General Description

The DS1624 consists of two separate functional units: a 256-byte nonvolatile E^2 memory and a direct-to-digital temperature sensor.

The nonvolatile memory is made up of 256 bytes of E^2 memory. This memory can be used to store any type of information the user wishes. These memory locations are accessed through the 2-wire serial bus.

The direct-to-digital temperature sensor allows the DS1624 to measure the ambient temperature and report the temperature in a 12-bit word with 0.0625°C resolution. The temperature sensor and its related registers are accessed through the 2-wire serial interface. Figure 1 shows a block diagram of the DS1624.

Benefits and Features

- Reduces Component Count with Integrated Temperature Sensor and Nonvolatile E^2 Memory
  - Measures Temperatures from -55°C to +125°C in 0.0625°C Increments
  - ±0.5°C Accuracy from 0°C to 70°C
  - 256 Bytes of E^2 Memory for Storing Information Such as Frequency Compensation Coefficients
  - No External Components
- Easy-to-Use 2-Wire Serial Interface
  - Temperature is Read as a 12-Bit Value (2-Byte Transfer)
- Available in 8-Pin SO and DIP Packages

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

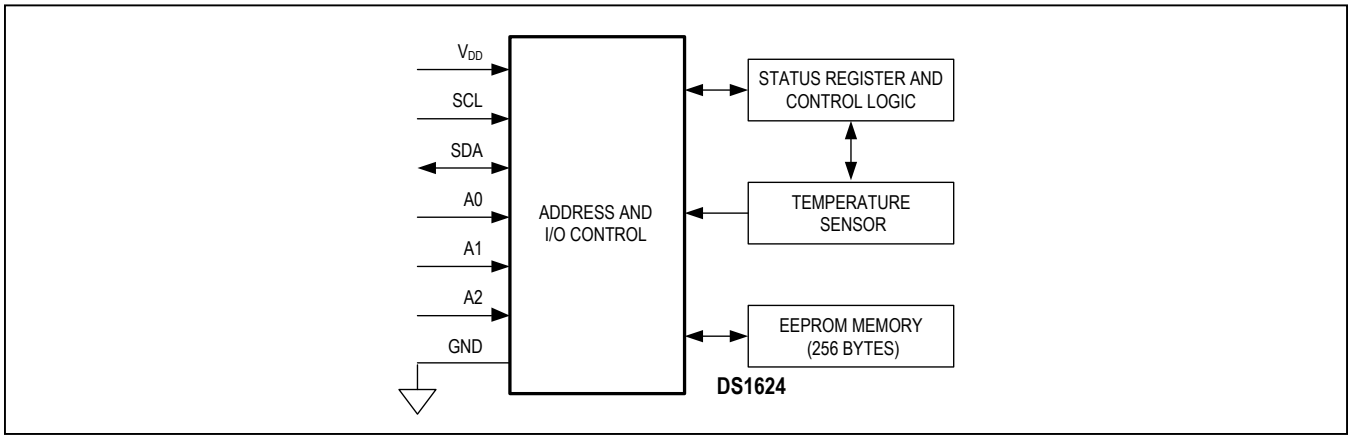


Figure 1. Block Diagram

maxim integrated™

Absolute Maximum Ratings

Voltage Range on Any Pin Relative to Ground ....-0.5V to +6.0V
Continuous Power Dissipation (T_A = +70°C)
PDIP (derate 9.10mW/°C above +70°C).....727.30mW
Operating Temperature Range..... -55°C to +125°C
Storage Temperature Range..... -55°C to +125°C
Soldering Temperature (reflow).....+260°C
Lead Temperature (soldering, 10s) .....+300°C

This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operation sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods of time may affect device reliability.

Package Thermal Characteristics (Note 1)

PDIP
Junction-to-Ambient Thermal Resistance (θ_{JA}) ..... 110°C/W
Junction-to-Case Thermal Resistance (θ_{JC}).....40°C/W

Note 1: Package thermal resistances were obtained using the method described in JEDEC specification JESD51-7, using a four-layer board. For detailed information on package thermal considerations, refer to **www.maximintegrated.com/thermal-tutorial**.

