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

ON Semiconductor®

*dB*COOL™ Remote Thermal Controller and Voltage Monitor

ADM1027*

FEATURES

Monitors up to 5 Supply Voltages
Controls and Monitors up to 4 Fan Speeds
1 On-Chip and 2 Remote Temperature Sensors
Monitors up to 5 Processor VID Bits
Automatic Fan Speed Control Mode Controls System Cooling Based on Measured Temperature
Enhanced Acoustic Mode Dramatically Reduces User Perception of Changing Fan Speeds
2-Wire and 3-Wire Fan Speed Measurement
Limit Comparison of All Monitored Values
Meets SMBus 2.0 Electrical Specifications (Fully SMBus 1.1 Compliant)

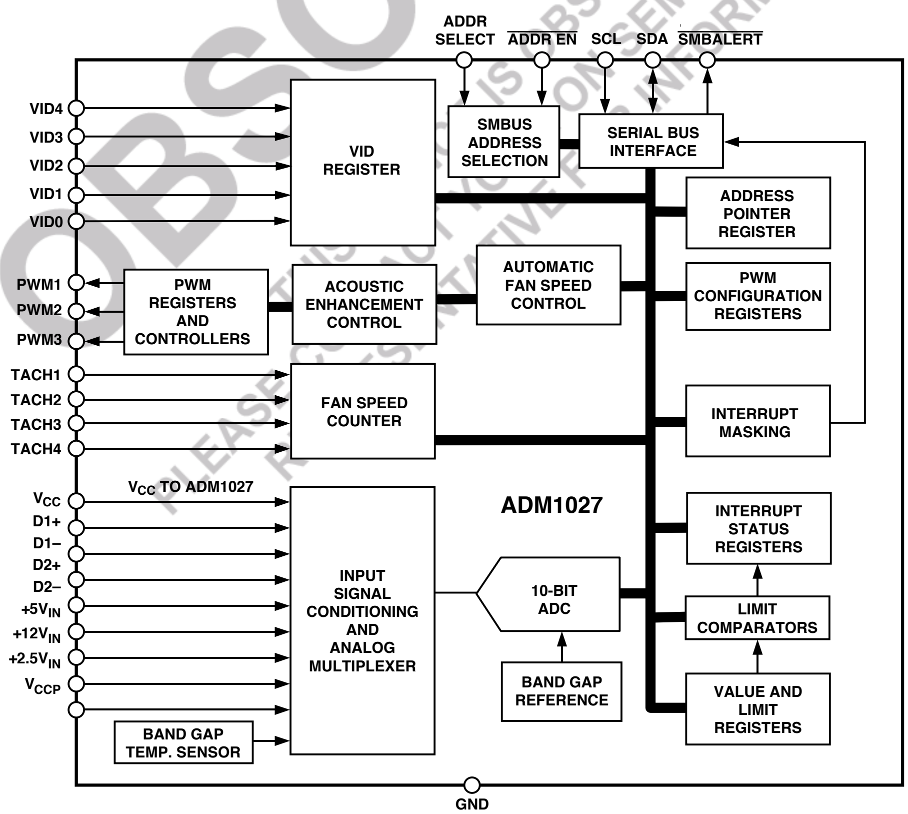
APPLICATIONS

Low Acoustic Noise PCs
Networking and Telecommunications Equipment

GENERAL DESCRIPTION

The ADM1027 *dB*COOL controller is a complete systems monitor and multiple PWM fan controller for noise sensitive applications requiring active system cooling. It can monitor 12 V, 5 V, 2.5 V CPU supply voltage, plus its own supply voltage. It can monitor the temperature of up to two remote sensor diodes, plus its own internal temperature. It can measure and control the speed of up to four fans so that they operate at the lowest possible speed for minimum acoustic noise. The automatic fan speed control loop optimizes fan speed for a given temperature. Once the control loop parameters are programmed, the ADM1027 can vary fan speed without CPU intervention.

FUNCTIONAL BLOCK DIAGRAM



*Protected by U.S. Patent Nos. 6,188,189; 6,169,442; 6,097,239; 5,982,221; and 5,867,012. Other patents pending.

May 2010 - Rev. 3

Publication Order Number:
ADM1027/D

ADM1027–SPECIFICATIONS[1, 2, 3, 4] ($T_A = T_{MIN}$ to T_{MAX} (0°C to 105°C), $V_{CC} = V_{MIN}$ to V_{MAX} (3 V to 5.5 V), unless otherwise noted.)

Parameter	Min	Typ	Max	Unit	Test Conditions/Comments
POWER SUPPLY					
Supply Voltage	3.0	3.3	5.5	V	
Supply Current, I_{CC}		1.4	3	mA	Interface Inactive, ADC Active
TEMP-TO-DIGITAL CONVERTER					
Local Sensor Accuracy			±3	°C	$0°C \le T_A \le 105°C$
			±2	°C	$0°C \le T_A \le 70°C$
		±1		°C	$T_A = 40°C$
Resolution		0.25		°C	
Remote Diode Sensor Accuracy			±3	°C	$0°C \le T_D \le 120°C$
			±1.5	°C	$0°C \le T_D \le 120°C$; $0°C \le T_A \le 70°C$
		±1		°C	$T_A = 40°C$
				°C	$0°C \le T_D \le 120°C$; $T_A = 40°C$
Resolution		0.25		°C	
Remote Sensor Source Current		200		µA	High Level
		12		µA	Low Level
ANALOG-TO-DIGITAL CONVERTER (INCLUDING MUX AND ATTENUATORS)					
Total Unadjusted Error, TUE		±0.5	±1	%	All ADC Inputs except 12 V
			±1.5	%	12 V Input
Differential Nonlinearity, DNL			±1	LSB	8 Bits
Power Supply Sensitivity		±0.1		%/V	
Conversion Time (Voltage Input)		11.38	12.29	ms	Averaging Enabled
Conversion Time (Local Temperature)		12.09	13.05	ms	Averaging Enabled
Conversion Time (Remote Temperature)		25.59	27.64	ms	Averaging Enabled
Total Monitoring Cycle Time		120.17	129.78	ms	Averaging Enabled
Total Monitoring Cycle Time		13.51	14.59	ms	Averaging Disabled
Input Resistance	80	140	250	kΩ	
FAN RPM-TO-DIGITAL CONVERTER					
Accuracy			±6	%	$0°C \le T_A \le 70°C$
			±8	%	$3.0\ V \le V_{CC} \le 3.6\ V$
Full-Scale Count			65,535		
Nominal Input RPM		109		RPM	Fan Count = 0xBFFF
		329		RPM	Fan Count = 0x3FFF
		5,000		RPM	Fan Count = 0x0438
		10,000		RPM	Fan Count = 0x021C
Internal Clock Frequency	82.8	90	97.2	kHz	
OPEN-DRAIN DIGITAL OUTPUTS, PWM1–PWM3, XTO					
Current Sink, I_{OL}			8.0	mA	
Output Low Voltage, V_{OL}			0.4	V	$I_{OUT} = -8.0$ mA, $V_{CC} = 3.3$ V
High Level Output Current, I_{OH}		0.1	1	µA	$V_{OUT} = V_{CC}$

Parameter	Min	Typ	Max	Unit	Test Conditions/Comment
OPEN-DRAIN SERIAL DATA BUS OUTPUT (SDA)					
Output Low Voltage, V_{OL}			0.4	V	I_{OUT} = –4.0 mA, V_{CC} = 3.3 V
High Level Output Current, I_{OH}		0.1	1	µA	$V_{OUT} = V_{CC}$
SMBUS DIGITAL INPUTS (SCL, SDA)					
Input High Voltage, V_{IH}	2.0			V	
Input Low Voltage, V_{IL}			0.4	V	
Hysteresis		500		mV	
DIGITAL INPUT LOGIC LEVELS (VID0–4)					
Input High Voltage, V_{IH}	1.7			V	
Input Low Voltage, V_{IL}			0.8	V	
DIGITAL INPUT LOGIC LEVELS (TACH INPUTS)					
Input High Voltage, V_{IH}	2.0			V	
			5.5	V	Maximum Input Voltage
Input Low Voltage, V_{IL}			0.8	V	
	–0.3			V	Minimum Input Voltage
Hysteresis		0.5		V p-p	
DIGITAL INPUT CURRENT					
Input High Current, I_{IH}	–1			µA	$V_{IN} = V_{CC}$
Input Low Current, I_{IL}			1	µA	V_{IN} = 0
Input Capacitance, C_{IN}		5		pF	
SERIAL BUS TIMING					
Clock Frequency, f_{SCLK}	10		100	kHz	See Figure 1
Glitch Immunity, t_{SW}			50	ns	See Figure 1
Bus Free Time, t_{BUF}	4.7			µs	See Figure 1
Start Setup Time, $t_{SU;STA}$	4.7			µs	See Figure 1
Start Hold Time, $t_{HD;STA}$	4.0			µs	See Figure 1
SCL Low Time, t_{LOW}	4.7			µs	See Figure 1
SCL High Time, t_{HIGH}	4.0		50	µs	See Figure 1
SCL, SDA Rise Time, t_r			1000	ns	See Figure 1
SCL, SDA Fall Time, t_f			300	µs	See Figure 1
Data Setup Time, $t_{SU;DAT}$	250			ns	See Figure 1
Data Hold Time, $t_{HD;DAT}$	300			ns	See Figure 1
Detect Clock Low Timeout, $t_{TIMEOUT}$	15		35	ms	Can Be Optionally Disabled

NOTES
[1]All voltages are measured with respect to GND, unless otherwise specified.
[2]Typicals are at T_A = 40°C and represent the most likely parametric norm.
[3]Logic inputs will accept input high voltages up to V_{MAX} even when the device is operating down to V_{MIN}.
[4]Timing specifications are tested at logic levels of V_{IL} = 0.8 V for a falling edge and V_{IH} = 2.0 V for a rising edge.

Specifications subject to change without notice.

ABSOLUTE MAXIMUM RATINGS*

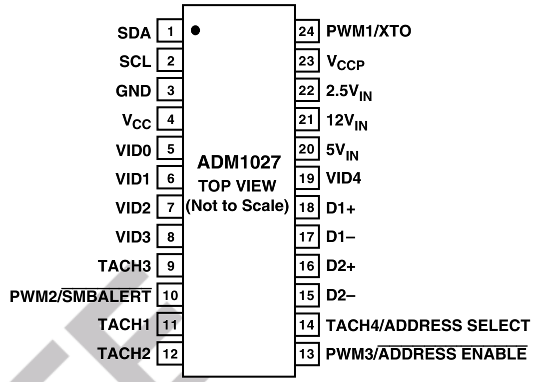
Positive Supply Voltage (V_{CC}) . . . . 6.5 V
Voltage on 12 V_{IN} Pin . . . . 20 V
Voltage on Any Other Input or Output Pin . . . . –0.3 V to +6.5 V
Input Current at Any Pin . . . . ±5 mA
Package Input Current . . . . ±20 mA
Maximum Junction Temperature ($T_{J\ MAX}$) . . . . 150°C
Storage Temperature Range . . . . –65°C to +150°C
Lead Temperature, Soldering
  Vapor Phase (60 sec) . . . . 215°C
  Infrared (15 sec) . . . . 200°C
ESD Rating . . . . 2000 V

*Stresses above those listed under Absolute Maximum Ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

THERMAL CHARACTERISTICS

24-Lead QSOP Package:
θ_{JA} = 123°C/W, θ_{JC} = 27°C/W

PIN CONFIGURATION



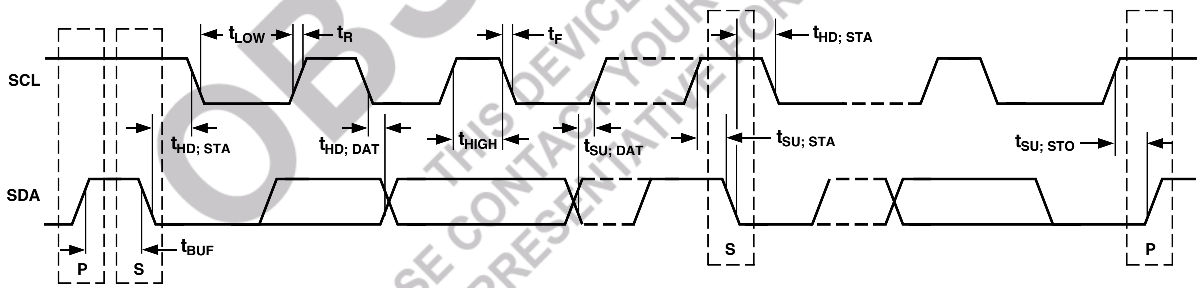


Figure 1. Diagram for Serial Bus Timing

CAUTION

ESD (electrostatic discharge) sensitive device. Electrostatic charges as high as 4000 V readily accumulate on the human body and test equipment and can discharge without detection. Although the ADM1027 features proprietary ESD protection circuitry, permanent damage may occur on devices subjected to high energy electrostatic discharges. Therefore, proper ESD precautions are recommended to avoid performance degradation or loss of functionality.

