If a CRC error occurs in the response frame from address N+1, device N does not append its message and an invalid CRC fault is generated.

Use Figure 8-22 with the example of using reading 16 VS voltages from S1 to S3. The response to this command is three separate response frames (one response frame per device), each frame with a total length of 38 bytes (32 data bytes + 6 protocol bytes). Although the stack read command does not contain the device address field, each response frame will contain the corresponding device address field associating the data to a particular device. The host will receive a response frame from S3 first (ToS), following with a response frame from S2, and finally the response frame from S1.

Stack Write:

The *COMM_CTRL[STACK_DEV]* must be configured before using this command. A stack write command enables the host to update up to eight consecutive registers for the stack devices (that is, device with *COMM_CTRL[STACK_DEV]* = 1) with one command. The command frame must contain the register address to start at (address field) and the data bytes to write to the registers. The DATA_SIZE field in the initialization frame is the number of registers to update.

The command frame travels to all devices in the daisy-chain, but only the device with *COMM_CTRL[STACK_DEV]* = 1 will execute the command.

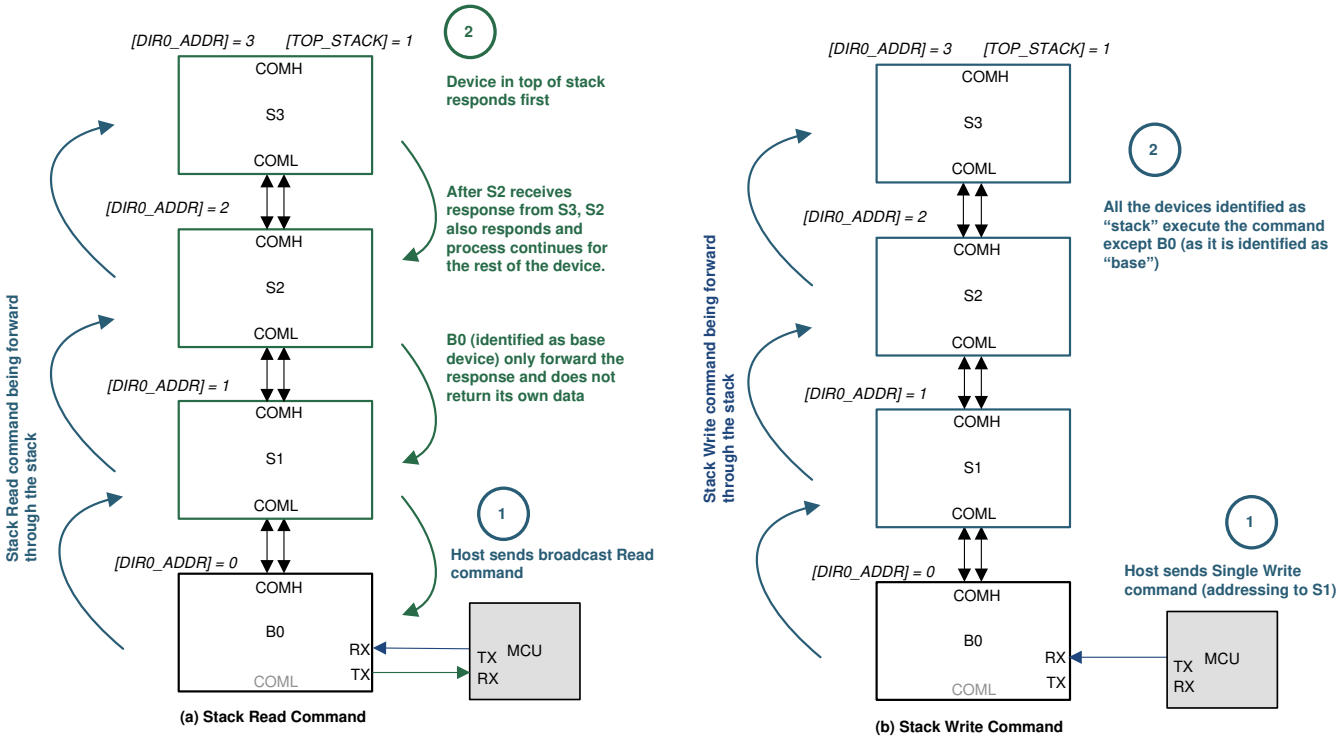


Figure 8-22. Stack Read/Write

Table 8-15. Stack Read/Write

	Stack Read Command Sent by Host		Stack Write Command Sent by Host	
Example	**Read 16 VS Voltages from S1 to S3**		**Write OTP Unlock Code to OTP_PROG_UNLOCK1A to 1D Registers to S1, S2, and S3**	
Frame Field	**Data**	**Comments**	**Data**	**Comments**
Initialization Byte	0xA0	Always 0xA0 FRAME_TYPE = 1 REQ_TYPE = 0b010 = Stack Read DATA_SIZE = 0b000	0xB3	0xB0 for 1 byte data read, 0xB1 for 2 bytes data read, 0xB2 for 3 bytes data read, and so on. For this example: FRAME_TYPE = 1 REQ_TYPE = 0b011= Stack Write
				DATA_SIZE = 0b011 = 4 bytes
Device Address	N/A	No need to include the device address byte in command frame	N/A	No need to include the device address byte in command frame
Register Address	0x0568	Start address of the register block to read (address of VS16_HI in this example)	0x0300	Start address of the register block to write (address of OTP_PROG_UNLOCK1A in this example)
Data	0x1F	Instruct each device to return 32 bytes of data (that is, from address 0x0568 to 0x0587), assuming each VSn_HI = 0x80, VSn_LO = 0x00, where n = 1 to 16.	0x02B7 78BC	The unlock value to OTP_PROG_UNLOCK1A to OTP_PROG_UNLOCK1D
CRC	0x5C2D		0x0BD7	

8.3.5.1.1.2.2.3 Broadcast Read/Write

Broadcast Read:

The device address and *[TOP_STACK]* bit must be configured before using this command. A broadcast read command generates a number of response frames depending on the number of devices in the daisy-chain (both stack and base devices), whose length depends on the requested number of register bytes read. The broadcast read command frame must contain the register address to start at (address field) and the number of bytes to return (number of registers to read). The DATA_SIZE field in the initialization byte for the read command is always 0b000.

The command frame travels to all devices in the daisy-chain, every device will respond. During the response, the device with *COMM_CTRL[TOP_STACK]* = 1 will return the response frame first, each device (address N) in the stack waits until the device above (address N+1) responds before appending its response frame. The CRC is validated while receiving the responses. If a CRC error occurs in the response frame from address N+1, device N does not append its message and an invalid CRC fault is generated.

Use Table 8-16 with the example of reading 16 VS voltages from B0 to S3. The response to this command is four separate response frames (one response frame per device), each frame with a total length of 38 bytes (32 data bytes + 6 protocol bytes). Although the broadcast read command does not contain the device address field, each response frame will contain the corresponding device address field, associated the data to a particular device. The host will receive the response frame from S3 first (ToS), following with the response frame from S2, then S1, and finally the response frame from B0.

Broadcast Write:

This command can be used without auto-addressing. A broadcast write command enables the host to update up to eight consecutive registers for all devices in the daisy-chain with one command. The command frame must contain the register address to start at (address field) and the data bytes to write to the registers. The DATA_SIZE field in the initialization frame is the number of registers to update.

The command frame travels to all the devices in the daisy-chain, and every devices in the daisy-chain will execute the command.

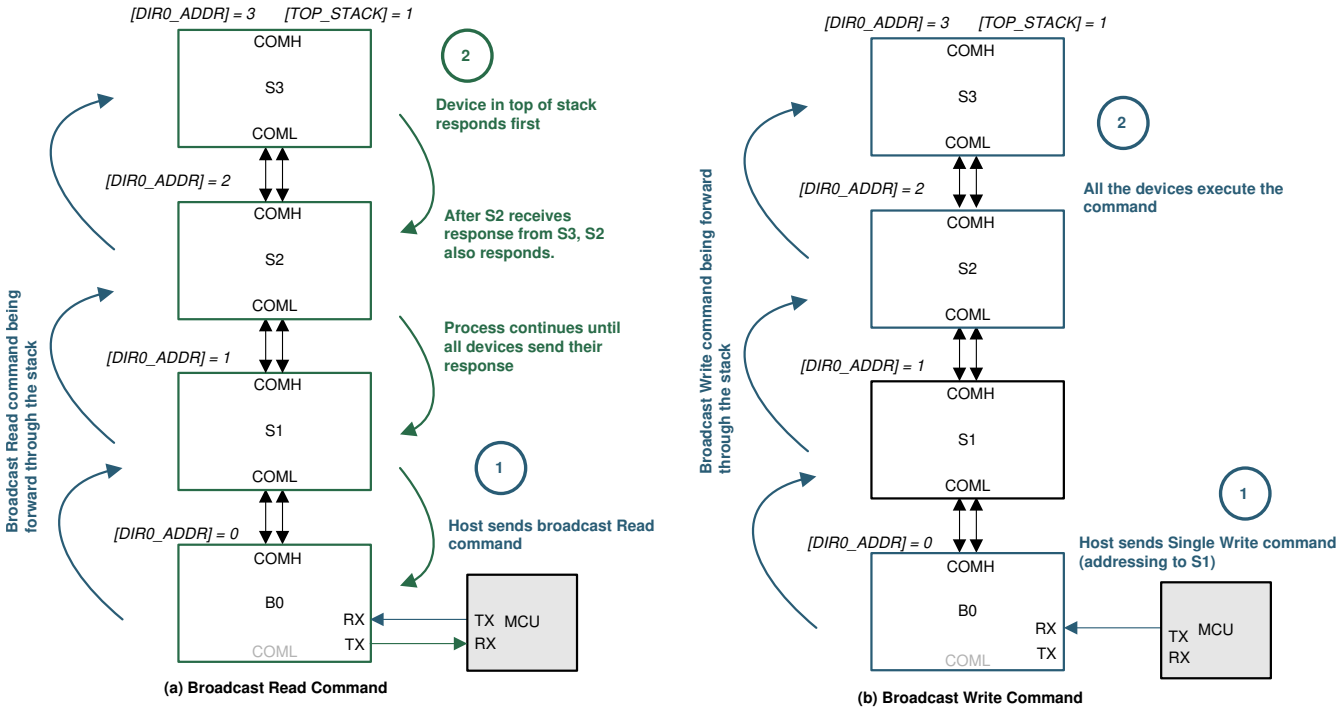


Figure 8-23. Broadcast Read/Write

Table 8-16. Broadcast Read/Write

	Broadcast Read Command Sent by Host		Broadcast Write Command Sent by Host	
Example	**Read 16 VS Voltages from B0 to S3**		**Write OTP Unlock Code to OTP_PROG_UNLOCK1A to 1D Registers to B0, S1, S2, and S3**	
Frame Field	**Data**	**Comments**	**Data**	**Comments**
Initialization Byte	0xC0	Always 0xC0 FRAME_TYPE = 1 REQ_TYPE = 0b100 = Broadcast Read DATA_SIZE = 0b000	0xD3	0xD0 for 1 byte data read, 0xD1 for 2 bytes data read, 0xD2 for 3 bytes data read, and so on. For this example: FRAME_TYPE = 1 REQ_TYPE = 0b101= Broadcast Write DATA_SIZE = 0b011 = 4 bytes
Device Address	N/A	No need to include the device address byte in command frame	N/A	No need to include the device address byte in command frame
Register Address	0x0568	Start address of the register block to read (address of VS16_HI in this example)	0x0300	Start address of the register block to write (address of OTP_PROG_UNLOCK1A in this example)
Data	0x1F	Instruct each device to return 32-bytes of data (that is, from address 0x0568 to 0x0587), assuming each VSn_HI = 0x80, VSn_LO = 0x00, where n = 1 to 16.	0x02B7 78BC	The unlock value to OTP_PROG_UNLOCK1A to OTP_PROG_UNLOCK1D
CRC	0x422D		0x6BD1	

8.3.5.1.1.2.2.4 Broadcast Write Reverse Direction

Usually, device is expecting to receive communication based on the *[DIR_SEL]* setting. If a device receives communication frame opposite to the *[DIR_SEL]* setting, such as receiving command frame from COMH while *[DIR_SEL]* = 0, it will flag the communication as error. The broadcast write reverse direction is a command used to change flip the *[DIR_SEL]* setting when host needs to switch the daisy-chain communication direction. This command is expected to receive from an opposite direction than the [DIR_SEL] setting during reverse communication direction procedure. See Section 8.3.5.1.3.4 for details.

Although the broadcast write reverse direction is allowed to write any register value to the device, it is not recommended to write any other register setting other than the *CONTROL1[DIR_SEL]* to avoid communication collisions. Communication collisions are not detected and result in corrupted communication on the stack interface.

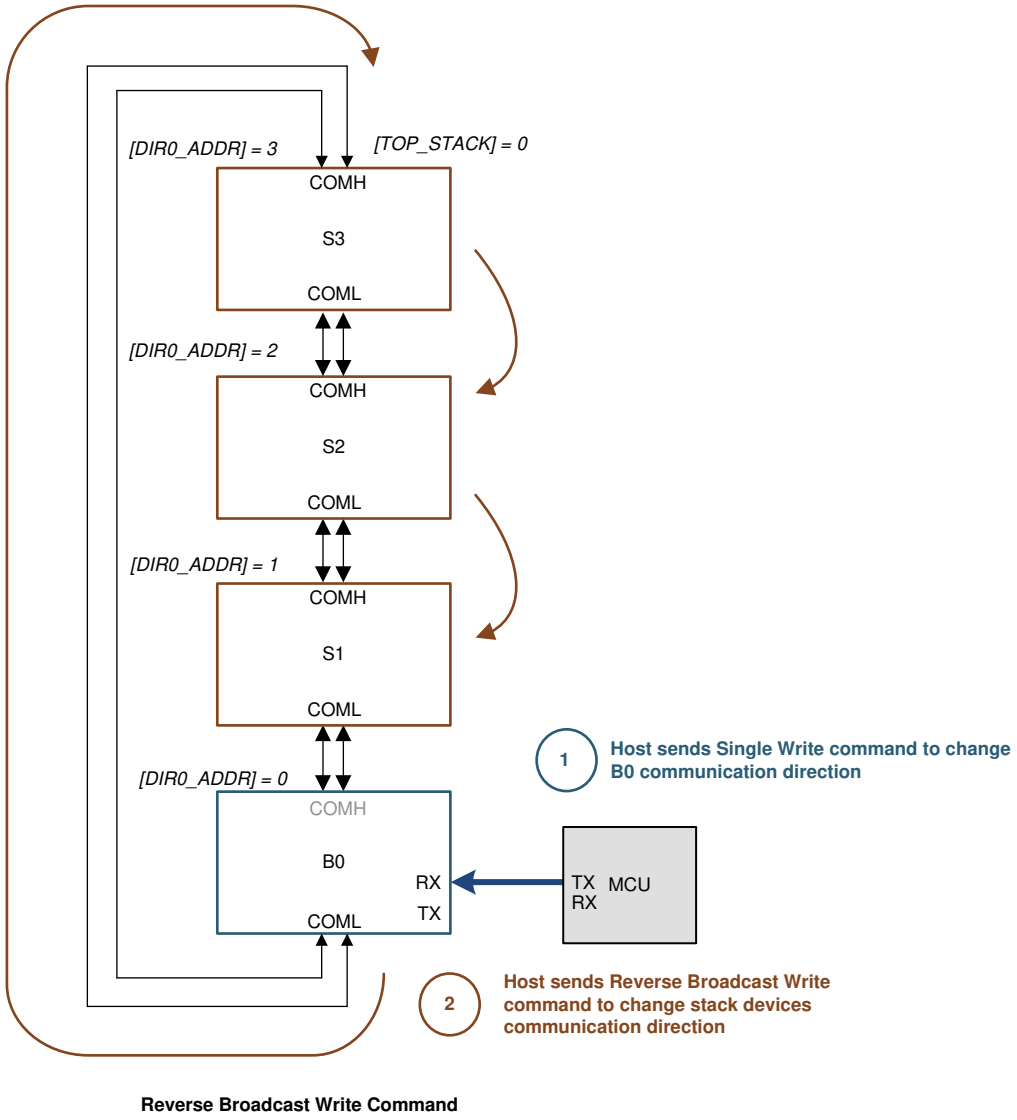


Figure 8-24. Broadcast Write Reverse Direction

Table 8-17. Broadcast Write Reverse Direction

	Broadcast Write Reverse Direction Command Sent by Host	
Example	**Set the *[DIR_SEL]* = 1 on All Devices in the Daisy-Chain**	
Frame Field	**Data**	**Comments**
Initialization Byte	0xE0	Always 0xE0 FRAME_TYPE = 1 REQ_TYPE = 0b110 = Broadcast Write Reverse Direction DATA_SIZE = 0b000
Device Address	N/A	No need to include the device address byte in command frame
Register Address	0x0309	Address of *CONTROL1* register
Data	0x80	Set *CONTROL1[DIR_SEL]* = 1

Table 8-17. Broadcast Write Reverse Direction (continued)

	Broadcast Write Reverse Direction Command Sent by Host	
Example	**Set the *[DIR_SEL]* = 1 on All Devices in the Daisy-Chain**	
Frame Field	**Data**	**Comments**
CRC	0xC014	

8.3.5.1.2 Daisy-Chain Interface

The daisy-chain communication is created using differential signaling to minimize Electro-Magnetic Susceptibility (EMS) and Bulk Current Injection (BCI) immunity. The differential communication transmits true and complement data on the COM*P and COM*N pins, respectively. In a multiple device stack, there are configurations where the devices are physically located on the same board or located in entirely separate packs connected with twisted-pair wiring.

The device supports the use of transformers or capacitors to electrically isolate the signals between devices in the stack. For applications that have multiple devices on the same PCB, a single level-shifting capacitor is connected between the COMH/L pins of the devices. For extremely noisy environments, additional filtering may be necessary. For devices that are separated by cabling, additional isolation components are used. See Section 9 for specific details on selecting components.

8.3.5.1.2.1 Daisy-Chain Transmitter and Receiver Functionality

The daisy-chain is bi-directional and half duplex, and, therefore, has a transmitter (TX) and receiver (RX) on the COMH and COML interfaces. The TX and RX functions are controlled automatically by the hardware based on the device's base/stack detection. When a WAKE ping/tone is received, the communication direction is set by *CONTROL1[DIR_SEL]* and the *COMM_CTRL[TOP_STACK]* configurations. See Section 8.3.5.1.3 for details. Additionally, a user overwrite to take over the complete control of the COMH and COML is available under communication debug mode using the *DEBUG_CTRL_UNLOCK*, *DEBUG_COMM_CTRL1*, and *DEBUG_COMM_CTRL2* registers. See Section 8.5.4.13 for details.

8.3.5.1.2.2 Daisy-Chain Protocol

The differential daisy-chain (vertical) interface uses an asynchronous 13-bit byte-transfer protocol. Data is transferred LSB first and every bit is duplicated (with a complement) to ensure the transmission has no DC content.

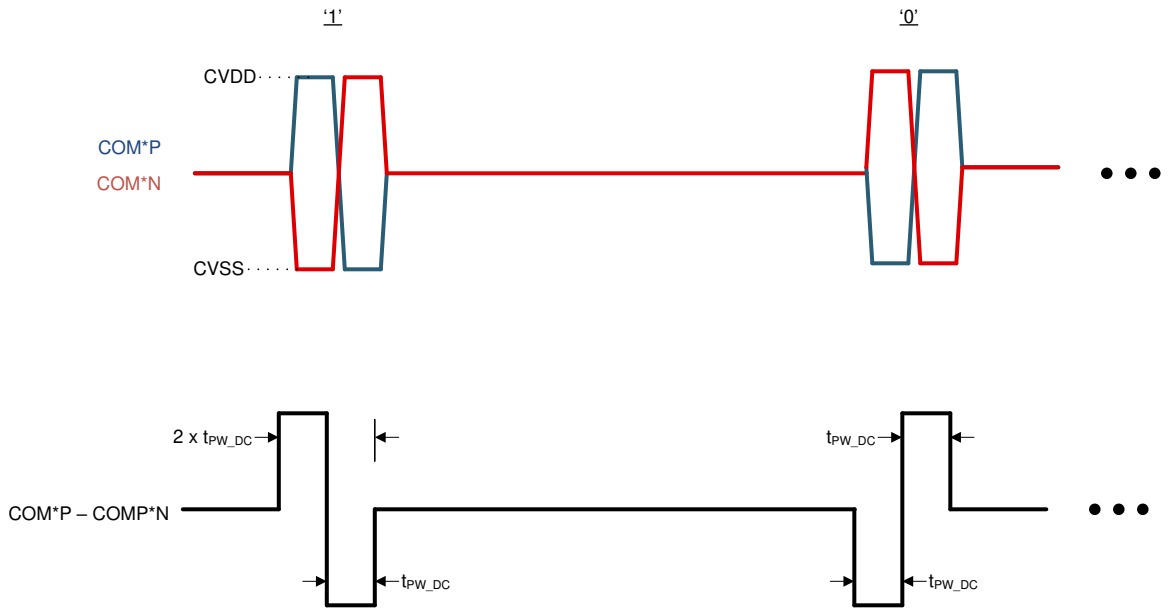


Figure 8-25. Daisy-Chain Bit Definition

A byte starts with a Preamble, followed by two SYNC bits, a start-of-frame bit, eight data bits starting from the LSB D0 to MSB D7 (D0 is transmitted just after State-Of-Frame and D7 comes last before the Byte Error and Postamble).

The device extracts timing information using the Preamble and SYNC bits to decode the rest of the bit value in the byte. If any of the following errors is detected, the byte is not processed and register error bit is set.

- The Preamble and SYNC bits are known values, if the decoded value has error, the *DEBUG_COMH/L_BIT[SYNC1]* = 1 depends on which COM port receives this data.
- If timing extracted from the Preamble and SYNC bits is outside of the expected range, the *DEBUG_COMH/L_BIT[SYNC2]* = 1.

Once the two valid SYNC bits are received, the additional bits are decoded and sent to the command processor. The device continues to detect any error on this byte, and if error is detected, the Byte Error (BERR) bit will be set in this byte. The *DEBUG_COMH/L_BIT[PERR]* = 1 depends on which COM port detects the error. The following condition will set the BERR bit in the byte.

- Not sufficient samples to indicate the logic level of a bit. That is, a bit is decoded as not a strong 1 or strong 0. The *DEBUG_COMH/L_BIT[BIT]* = 1 depends on which COM port detects the error.

In the meantime, each bit is still being retransmitted to the next device. If the device is unable to decode a 1 or a 0 for the bit, it will retransmit with 0 with the BERR bit set in the byte. When the new device detects the BERR bit is set to 1 in the receiving byte, it will ignore the questionable byte and set the *DEBUG_COMH/L_BIT[BERR_TAG]* = 1, indicating a byte is received with BERR. The questionable byte being ignored is likely to cause other communication errors and is likely to trigger the *DEBUG_COMH/L_BIT[PERR]* = 1 being set in the new device as well. The questionable byte continues to be retransmitted up the daisy-chain with BERR set and the process continues.

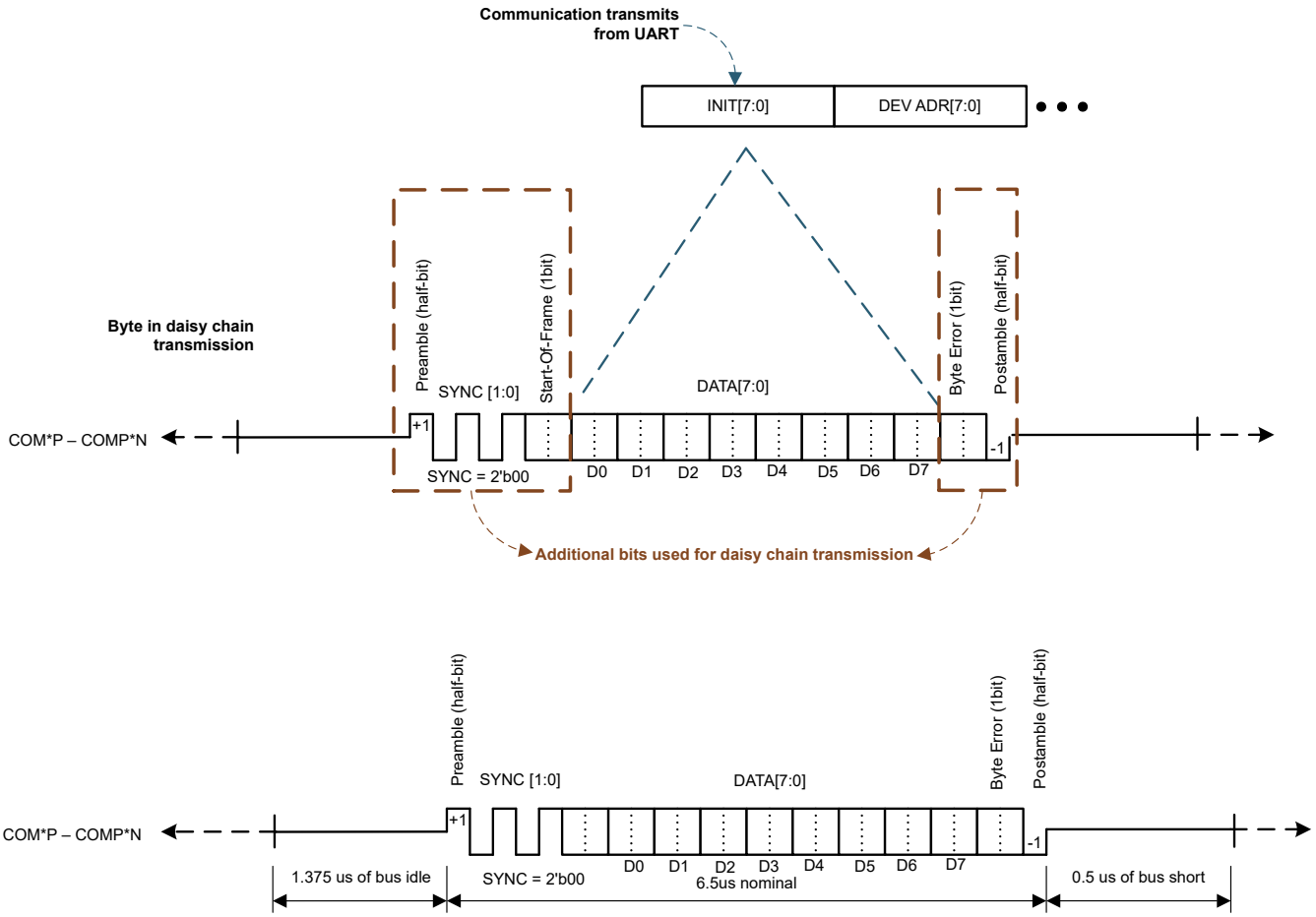


Figure 8-26. Daisy-Chain Byte Definition

Table 8-18. Daisy-Chain Byte Definition

Bit Field	Description
Preamble (half-bit)	Indicates a start of transaction, signaling the receiver to start sampling. This half-bit and the following two SYNC bits are used to extra timing information.
SYNC[1:0]	Always 0b00. The SYNC bits are used for the digital to assess the timing and noise level on the byte, improving the detection of a 1 and 0 in a noisy environment.
Start-Of-Frame (1-bit)	The Start-Of-Frame (SOF) bit indicates the follow-on data byte is the initialization byte, a start of a communication transaction frame. Stack device needs this information to process the communication. For command frame transaction, the base device is responsible to set the SOF bit as it translates the UART communication to the daisy-chain communication. The initialization byte contains data size information. Based on the data size information, the base device would count the number of bytes received and set the next SOF bit accordingly. The UART COMM CLEAR signal resets the UART receiver which includes the frame handling of the logic. Hence, the next byte after COMM CLEAR must have SOF set to 1 because the COMM CLEAR indicates the system clears UART and re-starts the communication.
Data[7:0]	The actual byte of the communication transaction frame
Byte Error BERR (1-bit)	Indicates an error detected in this byte. When a device receives a byte with BERR set by the lower device, it will retransmit the byte also with *BERR* = 1. Because each data bit is re-clocked from one device to the next, the next device may not detect a communication error. However, the tag of the *[BERR]* bit would indicate this communication frame has an error during its previous transaction.
Postamble (half-bit)	Indicates the end of transaction

Each byte is transmitted at 2 MHz (250 ns per pulse or 500 ns per couplet). The time between each byte depends on the UART baud rate (1 Mbps in normal operation), but the byte time is always the same. The communication frame is defined with idle time between byte. In some rare cases, communication signal may not terminate cleanly, leaving ringing at the end of a byte. In such case, increasing the byte to byte gap can improve the communication robustness. The device allows additional byte gap insert between bytes in the response frame through *STACK_RESPONSE* register setting.

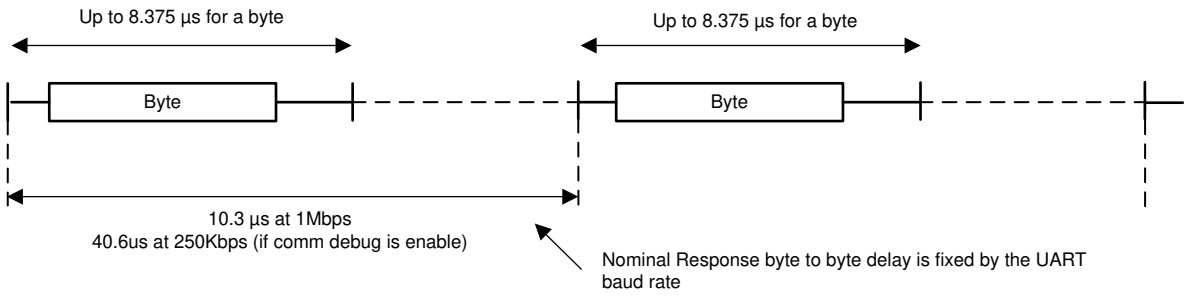


Figure 8-27. Daisy-Chain Byte Transfer

8.3.5.1.3 Start Communication

From SHUTDOWN or after device reset, host follows the following steps to bring up the devices for communication.

- Host sends a WAKE ping to reset or bring the devices to ACTIVE mode. In this process, the devices in the daisy-chain will configure their own COMH and COML ports based on their position in the daisy-chain (base device or stack device)
  - After this step, the broadcast write is supported.
- Host performs auto-addressing to assign a device address to each device
  - After this step, the broadcast read/write and single device read/write are supported.
- Host configures the *COMM_CTRL[STACK_DEV]* and *[TOP_STACK]* bits. The Top of Stack (ToS) device will disable its transmitter of the COMH (or COML based on communication direction)
  - After this step, all commands, broadcast read/write, single device read/write, and stack read/write are supported.

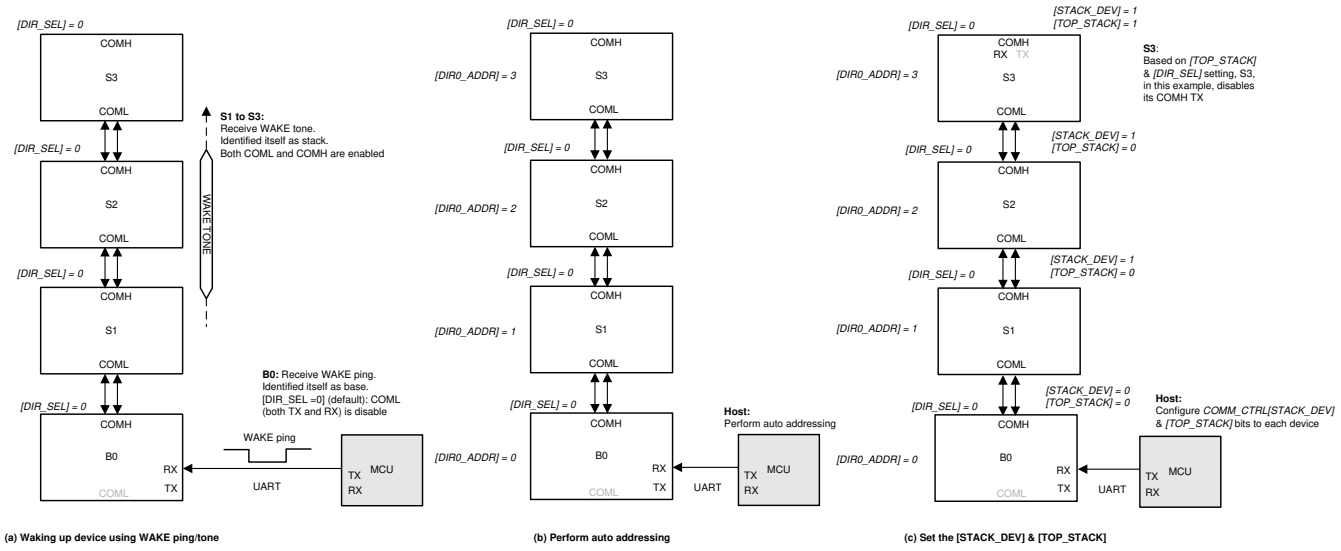


Figure 8-28. Configure Device for Communication

8.3.5.1.3.1 Identify Base and Stack

A WAKE ping/tone is used for the device to identify its position in the daisy-chain.

- Base device: a device interfaces with host through UART
- Stack device: a device interfaces with the base device through COMH and COML

A base device will be woke up by a WAKE ping through RX pin, while a stack device will be woke up by WAKE tone via the COMH/COML port. Hence, a device is using a WAKE ping or WAKE tone to identify itself as base or stack. This information is stored in the AVAO_REF block which is available in all power modes and is refreshed whenever a WAKE ping/tone is received.

Using the *CONTROL1[DIR_SEL]* setting, a base device will disable the unused daisy-chain ports (transmitter and receiver). If host changes the *CONTROL1[DIR_SEL]* setting, the base device will reconfigure its COMH/COML.

Note

The host starts communication at least 100 µs after changing the *[DIR_SEL]* setting to ensure the device finishes the COMH/COML reconfiguration.

8.3.5.1.3.2 Auto-Addressing

Every device must have a unique device address for the read protocol to work. If, for any reason, two devices are assigned with the same device address, it is likely that broadcast and stack reads do not work. Additionally, single device read to the doubled address results in destroyed communication.

The default device address, assuming the device address in OTP is not programmed, is 0x00. For a host to talk to a standalone device (that is, a stack consisting with only one device), host can simply use the default 0x00 device address. Otherwise, device address follows the rules below:

- Base device address can start with any value, it is not necessary for it to be 0x00
- All device addresses must be sequential. That is, if base is 0x00, the next device must be 0x01, and next must be 0x02, and so on.

Before starting the auto-addressing procedure, all devices must be in ACTIVE mode. In this state, the device will only be able to process broadcast write command, which will be the command used for the auto-addressing procedure. Based on the *CONTROL1[DIR_SEL]* setting, the auto-addressing procedure sets up the device address to either *DIR0_ADDR* register (when *[DIR_SEL]* = 0) or *DIR1_ADDR* register (when *[DIR_SEL]* = 1).

8.3.5.1.3.2.1 Setting Up the Device Addresses

The *CONTROL1[ADDR_WR]* bit enables the auto-addressing mode. In this mode, the device turns off its COMH/COML (depends on the *[DIR_SEL]* setting) transmitter for one communication frame (following the auto-addressing procedure, that will be its own device's address), clear the *CONTROL1[ADDR_WR]* = 0. When the next communication is received (following the auto-addressing procedure, it will be the next device's address), the device will forward the communication to the next device.

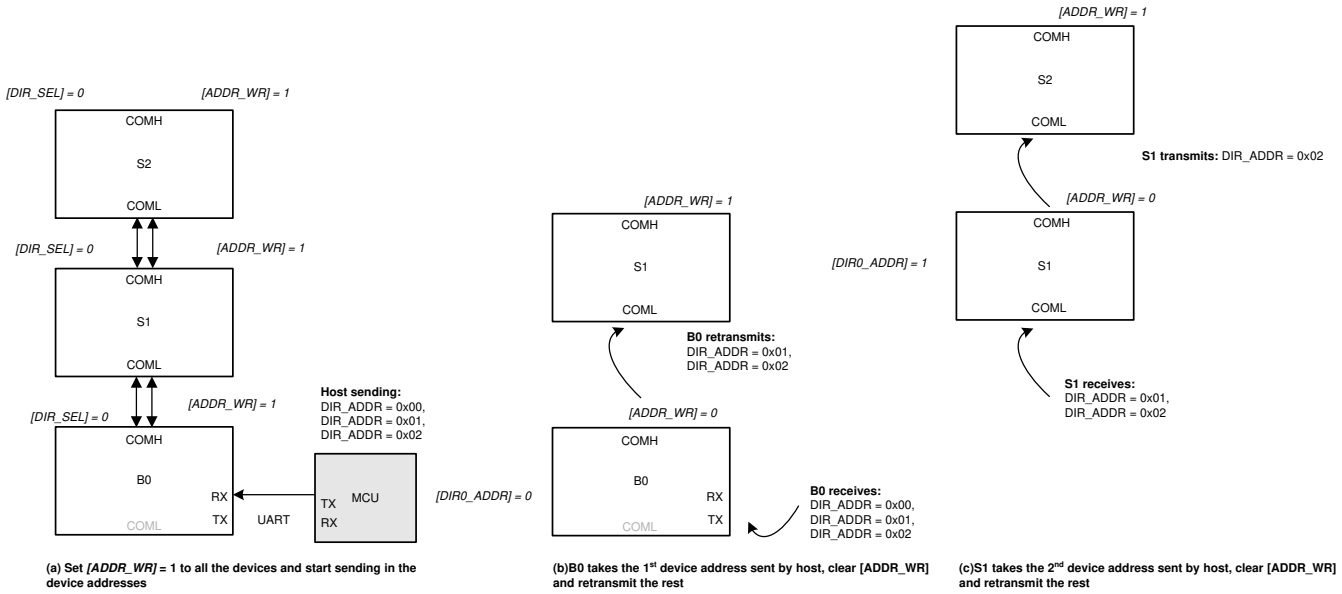


Figure 8-29. Auto-Addressing

8.3.5.1.3.2.2 Setting Up COMM_CTRL[STACK_DEV] and [TOP_STACK]

The last procedure in the auto-addressing is to configure the *COMM_CTRL[STACK_DEV]* and *[TOP_STACK]* settings. These bits need to be configured for the broadcast read and stack read/write to work properly.

- Base device: *[STACK_DEV]* = 0 and *[TOP_STACK]* = 0
- Stack devices (except ToS device): *[STACK_DEV]* = 1 and *[TOP_STACK]* = 0
- ToS device: *[STACK_DEV]* = 1 and *[TOP_STACK]* = 1

Table 8-19 shows the auto-addressing steps, assuming *CONTROL1[DIR_SEL]* = 0 (that is, each device will be set up to transmit command frame sent by host from its COML to COMH).

Table 8-19. Auto-Addressing

Step	Procedure
1	This step is required if a device reset has occurred before performing the auto-addressing procedure. Dummy Write to synchronize all daisy-chain devices DLL (delay-locked loop) ramp in write direction. Host sends broadcast write to write 0x00 to *ECC_DATA1* to *ECC_DATA8* registers.
2	Enable auto-addressing procedure. Host sends broadcast write to set *CONTROL1[ADDR_WR]* = 1.
3	Sending in the device addresses. Host sends broadcast write to set the consecutive addresses to *DIR0_ADDR[ADDRESS5:0]*. With an example of a total of three devices in a daisy-chain: 1. Send broadcast write to *DIR0_ADDR* register with data 0x00. 2. Send broadcast write to *DIR0_ADDR* register with data 0x01. 3. Send broadcast write to *DIR0_ADDR* register with data 0x02.
4	Set up the *COMM_CTRL[STACK_DEV]* and *[TOP_STACK]* bits for each device. Option 1: Host sends single device write to each device to set the proper *[STACK_DEV]* and *[TOP_STACK]* values. Option 2 (less communication steps): 1. Host sends broadcast write to set *[STACK_DEV]* = 1 and *[TOP_STACK]* = 0. 2. Host sends single device write to base device (device address 0x00 in this example) with *[STACK_DEV]* = 0. 3. Host send single device write to the ToS device address 0x02 in this example) with *[TOP_STACK]* = 1.
5	This step is required if a device reset has occurred before performing the auto-addressing procedure. Dummy read to synchronize all daisy-chain devices DLL ramp in read direction. Host sends broadcast read to read *ECC_DATA1* to *ECC_DATA8* registers. Host may not receive all of the data as this step synchronizes the DLL.
6	Recommended as good practice. Use broadcast read to read *DIR0_ADDR* registers to read back all device addresses to ensure all devices are addressed properly.

Table 8-19. Auto-Addressing (continued)

Step	Procedure
7	If the dummy write and dummy read steps are performed to synchronize the DLL, it is normal if communication fault is triggered. Clear the fault registers if that is the case.

8.3.5.1.3.2.3 Storing Device Address to OTP

The device uses *DIR0_ADDR* (used with *[DIR_SEL]* = 0) and *DIR1_ADDR* (used with *[DIR_SEL]* = 1) registers for its device address. In the auto-addressing procedure, device address is written to one of these registers and the new device address takes effect immediately.

The host has an option to program the device addresses for the *[DIR_SEL]* = 0 and 1 directions to the OTP, allowing the programmed addresses to be loaded whenever the device is reset. To program the device address to OTP, host writes the desired address to the OTP shadow registers, *DIR0_ADDR_OTP* (used when *[DIR_SEL]* = 0) and *DIR1_ADDR_OTP* (used when *[DIR_SEL]* = 1) and performs OTP programming. These two shadow registers only reflect the value programmed in OTP or use for the host to program the desired value to OTP. These two shadow registers are not the device address setting during communication. See Section 8.3.5.3.2 for programming details.

8.3.5.1.3.3 Synchronize Daisy-Chain DLL

When device is reset or enter ACTIVE from SLEEP. MCU should perform dummy write and read to synchronize the DLL on the daisy-chain devices.

In the device reset case, if device address is not programmed in OTP. MCU must perform an auto-address. The DLL synchronization is part of the step. If device address is programmed in OTP, auto-address is not required after device reset. However, MCU should perform a dummy write and dummy read steps shown in Table 8-19, step1 and step5 to synchronize the DLL.

When device goes from SLEEP to ACTIVE using SLEEPtoACTIVE signal, the device is not reset. However, it is recommend to do a 1-data-byte dummy write and read to ensure robustness. Follow the similar dummy write and read steps in Table 21, but only write and read to *OTP_ECC_DATAIN1*.

8.3.5.1.3.4 Ring Communication

The daisy-chain communication for the device uses a Ring architecture. In this architecture, a cable break between two devices does not prevent communication to all upstream devices as in a normal non-Ring scheme. When the host detects a broken communication, the device allows the host to switch the communication direction to communicate with devices on both sides of the break. This allows for safe operation until the break in the lines is repaired.

The *CONTROL1[DIR_SEL]* controls the communication direction. The devices will reconfigure the COMH and COML ports depending on the *[DIR_SEL]* and the *[TOP_STACK]* settings. Auto-addressing procedure is needed to re-address the device address for the reverse communication direction.

Example to change the communication direction to *[DIR_SEL]* = 1 to the entire daisy-chain:

1. Host sends Single Device Write to change the base device *[DIR_SEL]* = 1. The base device will disable its COMH and enable its COML.
2. Host sends Broadcast Write Reverse Direction to clear the *COMM_CTRL* register settings on all devices.
3. Host sends Broadcast Write Reverse Direction to change the rest of the devices' *[DIR_SEL]* = 1. In this step, the entire daisy-chain set up to transmitting communication in the *[DIR_SEL]* = 1 direction (that is, each device set up to transmit command frames sent by host from its COMH to its COML).
4. Host performs auto-addressing procedure to set up device address in the *DIR1_ADDR* register. Unless the devices have been reset, host can skip the dummy read/write steps to synchronize the DLL in the auto-addressing procedure.
5. Host sets up the new Top of Stack device and the new ToS device will disable its COML transmitter.

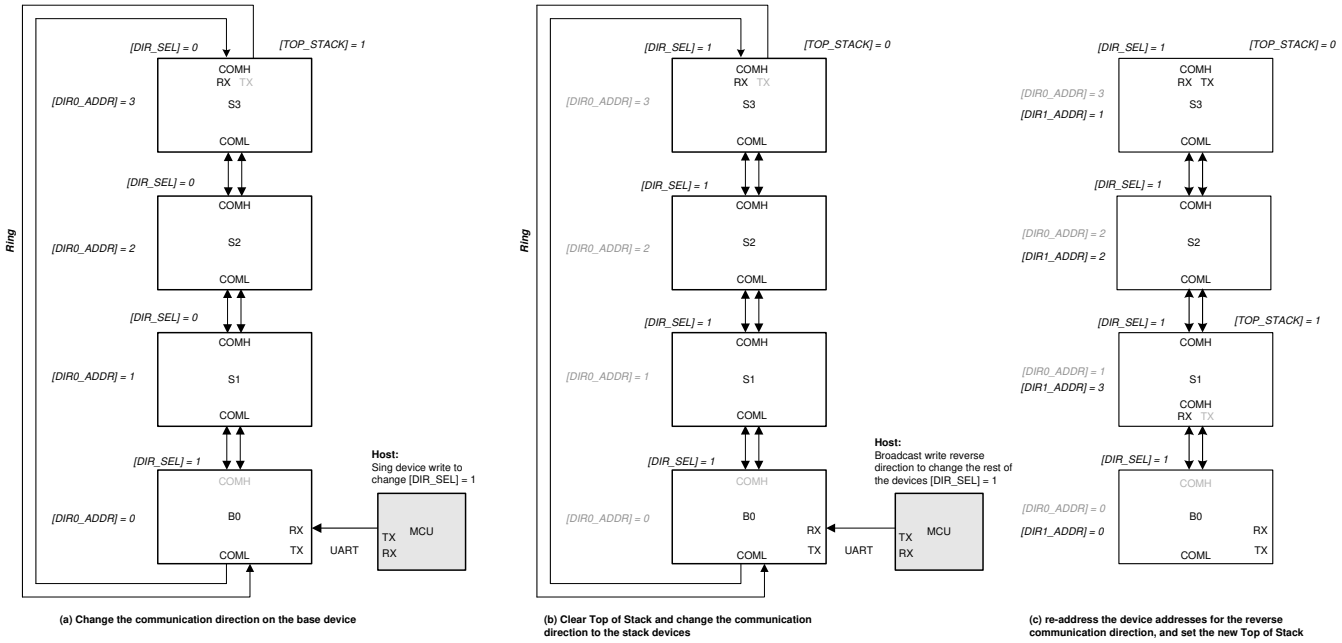
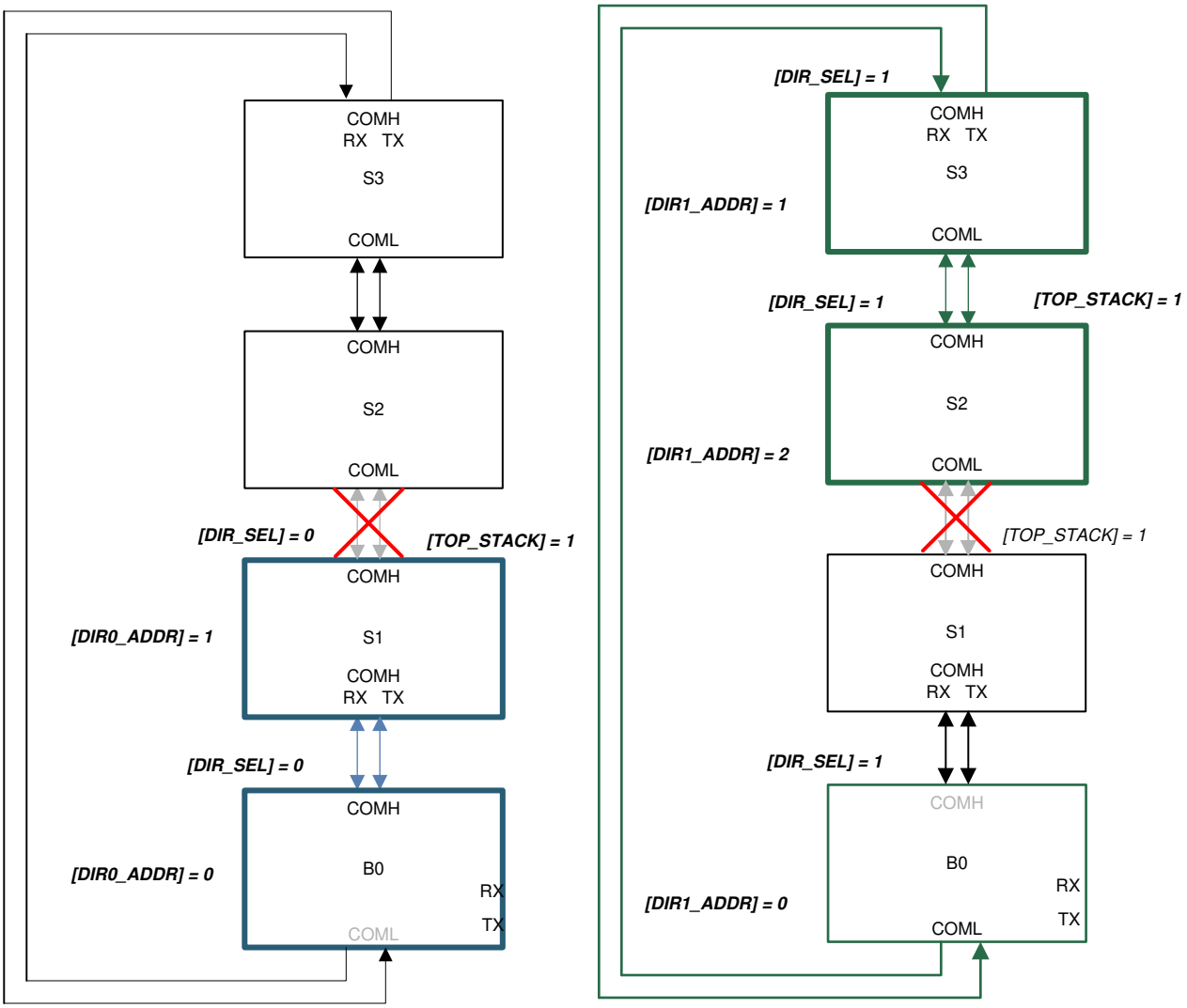


Figure 8-30. Example to Change Communication Direction in Daisy-Chain

Once the device address in both communication directions is set up, host can skip auto-address step when switching communication direction.

In a broken cable case, host follows the same procedure to change the communication direction. To access all devices in the daisy-chain, host will have to communicate with *[DIR_SEL]* = 0 on some devices and communicate with *[DIR_SEL]* = 1 on other devices in the daisy-chain. The chain will also have two ToS devices, one for each communication direction.



(a) Use [DIR_SEL] = 0 direction to communicate to S1

(a) Use [DIR_SEL] = 1 direction to communicate to S3 and S2

Figure 8-31. Using Ring Architecture to Access All Devices in a Broken Cable Case

8.3.5.1.4 Communication Timeout

There are two programmable communication timeout thresholds, CTS timer and CTL timer, that monitor the absence of a valid frame from either UART or daisy-chain communication. A valid frame is defined as any frame (response or command) that does NOT contain any errors that prevent the frame from being processed. The communication timeouts are only actively counting while in ACTIVE mode. The counters are disabled and reset during SHUTDOWN mode. In SLEEP mode, the last counter values are held frozen.

8.3.5.1.4.1 Short Communication Timeout

The short communication timeout acts like an alert to the host when triggered. The timeout period is programmable through the *COMM_TIMEOUT_CONF[CTS_TIME2:0]* bits. If enabled, the timer is reset every time a valid response or command frame is received. If the timer expires, the *FAULT_SYS[CTS]* bit is set.

8.3.5.1.4.2 Long Communication Timeout

The long communication timeout allows the host to put the device in SLEEP or SHUTDOWN mode for power saving. The timeout period is programmable through *COMM_TIMEOUT_CONF[CTL_TIME2:0]* bits. If enabled,

the timer is reset every time a valid response or command frame is received. If the timer expires, host can choose one of the following actions through *COMM_TIMEOUT_CONF[CTL_ACT]* bit.

- Set *FAULT_SYS[CTL]* = 1 and enter SLEEP mode.
- Enter SHUTDOWN mode.

8.3.5.1.5 Communication Debug Mode

The device provides a communication debug mode to ease the initial development phase. To enter this debug mode, host writes an unlock code 0xA5 to register *DEBUG_CTRL_UNLOCK*. Once the debug mode is unlocked, the settings in *DEBUG_COMM_CTRL1* and *DEBUG_COMM_CTRL2* become effective.

To exit the debug mode simply write any value but 0xA5 (for example, writing 0x00) to the *DEBUG_CTRL_UNLOCK*. The COMH, COML, and UART will return to their normal operation status regardless of the settings in the *DEBUG_COMM_CTRL1* and *DEBUG_COMM_CTRL2* registers.

