Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!

Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832
Email & Skype: info@chipsmall.com Web: www.chipsmall.com
Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China

ADM1033

Thermal Monitor and Fan Speed (RPM) Controller

The ADM1033 is a one channel remote- and local-temperature sensor and fan controller. The remote channel monitors the temperature of the remote thermal diode, which may be discrete 2N3904/6s or may be located on a microprocessor die. The device also monitors its own ambient temperature.

The ADM1033 can monitor and control the speed of cooling fan. The user can program a target fan speed, or else use the look-up table to input a temperature-to-fan speed profile. The look-up table can be configured to run the fans at discrete speeds (discrete mode) or to ramp the fan speed with temperature (linear mode).

The ADM1033 communicates over a 2-wire SMBus 2.0 interface. An 8-level LOCATION input allows the user to choose between SMBus 1.1 and SMBus 2.0. An $\overline{\text{ALERT}}$ output indicates error conditions. The $\overline{\text{THERM}}$ I/O signals overtemperature as an output and times $\overline{\text{THERM}}$ assertions as an input. Pin 8 can be configured as a reference for the $\overline{\text{THERM}}$ ($\overline{\text{PROCHOT}}$) input.

Features

- 1 Local and Remote Temperature Channels
- ±1°C Accuracy on Local and Remote Channels
- Automatic Remote Temperature Channel, Up to 1 kΩ
- Fast (Up to 64 Measurements per Second)
- SMBus 2.0, 1.1, and 1.0 Compliant
- SMBus Address Input/LOCATION Input to UDID
- Programmable Over/Undertemperature Limits
- Programmable Fault Queue
- $\overline{\text{SMBusALERT}}$ Output
- Fail-Safe Overtemperature Comparator Output
- Fan Speed (RPM) Controller
- Look-up Table for Temperature-to-Fan Speed Control
- Linear and Discrete Options for Look-up Table
- $\overline{\text{FAN\_FAULT}}$ Output
- $\overline{\text{THERM}}$ Input, Used to Time $\overline{\text{PROCHOT}}$ Assertions
- REF Input, Used as Reference for $\overline{\text{THERM}}$ ($\overline{\text{PROCHOT}}$)
- 3.0 V to 5.5 V Supply
- Small 16-lead QSOP Package
- This is a Pb-Free Device*

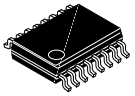
Applications

- Desktop and Notebook PCs
- Embedded Systems
- Telecommunications Equipment
- LCD Projectors

* For additional information on our Pb-Free strategy and soldering details, please download the ON Semiconductor Soldering and Mounting Techniques Reference Manual, SOLDERRM/D.

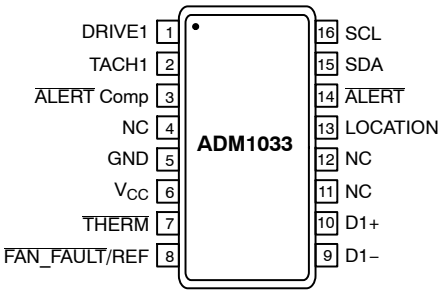
ON Semiconductor®

http://onsemi.com



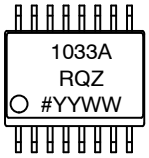
QSOP−16
CASE 492

PIN ASSIGNMENT



(Top View)

MARKING DIAGRAM



1033ARQZ	= Specific Device Code
#	= Pb-Free Package
YY	= Date Code
WW	= Work Week

ORDERING INFORMATION

See detailed ordering and shipping information in the package dimensions section on page 34 of this data sheet.

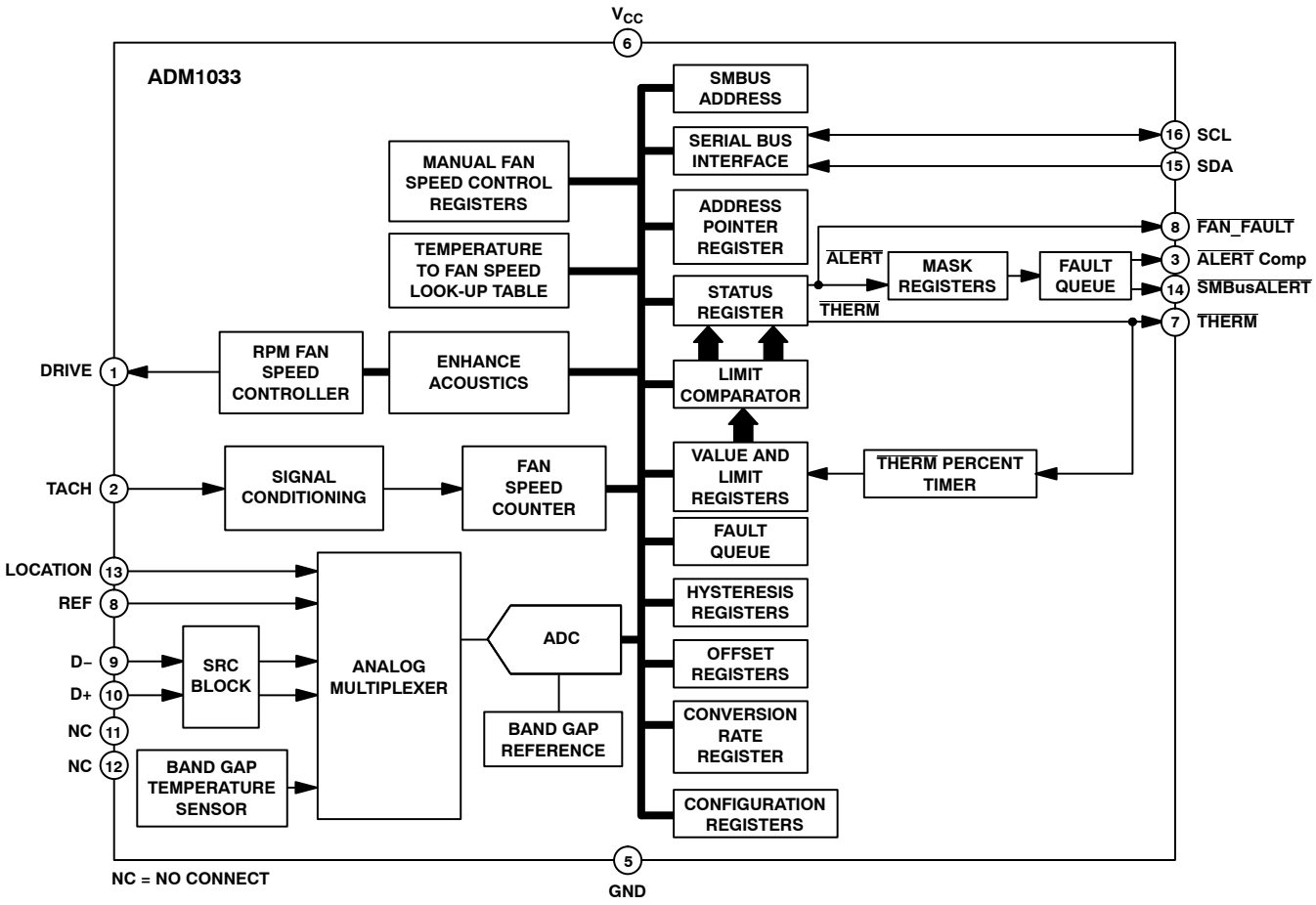


Figure 1. Functional Block Diagram

Table 1. ABSOLUTE MAXIMUM RATINGS

Parameter	Rating	Unit
Positive Supply Voltage (V_{CC})	−0.3, +6.5	V
Voltage on Any Input or Output Pin except $\overline{\text{FAN\_FAULT}}$ and LOCATION	−0.3 to V_{DD} + 6.5	V
Voltage on $\overline{\text{FAN\_FAULT}}$ (Note 1)	V_{CC}	
Voltage on LOCATION	V_{CC} + 0.3	V
Input Current at Any Pin	±20	mA
Maximum Junction Temperature ($T_{J\ MAX}$)	150	°C
Storage Temperature Range	−65 to +150	°C
Lead Temperature, Soldering (10 s)	300	°C
IR Reflow Peak Temperature	220	°C
ESD Rating − All Pins	1500	V

Stresses exceeding Maximum Ratings may damage the device. Maximum Ratings are stress ratings only. Functional operation above the Recommended Operating Conditions is not implied. Extended exposure to stresses above the Recommended Operating Conditions may affect device reliability.

1. During powerup the voltage on $\overline{\text{FAN\_FAULT}}$ should not be higher than V_{CC}.

NOTE: This device is ESD sensitive. Use standard ESD precautions when handling.

Table 2. THERMAL CHARACTERISTICS

Package Type	θ_{JA}	θ_{JC}	Unit
16-lead QSOP Package	150	39	°C/W

Table 3. PIN ASSIGNMENT

Pin No.	Mnemonic	Description
1	DRIVE1	DRIVE1 Pin Drives Fan 1. Open-drain output. Requires a pullup resistor.
2	TACH1	Fan 1 Fan Speed Measurement Input. Connects to the fan's TACH output to measure the fan speed.
3	$\overline{\text{ALERT}}$ Comp	Open-Drain Active Low Output. Assets low whenever a measurement goes outside its programmed limits if not masked. Automatically goes high again when the measured parameter falls back within its limits.
4	NC	No Connect.
5	GND	Ground for Analog and Digital Circuitry.
6	VCC	Power. Can be powered by 3.3 V standby power if monitoring in low power states is required.
7	$\overline{\text{THERM}}$	Can be configured as an overtemperature interrupt output, or as an input (to monitor $\overline{\text{PROCHOT}}$ output of an INTEL CPU). A timer measures assertion times on the $\overline{\text{THERM}}$ pin (either input or output).
8	$\overline{\text{FAN\_FAULT}}$/REF	$\overline{\text{FAN\_FAULT}}$: Open-Drain Output. Asserted low when one or both fans stall. Requires a pullup resistor to V_{CC}. REF: Analog Input Reference for the $\overline{\text{THERM}}$ Input.
9	D1−	Cathode Connection for the First Thermal Diode or Diode-Connected Transistor.
10	D1+	Anode Connection for the First Thermal Diode or Diode-Connected Transistor.
11	NC	No Connect.
12	NC	No Connect.
13	LOCATION	8-level Analog Input. Used to determine the correct SMBus version and the SMBus address (in fixed and discoverable mode) and to set the LLL bits in the UDID (in ARP-capable mode).
14	$\overline{\text{ALERT}}$	Open-Drain Output. $\overline{\text{SMBusALERT}}$ pin. Alerts the system in the case of out-of-limit events, such as over temperature. Can be configured as sticky SMBus mode or comparator mode.
15	SDA	Serial Bus Bidirectional Data. Connects to the SMBus master's data line. Requires pullup resistor if not provided elsewhere in the system.
16	SCL	Serial SMBus Clock Input. Connects to the SMBus master's clock line. Requires pullup resistor if not already provided in the system.

Table 4. ELECTRICAL CHARACTERISTICS ($T_A = T_{MIN}$ to T_{MAX}, $V_{CC} = V_{MIN}$ to V_{MAX}, unless otherwise noted.) (Note 1)

Parameter	Test Conditions/Comments	Min	Typ	Max	Unit
Power Supply					
Supply Voltage, V_{CC} (Note 2)		3.0	3.30	3.6	V
Supply Current, I_{CC}	Interface inactive, ADC Active	–	–	3.0	mA
	Standby Mode	–	–	900	μA
Undervoltage Lockout Threshold		–	2.5	–	V
Power-On Reset Threshold		1.0	–	2.4	V
Temperature-to-Digital Converter					
Internal Sensor Accuracy	20°C ≤ T_A ≤ 60°C −40°C ≤ T_A ≤ +100°C	– −4.0	±1.0 –	±2.0 +2.0	°C
Resolution		–	0.03125	–	°C
External Diode Sensor Accuracy	−40°C ≤ T_D ≤ +100°C; T_A = +40°C −40°C ≤ T_D ≤ +100°C; +20°C ≤ T_A ≤ +60°C −40°C ≤ T_D ≤ +100°C; −40°C ≤ T_A ≤ +100°C	– – −3.0	±0.5 ±1.0 –	±1.0 ±1.25 +2.0	°C
Resolution		–	0.03125	–	°C
Remote Sensor Source Current	High Level Mid Level Low Level	– – –	85 34 5.0	– – –	μA
Series Resistance Cancellation		–	–	1000	Ω
Power Supply Sensitivity		–	±1.0	–	%/V
Conversion Time (Local Temperature)	Averaging Enabled	–	11	–	ms
Conversion Time (Remote Temperature)	Averaging Enabled	–	32	–	ms
Total Conversion Time	Averaging Enabled	–	43	–	ms
Open-Drain Digital Outputs ($\overline{\text{ALERT}}$, $\overline{\text{THERM}}$, $\overline{\text{FAN\_FAULT}}$, DRIVE1, DRIVE2) (Note 3)					
Output Low Voltage, V_{OL}	I_{OUT} = −6.0 mA; V_{CC} = +3.0 V	–	–	0.4	V
High Level Output Leakage Current, I_{OH}	V_{OUT} = V_{CC}; V_{CC} = 3.0 V	–	0.1	1.0	μA
Digital Input Leakage Current (TACH1, TACH2)					
Input High Current, I_{IH}	$-V_{IN}$ = V_{CC}	−1.0	–	–	μA
Input Low Current, I_{IL}	V_{IN} = 0	–	–	1.0	μA
Input Capacitance, C_{IN}		–	7.0	–	pF
Digital Input Logic Levels (TACH1, TACH2)					
Input High Voltage, V_{IH}		2.0	–	5.5	V
Input Low Voltage, V_{IL}		−0.3	–	+0.8	V
Hysteresis		–	500	–	mV p−p
Open-Drain Serial Data Bus Output (SDA)					
Output Low Voltage, V_{OL}	I_{OUT} = −6.0 mA; V_{CC}	–	–	0.4	V
High Level Output Leakage Current, I_{OH}	V_{OUT} = V_{CC}	–	0.1	1.0	μA
Serial Bus Digital Inputs (SCL, SDA)					
Input High Voltage, V_{IH}		2.1	–	–	V
Input Low Voltage, V_{IL}		–	–	0.8	V
Hysteresis		–	500	–	mV
Analog Inputs (Location, REF)					
Input Resistance		80	125	160	kΩ
Tachometer Accuracy					
Fan Speed Measurement Accuracy		–	–	±4.0	%