PIN FUNCTION DESCRIPTIONS

Pin	Mnemonic	Description
1	SDA	Digital I/O (Open-Drain). SMBus bidirectional serial data. Requires SMBus pull-up.
2	SCL	Digital Input (Open-Drain). SMBus serial clock input. Requires SMBus pull-up.
3	GND	Ground Pin for the ADM1027.
4	V_{CC}	Power Supply. Can be powered by 3.3 V standby if monitoring in low power states is required. V_{CC} is also monitored through this pin. The ADM1027 can also be powered from a 5 V supply. Setting Bit 7 of Configuration Register 1 (Reg. 0x40) rescales the V_{CC} input attenuators to correctly measure a 5 V supply.
5	VID0	Digital Input (Open-Drain). Voltage supply readouts from CPU. This value is read into the VID register (Reg. 0x43).
6	VID1	Digital Input (Open-Drain). Voltage supply readouts from CPU. This value is read into the VID register (Reg. 0x43).
7	VID2	Digital Input (Open-Drain). Voltage supply readouts from CPU. This value is read into the VID register (Reg. 0x43).
8	VID3	Digital Input (Open-Drain). Voltage supply readouts from CPU. This value is read into the VID register (Reg. 0x43).
9	TACH3	Digital Input (Open-Drain). Fan tachometer input to measure speed of Fan 3. Can be reconfigured as an analog input (AIN3) to measure the speed of 2-wire fans.
10	PWM2/$\overline{\text{SMBALERT}}$	Digital Output (Open-Drain). Requires 10 kΩ typical pull-up. Pulsewidth modulated output to control Fan 2 speed. This pin may be reconfigured as an $\overline{\text{SMBALERT}}$ interrupt output to signal out-of-limit conditions.
11	TACH1	Digital Input (Open-Drain). Fan tachometer input to measure speed of Fan 1. Can be reconfigured as an analog input (AIN1) to measure the speed of 2-wire fans.
12	TACH2	Digital Input (Open-Drain). Fan tachometer input to measure speed of Fan 2. Can be reconfigured as an analog input (AIN2) to measure the speed of 2-wire fans.
13	PWM3/$\overline{\text{ADDRESS ENABLE}}$	Digital I/O (Open-Drain). Pulsewidth modulated output to control Fan 3 speed. Requires 10 kΩ typical pull-up. If pulled low on power-up, this places the ADM1027 into address select mode, and the state of Pin 14 will determine the ADM1027's slave address.
14	TACH4/ADDRESS SELECT	Digital Input (Open-Drain). Fan tachometer input to measure speed of Fan 4. Can be reconfigured as an analog input (AIN4) to measure the speed of 2-wire fans. If in address select mode, this pin determines the SMBus device address.
15	D2–	Cathode Connection to Second Thermal Diode.
16	D2+	Anode Connection to Second Thermal Diode.
17	D1–	Cathode Connection to First Thermal Diode.
18	D1+	Anode Connection to First Thermal Diode.
19	VID4	Digital Input (Open-Drain). Voltage supply readouts from CPU. This value is read into the VID register (Reg. 0x43).
20	$5V_{IN}$	Analog Input. Monitors 5 V power supply.
21	$12V_{IN}$	Analog Input. Monitors 12 V power supply.
22	$2.5V_{IN}$	Analog Input. Monitors 2.5 V supply, typically a chipset voltage.
23	V_{CCP}	Analog Input. Monitors processor core voltage (0 V to 3 V).
24	PWM1/XTO	Digital Output (Open-Drain). Pulsewidth modulated output to control Fan 1 speed. Requires 10 kΩ typical pull-up. Also functions as the output from the XOR tree in XOR test mode.

FUNCTIONAL DESCRIPTION

General Description

The ADM1027 is a complete systems monitor and multiple fan controller for any system requiring monitoring and cooling. The device communicates with the system via a serial system management bus. The serial bus controller has an optional address line for device selection (Pin 14), a serial data line for reading and writing addresses and data (Pin 1), and an input line for the serial clock (Pin 2). All control and programming functions of the ADM1027 are performed over the serial bus. In addition, one of the pins can be reconfigured as an $\overline{\text{SMBALERT}}$ output to indicate out-of-limit conditions.

Measurement Inputs

The device has six measurement inputs, four for voltage and two for temperature. It can also measure its own supply voltage and can measure ambient temperature with its on-chip temperature sensor.

Pins 20 to 23 are analog inputs with on-chip attenuators, configured to monitor 5 V, 12 V, 2.5 V, and the processor core voltage (2.25 V input), respectively.

Power is supplied to the chip via Pin 4, which the system also uses to monitor V_{CC}. In PCs, this pin is normally connected to a 3.3 V standby supply. This pin can, however, be connected to a 5 V supply and monitor it without overranging.

Remote temperature sensing is provided by the D1+/– and D2+/– inputs, to which diode-connected, external temperature-sensing transistors such as a 2N3906 or CPU thermal diode may be connected.

The ADC also accepts input from an on-chip band gap temperature sensor that monitors system ambient temperature.

Sequential Measurement

When the ADM1027 monitoring sequence is started, it cycles sequentially through the measurement of analog inputs and the temperature sensors. Measured values from these inputs are stored in value registers. These can be read out over the serial bus, or can be compared with programmed limits stored in the limit registers. The results of out-of-limit comparisons are stored in the status registers, which can be read over the serial bus to flag out-of-limit conditions.

Processor Voltage ID

Five digital inputs (VID0 to VID4 — Pins 5 to 8 and 19) read the processor Voltage ID code and store it in the VID register, from which it can be read out by the management system over the serial bus. The VID code monitoring function is compatible with both VRM9.x and future VRM10 solutions. The VID code monitoring function is compatible with VRM9.x.

ADM1027 Address Selection

Pin 13 is the dual function PWM3/$\overline{\text{ADDRESS ENABLE}}$ pin. If Pin 13 is pulled low on power-up, the ADM1027 will read the state of Pin 14 (TACH4/ADDRESS SELECT pin) to determine the ADM1027 slave address. If Pin 13 is high on power-up, then the ADM1027 will default to SMBus slave address 0x5C. This function is described later in more detail.

Internal Registers of the ADM1027

A brief description of the ADM1027's principal internal registers follows. More detailed information on the function of each register is given in Tables IV to XXXVI.

Configuration Registers

Provide control and configuration of the ADM1027, including alternate pinout functionality.

Address Pointer Register

Contains the address that selects one of the other internal registers. When writing to the ADM1027, the first byte of data is always a register address, which is written to the Address Pointer Register.

Status Registers

Provide the status of each limit comparison and are used to signal out-of-limit conditions on the temperature, voltage, or fan speed channels. If Pin 10 is configured as $\overline{\text{SMBALERT}}$, then this pin will assert low whenever a status bit gets set.

Interrupt Mask Registers

Allow each interrupt status event to be masked when Pin 10 is configured as an $\overline{\text{SMBALERT}}$ output. This affects only the $\overline{\text{SMBALERT}}$ output and not the bits in the status register.

VID Register

The status of the VID0 to VID4 pins of the processor can be read from this register.

Value and Limit Registers

The results of analog voltage inputs, temperature, and fan speed measurements are stored in these registers, along with their limit values.

Offset Registers

Allow each temperature channel reading to be offset by a twos complement value written to these registers.

T_{MIN} Registers

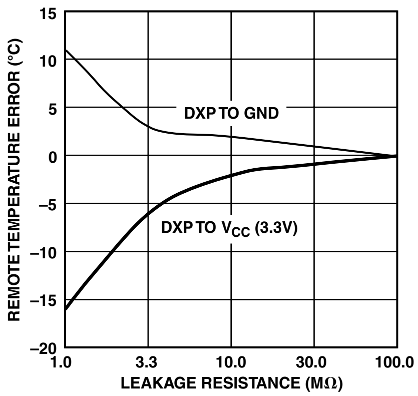
Program the starting temperature for each fan under automatic fan speed control.

T_{RANGE} Registers

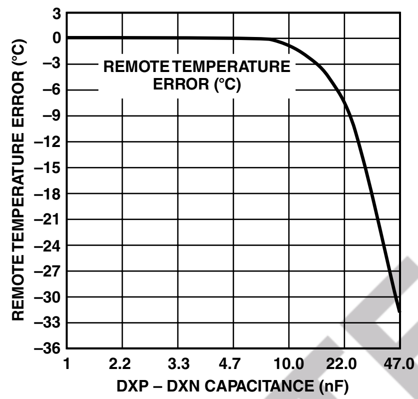
Program the temperature-to-fan speed control slope in automatic Fan Speed Control Mode for each PWM output.

Enhance Acoustics Registers

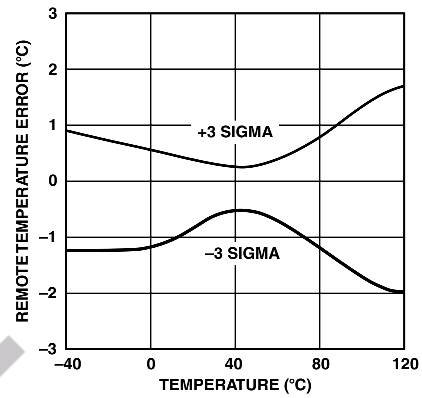
Allow each PWM output controlling fan to be tweaked to enhance the system's acoustics.



