

TXE8148 48-Bit SPI Bus I/O Expander With Interrupt Output, Reset Input, and I/O Configuration Registers

1 Features

- [Functional Safety-Capable](#)
 - [Documentation available to aid in functional safety system design](#)
- Operating supply voltage range of 1.65V to 5.5V
- Low standby current consumption of 6µA typical
- 10MHz SPI SCLK frequency
- SPI daisy-chain supported
- SPI read and write with burst mode
- Multi-port SPI command to configure multiple ports simultaneously
- I_{OFF} supported input port pins
- Active-low reset input ($\overline{\text{RESET}}$)
- Open-drain active-low interrupt output ($\overline{\text{INT}}$)
 - Interrupt mask and status per I/O
 - Interrupt status per port
- Built-in fail safe I/O feature
- Individual I/O configuration for:
 - Input and output function
 - Polarity inversion
 - Output push-pull and open-drain selection
 - Integrated pullup or pulldown selection
 - Bus-hold feature to maintain last I/O state
 - Glitch filter enable selection
- Latched outputs with 10mA drive capability for directly driving LEDs
- Latch-up performance exceeds 100mA per AEC Q100-004
- Latch-up performance exceeds 100mA per JESD 78, class II

2 Applications

- [Industrial transportation](#)
- [Industrial automation](#)
- [Testing and measurement](#)
- [Factory automation and control](#)
- [Medical and healthcare](#)
- Servers
- Routers (telecom switching equipment)
- Products with GPIO-limited processors

3 Description

The TXE8148 device provides general purpose parallel input/output (I/O) expansion for the four wire Serial Peripheral Interface (SPI) protocol and is designed for 1.65V to 5.5V V_{CC} operation. The SPI protocol supports standard point-to-point communication along with daisy-chaining of multiple TXE8148.

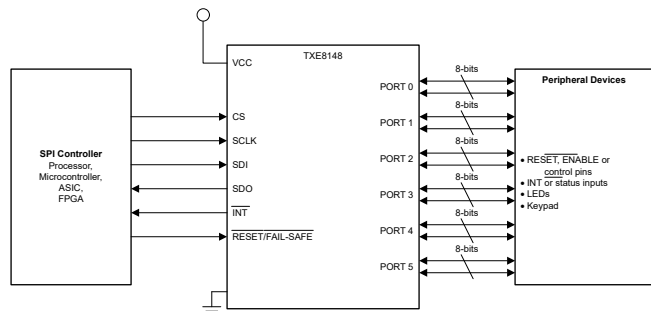
The device supports SPI bus frequency of 10MHz from 1.65V to 5.5V. I/O expanders such as the TXE8148 provide a simple design when additional I/Os are needed for switches, sensors, push-buttons, LEDs, fans, and so on.

The TXE8148 has six I/O ports each with eight IOs per port, which include additional features designed to enhance the I/O performance in terms of speed, power consumption and EMI. These features include per I/O programmable open-drain or push-pull outputs, programmable pullup and pulldown resistors, bus-hold latchable inputs, maskable interrupt, interrupt status register, and a fail-safe register mode which is enabled by the $\overline{\text{FAIL-SAFE}}$ pin.

Package Information

PART NUMBER	PACKAGE ⁽¹⁾	PACKAGE SIZE ⁽²⁾
TXE8148	DGG (TSSOP, 56)	14mm × 8.1mm

- (1) For more information, see [Section 11](#).
- (2) The package size (length × width) is a nominal value and includes pins, where applicable.

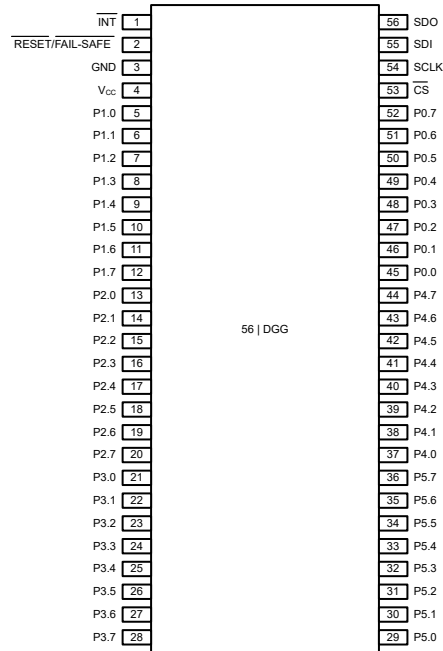


Simplified Schematic

Table of Contents

1 Features	1	7.3 Feature Description.....	18
2 Applications	1	7.4 Device Functional Modes.....	22
3 Description	1	7.5 Programming.....	23
4 Pin Configuration and Functions	3	7.6 TXE8148 Registers.....	29
5 Specifications	5	8 Application and Implementation	55
5.1 Absolute Maximum Ratings.....	5	8.1 Application Information.....	55
5.2 ESD Ratings.....	5	8.2 Typical Application.....	55
5.3 Recommended Operating Conditions.....	5	8.3 Power Supply Recommendations.....	56
5.4 Thermal Information.....	6	8.4 Layout.....	57
5.5 Electrical Characteristics.....	6	9 Device and Documentation Support	59
5.6 Timing Requirements.....	8	9.1 Receiving Notification of Documentation Updates....	59
5.7 SPI Bus Timing Requirements.....	8	9.2 Support Resources.....	59
5.8 Switching Characteristics.....	9	9.3 Trademarks.....	59
5.9 Typical Characteristics.....	10	9.4 Electrostatic Discharge Caution.....	59
6 Parameter Measurement Information	13	9.5 Glossary.....	59
7 Detailed Description	17	10 Revision History	59
7.1 Overview.....	17	11 Mechanical, Packaging, and Orderable Information	59
7.2 Functional Block Diagrams.....	17		

4 Pin Configuration and Functions



**Figure 4-1. DGG (TSSOP) Package, 56-Pin
(Top View)**

Table 4-1. Pin Functions

PIN		TYPE ⁽¹⁾	DESCRIPTION
NAME	TSSOP (DGG)		
INT	1	O	Interrupt output. Connect to V_{CC} through a pullup resistor.
RESET/FAIL-SAFE	2	I	Active Low reset input. Connect to V_{CC} through a pullup resistor.
GND	3	G	Ground
V_{CC}	4	P	Supply voltage
P1.0	5	I/O	P-port input/output. At power on, Port 1 - IO #0 is configured as an input.
P1.1	6	I/O	P-port input/output. At power on, Port 1 - IO #1 is configured as an input.
P1.2	7	I/O	P-port input/output. At power on, Port 1 - IO #2 is configured as an input.
P1.3	8	I/O	P-port input/output. At power on, Port 1 - IO #3 is configured as an input.
P1.4	9	I/O	P-port input/output. At power on, Port 1 - IO #4 is configured as an input.
P1.5	10	I/O	P-port input/output. At power on, Port 1 - IO #5 is configured as an input.
P1.6	11	I/O	P-port input/output. At power on, Port 1 - IO #6 is configured as an input.
P1.7	12	I/O	P-port input/output. At power on, Port 1 - IO #7 is configured as an input.
P2.0	13	I/O	P-port input/output. At power on, Port 2 - IO #0 is configured as an input.
P2.1	14	I/O	P-port input/output. At power on, Port 2 - IO #1 is configured as an input.
P2.2	15	I/O	P-port input/output. At power on, Port 2 - IO #2 is configured as an input.
P2.3	16	I/O	P-port input/output. At power on, Port 2 - IO #3 is configured as an input.
P2.4	17	I/O	P-port input/output. At power on, Port 2 - IO #4 is configured as an input.
P2.5	18	I/O	P-port input/output. At power on, Port 2 - IO #5 is configured as an input.
P2.6	19	I/O	P-port input/output. At power on, Port 2 - IO #6 is configured as an input.
P2.7	20	I/O	P-port input/output. At power on, Port 2 - IO #7 is configured as an input.
P3.0	21	I/O	P-port input/output. At power on, Port 3 - IO #0 is configured as an input.

Table 4-1. Pin Functions (continued)

PIN		TYPE ⁽¹⁾	DESCRIPTION
NAME	TSSOP (DGG)		
P3.1	22	I/O	P-port input/output. At power on, Port 3 - IO #1 is configured as an input
P3.2	23	I/O	P-port input/output. At power on, Port 3 - IO #2 is configured as an input.
P3.3	24	I/O	P-port input/output. At power on, Port 3 - IO #3 is configured as an input.
P3.4	25	I/O	P-port input/output. At power on, Port 3 - IO #4 is configured as an input.
P3.5	26	I/O	P-port input/output. At power on, Port 3 - IO #5 is configured as an input.
P3.6	27	I/O	P-port input/output. At power on, Port 3 - IO #6 is configured as an input.
P3.7	28	I/O	P-port input/output. At power on, Port 3 - IO #7 is configured as an input.
P5.0	29	I/O	P-port input/output. At power on, Port 5 - IO #0 is configured as an input.
P5.1	30	I/O	P-port input/output. At power on, Port 5 - IO #1 is configured as an input.
P5.2	31	I/O	P-port input/output. At power on, Port 5 - IO #2 is configured as an input.
P5.3	32	I/O	P-port input/output. At power on, Port 5 - IO #3 is configured as an input.
P5.4	33	I/O	P-port input/output. At power on, Port 5 - IO #4 is configured as an input.
P5.5	34	I/O	P-port input/output. At power on, Port 5 - IO #5 is configured as an input.
P5.6	35	I/O	P-port input/output. At power on, Port 5 - IO #6 is configured as an input.
P5.7	36	I/O	P-port input/output. At power on, Port 5 - IO #7 is configured as an input
P4.0	37	I/O	P-port input/output. At power on, Port 4 - IO #0 is configured as an input
P4.1	38	I/O	P-port input/output. At power on, Port 4 - IO #1 is configured as an input
P4.2	39	I/O	P-port input/output. At power on, Port 4 - IO #2 is configured as an input
P4.3	40	I/O	P-port input/output. At power on, Port 4 - IO #3 is configured as an input
P4.4	41	I/O	P-port input/output. At power on, Port 4 - IO #4 is configured as an input
P4.5	42	I/O	P-port input/output. At power on, Port 4 - IO #5 is configured as an input
P4.6	43	I/O	P-port input/output. At power on, Port 4 - IO #6 is configured as an input
P4.7	44	I/O	P-port input/output. At power on, Port 4 - IO #7 is configured as an input
P0.0	45	I/O	P-port input/output. At power on, Port 0 - IO #0 is configured as an input
P0.1	46	I/O	P-port input/output. At power on, Port 0 - IO #1 is configured as an input
P0.2	47	I/O	P-port input/output. At power on, Port 0 - IO #2 is configured as an input
P0.3	48	I/O	P-port input/output. At power on, Port 0 - IO #3 is configured as an input.
P0.4	49	I/O	P-port input/output. At power on, Port 0 - IO #4 is configured as an input.
P0.5	50	I/O	P-port input/output. At power on, Port 0 - IO #5 is configured as an input.
P0.6	51	I/O	P-port input/output. At power on, Port 0 - IO #6 is configured as an input.
P0.7	52	I/O	P-port input/output. At power on, Port 0 - IO #7 is configured as an input.
\overline{CS}	53	I	SPI chip select input. Internal pullup resistor.
SCLK	54	I	SPI serial clock input. Internal pulldown resistor.
SDI	55	I	SPI serial data input.
SDO	56	O	SPI serial data output. Push-pull output

(1) I = Input, O = Output, I/O = Input or Output, G = Ground, P = Power

5 Specifications

5.1 Absolute Maximum Ratings

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

		MIN	MAX	UNIT	
V _{CC}	Supply voltage	-0.5	6.5	V	
V _I	Input voltage ⁽²⁾	-0.5	6.5	V	
V _O	Output voltage ⁽²⁾	-0.5	6.5	V	
I _{IK}	Input clamp current	RESET, SCLK, \overline{CS}	V _I < 0	-20	mA
I _{OK}	Output clamp current	\overline{INT}	V _O < 0	-20	mA
I _{IOK}	Input-output clamp current	P0.0 - P5.7	V _O < 0 or V _O > V _{CC}	±20	mA
I _{CC}	Continuous current through GND			-200	mA
I _{CC}	Continuous current through V _{CC}			160	mA
T _J	Junction temperature	-40	150	°C	
T _{stg}	Storage temperature	-40	150	°C	

- (1) Operation outside the Absolute Maximum Ratings may cause permanent device damage. Absolute Maximum Ratings do not imply functional operation of the device at these or any other conditions beyond those listed under Recommended Operating Conditions. If used outside the Recommended Operating Conditions but within the Absolute Maximum Ratings, the device may not be fully functional, and this may affect device reliability, functionality, performance, and shorten the device lifetime.
- (2) The input negative-voltage and output voltage ratings can be exceeded if the input and output current ratings are observed.

5.2 ESD Ratings

		VALUE	UNIT	
V _(ESD)	Electrostatic discharge	Human body model (HBM), per ANSI/ESDA/ JEDEC JS-001, all pins ⁽¹⁾	±2000	V
		Charged device model (CDM), per ANSI/ESDA/ JEDEC JS-002, all pins ⁽²⁾	±1000	

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

5.3 Recommended Operating Conditions

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

		MIN	MAX	UNIT	
V _{CC}	Supply voltage	1.65	5.5	V	
V _{IH}	High-level input voltage	P Port	0.7 × V _{CC}	V _{CC}	V
		SCLK, SDI, CS, RESET,	0.7 × V _{CC}	V _{CC}	V
V _{IL}	Low-level input voltage	P Port	-0.5	0.3 × V _{CC}	V
		SCLK, SDI, CS, RESET,	-0.5	0.3 × V _{CC}	V
I _{OH}	High-level output current (V _{CC} ≥ 2.3V)	P Port		-10	mA
	High-level output current (V _{CC} < 2.3V)			-5	mA
I _{OL}	Low-level output current (V _{CC} ≥ 2.3V)			10	mA
	Low-level output current (V _{CC} < 2.3V)			5	mA
T _A	Ambient temperature	-40	130	°C	

5.4 Thermal Information

THERMAL METRIC ⁽¹⁾		TXE8148	
		DGG (TSSOP)	
		56 PINS	
			UNIT
R _{θJA}	Junction-to-ambient thermal resistance	61.73	°C/W
R _{θJC(top)}	Junction-to-case (top) thermal resistance	26.69	°C/W
R _{θJB}	Junction-to-board thermal resistance	39.70	°C/W
Ψ _{JT}	Junction-to-top characterization parameter	3.68	°C/W
Ψ _{JB}	Junction-to-board characterization parameter	39.32	°C/W

(1) For more information about traditional and new thermal metrics, see the [Semiconductor and IC Package Thermal Metrics](#) application report.

5.5 Electrical Characteristics

over operating free-air temperature range and V_{CC} = 1.65V to 5.5V (unless otherwise noted)

PARAMETER		TEST CONDITIONS	V _{CC}	MIN	TYP	MAX	UNIT
V _{IK}	Input diode clamp voltage	I _I = -18mA		-1.2			V
V _{PORR}	Power-on reset voltage, V _{CC} rising	V _I = V _{CC} or GND, I _O = 0				1.3	V
V _{PORF}	Power-on reset voltage, V _{CC} falling				1.05		V
V _{OH}	High-level output voltage	P Port	I _{OH} = 4mA	V _{CC} = 1.65V	1.23		V
			I _{OH} = 8mA	V _{CC} = 2.3V	1.73		V
				V _{CC} = 3V	2.55		V
				V _{CC} = 4.5V	4.0		V
		I _{OH} = 10mA	V _{CC} = 5.5V	4.95		V	
			V _{CC} = 3.3V	2.53		V	
			V _{CC} = 5V	4.3		V	
			V _{CC} = 5.5V	4.85		V	
SDO	I _{OH} = 3mA		V _{CC} - 0.4			V	
V _{OL}	Low-level output voltage	P Ports	I _{OL} = -4mA	V _{CC} = 1.65V		0.23	V
			I _{OL} = -8mA	V _{CC} = 2.3V		0.36	V
				V _{CC} = 3V		0.25	V
				V _{CC} = 4.5V		0.17	V
		I _{OL} = -10mA	V _{CC} = 5.5V		0.16	V	
			V _{CC} = 3.3V		0.40	V	
			V _{CC} = 5V		0.33	V	
			V _{CC} = 5.5V		0.32	V	
SDO	I _{OL} = -3mA				0.4	V	
I _{OL}	Low-level output current	INT	V _{OL} = 0.4V		4		mA
I _I	Input leakage current	P Ports	V _I = V _{CC} or GND			±1	μA
			V _I = 3.6V	V _{CC} = 0V		±1	
		SDI, RESET	V _I = V _{CC} or GND			±1	
		SCLK	V _I = GND			±1	μA
			V _I = V _{CC}			±65	μA
		CS	V _I = V _{CC}			±1	μA
		V _I = GND			±65	μA	

5.5 Electrical Characteristics (continued)

over operating free-air temperature range and $V_{CC} = 1.65V$ to $5.5V$ (unless otherwise noted)

PARAMETER		TEST CONDITIONS		V_{CC}	MIN	TYP	MAX	UNIT		
I_{CC}	Quiescent current	Standby mode	SDI, \overline{CS} and $\overline{RESET} = V_{CC}$, P port = V_{CC} or GND, I/O = inputs, $I_O = 0mA$ $f_{SCLK} = 0MHz$, $-40^{\circ}C < T_A \leq 85^{\circ}C$, IO resistors disabled	$V_{CC} = 5.5V$			8	μA		
				$V_{CC} = 3.6V$			7.5	μA		
				$V_{CC} = 2.7V$			7.2	μA		
				$V_{CC} = 1.95V$			7.0	μA		
			Active current	Active mode (5MHz)	SDI, \overline{CS} and $\overline{RESET} = V_{CC}$, P port = V_{CC} or GND, I/O = inputs, $I_O = 0mA$ $f_{SCLK} = 5MHz$, 100pF load on SDO $-40^{\circ}C < T_A \leq 125^{\circ}C$, IO resistors disabled	$V_{CC} = 5.5V$			26	μA
						$V_{CC} = 3.6V$			24	μA
						$V_{CC} = 2.7V$			23.6	μA
						$V_{CC} = 1.95V$			23.4	μA
	Active current	Active mode (10MHz)	SDI and \overline{CS} in continuous read, $\overline{RESET} = V_{CC}$, P port = V_{CC} or GND, I/O = inputs, $I_O = 0mA$ $f_{SCLK} = 10MHz$, 100pF load on SDO $-40^{\circ}C < T_A \leq 125^{\circ}C$, IO resistors disabled	$V_{CC} = 5.5V$			2.5	mA		
				$V_{CC} = 3.6V$			1.6	mA		
				$V_{CC} = 2.7V$			1.3	mA		
				$V_{CC} = 1.95V$			1.0	mA		
I_{BHL}	Bus-hold low sustaining current	Ramp V_I from 0 to $0.3 \times V_{CC}$	$V_{CC} = 1.65V$			42	μA			
			$V_{CC} = 2.3V$			62	μA			
			$V_{CC} = 3V$			72	μA			
			$V_{CC} = 4.5V$			125	μA			
I_{BHH}	Bus-hold high sustaining current	Ramp V_I from V_{CC} to $0.7 \times V_{CC}$	$V_{CC} = 1.65V$			-32	μA			
			$V_{CC} = 2.3V$			-42	μA			
			$V_{CC} = 3V$			-72	μA			
			$V_{CC} = 4.5V$			-115	μA			
I_{BHLO}	Bus-hold low overdrive current	Ramp input up $V_I = 0$ to V_{CC}	$V_{CC} = 1.95V$		170		μA			
			$V_{CC} = 2.7V$		260		μA			
			$V_{CC} = 3.6V$		340		μA			
			$V_{CC} = 5.5V$		500		μA			
I_{BHHO}	Bus-hold high overdrive current	Ramp input down $V_I = V_{CC}$ to 0	$V_{CC} = 1.95V$		-170		μA			
			$V_{CC} = 2.7V$		-260		μA			
			$V_{CC} = 3.6V$		-340		μA			
			$V_{CC} = 5.5V$		-500		μA			
$R_{pu(int)}$	internal pullup resistance	\overline{CS}			70	100	140	k Ω		
		P port			70	100	140	k Ω		
$R_{pd(int)}$	internal pulldown resistance	P port			70	100	140	k Ω		
		SCLK			70	100	140	k Ω		
C_i	Input pin capacitance	SCLK	$V_I = V_{CC}$ or GND				3	pF		
		SDI					3	pF		
		\overline{CS}					3	pF		
		\overline{RESET}					3	pF		

