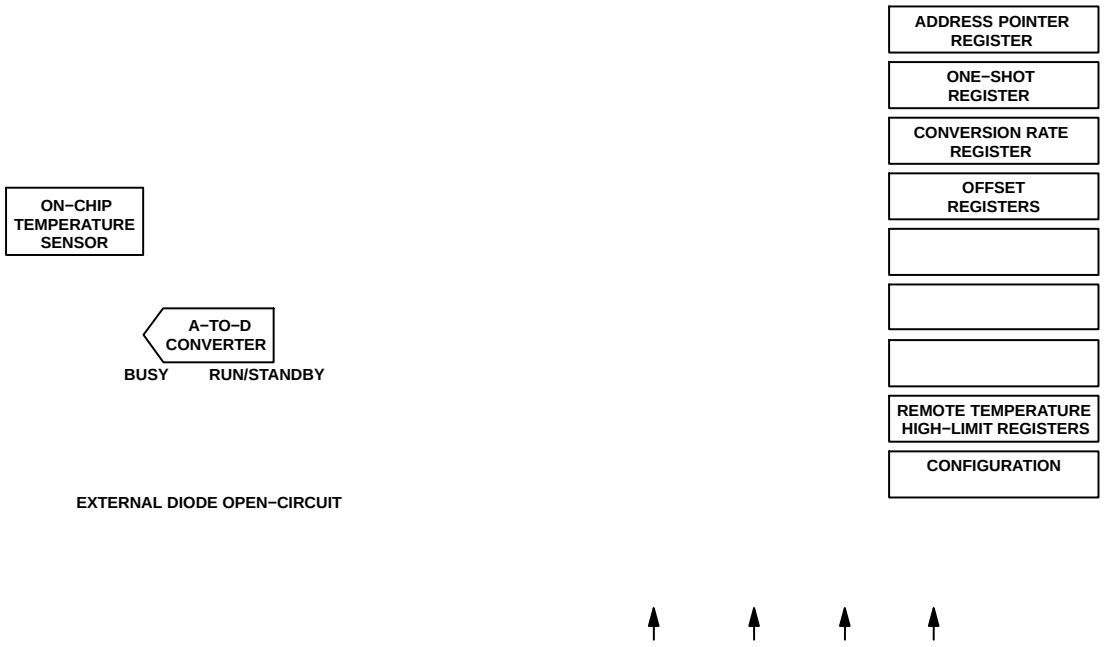


Figure 1. Functional Block Diagram

Table 3. PIN ASSIGNMENT

Pin No.	Mnemonic	Description
	–	
	———	———
	———	

Table 4. ELECTRICAL CHARACTERISTICS

Parameter	Test Condit	omments	in	T	Max	U
Power Supply and ADC						
					–	
					–	
			– –			
			–	–		

Table 4. ELECTRICAL CHARACTERISTICS

TYPICAL PERFORMANCE CHARACTERISTICS

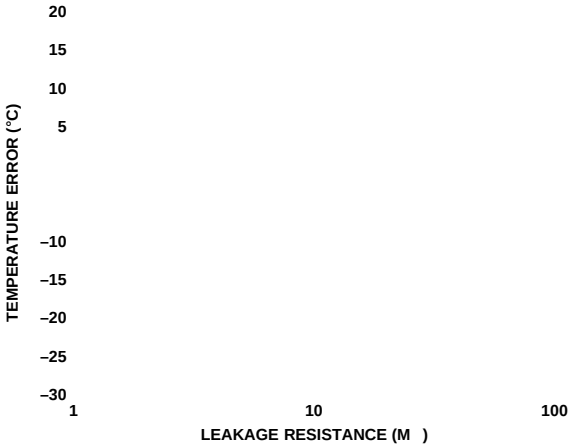


Figure 3. Temperature Error vs. Resistance from Track to V_{DD} and GND

Figure 4. Remote Temperature Error vs. Supply Noise Frequency

Figure 5. Temperature Error vs. Common-mode Noise Frequency

Figure 6. Temperature Error of ADM1023 vs. Pentium III Temperature

Figure 7. Temperature Error vs. Capacitance Between D+ and D−

Figure 8. Standby Supply Current vs. SCLK Frequency

THEORY OF OPERATION

Functional Description

The ADM1023 contains a two-channel analog-to-digital converter (ADC) with special input-signal conditioning to enable operation with remote and on-chip diode temperature sensors. When the ADM1023 is operating normally, the ADC operates in a free-running mode. The analog input multiplexer alternately selects either the on-chip temperature sensor to measure its local temperature or the remote temperature sensor. These signals are digitized by the ADC, and the results are stored in the local and remote temperature value registers. Only the eight most significant bits (MSBs) of the local temperature value are stored as an 8-bit binary word. The remote temperature value is stored as an 11−bit binary word in two registers. The eight MSBs are stored in the remote temperature value high byte register at Address 0x01. The three least significant bits (LSBs) are stored, left justified, in the remote temperature value low byte register at Address 0x10.