Table 4. ELECTRICAL CHARACTERISTICS ($T_A = T_{MIN}$ to T_{MAX}, $V_{CC} = V_{MIN}$ to V_{MAX}, unless otherwise noted.) (Note 1)

Parameter	Test Conditions/Comments	Min	Typ	Max	Unit
AGTL + INPUT ($\overline{\text{THERM}}$)					
Input High Level		–	0.75 x REF	–	V
Input Low Level		–	–	0.4	V
$\overline{\text{SERIAL BUS TIMING}}$ (Note 4)					
Clock Frequency, f_{SCLK}	See Figure 2 for All Parameters	–	–	400	kHz
Glitch Immunity, t_{SW}		–	50	–	ns
Bus Free Time, t_{BUF}		1.3	–	–	μs
Start Setup Time, $t_{SU:STA}$		0.6	–	–	μs
Start Hold Time, $t_{HD:STA}$		0.6	–	–	μs
Stop Condition Setup Time $t_{SU:STO}$		0.6	–	–	μs
SCL Low Time, t_{LOW}		1.3	–	–	μs
SCL High Time, t_{HIGH}		0.6	–	–	μs
SCL, SDA Rise Time, t_r		–	–	1000	ns
SCL, SDA Fall Time, t_f		–	–	300	ns
Data Setup Time, $t_{SU:DAT}$		100	–	–	ns
Detect Clock Low Timeout, $t_{TIMEOUT}$	(Note 5)	25	–	35	ms

1. Typicals are at T_A = 25°C and represent most likely parametric norm. Standby current typ. is measured with V_{CC} = 3.3 V. Timing specifications are tested at logic levels of V_{IL} = 0.8 V for a falling edge and V_{IH} = 2.1 V for a rising edge.
2. Operation at 5.5 V is guaranteed by design, not production tested.
3. Recommend use of 100 kΩ pullup resistors for all open-drain outputs from the ADM1033.
4. Guaranteed by design, not production tested.
5. SMBus timeout disabled by default. See the SMBus section for more information.

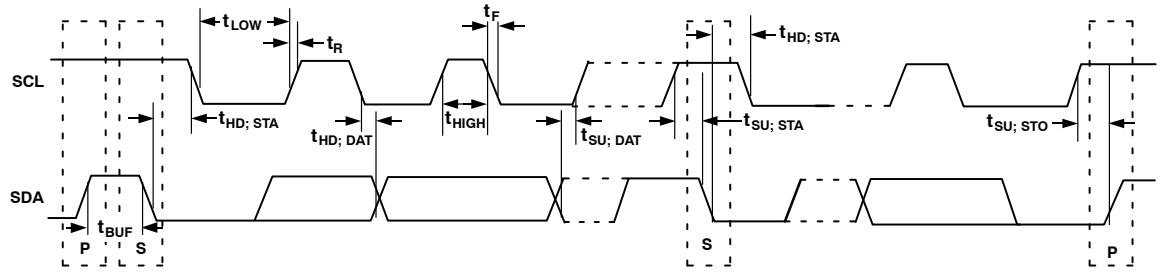


Figure 2. Serial Bus Timing Diagram

TYPICAL PERFORMANCE CHARACTERISTICS

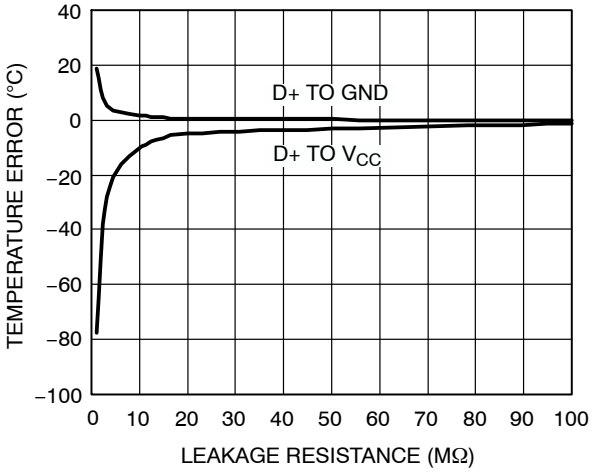


Figure 3. Temperature Error vs. PCB Track Resistance DXP to GND and V_{CC}

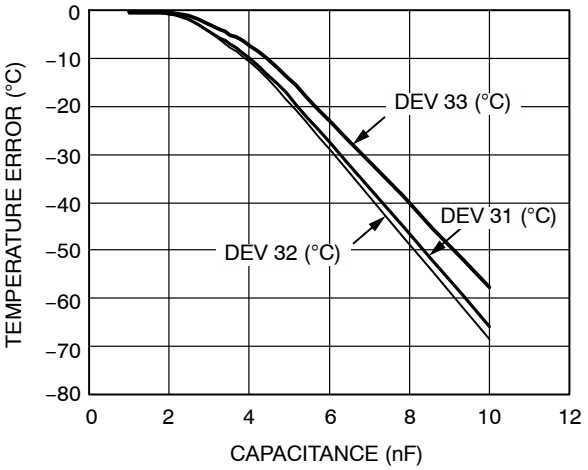


Figure 4. Remote Temperature Error vs. D+, D− Capacitance

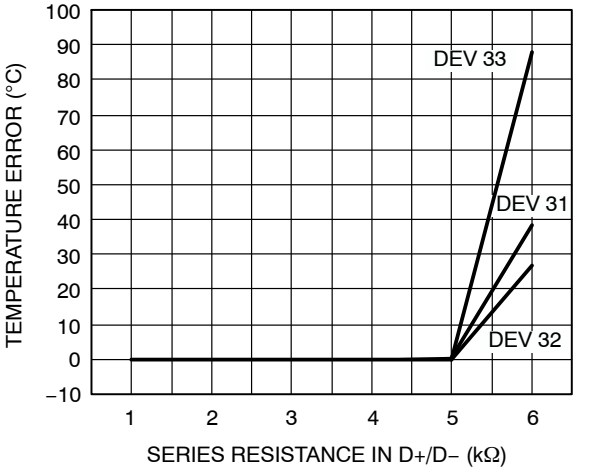


Figure 5. Remote Temperature Error vs. Series Resistance on D+ and D−

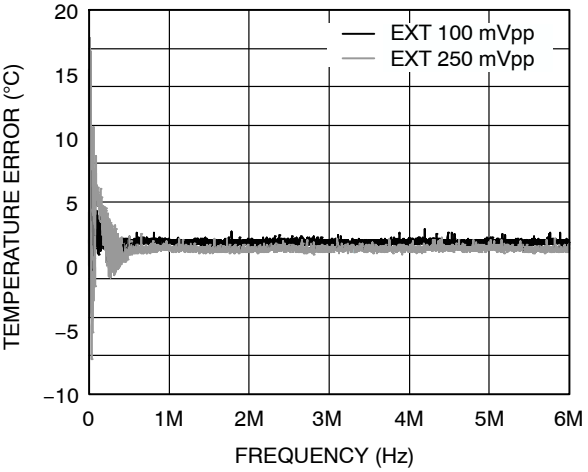


Figure 6. Remote Temperature Error vs. Power Supply Noise Frequency

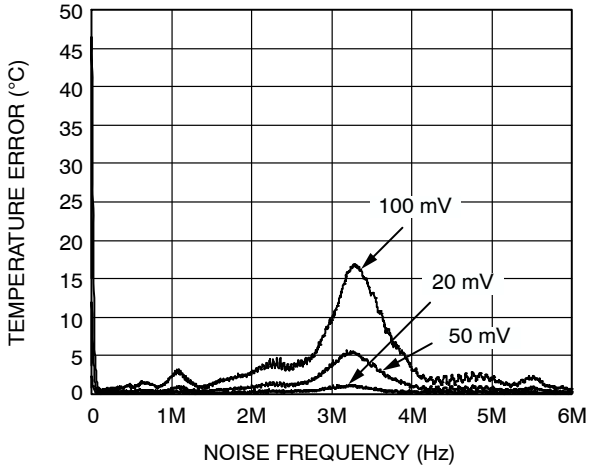


Figure 7. Remote Temperature Error vs. Common-Mode Noise Frequency Coupled on D+ and D−

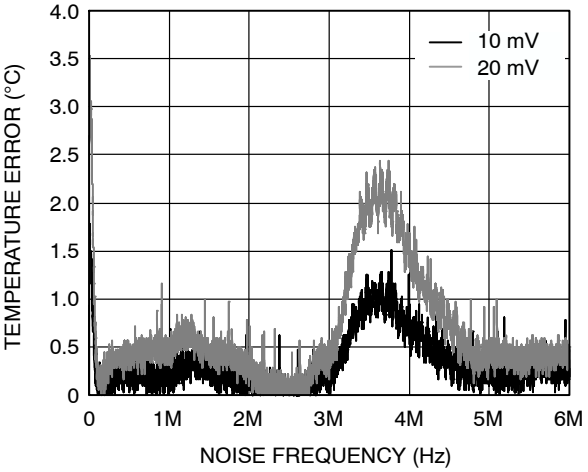


Figure 8. Remote Temperature Error vs. Differential Mode Noise Frequency Coupled on D+ and D−

TYPICAL PERFORMANCE CHARACTERISTICS (Cont'd)

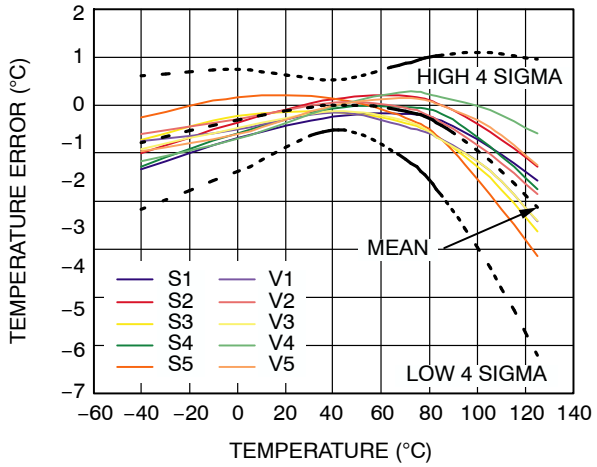


Figure 9. Remote 1 Temperature Error vs. Actual Temperature

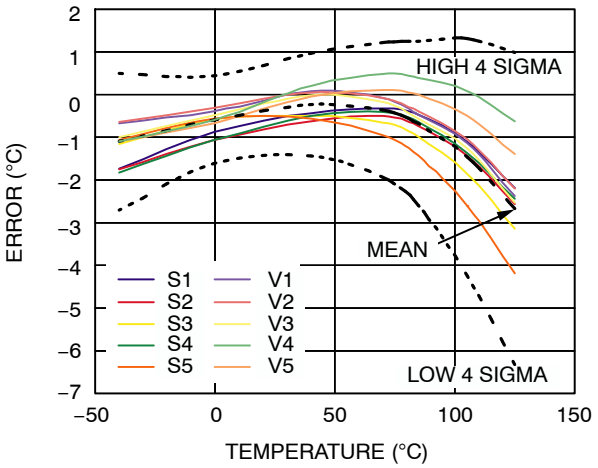


Figure 10. Local Temperature Error vs. Actual Temperature

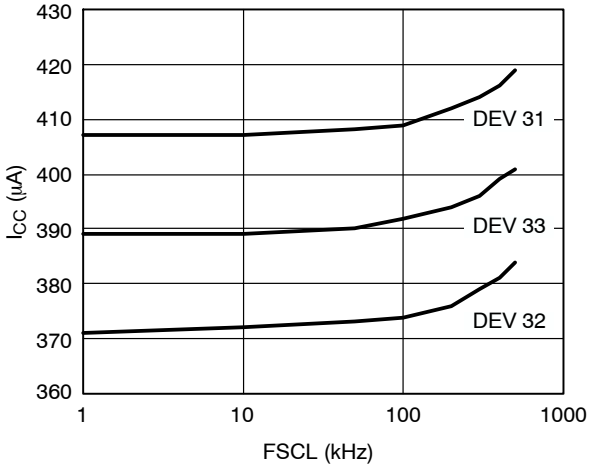


Figure 11. Standby Supply Current vs. SCLK Frequency

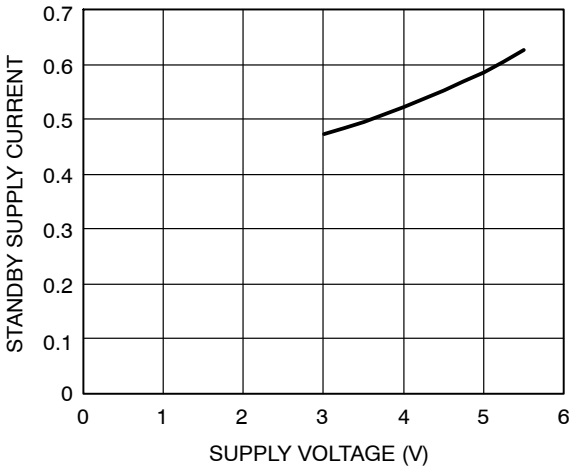


Figure 12. Standby Supply Current vs. Supply Voltage

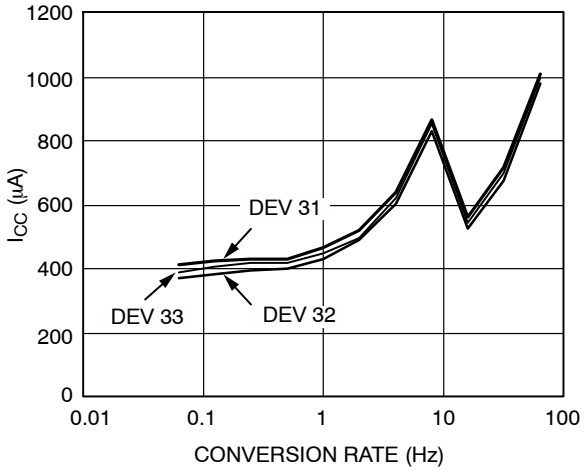


Figure 13. Supply Current vs. Conversion Rate

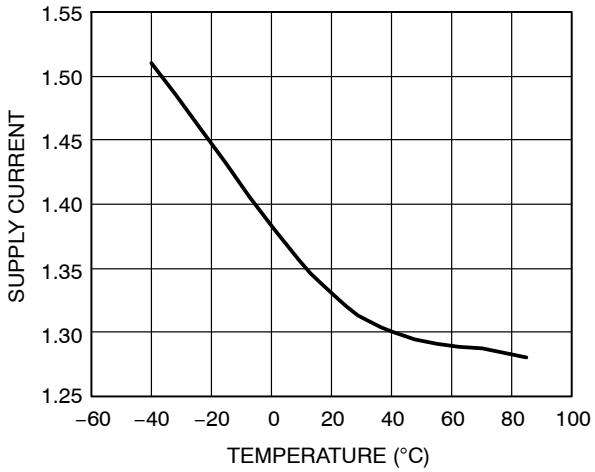


Figure 14. Supply Current vs. ADM1033 Temperature

Functional Description

The ADM1033 is a local- and remote-temperature monitor and fan controller for use in a variety of applications, including microprocessor-based systems. The device accurately monitors remote and ambient temperature and uses that information to quietly control the speed of a cooling fan. Whenever a fan stalls, the device asserts a $\overline{\text{FAN\_FAULT}}$ output.