5.5 Electrical Characteristics (continued)

over operating free-air temperature range and $V_{CC} = 1.65V$ to $5.5V$ (unless otherwise noted)

PARAMETER		TEST CONDITIONS	V_{CC}	MIN	TYP	MAX	UNIT
C_{IO}	Input-output pin capacitance	P port	$V_{IO} = V_{CC}$ or GND			4	pF

5.6 Timing Requirements

over operating free-air temperature range and $V_{CC} = 1.65V$ to $5.5V$ (unless otherwise noted)

			MIN	MAX	UNIT
RESET					
t_w	Reset pulse duration, SDO $C_{LOAD} = 100pF$ (Figure 6-1)		100		ns
t_{REC}	Reset recovery time, SDO $C_{LOAD} = 100pF$ (Figure 6-1)			100	ns
t_{RESET}	Time to reset, SDO $C_{LOAD} = 100pF$ (Figure 6-1)			80	ns
Power-On Reset					
t_{FT}	Fall rate (Figure 8-2) (Figure 8-3)		0.1	2000	ms
t_{RT}	Rise rate (Figure 8-2) (Figure 8-3)		0.1	2000	ms
t_{TRR_GND}	Time to re-ramp (when V_{CC} drops to GND) (Figure 8-2)		1		μs
t_{TRR_POR50}	Time to re-ramp (when V_{CC} drops to $VPOR_MIN - 50mV$) (Figure 8-3)		1		μs
V_{CC_GH}	Level that V_{CC} can glitch down to, but not cause a functional disruption when $t_{VCC_GW} = 1\mu s$ (Figure 8-4)			1.2	V
t_{VCC_GW}	Glitch width that does not cause a functional disruption when $V_{CC_GH} = 0.5 \times V_{CC}$ (Figure 8-4)			10	μs
Failsafe					
f_{SEN}	Failsafe IO enable time (100pF load) (Figure 6-2)	Output High in normal mode and Output Low in fail safe mode		100	ns
		Output Low in normal mode and output high in fail safe mode		100	ns
		Output high in normal mode and input in fail safe mode (500 Ω pulldown load)		35	ns
		Output low in normal mode and input in fail safe mode (500 Ω pulldown load)		29	ns
f_{SDIS}	Failsafe IO disable time (100pF load) (Figure 6-2)	Output High in normal mode and Output Low in fail safe mode		100	ns
		Output Low in normal mode and output high in fail safe mode		100	ns
		Input in normal mode and output high in fail safe mode (500 Ω pulldown load)		91	ns
		Input in normal mode and output low in fail safe mode (500 Ω pulldown load)		63	ns
Digital IO					
t_{GW}	Digital glitch filter width		90	230	ns

5.7 SPI Bus Timing Requirements

over operating free-air temperature range, $V_{CC} = 1.65V$ to $5.5V$ and SDO $C_{LOAD} = 100pF$ (unless otherwise noted) (Figure 6-3)

		MIN	MAX	UNIT
SPI Bus				
f_{SCLK}	SPI clock frequency		10	MHz
t_{CSS}	\overline{CS} to SCLK Rise Setup Time	50		ns

5.7 SPI Bus Timing Requirements (continued)

over operating free-air temperature range, $V_{CC} = 1.65V$ to $5.5V$ and $SDO C_{LOAD} = 100pF$ (unless otherwise noted) (Figure 6-3)

		MIN	MAX	UNIT
t_{CSH}	SCLK Fall to \overline{CS} De-asserted Hold Time	50		ns
t_{CSD}	\overline{CS} Disable Time	50		ns
t_{DS}	SDI to SCLK Setup Time	10		ns
t_{DH}	SDI to SCLK Hold Time	10		ns
t_{LOW}	SCLK Low Time	50		ns
t_{HIGH}	SCLK High Time	50		ns
$t_{V(SDO)}$	SDO Valid Time	$V_{CC} < 2.3V$	46	ns
		$V_{CC} \geq 2.3V$	28	ns
$t_{DIS(SDO)}$	SDO Disable Time		100	ns

5.8 Switching Characteristics

over operating free-air temperature range and $V_{CC} = 1.65V$ to $5.5V$ (unless otherwise noted)

PARAMETER		FROM (INPUT)	TO (OUTPUT)	MIN	TYP	MAX	UNIT
t_{iv}	Interrupt valid time, $INT C_{LOAD} = 100pF$ and $R_{PU} = 4.7k\Omega$ (Figure 6-4)	P port	\overline{INT}			0.2	μs
t_{ir}	Interrupt reset delay time, $INT C_{LOAD} = 100pF$ and $R_{PU} = 4.7k\Omega$ (Figure 6-4)	SCLK	\overline{INT}			0.4	μs
t_{pv}	Output data valid time, $SDO C_{LOAD} = 100pF$ (Figure 6-5)	SCLK	P port			100	ns
t_{ps}	Input data setup time, $SDO C_{LOAD} = 100pF$ (Figure 6-5)	P port	SCLK	26			ns
t_{ph}	Input data hold time, $SDO C_{LOAD} = 100pF$ (Figure 6-5)	SCLK	P port	2.5			ns

5.9 Typical Characteristics

$T_A = 25^\circ\text{C}$ (unless otherwise noted)

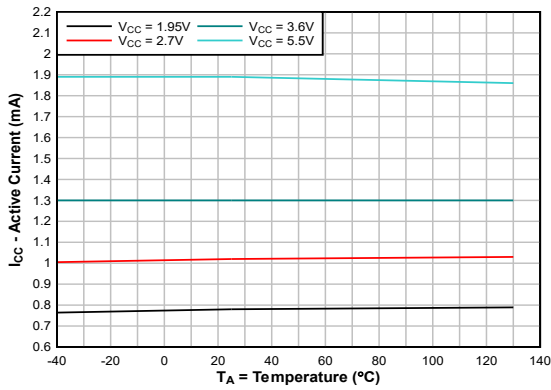


Figure 5-1. Active Current vs Temperature ($f_{SCLK} = 5\text{MHz}$)

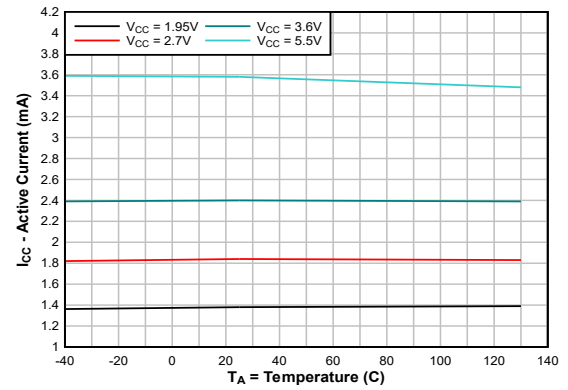


Figure 5-2. Active Current vs Temperature ($f_{SCLK} = 10\text{MHz}$)

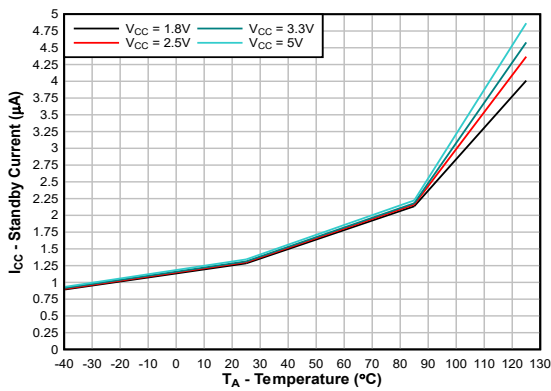


Figure 5-3. Standby Supply Current vs Temperature

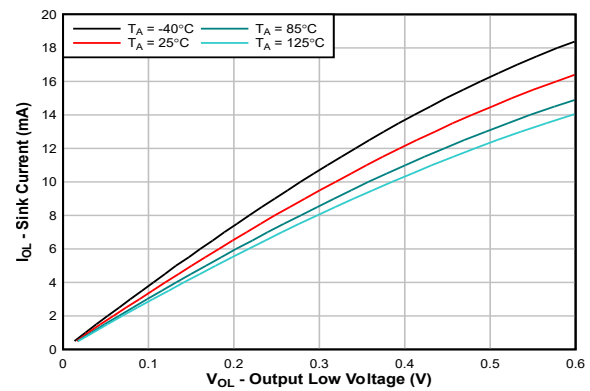


Figure 5-4. V_{OL} vs I_{OL} for $V_{CC} = 1.8\text{V}$

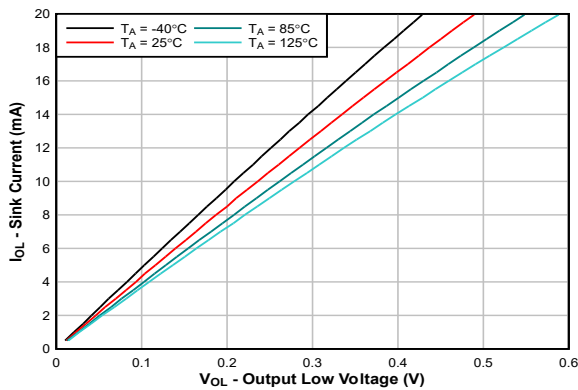


Figure 5-5. V_{OL} vs I_{OL} for $V_{CC} = 2.5\text{V}$

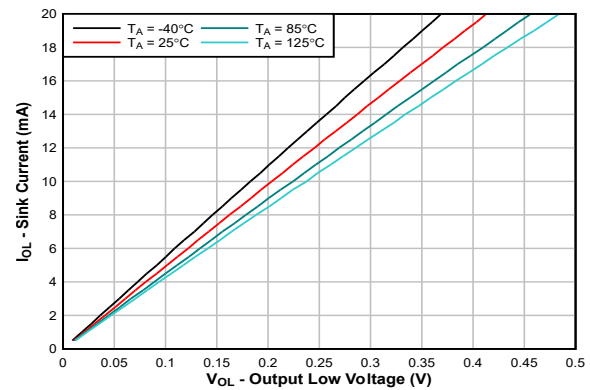


Figure 5-6. V_{OL} vs I_{OL} for $V_{CC} = 3.3\text{V}$

5.9 Typical Characteristics (continued)

$T_A = 25^\circ\text{C}$ (unless otherwise noted)

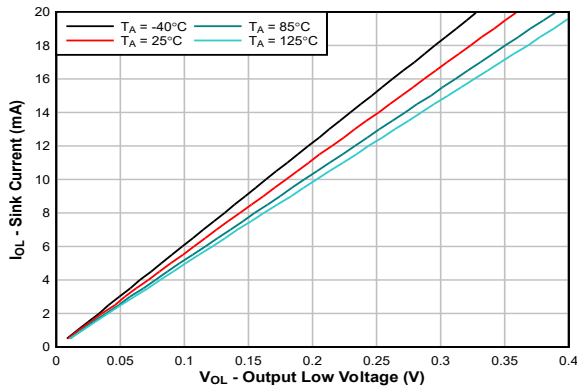


Figure 5-7. V_{OL} vs I_{OL} for $V_{CC} = 5.0\text{V}$

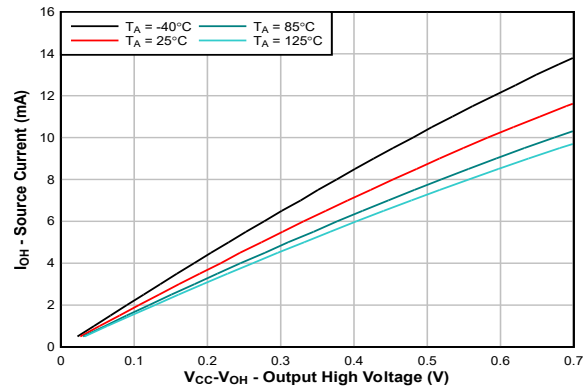


Figure 5-8. V_{OH} vs I_{OH} for $V_{CC} = 1.8\text{V}$

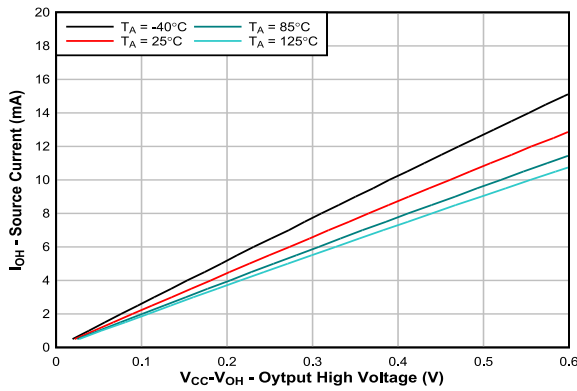


Figure 5-9. V_{OH} vs I_{OH} for $V_{CC} = 2.5\text{V}$

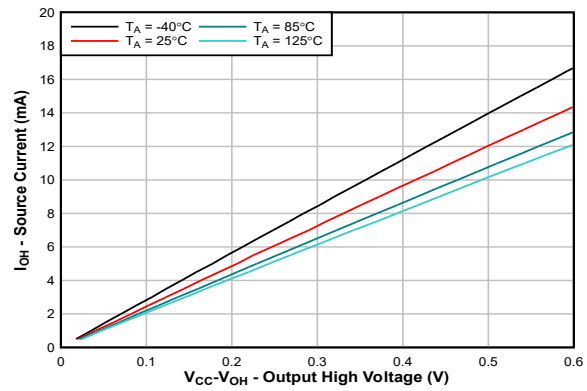


Figure 5-10. V_{OH} vs I_{OH} for $V_{CC} = 3.3\text{V}$

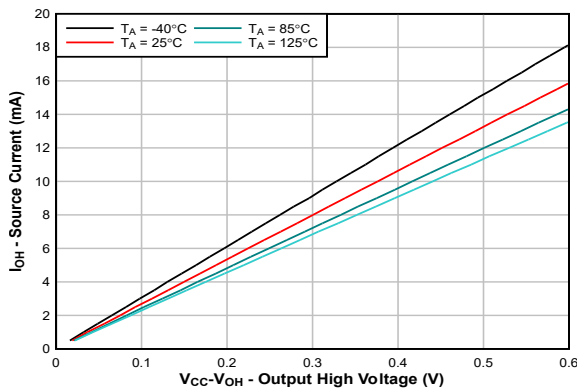


Figure 5-11. V_{OH} vs I_{OH} for $V_{CC} = 5.0\text{V}$

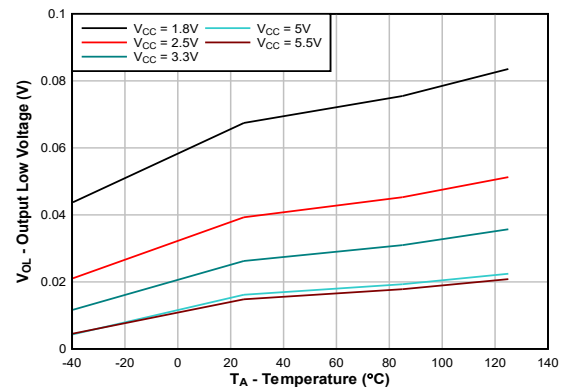


Figure 5-12. V_{OL} vs Temperature - $|I_{OL}| = 4\text{mA}$

5.9 Typical Characteristics (continued)

$T_A = 25^\circ\text{C}$ (unless otherwise noted)

