Four-Channel Remote Temperature Sensor

MAX31732

General Description

The MAX31732 is a multi-channel temperature sensor that monitors its own temperature and the temperatures of up to four external diode-connected transistors. A resistance cancellation feature compensates for up to 300Ω of series resistance between the external thermal diode and the MAX31732, while beta compensation corrects for temperature-measurement errors due to low-beta sensing transistors. The sensor also allows ideality factors between 0.9844 and 1.311.

The MAX31732 offers two open-drain, active-low alarm outputs, $\overline{ALARM1}$ and $\overline{ALARM2}$. When the measured temperature of a channel crosses the respective primary overtemperature/undertemperature threshold levels, $\overline{ALARM1}$ asserts low, and a status bit is set in the corresponding thermal status registers. When the measured temperature of a channel crosses the secondary overtemperature/undertemperature threshold levels, $\overline{ALARM2}$ asserts low and a status bit is set in the corresponding thermal status registers. The highest temperature register allows the controller to obtain the temperature of the hottest channel.

A 512-bit nonvolatile memory (NVM) allows the MAX31732 to program the configuration registers during power-up without software/firmware intervention. The NVM also captures the fault temperature for each remote and local channel for further analysis at a later time.

The 2-wire serial interface accepts SMBus protocols (write byte, read byte, send byte, and receive byte) for reading the temperature data and programming the temperature thresholds. Any one of the eight available target addresses can be selected using the address selection input (ADD), which can be connected to ground or connected to a grounded resistor.

The is specified over the -40°C to +125°C operating temperature range and with a 3.0V to 3.6V input supply range. MAX31732 is available in a 24-pin, 4mm x 4mm TQFN package.

Applications

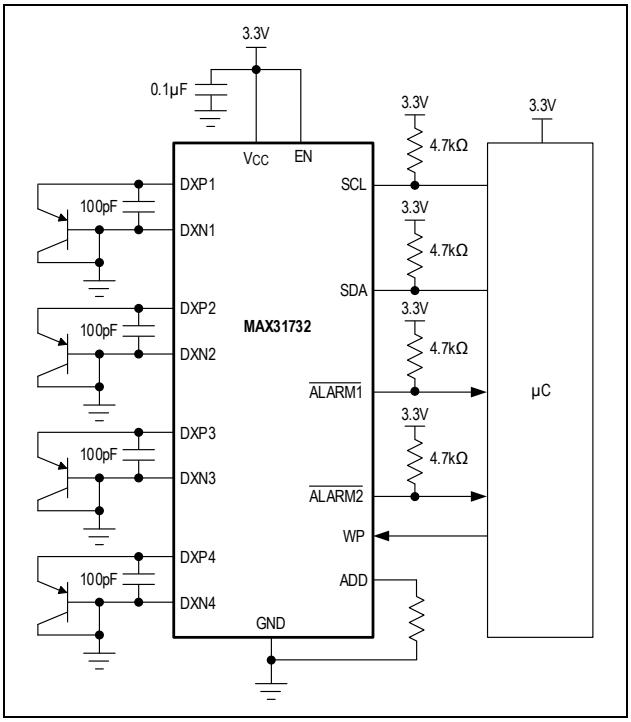
- Temperature Measurement of On-Chip CPU, ASIC, SOC, or FPGA diodes or remote discrete diodes
- Industrial-Thermal Management

Features

- One Local and Four Remote Temperature-Sensing Channels
- ±1°C Remote Temperature-Measurement Accuracy (-40°C to +125°C)
- -64°C to +150°C Remote Temperature-Measurement Range
- Up to 300Ω Resistance Cancellation for Remote Channels
- Compensation for Low Beta Transistors
- 12 Bit, 0.0625°C Resolution
- Two Programmable Alarm Temperature Thresholds
- Eight Selectable Target Addresses
- Flexible I²C/SMBus Interfaces to a Variety of Microcontrollers
- Ideality Factor of up to 1.311
- NVM Facilitates Configuration and Fault Logging

***Ordering Information* appears at end of data sheet.**

Typical Application Circuit



19-101906; Rev 0; 6/24

TABLE OF CONTENTS

Absolute Maximum Ratings

V_{CC}, SCL, SDA, $\overline{ALARM1}$, $\overline{ALARM2}$, ADD, WP, EN .....-0.3V to 3.7V

DXP 1, 2, 3, 4 and DXN 1, 2, 3, 4..... -0.3V to V_{CC} + 0.3V

ESD Protection (All Pins, Human Body Model)..... 2kV

Continuous Power Dissipation (T_A = +70°C) (TQFN (derate 27.8mW/°C above +70°C))..... 2222mW

Operating Temperature Range ..... -40°C to +125°C

Junction Temperature ..... +150°C

Storage Temperature Range ..... -65°C to +150°C

Lead Temperature (soldering, 10s) ..... +300°C

Soldering Temperature (reflow)..... +260°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

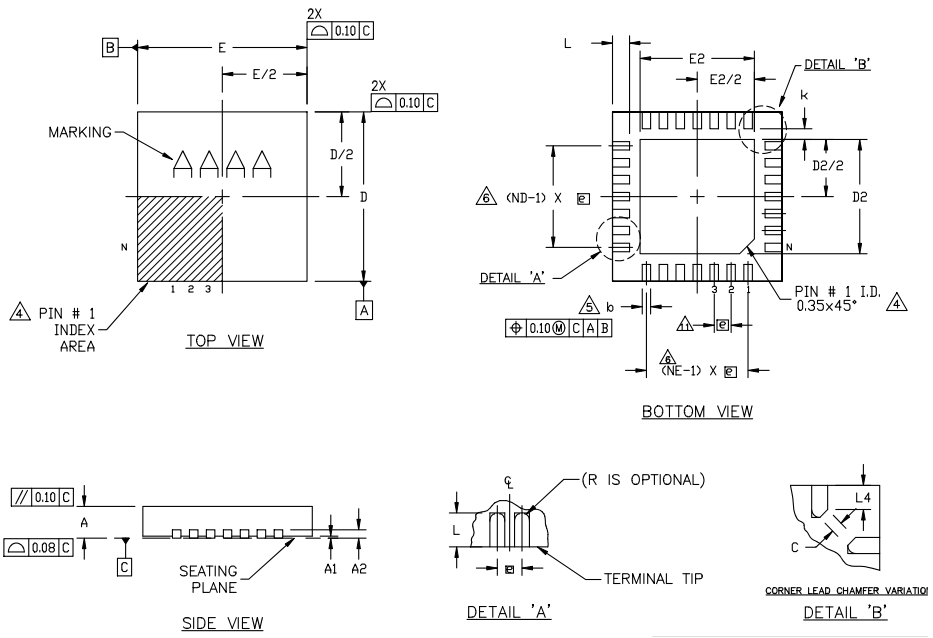
Package Information

Package Code	T2444+4C
Outline Number	**21-0139**
Land Pattern Number	**90-0022**
Thermal Resistance, Single Layer Board:	
Junction-to-Ambient (θ_{JA})	48°C/W
Junction-to-Case Thermal Resistance (θ_{JC})	3°C/W
Thermal Resistance, Four Layer Board:	
Junction-to-Ambient (θ_{JA})	36°C/W
Junction-to-Case Thermal Resistance (θ_{JC})	3°C/W

For the latest package outline information and land patterns (footprints), go to Packaging Index. Note that a "+", "#", or "-" in the package code indicates RoHS status only. Package drawings may show a different suffix character, but the drawing pertains to the package regardless of RoHS status.

Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to Thermal Characterization of IC Packages.

Package Information (continued)



COMMON DIMENSIONS															
PKG	12L 4x4			16L 4x4			20L 4x4			24L 4x4			28L 4x4		
REF.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.	MIN.	NOM.	MAX.
A	0.70	0.75	0.80	0.70	0.75	0.80	0.70	0.75	0.80	0.70	0.75	0.80	0.70	0.75	0.80
A1	0.0	0.02	0.05	0.0	0.02	0.05	0.0	0.02	0.05	0.0	0.02	0.05	0.0	0.02	0.05
A2	0.20 REF			0.20 REF			0.20 REF			0.20 REF			0.20 REF		
b	0.25	0.30	0.35	0.25	0.30	0.35	0.20	0.25	0.30	0.18	0.23	0.30	0.15	0.20	0.25
D	3.90	4.00	4.10	3.90	4.00	4.10	3.90	4.00	4.10	3.90	4.00	4.10	3.90	4.00	4.10
E	3.90	4.00	4.10	3.90	4.00	4.10	3.90	4.00	4.10	3.90	4.00	4.10	3.90	4.00	4.10
e	0.80 BSC.			0.65 BSC.			0.50 BSC.			0.50 BSC.			0.40 BSC.		
k	0.25	–	–	0.25	–	–	0.25	–	–	0.25	–	–	0.25	–	–
N	12			16			20			24			28		
ND	3			4			5			6			7		
NE	3			4			5			6			7		
Jedec Var.	WGGB			WGGC			WGGD-1			WGGD-2			WGGE		

NOTES:
1. DIMENSIONING & TOLERANCING CONFORM TO ASME Y14.5M-2009.
2. ALL DIMENSIONS ARE IN MILLIMETERS. ANGLES ARE IN DEGREES.
3. N IS THE TOTAL NUMBER OF TERMINALS.
4. THE TERMINAL #1 IDENTIFIER AND TERMINAL NUMBERING CONVENTION SHALL CONFORM TO JESD 95-1 SPP-012. DETAILS OF TERMINAL #1 IDENTIFIER ARE OPTIONAL, BUT MUST BE LOCATED WITHIN THE ZONE INDICATED. THE TERMINAL #1 IDENTIFIER MAY BE EITHER A MOLD OR MARKED FEATURE.
5. DIMENSION b APPLIES TO METALLIZED TERMINAL AND IS MEASURED BETWEEN 0.25mm AND 0.30mm FROM TERMINAL TIP.
6. ND AND NE REFER TO THE NUMBER OF TERMINALS ON EACH D AND E SIDE RESPECTIVELY.
7. DEPOPULATION IS POSSIBLE IN A SYMMETRICAL FASHION.
8. DRAWING CONFORMS TO JEDEC MO220, EXCEPT FOR T2444-3, T2444-4 AND T2844-1.
9. MARKING IS FOR PACKAGE ORIENTATION REFERENCE ONLY.
10. WARPAGE SHALL NOT EXCEED 0.10mm.
11. LEAD CENTERLINES TO BE AT TRUE POSITION AS DEFINED BY BASIC DIMENSION 'e', ±0.05.
12. NUMBER OF LEADS SHOWN ARE FOR REFERENCE ONLY.
13. MATERIAL MUST COMPLY WITH BANNED AND RESTRICTED SUBSTANCES SPEC # 10-0131.
14. ALL DIMENSIONS ARE THE SAME FOR LEADED (-) & PbFREE (+) PACKAGE CODES.

Electrical Characteristics

(3.0V ≤ V_{CC} ≤ 3.6V = Power Supply Voltage, T_A = -40°C to +125°C = Ambient Temperature. All specifications are subject to change.) (Note 3)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Supply Voltage	V_{CC}			3		3.6	
Shutdown Supply Current	I_{SD}	Shutdown: EN = 0V			175		µA
Active Supply Current	I_Q	Operating, β compensation disabled			700	1200	µA
POR Threshold	V_{POR}	V_{CC} Falling Edge			2.6	2.8	V
POR Threshold Hysteresis					125		mV
Temperature Resolution					0.0625		°C
Remote Temperature Accuracy		T_A = -40°C to +125°C, T_{RJ} = -40°C to +150°C	3 sigma w/ diodes batch calibrated	-2	±1	2	°C
Local Temperature Accuracy		T_A = -40°C to +125°C	3 sigma	-2		2	°C
Temperature Measurement Noise		T_A = -40°C to +125°C	No Filtering		0.25		$°C_{RMS}$
Temperature Hysteresis		Comparator Mode Only			2		°C
Conversion Time Per Channel		β compensation disabled				100	ms
		β compensation enabled				150	
Conversion Time for all the channels		β compensation disabled				450	ms
		β compensation enabled				700	
Remote-Diode Source Current	I_{RJ}	High Level			200		µA
		Low Level			12		
DXN# Bias Voltage		β compensation disabled				0.3	V
		β compensation enabled			0.73	1.0	
$\overline{ALARM1}$, $\overline{ALARM2}$ Output Low Voltage	V_{OL}	I_{SINK} = 1 mA				100	mV
		I_{SINK} = 6 mA				300	
EN, WP, SCL SDA, ADD Input Leakage Current	I_{LEAK}				0.01	1	µA
Output High Leakage Current		$\overline{ALARM1}$, $\overline{ALARM2}$, SDA				1	µA
MTP Programming Time	T_{MTP}	2 Bytes				25	ms
Total MTP Programming Time	T_{MTP_TOTAL}	Note 1				650	ms
Total MTP Loading Time	T_{MTP_LOAD}					10	ms
MTP Write Cycles		T_A = +85°C (Note 2)		1000			cycles
LOGIC INPUT (EN, WP, SDA, SCL)							
Input Voltage Low	V_{IL}					0.4	V
Input Voltage High	V_{IH}			1.4			
I^2C/SMBus-COMPATIBLE TIMING							
Serial-Clock Frequency	f_{CLK}					400	kHz

(3.0V ≤ V_{CC} ≤ 3.6V = Power Supply Voltage, T_A = -40°C to +125°C = Ambient Temperature. All specifications are subject to change.) (Note 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Bus Free Time Between STOP and START Conditions	t_{BUF}	f_{CLK} = 400kHz	1.3			µs
Repeated START Condition Setup Time	$t_{SU:STA}$		0.6			µs
START Condition Setup Time		90% of SCL to 90% of SDA, f_{CLK} = 400kHz	0.6			µs
START Condition Hold Time	$t_{HD:STA}$	90% of SDA to 90% of SCL, f_{CLK} = 400kHz	0.6			µs
STOP Condition Setup Time	$t_{SU:STO}$	90% of SCL to 90% of SDA, f_{CLK} = 400kHz	0.6			µs
Clock Low Period	t_{LOW}	10% to 10%	1			µs
Clock High Period	t_{HIGH}	90% to 90%	1			µs
Data-In Hold Time	$t_{HD:DAT}$		0.3			µs
Data-In Setup Time	$t_{SU:DAT}$		100			ns
Receive Clock/Data Rise Time	t_R				300	ns
Receive Clock/Data Fall Time	t_F				300	ns
Pulse Width of Spike Suppressed	t_{SP}		0		50	ns
Bus Timeout	$t_{TIMEOUT}$		25		45	ms

Note 1: The MTP memory programming temperature range is -40°C to +85°C.

Note 2: Guaranteed by design.

Note 3: All devices are production tested at T_A = 25°C. Specifications over temperature limits are guaranteed by design.