Error sources such as PCB track resistance and clock noise can introduce offset errors into measurements on the remote channel. To achieve the specified accuracy on this channel, these offsets must be removed, and two offset registers are provided for this purpose at Address 0x11 and Address 0x12.

An offset value may automatically be added to or subtracted from the measurement by writing an 11-bit, twos complement value to Register 0x11 (high byte) and Register 0x12 (low byte, left-justified).

The offset registers default to 0 at powerup and have no effect if nothing is written to them.

The measurement results are compared with local and remote, high and low temperature limits, stored in six on-chip limit registers. As with the measured value, the local temperature limits are stored as 8-bit values and the remote temperature limits as 11-bit values. Out-of-limit comparisons generate flags that are stored in the status register, and one or more out-of-limit results cause the $\overline{\text{ALERT}}$ output to pull low.

Registers can be programmed, and the device controlled and configured, via the serial system management bus (SMBus). The contents of any register can also be read back via the SMBus.

Control and configuration functions consist of:

Switching the Device between Normal Operation and Standby Mode

Masking or Enabling the $\overline{\text{ALERT}}$ Output

Selecting the Conversion Rate

On initial powerup, the remote and local temperature values default to −128 C. The device normally powers up converting, making a measure of local and remote temperature. These values are then stored before making a comparison with the stored limits. However, if the part is powered up in standby mode ($\overline{\text{STBY}}$ pin pulled low), no new values are written to the register before a comparison is made. As a result, both RLOW and LLOW are tripped in the status register, thus generating an $\overline{\text{ALERT}}$ output. This may be cleared in one of two ways:

Change both the local and remote lower limits to –128 C and read the status register (which in turn clears the $\overline{\text{ALERT}}$ output).

Take the part out of standby and read the status register (which in turn clears the $\overline{\text{ALERT}}$ output). This works only when the measured values are within the limit values.

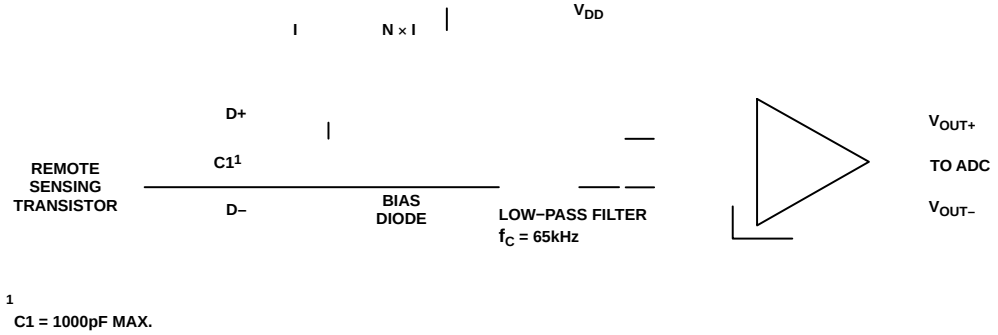


Figure 13. Input Signal Conditioning

This is given by:

Δ = —— ×

where:

K is Boltzmann's constant.
q is the charge on the electron (1.6 10^{-19} Coulombs).
T is the absolute temperature in Kelvins.
N is the ratio of the two collector currents.
n is the ideality factor of the thermal diode (TD).

To measure ΔV_{BE}, the sensor is switched between operating currents of I and NI. The resulting waveform is passed through a low-pass filter to remove noise, then 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, which gives a temperature output in binary format. To further reduce the effects of noise, digital filtering is performed by averaging the results of 16 measurement cycles. Signal conditioning and measurement of the internal temperature sensor are performed in a similar manner.