Recommended Operating Conditions

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Supply Voltage	V_{DD}	(Note 2)	2.7	5.0	5.5	V

DC Electrical Characteristics

(V_{DD} = 2.7V to 5.5V, T_A = -55°C to +125°C, unless otherwise noted.) (Note 3)

PARAMETER	SYMBOL	CONDITIONS	MIN	TYP	MAX	UNITS
Thermometer Error	T_{ERR}	0°C to +70°C			±0.5	°C
		-55°C to +125°C			±2.0	
Thermometer Resolution		12-bit		0.0625		°C
Low-Level Input Voltage	V_{IL}		-0.3		0.3 x V_{DD}	V
High-Level Input Voltage	V_{IH}		0.7 x V_{DD}		V_{DD} + 0.3	V
Pulse Width of Spikes That Must Be Suppressed by the Input Filter	t_{SP}	Fast mode	0		50	ns
Low-Level Output Voltage (SDA)	V_{OL1}	3mA sink current (Note 2)	0		0.4	V
	V_{OL2}	6mA sink current (Note 2)	0		0.6	
Input Current Each I/O Pin		0.4 < $V_{I/O}$ < 0.9V_{DD} (Note 4)	-1		+1	µA
I/O Capacitance	$C_{I/O}$				10	pF
Active Supply Current	I_{CC}	Temperature conversion			1250	µA
		E^2 write (Notes 5, 6)			400	
		Communication only			125	
Standby Supply Current	I_{STBY}	(Notes 5, 6, 7)		1	3	µA