Timing Diagrams

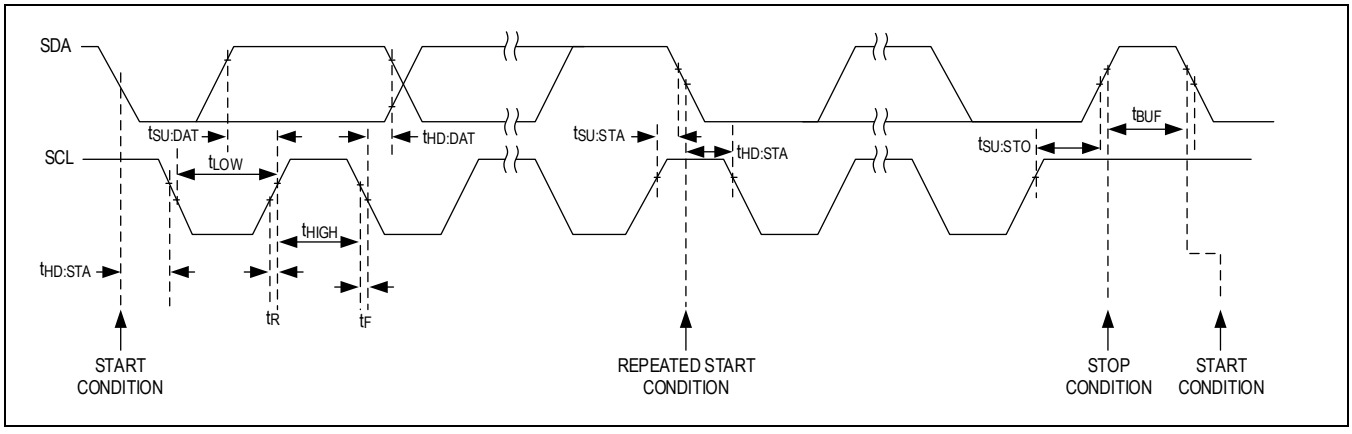
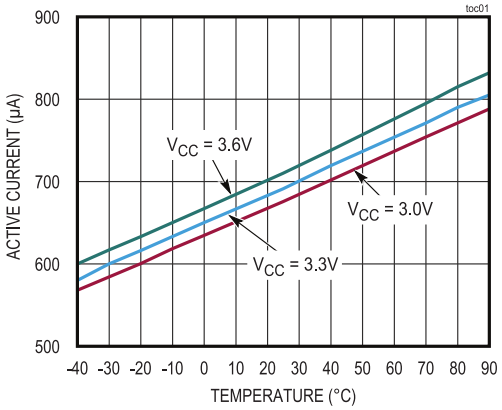


Figure 1. I²C/SMBus Timing Diagram

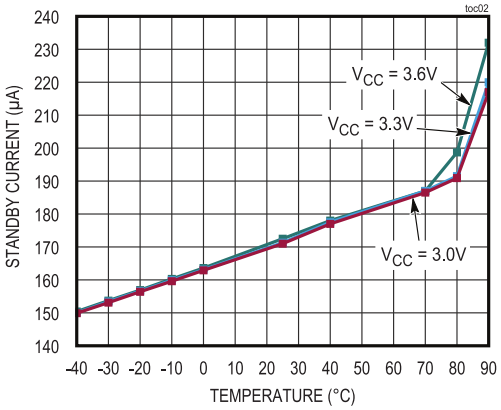
Typical Performance Characteristics

T_A = 25°C, unless otherwise noted.

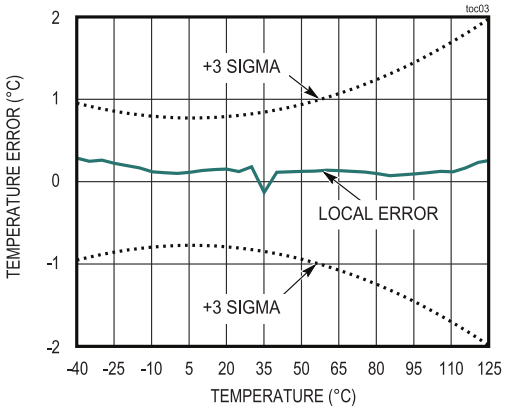
ACTIVE CURRENT vs. TEMPERATURE



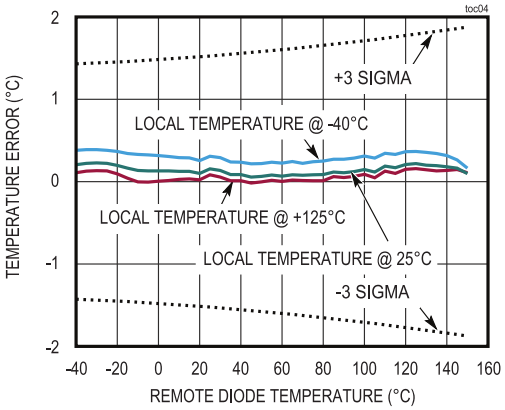
STANDBY CURRENT vs. TEMPERATURE



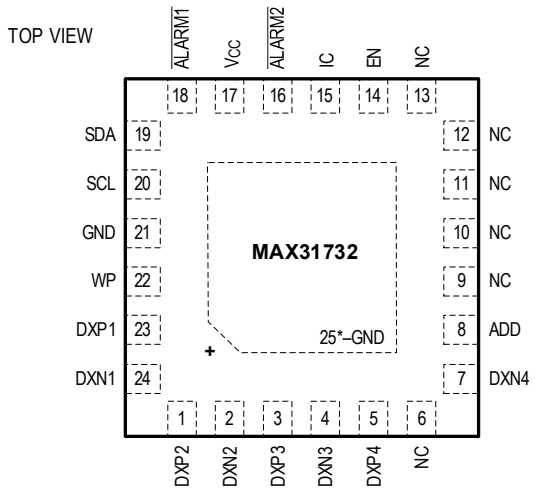
LOCAL TEMPERATURE ACCURACY



REMOTE TEMPERATURE ACCURACY



Pin Configurations



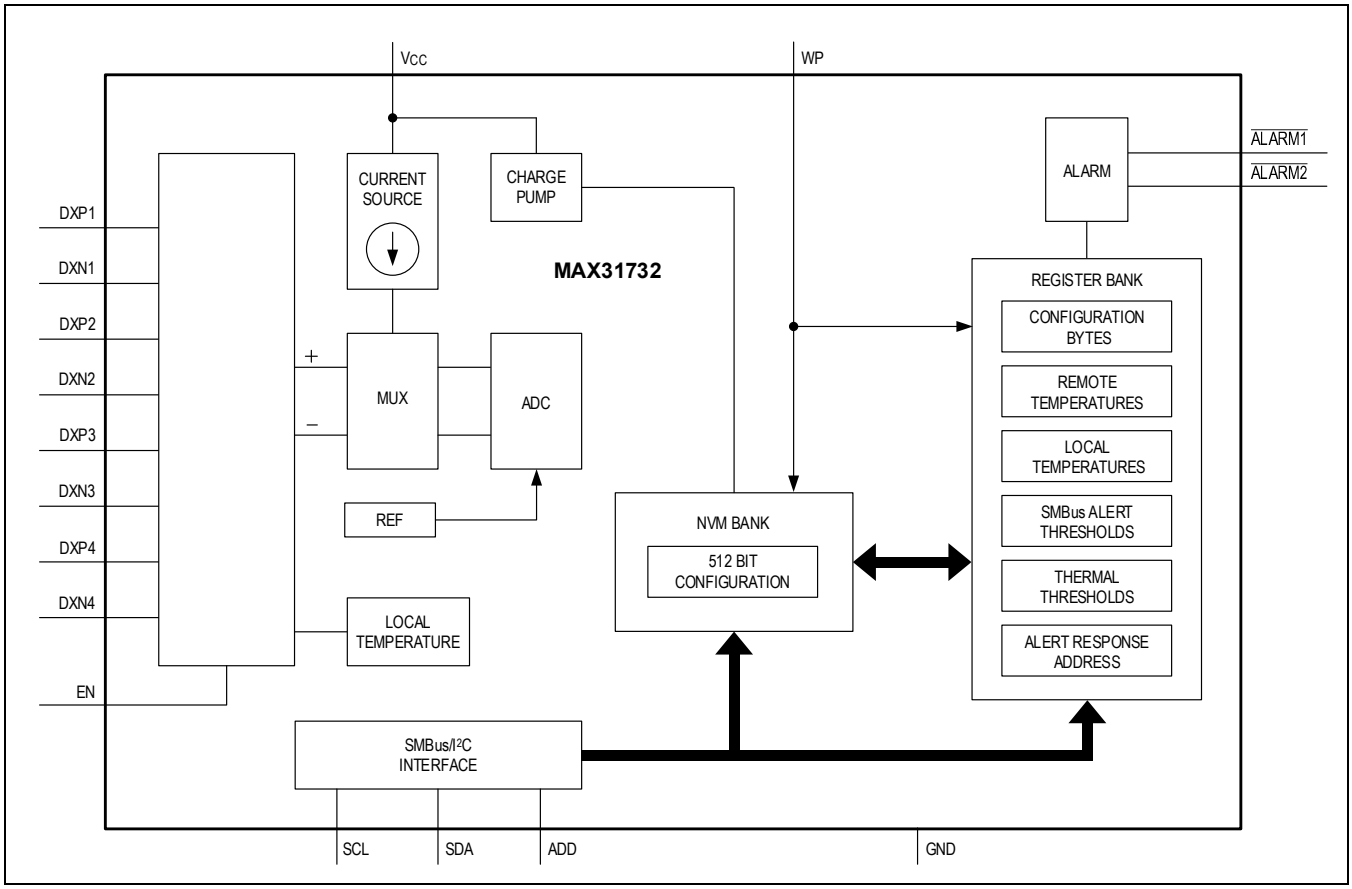
24-TQFN EP
*** EXPOSED PAD, CONNECT TO GND**

Pin Descriptions

PIN	NAME	FUNCTION
1	DXP2	Combined Current Source and Anode Input for Channel 2 Remote Diode. Connect DXP2 to the anode of a remote-diode-connected, temperature-sensing transistor. Leave DXP2 unconnected or connect to DXN2 if the channel 2 remote diode is not used. Connect a 100pF capacitor between DXP2 and DXN2 for noise filtering.
2	DXN2	Cathode Input for Channel 2 Remote Diode. Connect the DXN2 pin to GND and the negative input of the sensor, except in the case of a transistor where base-current compensation is used (connect DXN2 to the base of the transistor).
3	DXP3	Combined Current Source and Anode Input for Channel 3 Remote Diode. Connect DXP3 to the anode of a remote-diode-connected, temperature-sensing transistor. Leave DXP3 unconnected or connect to DXN3 if the channel 3 remote diode is not used. Connect a 100pF capacitor between DXP3 and DXN3 for noise filtering.
4	DXN3	Cathode Input for Channel 3 Remote Diode. Connect the DXN3 pin to GND and the negative input of the sensor except in the case of a transistor where the base-current compensation is used (connect DXN3 to the base of the transistor).
5	DXP4	Combined Current Source and Anode Input for Channel 4 Remote Diode. Connect DXP4 to the anode of a remote-diode-connected, temperature-sensing transistor. Leave DXP4 unconnected or connect to DXN4 if the channel 4 remote diode is not used. Connect a 100pF capacitor between DXP4 and DXN4 for noise filtering.
6, 9, 10, 11, 12, 13	NC	No Connection. Connect to other NCs or leave unconnected.
7	DXN4	Cathode Input for Channel 4 Remote Diode. Connect the DXN4 pin to GND and the negative input of the sensor except in the case of a transistor where the base-current compensation is used (connect DXN4 to the base of the transistor).
8	ADD	Address-Select Input. Sampled at power-up. One of the eight possible addresses can be selected by connecting ADD to GND.
14	EN	Enable Input. Drive EN high to enable the MAX31732. Drive EN low to place the MAX31732 in shutdown mode. In shutdown mode, DXN# and DXP# are high-impedance.
15	IC	Internally Connected. IC is internally connected to V_{CC}. Connect IC to V_{CC} or leave unconnected.
16	$\overline{\text{ALARM2}}$	Active-Low, Open-Drain Secondary Overtemperature/Undertemperature Alarm Output. $\overline{\text{ALARM2}}$ can also be used as an SMBus alert output by setting the device to interrupt mode using the Configuration

		register. When enabled, the pin asserts low when the temperature of any channel goes beyond the secondary programmed thresholds.
17	V_{CC}	Supply Voltage Input. Bypass to GND with a 0.1µF capacitor very close to the device.
18	$\overline{ALARM1}$	Active-Low, Open-Drain Primary Overtemperature/Undertemperature Output. $\overline{ALARM1}$ can also be used as an SMBus alert output by setting the device to interrupt mode using the Configuration register. When enabled, the pin asserts low when the temperature of any channel goes beyond a primary programmed threshold.
19	SDA	I^2C/SMBus Serial-Data Input/Output. Connect SDA to a pullup resistor.
20	SCL	I^2C/SMBus Serial-Clock Input. Connect SCL to a pullup resistor.
21	GND	Ground.
22	WP	Active-High Write Protect Input. Drive WP high to disable writing to volatile and nonvolatile memory. Drive WP low to enable writing to volatile and nonvolatile memory.
23	DXP1	Combined Current Source and Anode Input for Channel 1 Remote Diode. Connect DXP to the anode of a remote-diode-connected, temperature-sensing transistor. Leave DXP1 unconnected or connect to DXN1 if the channel 1 remote diode is not used. Connect a 100pF between DXP1 and DXN1 for noise filtering.
24	DXN1	Cathode Input for Channel 1 Remote Diode. Connect the DXN1 pin to GND and the negative input of the sensor except in the case of a transistor where the base-current compensation is used (connect DXN1 to the base of the transistor).
25	GND	Exposed Pad. Connect to GND.

Functional Diagram



Detailed Description

Overview

The MAX31732 is a precision temperature monitor that features one local and four remote temperature-sensing channels with programmable Overtemperature/Undertemperature thresholds for each channel. Each remote temperature channel is independently programmable to a custom ideality factor between 0.9844 and 1.311.

Communication with the device is achieved through the I²C/SMBus-compatible serial interface and the alarm outputs, $\overline{ALARM1}$ and $\overline{ALARM2}$. $\overline{ALARM1}$ asserts low when the temperature of any channel exceeds the software-programmed temperature thresholds in the *Primary Threshold Limit* registers. $\overline{ALARM2}$ asserts low when the temperature of any channel exceeds the software-programmed temperature thresholds in the *Secondary Threshold Limit* registers. $\overline{ALARM1}$ and $\overline{ALARM2}$ normally operate in comparator mode and can be connected to a fan, system shutdown, or other thermal-management circuitry. Alarm outputs can also operate in interrupt mode to serve as an SMBus alert interrupt.

The MAX31732 includes a 512-bit multi-time programmable nonvolatile memory (MTP) that helps facilitate device configuration during power-up or during normal operation if desired. During the power-up process, when V_{CC} exceeds the POR threshold voltage of 2.65V (typ), all configuration registers from the MTP are loaded into their respective registers in SRAM. The fault-logging registers in the MTP memory help preserve temperature fault data and the status bit for the channel that measures temperature outside of the programmed undertemperature and overtemperature threshold limits in the *Primary Threshold Limit* registers (0x1D–0x28) in *SRAM Register Map*.

ADC Conversion Sequence

When all channels are active, the MAX31732 measures the temperature on remote channel 1, followed by remote channel 2, remote channel 3, remote channel 4, and the local channel. The conversion result for each enabled channel is stored in the temperature data register (0x02–0x0B) in SRAM. No conversion can be performed on any channel with the conditions listed below:

- No diode connected between DXP# and DXN# inputs
- DXP# and DXN# are shorted together
- DXP# and DXN# are shorted to V_{CC}
- DXP# is shorted to GND (DXN# shorted to GND can also be enabled)
- Remote or local channel is not enabled in the Temperature Enable Register, 0x0E (SRAM)
- Beta Compensation is enabled, and the measured beta is less than 0.09 (Low Beta Fault)

Series-Resistance Cancellation

The series resistance of remote diodes can cause temperature measurement errors when used with conventional remote temperature sensors. The MAX31732 offers an always-on series-cancellation feature for remote channels 1–4 that eliminates the effect of diode series resistance and interconnection resistance. The cancellation range is from 0Ω to 300Ω.

Low Power Mode

Enter software-standby mode by setting the STOP bit to 1 in the Configuration1 register. Software-standby mode disables the ADC and reduces the supply current to approximately 150µA. During software standby, data is retained in memory and the bus interface is active and listening for commands. If a START condition is recognized, activity on the bus causes the supply current to increase. If a standby command is received while a conversion is in progress, the conversion cycle is finished, then the device enters shutdown, and the temperature registers are updated.

Enable Input (EN)

EN is an active-high logic-level input. When EN is pulled low, the following is true:

- The remote channel inputs (DXP# and DXN#) are forced into high-impedance mode.
- The internal ADC is disabled, and the current conversion is interrupted.
- The I²C/SMBus interface remains active to provide access to device registers.
- $\overline{ALARM1}$ and $\overline{ALARM2}$ outputs are disabled and pulled high through the external pull-up resistors.

Write-Protect Input (WP)

WP is an active-high logic-level input that write-protects the configuration registers in the MTP and the volatile SRAM when pulled high. Drive WP low to disable the write-protect mode and allow updates to device configuration registers through the I2C/SMBus interface. During write-protect mode, the I2C/SMBus communication interface can perform read operation only.