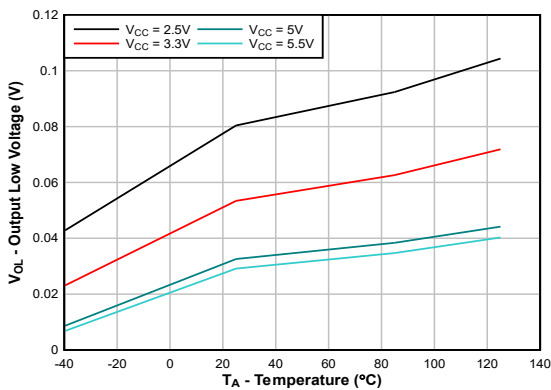


Figure 5-13. V_{OL} vs Temperature - $|I_{OL}| = 8\text{mA}$

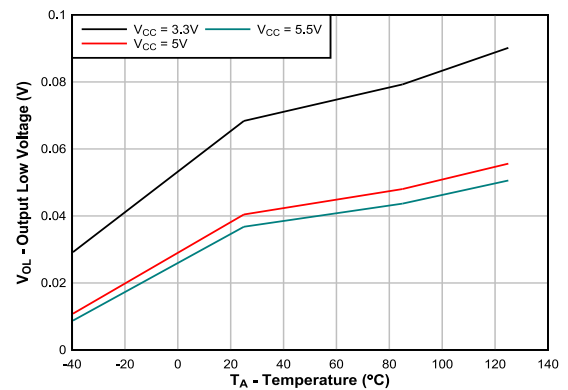


Figure 5-14. V_{OL} vs Temperature - $|I_{OL}| = 10\text{mA}$

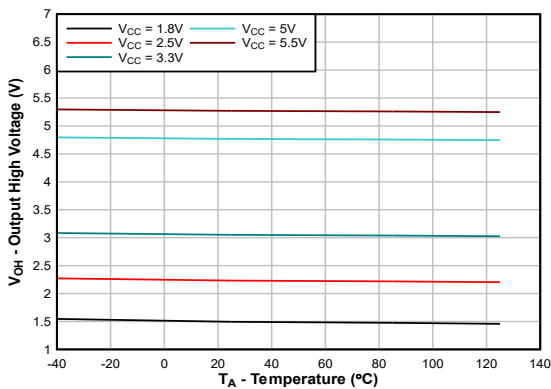


Figure 5-15. V_{OH} vs Temperature - $|I_{OH}| = 4\text{mA}$

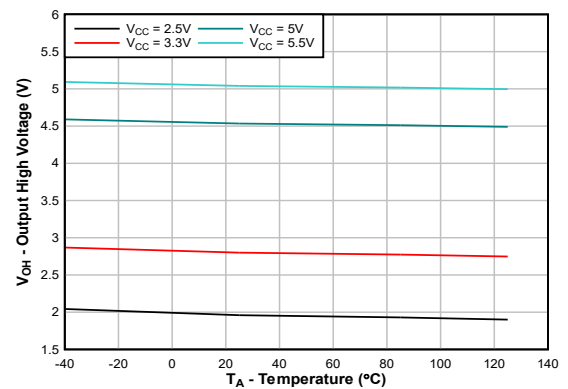


Figure 5-16. V_{OH} vs Temperature - $|I_{OH}| = 8\text{mA}$

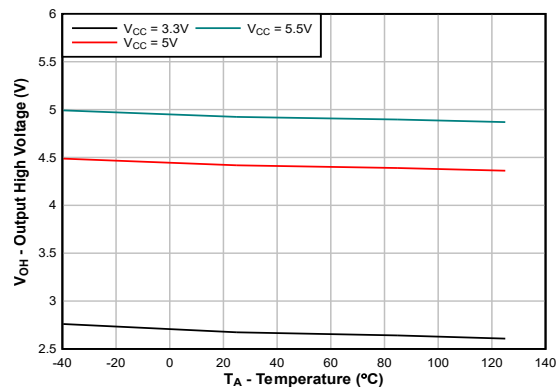


Figure 5-17. V_{OH} vs Temperature - $|I_{OH}| = 10\text{mA}$

6 Parameter Measurement Information

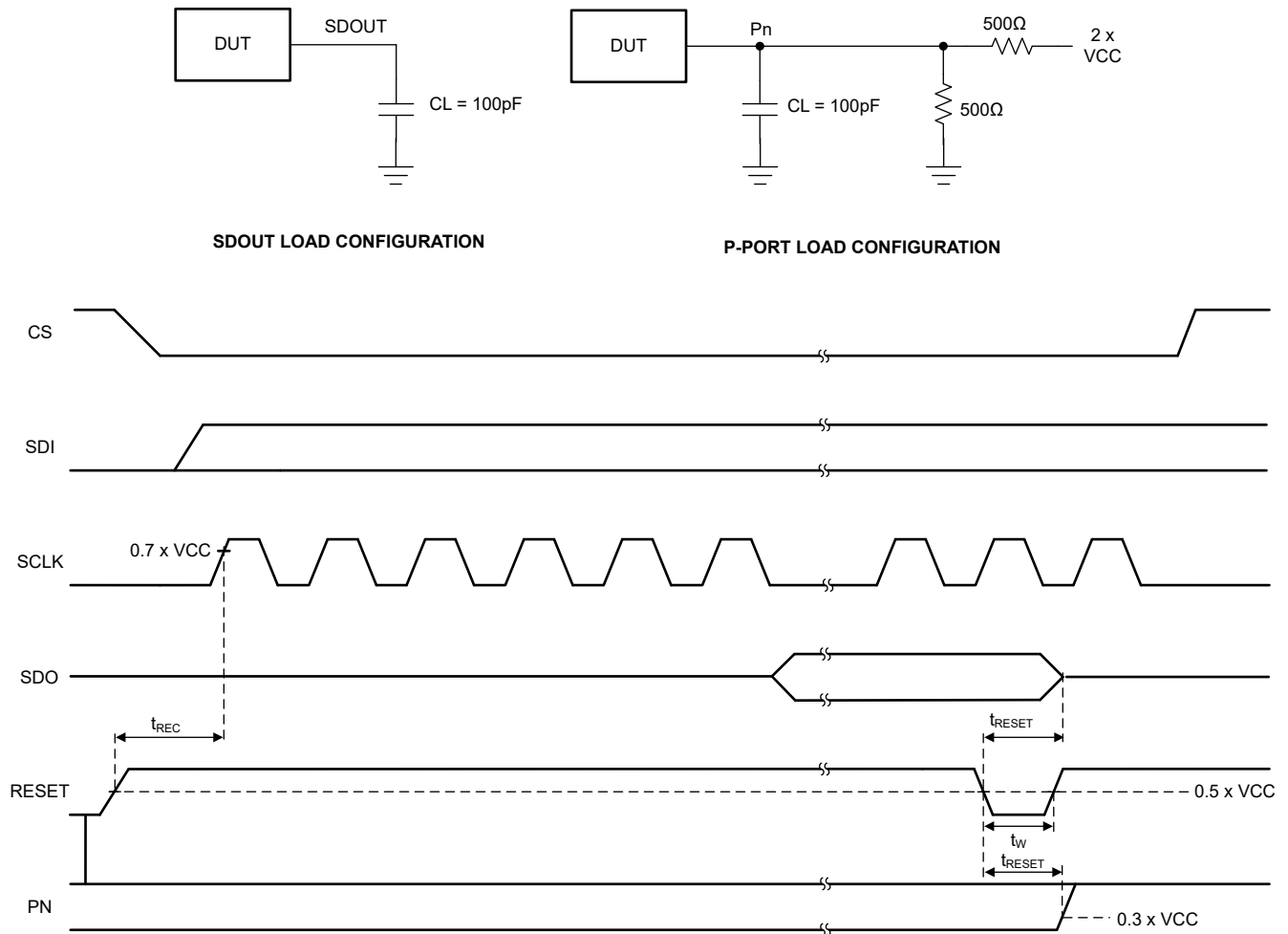


Figure 6-1. Reset Load Configuration

- A.
1. C_L includes probe and jig capacitance.
 2. All inputs are supplied by generators having the following characteristics: $PRR \leq 1$; $Z_o = 5$; $tr/tf \leq 3$.
 3. All parameters and waveforms are not applicable to all devices.

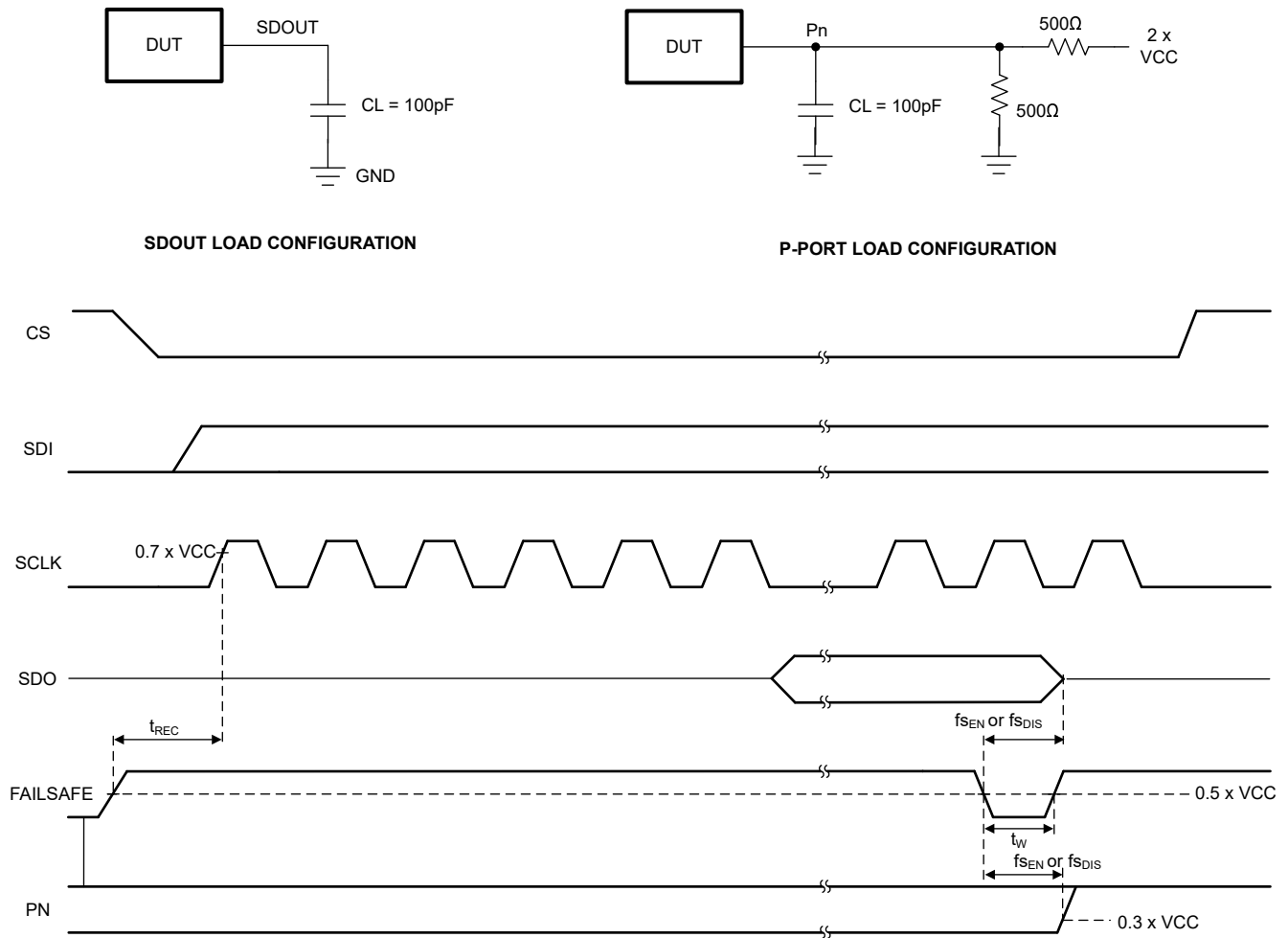


Figure 6-2. Fail-safe Load Configuration

- A.
1. C_L includes probe and jig capacitance.
 2. All inputs are supplied by generators having the following characteristics: $PRR \leq 1$; $Z_o = 5$; $t_r/t_f \leq 1$.
 3. FAILSAFE pin a shared pin with RESET pin.
 4. All parameters and waveforms are not applicable to all devices.

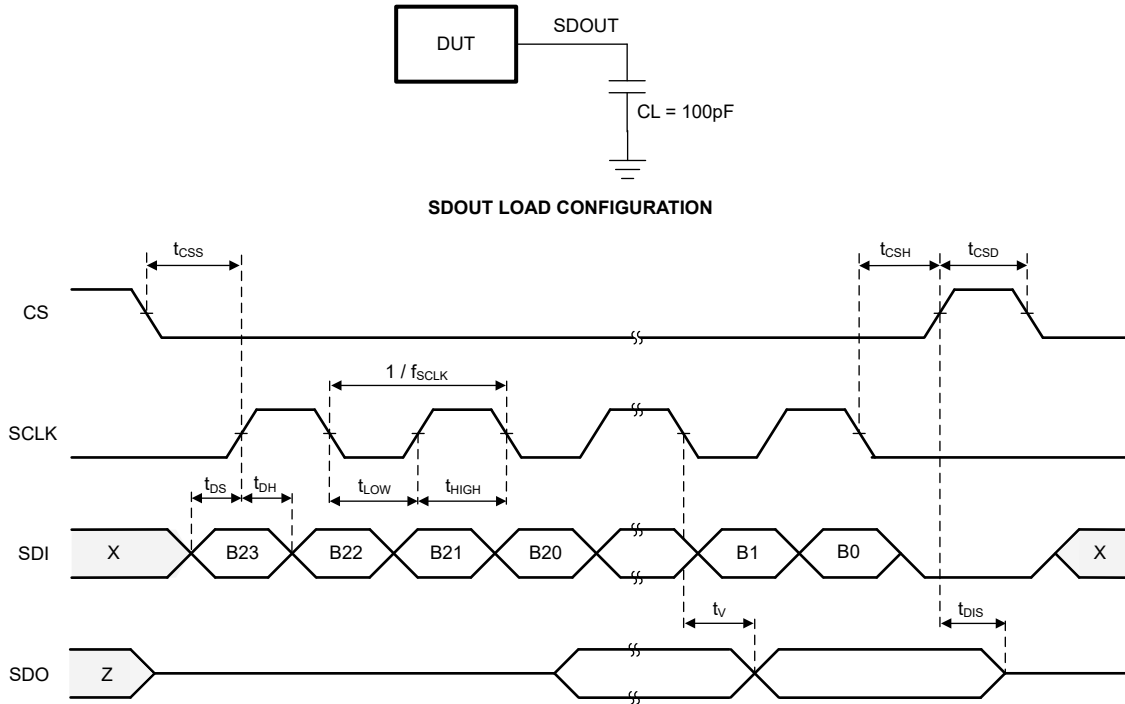


Figure 6-3. SPI Timing Diagram - Input

A. C_L includes probe and jig capacitance.

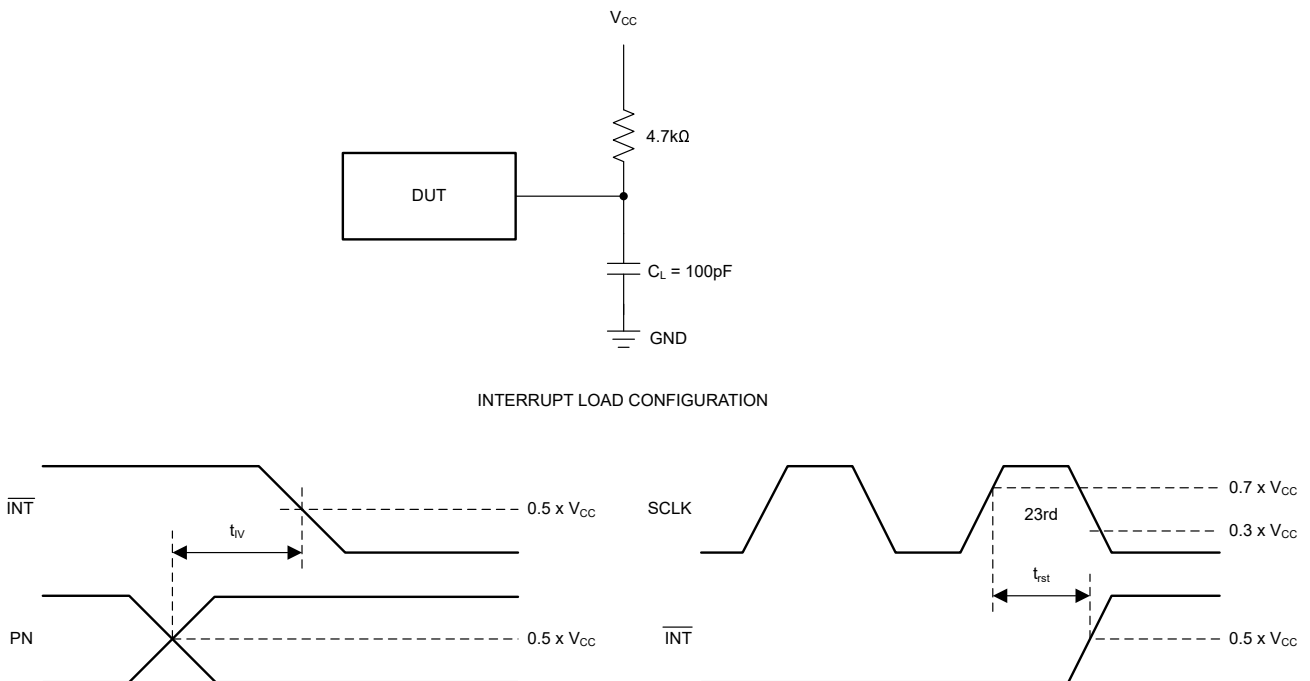
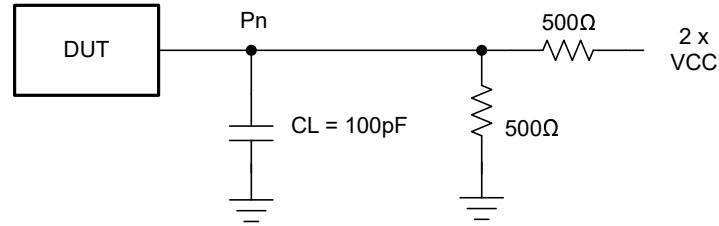
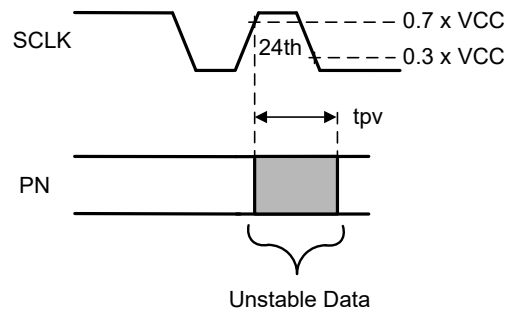


Figure 6-4. Interrupt Load Configuration

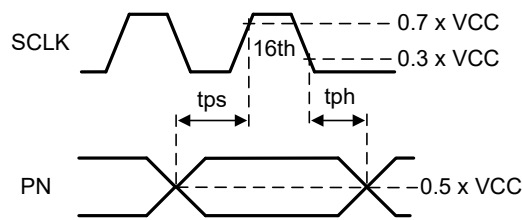
- A.
1. C_L includes probe and jig capacitance.
 2. All inputs are supplied by generators having the following characteristics: $PRR \leq 1$; $Z_o = 5$; $tr/tf \leq 1$.
 3. All parameters and waveforms are not applicable to all devices.



P-PORT LOAD CONFIGURATION



WRITE MODE ($\overline{R/W} = 0$)



READ MODE ($\overline{R/W} = 1$)

Figure 6-5. P-Port Load Configuration and Timing Waveforms

- A.
1. C_L includes probe and jig capacitance.
 2. t_{pv} is measured from $0.7 \times V_{CC}$ on SCLK to I/O (Pn) output.
 3. All inputs are supplied by generators having the following characteristics: $PRR \leq 1$; $Z_o = 5$; $t_r/t_f \leq 1$.
 4. All parameters and waveforms are not applicable to all devices.