Once the communication debug mode is entered, the host gains control of the following:

Table 8-20. Communication Debug Mode Functions

Control Function	Enable Bit	Description
Full COMH/L transmitter and receiver control	*[USER_DAISY_EN]*	If *[USER_DAISY_EN]* = 1, device will enable or disable its COMH/L transmitter and receiver based on the *DEBUG_COMM_CTRL2* register setting. If *[USER_DAISY_EN]* = 0, COMH/L will be in its normal operation status even under communication debug mode.
Mirror out the data in daisy-chain onto UART	*[USER_UART_EN]*	If *[USER_UART_EN]* = 1, host can set *[UART_MIRROR_EN]* = 1 to instruct the device to translate the daisy-chain onto the UART, allowing host to read the data being received or forwarded in the daisy-chain from the UART interface. Data will be presented in UART communication frame format. For stack devices, the UART TX is disabled by default. To use this feature, host also sets *[UART_TX_EN]* = 1. If *[USER_UART_EN]* = 0, any UART related debug functions are disabled. The UART will be in its normal operation status regardless of the *[UART_MIRROR_EN]* and *[UART_TX_EN]* settings.
Slow down UART baud rate to 250 kbps	*[USER_UART_EN]*	If *[USER_UART_EN]* = 1, host can set *[UART_BAUD]* = 1 to change the UART baud rate to 250 kbps. This will result in slow throughput rate on the daisy-chain. If *[USER_UART_EN]* = 0, UART baud rate will stay on 1 Mbps regardless of the *[UART_BAUD]* setting.

The *DEBUG_COMM_STAT* register has status bits indicating if UART and COMH/L are under user or hardware (device) control. The register also indicates the status of the COMH/L transmitter and receiver. This debug status register is updated per device status and is readable with or without the communication debug mode enabled.

In fact, the read-only debug registers are all readable in ACTIVE mode without communication debug mode enabled. Most of them are lower level communication fault status registers to provide extra information in a communication failure event like the *DEBUG_UART**, *DEBUG_COMH**, and *DEBUG_COML** registers. See Section 8.3.5.2 and Section 8.5.4 for more details.

8.3.5.1.6 Multidrop Configuration

A multidrop configuration is a configuration of multiple devices in a system communicating through UART to the host system. There is no daisy-chain communication between devices. When *[MULTIDROP]* = 1, the device COMH and COML ports are disabled. All the communication protocols, single device read/write, broadcast read/write, stack read/write, reverse broadcast write are still supported as in daisy-chain configuration (that is, *[MULTIDROP]* = 0). However, in a multidrop configuration, it is unlikely to have a use of the stack and reverse broadcast commands. If broadcast command is used, it is still required to set up the devices with sequential device address and set the *[TOP_STACK]* bit on the device with highest device address. The device with *[TOP_STACK]* = 1 will initiate the data return when a broadcast read command is received, and the device with one lower device address will respond next, as in a daisy-chain communication. Additionally, a COMM_CLR must be used before every frame to ensure consistent communication in multidrop configuration.

8.3.5.1.7 SPI Master

The GPIO4 thru GPIO7 are configurable as a SPI master interface when *GPIO_CONF1[SPI_EN]* = 1. The SPI master includes four I/Os:

- SCLK: SPI clock, generated by the device and is used for synchronization
- MOSI: Master data output, driven by the device to output data to slave
- MISO: Master data input, detecting data from slave
- SS: slave select, driven by the device during SPI communication.

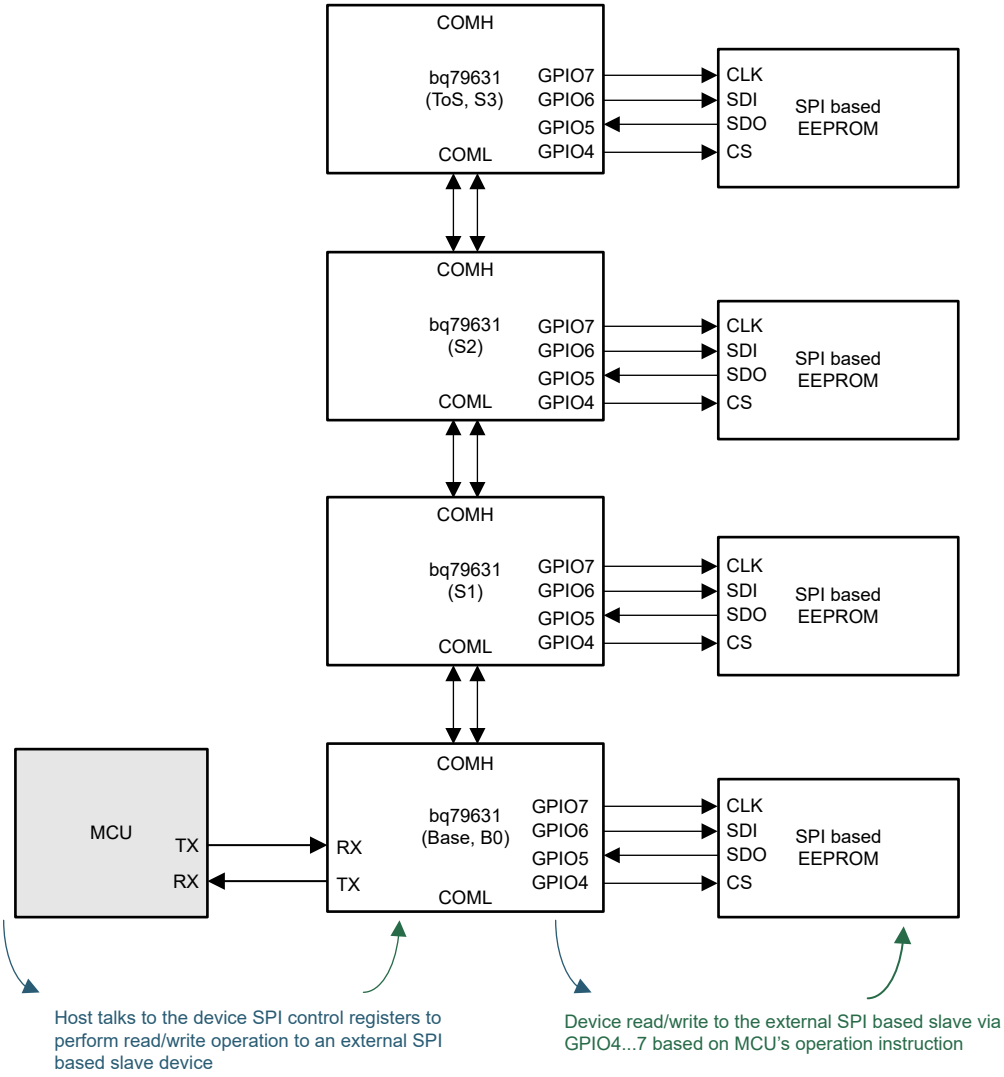


Figure 8-32. SPI Master Stack Configuration

The *SPI_CONF[CPOL]* (clock polarity) and *[CPHA]* (clock phase) define the SPI clock format. The *[CPOL]* is defined if the SPI clock is inverted or non-inverted. The *[CPHA]* is defined if the MISO and MOSI are sampled on the leading (first) clock edge or on the trailing (second) clock edge, regardless of whether that clock edge is rising or falling. The *SPI_CONF[NUMBIT4:0]* defines how many bits the transaction is (1-bit to 24-bit transaction).

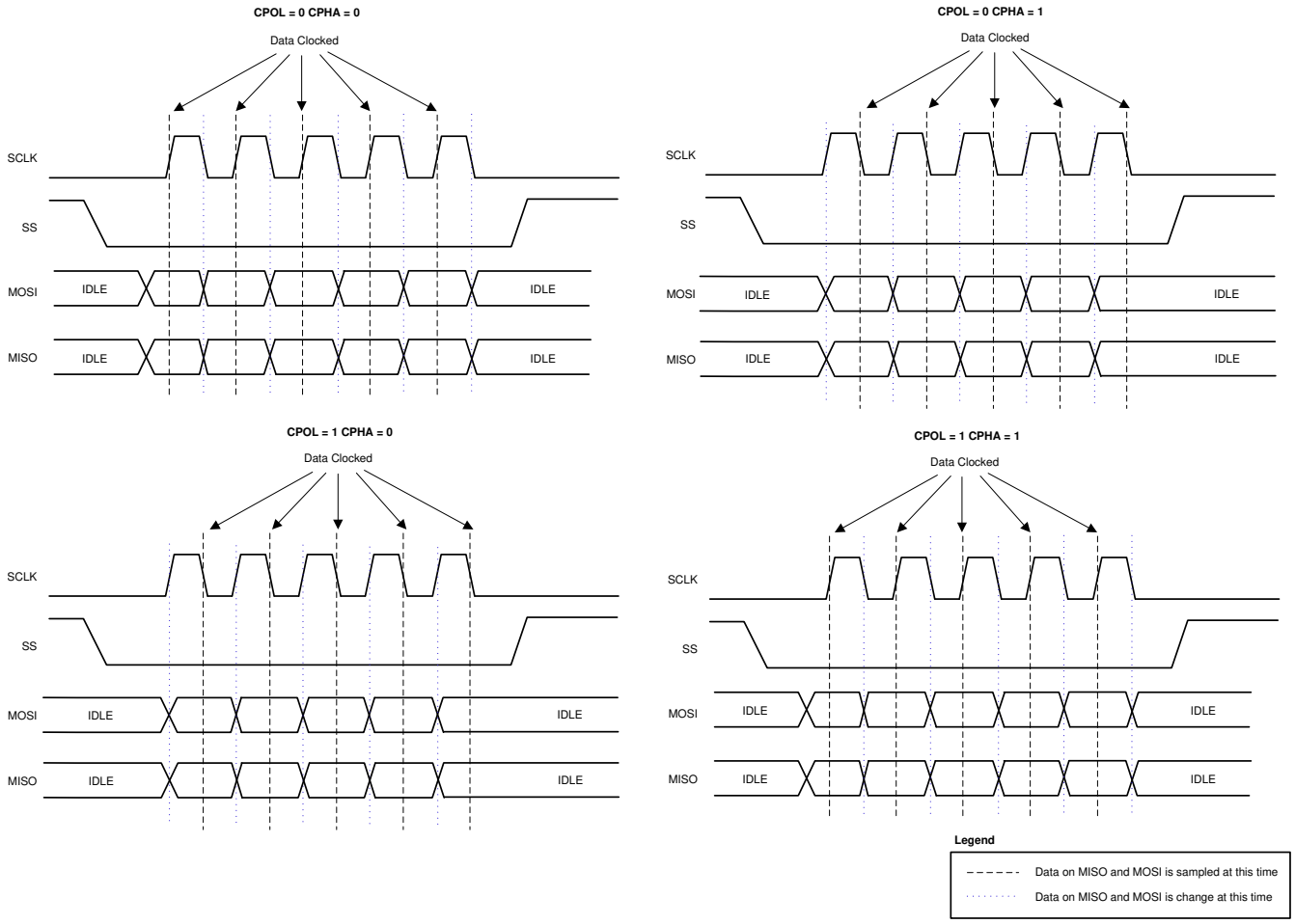


Figure 8-33. SPI Master CPOL and CPHA

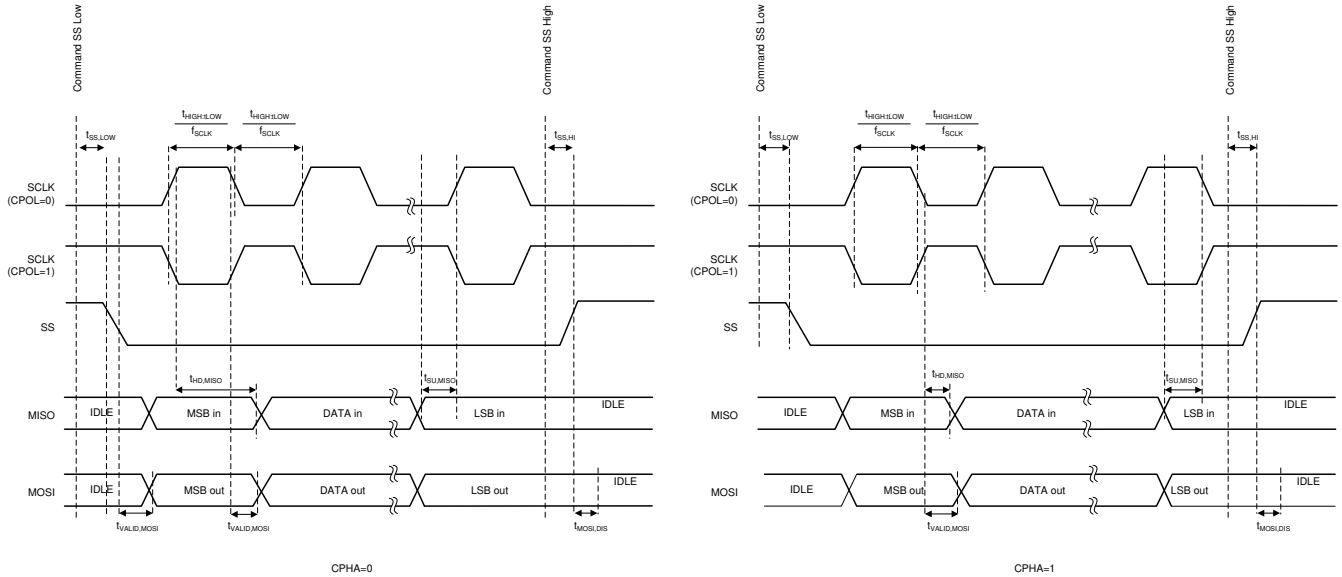


Figure 8-34. SPI Master Timing Diagram

Table 8-21. Write to External SPI Slave

Step	Description
1	Configure the SPI clock polarity, clock phase, number of bit transactions: a. Write to *SPI_CONF* register to configure SPI communication
2	Write the data (from 1 to 24 bits, specified in the *SPI_CONF[NUMBIT4:0]* setting): a. Set up the data to send to SPI slave to the *SPI_TX1* to *SPI_TX3* registers b. *SPI_TX1* is the LSByte and *SPI_TX3* is MSByte
3	Select the slave (assuming active low) and execute the SPI write action: a. Send *SPI_EXE* register = 0x01 (that is, *[SS_CTRL]* = 0 and *[SPI_GO]* = 1)
4	Wait for the SPI communication to complete
5	Deselect the SS port (assuming active low, so deselecting means pull the SS pin high): a. Send *SPI_EXE* register = 0x02 (that is, *[SS_CTRL]* = 1 and *[SPI_GO]* = 0)

Table 8-22. Read from External SPI Slave

Step	Description
1	Configure the SPI clock polarity, clock phase, number of bit transactions: a. Write to *SPI_CONF* register to configure SPI communication
2	Select the slave and execute the SPI communication: a. Send *SPI_EXE* register = 0x01 (that is, *[SS_CTRL]* = 0 and *[SPI_GO]* = 1)
3	Wait for the data transaction to complete
4	Read the data (from 1 to 24 bits, specified in the *SPI_CONF[NUMBIT4:0]* setting): a. Read data from SPI slave from the *SPI_RX1* to *SPI_RX3* registers b. *SPI_TX1* is the LSByte and *SPI_TX3* is MSByte
5	Deselect the SS port (assuming active low, so deselecting means pull the SS pin high): a. Send *SPI_EXE* register = 0x02 (that is, *[SS_CTRL]* = 1 and *[SPI_GO]* = 0)

8.3.5.1.8 SPI Loopback

The SPI master has a loopback function that is enabled using the *DIAG_COMM_CTRL[SPI_LOOPBACK]* bit. When enabled, the byte in the *SPI_TX** registers are clocked directly to the MISO pin of the SPI master to verify the SPI master functionality. This is performed internally, so no external connection is needed to run this test. This verifies that the SPI function is working correctly. The *SPI_CFG*, *SPI_TX**, and *SPI_EXE* registers are written as a normal SPI transaction, but the external pins do not toggle during this mode. That is, the external pins stay static in their last state and do not change state during the loopback operation.

The expected result of the test is that the byte in the *SPI_TX** register is read into the *SPI_RX** register. The SS pin is latched to the setting in *SPI_EXE[SS_CTRL]* that existed when the LOOPBACK mode was enabled. The CPHA and CPOL parameters must be set before entering LOOPBACK mode to ensure proper operation. Changing the CPOL or CPHA parameters while in LOOPBACK mode may result in errant pulses on the SPI outputs and is not recommended.

8.3.5.2 Fault Handling

8.3.5.2.1 Fault Status Hierarchy

The device monitors multiple types of faults such as:

- VS monitoring through the hardware protector, like OV/UV
- System operation driven like device reset, communication timeout, thermal warning, and so on
- Command-based diagnostic check related like the various comparison through the main and AUX ADCs, BIST run, and so on
- Automatic diagnostic check running in the background like the internal power supplies, OTP CRC, and so on
- Communication fault.

Each bit in the *FAULT_SUMMARY* register represents a group of faults which are stored in one or more lower level fault registers. The *FAULT_SUMMARY* register represents the highest hierarchy level of fault status detected by the device. Host system can periodically poll the *FAULT_SUMMARY* register to check the fault status and only read the lower level fault registers if needed (for example, if *FAULT_SUMMARY[FAULT_OVUV]*

= 1, host can read *FAULT_OV1/2* and *FAULT_UV1/2* registers to determine which VS channel triggered the fault).

Table 8-23 shows which lower level register corresponds to the *FAULT_SUMMARY* register bit. The description of the register is covered in the Section 8.5.

Table 8-23. Low-Level Fault Registers

FAULT_SUMMARY Bit Name	FAULT_PROT	FAULT_COMP_ADC	FAULT_OTP	FAULT_COMM	FAULT_OVUV	FAULT_SYS	FAULT_PWR
Lower level register name	*FAULT_PROT1*	*FAULT_COMP_GPIO*	*FAULT_OTP* (1)	*FAULT_ COMM1* (1)	*FAULT_OV1*	*FAULT_SYS*	*FAULT_PWR1*
	FAULT_PROT2	*FAULT_COMP_VSAUX1*		*FAULT_ COMM2* (1)	*FAULT_OV2*		*FAULT_PWR2*
		FAULT_COMP_VSAUX2		*FAULT_ COMM3*	*FAULT_UV1*		*FAULT_PWR3*
		FAULT_COMP_VSOW1			*FAULT_UV2*		
		FAULT_COMP_VSOW2					
		FAULT_COMP_AUXOW1					
		FAULT_COMP_AUXOW2					
		FAULT_COMP_MISC					

(1) Some of the bits in the *FAULT_COMM1/2* and *FAULT_OTP* registers have a lower level of fault information than shown in the *DEBUG_COMM** and *DEBUG_OTP* registers.

8.3.5.2.1.1 Debug Registers

The *DEBUG_COMM** and *DEBUG_OTP* registers are a form of fault status showing lower hierarchy level of fault information for some of the bits in *FAULT_COMM1* , *FAULT_COMM2*, and *FAULT_OTP*.

Table 8-24 shows the hierarchy relationship. See Section 8.5 for the register description details.

Table 8-24. Debug Registers

Low-level Fault Register	Low-level Register Bit		Associated DEBUG Registers
FAULT_COMM1	*[UART_RC]*	Fault related to received command frame from UART	*DEBUG_UART_RC*
	[UART_RR] [UART_TR]	Fault related to received or transmitted response frame from UART	*DEBUG_UART_RR_TR*
FAULT_COMM2	*[COMH_BIT]*	Fault related to error in a byte from COMH	*DEBUG_COMH_BIT*
	[COMH_RC]	Fault related to received command frame from COMH	*DEBUG_COMH_RC*
	[COMH_RR] [COMH_TR]	Fault related to received or transmitted response frame from COMH	*DEBUG_COMH_RR_TR*
	[COML_BIT]	Fault related to error in a byte from COML	*DEBUG_COML_BIT*
	[COML_RC]	Fault related to received command frame from COML	*DEBUG_COML_RC*
	[COML_RR] [COML_TR]	Fault related to received or transmitted response frame from COML	*DEBUG_COML_RR_TR*
FAULT_OTP	*[SEC_DET]*	Single error correction in OTP	*DEBUG_OTP_SEC_BLK*
	[DED_DET]	Double error correction in OTP	*DEBUG_OTP_DED_BLK*

8.3.5.2.2 Fault Masking and Reset

8.3.5.2.2.1 Fault Masking

When a device detects a fault, the corresponding low-level register bit, including the one in the related bit in the *DEBUG_** registers is set. Based on the fault hierarchy relationship, the fault will be reflected in the *FAULT_SUMMARY* register.

A group of faults can be masked, which the related low-level register flag will still be set, but the fault will not be reflected to the corresponding *FAULT_SUMMARY* register. The faults can be masked through the *FAULT_MSK1* and *FAULT_MSK2* registers.

For example, to mask the *FAULT_SUMMARY[FAULT_PWR]* being set, host sets *FAULT_MSK1[MSK_PWR]* = 1.

When fault is masked, it will also prevent the device from asserting the NFAULT pin when the masked fault occurs. See Section 8.3.5.2.3 for details on NFAULT signal.

Table 8-25. Fault Masking

	Masking Bit Name	Related Low-level Register(s) Affected	FAULT_SUMMARY Register Bit That Will Be Masked
FAULT_MSK1	*[MSK_PROT]*	*FAULT_PROT**	*[FAULT_PROT]*
	[MSK_UV]	*FAULT_UV**	*[FAULT_OVUV]*
	[MSK_OV]	*FAULT_OV**	
	[MSK_COMP]	*FAULT_COMP_**	*[FAULT_COMP]*
	[MSK_SYS]	*FAULT_SYS*	*[FAULT_SYS]*
	[MSK_PWR]	*FAULT_PWR**	*[FAULT_PWR]*
FAULT_MSK2	*[MSK_OTP_CRC]*	*FAULT_OTP[CUST_CRC][FACT_CRC]*	*[FAULT_OTP]*
	[MSK_OTP_DATA]	All non-CRC bits in *FAULT_OTP*, *DEBUG_OTP_**	
	[MSK_COMM3_FCOMM]	*FAULT_COMM3[FCOMM_DET]*	*[FAULT_COMM3]*
	[MSK_COMM3_FTONE]	*FAULT_COMM3[FTONE_DET]*	
	[MSK_COMM3_HB]	*FAULT_COMM3[HB_FAIL][HB_FAST]*	
	[MSK_COMM2]	*FAULT_COMM2*, *DEBUG_COMH_*, DEBUG_COML_**	*[FAULT_COMM2]*
	[MSK_COMM1]	*FAULT_COMM1*, *DEBUG_UART_**	*[FAULT_COMM1]*

8.3.5.2.2.2 Fault Reset

Once fault is detected, the fault status bit is latched until cleared using the reset bit. Similar to fault masking, when the specific fault reset bit is set, the associated low-level fault registers, including the *DEBUG_** registers are cleared. The corresponding bit in the *FAULT_SUMMARY* register will clear if all its associated low-level registers are cleared. If the fault condition persists and the reset bit is written, the fault status bit is not reset. The fault indicator cannot be reset until the underlying fault condition is eliminated.

The fault is reset through the *FAULT_RST1* and *FAULT_RST2* registers; the fault reset bits are structured in the same corresponding fault status registers as the fault masking bits.

8.3.5.2.3 Fault Signaling

Host can acquire the fault status with the following methods:

- Constantly polling the *FAULT_SUMMARY* status on each device in the daisy-chain. If *FAULT_SUMMARY* is non-zero, read the low-level fault status registers to obtain more information.
- Enable fault status to pass down the daisy-chain to the base device. Enable base device's NFAULT pin to be asserted when the *FAULT_SUMMARY* is non-zero in any of the devices in the daisy-chain. Host monitors NFAULT. When NFAULT is triggered, host does a broadcast read on the *FAULT_SUMMARY* to determine which device(s) is at fault.

When using the NFAULT pin in the base device to signal the host under a fault detection, the stack devices have to transfer their fault status information to the base device. The information is transmitted through COMH/L through the same communication cables. In ACTIVE mode, each device embeds the fault status to the communication when a response frame is forwarded. In SLEEP mode, or using Heartbeat and Fault Tone in SLEEP mode.

The NFAULT pin can be masked by configuring *DEV_CONF[NFAULT_EN]* = 0. When NFAULT is disabled, the device will set the corresponding flag in *FAULT_SUMMARY* register but will not assert NFAULT.

8.3.5.2.3.1 Fault Status Transmitting in ACTIVE Mode

In ACTIVE mode, stack devices can embed their fault status before retransmitting a response frame if *DEV_CONF[FCOMM_EN]* = 1. When the *[FCOMM_EN]* = 1, the stack devices repurpose the SOF bit in the

response frame's device address byte, register address bytes (both high and low address bytes) to a fault status bit instead. See Figure 8-35. This will be referred to as fault status bits in the rest of this section.

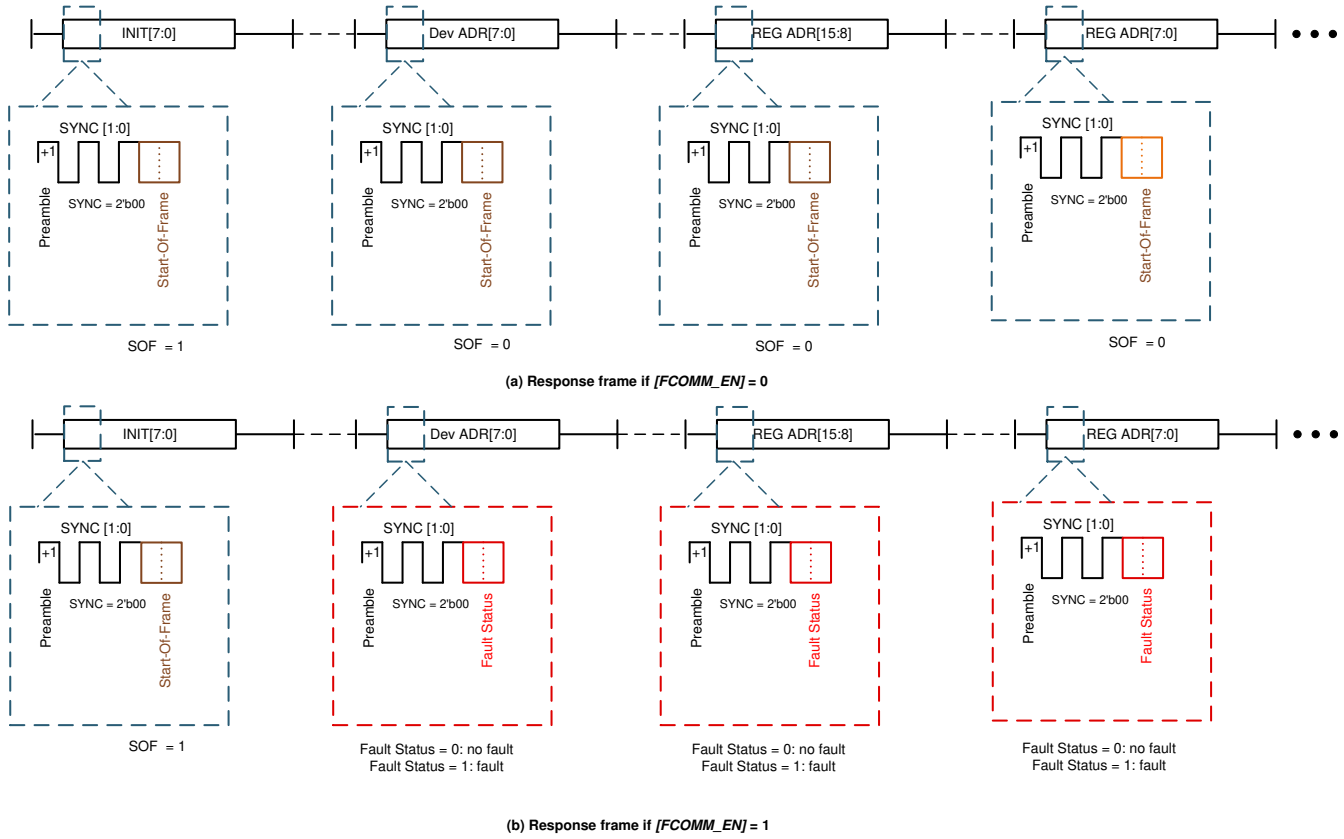


Figure 8-35. Embed Fault Status in Communication Response Frame

To pass on the fault status of the stack devices, the host sends a broadcast read or sends a single device read to the ToS device. Both types of reads will result in response frames passing through every device in the daisy-chain, giving each device an opportunity to OR their fault status to the fault status bits in the response frame.

An example of a response frame going through a daisy-chain from a single device read command to the top device is shown in Figure 8-36.

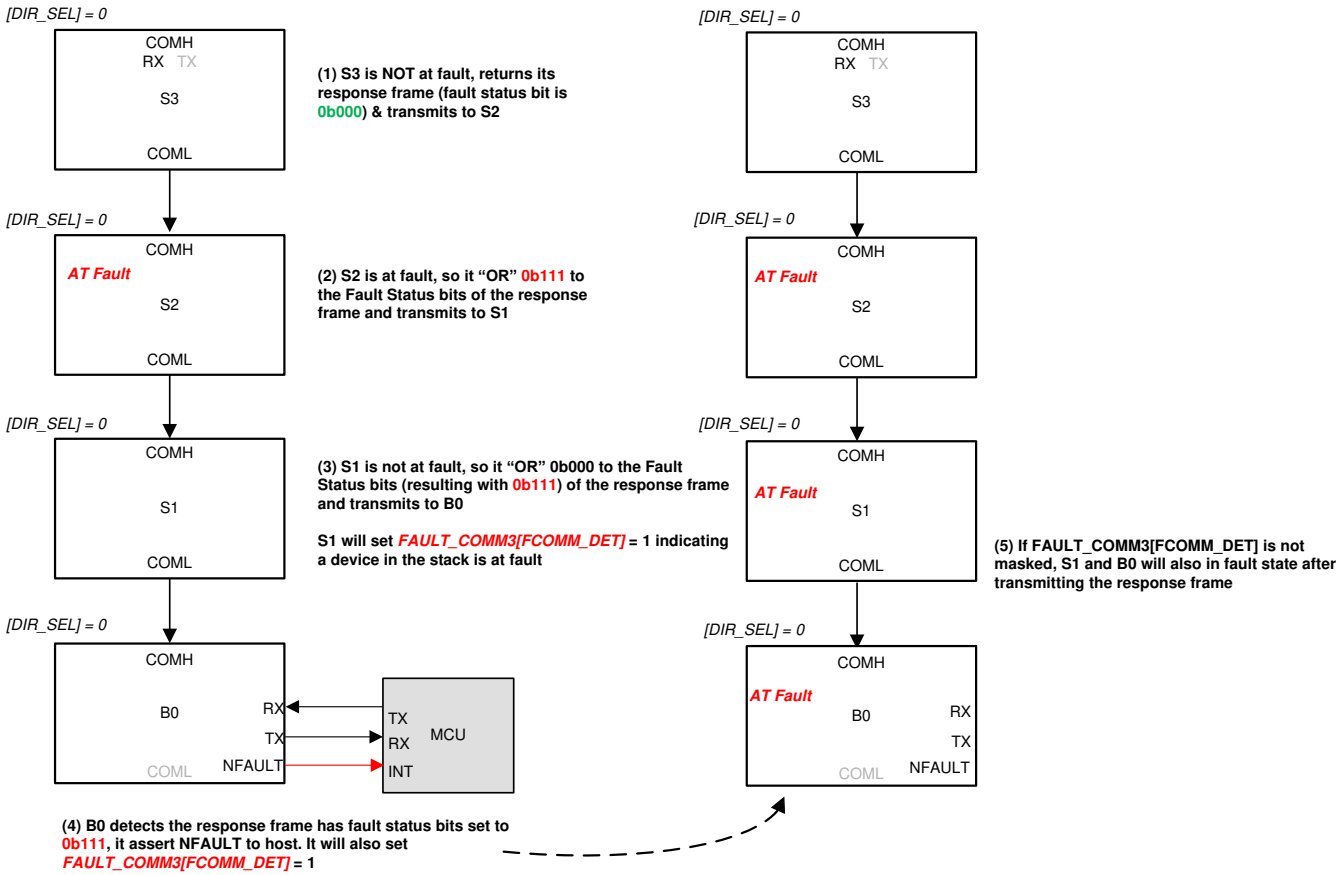


Figure 8-36. Transfer Fault Status in ACTIVE Mode (Respond to a Single Device Read)

When a device has no fault, it will OR the fault status bits with 0b000; otherwise, it will OR the fault status bits with 0b111. Hence, if a fault exists in any device in the daisy-chain, the fault status bits will be 0b111. For the base device to assert the NFAULT pin, it requires at least two bits of the fault status bits to be 1.

Additionally, when a device detects a response frame with at least two of the fault status bits being 1, the device will also set the *FAULT_COMM2[FCOMM_DET]* = 1. If this fault is not masked, the device will be in fault state as well. Next time a response frame is transmitted, this device will OR the fault status bits with 0b111.

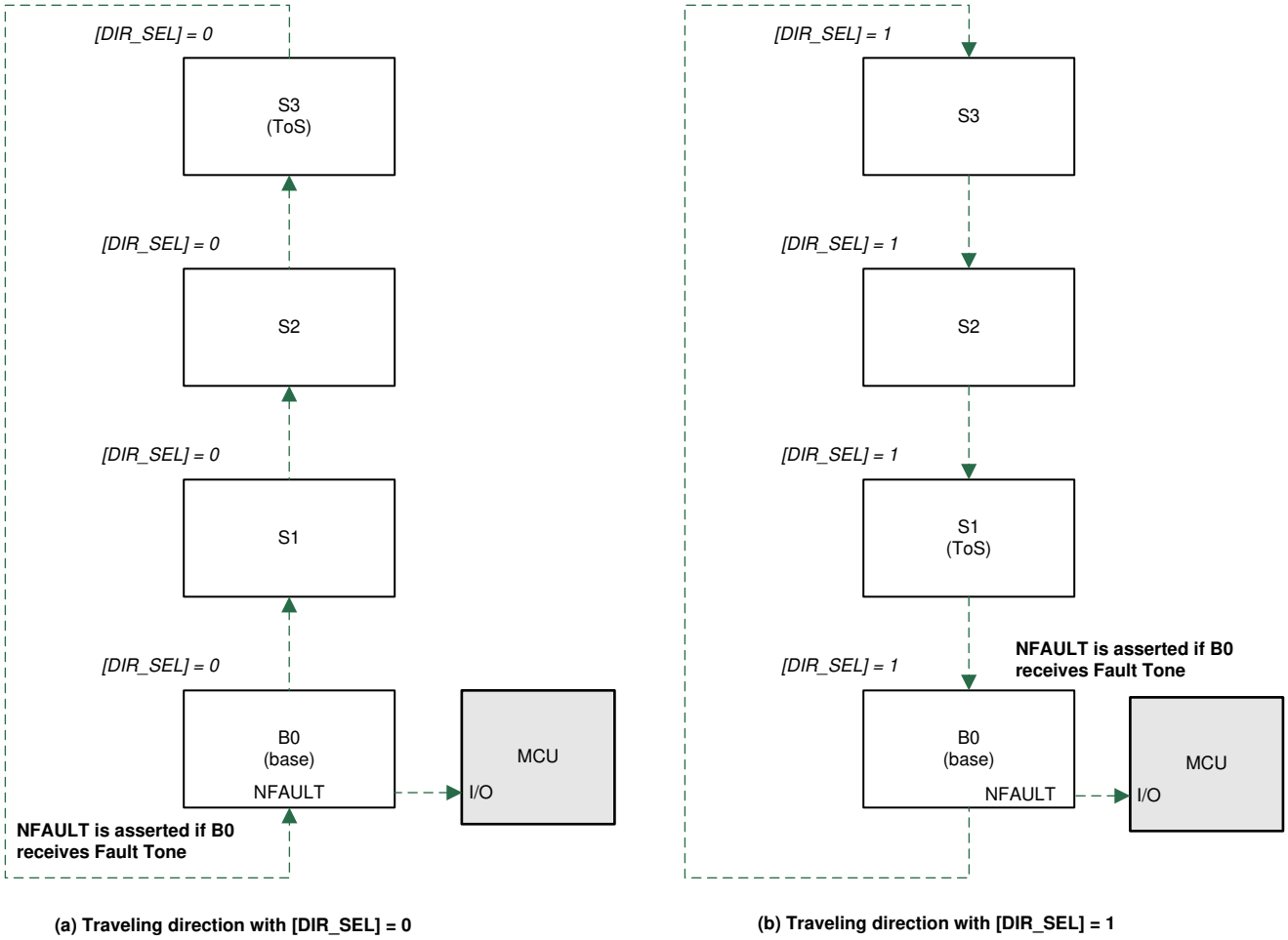
Host performs a broadcast read to detect which device in the daisy-chain is at fault and what type of fault.

8.3.5.2.3.2 Fault Status Transmitting in SLEEP Mode

In SLEEP mode, the following fault detections are still active:

- Customer and Factory OTP shadow registers CRC check
- Device thermal warning
- Power supplies OV, UV, and oscillation detection
- If OVUV protectors are enabled, VS OV and UV detection.

Because communication is not available in SLEEP mode, the device provides an option to transmit the fault status through Heartbeat (device in no fault state) and Fault (device in fault state) Tones. These tones are transmitted in the same direction as a communication command frame, which is based on the *CONTROL1[DIR_SEL]* setting. For the tone signal to return back to the base device (so NFAULT can be triggered if needed), a Ring architecture must be used to support transmitting fault status in SLEEP mode.



(a) Traveling direction with [DIR_SEL] = 0

(b) Traveling direction with [DIR_SEL] = 1

Figure 8-37. Heartbeat or Fault Tone Traveling Direction

Both the Heartbeat and Fault Tones are a type of tone similar to the communication tone. One main difference is a communication tone only transmits with a single burst of couplets, but Heartbeat and Fault Tones are sent with a burst of couplets periodically. See Section 8.3.5.2.3.3 for details.

8.3.5.2.3.3 Heartbeat and Fault Tone

The tones are enabled by setting *DEV_CONF[HB_EN]* = 1 and *DEV_CONF[FTONE_EN]* = 1 to enable the Heartbeat and Fault Tone transmitters, respectively. The Heartbeat and Fault Tone receivers are always on in SLEEP mode regardless of the *[HB_EN]* and *[FTONE_EN]* settings. To avoid fault detection (asserting NFAULT or *FAULT_SUMMARY* register) by Heartbeat or Fault Tone fault, mask the fault by *[MSK_COMM3_HB]* = 1 or *[MSK_COMM3_FTONE]* = 1.

The Heartbeat and Fault Tone are formed with couplets with "–" polarity. They are differentiated by the number of couplets. Unlike communication tones, Heartbeat and Fault Tone are transmitted periodically. The period between tones is referring as Burst period. The number of couplets transmitted is always greater than the number of couplets needed for detection.

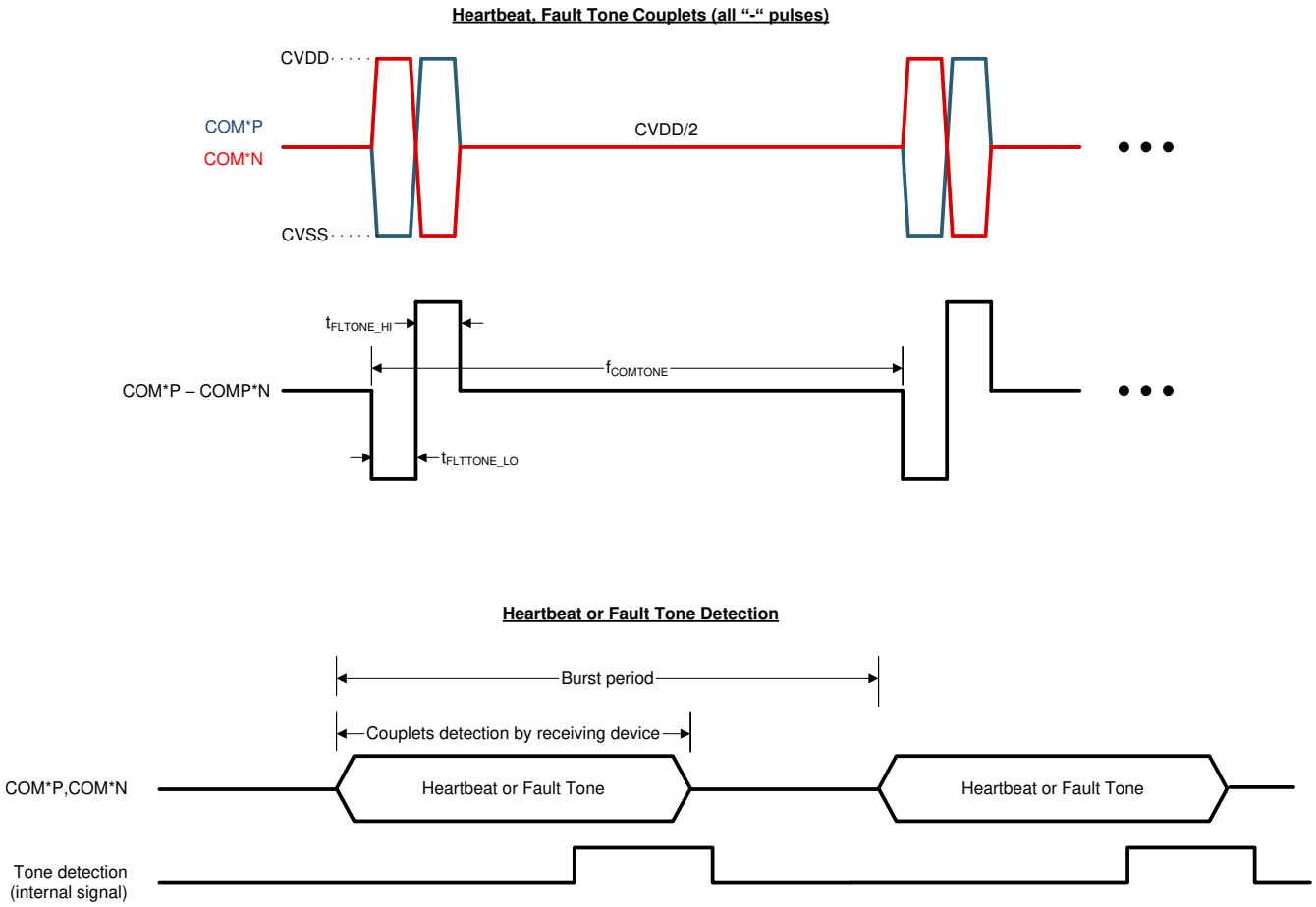


Figure 8-38. Heartbeat and Fault Tone

8.3.5.3 Nonvolatile Memory

There are memory locations that are programmable in nonvolatile memory (NVM) using OTP (One Time Programmable). The memory space is divided in two groups, factory space and customer space. The factory space stores the device configurations that are essential for normal operation. This space is not accessible by the host. The customer space contains the device default setting that host system can customize for their application configuration. This space is readable and programmable by the host.

When a device reset occurs, factory and customer OTP values are reloaded to their shadow registers. Error check and correction (ECC), single error correction (SEC) and double error detection (DED), are performed during the factory and customer space OTP load. The corresponding *FAULT_OTP[SEC_DET]* or *FAULT_OTP[DED_DET]* will be set if an error is detected.

Any load errors of the factory OTP space signal a fault using the *FAULT_OTP[FACTLDERR]*. Any load errors of the customer OTP space signal a fault using the *FAULT_OTP[CUSTLDERR]*. Additionally, the OTP space (factory and customer) are protected from data integrity problems using CRC. The corresponding *FAULT_OTP[FACT_CRC]* and *[CUST_CRC]* bits will be set if a CRC error is detected.

If any overvoltage error conditions exist in the OTP pages space (factory and customer) during programming, the *OTP_FAULT[GBLOVERR]* bit is set. Information received from the device with this error must not be considered reliable.

8.3.5.3.1 OTP Page Status

There are two unused pages of OTP memory available for the customer to program. Each page status is held in the *OTP_CUST1_STAT* and *OTP_CUST2_STAT* registers. The registers provide information on the current status of the page such as:

- Load status (if loaded, loaded with error, loaded but failed)
- Programmed successfully or available to be programmed
- Programmed status

When a reset occurs, the device evaluates the OTP page status and chooses the latest and valid OTP page to load. Page 2 has priority over Page 1. If both pages have not been written, the factory OTP default are loaded. Section 8.5.1 shows all customer programmable OTP parameters. The register summary also shows the default values when Customer OTP Page 1 and Page 2 are not programmed.

- A valid page is one where the *OTP_CUST*_STAT[PROGOK]* = 1.
- When the page is selected for loading, the *OTP_CUST*_STAT1[LOADED]* = 1.
- If a single error occurs in the loading of the page, the page is loaded after the single error is corrected and the *OTP_CUST*_STAT1[LOADWRN]* = 1.
  - Additionally, the *DEBUG_OTP_SEC_BLK* register is updated with the location of the error corrected block.
- If a double error occurs, the loading of that block is terminated and the hardware defaults of that block are loaded (as indicated in Section 8.5.1).
  - The overall page loading process is not terminated for a DED, only the affected block is terminated.
  - When a DED occurs, the *OTP_CUST*_STAT1[LOADERR]* = 1. Additionally, the *DEBUG_OTP_DED_BLK* register is updated with the block where the double error occurred.

8.3.5.3.2 OTP Programming

Section 8.5.1 shows all parameters that can be programmed to the customer OTP page. There are two pages of OTP memory available for customer to use.

Before programming the OTP, host ensures:

- All OTP shadow registers have the correct settings
- A customer OTP page is valid to be programmed. A valid page is one with *OTP_CUST*_STAT1[TRY]* = 0 and *OTP_CUST*_STAT1[FMTERR]* = 0.

Table 8-26. Program the OTP

Step	Procedure
1	Unlock the OTP programming: a. Write the following data to *OTP_PROG_UNLOCK1A* to *OTP_PROG_UNLOCK1D* registers. • *OTP_PROG_UNLOCK1A* <- data 0x02 • *OTP_PROG_UNLOCK1B* <- data 0xB7 • *OTP_PROG_UNLOCK1C* <- data 0x78 • *OTP_PROG_UNLOCK1D* <- data 0xBC b. Do another write with the following data to *OTP_PROG_UNLOCK2A* to *OTP_PROG_UNLOCK2D* registers. • *OTP_PROG_UNLOCK2A* <- data 0x7E • *OTP_PROG_UNLOCK2B* <- data 0x12 • *OTP_PROG_UNLOCK2C* <- data 0x08 • *OTP_PROG_UNLOCK2D* <- data 0x6F Each block of registers must be written in order (that is, A, B, C, then D) with no other writes or reads between. The best practice is to use the same Write command to update. Any attempt to update the registers out of sequence, or if another register is written or read between writes, the entire sequence must be redone.
2	Check to confirm the OTP unlock procedure is successful: a. Read to confirm *OTP_PROG_STAT[UNLOCK]* = 1 Issuing a Read command after step 1 is ok, but issuing the *[PROG_GO]* must be the next write command after the unlock procedures.
3	Select the proper OTP page and start the OTP programming: a. To program page1, set *OTP_PROG_CTRL[PAGESEL][PROG_GO]* = 0x01, or b. To program page2, set *OTP_PROG_CTRL[PAGESEL][PROG_GO]* = 0x03

Table 8-26. Program the OTP (continued)

Step	Procedure
4	Wait t_{PROG} for the OTP programming to complete
5	Check to ensure there is no error during OTP programming. The following bits are expected to be 1 after a successful OTP programming: a. *OTP_PROG_STAT[DONE]* = 1, OTP programming is done. No other bit will be set in this register. b. If page 1 is programmed, *OTP_CUST1_STAT[PROGOK]*, *[TRY]*, *[OVOK]*, and *[UVOK]* bits are 1. Other bits are 0. c. If page 2 is programmed, *OTP_CUST2_STAT[LOADED]*, *[PROGOK]*, *[TRY]*, *[OVOK]*, and *[UVOK]* bits are 1. Other bits are 0.
6	Issue a digital reset to reload the registers with the updated OTP values: a. *CONTROL1[SOFT_RESET]* = 1

During programming, if a programming voltage OV or UV event occurs, the *OTP_CUST*_STAT[UVOK]* or *OTP_CUST_STAT2[OVOK]* bit is 0 to indicate the programming voltage under- or overvoltage condition is detected during the programing attempts. In addition, the *[UVERR]*, *[OVERR]*, *[SUVERR]*, and *[SOVERR]* bits in the *OTP_PROG_STAT* register indicate if there is programming voltage error during programming and stability test.

Note

- During the programming procedure, device performs a programming voltage stability test before actually programming the OTP. If a programming voltage fails the stability test, the device will not set the *OTP_CUST*_STAT[TRY]* bit, giving the customer another attempt to program the page again.
- If the host incorrectly selects a page for programming, the *OTP_PROG_STAT[PROGERR]* bit is set. This indicates that the selected page was not available to be programmed. Select the correct page and retry the programming.

8.3.5.4 Diagnostic Control/Status

The device complies with applicable component level requirements for ASIL-D on voltage measurement, temperature measurement and communication. The following subsections describe the diagnostic control and fault status that can be used as part of the safety mechanisms.

The Safety Manual for BQ79631-Q1 and the BQ79606-Q1 FMEDA documents are/will be available separately from Texas Instruments. Contact TI Sales Associate or Applications Engineer for further information.

8.3.5.4.1 Power Supplies Check

8.3.5.4.1.1 Power Supply Diagnostic Check

The internal power supply circuits have overvoltage, undervoltage, oscillation detection, and/or current limit checks. All these detections are continuously running in the background when the device is in ACTIVE or SLEEP mode. If a failure is detected, the corresponding flags in the *FAULT_PWR** registers will be set or in certain failure modes, the device will reset. Table 8-27 summarizes the diagnostics that apply for each power supply and the corresponding action when failure is detected.

Table 8-27. Power Supply Diagnostic Checks

Supply/ Ground Pin	OV Check	UV Check	OSC Check	Current Limit	Pin Open
LDOIN					
AVDD	If this fails, set *FAULT_PWR1[AVDD_OV]*	If this fails, disable DVDD and trigger a digital reset. After soft reset, device sets *[AVDDUV_DRST]* to indicate a reset is caused by AVDD UV.	If fails, set *FAULT_PWR1[AVDD_OSC]*	Limit current to EC table current limit specification	

Table 8-27. Power Supply Diagnostic Checks (continued)

Supply/ Ground Pin	OV Check	UV Check	OSC Check	Current Limit	Pin Open
DVDD	If this fails, set *FAULT_PWR1[DVDD_OV]*	If this fails, trigger a digital reset		Limit current to EC table current limit specification	
CVDD	If this fails, set *FAULT_PWR1[CVDD_OV]*	If this fails, set *FAULT_PWR1[CVDD_UV]*		Limit current to EC table current limit specification	
TSREF	If this fails, set *FAULT_PWR2[TSREF_OV]* registers to all 1s.	If this fails, set *FAULT_PWR2[TSREF_UV]* registers to all 1s.	If fails, set *FAULT_PWR2[TSREF_OSC]* registers to all 1s.	Limit current to EC table current limit specification	
NEG5V		If this fails, set *FAULT_PWR2[NEG5V_UV]*			
REFHP/REFHM			If REFHP fails, set *FAULT_PWR2[REFH_OSC]*		If REFHM opens, set the *FAULT_PWR1[REFHM_OPEN]*
DVSS					If this opens, set the *FAULT_PWR1[DVSS_OPEN]*
CVSS					If this opens, set the *FAULT_PWR1[CVSS_OPEN]*

Note

Due to the detection logic implemented, when AVDD OV or UV is detected, the AVDD OSC fault can also be triggered. Similarly, when TSREF OV or UV, the TSREF OSC fault can also be triggered.

8.3.5.4.1.2 Power Supply BIST

The device implements a power supply BIST (Built-In Self-Test) function to test the primary power supply failure diagnostic paths that cover the following detections:

- *FAULT_PWR1[AVDD_OV]*, *[AVDD_OSC]*, *[DVDD_OV]*, *[CVDD_OV]* and *[CVDD_UV]*
- *FAULT_PWR2[TSREF_OV]*, *[TSREF_UV]* and *[NEG5V_UV]*

The power supply BIST is essentially a check on the checker and it is a command base function initiated by host.

The power supply BIST, once started, will force a fault on failure detection path on each supply. Take AVDD OV diagnostic path as an example, when the BIST engine tests the AVDD OV path, the following occur:

1. The BIST engine forces a fail to the AVDD OV comparator
2. The BIST engine then checks to ensure the signal to trigger *FAULT* register is asserted, and the signal to trigger NFAULT is also asserted
3. The BIST engine resets the *FAULT* register and NFAULT signal (that is, clears the *FAULT_PWR1/2/3* registers and deasserts NFAULT)
4. The BIST engine repeats step 1 to step 3 on the next power supply diagnostic path check (for example, AVDD OSC) until all intended diagnostic paths covered by BIST are tested.

Note

- During the BIST run, the NFAULT pin will be toggled on and off. Host ignores the NFAULT pin status or can disable the NFAULT pin output by setting *DEV_CONF[NFAULT_EN]* = 0.
- Among all internal power supplies, TSREF is one that can be enabled or disabled by host. To ensure TSREF diagnostic paths are tested during BIST run, host enables TSREF before starting the power supply BIST. Otherwise, the BIST engine will ignore the TSREF diagnostic paths test result during the BIST run.
- Because other nonpower supply-related faults can also trigger NFAULT, it is recommended to mask all nonpower supply-related faults through *FAULT_MSK1/2* registers before the power supply BIST run.
- Host also ensures there are no power supply faults before starting the power supply BIST run.

Start power supply BIST by sending *DIAG_PWR_CTRL[PWR_BIST_GO]* = 1. The BIST run will not abort even if a failure is detected during the run. At the end of the BIST run, the result is indicated by the *FAULT_PWR2[PWRBIST_FAIL]* flag.