I2C/SMBus Digital Interface

The device is I2C/SMBus 2.0 compatible and supports four standard SMBus protocols: write byte, read byte, send byte, and receive byte, as well as block reads and writes (See *Figure 2)*. The shorter receive-byte protocol allows quicker transfers, provided that the correct register was previously selected by a read-byte instruction. Use caution with the shorter protocols in multi-controller systems since a second controller could overwrite the register byte without informing the first controller. *Figure 3* is the SMBus write timing diagram, and *Figure 4* is the SMBus read timing diagram. I2C burst read is limited to 16 data bytes for registers 0x0E and above. When reading back from the remote or local temperature or status registers (0x02 to 0x0D), the I2C burst read is limited to 2 bytes (LSB register followed by MSB register for a given sensor channel).

The **write-byte** format consists of the controller transmitting the target address, followed by the address for the targeted register, followed by the 8 bits of data to be written to the targeted register. To write multiple bytes to two or more contiguous registers, write a new byte after each ACK. The register address then increments after each byte is written. End the transaction with a STOP condition.

The **read-byte** format consists of the controller transmitting the target address followed by the address for the register to be read. The controller then begins a new transaction by sending the target address again, after which the target returns the data from the selected register. To read multiple bytes from two or more contiguous registers, continue reading after each ACK. The register address then increments after each byte is read. Conclude the overall transaction with a NACK and a STOP condition.

When the MSB byte of a 2-byte temperature value is read, the device prevents updates of the second byte's contents until the second byte has been read. If the second byte has not been read within an SMBus timeout period (nominally 35ms), it is again allowed to update.

The **send-byte** format can be used to transmit a register address without a transfer of data. It consists of the controller transmitting the target address followed by the address of the targeted register.

The **receive-byte** format can be used to read data from a register that was previously selected. It consists of the controller transmitting the target address, after which the target returns the data from the register that was previously selected. After this command completes, the address pointer does not increment.

WRITE-BYTE FORMAT

S	ADDRESS	WR	ACK	REGISTER	ACK	DATA	ACK	PEC	ACK	P
	7 BITS		0	8 BITS		8 BITS		8 BITS		

TARGET ADDRESS: EQUIVALENT TO CHIP-SELECT LINE OF A 3-WIRE INTERFACE

DATA BYTE: DATA GOES INTO THE REGISTER SET BY THE COMMAND BYTE (TO SET THRESHOLDS, CONFIGURATION MASKS, AND SAMPLING RATE)

MULTIPLE WRITE-BYTE FORMAT

S	ADDRESS	WR	ACK	REGISTER	ACK	DATA	ACK	.....	.....	DATA	ACK	DATA	ACK
	7 BITS	0		8 BITS		8 BITS		.....	.....	8 BITS		8 BITS	

.....	.....	DATA	ACK	.....	.....	DATA	ACK	DATA	ACK	PEC	ACK	P
.....	.....	8 BITS		.....	.....	8 BITS		8 BITS		8 BITS		

READ-BYTE FORMAT

S	ADDRESS	WR	ACK	REGISTER	ACK	S	ADDRESS	RD	ACK	DATA	PEC	NACK	P
	7 BITS	0		8 BITS			7 BITS	1		8 BITS	8 BITS		

TARGET ADDRESS: EQUIVALENT TO CHIP SELECT LINE

REGISTER BYTE: SELECTS WHICH REGISTER YOU ARE READING FROM

TARGET ADDRESS: REPEATED DUE TO CHANGE IN DATA FLOW DIRECTION

DATA BYTE: READS FROM THE REGISTER SET BY THE COMMAND BYTE

MULTIPLE READ-BYTE FORMAT

S	ADDRESS	WR	ACK	REGISTER	ACK	S	ADDRESS	RD	ACK	.....	.....	DATA	ACK	DATA	ACK
	7 BITS	0		8 BITS			7 BITS	1		.....	.....	8 BITS		8 BITS	

.....	.....	.....	DATA	ACK	.....	.....	DATA	ACK	DATA	ACK	PEC	NACK	P
.....	.....	.....	8 BITS		.....	.....	8 BITS		8 BITS		8 BITS		

SEND-BYTE FORMAT

S	ADDRESS	WR	ACK	REGISTER	ACK	PEC	P
	7 BITS	0		8 BITS		8 BITS	

RECEIVE-BYTE FORMAT

S	ADDRESS	RD	ACK	DATA	ACK	PEC	NACK	P
	7 BITS	1		8 BITS		8 BITS		

REGISTER BYTE: SENDS COMMAND WITH NO DATA, USUALLY USED FOR ONE SHOT COMMAND

DATA BYTE: READS DATA FROM THE REGISTER COMMANDED BY THE LAST READ-BYTE OR WRITE-BYTE TRANSMISSION; ALSO USED FOR SMBus ALERT RESPONSE RETURN ADDRESS

S = START CONDITION P = STOP CONDITION SHADED = TARGET TRANSMISSION

Figure 2. I^2C/SMBus Format

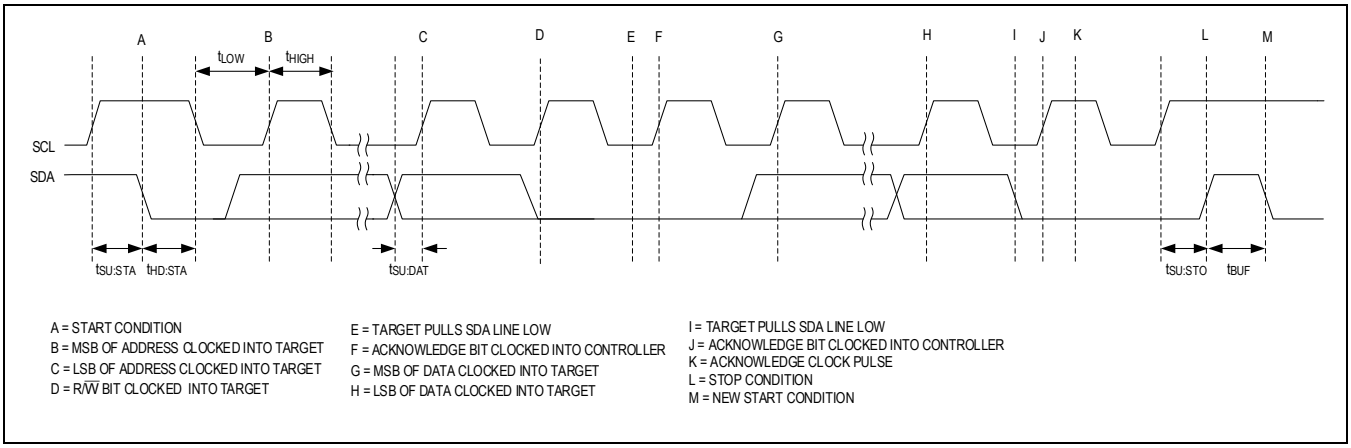


Figure 3. I²C/SMBus Write Timing Diagram

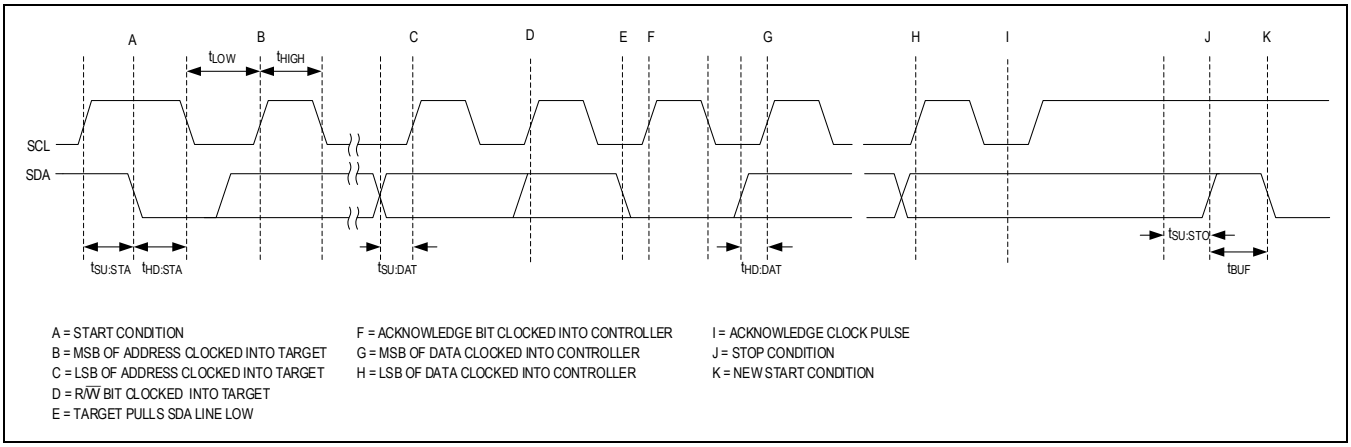


Figure 4. I²C/SMBus Read Timing Diagram

SRAM Register Map

SRAM REGISTER ADD (HEX)	DEFAULT VALUE (HEX)	READ/ WRITE	REGISTERS NAMES	FUNCTIONS
0x00	4F	R	Manufacturer ID	Reads Manufacturer ID.
0x01	01	R	Revision Code	Reads Die Revision.
0x02	00	R	Remote 1 Temperature Data (MSB)	Reads Channel 1 Remote Temperature MSB.
0x03	00	R	Remote 1 Temperature Data (LSB)	Reads Channel 1 Remote Temperature LSB.
0x04	00	R	Remote 2 Temperature Data (MSB)	Reads Channel 2 Remote Temperature MSB.
0x05	00	R	Remote 2 Temperature Data (LSB)	Reads Channel 2 Remote Temperature LSB.
0x06	00	R	Remote 3 Temperature Data (MSB)	Reads Channel 3 Remote Temperature MSB.
0x07	00	R	Remote 3 Temperature Data (LSB)	Reads Channel 3 Remote Temperature LSB.
0x08	00	R	Remote 4 Temperature Data (MSB)	Reads Channel 4 Remote Temperature MSB.
0x09	00	R	Remote 4 Temperature (LSB)	Reads Channel 4 Remote Temperature LSB.
0x0A	00	R	Local Temperature Data (MSB)	Reads Local Channel Temperature MSB.

SRAM REGISTER ADD (HEX)	DEFAULT VALUE (HEX)	READ/ WRITE	REGISTERS NAMES	FUNCTIONS
0x0B	00	R	Local Temperature Data (LSB)	Reads Local Channel Temperature LSB.
0x0C	00	R	Primary Thermal High Status	Reads the High-temperature Thermal Status for each Channel set up for the Primary Temperature limit.
0x0D	00	R	Primary Thermal Low Status	Reads the Low-temperature Thermal Status for each Channel set up for the Primary Temperature limit.
0x0E	9F	R/W	Temperature Channel Enable	Reads/Writes the Temperature Channel Enable bits and MTP Configuration Load bit.
0x0F	10	R/W	Configuration 1	Configures the ADCs Conversion Mode, $\overline{\text{ALARM1}}$ Mode of Operation, Thermal Fault Queue, and Temperature Data Format (extended or standard), and Enables Single-cycle Conversion.
0x10	11	R/W	Configuration 2	Configures $\overline{\text{ALARM2}}$ Mode of Operation, sets Thermal Fault Queue, and Enables Overtemperature and UnderTemperature Packet-Error Checking.
0x11	18	R/W	Remote 1 Channel Custom Ideality	Sets a Custom Ideality Factor for Remote 1–Sensing diodes.
0x12	18	R/W	Remote 2 Channel Custom Ideality	Sets a Custom Ideality Factor for Remote 2–Sensing diodes.
0x13	18	R/W	Remote 3 Channel Custom Ideality	Sets a Custom Ideality Factor for Remote 3–Sensing diodes.
0x14	18	R/W	Remote 4 Channel Custom Ideality	Sets a Custom Ideality Factor for Remote 4–Sensing diodes.
0x15	00	R/W	Custom Ideality Factor Enable Register	Sets the Nominal Ideality (1.008) or the Custom Ideality for each Remote Channel.
0x16	77	R/W	Custom Offset	Selects an Offset Value for Temperature Measurement. The device default is -14.875°C, with a programmed value of +14.875°C, which leaves a summed offset of 0°C default from the Factory/POR.
0x17	00	R/W	Custom Offset Enable	Enables/Disables the Custom Offset Temperature value for each channel.
0x18	00	R/W	Filter Enable	Enables/Disables filter for each Remote Channel (should be disabled when not in constant conversion mode).
0x19	00	R/W	Beta Compensation Enable	Enables/disables Beta Compensation for each Remote Channel.
0x1A	1E	R/W	Highest Temperature Enable	Selects which Channels are used in determining the contents of the highest Temperature Registers.

SRAM REGISTER ADD (HEX)	DEFAULT VALUE (HEX)	READ/ WRITE	REGISTERS NAMES	FUNCTIONS
0x1B	00	R/W	$\overline{\text{ALARM1}}$ Mask	Masks Faults from asserting $\overline{\text{ALARM1}}$ for each Channel.
0x1C	00	R/W	$\overline{\text{ALARM2}}$ Mask	Masks Faults from asserting $\overline{\text{ALARM2}}$ for each Channel.
0x1D	7F	R/W	Remote 1 Primary Overtemperature Threshold (MSB)	Read/Write the Remote 1 Channel's Primary Overtemperature Threshold MSB.
0x1E	F0	R/W	Remote 1 Primary Overtemperature Threshold (LSB)	Read/Write the Remote 1 Channel's Primary Overtemperature threshold LSB.
0x1F	7F	R/W	Remote 2 Primary Overtemperature Threshold (MSB)	Read/Write the Remote 2 Channel's Primary Overtemperature Threshold MSB.
0x20	F0	R/W	Remote 2 Primary Overtemperature Threshold (LSB)	Read/Write the Remote 2 Channel's Primary Overtemperature Threshold LSB.
0x21	7F	R/W	Remote 3 Primary Overtemperature Threshold (MSB)	Read/Write the Remote 3 Channel's Primary Overtemperature Threshold MSB.
0x22	F0	R/W	Remote 3 Primary Overtemperature Threshold (LSB)	Read/Write the Remote 3 Channel's Primary Overtemperature Threshold LSB.
0x23	7F	R/W	Remote 4 Primary Overtemperature Threshold (MSB)	Read/Write the Remote 4 Channel's Primary Overtemperature Threshold MSB.
0x24	F0	R/W	Remote 4 Primary Overtemperature Threshold (LSB)	Read/Write the Remote 4 Channel's Primary Overtemperature Threshold LSB.
0x25	7F	R/W	Local Primary Overtemperature Threshold (MSB)	Read/Write the Local Channel's Primary Overtemperature Threshold MSB.
0x26	F0	R/W	Local Primary Overtemperature Threshold (LSB)	Read/write the Local Channel's Primary Overtemperature Threshold LSB.
0x27	C9	R/W	All Channel Primary Undertemperature Low Threshold (MSB)	Read/Write the Primary Undertemperature threshold for all Channel MSB
0x28	00	R/W	All Channel Primary Undertemperature Low Threshold (LSB)	Read/Write the Primary Undertemperature Threshold for all Channel LSB
0x29	7F	R/W	Remote 1 Secondary Overtemperature Threshold	Read/Write the Remote 1 Channel's Secondary Overtemperature Threshold.
0x2A	7F	R/W	Remote 2 Secondary Overtemperature Threshold	Read/Write the Remote 2 Channel's Secondary Overtemperature threshold.
0x2B	7F	R/W	Remote 3 Secondary Overtemperature Threshold	Read/Write the Remote 3 Channel's Secondary Overtemperature Threshold.
0x2C	7F	R/W	Remote 4 Secondary Overtemperature Threshold	Read/Write the Remote 4 Channel's Secondary Overtemperature Threshold.
0x2D	7F	R/W	Local Secondary Overtemperature Threshold	Read/Write the Local Channel's Secondary Overtemperature Threshold.
0x2E	C9	R/W	All Channel Secondary Undertemperature Threshold	Read/Write the Secondary Undertemperature Threshold for all Channels.
0x2F	00	R/W	Remote 1 Reference Temperature (MSB)	Read/Write the Reference Temperature for Remote 1 Channel MSB.