Figure 13 shows the input signal conditioning used to measure the output of an external temperature sensor. This figure shows the external sensor as a substrate PNP transistor, provided for temperature monitoring on some microprocessors, but it could equally well be a discrete transistor. If a discrete transistor is used, the collector is not grounded and should be connected to the base. 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 Dĭ input. If the sensor is operating in a noisy environment, C1 may optionally be added as a noise filter. Its value is 1000 pF maximum. See the Layout Considerations section for more information on C1.

Sources of Errors on Thermal Transistors Measurement Method; The Effect of Ideality Factor (n)

The effects of ideality factor (n) and beta (β) of the temperature measured by a thermal transistor are described in this section. For a thermal transistor implemented on a submicron process, such as the substrate PNP used on a Pentium III processor, the temperature errors due to the combined effect of the ideality factor and beta are shown to be less than 3 C. Equation 2 is optimized for a substrate PNP transistor (used as a thermal diode) usually found on CPUs designed on submicron CMOS processes such as the Pentium III processor. There is a thermal diode on board each of these processors. The n in Equation 2 represents the ideality factor of this thermal diode. This ideality factor is a measure of the deviation of the thermal diode from ideal behavior.

According to Pentium III processor manufacturing specifications, measured values of n at 100 C are:

= < = < =

=

The ADM1023 takes this ideality factor into consideration when calculating temperature T_{TD} of the thermal diode. The ADM1023 is optimized for $n_{TYPICAL}$ = 1.008; any deviation on n from this typical value causes a temperature error that is calculated below for the n_{MIN} and n_{MAX} of a Pentium III processor at T_{TD} = 100 C.

Δ = ———— × (+ °) =

= – °

Δ = ———— × (+ °) =

= + °

Thus, the temperature error due to variation on n of the thermal diode for a Pentium III processor is about 2.5 C.

In general, this additional temperature error of the thermal diode measurement due to deviations on n from its typical value is given by:

Δ = ——— × (+)

where T_{TD} is in C.

Beta of Thermal Transistor β

In Figure 13, the thermal diode is a substrate PNP transistor where the emitter current is forced into the device. The derivation of Equation 2 assumed that the collector currents were scaled by N as the emitter currents were also scaled by N. Thus, this assumes that beta (β) of the transistor is constant for various collector currents. Figure 14 shows typical β variation vs. collector current for Pentium III processors at 100 C. The maximum β is 4.5 and varies less than 1% over the collector current range from 7 A to 300 A.

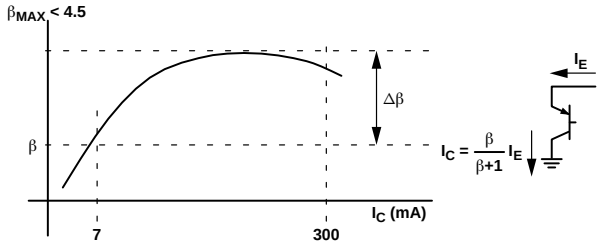


Figure 14. Variation of β with Collector Currents

Expressing the collector current in terms of the emitter current.

= [β/ β +

where:

β( ) = β( ) +

= Δ β/β β = β

Rewriting the equation for ΔV_{BE}, to include the ideality factor, n, and beta, β yields:

Δ = —— × [+ × (β +) / + β + ×]

All β variations of less than 1% (< 0.01) contribute to temperature errors of less than 0.4 C.

Temperature Data Format

One LSB of the ADC corresponds to 0.125 C, so the ADM1023 can measure from 0 C to 127.875 C. The temperature data format and extended temperature resolution are shown in Tables 5 and 6.

Table 5. TEMPERATURE DATA FORMAT (LOCAL AND REMOTE TEMPERATURE HIGH BYTE)

Temperature (°C)	Digital Output

The results of the local and remote temperature measurements are stored in the local and remote temperature value registers and are compared with limits programmed into the local and remote high and low limit registers.

Table 6. EXTENDED TEMPERATURE RESOLUTION (REMOTE TEMPERATURE LOW BYTE)

Extended Resolution (°C)	Temperature Low Bits

Register Functions

The ADM1023 contains registers that are used to store the results of remote and local temperature measurements and high and low temperature limits, and to configure and control the device. A description of these registers follows, and further details are given in Tables 7 to 11. Most of the registers for the ADM1023 are dual-port and have different addresses for read and write operations. Attempting to write to a read address or to read from a write address produces an invalid result. Register addresses above 0x14 are reserved for future use or factory test purposes and should not be written to.