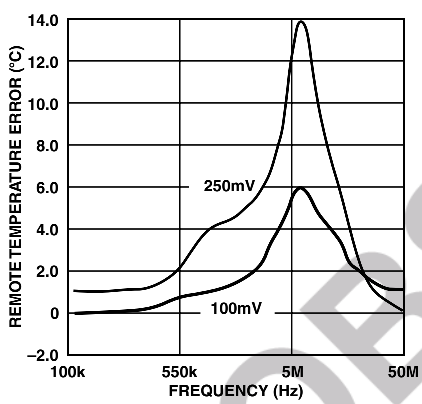
TPC 1. Remote Temperature Error vs. Leakage Resistance



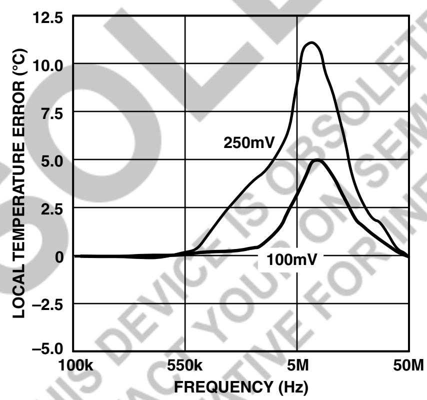
TPC 2. Remote Temperature Error vs. Capacitance between D+ and D–



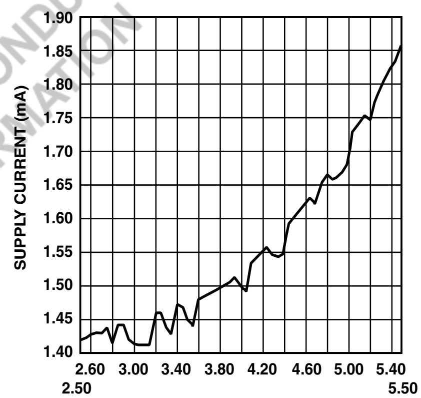
TPC 3. Remote Temperature Error vs. Actual Temperature



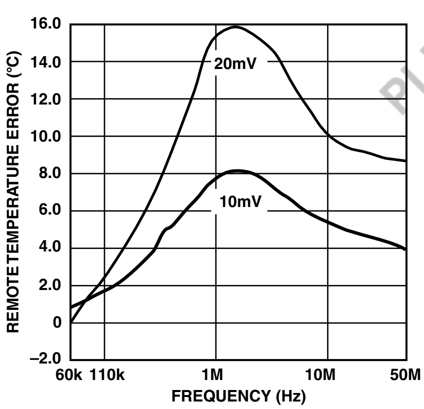
TPC 4. Remote Temperature Error vs. Power Supply Noise Frequency



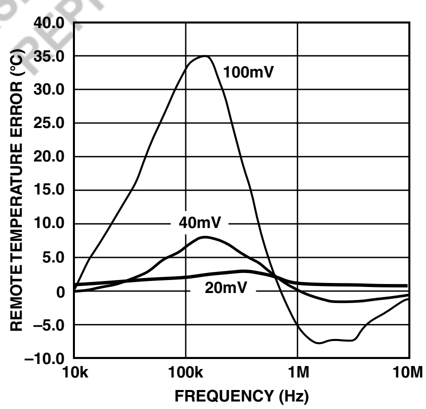
TPC 5. Local Temperature Error vs. Power Supply Noise Frequency



TPC 6. Supply Current vs. Supply Voltage



TPC 7. Remote Temperature Error vs. Differential Mode Noise Frequency



TPC 8. Remote Temperature Error vs. Common Mode Noise Frequency

SERIAL BUS INTERFACE

Control of the ADM1027 is carried out using the serial System Management Bus (SMBus). The ADM1027 is connected to this bus as a slave device, under the control of a master controller.

The ADM1027 has a 7-bit serial bus address. When the device is powered up with Pin 13 (PWM3/$\overline{\text{ADDRESS ENABLE}}$) high, the ADM1027 will have a default SMBus address of 0101110 or 0x5C. If more than one ADM1027 is to be used in a system, then each ADM1027 should be placed in address select mode by strapping Pin 13 low on power-up. The logic state of Pin 14 then determines the device's SMBus address.

Table I. ADM1027 Address Select Mode

Pin 13 State	Pin 14 State	Address
0	Low (10 kΩ to GND)	0101100 (0x58)
0	High (10 kΩ pull-up)	0101101 (0x5A)
1	Don't Care	0101110 (0x5C) (default)

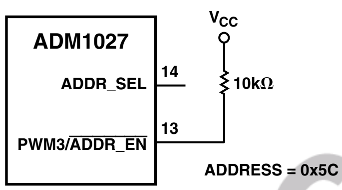


Figure 2. Default SMBus Address = 0x5C

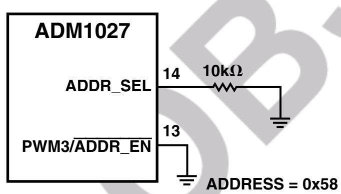


Figure 3. SMBus Address = 0x58 (Pin 14 = 0)

The device address is sampled and latched on the first valid SMBus transaction, so any attempted addressing changes made thereafter will have no immediate effect.

The facility to make hardwired changes to the SMBus slave address allows the user to avoid conflicts with other devices sharing the same serial bus (for example, if more than one ADM1027 is used in a system).

Once the SMBus address has been assigned, these pins return to their original function. However, since the circuits required to set up the SMBus address are unworkable with the PWM and TACH circuits, it would require the use of muxes to switch in and out the correct circuit at the correct time.

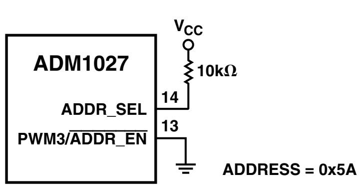


Figure 4. SMBus Address = 0x5A (Pin 14 = 1)

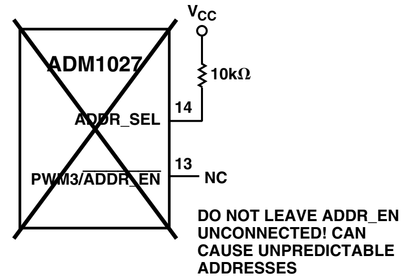


Figure 5. Unpredictable SMBus Address if Pin 13 is Unconnected

Care should be taken to ensure that Pin 13 (PWM3/$\overline{\text{ADDR\_EN}}$) is either tied high or low. Leaving Pin 13 floating could cause the ADM1027 to power up with an unexpected address.

Note that if the ADM1027 is placed into address select mode, Pins 13 and 14 can be used as their alternate functions once address assignment has taken place (PWM3, TACH4). Care should be taken using muxes to connect in the appropriate circuit at the appropriate time.

The serial bus protocol operates as follows:

1. 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 will follow. All slave peripherals connected to the serial bus respond to the start condition and shift in the next eight bits, consisting of a 7-bit address (MSB first) plus the R/$\overline{W}$ bit, which determines the direction of the data transfer, i.e., whether data will be written to or read from the slave device.

   The peripheral whose address corresponds to the transmitted address responds by pulling the data line low during the low period before the ninth clock pulse, known as the acknowledge bit. All other devices on the bus now 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 will write to the slave device. If the R/$\overline{W}$ bit is a 1, the master will read from the slave device.

2. Data is sent over the serial bus in sequences of nine clock pulses, eight 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, as 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 will pull the data line high during the 10th clock pulse to assert a stop condition. In read mode, the master device will override the acknowledge bit by pulling the data line high during the low period before the ninth clock pulse. This is known as No Acknowledge. The master will then take 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.

Any number of bytes of data can be transferred over the serial bus in one operation. However, it is not possible to mix read and write in one operation because the type of operation is determined at the beginning and subsequently cannot be changed without starting a new operation.

In the case of the ADM1027, write operations contain either one or two bytes, and read operations contain one byte and perform the following functions:

To write data to one of the device data registers or read data from it, the address pointer register must be set so 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 address pointer register. If data is to be written to the device, then the write operation contains a second data byte that is written to the register selected by the address pointer register.

This is illustrated in Figure 6. The device address is sent over the bus followed by R/$\overline{W}$ being 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 address pointer register. 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:

1. If the ADM1027 address pointer register value is unknown or not the desired value, it is first necessary to set it to the correct value before data can be read from the desired data register. This is done by performing a write to the ADM1027 as before, but only sending the data byte containing the register address, as data is not to be written to the register. This is shown in Figure 7.

   A read operation is then performed consisting of the serial bus address, R/$\overline{W}$ bit set to 1, followed by the data byte read from the data register. This is shown in Figure 8.

2. If the address pointer register is known to be already at the desired address, data can be read from the corresponding data register without first writing to the address pointer register, so Figure 7 can be omitted.

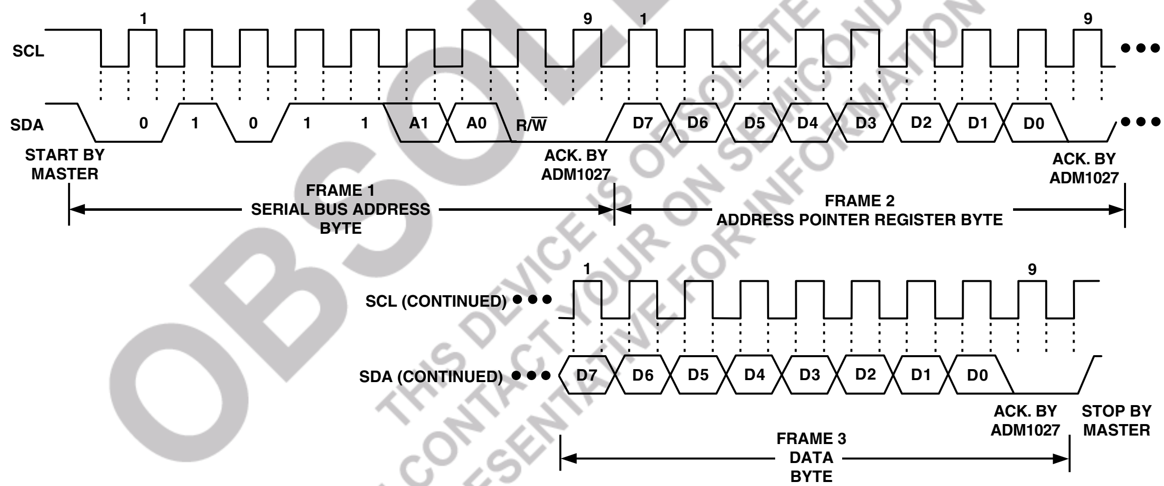


Figure 6. Writing a Register Address to the Address Pointer Register, Then Writing Data to the Selected Register

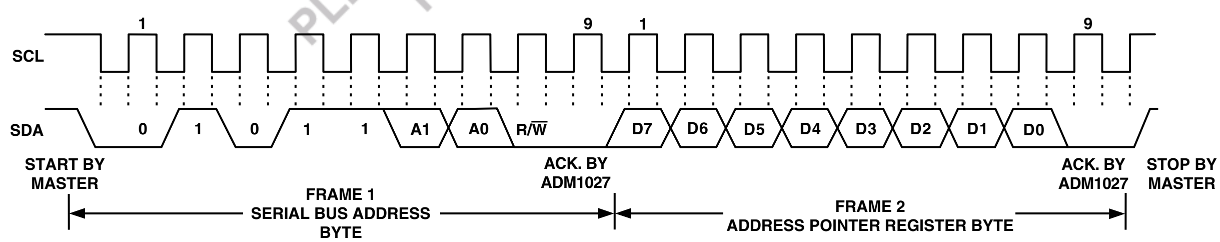
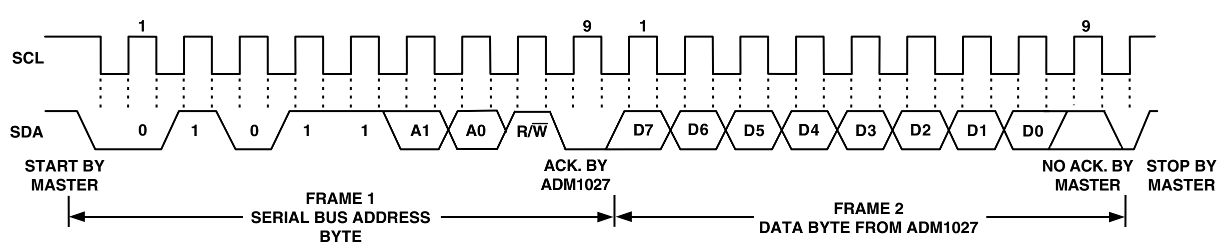


Figure 7. Writing to the Address Pointer Register Only



Notes

1. It is possible to *read* a data byte from a data register without first writing to the address pointer register if the address pointer register is already at the correct value. However, it is not possible to *write* data to a register without writing to the address pointer register, because the first data byte of a write is always written to the address pointer register.
2. In Figures 6 to 8, the serial bus address is shown as the default value 01011(A1)(A0), where A1 and A0 are set by the address select mode function previously defined.
3. In addition to supporting the send byte and receive byte protocols, the ADM1027 also supports the read byte protocol (see System Management Bus specifications Rev. 2.0 for more information).
4. If it is required to perform several read or write operations in succession, the master can send a repeat start condition instead of a stop condition to begin a new operation.

ADM1027 WRITE OPERATIONS

The SMBus specification defines several protocols for different types of read and write operations. The ones used in the ADM1027 are discussed below. The following abbreviations are used in the diagrams:

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

The ADM1027 uses the following SMBus write protocols:

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

For the ADM1027, the send byte protocol is used to write a register address to RAM for a subsequent single byte read from the same address. This is illustrated in Figure 9.