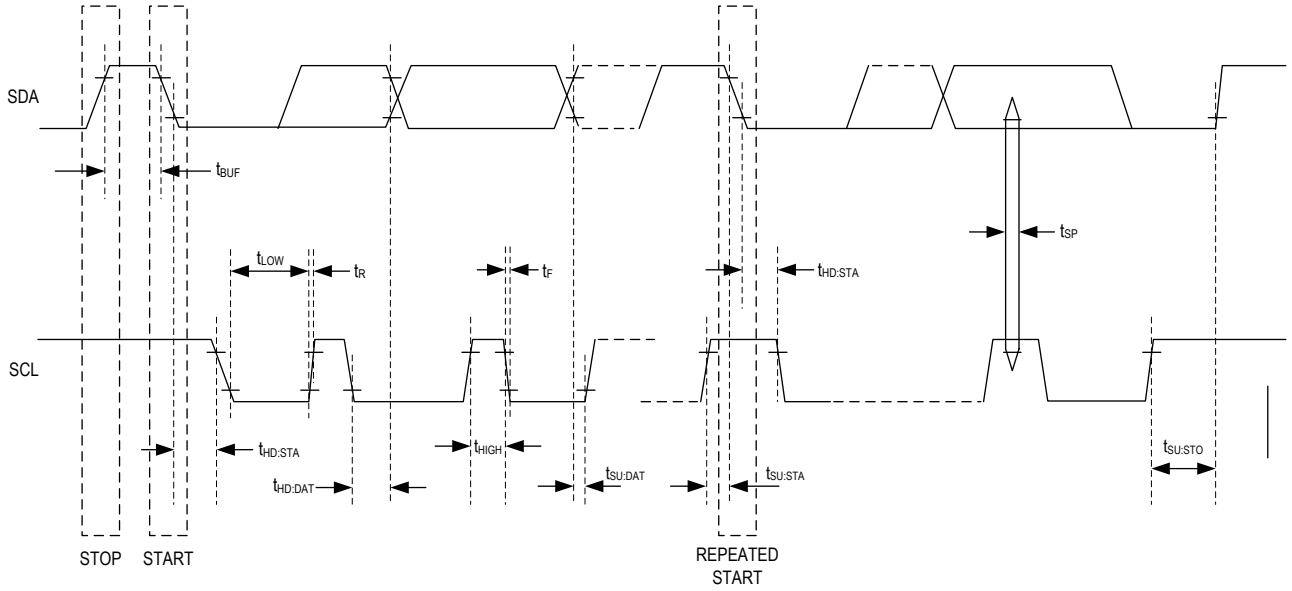
AC Electrical Characteristics

(V_{DD} = 2.7V to 5.5V, T_A = -55°C to +125°C, unless otherwise noted. All values referred to V_{IH} = 0.9V_{DD} and V_{IL} = 0.1V_{DD}.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
Temperature Conversion Time	t_{TC}					200	ms
EEPROM Write Cycle Time	t_{WR}	0°C to +70°C (Note 8)				50	ms
EEPROM Endurance	N_{EEWR}	-20°C to +70°C	(Note 9)	10k	20k		Write Cycles
		T_A = +25°C		40k	80k		
EEPROM Data Retention	t_{EEDR}	-40°C to +70°C		10	20		Years
SLK Clock Frequency	f_{SCL}	Fast mode	(Note 10)	0		400	kHz
		Standard mode		0		100	
Bus Free Time Between a STOP and START Condition	t_{BUF}	Fast mode	(Note 10)	1.3			µs
		Standard mode		4.7			
Hold Time (Repeated) START Condition	$t_{HD:STA}$	Fast mode	(Notes 10, 11)	0.6			µs
		Standard mode		4.0			
Low Period of SCL Clock	t_{LOW}	Fast mode	(Note 10)	1.3			µs
		Standard mode		4.7			
High Period of SCL Clock	t_{HIGH}	Fast mode	(Note 10)	0.6			µs
		Standard mode		4.0			
Setup Time for a Repeated START Condition	$t_{SU:STA}$	Fast mode	(Note 10)	0.6			µs
		Standard mode		4.7			
Data Hold Time	$t_{HD:DAT}$	Fast mode	(Note 10)	0		0.9	µs
		Standard mode		0		0.9	
Data Setup Time	$t_{SU:DAT}$	Fast mode	(Notes 10, 11, 12)	100			ns
		Standard mode		250			
Rise Time of Both SDA and SCL Signals	t_R	Fast mode	(Notes 8, 10, 12)	20 + 0.1C_B		300	ns
		Standard mode		20 + 0.1C_B		1000	
Fall Time of Both SDA and SCL Signals	t_F	Fast mode	(Notes 8, 10, 12)	20 + 0.1C_B		300	ns
		Standard mode		20 + 0.1C_B		300	
Setup Time for STOP Condition	$t_{SU:STO}$	Fast mode	(Note 10)	0.6			µs
		Standard mode		4.0			
Capacitive Load for Each Bus Line	C_B					400	pF
Input Capacitance	C_I			5			pF