Address Pointer Register

The address pointer register does not have, nor does it require, an address, because it is the register to which the first data byte of every write operation is automatically written. This data byte is an address pointer that sets up one of the other registers for the second byte of the write operation or for a subsequent read operation.

Value Registers

The ADM1023 has three registers to store the results of local and remote temperature measurements. These registers are written to by the ADC and can only be read over the SMBus.

The Offset Register

Two offset registers are provided at Address 0x11 and Address 0x12. These are provided so that the user may remove errors from the measured values of remote temperature. These errors may be introduced by clock noise and PCB track resistance. See Table 8 for an example of offset values.

The offset value is stored as an 11-bit, twos complement value in Register 0x11 (high byte) and Register 0x12 (low byte, left justified). The value of the offset is negative if the MSB of Register 0x11 is 1, and it is positive if the MSB of Register 0x11 is 0. This value is added to the remote temperature. These registers default to 0 at powerup and have no effect if nothing is written to them. The offset register can accept values from ĭ128.875 C to +127.875 C. The ADM1023 detects overflow so the remote temperature value register does not wrap around +127 C or ĭ128 C.

Table 7. LIST OF ADM1023 REGISTERS

Read Address (Hex)	Write Address (Hex)	Name	Power-on Default
			–
			–

Table 9. STATUS REGISTER BIT ASSIGNMENTS

Bit	Name	Function

Configuration Register

Two bits of the configuration register are used. If Bit 6 is 0, which is the power-on default, the device is in operating mode with the ADC converting (see Table 10). If Bit 6 is set to 1, the device is in standby mode and the ADC does not convert. Standby mode can also be selected by taking the $\overline{\text{STBY}}$ pin low. In standby mode, the values of remote and local temperature remain at the value they were before the part was placed in standby mode.

Bit 7 of the configuration register is used to mask the $\overline{\text{ALERT}}$ output. If Bit 7 is 0, which is the power-on default, the $\overline{\text{ALERT}}$ output is enabled. If Bit 7 is set to 1, the $\overline{\text{ALERT}}$ output is disabled.

Table 10. CONFIGURATION REGISTER BIT ASSIGNMENTS

Bit	Name	Function	Power-on Default

Conversion Rate Register

The lowest three bits of this register are used to program the conversion rate by dividing the ADC clock by 1, 2, 4, 8, 16, 32, 64, or 128, to give conversion times from 125 ms (Code 0x07) to 16 seconds (Code 0x00). This register can be written to and read back over the SMBus. The higher five bits of this register are unused and must be set to 0. Use of slower conversion times greatly reduces the device's power consumption, as shown in Table 11.

Table 11. CONVERSION RATE REGISTER CODE

Data	Conversion/Sec	Average Supply Current μA Typ at V_{CC} = 3.3 V

Table 12. DEVICE ADDRESSES

ADD0	ADD1	Device Address

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, SDATA, while the serial clock line, SCLK, 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 8 bits. These bits consist of a 7-bit address (MSB first) plus an R/$\overline{W}$ bit, which determines the direction of the data transfer, that is, whether data is 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 remain idle while the selected device waits for data to be read from or written to it. If the R/$\overline{W}$ bit is 0, the master writes to the slave device. If the R/$\overline{W}$ bit is 1, the master reads from the slave device.
2. Data is sent over the serial bus in sequences of nine 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 ninth clock pulse. This is known as No Acknowledge. The master then 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.