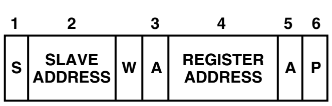


Figure 9. Setting a Register Address for Subsequent Read

If it is required to read data from the register immediately after setting up the address, 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 command byte and one data byte to the 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 write bit (low).
3. The addressed slave device asserts ACK on SDA.
4. The master sends a command code.
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.

This is illustrated in Figure 10.

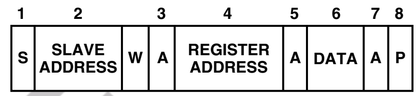


Figure 10. Single Byte Write to a Register

ADM1027 READ OPERATIONS

The ADM1027 uses the following SMBus read protocols:

Receive Byte

This is useful when repeatedly reading a single register. The register address needs to have been set up previously. 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 asserts NO ACK on SDA.
6. The master asserts a stop condition on SDA and the transaction ends.

In the ADM1027, the receive byte protocol is used to read a single byte of data from a register whose address has previously been set by a send byte or write byte operation.

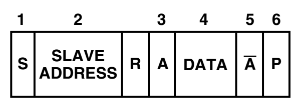


Figure 11. Single Byte Read from a Register

Alert Response Address

Alert Response Address (ARA) is a feature of SMBus devices, which allows an interrupting device to identify itself to the host when multiple devices exist on the same bus.

The $\overline{\text{SMBALERT}}$ output can be used as an interrupt output or can be used as an $\overline{\text{SMBALERT}}$. One or more outputs can be connected to a common $\overline{\text{SMBALERT}}$ line connected to the master. If a device's $\overline{\text{SMBALERT}}$ line goes low, the following procedure occurs:

1. $\overline{\text{SMBALERT}}$ is pulled low.
2. Master initiates a read operation and sends the alert response address (ARA = 0001 100). This is a general call address that must not be used as a specific device address.
3. The device whose $\overline{\text{SMBALERT}}$ output is low responds to the alert response address, and the master reads its device address. The address of the device is now known and it can be interrogated in the usual way.
4. If more than one device's $\overline{\text{SMBALERT}}$ output is low, the one with the lowest device address will have priority, in accordance with normal SMBus arbitration.

5. Once the ADM1027 has responded to the alert response address, the master must read the status registers and the $\overline{\text{SMBALERT}}$ will only be cleared if the error condition has gone away.

SMBus Timeout

The ADM1027 includes an SMBus timeout feature. If there is no SMBus activity for a minimum of 15 ms and a maximum of 35 ms, the ADM1027 assumes that the bus is locked and releases the bus. This prevents the device from locking or holding the SMBus expecting data. Some SMBus controllers cannot handle the SMBus timeout feature, so it can be disabled.

CONFIGURATION REGISTER 1 – Register 0x40

<6> TODIS = 0; SMBus timeout enabled (default)

<6> TODIS = 1; SMBus timeout disabled

VOLTAGE MEASUREMENT INPUTS

The ADM1027 has four external voltage measurement channels. It can also measure its own supply voltage, V_{CC}.

Pins 20 to 23 are dedicated to measuring 5 V, 12 V, 2.5 V supplies and the processor core voltage V_{CCP} (0 V to 3 V input). The V_{CC} supply voltage measurement is carried out through the V_{CC} pin (Pin 4). Setting Bit 7 of Configuration Register 1 (Reg. 0x40) allows a 5 V supply to power the ADM1027 and be measured without overranging the V_{CC} measurement channel. The 2.5 V input can be used to monitor a chipset supply voltage in computer systems.

ANALOG-TO-DIGITAL CONVERTER

All analog inputs are multiplexed into the on-chip, successive approximation, analog-to-digital converter. This has a resolution of 10 bits. The basic input range is 0 V to 2.25 V, but the inputs have built-in attenuators to allow measurement of 2.5 V, 3.3 V, 5 V, 12 V and the processor core voltage V_{CCP}, without any external components. To allow for the tolerance of these supply voltages, the ADC produces an output of 3/4 full scale (768 decimal or 300 hex) for the nominal input voltage, and so has adequate headroom to cope with overvoltages.

INPUT CIRCUITRY

The internal structure for the analog inputs is shown in Figure 12. Each input circuit consists of an input protection diode, an attenuator, and a capacitor to form a first order low-pass filter that gives the input immunity to high frequency noise.

VOLTAGE MEASUREMENT REGISTERS

Reg. 0x20 **2.5 V Reading** = 0x00 default

Reg. 0x21 **V_{CCP} Reading** = 0x00 default

Reg. 0x22 **V_{CC} Reading** = 0x00 default

Reg. 0x23 **5 V Reading** = 0x00 default

Reg. 0x24 **12 V Reading** = 0x00 default

VOLTAGE MEASUREMENT LIMIT REGISTERS

Associated with each voltage measurement channel are high and low limit registers. Exceeding the programmed high or low limit causes the appropriate status bit to be set. Exceeding either limit can also generate $\overline{\text{SMBALERT}}$ interrupts.

Reg. 0x44 **2.5 V Low Limit** = 0x00 default

Reg. 0x45 **2.5 V High Limit** = 0xFF default

Reg. 0x46 **V_{CCP} Low Limit** = 0x00 default

Reg. 0x47 **V_{CCP} High Limit** = 0xFF default

Reg. 0x48 **V_{CC} Low Limit** = 0x00 default

Reg. 0x49 **V_{CC} High Limit** = 0xFF default

Reg. 0x4A **5 V Low Limit** = 0x00 default

Reg. 0x4B **5 V High Limit** = 0xFF default

Reg. 0x4C **12 V Low Limit** = 0x00 default

Reg. 0x4D **12 V High Limit** = 0xFF default

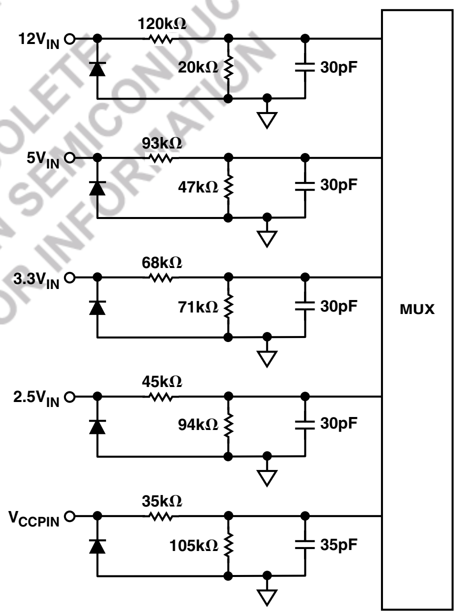


Figure 12. Structure of Analog Inputs

Table II shows the input ranges of the analog inputs and output codes of the 10-bit A/D converter.

When the ADC is running, it samples and converts a voltage input in 711 μs, and averages 16 conversions to reduce noise. Therefore a measurement on any input takes nominally 11.38 ms.

Table II. 10-Bit A/D Output Code vs. V_{IN}

Input Voltage					A/D Output	
12 V_{IN}	**5 V_{IN}**	**V_{CC} (3.3 V_{IN})***	**2.5 V_{IN}**	**V_{CCPIN}**	**Decimal**	**Binary (10 Bits)**
<0.0156	<0.0065	<0.0042	<0.0032	<0.00293	0	00000000 00
0.0156 – 0.0312	0.0065 – 0.0130	0.0042 – 0.0085	0.0032 – 0.0065	0.0293 – 0.0058	1	00000000 01
0.0312 – 0.0469	0.0130 – 0.0195	0.0085 – 0.0128	0.0065 – 0.0097	0.0058 – 0.0087	2	00000000 10
0.0469 – 0.0625	0.0195 – 0.0260	0.0128 – 0.0171	0.0097 – 0.0130	0.0087 – 0.0117	3	00000000 11
0.0625 – 0.0781	0.0260 – 0.0325	0.0171 – 0.0214	0.0130 – 0.0162	0.0117 – 0.0146	4	00000001 00
0.0781 – 0.0937	0.0325 – 0.0390	0.0214 – 0.0257	0.0162 – 0.0195	0.0146 – 0.0175	5	00000001 01
0.0937 – 0.1093	0.0390 – 0.0455	0.0257 – 0.0300	0.0195 – 0.0227	0.0175 – 0.0205	6	00000001 10
0.1093 – 0.1250	0.0455 – 0.0521	0.0300 – 0.0343	0.0227 – 0.0260	0.0205 – 0.0234	7	00000001 11
0.1250 – 0.1406	0.0521 – 0.0586	0.0343 – 0.0386	0.0260 – 0.0292	0.0234 – 0.0263	8	00000010 00
4.0000 – 4.0156	1.6675 – 1.6740	1.1000 – 1.1042	0.8325 – 0.8357	0.7500 – 0.7529	256 (1/4 scale)	01000000 00
8.0000 – 8.0156	3.3300 – 3.3415	2.2000 – 2.2042	1.6650 – 1.6682	1.5000 – 1.5029	512 (1/2 scale)	10000000 00
12.0000 – 12.0156	5.0025 – 5.0090	3.3000 – 3.3042	2.4975 – 2.5007	2.2500 – 2.2529	768 (3/4 scale)	11000000 00
15.8281 – 15.8437	6.5983 – 6.6048	4.3527 – 4.3570	3.2942 – 3.2974	2.9677 – 2.9707	1013	11111101 01
15.8437 – 15.8593	6.6048 – 6.6113	4.3570 – 4.3613	3.2974 – 3.3007	2.9707 – 2.9736	1014	11111101 10
15.8593 – 15.8750	6.6113 – 6.6178	4.3613 – 4.3656	3.3007 – 3.3039	2.9736 – 2.9765	1015	11111101 11
15.8750 – 15.8906	6.6178 – 6.6244	4.3656 – 4.3699	3.3039 – 3.3072	2.9765 – 2.9794	1016	11111110 00
15.8906 – 15.9062	6.6244 – 6.6309	4.3699 – 4.3742	3.3072 – 3.3104	2.9794 – 2.9824	1017	11111110 01
15.9062 – 15.9218	6.6309 – 6.6374	4.3742 – 4.3785	3.3104 – 3.3137	2.9824 – 2.9853	1018	11111110 10
15.9218 – 15.9375	6.6374 – 6.4390	4.3785 – 4.3828	3.3137 – 3.3169	2.9853 – 2.9882	1019	11111110 11
15.9375 – 15.9531	6.6439 – 6.6504	4.3828 – 4.3871	3.3169 – 3.3202	2.9882 – 2.9912	1020	11111111 00
15.9531 – 15.9687	6.6504 – 6.6569	4.3871 – 4.3914	3.3202 – 3.3234	2.9912 – 2.9941	1021	11111111 01
15.9687 – 15.9843	6.6569 – 6.6634	4.3914 – 4.3957	3.3234 – 3.3267	2.9941 – 2.9970	1022	11111111 10
>15.9843	>6.6634	>4.3957	>3.3267	>2.9970	1023	11111111 11

*The V_{CC} output codes listed assume that V_{CC} is 3.3 V. If V_{CC} input is reconfigured for 5 V operation (by setting Bit 7 of Configuration Register 1), then the V_{CC} output codes are the same as for the 5 V_{IN} column.

VID CODE MONITORING

The ADM1027 has five dedicated voltage ID (VID code) inputs. These are digital inputs that can be read back through the VID register (Reg. 0x43) to determine the processor voltage required/being used in the system. Five VID code inputs support VRM9.x solutions.

VID CODE REGISTER – Register 0x43

<0> = VID0 (reflects logic state of Pin 5)

<1> = VID1 (reflects logic state of Pin 6)

<2> = VID2 (reflects logic state of Pin 7)

<3> = VID3 (reflects logic state of Pin 8)

<4> = VID4 (reflects logic state of Pin 19)

ADDITIONAL ADC FUNCTIONS

A number of other functions are available on the ADM1027 to offer the systems designer increased flexibility:

Turn Off Averaging

For each voltage measurement read from a value register, 16 readings have actually been made internally and the results averaged before being placed into the value register. There may be an instance where the user would like to speed up conversions. Setting Bit 4 of Configuration Register 2 (Reg. 0x73) turns averaging off. This effectively gives a reading 16× faster than 711 µs, but the reading may be noisier.