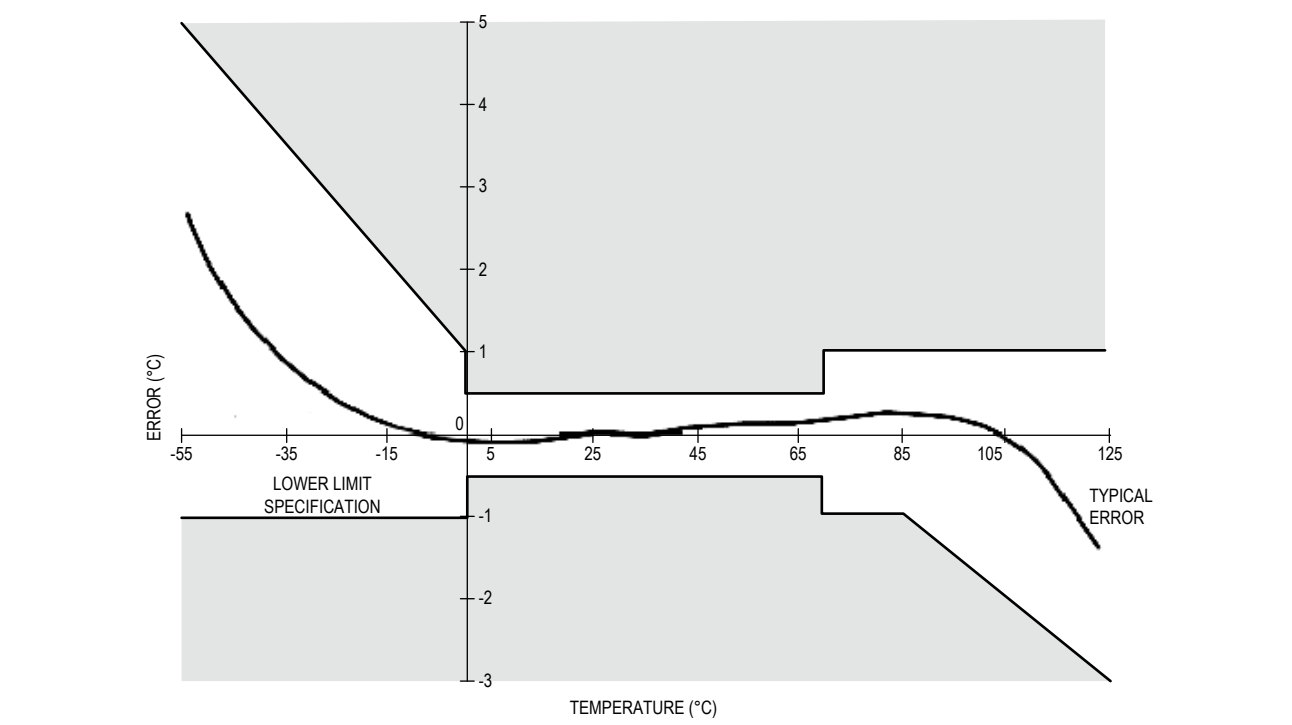
Note 2: All voltages are referenced to ground.
Note 3: Limits are 100% production tested at T_A = +25°C and/or T_A = +85°C. Limits over the operating temperature range and relevant supply voltage are guaranteed by design and characterization.
Note 4: I/O pins of fast mode devices must not obstruct the SDA and SCL lines if V_{DD} is switched off.
Note 5: I_{CC} specified with SDA pin open.
Note 6: I_{CC} specified with V_{CC} at 5.0V and SDA, SCL = 5.0V, 0°C to +70°C.
Note 7: EEPROM inactive, temperature sensor in shutdown mode.
Note 8: For example, if CB = 300pF, then $t_{R(MIN)}$ = $t_{F(MIN)}$ = 50ns.
Note 9: Write occurs between 0°C and +70°C.
Note 10: See the timing diagram (Figure 2). All timing is referenced to 0.9V_{DD} and 0.1V_{DD}.
Note 11: After this period, the first clock pulse is generated.
Note 12: A fast mode device can be used in a standard mode system, but the requirement $t_{SU:DAT}$ ≥ 250ns must then be met. This is automatically the case if the device does not stretch the low period of the SCL signal. If such a device does stretch the low period of the SCL signal, it must output the next data bit to the SDA line $t_{R(MAX)}$ + $t_{SU:DAT}$ = 1000 + 250 = 1250ns before the SCL line is released.