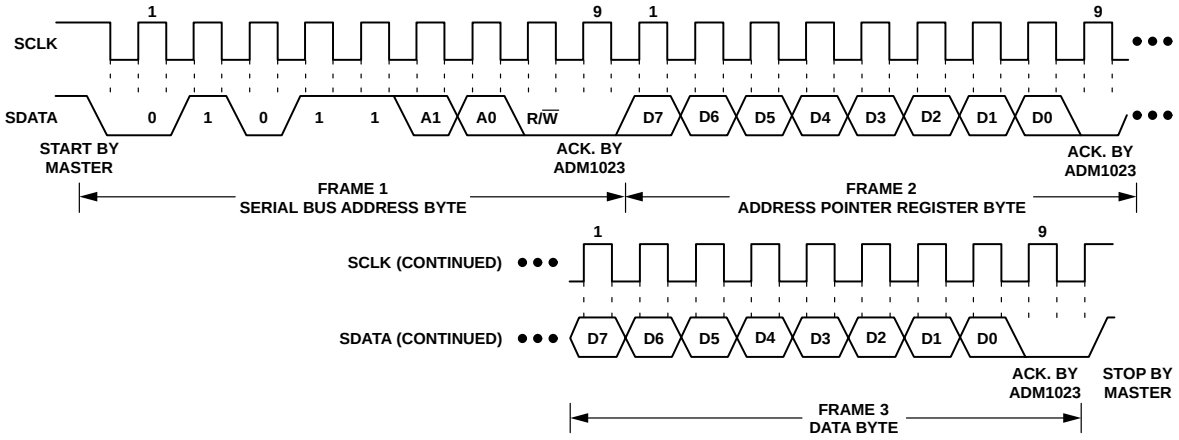


Figure 15. Writing a Register Address to the Address Pointer Register, then Writing Data to the Selected Register

Figure 16. Writing to the Address Pointer Register Only

One or more $\overline{\text{ALERT}}$ outputs are connected to a common $\overline{\text{SMBALERT}}$ line connected to the master. When the $\overline{\text{SMBALERT}}$ line is pulled low by one of the devices, the procedure shown in Figure 18 occurs.

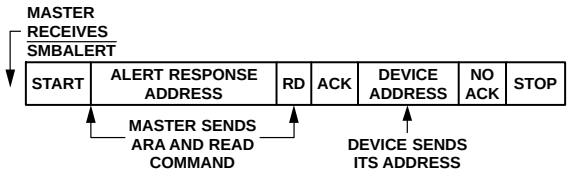


Figure 18. Use of $\overline{\text{SMBALERT}}$

$\overline{\text{SMBALERT}}$ Process

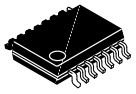
1. $\overline{\text{SMBALERT}}$ pulled low.
2. Master initiates a read operation and sends the alert response address (ARA = 0001 100). This is a general call address that must not be used as a specific device address.
3. The device whose $\overline{\text{ALERT}}$ output is low responds to the ARA and 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 $\overline{\text{ALERT}}$ output is low, the one with the lowest device address has priority, in accordance with normal SMBus arbitration.
5. Once the ADM1023 has responded to the ARA, it resets its $\overline{\text{ALERT}}$ output, provided that the error condition that caused the $\overline{\text{ALERT}}$ no longer exists. If the $\overline{\text{SMBALERT}}$ line remains low, the master sends ARA again, and so on until all devices whose $\overline{\text{ALERT}}$ outputs were low have responded.

Low Power Standby Modes

The ADM1023 can be put into a low power standby mode using hardware or software, that is, by taking the $\overline{\text{STBY}}$ input low or by setting Bit 6 of the configuration register. When $\overline{\text{STBY}}$ is high or Bit 6 is low, the ADM1023 operates

Table 13. ORDERING INFORMATION

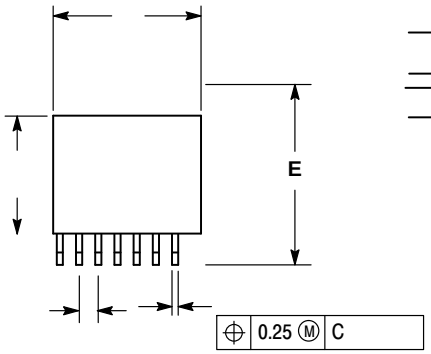
Device Number	Temperature Range	Package Type	Package Option	Shipping
		ĭ	–	

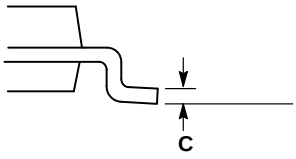


SCALE 2:1

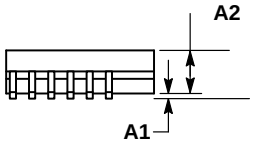
QSOP16
CASE 492−01
ISSUE A

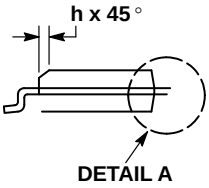
DATE 23 MAR 2011





DETAIL A





DETAIL A

NOTES:

DIM	INCHES MIN	MA
A	0.053	0.069
A1	0.004	0.010
	0.008	0.012
\	0.007	0.010
	0.025 BSC	
	0.009	0.020
L	0.016	0.050
M	0	8