7 Detailed Description

7.1 Overview

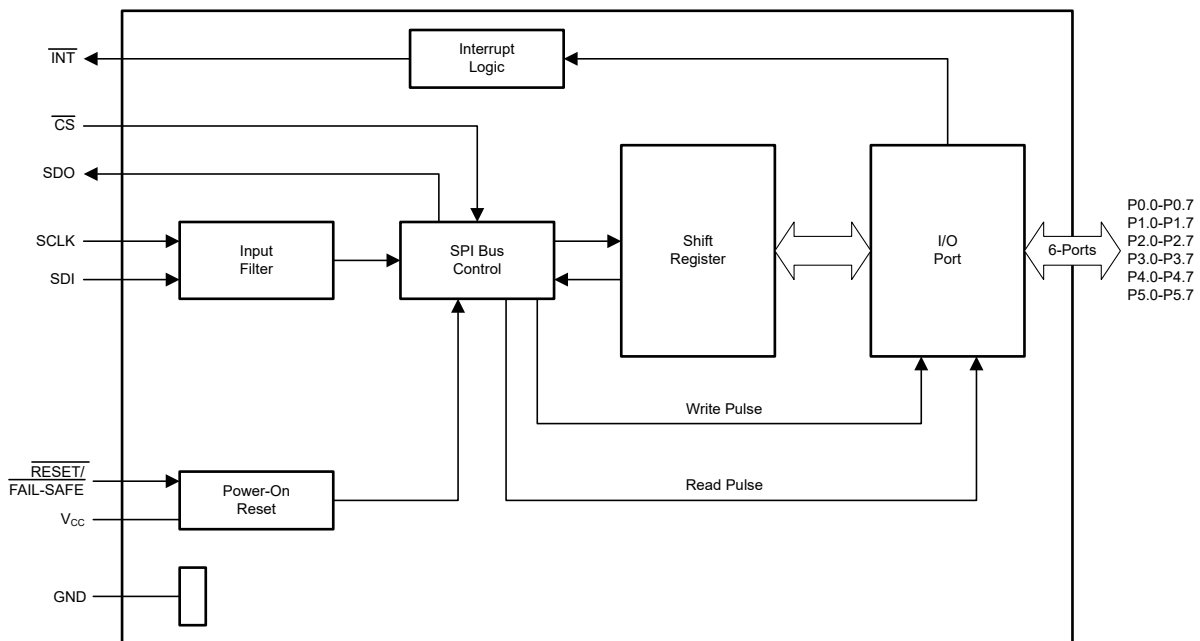
The TXE8148 digital core consists of 8-bit data registers, which allow the user to configure the I/O port characteristics. At power on or after a reset, the I/Os are configured as inputs. However, the system controller can configure the I/Os as either inputs or outputs by writing to the direction configuration registers. The data for each input or output is kept in the corresponding Input Port or Output Port register. The polarity of the Input Port register can be inverted with the Polarity Inversion register. All registers, except software reset register can be read by the system controller.

The TXE8148 has configurable I/O functionality which is specifically targeted to enhance the I/O ports. The configurable I/O features and registers include enabling or disabling pullup and pulldown resistors, bus-hold latching inputs, maskable interrupts, interrupt status register, and individually programmable open-drain or push-pull outputs. These configuration registers improve the I/O by increasing flexibility and allowing the user to optimize the design for power consumption and speed.

Other features of the device include an interrupt that is generated on the $\overline{\text{INT}}$ pin whenever an input port changes state. The device can be reset to the default state by applying a low logic level to the $\overline{\text{RESET}}$ pin, issuing a software reset command, or by cycling power to the device and causing a power-on reset. The TXE8148 open-drain interrupt ($\overline{\text{INT}}$) output is activated when any input state differs from the last read state and is used to indicate to the system controller that an input state has changed. The $\overline{\text{INT}}$ pin can be connected to the interrupt input of a processor. By sending an interrupt signal on this line, the device can inform the processor if there is incoming data on the remote I/O ports Bus without having to communicate through the SPI bus. The device remains a simple target device.

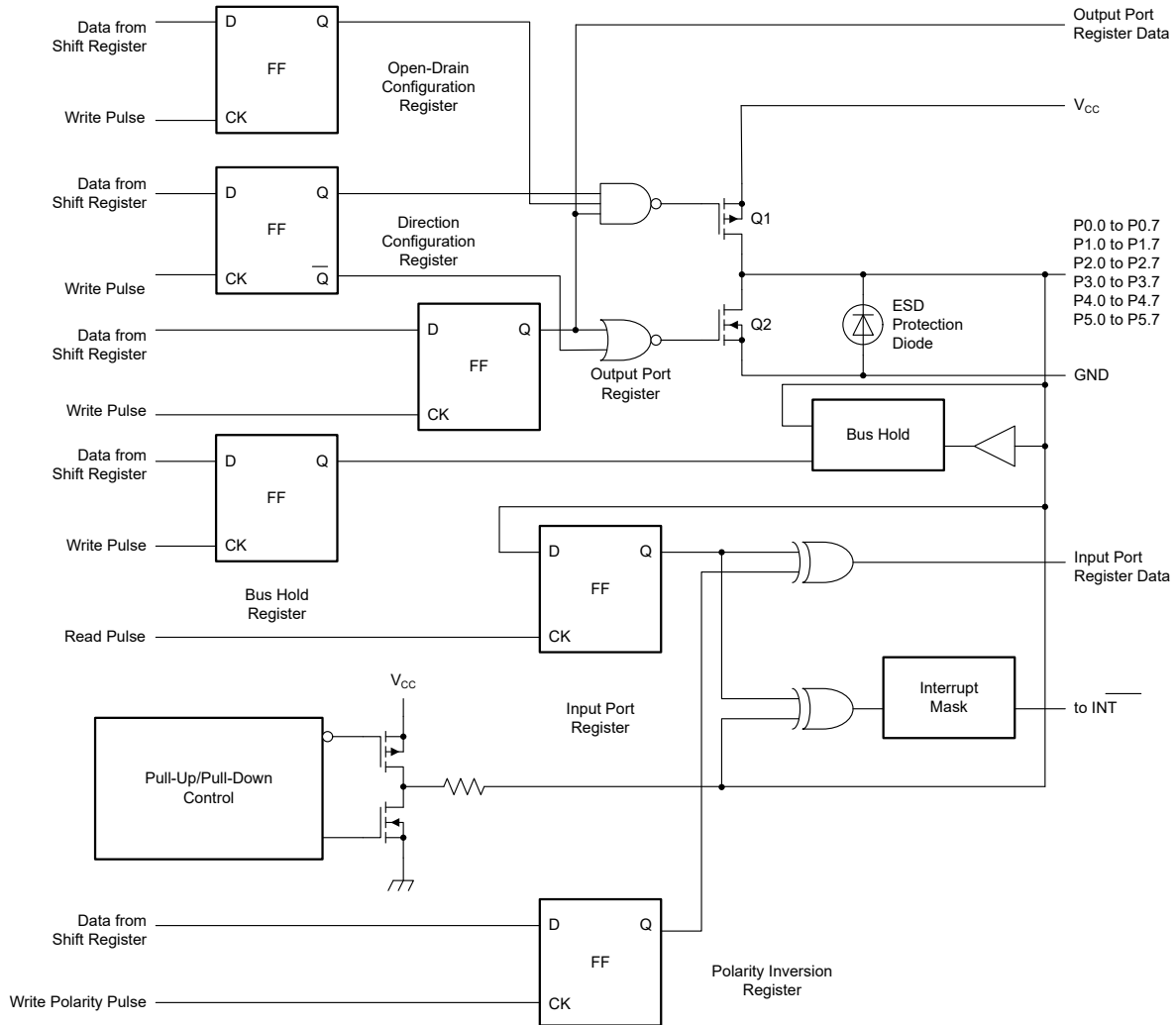
In the event of a timeout or other improper operation, the system controller resets the device by asserting a low on the $\overline{\text{RESET}}$ input pin or by cycling the power to the V_{CC} pin and causing a power-on reset (POR). A reset puts the registers in the default state and initializes the SPI state machine. The $\overline{\text{RESET}}$ feature and a POR cause the same reset, initialization to occur, but the $\overline{\text{RESET}}$ feature does so without needing to power down the device.

7.2 Functional Block Diagrams



All I/Os are set to inputs at reset.

Figure 7-1. Logic Diagram



On power up or reset, all registers return to default values.

Figure 7-2. Simplified Schematic of P0.0 to P5.7

7.3 Feature Description

7.3.1 I/O Port

When an I/O is configured as an input, FETs Q1 and Q2 are off (see Figure 7-2), which creates a high-impedance input.

If the I/O is configured as an output, Q1 or Q2 is enabled, depending on the state of the output port register. In this case, there are low-impedance paths between the I/O pin and either supply or GND. The external voltage applied to this I/O pin must not exceed the recommended levels for proper operation.

7.3.2 Interrupt Output (\overline{INT})

The TXE8148 devices generate an interrupt on any rising or falling edge of an input I/O, provided that the interrupt for that I/O is not masked. When an input pin state is different than the input read register state, the corresponding interrupt flag bit is set, and the \overline{INT} output is asserted. The \overline{INT} pin is open-drain and requires an external pullup resistor to V_{CC} use the interrupt feature, otherwise this pin can be left floating.

I/Os configured as outputs do not generate interrupts. Switching a pin from output to input can generate a fault interrupt if the actual pin level does not match the stored input port register value. If an I/O port is previously

in input state and detected an interrupt as switching to output, this action does not clear the interrupt flag. This action only masks the interrupt pin. Then when the port is reconfigured as input, the interrupt comes back.

With the following conditions, the interrupt status bits can be cleared and the $\overline{\text{INT}}$ pin deasserted.

- Hardware reset from $\overline{\text{RESET}}$ pin - this deasserts the interrupt temporarily as POR is going to assert the interrupt
- Entering fail-safe mode - this mode disables and de-assert the interrupt
- Reading Interrupt Flag Status Register
- Setting the corresponding bit as 1 in Interrupt Mask Register

There are four sources of interrupts in TXE8148:

1. **Smart Input Pin Interrupt:** Smart Interrupt is enabled or disabled at I/O port level by setting the corresponding port bit in the Smart Interrupt Register. If Smart Interrupt (the corresponding register bit as 0) is enabled and an interrupt is generated, the interrupt clears if the I/O state goes back to the initial logic state or the interrupt reads the Interrupt Flag Status Register. Refer to [Table 7-1](#) for the different interrupt clearing scenarios. To avoid missing the interrupt clear due to false IO toggle, TI strongly recommends to enable the glitch filter enable in the Input Glitch Filter Enable Register.
2. **Regular Input Pin Interrupt:** When Smart Interrupt is disabled (the corresponding register bit as 1) in the Smart Interrupt Register, the I/O state going back to the initial logic state cannot clear the interrupt, only reading the Interrupt Flag Status Register clears the interrupt.

Table 7-1. Interrupt Flag Clearing Scenarios for Smart Interrupt

SMART INTERRUPT	$\overline{\text{CS}}$ STATE WHEN IO INPUT CHANGES	INTERRUPT FLAG CLEARS
Disable	$\overline{\text{CS}} = \text{High}$	$\overline{\text{CS}}$ to be low and SPI reading Interrupt Flag Status Register
Disable	$\overline{\text{CS}} = \text{Low}$	Reading Interrupt Flag Status Register
Enable	$\overline{\text{CS}} = \text{High}$	<ol style="list-style-type: none"> IO state going back to the initial state. $\overline{\text{CS}}$ to be low and SPI reading Interrupt Flag Status Register.
Enable	$\overline{\text{CS}} = \text{Low}$	<ol style="list-style-type: none"> IO states going back to the initial state shall automatically clear the interrupt flag at least 30ns after $\overline{\text{CS}}$ becomes high. Reading Interrupt Flag Status Register.

3. **POR Interrupt :** the POR fault bit is set in the Fault Status Register for each POR recovery, which also generates an interrupt. The interrupt is only cleared when the Fault Status Register is read.
4. **Fail-safe Redundancy Failure Interrupt:** When the fail-safe redundancy check is enabled, and if any fail-safe redundancy check failure occurs, a fail-safe sync fault bit is set in the Fault Status Register. This action also generates an interrupt. The interrupt is only cleared when the Fault Status Register is read.

Interrupt Masking

Interrupts from all input I/Os are unmasked by default. To mask an interrupt, the corresponding I/O bit needs to be set in the interrupt mask register. The interrupt generated by POR recovery cannot be masked.

If the state of an input I/O is changed and the corresponding bit in the Interrupt mask register is set to 1, the interrupt is masked and the $\overline{\text{INT}}$ pin is not asserted. The corresponding bit in the interrupt flag status register also stays at 0 and is blocked by the interrupt mask bit.

The interrupts generated by fail-safe redundancy check fail is disabled if the fail-safe redundancy check enable bit is 0.

Multiple ports can be configured for interrupt masking at the same time by using multi-port command.

7.3.3 Reset Input (\overline{RESET})

The \overline{RESET} input can be asserted to initialize the system while keeping the V_{CC} supply at the operating level. A reset can be accomplished by holding the \overline{RESET} pin low for a minimum of t_W . The TXE8148 registers and SPI state machine are changed to the default state after \overline{RESET} is set LOW. When \overline{RESET} is set HIGH, the I/O levels at the P port can be changed externally or through the controller. This input requires a pullup resistor to V_{CC} , if no active connection is used. When \overline{RESET} is toggled the input port register is updated to reflect the state of the GPIO pins.

7.3.4 Bus Hold

TXE8148 supports bus hold (bus-keeper) function on all input I/Os which is very useful when the system has to go into low-power or standby state as external pullup or pulldown corresponding to the low-power state are no longer required. By default the feature is disabled and must be enabled through software programming.

On enabling the bus hold feature, the I/O holds the last known active level on the I/O. If the external device drives a logic 1, TXE8148 activates the internal bus hold pullup. When the controller goes into low-power state making the output pins high-impedance, the internal bus hold pullup holds the I/O high, thus avoiding a floating input. The same happens when the external device is drives a logic 0, with the TXE8148 activating the internal bus hold pulldown. This also makes sure that the TXE8148 does not generate an interrupt due to noise on the floating inputs, as the device is held in the last known state.

When the bus hold feature is enabled, the external driver must make sure that the current sink or source (indicated by the + or – sign on the specification) due to leakage or other circuits on the board must meet the sustaining or override current. As an example, if the bus hold is activated in high state, then the external devices on the line must not source current from TXE8148 that exceeds the $I_{BHH(MAX)}$ limit. If the limit is exceeded, the voltage can drop below the $V_{IL(MAX)}$ on the corresponding port pin and trigger the bus-hold pulldown. Similarly, if the bus-hold has to be overridden, then the external devices must be able to sink $I_{BHO(MIN)}$ current from the TXE8148, to make sure that the device releases the bus-hold.

The software must make sure that the pullup or pulldown selection must not be used along with the bus-hold feature to avoid additional leakage current in the device.

7.3.5 Fail-safe Mode

The SPI controller has the option to set TXE8148 to be in a fail-safe state by programming the Fail-safe Enable Register to enable this feature and change the functionality of the pin from reset to fail-safe.

This register can get cleared during a POR event or other fault scenarios. The SPI controller has to rewrite this register every time if there is a fault scenario which generates an interrupt to the SPI controller. After the interrupt is generated, the SPI controller can read the Fault Status Register to understand the source of the interrupt.

The bit 0 in Fail-safe Enable Register must be 1 to configure TXE8148 to be fail-safe mode.

Two Device Configuration Registers have to be written to program I/O configuration to ensure redundancy. If either of these registers get corrupted, and the contents don not match, an interrupt is generated.

For example, if setting I/O pin P0.1 to be output and high under fail-safe mode, the sequence to configure fail-safe mode:

1. Configure bit 0 in the Fail-safe Enable Register 1 as 1
2. Configure bit 0 in the Fail-safe Enable Register 2 as 1
3. Set bit 1 (P0.1) in port 0 of Fail-safe Direction Configuration Register 1 for Port-0 to be 1
4. Set bit 1 (P0.1) in port 0 of Fail-safe Direction Configuration Register 2 for Port-0 to be 1
5. Set bit 1 (P0.1) in port 0 of Fail-safe Output Register 1 for Port-0 to be 1
6. Set bit 1 (P0.1) in port 0 of Fail-safe Output Register 2 for Port-0 to be 1
7. Set bit 0 in Fail-safe Redundancy Check Register to be 1
8. Assert $\overline{RESET}/\overline{FAIL-SAFE}$ pin

7.3.6 Software Reset Call

The software reset call is an SPI bus transaction to the TXE8148 device, to write to the software reset register, that instructs the device to be reset to the power-up default state. This action allows customer application to reset the device without having to either power cycle the device or assert, de-assert the $\overline{\text{RESET}}$ pin. This is especially useful if the $\overline{\text{RESET}}$ pin is being used in fail-safe mode.

7.3.7 Daisy Chain

Multiple TXE8148 devices can be connected in a daisy chain configuration as shown in example implementation [Figure 7-3](#), to expand the number of I/O ports supported. In the daisy chain mode of operation, the SDO of the controller is connected to the SDI of the first TXE8148. The SDO of the first TXE8148 is in turn connected to the SDI of the next TXE8148. This connection can be done until the last TXE8148, where the SDO is connected to the SDI of the controller. When connecting devices in a daisy chain, sampling of the SDO from the last devices may require the SCLK frequency to be reduced, because the data valid time from each of the devices in the chain add up, along with the propagation delay of the SCLK to the furthest device.

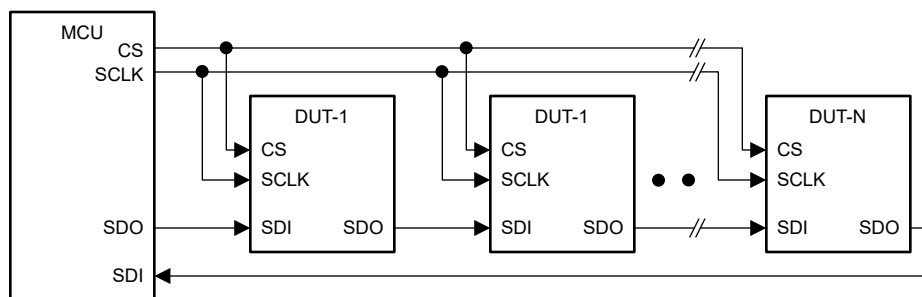


Figure 7-3. SPI Daisy Chain Diagram

The controller transmits a header to auto-configure the daisy chain of operation, followed by the register address of the devices in the chain with the register address of the farthest device in the chain (the device furthest from the controller SDI and closest to the controller SDO) first. The data is transmitted after the address with data for the furthest device first. When receiving, the data from the furthest device is received first.

Refer to [Figure 7-4](#) for the frames of daisy chained transaction. The same sequencing is repeated throughout the entire chain until the final device is reached.