The power supply BIST forces a failure and ensures the diagnostic path triggers the fault accordingly. A failure on the BIST run indicates a diagnostic path is unable to trigger in a fault condition. To further examine which path is unable to indicate a failure, host can set the *DIAG_PWR_CTRL[BIST_NO_RST]* = 1. This bit disables the reset step during the BIST run. Re-start power supply BIST with this option enabled. At the end of the BIST run, examine the *FAULT_PWR1* and *FAULT_PWR2* registers. Any register flag that remains 0 indicates it is unable to flag a failure.

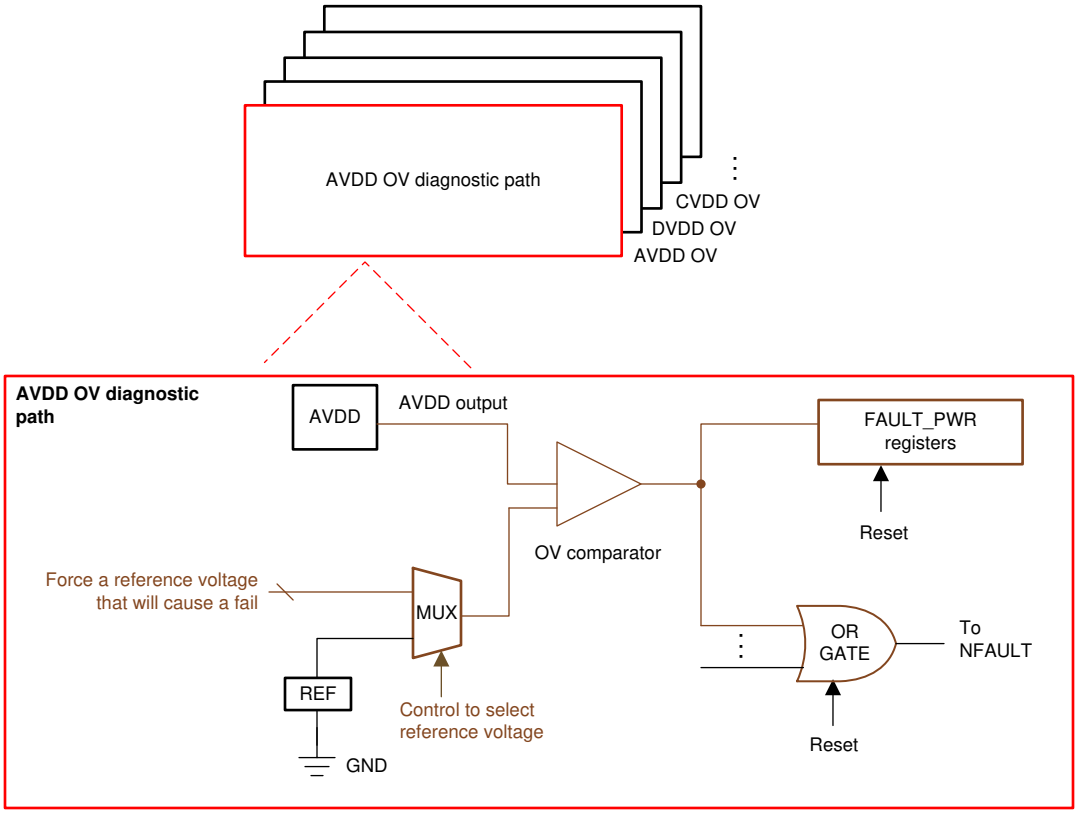


Figure 8-39. Power Supply BIST

8.3.5.4.2 Thermal Shutdown and Warning Check

8.3.5.4.2.1 Thermal Shutdown

Thermal shutdown occurs when the thermal shutdown sensor senses an overtemperature condition of the device. The sensor operates without interaction and is separated from the ADC measured die sensor. The thermal shutdown function has a register-status indicator flag (*FAULT_SYS[TSHUT]*) that is saved during the shutdown event and can be read after the device is awaken back up. When a TSHUT fault occurs, the part immediately enters the SHUTDOWN mode. Any pending transactions on UART or daisy-chain are discarded. There is no fault signaling performed when a thermal shutdown event occurs as the device immediately shuts down.

To awaken the device, host ensures the ambient temperature is below T_{SHUT_FALL} and sends a WAKE ping to the base device. Host will not attempt to wake the device if the ambient temperature is still above T_{SHUT_FALL}.

Upon waking up, the *FAULT_SYS[TSHUT]* bit is set. The *FAULT_SYS[SHUTDOWN_REC]* = 1 indicating the prior shutdown is caused by abnormal event. See Section 8.4.1.1 for more details. If the system faults are unmasked, *FAULT_MSK1[MSK_SYS]* = 0, the thermal shutdown will be reflected as a fault and will be indicated in the *FAULT_SUMMARY* register and the assertion of the NFAULT pin.

8.3.5.4.2.2 Thermal Warning

To warn the host of an impending thermal overload the device includes an overtemperature warning that signals a fault when the die temperature approaches thermal shutdown. The device detects the die temperature through the TWARN sensor against the thermal warning threshold. There are four threshold options configured by the *PWR_TRANSIT_CONF[TWARN_THR1:0]* setting.

When the system fault is unmasked, and the temperature warning fault occurs, the *FAULT_SYS[TWARN]* = 1. Host can take action to avoid a thermal shutdown.

8.3.5.4.3 Oscillators Watchdog

The oscillators are monitored by watchdog circuits. There are two oscillators in the device, the HFO and the LFO. If these oscillators are not functioning, the device does not operate. If the HFO or LFO does not transition within the expected time, the watchdog circuits causes a digital reset.

When this unexpected reset occurs, it is recommended that the host sends a SHUTDOWN ping/tone to the problem device and then send a WAKE ping to reset the daisy-chain. If the oscillators are truly damaged, the device will not restart and must be replaced.

In addition to the watchdog, the LFO frequency is monitored to ensure it stays within acceptable limits. If the LFO frequency falls outside of the expected range, the *FAULT_SYS_FAULT[LFO]* bit is set.

8.3.5.4.4 OTP Error Check

8.3.5.4.4.1 OTP CRC Test and Faults

CRC Test:

The factory registers and customer OTP shadow registers are covered by a CRC check that constantly runs in the background. The *CUST_CRC_RSLTH* and *CUST_CRC_RSLTL* registers hold the current device's computed CRC value. This value is compared against the customer programmed value in the CRC registers, *CUST_CRCH* and *CUST_CRCL*. When updating any customer OTP shadow register covered in the CRC, the host must update a new CRC value to *CUST_CRCH* and *CUST_CRCL* registers. The CRC calculation is performed in the same manner (including the bit stream ordering) and with the same polynomial as described in Section 8.3.5.1.1.2.1.6. The CRC check and comparison for factory and customer spaces is performed periodically and the *DEV_STAT[CUST CRC_DONE]* and *[FACT_CRC_DONE]* bits are set after the check is complete. If the bit is already set, it remains set until cleared with a read.

CRC Faults:

When *CUST_CRC_HI/L* and *CUST_CRC_RSLT_HI/L* do not match, the *FAULT_OTP[CUST_CRC]* flag is set until the condition is corrected. Continuous monitoring of the factory NVM space occurs in a similar fashion, concurrently with the monitoring of the customer space. When a factory register change is detected, the

FAULT_OTP[FACT_CRC] flag is set. When this fault occurs, the host should reset the fault flag to see if the fault persists. If the fault persists, the host must perform a reset of the part. If reset does not correct the issue, the device is corrupted and must not be used.

8.3.5.4.4.2 OTP Margin Read

The device provides OTP margin read test modes, with which host can set up to reload the OTP with margin 1 or margin 0. To start the margin read test, host selects the desired test mode through *DIAG_OTP_CTRL[MARGIN_MODE2:0]* and sets *DIAG_OTP_CTRL[MARGIN_GO]* = 1. The device will reload the OTP per the *[MARGIN_MODE2:0]* setting. Any OTP related error will be flagged to the *FAULT_OTP* register.

8.3.5.4.4.3 Error Check and Correct (ECC) OTP

ECC:

Register values for selected registers (0x0000 to 0x002F) are permanently stored in OTP. All registers also exist as volatile storage locations at the same addresses, referred to as shadow registers. The volatile registers are for reading, writing, and device control. For a list of registers included in the OTP, see Section 8.5.1.

During wakeup, the device first loads all shadow registers with hardware default values listed in Section 8.5.1. Then the device loads the registers conditionally with OTP contents from the results of the Error Check and Correct (ECC) evaluation of the OTP. The OTP is loaded to shadow registers in 64-bit blocks; each block has its own Error Check and Correct (ECC) value stored. The ECC detects a single-bit (Single-Error-Correction) or double-bit (Double-Error-Detection) changes in OTP stored data. The ECC is calculated for each block, individually.

Single-bit errors are corrected, double-bit errors are only detected, not corrected. A block with good ECC is loaded. A block with a single-bit error is corrected, and the *FAULT_OTP[SEC_DET]* bit is set to flag the corrected error event. Additionally, the *DEBUG_OTP_SEC_BLK* register is updated with the location of the error corrected block. This enables the host to keep track of potentially damaged memory. The block is loaded to shadow registers after the single-bit error correction. Because the evaluation is on a block-by-block basis, it is possible for multiple blocks to have a single-correctable error and still be loaded correctly. Multiple-bit errors can exist with full correction, as long as they are limited to a single error per block.

A block with a bad ECC comparison (two-bit errors in one block) is not loaded and the *FAULT_OTP[DED_DET]* bit is set to flag the failed bit-error event. Additionally, the *DEBUG_OTP_DED_BLK* register is updated with the block where the double error occurred. The hardware default value remains in the register. This allows some blocks to be loaded correctly (no fail or single-bit corrected value) and some blocks not to load. When the *FAULT_OTP[SEC_DET]* or *FAULT_OTP[DED_DET]* bit is set and the condition is not cleared by a device reset, the device is corrupted and must not be used.

The ECC engine uses the industry standard 72,64 SEC DEC ECC implementation. The OTP is protected by a (72, 64) Hamming code, providing single error correction, double error detection (SECDED). For each 64 bits of data stored in OTP, an additional eight bits of parity information are stored. The parity bits are designated p0, p1, p2, p4, p8, p16, p32, and p64. Bit p0 covers the entire encoded 72-bit ECC block. The remaining seven parity bits are assigned according to the following rule:

- Parity bit p1 covers odd bit positions, that is, bit positions which have the least significant bit of the bit position equal to 1 (1, 3, 5, and so on), including the p1 bit itself (bit 1).
- Parity bit p2 covers bit positions which have the second least significant bit of the bit position equal to 1 (2, 3, 6, 7, 10, 11, and so on), including the p2 bit itself (bit 2).
- The pattern continues for p4, p8, p16, p32, and p64. Table 8-28 specifies the complete encoding.

Table 8-28. (72, 64) Parity Encoding

Bit Position		71	70	69	68	67	66	65	64	63	62	61	60	59	58	57	56	55	54
Encoded Bits		d63	d62	d61	d60	d59	d58	d57	p64	d56	d55	d54	d53	d52	d51	d50	d49	d48	d47
Parity Bit Coverage	p0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	p1	x		x		x		x		x		x		x		x		x	
	p2	x	x			x	x			x	x			x	x			x	x
	p4	x	x	x	x					x	x	x	x					x	x
	p8									x	x	x	x	x	x	x	x		
	p16									x	x	x	x	x	x	x	x	x	x
	p32									x	x	x	x	x	x	x	x	x	x
	p64	x	x	x	x	x	x	x	x										
Bit Position		53	52	51	50	49	48	47	46	45	44	43	42	41	40	39	38	37	36
Encoded Bits		d46	d45	d44	d43	d42	d41	d40	d39	d38	d37	d36	d35	d34	d33	d32	d31	d30	d29
Parity Bit Coverage	p0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	p1	x		x		x		x		x		x		x		x		x	
	p2			x	x			x	x			x	x			x	x		
	p4	x	x					x	x	x	x					x	x	x	x
	p8							x	x	x	x	x	x	x	x				
	p16	x	x	x	x	x	x												
	p32	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	p64																		
Bit Position		35	34	33	32	31	30	29	28	27	26	25	24	23	22	21	20	19	18
Encoded Bits		d28	d27	d26	p32	d25	d24	d23	d22	d21	d20	d19	d18	d17	d16	d15	d14	d13	d12
Parity Bit Coverage	p0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	p1	x		x		x		x		x		x		x		x		x	
	p2	x	x			x	x			x	x			x	x			x	x
	p4					x	x	x	x					x	x	x	x		
	p8					x	x	x	x	x	x	x	x						
	p16					x	x	x	x	x	x	x	x	x	x	x	x	x	x
	p32	x	x	x	x														
	p64																		
Bit Position		17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Encoded Bits		d11	p16	d10	d9	d8	d7	d6	d5	d4	p8	d36	d2	d1	p4	d0	p2	p1	p0
Parity Bit Coverage	p0	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
	p1	x		x		x		x		x		x		x		x		x	
	p2			x	x			x	x			x	x			x	x		
	p4			x	x	x	x					x	x	x	x				
	p8			x	x	x	x	x	x	x	x								
	p16	x	x																
	p32																		
	p64																		

Table 8-29. Encoder and Decoder Data IN and OUT Positioning

ENCODER			
DATA IN	**Encoded Bits**	**DATA OUT**	**Bit Positions**
OTP_ECC_DATAIN 1	d0 to d7	OTP_ECC_DATAOUT 1	0 to 7
OTP_ECC_DATAIN 2	d8 to d15	OTP_ECC_ DATAOUT 2	8 to 15
OTP_ECC_DATAIN 3	d16 to d23	OTP_ECC_ DATAOUT 3	16 to 23
OTP_ECC_DATAIN 4	d24 to d31	OTP_ECC_ DATAOUT 4	24 to 31
OTP_ECC_DATAIN 5	d32 to d39	OTP_ECC_ DATAOUT 5	32 to 39
OTP_ECC_DATAIN 6	d40 to d47	OTP_ECC_ DATAOUT 6	40 to 47
OTP_ECC_DATAIN 7	d48 to d55	OTP_ECC_ DATAOUT 7	48 to 55
OTP_ECC_DATAIN 8	d56 to d63	OTP_ECC_ DATAOUT 8	56 to 63
		OTP_ECC_ DATAOUT 9	64 to 71
DECODER			
DATA IN	**Bit Positions**	**DATA IN**	**Encoded Bits**
OTP_ECC_DATAIN 1	0 to 7	OTP_ECC_DATAOUT 1	d0 to d7
OTP_ECC_DATAIN 2	8 to 15	OTP_ECC_ DATAOUT 2	d8 to d15
OTP_ECC_DATAIN 3	16 to 23	OTP_ECC_ DATAOUT 3	d16 to d23
OTP_ECC_DATAIN 4	24 to 31	OTP_ECC_ DATAOUT 4	d24 to d31
OTP_ECC_DATAIN 5	32 to 39	OTP_ECC_ DATAOUT 5	d32 to d39
OTP_ECC_DATAIN 6	40 to 47	OTP_ECC_ DATAOUT 6	d40 to d47
OTP_ECC_DATAIN 7	48 to 55	OTP_ECC_ DATAOUT 7	d48 to d55
OTP_ECC_DATAIN 8	56 to 63	OTP_ECC_ DATAOUT 8	d56 to d63
OTP_ECC_DATAIN 9	64 to 71		

ECC Diagnostic Test: The device provides a diagnostic tool to test the ECC function. There are two modes that are available to run the diagnostic. The first, auto mode (*OTP_ECC_TEST[MANUAL_AUTO]* = 0), uses internal data to run the tests. In auto mode, the *OTP_ECC_TEST[DED_SEC]* bit selects the type of test that is to be performed and the *OTP_ECC_TEST[ENC_DEC]* bit determines if the encoder or decoder function is to be tested. The result of the ECC test is provided in the *OTP_ECC_DATAOUT** registers. The test steps and expected results from each test are shown below.

Automatic Decoding steps:

1. Set ECC Test to automatic *OTP_ECC_TEST[MANUAL_AUTO]* = 0
2. Set decoder setting *OTP_ECC_TEST[ENC_DEC]* = 0
3. Set decoder to single or double encoding setting with *OTP_ECC_TEST[DED_SEC]* (1 for DED or 0 for SEC)
4. Clear all SEC/DED faults by *FAULT_RST2[RST_OTP_DATA]* = 1
5. Enable ECC test *OTP_ECC_TEST[ENABLE]* = 1
6. Read *FAULT_OTP[SEC_DET]* flag for SEC or *FAULT_OTP[DED_DET]* flag for DED
7. Block read *OTP_ECC_DATAOUT1* to *OTP_ECC_DATAOUT8* to verify the decoder test results as in Table 8-30
8. Disable ECC test *OTP_ECC_TEST[ENABLE]* = 0

Automatic Encoding steps:

1. Set ECC TEST to automatic *OTP_ECC_TEST[MANUAL_AUTO]* = 0
2. Set the encoder setting using *OTP_ECC_TEST[ENC_DEC]* = 1
3. Enable the ECC test with *OTP_ECC_TEST[ENABLE]* = 1
4. Block read *OTP_ECC_DATAOUT1* to *OTP_ECC_DATAOUT9* to verify the encoder test results as in Table 8-30
5. Disable ECC test *OTP_ECC_TEST[ENABLE]* = 0

Table 8-30. Decoder and Encoder Test Verification

[DED_SEC]	*[ENC_DEC]*	*[SEC_DET]*	*[DED_DET]*	*OTP_DATAOUT**
0 (SEC test)	0 (Decoder test)	1	0	0x18C3 FF8A 68A9 8069
0 (SEC test)	1 (Encoder test)	N/A	N/A	0xCD 3968 C140 2EA5 ED6D
1 (DED test)	0 (Decoder test)	0	1	0x0000 0000 0000 0000
1 (DED test)	1 (Encoder test)	N/A	N/A	0xCD 3968 C140 2EA5 ED6D

8.3.5.4.5 OVUV Detection Check

8.3.5.4.5.1 Parity Check

When the OVUV protectors are enabled, the register settings related to the OVUV configurations are latched to protector blocks. The device will check periodically in the background to ensure the latched configurations remain the same throughout the protector operation.

The parity check covers the following latched setting. If a parity fault in the OVUV protector is detected, the device will set the *FAULT_PROT1[VPARITY_FAIL]* = 1.

Table 8-31. Protector Parity Check Settings

OVUV Protector	Note
OV threshold, UV threshold	Ensure threshold settings remains the same during the operation
OVUV_MODE setting	Ensure the protector doesn't switch to a different operation mode

8.3.5.4.5.2 OVUV DAC Check

The OV, UV DAC values are multiplexed to the AUX ADC from which the host can read out the values as part of the diagnostic check on the protector threshold settings.

To measure the protector's DAC value, it is recommended to lock the OVUV protectors to a single channel through *OVUV_CTRL[OVUV_LOCK3:0]* for OV and UV DAC measurement and restart the OVUV protector to run in the single channel run mode. Host ensures the VS channel is not under OV or UV fault. Otherwise, the DAC measurement will not be reflecting the triggering threshold value. Note that the OV and UV DAC value is (0.8 x the threshold setting).

8.3.5.4.5.3 OVUV Protector BIST

The device implemented an OVUV BIST (Built-In-Self-Test) function to test the primary OVUV protector path. Host can start the BIST run by setting *[OVUV_MODE1:0]* = 0b10 and *[OVUV_GO]* = 1. The BIST run covers:

1. OV and UV comparators thresholds:
   a. A higher and lower than the set threshold are checked to ensure the comparator is triggered correctly.
   b. If failure is detected, the corresponding *FAULT_PROT2[OVCOMP_FAIL]* or *[UVCOMP_FAIL]* bit will be set.
2. The path from the OVUV MUX to UV fault status bit and NFAULT pin:
   a. For each VS channel, a switch is open so that input to the OVUV MUX is open and will lead to a UV detection to the channel under test
   b. The BIST engine then checks the logic to assert corresponding *FAULT_UV* register bit and the NFAULT is set properly.
   c. The BIST engine resets the corresponding *FAULT_UV* bit and deasserts the NFAULT, then switches to test the next channel and repeats the process until all active channels are tested.
   d. If failure is detected, the corresponding *[VPATH_FAIL]* bit is set.
3. OV fault bit and NFAULT path
   a. The BIST engine forces 1 to the *FAULT_OV** register, one bit at time, to ensure each *FAULT_OV** register bit can be set and the NFAULT can be asserted, accordingly.
   b. If failure is detected, the corresponding *[VPATH_FAIL]* bit will be set.

If NFAULT is enabled, host observes NFAULT toggling during the BIST run. Upon completion of the BIST run, the OVUV comparators will be turned off. Host starts the regular OVUV mode by sending *[OVUV_GO]* = 1 with *[OVUV_MODE1:0]* = 0b11 (single channel run mode).

Note

- If a *[OVUV_GO]* = 1 is sent during the OVUV BIST run, device will execute the new GO command based on the *[OVUV_MODE1:0]* setting.
- Before starting the OVUV Protector BIST, host masks out all the non-OVUV related faults, and ensures there are no OV and UV faults on any VS channels (recommended all VS voltages to be at least 100 mV apart from the OV or UV threshold during the BIST run). Otherwise, the BIST result is not invalid.
- After BIST starts, if pre-existing fault is detected before starting step 2, the BIST engine will be aborted and the *FAULT_PROT2[BIST_ABORT]* = 1.
- A no reset option, *DIAG_PROT_CTRL[PROT_BIST_NO_RST]* = 1, is available to command the BIST engine not to reset the fault status and NFAULT pin after testing each channel. If a BIST run fails, host can select this option and re-run BIST to detect which VS channel path is unable reflect a fault condition in the fault registers.

8.3.5.4.6 Diagnostic Through ADC Comparison

8.3.5.4.6.1 VS Voltage Measurement Check

VS voltage measurement path comparison:

The VS voltage measurement check is performed by comparing the prefiltered measurement result from Main ADC versus measurement result from AUX ADC. To read the compared value measured by Main ADC and AUX ADC, MCU has to set up this diagnostic check to lock on a single channel using *[AUX_IN_SEL]* setting and the start this diagnostic check. In this configuration, the compared values from Main ADC and AUX ADC are reported to *DIAG_MAIN_HI/LO* registers and *DIAG_AUX_HI/LO* registers respectively.

Both Main and AUX ADC has the same front end filters. This diagnostic time is mostly spend on waiting for the AAF on the AUX ADC path to settle. The *[AUX_SETTLE]* setting allows the MCU to make trade-off between diagnostic time and noise filter level. Additionally, when AUX ADC starts, by default, AUX slot always align to the Main ADC VS1 slot. The *[AUX_IN_ALIGN]* setting allows MCU to change this alignment to Main ADC VS8 slot, resulting with less sampling time delta between Main and AUX ADC on the higher channels. The device does not do on-chip measurement check for SRP/SRN signal.

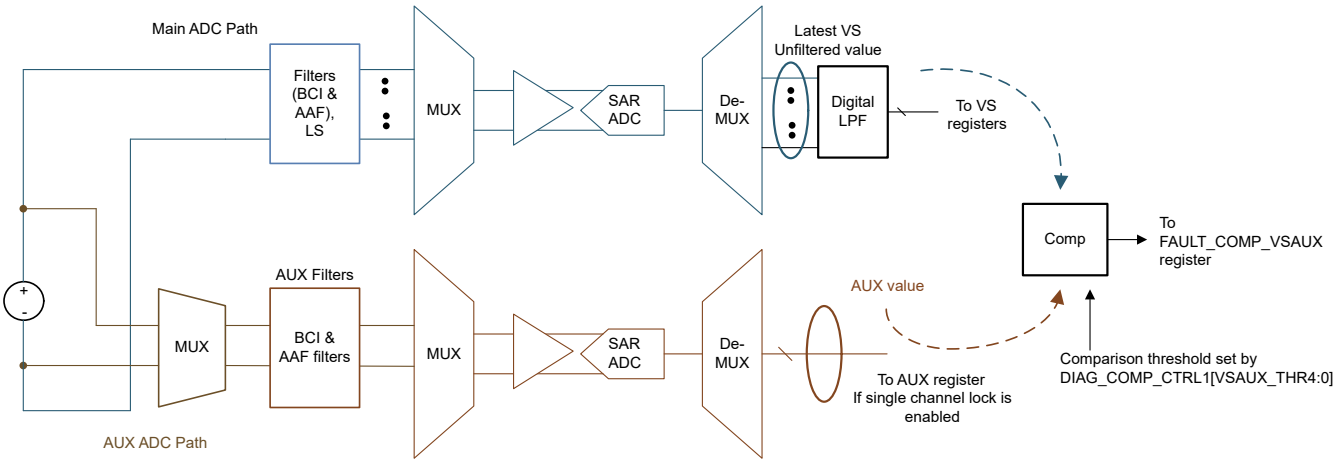


Figure 8-40. VS Voltage Measurement Diagnostic

Before starting the voltage measurement comparison between VS and AUX, host ensures:

- The desired AUX channels to be tested are configured in the *ADC_CTRL2[AUX_IN_SEL4:0]* setting and AUX ADC is enabled and in continuous mode.
- Allow AUX ADC to run through all AUX channels for the devie to compensate for common mode error before starting this diagnostic check.
- Main ADC must be enabled and is in continuous mode.

- Select the (VS – AUX) comparison threshold through *DIAG_COMP_CTRL1[VSAUX4:0]* setting.
- Select the desired settling time for the AUX channel through *ADC_CONF1[AUX_SETTLE1:0].*

To start the VS and AUX voltage measurement comparison:

1. Set *DIAG_COMP_CTRL3[COMP_ADC_SEL2:0]* = VS voltage measurement check (that is, 0b001) and set *[COMP_ADC_GO]* = 1.
2. For each channel enabled by *[AUX_IN_SEL4:0]*, the device will compare abs[(VS – AUX)] < *[VSAUX4:0]*.
3. Wait for the comparison to be accomplished, roughly [(number of channel) * (AUX settling time + one round robin cycle time)].
4. The input voltage measurement comparison is completed when *ADC_STAT2[DRDY_VSAUX]* = 1.

Host checks the *FAULT_COMP_VSAUX1* and *FAULT_COMP_VSAUX2* registers for the comparison result.

ADC comparison abort conditions:

The device will not start the VS voltage measurement comparison under the invalid conditions listed below. When the comparison is aborted, the *FAULT_COMP_MISC[COMP_ADC_ABORT]* = 1, *[DRDY_AUX_IN]* = 1, *[DRDY_VSAUX]* = 1, and *FAULT_COMP_VSAUX1/2* registers = 0xFF. If *[AUX_IN_SEL4:0]* is set to locked at a single channel, the *AUX_IN_HI/LO* registers will be reset to default value 0x8000 if the comparison run is aborted.

Invalid conditions or settings which will prevent the start of the VS voltage measurement comparison:

- Invalid *[AUX_IN_SEL]* setting: results in no AUX ADC measurement on the selected channel. The *AUX_IN_HI/LO* registers are kept in default value.
- Main or AUX ADCs are off or not set in continuous mode.

Post-ADC digital LPF check:

The digital LPF is checked continuous whenever the Main ADC is running. A duplicate diagnostic LPF is implemented to check against each LPF for each VS channel and the CSAUX channel. The check is performed with one LPF at a time.

Example, to test LPF1 for VS1, the input (that is, ADC measurement result from VS1) is fed to the LPF1 and the diagnostic LPF for a period of time. The output of the LPF1 and the diagnostic LPF are compared against each other. Several outputs from LPF1 and diagnostic LPF will be compared to ensure the operation of the LFP1 before moving to check the next LFP. If any of the LPFs fail the diagnostic check, *FAULT_COMP_MISC[LPF_FAIL]* = 1.

When the LPF for each VS channel is tested once, *ADC_STAT2[DRDY_LPF]* = 1. This diagnostic check of the LPFs will continuously run in the background as long as the Main ADC is running.

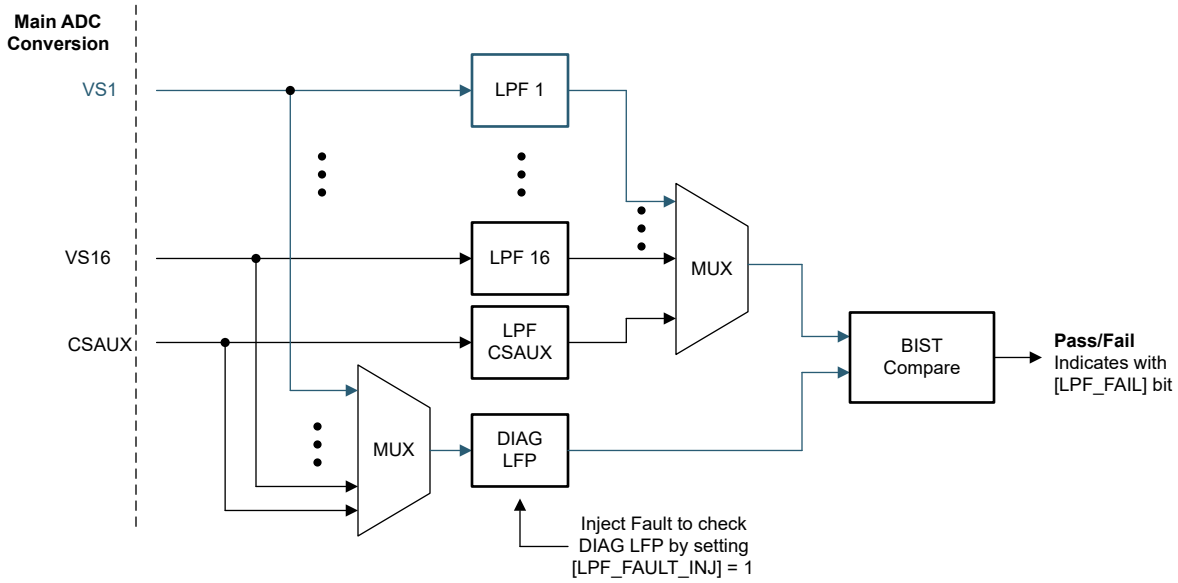


Figure 8-41. Post-ADC LPF Diagnostic (Blue Path as Example of Checking LPF1)

Furthermore, the device also implements a check to verify the functionality of the diagnostic LPF itself. By setting *DIAG_COMP_CTRL4[LPF_FAULT_INJ]* = 1 and restarting the Main ADC, the device will inject a fault into the diagnostic LPF, forcing a failure during the LPF diagnostic check which then sets the *[LPF_FAIL]* = 1. When the test is completed, simply set the *[LPF_FAULT_INJ]* = 0.

8.3.5.4.6.2 Temperature Measurement Check

Similar to the VS voltage measurement check, the device checks the thermistor temperature measurement by comparing the Main ADC measurement to the AUX ADC measurement. To read the compared value measured by Main ADC and AUX ADC, MCU has lock on a single channel using *[AUX_GPIO_SEL]* setting and the start this diagnostic check. In this configuration, the compared values from Main ADC and AUX ADC are reported to *DAIG_MAIN_HI/LO* registers and *DIAG_AUX_HI/LO* registers respectively.

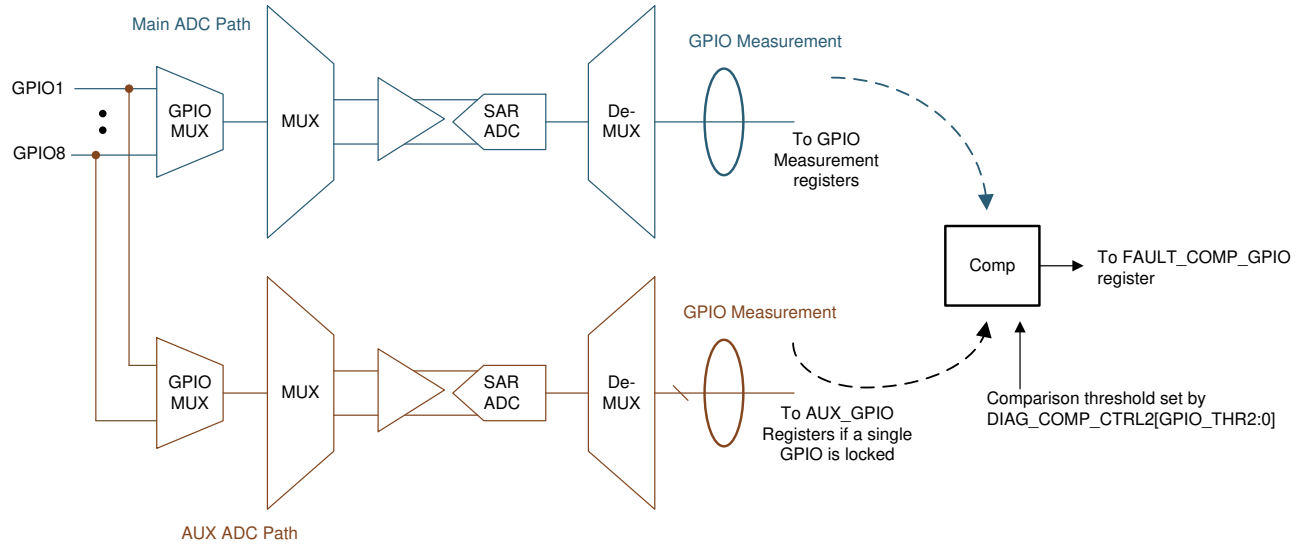


Figure 8-42. Thermistor Temperature (GPIO) Measurement Diagnostic

Before starting the temperature measurement comparison, host ensures:

- Main ADC must be enabled and is in continuous mode.

- The desired GPIO channels to be tested are configured in the *ADC_CTRL3[AUX_GPIO_SEL3:0]* setting and AUX ADC is enabled and in continuous mode.
- Select the comparison threshold through *DIAG_COMP_CTRL2[GPIO_THR2:0]* setting.

To start the voltage measurement comparison:

1. Set *DIAG_COMP_CTRL3[COMP_ADC_SEL2:0]* = GPIO measurement check (that is, 0b101) and set *[COMP_ADC_GO]* = 1.
2. For each channel enabled by *[AUX_GPIO_SEL4:0]*, the device will compare abs[(GPIO from Main – GPIO from AUX)] < *[GPIO_THR2:0]*.
3. Wait for the comparison to be accomplished which can take up to 64 ADC round robin times.
4. The GPIO measurement comparison is completed when *ADC_STAT2[DRDY_GPIO]* = 1.

Host checks the *FAULT_COMP_GPIO* register for the comparison result.

ADC comparison abort conditions:

The device will not start the temperature measurement comparison under the invalid conditions listed below. When the comparison is aborted, the *FAULT_COMP_MISC[COMP_ADC_ABORT]* = 1, *[DRDY_GPIO]* = 1, and *FAULT_COMP_GPIO* = 0xFF. If *[AUX_GPIO_SEL3:0]* is set to locked at a single channel, the *AUX_GPIO_HI/LO* registers will be reset to default value 0x8000 if the comparison run is aborted.

Invalid conditions or settings which will prevent the start of the temperature measurement comparison:

- Invalid *[AUX_GPIO_SEL]* setting which the selected GPIO is not configured for ADC measurement. The *AUX_GPIO_HI/LO* registers are kept in default value. This also applies to the case if *[AUX_GPIO_SEL]* is selected for all GPIOs but none of the GPIOs are configured for ADC measurement.
- Main or AUX ADCs are off or not set in continuous mode.

8.3.5.4.6.3 VS and AUX Open Wire Check

The device can detect an open wire connection on the VS and AUX pins. A current sink is connected to each VS and AUX pin, except VS0 and AUX0 pins which are connected with a current source.

When the current sink (or current source) is enabled and if there is an open wire connection, the VS voltage measurement will drop to an abnormal level over time. Similar detection concept applies to the VS0 and AUX0 pins with a current source. If there is an open wire connection, the VS0 or AUX0 will be pulled up by the current source, resulting in a reduced voltage measurement over time.

When the diagnostic comparison is enabled, the device will compare the voltage measurement from Main ADC (for VS pins open wire detection) against a host-programmed threshold; or comparing the AUX measurement from the AUX ADC (for AUX pins open wire detection) against a host-programmed threshold.

If MCU lock to a single AUX channel though *[AUX_IN_SEL]* before starting the AUX open wire check. The device will report the AUX measurement used for the check comparison. The value is reported in *DIAG_AUX_HI/LO* registers. Since there is no single channel lock mechanism in Main ADC, VS channel measurement used for VS open wire will not be reported in *DIAG_MAIN_HI/LO* registers.

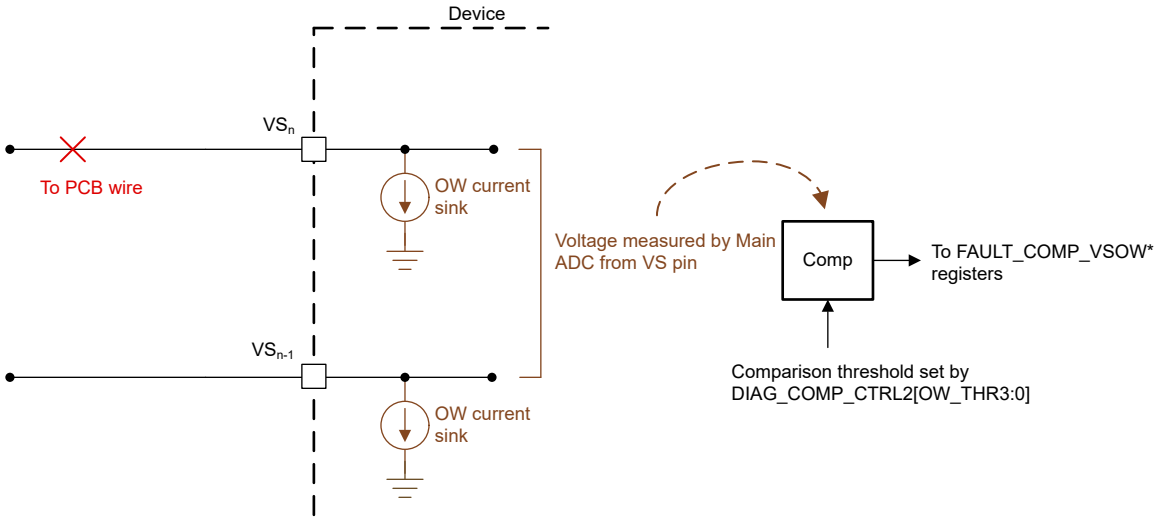


Figure 8-43. Open Wire Detection

Before starting the open wire comparison, host ensures:

- For VS open wire detection, Main ADC is running in continuous mode.
- For AUX open wire detection, AUX ADC is running in continuous mode
  - Configured in the *ADC_CTRL2[AUX_IN_SEL4:0]* to select the AUX channels
  - Select the desired settling time for the AUX channel through *ADC_CONF1[AUX_SETTLE1:0]*.
- Configure the open wire detection threshold through *DIAG_COMP_CTRL2[OW_THR3:0]*.

To start the open wire comparison:

1. Turn on the VS pins (or AUX pins) current sink or source through *DIAG_COMP_CTRL3[OW_SNK1:0]*.
2. Wait for dV/dt time of the external capacitor to deplete to the detection threshold if there is an open wire fault.
3. For VS open wire detection, select *DIAG_COMP_CTRL3[COMP_ADC_SEL2:0]* = OW VS check (that is, 0b010) and set *[COMP_ADC_GO]* = 1. Or for AUX open wire detection, *[COMP_ADC_SEL2:0]* = OW AUX check (that is, 0b011).
4. The device compares all active VS measurement (for VS open wire) or AUX measurement (for AUX open wire) against the *[OW_THR3:0]* threshold setting.
5. When the comparison is completed, *ADC_STAT2[DRDY_VSOW]* = 1 for VS open wire (or *[DRDY_AUXOW]* = 1 for AUX open wire).
6. Host then turns off all current sinks and sources through *DIAG_COMP_CTRL3[OW_SNK1:0]*.

Host checks the *FAULT_COMP_VSOW1/2* or *FAULT_COMP_AUXOW1/2* registers for the comparison result.

8.4 Device Functional Modes

The device has three power modes plus an POR state.

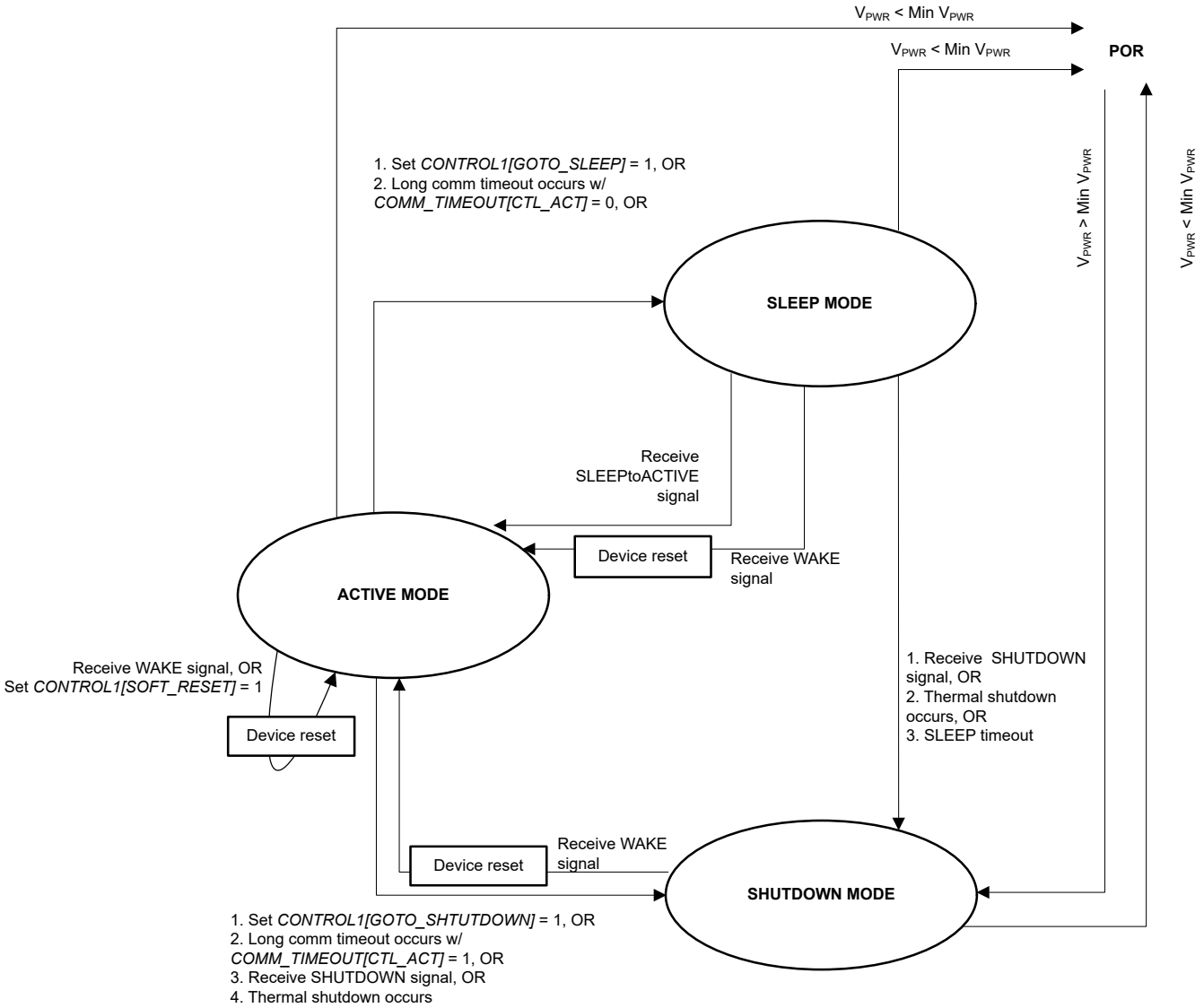
- POR: This is not a power mode. This is a condition in which the voltage at the PWR pin is less than VPWR min, and all circuits including the AVAO_REF block in the device are powered off.
- SHUTDOWN: This is the lowest power mode. AVDD, DVDD and CVDD supplies are off. Only a gross regulation at LDOIN pin is maintained. CVDD pin is will have a similar voltage as the LDOIN pin through internal circuit in order to support WAKE detection.
- SLEEP: This is the low power operation mode. Only limited functions are available.
- ACTIVE: This is the full power operation mode. All functions are supported under this state.

The various functions supported under different power modes are summarized in Table 8-32 and the power state diagram is shown in Figure 8-44.

Table 8-32. Active Functions Summary

Functional Block	SHUTDOWN	SLEEP	ACTIVE	POR
Main ADC			√	This is not a power state. All circuits are off. A sufficient voltage on VPWR will POR the device and put it to SHUTDOWN mode
AUX ADC			√	
OV/UV protector		√(1)	√	
UART			√	
Comm Vertical Communication			√	
Fault Status and NFAULT Communication		√	√	
Comm timeout			√	
SLEEP timeout		√		
Thermal Shutdown Detection		√	√	
SPI Master			√	
OTP programming			√	
Always-on block to detect POR of the device	√	√	√	

(1) To enable OV/UV protector in SLEEP mode, host must enable the function(s) in ACTIVE mode first, then put the device to SLEEP.



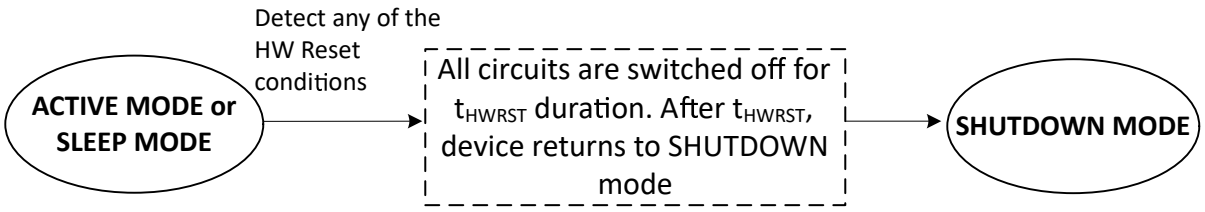


Figure 8-44. Power State Diagram

8.4.1 Power Modes

8.4.1.1 SHUTDOWN Mode

This is the lowest power mode. In SHUTDOWN mode, most of the functions are off. The device remains idle to simply monitor the WAKE ping/tone (see Section 8.4.3 for details) to wake up from this state. Only a gross regulation on LDOIN and CVDD pins are maintain for WAKE ping/tone detection.

8.4.1.1.1 Exit SHUTDOWN Mode

Communication is not supported in SHUTDOWN mode, host must send a WAKE ping or WAKE tone to enter ACTIVE mode. Once device transitions from SHUTDOWN mode to ACTIVE mode, the following table indicates the expected fault bits being set under such transition has occurred.

Table 8-33. Expected Fault Bit After Device Wake From SHUTDOWN

Device Position In The Daisy-Chain	Expected Fault Bits After Waking Up From SHUTDOWN	
Base device	*FAULT_SYS[DRST]* = 1	Digital reset by the wake ping
	FAULT_COMM3[FCOMM_DET] = 1	*[DRST]* = 1 from the upper device
	FAULT_COMM1[COMMCLR_DET] = 1	UART engine is reset
Stack device (except top of stack)	*FAULT_SYS[DRST]* = 1	Digital reset by the wake tone
	FAULT_COMM3[FCOMM_DET] = 1	*[DRST]* = 1 from the upper device
Top of stack device	*FAULT_SYS[DRST]* = 1	Digital reset by the wake tone

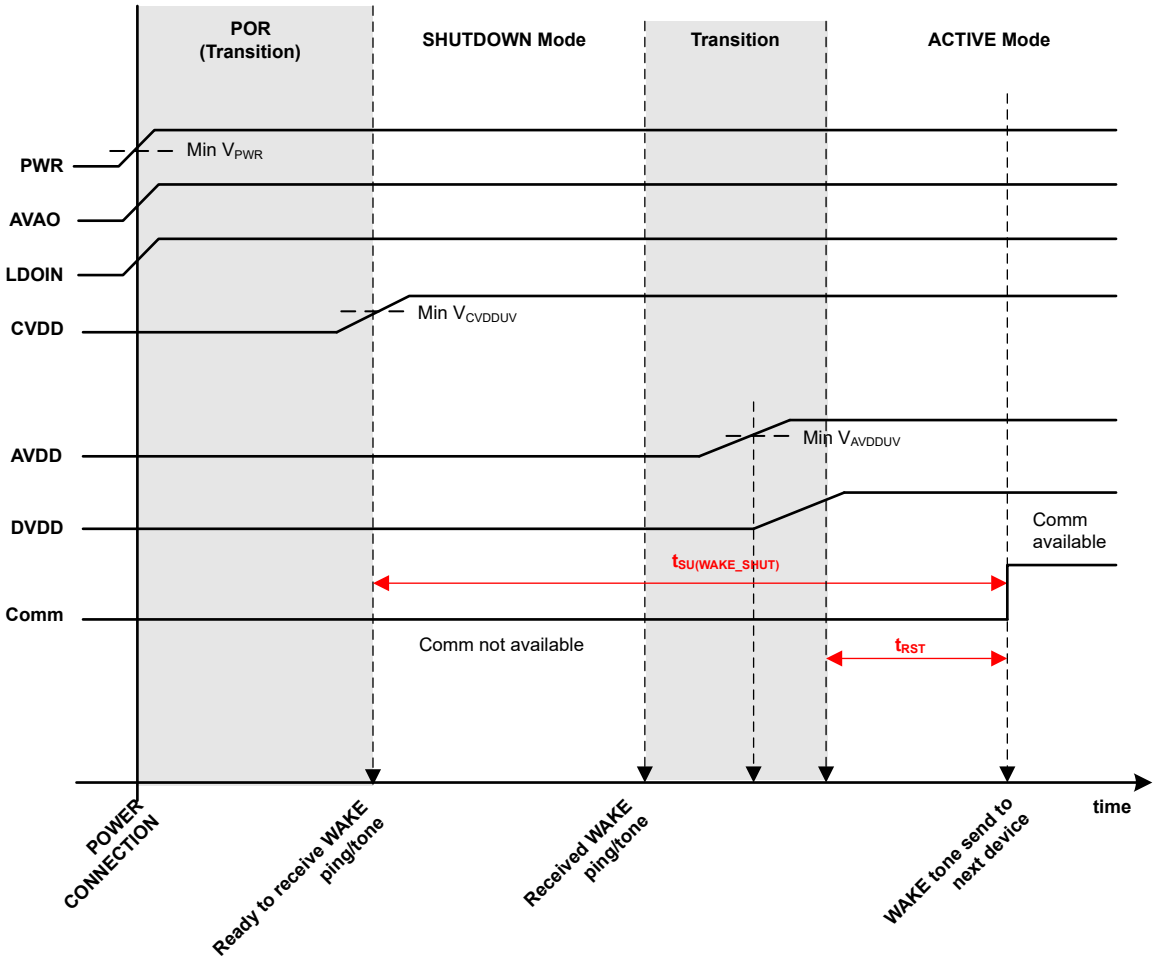


Figure 8-45. SHUTDOWN to ACTIVE Mode Transition

8.4.1.1.2 Enter SHUTDOWN Mode

During normal operation, host puts the device in SHUTDOWN mode through communication by sending *CONTROL1[GOTO_SHUTDOWN]* = 1. In a daisy-chain configuration, using broadcast write to send this command will put all devices in the daisy-chain in SHUTDOWN mode.

The device can also enter SHUTDOWN mode by one of the following conditions:

- Communication timeout: automatically transitions from ACTIVE mode to SHUTDOWN mode if there is no valid communication for the configured time. Host can enable this option through the *COMM_TIMEOUT_CONF* register.
- SLEEP mode timeout: automatically transitions from SLEEP mode to SHUTDOWN mode if device is in SLEEP mode for the configured time. Host can enable this option through *PWR_TRANSIT_CONF[SLP_TIME2:0]*.
- Thermal shutdown: shuts down the device when the internal die temperature is greater than T_{SHUT}
- SHUTDOWN or HW_RESET ping/tone: These pings/tones are used as a recovery attempt on a loss communication situation. A SHUTDOWN ping/tone puts the device into SHUTDOWN mode without using communication, forcing most of the circuits to be off. A more aggressive recovery attempt uses HW_RESET ping/tone which turns off all circuits except a bandgap and restarts the device in SHUTDOWN mode.

When a device is shutting down through SHUTDOWN or HW_RESET ping/tone, upon device wake-up, the *FAULT_SYS[SHUTDOWN_REC]* bit is set to indicate the previous shutdown is abnormal. For the base device, when device wakes up with *[SHUTDOWN_REC]* = 1, the COMH and COML ports are disabled thus isolating the base device from the daisy-chain to help checking on any communication issue. Once the host confirms the

communication to the base device is good, host can issue a WAKE ping to reset the entire daisy-chain, which will clear the *[SHUTDOWN_REC]* bit as well as put the COMH/L ports of the base device back to their normal condition.

If *[SHUTDOWN_REC]* = 1 is on any of the stack devices, the COML and COMH will not be disabled upon device wake-up as the COMH/L ports are the only way to communicate to a stack device.

8.4.1.2 SLEEP Mode

This is the low power operation mode. In SLEEP mode, all internal power supplies are still on, but functions are limited to OVUV protectors, Heartbeat/Fault Tone/NFAULT transmission and detection.

8.4.1.2.1 Exit SLEEP Mode

Because host cannot communicate to the device, to exit SLEEP mode, host must send either a WAKE ping/tone or SLEEPtoACTIVE ping/tone to transition to ACTIVE mode. A WAKE wakes up and resets the device, which host will need to reconfigure the device setting; a SLEEPtoACTIVE only wakes up the device.

8.4.1.2.2 Enter SLEEP Mode

The device can enter SLEEP mode from ACTIVE mode only. During normal operation, host puts the device to SLEEP mode through communication by sending *CONTROL1[GOTO_SLEEP]* = 1. In a daisy-chain configuration, using broadcast write to send this command will put all devices into SLEEP mode.