Bypass Voltage Input Attenuators

Setting Bit 5 of Configuration Register 2 (Reg. 0x73) removes the attenuation circuitry from the 2.5 V, V_{CCP}, V_{CC}, 5 V, and 12 V inputs. This allows the user to directly connect external sensors or rescale the analog voltage measurement inputs for other applications. The input range of the ADC without the attenuators is 0 V to 2.25 V.

Single-Channel ADC Conversions

Setting Bit 6 of Configuration Register 2 (Reg. 0x73) places the ADM1027 into single-channel ADC conversion mode. In this mode, the ADM1027 can be made to read a single voltage channel only. If the internal ADM1027 clock is used, the selected input will be read every 711 µs. The appropriate ADC channel is selected by writing to Bits <7:5> of TACH1 minimum high byte register (0x55).

Bits <7:5> Reg. 0x55	Channel Selected
000	2.5 V
001	V_{CCP}
010	V_{CC}
011	5 V
100	12 V

Configuration Register 2 (Reg. 0x73)
<4> = 1 Averaging off
<5> = 1 Bypass input attenuators
<6> = 1 Single-channel convert mode

TACH1 Minimum High Byte (Reg. 0x55)
<7:5> Selects ADC channel for single-channel convert mode

TEMPERATURE MEASUREMENT SYSTEM

Local Temperature Measurement

The ADM1027 contains an on-chip band gap temperature sensor whose output is digitized by the on-chip 10-bit ADC. The 8-bit MSB temperature data is stored in the local temp register (Address 0x26). As both positive and negative temperatures can be measured, the temperature data is stored in twos complement format, as shown in Table III. Theoretically, the temperature sensor and ADC can measure temperatures from –128°C to +127°C with a resolution of 0.25°C. However, this exceeds the operating temperature range of the device (0°C to 105°C), so local temperature measurements outside this range are not possible. Temperature measurement from –127°C to +127°C is possible using a remote sensor.

Remote Temperature Measurement

The ADM1027 can measure the temperature of two remote diode sensors or diode-connected transistors connected to Pins 15 and 16, or 17 and 18.

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, so the technique is unsuitable for mass production. The technique used in the ADM1027 is to measure the change in V_{be} when the device is operated at two different currents. This is given by

$$\Delta V_{be} = KT/q \times \ln(N)$$

where:

K is Boltzmann's constant.

q is charge on the carrier.

T is absolute temperature in kelvins.

N is the ratio of the two currents.

Figure 13 shows the input signal conditioning used to measure the output of a remote temperature sensor. This figure shows the external sensor as a substrate transistor, provided for temperature monitoring on some microprocessors. It could equally well be a discrete transistor such as a 2N3904/06.

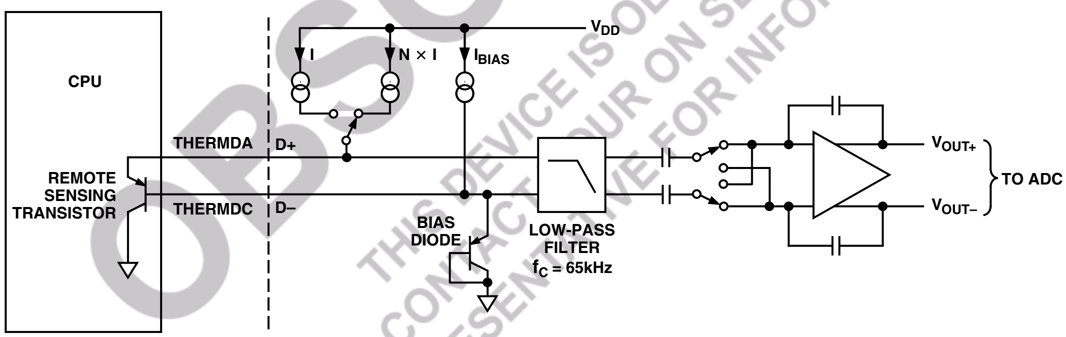


Figure 13. Signal Conditioning for Remote Diode Temperature Sensors

If a discrete transistor is used, the collector will not be 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. Figure 14 shows how to connect the ADM1027 to an NPN or PNP transistor for temperature measurement. To prevent ground noise from interfering with the measurement, the more negative terminal of the sensor is not referenced to ground, but is biased above ground by an internal diode at the D– input.

To measure ΔV_{be}, the sensor is switched between operating currents of I and N × I. The resulting waveform is passed through a 65 kHz low-pass filter to remove noise, and to a chopper-stabilized amplifier that performs the functions of amplification and rectification of the waveform to produce a dc voltage proportional to ΔV_{be}. This voltage is measured by the ADC to give a temperature output in 10-bit, twos complement format. To further reduce the effects of noise, digital filtering is performed by averaging the results of 16 measurement cycles. A remote temperature measurement takes nominally 25.5 ms. The results of remote temperature measurements are stored in 10-bit, twos complement format, as illustrated in Table III. The extra resolution for the temperature measurements is held in the Extended Resolution Register 2 (Reg. 0x77). This gives temperature readings with a resolution of 0.25°C.

Table III. Temperature Data Format*

Temperature	Digital Output (10-Bit)
–128°C	1000 0000 **00**
–125°C	1000 0011 **00**
–100°C	1001 1100 **00**
–75°C	1011 0101 **00**
–50°C	1100 1110 **00**
–25°C	1110 0111 **00**
–10°C	1111 0110 **00**
0°C	0000 0000 **00**
+10.25°C	0000 1010 **01**
+25.5°C	0001 1001 **10**
+50.75°C	0011 0010 **11**
+75°C	0100 1011 **00**
+100°C	0110 0100 **00**
+125°C	0111 1101 **00**
+127°C	0111 1111 **00**

*Bold denotes 2 LSBs of measurement in Extended Resolution Register 2 (Reg. 0x77) with 0.25°C resolution.

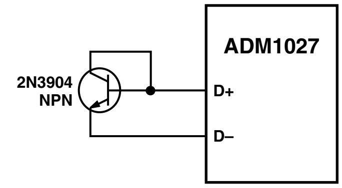


Figure 14a. Measuring Temperature Using an NPN Transistor

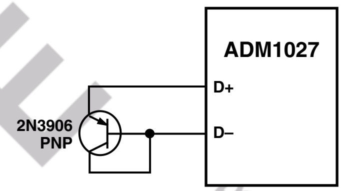


Figure 14b. Measuring Temperature Using a PNP Transistor

NULLING OUT TEMPERATURE ERRORS

As CPUs run faster, it is getting more difficult to avoid high frequency clocks when routing the D–/D+ traces around a system board. Even when recommended layout guidelines are followed, there may still be temperature errors attributed to noise being coupled onto the D+/D– lines. High frequency noise generally has the effect of giving temperature measurements that are too high by a constant amount. The ADM1027 has temperature offset registers at addresses 0x70, 0x71, and 0x72 for the Remote 1, Local, and Remote 2 temperature channels. By doing a one-time calibration of the system, you can determine the offset caused by system board noise and null it out using the offset registers. The offset registers automatically add a twos complement 8-bit reading to every temperature measurement. The LSB adds a 1°C offset to the temperature reading so the 8-bit register effectively allows temperature offsets of up to ±127°C with a resolution of 1°C. This ensures that the readings in the temperature measurement registers are as accurate as possible.

TEMPERATURE OFFSET REGISTERS

Reg. 0x70 **Remote 1 Temperature Offset** = 0x00 (0°C default)

Reg. 0x71 **Local Temperature Offset** = 0x00 (0°C default)

Reg. 0x72 **Remote 2 Temperature Offset** = 0x00 (0°C default)

TEMPERATURE MEASUREMENT REGISTERS

Reg. 0x25 **Remote 1 Temperature** = 0x80 default

Reg. 0x26 **Local Temperature** = 0x80 default

Reg. 0x27 **Remote 2 Temperature** = 0x80 default

Reg. 0x77 **Extended Resolution 2** = 0x00 default

<7:6> TDM2 = Remote 2 Temperature LSBs

<5:4> LTMP = Local Temperature LSBs

<3:2> TDM1 = Remote 1 Temperature LSBs

TEMPERATURE MEASUREMENT LIMIT REGISTERS

Associated with each temperature measurement channel are high and low limit registers. Exceeding the programmed high or low limit causes the appropriate status bit to be set. Exceeding either limit can also generate $\overline{\text{SMBALERT}}$ interrupts.

Reg. 0x4E **Remote 1 Temperature Low Limit** = 0x81 default

Reg. 0x4F **Remote 1 Temperature High Limit** = 0x7F default

Reg. 0x50 **Local Temperature Low Limit** = 0x81 default

Reg. 0x51 **Local Temperature High Limit** = 0x7F default

Reg. 0x52 **Remote 2 Temperature Low Limit** = 0x81 default

Reg. 0x53 **Remote 2 Temperature High Limit** = 0x7F default

READING TEMPERATURE FROM THE ADM1027

It is important to note that temperature can be read from the ADM1027 as an 8-bit value (with 1°C resolution), or as a 10-bit value (with 0.25°C resolution). If only 1°C resolution is required, the temperature readings can be read back at any time and in no particular order.

If the 10-bit measurement is required, this involves a 2-register read for each measurement. The extended resolution register (Reg. 0x77) should be read first. This causes all temperature reading registers to be frozen until all temperature reading registers have been read from. This prevents an MSB reading from being updated while its two LSBs are being read, and vice versa.

ADDITIONAL ADC FUNCTIONS

A number of other functions are available on the ADM1027 to offer the systems designer increased flexibility:

Turn Off Averaging

For each temperature measurement read from a value register, 16 readings have actually been made internally and the results averaged before being placed into the value register. There may be an instance where the user would like to take a very fast measurement, e.g., of CPU temperature. Setting Bit 4 of Configuration Register 2 (Reg. 0x73) turns averaging off. This takes a reading every 13 ms. The measurement itself takes 4 ms.

Single-Channel ADC Conversions

Setting Bit 6 of Configuration Register 2 (Reg. 0x73) places the ADM1027 into single-channel ADC conversion mode. In this mode, the ADM1027 can be made to read a single temperature channel only. If the internal ADM1027 clock is used, the selected input will be read every 1.4 ms. The appropriate ADC channel is selected by writing to Bits <7:5> of TACH1 minimum high byte register (Reg. 0x55).

Bits <7:5> Reg 0x55	Channel Selected
101	Remote 1 Temp
110	Local Temp
111	Remote 2 Temp

Configuration Register 2 (Reg. 0x73)

<4> = 1 Averaging off

<6> = 1 Single-channel convert mode

TACH1 Minimum High Byte (Reg. 0x55)

<7:5> Selects ADC channel for single-channel convert mode

OVERTEMPERATURE EVENTS

Overtemperature events on any of the temperature channels can be detected and dealt with automatically. Registers 0x6A to 0x6C are the THERM limits. When a temperature exceeds its THERM limit, all fans will run at 100% duty cycle. The fans will stay running at 100% until the temperature drops below THERM – 4°C.