The ADM1033 features a $\overline{\text{THERM}}$ I/O. As an input, this measures assertions on the $\overline{\text{THERM}}$ pin. As an output, it asserts a low signal to indicate when the measured temperature exceeds the programmed $\overline{\text{THERM}}$ temperature. The ADM1033 communicates over an SMBus 2.0 interface. Its LOCATION input determines which version of SMBus to use, as well as the SMBus address (in fixed and discoverable mode) and the LOCATION bits in the UDID (in ARP-capable mode).

Internal Registers

Table 5 gives a brief description of the ADM1033's principal internal registers. For more detailed information on the function of each register, refer to Table 35.

Serial Bus Interface

The ADM1033 communicates with the master via the 2-wire SMBus 2.0 interface. It supports two versions of SMBus 2.0, determined by the value of the LOCATION input's resistors.

The first version is fully ARP-capable. This means that it supports address resolution protocol (ARP), allowing the master to dynamically address the device on powerup. It responds to ARP commands such as "Prepare to ARP."

The second SMBus version, fixed and discoverable, is backwards compatible with SMBus 1.0 and 1.1. In this mode, the ADM1033 powers up with a fixed address, which is determined by the state of the LOCATION pin on powerup.

NOTE: When using the ADM1033, Addresses 0xC2 and 0xCA should not be used by any other device on the bus.

Location Input

The LOCATION input is a resistor divider input. It has multiple functions and can specify the SMBus version (in fixed and discoverable or ARP-capable modes); the SMBus address (in fixed and discoverable mode); and the LLL bits (in UDID in ARP-capable mode).

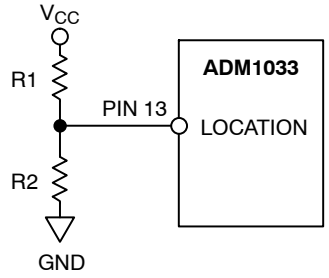


Figure 15. Bootstrapping the LOCATION Input

The voltage of this 8-level input is set by a potential divider. The voltage on LOCATION is sampled on powerup and digitized by the on-chip ADC to determine the LOCATION input value. Because the LOCATION input is sampled only at powerup, changes made while power is applied have no effect.

SMBus 2.0 ARP-Capable Mode

In ARP-capable mode, the ADM1033 supports features such as address resolution protocol (ARP) and unique device identifier (UDID). The UDID is a 128-bit message that describes the ADM1033's capabilities to the master. The UDID also includes a vendor specific ID for functionally equivalent devices.

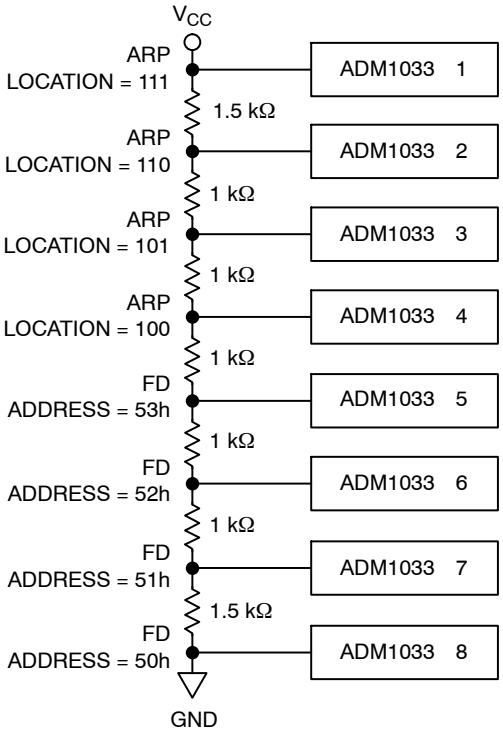


Figure 16. Setting Up Multiple ADM1033 Addresses in SMBus 2.0 ARP-capable Mode

In SMBus 2.0 mode, this vendor specific ID is generated by an on-chip random number generator. This should enable two adjacent ADM1033s in the same system to powerup with a different vendor specific ID, allowing the master to identify the two separate ADM1033's and assign a different address to each.

The state of the LOCATION input on powerup is also reflected in the UDID. This is useful when there is more than one ADM1033 in the system, so the master knows which one it is communicating with. The complete UDID is listed in Table 7.

The SMBus 2.0 master issues both general and directed ARP commands. A general command is directed at all ARP devices. A directed command is targeted at a single device once an address has been established. The PEC byte must be used for ARP commands. (Refer to the Packet Error Checking (PEC) section.)

The ADM1033 responds to the following commands:

- Prepare to ARP (General)
- Reset Device (General and Directed)
- Get UDID (General and Directed)
- Assign Address (General)

Table 5. INTERNAL REGISTER DESCRIPTIONS

Register	Description
Configuration	Provides control and configuration of various functions on the device.
Conversion Rate	Determines the number of measurements per second completed by the ADM1033.
Address Pointer	Contains the address that selects one of the other internal registers. When writing to the ADM1033, the first byte of data is always a register address, written to the address pointer register.
Status	Provides the status of each limit comparison.
Interrupt Mask	Allows the option to mask $\overline{\text{ALERT}}$s due to particular out-of-limit conditions.
Value and Limit	Stores the results of temperature and fan speed measurements, along with their limit values.
Offset	Allows the local and remote temperature channel readings to be offset by a twos complement value written to them. These values are automatically added to the temperature values (or subtracted from them if negative). This allows the systems designer to optimize the system if required, by adding or subtracting up to 15.875°C from a temperature reading.
$\overline{\text{THERM}}$ Limit and Hysteresis	Contains the temperature value at which $\overline{\text{THERM}}$ is asserted and indicates the level of hysteresis.
Look-up Table	Used to program the look-up table for the fan speed-to-temperature profile.
$\overline{\text{THERM}}$ % On-time and $\overline{\text{THERM}}$ % Limit	Reflects the state of the $\overline{\text{THERM}}$ input and monitors the duration of the assertion time of the signal as a percentage of a time window. The user can program the length of the time window.

Table 6. RESISTOR RATIOS FOR SETTING LOCATION BITS

Ideal Ratio R2/(R1 + R2)	R1 kΩ	R2 Ω	Actual R2/(R1 + R2)	Error %	SMBus Ver (Note 1)	SMBus Address	UDID LLL
N/A	0	O/C	1	0	ARP	N/A	111
0.8125	18	82	0.82	+0.75	ARP	N/A	110
0.6875	22	47	0.6812	−0.63	ARP	N/A	101
0.5625	12	15	0.5556	−0.69	ARP	N/A	100
0.4375	15	12	0.4444	+0.69	FD	0x53	N/A
0.3125	47	22	0.3188	+0.63	FD	0x52	N/A
0.1875	82	18	0.18	−0.75	FD	0x51	N/A
N/A	O/C	0	0	0	FD	0x50	N/A

1. ARP denotes ARP-capable mode, FD denotes fixed and discoverable mode.

Table 7. UDID VALUES

Bit No.	Name	Function	Value
<127:120>	Device Capabilities	Describes the ADM1033's capabilities (for instance, that it supports PEC and uses a random number address device).	11000001
<119:112>	Version/Revision	UDID version number (Version 1) and silicon revision identification	00001010
<111:96>	Vendor ID	Analog Devices vendor ID number, as assigned by the SBS Implementer's Forum or the PCI SIG.	00010001 11010100
<95:80>	Device ID	Device ID.	00010000 00110100
<79:64>	Interface	Identifies the protocol layer interfaces supported by the ADM1033. This represents SMBus 2.0 as the Interface version..	00000000 00000100
<63:48>	Subsystem Vendor ID	Subsystem Vendor ID = 0 (subsystem fields are unsupported).	00000000 00000000
<47:32>	Subsystem Device ID	Subsystem Device ID = 0 (subsystem fields are unsupported).	00000000 00000000
<31:0>	Vendor Specific ID	A unique number per device. Contains LOCATION information (LL) and a 16-bit random number (x). See Table 9 for information on setting the LLL bits.	00000000 00000LLL xxxxxxxx xxxxxxxx

SMBus 2.0 Fixed and Discoverable Mode

The ADM1033 also supports fixed and discoverable mode, which is backwards compatible with SMBus 1.0 and 1.1. Fixed and discoverable mode supports all the same functionality as ARP-capable mode, except for assign address in which case it powers up with a fixed address and is not changed by the assign address call. The fixed address is determined by the state of the LOCATION pin on powerup.

SMBus 2.0 Read and Write Operations

The master initiates data transfer by establishing a start condition, defined as a high-to-low transition on the serial data line (SDA) while the serial clock line (SCL) remains high. This indicates that an address/data stream is to follow. All slave peripherals connected to the serial bus respond to the start condition and shift in the next 8 bits, which consist of a 7-bit address (MSB first) plus an R/$\overline{W}$ bit. This last bit determines the direction of the data transfer (whether data is written to or read from the slave device).

1. The peripheral that corresponds to the transmitted address responds by pulling the data line low during the low period before the 9th clock pulse, which is known as the acknowledge bit. All other devices on the bus remain idle while the selected device waits for data to be read from or written to it. If the R/$\overline{W}$ bit is a 0, the master writes to the slave device. If the R/$\overline{W}$ bit is a 1, the master reads from it.
2. Data is sent over the serial bus in sequences of 9 clock pulses − 8 bits of data followed by an acknowledge bit from the slave device. Transitions on the data line must occur during the low period of the clock signal and remain stable during the high period, because a low-to-high transition when the clock is high may be interpreted as a stop signal. The number of data bytes that can be transmitted over the serial bus in a single read or write operation is limited only by what the master and slave devices can handle.
3. When all data bytes have been read or written, stop conditions are established. In write mode, the master pulls the data line high during the 10th clock pulse to assert a stop condition. In read mode, the master device overrides the acknowledge bit by pulling the data line high during the low period before the 9th clock pulse. This is known as no acknowledge. The master takes the data line low during the low period before the 10th clock pulse, then high during the 10th clock pulse to assert a stop condition.

It is not possible to mix read and write in one operation, because the type of operation is determined at the beginning and cannot be changed without starting a new operation.

To write data to one of the device data registers or to read data from it, the address pointer register (APR) must be set so that the correct data register is addressed; then data can be written into that register or read from it. The first byte of a write operation always contains an address that is stored in the APR. If data is to be written to the device, then the write operation contains a second data byte, which is written to the register selected by the APR.

As illustrated in Figure 17, the device address is sent over the bus, followed by R/$\overline{W}$ set to 0. This is followed by two data bytes. The first data byte is the address of the internal data register to be written to, which is stored in the APR. The second data byte is the data to be written to the internal data register.

When reading data from a register there are two possibilities.

If the ADM1033's APR value is unknown or incorrect, it must be set to the correct value before data can be read from the desired data register. To do this, perform a write to the ADM1033 as before, but send only the data byte containing the register (See Figure 18.) A read operation is then performed, using the serial bus address and the R/$\overline{W}$ bit set to 1, followed by the data byte read from the data register. (See Figure 19.)

However, if the APR is already at the desired address, data can be read from the corresponding data register without first writing to the APR. In this case, see Figure 18 can be omitted.

In Figure 17 to Figure 19, the serial bus address is determined by the state of the LOCATION pin on powerup.