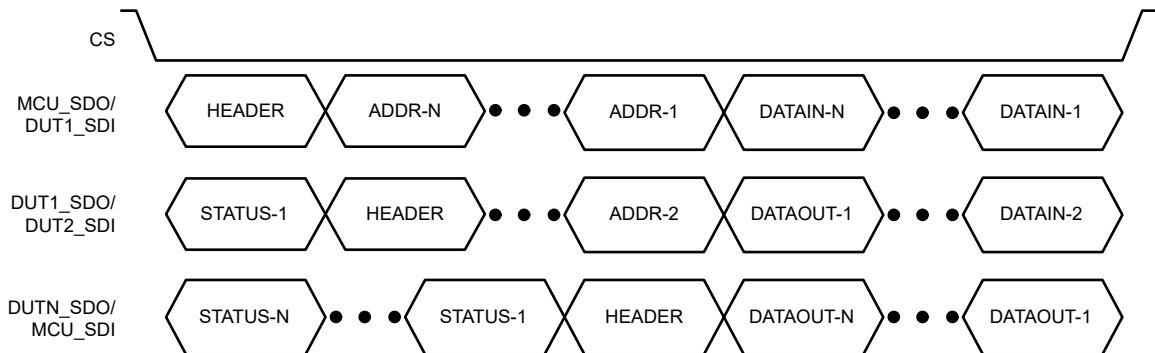


Figure 7-4. SPI Daisy Chain Data Frame

Header segment

Bit-15 and 14 in Header segment are the Header ID that is used by the device controller to detect that a header segment is being received. The Header ID bits have the value 01 to indicate this is a Header segment.

Bit-13 to 5 are reserved and bits 4 to 0 indicate the number of devices in the chain. The maximum number of devices that can be daisy chained is 31.

Address segment (Register Address)

Bit 15 indicates SPI mode of operation (1 = Read operation 0 = Write operation). Refer to the first and second byte in [Figure 7-5](#) for the register address.

Status segment

Bit-15 and 14 indicate a status segment and set to 11 to indicate status segment. Bit 13 to 8 indicate the fault status register and bit 7 to 0 are always transmitted as 0.

7.3.8 Feature Register Mapping

The TXE8148 device registers are implemented as a register bank, where the actual register address for a port is formed by concatenating the feature address, 0 and the port number as shown in [Figure 7-5](#). This allows the customer software to configure or read a specific feature for all ports more effectively with a burst transfer.

[Table 7-2](#) lists the registers as per the feature address and support for multi-port.

Table 7-2. TXE8148 Feature Map

FEATURE ADDRESS					REGISTER NAME	MULTI PORT	ACCESS
B20	B19	B18	B17	B16			
0	0	0	0	0	Scratch Register	No	RW
0	0	0	0	1	Device_ID	No	R
0	0	0	1	0	Input Port Register	Yes	R
0	0	0	1	1	Output Port Register	Yes	RW
0	0	1	0	0	Direction Configuration Register	Yes	RW
0	0	1	0	1	Polarity Inversion Register	Yes	RW
0	0	1	1	0	Push Pull / Open Drain Selection Register	Yes	RW
0	1	0	0	0	Pull Up or Pull Down Enable Register	Yes	RW
0	1	0	0	1	Pull Up or Pull Down Selection Register	Yes	RW
0	1	0	1	0	Bus Holder Register	Yes	RW
0	1	0	1	1	Smart Interrupt Register	No	RW
0	1	1	0	0	Interrupt Mask Register	Yes	RW
0	1	1	0	1	Input Glitch Filter Enable Register	No	RW
0	1	1	1	0	Interrupt Flag Status Register	No	R
0	1	1	1	1	Interrupt Port Status Register	No	R
1	0	0	1	0	Fail-safe Enable Register 1	No	RW
1	0	0	1	1	Fail-safe Enable Register 2	Yes	RW
1	0	1	0	0	Fail-safe Direction Configuration Register 1	Yes	RW
1	0	1	0	1	Fail-safe Direction Configuration Register 2	Yes	RW
1	0	1	1	0	Fail-safe Output Register 1	Yes	RW
1	0	1	1	1	Fail-safe Output Register 2	Yes	RW
1	1	0	0	0	Fail-safe Redundancy Check Register	No	RW
1	1	0	0	1	Fault Status Register	No	R
1	1	0	1	0	Software Reset Register	No	WO

7.4 Device Functional Modes

7.4.1 Power-On Reset

When powering the device from 0V is applied to V_{CC} , an internal power-on reset holds the TXE8148 in a reset condition until the supply has reached V_{POR} . At that time, the reset condition is released, and the TXE8148 registers and SPI state machine initializes to the default states. After that action, V_{CC} must be lowered to below V_{PORF} for time $t_{TRR_POR_50}$ and back up to the operating voltage for a power-reset cycle.

7.5 Programming

7.5.1 SPI

The TXE8148 devices use an SPI to set device configurations, operating parameters and read out diagnostic information. The SPI protocol uses three inputs and one output; serial clock (SCLK), active LOW chip select (\overline{CS}), serial data in (SDI) and serial data out (SDO). \overline{CS} must be driven low before clock pulses and data into the device. When \overline{CS} is high, the device ignores all activity on SCLK and SDI.

The TXE8148 devices support SPI mode 0 (CPOL = 0, CPHA = 0). The clock (SCLK) is low when idle. Data is sampled on the rising edge of SCLK and changed on the falling edge.

Besides SPI bus with independent chip select, daisy chain configuration is also supported in TXE8148. Daisy chain configuration allows multiple peripherals to be connected in series, with the output of one device feeding into the input of the next. Daisy chain is beneficial to reduce the number of \overline{CS} lines, as only one is needed for the entire chain. Data is shifted through all devices in the chain during each clock cycle.

7.5.2 SPI Data Format

The length of TXE8148 SPI data word is $[16 + (N \times 8)]$ bits (N is the number of data bytes to write). For a single byte write or read, 24 bits of data are sent and received MSB first. SPI data must be stable during the rising edge of SCLK where sampled.

The data format on the SDI pin from the controller to TXE8148 is shown in [Figure 7-5](#).

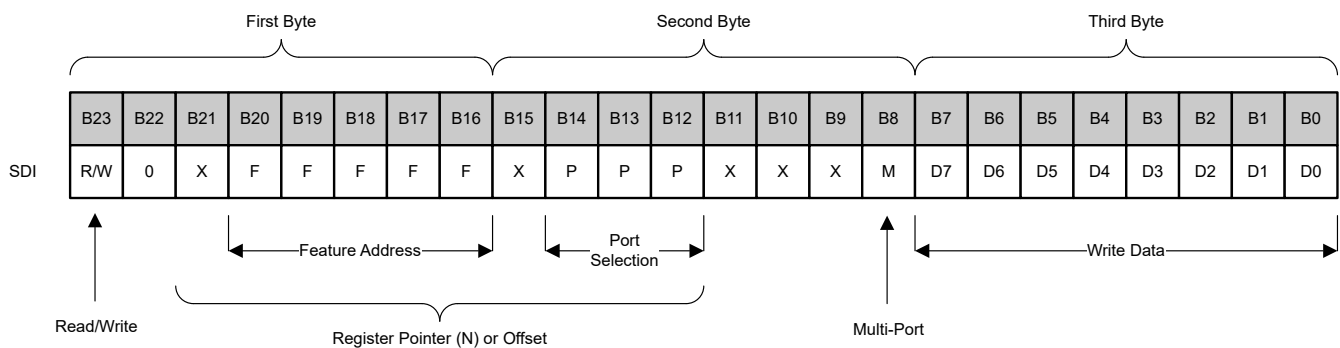


Figure 7-5. TXE8148 SPI SDI Data Frame Format

The first and second byte are mandatory portion of the frame. The B23 bit in the SDI frame is R/\overline{W} , where 1 is for a read operation and 0 is for a write operation. The B22 bit is always 0 for a point-to-point connection (non-daisy chain topology). B20 to B16 correspond to the feature register of the port. B14 to B12 are for selecting the port. The B8 bit is used for multi-port operation, and must be set to 1 only if the multi-port function is used. The bits B21 to B12 are referred to as the register pointer (or offset in the register table) for the device.

Third byte onwards, the 8-bit write data is shifted in and written to the register indicated by the register pointer. For single transfer $N = 1$ while for a burst mode $N > 1$. For a burst transfer, the register pointer is internally incremented and the shifted data written to the updated internal register pointer. Any bit which is not defined in the frame must be sent as 0 for future compatibility.

[Figure 7-6](#) shows the data format on the SDO pin from the TXE8148 to the controller.

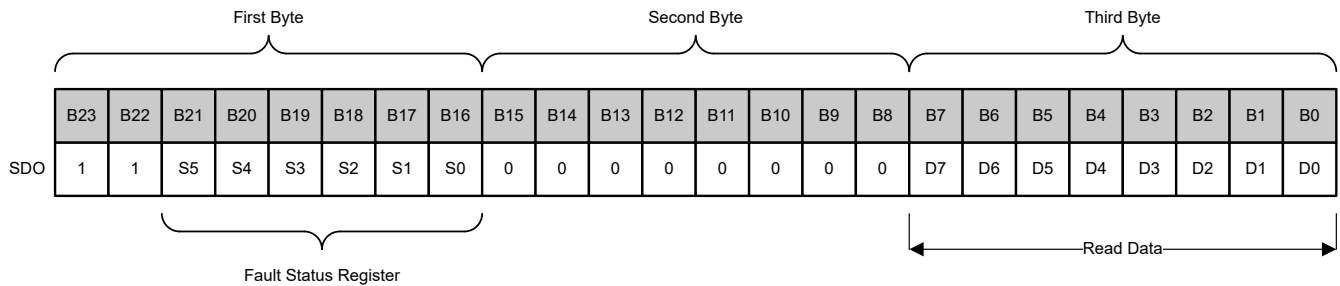


Figure 7-6. SPI SDO Data Format

The first and second byte are standard for any SPI bus communication operation. The B23 and B22 are set as 1 to indicate valid status. B21 to B16 are the status bit from the fault status register. The second byte is always set to 0. Third byte onwards the 8-bit read is shifted out for the register pointer shifted into the device on the SDI pin. For a burst transfer ($N > 1$) the register pointer is internally incremented and the data shifted out.

7.5.3 Burst Mode

In burst mode transactions, the initial register pointer is specified by the controller device and sent to the peripheral. For subsequent accesses, the register pointer is automatically incremented to the next valid address (second address byte) corresponding to the next port. This automatic address increment continues as long as the CS remains active low and SCLK pulses are received by the peripheral device.

As the burst mode transaction continues sequentially, the register pointer automatically advances the address. If there is no valid register corresponding to the updated register pointer or the register is a read-only type, the data written is ignored. Similarly on the SDO, the peripheral outputs 0s for a register, which is not mapped in the address space of the device.

TI strongly recommends to use burst mode when configuring or accessing the same register for multiple ports together to improve the effective data throughput on the SPI bus.

7.5.4 SPI Write

SPI write operation is used to send data from the controller device to the peripheral device. SPI write can be a single byte write as shown in Figure 7-7 or burst write as shown in Figure 7-8 where multiple bytes can be written to the device. The operation is performed over the SPI bus, where the controller device drives \overline{CS} low, generates the serial clock (SCLK) and sends data to the peripheral. SPI Write is commonly used to configure peripherals, send control commands, or transfer data. During a SPI single write, the device sends the last data from the register indicated by the register pointer. On a SPI burst write, the address is automatically incremented and the device sends the last data from the updated register pointer.

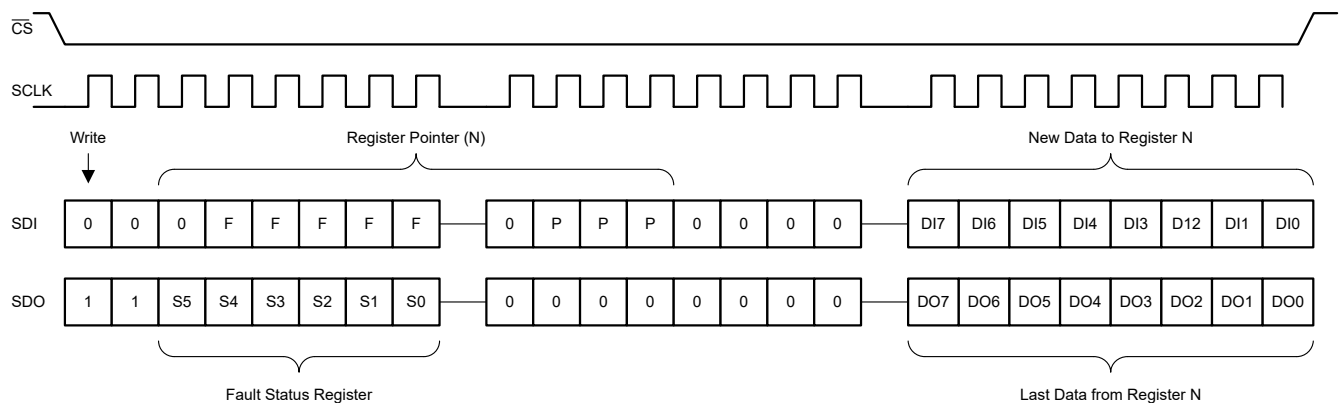


Figure 7-7. SPI Single Write

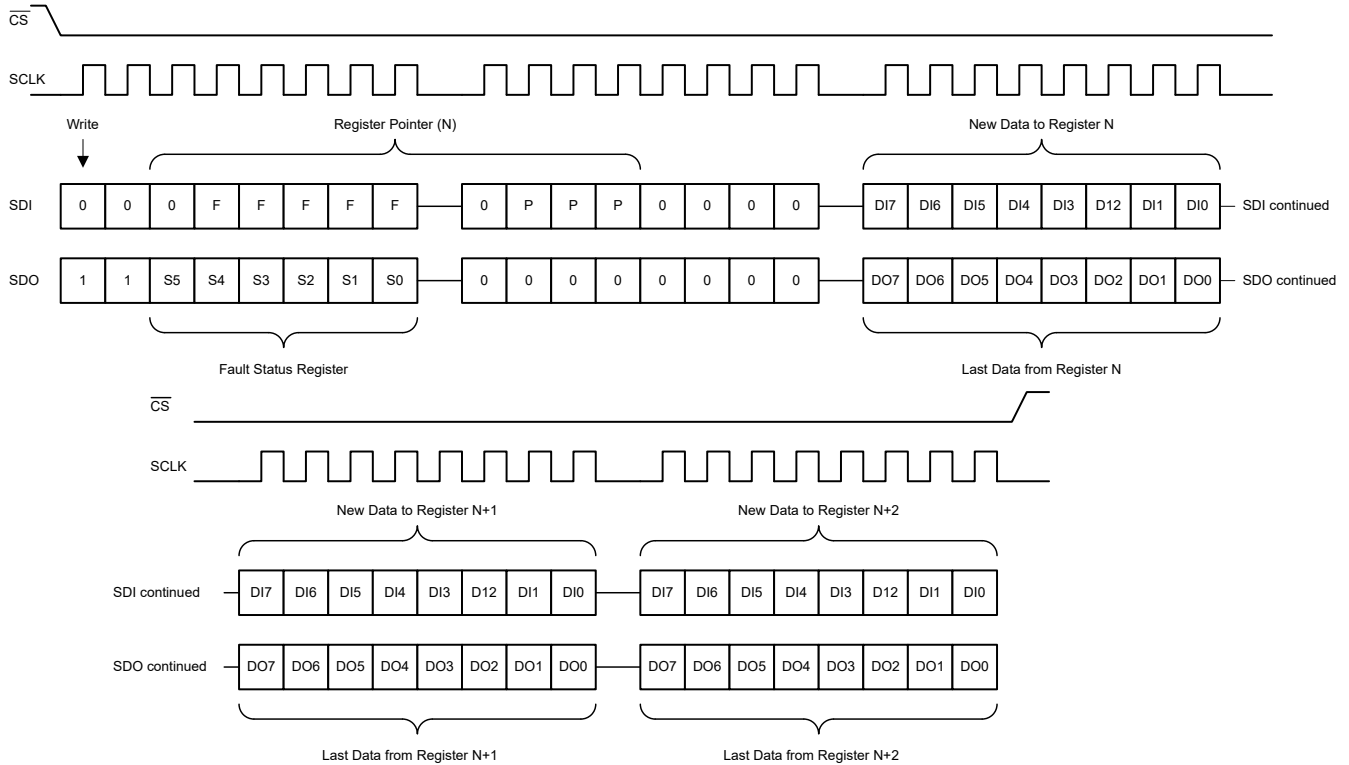


Figure 7-8. SPI Burst Write

SPI Write Steps:

1. Drive \overline{CS} low. This action enables the internal shift register.
2. Shift 24 bits of data into the device in a MSB first fashion, MSB bit . Data must be stable during the rising edge of SCLK.
3. The MSB bit must be a '0' indicating a write operation.
4. On the SDO pin, the device sends the first 2 bits as 2'b11 (indicating a status segment). Followed by 6 bits are the Bit 5 to 0 of the Fault status register. The next 8 bits are all 0s followed by 8-bit current data in the register selected by register pointer.
5. After the last bit of data is transferred, drive SCLK low if there is no more data to be transferred.
6. The previous content of the register is sent out on SDO as the data byte is driven on SDI.
7. De-assert \overline{CS} (drive high) to end the write cycle.

7.5.5 SPI Read

The SPI read operation for TXE8148 is used to retrieve data from a specific register in single read mode as shown in [Figure 7-9](#) or sequential registers in burst mode as shown in [Figure 7-10](#).