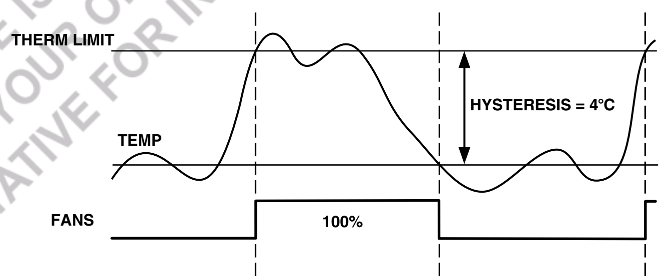


Figure 15. THERM Limit Operation

$\overline{\text{SMBALERT}}$, STATUS, AND MASK REGISTERS

$\overline{\text{SMBALERT}}$ CONFIGURATION

Pin 10 of the ADM1027 can be configured as either PWM2 or as an $\overline{\text{SMBALERT}}$ output. The $\overline{\text{SMBALERT}}$ output may be used to signal out-of-limit conditions as explained below. The default state of Pin 10 is PWM2. To configure Pin 10 as $\overline{\text{SMBALERT}}$:

Configuration Reg. 3 (Addr = 0x78), Bit 0 = 1 = $\overline{\text{SMBALERT}}$

Configuration Reg. 3 (Addr = 0x78), Bit 0 = 0 = PWM2 = default

LIMIT VALUES

Associated with each measurement channel on the ADM1027 are high and low limits. 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, $\overline{\text{SMBALERT}}$ interrupts can be generated to flag a processor or microcontroller of out-of-limit conditions.

8-BIT LIMITS

The following is a list of 8-bit limits on the ADM1027:

Voltage Limit Registers

Reg. 0x44 **2.5 V Low Limit** = 0x00 default

Reg. 0x45 **2.5 V High Limit** = 0xFF default

Reg. 0x46 **V_{CCP} Low Limit** = 0x00 default

Reg. 0x47 **V_{CCP} High Limit** = 0xFF default

Reg. 0x48 **V_{CC} Low Limit** = 0x00 default

Reg. 0x49 **V_{CC} High Limit** = 0xFF default

Reg. 0x4A **5 V Low Limit** = 0x00 default

Reg. 0x4B **5 V High Limit** = 0xFF default

Reg. 0x4C **12 V Low Limit** = 0x00 default

Reg. 0x4D **12 V High Limit** = 0xFF default

Temperature Limit Registers

Reg. 0x4E **Remote 1 Temp Low Limit** = 0x81 default

Reg. 0x4F **Remote 1 Temp High Limit** = 0x7F default

Reg. 0x6A **Remote 1 THERM Limit** = 0x64 default

Reg. 0x50 **Local Temp Low Limit** = 0x81 default

Reg. 0x51 **Local Temp High Limit** = 0x7F default

Reg. 0x6B **Local THERM Limit** = 0x64 default

Reg. 0x52 **Remote 2 Temp Low Limit** = 0x81 default

Reg. 0x53 **Remote 2 Temp High Limit** = 0x7F default

Reg. 0x6C **Remote 2 THERM Limit** = 0x64 default

16-Bit Limits

The fan TACH measurements are 16-bit results. The fan TACH limits are also 16 bits, consisting of a high byte and low byte. Since fans running underspeed or stalled are normally the only conditions of interest, only high limits exist for fan TACHs. Since fan TACH period is actually being measured, exceeding the limit indicates a slow or stalled fan.

Fan Limit Registers

Reg. 0x54 **TACH1 Minimum Low Byte** = 0xFF default

Reg. 0x55 **TACH1 Minimum High Byte** = 0xFF default

Reg. 0x56 **TACH2 Minimum Low Byte** = 0xFF default

Reg. 0x57 **TACH2 Minimum High Byte** = 0xFF default

Reg. 0x58 **TACH3 Minimum Low Byte** = 0xFF default

Reg. 0x59 **TACH3 Minimum High Byte** = 0xFF default

Reg. 0x5A **TACH4 Minimum Low Byte** = 0xFF default

Reg. 0x5B **TACH4 Minimum High Byte** = 0xFF default

OUT-OF-LIMIT COMPARISONS

The ADM1027 will measure all parameters in round-robin format and set the appropriate status bit for out-of-limit conditions. Comparisons are done differently depending on whether the measured value is being compared to a high or low limit.

HIGH LIMIT: > COMPARISON PERFORMED

LOW LIMIT: < OR = COMPARISON PERFORMED

ANALOG MONITORING CYCLE TIME

The analog monitoring cycle begins when a 1 is written to the start bit (Bit 0) of Configuration Register 1(Reg. 0x40). The ADC measures each analog input in turn and as each measurement is completed, the result is automatically stored in the appropriate value register. This round-robin monitoring cycle continues unless disabled by writing a 0 to Bit 0 of Configuration Register 1.

Since the ADC will normally be left to free-run in this manner, the time taken to monitor all the analog inputs will normally not be of interest as the most recently measured value of any input can be read out at any time.

For applications where the monitoring cycle time is important, it can easily be calculated.

The total number of channels measured is

- Four dedicated supply voltage inputs
- 3.3 V_{STBY} or 5 V supply (V_{CC} pin)
- Local temperature
- Two remote temperatures

As mentioned previously, the ADC performs round-robin conversions and takes 11.38 ms for each voltage measurement, 12 ms for a local temperature reading, and 25.5 ms for a remote temperature reading.

The total monitoring cycle time for averaged voltage and temperature monitoring is therefore nominally

$(5 \times 11.38) + 12 + (2 \times 25.5) = 120$ ms

Fan TACH measurements are made in parallel and are not synchronized with the analog measurements in any way.

STATUS REGISTERS

The results of limit comparisons are stored in Status Registers 1 and 2. The status register bit for each channel reflects the status of the last measurement and limit comparison on that channel. If a measurement is within limits, the corresponding status register bit will be cleared to 0. If the measurement is out-of-limits, the corresponding status register bit will be set to 1.

The state of the various measurement channels may be polled by reading the status registers over the serial bus. When 1, Bit 7 (OOL) of Status Register 1 (Reg. 0x41) means that an out-of-limit event has been flagged in Status Register 2. This means that the user need read only Status Register 2 when this bit is set. Alternatively, Pin 10 can be configured as an $\overline{\text{SMBALERT}}$ output. This will automatically notify the system supervisor of an out-of-limit condition. Reading the status registers clears the appropriate status bit as long as the error condition that caused the interrupt has cleared. Status register bits are "sticky." Whenever a status bit gets set, indicating an out-of-limit condition, it will remain set even if the event that caused it has gone away (until read). The only way to clear the status bit is to read the status register after the event has gone away. Interrupt status mask registers (Reg. 0x74, 0x75) allow individual interrupt sources to be masked from causing an $\overline{\text{SMBALERT}}$. However, if one of these masked interrupt sources goes out-of-limit, its associated status bit will get set in the interrupt status registers.

STATUS REGISTER 1 (REG. 0x41)

Bit 7 (OOL) = 1, denotes a bit in Status Register 2 is set and Status Register 2 should be read.

Bit 6 (R2T) = 1, Remote 2 temp high or low limit has been exceeded.

Bit 5 (LT) = 1, Local temp high or low limit has been exceeded.

Bit 4 (R1T) = 1, Remote 1 temp high or low limit has been exceeded.

Bit 3 (5 V) = 1, 5 V high or low limit has been exceeded.

Bit 2 (V_{CC}) = 1, V_{CC} high or low limit has been exceeded.

Bit 1 (V_{CCP}) = 1, V_{CCP} high or low limit has been exceeded.

Bit 0 (2.5 V) = 1, 2.5 V high or low limit has been exceeded.

STATUS REGISTER 2 (REG. 0x42)

Bit 7 (D2) = 1, indicates an open or short on D2+/D2– inputs.

Bit 6 (D1) = 1, indicates an open or short on D2+/D2– inputs.

Bit 5 (FAN4) = 1, indicates Fan 4 has dropped below minimum speed.

Bit 4 (FAN3) = 1, indicates Fan 3 has dropped below minimum speed.

Bit 3 (FAN2) = 1, indicates Fan 2 has dropped below minimum speed.

Bit 2 (FAN1) = 1, indicates Fan 1 has dropped below minimum speed.

Bit 1 (OVT) = 1, indicates that a THERM overtemperature limit has been exceeded.

Bit 0 (12 V) = 1, 12 V high or low limit has been exceeded.

$\overline{\text{SMBALERT}}$ INTERRUPT BEHAVIOR

The ADM1027 can be polled for status, or an $\overline{\text{SMBALERT}}$ interrupt can be generated for out-of-limit conditions. It is important to note how the $\overline{\text{SMBALERT}}$ output and status bits behave when writing interrupt handler software.

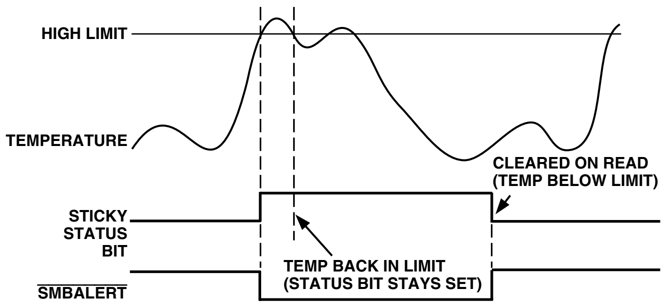


Figure 16. $\overline{\text{SMBALERT}}$ and Status Bit Behavior

Figure 16 shows how the $\overline{\text{SMBALERT}}$ output and sticky status bits behave. Once a limit is exceeded, the corresponding status bit is set to 1. The status bit remains set until the error condition subsides and the status register is read. The status bits are referred to as sticky since they remain set until read by software. This ensures that an out-of-limit event cannot be missed if software is polling the device periodically. Note that the $\overline{\text{SMBALERT}}$ output remains low for the entire duration that a reading is out-of-limit and until the status register has been read. This has implications on how software handles the interrupt.

HANDLING $\overline{\text{SMBALERT}}$ INTERRUPTS

To prevent the system from being tied up servicing interrupts, it is recommend to handle the $\overline{\text{SMBALERT}}$ interrupt as follows:

1. Detect the $\overline{\text{SMBALERT}}$ assertion.
2. Enter the interrupt handler.
3. Read the status registers to identify the interrupt source.
4. Mask the interrupt source by setting the appropriate mask bit in the interrupt mask registers (Reg. 0x74, 0x75).
5. Take the appropriate action for a given interrupt source.
6. Exit the interrupt handler.
7. Periodically poll the status registers. If the interrupt status bit has cleared, reset the corresponding interrupt mask bit to 0. This will cause the $\overline{\text{SMBALERT}}$ output and status bits to behave as shown in Figure 17.

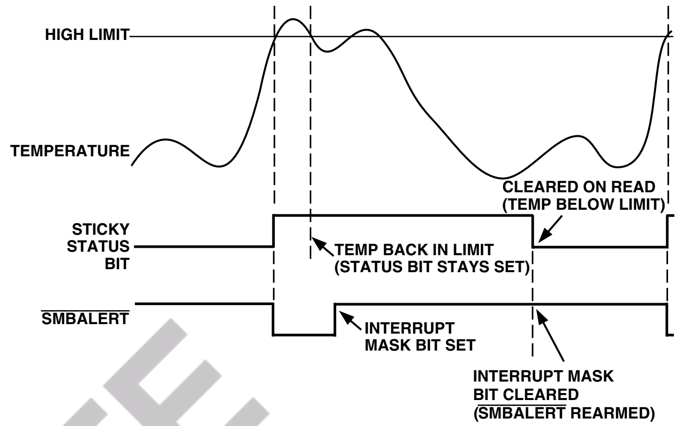


Figure 17. How Masking the Interrupt Source Affects $\overline{\text{SMBALERT}}$ Output

MASKING INTERRUPT SOURCES

Interrupt Mask Registers 1 and 2 are located at Addresses 0x74 and 0x75. These allow individual interrupt sources to be masked out to prevent $\overline{\text{SMBALERT}}$ interrupts. Note that masking an interrupt source prevents only the $\overline{\text{SMBALERT}}$ output from being asserted; the appropriate status bit will be set as normal.

INTERRUPT MASK REGISTER 1 (REG. 0x74)

Bit 7 (OOL) = 1, set this bit to 1 to allow masking of interrupts by Status Register 2. If this bit = 0, then setting a bit in Mask Register 2 to 1 will have no effect.

Bit 6 (R2T) = 1, masks $\overline{\text{SMBALERT}}$ for Remote 2 temperature.

Bit 5 (LT) = 1, masks $\overline{\text{SMBALERT}}$ for local temperature.

Bit 4 (R1T) = 1, masks $\overline{\text{SMBALERT}}$ for Remote 1 temperature.