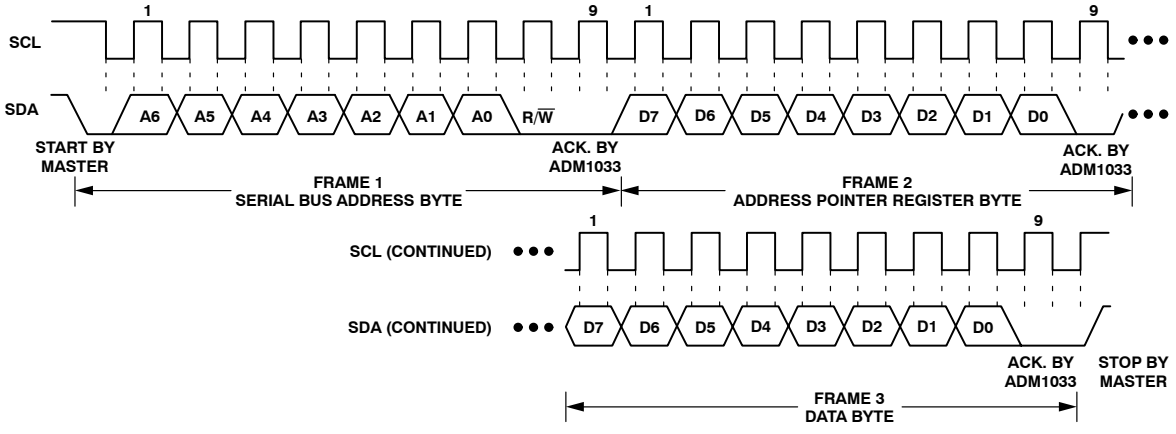


Figure 17. Writing a Register Address to the Address Pointer Register, then Writing Data to the Selected Register

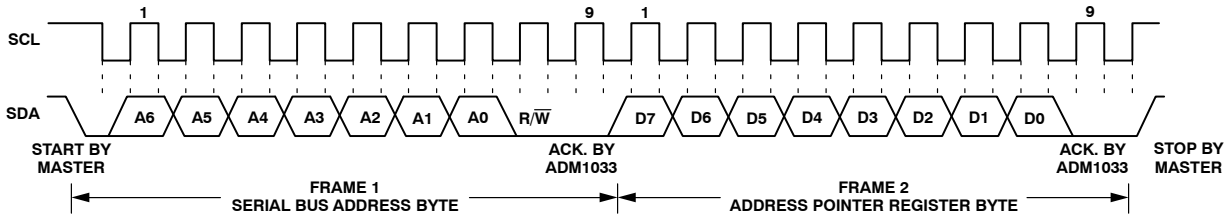


Figure 18. Writing to the Address Pointer Register Only (Send Byte)

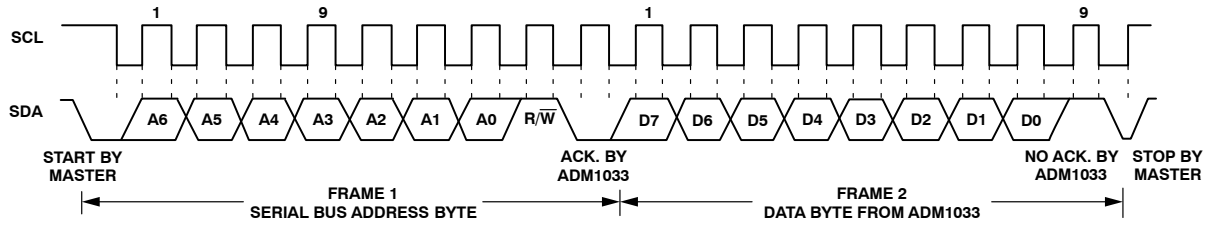


Figure 19. Reading Data from a Previously Selected Register

Register Addresses for Single/Block Byte Modes

The ADM1033 supports single byte as well as block read and write operations. The register address determines whether a single byte or multiple byte (block) operation is run. For a single byte operation, the MSB of the register address is set to 0; for a multiple byte operation, it is set to 1. The number of bytes read in a multiple byte operation is set in the #Bytes/Block Read Register at Address 0x00. The number of bytes written to the ADM1033 is specified during the block write operation. The addresses quoted in the register map and throughout this data sheet assume single byte operation. For multiple byte operations, set the MSB of each register address to 1.

Write Operations

The SMBus specifications define protocols for read and write operations. The ADM1033 supports send byte, write byte, and block byte SMBus write protocols. The following abbreviations are used in the diagrams:

S – START
P – STOP
R – READ
W – WRITE
A – ACKNOWLEDGE
$\overline{A}$ – NO ACKNOWLEDGE

Send Byte

In this operation, the master device sends a single-command byte to a slave device as follows:

1. The master device asserts a start condition on SDA.
2. The master sends a 7-bit address followed by the write bit (low).
3. The addressed slave device asserts ACK on SDA.
4. The master sends the register address.
5. The slave asserts ACK on SDA.
6. The master asserts a stop condition on SDA, and the transaction ends.

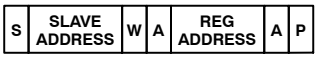


Figure 20. Send Byte

The ADM1033 uses the send byte operation to write a register address to the APR for a subsequent read from the same address. (See Figure 24). The user may be required to read data from the register immediately after setting up the address. If so, the master can assert a repeat start condition immediately after the final ACK and carry out a single byte read without asserting an intermediate stop condition.

Write Byte

In this operation, the master device sends a register address and one data byte to the slave device as follows:

1. The master asserts a start condition on SDA.
2. The master sends the 7-bit slave address followed by a write bit (low).
3. The addressed slave device asserts ACK on SDA.
4. The master sends the register address. The MSB of the register address should equal 0 for a write byte operation. If the MSB equals 1, a block write operation takes place.
5. The slave asserts ACK on SDA.
6. The master sends a data byte.
7. The slave asserts ACK on SDA.
8. The master asserts a stop condition on SDA to end the transaction.

Figure 21. Write Byte Operation

Block Write

In this operation, the master device writes a block of data to a slave address as follows. A maximum of 32 bytes can be written.

1. The master asserts a start condition on SDA.
2. The master sends the 7-bit slave address followed by a write bit (low).
3. The addressed slave device asserts ACK on SDA.
4. The master sends the register address. The register address sets up the address pointer register and determines whether a block write (MSB = 1) or a byte write (MSB = 0) takes place.
5. The slave asserts ACK on SDA.
6. The master sends the byte count.
7. The slave asserts ACK on SDA.
8. The master sends N data bytes.
9. The slave asserts ACK on SDA after each byte.
10. The master asserts a stop condition on SDA to end the transaction.

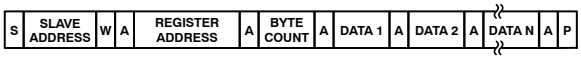


Figure 22. Block Write to RAM

Read Operations

Receive Byte

This is useful when repeatedly reading a single register. The register address must be set up prior to this, with the MSB at 0 to read a single byte. In this operation, the master device receives a single byte from a slave device as follows:

1. The master device asserts a start condition on SDA.
2. The master sends the 7-bit slave address followed by the read bit (high).
3. The addressed slave device asserts ACK on SDA.
4. The master receives a data byte.
5. The master sends NO ACK on SDA.
6. The master asserts a stop condition on SDA, and the transaction ends.

In the ADM1033, the receive byte protocol is used to read a single byte from a register whose address has previously been set by a send byte or write byte operation.

Figure 23. Receive Byte

Block Read

In this operation, the master reads a block of data from a slave device. The number of bytes to be read must be set in advance. To do this, use a write byte operation to the #Bytes/Block Read Register at Address 0x00. The register address determines whether a block-read or a read-byte operation is to be completed (set MSB to 1 to specify a block-read operation). A maximum of 32 bytes can be read.

1. The master asserts a start condition on SDA.
2. The master sends the 7-bit slave address followed by the write bit (low).
3. The addressed slave device asserts ACK on SDA.
4. The master sends the register address (MSB = 1).
5. The slave asserts ACK on SDA.
6. The master asserts a repeated start on SDA.
7. The master sends the 7-bit slave address followed by the read bit (high).
8. The slave asserts ACK on SDA.
9. The slave sends the byte count.
10. The master asserts ACK on SDA.
11. The slave sends N data bytes.
12. The master asserts ACK on SDA after each data byte.
13. The master does not acknowledge after the Nth data byte.
14. The master asserts a stop condition on SDA to end the transaction.

Figure 24. Block Read from RAM

SMBus Timeout

The ADM1033 has a programmable SMBus timeout feature. When this is enabled, the SMBus typically times out after 25 ms of no activity. The timeout is disabled by default. It prevents hangups by releasing the bus after a period of inactivity.

To enable the SDA timeout, set the SDA timeout bit (Bit 5) of Configuration Register 1 (Address 0x01) to 1.

To enable the SCL timeout, set the SCL timeout bit (Bit 4) of Configuration Register 1 (Address 0x01) to 1.

Packet Error Checking (PEC)

The ADM1033 also supports packet error checking (PEC). This optional feature is triggered by the extra clock for the PEC byte. The PEC byte is calculated using CRC-8. The frame check sequence (FCS) conforms to CRC-8 by the following:

$$C(x) = x^8 + x^2 + x + 1 \qquad \text{(eq. 1)}$$

For more information, consult www.SMBus.org.

Alert Response Address (ARA)

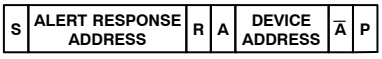


Figure 25. ALERT Response Address

When multiple devices exist on the same bus, the ARA feature allows an interrupting device to identify itself to the host.

The $\overline{\text{ALERT}}$ output can be used as an interrupt output or as an $\overline{\text{SMBusALERT}}$. One or more $\overline{\text{ALERT}}$ outputs can be connected to a common $\overline{\text{SMBusALERT}}$ line, connected to the master.

If a device's $\overline{\text{ALERT}}$ line goes low, the following occurs:

1. $\overline{\text{SMBusALERT}}$ is pulled low.
2. The master initiates a receive byte operation and sends the alert response address (ARA 0001 100). This is a general call address that must not be used as a specific address.
3. The device with the low $\overline{\text{ALERT}}$ output responds to the ARA, and the master reads its device address. Once the address is known, it can be interrogated in the usual way.
4. If low $\overline{\text{ALERT}}$ output is detected in more than one device, the one with the lowest device address has priority, in accordance with normal SMBus arbitration.
5. Once the ADM1033 has responded to the ARA, it resets its $\overline{\text{ALERT}}$ output. However, if the error persists, the $\overline{\text{ALERT}}$ is re-asserted on the next monitoring cycle.

Temperature Measurement System

Internal Temperature Measurement

The ADM1033 contains an on-chip band gap temperature sensor. The on-chip ADC performs conversions on the sensor's output, outputting the data in 13-bit format. The resolution of the local temperature sensor is 0.03125°C.

Table 8 shows the format of the temperature data MSBs. Table 9 shows the same for the LSBs. To ensure accurate readings, read the LSBs first. This locks the current LSBs and MSBs until the MSBs are read. They then start to update again. (Reading only the MSBs does not lock the registers.) Temperature updates to the look-up table take place in parallel; so fan speeds may be updated even if the MSBs are locked.

Table 8. TEMPERATURE DATA FORMAT (LOCAL TEMPERATURE AND REMOTE TEMPERATURE HIGH BYTES)

Temperature (°C)	Digital Output
−64°C	0000 0000
−40°C	0001 1000
−32°C	0010 0000
−2°C	0011 1110
−1°C	0011 1111
0°C	0100 0000
1°C	0100 0001
2°C	0100 0010
10°C	0100 1010
20°C	0101 0100
50°C	0111 0010
75°C	1000 1011
100°C	1010 0100
125°C	1011 1101
150°C	1101 0110
191°C	1111 1111

Table 9. LOCAL AND REMOTE SENSOR EXTENDED RESOLUTION

Extended Resolution (°C)	Temperature Low Bits
0.0000	00000
0.03125	00001
0.0625	00010
0.125	00100
0.250	01000
0.375	01100
0.500	10000
0.625	10100
0.750	11000
0.875	11100

Temperature (°C) = (MSB − 64°C) + (LSB x 0.03125)
Example: MSB = 0101 0100 = 84d
LSB = 11100 = 28
Temperature °C = (84 – 64) + (28 x 0.03125) = 20.875

Remote Temperature Measurement

The ADM1033 can measure the temperature of external diode sensor or diode-connected transistor, which are connected to Pins 9 and 10. These pins are dedicated temperature input channels. The series resistance cancellation (SRC) feature can automatically cancel out the effect of up to 1 kΩ of resistance in series with the remote thermal diode.

The forward voltage of a diode or diode-connected transistor, operated at a constant current, exhibits a negative temperature coefficient of about −2 mV/°C. Unfortunately, the absolute value of V_{BE} varies from device to device, and individual calibration is required to null this out. Therefore, the technique is unsuitable for mass production.

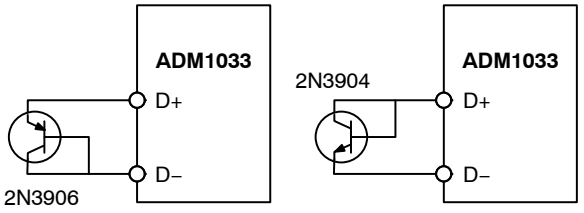


Figure 26. Measuring Temperature by Using Discreet Transistors

The ADM1033 operates at three different currents to measure the change in V_{BE}. Figure 27 shows the input signal conditioning used to measure the output of an external temperature sensor. It also shows the external sensor as a substrate transistor, provided for temperature monitoring on some microprocessors. The external sensor could work equally well as a discrete transistor.

If a discrete transistor is used, the collector is not grounded, and should be linked to the base. If a PNP transistor is used, the base is connected to the D− input and the emitter to the D+ input. If an NPN transistor is used, the emitter is connected to the D− input and the base to the D+ input.

If the sensor is used in a very noisy environment, a capacitor value up to 1000 pF may be placed between the D+ and D− inputs to filter the noise. However, additional parasitic capacitance on the lines between D+, D−, and the thermal diode should also be considered. The total capacitance should never be greater than 1000 pF.

To measure each ΔV_{BE}, the sensor is switched between operating currents of I, (N1 × I), and (N2 × I). The resulting waveform is passed through a 65 kHz low-pass filter to remove noise, then to a chopper-stabilized amplifier that amplifies and rectifies the waveform. This produces a dc voltage proportional to ΔV_{BE}. These voltage measurements determine the temperature of the thermal diode, while automatically compensating for any series resistance on the D+ and/or D− lines. The temperature is stored in two registers as a 13-bit word.