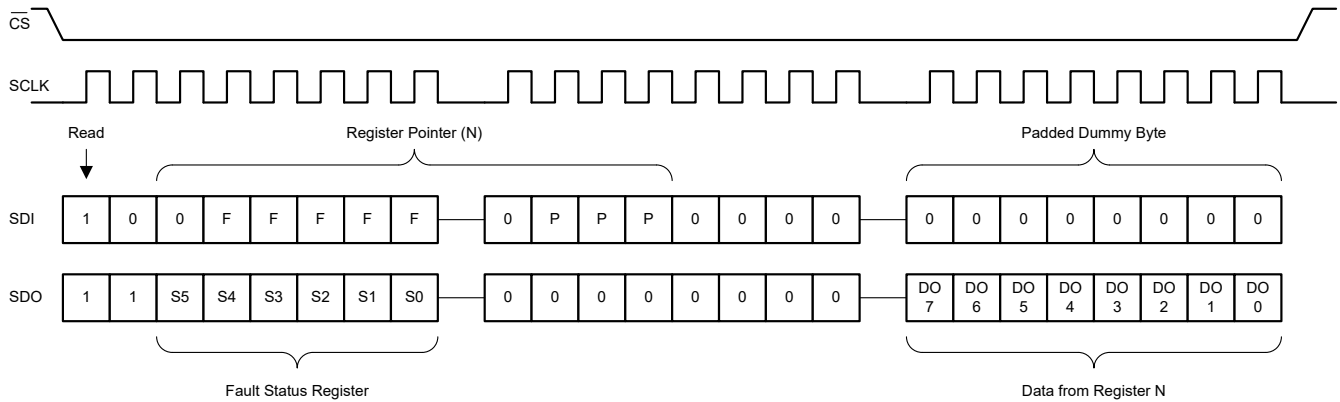


Figure 7-9. SPI Single Read

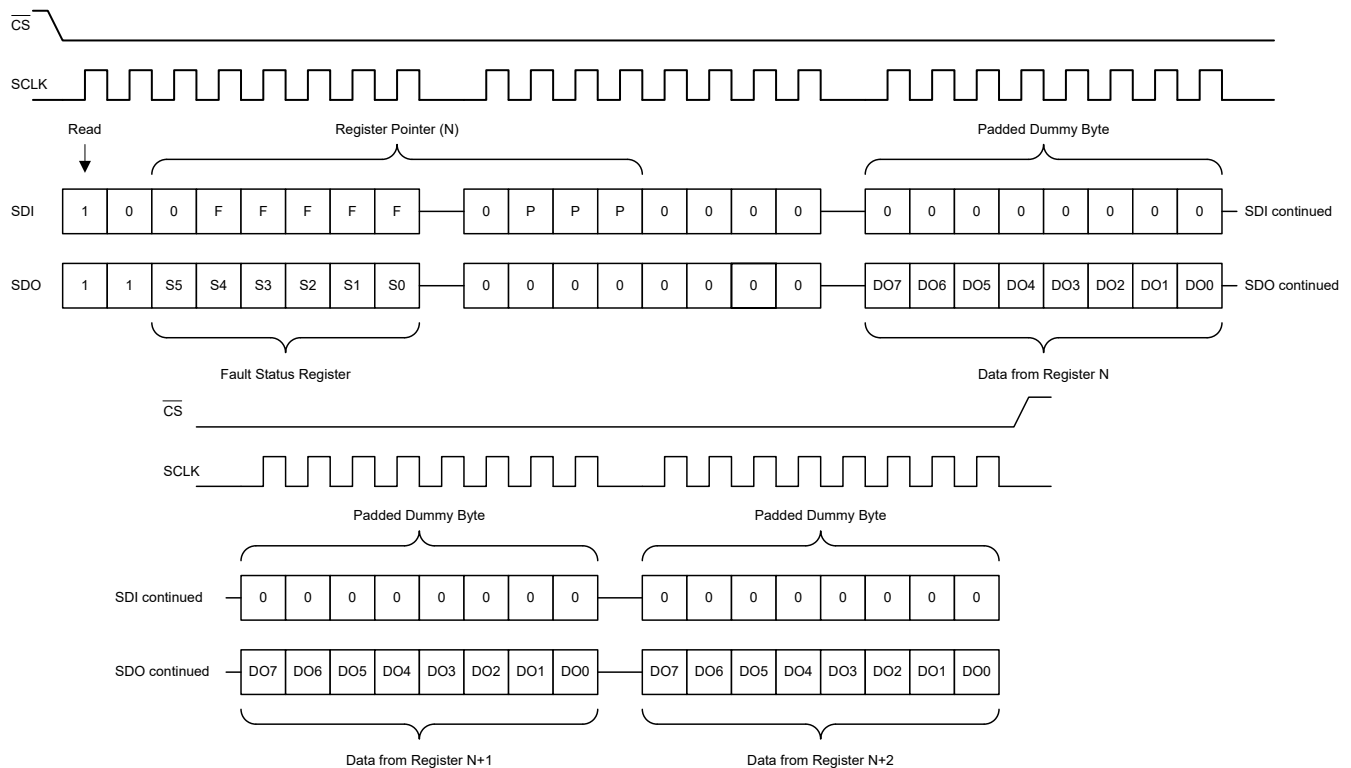


Figure 7-10. SPI Burst Read

SPI Read Operation Steps:

1. Drive \overline{CS} low. This action enables the internal shift register.
2. Shift 24 bits of data into the device in a MSB first fashion. Data must be stable during the rising edge of SCLK
3. The MSB bit must be a '1' indicating a read only transfer.
4. The third data byte is NOP (no operation) which is dummy data byte.
5. On the SDO pin, the device sends the first 2 bits as 2'b11 (indicating a status segment), followed by 6 bits that are the Bit 5 to 0 of Fault status register. The next 8 bits are all 0s followed by 8-bit current data in the register selected by register pointer.
6. After the last bit of data is transferred, drive SCLK low if there is no more data to be transferred.
7. De-assert \overline{CS} (drive high) to end the read cycle.

7.5.6 SPI Daisy Chain

In daisy chain mode of operation, the SPI transaction from the controller consists of three types of segments: Header (16-bit), Address (Nx16-bit,) and Data (Nx8-bit). As the TXE8148 receives the same and transmits to the next device in the chain, the device appends a Status (16-bit) to the transaction, while removing the Address segment and replacing the Data segment that corresponds to the position in the chain.

Figure 7-11 shows an example of a daisy chain write for two devices in a daisy chain. The header byte is transmitted with the chain length (CL) value of 2, followed by the address for DUT2, DUT1 and then the 8-bit data-in for DUT2, DUT1. As the first device shifts the data out, the device transmits the status segment, followed by the original header, address for DUT2, data-out from DUT1 and data-in for DUT2. The final device in the chain sends the status, followed by status for DUT1 and finally data-out for DUT2, DUT1.

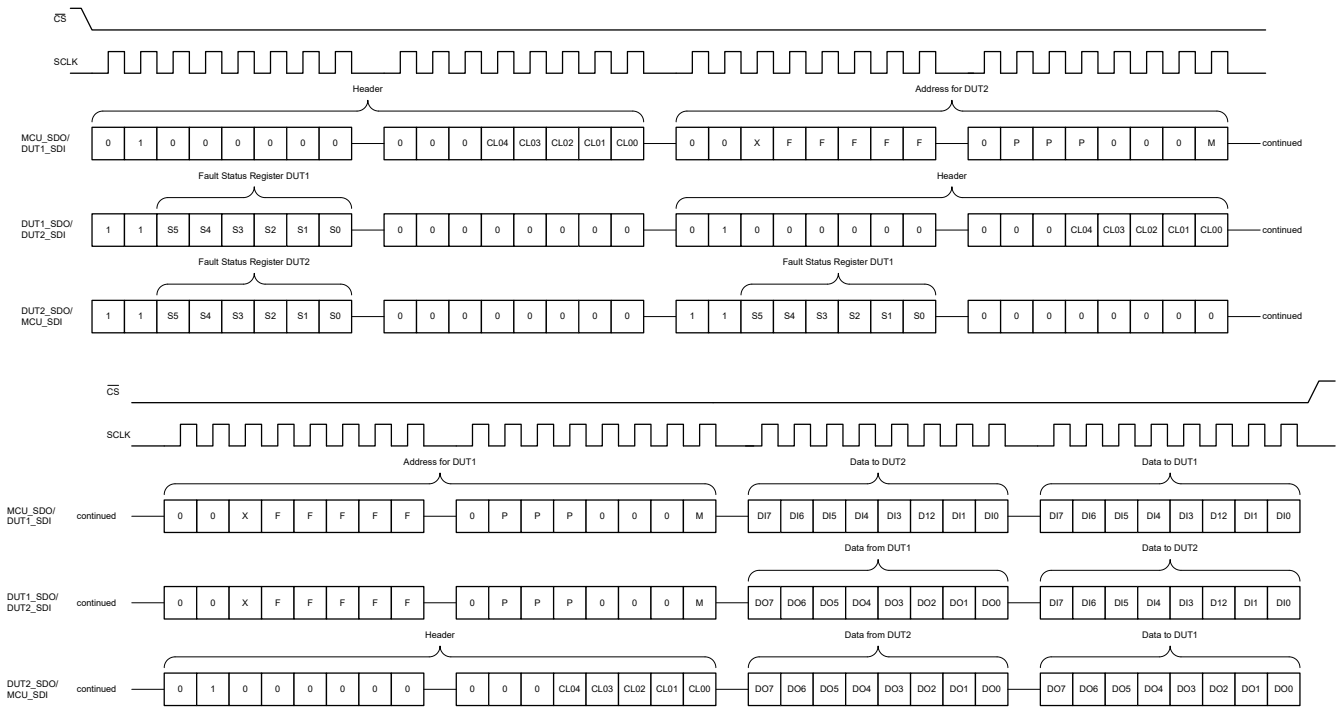


Figure 7-11. SPI Daisy Chain Write

Figure 7-12 shows an example of a daisy chain read for two devices in daisy chain. The header byte is transmitted with the chain length (CL) value of 2, followed by the address for DUT2, DUT1 and then the 8-bit dummy data-in for DUT2, DUT1. As the first device shifts the data out, the device transmits the status segment, followed by the original header, address for DUT2, data-out from DUT1 and dummy data-in for DUT2. The final device in the chain sends the status, followed by status for DUT1 and finally data-out for DUT2, DUT1.

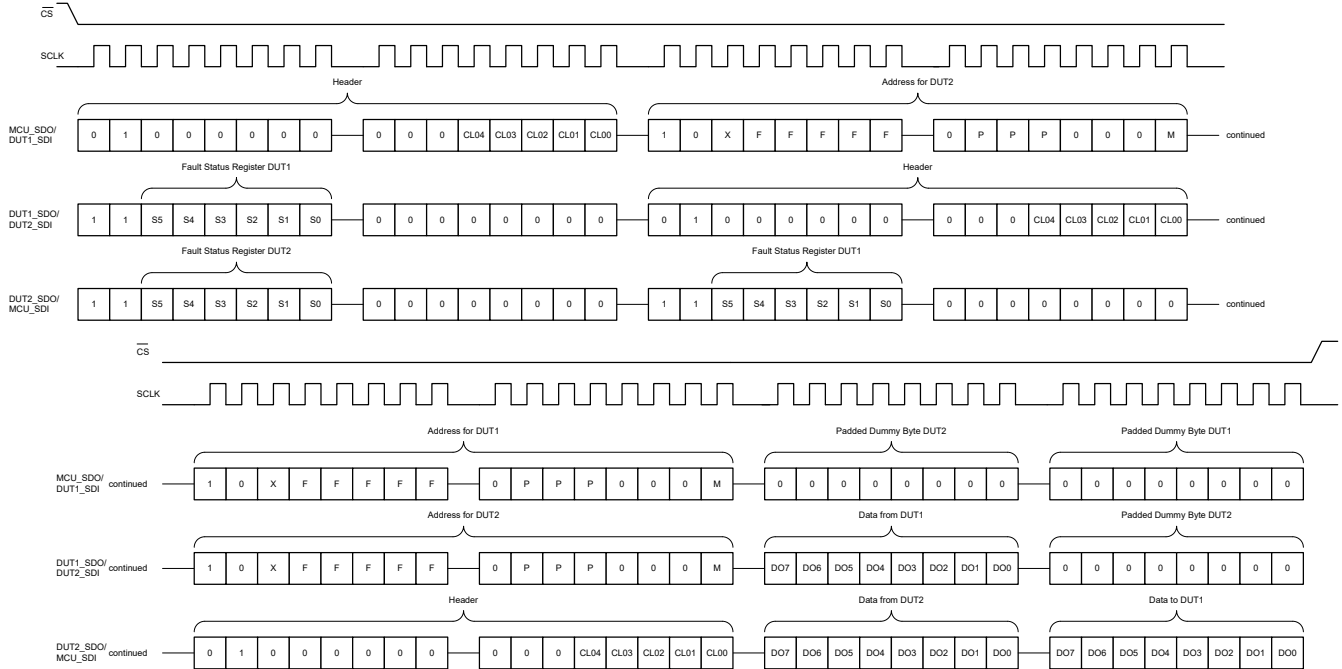


Figure 7-12. SPI Daisy Chain Read

7.6 TXE8148 Registers

Table 7-3 lists the memory-mapped registers for the TXE8148 registers. All register offset addresses not listed in Table 7-3 should be considered as reserved locations and the register contents should not be modified.

Table 7-3. TXE8148 Registers

Offset	Acronym	Register Name	Section
0h	Scratch_Register	Scratch Register	Section 7.6.1
10h	Device_ID_Register	Device ID Register	Section 7.6.2
20h + formula	Input_Port_Register_y	Input Port Register	Section 7.6.3
30h + formula	Output_Port_Register_y	Output Port Register	Section 7.6.4
40h + formula	Direction_Configuration_Register_y	Direction Configuration Register	Section 7.6.5
50h + formula	Polarity_Inversion_Register_y	Polarity Inversion Register	Section 7.6.6
60h + formula	Push_Pull/ Open_Drain_Selection_Register_y	Push Pull/Open Drain Selection Register	Section 7.6.7
80h + formula	Pull-up/Pull-down_Enable_Register_y	Pull-up/Pull-down Enable Register	Section 7.6.8
90h + formula	Pull-up/Pull-down_Selection_Register_y	Pull-up/Pull-down Selection Register	Section 7.6.9
A0h + formula	Bus_Holder_Register_y	Bus Holder Register	Section 7.6.10
B0h	Smart_Interrupt_Register	Smart Interrupt Register	Section 7.6.11
C0h + formula	Interrupt_Mask_Register_y	Interrupt Mask Register	Section 7.6.12
D0h + formula	Interrupt_Glitch_Filter_Enable_Register_y	Interrupt Glitch Filter Enable Register	Section 7.6.13
E0h + formula	Interrupt_Flag_Status_Register_y	Interrupt Flag Status Register	Section 7.6.14
F0h	Interrupt_Port_Status_Register	Interrupt Port Status Register	Section 7.6.15
120h	Fail-Safe_Enable_Register-1	Fail-safe Enable Register-1	Section 7.6.16
130h	Fail-Safe_Enable_Register-2	Fail-Safe Enable Register-2	Section 7.6.17
140h + formula	Fail-Safe_Direction_Configuration_Register-1_y	Fail-Safe Direction Configuration Register-1	Section 7.6.18
150h + formula	Fail-Safe_Direction_Configuration_Register-2_y	Fail-Safe Direction Configuration Register-2	Section 7.6.19
160h + formula	Fail-Safe_Output_Register-1_y	Fail-Safe Output Register-1	Section 7.6.20
170h + formula	Fail-Safe_Output_Register-2_y	Fail-Safe Output Register-2	Section 7.6.21
180h	Fail-Safe_Redundancy_Check_Register	Fail-Safe Redundancy Check Register	Section 7.6.22
190h	Fault_Status_Register	Fault Status Register	Section 7.6.23
1A0h	Software_Reset_Register	Software Reset Register	Section 7.6.24

Complex bit access types are encoded to fit into small table cells. Table 7-4 shows the codes that are used for access types in this section.

Table 7-4. TXE8148 Access Type Codes

Access Type	Code	Description
Read Type		

Table 7-4. TXE8148 Access Type Codes (continued)

Access Type	Code	Description
R	R	Read
Write Type		
W	W	Write
W1S	W 1S	Write 1 to set
Reset or Default Value		
-n		Value after reset or the default value
Register Array Variables		
i,j,k,l,m,n		When these variables are used in a register name, an offset, or an address, they refer to the value of a register array where the register is part of a group of repeating registers. The register groups form a hierarchical structure and the array is represented with a formula.
y		When this variable is used in a register name, an offset, or an address it refers to the value of a register array.

7.6.1 Scratch_Register (Offset = 0h) [Reset = 00h]

Scratch_Register is shown in [Figure 7-13](#) and described in [Table 7-5](#).

Return to the [Summary Table](#).

The scratch register is a test register to read/write code from/to a blank register and resolve any coding issues

Figure 7-13. Scratch_Register

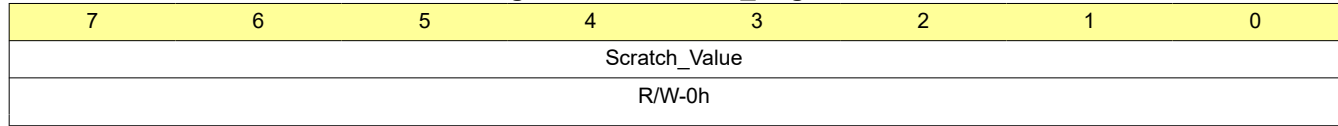


Table 7-5. Scratch_Register Field Descriptions

Bit	Field	Type	Reset	Description
7-0	Scratch_Value	R/W	0h	8-bit scratch value

7.6.2 Device_ID_Register (Offset = 10h) [Reset = 04h]

Device_ID_Register is shown in [Figure 7-14](#) and described in [Table 7-6](#).

Return to the [Summary Table](#).

The device identification register that has the device ID.

Figure 7-14. Device_ID_Register

7	6	5	4	3	2	1	0
Device_ID							
R-4h							

Table 7-6. Device_ID_Register Field Descriptions

Bit	Field	Type	Reset	Description
7-0	Device_ID	R	4h	Device identification 4: TXE8148

7.6.3 Input_Port_Register_y (Offset = 20h + formula) [Reset = XXh]

Input_Port_Register_y is shown in [Figure 7-15](#) and described in [Table 7-7](#).

Return to the [Summary Table](#).

The input port register reflects the incoming logic level of the pins for the port, when a pin is configured as input. When configured as an output, the corresponding pin reads as 0. The input port registers are read only. Writes to these registers have no effect. The externally applied logic level determines the default value.

Offset = 20h + (y * 1h); where y = 0h to 5h

Figure 7-15. Input_Port_Register_y

7	6	5	4	3	2	1	0
IPy.7	IPy.6	IPy.5	IPy.4	IPy.3	IPy.2	IPy.1	IPy.0
R-Xh	R-Xh	R-Xh	R-Xh	R-Xh	R-Xh	R-Xh	R-Xh

Table 7-7. Input_Port_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	IPy.7	R	Xh	Py.7 input pin value
6	IPy.6	R	Xh	Py.6 input pin value
5	IPy.5	R	Xh	Py.5 input pin value
4	IPy.4	R	Xh	Py.4 input pin value
3	IPy.3	R	Xh	Py.3 input pin value
2	IPy.2	R	Xh	Py.2 input pin value
1	IPy.1	R	Xh	Py.1 input pin value
0	IPy.0	R	Xh	Py.0 input pin value

7.6.4 Output_Port_Register_y (Offset = 30h + formula) [Reset = 00h]

Output_Port_Register_y is shown in [Figure 7-16](#) and described in [Table 7-8](#).

Return to the [Summary Table](#).

The output port register reflect the outgoing logic levels of the pins Py.0 to Py.7, when defined as outputs by the configuration register. Bit values in the register have no effect on pins defined as inputs. Reads from the register reflect the value that is in the register and not the actual pin value.