Bit 3 (5 V) = 1, masks $\overline{\text{SMBALERT}}$ for 5 V channel.

Bit 2 (V_{CC}) = 1, masks $\overline{\text{SMBALERT}}$ for V_{CC} channel.

Bit 1 (V_{CCP}) = 1, masks $\overline{\text{SMBALERT}}$ for V_{CCP} channel.

Bit 0 (2.5 V) = 1, masks $\overline{\text{SMBALERT}}$ for 2.5 V channel.

INTERRUPT MASK REGISTER 2 (REG. 0x75)

Bit 7 (D2) = 1, masks $\overline{\text{SMBALERT}}$ for Diode 2 errors.

Bit 6 (D1) = 1, masks $\overline{\text{SMBALERT}}$ for Diode 1 errors.

Bit 5 (FAN4) = 1, masks $\overline{\text{SMBALERT}}$ for Fan 4.

Bit 4 (FAN3) = 1, masks $\overline{\text{SMBALERT}}$ for Fan 3.

Bit 3 (FAN2) = 1, masks $\overline{\text{SMBALERT}}$ for Fan 2.

Bit 2 (FAN1) = 1, masks $\overline{\text{SMBALERT}}$ for Fan 1.

Bit 1 (OVT) = 1, masks $\overline{\text{SMBALERT}}$ for overtemperature (exceeding THERM limits).

Bit 0 (12 V) = 1, masks $\overline{\text{SMBALERT}}$ for 12 V channel.

FAN DRIVE CIRCUITRY

Fan Drive Using PWM Control

The ADM1027 uses Pulsewidth Modulation (PWM) to control fan speed. This relies on varying the duty cycle (or on/off ratio) of a square wave applied to the fan to vary the fan speed. The external circuitry required to drive a fan using PWM control is extremely simple. A single NMOSFET is the only drive device required. The specifications of the MOSFET depend on the maximum current required by the fan being driven. Typical notebook fans draw a nominal 170 mA, so SOT devices can be used where board space is a concern. In desktops, fans can typically draw 250 mA to 300 mA each. If you drive several fans in parallel from a single PWM output or drive larger server fans, the MOSFET will need to handle the higher current requirements. The only other stipulation is that the MOSFET should have a gate voltage drive, V_{GS} < 3.3 V for direct interfacing to the PWM_OUT pin. V_{GS} can be greater than 3.3 V as long as the pull-up on the gate is tied to 5 V. The MOSFET should also have a low on resistance to ensure that there is not significant voltage drop across the FET. This would reduce the voltage applied across the fan and thus the maximum operating speed of the fan.

Figure 18 shows how a 3-wire fan may be driven using PWM control.

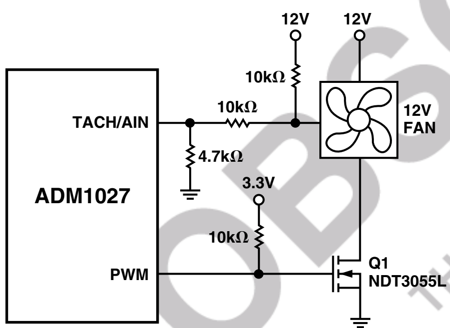


Figure 18. Driving a 3-Wire Fan Using an N-Channel MOSFET

Figure 18 uses a 10 kΩ pull-up resistor for the TACH signal. This assumes that the TACH signal is open-collector from the fan. In all cases, the TACH signal from the fan *must* be kept below 5 V maximum to prevent damaging the ADM1027. If in doubt as to whether the fan used has an open-collector or totem pole TACH output, use one of the input signal conditioning circuits shown in the Fan Speed Measurement section.

Figure 19 shows a fan drive circuit using an NPN transistor such as a general-purpose MMBT2222. While these devices are inexpensive, they tend to have much lower current handling capabilities and higher on resistance than MOSFETs. When choosing a transistor, care should be taken to ensure that it meets the fan's current requirements. Ensure that the base resistor is chosen such that the transistor is saturated when the fan is powered on.

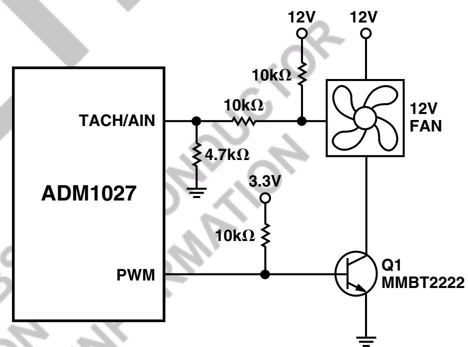


Figure 19. Driving a 3-Wire Fan Using an NPN Transistor

Driving 2 Fans From PWM3

Note that the ADM1027 has four TACH inputs available for fan speed measurement, but only three PWM drive outputs. If a fourth fan is being used in the system, it should be driven from the PWM3 output in parallel with the third fan. Figure 20 shows how to drive two fans in parallel using low cost NPN transistors. Figure 21 is the equivalent circuit using the NDT3055L MOSFET. Note that since the MOSFET can handle up to 3.5 A, it is simply a matter of connecting another fan directly in parallel with the first.

Care should be taken in designing drive circuits with transistors and FETs to ensure that the PWM pins are not required to source current, and that they sink less than the 8 mA max current specified on the data sheet.

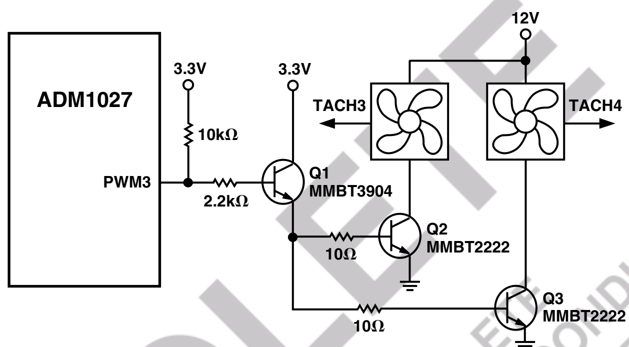


Figure 20. Interfacing Two Fans in Parallel to the PWM3 Output Using Low Cost NPN Transistors

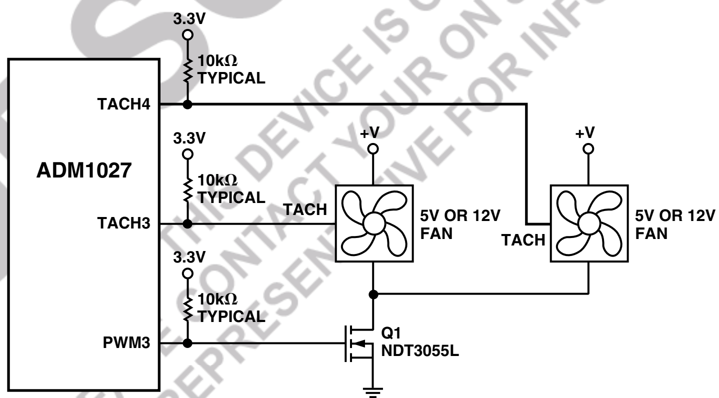


Figure 21. Interfacing Two Fans in Parallel to the PWM3 Output Using a Single N-Channel MOSFET

Driving 2-Wire Fans

Figure 22 shows how a 2-wire fan may be connected to the ADM1027. This circuit allows the speed of a 2-wire fan to be measured even though the fan has no dedicated TACH signal. A series resistor, R_{SENSE}, in the fan circuit converts the fan commutation pulses into a voltage. This is ac-coupled into the ADM1027 through the 0.01 μF capacitor. On-chip signal conditioning allows accurate monitoring of fan speed. For fans drawing approximately 200 mA, a 2 Ω R_{SENSE} value is suitable. For fans that draw more current, such as larger desktop or server fans, R_{SENSE} may be reduced. The smaller R_{SENSE} is the better, since more voltage will be developed across the fan, and the fan will spin faster. Figure 23 shows a typical plot of the sensing waveform at a TACH/AIN pin. The most important thing is that the negative going spikes are more than 250 mV in amplitude. This allows fan speed to be reliably determined.

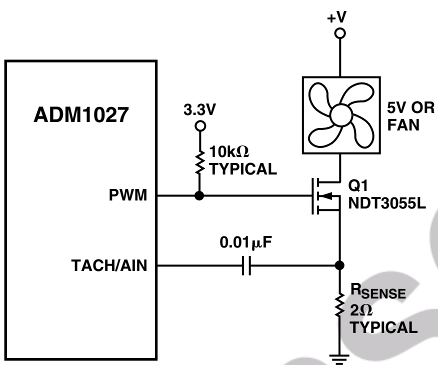


Figure 22. Driving a 2-Wire Fan

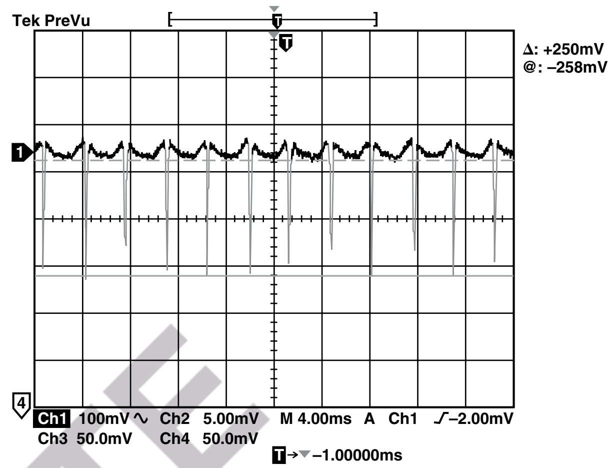


Figure 23. Fan Speed Sensing Waveform at TACH/AIN Pin

Laying Out for 2-Wire and 3-Wire Fans

Figure 24 shows how to lay out a common circuit arrangement for 2-wire and 3-wire fans. Some components will not be populated depending on whether a 2-wire or 3-wire fan is being used.

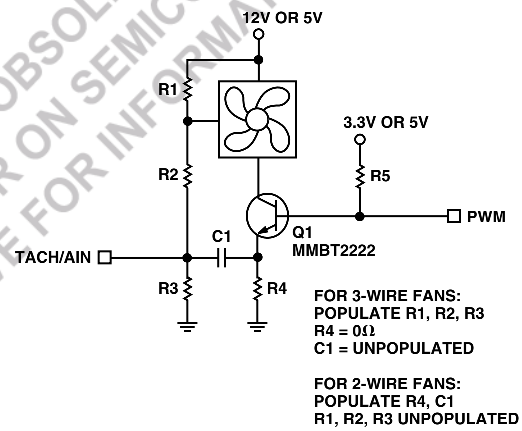


Figure 24. Planning for 2-Wire or 3-Wire Fans on a PCB

FAN SPEED MEASUREMENTS

TACH Inputs

Pins 11, 12, 9, and 14 are open-drain TACH inputs intended for fan speed measurement.

Signal conditioning in the ADM1027 accommodates the slow rise and fall times typical of fan tachometer outputs. The maximum input signal range is 0 V to 5 V, even where V_{CC} is less than 5 V. In the event that these inputs are supplied from fan outputs that exceed 0 V to 5 V, either resistive attenuation of the fan signal or diode clamping must be included to keep inputs within an acceptable range.

Figures 25a to 25d show circuits for most common fan TACH outputs. If the fan TACH output has a resistive pull-up to V_{CC}, it can be connected directly to the fan input, as shown in Figure 25a.

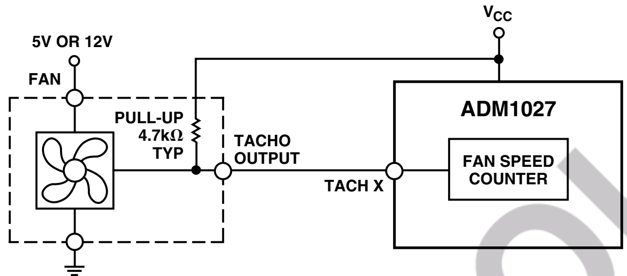


Figure 25a. Fan With TACH Pull-Up to +V_{CC}

If the fan output has a resistive pull-up to 12 V (or other voltage greater than 5 V), then the fan output can be clamped with a Zener diode, as shown in Figure 25b. The Zener diode voltage should be chosen so that it is greater than V_{IH} of the TACH input but less than 5 V, allowing for the voltage tolerance of the Zener. A value of between 3 V and 5 V is suitable.