To further reduce the effects of noise, digital filtering is performed by averaging the results of 16 measurement cycles at conversion rates of less than or equal to 8 Hz. An external temperature measurement takes nominally 32 ms when averaging is enabled and 6 ms when averaging is disabled.

One LSB of the ADC corresponds to 0.03125°C. The ADM1033 can theoretically measure temperatures from −64°C to +191.96875°C, although these are outside its operating range. The extended temperature resolution data format is shown in Table 9. The data for the local and remote channels is stored in the extended temperature resolution registers (Reg. 0x40 = Local, Reg. 0x42 = Remote 1).

Table 10. TEMPERATURE MEASUREMENT REGISTERS

Register	Description	Default
0x40	Local Temperature, LSBs	0x00
0x41	Local Temperature, MSBs	0x00
0x42	Remote 1 Temperature, LSBs	0x00
0x43	Remote 1 Temperature, MSBs	0x00

High and low temperature limit registers are associated with each temperature measurement channel. Exceeding the programmed high and low limits sets the appropriate status bit. Exceeding either limit can cause an $\overline{\text{SMBusALERT}}$ interrupt.

Table 11. TEMPERATURE MEASUREMENT LIMIT REGISTERS

Register	Description	Default
0x0B	Local High Limit	0x8B (75°C)
0x0C	Local Low Limit	0x54 (20°C)
0x0D	Local $\overline{\text{THERM}}$ Limit	0x95 (85°C)
0x0E	Remote 1 High Limit	0x8B (75°C)
0x0F	Remote 1 Low Limit	0x54 (20°C)
0x10	Remote 1 $\overline{\text{THERM}}$ Limit	0x95 (85°C)

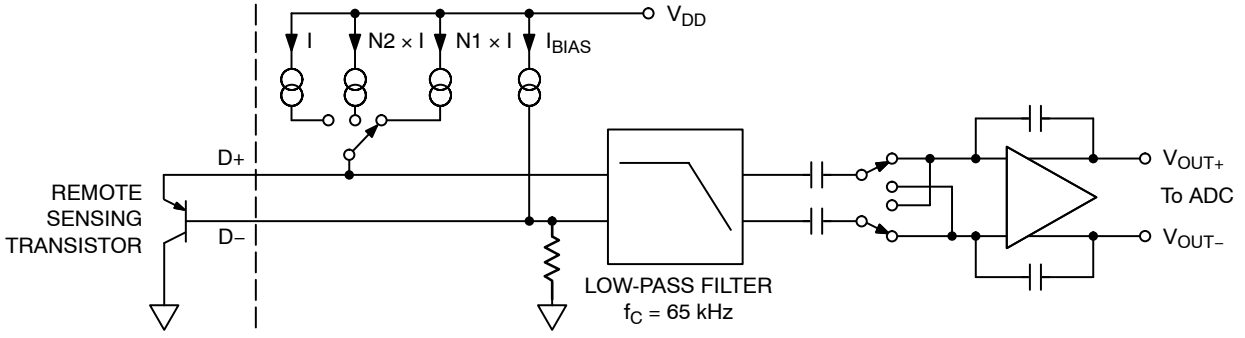


Figure 27. ADM1033 Signal Conditioning

Additional Functions

Several other temperature measurement functions available on the ADM1033 offer the systems designer added flexibility.

Turn-off Averaging

The ADM1033 performs averaging at conversion rates of less than or equal to 8 conversions per second. This means that the value in the measurement register is the average of 16 measurements. For faster measurements, set the conversion rate to 16 conversions per second or greater. (Averaging is not carried out at these conversion rates.) Alternatively, switch off averaging at the slower conversion rates by setting Bit 1 (AVG) of Configuration 1 Register (Address 0x01).

Single-channel ADC Conversions

In normal operating mode, the ADM1033 converts on two temperature channels: the local temperature channel, and the remote channel. However, the user has the option to set up the ADM1033 to convert on one channel only. To enable single-channel mode, the user sets the round-robin bit (Bit 7) in Configuration Register 2 (Address 0x02) to 0. When the round-robin bit equals 1, the ADM1033 converts on all temperature channels. In single-channel mode, it converts on one channel only, to be determined by the state of the channel selector bits (Bits 5 and 4) of the Configuration Register 2 (Address 0x02).

Table 12. CHANNEL SELECTOR

Bits 5:4	Channel Selector (Configuration 2)
00	Local Channel = Default
01	Remote 1 Channel
10	Reserved
11	Reserved

Removing Temperature Errors

As CPUs run faster and faster, it gets more difficult to avoid high frequency clocks when routing the D+ and D− traces around a system board. Even when the recommended layout guidelines are followed, temperature errors attributed to noise coupled onto the D+ and D− lines remain. High frequency noise generally gives temperature measurements that are consistently too high. The ADM1033 has Local and Remote temperature offset registers at 0x16 and 0x17; one for each channel. By completing a one-time calibration, the user can determine the offset caused by the system board noise and remove it using the offset registers. The registers automatically add a twos compliment word to the remote temperature measurements, ensuring correct readings in the value registers.

Table 13. OFFSET REGISTERS

Registration	Description	Default
0x16	Local Offset	0x00
0x17	Remote 1 Offset	0x00

Table 14. OFFSET REGISTER VALUES

Code	Offset Value
0 0000 000	0°C (Default Value)
0 0000 001	0.125°C
0 0000 111	0.875°C
0 0001 111	1.875°C
0 0111 111	7.875°C
0 1111 111	15.875°C
1 0000 000	−16°C
1 1111 000	−0.875°C

Layout Considerations

Digital boards can be electrically noisy environments. Try to protect the analog inputs from noise, particularly when measuring the very small voltages from a remote diode sensor. Take the following precautions:

- Place the ADM1033 as close as possible to the remote sensing diode. A distance of 4 inches to 8 inches is adequate, provided that the worst noise sources such as clock generators, data/address buses, and CRTs are avoided.
- Route the D+ and D− tracks close together, in parallel, with grounded guard tracks on each side. Provide a ground plane under the tracks if possible.
- Use wide tracks to minimize inductance and reduce noise pickup. At least 5 mil track width and spacing are recommended.

Figure 28. Arrangement of Signal Tracks

- Try to minimize the number of copper/solder joints, because they can cause thermocouple effects. Where copper/solder joints are used, make sure that they are in both the D+ and D− paths and at the same temperature. Thermocouple effects are not a major problem because 1°C corresponds to approximately 200 μV, and thermocouple voltages are approximately 3 μV/°C of temperature difference. Unless there are two thermocouples with a big temperature differential between them, the voltages should be much less than 200 μV.
- Place a 0.1 μF bypass capacitor close to the ADM1033.
- If the distance to the remote sensor is more than 8 inches, twisted pair cable is recommended. This works up to about 6 feet to 12 feet.
- For very long distances (up to 100 feet), use shielded twisted pair such as Belden #8451 microphone cable. Connect the twisted pair to D+ and D− and the shield to GND, close to the ADM1033. Leave the remote end of the shield unconnected to avoid ground loops.

Because the measurement technique uses switched current sources, excessive cable and/or filter capacitance can affect the measurement. When using long cables, the filter capacitor C1 may be reduced or removed. In any case, the total shunt capacitance should never exceed 1000 pF.

Noise Filtering

For temperature sensors operating in noisy environments, common practice is to place a capacitor across the D+ and D− pins to help combat the effects of noise. However, large capacitances affect the accuracy of the temperature measurement, leading to a recommended maximum capacitor value of 1000 pF. While this capacitor reduces the noise, it does not eliminate it, making it difficult to use the sensor in a very noisy environment.

The ADM1033 has a major advantage over other devices when it comes to eliminating the effects of noise on the external sensor. The series resistance cancellation feature allows a filter to be constructed between the external temperature sensor and the part. The effect of any filter resistance seen in series with the remote sensor is automatically cancelled from the temperature.

The construction of a filter allows the ADM1033 and the remote temperature sensor to operate in noisy environments. Figure 29 shows a low-pass R-C-R filter with the following values: R = 100 Ω and C = 1 nF. This filtering reduces both common-mode noise and differential noise.

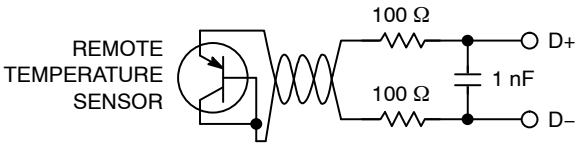


Figure 29. Filter between Remote Sensor and ADM1033

Limits, Status Registers, and Interrupts

High and low limits are associated with each measurement channel on the ADM1033. These can form the basis of system status monitoring. A status bit can be set for any out-of-limit condition and detected by polling the device. Alternatively, $\overline{\text{SMBusALERT}}$s can be generated to flag a processor or microcontroller of an out-of-limit condition.

8-bit Limits

The following is a list of all the 8-bit limits on the ADM1033:

Table 15. TEMPERATURE LIMIT REGISTERS

Register	Description	Default
0x0B	Local High Limit	0x8B (75°C)
0x0C	Local Low Limit	0x54 (20°C)
0x0D	Local $\overline{\text{THERM}}$ Limit	0x95 (85°C)
0x0E	Remote 1 High Limit	0x8B (75°C)
0x0F	Remote 1 Low Limit	0x54 (20°C)
0x10	Remote 1 $\overline{\text{THERM}}$ Limit	0x95 (85°C)

Table 16. $\overline{\text{THERM}}$ LIMIT REGISTERS

Register	Description	Default
0x19	$\overline{\text{THERM}}$ % Limit	0xFF default

Out-of-Limit Comparisons

The ADM1033 measures all parameters in a round-robin format and sets the appropriate status bit for out-of-limit conditions. Comparisons are made differently, depending on whether the measured value is compared to a high or low limit.

High Limit: ≥ Comparison Performed

Low Limit: < Comparison Performed

Analog Monitoring Cycle Time

The analog monitoring cycle time begins on powerup, or, if monitoring has been disabled, by writing a 1 to the monitor/STBY bit of Configuration Register 1, (Address 0x01). The ADC measures each one of the analog inputs in turn; as each measurement is completed, the result is automatically stored in the appropriate value register. The round-robin monitoring cycle continues unless it is disabled by writing a 0 to the monitor/STBY bit (Bit 0) of Configuration Register 1 (Address 0x01).

The ADC performs round-robin conversions and takes 11 ms for the local temperature measurement and 32 ms for each remote temperature measurement with averaging enabled.

The total monitoring cycle time for the average temperatures is therefore nominally.

$$32 + 11 = 43 \text{ ms} \quad \text{(eq. 2)}$$

Once the conversion time elapses, the round robin starts again. For more information, refer to the Conversion Rate Register section.

Fan TACH measurements take place in parallel and are not synchronized with the temperature measurements in any way.

Status Registers

The results of limit comparisons are stored in the status registers. A 1 represents an out-of-limit measurement; a 0 represents an in-limit measurement. The status registers are located at Addresses 0x4F to 0x51.

If the measurement is outside its limits, the corresponding status register bit is set to 1. It remains set at 1 until the measurement falls back within its limits and it is read or until an ARA is completed.

Poll the state of the various measurements by reading the status registers over the serial bus. If Bit 0 ($\overline{\text{ALERT}}$ low) of Status Register 3 (Address 0x51) is set, this means that the $\overline{\text{ALERT}}$ output has been pulled low by the ADM1033.

Pin 14 can be configured as a $\overline{\text{SMBusALERT}}$ output. This automatically notifies the system supervisor of an out-of-limit condition. Reading the status register clears the status bit as long as the error condition is gone.

Status register bits are sticky. Whenever a status bit is set due to an out-of-limit condition, it remains set even after the triggering event has gone. The only way to clear the status bit is to read the status register (after the event has gone). Interrupt mask registers (Reg. 0x08, Reg. 0x09, Reg. 0x0A) allow individual interrupt sources to be masked from causing an $\overline{\text{ALERT}}$. However, if one of these masked interrupt sources goes out of limit, its associated status bit is set in the status register.

Table 17. INTERRUPT STATUS REGISTER 1 (REG. 0X4F)

Bit #	Name	Description
7	LH	1 = Local high temperature limit has been exceeded.
6	LL	1 = Local low temperature limit has been exceeded.
5	R1H	1 = Remote 1 high temperature limit has been exceeded
4	R1L	1 = Remote 1 low temperature limit has been exceeded.
3	R1D	1 = Remote 1 diode error; indicates an open or short on the D1+/D1− pins.
2	Unused	Reserved
1	Unused	Reserved
0	Unused	Reserved

Table 18. STATUS REGISTER 2 (REG. 0X50)

Bit #	Name	Description
7	LT	1 = Local $\overline{\text{THERM}}$ temperature limit has been exceeded.
6	R1T	1 = Remote 1 $\overline{\text{THERM}}$ temperature limit has been exceeded.
5	Unused	Reserved
4	T%	1 = $\overline{\text{THERM}}$ % on-time limit has been exceeded.
3	TA	1 = One of the $\overline{\text{THERM}}$ limits has been exceeded; and the $\overline{\text{THERM}}$ output signal has been asserted.
2	TS	1 = $\overline{\text{THERM}}$ state. Indicates the $\overline{\text{THERM}}$ pin is active; clears on a read if $\overline{\text{THERM}}$ is not active. Does not generate an $\overline{\text{ALERT}}$ in $\overline{\text{ALERT}}$ comp mode.
1	Res	Reserved
0	Res	Reserved

Table 19. STATUS REGISTER 3 (REG. 0X51)

Bit #	Name	Description
7	F1S	1 = Fan 1 has stalled.
6	FA	1 = Fan alarm speed. Fan 1 and Fan 2 are running at alarm speed.
5	Res	Reserved
4	Res	Reserved
3	Res	Reserved
2	Res	Reserved
1	Res	Reserved
0	$\overline{\text{ALERT}}$	1 = $\overline{\text{ALERT}}$ low; indicates the $\overline{\text{ALERT}}$ line has been pulled low.