SRAM REGISTER ADD (HEX)	DEFAULT VALUE (HEX)	READ/ WRITE	REGISTERS NAMES	FUNCTIONS
0x30	00	R/W	Remote 1 Reference Temperature (LSB)	Read/Write the Reference Temperature for Remote 1 Channel LSB.
0x31	00	R/W	Remote 2 Reference Temperature (MSB)	Read/Write the Reference Temperature for Remote 2 Channel MSB.
0x32	00	R/W	Remote 2 Reference Temperature (LSB)	Read/Write the Reference Temperature for Remote 2 Channel LSB.
0x33	00	R/W	Remote 3 Reference Temperature (MSB)	Read/Write the Reference Temperature for Remote 3 Channel MSB.
0x34	00	R/W	Remote 3 Reference Temperature (LSB)	Read/Write the Reference Temperature for Remote 3 channel LSB.
0x35	00	R/W	Remote 4 Reference Temperature (MSB)	Read/Write the Reference Temperature for Remote 4 Channel MSB.
0x36	00	R/W	Remote 4 Reference Temperature (LSB)	Read/Write the Reference Temperature for Remote 4 Channel LSB.
0x37	00	R/W	Local Reference Temperature (MSB)	Read/Write the Reference Temperature for Local Channel MSB.
0x38	00	R/W	Local Reference Temperature (LSB)	Read/Write the Reference Temperature for Local Channel LSB.
0x39	1F	R/W	MTP Configuration	Enables fault-logging function into the MTP.
0x3A	00	R/W	MTP Configuration 2	Enables Storing Configuration Data into the MTP, Loading Configuration Data from the MTP, and Reading the MTP registers.
0x3B	00	R/W	MTP Address	Read/Write the Address of Customer Software Revision Registers.
0x3C	00	R/W	MTP_DIN (MSB) – Data In	Read/Write the Customer Software Revision Code MSB.
0x3D	00	R/W	MTP_DIN (LSB) – Data In	Read/Write the Customer Software Revision Code LSB.
0x42	00	R	Secondary Thermal High Status	Read the High-temperature Thermal Status for each Channel set up for the Secondary Temperature Limit.
0x43	00	R	Secondary Thermal Low Status	Read the Low-temperature Thermal Status for each Channel set up for the Secondary Temperature Limit.
0x44	00	R	Diode Fault Status	Read the Diode Fault Status for each Channel.
0x45	00	R	Highest Temperature (MSB)	Highest Current Temperature Value MSB. The value in the highest temperature register is the greater of all (temperature channel value minus the channel reference temperature value).
0x46	00	R	Highest Temperature (LSB)	Highest Current Temperature Value LSB. The value in the highest temperature register is the greater of all (temperature channel value minus the channel reference temperature value).

SRAM REGISTER ADD (HEX)	DEFAULT VALUE (HEX)	READ/ WRITE	REGISTERS NAMES	FUNCTIONS
0x47	00	R	Beta Value Remote 1	Contains the Beta compensation Value for Channel 1.
0x48	00	R	Beta Value Remote 2	Contains the Beta compensation Value for Channel 2.
0x49	00	R	Beta Value Remote 3	Contains the Beta compensation Value for Channel 3.
0x4A	00	R	Beta Value Remote 4	Contains the Beta compensation Value for Channel 4.

Temperature Measurement (Address = 0x02 to 0x0B)

Each temperature sensor in the MAX31732 is capable of monitoring two programmable threshold levels, referred to as Primary Threshold Limit and Secondary Threshold Limit. Bi-level temperature threshold monitoring allows systems to perform more reliably as one level can be used as a warning to the system. The temperature resolution for the Primary Threshold Limit is 12 bits with an LSB of 0.0625°C. The conversion result is presented in two's-complement with the MSB as a sign bit. After each conversion is completed, the result is stored in the upper byte (MSB) and lower byte (LSB) of the Temperature Data registers (0x02–0x0B). The MSB register must be read first, followed by the LSB register for any given channel or highest temperature read-back. See *Table 1* for the Temperature Data register format.

Table 1. Temperature Data Format

UPPER BYTE (MSB)								LOWER BYTE (LSB)							
D15	**D14**	**D13**	**D12**	**D11**	**D10**	**D9**	**D8**	**D7**	**D6**	**D5**	**D4**	**D3**	**D2**	**D1**	**D0**
SIGN BIT	MSB										LSB				
	64°C	32°C	16°C	8°C	4°C	2°C	1°C	0.5°C	0.25°C	0.125°C	0.0625°C	0	0	0	0
	2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}

The temperature resolution for the Secondary Threshold Limit is 8 bits with an LSB equal to 1°C. The conversion result is presented in two's-complement with the MSB as a sign bit. After each conversion is completed, the result is stored in the upper byte of the Temperature Data registers shown in *Table 1*.

Extended Temperature Measurement

In addition to the normal two's-complement temperature data format, the device offers an optional extended data format that allows temperatures greater than +127.9375°C to be read for the Primary threshold limits and +127°C for the Secondary threshold limits. In the extended format (selected by bit 1 of the Configuration1 register, 0x0F), the measured temperature is the value in the Temperature Data register plus 64°C, as shown in *Table 2* and *Table 3*.

Table 2. Primary Temperature Data Format

ACTUAL TEMPERATURE (°C)	NORMAL FORMAT		EXTENDED FORMAT	
	BINARY	HEX	BINARY	HEX
+150	0111 1111 1111 0000*	0x7FF0*	0101 0110 0000 0000	0x5600
+128	0111 1111 1111 0000*	0x7FF0*	0100 0000 0000 0000	0x4000
+127	0111 1111 0000 0000	0x7F00	0011 1111 0000 0000	0x3F00
+125	0111 1101 0000 0000	0x7D00	0011 1101 0000 0000	0x3D00
+64	0100 0000 0000 0000	0x4000	0000 0000 0000 0000	0x0000
+25	0001 1001 0000 0000	0x1900	1101 1001 0000 0000	0xD900
+0.5	0000 0000 1000 0000	0x0080	1100 0000 1000 0000	0xC080
0	0000 0000 0000 0000	0x0000	1100 0000 0000 0000	0xC000
-0.5	1111 1111 1000 0000	0xFF80	1011 1111 1000 0000	0xBF80
-25	1110 0111 0000 0000	0xE700	1010 0111 0000 0000	0xA700
-55	1100 1001 0000 0000	0xC900	1000 1001 0000 0000	0x8900
-64	1100 0000 0000 0000	0xC000	1000 0000 0000 00000	0x8000
DIODE FAULT	0000 0000 0000 0000	0x0000	0000 0000 0000 0000	0x0000

Note: When the extended format is selected, all temperature threshold limits and reference temperature registers must be written in this format. Toggling the extended format bit does not automatically translate them.
**overrange*

Table 3. Secondary Temperature Data Format

ACTUAL TEMPERATURE (°C)	NORMAL FORMAT		EXTENDED FORMAT	
	BINARY	HEX	BINARY	HEX
+150	0111 1111	0x7F*	0101 0110	0x56
+128	0111 1111	0x7F*	0100 0000	0x40
+127	0111 1111	0x7F	0011 1111	0x3F
+125	0111 1101	0x7D	0011 1101	0x3D
+64	0100 0000	0x40	0000 0000	0x00
+25	0001 1001	0x19	1101 1001	0xD9
0	0000 0000	0x00	1100 0000	0xC0
-25	1110 0111	0xE7	1010 0111	0xA7
-55	1100 1001	0xC9	1000 1001	0x89
-64	1100 0000	0xC0	1000 0000	0x80
DIODE FAULT	0000 0000	0x00	0000 0000	0x00

Note: *When the extended format is selected, all temperature threshold limits and reference temperature registers must be written in this format. Toggling the extended format bit does not automatically translate them.*
**overrange.*

Temperature Channel Enable Register (Address = 0x0E)

Bit 7 of the *Temperature Channel Enable* register (0x0E) enables MTP loading during power or when loading of the MTP is required during normal operation. The default value for bit 7 is 1 but can be set to 0 during power-up by the matching register (0x8E) in the MTP memory. See the *MTP Register Map* (*Table 33*) for more details. If that happens, the MAX31732 does not allow the loading of the MTP configuration registers into SRAM. To properly load the MTP registers during power up, ensure bit 7 of the *Temperature Channel Enable* register is set to 1 in SRAM and the MTP memory.

The *Temperature Channel Enable* register also selects which temperature-sensing channels are enabled using bit 4–0. Channels not selected are skipped during the temperature-conversion cycle, and diode fault detection is not performed on them. If a channel is deselected while a thermal or diode fault is indicated in the corresponding fault register, the fault bit(s) remain asserted until the register contents are read, and then do not reassert until the channel is again enabled and a fault detected, see *Table 4* for more details. Disabling a temperature-sensing channel or setting the EN pin to logic low will not clear its respective temperature data register.

Table 4. Temperature Enable Register

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	EN_MTP_PU_LOAD	1	MTP loading enable Bit. This bit is automatically set to either 0 or 1 during power-up from the MTP register. Setting this bit to 0 during power-up prevents the configuration registers in the MTP from being loaded into the SRAM configuration registers. Setting this bit to 1 at power-up allows all the configuration registers from MTP to be loaded into the SRAM configuration registers.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	EN_REMOTE4	1	Channel 4 Enable Bit. Set this bit to Logic 0 to disable temperature measurement on temperature Channel 4. Set this bit to 1 to enable temperature measurement on Channel 4.
3	EN_REMOTE3	1	Channel 3 Enable Bit. Set this bit to Logic 0 to disable temperature measurement on temperature Channel 3. Set this bit to 1 to enable temperature measurement on Channel 3.
2	EN_REMOTE2	1	Channel 2 Enable Bit. Set this bit to logic 0 to disable temperature measurement on temperature Channel 2. Set this bit to 1 to enable temperature measurement on Channel 2.
1	EN_REMOTE1	1	Channel 1 Enable Bit. Set this bit to logic 0 to disable temperature measurement on temperature Channel 1. Set this bit to 1 to enable temperature measurement on Channel 1.
0	EN_LOCAL	1	Local Channel Enable Bit. Set this bit to logic 0 to disable temperature measurement on the Local channel. Set this bit to 1 to enable temperature measurement on the Local channel.

Highest Temperature Registers (Address = 0x45 and 0x46)

The *Highest Temperature Data* registers (0x45/0x46) work with the *Reference Temperature* registers' values for each temperature channel. The *Reference Temperature* registers can effectively serve as an offset temperature margin, or their contents can simply be set to zero.

After each temperature conversion, the *Reference Temperature* value is subtracted from the measured temperature for the corresponding channel (e.g., remote 2 temperature minus remote 2 reference temperature), and the result is compared to the most recent results for the other channels. The highest of all these values is loaded into the *Highest Temperature Data* registers. See *Table 5* and *Table 6* below for the bit format. If configuration register, 0x0F, bit 1 is set, add 64°C to the read-back value.

Table 5. Highest Temperature Data Format (MSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D15	D14	D13	D12	D11	D10	D9	D8
Sign	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Table 6. Highest Temperature Data Format (LSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D7	D6	D5	D4	D3	D2	D1	D0
2^{-1}	2^{-2}	2^{-3}	2^{-4}	0	0	0	0

Value in the highest temperature registers will be the greater of all (the temperature channel value minus the channel reference temperature value).

Highest Temperature Enable Register (Address = 0x1A)

The *Highest Temperature Enable* register (0x1A) selects the temperature channels from which the contents of the Highest Temperature registers are obtained, see *Table 7* for more details.

Table 7. Highest Temperature Enable Register

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	1	Channel 4 Select Bit. Set to Logic 1 to use Remote Channel 4 in determining the highest temperature.
3	Remote 3	1	Channel 3 Select Bit. Set to Logic 1 to use Remote Channel 3 in determining the highest temperature.
2	Remote 2	1	Channel 2 Select Bit. Set to Logic 1 to use Remote Channel 2 in determining the highest temperature.
1	Remote 1	1	Channel 1 Select Bit. Set to Logic 1 to use Remote Channel 1 in determining the highest temperature.
0	Local	1	Local Channel Select Bit. Set to Logic 1 to use the Local Channel in determining the highest temperature.

Reference Temperature Registers (Address = 0x2F to 0x38)

The Reference Temperature registers (0x2F-0x38) are used to determine the highest temperature channel. See the *Highest Temperature Registers (Address = 0x45 and 0x46)* section for more details. See *Table 8* and *Table 9* for the Reference Temperature register format.

Table 8. Reference Temperature (MSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D15	D14	D13	D12	D11	D10	D9	D8
Sign	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Note: *When the extended format is selected, reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

Table 9. Reference Temperature (LSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D7	D6	D5	D4	D3	D2	D1	D0
2^{-1}	2^{-2}	2^{-3}	2^{-4}	0	0	0	0

Note: *When the extended format is selected, all limit and reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

Overtemperature Threshold Registers
(Primary Address = 0x1D to 0x26, Secondary Address = 0x29 to 0x2D)

The over-temperature threshold registers store overtemperature thermal-threshold values. Access to these registers is provided through the I²C/SMBus-compatible interface. The *Primary Overtemperature Threshold* registers (0x1D–0x26) store over-temperature threshold limits for the primary threshold monitoring. If the temperature of a channel exceeds the programmed threshold, a bit is set in the *Primary Thermal Status High* register to indicate the thermal fault, and $\overline{\text{ALARM1}}$ is asserted unless masked by the *$\overline{\text{ALARM1}}$ Mask* register. See Table 10 and Table 11 for data format.

Table 10. Primary Overtemperature Threshold (MSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D15	D14	D13	D12	D11	D10	D9	D8
Sign	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}

Note: *When the extended format is selected, all limit and reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

Table 11. Primary Overtemperature Threshold (LSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D7	D6	D5	D4	D3	D2	D1	D0
2^{-1}	2^{-2}	2^{-3}	2^{-4}	0	0	0	0

Note: *When the extended format is selected, all limit and reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

The *Secondary Overtemperature Threshold* registers (0x29–0x2D) store data for the secondary threshold monitoring. If a channel's temperature exceeds the programmed threshold, a bit is set in the *Secondary Thermal Status High* register to indicate the thermal fault, and $\overline{\text{ALARM2}}$ is asserted unless masked by the *$\overline{\text{ALARM2}}$ Mask* register. See Table 12 for the data format.

To minimize the chance of excessive MTP writes (if MTP write is enabled), it is recommended that the Secondary Over-Temperature Threshold be set to a value lower than the Primary Overtemperature Threshold (only crossing the primary threshold forces an MTP write). In doing so, the $\overline{\text{ALARM2}}$ pin would alert the system that the monitored temperatures are above the secondary threshold and provide time for any necessary action before the temperature potentially rises above the primary threshold (which triggers a write to the MTP if enabled).

Table 12. Secondary Overtemperature Threshold

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D7	D6	D5	D4	D3	D2	D1	D0
Sign	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}

Note: *When the extended format is selected, all limit and reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

Undertemperature Threshold Registers

(Primary Address = 0x27 to 0x28, Secondary Address = 0x2E)

The undertemperature threshold registers store undertemperature thermal-threshold values. Access to these registers is provided through the I2C/SMBus-compatible interface. The *All-Channel Primary Undertemperature Threshold* registers (0x27/0x28) store primary threshold data for all channels. If the temperature of any channel drops below the programmed threshold, a bit is set in the *Primary Thermal Status Low* register to indicate the thermal fault, and $\overline{\text{ALARM1}}$ is asserted unless masked by the *$\overline{\text{ALARM1}}$ Mask* register. See Table 13 and Table 14 for data format.

Table 13. All-Channel Primary Undertemperature Threshold (MSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D15	D14	D13	D12	D11	D10	D9	D8
Sign	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Note: *When the extended format is selected, all limit and reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

Table 14. All-Channel Primary Undertemperature Threshold (LSB)

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D7	D6	D5	D4	D3	D2	D1	D0
2^{-1}	2^{-2}	2^{-3}	2^{-4}	0	0	0	0

Note: *When the extended format is selected, all limit and reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

The *All Channel Secondary Undertemperature Threshold* register (0x2E) stores secondary threshold data for all channels. If the temperature of any channel drops below the programmed threshold, a bit is set in the *Secondary Thermal Status Low* register to indicate the thermal fault, and $\overline{\text{ALARM2}}$ is asserted unless masked by the *$\overline{\text{ALARM2}}$ Mask* register. See Table 15 for the data format.