Timing Diagram

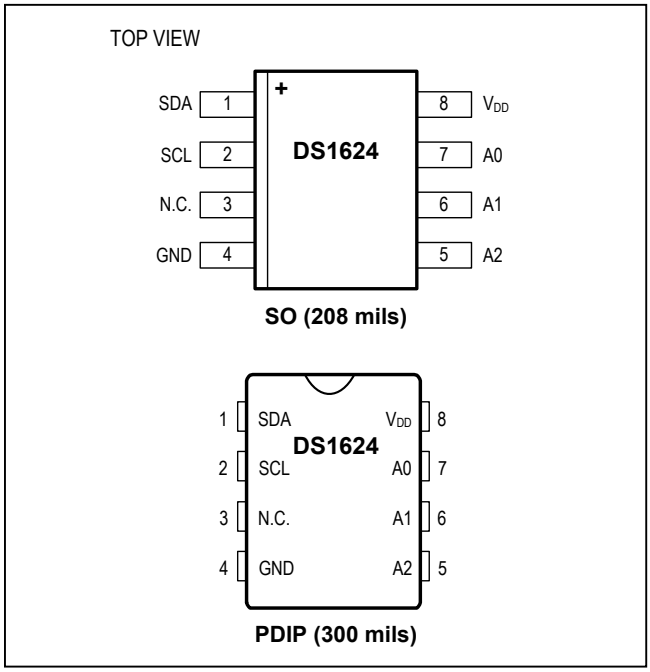


NOTE: THE DS1624 DOES NOT DELAY THE SDA LINE INTERNALLY WITH RESPECT TO SCL FOR ANY LENGTH OF TIME

Typical Performance Curve



Pin Configurations



Pin Description

PIN	NAME	FUNCTION
1	SDA	Data Input/Output Pin for 2-Wire Serial Communication Port
2	SCL	Clock Input/Output Pin for 2-Wire Serial Communication Port
3	N.C.	No Connection. No Internal Connection.
4	GND	Ground
5	A2	Address Input
6	A1	Address Input
7	A0	Address Input
8	V_{DD}	2.7V to 5.5V Input Power-Supply Voltage

Detailed Description

2-Wire Serial Data Bus

The DS1624 supports a bidirectional two-wire bus and data transmission protocol. A device that sends data onto the bus is defined as a transmitter, and a device receiving data as a receiver. The device that controls the message is called a “master.” The devices that are controlled by the master are “slaves.” The bus must be controlled by a master device which generates the serial clock (SCL), controls the bus access, and generates the START and STOP conditions. The DS1624 operates as a slave on the two-wire bus. Connections to the bus are made via the open-drain I/O lines SDA and SCL. The following bus protocol has been defined (see Figure 2):

- Data transfer can be initiated only when the bus is not busy.
- During data transfer, the data line must remain stable whenever the clock line is high. Changes in the data line while the clock line is high are interpreted as control signals.

Accordingly, the following bus conditions have been defined:

Bus Not Busy: Both data and clock lines remain high.

Start Data Transfer: A change in the state of the data line, from high to low, while the clock is high, defines a START condition.

Stop Data Transfer: A change in the state of the data line, from low to high, while the clock line is high, defines the STOP condition.

Data Valid: The state of the data line represents valid data when, after a START condition, the data line is stable for the duration of the high period of the clock signal. The data on the line must be changed during the low period of the clock signal. There is one clock pulse per bit of data.

Each data transfer is initiated with a START condition and terminated with a STOP condition. The number of data bytes transferred between START and STOP conditions is not limited, and is determined by the master device.

The information is transferred byte-wise and each receiver acknowledges with a ninth bit.

Within the bus specifications a standard mode (100kHz clock rate) and a fast mode (400kHz clock rate) are defined. The DS1624 works in both modes.

Acknowledge: Each receiving device, when addressed, is obliged to generate an acknowledge after the reception of each byte. The master device must generate an extra clock pulse, which is associated with this acknowledge bit.

A device that acknowledges must pull down the SDA line during the acknowledge clock pulse in such a way that the SDA line is stable low during the high period of the acknowledge related clock pulse. Of course, setup and hold times must be taken into account. A master must signal an end of data to the slave by not generating an acknowledge bit on the last byte that has been clocked out of the slave. In this case, the slave must leave the data line high to enable the master to generate the STOP condition.

Figure 2 details how data transfer is accomplished on the two-wire bus. Depending upon the state of the R/$\overline{W}$ bit, two types of data transfer are possible:

1. **Data transfer from a master transmitter to a slave receiver**. The first byte transmitted by the master is the slave address. Next follows a number of data bytes. The slave returns an acknowledge bit after each received byte.
2. **Data transfer from a slave transmitter to a master receiver**. The first byte (the slave address) is transmitted by the master. The slave then returns an acknowledge bit. Next follows a number of data bytes transmitted by the slave to the master. The master returns an acknowledge bit after all received bytes other than the last byte. At the end of the last received byte, a ‘not acknowledge’ is returned.

The master device generates all of the serial clock pulses and the START and STOP conditions. A transfer is ended with a STOP condition or with a repeated START condition. Since a repeated START condition is also the beginning of the next serial transfer, the bus is not released.

The DS1624 can operate in the following two modes:

1. **Slave receiver mode:** Serial data and clock are received through SDA and SCL. After each byte is received an acknowledge bit is transmitted. START and STOP conditions are recognized as the beginning and end of a serial transfer. Address recognition is performed by hardware after reception of the slave address and direction bit.
2. **Slave transmitter mode:** The first byte is received and handled as in the slave receiver mode. However, in this mode the direction bit indicates that the transfer direction is reversed. Serial data is transmitted on SDA by the DS1624 while the serial clock is input on SCL. START and STOP conditions are recognized as the beginning and end of a serial transfer.