The device can also enter SLEEP mode in the following condition:

- Communication timeout: automatically transitions from ACTIVE mode to SLEEP mode if there is no valid communication for the configured time. Host can enable this option through the *COMM_TIMEOUT_CONF* register.

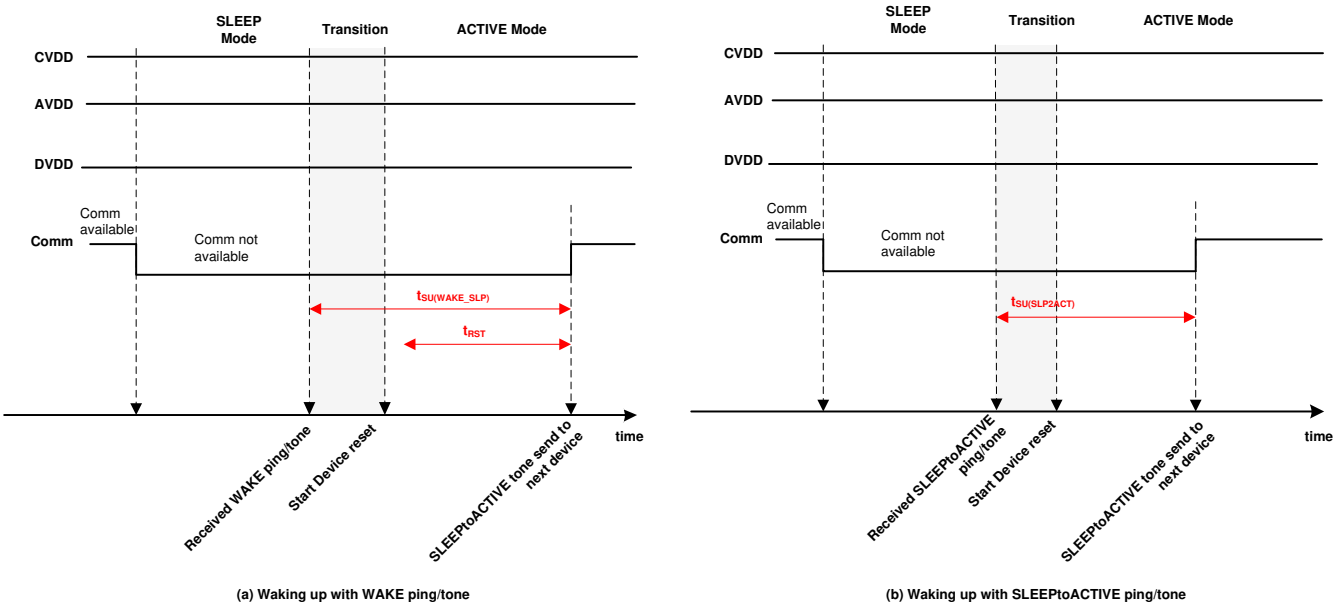


Figure 8-46. SLEEP to ACTIVE Mode Transition

8.4.1.3 ACTIVE Mode

This is the operation mode with full functionality support. Host can communicate to the device with full control on various features as well as performance diagnostic in this mode.

8.4.1.3.1 Exit ACTIVE Mode

From ACTIVE mode, device can enter SLEEP mode or SHUTDOWN mode through command, ping/tone, timer, or specific event. See Section 8.4.1.1 and Section 8.4.1.2 for details.

8.4.1.3.2 Enter ACTIVE Mode From SHUTDOWN Mode

Device can transition to ACTIVE mode from SHUTDOWN mode only through a WAKE ping/tone. Once in ACTIVE mode, host clears some of the reset-related faults which are expected faults (see Section 8.4.1.1 for details) indicating a POR on certain blocks due to the transition from SHUTDOWN mode to ACTIVE mode. Registers are reset to default; the OTP shadow registers are reloaded with the OTP programmed values.

8.4.1.3.3 Enter ACTIVE Mode From SLEEP Mode

From SLEEP mode, either a WAKE or SLEEPtoACTIVE ping/tone can put the device in ACTIVE mode. A WAKE ping/tone will generate a digital reset to the device. Because the LDO supplies remain on during SLEEP mode, only the *FAULT_SYS[DRST]* = 1 is set, indicating a digital reset has occurred. Certain expected faults related to being reset are set. See SHUTDOWN mode for detail. Registers are reset to default, the OTP shadow registers are reloaded with the OTP programmed values.

If a SLEEPtoACTIVE ping/tone is used to wake up the device from SLEEP mode to ACTIVE mode, device will simply enter ACTIVE mode without digital resetting but the UART engine will be reset. Hence, in the base device, the *FAULT_COMM1[COMMCLR_DET]* = 1 and host clears it after entering ACTIVE mode.

8.4.2 Device Reset

There are several conditions which the device will go through: a digital reset, putting the registers to their default settings and reloading the OTP.

- A WAKE ping/tone is sent to transition from SHUTDOWN mode or SLEEP mode to ACTIVE mode.
- A WAKE ping/tone is received in ACTIVE mode.
- The *CONTROL1[SOFT_RESET]* = 1 command is sent in ACTIVE mode.
- A HW_RESET ping/tone is sent under any power mode. This generates a POR-like event to the device. Upon the detection of a HW_RESET ping/tone, the device will turn off all internal blocks except a bandgap for t_{HWRST} duration. Afterward, the device will restart in SHUTDOWN mode.
- Internal power supply faults. See Section 8.3.5.4 for details.
  - AVDD UV, DVDD UV is detected.
- A HFO or LFP watchdog fault will reset the digital.

Apart from the full reset cases, the following conditions will only reset the UART engine. These conditions mainly affect the base device because UART is used to talk to the host MCU. In the base device, the *FAULT_COMM1[COMMCLR_DET]* = 1 will be set. These conditions do not affect the stack devices because UART is inactive in those devices.

- A SLEEPtoACTIVE ping is sent to transition from SLEEP mode to ACTIVE mode.
- The following conditions not only clear the UART engine and set the *[COMMCLR_DET]* = 1, they also set *FAULT_COMM1[STOP_DET]* = 1 as an indication that an unexpected UART STOP is detected.
  - A SLEEPtoACTIVE ping is sent in ACTIVE mode.
  - A COMM CLEAR signal is sent. This is a dedicated signal to clear the UART engine and instruct the engine to look for a new start of communication frame. See Section 8.3.5.1.1.1 for more details.

8.4.3 Ping and Tone

In the noncommunicable conditions such as in SHUTDOWN or SLEEP mode, or in the loss of communication situations when host would like to instruct for a reset or power down as a communication recovery attempt, a Ping or Tone is used as a form of communication to the device for a specific action.

Table 8-34. Supported Ping/Tone in Different Power Modes

Ping/Tone Detection	Detected Pin(s)	SHUTDOWN	SLEEP	ACTIVE
SHUTDOWN ping	RX		√	√
SLEEPtoACTIVE ping	RX		√	√
WAKE ping	RX	√	√	√
HW_RESET ping	RX		√	√
SHUTDOWN tone	COMH/L		√	√
SLEEPtoACTIVE tone	COMH/L		√	√

Table 8-34. Supported Ping/Tone in Different Power Modes (continued)

Ping/Tone Detection	Detected Pin(s)	SHUTDOWN	SLEEP	ACTIVE
WAKE tone	COMH/L	√	√	√
HW_RESET tone	COMH/L		√	√
Fault tone	COMH/L		√	Only fault tone detection is available
HEARTBEAT	COMH/L		√	

8.4.3.1 Ping

A ping is a specific high-low-high signal to the RX pin of the device. Ping is used on the base device as only the base device is connected to the host which the UART RX is accessible. The device detects different low times of the ping signal to differentiate the different ping signals.

The communication pings are referring to the WAKE ping, SLEEPtoACTIVE ping, SHUTDOWN ping, and HW_RESET ping. These pings instruct the device to a specific power mode when normal communication is not available. By definition, a COMM CLEAR signal on the RX pin is a form of a ping. Because a COMM_CLR is to clear the UART engine, this signal is covered in Section 8.3.5.1.1.1.

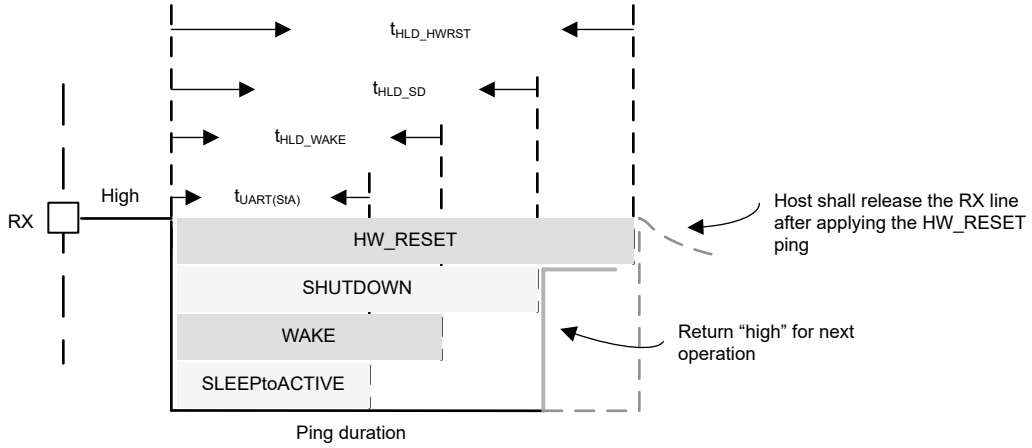


Figure 8-47. Communication Pings

8.4.3.2 Tone

A tone is a fixed number of couplets (pulses) with a specified polarity (all “+” or all “–”) sent through the differential vertical interface COMH and COML ports. Tone is used on stack devices as only the COMH/L ports are accessible. The number of couplets for transmission is always greater than the number of couplets needed for detection.

There are four communication tones corresponding to the four communication pings. They are WAKE tone, SLEEPtoACTIVE tone, SHUTDOWN tone, and HW_RESET tone. In addition to the communication tones, there are two extra tones related to device fault status: Heartbeat tone and Fault tone. These two fault status tones are only available in SLEEP mode. See Section 8.3.5.2.3.3 for details.

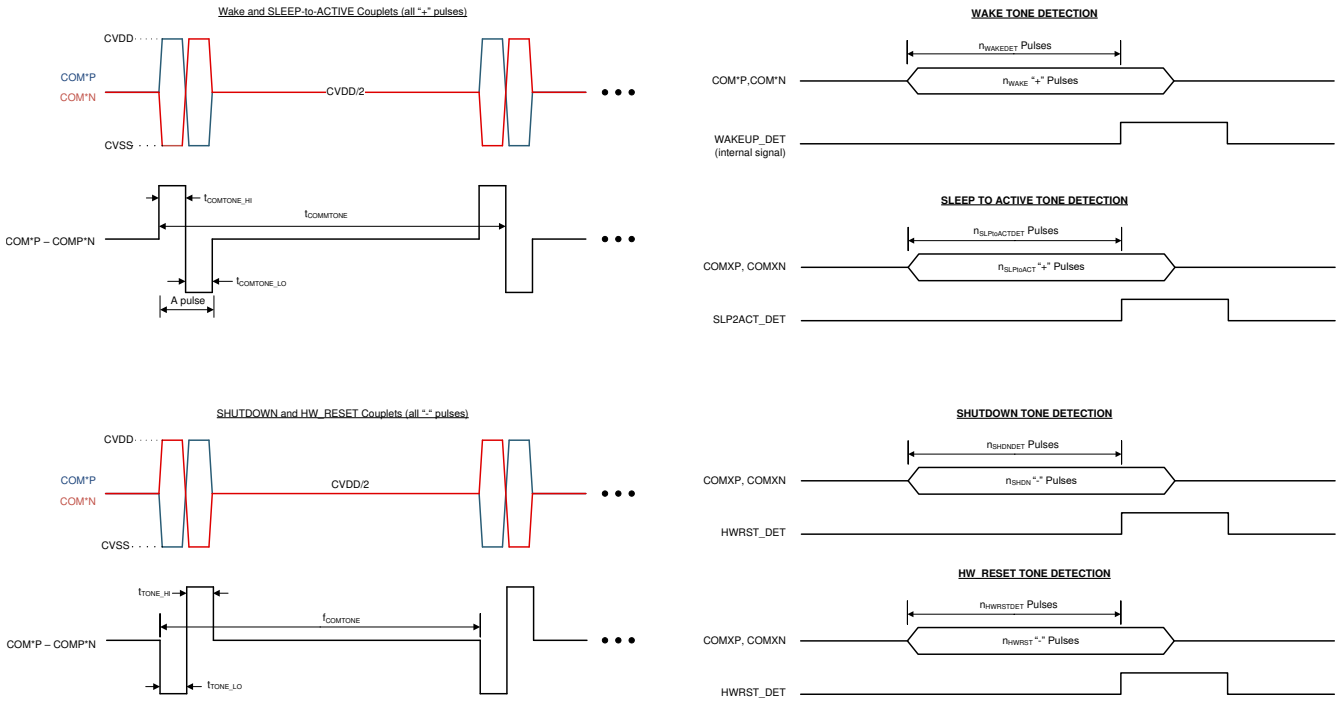


Figure 8-48. Communication Tones

8.4.3.3 Ping and Tone Propagation

Propagates:

The WAKE and SLEEPtoACTIVE pings/tones are part of the normal operation to wake up the device; hence, these two pings/tones can propagate to the next device in a daisy-chain configuration. That is, when a device receives a WAKE ping/tone, it generates a WAKE tone and forwards it to the next device. Similar action applies to SLEEPtoACTIVE ping/tone.

The direction of the tone forwarding follows the communication direction, which is set by the *CONTROL1[DIR_SEL]* bit. See Section 8.3.5.1 for more details. The detection of the tone is supported from the COMH and COML ports on stack devices regardless of the *[DIR_SEL]* setting. This does not apply to base device because base device detects pings instead.

During normal operation, host can simply send a WAKE or SLEEPtoACTIVE ping to the base device and the corresponding tone will be generated to the rest of the stack devices. During system development, if there is a need to send WAKE or SLEEPtoACTIVE to only some of the devices in the daisy-chain, host can use the *CONTROL1[SEND_WAKE]* or *CONTROL1[SEND_SLPTOACT]* bit. Device that receives this command will send the corresponding tone to the next device in the daisy-chain. Because the WAKE and SLEEPtoACTIVE tones propagate, the rest of the daisy-chain connected above also receives the corresponding tone.

Does Not Propagate:

The SHUTDOWN and HW_RESET pings/tones are mostly used as a communication recovery attempt. Hence these pings/tones do not propagate. That is, when a device receives a SHUTDOWN ping/tone, it starts the shutdown process but the device does not generate another SHUTDOWN tone to the next device. Similar action applies to HW_RESET ping/tone.

For a base device, as RX pin is connected to the host, SHUTDOWN or HW_RESET ping can be used on the base device. For stack devices, it is required at least one stack device is connected to the problem device is communicable. Host has to talk to the neighboring device and sets the *CONTROL1[SEND_SHUTDOWN]* = 1 or *CONTROL2[SEND_HW_RESET]* = 1 to instruct the neighboring device to issue the corresponding tone to the problem device. The detection of the tone is supported from the COMH and COML ports on stack devices

regardless of the *[DIR_SEL]* setting. This does not apply to a base device because a base device detects pings instead.

Table 8-35. Ping and Tone Propagation Summary

Ping/Tone	Propagable	Non-Propagable
WAKE	Receiving device will generate a WAKE tone to the next device	
SLEEPtoACTIVE	Receiving device will generate a SLEEPtoACTIVE tone to the next device	
SHUTDOWN		Receiving device will initialize the shutdown process
HW_RESET		Receiving device will initialize the HW reset process

8.5 Register Maps

This section has three register map summary tables with registers listed per the order of the register address:

- The NVM (OTP) shadow registers. These read/write-able shadow registers are reset with OTP values programmed in the customer OTP space. To program the custom OTP space, host writes the desired values to these OTP shadow registers and follows the programming procedure. These registers are included in the OTP CRC check. If customer OTP space is not programmed. The shadow registers are loaded with factory configuration default value. If the OTP (either factory configuration default or value programmed in customer OTP space) is failing to load after a device reset, the shadow registers will be loaded with the hardware reset default value instead. The hardware reset default value and the factory configuration default values are the same for the majority of the OTP shadow registers. Only the *DIR0_ADDR_OTP, DIR1_ADD_OTP*, *PWR_TRANSIT_CONF*, *CUST_CRC_HI/LO* registers have a reset value versus factory default, and are specified in Section 8.5.1 and their register field descriptions.
- The Read/Write registers. These are registers that the host can read/write to during runtime. A device reset will put these registers back to their reset value.
- The Read registers. These are registers that the host only has read access. A device reset will put these registers back to their reset value.

The register summary tables use the following key:

- Addr = Register address
- Hex = Hexidecimal value
- NVM = Non-volatile memory (OTP) shadow registers
- RSVD = Reserved. Reserved register addresses or bits are not implemented in the device. Any write to these bits is ignored. Reads to these bits always return 0.
- OTP_SPARE: These are spare OTP and shadow register bits that are implemented in the device. These spare bits are included as part of the CRC calculation. These bits are read/write as normal, but do not perform any function or influence any device behaviors.
- OTP_RSVDn = OTP and shadowed registers that are implemented but are reserved for device internal usage, where n refers to the register address. MCU must keep these registers in their default value

Section 8.5.4 describes the definition of each bit in the registers. The registers in this section are grouped per functional blocks.

8.5.1 OTP Shadow Register Summary

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DIR0_ADDR_OTP	0	NVM	HW Reset Default = 0x00 Factory Configuration Default = 0x01	SPARE[1:0]		ADDRESS[5:0]					
DIR1_ADDR_OTP	1	NVM	HW Reset Default = 0x00 Factory Configuration Default = 0x01	SPARE[1:0]		ADDRESS[5:0]					
DEV_CONF	2	NVM	0x54	RSVD	RSVD	MULTIDROP_EN	FCOMM_EN	TWO_STOP_EN	NFAULT_EN	FTONE_EN	HB_EN
OTP_RSVD	3		INTERNAL USE. DO NOT WRITE TO THIS ADDRESS								
OTP_SPARE15	4	NVM	0x00	SPARE[7:0]							
OTP_RSVD	5	NVM	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS								
OTP_RSVD	6	NVM	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS								

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_CONF1	7	NVM	0x00	AUX_SETTLE[1:0]		LPF_CSAUX[2:0]			LPF_VS[2:0]		
ADC_CONF2	8	NVM	0x00	SPARE[1:0]		ADC_DLY[5:0]					
OV_THRESH	9	NVM	0x3F	SPARE	SPARE	OV_THR[5:0]					
UV_THRESH	A	NVM	0x00	SPARE	SPARE	UV_THR[5:0]					
OTP_RSVD	B	NVM	0xE0	INTERNAL USE.							
UV_DISABLE1	C	NVM	0x00	VS16	VS15	VS14	VS13	VS12	VS11	VS10	VS9
UV_DISABLE2	D	NVM	0x00	VS8	VS7	VS6	VS5	VS4	VS3	VS2	VS1
GPIO_CONF1	E	NVM	0x00	FAULT_IN_EN	SPI_EN	GPIO2[2:0]			GPIO1[2:0]		
GPIO_CONF2	F	NVM	0x00	SPARE	CS_RDY_EN	GPIO4[2:0]			GPIO3[2:0]		
GPIO_CONF3	10	NVM	0x00	SPARE[1:0]		GPIO6[2:0]			GPIO5[2:0]		
GPIO_CONF4	11	NVM	0x00	SPARE[1:0]		GPIO8[2:0]			GPIO7[2:0]		
OTP_SPARE14	12	NVM	0x00	SPARE[7:0]							
OTP_SPARE13	13	NVM	0x00	SPARE[7:0]							
OTP_SPARE12	14	NVM	0x00	SPARE[7:0]							
OTP_SPARE11	15	NVM	0x00	SPARE[7:0]							
FAULT_MSK1	16	NVM	0x00	MSK_PROT	RSVD	RSVD	MSK_UV	MSK_OV	MSK_COMP	MSK_SYS	MSK_ PWR
FAULT_MSK2	17	NVM	0x00	SPARE[1]	MSK_OTP_CRC	MSK_OTP_DATA	MSK_COMM3_FCOMM	MSK_COMM3_FTONE	MSK_COMM3_HB	MSK_COMM2	MSK_COMM1
PWR_TRANSIT_CONF	18	NVM	HW Reset Default = 0x18 Factory Configuration Default = 0x10	SPARE[2:0]			TWARN_THR[1:0]		SLP_TIME[2:0]		
COMM_TIMEOUT_CONF	19	NVM	0x00	SPARE	CTS_TIME[2:0]			CTL_ ACT	CTL_TIME[2:0]		
TX_HOLD_OFF	1A	NVM	0x00	DLY[7:0]							
MAIN_ADC_CAL1	1B	NVM	0x00	GAINL[7:0]							
MAIN_ADC_CAL2	1C	NVM	0x00	GAINH	OFFSET[6:0]						
AUX_ADC_CAL1	1D	NVM	0x00	GAINL[7:0]							
AUX_ADC_CAL2	1E	NVM	0x00	GAINH	OFFSET[6:0]						
CS_ADC_CAL1	1F	NVM	0x00	GAINL[7:0]							

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CS_ADC_CAL2	20	NVM	0x00	GAINH[2:0]			OFFSET [4:0]				
CUST_MISC1 through CUST_MISC8	21	NVM	0x00	DATA[7:0]							
	22	NVM	0x00	DATA[7:0]							
	23	NVM	0x00	DATA[7:0]							
	24	NVM	0x00	DATA[7:0]							
	25	NVM	0x00	DATA[7:0]							
	26	NVM	0x00	DATA[7:0]							
	27	NVM	0x00	DATA[7:0]							
	28	NVM	0x00	DATA[7:0]							
STACK_RESPONSE	29	NVM	0x00	SPARE[1:0]		DELAY[5:0]					
OTP_RSVD2A	2A	NVM	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
OTP_RSVD2B	2B	NVM	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
OTP_SPARE10	2C	NVM	0x00	SPARE[7:0]							
OTP_SPARE9	2D	NVM	0x00	SPARE[7:0]							
OTP_SPARE8	2E	NVM	0x00	SPARE[7:0]							
OTP_SPARE7	2F	NVM	0x00	SPARE[7:0]							
OTP_SPARE6	30	NVM	0x00	SPARE[7:0]							
OTP_SPARE5	31	NVM	0x00	SPARE[7:0]							
OTP_SPARE4	32	NVM	0x00	SPARE[7:0]							
OTP_SPARE3	33	NVM	0x00	SPARE[7:0]							
OTP_SPARE2	34	NVM	0x00	SPARE[7:0]							
OTP_SPARE1	35	NVM	0x00	SPARE[7:0]							
CUST_CRC_HI	36	NVM	HW Reset Default = 0x57 Factory Configuration Default = 0x31	CRC[7:0]							
CUST_CRC_LO	37	NVM	HW Reset Default = 0x89 Factory Configuration Default = 0xF3	CRC[7:0]							

8.5.2 Read/Write Register Summary

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OTP_PROG_UNLOCK1A through OTP_PROG_UNLOCK1D	300	RW	0x00	CODE[7:0]							
	301	RW	0x00	CODE[7:0]							
	302	RW	0x00	CODE[7:0]							
	303	RW	0x00	CODE[7:0]							
DIR0_ADDR	306	RW	0x00	RSVD		ADDRESS[5:0]					
DIR1_ADDR	307	RW	0x00	RSVD		ADDRESS[5:0]					
COMM_CTRL	308	RW	0x00	RSVD						STACK_DEV	TOP_STACK
CONTROL1	309	RW	0x00	DIR_SEL	SEND_SHUTDOWN	SEND_WAKE	SEND_SLPTOACT	GOTO_SHUTDOWN	GOTO_SLEEP	SOFT_RESET	ADDR_WR
CONTROL2	30A	RW	0x00	RSVD						SEND_HW_RESET	TSREF_EN
OTP_PROG_CTRL	30B	RW	0x00	RSVD						PAGE SEL	PROG_GO
ADC_CTRL1	30D	RW	0x00	RSVD	CS_DR[1:0]		LPF_CSAUX_EN	LPF_VS_EN	CS_MAIN_GO	MAIN_MODE[1:0]	
ADC_CTRL2	30E	RW	0x00	RSVD	MAINAUXCS_AFE_DIS	AUX_IN_ALIGN	AUX_IN_SEL[4:0]				
ADC_CTRL3	30F	RW	0x00	RSVD	AUX_GPIO_SEL[3:0]				AUX_GO	AUX_MODE[1:0]	
REG_INT_RSVD	310	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
REG_INT_RSVD	318	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	319	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	31A	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	31B	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	31C	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	31D	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	31E	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	31F	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	320	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	321	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	322	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	323	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	324	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	325	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	326	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	327	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
REG_INT_RSVD	328	RW	0x3F	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	329	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	32A	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
	32B	RW	0x0F	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
OVUV_CTRL	32C	RW	0x00	RSVD	OVUV_LOCK[3:0]				OVUV_GO	OVUV_MODE[1:0]	
REG_INT_RSVD	32D	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
REG_INT_RSVD	32E	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
REG_INT_RSVD	32F	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
REG_INT_RSVD	330	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
FAULT_RST1	331	RW	0x00	RST_PROT	RSVD	RSVD	RST_UV	RST_OV	RST_COMP	RST_SYS	RST_PWR
FAULT_RST2	332	RW	0x00	RSVD	RST_OTP_CRC	RST_OTP_DATA	RST_COMM3_FCOMM	RST_COMM3_FTONE	RST_COMM3_HB	RST_COMM2	RST_COMM1
DIAG_OTP_CTRL	335	RW	0x00	RSVD			FLIP_FACT_CRC	MARGIN_MODE[2:0]			MARGIN_GO
DIAG_COMM_CTRL	336	RW	0x00	RSVD						SPI_LOOPBACK	FLIP_TR_CRC
DIAG_PWR_CTRL	337	RW	0x00	RSVD						BIST_NO_RST	PWR_BIST_GO
REG_INT_RSVD	338	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
REG_INT_RSVD	339	RW	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
DIAG_COMP_CTRL1	33A	RW	0x00	VSAUX_THR[4:0]					RSVD		
DIAG_COMP_CTRL2	33B	RW	0x00	RSVD	GPIO_THR[2:0]			OW_THR[3:0]			
DIAG_COMP_CTRL3	33C	RW	0x00	RSVD	RSVD	OW_SNK[1:0]		COMP_ADC_SEL[2:0]			COMP_ADC_GO
DIAG_COMP_CTRL4	33D	RW	0x00	RSVD						COMP_FAULT_INJ	LPF_FAULT_INJ
DIAG_PROT_CTRL	33E	RW	0x00	RSVD							PROT_BIST_NO_RST
OTP_ECC_DATAIN1 through OTP_ECC_DATAIN9	343	RW	0x00	DATA[7:0]							
	344	RW	0x00	DATA[7:0]							
	345	RW	0x00	DATA[7:0]							
	346	RW	0x00	DATA[7:0]							
	347	RW	0x00	DATA[7:0]							
	348	RW	0x00	DATA[7:0]							
	349	RW	0x00	DATA[7:0]							
	34A	RW	0x00	DATA[7:0]							
	34B	RW	0x00	DATA[7:0]							
OTP_ECC_TEST	34C	RW	0x00	RSVD				DED_SEC	MANUAL_AUTO	ENC_DEC	ENABLE
SPI_CONF	34D	RW	0x00	RSVD	CPOL	CPHA	NUMBIT[4:0]				
SPI_TX3, SPI_TX2, and SPI_TX1	34E	RW	0x00	DATA[7:0]							
	34F	RW	0x00	DATA[7:0]							
	350	RW	0x00	DATA[7:0]							
SPI_EXE	351	RW	0x02	RSVD						SS_CTRL	SPI_GO
OTP_PROG_UNLOCK2A through OTP_PROG_UNLOCK2D	352	RW	0x00	CODE[7:0]							
	353	RW	0x00	CODE[7:0]							
	354	RW	0x00	CODE[7:0]							
	355	RW	0x00	CODE[7:0]							

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DEBUG_CTRL_UNLOCK	700	RW	0x00	CODE[7:0]							
DEBUG_COMM_CTRL1	701	RW	0x04	RSVD			UART_BAUD	UART_MIRROR_EN	UART_TX_EN	USER_UART_EN	USER_DAISY_EN
DEBUG_COMM_CTRL2	702	RW	0x0F	RSVD				COML_TX_EN	COML_RX_EN	COMH_TX_EN	COMH_RX_EN

8.5.3 Read-Only Register Summary

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PARTID	500	R	0x00	REV[7:0]							
DEV_REVID	E00	R	0x00	DEV_REVID[7:0]							
DIE_ID1 through DIE_ID9	501	R	0x00	ID[7:0]							
	502	R	0x00	ID[7:0]							
	503	R	0x00	ID[7:0]							
	504	R	0x00	ID[7:0]							
	505	R	0x00	ID[7:0]							
	506	R	0x00	ID[7:0]							
	507	R	0x00	ID[7:0]							
	508	R	0x00	ID[7:0]							
	509	R	0x00	ID[7:0]							
CUST_CRC_RSLT_HI	50C	R	0x31	CRC[7:0]							
CUST_CRC_RSLT_LO	50D	R	0xF3	CRC[7:0]							
OTP_ECC_DATAOUT1 through OTP_ECC_DATAOUT9	510	R	0x00	DATA[7:0]							
	511	R	0x00	DATA[7:0]							
	512	R	0x00	DATA[7:0]							
	513	R	0x00	DATA[7:0]							
	514	R	0x00	DATA[7:0]							
	515	R	0x00	DATA[7:0]							
	516	R	0x00	DATA[7:0]							
	517	R	0x00	DATA[7:0]							
	518	R	0x00	DATA[7:0]							
OTP_PROG_STAT	519	R	0x00	UNLOCK	OTERR	UVERR	OVERR	SUVERR	SOVERR	PROG ERR	DONE
OTP_CUST1_STAT	51A	R	0x00	LOADED	LOAD WRN	LOAD ERR	FMTERR	PROGOK	UVOK	OVOK	TRY
OTP_CUST2_STAT	51B	R	0x00	LOADED	LOAD WRN	LOAD ERR	FMTERR	PROGOK	UVOK	OVOK	TRY
SPI_RX3, SPI_RX2, and SPI_RX1	520	R	0x00	DATA[7:0]							
	521	R	0x00	DATA[7:0]							
	522	R	0x00	DATA[7:0]							
DIAG_STAT	526	R	0x00	RSVD			RSVD	DRDY_OVUV	RSVD	DRDY_BIST_OVUV	DRDY_BIST_PWR

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADC_STAT1	527	R	0x00	RSVD			RSVD	DRDY_ AUX_ GPIO	DRDY_ AUX_ IN	DRDY_ AUX_ MISC	DRDY_ MAIN_ ADC
ADC_STAT2	528	R	0x00	RSVD		DRDY_ LPF	DRDY_ GPIO	DRDY_ VSOW	DRDY_ AUXOW	RSVD	DRDY_ VSAUX
GPIO_STAT	52A	R	0x00	GPIO8	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1
REG_INT_RSVD	52B	R	0x00	RSVD							
DEV_STAT	52C	R	0x00	RSVD	FACT_ CRC_ DONE	CUST_ CRC_ DONE	RSVD	OVUV_ RUN	CS_RUN	AUX_ RUN	MAIN_ RUN
FAULT_SUMMARY	52D	R	0x00	FAULT_ PROT	FAULT_ COMP_ ADC	FAULT_ OTP	FAULT_ COMM	RSVD	FAULT_ OVUV	FAULT_ SYS	FAULT_ PWR
FAULT_COMM1	530	R	0x00	RSVD			UART_TR	UART_ RR	UART_ RC	COMM CLR_ DET	STOP_ DET
FAULT_COMM2	531	R	0x00	COML_ TR	COML_ RR	COML_ RC	COML_ BIT	COMH_ TR	COMH_ RR	COMH_ RC	COMH_ BIT
FAULT_COMM3	532	R	0x00	RSVD				FCOMM _DET	FTONE _DET	HB_FAIL	HB_FAST
FAULT_OTP	535	R	0x00	RSVD	DED_ DET	SEC_DET	CUST_ CRC	FACT_ CRC	CUSTLD ERR	FACTLD ERR	GBLOV ERR
FAULT_SYS	536	R	0x00	LFO	RSVD	GPIO	DRST	CTL	CTS	TSHUT	TWARN
FAULT_PROT1	53A	R	0x00	RSVD						RSVD	VPARITY _FAIL
FAULT_PROT2	53B	R	0x00	RSVD	BIST_ ABORT	TPATH _FAIL	VPATH _FAIL	RSVD	RSVD	OVCOMP _FAIL	UVCOMP _FAIL
FAULT_OV1	53C	R	0x00	OV16_ DET	OV15_ DET	OV14_ DET	OV13_ DET	OV12_ DET	OV11_ DET	OV10_ DET	OV9_DET
FAULT_OV2	53D	R	0x00	OV8_DET	OV7_DET	OV6_DET	OV5_DET	OV4_DET	OV3_DET	OV2_DET	OV1_DET
FAULT_UV1	53E	R	0x00	UV16_ DET	UV15_ DET	UV14_ DET	UV13_ DET	UV12_ DET	UV11_ DET	UV10_ DET	UV9_DET
FAULT_UV2	53F	R	0x00	UV8_DET	UV7_DET	UV6_DET	UV5_DET	UV4_DET	UV3_DET	UV2_DET	UV1_DET
REG_INT_RSVD	540	R	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
REG_INT_RSVD	541	R	0x00	INTERNAL USE. DO NOT WRITE TO THIS ADDRESS							
FAULT_COMP_GPIO	543	R	0x00	GPIO8_ FAIL	GPIO7_ FAIL	GPIO6_ FAIL	GPIO5_ FAIL	GPIO4_ FAIL	GPIO3_ FAIL	GPIO2_ FAIL	GPIO1_ FAIL
FAULT_COMP_VSAUX1	545	R	0x00	VS16_ FAIL	VS15_ FAIL	VS14_ FAIL	VS13_ FAIL	VS12_ FAIL	VS11_ FAIL	VS10_ FAIL	VS9_ FAIL
FAULT_COMP_VSAUX2	546	R	0x00	VS8_ FAIL	VS7_ FAIL	VS6_ FAIL	VS5_ FAIL	VS4_ FAIL	VS3_ FAIL	VS2_ FAIL	VS1_ FAIL
FAULT_COMP_VSOW1	548	R	0x00	VSOW16 _FAIL	VSOW15 _FAIL	VSOW14 _FAIL	VSOW13 _FAIL	VSOW12 _FAIL	VSOW11 _FAIL	VSOW10 _FAIL	VSOW9 _FAIL
FAULT_COMP_VSOW2	549	R	0x00	VSOW8 _FAIL	VSOW7 _FAIL	VSOW6 _FAIL	VSOW5 _FAIL	VSOW4 _FAIL	VSOW3 _FAIL	VSOW2 _FAIL	VSOW1 _FAIL
FAULT_COMP_AUXOW1	54B	R	0x00	AUXOW1 6 _FAIL	AUXOW1 5 _FAIL	AUXOW1 4 _FAIL	AUXOW1 3 _FAIL	AUXOW1 2 _FAIL	AUXOW1 1 _FAIL	AUXOW1 0 _FAIL	AUXOW9 _FAIL
FAULT_COMP_AUXOW2	54C	R	0x00	AUXOW8 _FAIL	AUXOW7 _FAIL	AUXOW6 _FAIL	AUXOW5 _FAIL	AUXOW4 _FAIL	AUXOW3 _FAIL	AUXOW2 _FAIL	AUXOW1 _FAIL
REG_INT_RSVD	54E	R	0x00	RSVD							
REG_INT_RSVD	54F	R	0x00	RSVD							

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
FAULT_COMP_MISC	550	R	0x00	RSVD						COMP_ADC_ABORT	LPF_FAIL
FAULT_PWR1	552	R	0x00	CVSS_OPEN	DVSS_OPEN	REFHM_OPEN	CVDD_UV	CVDD_OV	DVDD_OV	AVDD_OSC	AVDD_OV
FAULT_PWR2	553	R	0x00	RSVD	PWRBIST_FAIL	RSVD	REFH_OSC	NEG5V_UV	TSREF_OSC	TSREF_UV	TSREF_OV
FAULT_PWR3	554	R	RSVD	RSVD					RSVD	RSVD	AVDDUV_DRST
REG_INT_RSVD	556	R	0x00	RSVD							
REG_INT_RSVD	557	R	0x00	RSVD							
REG_INT_RSVD	558	R	0x00	RSVD							
VS16_HI/LO	568	R	0x80	RESULT[7:0]							
	569	R	0x00	RESULT[7:0]							
VS15_HI/LO	56A	R	0x80	RESULT[7:0]							
	56B	R	0x00	RESULT[7:0]							
VS14_HI/LO	56C	R	0x80	RESULT[7:0]							
	56D	R	0x00	RESULT[7:0]							
VS13_HI/LO	56E	R	0x80	RESULT[7:0]							
	56F	R	0x00	RESULT[7:0]							
VS12_HI/LO	570	R	0x80	RESULT[7:0]							
	571	R	0x00	RESULT[7:0]							
VS11_HI/LO	572	R	0x80	RESULT[7:0]							
	573	R	0x00	RESULT[7:0]							
VS10_HI/LO	574	R	0x80	RESULT[7:0]							
	575	R	0x00	RESULT[7:0]							
VS9_HI/LO	576	R	0x80	RESULT[7:0]							
	577	R	0x00	RESULT[7:0]							
VS8_HI/LO	578	R	0x80	RESULT[7:0]							
	579	R	0x00	RESULT[7:0]							
VS7_HI/LO	57A	R	0x80	RESULT[7:0]							
	57B	R	0x00	RESULT[7:0]							
VS6_HI/LO	57C	R	0x80	RESULT[7:0]							
	57D	R	0x00	RESULT[7:0]							
VS5_HI/LO	57E	R	0x80	RESULT[7:0]							
	57F	R	0x00	RESULT[7:0]							
VS4_HI/LO	580	R	0x80	RESULT[7:0]							
	581	R	0x00	RESULT[7:0]							
VS3_HI/LO	582	R	0x80	RESULT[7:0]							
	583	R	0x00	RESULT[7:0]							
VS2_HI/LO	584	R	0x80	RESULT[7:0]							
	585	R	0x00	RESULT[7:0]							
VS1_HI/LO	586	R	0x80	RESULT[7:0]							
	587	R	0x00	RESULT[7:0]							
CSAUX_HI/LO	588	R	0x80	RESULT[7:0]							
	589	R	0x00	RESULT[7:0]							

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TSREF_HI/LO	58C	R	0x80	RESULT[7:0]							
	58D	R	0x00	RESULT[7:0]							
GPIO1_HI/LO	58E	R	0x80	RESULT[7:0]							
	58F	R	0x00	RESULT[7:0]							
GPIO2_HI/LO	590	R	0x80	RESULT[7:0]							
	591	R	0x00	RESULT[7:0]							
GPIO3_HI/LO	592	R	0x80	RESULT[7:0]							
	593	R	0x00	RESULT[7:0]							
GPIO4_HI/LO	594	R	0x80	RESULT[7:0]							
	595	R	0x00	RESULT[7:0]							
GPIO5_HI/LO	596	R	0x80	RESULT[7:0]							
	597	R	0x00	RESULT[7:0]							
GPIO6_HI/LO	598	R	0x80	RESULT[7:0]							
	599	R	0x00	RESULT[7:0]							
GPIO7_HI/LO	59A	R	0x80	RESULT[7:0]							
	59B	R	0x00	RESULT[7:0]							
GPIO8_HI/LO	59C	R	0x80	RESULT[7:0]							
	59D	R	0x00	RESULT[7:0]							
DIETEMP1_HI/LO	5AE	R	0x80	RESULT[7:0]							
	5AF	R	0x00	RESULT[7:0]							
DIETEMP2_HI/LO	5B0	R	0x80	RESULT[7:0]							
	5B1	R	0x00	RESULT[7:0]							
AUX_IN_HI/LO	5B2	R	0x80	RESULT[7:0]							
	5B3	R	0x00	RESULT[7:0]							
AUX_GPIO_HI/LO	5B4	R	0x80	RESULT[7:0]							
	5B5	R	0x00	RESULT[7:0]							
AUX_PWR_HI/LO	5B6	R	0x80	RESULT[7:0]							
	5B7	R	0x00	RESULT[7:0]							
AUX_REFL_HI/LO	5B8	R	0x80	RESULT[7:0]							
	5B9	R	0x00	RESULT[7:0]							
AUX_VBG2_HI/LO	5BA	R	0x80	RESULT[7:0]							
	5BB	R	0x00	RESULT[7:0]							
AUX_AVAO_REF_HI/LO	5BE	R	0x80	RESULT[7:0]							
	5BF	R	0x00	RESULT[7:0]							
AUX_AVDD_REF_HI/LO	5C0	R	0x80	RESULT[7:0]							
	5C1	R	0x00	RESULT[7:0]							
AUX_OV_DAC_HI/LO	5C2	R	0x80	RESULT[7:0]							
	5C3	R	0x00	RESULT[7:0]							
AUX_UV_DAC_HI/LO	5C4	R	0x80	RESULT[7:0]							
	5C5	R	0x00	RESULT[7:0]							
RSVD	5C6	R	0x80	RSVD							
	5C7	R	0x00	RSVD							
RSVD	5C8	R	0x80	RSVD							
	5C9	R	0x00	RSVD							

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RSVD	5CA	R	0x80	RSVD							
	5CB	R	0x00	RSVD							
AUX_VCM1_HI/LO	5CC	R	0x80	RESULT[7:0]							
	5CD	R	0x00	RESULT[7:0]							
REFH_HI/LO	5D0	R	0x00	RESULT[7:0]							
	5D1	R	0x00	RESULT[7:0]							
DIAG_MAIN_HI/LO	5D2	R	0x00	RESULT[7:0]							
	5D3	R	0x00	RESULT[7:0]							
DIAG_AUX_HI/LO	5D4	R	0x00	RESULT[7:0]							
	5D5	R	0x00	RESULT[7:0]							
CURRENT_HI/MID/LO	5D6	R	0x80	RESULT[7:0]							
	5D7	R	0x00	RESULT[7:0]							
	5D8	R	0x00	RESULT[7:0]							
DEBUG_COMM_STAT	780	R	0x33 for base 0x3F for stack	RSVD		HW_UART_DRV	HW_DAISY_DRV	COML_TX_ON	COML_RX_ON	COMH_TX_ON	COMH_RX_ON
DEBUG_UART_RC	781	R	0x00	RSVD		RC_IERR	RC_TXDIS	RC_SOF	RC_BYTE_ERR	RC_UNEXP	RC_CRC
DEBUG_UART_RR_TR	782	R	0x00	RSVD			TR_SOF	TR_WAIT	RR_SOF	RR_BYTE_ERR	RR_CRC
DEBUG_COMH_BIT	783	R	0x00	RSVD			PERR	BERR_TAG	SYNC2	SYNC1	BIT
DEBUG_COMH_RC	784	R	0x00	RSVD		RC_IERR	RC_TXDIS	RC_SOF	RC_BYTE_ERR	RC_UNEXP	RC_CRC
DEBUG_COMH_RR_TR	785	R	0x00	RSVD		TR_WAIT	RR_TXDIS	RR_SOF	RR_BYTE_ERR	RR_UNEXP	RR_CRC
DEBUG_COML_BIT	786	R	0x00	RSVD			PERR	BERR_TAG	SYNC2	SYNC1	BIT
DEBUG_COML_RC	787	R	0x00	RSVD		RC_IERR	RC_TXDIS	RC_SOF	RC_BYTE_ERR	RC_UNEXP	RC_CRC
DEBUG_COML_RR_TR	788	R	0x00	RSVD		TR_WAIT	RR_TXDIS	RR_SOF	RR_BYTE_ERR	RR_UNEXP	RR_CRC
DEBUG_UART_DISCARD	789	R	0x00	COUNT[7:0]							
DEBUG_COMH_DISCARD	78A	R	0x00	COUNT[7:0]							
DEBUG_COML_DISCARD	78B	R	0x00	COUNT[7:0]							
DEBUG_UART_VALID_HI/LO	78C	R	0x00	COUNT[7:0]							
	78D	R	0x00	COUNT[7:0]							
DEBUG_COMH_VALID_HI/LO	78E	R	0x00	COUNT[7:0]							
	78F	R	0x00	COUNT[7:0]							

Register Name	Addr Hex	RW Type	Reset Value	Data							
				Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DEBUG_COML_VALID_HI/LO	790	R	0x00	COUNT[7:0]							
	791	R	0x00	COUNT[7:0]							
DEBUG_OTP_SEC_BLK	7A0	R	0x00	BLOCK[7:0]							
DEBUG_OTP_DED_BLK	7A1	R	0x00	BLOCK[7:0]							

8.5.4 Register Field Descriptions

8.5.4.1 Device Addressing Setup

8.5.4.1.1 DIR0_ADDR_OTP

Address	0x0000							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	SPARE[1:0]		ADDRESS[5:0]					
Reset	0	0	0	0	0	0	0	0
SPARE[1:0] = Spare								
ADDRESS[5:0] = This register shows the default device address used when *[DIR_SEL]* = 0 and programmed in the OTP. Writing to this register will not change the device address actively in use. This register is used for the system to program the device address to OTP, which will be loaded to the *DIR0_ADDR* register at POR. For programming, follow the OTP programming procedure.								

8.5.4.1.2 DIR1_ADDR_OTP

Address	0x0001							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	SPARE[1:0]		ADDRESS[5:0]					
Reset	0	0	0	0	0	0	0	0
SPARE[1:0] = Spare								
ADDRESS[5:0] = This register shows the default device address used when *[DIR_SEL]* = 1 and programmed in the OTP. Writing to this register will not change the device address actively in use. This register is used for the system to program the device address to OTP, which will be loaded to the *DIR1_ADDR* register at POR. For programming, follow the OTP programming procedure.								

8.5.4.1.3 CUST_MISC1 through CUST_MISC8

Address	0x0021 to 0x0028							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	DATA[7:0]							
Reset	0	0	0	0	0	0	0	0
DATA[7:0] = Customer scratch pad								

8.5.4.1.4 DIR0_ADDR

Address	0x0306							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD		ADDRESS[5:0]					
Reset	0	0	0	0	0	0	0	0
RSVD = Reserved								
ADDRESS[5:0] = Always shows the current device address used by the device when *[DIR_SEL]* = 0. At POR, this register is loaded from the device address value in the OTP (same OTP device address loaded to *DIR0_ADDR_OTP* register). Host can re-address the device by writing a different device address to this register, and the device will take on the new address immediately. Note: *CONTROL1[ADDR_WR]* = 1 is required to write to this register. See Section 8.5.4.3.7 for details.								

8.5.4.1.5 DIR1_ADDR

Address	0x0307							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD		ADDRESS[5:0]					
Reset	0	0	0	0	0	0	0	0
RSVD = Reserved								

ADDRESS[5:0] =	Always shows the current device address used by the device when *[DIR_SEL]* = 1. At POR, this register is loaded from the device address value in the OTP (same OTP device address loaded to *DIR1_ADDR_OTP* register). Host can re-address the device by writing a different device address to this register, and the device will take on the new address immediately. Note: *CONTROL1[ADDR_WR]* = 1 is required to write to this register. See Section 8.5.4.3.7 for details.

8.5.4.2 Device ID and Scratch Pad

8.5.4.2.1 PARTID

Address	**0x0500**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	PARTID[7:0]							
Reset	0	0	0	0	0	0	0	0
	REV[7:0] = Device Revision 0x00 = Revision A0 0x20 = Revision B0 All other codes = Reserved							

8.5.4.2.2 DEV_REVID

Address	0xE00							
Read Only	**Bit7**	**Bit6**	**Bit5**	**Bit4**	**Bit3**	**Bit2**	**Bit1**	**Bit0**
Reset	0	0	0	0	0	0	0	0
	A value of 0x00 indicates that the device is in normal operating mode. If a fault activates the Factory Testmode Detection, the value will be non-zero. Refer to the Safety Manual for details on SM426: Fact Testmode Detection.							

8.5.4.2.3 DIE_ID1 through DIE_ID9

Address	**0x0501**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	ID[7:0]							
Reset	0	0	0	0	0	0	0	0
	ID[7:0] = Device Revision 0x10 = Revision A0 0x11 = Revision A1 0x20 = Revision B0 0x21 = Revision B1 0x22 = Revision B2 All other codes = Reserved							

Address	**0x0502 to 0x0509**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	ID[7:0]							
Reset	0	0	0	0	0	0	0	0
	ID[7:0] = Die ID for TI factory use							

8.5.4.3 General Configuration and Control

8.5.4.3.1 DEV_CONF

Address	**0x0002**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**

Name	RSVD	RSVD	MULTIDROP_EN	FCOMM_EN	TWO_STOP_EN	NFAULT_EN	FTONE_EN	HB_EN
Reset	0	1	0	1	0	1	0	0

RSVD = RESERVED

MULTIDROP_EN = Defines if the device is used in a multidrop or daisy-chain configuration. The TX and RX for COML and COMH will be enabled or disabled based on the configuration.
0 = Daisy-chain of base device
1 = Multidrop

FCOMM_EN = Enables the fault state detection through communication in ACTIVE mode.
0 = Disable
1 = Enable

TWO_STOP_EN = Enables two stop bits for the UART in case of severe oscillator error in the host and device.
0 = One STOP bit
1 = Two STOP bits

NFAULT_EN = Enables the NFAULT function.
0 = NFAULT always pulled up
1 = NFAULT pulled low to indicate an unmasked fault is detected.
Note: This bit setting does not affect the *FAULT_SUMMARY* register.

FTONE_EN = Enables FAULT TONE transmitter when device is in SLEEP mode.
0 = Disable
1 = Enable

HB_EN = Enables HEARTBEAT transmitter when device is in SLEEP mode.
0 = Disable
1 = Enable

8.5.4.3.2 PWR_TRANSIT_CONF

Address	**0x0018**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	SPARE[2:0]			TWARN_THR[1:0]		SLP_TIME[2:0]		
Reset	0	0	0	1	1	0	0	0
Factory Configuration default	0	0	0	1	0	0	0	0

SPARE[2:0] = Spare

TWARN_THR[1:0] = Sets the TWARN threshold.
00 = 85°C
01 = 95°C
10 = 105°C (default)
11 = 115°C

SLP_TIME[2:0] = A timeout in SLEEP mode. This timer starts counting when device enters SLEEP mode. When the timer expires, the device enters SHUTDOWN mode. The timer resets if device wakes up to ACTIVE mode.
000 = No timeout. Device remains in SLEEP mode (default at reset)
001 = 5 s
010 = 10 s
011 = 1 min
100 = 10 min
101 = 30 min
110 = 1 hour
111 = 2 hour

8.5.4.3.3 COMM_TIMEOUT_CONF

Address	**0x0019**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	SPARE	CTS_TIME[2:0]			CTL_ACT	CTL_TIME[2:0]		
Reset	0	0	0	0	0	0	0	0

SPARE = Spare

CTS_TIME[2:0] =	Sets the short communication timeout. When this timer expires, the device sets the *FAULT_SYS[CTL]* bit. This can be used as an alert to the system to prevent a long communication timeout. 000 = Disables short communication timeout (default at reset) 001 = 100 ms 010 = 2 s 011 = 10 s 100 = 1 min 101 = 10 min 110 = 30 min 111 = 1 hr
CTL_ACT =	Configures the device action when long communication timeout timer expires. 0 = Sets *FAULT_SYS[CTL]* and sends device to SLEEP mode (default at reset) 1 = Sends the device to SHUTDOWN. *FAULT_SYS[CTL]* bit will not be set.
CTL_TIME[2:0] =	Sets the long communication timeout. When this timer expires, the device takes the action configured by the *[CTL_ACT]* bit. 000 = Disables long communication timeout (default at reset) 001 = 100 ms 010 = 2 s 011 = 10 s 100 = 1 min 101 = 10 min 110 = 30 min 111 = 1 hr

8.5.4.3.4 TX_HOLD_OFF

Address	**0x001A**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	DLY[7:0]							
Reset	0	0	0	0	0	0	0	0
DLY[7:0] =	Sets the number of bit periods from 0 to 255 to delay after receiving the STOP bit of a command frame and before transmitting the 1st bit of response frame.							

8.5.4.3.5 STACK_RESPONSE

Address	**0x0029**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	SPARE[1:0]		DELAY[5:0]					
Reset	0	0	0	0	0	0	0	0
DELAY[5:0] =	Add additional byte delay gap in daisy-chain data response frame 0x00 = 0-µs 0x01 to 0x3F = 0.25-µs to 15.75-µs in 0.25-µs step							

8.5.4.3.6 COMM_CTRL

Address	**0x0308**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD						STACK_DEV	TOP_STACK
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
STACK_DEV =	Defines device as a base or stack device in daisy-chain configuration. 0 = Base device 1 = Stack device							
TOP_STACK =	Defines device as highest addressed device in the stack. 0 = Not the ToS device 1 = Is the ToS device							

8.5.4.3.7 CONTROL1

Address	0x0309							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	DIR_SEL	SEND_ SHUTDOWN	SEND_WAKE	SEND_ SLPTOACT	GOTO_ SHUTDOWN	GOTO_ SLEEP	SOFT_RESET	ADDR_WR
Reset	0	0	0	0	0	0	0	0

DIR_SEL = Selects daisy-chain communication direction.
0 = With two devices connected in daisy-chain, command frame travels from COMH of the lower device to COML of the next device.
1 = With two devices connected in daisy-chain, command frame travels from COML of the lower device to COMH of the next device.