Table 15. All Channel Secondary Undertemperature Threshold

BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D7	D6	D5	D4	D3	D2	D1	D0
Sign	2^6	2^5	2^4	2^3	2^2	2^1	2^0

Note: *When the extended format is selected, all limit and reference temperature registers must be written in the extended temperature format. Toggling the extended format bit does not automatically translate them.*

Status Registers

The thermal Status registers indicate overtemperature and undertemperature faults. The *Primary Thermal High-Status* register (0x0C) indicates whether a measured local or remote temperature has exceeded the threshold limit set in the associated Primary Overtemperature Threshold registers. See Table 16 for more details. The Primary Thermal Low-Status register (0x0D) indicates whether the measured temperature has fallen below the threshold limit set in the All Channel Primary Undertemperature Threshold registers for the local or remote-sensing diodes. See Table 17 for more details. Bits in the thermal status registers are cleared by a successful read but set again after the next conversion unless the fault is corrected, either by a change in the measured temperature or by a change in the threshold temperature.

Table 16. Primary Thermal High Status Register (Address = 0x0C)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Remote 4–Diode Primary Status High Bit. This bit is set to Logic 1 when the Channel 4 remote-diode temperature exceeds the primary threshold in the Remote 4 Thermal High Threshold registers.
3	Remote 3	0	Remote 3–Diode Primary Status High Bit. This bit is set to Logic 1 when the Channel 3 remote-diode temperature exceeds the primary threshold in the Remote 3 Thermal High Threshold registers.
2	Remote 2	0	Remote 2–Diode Primary Status High Bit. This bit is set to Logic 1 when the Channel 2 remote-diode temperature exceeds the primary threshold in the Remote 2 Thermal High Threshold registers.
1	Remote 1	0	Remote 1–Diode Primary Status High Bit. This bit is set to Logic 1 when the Channel 1 remote-diode temperature exceeds the primary threshold in the Remote 1 Thermal High Threshold registers.
0	Local	0	Local Channel Primary Statis High Bit. This bit is set to Logic 1 when the local temperature exceeds the primary threshold in the local Thermal High Threshold registers.

Table 17. Primary Thermal Low Status Register (Address = 0x0D)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Remote 4–Diode Primary Status Low Bit. This bit is set to Logic 1 when the Channel 4 remote-diode temperature exceeds the primary threshold in the Thermal Low Threshold registers.
3	Remote 3	0	Remote 3–Diode Primary Status Low Bit. This bit is set to Logic 1 when the Channel 3 remote-diode temperature exceeds the primary threshold in the Thermal Low Threshold registers.
2	Remote 2	0	Remote 2–Diode Primary Status Low Bit. This bit is set to Logic 1 when the Channel 2 remote-diode temperature exceeds the primary threshold in the Thermal Low Threshold registers.
1	Remote 1	0	Remote 1-Diode Primary Status Low Bit. This bit is set to Logic 1 when the Channel 1 remote-diode temperature exceeds the primary threshold in the Thermal Low Threshold registers
0	Local	0	Local Channel Primary Status Low Bit. This bit is set to Logic 1 when the local temperature falls below the primary threshold in the Thermal Low Threshold registers

The *Secondary Thermal High-Status* register (0x42) indicates whether a measured local or remote temperature has exceeded the threshold limit set in the Secondary Overtemperature Threshold registers. See *Table 18* for more details. The Secondary Thermal Low-Status register indicates whether the measured temperature has fallen below the threshold limit set in the *All-Channel Secondary Undertemperature Threshold* register for the local or remote-sensing diodes.

See *Table 19* for more details. Bits in the thermal status registers are cleared by a successful read but set again after the next conversion unless the fault is corrected, either by a change in the measured temperature or by a change in the threshold temperature.

Table 18. Secondary Thermal High-Status Register (Address = 0x42)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Remote 4-Diode Secondary Status High Bit. This bit is set to Logic 1 when the channel 4 remote-diode temperature exceeds the secondary threshold in the Remote 4 Thermal High Threshold registers.
3	Remote 3	0	Remote 3-Diode Secondary Status High Bit. This bit is set to Logic 1 when the channel 3 remote-diode temperature exceeds the secondary threshold in the Remote 3 Thermal High Threshold registers.
2	Remote 2	0	Remote 2-Diode Secondary Status High Bit. This bit is set to Logic 1 when the channel 2 remote-diode temperature exceeds the secondary threshold in the Remote 2 Thermal High Threshold registers.
1	Remote 1	0	Remote 1-Diode Secondary Status High Bit. This bit is set to Logic 1 when the channel 1 remote-diode temperature exceeds the secondary threshold in the Remote 1 Thermal High Threshold registers.
0	Local	0	Local Channel Secondary Status High Bit. This bit is set to Logic 1 when the local temperature exceeds the secondary threshold in the Thermal High Threshold registers.

Table 19. Secondary Thermal Low Status Register (Address = 0x43)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Remote4-Diode Secondary Status Low Bit. This bit is set to logic 1 when the channel 4 remote-diode temperature exceeds the Secondary threshold in the Thermal Low Threshold registers.
3	Remote 3	0	Remote3-Diode Secondary Status Low Bit. This bit is set to logic 1 when the channel 3 remote-diode temperature exceeds the Secondary threshold in the Thermal Low Threshold registers.
2	Remote 2	0	Remote2-Diode Secondary Status Low Bit. This bit is set to logic 1 when the channel 2 remote-diode temperature exceeds the Secondary threshold in the Thermal Low Threshold registers.
1	Remote 1	0	Remote1-Diode Secondary Status Low Bit. This bit is set to logic 1 when the channel 1 remote-diode temperature exceeds the secondary threshold in the Thermal Low Threshold registers.
0	Local	0	Local channel Secondary Status Low Bit. This bit is set to logic 1 when the local temperature exceeds the secondary threshold in the Thermal Low Threshold registers.

When in comparator mode, reading the status registers has no effect on the $\overline{ALARM1}$ and $\overline{ALARM2}$ outputs state; the state depends on the current temperature, threshold, and mask values.

In interrupt mode, the $\overline{ALARM1}$ and $\overline{ALARM2}$ outputs follow the status bits for all unmasked channels. Once the $\overline{ALARM1}$ and $\overline{ALARM2}$ outputs are asserted while in interrupt mode, they can be deasserted either by reading the thermal status register or by successfully responding to the ARA (Alert Response Address) command. In both cases, the $\overline{ALARM1}$ and $\overline{ALARM2}$ pins are cleared even if the fault condition remains in effect, but the $\overline{ALARM1}$ and $\overline{ALARM2}$ output reasserts at the end of the next conversion if the fault condition is still present.

Diode Fault Detection (Address =0x44)

If a remote channel's DXP# and DXN# inputs are unconnected or are shorted to each other, to ground, or to the supply voltage, the device detects a diode fault. A diode fault does not cause alarm outputs to assert and does not allow an over-temperature or under-temperature event to be detected for the affected channel. A bit in the *Diode Fault Status* register (0x44) corresponding to the channel is set to 1 and the temperature data for the channel is stored as 0°C (0x0000 in normal format).

A period of approximately 3ms at the beginning of each channel's temperature conversion cycle is dedicated to diode fault detection. Once a diode fault is detected, the temperature conversion for that channel is abandoned and fault detection/temperature conversion begins on the next channel in the conversion sequence. See *Table 20* for the Diode *Fault Status* register format.

Bits in the *Diode Fault Status* register (0x44) are cleared by a successful read but will be set again after the next conversion if the fault is still in effect.

Table 20. Diode Fault Status (Address = 0x44)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Remote 4–Diode Fault Bit. When this is set to 1, it indicates that there is an open or short condition on Channel 4.
3	Remote 3	0	Remote 3–Diode Fault Bit. When this is set to 1, it indicates that there is an open or short condition on Channel 3.
2	Remote 2	0	Remote 2–Diode Fault Bit. When this is set to 1, it indicates that there is an open or short condition on Channel 2.
1	Remote 1	0	Remote 1–Diode Fault Bit. When this is set to 1, it indicates that there is an open or short condition on Channel 1.
0	Reserved	0	Reserved

Configuration Registers

There are two configuration registers to control the operation of the MAX31732: Configuration 1 and Configuration 2. The bit arrangement for the *Configuration 1* register (0x0F) is shown in *Table 21*. Bit 7 (MSB) is used to put the device either in software-standby mode (stop) or continuous-conversion mode. In standby mode, the ADC shuts down, and the supply current is reduced, but the I^2C/SMBus communication interface remains active. Bit 6 resets all SRAM registers to their POR values and then clears itself. See SRAM Register Map for POR values. Bit 5 enables the SMBus timeout function. Bit 4 selects the $\overline{ALARM1}$ output to function either as an interrupt or as a comparator. Bit 3 and Bit 2 enable the fault queue for the primary threshold faults. These bits set the number of consecutive thermal faults required before asserting $\overline{ALARM1}$ and setting the thermal status bits. Bit 1 selects the extended range temperature data format which allows reading temperature values of 128°C or greater. Bit 0 selects a single conversion on all enabled temperature channels when set to 1. This one-shot function can be enabled only when in stop mode.

Table 21. Configuration1 Register (Address = 0x0F)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	STOP	0	STOP Bit. Setting this bit to 1 allows the ADC to enter standby mode after the current conversion sequence is completed. Setting this to 0 keeps the ADC in continuous conversion mode.
6	SOFT POR	0	Soft Power-On-Reset Bit. Setting this bit to 1 resets the MAX31732 and all the registers will default to their POR values. Wait for POR before accessing the I^2C bus.
5	TIMEOUT	0	SMBus Timeout Enable Bit. Setting this bit to 1 allows the MAX31732 to monitor the SCL line. If the SCL line stops toggling for about 35ms (typ) after the bus START condition, the communication interface will be reset, and the current data will be abandoned.
4	$\overline{ALARM1}$ INT/COMP.	1	$\overline{ALARM1}$ Interrupt/Comparator Mode-Select Bit. Setting this bit to 0 enables the $\overline{ALARM1}$ output to function in interrupt mode. Setting this bit to 1, enables the $\overline{ALARM1}$ output to function in comparator mode.
3	$\overline{ALARM1}$ FAULT QUEUE	0	$\overline{ALARM1}$ Fault Queue Bits. These bits determine the number of consecutive faults required to assert $\overline{ALARM1}$ and set the status bits. 00 = 1; 01 = 2; 10 = 4, 11 = 6.
2		0	
1	EXTENDED RANGE	0	Extended-Range Enable Bit. Set this bit to 1 to set the temperature limit, and reference data range to the maximum reportable temperature of +191.9375°C for the primary temperature limits and +191°C for the secondary temperature limits. Set this bit to 0 to set the data range to a maximum reportable temperature of +127.9375°C for the primary temperature limits and +127°C for the secondary temperature limits. **Note:** This bit sets the extended temperature for both primary and secondary temperature monitoring.
0	ONE SHOT	0	One-Shot Conversion Bit. Setting this bit to 1 while the STOP bit = 1, initiates a single cycle of temperature conversions. All other bits in the configuration register are ignored. This bit reset to 0 after the one-shot conversion is completed. Note: One-Shot conversion mode (Bit 0, Address 0x0F) is not compatible with MTP fault logging and the WR_EN bit (Bit 7, Address 0x39) must be disabled (logic '0') to prevent corrupt data from being recorded.

The bits arrangement for the Configuration 2 register (0x10) is shown in *Table 22*. Bit 7 (MSB) enables packet error code. Bit 4 configures $\overline{\text{ALARM2}}$ output to function either as an interrupt or as a comparator. Bit 3 and Bit 2 enable the fault queue for secondary thermal faults, which sets the number of consecutive thermal faults required before asserting the thermal status bits and the $\overline{\text{ALARM2}}$ output. Bit 0 enables or disables the diode short to ground fault reporting. This is useful for configurations where any DXN# pins are grounded.

Table 22. Configuration 2 Register (Address = 0x10)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	PEC_EN	0	Packet Error Code Enable Bit. Setting this bit to 1 indicates the SMBus transactions will have a PEC byte at the end of message transfer. **Note:** When this is PEC_EN is set to 1, I²C Burst transmissions will not work.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	$\overline{\text{ALARM2}}$ INT/COMP.	1	$\overline{\text{ALARM2}}$ Interrupt/Comparator Mode-Select Bit. Setting this bit to 0 enables the $\overline{\text{ALARM2}}$ output to function in interrupt mode. Setting this bit to 1, enables the $\overline{\text{ALARM2}}$ output to function in comparator mode.
3	$\overline{\text{ALARM2}}$ FAULT QUEUE	0	$\overline{\text{ALARM2}}$ Fault Queue Bits. These bits determine the number of consecutive faults required to assert $\overline{\text{ALARM2}}$ and set the status bits. 00 = 1; 01 = 2; 10 = 4, 11 = 6.
2		0	
1	Reserved	0	Reserved.
0	ISC2GND	1	Ignore Short to Ground Faults. 0 = Diode Short to Ground Fault Enabled; 1 = Diode Short to Ground Fault Ignored (default).

Alarm Outputs

$\overline{\text{ALARM1}}$ and $\overline{\text{ALARM2}}$ are active-low, open-drain fault outputs that assert when the temperature on local/remote sensors falls outside of the limits set in overtemperature/undertemperature limit registers.

$\overline{\text{ALARM1}}$ responds to overtemperature and undertemperature threshold faults set in the Primary Overtemperature/ Undertemperature Limit registers for local and remote sensors. $\overline{\text{ALARM1}}$ asserts when the number of temperature faults for a channel exceeds the number set by FAULT QUEUE bits 2 and 3 in the Configuration 1 register. Upon the last temperature fault in a channel when $\overline{\text{ALARM1}}$ asserts, a corresponding status bit is set in the Status register in SRAM. $\overline{\text{ALARM2}}$ responds to overtemperature and undertemperature threshold faults set in the Secondary Threshold Limits registers for local and remote sensors. $\overline{\text{ALARM2}}$ asserts when the number of temperature faults for a channel exceeds the number set by FAULT QUEUE Bits 2 and 3 in the Configuration 2 register. Upon the last temperature fault in a channel, a corresponding status bit is set in the *Status* register in the SRAM**.** Both alarm outputs operate in comparator mode but can be changed to interrupt mode in the *Configuration 1* and *Configuration 2* registers, independently.

Interrupt Mode

Thermal interrupts occur when the temperature reading of the local or remote channels exceeds the user-programmable temperature threshold limit registers. $\overline{\text{ALARM1}}$ and $\overline{\text{ALARM2}}$ interrupt output signals can be cleared by reading the corresponding status register associated with the fault or by successfully responding to an ARA (Alert Response Address) transmission by the controller. In both cases, the thermal fault is cleared but is reasserted at the end of the next conversion if the fault condition still exists. The interrupt does not halt automatic conversions. The $\overline{\text{ALARM1}}$, and $\overline{\text{ALARM2}}$ outputs are open drain so that multiple devices can share a common interrupt line. All thermal interrupts can be masked using the $\overline{\text{ALARM1}}$, and $\overline{\text{ALARM2}}$ Mask registers. See *Configuration 1* and *Configuration 2* registers on how to set the alarm outputs in interrupt mode.

Comparator Mode

In Comparator mode, $\overline{\text{ALARM1}}$ and $\overline{\text{ALARM2}}$ assert based on a temperature measurement exceeding the temperature threshold in the temperature threshold limit registers. However, unlike the Interrupt mode, the outputs de-assert automatically when the temperature of a channel crosses the threshold back into the acceptable range. A 2°C hysteresis

is applied in comparator mode, so clearing $\overline{ALARM1}$, and $\overline{ALARM2}$ in this mode requires the temperature to be 2°C less than the high thermal limit and 2°C greater than the low thermal limit. See *Configuration 1 and Configuration 2* registers on how to set the alarm outputs in comparator mode.

Note: *In comparator mode, alarm outputs de-assert if a channel is disabled.*

Alarm Output Mask Registers (Address = 0x1B and 0x1C)

Mask registers allow the MAX31732 to mask overtemperature and undertemperature faults for each channel. *$\overline{ALARM1}$ Mask register* (0x1B) masks temperature faults that occur due to the temperature exceeding the threshold set in the *Primary Overtemperature/Undertemperature Threshold* registers. The register bit format and functionality are described in Table 23. *$\overline{ALARM2}$ Mask* register (0x1C) masks temperature faults that occur due to the temperature exceeding threshold limits set in the *Secondary Overtemperature/Undertemperature Threshold* registers. The register bit format and functionality are described in Table 24. The power-up state of these registers is 0x00.