$\overline{\text{ALERT}}$ Interrupt Behavior

The ADM1033 generates an $\overline{\text{ALERT}}$ whenever an out-of-limit measurement is made (if it is not masked out). The user can also detect out-of-limit conditions by polling the ADM1033 status registers. It is important to note how the SMBus $\overline{\text{ALERT}}$ output behaves when writing interrupt handler software.

The $\overline{\text{ALERT}}$ output on the ADM1033 can be programmed to operate in either $\overline{\text{SMBusALERT}}$ mode or in comp mode.

In $\overline{\text{SMBusALERT}}$ mode, the $\overline{\text{ALERT}}$ output remains low until the measurement falls back within its programmed limits and either the status register is read or an ARA is completed. In comp mode, the $\overline{\text{ALERT}}$ output automatically resets once the temperature measurement falls back within the programmed limits.

Configuring the $\overline{\text{ALERT}}$ Output

For $\overline{\text{SMBusALERT}}$ mode, set the $\overline{\text{ALERT}}$ configuration bit (Bit 3) of the Configuration Register 1 (Address 0x01) to 0.

In $\overline{\text{SMBusALERT}}$ mode, a status bit is set when a measurement goes outside of its programmed limit. If the corresponding mask bit is not set, the $\overline{\text{ALERT}}$ output is pulled low. If the measured value falls back within the limits, the $\overline{\text{ALERT}}$ output remains low until the corresponding status register is read or until an ARA is completed (as long as no other measurement is outside its limits).

For comp mode, set the $\overline{\text{ALERT}}$ configuration bit (Bit 3) of Configuration Register 1 (Address 0x01) to1.

In comp mode, the $\overline{\text{ALERT}}$ output is automatically pulled low when a measurement goes outside its programmed limits. Once the measurement falls back within its limits (and assuming no other measurement channel is outside its limits), the $\overline{\text{ALERT}}$ output is automatically pulled high again.

The main difference between the two modes is that the $\overline{\text{SMBusALERT}}$ does not reset without software intervention, whereas the comp mode $\overline{\text{ALERT}}$ output automatically resets.

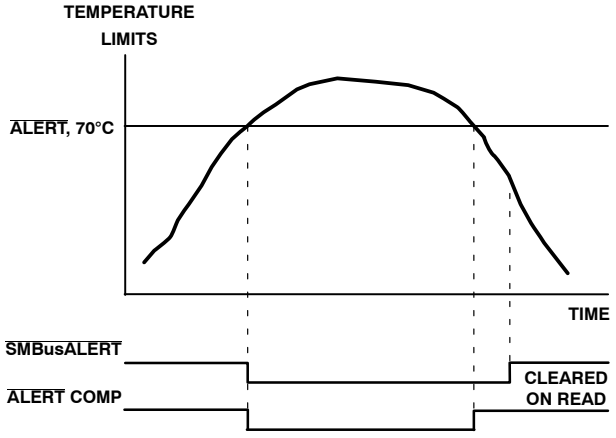


Figure 30. $\overline{\text{ALERT}}$ Comparator and $\overline{\text{SMBusALERT}}$ Outputs

Handling $\overline{\text{SMBusALERT}}$ Interrupts

To prevent tie-ups due to service interrupts, follow these steps:

1. Detect an SMBus assertion.
2. Enter the interrupt handler.
3. Read the status register to identify the interrupt source.

4. Mask the interrupt source by setting the appropriate mask bit in the interrupt mask registers (from Reg. 0x08 to Reg. 0x0A).
5. Take the appropriate action for a given interrupt source.
6. Exit the interrupt handler.
7. Periodically poll the status register. If the interrupt status bit clears, reset the corresponding interrupt mask bit to 0. The $\overline{\text{SMBusALERT}}$ output and status bits then behave as shown in Figure 31.

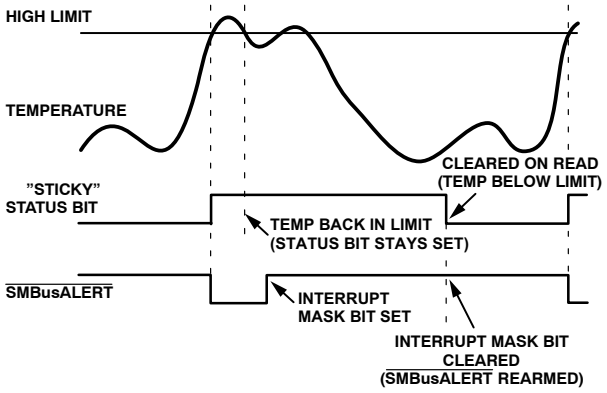


Figure 31. Handling $\overline{\text{SMBusALERT}}$

Interrupt Masking Register

Mask Registers 1, 2, and 3 are located at Addresses 0x08, 0x09, and 0x0A. These allow individual interrupt sources to be masked out to prevent the $\overline{\text{SMBusALERT}}$ interrupts. Masking the interrupt source prevents only the $\overline{\text{SMBusALERT}}$ from being asserted; the appropriate status bit is still set as normal.

Table 20. MASK REGISTER 1 (REG. 0X08)

Bit #	Name	Description
7	LH	1 masks the $\overline{\text{ALERT}}$ for the local high temperature.
6	LL	1 masks the $\overline{\text{ALERT}}$ for the local low temperature.
5	R1H	1 masks the $\overline{\text{ALERT}}$ for the Remote 1 high temperature.
4	R1L	1 masks the $\overline{\text{ALERT}}$ for the Remote 1 low temperature.
3	R1D	1 masks the $\overline{\text{ALERT}}$ for the Remote 1 diode errors.
2	Res	Reserved
1	Res	Reserved
0	Res	Reserved

Table 21. MASK REGISTER 2 (REG. 0X09)

Bit #	Name	Description
7	Res	Reserved
6	Res	Reserved
5	Res	Reserved
4	T%	1 masks the $\overline{\text{ALERT}}$ for the $\overline{\text{THERM}}$ % on-time limit.
3	TA	1 masks the $\overline{\text{ALERT}}$ for the $\overline{\text{THERM}}$ limit being exceeded and the $\overline{\text{THERM}}$ output signal being asserted.
2	TS	1 masks the $\overline{\text{ALERT}}$ for the $\overline{\text{THERM}}$ state; has no effect on $\overline{\text{ALERT}}$ in $\overline{\text{ALERT}}$ comp mode.
1	Res	Reserved
0	Res	Reserved

Table 22. MASK REGISTER 3 (REG. 0X0A)

Bit #	Name	Description
7	F1S	1 mask the $\overline{\text{ALERT}}$ for Fan 1 stalling
6	FA	1 mask the $\overline{\text{ALERT}}$ for fans at $\overline{\text{ALARM}}$ speed
5	Res	Reserved
4	Res	Reserved
3	Res	Reserved
2	Res	Reserved
1	Res	Reserved
0	Res	Reserved

$\overline{\text{FAN\_FAULT}}$ Output

The $\overline{\text{FAN\_FAULT}}$ output signals when one or both of the fans stall. Pin 8, the $\overline{\text{FAN\_FAULT}}$ output, is a dual-function pin. It defaults to being a $\overline{\text{FAN\_FAULT}}$ output but can be reconfigured as an analog input reference for the $\overline{\text{THERM}}$ input. To do this, set the $\overline{\text{FAN\_FAULT}}$/REF (Bit 7) in Configuration Register 4 (Address 0x04) to 1.

Fault Queue

The ADM1033 has a programmable fault queue option that lets the user program the number of out-of-limit measurements allowable before generating an $\overline{\text{ALERT}}$. The fault queue affects only temperature measurement channels and is only operational in $\overline{\text{SMBusALERT}}$ mode. It performs some simple filtering, which is particularly useful at the higher conversion rates (16, 32, and 64 conversions per second), where averaging is not carried out.

There is a queue for each of the temperature channels. If L (the number programmed to the fault queue) or more

consecutive out-of-limit readings are made on the same temperature channel, the fault queue fills, and the $\overline{\text{SMBusALERT}}$ output triggers. To fill the fault queue, one needs L or more consecutive out of limits on the internal temperature channel; L or more consecutive out-of-limits on the external 1 temperature channel; or L or more consecutive out-of-limits on the external 2 temperature channel. The fault queue is independent of the state of the bits in the $\overline{\text{ALERT}}$ status registers.

Table 23. FAULT QUEUE ADDRESS 0X06

Bits <3:0>	Fault Queue
000X	1
001X	2
01XX	3
1XXX	4

To reset the fault queue, do one of the following:

- SMBus ARA Command
- Read Status Register 1
- Power-On Reset

The $\overline{\text{SMBusALERT}}$ clears, even if the condition that caused the $\overline{\text{SMBusALERT}}$ remains. The $\overline{\text{SMBusALERT}}$ is reasserted if the fault queue fills up.

Conversion Rate Register

The ADM1033 makes up to 64 measurements per second. However, for the sake of reduced power consumption and better noise immunity, users may run the ADM1033 at a slower conversion rate. Better noise immunity results from the averaging that occurs at the slower conversion rates. Averaging does not occur at rates of 16, 32, or 64 conversions per second. Table 24 lists the available conversion rates. Note that the current round-robin loop must be finished for conversion rates changes to take effect.

Table 24. CONVERSION RATES

Code	Conversion Rate
0x00	0.0625
0x01	0.125
0x02	0.25
0x03	0.5
0x04	1
0x05	2
0x06	4
0x07	8
0x08	16
0x09	32
0x0A	64
0x0B to 0xFF	Reserved

$\overline{\text{THERM}}$ I/O Timer and Limits

Pin 7 can be configured as either an input or output. As an output it is asserted low to signal that the measured temperature has exceeded preprogrammed temperature limits. The output is automatically pulled high again when the temperature falls below the $\overline{\text{THERM}}$ − Hys limit. The value of hysteresis is programmable in Register 0x1A. $\overline{\text{THERM}}$ is enabled as an output by default on powerup.

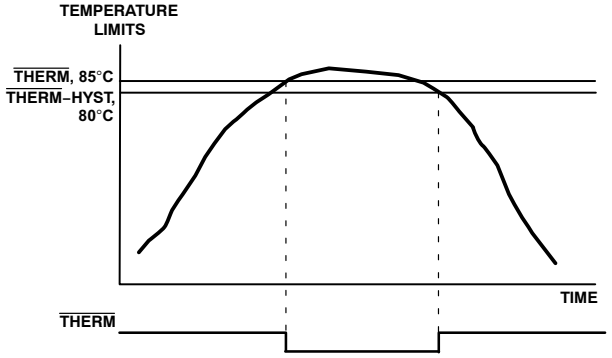


Figure 32. $\overline{\text{THERM}}$ Behavior

Once the $\overline{\text{THERM}}$ limits are exceeded, the fans are boosted to full speed, that is, as long as the Boost Disable Bit (Bit 1) is not set in Configuration Register 2 (Address 0x02).

To configure $\overline{\text{THERM}}$ as an input, the user must set the $\overline{\text{THERM}}$ timer bit (Bit 2) of Configuration Register 1 (Address 0x01) to 1. (It no longer operates as an output.) The ADM1033 can then detect when the $\overline{\text{THERM}}$ input is asserted low. This may be connected to a trip point temperature sensor or to the $\overline{\text{FAN\_FAULT}}$ $\overline{\text{PROCHOT}}$ output of a CPU. With processor core voltages reducing all the time, the threshold for the AGTL + $\overline{\text{PROCHOT}}$ output also reduces down as new processors become available. The default threshold on the input is the normal CMOS threshold. However, Pin 8 ($\overline{\text{FAN\_FAULT}}$/REF) can also be reconfigured as a REF input. This is done by setting Bit 7 ($\overline{\text{FAN\_FAULT}}$/REF) in Configuration Register 4 (Address 0x04). Connect the processor V_{CCP} to this input to provide a reference for the $\overline{\text{THERM}}$ input. The resulting $\overline{\text{THERM}}$ threshold is 0.75 × V_{CCP}, which is the correct threshold for an AGTL + signal.

The ADM1033 also measures assertion times on the $\overline{\text{THERM}}$ input as a percentage of a time window. This time window is programmable in Configuration Register 4 (Address 0x04) by using Bits <6:4> ($\overline{\text{THERM}}$ % Time Window). Values between 0.25 seconds and 8 seconds are programmable. The assertion time as a percentage of the time window is stored in the $\overline{\text{THERM}}$ % On-Time Register (Address 0x4E).

A $\overline{\text{THERM}}$ % limit is also associated with this register. Once the measured percentage exceeds the percentage limit, the

$\overline{\text{THERM}}$ % Exceeded Bit (Bit 4) in Status Register 2 (Address 0x50) is asserted and an $\overline{\text{ALERT}}$ is generated, that is, if the mask bit is not set. If the limit is set to 0x00, an $\overline{\text{ALERT}}$ is generated on the first assertion. If the limit is set to 0xFF, an $\overline{\text{ALERT}}$ is never generated. This is because 0xFF corresponds to the $\overline{\text{THERM}}$ input, which is asserted continuously.

Table 25. CONVERSION RATES

Code	$\overline{\text{THERM}}$ % On-Time Window
000	0.25 s
001	0.5 s
010	1 s
011	2 s
100	4 s
101	8 s
110	8 s
111	8 s

When $\overline{\text{THERM}}$ is configured as an input only, setting the Enable $\overline{\text{THERM}}$ Events bits in Configuration Register 4 allows Pin 7 to operate as an I/O.

The user can configure the $\overline{\text{THERM}}$ pin to be pulled low as an output whenever the local temperature exceeds the local $\overline{\text{THERM}}$ limit. To do this, set the Enable Local $\overline{\text{THERM}}$ events bit (Bit 0) of Configuration Register 4 (Address 0x04).

The user can also configure the $\overline{\text{THERM}}$ pin to be pulled low as an output whenever the Remote 1 temperature exceeds the Remote 1 $\overline{\text{THERM}}$ limit. Set the Enable Remote 1 $\overline{\text{THERM}}$ events bit (Bit 1) of Configuration Register 4 (Address 0x04).