SEND_SHUTDOWN = Sends SHUTDOWN tone to next device up the stack. The device receiving this bit set is unaffected. Bit is cleared on read.
0 = Ready
1 = Send SHUTDOWN tone up the stack

SEND_WAKE = Sends WAKE tone to next device up the stack. Bit is cleared on read.
0 = Ready
1 = Send WAKE tone to next device up the stack.

SEND_SLPTOACT = Sends SLEEPtoACTIVE tone up the stack. Bit is cleared on read.
0 = Ready
1 = Send SLEEPtoACTIVE tone up the stack

GOTO_SHUTDOWN = Transitions device to SHUTDOWN mode. Bit is cleared on read.
0 = Ready
1 = Enter SHUTDOWN mode

GOTO_SLEEP = Transitions device to SLEEP mode. Bit is cleared on read.
0 = Ready
1 = Enter SLEEP mode

SOFT_RESET = Resets the digital to OTP default. Bit is cleared on read. Setting this bit will cause the device to generate WAKE tone to the upper stack devices.
0 = Ready
1 = Reset device

ADDR_WR = Enables device to start auto-addressing. When this bit is set, device will not forward the first transition it receives, allowing the device address to be written to a single device. See Section 8.3.5.1.3.2 for details.
0 = Not performing auto-address. Device forwards communication transaction as normal.
1 = Device is being auto-addressed; the first communication transaction it receives will not be forwarded.

8.5.4.3.8 CONTROL2

Address	0x030A							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD						SEND_HW_ RESET	TSREF_EN
Reset	0	0	0	0	0	0	0	0

RSVD = Reserved

SEND_HW_RESET = Sends HW_RESET tone up the stack. Bit is cleared on read.
0 = Ready
1 = Send HW_RESET tone to next stack device up

TSREF_EN = Enables TSREF LDO output. Used to bias NTC thermistor.
0 = Disabled
1 = Enabled

8.5.4.3.9 CUST_CRC_HI

Address	0x0036							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CRC[7:0]							
Reset	0	1	0	1	0	1	1	1

Factory Configuration Reset	0	0	1	1	0	0	0	1
CRC[7:0] = High-byte of the host-calculated CRC for customer OTP space.								

8.5.4.3.10 CUST_CRC_LO

Address	0x0037							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CRC[7:0]							
Reset	1	0	0	0	1	0	0	1
Factory Configuration Reset	1	1	1	1	0	0	1	1
CRC[7:0] = Low-byte of the host-calculated CRC for customer OTP space.								

8.5.4.3.11 CUST_CRC_RSLT_HI

Address	0x050C							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CRC[7:0]							
Reset	0	0	1	1	0	0	0	1
CRC[7:0] = High-byte of the device-calculated CRC for customer OTP space.								

8.5.4.3.12 CUST_CRC_RSLT_LO

Address	0x050D							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CRC[7:0]							
Reset	1	1	1	1	0	0	1	1
CRC[7:0] = Low-byte of the device-calculated CRC for customer OTP space.								

8.5.4.4 Operation Status

8.5.4.4.1 DIAG_STAT

Address	0x0526							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			RSVD	DRDY_OVUV	RSVD	DRDY_BIST_OVUV	DRDY_BIST_PWR
Reset	0	0	0	0	0	0	0	0
RSVD = Reserved								
DRDY_OVUV = Indicates the OVUV round robin has at least run once. This bit is cleared when *[OVUV_GO]* = 1 with *[OVUV_MODE1:0]* = 01 (start the OVUV round robin run) and set when at least 1 cycle of round robin is completed. 0 = OVUV has not started or first round robin has not completed yet. 1 = At least 1 cycle of round robin has completed.								
DRDY_BIST_OVUV = Indicates the status of the OVUV protector diagnostic. This bit is cleared when *[OVUV_GO]* = 1 with *[OVUV_MODE1:0]* = 10 (start the BIST run) and set when the BIST cycle is completed. 0 = Not started or still running. 1 = BIST cycle completed.								

DRDY_BIST_PWR =	Indicates the status of the power supplies diagnostic. This bit is cleared when *[PWR_BIST_GO]* = 1 (start the BIST run) and set when the BIST cycle is completed. 0 = Not started or still running. 1 = BIST cycle completed.

8.5.4.4.2 ADC_STAT1

Address	**0x0527**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			RSVD	DRDY_AUX_GPIO	DRDY_AUX_IN	DRDY_AUX_MISC	DRDY_MAIN_ADC
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
DRDY_AUX_GPIO =	AUX ADC has completed at least a single measurement on all active GPIO channels configured for ADC measurement. This bit is cleared when *[AUX_GO]* is changed from 0 to 1. 0 = Not ready 1 = All GPIO inputs have completed at least a single measurement by the AUX ADC
DRDY_AUX_IN =	Device has completed at least a single measurement on all AUX channel(s) set by *[AUX_IN_SEL4:0]*. This bit is cleared when *[AUX_GO]* is changed from 0 to 1. 0 = Not ready 1 = All *[AUX_IN_SEL4:0]* configured channels have completed at least a single measurement
DRDY_AUX_MISC =	Device has completed at least a single measurement on all AUX ADC MISC input channels (that is, completed a single round robin run). This bit is cleared when *[AUX_GO]* is changed from 0 to 1. 0 = Not ready 1 = All AUX ADC MISC inputs have completed at least a single measurement
DRDY_MAIN_ADC =	Device has completed at least a single measurement on all Main ADC input channels, including all GPIOs (that is, completed a single round robin run). This bit is cleared when *[CS_MAIN_GO]* is changed from 0 to 1. 0 = Not ready 1 = All Main ADC inputs have completed at least a single measurement

8.5.4.4.3 ADC_STAT2

Address	**0x0528**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD		DRDY_LPF	DRDY_GPIO	DRDY_VSOW	DRDY_AUXOW	RSVD	DRDY_VSAUX
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
DRDY_LPF =	Device has finished at least 1 round of LPF checks on all VS channels. The comparison continues in the background as long as the Main ADC is running. This bit is cleared when *[COMP_ADC_GO]* = 1. This data ready bit is also used when a fault is injected to test the DIAG_LPF engine using the *[LPF_FLT_INJ]* bit. When *[LPF_FLT_INJ]* = 1, this bit is cleared to 0 and the device will restart the VS and CSAUX channel LPF checks from the beginning using the fault inject *[DIAG_LPF]*. Once all channel LPFs are checked, the *[DRDY_LPF]* = 1. 0 = Not ready 1 = Diagnostic comparison finished
DRDY_GPIO =	Device has finished the GPIO Main and AUX ADC diagnostic comparisons on all active channels and the comparisons are stopped. This bit is cleared when *[COMP_ADC_GO]* = 1. 0 = Not ready 1 = Diagnostic comparison finished
DRDY_VSOW =	Device has finished VS OW diagnostic comparison on all active channels and the comparison is stopped. This bit is cleared when *[COMP_ADC_GO]* = 1. 0 = Not ready 1 = Diagnostic comparison finished
DRDY_AUXOW =	Device has finished AUX OW diagnostic comparison on all active channels and the comparison is stopped. This bit is cleared when *[COMP_ADC_GO]* = 1. 0 = Not ready 1 = Diagnostic comparison finished

DRDY_VSAUX= Device has finished VS vs. AUX diagnostic comparison on all active channels. This bit is cleared when *[COMP_ADC_GO]* = 1.
0 = Not ready
1 = Diagnostic comparison finished

8.5.4.4.4 GPIO_STAT

Address	0x052A							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	GPIO8	GPIO7	GPIO6	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1
Reset	0	0	0	0	0	0	0	0

GPIO1 through GPIO8 = When GPIO is configured as digital input or output, this register shows the GPIO status.
0 = Low
1 = High

8.5.4.4.5 DEV_STAT

Address	0x052C							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD	FACT_CRC_DONE	CUST_CRC_DONE	RSVD	OVUV_RUN	CS_RUN	AUX_RUN	MAIN_RUN
Reset	0	0	0	0	0	0	0	0

RSVD = Reserved

FACT_CRC_DONE = Indicates the status of the factory CRC state machine. This bit is set when the factory CRC is calculated and verified internally at least once. A read from this register will clear this bit.
0 = Not complete
1 = Complete (cleared on read)

CUST_CRC_DONE = Indicates the status of the customer CRC state machine. This bit is set when the CRC is calculated and compared to the *CUST_CRC** registers at least once. A read from this register will clear this bit.
0 = Not complete
1 = Complete (cleared on read)

OVUV_RUN = Shows the status of the OVUV protector comparators. This bit is set when OVUV BIST starts. When BIST is completed or aborted, the device will turn off the OV and UV comparators automatically, and then this bit will be cleared).
0 = off (that is, OVUV is not started or when *[OVUV_GO]* = 1 and *[OVUV_MODE1:0]* = 0)
1 = on (that is, when *[OVUV_GO]* = 1 and *[OVUV_MODE1:0]* is non-zero)

CS_RUN = Shows the status of the CS ADC.
0 = off
1 = on

AUX_RUN = Shows the status of the AUX ADC.
0 = off
1 = on

MAIN_RUN = Shows the status of the Main ADC.
0 = off
1 = on

8.5.4.5 ADC Configuration and Control

8.5.4.5.1 ADC_CONF1

Address	0x0007							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	AUX_SETTLE[1:0]		LPF_CSAUX[2:0]			LPF_VS[2:0]		
Reset	0	0	0	0	0	0	0	0

AUX_SETTLE[1:0] =	The AUX configures the AUX IN settling time. Each AUX has to wait for the anti-aliasing filter (AAF) settling time in order to consider as a valid measurement. These bits provide the option to use different AAF or bypass an AAF to trade for a fast measurement. 00 = 4.3 ms 01 = 2.3 ms 10 = 1.3 ms 11 = Reserved
LPF_CSAUX[2:0] =	Configures the post main SAR ADC low-pass filter cut-off frequency for CSAUX measurement. Same options as the *LPF_VS[2:0]*.
LPF_VS[2:0] =	Configures the post ADC low-pass filter cut-off frequency for VS measurement. 0x0 = 6.5 Hz (154 ms average) 0x1 = 13 Hz (77 ms average) 0x2 = 26 Hz (38 ms average) 0x3 = 53 Hz (19 ms average) 0x4 = 111 Hz (9 ms average) 0x5 = 240 Hz (4 ms average) 0x6 = 600 Hz (1.6 ms average) 0x7 = 240 Hz

8.5.4.5.2 ADC_CONF2

Address	**0x0008**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	SPARE[1:0]		ADC_DLY[5:0]					
Reset	0	0	0	0	0	0	0	0
SPARE[1:0] =	Spare							
ADC_DLY[5:0] =	If *[CS_MAIN_GO]* bit is written to 1, bit Main ADC is delayed for this setting time before being enabled to start the conversion. This setting synchronizes the start of Main ADC throughout the daisy-chained stack. The option ranges from 0 µs (no delay) to 200 µs in 5-µs steps. Undefined code = 0 µs (no delay)							

8.5.4.5.3 MAIN_ADC_CAL1

Address	**0x001B**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	GAINL[7:0]							
Reset	0	0	0	0	0	0	0	0
GAINL[7:0] =	Main ADC 25°C gain calibration result (lower 8-bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step. Range from -0.78125% to 0.7782% in 0.0031% steps.							

8.5.4.5.4 MAIN_ADC_CAL2

Address	**0x001C**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	GAINH	OFFSET[6:0]						
Reset	0	0	0	0	0	0	0	0
GAINH	Main ADC 25°C gain calibration result (MS bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step. Range from -0.78125% to 0.7782% in 0.0031% steps.							
OFFSET[6:0] =	Main ADC 25°C offset calibration result. If customer performs offset calibration during production flow, the offset result can be programmed to OTP and will be sent to this offset register at device reset. The device automatically applies this data during ADC correction step. Range from -12.20703-mV to 12.01630-mV in 0.19073-mV steps							

8.5.4.5.5 AUX_ADC_CAL1

Address	0x001D							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GAINL[7:0]							
Reset	0	0	0	0	0	0	0	0

GAINL[7:0] = AUX ADC 25°C gain calibration result (lower 8-bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step.
Range from -0.78125% to 0.7782% in 0.0031% steps.

8.5.4.5.6 AUX_ADC_CAL2

Address	0x001E							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GAINH	OFFSET[6:0]						
Reset	0	0	0	0	0	0	0	0

GAINH AUX ADC 25°C gain calibration result (MS bit). If customer performs gain calibration during production flow, the gain result can be programmed to OTP and will be sent to this gain register at device reset. The device automatically applies this data during ADC correction step.
Range from -0.78125% to 0.7782% in 0.0031% steps.

OFFSET[6:0] = AUX ADC 25°C offset calibration result. If customer performs offset calibration during production flow, the offset result can be programmed to OTP and will be sent to this offset register at device reset. The device automatically applies this data during ADC correction step.
Range from -12.20703-mV to 12.01630-mV in 0.19073-mV steps

8.5.4.5.7 CS_ADC_CAL1

Address	0x001F							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GAINL[7:0]							
Reset	0	0	0	0	0	0	0	0

GAINl[7:0] = CS ADC gain correction, lower 8-bits
Range from -0.78125% to 0.78049% in 0.0008% steps.

8.5.4.5.8 CS_ADC_CAL2

Address	0x0020							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GAINH[2:0]			OFFSET[4:0]				
Reset	0	0	0	0	0	0	0	0

OFFSET[4:0] = 8-bit register for CS ADC offset correction.
Range from -3.8147-µV to 3.57628-µV in 0.23842-µV steps.

GAINH[2:0] CS ADC gain correction, upper 3-bits
Range from -0.78125% to 0.78049% in 0.0008% steps.

8.5.4.5.9 ADC_CTRL1

Address	0x030D							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD	CS_DR[1:0]		LPF_CSAUX_EN	LPF_VS _EN	CS_MAIN_GO	MAIN_MODE[1:0]	
Reset	0	0	0	0	0	0	0	0

RSVD = Reserved

CS_DR[1:0] =	Configures the desired single measurement time of the CS ADC. 00 = 768 µs 01 = 1.536 ms 10 = 3.072 ms 11 = 12.288 ms
LPF_CSAUX_EN =	Enables digital low-pass filter post-ADC conversion. LPF applies to CSAUX measurements only. The cut-off frequency is configured by *ADC_CONFIG1[LPF_CSAUX[2:0]*.
LPF_VS_EN =	Enables digital low-pass filter post-ADC conversion. LPF applies to VS measurements only. The cut-off frequency is configured by *ADC_CONFIG1[LPF_VS[2:0]*. NOTE: Need an OTP bit as a global enable or disable to the digital LPF function for VS and SRP/N.
CS_MAIN_GO =	Starts main ADC conversion. When this bit is written to 1, all Main ADC inputs are sampled. Once the Main ADC is started, any change to the Main ADC control setting has no effect until this bit is written to 1 again. This bit is cleared to 0 in read. 0 = Ready. Writing 0 has no effect 1 = Start Main ADC This control also applies to the CS ADC.In sleep mode, CS ADC need to be disabled, otherwise, it consumes extra current ICS_ADC.
MAIN_MODE[1:0] =	Sets the Main ADC run mode. In continuous run, if user would like to stop ADC, user must read all the ADC conversion results, then stop it. ADC results are not valid before ADC is re-enabled next time. 00 = Main ADC not running 01 = Single run. Run the main ADC round robin 8 times and then stop 10 = Continuous run. Continuous running the Main ADC round robin until host sends command to stop 11 = Reserved

8.5.4.5.10 ADC_CTRL2

Address	0x030E							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD	MAINAUXCS_AFE_DIS	AUX_IN_ALIGN	AUX_IN_SEL[4:0]				
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
MAINAUXCS_AFE_DIS =	Disconnected main ADC SRP/SRN AFE from SRP/SRN pin, this would leave SRP/SRN sensed by CS ADC stand alone. 0 = Connected 1 = Disconnected
AUX_IN_ALIGN =	Align the AUX ADC AUX measurement to Main ADC VS1 or VS8 0 = Align to Main ADC VS1 1 = Align to Main ADC VS8
AUX_IN_SEL[4:0] =	Selects which AUX channel(s) will be multiplexed through the AUX ADC. 0x00 = Run all active channels 0x01 = SRP/SRN are not connected to AUX ADC 0x02 = Lock to AUX1 0x03 = Lock to AUX2 0x04 = Lock to AUX3 : 0x05 = Lock to AUX4 0x06 to 0x1F = RSVD NOTE: If inactive channel or RSVD code is selected, device will not perform AUX ADC conversion on the AUX slot and the *AUX_IN_HI/LO* registers will be kept in reset value.

8.5.4.5.11 ADC_CTRL3

Address	0x030F							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD	AUX_GPIO_SEL[3:0]				AUX_GO	AUX_MODE[1:0]	
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved

AUX_GPIO_SEL[3:0] =	Selects which GPIO channel(s) will be multiplexed through the AUX ADC to use for temperature measurement diagnostic. If this selection is not set to 0x00, the AUX ADC will lock onto a single GPIO channel and the measurement result is output to the *AUX_GPIO_HI/LO* registers. 0x00 = AUX ADC cycles through all GPIO channel(s) that are configured as ADC. 0x01 = Lock to GPIO1 0x02 = Lock to GPIO2 : 0x08 = Lock to GPIO8 All other codes are RSVD. NOTE: If GPIO is not configured for ADC measurement or RSVD codes are selected, device will not perform AUX ADC conversion on the GPIO slot and the *AUX_GPIO_HI/LO* registers will be kept in reset value.
AUX_GO =	Starts AUX ADC conversion. When this bit is written to 1, all AUX ADC inputs are sampled. Once the AUX ADC is started, any change to the AUX ADC control setting has no effect until this bit is written to 1 again. This bit is cleared to 0 in read. 0 = Ready. Writing 0 has no effect. 1 = Start AUX ADC
AUX_MODE[1:0] =	Sets the Main ADC run mode. In continuous run, if user would like to stop ADC, user must read all the ADC conversion results, then stop it. ADC results are not valid before ADC is re-enabled next time. 00 = AUX ADC not running 01 = Single run. Run the AUX ADC round robin once and then stop. 10 = Continuous run. Continually run the AUX ADC round robin until host sends command to stop. 11 = 8-round-robin run to measure all eight GPIOs once.

8.5.4.6 ADC Measurement Results

8.5.4.6.1 VS16_HI/LO

VS16_HI

Address	0x0568							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS16 voltage in 2s complement. When host reads this register, the device locks the VS16 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS16_LO

Address	0x0569							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS16 voltage in 2s complement.								

8.5.4.6.2 VS15_HI/LO

VS15_HI

Address	0x056A							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS15 voltage in 2s complement. When host reads this register, the device locks the VS15 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS15_LO

Address	0x056B							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS15 voltage in 2s complement.								

8.5.4.6.3 VS14_HI/LO

VS14_HI

Address	0x056C							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS14 voltage in 2s complement. When host reads this register, the device locks the VS14 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS14_LO

Address	0x056D							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS14 voltage in 2s complement.								

8.5.4.6.4 VS13_HI/LO

VS13_HI

Address	0x056E							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS13 voltage in 2s complement. When host reads this register, the device locks the VS13 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS13_LO

Address	0x056F							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS13 voltage in 2s complement.								

8.5.4.6.5 VS12_HI/LO

VS12_HI

Address	0x0570							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS12 voltage in 2s complement. When host reads this register, the device locks the VS12 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS12_LO

Address	0x0571							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS12 voltage in 2s complement.								

8.5.4.6.6 VS11_HI/LO

VS11_HI

Address	0x0572							

Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS11 voltage in 2s complement. When host reads this register, the device locks the VS11 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS11_LO

Address	0x0573							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS11 voltage in 2s complement.								

8.5.4.6.7 VS10_HI/LO

VS10_HI

Address	0x0574							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS10 voltage in 2s complement. When host reads this register, the device locks the VS10 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS10_LO

Address	0x0575							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS10 voltage in 2s complement.								

8.5.4.6.8 VS9_HI/LO

VS9_HI

Address	0x0576							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS9 voltage in 2s complement. When host reads this register, the device locks the VS9 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS9_LO

Address	0x0577							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							

Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS9 voltage in 2s complement.								

8.5.4.6.9 VS8_HI/LO

VS8_HI

Address	0x0578							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS8 voltage in 2s complement. When host reads this register, the device locks the VS8 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS8_LO

Address	0x0579							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS8 voltage in 2s complement.								

8.5.4.6.10 VS7_HI/LO

VS7_HI

Address	0x057A							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the VS7 voltage in 2s complement. When host reads this register, the device locks the VS7 voltage low-byte from updating until the high-byte and low-byte registers are read.								

VS7_LO

Address	0x057B							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS7 voltage in 2s complement.								

8.5.4.6.11 VS6_HI/LO

VS6_HI

Address	0x057C							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0

RESULT[7:0] =	The ADC measurement result of the high-byte of the VS6 voltage in 2s complement. When host reads this register, the device locks the VS6 voltage low-byte from updating until the high-byte and low-byte registers are read.

VS6_LO

Address	0x057D							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the low-byte of the VS6 voltage in 2s complement.							

8.5.4.6.12 VS5_HI/LO

VS5_HI

Address	0x057E							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the high-byte of the VS5 voltage in 2s complement. When host reads this register, the device locks the VS5 voltage low-byte from updating until the high-byte and low-byte registers are read.							

VS5_LO

Address	0x057F							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the low-byte of the VS5 voltage in 2s complement.							

8.5.4.6.13 VS4_HI/LO

VS4_HI

Address	0x0580							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the high-byte of the VS4 voltage in 2s complement. When host reads this register, the device locks the VS4 voltage low-byte from updating until the high-byte and low-byte registers are read.							

VS4_LO

Address	0x0581							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the low-byte of the VS4 voltage in 2s complement.							

8.5.4.6.14 VS3_HI/LO

VS3_HI

Address	0x0582							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the high-byte of the VS3 voltage in 2s complement. When host reads this register, the device locks the VS3 voltage low-byte from updating until the high-byte and low-byte registers are read.							

VS3_LO

Address	0x0583							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the low-byte of the VS3 voltage in 2s complement.							

8.5.4.6.15 VS2_HI/LO

VS2_HI

Address	0x0584							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the high-byte of the VS2 voltage in 2s complement. When host reads this register, the device locks the VS2 voltage low-byte from updating until the high-byte and low-byte registers are read.							

VS2_LO

Address	0x0585							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the low-byte of the VS2 voltage in 2s complement.							

8.5.4.6.16 VS1_HI/LO

VS1_HI

Address	0x0586							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the high-byte of the VS1 voltage in 2s complement. When host reads this register, the device locks the VS1 voltage low-byte from updating until the high-byte and low-byte registers are read.							

VS1_LO

Address	0x0587							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the VS1 Voltage in 2s complement.								

8.5.4.6.17 CSAUX_HI/LO

CSAUX_HI

Address	0x0588							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the differential CSAUX (SRP – SRN) in 2s complement. When host reads this register, the device locks the low-byte from updating until the high-byte and low-byte registers are read.								

CSAUX_LO

Address	0x0589							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the low-byte of the differential CSAUX (SRP – SRN) in 2s complement.								

8.5.4.6.18 TSREF_HI/LO

TSREF_HI

Address	0x058C							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The TSREF high-byte result from Main ADC. When host reads this register, the device locks the TSREF low-byte from updating until the high-byte and low-byte registers are read.								

TSREF_LO

Address	0x058D							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The TSREF low-byte result from Main ADC								

8.5.4.6.19 GPIO1_HI/LO

GPIO1_HI

Address	0x058E							

Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO1. When host reads this register, the device locks the GPIO1 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO1_LO

Address	0x058F							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO1.								

8.5.4.6.20 GPIO2_HI/LO

GPIO2_HI

Address	0x0590							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO2. When host reads this register, the device locks the GPIO2 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO2_LO

Address	0x0591							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO2.								

8.5.4.6.21 GPIO3_HI/LO

GPIO3_HI

Address	0x0592							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO3. When host reads this register, the device locks the GPIO3 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO3_LO

Address	0x0593							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							

Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO3.								

8.5.4.6.22 GPIO4_HI/LO

GPIO4_HI

Address	0x0594							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO4. When host reads this register, the device locks the GPIO4 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO4_LO

Address	0x0595							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO4.								

8.5.4.6.23 GPIO5_HI/LO

GPIO5_HI

Address	0x0596							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO5. When host reads this register, the device locks the GPIO5 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO5_LO

Address	0x0597							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO5.								

8.5.4.6.24 GPIO6_HI/LO

GPIO6_HI

Address	0x0598							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO6. When host reads this register, the device locks the GPIO6 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO6_LO

Address	0x0599							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO6.								

8.5.4.6.25 GPIO7_HI/LO

GPIO7_HI

Address	0x059A							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO7. When host reads this register, the device locks the GPIO7 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO7_LO

Address	0x059B							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO7.								

8.5.4.6.26 GPIO8_HI/LO

GPIO8_HI

Address	0x059C							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement high-byte result of the GPIO8. When host reads this register, the device locks the GPIO8 low-byte from updating until the high-byte and low-byte registers are read.								

GPIO8_LO

Address	0x059D							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement low-byte result of the GPIO8.								

8.5.4.6.27 DIETEMP1_HI/LO

DIETEMP1_HI

Address	0x05AE							

Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The DieTemp1 high-byte result from Main ADC. When host reads this register, the device locks the DIETEMP1 low-byte from updating until the high-byte and low-byte registers are read.								

DIETEMP1_LO

Address	0x05AF							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The DieTemp1 low-byte (temperature used for ADC correction) result from Main ADC.								

8.5.4.6.28 DIETEMP2_HI/LO

DIETEMP2_HI

Address	0x05B0							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The DieTemp2 high-byte result from AUX ADC. When host reads this register, the device locks the DIETEMP2 low-byte from updating until the high-byte and low-byte registers are read.								

DIETEMP2_LO

Address	0x05B1							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The DieTemp2 low-byte (temperature used for ADC correction) result from AUX ADC								

8.5.4.6.29 AUX_IN_HI/LO

AUX_IN_HI

Address	0x05B2							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The ADC measurement result of the high-byte of the AUX voltage in 2s complement. These *AUX_IN_HI/LO* registers will only report AUX voltage measurement if host configures *[AUX_IN_SEL4:0]* to lock to a single AUX channel. When host reads this register, the device locks the AUX voltage low-byte from updating until the high-byte and low-byte registers are read.								

AUX_IN_LO

Address	0x05B3							

Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The ADC measurement result of the low-byte of the AUX IN voltage in 2s complement. These *AUX_IN_HI/LO* registers will only report AUX voltage measurement if host configures *[AUX_IN_SEL4:0]* to lock to a single AUX channel.							

8.5.4.6.30 AUX_GPIO_HI/LO

AUX_GPIO_HI

Address	0x05B4							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The AUX ADC measurement high-byte result of the GPIO that is locked by the *[AUXGPIO_SEL3:0]* bits. When host reads this register, the device locks the AUX_GPIO low-byte from updating until the high-byte and low-byte registers are read.							

AUX_GPIO_LO

Address	0x05B5							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The AUX ADC measurement low-byte result of the GPIO that is locked by the *[AUXGPIO_SEL3:0]* bits.							

8.5.4.6.31 AUX_PWR_HI/LO

AUX_PWR_HI

Address	0x05B6							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The high-byte result of the PWR pin measurement from AUX ADC. When host reads this register, the device locks the AUX_PWR low-byte from updating until the high-byte and low-byte registers are read.							

AUX_PWR_LO

Address	0x05B7							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The low-byte result of the PWR pin measurement from AUX ADC.							

8.5.4.6.32 AUX_REFL_HI/LO

AUX_REFL_HI

Address	0x05B8							

Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of the internal reference, REFL, measurement from AUX ADC. When host reads this register, the device locks the AUX_REL low-byte from updating until the high-byte and low-byte registers are read.								

AUX_REFL_LO

Address	0x05B9							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The low-byte result of the internal reference, REFL, measurement from AUX ADC.								

8.5.4.6.33 AUX_VBG2_HI/LO

AUX_VBG2_HI

Address	0x05BA							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of the internal reference, VBG2, measurement from AUX ADC. When host reads this register, the device locks the AUX_VBG2 low-byte from updating until the high-byte and low-byte registers are read.								

AUX_VBG2_LO

Address	0x05BB							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The low-byte result of the internal reference, VBG2, measurement from AUX ADC.								

8.5.4.6.34 AUX_AVAO_REF_HI/LO

AUX_AVAO_REF_HI

Address	0x05BE							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of the AVAO_REF measurement from AUX ADC. When host reads this register, the device locks the AUX_AVAO_REF low-byte from updating until the high-byte and low-byte registers are read.								

AUX_AVAO_REF_LO

Address	0x05BF							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The low-byte result of the AVAO_REF measurement from AUX ADC.								

8.5.4.6.35 AUX_AVDD_REF_HI/LO

AUX_AVDD_REF_HI

Address	0x05C0							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of the AVDD_REF measurement from AUX ADC. When host reads this register, the device locks the AUX_AVDD_REF low-byte from updating until the high-byte and low-byte registers are read.								

AUX_AVDD_REF_LO

Address	0x05C1							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The low-byte result of the AVDD_REF measurement from AUX ADC.								

8.5.4.6.36 AUX_OV_DAC_HI/LO

AUX_OV_DAC_HI

Address	0x05C2							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of the OV comparator DAC measurement, which is (0.8 x OV threshold), from AUX ADC. When host reads this register, the device locks the AUX_OV_DAC low-byte from updating until the high-byte and low-byte registers are read.								

AUX_OV_DAC_LO

Address	0x05C3							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The low-byte result of the OV comparator DAC measurement, which is (0.8 x OV threshold), from AUX ADC.								

8.5.4.6.37 AUX_UV_DAC_HI/LO

AUX_UV_DAC_HI

Address	0x05C4							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0

RESULT[7:0] =	The high-byte result of the UV comparator DAC measurement, which is (0.8 x UV threshold), from AUX ADC. When host reads this register, the device locks the AUX_UV_DAC low-byte from updating until the high-byte and low-byte registers are read.

AUX_UV_DAC_LO

Address	0x05C5							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The low-byte result of the UV comparator DAC measurement, which is (0.8 x UV threshold), from AUX ADC.							

8.5.4.6.38 AUX_VCM1_HI/LO

AUX_VCM1_HI

Address	0x05CC	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The high-byte result of the VCM1 (common mode voltage on Main ADC) measurement from AUX ADC. When host reads this register, the device locks the AUX_VCM1 low-byte from updating until the high-byte and low-byte registers are read.							

AUX_VCM1_LO

Address	0x05CD							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The low-byte result of the VCM1 (common mode voltage on Main ADC) measurement from AUX ADC.							

8.5.4.6.39 REFH_HI/LO

REFH_HI

Address	0x05D0							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] =	The high-byte result of the recorded REFH reference voltage trimmed at factory.							

REFH_LO

Address	0x05D1							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] =	The low-byte result of the recorded REFH reference voltage trimmed at factory.							

8.5.4.6.40 DIAG_MAIN_HI/LO

DIAG_MAIN_HI

Address	0x05D2							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of reported Main ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked								

DIAG_MAIN_LO

Address	**0x05D3**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The low-byte result of reported Main ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked								

8.5.4.6.41 DIAG_AUX_HI/LO

DIAG_AUX_HI

Address	**0x05D4**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	1	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of reported AUX ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked								

DIAG_AUX_LO

Address	**0x05D5**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The low-byte result of reported AUX ADC comparison value used in the diagnostic ADC comparison. Valid if the diagnostic ADC comparison is run when a single channel is locked								

8.5.4.6.42 CURRENT_HI/MID/LO

CURRENT_HI

Address	**0x05D6**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The high-byte result of reported Current Sense ADC (CSADC)								

CURRENT_MID

Address	**0x05D7**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The Mid-byte result of reported Current Sense ADC (CSADC)								

CURRENT_LO

Address	0x05D8							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RESULT[7:0]							
Reset	0	0	0	0	0	0	0	0
RESULT[7:0] = The Low-byte result of reported Current Sense ADC (CSADC)								

8.5.4.7 Protector Configuration and Control

8.5.4.7.1 OV_THRESH

Address	0x0009							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	SPARE	SPARE	OV_THR[5:0]					
Reset	0	0	1	1	1	1	1	1
SPARE = Spare								
OV_THR[5:0] = Sets the overvoltage threshold for the OV comparator. Changes on these bits require host to send another *[OVUV_GO]* = 1 command. All settings are at 25-mV steps. 0x02 to 0x0E: range from 2700 mV to 3000 mV 0x12 to 0x1E: range from 3500 mV to 3800 mV 0x22 to 0x2E: range from 4175 mV to 4475 mV All other settings will default to 2700 mV.								

8.5.4.7.2 UV_THRESH

Address	0x000A							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	SPARE	SPARE	UV_THR[5:0]					
Reset	0	0	0	0	0	0	0	0
SPARE = Spare								
UV_THR[5:0] = Sets the undervoltage threshold for the UV comparator. Changes on these bits require host to send another *[OVUV_GO]* = 1 command. All settings are at 50-mV steps. 0x00 to 0x26: range from 1200 mV to 3100 mV All other settings will default to 3100 mV.								

8.5.4.7.3 UV_DISABLE1

Address	0x000C							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	VS16	VS15	VS14	VS13	VS12	VS11	VS10	VS9
Reset	0	0	0	0	0	0	0	0
VS9 to VS16 = Indicate which channels shall be excluded from UV detection 0 = UV monitoring apply to the channel 1 = UV monitoring are excluded from the channel								

8.5.4.7.4 UV_DISABLE2

Address	0x000D							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	VS8	VS7	VS6	VS5	VS4	VS3	VS2	VS1
Reset	0	0	0	0	0	0	0	0
VS8 to VS1 = Indicate which channels shall be excluded from UV detection 0 = UV monitoring apply to the channel 1 = UV monitoring are excluded from the channel								

8.5.4.7.5 OVUV_CTRL

Address	0x032C							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD	OVUV_LOCK[3:0]				OVUV_GO	OVUV_MODE[1:0]	
Reset	0	0	0	0	0	0	0	0

RSVD = RESERVED: LEAVE AT DEFAULT VALUE

OVUV_LOCK[3:0] = Configures a particular single channel as the OV and UV comparators input when *[OVUV_MOD1:0]* = 0b11. Changes on these bits require host to send another *[OVUV_GO]* = 1 command.
0x0 = Lock to VS1
0x1 = Lock to VS2
0x2 = Lock to VS3
:
0xF = Lock to VS16

OVUV_GO = Starts the OV and UV comparators. When written to 1, all OVUV configuration settings are sampled. This bit is self-clearing.
0 = Ready
1 = Start OV and UV comparators

OVUV_MODE[1:0] = Sets the OV and UV comparators operation mode when *[OVUV_GO]* = 1. Changes on these bits require host to send another *[OVUV_GO]* = 1 command.
00 = Do not run OV and UV comparators
01 = Run the OV and UV round robin with all VS
10 = Run the OV and UV BIST cycle.
11 = Lock OV and UV comparators to a single channel configured by *[OVUV_LOCK3:0]*

8.5.4.8 GPIO Configuration

8.5.4.8.1 GPIO_CONF1

Address	0x000E							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	FAULT_IN_EN	SPI_EN	GPIO2[2:0]			GPIO1[2:0]		
Reset	0	0	0	0	0	0	0	0

FAULT_IN_EN = Enables GPIO8 as an active-low input to trigger the NFAULT pin when the input signal is low.
0 = No fault input function. GPIO8 is configured based on *[GPIO8_CONF2:0]* setting.
1 = GPIO8 is set as active-low input to trigger NFAULT pin, *[GPIO8_CONF2:0]* setting is ignored.

SPI_EN = Enables SPI master on GPIO4, GPIO5 and GPIO6, GPIO7.
0 = SPI master disabled.
1 = SPI master enabled. Overwrite the *[GPIO4_CONF2:0]*, *[GPIO5_CONF2:0]*, *[GPIO6_CONF2:0]*, and *[GPIO7_CONF2:0]* settings.

GPIO2[2:0] = Configures GPIO1.
000 = As disabled, high-Z
001 = RSVD
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

GPIO1[2:0] = Configures GPIO1.
000 = As disabled, high-Z
001 = RSVD
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

8.5.4.8.2 GPIO_CONF2

Address	0x000F							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	SPARE	CS_RDY_EN	GPIO4[2:0]			GPIO3[2:0]		
Reset	0	0	0	0	0	0	0	0

SPARE = Spare

CS_RDY_EN = Enables GPIO1 as digital output to toggle low when CS ADC conversion is complete. Reset to high when host reads CURRENT_HI register.
0 = No CS ADC toggle function. GPIO1 is configured based on [GPIO1_CONF2:0] setting.
1 = GPIO1 is used for CS ADC conversion toggle function, [GPIO1_CONF2:0] setting is ignored.

GPIO4[2:0] = Configures GPIO4. If *[SPI_EN]* = 1, these configuration bits are ignored and the pin is used as SS for SPI master. See Section 8.3.5.1.7 for details.
000 = As disabled, high-Z
001 = RSVD
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

GPIO3[2:0] = Configures GPIO3.
000 = As disabled, high-Z
001 = RSVD
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

8.5.4.8.3 GPIO_CONF3

Address	0x0010							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	SPARE[1:0]		GPIO6[2:0]			GPIO5[2:0]		
Reset	0	0	0	0	0	0	0	0

SPARE[1:0] = Spare

GPIO6[2:0] = Configures GPIO6. If *[SPI_EN]* = 1, these configuration bits are ignored and the pin is used as MOSI for SPI master. See Section 8.3.5.1.7 for details.
000 = As disabled, high-Z
001 = RSVDs
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

GPIO5[2:0] = Configures GPIO5. If *[SPI_EN]* = 1, these configuration bits are ignored and the pin is used as MISO for SPI master. See Section 8.3.5.1.7 for details.
000 = As disabled, high-Z
001 = RSVD
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

8.5.4.8.4 GPIO_CONF4

Address	0x0011							
NVM	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	SPARE[1:0]		GPIO8[2:0]			GPIO7[2:0]		
Reset	0	0	0	0	0	0	0	0

SPARE[1:0] = Spare

GPIO8[2:0] = Configures GPIO8. If *[FAULT_IN_EN]* = 1, these configuration bits are ignored and the pin is used as an input such that an active low will trigger NFAULT.
000 = As disabled, high-Z
001 = RSVD
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

GPIO7[2:0] = Configures GPIO7. If *[SPI_EN]* = 1, these configuration bits are ignored and the pin is used as SCLK for SPI master. See Section 8.3.5.1.7 for details.
000 = As disabled, high-Z
001 = RSVD
010 = As ADC only input
011 = As digital input
100 = As output high
101 = As output low
110 = As ADC input and weak pullup enabled
111 = As ADC input and weak pulldown enabled.

8.5.4.9 SPI Master

8.5.4.9.1 SPI_CONF

Address	0x034D							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD	CPOL	CPHA	NUMBIT[4:0]				
Reset	0	0	0	0	0	0	0	0

RSVD = Reserved

CPOL = Sets the SCLK polarity.
0 = Idles low and clocks high
1 = Idles high and clocks low

CPHA = Sets the edge of SCLK where data is sampled on MISO.
0 = Leading clock transition
1 = Trailing clock transition

NUMBIT[4:0] = SPI transaction length. Set the number of SPI bits to read/write.
00000 = 24-bit
00001 = 1-bit
00010 = 2-bit
:
10111 = 23-bit
All others = 23-bit

8.5.4.9.2 SPI_EXE

Address	0x0351							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD						SS_CTRL	SPI_GO
Reset	0	0	0	0	0	0	1	0

RSVD = Reserved

SS_CTRL =	Programs the state of SS. 0 = Output low 1 = Output high
SPI_GO =	Executes the SPI transaction. This bit is self-clearing. 0 = Idle 1 = Execute the SPI

8.5.4.9.3 SPI_TX3, SPI_TX2, and SPI_TX1

Address	0x034E to 0x0350							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	DATA[7:0]							
Reset	0	0	0	0	0	0	0	0
DATA[7:0] =	Data to be used to write to SPI slave device. The bits are programmed by using *SPI_CONF[NUMBIT4:0]* and are clocked out of MOSI starting from the LSB *SPI_TX1* -> LSB *SPI_TX2* -> LSB *SPI_TX3*.							

8.5.4.9.4 SPI_RX3, SPI_RX2, and SPI_RX1

Address	0x0520 to 0x522							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	DATA[7:0]							
Reset	0	0	0	0	0	0	0	0
	R	R	R	R	R	R	R	R
DATA[7:0] =	Data returned from a read during SPI transaction. Updated, starting with LSB *SPI_RX1* -> LSB *SPI_RX2* -> LSB *SPI_RX3*, with the number of bits set by *SPI_CONF[NUMBIT4:0]* clocked in from MISO.							

8.5.4.10 Diagnostic Control

8.5.4.10.1 DIAG_OTP_CTRL

Address	0x0335							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			FLIP_FACT_CRC	MARGIN_MODE[2:0]			MARGIN_GO
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
FLIP_FACT_CRC =	An enable bit to flip the factory CRC value. This is for factory CRC diagnostic. 0 = Normal operation. No modification of the factory CRC 1 = Flip the CRC value. This causes a factory CRC fault, *FAULT_OTP[FACT_CRC]*.							
MARGIN_MODE[2:0] =	Configures OTP Margin read mode: 0b000 = Normal Read 0b001 = Reserved 0b010 = Margin 1 Read 0b011 to 0b111 = Reserved							
MARGIN_GO =	Starts OTP Margin test set by the *[MARGIN_MOD]* bit. This bit self-clears and always reads 0. 0 = Ready 1 = Start the test							

8.5.4.10.2 DIAG_COMM_CTRL

Address	0x0336							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD						SPI_LOOPBACK	FLIP_TR_CRC
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
SPI_LOOPBACK =	Enables SPI loopback function to verify SPI functionality. See Section 8.3.5.1.7 for more details. 0 = Disable 1 = Enable
FLIP_TR_CRC =	Sends a purposely incorrect communication (during transmitting response) CRC by inverting all of the calculated CRC bits. 0 = Send CRC as calculated 1 = Send inverted CRC

8.5.4.10.3 DIAG_PWR_CTRL

Address	0x0337							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD						BIST_NO_RST	PWR_BIST_GO
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
BIST_NO_RST =	Use for further diagnostic if the power supply BIST detects a failure. When this bit is set to 1, the device will either run a force pass or force fail BIST cycle based on the *[BIST_FORCE_FAIL]* setting and will not clear the *FAULT_PWR1* register, and the *FAULT_PWR2* register does not deassert the NFAULT signal. 0 = Cycle through the force pass and force fail BIST on the LDO comparators. The *FAULT_PWR1* and *FAULT_PWR2* registers are reset to 0 and NFAULT is deasserted at the end of each LDO BIST run. 1 = Only run the force pass or force fail cycle based on the *[BIST_FORCE_FAIL]* setting. The *FAULT_PWR1* and *FAULT_PWR2* registers are not reset to 0, and NFAULT remains asserted at the end of each LDO BIST run.
PWR_BIST_GO =	When written to 1, the power supply BIST diagnostic will start. Any change *[BIST_NO_CLR]* or *[BIST_FORCE_FAULT]* has no effect until this bit is written to 1 again. The bit self-clears. 0 = Ready 1 = Start power supply BIST diagnostic

8.5.4.10.4 DIAG_COMP_CTRL1

Address	0x033A							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	VSAUX_THR[4:0]							
Reset	0	0	0	0	0	0	0	0

VSAUX_THR[4:0] =	Configures the VS vs. AUX delta. The VS vs. AUX check is considered pass if the measured delta is less than this threshold. Range from 6 to 99 mV in 3-mV step.
RSVD	RESERVED

8.5.4.10.5 DIAG_COMP_CTRL2

Address	0x033B							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	RSVD	GPIO_THR[2:0]			OW_THR[3:0]			
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
GPIO_THR[2:0] =	Configures the GPIO comparison delta threshold between Main and AUX ADC measurements. Range is from 4-mV to 32-mV in 4-mV steps.
OW_THR[3:0] =	Configures the OW detection threshold for diagnostic comparison. This threshold applies to the AUX OW and VS OW diagnostics. Range is from 500 mV to 5 V in 300-mV steps.

8.5.4.10.6 DIAG_COMP_CTRL3

Address	0x033C							
RW	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

Name	RSVD	RSVD	OW_SNK[1:0]		COMP_ADC_SEL[2:0]			COMP_ADC_GO
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
OW_SNK[1:0] =	Turns on current sink on VS pins, AUX pins, or SRP/N pins. Changes to these bits take effect immediately. Host MCU is responsible to turn on the correct sink current before performing open wire (OW) test and to turn off the sink current after OW test is completed. 00 = All VS, SRP/N, AUX pins current sink is off. 01 = Turn on current sink on all VS pins. 10 = Turn on current sink on all AUX pins 11 = Turn on current sink on SRP/N pins.							
COMP_ADC_SEL[2:0] =	Enables the device diagnostic comparison through the ADC measurements. Host enables the corresponding ADCs in continuous mode before enabling this diagnostic. These bits are sampled when *[COMP_ADC_GO]* = 1. 000 = No ADC comparison is performed 001 = VS voltage measurement check. Device compares the INPUT channels specified by *[AUX_IN_SEL4:0]* against the following criteria: VS (from Main ADC) vs. AUX (from AUX ADC) delta is less than *[VSAUX_THR4:0]*. The *[DRDY_VSAUX]* = 1 when this comparison is completed. 010 = Open wire (OW) check on VS pins. MCU enables the current sink on all VS pins through the *[OW_SNK1:0]* before enabling this comparison. Device compares corresponding VS pins against the following criteria: VS (from Main ADC) is less than *DIAG_COMP_CTRL2 [OW_THR3:0]*. The *[DRDY_VS_OW]* = 1 when the comparison is completed. 011 = Open wire (OW) check on VS pins MCU enables the current sink on all VS pins through the *[OW_SNK1:0]* before enabling this comparison. Device compares corresponding AUX pins specified by *[AUX_IN_SEL4:0]* against the following criteria: AUX (from AUX ADC) is less than *DIAG_COMP_CTRL2 [OW_THR3:0]*. The *[DRDY_AUXOW]* = 1 when the comparison is completed. 100 = RSVD 101 = GPIO measurement check (applies to GPIO configured as ADC input). Device compares main GPIO measurement vs. AUX GPIO measurements delta is less than *[GPIO_THR2:0]*. The *[DRDY_GPIO]* = 1 when the comparison is completed. Other codes: No ADC comparison is performed.							
COMP_ADC_GO =	Device starts diagnostic test specified by *[COMP_ADC_SEL2:0]* setting. When this bit is written to 1, the selected *[COMP_ADC_SEL2:0]* is sampled. Change of *[COMP_ADC_SEL2:0]* setting has no effect unless this GO bit is written to 1 again. This bit is cleared to 0 in read. 0 = Ready. Writing 0 has no effect 1 = Star diagnostic selected by *[COMP_ADC_SEL2:0]*.							

8.5.4.10.7 DIAG_COMP_CTRL4

Address	**0x033D**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD						COMP_FAULT_INJ	LPF_FAULT_INJ
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
COMP_FAULT_INJ =	Injects fault to the ADC comparison logic. If any ADC comparison diagnostic is run with this bit set, the comparison result is expected to fail. 0 = Disable 1 = Enable							
LPF_FAULT_INJ =	Injects fault condition to the diagnostic LPF during LPF diagnostic. The *FAULT_COMP_MISC[LPF_FAIL]* is expected to be set. 0 = Disable 1 = Enable							

8.5.4.10.8 DIAG_PROT_CTRL

Address	**0x033E**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**

Name	RSVD							PROT_BIST_NO_RST
Reset	0	0	0	0	0	0	0	0

RSVD = Reserved

PROT_BIST_NO_RST = Use for further diagnostic if the protector BIST detects a failure. When this bit is set to 1, the device will not clear the *FAULT_OV1/2*, *FAULT_UV1/2*, registers. The NFAULT signal will be latched once it is asserted.
Note: Host ensures there is no fault before starting the BIST run with this bit set to 0.
0 = During BIST run, when the device asserts a fault to check the protector comparators and MUX and asserts the correct OV, UV fault bits the NFAULT pin. When this bit is 0, the device clears the fault and deasserts NFAULT before switching to the next channel.
1 = During BIST run, the fault created during the test will not be cleared before switching to next VS or GPIO channel. The NFAULT pin is latched once it is asserted.

8.5.4.11 Fault Configuration and Reset

8.5.4.11.1 FAULT_MSK1

Address	**0x0016**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	MSK_PROT	RSVD	RSVD	MSK_UV	MSK_OV	MSK_COMP	MSK_SYS	MSK_PWR
Reset	0	0	0	0	0	0	0	0

MSK_PROT = Masks the *FAULT_PROT** registers to trigger NFAULT.
0 = Assert NFAULT if any bit from *FAULT_PROT** is set to 1.
1 = No NFAULT action regardless of *FAULT_PROT** bit status.

RSVD = Reserved

MSK_UV = Masks the *FAULT_UV** registers to trigger NFAULT.
0 = Assert NFAULT if any bit from *FAULT_UV** is set to 1.
1 = No NFAULT action regardless of *FAULT_UV** bit status.

MSK_OV = Masks the *FAULT_OV** registers to trigger NFAULT.
0 = Assert NFAULT if any bit from *FAULT_OV** is set to 1.
1 = No NFAULT action regardless of *FAULT_OV** bit status.

MSK_COMP = Masks the *FAULT_COMP_** registers to trigger NFAULT.
0 = Assert NFAULT if any bit from *FAULT_COMP_** is set to 1.
1 = No NFAULT action regardless of *FAULT_COM_** bit status.

MSK_SYS = To mask the NFAULT assertion from any *FAULT_SYS* register bit.
0 = Assert NFAULT if any bit from *FAULT_SYS1* is set to 1.
1 = No NFAULT action regardless of *FAULT_SYS1* bit status.

MSK_PWR = To mask the NFAULT assertion from any *FAULT_PWR1* to *FAULT_PWR3* register bit.
0 = Assert NFAULT if any bit from *FAULT_PWR1* to *FAULT_PWR3* is set to 1.
1 = No NFAULT action regardless of *FAULT_PWR1* to *FAULT_PWR3* bit status.

8.5.4.11.2 FAULT_MSK2

Address	**0x0017**							
NVM	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	SPARE[1]	MSK_OTP_CRC	MSK_OTP_DATA	MSK_COMM3_FCOMM	MSK_COMM3_FTONE	MSK_COMM3_HB	MSK_COMM2	MSK_COMM1
Reset	0	0	0	0	0	0	0	0

SPARE[1] = Spare

MSK_OTP_CRC = Masks the *FAULT_OTP* register (*[CUST_CRC]* and *[FACT_CRC]* only) on NFAULT triggering.
0 = Assert NFAULT if any bit described above is set to 1.
1 = No NFAULT action regardless of the status of the bits described above.

MSK_OTP_DATA = Masks the *FAULT_OTP* register (all bits except *[CUST_CRC]* and *[FACT_CRC]*) on NFAULT triggering.
0 = Assert NFAULT if any bit described above is set to 1.
1 = No NFAULT action regardless of the status of the bits described above.

MSK_COMM3_FCOMM =	Masks *FAULT_COMM3[FCOMM_DET]* fault on NFAULT triggering. 0 = Assert NFAULT if *FAULT_COMM3[FCOMM_DET]* is set to 1. 1 = No NFAULT action regardless of *FAULT_COMM3[FCOMM_DET]* status.
MSK_COMM3_FTONE =	Masks *FAULT_COMM3[FTONE_DET]* fault on NFAULT triggering. 0 = Assert NFAULT if *FAULT_COMM3[FTONE_DET]* is set to 1. 1 = No NFAULT action regardless of *FAULT_COMM3[FTONE_DET]* status.
MSK_COMM3_HB =	Masks *FAULT_COMM3[HB_FAST]* or *[HB_FAIL]* faults on NFAULT triggering. 0 = Assert NFAULT if *FAULT_COMM3[HB_FAST]* or *[HB_FAIL]* is set to 1. 1 = No NFAULT action regardless of *FAULT_COMM3[HB_FAST]* or *[HB_FAIL]* status.
MSK_COMM2 =	Masks *FAULT_COMM2* register on NFAULT triggering. 0 = Assert NFAULT if any bit from *FAULT_COMM2* register is set to 1. 1 = No NFAULT action regardless of *FAULT_COMM2* register bit status.
MSK_COMM1 =	Masks *FAULT_COMM1* register on NFAULT triggering. 0 = Assert NFAULT if any bit from *FAULT_COMM1* register is set to 1. 1 = No NFAULT action regardless of *FAULT_COMM1* register bit status.