Table 23. $\overline{ALARM1}$ Mask Register (Address = 0x1B)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Channel 4 Remote Mask Bit. '0' = $\overline{ALARM1}$ asserts due to remote Channel 4 temperature fault. '1' = $\overline{ALARM1}$ **does not** assert due to remote Channel 4 temperature fault.
3	Remote 3	0	Channel 3 Remote Mask Bit. '0' = $\overline{ALARM1}$ asserts due to remote Channel 3 temperature fault. '1' = $\overline{ALARM1}$ **does not** assert due to remote Channel 3 temperature fault.
2	Remote 2	0	Channel 2 Remote Mask Bit. '0' = $\overline{ALARM1}$ asserts due to remote Channel 2 temperature fault. '1' = $\overline{ALARM1}$ **does not** assert due to remote Channel 2 temperature fault.
1	Remote 1	0	Channel 1 Remote Mask Bit. '0' = $\overline{ALARM1}$ asserts due to remote Channel 1 temperature fault. '1' = $\overline{ALARM1}$ **does not** assert due to remote Channel 1 temperature fault.
0	Local	0	Local Channel Mask Bit. '0' = $\overline{ALARM1}$ asserts due to remote Channel 4 temperature fault. '1' = $\overline{ALARM1}$ **does not** assert due to a local temperature fault.

Table 24. $\overline{ALARM2}$ Mask Register (Address = 0x1C)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Channel 4 Remote Mask Bit. '0' = $\overline{ALARM2}$ asserts due to remote Channel 4 temperature fault. '1' = $\overline{ALARM2}$ **does not** assert due to remote Channel 4 temperature fault.
3	Remote 3	0	Channel 3 Remote Mask Bit. '0' = $\overline{ALARM2}$ asserts due to remote Channel 3 temperature fault. '1' = $\overline{ALARM2}$ **does not** assert due to remote Channel 3 temperature fault.
2	Remote 2	0	Channel 2 Remote Mask Bit. '0' = $\overline{ALARM2}$ asserts due to remote Channel 2 temperature fault. '1' = $\overline{ALARM2}$ **does not** assert due to remote Channel 2 temperature fault.
1	Remote 1	0	Channel 1 Remote Mask Bit. '0' = $\overline{ALARM2}$ asserts due to remote Channel 1 temperature fault. '1' = $\overline{ALARM2}$ **does not** assert due to remote Channel 1 temperature fault.
0	Local	0	Local Channel Mask Bit. '0' = $\overline{ALARM2}$ asserts due to remote Channel 4 temperature fault. '1' = $\overline{ALARM2}$ **does not** assert due to a local temperature fault.

Effect of Ideality Factor

The accuracy of the remote temperature measurements depends on the ideality factor (n) of the remote "diode" (actually a diode-connected transistor). The default value for the MAX31732 is n = 1.008 for all fou remote channels. If necessary, a different ideality factor value can be written to the Custom Ideality Factor registers (0x11 to 0x14). The Custom Ideality Enable register (See *Table 25*) allows each channel to have the default ideality of 1.008 or a value selected between 0.9844 to 1.311. Note that any change in the ideality selections occurs on subsequent conversions; current temperature register values do not change until a new conversion has been completed.

$$T_{ACTUAL} = \left[(T_{MEASURED} + 273.15) \times \left(\frac{\eta_{CUSTOM}}{1.008}\right)\right] - 273.15 \ \ (in\ °C)$$

Table 25. Custom Ideality Enable Register (Address = 0x15)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Channel 4 Remote-Diode Customer Ideality Enable Bit. Set this bit to Logic 0 to select ideality factor = 1.008 for Channel 4. Set this bit to Logic 1 to select the ideality factor determined by the Customer Ideality Factor register.
3	Remote 3	0	Channel 2 Remote-Diode Customer Ideality Enable Bit. Set this bit to Logic 0 to select ideality factor = 1.008 for Channel 3. Set this bit to Logic 1 to select the ideality factor determined by the Customer Ideality Factor register.
2	Remote 2	0	Channel 2 Remote-Diode Customer Ideality Enable Bit. Set this bit to Logic 0 to select ideality factor = 1.008 for Channel 2. Set this bit to Logic 1 to select the ideality factor determined by the Customer Ideality Factor register.
1	Remote 1	0	Channel 1 Remote-Diode Customer Ideality Enable Bit. Set this bit to Logic 0 to select ideality factor = 1.008 for Channel 1. Set this bit to Logic 1 to select the ideality factor determined by the Customer Ideality Factor register.
0	Reserved	0	Reserved.

Each remote channel can be programmed to any value between 0.9844 to 1.311 using *Custom Ideality Factor* registers (0x11–0x14). The bit arrangement for the Custom Ideality Factor is shown in *Table 26*.

Table 26. Customer Ideality Factor Bits Format (Address = 0x11 to 0x14)

BIT 7 (MSB)	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	1	1	0	0	0

To change the ideality factor for a remote channel, use the following formula:

$$CODE = 1024 - (\frac{1008}{IDF})$$

where IDF is the desired ideality factor within the 0.9844 to 1.311 range, and CODE is its decimal equivalent in binary that must be stored in the remote channel's Custom Ideality Factor register. For example, if IDF = 1.21 the value for CODE using the above formula is approximately 191 (0xBF).

Beta Compensation

Beta compensation corrects for errors caused by low beta-sensing transistors. **Note:** It applies only to PNP transistors with their collectors grounded and their bases and emitters connected to DXN# and DXP#, respectively (see *Figure 5)*. Select the remote channels for which beta compensation is active using the Beta Compensation Enable register (0x19). See *Table 27* for the arrangement of bits and descriptions. Note that any changes to this register do not change the results currently in the temperature registers or temperature conversion in progress; changes affect subsequent conversion results.

Before beginning a temperature measurement with beta compensation enabled, the device first measures the beta of the targeted transistor, and then adjusts the drive current level to produce accurate collector current ratios. The beta value registers (0x47-0x4A) for the four remote channels contain the minimum beta values for the corresponding transistors. See *Table 28* and *Table 29* for more details.

If a targeted transistor has a beta less than 0.09, temperature measurement does not work reliably, and temperature measurement is not initiated for that transistor. The diode fault bit is set for the corresponding channel, and the temperature registers are updated with 0000h. If an attempt at temperature measurement is desired for that remote channel, set the associated Beta Compensation Enable bit to 0. Note that if beta compensation is enabled, the series resistance in the diode path must be minimized, as the series resistance cancellation circuitry will interfere with the beta compensation.

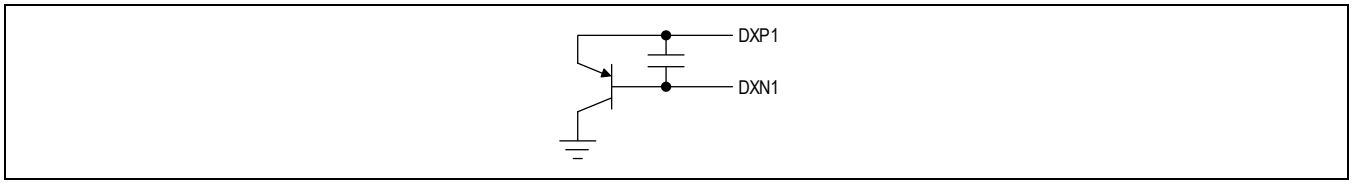


Figure 5. PNP Configuration for use with Beta Compensation

Table 27. Beta Compensation Enable Register (Address = 0x19)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Channel 4 Beta Compensation Enable Bit. Set this bit to Logic 1 to enable beta compensation for remote Channel 4. Set this bit to Logic 0 to disable beta compensation.
3	Remote 3	0	Channel 3 Beta Compensation Enable Bit. Set this bit to Logic 1 to enable beta compensation for remote Channel 3. Set this bit to Logic 0 to disable beta compensation.
2	Remote 2	0	Channel 2 Beta Compensation Enable Bit. Set this bit to Logic 1 to enable beta compensation for remote Channel 2. Set this bit to Logic 0 to disable beta compensation.
1	Remote 1	0	Channel 1 Beta Compensation Enable Bit. Set this bit to Logic 1 to enable beta compensation for remote Channel 1. Set this bit to Logic 0 to disable beta compensation.
0	Reserved	0	Reserved.

Table 28. Beta Compensation and Fault Registers (Read Only) (Address = 0x47 to 0x4A) 0x47 = Remote Channel #1, 0x48 = Remote Channel #2, 0x49 = Remote Channel #3, 0x4A = Remote Channel #4

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	OC	0	DXN# / DXP# Open Circuit Fault. **NOTE:** When an OC fault is detected, a SC2VCC fault can also be triggered.
6	SC2GND	0	DXN# Short Circuit to Ground.
5	SC2VCC	0	DXN#/DXP# Short Circuit to V_{CC}.
4	SC2DXP	0	DXN# Short Circuit to DXP# or DXP# to Ground.
3:0	BETA3:0	0	Beta Compensation Values Applied (See *Table 29* for values).

Table 29. Beta Compensation Values (Read Only) (Address = 0x47 to 0x4A)

VALUE (HEX)	BIT 3	BIT 2	BIT 1	BIT 0	BETA (MIN)
0 (default)	0	0	0	0	0.67
1	0	0	0	1	0.50
2	0	0	1	0	0.36
3	0	0	1	1	0.30
4	0	1	0	0	0.25
5	0	1	0	1	0.20
6	0	1	1	0	0.15
7	0	1	1	1	0.13
8	1	0	0	0	0.11
9	1	0	0	1	0.09
F	1	1	1	1	Low B Fault

Noise Filter

In noisy environments, it can be useful to average the results of multiple temperature conversion results. Use the *Filter Enable* register (0x18) to average the previous four conversions to determine the value stored in the temperature registers. Even when enabled, averaging occurs when performing a one-shot conversion sequence (selected by bit 0 in the *Configuration 1* register), so caution should be exercised when long delays occur between one-shot conversions. Note that filtering begins after enabling the filter; the current register contents do not change. See *Table 30* for bits arrangement and description.

Table 30. Filter Enable Register (0x18)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Channel 4 Filter Enable Bit. Set this bit to logic 1 to enable the filter for remote Channel 4. Set this bit to Logic 0 to disable the filter.
3	Remote 3	0	Channel 3 Filter Enable Bit. Set this bit to logic 1 to enable the filter for remote Channel 3. Set this bit to Logic 0 to disable the filter.

BIT	NAME	POR VALUE	FUNCTION
2	Remote 2	0	Channel 2 Filter Enable Bit. Set this bit to logic 1 to enable the filter for remote Channel 2. Set this bit to logic 0 to disable the filter.
1	Remote 1	0	Channel 1 Filter Enable Bit. Set this bit to logic 1 to enable the filter for remote Channel 1. Set this bit to logic 0 to disable the filter.
0	Reserved	0	Reserved.

Custom Offset Register

If desired, an offset value can be applied to the data in any selected temperature channel. Select the offset value using the *Custom Offset* register (0x16). The resolution of the custom offset value is 0.125°C, and the MSB is 16°C. The temperature offset is calculated using the following equation:

-14.875°C + b[7:0]/8 = Temperature Offset

The resulting offset range is -14.875°C to +17°C. With a default power-on value of 0x77, the device has a default temperature offset of 0°C. See *Table 31* below for bits arrangement and description.

Choose the temperature channels to which custom offset is applied using the *Custom Offset Enable* register (0x17). The offset value does not affect the value in the highest temperature registers. See *Table 32* for bits arrangement and description.

Table 31. Custom Offset Register (Address = 0x16)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	16°C	0	Digital offset (weighted).
6	8°C	1	Digital offset (weighted).
5	4°C	1	Digital offset (weighted).
4	2°C	1	Digital offset (weighted).
3	1°C	0	Digital offset (weighted).
2	0.5°C	1	Digital offset (weighted).
1	0.25°C	1	Digital offset (weighted).
0	0.125°C	1	Digital offset (weighted).

Table 32. Custom Offset Enable Register (Address = 0x17)

BIT	NAME	POR VALUE	FUNCTION
7(MSB)	Reserved	0	Reserved.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	Remote 4	0	Remote 4 Offset Enable Bit. Set to Logic 1 to enable offset in the Customer Offset Register. Set to Logic 0 to disable offset in the Customer Offset Register.
3	Remote 3	0	Remote 3 Offset Enable Bit. Set to Logic 1 to enable offset in the Customer Offset Register. Set to Logic 0 to disable offset in the Customer Offset Register.
2	Remote 2	0	Remote 2 Offset Enable Bit. Set to Logic 1 to enable offset in the Customer Offset Register. Set to Logic 0 to disable offset in the Customer Offset Register.
1	Remote 1	0	Remote 1 Offset Enable Bit. Set to Logic 1 to enable offset in the Customer Offset Register. Set to Logic 0 to disable offset in the Customer Offset Register.
0	Reserved	0	Reserved.

Multi-Time Programmable Nonvolatile Memory (MTP)

The MAX31732 includes a 512-bit nonvolatile MTP memory that helps facilitate system operation and performance. Part of the MTP memory bank is used as device configuration registers (0x8E – 0xB8). These are read/write registers that retain user-defined configuration data and are copied to the corresponding configuration registers in SRAM during power-up or, if desired, during normal operation. See the *Programming MTP Registers* section for more details. Registers 0x80 and 0x81 are available for a user software revision code.

The MTP memory configuration registers are not directly write-accessible by the controller. To update the configuration registers in the MTP memory, the controller must first update the configuration registers in the SRAM and then copy them over to the corresponding configuration registers in the MTP memory using the *MTP_Configuration 2* register (0x3A) in the SRAM. See *Table 33* for the list of common registers between the MTP memory and the SRAM bank.

Table 33. MTP and SRAM Common Configuration Registers

NAMES	MTP ADD (HEX)	SRAM ADD (HEX)	READ/ WRITE	FUNCTION
User Software Revision Registers	0x80–0x81	—	R/W	Stores User-defined Software Revision.
Fault Status	0x82–0x8D	0x02–0x0D	R	See *Table 38* for more details.
Temperature Channel Enable	0x8E	0x0E	R/W	Stores Temperature Channel Enable bits and MTP Loading Enable Bit.
Configuration 1	0x8F	0x0F	R/W	Configures the ADCs conversion mode, $\overline{\text{ALARM1}}$ mode of operation, thermal fault queue, and temperature data format (extended or standard) and enables single-cycle conversion. **Note:** The POR bit is not available in the Configuration 1 register in the MTP memory.
Configuration 2	0x90	0x10	R/W	Configures $\overline{\text{ALARM2}}$ mode of operation, sets thermal fault queue, enables Packet Error Checking, and sets Ignore Short to Ground Faults.
Remote 1 Channel Custom Ideality	0x91	0x11	R/W	Stores Custom Ideality Factor for Remote 1 Sensing Diode in the range of 0.9844 to 1.311.
Remote 2 Channel Custom Ideality	0x92	0x12	R/W	Stores Custom Ideality Factor for Remote 2 Sensing Diode in the range of 0.9844 to 1.311.
Remote 3 Channel Custom Ideality	0x93	0x13	R/W	Stores Custom Ideality Factor for Remote 3 Sensing Diode in the range of 0.9844 to 1.311.
Remote 4 Channel Custom Ideality	0x94	0x14	R/W	Stores Custom Ideality Factor for Remote 4 Sensing Diode in the range of 0.9844 to 1.311.
Custom Ideality Factor Enable	0x95	0x15	R/W	Enables/disables each Remote Sense Channel to be programmed for Custom Ideality Factor.
Custom Offset	0x96	0x16	R/W	Stores Custom Offset Temperature value to be applied to selected remote sense channels.
Custom Offset Enable	0x97	0x17	R/W	Enables/disables each Remote Sense Channel to be programmed for custom offset temperature.
Filter Enable	0x98	0x18	R/W	Enables/disables filter for each Remote Sense Channel (should be disabled when not in constant conversion mode).