Offset = 30h + (y * 1h); where y = 0h to 5h

Figure 7-16. Output_Port_Register_y

7	6	5	4	3	2	1	0
OPy.7	OPy.6	OPy.5	OPy.4	OPy.3	OPy.2	OPy.1	OPy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-8. Output_Port_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	OPy.7	R/W	0h	Py.7 output value
6	OPy.6	R/W	0h	Py.6 output value
5	OPy.5	R/W	0h	Py.5 output value
4	OPy.4	R/W	0h	Py.4 output value
3	OPy.3	R/W	0h	Py.3 output value
2	OPy.2	R/W	0h	Py.2 output value
1	OPy.1	R/W	0h	Py.1 output value
0	OPy.0	R/W	0h	Py.0 output value

7.6.5 Direction_Configuration_Register_y (Offset = 40h + formula) [Reset = 00h]

Direction_Configuration_Register_y is shown in [Figure 7-17](#) and described in [Table 7-9](#).

Return to the [Summary Table](#).

The direction configuration registers configure the direction of the I/O pins. If a bit in the register is set to 0, the corresponding port pin is enabled as a high-impedance input. If a bit in the register is set to 1, the corresponding port pin is enabled as an output.

Offset = 40h + (y * 1h); where y = 0h to 5h

Figure 7-17. Direction_Configuration_Register_y

7	6	5	4	3	2	1	0
DIRy.7	DIRy.6	DIRy.5	DIRy.4	DIRy.3	DIRy.2	DIRy.1	DIRy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-9. Direction_Configuration_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	DIRy.7	R/W	0h	Py.7 configuration 0: Pin is in input mode 1: Pin is in output mode
6	DIRy.6	R/W	0h	Py.6 configuration 0: Pin is in input mode 1: Pin is in output mode
5	DIRy.5	R/W	0h	Py.5 configuration 0: Pin is in input mode 1: Pin is in output mode
4	DIRy.4	R/W	0h	Py.4 configuration 0: Pin is in input mode 1: Pin is in output mode
3	DIRy.3	R/W	0h	Py.3 configuration 0: Pin is in input mode 1: Pin is in output mode
2	DIRy.2	R/W	0h	Py.2 configuration 0: Pin is in input mode 1: Pin is in output mode
1	DIRy.1	R/W	0h	Py.1 configuration 0: Pin is in input mode 1: Pin is in output mode
0	DIRy.0	R/W	0h	Py.0 configuration 0: Pin is in input mode 1: Pin is in output mode

7.6.6 Polarity_Inversion_Register_y (Offset = 50h + formula) [Reset = 00h]

Polarity_Inversion_Register_y is shown in [Figure 7-18](#) and described in [Table 7-10](#).

Return to the [Summary Table](#).

The polarity inversion register allows polarity inversion of pins Py.0 to Py.7, defined as inputs by the configuration register. If a bit in this register is set, the corresponding pin's polarity is inverted. If a bit in the register is cleared, the corresponding pin's polarity is retained.

Offset = 50h + (y * 1h); where y = 0h to 5h

Figure 7-18. Polarity_Inversion_Register_y

7	6	5	4	3	2	1	0
Ply.7	Ply.6	Ply.5	Ply.4	Ply.3	Ply.2	Ply.1	Ply.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-10. Polarity_Inversion_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	Ply.7	R/W	0h	Py.7 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted
6	Ply.6	R/W	0h	Py.6 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted
5	Ply.5	R/W	0h	Py.5 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted
4	Ply.4	R/W	0h	Py.4 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted
3	Ply.3	R/W	0h	Py.3 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted
2	Ply.2	R/W	0h	Py.2 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted
1	Ply.1	R/W	0h	Py.1 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted
0	Ply.0	R/W	0h	Py.0 polarity inversion 0: Polarity is unchanged 1: Polarity is inverted

7.6.7 Push_Pull/Open_Drain_Selection_Register_y (Offset = 60h + formula) [Reset = 00h]

Push_Pull/Open_Drain_Selection_Register_y is shown in [Figure 7-19](#) and described in [Table 7-11](#).

Return to the [Summary Table](#).

The push pull / open drain selection register is used to select the pin as push-pull or open-drain. When the bit is set, the corresponding pin is configured as open-drain (Q1 is disabled and Q2 is active). When the bit is cleared, the corresponding pin is configured as push-pull (Q1 and Q2 are active).

Offset = 60h + (y * 1h); where y = 0h to 5h

Figure 7-19. Push_Pull/Open_Drain_Selection_Register_y

7	6	5	4	3	2	1	0
PPODy.7	PPODy.6	PPODy.5	PPODy.4	PPODy.3	PPODy.2	PPODy.1	PPODy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-11. Push_Pull/Open_Drain_Selection_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	PPODy.7	R/W	0h	Py.7 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain
6	PPODy.6	R/W	0h	Py.6 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain
5	PPODy.5	R/W	0h	Py.5 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain
4	PPODy.4	R/W	0h	Py.4 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain
3	PPODy.3	R/W	0h	Py.3 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain
2	PPODy.2	R/W	0h	Py.2 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain
1	PPODy.1	R/W	0h	Py.1 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain
0	PPODy.0	R/W	0h	Py.0 output configuration bit 0: Pin is configured as output push-pull 1: Pin is configured as output open-drain

7.6.8 Pull-up/Pull-down_Enable_Register_y (Offset = 80h + formula) [Reset = 00h]

Pull-up/Pull-down_Enable_Register_y is shown in [Figure 7-20](#) and described in [Table 7-12](#).

Return to the [Summary Table](#).

The pull-up/pull-down enable register allows the user to enable or disable pull-up/pull-down resistors on the GPIO pins. Setting the bit enables the selection of pull-up/pull-down resistors on the pins. Clearing the bit disconnects the pull-up/pull-down resistors from the pins. The resistors are disabled when the corresponding pins are configured as outputs.

Offset = 80h + (y * 1h); where y = 0h to 5h

Figure 7-20. Pull-up/Pull-down_Enable_Register_y

7	6	5	4	3	2	1	0
PEy.7	PEy.6	PEy.5	PEy.4	PEy.3	PEy.2	PEy.1	PEy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-12. Pull-up/Pull-down_Enable_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	PEy.7	R/W	0h	Py.7 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin
6	PEy.6	R/W	0h	Py.6 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin
5	PEy.5	R/W	0h	Py.5 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin
4	PEy.4	R/W	0h	Py.4 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin
3	PEy.3	R/W	0h	Py.3 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin
2	PEy.2	R/W	0h	Py.2 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin
1	PEy.1	R/W	0h	Py.1 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin
0	PEy.0	R/W	0h	Py.0 pull-up/pull-down enable 0: Pull-up/Pull-down is disabled on the pin 1: Pull-up/Pull-down is enabled on the pin

7.6.9 Pull-up/Pull-down_Selection_Register_y (Offset = 90h + formula) [Reset = 00h]

Pull-up/Pull-down_Selection_Register_y is shown in [Figure 7-21](#) and described in [Table 7-13](#).

Return to the [Summary Table](#).

The pull-up/pull-down selection register allows the user to configure each GPIO pin to have a pull-up or pull-down resistors. Setting the bit enables the pull-up resistors on the pins. Clearing the bit enables the pull-down resistors on the pins. If the pull-up/pull-down feature is disabled, then writing to the register will have no effect.

Offset = 90h + (y * 1h); where y = 0h to 5h

Figure 7-21. Pull-up/Pull-down_Selection_Register_y

7	6	5	4	3	2	1	0
PUDy.7	PUDy.6	PUDy.5	PUDy.4	PUDy.3	PUDy.2	PUDy.1	PUDy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-13. Pull-up/Pull-down_Selection_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	PUDy.7	R/W	0h	Py.7 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin
6	PUDy.6	R/W	0h	Py.6 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin
5	PUDy.5	R/W	0h	Py.5 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin
4	PUDy.4	R/W	0h	Py.4 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin
3	PUDy.3	R/W	0h	Py.3 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin
2	PUDy.2	R/W	0h	Py.2 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin
1	PUDy.1	R/W	0h	Py.1 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin
0	PUDy.0	R/W	0h	Py.0 pull-up/pull-down selection 0: Pull-down resistor is enabled on the pin 1: Pull-up resistor is enabled on the pin

7.6.10 Bus_Holder_Register_y (Offset = A0h + formula) [Reset = 00h]

Bus_Holder_Register_y is shown in [Figure 7-22](#) and described in [Table 7-14](#).

Return to the [Summary Table](#).

The bus holder register enables or disables the bus-hold or bus-keeper function of the GPIO pins. The feature is available only when the pin is configured as an input pin. When the register bit is set, the corresponding input pin state holds the line at its last known logical state when the device driving the pin floats its output.

Offset = A0h + (y * 1h); where y = 0h to 5h

Figure 7-22. Bus_Holder_Register_y

7	6	5	4	3	2	1	0
BHy.07	BHy.6	BHy.5	BHy.4	BHy.3	BHy.2	BHy.1	BHy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-14. Bus_Holder_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	BHy.07	R/W	0h	Py.7 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled
6	BHy.6	R/W	0h	Py.6 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled
5	BHy.5	R/W	0h	Py.5 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled
4	BHy.4	R/W	0h	Py.4 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled
3	BHy.3	R/W	0h	Py.3 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled
2	BHy.2	R/W	0h	Py.2 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled
1	BHy.1	R/W	0h	Py.1 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled
0	BHy.0	R/W	0h	Py.0 bus holder configuration 0: Bus hold on the pin is disabled 1: Bus hold on the pin is enabled

7.6.11 Smart_Interrupt_Register (Offset = B0h) [Reset = 00h]

Smart_Interrupt_Register is shown in [Figure 7-23](#) and described in [Table 7-15](#).

Return to the [Summary Table](#).

The register is used to enable or disable the smart interrupt function. When the bit is set to 0 (smart interrupt enabled), a state change in any input pin generates an interrupt and if the input goes back to its initial state, the interrupt is cleared.

When the bit is set to 1 (smart interrupt disabled), a state change in any input pin generates an interrupt and if the input goes back to its initial state, the interrupt is not cleared. A read of the interrupt status flag register will clear the interrupt.

This feature is enabled at the port level only.

Figure 7-23. Smart_Interrupt_Register

7	6	5	4	3	2	1	0
RESERVED						SI.P1	SI.P0
R-0h						R/W-0h	R/W-0h

Table 7-15. Smart_Interrupt_Register Field Descriptions

Bit	Field	Type	Reset	Description
7-2	RESERVED	R	0h	Do not use
1	SI.P1	R/W	0h	Smart Interrupt for Port 1 0: Enabled 1: Disabled
0	SI.P0	R/W	0h	Smart Interrupt for Port 0 0: Enabled 1: Disabled

7.6.12 Interrupt_Mask_Register_y (Offset = C0h + formula) [Reset = FFh]

Interrupt_Mask_Register_y is shown in [Figure 7-24](#) and described in [Table 7-16](#).

Return to the [Summary Table](#).

The interrupt mask register enables or disable interrupt generation from pins. If the mask register bit is set, and the input state of a pin changes, then interrupt is not asserted. If the mask register bit is cleared, and the input state of a pin changes, then interrupt is asserted.

Offset = C0h + (y * 1h); where y = 0h to 5h

Figure 7-24. Interrupt_Mask_Register_y

7	6	5	4	3	2	1	0
IMy.7	IMy.6	IMy.5	IMy.4	IMy.3	IMy.2	IMy.1	IMy.0
R/W-1h	R/W-1h	R/W-1h	R/W-1h	R/W-1h	R/W-1h	R/W-1h	R/W-1h

Table 7-16. Interrupt_Mask_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	IMy.7	R/W	1h	Py.7 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin
6	IMy.6	R/W	1h	Py.6 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin
5	IMy.5	R/W	1h	Py.5 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin
4	IMy.4	R/W	1h	Py.4 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin
3	IMy.3	R/W	1h	Py.3 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin
2	IMy.2	R/W	1h	Py.2 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin
1	IMy.1	R/W	1h	Py.1 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin
0	IMy.0	R/W	1h	Py.0 interrupt mask bit 0: Interrupt is not masked on state change of the pin 1: Interrupt is masked on state change of the pin

7.6.13 Interrupt_Glitch_Filter_Enable_Register_y (Offset = D0h + formula) [Reset = 00h]

Interrupt_Glitch_Filter_Enable_Register_y is shown in [Figure 7-25](#) and described in [Table 7-17](#).

Return to the [Summary Table](#).

The interrupt glitch filter register enables or disable the glitch filter on the I/O. If the bit is set to 1, the glitch filter for the corresponding pin is enabled. If the bit is set to 0, the glitch filter for the corresponding pin is disabled.

Offset = D0h + (y * 1h); where y = 0h to 5h

Figure 7-25. Interrupt_Glitch_Filter_Enable_Register_y

7	6	5	4	3	2	1	0
GFy.7	GFy.6	GFy.5	GFy.4	GFy.3	GFy.2	GFy.1	GFy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-17. Interrupt_Glitch_Filter_Enable_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	GFy.7	R/W	0h	Py.7 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled
6	GFy.6	R/W	0h	Py.6 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled
5	GFy.5	R/W	0h	Py.5 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled
4	GFy.4	R/W	0h	Py.4 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled
3	GFy.3	R/W	0h	Py.3 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled
2	GFy.2	R/W	0h	Py.2 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled
1	GFy.1	R/W	0h	Py.1 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled
0	GFy.0	R/W	0h	Py.0 glitch filter enable bit 0: Glitch filter is disabled 1: Glitch filter is enabled

7.6.14 Interrupt_Flag_Status_Register_y (Offset = E0h + formula) [Reset = 00h]

Interrupt_Flag_Status_Register_y is shown in [Figure 7-26](#) and described in [Table 7-18](#).

Return to the [Summary Table](#).

The interrupt status register is read only register bits used to identify the source of an interrupt. When read, a logic 1 indicates that the corresponding input pin was the source of the interrupt. A logic 0 indicates that the input pin is not the source of the an interrupt. When the corresponding bit in the interrupt mask register is set to 1 (masked), the interrupt status bit will return a logic 0.

Offset = E0h + (y * 1h); where y = 0h to 5h

Figure 7-26. Interrupt_Flag_Status_Register_y

7	6	5	4	3	2	1	0
ISy.7	ISy.6	ISy.5	ISy.4	ISy.3	ISy.2	ISy.1	ISy.0
R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h	R-0h

Table 7-18. Interrupt_Flag_Status_Register_y Field Descriptions

Bit	Field	Type	Reset	Description
7	ISy.7	R	0h	Py.7 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt
6	ISy.6	R	0h	Py.6 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt
5	ISy.5	R	0h	Py.5 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt
4	ISy.4	R	0h	Py.4 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt
3	ISy.3	R	0h	Py.3 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt
2	ISy.2	R	0h	Py.2 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt
1	ISy.1	R	0h	Py.1 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt
0	ISy.0	R	0h	Py.0 interrupt status bit 0: Pin is not the source of interrupt 1: Pin is the source of the interrupt

7.6.15 Interrupt_Port_Status_Register (Offset = F0h) [Reset = 00h]

Interrupt_Port_Status_Register is shown in [Figure 7-27](#) and described in [Table 7-19](#).

Return to the [Summary Table](#).

The register is used to identify the IO port for the interrupt source. If the value of the bit is 1, it indicates that the source of the interrupt is from the corresponding IO port. If the value is 0, it indicates that none of the input pins in the IO port generated an interrupt.

Figure 7-27. Interrupt_Port_Status_Register

7	6	5	4	3	2	1	0
RESERVED						IPS.P1	IPS.P0
R-0h						R/W-0h	R/W-0h

Table 7-19. Interrupt_Port_Status_Register Field Descriptions

Bit	Field	Type	Reset	Description
7-2	RESERVED	R	0h	Do not use
1	IPS.P1	R/W	0h	Interrupt Status for Port 1 0: No interrupt from Port 1 1: Interrupt from Port 1
0	IPS.P0	R/W	0h	Interrupt Status for Port 0 0: No interrupt from Port 0 1: Interrupt from Port 0

7.6.16 Fail-Safe_Enable_Register-1 (Offset = 120h) [Reset = 00h]

Fail-Safe_Enable_Register-1 is shown in [Figure 7-28](#) and described in [Table 7-20](#).

Return to the [Summary Table](#).

This register reconfigures the reset pin as a fail-safe pin. The register is cleared on POR or fault condition as mentioned in the fault status register.

There are two fail-safe registers in the device that have to be written to program the I/O configuration with built-in redundancy. If the value in the two registers mismatch, it will generate an interrupt with fault status set accordingly.

Figure 7-28. Fail-Safe_Enable_Register-1

7	6	5	4	3	2	1	0
RESERVED							FS1EN
R-0h							R/W-0h

Table 7-20. Fail-Safe_Enable_Register-1 Field Descriptions

Bit	Field	Type	Reset	Description
7-1	RESERVED	R	0h	Do not use
0	FS1EN	R/W	0h	Fail-Safe Enable bit 0: Disable 1: Enable

7.6.17 Fail-Safe_Enable_Register-2 (Offset = 130h) [Reset = 00h]

Fail-Safe_Enable_Register-2 is shown in [Figure 7-29](#) and described in [Table 7-21](#).

Return to the [Summary Table](#).

This register reconfigures the reset pin as a fail-safe pin. The register is cleared on POR or fault condition as mentioned in the fault status register.

There are two fail-safe registers in the device that have to be written to program the I/O configuration with built-in redundancy. If the value in the two registers mismatch, it will generate an interrupt with fault status set accordingly.

Figure 7-29. Fail-Safe_Enable_Register-2

7	6	5	4	3	2	1	0
RESERVED							FS2EN
R-0h							R/W-0h

Table 7-21. Fail-Safe_Enable_Register-2 Field Descriptions

Bit	Field	Type	Reset	Description
7-1	RESERVED	R	0h	Do not use
0	FS2EN	R/W	0h	Fail-Safe Enable bit 0: Disable 1: Enable

7.6.18 Fail-Safe_Direction_Configuration_Register-1_y (Offset = 140h + formula) [Reset = 00h]

Fail-Safe_Direction_Configuration_Register-1_y is shown in [Figure 7-30](#) and described in [Table 7-22](#).

Return to the [Summary Table](#).

The fail-safe direction configuration register-1 configure the direction of the I/O pins when the device enters a fail-safe state. If a bit in these register is set to 0, the corresponding IO pin is enabled as a high-impedance input during fail-safe mode. If a bit is set to 1, the corresponding IO pin is enabled as an output during fail-safe mode. There are two registers per port, that have to be written to program the I/O configuration to ensure redundancy. If the registers mismatch, it will generate an interrupt with fault status set accordingly.