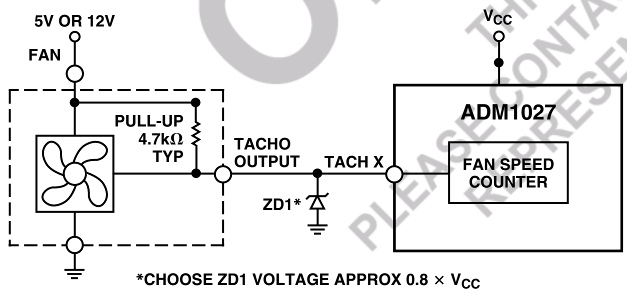


Figure 25b. Fan with TACH Pull-Up to Voltage > 5 V (e.g., 12 V) Clamped with Zener Diode

If the fan has a strong pull-up (less than 1 kΩ) to 12 V, or a totem-pole output, then a series resistor can be added to limit the Zener current, as shown in Figure 25c. Alternatively, a resistive attenuator may be used, as shown in Figure 25d.

R1 and R2 should be chosen such that

$$2\,V < V_{PULLUP} \times R2 / \left(R_{PULLUP} + R1 + R2\right) > 5\,V$$

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 pull-up voltage of 12 V and pull-up resistor less than 1 kΩ, suitable values for R1 and R2 would be 100 kΩ and 47 kΩ. This will give a high input voltage of 3.83 V.

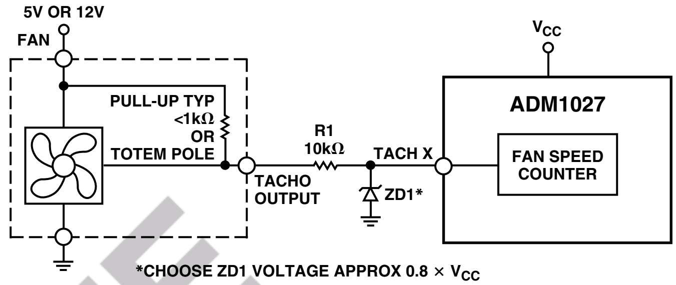


Figure 25c. Fan with Strong TACH Pull-Up to > V_{CC} or Totem-Pole Output, Clamped with Zener and Resistor

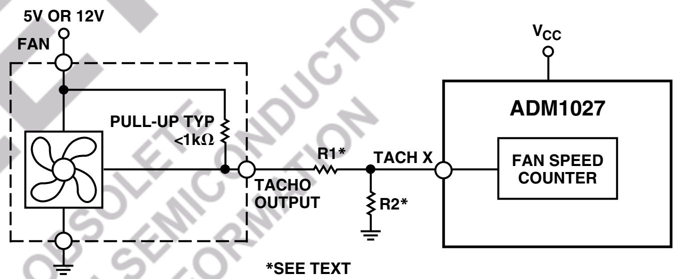


Figure 25d. Fan with Strong TACH Pull-Up to > V_{CC} or Totem-Pole Output, Attenuated with R1/R2

Fan Speed Measurement

The fan counter does not count the fan TACH output pulses directly because the fan speed may be 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 90 kHz oscillator into the input of a 16-bit counter for N periods of the fan TACHO output (Figure 26), so the accumulated count is actually proportional to the fan tachometer period and inversely proportional to the fan speed.

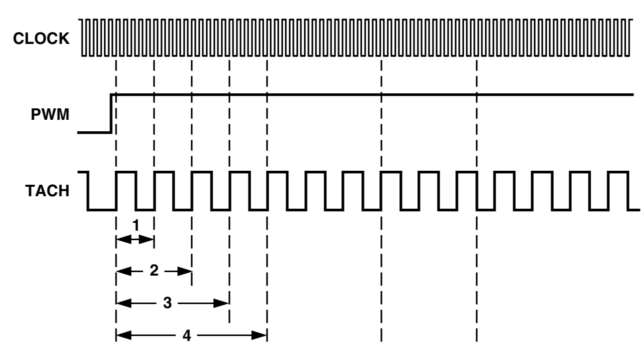


Figure 26. Fan Speed Measurement

N, the number of pulses counted, is determined by the settings of Register 0x7B (fan pulses per revolution register). This register contains two bits for each fan, allowing 1, 2 (default), 3, or 4 TACH pulses to be counted.

Fan Speed Measurement Registers

The fan tachometer readings are 16-bit values consisting of a 2-byte read from the ADM1027.

Reg. 0x28 **TACH1 Low Byte** = 0x00 default

Reg. 0x29 **TACH1 High Byte** = 0x00 default

Reg. 0x2A **TACH2 Low Byte** = 0x00 default

Reg. 0x2B **TACH2 High Byte** = 0x00 default

Reg. 0x2C **TACH3 Low Byte** = 0x00 default

Reg. 0x2D **TACH3 High Byte** = 0x00 default

Reg. 0x2E **TACH4 Low Byte** = 0x00 default

Reg. 0x2F **TACH4 High Byte** = 0x00 default

Reading Fan Speed From the ADM1027

If fan speeds are being measured, this involves a 2-register read for each measurement. The low byte should be read first. This causes the high byte to be frozen until both high and low byte registers have been read from. This prevents erroneous TACH readings.

The fan tachometer reading registers report back the number of 11.11 μs period clocks (90 kHz oscillator) gated to the fan speed counter, from the rising edge of the first fan TACH pulse to the rising edge of the third fan TACH pulse (assuming two pulses per revolution are being counted). Since the device is essentially measuring the fan TACH period, the higher the count value, the slower the fan is actually running. A 16-bit fan tachometer reading of 0xFFFF indicates that the fan either has stalled or is running very slowly (< 100 RPM).

HIGH LIMIT: > COMPARISON PERFORMED

Since actual fan TACH period is being measured, exceeding a fan TACH limit by 1 will set the appropriate status bit and can be used to generate an $\overline{\text{SMBALERT}}$.

Fan Tach Limit Registers

The fan TACH limit registers are 16-bit values consisting of two bytes.

Reg. 0x54 **TACH1 Minimum Low Byte** = 0xFF default

Reg. 0x55 **TACH1 Minimum High Byte** = 0xFF default

Reg. 0x56 **TACH2 Minimum Low Byte** = 0xFF default

Reg. 0x57 **TACH2 Minimum High Byte** = 0xFF default

Reg. 0x58 **TACH3 Minimum Low Byte** = 0xFF default

Reg. 0x59 **TACH3 Minimum High Byte** = 0xFF default

Reg. 0x5A **TACH4 Minimum Low Byte** = 0xFF default

Reg. 0x5B **TACH4 Minimum High Byte** = 0xFF default

Fan Speed Measurement Rate

The fan TACH readings are normally updated once every second.

The fast bit (Bit 3) of Configuration Register 3 (Reg. 0x78), when set, updates the fan TACH readings every 250 ms.

If any of the fans are not being driven by a PWM channel but are powered directly from 5 V or 12 V, their associated dc bit in Configuration Register 3 should be set. This allows TACH readings to be taken on a continuous basis for fans connected directly to a dc source.

Calculating Fan Speed

Assuming a fan with 2 pulses/revolution (and 2 pulses/rev being measured), fan speed is calculated by:

$$Fan\ Speed\ (RPM) = (90{,}000 \times 60)/Fan\ Tach\ Reading$$

where *Fan Tach Reading* = 16-bit fan tachometer reading.

Example:

TACH1 high byte (Reg. 0x29) = 0x17

TACH1 low byte (Reg. 0x28) = 0xFF

What is Fan 1 speed in RPM?

Fan 1 TACH reading = 0x17FF = 6143 decimal

RPM = (f × 60)/fan 1 TACH reading

RPM = (90000 × 60)/6143

Fan Speed = 879 RPM

FAN PULSES PER REVOLUTION

Different fan models can output either 1, 2, 3, or 4 TACH pulses per revolution. Once the number of fan TACH pulses has been determined, it can be programmed into the fan pulses per revolution register (Reg. 0x7B) for each fan. Alternatively, this register can be used to determine the number or pulses/revolution output by a given fan. By plotting fan speed measurements at 100% speed with different pulses/rev setting, the smoothest graph with the lowest ripple determines the correct pulses/rev value.

Fan Pulses Per Revolution Register

<1:0> FAN1 default = 2 pulses/rev

<3:2> FAN2 default = 2 pulses/rev

<5:4> FAN3 default = 2 pulses/rev

<7:6> FAN4 default = 2 pulses/rev

- 00 = 1 pulse/rev
- 01 = 2 pulses/rev
- 10 = 3 pulses/rev
- 11 = 4 pulses/rev

2-Wire Fan Speed Measurements

The ADM1027 is capable of measuring the speed of 2-wire fans, i.e., fans without TACH outputs. To do this, the fan must be interfaced as shown in the Fan Drive Circuitry section of the data sheet. In this case, the TACH inputs need to be reprogrammed as analog inputs, AIN.

CONFIGURATION REGISTER 2 (REG. 0x73)

Bit 3 (AIN4) = 1, Pin 14 is reconfigured to measure the speed of a 2-wire fan using an external sensing resistor and coupling capacitor.

Bit 2 (AIN3) = 1, Pin 9 is reconfigured to measure the speed of a 2-wire fan using an external sensing resistor and coupling capacitor.

Bit 1 (AIN2) = 1, Pin 12 is reconfigured to measure the speed of a 2-wire fan using an external sensing resistor and coupling capacitor.

Bit 0 (AIN1) = 1, Pin 11 is reconfigured to measure the speed of a 2-wire fan using an external sensing resistor and coupling capacitor.

FAN SPIN-UP

The ADM1027 has a unique fan spin-up function. It will spin the fan at 100% PWM duty cycle until two TACH pulses are detected on the TACH input. Once two pulses have been detected, the PWM duty cycle will go to the expected running value, e.g., 33%. The advantage of this is that fans have different spin-up characteristics and will take different times to overcome inertia. The ADM1027 just runs the fans fast enough to overcome inertia and will be quieter on spin-up than fans programmed to spin up for a given spin-up time.

FAN START-UP TIMEOUT

To prevent false interrupts being generated as a fan spins up (since it is below running speed), the ADM1027 includes a fan start-up timeout function. This is the time limit allowed for two TACH pulses to be detected on spin-up. For example, if 2 seconds fan start-up timeout is chosen, and no TACH pulses occur within 2 seconds of the start of spin-up, a fan fault is detected and flagged in the interrupt status registers.

PWM1 CONFIGURATION (REG. 0x5C)

<2:0> SPIN These bits control the start-up timeout for PWM1.

000 = No startup timeout
001 = 100 ms
010 = 250 ms (default)
011 = 400 ms
101 = 1 sec
110 = 2 sec
111 = 4 sec

PWM2 CONFIGURATION (REG. 0x5D)

<2:0> SPIN These bits control the start-up timeout for PWM2.

000 = No startup timeout
001 = 100 ms
010 = 250 ms (default)
011 = 400 ms
101 = 1 sec
110 = 2 sec
111 = 4 sec

PWM3 CONFIGURATION (REG. 0x5E)

<2:0> SPIN These bits control the start-up timeout for PWM3.

000 = No startup timeout
001 = 100 ms
010 = 250 ms (default)
011 = 400 ms
101 = 1 sec
110 = 2 sec
111 = 4 sec

Disabling Fan Start-Up Timeout

Although fan start-up makes fan spin-ups much quieter than fixed-time spin-ups, the option is there to use fixed spin-up times. Bit 5 (FSPDIS) = 1 in Configuration Register 1 (Reg. 0x40) disables the spin-up for two TACH pulses. Instead, the fan will spin up for the fixed time as selected in registers 0x5C to 0x5E.