NAMES	MTP ADD (HEX)	SRAM ADD (HEX)	READ/ WRITE	FUNCTION
Beta Compensation Enable	0x99	0x19	R/W	Enables/disables Beta Compensation for each remote channel.
Highest Temperature Enable	0x9A	0x1A	R/W	Selects/deselects which channels are used in determining the contents of the Highest Temperature Registers.
$\overline{\text{ALARM1}}$ Mask	0x9B	0x1B	R/W	Masks faults from asserting $\overline{\text{ALARM1}}$ for each channel.
$\overline{\text{ALARM2}}$ Mask	0x9C	0x1C	R/W	Masks faults from asserting $\overline{\text{ALARM2}}$ for each channel.
Remote 1 Primary Overtemperature Limit (MSB)	0x9D	0x1D	R/W	Stores Remote 1 channel's primary overtemperature limit MSB.
Remote 1 Primary Overtemperature Limit (LSB)	0x9E	0x1E	R/W	Stores Remote 1 channel's primary overtemperature limit LSB.
Remote 2 Primary Overtemperature Limit (MSB)	0x9F	0x1F	R/W	Stores Remote 2 channel's primary overtemperature limit MSB.
Remote 2 Primary Overtemperature Limit (LSB)	0xA0	0x20	R/W	Stores Remote 2 channel's primary overtemperature limit LSB.
Remote 3 Primary Overtemperature Limit (MSB)	0xA1	0x21	R/W	Stores Remote 3 channel's primary overtemperature limit MSB.
Remote 3 Primary Overtemperature Limit (LSB)	0XA2	0x22	R/W	Stores Remote 3 channel's primary overtemperature limit LSB.
Remote 4 Primary Overtemperature Limit (MSB)	0xA3	0x23	R/W	Stores Remote 4 channel's primary overtemperature limit MSB.
Remote 4 Primary Overtemperature Limit (LSB)	0xA4	0x24	R/W	Stores Remote 4 channel's primary overtemperature limit LSB.
Local Primary Overtemperature Limit (MSB)	0xA5	0x25	R/W	Stores local channel's primary overtemperature limit MSB.
Local Primary Overtemperature Limit (LSB)	0xA6	0x26	R/W	Stores local channel's primary overtemperature limit LSB.
All Channels Primary Undertemperature Limit (MSB)	0xA7	0x27	R/W	Stores primary undertemperature limit for all channel MSB

NAMES	MTP ADD (HEX)	SRAM ADD (HEX)	READ/ WRITE	FUNCTION
All Channels Primary Undertemperature Limit (LSB)	0xA8	0x28	R/W	Stores primary undertemperature limit for all channel LSB
Remote 1 Secondary Overtemperature Limit	0xA9	0x29	R/W	Stores Remote 1 channel's secondary overtemperature limit.
Remote 2 Secondary Overtemperature Limit	0xAA	0x2A	R/W	Stores Remote 2 channel's secondary overtemperature limit.
Remote 3 Secondary Overtemperature Limit	0xAB	0x2B	R/W	Stores Remote 3 channel's secondary overtemperature limit.
Remote 4 Secondary Overtemperature Limit	0xAC	0x2C	R/W	Stores Remote 4 channel's secondary overtemperature limit.
Local Secondary Overtemperature Limit	0xAD	0x2D	R/W	Stores local channel's secondary overtemperature limit.
All Channels Secondary Undertemperature Limit	0xAE	0x2E	R/W	Stores secondary undertemperature limit for all channels.
Remote 1 Reference Temperature (MSB)	0xAF	0x2F	R/W	Stores reference temperature for Remote 1 channel MSB.
Remote 1 Reference Temperature (LSB)	0xB0	0x30	R/W	Stores reference temperature for Remote 1 channel LSB.
Remote 2 Reference Temperature (MSB)	0xB1	0x31	R/W	Stores reference temperature for Remote 2 channel MSB.
Remote 2 Reference Temperature (LSB)	0xB2	0x32	R/W	Stores reference temperature for Remote 2 channel LSB.
Remote 3 Reference Temperature (MSB)	0xB3	0x33	R/W	Stores reference temperature for Remote 3 channel MSB.
Remote 3 Reference Temperature (LSB)	0xB4	0x34	R/W	Stores reference temperature for Remote 3 channel LSB.
Remote 4 Reference Temperature (MSB)	0xB5	0x35	R/W	Stores reference temperature for Remote 4 channel MSB.
Remote 4 Reference Temperature (LSB)	0xB6	0x36	R/W	Stores reference temperature for Remote 4 channel LSB.
Local Reference Temperature (MSB)	0xB7	0x37	R/W	Stores reference temperature for local channel MSB.
Local Reference Temperature (LSB)	0xB8	0x38	R/W	Stores reference temperature for local channel LSB.
MTP_Configuration	0xB9	0x39	R/W	Selects/deselects fault logging into the MTP memory for each channel.

The MTP memory also has a dedicated number of registers (0x82-0x8D) for fault logging purposes. These are read-only registers that store the thermal fault temperature and the status bits when the temperature of a channel exceeds the primary overtemperature and undertemperature limits. The fault logging functionality is enabled using the *MTP_Configuration* register (0x39) in SRAM. See *Table 34* below for MTP_Configuration's bit arrangement and description. See the *MTP Fault Logging Registers* section for more details.

Table 34. MTP_Configuration Register (Address = 0x39)

BIT	NAME	POR VALUE	FUNCTION
7(MSB)	WR_EN	0	Fault Write Enable Bit. Setting this Bit to 1, enables the transfer of thermal fault temperature to the selected channel. Setting this Bit to 0, disables the transfer of thermal fault temperature to the selected channels. **Note:** One-Shot conversion mode (Bit 0, Address 0x0F) is not compatible with MTP fault logging and the WR_EN bit must be disabled (Logic 0) to prevent corrupt data from being recorded.
6	Reserved	0	Reserved.
5	Reserved	0	Reserved.
4	WR_R4	1	Remote 4 Fault Write Enable Bit. Setting this Bit to 1, enables the transfer of thermal fault temperature on Remote 4 from SRAM to the corresponding registers in the MTP fault-logging registers.
3	WR_R3	1	Remote 3 Fault Write Enable Bit. Setting this Bit to 1, enables the transfer of thermal fault temperature on Remote 3 from SRAM to the corresponding registers in the MTP fault-logging registers.
2	WR_R2	1	Remote 2 Fault Write Enable Bit. Setting this Bit to 1, enables the transfer of thermal fault temperature on Remote 2 from SRAM to the corresponding registers in the MTP fault-logging registers.
1	WR_R1	1	Remote 1 Fault Write Enable Bit. Setting this Bit to 1, enables the transfer of thermal fault temperature on Remote 1 from SRAM to the corresponding registers in the MTP fault-logging registers.
0	WR_L	1	Local Fault Write Enable Bit. Setting this Bit to 1, enables the transfer of thermal fault temperature on local channel from SRAM to the corresponding registers in the MTP fault-logging registers.

Note: *Only 0x1F (No MTP Writes) and 0x9F (MTP Writes Enabled) are allowed in this register. Wait 0.7 seconds before writing anything additional to the MAX31732.*

Accessing MTP Registers

Any update to the MTP configuration registers must be done using a set of registers in SRAM. *Table 35* lists these registers and their functionalities. These registers are not present in the MTP memory.

Table 35. MTP Configuration Registers in SRAM (Address = 0x3A to 0x3D)

REGISTER NAME	REGISTER ADD (HEX)	POR VALUE	FUNCTION
MTP_Configuration 2	0x3A	0x00	Bits in the MTP_Configuration 2 register enable the MAX31732 to write a single-word or multiple-byte to the MTP memory, load MTP configuration registers into the configuration register in SRAM during normal operation, or do a single-register read operation via I^2C/SMBus. See *Table 36* for more details.
MTP_Address	0x3B	0x00	Bits in the MTP_Address register specify the address location of the target register in MTP to be updated.

REGISTER NAME	REGISTER ADD (HEX)	POR VALUE	FUNCTION
MTP_DIN (MSB)	0x3C	0x00	Bits in the MTP_DIN (MSB) specifies the MSB data to be written to the targeted register in MTP specified by the MTP_Address register. See *Table 37* for bit format.
MTP_DIN (LSB)	0x3D	0x00	Bits in the MTP_DIN (LSB) specifies the LSB data to be written to the targeted register in MTP specified by the MTP_Address register. See *Table 37* for bit format.

Table 36. MTP_Configuration2 Register (Address = 0x3A)

BIT	NAME	POR VALUE	FUNCTION
7 (MSB)	MTP_Config_Store	0	MTP Configuration Store Bit. Setting this Bit to 1, allows the MAX31732 to load all the configuration registers in the SRAM into the corresponding configuration registers in the MTP. This Bit resets to 0 after the loading is complete. See the *Electrical Characteristics* table for time duration.
6	MTP_Config_Word_Store	0	MTP Configuration Word Store Bit. Setting this Bit to 1, allows the MAX31732 to load a single word in the SRAM into the corresponding registers in the MTP. This Bit resets to 0 after the loading of the single word is complete. See the *Electrical Characteristics* table for time duration.
5	MTP_Config_Load	0	MTP Configuration Load Bit. Setting this Bit to 1 allows the MAX31732 to load all the configuration registers in the MTP into the configuration registers in the SRAM. This Bit resets to 0 after the loading of the single word is complete. See the *Electrical Characteristics* table for time duration.
4	MTP_Manual_Write	0	MTP Manual Write Bit. Setting this Bit to 1 allows the MAX31732 to load the customer software version code into registers 0x80 and 0x81 loading in the MTP. This Bit resets to 0 after the loading of the single word is complete.
3	MTP_I2C_Read	0	MTP I^2C Read Bit. Setting this Bit to 1 allows the controller to directly read the MTP registers one at a time. If this Bit to 0, the I^2C read operation will read all MTP registers as 0x00.
2	Reserved	0	Reserved.
1	Reserved	0	Reserved.
0	Reserved	0	Reserved.

Table 37. MTP_DIN Data Format (Address = 0x3C and 0x3D)

MTP_DIN (MSB)								MTP_DIN (LSB)							
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Applications Information

Reading MTP Registers

Burst I2C read is not available when reading MTP registers (0x80-0xB9), meaning MTP registers must be read one at a time. Program *MTP Configuration 2* (3Ah) bit 3, MTP_I2C_READ_EN = 1 to enable read back.

Programming MTP Registers

Use MTP_Configuration2 (0x3A) in the SRAM to either write a single word at a time or all MTP configuration registers in one shot during normal operation when the enable input (EN) is high or low. See *MTP Multi-Byte Configuration* and *MTP Single Word Configuration* sections for more details.

MTP Multi-Byte Configuration Store

To change the contents of the entire configuration registers in MTP memory use the following steps.

1. Write all desired configuration data into the RAM registers (0x0E to address 0x39).
2. Write “0x80” into the register called *MTP_Configuration 2* (0x3A).
3. Wait for 650ms (typ.) and then read *MTP_Configuration 2.* The register should read “0x00” instead of “0x80” which indicates that all the configuration registers in the MTP have been updated.

MTP Single Word Configuration Store

To change the contents of a single word in the MTP memory, use the following steps:

1. Choose one of the configuration registers in MTP that you want to modify. For example, 0x9D is the address of *Remote 1 Primary Overtemperature Limit (MSB)*. The corresponding address in SRAM is 0x1D.
2. Update the threshold limit in address 0x1D.
3. Write the address “0x9D” into a register called *MTP_ADDRESS* (0x3B).
4. Write “0x40” into *MTP_Configuration2* (0x3A).
5. Wait for 25ms and then read *MTP_Configuration 2.* The register should read “0x00” instead of “0x40” which indicates that the targeted registers in the MTP have been updated.

This writes two bytes at a time starting from an odd address. Writing an even address into 0x3B will result in writing the contents of that address and the contents of the previous MTP address.

Programming User Software Revision Registers

User Software Revision registers (0x80/0x81) are in the MTP memory only. These registers are used to store the customer’s software revision code. Use the following steps to update these registers.

- Write the software revision data in *MTP_DIN_MSB (0x3C)* and *MTP_DIN_LSB(0x3D).*
- Write the address of the targeted MTP registers (0x80/0x81) into the register called *MTP_ADDRESS* in the SRAM bank.
- Write “10” into *MTP_Configuration2* (0x3A) to initiate the update process.
- Wait for 25ms and then check the content of the register (0x80/0x81).

Power-Up Process

During power-up, when V_{CC} crosses the POR value of 2.65V (typ), all MTP configuration registers (0x8E–0xB9) are loaded into the SRAM configuration registers (0x0E–0xB9). To enable the loading process during power-up, Bit 7 (EN_MTP_PU_LOAD) of the *Temperature Channel Enable* register (0x8E) in the MTP memory must be set to 1. If Bit 7 is set to 0, the MTP configuration registers are not copied to the SRAM during power-up process. I^2C communication is paused for approximately 2ms during MTP load. The MAX31732 will issue a NACK if communication is attempted during this time.

MTP Fault Logging Registers

The MAX31732 logs the thermal data for under-temperature and over-temperature faults along with the primary thermal status in the MTP fault logging registers, (0x82–0x8D). When the temperature of any channel exceeds the set undertemperature or overtemperature threshold in the primary threshold limit registers, and $\overline{\text{ALARM1}}$ asserts low, the fault temperature and the status bit for that channel are copied to the corresponding fault logging registers in the MTP memory. In addition, the MAX31732 copies the thermal data of the remaining four channels to their appropriate locations in MTP memory. The thermal data in the fault logging registers are locked unless there is another thermal fault on any one of the four remaining channels. See *Table 38* for fault logging register address locations (SRAM and MTP). When fault logging occurs, temperature conversion will pause, and the I^2C will be unavailable to prevent any updates to the thermal limit registers. The MAX31732 will issue a NACK if communication is attempted for approximately 20ms.

Table 38. MTP and SRAM Common Temperature Data Registers

NAMES	MTP ADD (HEX)	SRAM ADD (HEX)	READ/ WRITE	FUNCTION
Remote 1 Primary Temperature Data (MSB)	0x82	0x02	R	Stores the upper byte of the thermal fault temperature of Remote 1 channel.
Remote 1 Primary Temperature Data (LSB)	0x83	0x03	R	Stores the lower byte of the thermal fault temperature of Remote 1 channel.
Remote 2 Primary Temperature Data (MSB)	0x84	0x04	R	Stores the upper byte of the thermal fault temperature of Remote 2 channel.
Remote 2 Primary Temperature Data (LSB)	0x85	0x05	R	Stores the lower byte of the thermal fault temperature of Remote 2 channel.
Remote 3 Primary Temperature Data (MSB)	0x86	0x06	R	Stores the upper byte of the thermal fault temperature of Remote 3 channel.
Remote 3 Primary Temperature Data (LSB)	0x87	0x07	R	Stores the lower byte of the thermal fault temperature of Remote 3 channel.
Remote 4 Primary Temperature Data (MSB)	0x88	0x08	R	Stores the upper byte of the thermal fault temperature of Remote 4 channel.
Remote 4 Primary Temperature Data (LSB)	0x89	0x09	R	Stores the lower byte of the thermal fault temperature of Remote 4 channel.
Local Primary Temperature Data (MSB)	0x8A	0x0A	R	Stores the upper byte of the thermal fault temperature of the local channel.
Local Primary Temperature Data (LSB)	0x8B	0x0B	R	Stores the lower byte of the thermal fault temperature of the local channel.
Primary Thermal Status High	0x8C	0x0C	R	Stores status bits when there is an overtemperature fault on any channel.
Primary Thermal Status Low	0x8D	0x0D	R	Stores status bits when there is an undertemperature fault on any channel.

On-Demand Loading

The on-demand loading command allows updating configuration registers in the SRAM during normal operation after the power-up process. This feature is useful to fine-tune the register settings for optimal performance or in the event the data in the registers are lost or corrupted. To initiate on-demand loading, set Bit 5 (MTP_Config_Load) of the MTP_Configuration 2 register to 1. The MAX31732 will issue a NACK if communication is attempted for approximately 2ms.