8.5.4.11.3 FAULT_RST1

Address	**0x0331**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RST_PROT	RSVD	RSVD	RST_UV	RST_OV	RST_COMP	RST_SYS	RST_PWR
Reset	0	0	0	0	0	0	0	0

RST_PROT =	Resets the *FAULT_PROT1* and *FAULT_PROT2* registers to 0x00. 0 = No reset 1 = Reset registers to 0x00
RSVD =	RESERVED
RSVD =	RESERVED
RST_UV =	Resets all *FAULT_UV** registers to 0x00. 0 = No reset 1 = Reset registers to 0x00
RST_OV =	Resets all *FAULT_OV** registers to 0x00. 0 = No reset 1 = Reset registers to 0x00
RST_COMP =	Resets all *FAULT_COMP_** registers to 0x00. 0 = No reset 1 = Reset registers to 0x00
RST_SYS =	To reset the *FAULT_SYS* register to 0x00. This bit self-clears to 0 after writing to 1. 0 = Do not reset 1 = Reset to 0x00
RST_PWR =	To reset the *FAULT_PWR1* to *FAULT_PWR3* registers to 0x00. This bit self-clears to 0 after writing to 1. 0 = Do not reset 1 = Reset to 0x00

8.5.4.11.4 FAULT_RST2

Address	**0x0332**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD	RST_OTP_CRC	RST_OTP_DATA	RST_COMM3_FCOMM	RST_COMM3_FTONE	RST_COMM3_HB	RST_COMM2	RST_COMM1
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
RST_OTP_CRC =	Resets the *FAULT_OTP* register (*[CUST_CRC]* and *[FACT_CRC]* only). 0 = No reset 1 = Reset the register to 0x00
RST_OTP_DATA =	Resets the *FAULT_OTP* register (*[SEC_DETECT]* and *[DED_DETECT]* only). 0 = No reset 1 = Reset the register to 0x00

RST_COMM3_FCOMM =	Resets *FAULT_COMM3[FCOMM_DET]*. 0 = No reset 1 = Reset the related bit to 0
RST_COMM3_FTONE =	Resets *FAULT_COMM3[FTONE_DET]*. 0 = No reset 1 = Reset the related bit to 0
RST_COMM3_HB =	Resets *FAULT_COMM3[HB_FAST]* and *[HB_FAIL]* bits. 0 = No reset 1 = Reset the related bits to 0
RST_COMM2 =	Resets *FAULT_COMM2*, *DEBUG_COML**, and *DEBUG_COMM_COMH** registers. 0 = No reset 1 = Reset registers to 0x00
RST_COMM1 =	Resets *FAULT_COMM1* and *DEBUG_COMM_UART** registers. 0 = No reset 1 = Reset registers to 0x00

8.5.4.12 Fault Status

8.5.4.12.1 FAULT_SUMMARY

This register is the soft version of the NFAULT.

Address	**0x052D**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	FAULT_PROT	FAULT_COMP_ADC	FAULT_OTP	FAULT_COMM	RSVD	FAULT_OVUV	FAULT_SYS	FAULT_PWR
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
FAULT_PROT =	This bit is set if *[MSK_PROT]* = 0 and any of the *FAULT_PROT1* or *FAULT_PROT2* register bits is set. 0 = No protector (OVUV comparators) fault. 1 = Protector fault is detected							
FAULT_COMP_ADC =	This bit is set if *[MSK_COMP]* = 0 and any of the following registers is set: • *FAULT_COMP_VSAUX1/2* • *FAULT_COMP_VSOW1/2* • *FAULT_COMP_AUXOW1/2* • *FAULT_COMP_GPIO* • *FAULT_COMP_MISC* 0 = No ADC comparison fault (that is, none of the *FAULT_COMP_** registers are set). 1 = ADC comparison fault is detected.							
FAULT_OTP =	This bit is set if *[MSK_OTP]* = 0 and any of the *FAULT_OTP* register bits is set. 0 = No OTP-related fault detected or OTP faults are masked. 1 = OTP-related fault is detected.							
FAULT_COMM =	This bit is set if any of the following is true: • *[MSK_COMM1]* = 0 and any of the *FAULT_COMM1* register bits is set. • *[MSK_COMM2]* = 0 and any of the *FAULT_COMM2* register bits is set. • *[MSK_COMM3_HB]* = 0 and the *FAULT_COMM3[HB_FAST]* bit or *[HB_FAIL]* bit is set. • *[MSK_COMM3_FTONE]* = 0 and the *FAULT_COMM3[FTONE_DET]* is set. • *[MSK_COMM3_FCOMM]* = 0 and if *FAULT_COMM3[FCOMM_DET]* is set. 0 = No UART, VIF, or FTONE fault is detected, or UART, VIF, and FTONE faults are masked. 1 = UART, VIF fault is detected.							
FAULT_OVUV =	This bit is set if any of the following is true: • *[MSK_OV]* = 0 and any of the *FAULT_OV1* or *FAULT_OV2* bits is set. • *[MSK_UV]* = 0 and any of the *FAULT_UV1* or *FAULT_UV2* bits is set. 0 = No OV or UV fault is detected, or OV and UV faults are masked. 1 = OV or UV fault is detected.							

FAULT_SYS =	This bit is set if *[MSK_SYS]* = 0 and any of the *FAULT_SYS1* register bits is set. 0 = No system related fault detected or system faults are masked. 1 = System related fault is detected.
FAULT_PWR =	This bit is set if *[MSK_PWR]* = 0 and any of the *FAULT_PWR1* to *FAULT_PWR3* register bits is set. 0 = No power rail related fault is detected or power rail faults are masked. 1 = Power rail related fault is detected.

8.5.4.12.2 FAULT_COMM1

Address	**0x0530**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			UART_TR	UART_RR	UART_RC	COMMCLR_DET	STOP_DET
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
UART_TR =	Indicates a UART FAULT is detected when transmitting a response frame. Further details of the fault information are available in the *DEBUG_UART_RR_TR* register. 0 = No fault 1 = Fault
UART_RR =	Indicates a UART FAULT is detected when receiving a response frame. Further details of the fault information are available in the *DEBUG_UART_RR_TR* register. 0 = No fault 1 = Fault
UART_RC =	Indicates a UART FAULT is detected during receiving a command frame. Further details of the fault information are available in the *DEBUG_UART_RC* register. 0 = No fault 1 = Fault
COMMCLR_DET =	A UART communication clear signal is detected. A detection of SLEEPtoACTIVE ping in ACTIVE or SLEEP mode or detection of WAKE pin in ACTIVE mode will also set this bit. 0 = No UART Clear 1 = UART Clear detected
STOP_DET =	Indicates an unexpected STOP condition is received. A detection of SLEEPtoACTIVE signal in ACTIVE mode will also set this bit. 0 = No fault 1 = Fault

8.5.4.12.3 FAULT_COMM2

Address	**0x0531**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COML_TR	COML_RR	COML_RC	COML_BIT	COMH_TR	COMH_RR	COMH_RC	COMH_BIT
Reset	0	0	0	0	0	0	0	0

COML_TR =	Indicates a COML byte level fault is detected when transmitting a response frame. Further details of the fault information are available in the *DEBUG_COML_RR_TR* register. 0 = No fault 1 = Fault
COML_RR =	Indicates a COML byte level fault is detected when receiving a response frame. Further details of the fault information are available in the *DEBUG_COML_RR_TR* register. 0 = No fault 1 = Fault
COML_RC =	Indicates a COML byte level fault is detected when receiving a command frame. Further details of the fault information are available in the *DEBUG_COML_RR_RC* register. 0 = No fault 1 = Fault
COML_BIT =	Indicates a COML bit level fault is detected which would cause at least one byte level fault. Further details of the fault information are available in the *DEBUG_COML_BIT* register. 0 = No fault 1 = Fault

COMH_TR =	Indicates a COMH byte level fault is detected when transmitting a response frame. Further details of the fault information are available in the *DEBUG_COMH_RR_TR* register. 0 = No fault 1 = Fault
COMH_RR =	Indicates a COMH byte level fault is detected when receiving a response frame. Further details of the fault information are available in the *DEBUG_COMH_RR_TR* register. 0 = No fault 1 = Fault
COMH_RC =	Indicates a COMH byte level fault is detected when receiving a command frame. Further details of the fault information are available in the *DEBUG_COMH_RR_RC* register. 0 = No fault 1 = Fault
COMH_BIT =	Indicates a COMH bit level fault is detected which would cause at least one byte level fault. Further details of the fault information are available in the *DEBUG_COMH_BIT* register. 0 = No fault 1 = Fault

8.5.4.12.4 FAULT_COMM3

Address	**0x0532**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD				FCOMM_DET	FTONE_DET	HB_FAIL	HB_FAST
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
FCOMM_DET =	Received communication transaction with the Fault Status bits set by any of the upper stack device(s). 0 = Fault Status are clear, indicating no fault is detected from any of the upper stack device(s). 1 = Fault Status are set from the receiving communication transaction.
FTONE_DET =	Indicates a FAULT TONE is received. Detection is monitoring the COML side if *[DIR_SEL]* = 0 and vice versa. 0 = No FAULT TONE detected 1 = FAULT TONE detected
HB_FAIL =	Indicates HEARTBEAT is not received within an expected time. Detection is monitoring the COML side if *[DIR_SEL]* = 0 and vice versa. 0 = No fault 1 = Fault
HB_FAST =	Indicates HEARTBEAT is received too frequently. Detection is monitoring the COML side if *[DIR_SEL]* = 0 and vice versa. This bit may also be set when *[FTONE_DET]* = 1 depends on how soon the FAULT TONE is detected from the previous HEARTBEAT. 0 = No fault 1 = Fault

8.5.4.12.5 FAULT_OTP

Address	**0x0535**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD	DED_DET	SEC_DET	CUST_CRC	FACT_CRC	CUSTLDERR	FACTLDERR	GBLOVERR
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
DED_DET =	Indicates a DED error has occurred during the OTP load. (Unknown during encoding) 0 = No fault 1 = Fault
SEC_DET =	Indicates a SEC error has occurred during the OTP load. (Unknown during encoding) 0 = No fault 1 = Fault
CUST_CRC =	Indicates a CRC error has occurred in the customer register space. 0 = No fault 1 = Fault

FACT_CRC =	Indicates a CRC error has occurred in the factory register space. 0 = No fault 1 = Fault
CUSTLDERR =	Indicates errors during the customer space OTP load process. Read *OTP_CUST1_STAT* and *OTP_CUST2_STAT* registers for the specific error condition. This error bit is set if one of the following is true: • No Customer OTP page is programmed. • The highest Customer OTP page has a *[FMTERR]*. • The highest Customer OTP page has *[TRY]* = 1 and is not *[PROGOK]*. • LOADERR happened on the selected Customer OTP page. Information received from the device with this error must not be considered reliable. Writing *[RST_OTP_DATA]* = 1 does not reset this bit. To recheck this error, a device reset or HW_RESET is needed. 0 = No fault 1 = Fault
FACTLDERR =	Indicates errors during the factory space OTP load process. This error bit is set if one of the following is true: • No factory OTP page is programmed. • The highest factory OTP page has a *[FMTERR]*. • The highest factory OTP page has *[TRY]* = 1 and is not *[PROGOK]*. • LOADERR happened on the selected factory OTP page. Information received from the device with this error must not be considered reliable. Writing *[RST_OTP_DATA]* = 1 does not reset this bit. To recheck this error, a device reset or HW_RESET is needed. 0 = No fault 1 = Fault
GBLOVERR =	Indicates that on overvoltage error is detected on one of the OTP pages. Read *OTP_CUST1_STAT* and *OTP_CUST2_STAT* registers to determine the specific page(s). Information received from the device with this error must not be considered reliable. Writing *[RST_OTP_DATA]* = 1 does not reset this bit. To clear this bit, a device reset or HW_RESET is needed. Repeat the programming procedure on a different page (if available) will force the device to re-evaluate the condition. 0 = No fault 1 = Fault

8.5.4.12.6 FAULT_SYS

Address	**0x0536**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	LFO	RSVD	GPIO	DRST	CTL	CTS	TSHUT	TWARN
Reset	0	0	0	0	0	0	0	0

LFO =	Indicated LFO frequency is outside an expected range 0 = No fault detected 1 = Fault detected
RSVD =	Reserved
GPIO =	Indicates GPIO8 detects a FAULT input when *GPIO_CONF1[FAULT_IN_EN]* = 1. 0 = No fault detected 1 = FAULT input detected
DRST =	Indicates a digital reset has occurred. 0 = No digital reset 1 = Digital reset has occurred
CTL =	Indicates a long communication timeout occurred. Device action is configured by *[CTL_ACT]*. This bit is not observable if the action is set to device shutdown. 0 = No fault 1 = Long communication timeout occurs. Observable if long timeout action is set to SLEEP.
CTS =	Indicates a short communication timeout occurred. No action from the device. This can be served as an alert to system before reaching long communication timeout. 0 = No fault 1 = Short communication timeout occurs

TSHUT =	Indicates the previous shutdown was a thermal shutdown, in which the die temperature (die temp 2) is higher than the thermal shutdown threshold. 0 = Die temperature is less than thermal shutdown threshold 1 = The previous shutdown was a thermal shutdown
TWARN =	Indicates the die temperature (die temp 2) is higher than the *TWARN_THR[1:0]* setting. No action is taken by the device at the moment yet. This serves as a warning signal that the die temperature is approaching thermal shutdown. 0 = Die temperature is less than *TWARN_THR[1:0]* 1 = Die temperature is greater than *TWARN_THR[1:0]*

8.5.4.12.7 FAULT_PROT1

Address	**0x053A**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD							VPARITY_FAIL
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
VPARITY_FAIL =	Indicates a parity fault is detected on any of the following OVUV related configurations: • OV or UV threshold setting • *[OVUV_MODE1:0]* setting 0 = No fault 1 = Fault

8.5.4.12.8 FAULT_PROT2

Address	**0x053B**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD	BIST_ABORT	RSVD	VPATH_FAIL	RSVD	RSVD	OVCOMP_FAIL	UVCOMP_FAIL
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
BIST_ABORT =	Indicates either OVUV BIST run is aborted. 0 = BIST runs to completion 1 = BIST abort
VPATH_FAIL =	Indicates a fault is detected along the OVUV signal path during BIST test. 0 = No fault 1 = Fault
OVCOMP_FAIL =	Indicates the OV comparator fails during BIST test. 0 = No fault 1 = Fault
UVCOMP_FAIL =	Indicates the UV comparator fails during BIST test. 0 = No fault 1 = Fault

8.5.4.12.9 FAULT_OV1

Address	**0x053C**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	OV16_DET	OV15_DET	OV14_DET	OV13_DET	OV12_DET	OV11_DET	OV10_DET	OV9_DET
Reset	0	0	0	0	0	0	0	0

OV9_DET to OV16_DET =	OV fault status for VS9 to VS16, results are from the OV comparator detection.

8.5.4.12.10 FAULT_OV2

Address	0x053D							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	OV8_DET	OV7_DET	OV6_DET	OV5_DET	OV4_DET	OV3_DET	OV2_DET	OV1_DET
Reset	0	0	0	0	0	0	0	0
OV1_DET to OV8_DET =	OV fault status for VS1 to VS8, results are from the OV comparator detection.							

8.5.4.12.11 FAULT_UV1

Address	0x053E							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	UV16_DET	UV15_DET	UV14_DET	UV13_DET	UV12_DET	UV11_DET	UV10_DET	UV9_DET
Reset	0	0	0	0	0	0	0	0
UV9_DET to UV16_DET =	UV fault status for VS9 to VS16, results are from the UV comparator detection.							

8.5.4.12.12 FAULT_UV2

Address	0x053F							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	UV8_DET	UV7_DET	UV6_DET	UV5_DET	UV4_DET	UV3_DET	UV2_DET	UV1_DET
Reset	0	0	0	0	0	0	0	0
UV1_DET to UV8_DET =	UV fault status for VS1 to VS8, results are from the UV comparator detection.							

8.5.4.12.13 FAULT_COMP_GPIO

Address	0x0543							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	GPIO8_FAIL	GPIO7_FAIL	GPIO6_FAIL	GPIO5_FAIL	GPIO4_FAIL	GPIO3_FAIL	GPIO2_FAIL	GPIO1_FAIL
Reset	0	0	0	0	0	0	0	0
GPIO1_FAIL to GPIO8_FAIL =	Indicates ADC vs. AUX ADC GPIO measurement diagnostic results for GPIO1 to GPIO8. 0 = Diagnostic pass 1 = Diagnostic fail. GPIO from Main ADC vs. AUX ADC measurement is greater than *[GPIO_THR2:0]*							

8.5.4.12.14 FAULT_COMP_VSAUX1

Address	0x0545							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	VS16_FAIL	VS15_FAIL	VS14_FAIL	VS13_FAIL	VS12_FAIL	VS11_FAIL	VS10_FAIL	VS9_FAIL
Reset	0	0	0	0	0	0	0	0
VS9_FAIL to VS16_FAIL =	Indicates voltage diagnostic results for VS9 to VS16. 0 = Diagnostic pass 1 = Diagnostic fail. VS vs. AUX measurement is greater than *[VSAUX_THR4:0]*							

8.5.4.12.15 FAULT_COMP_VSAUX2

Address	0x0546							
Read Only	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Name	VS8_FAIL	VS7_FAIL	VS6_FAIL	VS5_FAIL	VS4_FAIL	VS3_FAIL	VS2_FAIL	VS1_FAIL

Reset	0	0	0	0	0	0	0	0
VS1_FAIL to VS8_FAIL =	Indicates voltage diagnostic results for VS1 to VS8. 0 = Diagnostic pass 1 = Diagnostic fail. VS vs. AUX measurement is greater than *[VSAUX_THR4:0]*							

8.5.4.12.16 FAULT_COMP_VSOW1

Address	**0x0548**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	VSOW16_FAIL	VSOW15_FAIL	VSOW14_FAIL	VSOW13_FAIL	VSOW12_FAIL	VSOW11_FAIL	VSOW10_FAIL	VSOW9_FAIL
Reset	0	0	0	0	0	0	0	0
VSOW9_FAIL to VSOW16_FAIL =	Indicates VS OW diagnostic results for VS9 to VS16. 0 = Diagnostic pass 1 = Diagnostic fail. VS measurement is less than *[OW_THR3:0]*							

8.5.4.12.17 FAULT_COMP_VSOW2

Address	**0x0549**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	VSOW8_FAIL	VSOW7_FAIL	VSOW6_FAIL	VSOW5_FAIL	VSOW4_FAIL	VSOW3_FAIL	VSOW2_FAIL	VSOW1_FAIL
Reset	0	0	0	0	0	0	0	0
VSOW1_FAIL to VSOW8_FAIL =	Indicates VS OW diagnostic results for VS1 to VS8. 0 = Diagnostic pass 1 = Diagnostic fail. VS measurement is less than *[OW_THR3:0]*							

8.5.4.12.18 FAULT_COMP_AUXOW1

Address	**0x054B**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	AUXOW16_FAIL	AUXOW15_FAIL	AUXOW14_FAIL	AUXOW13_FAIL	AUXOW12_FAIL	AUXOW11_FAIL	AUXOW10_FAIL	AUXOW9_FAIL
Reset	0	0	0	0	0	0	0	0
AUXOW9_FAIL to AUXOW16_FAIL =	Results of the AUXOW diagnostic for AUX9 to AUX16. 0 = Pass 1 = Fail							

8.5.4.12.19 FAULT_COMP_AUXOW2

Address	**0x054C**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	AUXOW8_FAIL	AUXOW7_FAIL	AUXOW6_FAIL	AUXOW5_FAIL	AUXOW4_FAIL	AUXOW3_FAIL	AUXOW2_FAIL	AUXOW1_FAIL
Reset	0	0	0	0	0	0	0	0
AUXOW1_FAIL to AUXOW8_FAIL =	Results of the AUXOW diagnostic for AUX1 to AUX8. 0 = Pass 1 = Fail							

8.5.4.12.20 FAULT_COMP_MISC

Address	**0x0550**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**

Name	RSVD						COMP_ADC_ABORT	LPF_FAIL
Reset	0	0	0	0	0	0	0	0

RSVD = Reserved

COMP_ADC_ABORT = Indicates the most recent ADC comparison diagnostic is aborted due to improper setting. Valid only if one of the ADC comparison diagnostics has started.
0 = ADC comparison diagnostic run to completion
1 = ADC comparison diagnostic is aborted

LPF_FAIL = Indicates LPF diagnostic result.
0 = Pass
1 = Fail

8.5.4.12.21 FAULT_PWR1

Address	0x0552							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CVSS_OPEN	DVSS_OPEN	REFHM_OPEN	CVDD_UV	CVDD_OV	DVDD_OV	AVDD_OSC	AVDD_OV
Reset	0	0	0	0	0	0	0	0

CVSS_OPEN = Indicates an open condition on CVSS pin.
0 = No fault
1 = Fault

DVSS_OPEN = Indicates an open condition on DVSS pin.
0 = No fault
1 = Fault

REFHM_OPEN = Indicates an open condition on REFHM pin.
0 = No fault
1 = Fault

CVDD_UV = Indicates an undervoltage fault on the CVDD LDO.
0 = No fault
1 = Fault

CVDD_OV = Indicates an overvoltage fault on the CVDD LDO.
0 = No fault
1 = Fault

DVDD_OV = Indicates an overvoltage fault on the DVDD LDO.
0 = No fault
1 = Fault

AVDD_OSC = Indicates AVDD is oscillating outside of acceptable limits.
0 = No fault
1 = Fault
This fault could trigger when transitioning from SLEEP to ACTIVE mode. So, if this fault is set, ignore it and reset the fault.

AVDD_OV = Indicates an overvoltage fault on the AVDD LDO.
0 = No fault
1 = Fault

8.5.4.12.22 FAULT_PWR2

Address	0x0553							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD	PWRBIST_FAIL	RSVD	REFH_OSC	NEG5V_UV	TSREF_OSC	TSREF_UV	TSREF_OV
Reset	0	0	0	0	0	0	0	0

RSVD = Reserved

PWRBIST_FAIL =	Indicates a fail on the power supply BIST run. 0 = No fault 1 = Fault
REFH_OSC =	Indicates REGH reference is oscillating outside of an acceptable limit. 0 = No fault 1 = Fault
NEG5V_UV =	Indicates an undervoltage fault on the NEG5V charge pump. 0 = No fault 1 = Fault
TSREF_OSC =	Indicates TSREF is oscillating outside of an acceptable limit. 0 = No fault 1 = Fault
TSREF_UV =	Indicates an undervoltage fault on the TSREF LDO. 0 = No fault 1 = Fault
TSREF_OV =	Indicates an overvoltage fault on the TSREF LDO. 0 = No fault 1 = Fault

8.5.4.12.23 FAULT_PWR3

Address	0x0554							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD					RSVD	RSVD	AVDDUV_DRST
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
AVDDUV_DRST =	Indicates a digital reset occurred due to AVDD UV detected. This also applies when device wakes up after a SHUTDOWN or HW Reset event. 0 = No reset 1 = Digital reset occurred due to AVDD UV

8.5.4.13 Debug Control and Status

8.5.4.13.1 DEBUG_CTRL_UNLOCK

Address	0x0700							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CODE[7:0]							
Reset	0	0	0	0	0	0	0	0

CODE[7:0] =	Write the unlock code (0xA5) to this register to activate the setting in the *DEBUG_COMM_CTRL** register. Any other value than the unlock code will deactivate any effect in the *DEBUG_COMM_CTRL** setting and return to the normal settings of the device.

8.5.4.13.2 DEBUG_COMM_CTRL1

Address	0x0701							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			UART_BAUD	UART_MIRROR_EN	UART_TX_EN	USER_UART_EN	USER_DAISY_EN
Reset	0	0	0	0	0	1	0	0

RSVD =	Reserved
UART_BAUD =	This bit changes the UART baud rate to 250kb/s. Useful on VIF debug. When system sets all daisy-chain devices to the 250kb/s baud rate, it slows down the response byte through the VIF to increase the robustness of the VIF for debug purposes. 0 = Default 1Mb/s 1 = 250kb/s

UART_MIRROR_EN =	This bit enables the stack VIF communication to mirror to UART. To use this debug function, the stack device's UART TX has to be enabled first by setting *[UARTTX_EN]* = 1. 0 = Disable 1 = Enable
UART_TX_EN =	Stack device, by default, has the UART TX disabled. This bit enables the UART TX to allow read/write via UART on the stack device. 0 = Disable 1 = Enable
USER_UART_EN =	This bit enables the debug UART control bits, *[UART_TX_EN]* and *[UART_MIRROR_EN]*. 0 = The setting of the bits mentioned above has no effect. 1 = The device configures the UART per *[UART_TX_EN]* and *[UART_MIRROR_EN]* settings
USER_DAISY_EN =	This bit enables the debug COML and COMH control bits in the *DEBUG_COMM_CTRL2* register 0 = The setting of *DEBUG_COMM_CTRL2* register has no effect. 1 = The device configures the COML and COMH per *DEBUG_COMM_CTRL2* register setting.

8.5.4.13.3 DEBUG_COMM_CTRL2

Address	**0x0702**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD				COML_TX_EN	COML_RX_EN	COMH_TX_EN	COMH_RX_EN
Reset	0	0	0	0	1	1	1	1

RSVD =	Reserved
COML_TX_EN =	Enables COML transmitter. 0 = Disable 1 = Enable
COML_RX_EN =	Enables COML receiver. 0 = Disable 1 = Enable
COMH_TX_EN =	Enables COMH transmitter. 0 = Disable 1 = Enable
COMH_RX_EN =	Enables COMH receiver. 0 = Disable 1 = Enable

8.5.4.13.4 DEBUG_COMM_STAT

Address	**0x0780**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD		HW_UART_DRV	HW_DAISY_DRV	COML_TX_ON	COML_RX_ON	COMH_TX_ON	COMH_RX_ON
Reset (Base)	0	0	1	1	0	0	1	1
Reset (Stack)	0	0	1	1	1	1	1	1

RSVD =	Reserved
HW_UART_DRV =	Indicates the UART TX is controlled by the device itself or by MCU control. Applicable to the stack device in which the UART TX is disabled by default once a device is configured as 'STACK'. 0 = The *DEBUG_COMM_CTRL1[USER_UART_EN]* = 1. UART TX is under manual control through the *DEBUG_COMM_CTRL2* register. 1 = UART TX is controlled by the device
HW_DAISY_DRV =	Indicates the COML and COMH are controlled by the device itself or by MCU control. 0 = The *DEBUG_COMM_CTRL1[USER_DAISY_EN]* = 1. COML and COMH are under manual control through the *DEBUG_COMM_CTRL2* register. 1 = COML and COMH are controlled by the device

COML_TX_ON =	Shows the current COML transmitter status. 0 = off 1 = on
COML_RX_ON =	Shows the current COML receiver status. 0 = off 1 = on
COMH_TX_ON =	Shows the current COMH transmitter status. 0 = off 1 = on
COMH_RX_ON =	Shows the current COMH receiver status. 0 = off 1 = on

8.5.4.13.5 DEBUG_UART_RC

Address	**0x0781**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD		RC_IERR	RC_TXDIS	RC_SOF	RC_BYTE_ERR	RC_UNEXP	RC_CRC
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
RC_IERR =	Detects initialization byte error in the received command frame. This may be due to the frame initialization byte has a stop error, incorrect frame type is set, or reserved command type bit is set. All bytes that follow are ignored until a communication CLEAR is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected
RC_TXDIS =	Detects if UART TX is disabled, but the host MCU has issued a command to read data from the device. 0 = No error 1 = Error detected
RC_SOF =	Detects a start-of-frame (SOF) error. That is, an UART CLEAR is received on the UART before the current frame is finished. 0 = No error 1 = Error detected
RC_BYTE_ERR =	Detects any byte error, other than the error in the initialization byte, in the received command frame. All bytes that follow are ignored until a communication CLEAR is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected
RC_UNEXP =	In a stack device (that is, *[STACK_DEV]* = 1 and *[MULTIDROP]* = 0), it is not expected to receive a stack or broadcast command through the UART interface. If so, this is detected as an error and this bit is set. If device is configured with *[MULTIDROP]* = 1, this bit will not be set. 0 = No error 1 = Error detected
RC_CRC =	Detects a CRC error in the received command frame from UART. The frame will be considered as discarded frame. 0 = No error 1 = Error detected

8.5.4.13.6 DEBUG_UART_RR_TR

Address	**0x0782**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			TR_SOF	TR_WAIT	RR_SOF	RR_BYTE_ERR	RR_CRC
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
TR_SOF =	Indicates that a UART CLEAR is received while the device is still transmitting data. 0 = No error 1 = Error detected
TR_WAIT =	The device is waiting for its turn to transfer a response out but the action is terminated because either: • The device receives a UART CLEAR signal. • The device receives a new command. This bit is valid when broadcast or stack read command has been issued. 0 = No error 1 = Error detected
RR_SOF =	Indicates a UART CLEAR is received while receiving the response frame. Response frames on the UART only apply in multidrop mode. 0 = No error 1 = Error detected
RR_BYTE_ERR =	Detects any byte error, other than the error in the initialization byte, in the received response frame. All bytes that follow are ignored until a communication CLEAR is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected
RR_CRC =	Detects are CRC error in the received response frame from UART. The frame will be considered as a discarded frame. 0 = No error 1 = Error detected

8.5.4.13.7 DEBUG_COMH_BIT

Address	**0x0783**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			PERR	BERR_TAG	SYNC2	SYNC1	BIT
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
PERR =	Detects abnormality of the incoming communication frame and hence, the device will forward the communication frame with *[BERR]* bit set. Any error bit that is set in this register will also set the *[PERR]* bit. However, an abnormality that is not classified in the register can also trigger the *[PERR]* bit (for example, detecting missing data or wrong data order). 0 = No communication error detected, the forwarded communication frame does not have the *[BERR]* inserted. 1 = Detected abnormality of the received communication frame. *[BERR]* is asserted to the forwarded communication.
BERR_TAG =	Set when the received communication is tagged with *[BERR]* = 1. 0 = Received communication frame has no *[BERR]* 1 = Received communication frame has *[BERR]*
SYNC2 =	The Preamble half-bit and the *[SYNC1:0]* bits are detected. Device is using the timing information extracted from these bits but it is unable to detect valid data. 0 = No error 1 = Error detected
SYNC1 =	Unable to detect the preamble half-bit or any of the *[SYNC1:0]* bits. It could be the bit is missing or the signal is too distorted to be detectable. 0 = No error 1 = Error detected
BIT =	The device has detected a data bit; however, the detection samples are not enough to assure a strong 1 or 0. 0 = No error 1 = Error detected

8.5.4.13.8 DEBUG_COMH_RC

Address	**0x0784**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**

Name	RSVD		RC_IERR	RC_TXDIS	RC_SOF	RC_BYTE_ERR	RC_UNEXP	RC_CRC
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
RC_IERR =	Detects initialization byte error in the received command frame. This may due to improper formatting of a byte such as a frame initialization byte is expected, but start-of-frame (SOF) bit is not set, or an invalid frame type is selected. Because bytes received on the COMH/COML are propagated up the stack, it is likely devices in the upper stack will also detect this error. All bytes that follow are ignored until a SOF bit is set is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = No error 1 = Error detected							
RC_TXDIS =	Valid when *[DIR_SEL]* = 1. Device detects the COMH TX is disabled but the device receives a command to read data (that is, to transmit data out). The command frame will be counted as a discard frame. 0 = No error 1 = Error detected							
RC_SOF =	Valid when *[DIR_SEL]* = 1. Detects a start-of-frame (SOF) error on COMH. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected							
RC_BYTE_ERR =	Valid when *[DIR_SEL]* = 1. Detected any byte error, other than the error in the initialization byte, in the received command frame. This error can trigger one or more error bit set in the *DEBUG_COMMH_BIT* register. 0 = No error 1 = Error detected							
RC_UNEXP =	If *[DIR_SEL]* = 0, but device receives command frame from COMH which is an invalid condition and device will set this error bit. 0 = No error 1 = Error detected							
RC_CRC =	Indicates a CRC error that resulted in one or more COMH command frames being discarded. Any other errors in the frame are not indicated as the frame was discarded. 0 = No error 1 = Error detected							

8.5.4.13.9 DEBUG_COMH_RR_TR

Address	**0x0785**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD		TR_WAIT	RR_TXDIS	RR_SOF	RR_BYTE_ERR	RR_UNEXP	RR_CRC
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
TR_WAIT =	The device is waiting for its turn to transfer a response out but the action is terminated because the device receives a new command. This bit is valid when broadcast or stack read command has been issued. 0 = No error 1 = Error detected							
RR_TXDIS =	Valid when *[DIR_SEL]* = 0, device receives a response but fails to transmit to the next device because the COMH TX is disabled. The frame is counted as a discarded frame. 0 = No error 1 = Error detected							
RR_SOF =	Valid when *[DIR_SEL]* = 0. Detects a start-of-frame (SOF) error on COMH. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected							
RR_BYTE_ERR =	Valid when *[DIR_SEL]* = 0. Detects any byte error, other than the error in the initialization byte, in the received response frame. This error can trigger one or more error bits set in the *DEBUG_COMMH_BIT* register. 0 = No error 1 = Error detected							

RR_UNEXP =	If *[DIR_SEL]* = 1, but device received response frame from COMH which is an invalid condition and device sets this error bit. 0 = No error 1 = Error detected
RR_CRC =	Indicates a CRC error that resulted in one or more COMH response frames being discarded. Most other errors in the frame are not indicated as the frame was discarded. If *[RR_BYTE_ERR]* is observed on the final byte of the CRC, both CRC and BERR will be indicated. 0 = No error 1 = Error detected

8.5.4.13.10 DEBUG_COML_BIT

Address	**0x0786**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD			PERR	BERR_TAG	SYNC2	SYNC1	BIT
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
PERR =	Detect abnormality of the incoming communication frame and the outgoing communication frame will be set with BERR. Any error bit that is set in this register will also set the *[PERR]* bit. However, abnormality that is not classified in the register can also trigger the *[PERR]* bit (for example, detecting missing data or wrong data order. 0 = No communication error detected, the forwarded communication frame does not have the BERR inserted 1 = Detected abnormality of the received communication frame. BERR is asserted to the forwarded communication.
BERR_TAG =	Set when the received communication is tagged with BERR. 0 = Received communication frame has no BERR 1 = Received communication frame has BERR
SYNC2 =	The Preamble half-bit and the *[SYNC1:0]* bits are detected. Device is using the timing information that is extracted from these bits but it is unable to detect valid data. 0 = No error 1 = Error detected
SYNC1 =	Unable to detect the preamble half-bit or any of the *[SYNC1:0]* bits. It could be the bit is missing or the signal is too distorted to be detectable. 0 = No error 1 = Error detected
BIT =	The device has detected a data bit. However, the detection samples are not enough to assure a strong 1 or 0. 0 = No error 1 = Error detected

8.5.4.13.11 DEBUG_COML_RC

Address	**0x0787**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD		RC_IERR	RC_TXDIS	RC_SOF	RC_BYTE_ERR	RC_UNEXP	RC_CRC
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
RC_IERR =	Detected initialization byte error in the received command frame. This may due to improper formatting of a byte such as a frame initialization byte is expected, but start-of-frame (SOF) bit is not set, or an invalid frame type is selected. Because bytes received on the COMH/COML are propagated up the stack, it is likely devices in the upper stack will also detect this error. All bytes that follow are ignored until a SOF bit is received. When a communication frame is ignored, the device will not attempt to detect any communication error in the ignored frame nor counting it as valid/discard in the frame counters. 0 = Error not detected 1 = Error detected
RC_TXDIS =	Valid when *[DIR_SEL]* = 0. Device detects the COML TX is disabled but the device receives a command to read data (that is, to transmit data out). The command frame will be counted as a discarded frame. 0 = No error 1 = Error detected

RC_SOF =	Valid when *[DIR_SEL]* = 0. Detects a start-of-frame (SOF) error on COML. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected
RC_BYTE_ERR =	Valid when *[DIR_SEL]* = 0. Detected any byte error, other than the error in the initialization byte, in the received command frame. This error can trigger one or more error bits set in the *DEBUG_COMML_BIT* register. 0 = No error 1 = Error detected
RC_UNEXP =	If *[DIR_SEL]* = 1, but device received command frame from COML which is an invalid condition and device will set this error bit. 0 = No error 1 = Error detected
RC_CRC =	Indicates a CRC error that resulted in one or more COML command frames being discarded. Any other errors in the frame are not indicated as the frame was discarded. 0 = No error 1 = Error detected

8.5.4.13.12 DEBUG_COML_RR_TR

Address	**0x0788**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD		TR_WAIT	RR_TXDIS	RR_SOF	RR_BYTE_ERR	RR_UNEXP	RR_CRC
Reset	0	0	0	0	0	0	0	0

RSVD =	Reserved
TR_WAIT =	The device is waiting for its turn to transfer a response out but the action is terminated because the device receives a new command. This bit is valid when broadcast or stack read command has been issued. 0 = No error 1 = Error detected
RR_TXDIS =	Valid when *[DIR_SEL]* = 1, device receives a response but fails to transmit to the next device because the COML TX is disabled. The frame is counted as a discarded frame. 0 = No error 1 = Error detected
RR_SOF =	Valid when *[DIR_SEL]* = 1. Detects a start-of-frame (SOF) error on COML. The SOF bit is set only in the initialization frame but the SOF bit is set in the current frame that is not expected. 0 = No error 1 = Error detected
RR_BYTE_ERR =	Valid when *[DIR_SEL]* = 1. Detects any byte error, other than the error in the initialization byte, in the received response frame. This error can trigger one or more error bits set in the *DEBUG_COMML_BIT* register. 0 = No error 1 = Error detected
RR_UNEXP =	If *[DIR_SEL]* = 0, but device received a response frame from COML which is an invalid condition and device will set this error bit. 0 = No error 1 = Error detected
RR_CRC =	Indicates a CRC error that resulted in one or more COML response frames being discarded. Most other errors in the frame are not indicated as the frame was discarded. If *[RR_BYTE_ERR]* is observed on the final byte of the CRC, both CRC and BERR are indicated. 0 = No error 1 = Error detected

8.5.4.13.13 DEBUG_UART_DISCARD

Address	**0x0789**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0

COUNT[7:0] =	UART frame counter to track the number of discard frames received or transmitted. The registers of the *DEBUG_UART_DISCARD* and *DEBUG_UART_VALID** are latched and the related counters are reset when this register is read.

8.5.4.13.14 DEBUG_COMH_DISCARD

Address	0x078A							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0

COUNT[7:0] =	COMH frame counter to track the number of discard frames received or transmitted. The registers of the *DEBUG_COMH_DISCARD* and *DEBUG_COMH_VALID** are latched and the related counters are reset when this register is read.

8.5.4.13.15 DEBUG_COML_DISCARD

Address	0x078B							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0

COUNT[7:0] =	COML frame counter to track the number of discard frames received or transmitted. The registers of the *DEBUG_COML_DISCARD* and *DEBUG_COML_VALID** are latched and the related counters are reset when this register is read.

8.5.4.13.16 DEBUG_UART_VALID_HI/LO

DEBUG_UART_VALID_HI

Address	0x078C							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0

COUNT[7:0] =	The high-byte of UART frame counter to track the number of valid frames received or transmitted. Counter saturates when both *DEBUG_UART_VALID_HI/LO* is 0xFF. This register is latched and the related counter is reset when *DEBUG_UART_DISCARD* is read.

DEBUG_UART_VALID_LO

Address	0x078D							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0

COUNT[7:0] =	The low-byte of UART frame counter to track the number of valid frames received or transmitted. Counter saturates when both *DEBUG_UART_VALID_HI/LO* is 0xFF. This register is latched and the related counter is reset when *DEBUG_UART_DISCARD* is read.

8.5.4.13.17 DEBUG_COMH_VALID_HI/LO

DEBUG_COMH_VALID_HI

Address	0x078E							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							

Reset	0	0	0	0	0	0	0	0
	COUNT[7:0] =	The high-byte of COMH frame counter to track the number of valid frames received or transmitted. Counter saturates when both *DEBUG_COMH_VALID_HI/LO* is 0xFF. This register is latched and the related counter is reset when *DEBUG_COMH_DISCARD* is read.						

DEBUG_COMH_VALID_LO

Address	**0x078F**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0
	COUNT[7:0] =	The low-byte of COMH frame counter to track the number of valid frames received or transmitted. Counter saturates when both *DEBUG_COMH_VALID_HI/LO* is 0xFF. This register is latched and the related counter is reset when *DEBUG_COMH_DISCARD* is read.						

8.5.4.13.18 DEBUG_COML_VALID_HI/LO

DEBUG_COML_VALID_HI

Address	**0x0790**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0
	COUNT[7:0] =	The high-byte of COML frame counter to track the number of valid frames received or transmitted. Counter saturates when both *DEBUG_COML_VALID_HI/LO* is 0xFF. This register is latched and the related counter is reset when *DEBUG_COML_DISCARD* is read.						

DEBUG_COML_VALID_LO

Address	**0x0791**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	COUNT[7:0]							
Reset	0	0	0	0	0	0	0	0
	COUNT[7:0] =	The low-byte of COML frame counter to track the number of valid frames received or transmitted. Counter saturates when both *DEBUG_COML_VALID_HI/LO* is 0xFF. This register is latched and the related counter is reset when *DEBUG_COML_DISCARD* is read.						

8.5.4.13.19 DEBUG_OTP_SEC_BLK

Address	**0x07A0**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	BLOCK[7:0]							
Reset	0	0	0	0	0	0	0	0
	BLOCK[7:0] =	Holds last OTP block address where SEC occurred. Valid only when *FAULT_OTP[SEC_DET]* = 1.						

8.5.4.13.20 DEBUG_OTP_DED_BLK

Address	**0x07A1**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	BLOCK[7:0]							
Reset	0	0	0	0	0	0	0	0

BLOCK[7:0] =	Holds last OTP block address where DED occurred. Valid only when *FAULT_OTP[DED_DET]* = 1.

8.5.4.14 OTP Programming Control and Status

8.5.4.14.1 OTP_PROG_UNLOCK1A through OTP_PROG_UNLOCK1D

Address	**0x0300 to 0x0303**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CODE[7:0]							
Reset	0	0	0	0	0	0	0	0
CODE[7:0] =	The first 32-bit OTP programming unlock code is required as part of the OTP programming unlock sequence before performing OTP programming. This 32-bit code is entered in the sequence from *OTP_PROG_UNLOCK1A* to *OTP_PROG_UNLOCK1D*. These registers always read back 0.							

8.5.4.14.2 OTP_PROG_UNLOCK2A through OTP_PROG_UNLOCK2D

Address	**0x0352 to 0x0355**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	CODE[7:0]							
Reset	0	0	0	0	0	0	0	0
CODE[7:0] =	The second 32-bit OTP programming unlock code, required as part of the OTP programming unlock sequence before performing OTP programming. This 32-bit code is entered in the sequence from *OTP_PROG_UNLOCK2A* to *OTP_PROG_UNLOCK2D*. These registers always read back 0.							

8.5.4.14.3 OTP_PROG_CTRL

Address	**0x030B**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD						PAGESEL	PROG_GO
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
PAGESEL =	Selects which customer OTP page to be programmed. 0 = page 1 1 = page 2							
PROG_GO =	Enables programming for the OTP page selected by *OTP_PROG_CTRL[PAGESEL]*. Requires *OTP_PROG_UNLOCK1** and *OTP_PROG_UNLOCK2** registers are set to the correct codes. 0 = Ready 1 = Start OTP programming							

8.5.4.14.4 OTP_ECC_TEST

Address	**0x034C**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	RSVD				DED_SEC	MANUAL_ AUTO	ENC_DEC	ENABLE
Reset	0	0	0	0	0	0	0	0
RSVD =	Reserved							
DED_SEC =	Sets the decoder function (SEC or DED) to test. This bit is ignored during encoder testing. 0 = Test SEC functionality. Sets the *FAULT_OTP[SEC_DETECT]* flag and outputs test result to *OTP_ECC_DATAOUT** registers. 1 = Test DED functionality. Sets the *FAULT_OTP[DED_DETECT]* flag and outputs test result *OTP_ECC_DATAOUT**. Note: If SEC or DEC fault is detected, host sets *[RST_OTP_DATA]* = 1 to reset the corresponding fault. Switch to run SEC test does not clear DEC fault or vice versa.							

MANUAL_AUTO =	Sets the location of the data to use for the ECC test. 0 = Auto mode. Use the internal data for test. 1 = Manual mode. Uses data in *ECC_DATAIN_n* registers for test. Use for MPF test.
ENC_DEC =	Sets the encoder/decoder test to run when *OTP_ECC_TEST[ENABLE]* = 1. 0 = Run decoder test 1 = Run encoder test
ENABLE =	Executes the OTP ECC test configured by *[ENC_DEC]* and *[DED_SEC]* bits. 0 = Normal operation, ECC test disabled 1 = Initiate test

8.5.4.14.5 OTP_ECC_DATAIN1 through OTP_ECC_DATAIN9

Address	**0x0343 to 0x034B**							
RW	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	DATA[7:0]							
Reset	0	0	0	0	0	0	0	0
DATA[7:0] =	When ECC is enabled in manual mode, *CUST_ECC_TEST[MANUAL_AUTO]* = 1, *OTP_ECC_DATAIN1…9* registers are used to test the ECC encoder/decoder. If *CUST_ECC_TEST[ENC_DEC]* = 1, *ECC_DATAIN8* through *ECC_DATAIN1* are fed to the encoder. If *CUST_ECC_TEST[ENC_DEC]* = 0, *ECC_DATAIN9* through *ECC_DATAIN1* are fed to the decoder. The ECC_DATAOUT0…8 bytes must be read back to verify functionality.							

8.5.4.14.6 OTP_ECC_DATAOUT1 through OTP_ECC_DATAOUT9

Address	**0x0510 to 0x0518**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	DATA[7:0]							
Reset	0	0	0	0	0	0	0	0
DATA[7:0] =	*OTP_ECC_DATAOUT** bytes output the results of the ECC decoder and encoder tests. If *CUST_ECC_TEST[ENC_DEC]* = 0, *ECC_DATAOUT8* through *ECC_DATAOUT1* are read to determine a successful decoder test. If *CUST_ECC_TEST[ENC_DEC]* = 1, *ECC_DATAOUT9* through *ECC_DATAOUT1* are read to determine a successful encoder test. The correct result depends on the input to the test.							

8.5.4.14.7 OTP_PROG_STAT

Address	**0x0519**							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	UNLOCK	OTERR	UVERR	OVERR	SUVERR	SOVERR	PROGERR	DONE
Reset	0	0	0	0	0	0	0	0
UNLOCK =	Indicates the OTP programming function unlock status. After this bit is set (that is, OTP programming is enabled), the host writes to the *OTP_PROG_CTRL* register to start the OTP programming. Writing to any other register relocks the OTP programming function and clears this bit to 0. *[PROG_GO]* = 1 also clears this bit to 0. 0 = OTP programming locked 1 = OTP programming is unlocked							
OTERR =	Indicates the die temperature is greater than T_{OTP_PROG} and device does not start OTP programming. 0 = No fault 1 = Detected die temperature is greater than T_{OTP_PROG}. Abort OTP programming.							
UVERR =	Indicates an undervoltage error detected on the programming voltage during OTP programming. This bit is cleared with *[PROG_GO]* = 1. 0 = No error 1 = UV error detected							
OVERR =	Indicates an overvoltage error detected on the programming voltage during OTP programming. This bit is cleared with *[PROG_GO]* = 1. Information received from the device with this error must not be considered reliable. 0 = No error 1 = OV error detected							

SUVERR =	A programming voltage stability test is performed before starting the actual OTP programming. This bit indicates an undervoltage error is detected during the voltage stability test. This bit is cleared with *[PROG_GO]* = 1. 0 = No error 1 = UV error detected during OTP programming voltage stability test
SOVERR =	A programming voltage stability test is performed before starting the actual OTP programming. This bit indicates an overvoltage error is detected during the voltage stability test. This bit is cleared with *[PROG_GO]* = 1. 0 = No error 1 = OV error detected during OTP programming voltage stability test
PROGERR =	Indicates when an error is detected due to incorrect page setting caused by any of the following: • Trying to program but OTP programming *[UNLOCK]* = 0. • Trying to program a page that has *[TRY]* = 1. • Trying to program a page which has *[FMTERR]* = 1. This bit is cleared with *[PROG_GO]* = 1. 0 = No error or programming not attempted 1 = Error detected
DONE =	Indicates the status of the OTP programming for the selected page. This bit is cleared with *[PROG_GO]* = 1. 0 = Not completed or programming not attempted 1 = Complete.

8.5.4.14.8 OTP_CUST1_STAT

Address	0x051A							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
Name	LOADED	LOADWRN	LOADERR	FMTERR	PROGOK	UVOK	OVOK	TRY
Reset	0	0	0	0	0	0	0	0

LOADED =	Indicates OTP page 1 has been selected for loading into the related registers. See *[LOADERR]* and *[LOADWRN]* for error and warning status. 0 = Not selected for loading 1 = Page 1 selected and loaded
LOADWRN =	Indicates OTP page 1 was loaded but with one or more SEC warnings. 0 = No warning, or no load attempted 1 = Warning
LOADERR =	Indicates an error while attempting to load OTP page 1; that is, DED is detected while loading the selected page. 0 = No error, or no load was attempted. 1 = Error detected
FMTERR =	Indicates a formatting error in OTP page 1; that is, when *[UVOK]* or *[OVOK]* is set, but *[TRY]* = 0. Do not program if this bit is set. 0 = No error 1 = Error detected
PROGOK =	Indicates the validity for loading for OTP page 1. A valid page indicates that successful programming occurred. 0 = Not valid 1 = Valid
UVOK =	Indicates an OTP programming voltage undervoltage condition is detected during programming attempt for OTP page 1. The OV condition may also trigger the UV as part of the shutdown process. 0 = UV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No UV condition detected
OVOK =	Indicates an OTP programming voltage overvoltage condition is detected during programming attempt for OTP page 1. The OV condition will trigger the UV as part of the shutdown process. The device must be taken out of service. 0 = OV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No OV condition detected
TRY =	Indicates a first programming attempt for OTP page 1. 0 = No first attempt made 1 = First attempt made

8.5.4.14.9 OTP_CUST2_STAT

Address	0x051B							
Read Only	**Bit 7**	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**

Name	LOADED	LOADWRN	LOADERR	FMTERR	PROGOK	UVOK	OVOK	TRY
Reset	0	0	0	0	0	0	0	0

LOADED =	Indicates OTP page 2 has been selected for loading into the related registers. See *[LOADERR]* and *[LOADWRN]* for error and warning status. 0 = Not selected for loading 1 = Page 2 selected and loaded
LOADWRN =	Indicates OTP page 2 was loaded but with one or more SEC warnings. 0 = No warning, or no load attempted 1 = Warning
LOADERR =	Indicates an error while attempting to load OTP page 2; that is, DED is detected while loading the selected page. 0 = No error, or no load was attempted. 1 = Error detected
FMTERR =	Indicates a formatting error in OTP page 2; that is, when *[UVOK]* or *[OVOK]* is set, but *[TRY]* = 0. Do not program if this bit is set. 0 = No error 1 = Error detected
PROGOK =	Indicates the validity for loading for OTP page 2. A valid page indicates that successful programming occurred. 0 = Not valid 1 = Valid
UVOK =	Indicates an OTP programming voltage undervoltage condition is detected during programming attempt for OTP page 2. The OV condition may also trigger the UV as part of the shutdown process. 0 = UV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No UV condition detected
OVOK =	Indicates an OTP programming voltage overvoltage condition is detected during programming attempt for OTP page 2. The OV condition will trigger the UV as part of the shutdown process. The device must be taken out of service. 0 = OV condition detected. Also reads as 0 if no programming attempt is performed. 1 = No OV condition detected
TRY =	Indicates a first programming attempt for OTP page 2. 0 = No first attempt made 1 = First attempt made

9 Application and Implementation

Note

Information in the following applications sections is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes, as well as validating and testing their design implementation to confirm system functionality.

9.1 Application Information

The BQ79631-Q1 device family provides high-accuracy, voltage, current, and temperature monitoring for HV EV/BMS systems.

9.2 Typical Applications

9.2.1 Base Device Application Circuit

The following application circuit (see Figure 9-1) is based on the BQ79631-Q1 device operating in a BJB/BDU.