$\overline{\text{THERM}}$ % Limit Register

The $\overline{\text{THERM}}$ % limit is programmed to Register 0x19. An $\overline{\text{ALERT}}$ is generated, if $\overline{\text{THERM}}$ is asserted for longer than the programmed percentage limit. The limit is programmed as a percentage of the chosen time window.

$\overline{\text{THERM}}$ % limit register is an 8-bit register.

0x00 = 0%

0xFF = 100%

Therefore, 1 LSB = 0.39%.

Example:

If a time window of 8 seconds is chosen, and an $\overline{\text{ALERT}}$ is to be generated if $\overline{\text{THERM}}$ is asserted for more than 1 second, program the following value to the limit register:

% Limit = 1/8 × 100 = 12.5%

12.5%/0.39% = 32d = 0x20 = 0010 0000

An $\overline{\text{ALERT}}$ is generated if the $\overline{\text{THERM}}$ limit is exceeded after the time window has elapsed, assuming it is not masked.

Fan Drive Signal

The ADM1033 contols the speed of up to one cooling fan. Varying the duty cycle (on/off time) of a square wave applied to the fan varies the speed of the fan. The ADM1033 uses a control method called synchronous speed control, in which the PWM drive signal applied to the fan is synchronized with the fan's TACH signal. See the Synchronous Speed Control section for more information.

The external circuitry required to drive the fan is very simple. A single N-channel MOSFET is the only drive device required. The specifications of the MOSFET depend on the maximum current required by the fan and the gate voltage drive (V_{GS} < 3.0 V for direct interfacing to the drive pin). V_{GS} can be greater than 3.0 V, as long as the pullup on the gate is tied to 5.0 V. The MOSFET should also have a low on-resistance to ensure that there is no significant voltage drop across the FET. A high on-resistance reduces the voltage applied across the fan and therefore the maximum operating speed of the fan. Figure 33 shows a scheme for driving a 3-wire fan.

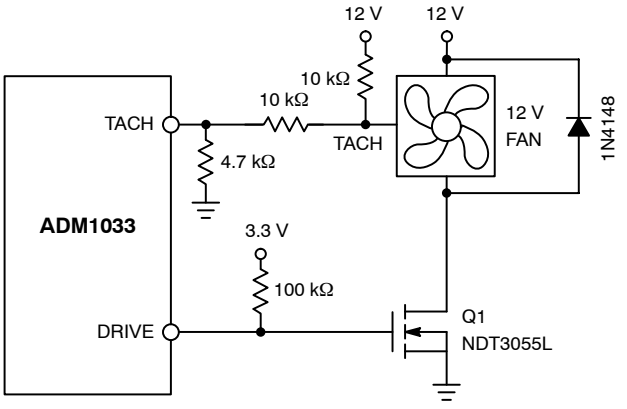


Figure 33. Interfacing a 3-wire Fan to the ADM1033 by Using an N-channel MOSFET

Figure 33 uses a 10 kΩ pullup resistor for the TACH signal. This assumes that the TACH signal is an open collector from the fan. In all cases, the fan's TACH signal must be kept below 5.0 V maximum to prevent damaging the ADM1033.

If in doubt as to whether a fan has an open-collector or totem pole TACH output, use one of the input signal conditioning circuits shown in the Fan Inputs section.

When designing drive circuits with transistors and FETs, make sure that the drive pins are not required to source current and that they sink less than the maximum current specified here.

Synchronous Speed Control

The ADM1033 drives the fan by using a control scheme called synchronous speed control. In this scheme, the PWM drive signal applied to the fan is synchronized with the TACH signal. Accurate and repeatable fan speed measurements are the main benefits. The fan is allowed to run reliably at speeds as low as 30 percent of the full capability.

The drive signal applied to the fan is synchronized with the TACH signal. The ADM1033 switches on the drive

signal and waits for a transition on the TACH signal. When a transition takes place on the TACH signal, the PWM drive is switched off for a period of time called t_{off}. The drive signal is then switched on again. The toff time is varied in order to vary the fan speed. If the fan is running too fast, the toff time is increased. If the fan is running too slow, the toff time is decreased.

Since the drive signal is synchronized with the TACH signal, the frequency with which the fan is driven depends on the current speed of the fan and the number of poles in it.

Figure 34 shows how the synchronous speed drive signal works. The ideal TACH signal is the TACH signal that would be output from the fan if power were applied 100 percent of the time. It is representative of the actual speed of the fan. The actual TACH signal is the signal the user would see on the TACH output from the fan if the user were to put a scope on it. In effect, the actual TACH signal is the ideal TACH signal chopped with the drive signal.

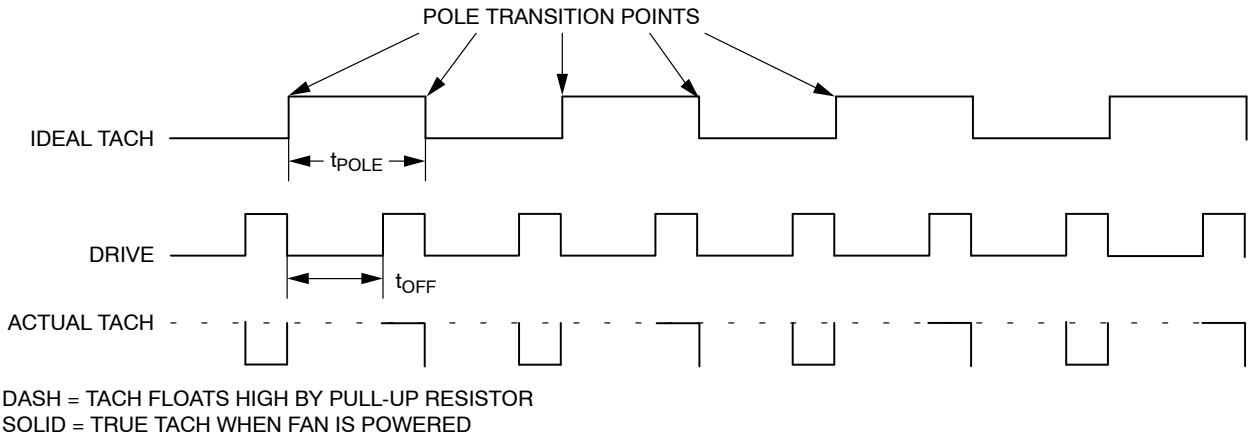


Figure 34. Drive Signal by Using Synchronous Control

Fan Inputs

Pin 2 is a TACH input intended for fan speed measurement. This input is open-drain.

Signal conditioning on the ADM1033 accommodates the slow rise and fall time of typical tachometer outputs. The maximum input signal range is from 0 V to 5.0 V, even when V_{CC} is less than 5.0 V. In the event that these inputs are supplied from fan outputs exceeding 0 V to 5.0 V, either resistive attenuation of the fan signal or diode clamping must be used to keep the fan inputs within an acceptable range.

Figure 35 to Figure 38 show examples of possible fan input circuits. If the fan TACH has a resistive pullup to V_{CC}, it can be connected directly to the fan output.

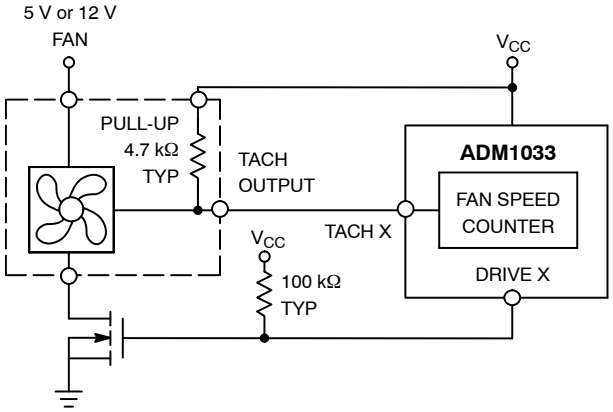


Figure 35. Fan with TACH Pullup to +V_{CC}

If the fan output has a resistive pullup to 12 V (or another voltage greater than 5.0 V), the fan output can be clamped with a Zener diode, as shown in Figure 36. The Zener voltage should be chosen so that it is greater than V_{IH} but less than 5.0 V. Allowing for the voltage tolerance of the Zener, a value of between 3.0 V and 5.0 V is suitable.

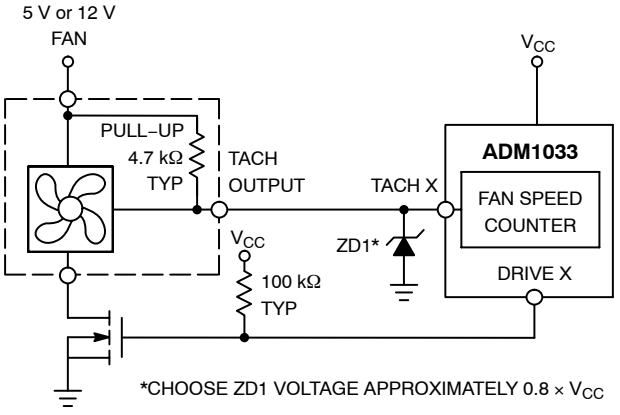


Figure 36. Fan with TACH Pullup to Voltage > 5.0 V, Clamped with Zener Diode

If the fan has a strong pullup (less than 1 kΩ to +12 V) or a totem-pole output, a series resistor can be added to limit the Zener current, as shown in Figure 37. Alternatively, a resistive attenuator may be used, as shown in Figure 38.

R1 and R2 should be chosen such that

$$2\ V < V_{PULLUP} \times R2/(R_{PULLUP} + R1 + R2) < 5\ V \quad \text{(eq. 3)}$$

The fan inputs have an input resistance of nominally 160 kΩ to ground. This should be taken into account when calculating resistor values.

With a pullup voltage of 12 V and pullup resistor less than 1 kΩ, suitable values for R1 and R2 would be 100 kΩ and 47 kΩ. This gives a high input voltage of 3.83 V.

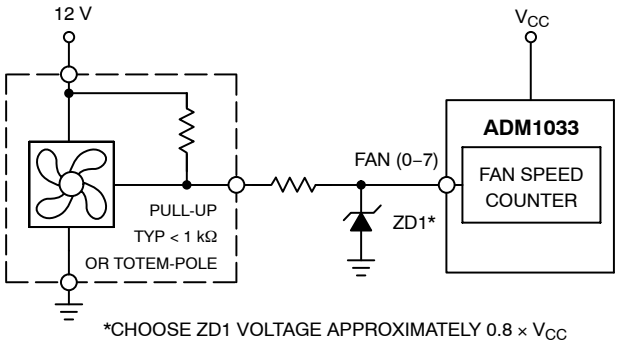


Figure 37. Fan with Strong TACH. Pullup to >V_{CC} or Totem-Pole Output, Clamped with Zener and Resistor

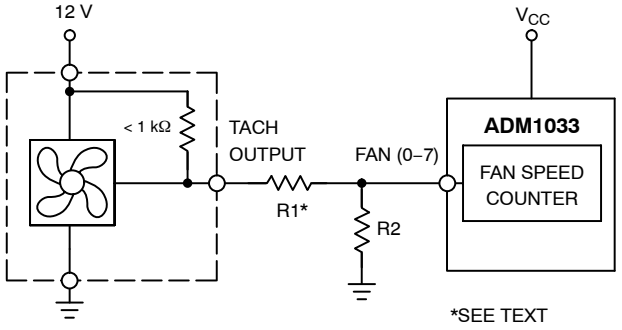


Figure 38. Fan with Strong TACH. Pullup to >V_{CC} or Totem-Pole Output, Attenuated with R1/R2

Fan Speed Measurement

The fan counter does not count the fan TACH output pulses directly. This is because the fan may be spinning at less than 1000 rpm and it would take several seconds to accumulate a reasonably large and accurate count. Instead, the period of the fan revolution is measured by gating an on-chip 81.92 kHz oscillator into the input of a 16-bit counter for one complete revolution of the fan. Therefore, the accumulated count is actually proportional to the fan tachometer period and inversely proportional to the fan speed.

The number of poles in the fan must be programmed in Configuration Register 3 (Address 0x03). Bits <3:0> set the number of poles for Fan 1, and Bits <7:4> set the number of poles for Fan 2. This number must be an even number only, because there cannot be an uneven number of poles in a fan. A TACH period is output for every two poles. Therefore, the number of poles must be known so that the ADM1033 can measure for a full revolution.

Figure 39 shows the fan speed measurement period, assuming that the fan outputs an ideal TACH signal. In reality, the TACH signal output by the fan is chopped by the drive signal. However, since the drive and the TACH signal are synchronized, there is enough information available for the ADM1033 to measure the fan speed accurately.

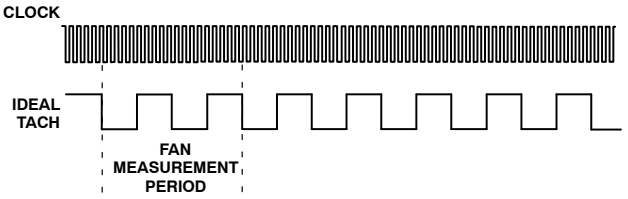


Figure 39. Fan Speed Measurement for a 4-pole Fan

Fan Speed Measurement Registers

These 16-bit measurements are stored in the TACH value registers.

Table 26. TACH VALUE REGISTERS

Register	Description	Default
0x4A	TACH1 Period, LSB	0xFF
0x4B	TACH1 Period, MSB	0xFF
0x4C	TACH2 Period, LSB	0xFF
0x4D	TACH2 Period, MSB	0xFF

Reading Fan Speed

Reading back fan speeds involves a 2-register read for each measurement. The low byte should be read first. This freezes the high byte until both high and low byte registers have been read, preventing erroneous fan speed measurement readings.

The fan tachometer reading registers report back the number of 12.2 µs period clocks (81.92 kHz oscillator) gated to the fan speed counter, for one full rotation of the fan, assuming the correct number of poles is programmed. Since the ADM1033 essentially measures the fan TACH period, the higher the count value, the slower the actual fan speed. A 16-bit fan TACH reading of 0xFFFF indicates that the fan has stalled or is running very slowly (< 75 rpm).