MTP Temperature and Status recording

While in continuous mode, the user can record all the temperature and status registers into the MTP in the case of an overtemperature or under-temperature. In the case of an undertemperature or over-temperature on any of the channels, the temperature of all active channels, and the status register is recorded into the MTP.

To enable this recording, use the following procedure:

1) Write 0xFF into register address 0x39.
2) Wait 0.7s

Once the MTP temperature recording is enabled, MTP writes are controlled, as shown in *Figure 6*.

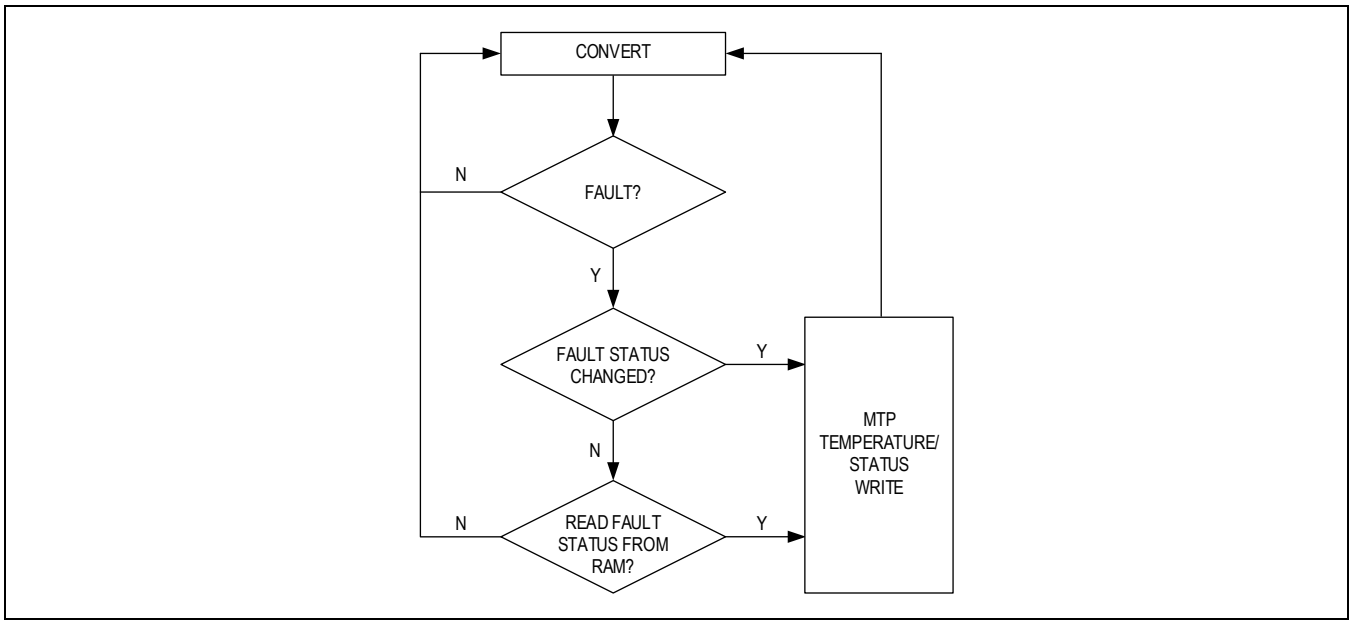


Figure 6. Flowchart showing MTP temperature and status write conditions.

After the MTP temperature recording is enabled, the MTP records all active channel temperatures and the fault status byte. After this, the MTP holds these values until another fault occurs and either:

a) The Fault status changed,
b) The status byte (0x0C or 0x0D) is read from the RAM.

Given the above flow and the limited number of MTP writes, the user may wish to limit fault status reads from the RAM when the MTP temperature and recording are enabled. The fault status bytes can be inferred without reading the RAM with the following procedure:

1) If No fault has occurred, read the status data from the MTP. The MTP will capture and hold the first fault condition.
2) If the MTP has recorded a fault, the user can infer the current fault status with $\overline{\text{ALARM1}}$ and the MTP fault status. If $\overline{\text{ALARM1}}$ is high, no fault is occurring. If $\overline{\text{ALARM1}}$ is low, the MTP fault status reflects the RAM fault status and, therefore, can be read directly.

Additionally, after a fault condition, the user can disable MTP reads and then monitor the temperature and status registers directly.

Finally, read MTP data before RAM status registers are read, to avoid conflicts with the MTP after the MTP temperature recording is re-enabled.

Avoiding MTP Conflicts

If the user has activated MTP fault logging and a fault occurs, the part will NACK user communications to avoid conflicts while the part records the fault to the MTP. To avoid this issue, before any MTP writes, the user should write 0x00 to 0x39 to stop conversions and wait for 0.7s for all conversions and MTP fault logging to finish.

Clearing MTP fault login registers

The MTP fault temperature and status registers (addresses 0x82 to 0x8D) can be cleared, 2 bytes at a time.

To clear MTP temperature or status registers, use the following procedure:

1) Write 0x00 to 0x39 to disable MTP recording.
2) Wait for 0.7s to finish any active conversion and avoid conflicts with MTP writes.
3) Write 0 into MTP_DIN_MSB(0x3C)
4) Write 0 into MTP_DIN_LSB (0x3D)

Write the MTP address for the temperature and status register using *Table 38*. To reset a temperature, use the MSB address. For example, Remote 1 Primary Temperature's MSB address is 0x82, and Remote 2 Primary Temperature's MSB address is 0x84. To reset the status, use address 0x8C. This will reset both the primary thermal status high and the primary thermal status low.

Write 0x10 into MTP_Configuration 2 (0x3a) to initiate the update process.

Wait for 25ms and then check the content of the registers. For example, after resetting Remote 2 Primary Temperature, check that MTP_Configuration2 (0x3A) reads 0 and the MTP data in addresses 0x3C and 0x3D equal zero.

Setting Overtemperature Limits

After setting overtemperature and undertemperature limits, wait for 0.7s before further reads or writes.

Remote-Diode Selection

The device directly measures the die temperature of CPUs and other ICs that have on-chip temperature-sensing diodes (see the *Typical Application Circuit*), or it can measure the temperature of a discrete diode-connected transistor.

Discrete Remote Diodes

When the remote-sensing diode is a discrete transistor, its collector and base must be connected together; PNP or NPN discrete transistors can be used. *Table 39* lists examples of discrete transistors that are appropriate for use with this device. The transistor must be a small-signal type with a relatively high forward voltage; otherwise, the A/D input voltage range can be violated. The forward voltage at the highest expected temperature must be greater than 0.25V at 10µA; at the lowest expected temperature, the forward voltage must be less than 0.95V at 100µA. Large power transistors must not be used. Also, ensure that the base resistance is less than 100Ω. Tight specifications for forward-current gain (e.g., $50 < \beta < 150$) indicate that the manufacturer has good process controls and that the devices have consistent V_{BE} characteristics. Manufacturers of discrete transistors do not normally specify or guarantee ideality factor. This normally is not a problem since good-quality discrete transistors tend to have ideality factors that fall within a relatively narrow range. Variations in remote temperature readings of less than ±2°C with a variety of discrete transistors have been observed. However, it is good design practice to verify good consistency of temperature readings with several discrete transistors from any supplier under consideration.

Unused Diode Channels

If one or more of the remote-diode channels is not needed, disconnect the DXP# and DXN# inputs for that channel, or connect the DXP# to the corresponding DXN#. The status register indicates a diode "fault" for this channel and the channel is ignored during the temperature-measurement sequence. It is also good practice to mask any unused channels immediately upon power-up by setting the appropriate bits in the $\overline{\text{ALARM1}}$ and $\overline{\text{ALARM2}}$ Mask register. This prevents unused channels from causing $\overline{\text{ALARM1}}$ and $\overline{\text{ALARM2}}$ to assert.

Table 39. Remote Sensors Transistors Supplies

SUPPLIER	PNP MODEL NUMBER
Central Semiconductor Corp. (USA)	CMPT3906
Fairchild Semiconductor (USA)	MMBT3906
Infineon (Germany)	SMBT3906
ON Semiconductor (USA)	MMBT3906
ROHM Semiconductor (USA)	SST3906
Samsung (Korea)	KST3906-TF
Siemens (Germany)	SMBT3906
Zetex (England)	FMMT3906CT-ND

Thermal Mass

When sensing local temperature, the device measures the temperature of the Printed Circuit Board (PCB) to which it is soldered. The leads provide a good thermal path between the PCB traces and the die. As with all IC temperature sensors, thermal conductivity between the die and the ambient air is poor by comparison, making air-temperature measurements impractical. Since the thermal mass of the PCB is far greater than that of the device, the device follows temperature changes on the PCB with little or no perceivable delay. When measuring the temperature of a CPU, or other IC with an on-chip sense junction, thermal mass has virtually no effect; the measured temperature of the junction tracks the actual temperature within a conversion cycle. When measuring temperature with discrete remote transistors, the best thermal response times are obtained with transistors in small packages (i.e., SOT23 or SC70). Take care to account for thermal gradients between the heat source and the sensor, and ensure that stray air currents across the sensor package do not interfere with measurement accuracy.

ADC Noise Filtering

The integrating ADC has good noise rejection for low-frequency signals, such as power-supply hum. In environments with significant high-frequency Electromagnetic interference (EMI), connect an external 100pF capacitor between DXP# and DXN#. Larger capacitor values can be used for added filtering; however, this can introduce errors due to the rise time of the switched current source. Noise can be minimized with careful PCB layout, as discussed in the *PCB Layout* section.

Target Address

Target addresses can be selected by connecting a resistor between ADD and GND, as shown in *Table 40*.

Table 40. Target Address Selection

RESISTOR BETWEEN ADD AND GND	7-BIT SLAVE ADDRESS (HEX)
15kΩ TO 39kΩ	0x4F
9.31kΩ	0x4E
6.81kΩ	0x4D
4.75kΩ	0x4C
3.01kΩ	0x1F
1.69kΩ	0x1E
750Ω	0x1D
0 (< 250Ω)	0x1C

Note: *Resistor value tolerance must be ±5% of the listed values.*

PCB Layout

Follow the guidelines to reduce the measurement error when measuring remote temperature:

- Place the device as close as possible to the thermal diode. In noisy environments, such as a computer motherboard, this distance is typically 10cm to 20cm. This length can be increased if the worst noise sources are avoided. Noise sources include displays, clock generators, memory buses, and PCI buses.
- Do not route the DXP# and DXN# traces across fast digital signals, which can easily introduce a +30°C error, even with good filtering) Route the DXP# and DXN# traces in parallel and in close proximity to each other. Each parallel pair of traces should go to a thermal diode.
- Route DXP# and DXN# traces away from any higher voltage traces, such as +12VDC. Leakage currents from PCB contamination must be dealt with carefully since a 20MΩ leakage path from DXP_ to ground causes approximately +1°C error. If high-voltage traces are unavoidable, connect guard traces to GND on either side of the DXP# - DXN# traces.
- Route through as few vias and crossunders as possible to minimize copper/solder thermocouple effects. Use wide traces when possible (5mil to 10mil traces are typical).
- Place 0.1μF bypass capacitor between V_{CC} and GND as close as possible to the MAX31732.
- Place 100pF capacitor between DXN# and DXP# or DXP# and GND if DXN# is connected to the ground close to the MAX31732.
- For single-ended configurations, ground DXN pins close to the MAX31732, and if possible, route DXP# traces as guarded strip-line between two ground planes.

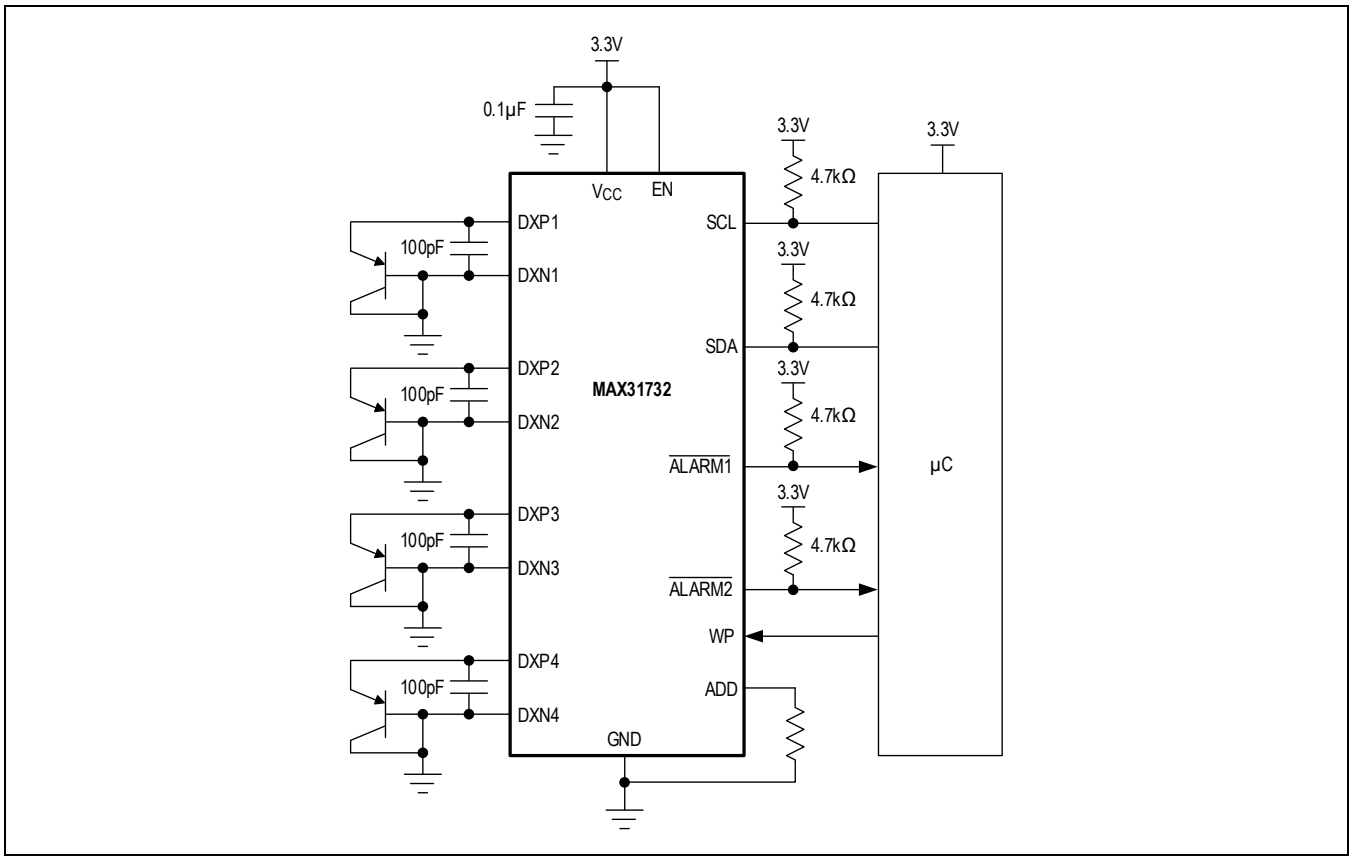
Twisted-Pair and Shielded Cables

Use a twisted-pair cable to connect the remote sensor for remote-sensor distances longer than 20cm or in very noisy environments. Twisted-pair cable lengths can be between 2m and 4m before noise introduces excessive errors. For longer distances, the best solution is a shielded twisted pair, such as those used for audio microphones.

For example, Belden No. 8451 works well for distances up to 100ft in a noisy environment. At the device, connect the twisted-pair cables to DXP# and DXN# and the shielded cable to GND. Leave the shielded cable unconnected at the remote sensor. For very long cable runs, the cable's parasitic capacitance often provides noise filtering; therefore, the 100pF capacitor can often be removed, or reduced in value.

For single-ended configurations, ground both the shield and the negative lead of the cable at the PCB and leave the shield unconnected at the remote sensor end of the cable.

Typical Application Circuit



Ordering Information

PART NUMBER	TEMPERATURE RANGE	PIN-PACKAGE
MAX31732ATG+	-40°C TO +125°C	24 TQFN-EP
MAX31732ATG+T	-40°C TO +125°C	24 TQFN-EP

+Denotes a lead(Pb)-free/RoHS-compliant package.

T = Tape and reel.

**EP = Exposed pad.*

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	6/24	Initial release	—