Offset = 140h + (y * 1h); where y = 0h to 5h

Figure 7-30. Fail-Safe_Direction_Configuration_Register-1_y

7	6	5	4	3	2	1	0
FS1DIRy.7	FS1DIRy.6	FS1DIRy.5	FS1DIRy.4	FS1DIRy.3	FS1DIRy.2	FS1DIRy.1	FS1DIRy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-22. Fail-Safe_Direction_Configuration_Register-1_y Field Descriptions

Bit	Field	Type	Reset	Description
7	FS1DIRy.7	R/W	0h	Py.7 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
6	FS1DIRy.6	R/W	0h	Py.6 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
5	FS1DIRy.5	R/W	0h	Py.5 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
4	FS1DIRy.4	R/W	0h	Py.4 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
3	FS1DIRy.3	R/W	0h	Py.3 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
2	FS1DIRy.2	R/W	0h	Py.2 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
1	FS1DIRy.1	R/W	0h	Py.1 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
0	FS1DIRy.0	R/W	0h	Py.0 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode

7.6.19 Fail-Safe_Direction_Configuration_Register-2_y (Offset = 150h + formula) [Reset = 00h]

Fail-Safe_Direction_Configuration_Register-2_y is shown in [Figure 7-31](#) and described in [Table 7-23](#).

Return to the [Summary Table](#).

The fail-safe direction configuration register-2 configure the direction of the I/O pins when the device enters a fail-safe state. If a bit in these register is set to 0, the corresponding IO pin is enabled as a high-impedance input during fail-safe mode. If a bit is set to 1, the corresponding IO pin is enabled as an output during fail-safe mode. There are two registers per port, that have to be written to program the I/O configuration to ensure redundancy. If the registers mismatch, it will generate an interrupt with fault status set accordingly.

Offset = 150h + (y * 1h); where y = 0h to 5h

Figure 7-31. Fail-Safe_Direction_Configuration_Register-2_y

7	6	5	4	3	2	1	0
FS2DIRy.7	FS2DIRy.6	FS2DIRy.5	FS2DIRy.4	FS2DIRy.3	FS2DIRy.2	FS2DIRy.1	FS2DIRy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-23. Fail-Safe_Direction_Configuration_Register-2_y Field Descriptions

Bit	Field	Type	Reset	Description
7	FS2DIRy.7	R/W	0h	Py.7 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
6	FS2DIRy.6	R/W	0h	Py.6 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
5	FS2DIRy.5	R/W	0h	Py.5 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
4	FS2DIRy.4	R/W	0h	Py.4 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
3	FS2DIRy.3	R/W	0h	Py.3 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
2	FS2DIRy.2	R/W	0h	Py.2 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
1	FS2DIRy.1	R/W	0h	Py.1 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode
0	FS2DIRy.0	R/W	0h	Py.0 fail-safe configuration bit 0: Pin is configured in input mode 1: Pin is configured in output mode

7.6.20 Fail-Safe_Output_Register-1_y (Offset = 160h + formula) [Reset = 00h]

Fail-Safe_Output_Register-1_y is shown in [Figure 7-32](#) and described in [Table 7-24](#).

Return to the [Summary Table](#).

The fail-safe output register-1 configure the output level of the pins defined as output by the fail-safe configuration register. Bit values in the register have no effect on pins defined as input.

There are two registers per port, that have to be written to with the same output value to ensure redundancy. If the registers mismatch, it will generate an interrupt with fault status set accordingly.

Offset = 160h + (y * 1h); where y = 0h to 5h

Figure 7-32. Fail-Safe_Output_Register-1_y

7	6	5	4	3	2	1	0
FS1OPy.7	FS1OPy.6	FS1OPy.5	FS1OPy.4	FS1OPy.3	FS1OPy.2	FS1OPy.1	FS1OPy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-24. Fail-Safe_Output_Register-1_y Field Descriptions

Bit	Field	Type	Reset	Description
7	FS1OPy.7	R/W	0h	Py.7 fail-safe output value
6	FS1OPy.6	R/W	0h	Py.6 fail-safe output value
5	FS1OPy.5	R/W	0h	Py.5 fail-safe output value
4	FS1OPy.4	R/W	0h	Py.4 fail-safe output value
3	FS1OPy.3	R/W	0h	Py.3 fail-safe output value
2	FS1OPy.2	R/W	0h	Py.2 fail-safe output value
1	FS1OPy.1	R/W	0h	Py.1 fail-safe output value
0	FS1OPy.0	R/W	0h	Py.0 fail-safe output value

7.6.21 Fail-Safe_Output_Register-2_y (Offset = 170h + formula) [Reset = 00h]

Fail-Safe_Output_Register-2_y is shown in [Figure 7-33](#) and described in [Table 7-25](#).

Return to the [Summary Table](#).

The fail-safe output register-2 configure the output level of the pins defined as output by the fail-safe configuration register. Bit values in the register have no effect on pins defined as input. There are two registers per port, that have to be written to with the same output value to ensure redundancy. If the registers mismatch, it will generate an interrupt with fault status set accordingly.

Offset = 170h + (y * 1h); where y = 0h to 5h

Figure 7-33. Fail-Safe_Output_Register-2_y

7	6	5	4	3	2	1	0
FS2OPy.7	FS2OPy.6	FS2OPy.5	FS2OPy.4	FS2OPy.3	FS2OPy.2	FS2OPy.1	FS2OPy.0
R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h	R/W-0h

Table 7-25. Fail-Safe_Output_Register-2_y Field Descriptions

Bit	Field	Type	Reset	Description
7	FS2OPy.7	R/W	0h	Py.7 fail-safe output value
6	FS2OPy.6	R/W	0h	Py.6 fail-safe output value
5	FS2OPy.5	R/W	0h	Py.5 fail-safe output value
4	FS2OPy.4	R/W	0h	Py.4 fail-safe output value
3	FS2OPy.3	R/W	0h	Py.3 fail-safe output value
2	FS2OPy.2	R/W	0h	Py.2 fail-safe output value
1	FS2OPy.1	R/W	0h	Py.1 fail-safe output value
0	FS2OPy.0	R/W	0h	Py.0 fail-safe output value

7.6.22 Fail-Safe_Redundancy_Check_Register (Offset = 180h) [Reset = 00h]

Fail-Safe_Redundancy_Check_Register is shown in [Figure 7-34](#) and described in [Table 7-26](#).

Return to the [Summary Table](#).

The fail-safe redundancy check register is used to enable the check on the fail-safe redundant registers (fail-safe enable register, fail-safe direction configuration register and fail-safe output register).

Figure 7-34. Fail-Safe_Redundancy_Check_Register

7	6	5	4	3	2	1	0
RESERVED							FSCHECKEN
R-0h							R/W-0h

Table 7-26. Fail-Safe_Redundancy_Check_Register Field Descriptions

Bit	Field	Type	Reset	Description
7-1	RESERVED	R	0h	Do not use
0	FSCHECKEN	R/W	0h	Fail-Safe Check Enable bit 0: Disable 1: Enable

7.6.23 Fault_Status_Register (Offset = 190h) [Reset = 01h]

Fault_Status_Register is shown in [Figure 7-35](#) and described in [Table 7-27](#).

Return to the [Summary Table](#).

The fault status register are set during fault conditions. Bit-0 is set to 1 for POR recovery. Bit-1 is set when the redundancy register has a mismatch. Bit-2 is set when the device is in fail safe mode. These flags are not cleared even if the fault condition goes away. They can only be cleared by performing a read of the fault status register.

Figure 7-35. Fault_Status_Register

7	6	5	4	3	2	1	0
RESERVED					FSMODEACTIV E	REGMISMATC H	POR
R-0h					R-0h	R-0h	R-1h

Table 7-27. Fault_Status_Register Field Descriptions

Bit	Field	Type	Reset	Description
9-3	RESERVED	R	0h	Do not use
2	FSMODEACTIVE	R	0h	Fail Safe function is Active
1	REGMISMATCH	R	0h	Fail Safe function cleared due to register mismatch
0	POR	R	1h	POR recovery status

7.6.24 Software_Reset_Register (Offset = 1A0h) [Reset = 00h]

Software_Reset_Register is shown in [Figure 7-36](#) and described in [Table 7-28](#).

Return to the [Summary Table](#).

The register is used to trigger a software reset of the device. The register is auto-cleared when the reset state is entered.

Figure 7-36. Software_Reset_Register

7	6	5	4	3	2	1	0
RESERVED						Register_Reset	Device_Reset
R-0h						R/W1S-0h	R/W1S-0h

Table 7-28. Software_Reset_Register Field Descriptions

Bit	Field	Type	Reset	Description
7-2	RESERVED	R	0h	Do not use
1	Register_Reset	R/W1S	0h	When set triggers a register reset
0	Device_Reset	R/W1S	0h	When set triggers a device reset

8 Application and Implementation

Note

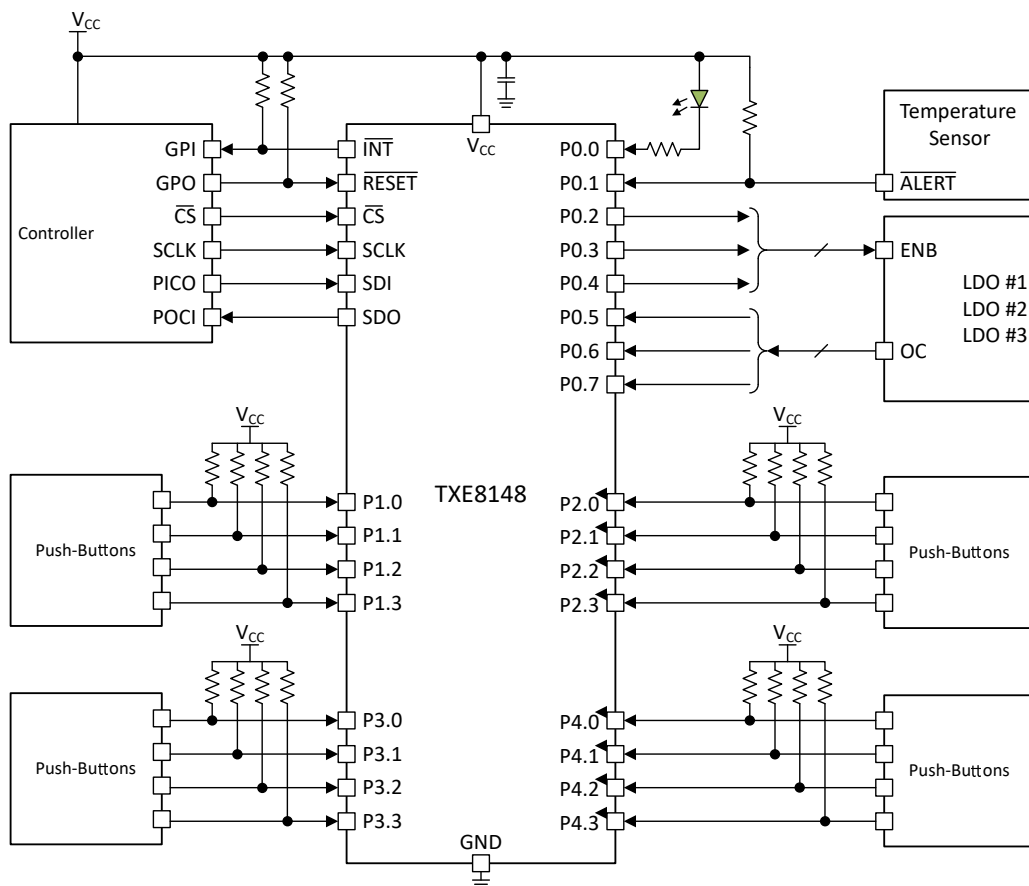
Information in the following applications sections is not part of the TI component specification, and TI does not warrant its accuracy or completeness. TI's customers are responsible for determining suitability of components for their purposes. Customers should validate and test their design implementation to confirm system functionality.

8.1 Application Information

Applications of the TXE8148 use this device connected as a target to an SPI controller (processor), and the SPI bus can contain any number of other target devices. The TXE8148 is in a remote location from the controller, placed close to the GPIOs to which the controller needs to monitor or control.

8.2 Typical Application

Figure 8-1 shows an application in which the TXE8148 devices can be used.



- A. P1.0 to P1.3, P2.0 to P2.3, P3.0 to P3.3, P4.0 to P4.3 are configured as inputs for specific set of switches. P0.1 and P0.5 to P0.7 are used for monitor inputs.
- B. P0.2 to P0.4, are configured as outputs
- C. Resistors are required for inputs (on P-port) that can float. If a driver to an input never lets the input float, a resistor is not needed. Outputs (in the P-port) do not need pullup resistors.

Figure 8-1. Typical Application Schematic

The following shows a sample application code example that can be used for accessing the device.

```

uint32_t SPIExchangeData(uint32_t transmitData)
{
    uint32_t receiveData=0x0;

    for(int i = 23; i >= 0; i--)
    {
        // Shift out MSB first
        SPISendBit((transmitData >> i) & 0x01);
        receiveData |= SPIGetBit() << i;
    }

    return(receiveData);
}

uint32_t SPITransmitReceive(uint32_t RnW, uint32_t regPointer, uint32_t multiPort, uint32_t
transmitData)
{
    uint32_t regWrData=0x0;
    uint32_t regRdData=0x0;

    regWrData = RnW        << 23 | \
                regPointer << 12 | \
                multiPort  <<  8 | \
                transmitData <<  0;

    regRdData = SPIExchangeData(regWrData);

    return(regRdData);
}

```

8.3 Power Supply Recommendations

8.3.1 Power-On Reset Requirements

In the event of a glitch or data corruption, TXE8148 can be reset to the default conditions by using the power-on reset feature. Power-on reset requires that the device go through a power cycle to be completely reset. This reset also happens when the device is powered on for the first time in an application.

The two types of power-on reset are shown in [Figure 8-2](#) and [Figure 8-3](#).

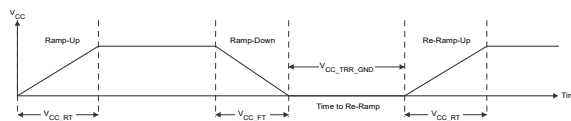


Figure 8-2. V_{CC} is Lowered to 0V and Then RAMPED Up

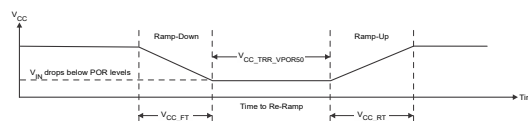


Figure 8-3. V_{CC} is Lowered Below the POR Threshold, Then Ramped Back Up

Glitches in the power supply can also affect the power-on reset performance of this device. The glitch width (t_{VCC_GW}) and height (V_{CC_GH}) are dependent on each other. The bypass capacitance, source impedance, and device impedance are factors that affect power-on reset performance. [Figure 8-4](#) provides more information on how to measure these specifications.

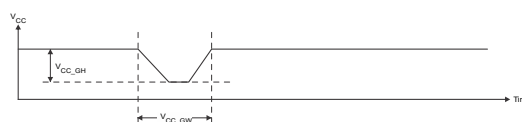


Figure 8-4. Glitch Width and Glitch Height

V_{POR} is critical to the power-on reset. V_{POR} is the voltage level at which the reset condition is released and all the registers and the SPI state machine are initialized to the default states. The value of V_{POR} differs based on the V_{CC} being lowered to or from 0. [Figure 8-5](#) provides more detail on this specification.

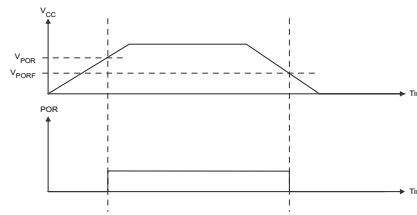


Figure 8-5. V_{POR}

8.4 Layout

8.4.1 Layout Guidelines

For printed circuit board (PCB) layout of the TXE8148, common PCB layout practices must be followed but additional concerns related to high-speed data transfer such as matched impedance and differential pairs are not a concern for SPI signal speeds.

In all PCB layouts, avoiding right angles in signal traces, fanning out signal traces away from each other upon leaving the vicinity of an integrated circuit (IC), and using thicker trace widths to carry higher amounts of current that commonly pass through power and ground traces is best practice. By-pass and decoupling capacitors are commonly used to control the voltage on the supply pins, using a larger capacitor to provide additional power in the event of a short power supply glitch and a smaller capacitor to filter out high-frequency ripple. These capacitors must be placed as close to the TXE8148 as possible. These best practices are shown in [Layout Example](#).

For the layout example provided in [Layout Example](#), fabricating a PCB with only 2 layers by using the top layer for signal routing and the bottom layer as a split plane for power and ground (GND) is possible. However, a 4 layer board is preferable for boards with higher density signal routing. On a 4 layer PCB, routing signals on the top and bottom layer, dedicating one internal layer to a ground plane, and dedicating the other internal layer to a power plane is common. In a board layout using planes or split planes for power and ground, vias are placed directly next to the surface mount component pad which needs to attach to power or GND and the via is connected electrically to the internal layer or the other side of the board. Vias are also used when a signal trace needs to be routed to the opposite side of the board, but this technique is not demonstrated in [Layout Example](#).

8.4.2 Layout Example

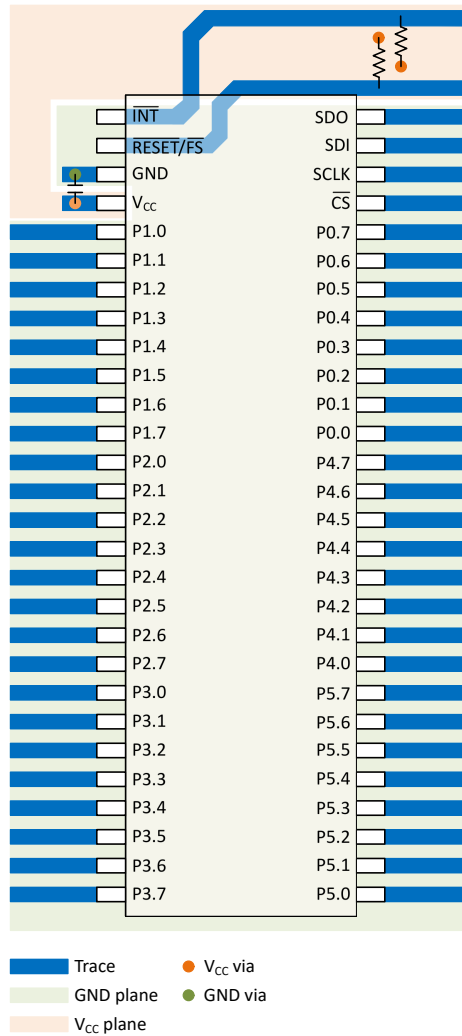


Figure 8-6. TXE8148 Layout

9 Device and Documentation Support

9.1 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.

9.2 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](#).

9.3 Trademarks

TI E2E™ is a trademark of Texas Instruments.
All trademarks are the property of their respective owners.

9.4 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.

9.5 Glossary

[TI Glossary](#) This glossary lists and explains terms, acronyms, and definitions.

10 Revision History

DATE	REVISION	NOTES
June 2026	*	Initial Release

11 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.

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), 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 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](#), [TI's General Quality Guidelines](#), 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. Unless TI explicitly designates a product as custom or customer-specified, TI products are standard, catalog, general purpose devices.

TI objects to and rejects any additional or different terms you may propose.

Copyright © 2026, Texas Instruments Incorporated

Last updated 10/2025