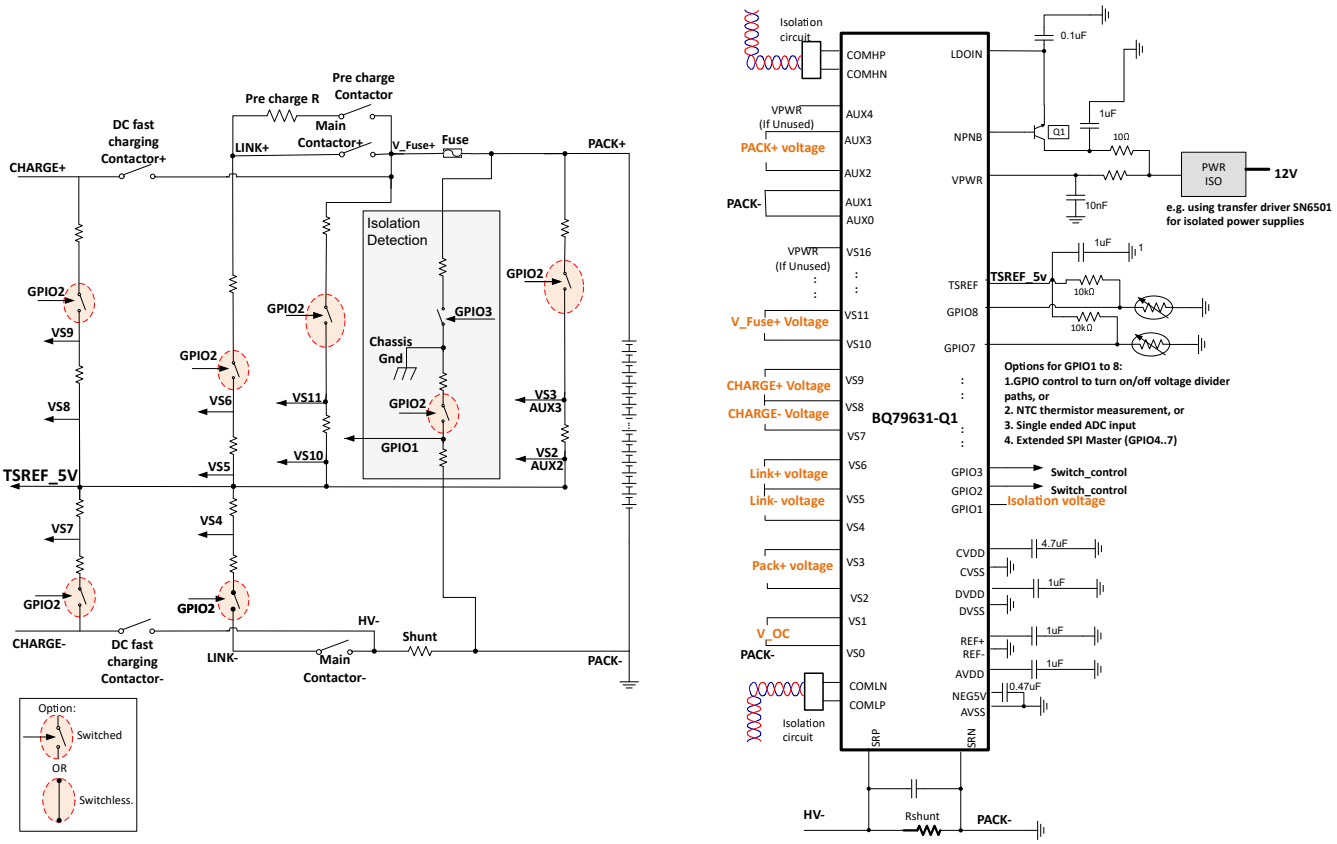


Figure 9-1. Typical BJB Application Circuit and Connections for Voltage (U), Current (I), and Insulation Resistance (R)

A typical BJB application circuit and connections for voltage (U), current (I), and insulation resistance (R) are shown in Figure 9-1

All the VS and AUX channels measure differential voltages. The GPIO, configured as ADC, measure single-ended inputs refered to AVSS. The GPIO also provides ratiometric measurement with respect to TSREF in order to measure thermistor temperature sensors.

9.2.1.1 Design Requirements

Table 9-1 below shows the design parameters.

Table 9-1. Recommended Design Requirements

PARAMETER	VALUE
VPWR voltage range	9 V to 15 V
Differential VSn to VSn-1 input voltage range	-1 V to 5 V
GPIO input voltage range	0.2 V to 4.8 V
Current sense input voltage range	-100 mV to +100 mV

9.2.1.2 Detailed Design Procedure

- Recommend to provide >1-mA current on the resistor divider ladder to allow faster charging time on the VS pin filter capacitor.
- Appropriate resistor dividers are needed to bring the high voltage signals within the dynamic range of the ADC.
- The resistors in the ladder can be comprised of several resistors in order to distribute the power dissipation. The portion of the divider below the ADC input can be made up of parallel resistors to ensure that the ADC input does not fly to a high voltage when any one resistor becomes open.
- Ensure that the voltages on AUX4 >= V_AUX3 >= V_AUX2 >= V_AUX1 >= V_AUX0.
- Ensure the input voltage from VS3 and higher to be ≥ 3 V. VS1 to VS2 input can be <3 V, but must be >0 V with respected to device ground.
- Short unused VS pins to VPWR as shown in Figure 9-7.
- Short unused NC pins between 2 and 24 (including 2, 24) to VPWR as shown in Figure 9-8.
- The highest VS voltage input and VPWR voltage should maintain the following relationship for best voltage measurement accuracy:
  - VPWR ≥ (1/2 the highest VS voltage + 2 V)
  - If the min VPWR is 9 V in the application, then when VPWR = 9 V, the highest VS voltage input shall ≤ (9 V -2) x 2 = 14 V
  - If using MOSFET switches , ensure that the highest measurement voltage is lower than (Vmosfet_gate – Vgs_mosfet).

9.2.1.2.1 PACK+ and FUSE Measurement

Table 9-2. PACK+ and FUSE Measurement

Pins	Components	Value	Description
VS2, VS3, VS10, VS11, AUX2, AUX3	Resistor string Top	1.32 MΩ	Resistor divider to divide down HV measurement to ADC input range. Component values are recommended ones. Other values can be chosen as long as signal is kept within ADC input range. Too large of value resistors will cause more measurement inaccuracy due to leakage at ADC pins.
	Resistor string Bot	12.45 kΩ	
	Optional MOSFET or Switch	1000-V standoff	Switch to block unwanted leakage current

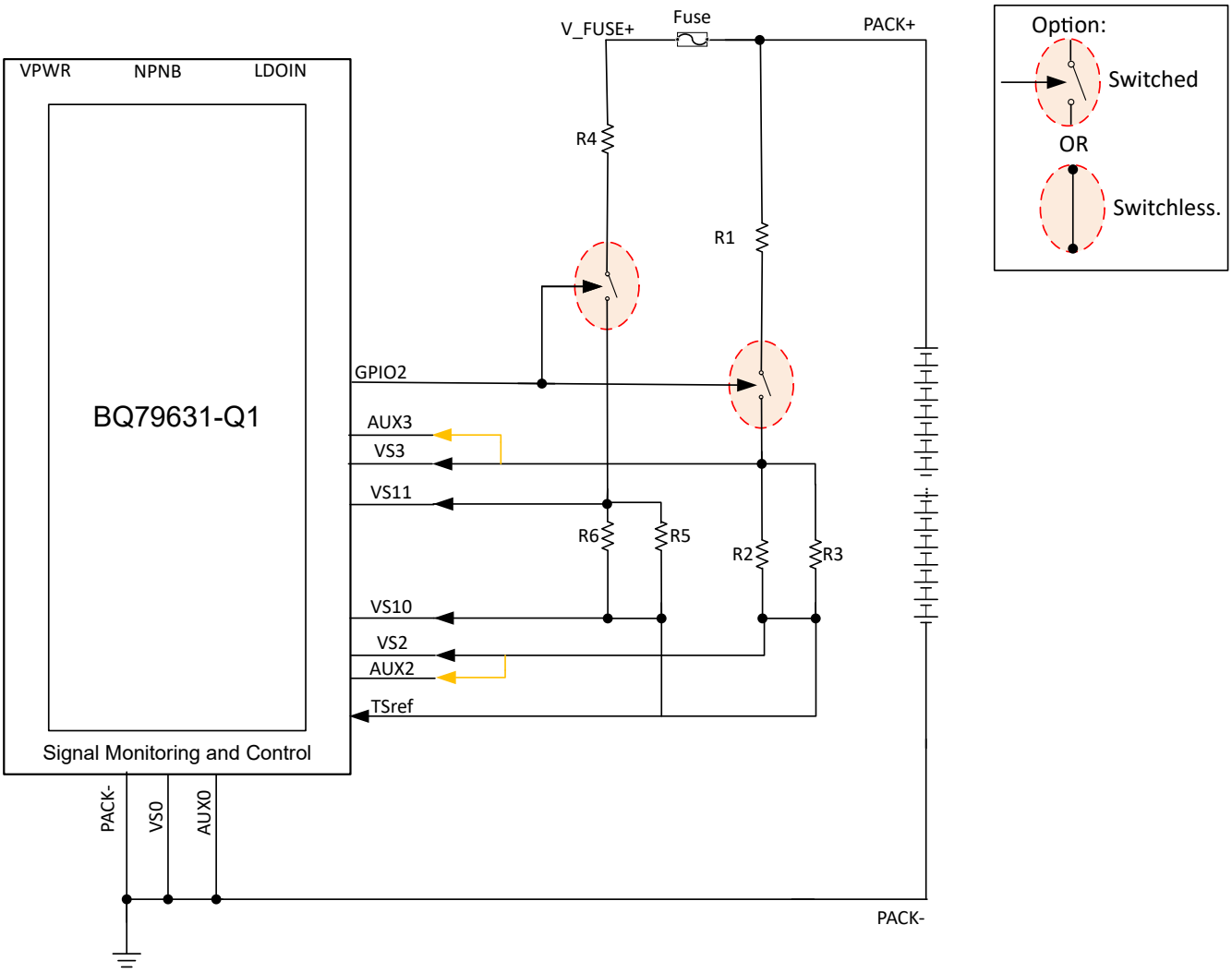


Figure 9-2. Pack+ and V_Fuse+ Measurements

Pack+ and V_Fuse+ measurements are shown in Figure 9-2.

For Pack+ and V_Fuse+ calculation, the measurements from Pack+ reading on VS3 and V_Fuse+ reading on VS11 need to be gained up by dividing gain factors G_{pack} and G_{fuse} respectively.

$$\text{Gain: } G_{pack} = \frac{R2//R3}{(R2//R3)+R1} \; ; \; G_{fuse} = \frac{R6//R5}{(R6//R5)+R4}$$

Aux channels can be used to provide redundant measurement to the VS channels. AUX channels could be used for independent measurements as well. When used for redundant measurement, there is an option to run diagnostics to compare AUX channel to its equally numbered VS counterpart.

Switches in the path are optional and are mainly used to minimize leakage current from the battery pack when measurement is not being performed. The BQ79631-Q1's integrated digital low-pass filters provide the necessary filtering for the input signals.

When the optional switches are used and these switches are MOSFET switches, care must be taken to ensure the maximum signal to the ADC input is always lower than $V_{MOSFET\_GATE} - V_{MOSFET\_VGS}$.

9.2.1.2.2 Insulation Measurement

Table 9-3. Insulation Measurement

Pins	Components	Value	Description
GPIO1, GPIO2, GPIO3	Resistor (R1)	1.32 MΩ	Known intentional resistor used to switch in and out of the circuit in order to detemine the unknown values of insulation resistances.
	Resistor string Top (R2)	1.32 MΩ	Resistor divider to divide down HV measurement to ADC input range. Component values are recommended ones. Other values can be chosen as long as signal is kept within ADC input range. Too large of value resistors will cause more measurement inaccuracy due to leakage at ADC pins.
	Resistor string Bot (R3//R4)	12.45 kΩ	
	SW1	TPSI2140-Q1	HV switch used to connect and disconnect R1 in the circuit.
	Optional MOSFET or Switch (SW2)	1000-V standoff	Switch to block unwanted leakage current.

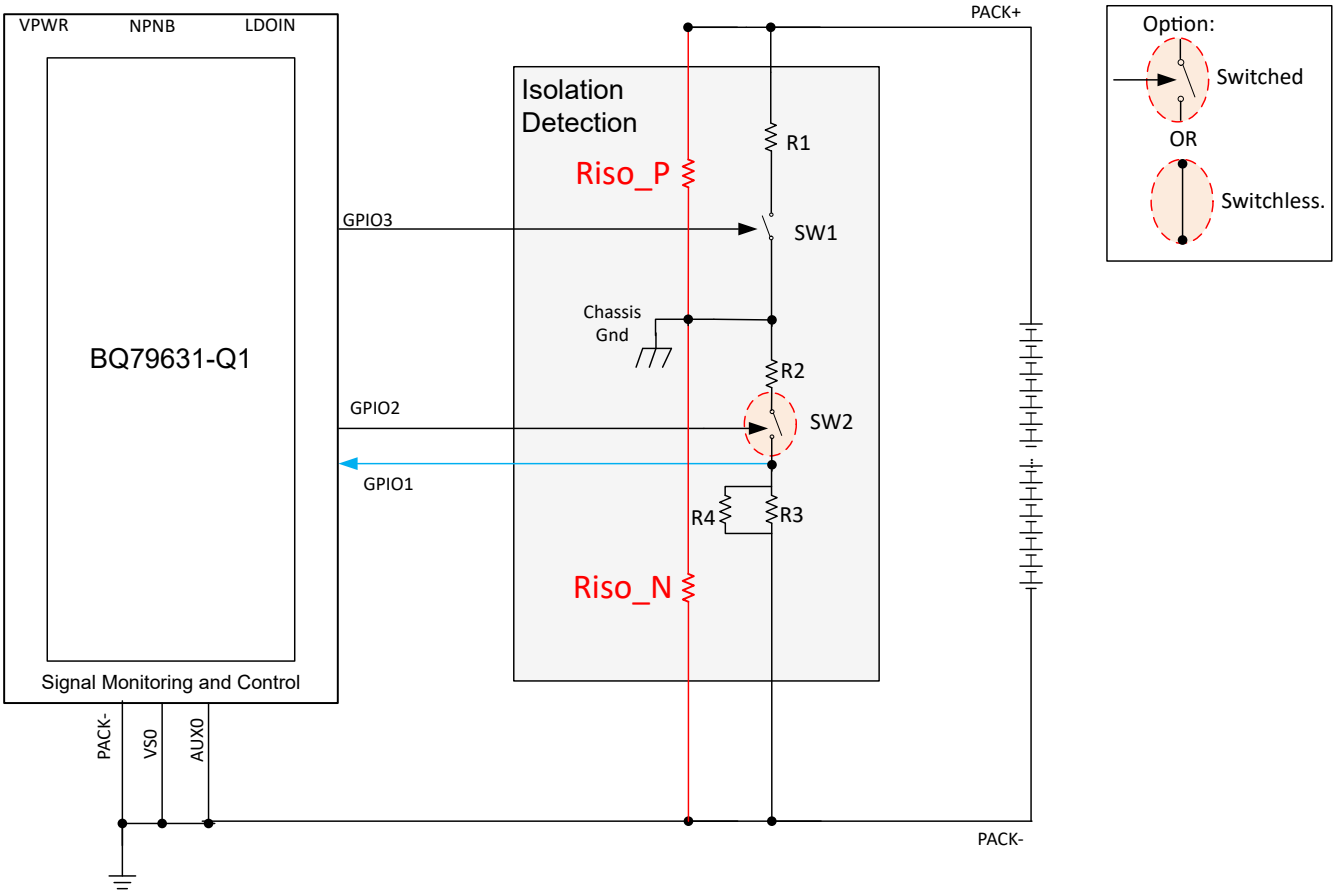


Figure 9-3. Insulation Detection Scheme

Insulation detection scheme is shown in Figure 9-3

Riso_P and Riso_N are the Insulation resistances in the system that are to be determined.

Riso_P is the resistance between PACK+ and chassis gnd. Riso_N is the resistance btweeen PACK- and chassis gnd.

A low value of Riso_P or Riso_N could provide a low impedance from the HV battery to the LV chassis gnd where sensitive electronics and/or operator could be endangered. Hence it is important to detect a low insulation resistace.

Riso_P and Riso_N are two unknown variables that need to be determined. A single measurement on GPIO1 can give the relation or ratio between Riso_P and Riso_N , but will not be able to determine their absolute resistance values. For this a known reference resistor is needed. R1 in the Figure 9-3 serves that purpose. To solve for the two unknowns Riso_P and Riso_N, two equations are needed. One equation is achieved by connecting R1 in parallel to Riso_P. The other equation is achieved by disconnecting R1 from Riso_P.

The procedure to determine the insulation resistances is as follows:

Step 1: SW_GPIO3 open, measure GPIO1. Get equation1 by deriving the output voltage expression for this configuration of switch.

Step 2: SW_GPIO3 close, measure GPIO1. Get equation2 by deriving the output voltage expression for this configuration of switch.

Step 3: Combine equation1 and 2 solve Riso_P and Riso_N.

9.2.1.2.3 LINK+/- Measurement

Table 9-4. LINK+/- Measurement

Pins	Components	Value	Description
V5, V6, TSREF	Resistor string Top	1.32 MΩ	Resistor divider to divide down HV+ measurement to ADC input range. Component values are recommended ones. Other values can be chosen as long as signal is kept within ADC input range. Too large of value resistors will cause more measurement inaccuracy due to leakage at ADC pins.
	Resistor string Bot	12.45 kΩ	
V4, V6, TSREF	Resistor string Top	5 kΩ	Resistor divider to divide down HV- measurement to ADC input range. Component values are recommended ones. Other values can be chosen as long as signal is kept within ADC input range. Too large of value resistors will cause more measurement inaccuracy due to leakage at ADC pins.
	Resistor string Bot	10 kΩ	
GPIO2	Optional MOSFET or Switch (SW1 and SW2)	1000-V standoff	Switch to block unwanted leakage current.

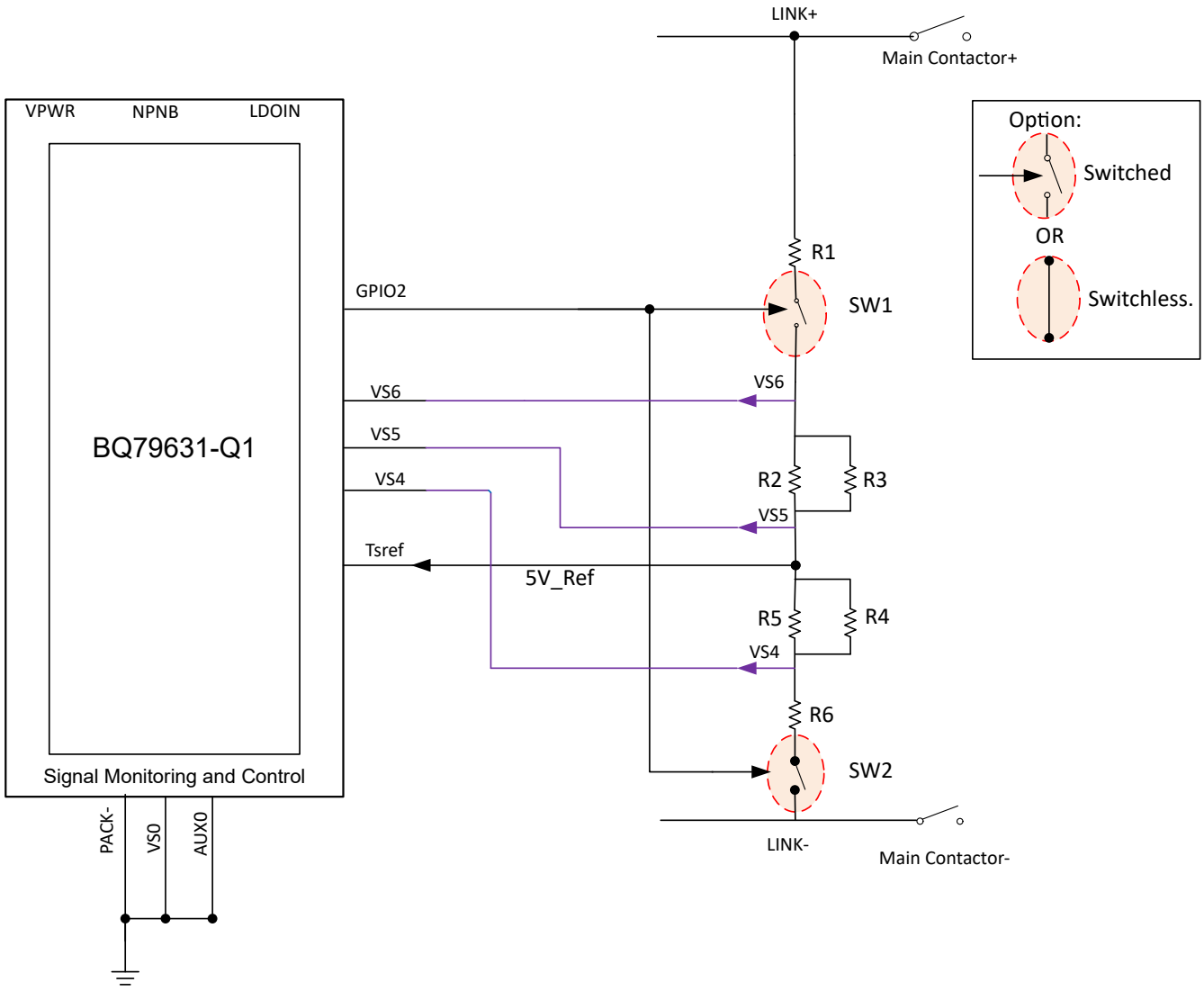


Figure 9-4. LINK+/– Measurement

LINK+/– Measurement is shown in Figure 9-4.

Differential ADC inputs V6-V5 measure the Link+ voltage. The Link+ voltage can be determined from the output of the ADC by using:

$$V\_LINK+ = VC6(ADC)/G_{Link+} + V\_TSREF$$

$$\text{Where: } G_{Link+} = \frac{R2//R3}{(R2//R3)+R1}$$

Differential ADC inputs V5-V4 measure the Link– voltage. The Link– voltage can be determined from the output of the ADC by using:

$$V\_LINK- = V\_TSREF - VC5(ADC)/G_{Link-}$$

$$\text{Where: } G_{Link-} = \frac{(R4//R5)}{(R4//R5)+R6}$$

Resistors R2 is in parallel with R3, R5 is in parallel with R4 to ensure that VS6 and VS4 are not disconnected from the resistor string in case of failure in any one of these resistors. VS6 and VS4 could exceed abs max if they get disconnected from the resistor string.

9.2.1.2.4 CHARGE+/– Measurement

Table 9-5. CHARGE+/- Measurement

Pins	Components	Value	Description
V9, V8, TSREF	Resistor string Top	1.32 MΩ	Resistor divider to divide down HV+ measurement to ADC input range. Component values are recommended ones. Other values can be chosen as long as signal is kept within ADC input range. Too large of value resistors will cause more measurement inaccuracy due to leakage at ADC pins.
	Resistor string Bot	12.45 kΩ	
V8, V7, TSREF	Resistor string Top	5 kΩ	Resistor divider to divide down HV- measurement to ADC input range. Component values are recommended ones. Other values can be chosen as long as signal is kept within ADC input range. Too large of value resistors will cause more measurement inaccuracy due to leakage at ADC pins.
	Resistor string Bot	10 kΩ	
GPIO2	Optional MOSFET or Switch (SW1 and SW2)	1000-V standoff	Switch to block unwanted leakage current.

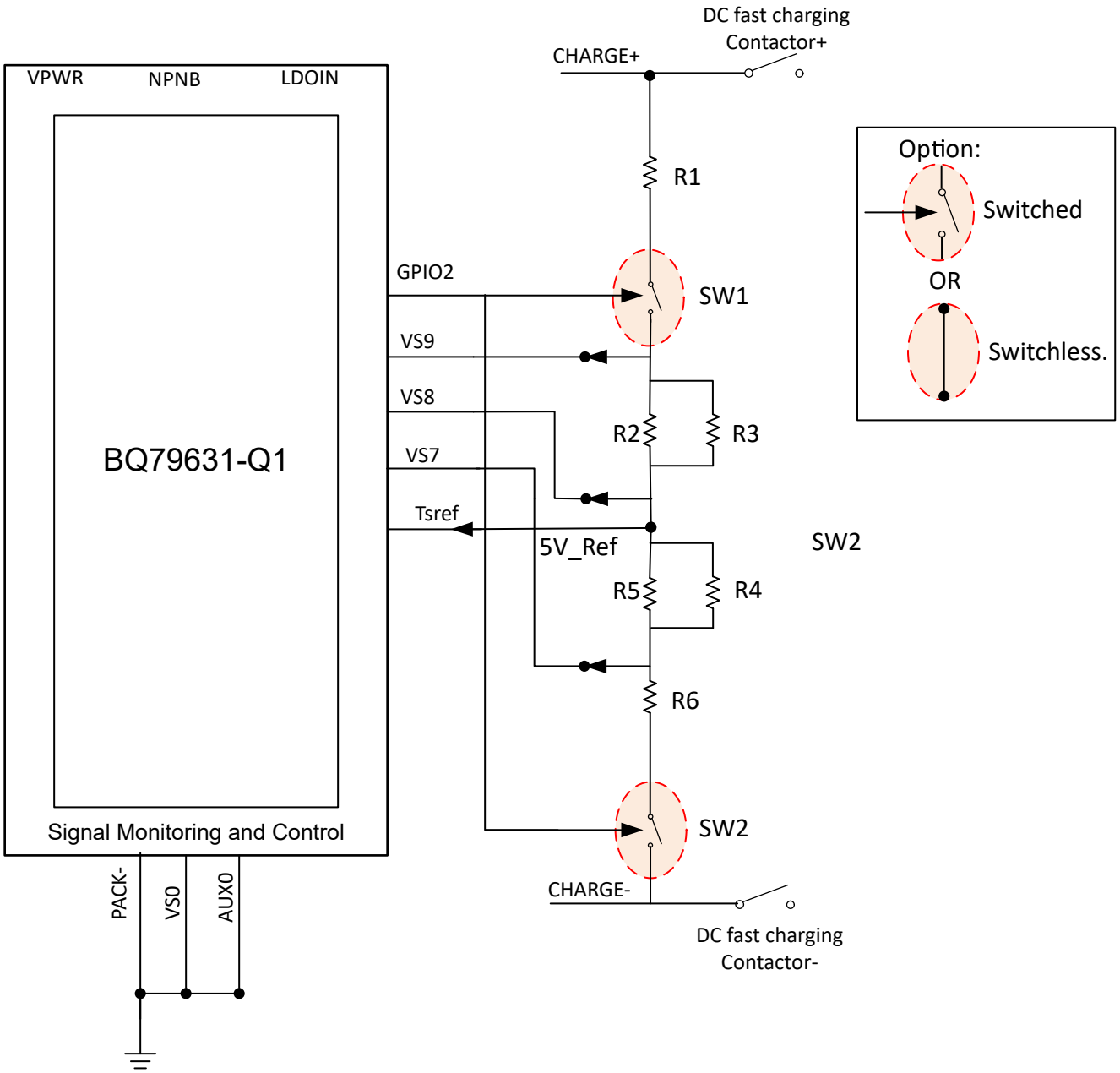


Figure 9-5. CHARGE +/– Measurement

CHARGE +/- Measurement is shown in Figure 9-5.

Differential ADC inputs V9-V8 measure the CHARGE+ voltage. The CHARGE+ voltage can be determined from the output of the ADC by using:

$$V\_CHARGE_{+} = VC9(ADC)/G_{Charge+} + V\_TSREF$$

$$\text{Where: } G_{Charge+} = \frac{R2//R3}{(R2//R3)+R1}$$

Differential ADC inputs V8-V7 measure the CHARGE– voltage. The CHARGE– voltage can be determined from the output of the ADC by using:

$$V\_CHARGE\text{-} = V\_TSREF - VC8(ADC)/G_{Charge-}$$
$$\text{Where: } G_{Charge-} = \frac{(R4//R5)}{(R4//R5)+R6}$$

Resistors R2 is in parallel with R3, R5 is in parallel with R4 to ensure that VS9 and VS7 are not disconnected from the resistor string in case of failure in any one of these resistors. VS9 and VS7 could exceed abs max if they get disconnected from the resistor string.

9.2.1.2.5 Overcurrent Detection Scheme Using OVUV Comparators

Table 9-6. Overcurrent Detection Scheme Using OVUV Comparators

Pins	Components	Value	Description
VS1, TSREF, SRP, SRN	R1, R2, R4, R5	10 kΩ	Resistors to provide a gain of 10x as well as set the output common mode voltage of V_Tsref/2 on the ADC input pin.
	R3, R6	100 kΩ	
	Amplifier	LM2904QDRQ1	General purpose op-amplifier.

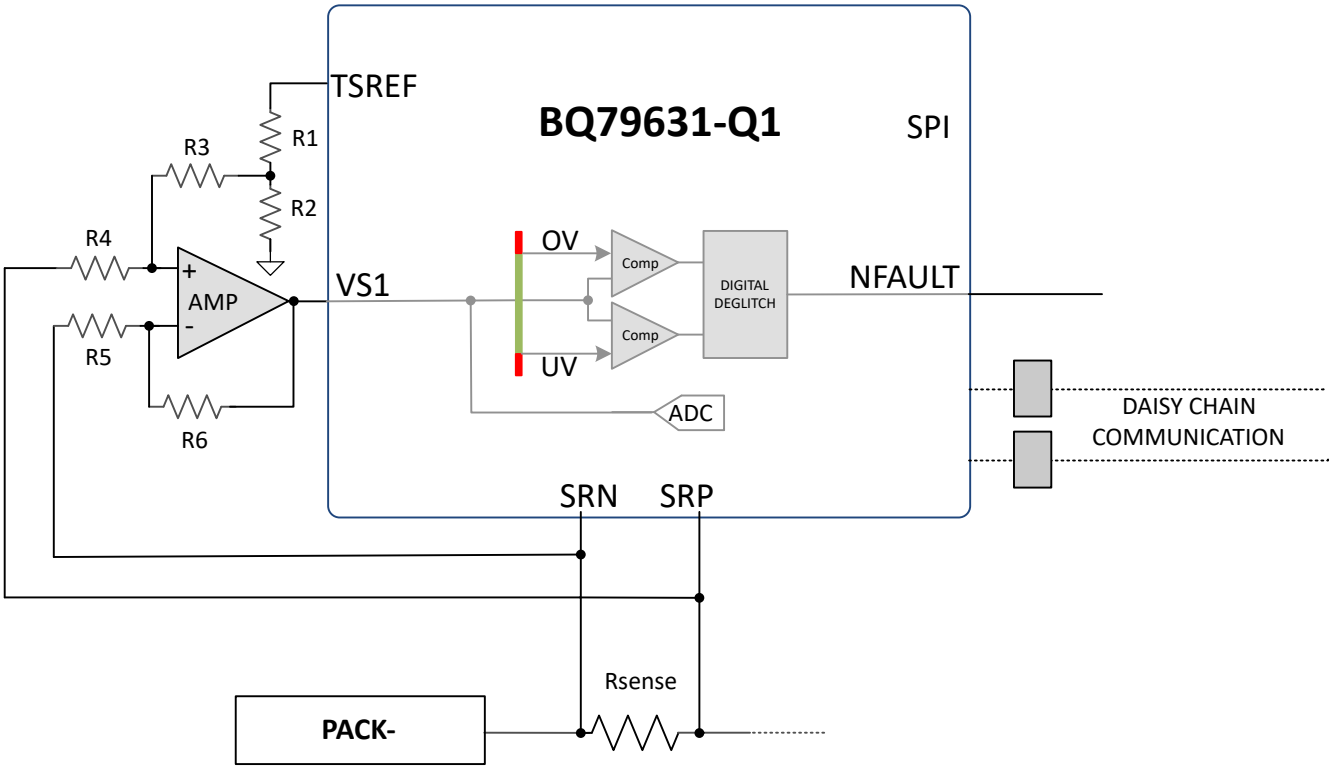


Figure 9-6. A Scheme for Overcurrent Measurement

A scheme for overcurrent measurement is shown in Figure 9-6.

Often overcurrent detects have large ranges that exceed the dynamic range of the main CSADC.

There are two options for overcurrent detect in the BQ79631-Q1. One option is to use the internal CSAUX ADC that has a wider dynamic range. The MCU then processes the data and compares against an OC threshold.

The other option is to use the OVUV comparator of the VS input. An external amplifier can be used to gain up the voltage across the RShunt and feed this to the OVUV comparator input. The comparator thresholds can be preset. A positive overcurrent event will trip the OV comparator and a negative overcurrent will trip the UV comparator. When either of the comparators trip, a fault is flagged on the nFault pin and in the communication to MCU. These hardware comparators are actively monitoring even when the device is in sleep mode.

As an example of the above scheme, the amplifier gain is set to 10V/V. In order to set an overcurrent fault when the Rsense voltage reaches 50 mV in either the positive or negative direction, an OV threshold of 3000 mV and UV threshold of 2000 mV is programmed. The amplifier gains the 50-mV overcurrent signal to 500 mV at the VS1 input. This signal at VS1 is on either side of the common mode of 2500 mV, which is set by the resistor divider from TSREF. Different values of the OV and UV thresholds can be programmed along with various amplifier gain and common-mode settings in order to achieve the desired overcurrent thresholds.

9.2.1.2.6 Unused Pins

Unused VS pins need to be tied to VPWR as shown in Figure 9-7.

Unused AUX pins must be tied to VPWR in similar fashion.

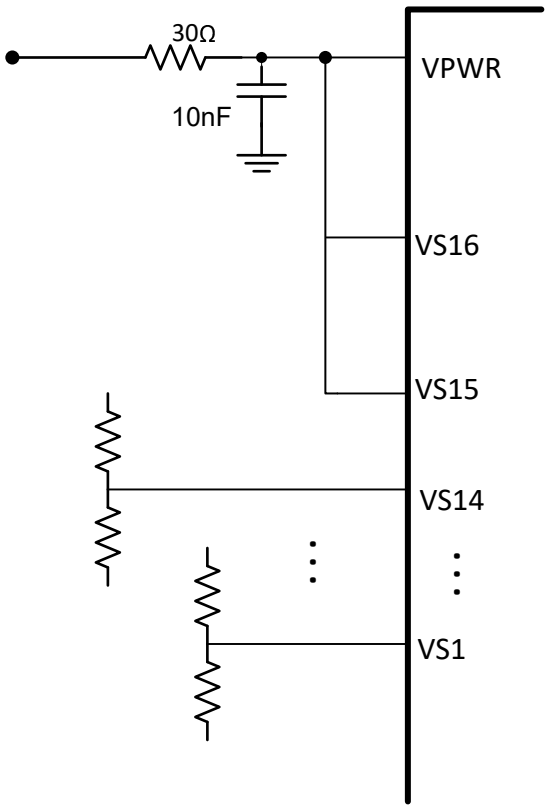


Figure 9-7. Unused Pins Tied to VPWR

All NC pins between pin 1 and Pin 25 need to be tied to VPWR as shown in Figure 9-8.

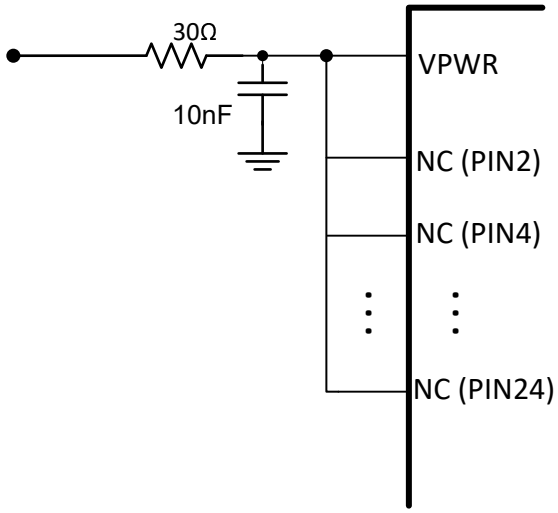


Figure 9-8. NC Pins Tied to VPWR

9.2.1.2.7 Current Sense Input

It is recommended to add an analog RC filter between the shut resistor and device current sensing pins (SPR/SRN). This filter helps remove high frequency components and improves the performance of the current sensing ADC.

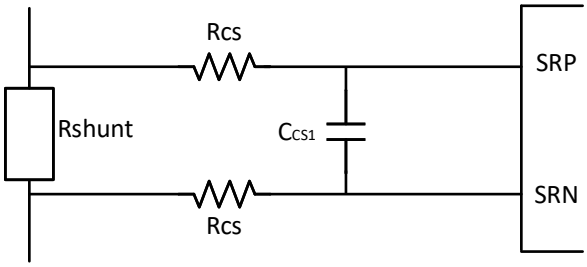


Figure 9-9. Current Sense Input Connection

Selection of the shut sensing resistor has many system level considerations, a few related to this device include:

- Maximum current multiplied with R-shunt should be within the Vcs_range absmax range.
- Maximum normal current multiplied with R-shunt should be within the Vcs_range recommended range.
- To fully utilize the current sense ADC input range thus better resolution, it is preferred to choose a higher value resistance, this is a tradeoff to thermal dissipation on the shunt resistor.
- User can perform room temperature two points calibration to trim out room temperature offset and gain error and store the coefficient in the device. Through this way, the residual error is offset drift and gain error drift.

Table 9-7. Current Sense Components

Related Pins	Components	Value	Description
SRP, SRN	Filter resistor Rcs	10 Ω	To avoid inducing a large gain error, the value of this resistor should be less than 10 Ω.
	Filter capacitor Ccs1	0.47 µF/16 V	Differential filtering capacitor, serves the purpose of filtering differential noise. It is recommended to use a value of not less than 1 µF.

It is not recommended to place a common-mode capacitor between ground and the SRP/SRN pins since this would couple ground noise to the input pin.

9.2.1.2.8 VPWR and External NPN

Table 9-8. VPWR and External NPN

Related Pins	Components	Value	Description
VPWR	Filter resistor	30 Ω	Single-ended RC filter, recommended values must be used for hot-plug performance.
	Filter capacitor	10 nF/20 V	
NPNB	NPN (Q1)	Collector–emitter breakdown voltage 15 V to 20 V, Power rating ≥ 1 W Gain > 80 at the expected load current Current handling >100 mA	The external NPN is used to form a pre-regulation circuit to provide a 6-V (typical) input to the LDOIN pin. The voltage rating of the NPN can be optimized by the following equation: **NPN voltage rating = Max VPWR – Min VLDOIN + Margin** Where: Max VPWR = supply of the device Min VLDOIN = minimum spec of the VLDOIN parameter Margin = system transient voltage + design margin per application requirement
	Resistor on external NPN collector (R_{NPN})	Various based on module voltage	The resistor has a couple of purposes: (a) For an RC filter for the NPN pre-regulation circuit (b) Share the thermal dissipation with the NPN
	Capacitor on external NPN collector	0.22 nF/100 V Can use lower voltage rating based on module size	The capacitor forms the RC filter for the NPN pre-regulation circuit The capacitor rating is based on the peak voltage spike seen on the module. For a smaller module size, <100-V rated capacitor can be used. System designer selects the optimized voltage-rated capacitor per their system tolerance and requirements.

To reduce the power rating needed for the external NPN (Q1), the system designer can put power resistors on the NPN collector to create IR drop from the VPWR. Figure 9-10 shows the current paths to power the BQ79631-Q1 device.

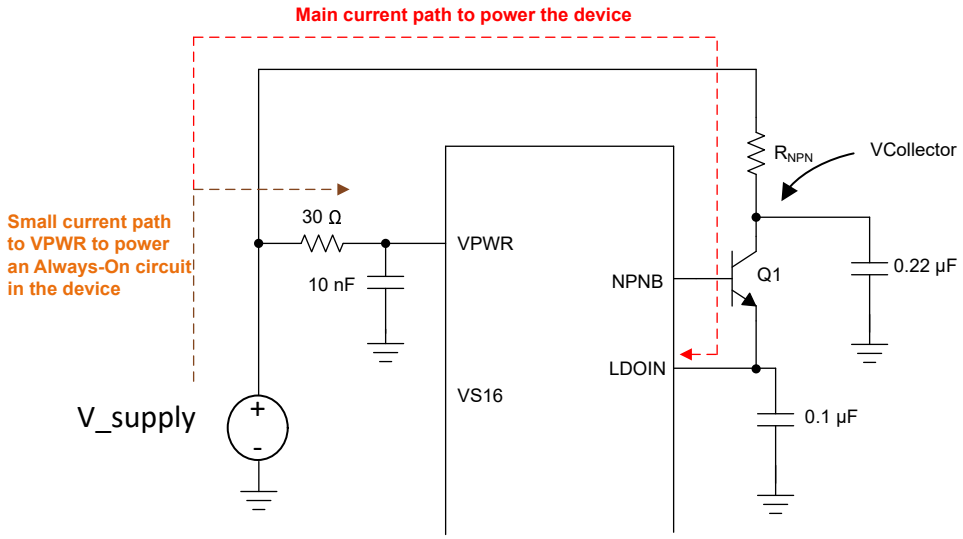


Figure 9-10. Power Consumption Paths

To ensure there is sufficient headroom to maintain a 6-V (typical) regulated voltage on the LDOIN pin, the system designer ensures VCollector has ≥ 8 V at any time with the assumption of about 2-V drop across the NPN.

Hence, maximum allowable R_{NPN} value = ((Min VPWR) – (VCollector)) / (Max peak current)

Where:

VCollector: 8 V with the assumption of about 2-V drop across NPN

Max peak current: highest operation current, which is the active current with all functions turned on. Note that different communication isolation components (for example, capacitor isolation versus transformer, or the type of transformer) contribute different loading to the total power consumption.

9.2.1.2.9 Power Supplies, Reference Input

Table 9-9. Power Supplies, Reference Input

Related Pins	Components	Value	Description
AVDD, DVDD, TSREF	Bypass capacitor	1 µF/10 V	Bypass capacitor for the internal LDOs
CVDD	Bypass capacitor	4.7 µF/10 V	Bypass capacitor for CVDD
NEG5V	Bypass capacitor	0.1 µF/10 V	Bypass capacitor for the negative charge pump

9.2.1.2.10 GPIO For Thermistor Inputs

When using an external thermistor, for ADC measurement only, there is no limitation of what type of thermistors (NTC or PTC) or bias resistor (R1) value or whether the thermistor is placed on high side or low side with respected to the bias resistor.

The device does not require external RC for temperature measurement. However, it is common for the system designer to add an RC filter on the GPIO pin for the thermistor circuit. The system designer can select RC values for application needs. Example: RGPIO = 1 kΩ, CGPIO = 0.1 µF to 1 µF

Unused GPIO must be grounded to AVSS with a 10-kΩ resistor.

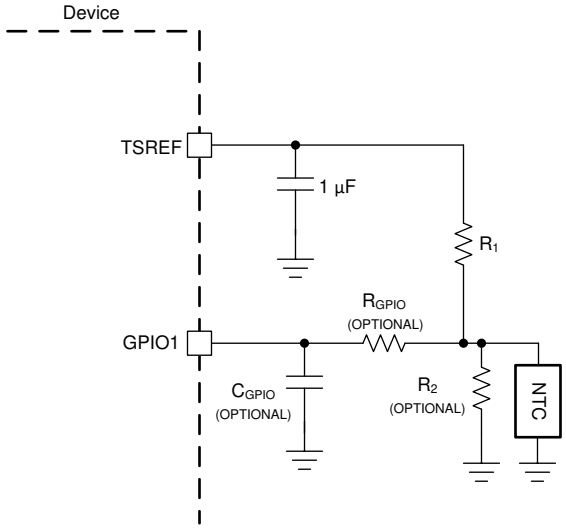


Figure 9-11. NTC Connection

9.2.1.2.11 UART, NFAULT

If the device is used as a base device, the UART interface requires the TX and RX pins are pulled up through a 10-kΩ to 100-kΩ resistor. Do not leave the TX and RX pins unconnected. The TX pin must be pulled high to prevent triggering an invalid communications frame during the idle state. When using a serial cable to connect to the host controller, connect the TX pullup on the host side and the RX pullup to the CVDD on the device side.

If the device is used as a stack device, the TX pin is disabled by default and is left floating. The RX pin is shorted to CVDD.

The NFAULT pin for a base device, if not used, must be left floating. Otherwise, pull it up with 100-kΩ to CVDD. The NFAULT pin on a stack device is floating.

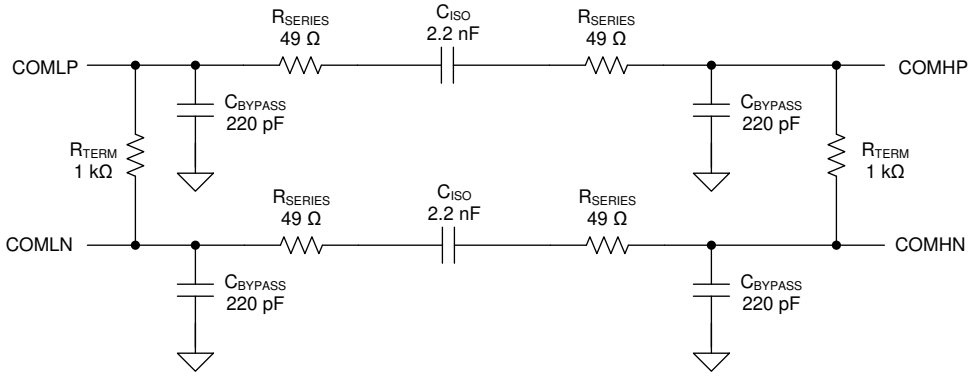
9.2.1.2.12 Daisy-Chain Isolation

The device works with multiple daisy-chain isolation types: capacitor isolation, capacitor-choke isolation, and transformer isolation. For devices that are daisy-chained on the same PCB, capacitor isolation without ESD diode as shown in Figure 9-12 is sufficient. Unused COMLP/H or COMHP/N pins must be connected with a 1-kΩ termination resistor.

9.2.1.2.12.1 Devices Connected on the Same PCB

Table 9-10. Isolation Components for Devices Connected on the Same PCB

Components	Value	Description (Capacitor Isolation on the Same PCB)
R_{TERM}	1 kΩ	Termination resistor
R_{SERIES}	49 Ω	Filter resistor and impedance matching resistor. The connection between devices must be ~120 Ω (~50 Ω on each end of the signal connection of the device plus 10-Ω internal resistance)
C_{BYPASS}	220 pF/50 V	Bypass capacitor
C_{ISO}	2.2 nF	Isolation capacitor Voltage rating depends on application requirement. It is common to select 2x of module voltage rating to provide standoff margin in the event of a fault in the system.



Components Required for Cap Coupled Daisy Chain on the same PCB

Figure 9-12. Capacitor Isolation with Devices on the Same PCB

9.2.1.2.12.2 Devices Connected on Different PCBs

For devices that are daisy-chained to different PCBs through a pair of twisted cables, all three isolation types can be used for daisy-chain isolation. It is also possible to use one type of isolation on one side of the daisy-chain (for example, transformer isolation on COMLP/N to the Battery Management Unit) while using a different type of isolation for the other side of the daisy-chain (for example, capacitor isolation on COMH/N to another BQ796xx Unit).

Option 1: Capacitor Isolation

Table 9-11. Components for Capacitor Isolation on Different PCBs

Components	Value	Description (Capacitor Isolation Between PCBs)
R_{TERM}	1 kΩ	Termination resistor
R_{SERIES}	49 Ω	Filter resistor and impedance matching resistor. The connection between devices must be ~120 Ω (~50 Ω on each end of the signal connection of the device plus 10-Ω internal resistance).
C_{BYPASS}	220 pF/50 V	Bypass capacitor
C_{ISO}	2.2 nF	Isolation capacitor Voltage rating depends on application requirement. It is common to select 2x of module voltage rating to provide standoff margin in the event of a fault in the system.

Table 9-11. Components for Capacitor Isolation on Different PCBs (continued)

Components	Value	Description (Capacitor Isolation Between PCBs)
ESD diode	TVS diode	The ESD protector should provide protection to the communication interface pins during hot-plug events and also for absorption of high-voltage transients during service disconnect or reconnect. Select the ESD diodes to limit the maximum voltage on the COM* bus to below the maximum rating. A voltage rating close to the maximum voltage to provide the highest possible common-mode voltage range is recommended for best EMC performance. The capacitance must be low compared to the coupling capacitance (if using capacitor coupling).

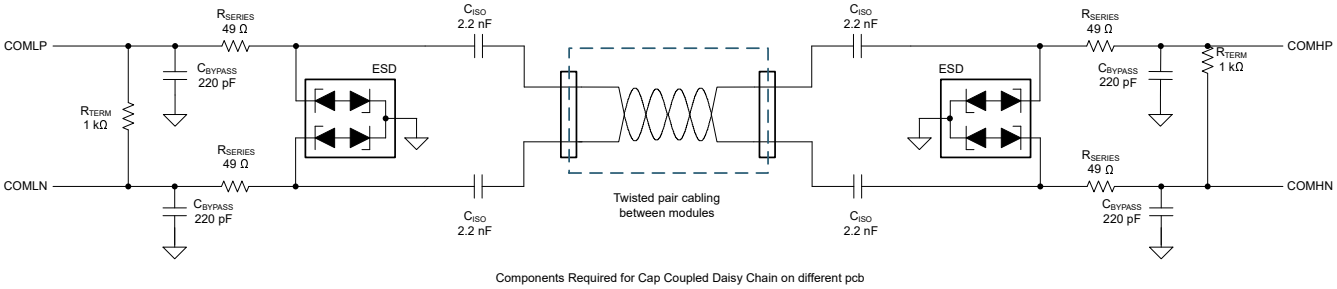


Figure 9-13. Capacitor Isolation on Different PCB

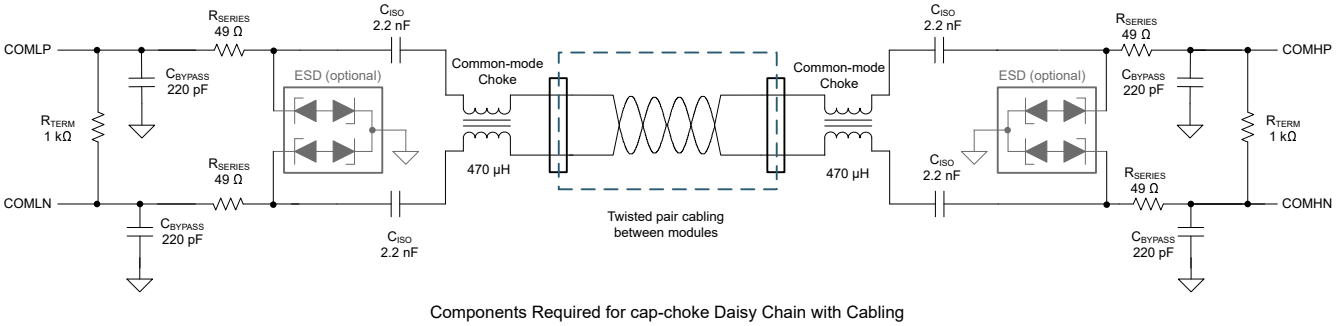
Figure 9-13 shows the capacitor isolation circuit for devices connecting between PCBs. Similar to the capacitor isolation on the same PCB case, the capacitor must be rated with a high enough voltage to provide standoff margin in the event of a fault in the system that exposes the device to a local hazardous voltage. The voltage is determined by the application requirement but it is common to select 2x of the module voltage.

The capacitance on the daisy-chain bus has a direct effect on performance. All parasitic capacitances from the support components and cabling must be taken into consideration when designing for communication robustness to EMC. Capacitance from the cables, ESD diodes, bypass capacitance, and chokes form a capacitive divider with the isolation capacitors that may affect performance. Additionally, the amount of capacitance on the bus has a direct impact to the operating current during communication (the capacitor charging or discharging).

Option 2: Capacitor Plus Common-Mode Choke Isolation

Table 9-12. Components for Capacitor Plus Common-Mode Choke Isolation

Components	Value	Description (Capacitor Plus Choke Isolation Between PCBs)
R_{TERM}	1 kΩ	Termination resistor
R_{SERIES}	49 Ω	Filter resistor and impedance matching resistor. The connection between devices must be ~120 Ω (~50 Ω on each end of the signal connection of the device plus 10-Ω internal resistance).
C_{BYPASS}	220 pF/50 V	Bypass capacitor
C_{ISO}	2.2 nF	Isolation capacitor Voltage rating depends on application requirement. It is common to select 2x of module voltage rating to provide a standoff margin in the event of a fault in the system.
Common-mode choke	100 µH to 500 µH	Common-mode choke An inductance range of 100 µH to 500 µH is a general guidance value, not a specified range as there are many parameters affecting the performance of a common-mode choke. When selecting a recommended part, refer to the SLVAEP4 BQ79616-Q1 Daisy Chain Communications Application Report. The user shall perform the through test in their environment.
ESD diode (optional)	TVS diode	Optional ESD protection depends on pcb level ESD requirement



Components Required for cap-choke Daisy Chain with Cabling

Figure 9-14. Capacitor Plus Choke Isolation

Longer cable lengths, or abnormally noisy applications may require the use of a common-mode choke filter. A capacitor plus choke isolation has better noise immunity than capacitor only. For these applications, use an automotive grade from 100-µH to 500-µH common-mode filter minimum for proper operation. To achieve the best performance in noisy environments, use dual common-mode filters (470 µH). The recommended impedance of the choke is at least 1 kΩ from 1 MHz to 100 MHz and above 300 Ω for higher frequencies.

Option 3: Transformer Isolation

Table 9-13. Components for Transformer Isolation

Components	Value	Description (Capacitor Plus Choke Isolation Between PCBs)
R_{TERM}	1 kΩ	Termination resistor
R_{SERIES}	49 Ω	Filter resistor and impedance matching resistor. The connection between devices must be ~120 Ω (~50 Ω on each end of the signal connection of the device plus 10-Ω internal resistance)
C_{BYPASS}	220 pF/50 V	Bypass capacitor
Transformer	Inductance: 150 µH to 1400 µH	The inductance range 150 µH to 1400 µH is a general guidance value, not a specified range as there are many parameters affecting the performance of the transformer. When selecting a recommended part, refer to the SLVAEP4 BQ79616-Q1 Daisy-Chain Communications Application Report. The user shall perform the through test in their environment.
ESD diode (optional)	TVS diode	Optional ESD protection depends on pcb level ESD requirement

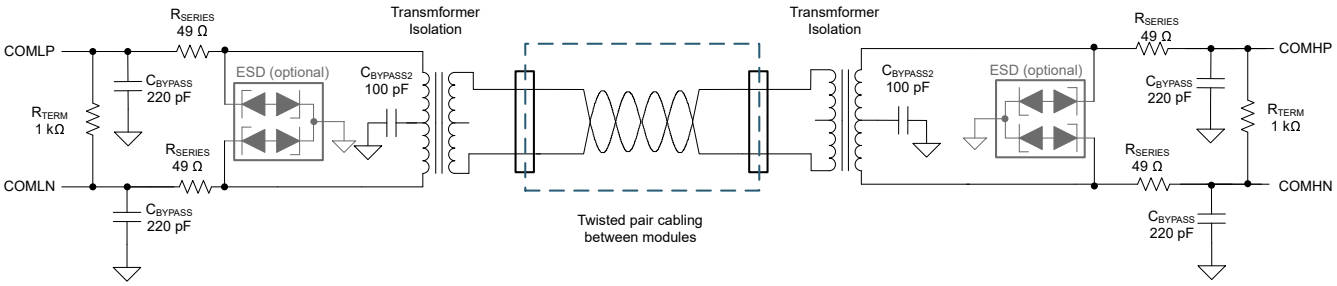


Figure 9-15. Transformer Isolation

Transformer isolation is supported and can be implemented as above. For example, transformer isolation can be used between the low-voltage and high-voltage boundary for galvanic isolation.