Calculating Fan Speed

Fan speed in rpm is calculated as follows. This assumes that the number of poles programmed in the Configuration Register 3 (Address 0x03) is correct for both fans.

Fan Speed (RPM) = (81920 × 60)/Fan TACH Reading

where:

Fan TACH Reading = 16-bit Fan TACHometer Reading

Example:

TACH1 High Byte (Reg. 0x4A) = 0x17
TACH1 Low Byte (Reg. 0x4B) = 0xFF

What is Fan 1 speed in rpm?

Fan 1 TACH Reading = 0x17FF = 6143d
RPM = (f × 60) / Fan 1 TACH reading

RPM = (81920 × 60) / 6143
Fan Speed = 800 RPM

Alarm Speed

The fan ALARM speed (Bit 6) in Status Register 3 (Address 0x51) is set whenever the fan runs at alarm speed. This occurs if the device is programmed to run the fan at full speed whenever the $\overline{\text{THERM}}$ temperature limits are exceeded. The device runs at alarm speed, for example, if the Boost Disable bit (Bit 1) of the Configuration 2 Register (Address 0x02) is not set to 1.

Fan Response Register

The ADM1033 fan speed controller operates by reading the current fan speed, comparing it with the programmed fan speed, and then updating the drive signal applied to the fan. The rate at which the ADM1033 looks at and updates the drive signal is determined by the fan response register. Different fans have different inertias and respond to a changing drive signal more or less quickly than others. The fan response register allows the user to tailor the ADM1033 to a particular fan to prevent situations like overshoot.

The user programs the number of updates the ADM1033 can make to the drive signal per second. Table 27 lists the available options.

Table 27. FAN RESPONSE CODES

Code	Update Rate
000	1.25 Updates/Second
001	2.5 Updates/Second = Default
010	5 Updates/Second
011	10 Updates/Second
100	20 Updates/Second
101	40 Updates/Second
110	80 Updates/Second
111	160 Updates/Second

Table 28. CONVERSION RATES

Bit #	Function
7	Reserved
<6:4>	Reserved
3	Reserved
<2:0>	Fan 1 Response

Look-up Table: Modes of Operation

The ADM1033 look-up table has two different modes of operation used to determine the behavior of the system:

- Manual Mode
- Look-up Table

Manual Mode

In manual mode, the ADM1033 is under software control. The software can program the required fan speed value or the target fan speed to the ADM1033, which then outputs that fan speed.

Programming Target Fan Speed

In this mode, the user programs the target fan speed as a TACH count for N poles or a TACH count for one full rotation of the fan, assuming the number of poles is programmed correctly in the Configuration 3 Register (Address 0x03).

Use the following steps to program the target fan speed:

1. Place the ADM1033 into manual mode. Set Bit 7 (Table/SW) of Configuration Register 1 (Address 0x01) = 0.
2. Program the target TACH count (fan speed) using the following equation:

$$\text{TACH Count} = (f \times 60)/R \qquad \text{(eq. 4)}$$

where:
f = clock frequency = 81.92 kHz
R = required RPM value

Example 1: If the desired speed for Fan 1 is 5000 rpm, program the following value to the TACH count registers:

TACH Count = (f × 60)/5000
TACH Count = 983d = 0x03D7

Example 2: If the desired speed for Fan 2 is 3500 rpm, program the following value to the TACH pulse period registers:

TACH Count = (f × 60)/3500
TACH Count = 1404d = 0x057C

Table 29. REGISTERS TO BE PROGRAMMED

Fan	Description	Address	Value
Fan 1	Look-up Table FS1, LSB	0x2A	0xD7
Fan 1	Look-up Table FS1, MSB	0x2B	0x03

Look-up Table

The ADM1033 allows the user to program a temperature-to-fan speed profile. There are 24 registers in the look-up table; 8 for temperature and 16 for target fan speed (each target fan speed is two registers). In total, there are eight available points.

There are two options when programming the look-up table. The ADM1033 can be programmed to make the fan speed run at discrete speeds and jump to the new fan speed once the temperature threshold is crossed. Or, it can linearly ramp the TACH count between the two temperature thresholds.

Figure 40 and Figure 41 show what the look-up table looks like if all eight points are used on the one curve.

Figure 40 shows the transfer curve when the fan is programmed to run at discrete speeds. The ADM1033 spins the fan at its new speed once a threshold is crossed.

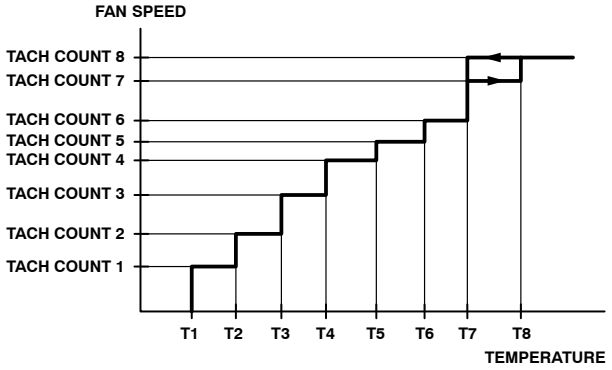


Figure 40. Programming the Look-up Table in Discreet Fan Speeds Mode

Figure 41 shows the transfer curve if the Linear Fan Speeds option is chosen. At temperature T1, the fan runs at Fan Speed 1. As the temperature increases, the fan speed increases until it reaches Fan Speed 2 at T2.

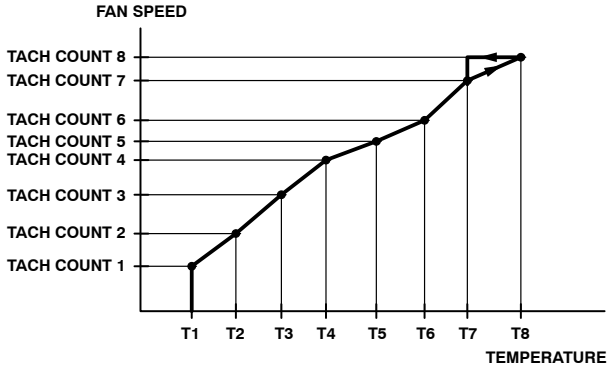


Figure 41. Programming the Look-up Table in Linear Fan Speeds Mode

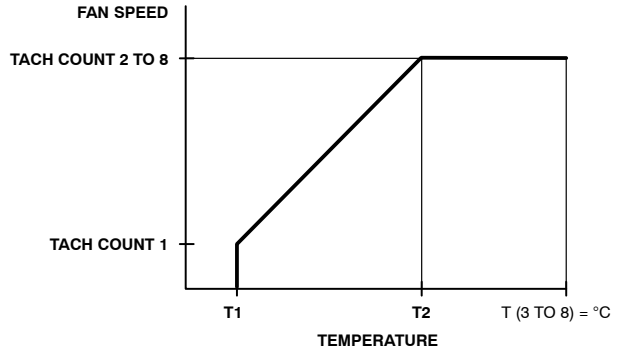


Figure 42. Programming Two Points on the Look-up Table

Once the temperature exceeds the highest temperature point in the look-up table, the fan speed remains at the highest speed until the temperature drops below the T7 temperature value. When the look-up table is split in two, the same applies.

If the temperatures in T1 to T8 are not programmed in succession, the fan speed moves to the next highest programmed temperature as the temperature increases. Similarly, when the temperature decreases, it ignores programmed higher temperatures and jumps to the next lower temperature. Therefore, the temperature-to-fan speed profile for increasing and decreasing temperature can be different.

When programming the look-up table, the user has the option to use between two and eight points for the fan. If the user just wants to program a transfer curve (and knows the starting temperature, minimum speed, maximum temperature, and maximum speed), then all the user needs to program are four parameters: T1, T2, FS1, and FS2. The remainder of the look-up temperature thresholds should remain at their default values of +191°C. If required, the FS3 should be programmed with the same value as FS2 to give the flat curve, if required. Or, the fan speeds can be left at the default value of 0. However, it is normal to program a $\overline{\text{THERM}}$ limit as well. Once this temperature is exceeded and the boost bit is set, the fan run to full speed. This overrides the look-up table.

Table 30. LOOK-UP TABLE REGISTER ADDRESS

x	Temperature, x	FSx, LSB	FSx, MSB
1	0x22	0x2A	0x2B
2	0x23	0x2C	0x2D
3	0x24	0x2E	0x2F
4	0x25	0x30	0x31
5	0x26	0x32	0x33
6	0x27	0x34	0x35
7	0x28	0x36	0x37
8	0x29	0x38	0x39

Setting Up the Look-up Table in Linear Mode

When Discrete/Linear Speed (Bit 2) is set to 1 (default), the TACH count decreases linearly (and therefore the fan speed increases) with temperature.

Example: At temperature T_X, the fans run at FS_X and fan speed increases with temperature to FS_{X+1} at temperature T_{X+1}.

Alternatively, the fan can be run at discrete fan speeds. When Discrete/Linear Speed (Bit 2) is set to 0, the fan runs at a new speed once the temperature threshold is exceeded.

Setting Which Temperature Channel Controls a Fan

Fan Behavior Register (Address 0x07)

Bits <1:0> = DRIVE1 Behavior (D1B)

The ADM1033 can be configured so that Fan 1 can be controlled by either the local temperature, or by the Remote 1 temperatures.

Table 31. DRIVE BHVR BITS

Bits	DRIVE x BHVR
00	Local Temperature Controls Fan
01	Remote 1 Temperature Controls Fan
10	Reserved
11	Fan Runs at Full Speed

Look-up Table Hysteresis

The user can program a hysteresis to be applied to the look-up table. The advantage of this is apparent if the temperature is cycling around one of the threshold temperatures and causing the fan speed to switch between the two speeds, particularly when the look-up table is configured in discrete mode. It would not be as important in the linear mode.

Table 32. PROGRAMMING THE HYSTERESIS

Code	Hysteresis Value
0000 0000	0°C
0000 0001	1°C
0000 0010	2°C
0000 0101	5°C
0000 1000	8°C
0000 1111	15°C

The look-up table's hysteresis register is at Address 0x3A. A hysteresis value of between 0°C and 15°C can be programmed with a resolution of 1°C and applied to all the temperature thresholds. Table 32 gives examples of values for programming.

Programming the $\overline{\text{THERM}}$ Limit for Temperature Channels

$\overline{\text{THERM}}$ is the absolute maximum temperature allowed on a temperature channel. Above this temperature, a component such as the CPU or VRM may be operating beyond its safe operating limit. When the temperature measured exceeds $\overline{\text{THERM}}$, all fans are driven at full speed to provide critical system cooling. The fans remain running at full speed until the temperature drops below $\overline{\text{THERM}}$ – Hysteresis. The hysteresis value is programmable; its default is 5°C. If the Boost Disable bit (Bit 1) is set in Configuration Register 2, the fan do not run to full speed.

The $\overline{\text{THERM}}$ limit is considered the maximum worst-case operating temperature of the system. Exceeding any $\overline{\text{THERM}}$ limit runs the fan at full speed, a condition with very negative acoustic effects. This limit should be set up as a fail-safe and not exceeded under normal system operating conditions. The $\overline{\text{THERM}}$ temperature limit registers are listed in Table 33.

Table 33. $\overline{\text{THERM}}$ HYSTERESIS REGISTERS

Address	Description	Default
0x0D	Local $\overline{\text{THERM}}$ Limit	0x95 (85°C)
0x10	Remote 1 $\overline{\text{THERM}}$ Limit	0x95 (85°C)

The $\overline{\text{THERM}}$ hysteresis register is at Address 0x1A. A hysteresis value is programmed and applied to all two temperature channels; Local and Remote 1. A $\overline{\text{THERM}}$ hysteresis value of between 0°C and 15°C can be programmed with a resolution of 1°C. Table 33 gives some examples.

Table 34. PROGRAMMING $\overline{\text{THERM}}$ HYSTERESIS

Code	Hysteresis Value
0000 0000	0°C
0000 0001	1°C
0000 0010	2°C
0000 0101	5°C = Default
0000 1000	8°C
0000 1111	15°C

XOR Tree Test Mode

The ADM1033 includes an XOR tree test mode. This is useful for in circuit test equipment at board level testing. By applying stimulus to the pins included in the XOR test, it is possible to detect opens or shorts on the system board. Figure 43 shows the signals that are exercised in the XOR tree test mode. The XOR tree test is enabled by setting the XOR bit (Bit 3) in Configuration 4 Register (0x04).

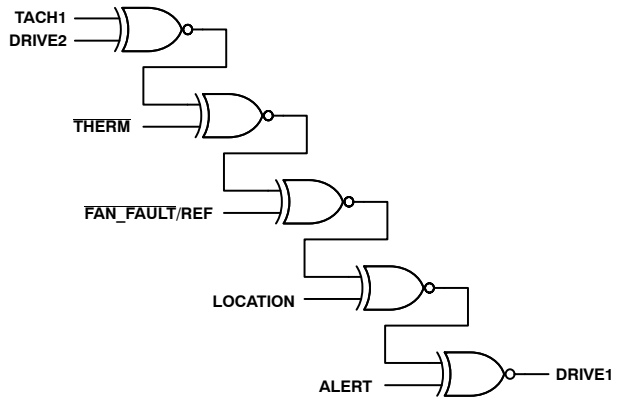


Figure 43. XOR Tree Test

Lock Bit

Setting the Lock bit (Bit 6) of Configuration Register 1 (Address 0x01) makes all the lockable registers read-only. These registers remain read-only until the ADM1033 is powered down and back up again. For more information on which registers are lockable, see Table 35.

SW Reset

Setting the Software Reset bit (Bit 0) of Configuration Register 2 (Address 0x02) resets all software resettable bits to their default value. For more information on resetting registers and their default values, see Table 35 to Table 69.