9.2.1.3 Application Curve

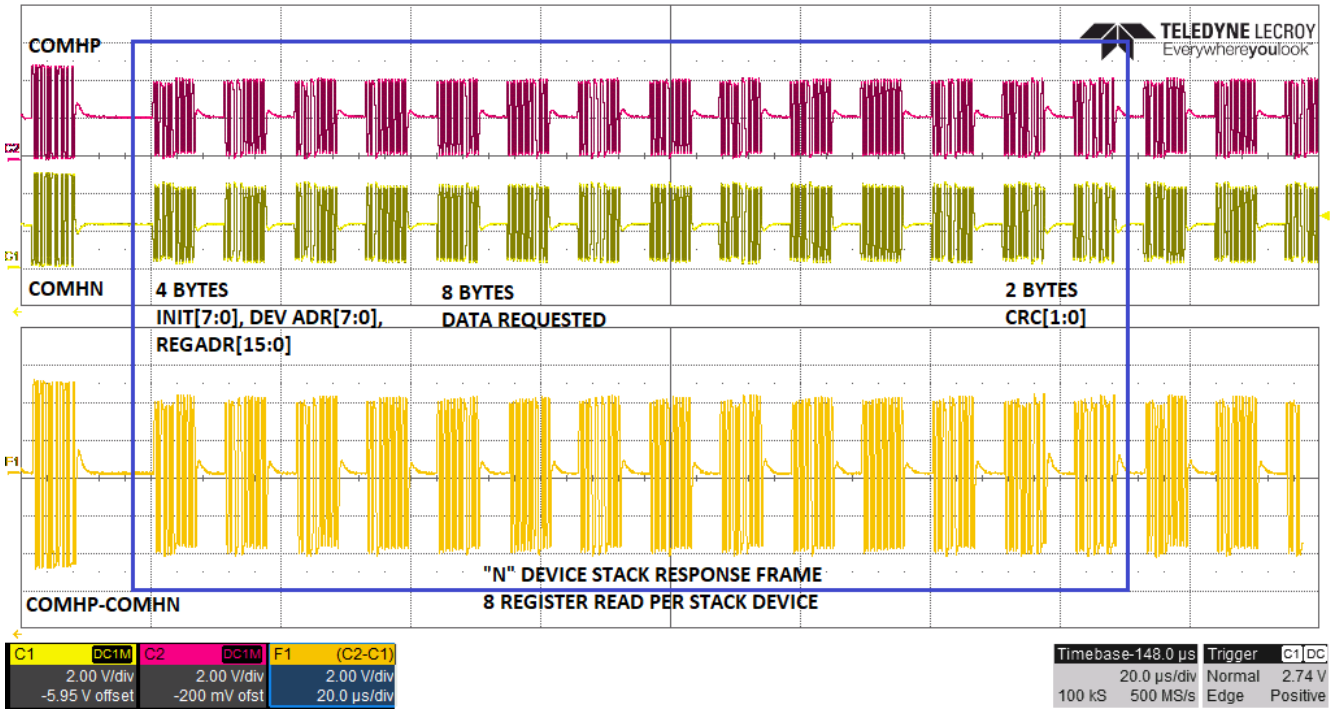


Figure 9-16. Response Frame for 8 Registers Read from Stack Devices

9.2.2 Daisy Device Application Circuit

The following application circuit (see Figure 9-17) is based on the BQ79631-Q1 device connecting to a 16S module.

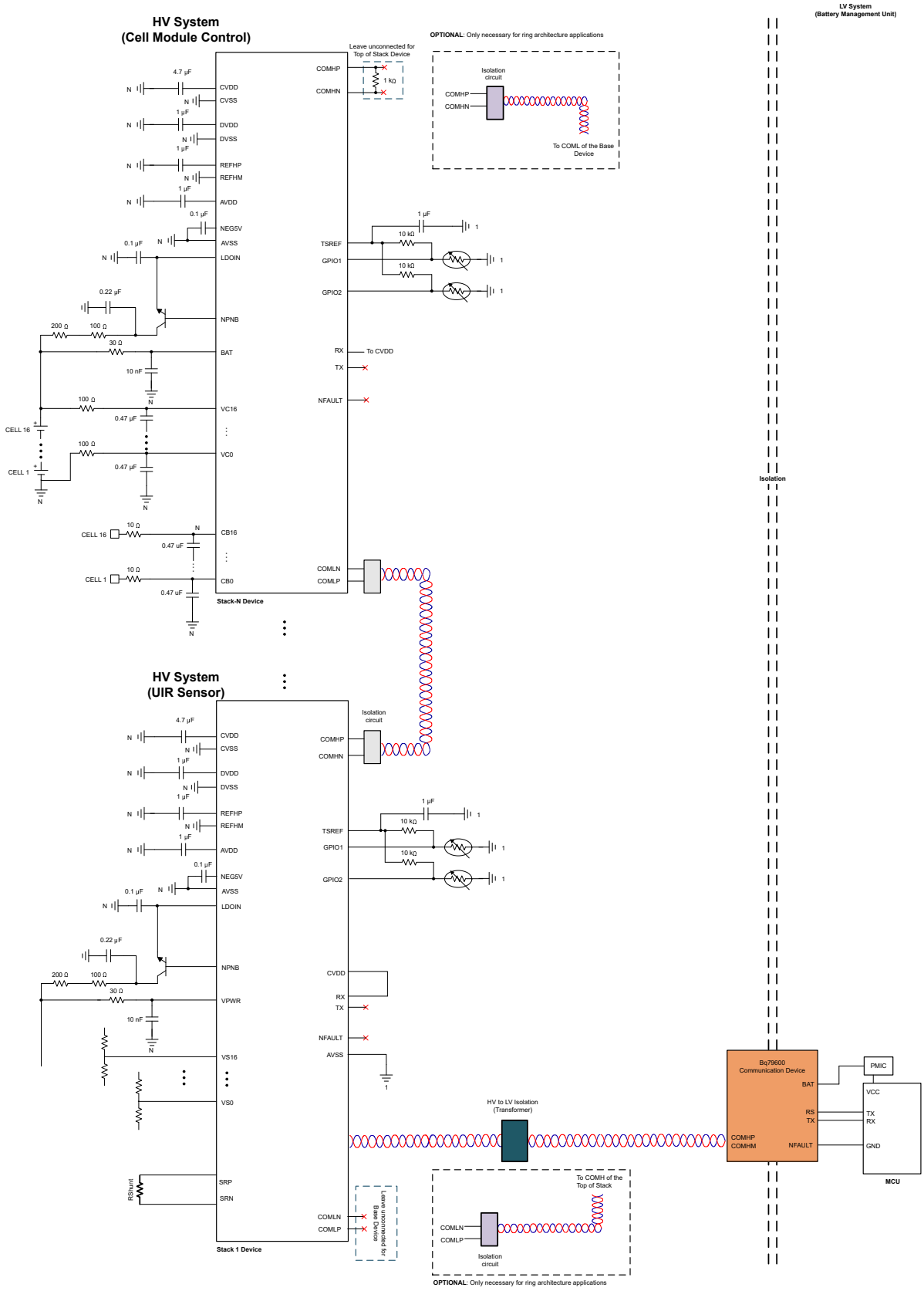


Figure 9-17. Daisy Device Application Circuit

10 Power Supply Recommendations

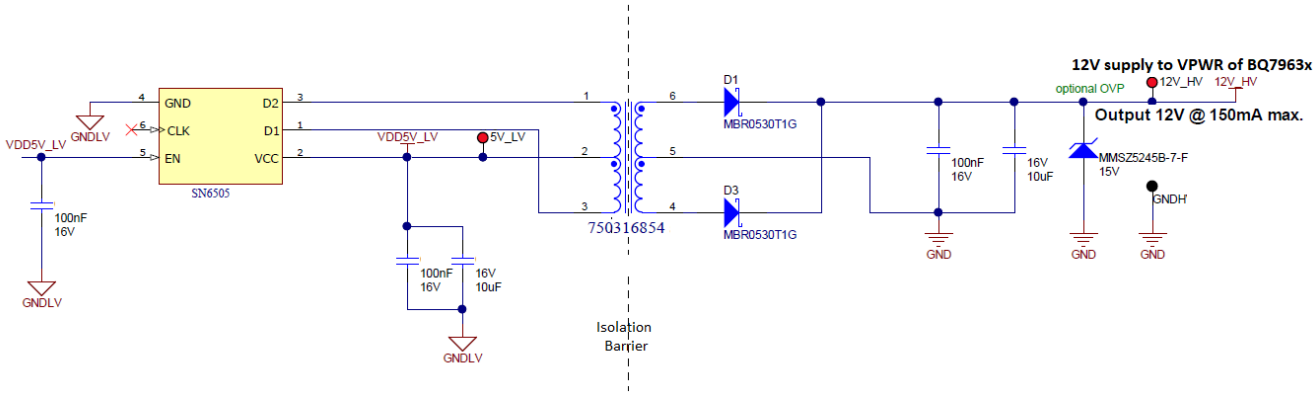


Figure 10-1. Device Powering Path

HV sensing devices such as the BQ79631-Q1 are generally powered by an isolated power supply to ensure that a device failure does not propagate the high voltage into the low voltage domain. Typical isolated power is generated using a simple push-pull switching technique using a transformer as shown in Figure 10-1. The input to the SN6505 is 5 V. This along with the turns ratio of the transformer provides a 12-V isolated supply to the BQ79631-Q1. The output has filters to smooth out the rectified waveform from the transformer and diodes. There is also a zener present to clamp any overvoltage events.

11 Layout

The layout for this device must be designed carefully. Any design outside these guidelines can affect ADC accuracy and EMI performance. Care must be taken in the layout of signals to and from the device to avoid coupling noise onto sensitive inputs. The layout of ground and power connections, as well as communication signals, should also be made carefully.

11.1 Layout Guidelines

11.1.1 Ground Planes

It is very important to establish a clean grounding scheme to ensure best performance of the device. There are three ground pins (AVSS, DVSS, CVSS) for the device's internal power supplies and one ground reference (REFHM) for the precision reference. There are noisy grounds and quiet grounds that must be separated in the layout initially and re-joined together in a lower PCB layer. The external components (for example, bypass capacitors) must be tied to the proper grounding group if possible to keep the separation of noisy and quiet grounds apart.

- AVSS ground:
  - Bypass capacitor for these pins: VPWR, VS0, AUX0, LDOIN and TSREF
  - Package PowerPAD
- DVSS ground:
  - Bypass capacitor for DVDD
  - GPIO filter capacitor (if used). It can also connect to AVSS ground plane, if needed.
- CVSS ground:
  - Bypass capacitor for CVDD
  - Bypass capacitors for COMHP/N, and COMLP/N
- REFHM ground:
  - Bypass capacitor for REFHP
  - If possible, separate out REFHM from AVSS on the signal connection layer and re-connect REFHM to AVSS ground plane in the lower layer.

Even on a PCB layer that is mainly for signal routing, it is good practice to pour have a small island of ground pour if possible to provide a low-impedance ground, rather than simply a via through the ground trace to an lower ground plane.

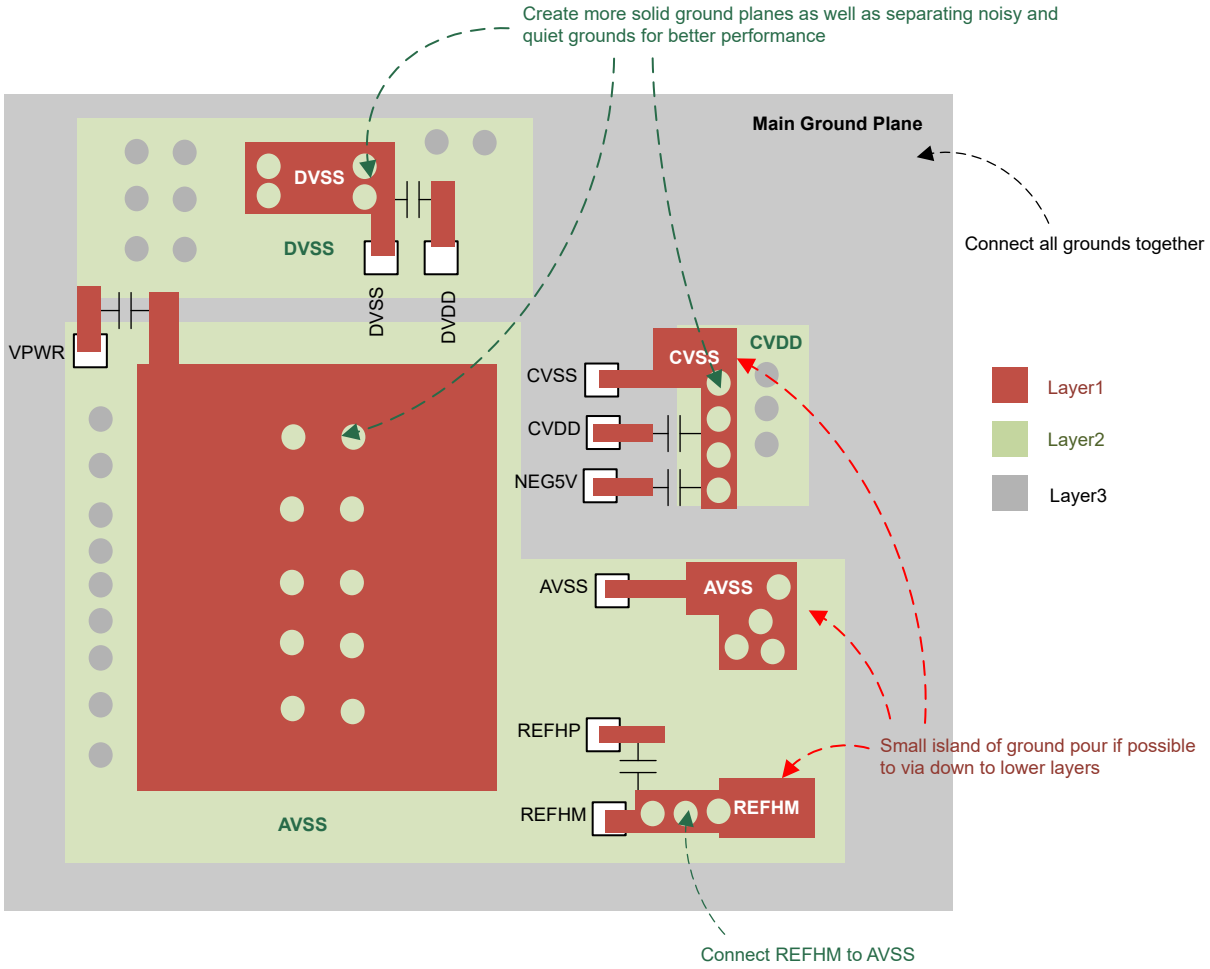


Figure 11-1. Grounding Layout Consideration

11.1.2 Bypass Capacitors for Power Supplies and Reference

The bypass capacitors of the following pins must be placed as close to the device pins as possible to ensure proper performance, especially for the REFHP capacitor.

- REFHP, VPWR, LDOIN, AVDD, DVDD, CVDD, TSREF, and NEG5V

11.1.3 Voltage Sensing

Voltage sensing traces (VS pins and AUX pins) must be placed in parallel with impedance matching.

High voltage traces must follow safe spacing rules. Adequte HV warning lables are recommended for operator safety.

11.1.4 Daisy-Chain Communication

It is important to have proper layout on the COMHP/N and COMLP/N circuits in order to have the best robust daisy-chain communication.

- Keep differential traces as short as possible and as straight as possible. Minimize turns and avoid any looping on the traces.
- Keep the differential traces on the same layers. Run the trace in parallel with shielding and matching trace impedance.
- Place the isolation components close to the connectors.
- When using capacitive isolation, place the high-voltage capacitor of the COMxP/N pair (where x = H or L) close to each other along the parallel traces.

- Create a keep-out area (no other traces and no ground plane) around the daisy-chain components in all PCB layers.

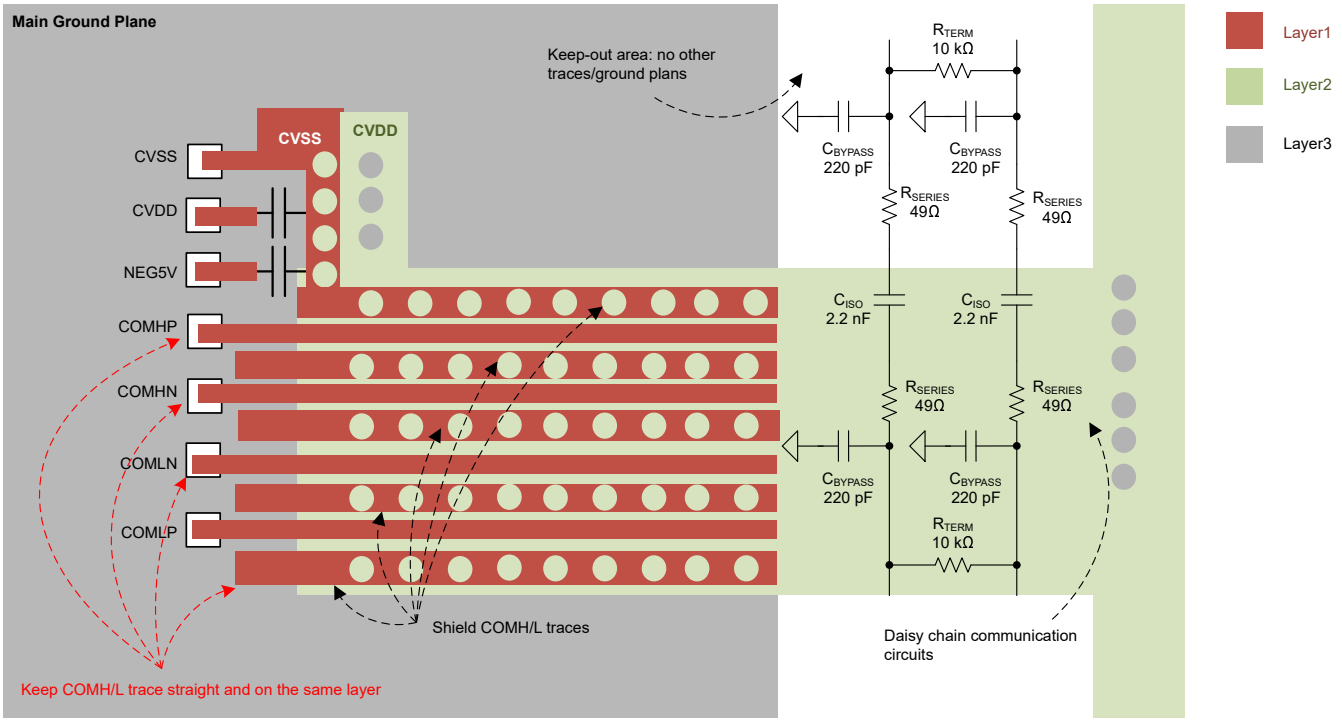


Figure 11-2. Daisy-Chain Layout Consideration

11.2 Layout Example

This section presents the BQ79631-Q1 Evaluation Module (EVM) design as a layout example.

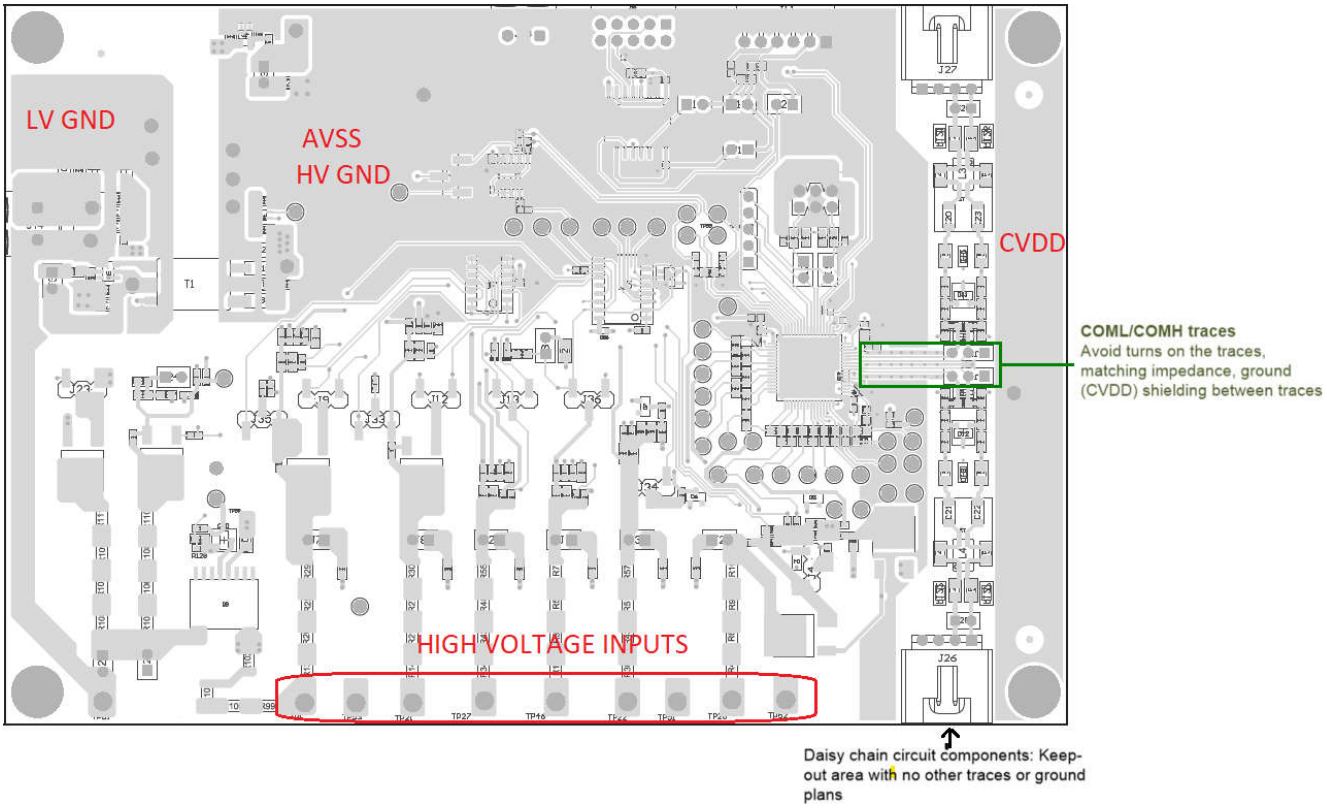


Figure 11-3. Top Signal Layer

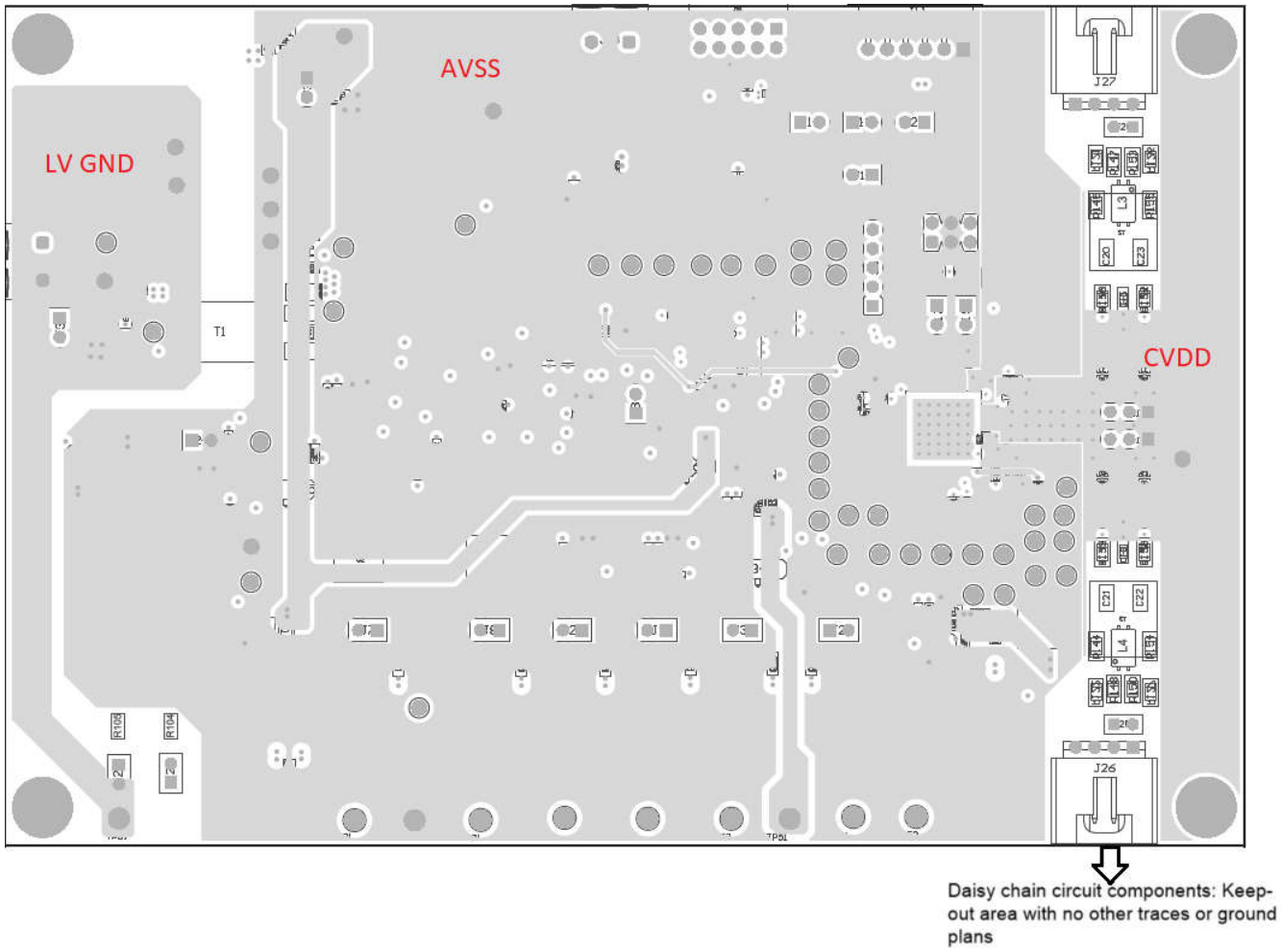


Figure 11-4. Second Layer with Solid, Separate Ground Planes

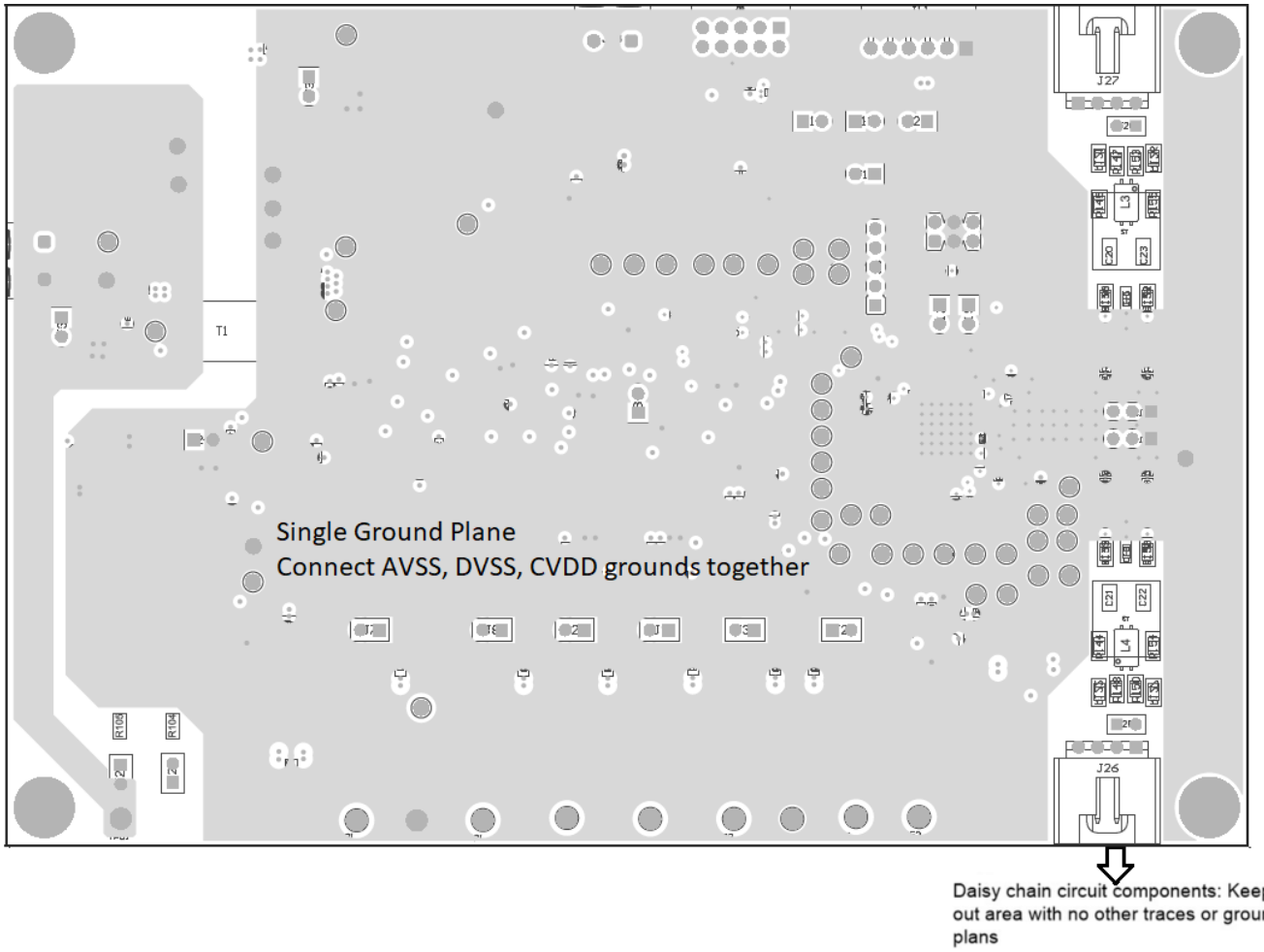


Figure 11-5. Thlrd Layer with Single Ground Plane

12 Device and Documentation Support

12.1 Device Support

12.1.1 Third-Party Products Disclaimer

TI'S PUBLICATION OF INFORMATION REGARDING THIRD-PARTY PRODUCTS OR SERVICES DOES NOT CONSTITUTE AN ENDORSEMENT REGARDING THE SUITABILITY OF SUCH PRODUCTS OR SERVICES OR A WARRANTY, REPRESENTATION OR ENDORSEMENT OF SUCH PRODUCTS OR SERVICES, EITHER ALONE OR IN COMBINATION WITH ANY TI PRODUCT OR SERVICE.

12.2 Receiving Notification of Documentation Updates

To receive notification of documentation updates, navigate to the device product folder on ti.com. Click on *Notifications* to register and receive a weekly digest of any product information that has changed. For change details, review the revision history included in any revised document.

12.3 Support Resources

TI E2E™ support forums are an engineer's go-to source for fast, verified answers and design help — straight from the experts. Search existing answers or ask your own question to get the quick design help you need.

Linked content is provided "AS IS" by the respective contributors. They do not constitute TI specifications and do not necessarily reflect TI's views; see TI's Terms of Use.

12.4 Trademarks

TI E2E™ is a trademark of Texas Instruments.
All trademarks are the property of their respective owners.

12.5 Electrostatic Discharge Caution

This integrated circuit can be damaged by ESD. Texas Instruments recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

12.6 Glossary

TI Glossary	This glossary lists and explains terms, acronyms, and definitions.

13 Mechanical, Packaging, and Orderable Information

The following pages include mechanical, packaging, and orderable information. This information is the most current data available for the designated devices. This data is subject to change without notice and revision of this document. For browser-based versions of this data sheet, refer to the left-hand navigation.

PACKAGING INFORMATION

Orderable Device	Status (1)	Package Type	Package Drawing	Pins	Package Qty	Eco Plan (2)	Lead finish/ Ball material (6)	MSL Peak Temp (3)	Op Temp (°C)	Device Marking (4/5)	Samples
BQ79631PAPRQ1	ACTIVE	HTQFP	PAP	64	1000	RoHS & Green	NIPDAU	Level-3-260C-168 HR	-40 to 125	BQ79631	Samples

(1) The marketing status values are defined as follows:
ACTIVE: Product device recommended for new designs.
LIFEBUY: TI has announced that the device will be discontinued, and a lifetime-buy period is in effect.
NRND: Not recommended for new designs. Device is in production to support existing customers, but TI does not recommend using this part in a new design.
PREVIEW: Device has been announced but is not in production. Samples may or may not be available.
OBSOLETE: TI has discontinued the production of the device.

(2) **RoHS:** TI defines "RoHS" to mean semiconductor products that are compliant with the current EU RoHS requirements for all 10 RoHS substances, including the requirement that RoHS substance do not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, "RoHS" products are suitable for use in specified lead-free processes. TI may reference these types of products as "Pb-Free".
RoHS Exempt: TI defines "RoHS Exempt" to mean products that contain lead but are compliant with EU RoHS pursuant to a specific EU RoHS exemption.
Green: TI defines "Green" to mean the content of Chlorine (Cl) and Bromine (Br) based flame retardants meet JS709B low halogen requirements of <=1000ppm threshold. Antimony trioxide based flame retardants must also meet the <=1000ppm threshold requirement.

(3) MSL, Peak Temp. - The Moisture Sensitivity Level rating according to the JEDEC industry standard classifications, and peak solder temperature.

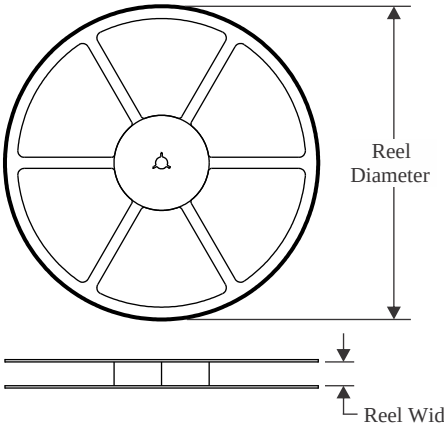
(4) There may be additional marking, which relates to the logo, the lot trace code information, or the environmental category on the device.

(5) Multiple Device Markings will be inside parentheses. Only one Device Marking contained in parentheses and separated by a "~" will appear on a device. If a line is indented then it is a continuation of the previous line and the two combined represent the entire Device Marking for that device.

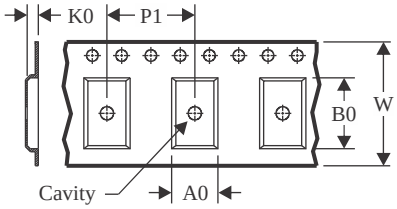
(6) Lead finish/Ball material - Orderable Devices may have multiple material finish options. Finish options are separated by a vertical ruled line. Lead finish/Ball material values may wrap to two lines if the finish value exceeds the maximum column width.

TAPE AND REEL INFORMATION

REEL DIMENSIONS

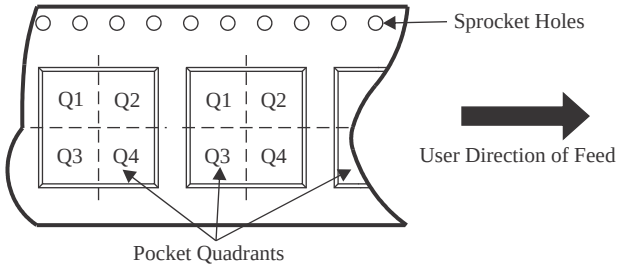


TAPE DIMENSIONS



A0	Dimension designed to accommodate the component width
B0	Dimension designed to accommodate the component length
K0	Dimension designed to accommodate the component thickness
W	Overall width of the carrier tape
P1	Pitch between successive cavity centers

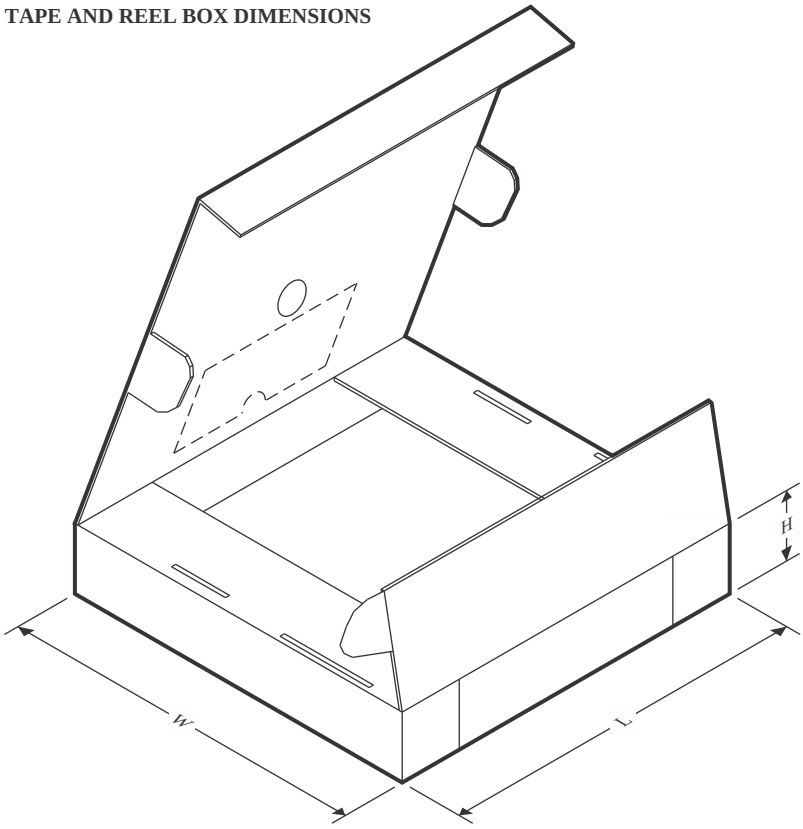
QUADRANT ASSIGNMENTS FOR PIN 1 ORIENTATION IN TAPE



*All dimensions are nominal

Device	Package Type	Package Drawing	Pins	SPQ	Reel Diameter (mm)	Reel Width W1 (mm)	A0 (mm)	B0 (mm)	K0 (mm)	P1 (mm)	W (mm)	Pin1 Quadrant
BQ79631PAPRQ1	HTQFP	PAP	64	1000	330.0	24.4	13.0	13.0	1.5	16.0	24.0	Q2

TAPE AND REEL BOX DIMENSIONS



*All dimensions are nominal

Device	Package Type	Package Drawing	Pins	SPQ	Length (mm)	Width (mm)	Height (mm)
BQ79631PAPRQ1	HTQFP	PAP	64	1000	367.0	367.0	55.0

GENERIC PACKAGE VIEW

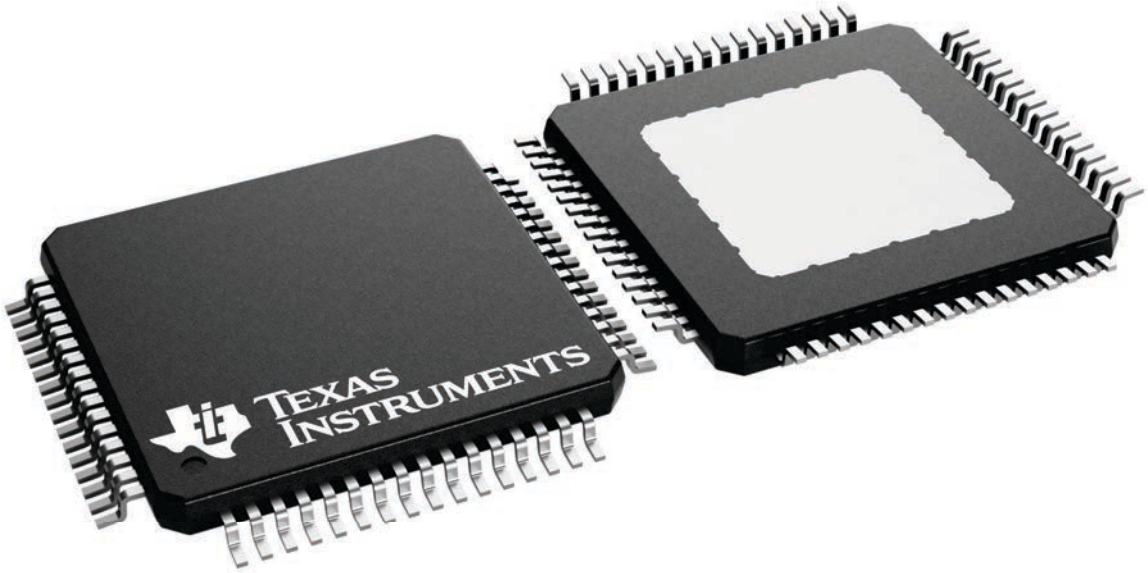
PAP 64

HTQFP - 1.2 mm max height

10 x 10, 0.5 mm pitch

QUAD FLATPACK

This image is a representation of the package family, actual package may vary.
Refer to the product data sheet for package details.



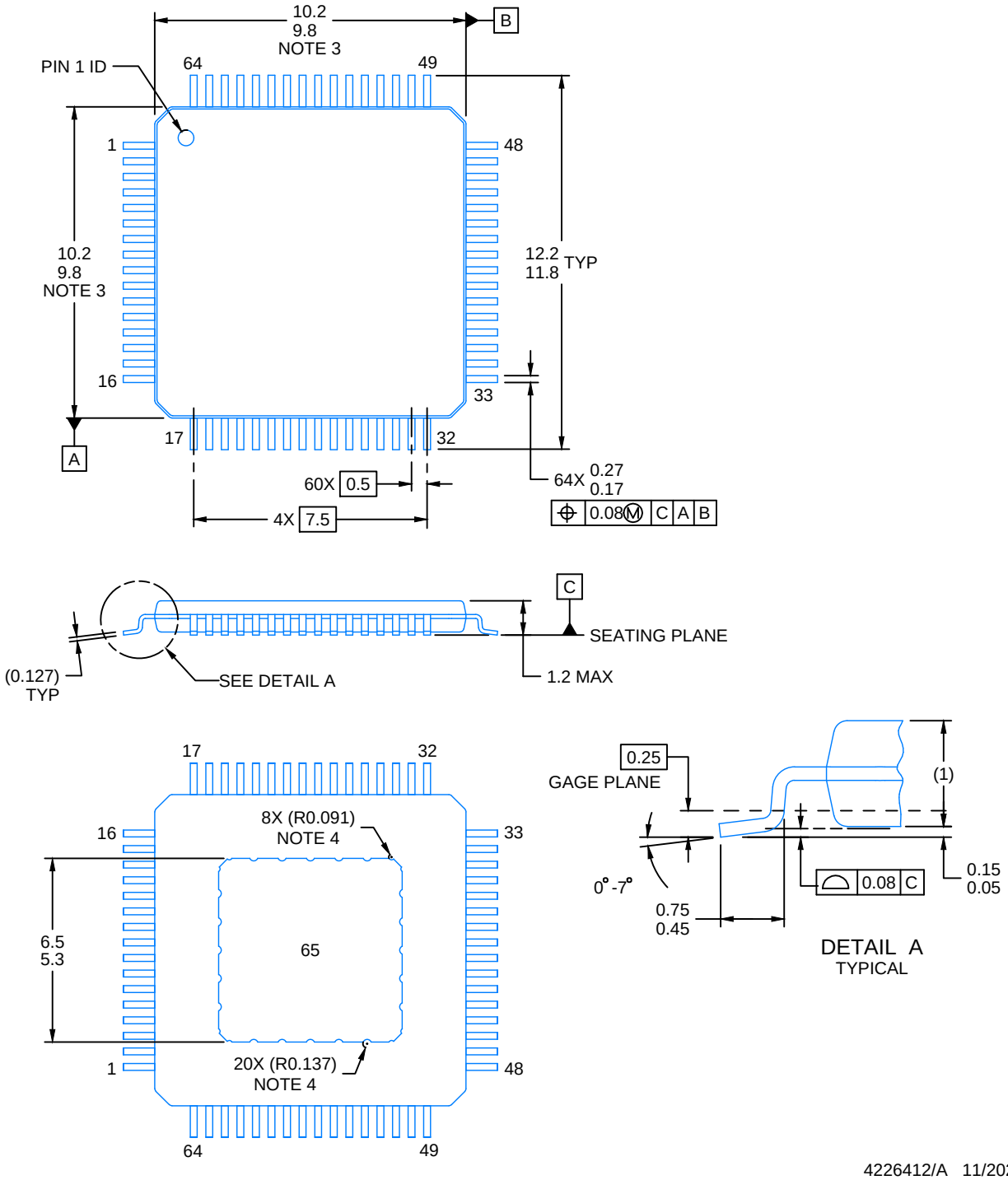
4226442/A

PACKAGE OUTLINE

PAP0064F

PowerPAD™ TQFP - 1.2 mm max height

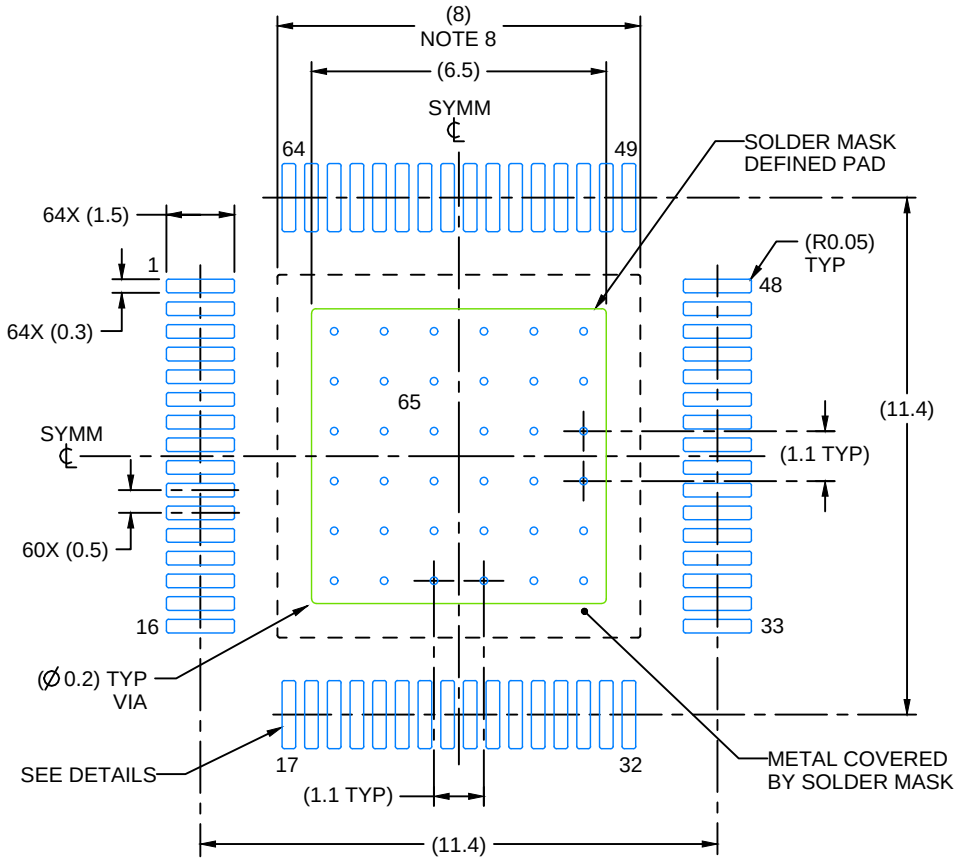
PLASTIC QUAD FLATPACK



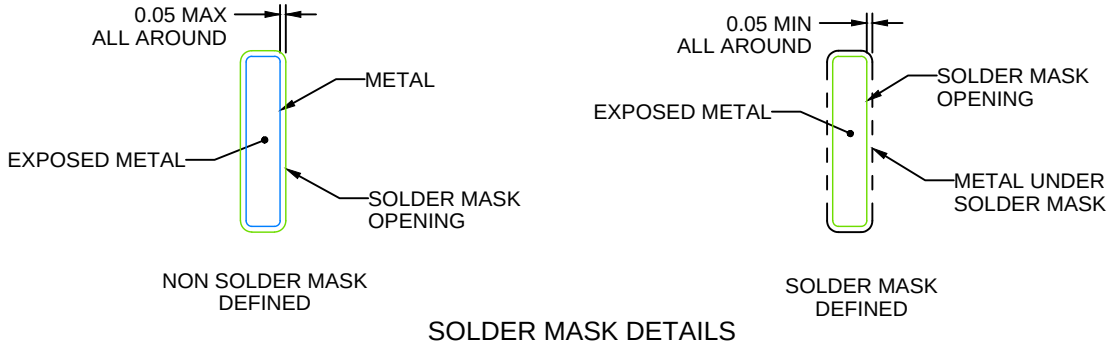
NOTES:

PowerPAD is a trademark of Texas Instruments.

1. All linear dimensions are in millimeters. Any dimensions in parenthesis are for reference only. Dimensioning and tolerancing per ASME Y14.5M.
2. This drawing is subject to change without notice.
3. This dimension does not include mold flash, protrusions, or gate burrs.
4. Strap features may not be present.
5. Reference JEDEC registration MS-026.



LAND PATTERN EXAMPLE
EXPOSED METAL SHOWN
SCALE:6X



NON SOLDER MASK
DEFINED

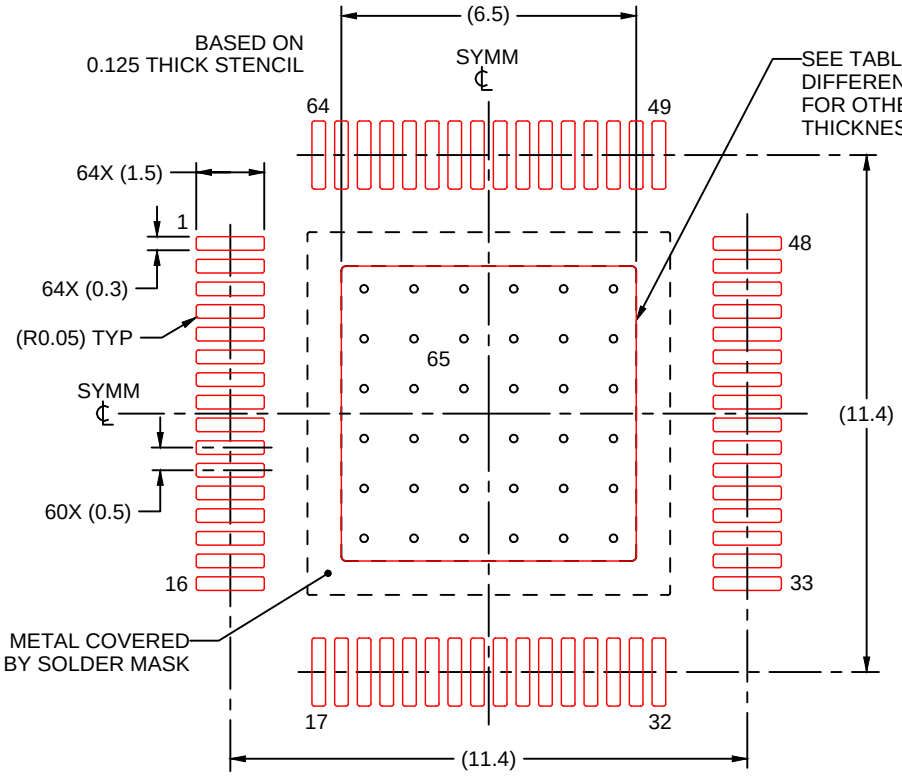
SOLDER MASK
DEFINED

SOLDER MASK DETAILS

4226412/A 11/2020

NOTES: (continued)

6. Publication IPC-7351 may have alternate designs.
7. Solder mask tolerances between and around signal pads can vary based on board fabrication site.
8. This package is designed to be soldered to a thermal pad on the board. See technical brief, Powerpad thermally enhanced package, Texas Instruments Literature No. SLMA002 (www.ti.com/lit/slma002) and SLMA004 (www.ti.com/lit/slma004).
9. Vias are optional depending on application, refer to device data sheet. It is recommended that vias under paste be filled, plugged or tented.
10. Size of metal pad may vary due to creepage requirement.



SOLDER PASTE EXAMPLE
EXPOSED PAD
100% PRINTED SOLDER COVERAGE BY AREA
SCALE:6X

STENCIL THICKNESS	SOLDER STENCIL OPENING
0.1	7.27 X 7.27
0.125	6.5 X 6.5 (SHOWN)
0.15	5.93 X 5.93
0.175	5.49 X 5.49

4226412/A 11/2020

NOTES: (continued)

11. Laser cutting apertures with trapezoidal walls and rounded corners may offer better paste release. IPC-7525 may have alternate design recommendations.
12. Board assembly site may have different recommendations for stencil design.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATA SHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES “AS IS” AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, regulatory or other requirements.

These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI’s products are provided subject to TI’s Terms of Sale or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI’s provision of these resources does not expand or otherwise alter TI’s applicable warranties or warranty disclaimers for TI products.

TI objects to and rejects any additional or different terms you may have proposed.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265