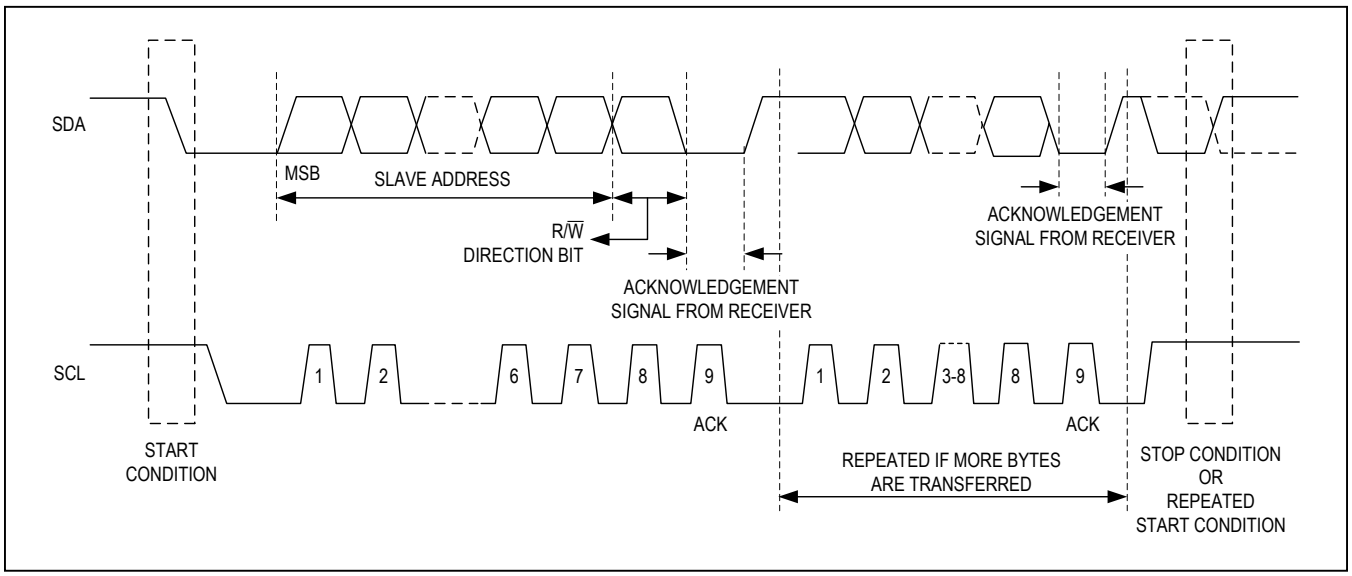


Figure 2. Data Transfer on 2-Wire Serial Bus

Slave Address

A control byte is the first byte received following the START condition from the master device. The control byte consists of a four bit control code; for the DS1624, this is set as 1001 binary for read and write operations. The next three bits of the control byte are the device select bits (A2, A1, A0). They are used by the master device to select which of eight devices are to be accessed. These bits are in effect the three least significant bits of the slave address. The last bit of the control byte (R/$\overline{W}$) defines the operation to be performed. When set to a "1", a read operation is selected, when set to a "0", a write operation is selected. Following the START condition the DS1624 monitors the SDA bus checking the device type identifier being transmitted. Upon receiving the 1001 code and appropriate device select bits, the slave device outputs an acknowledge signal on the SDA line.

Measuring Temperature

Figure 1 shows a block diagram of the DS1624. The DS1624 measures the temperature using a bandgap-based temperature sensor. A delta-sigma analog-to-digital (ADC) converts the temperature to a 12-bit digital value that is calibrated in °C; for °F applications a lookup table or conversion routine must be used. Throughout this data sheet the term "conversion" is used to refer to the entire temperature measurement and ADC sequence.

The temperature reading is stored as a 16-bit two's complement number in the 2-byte temperature register as shown in Figure 4.

Since data is transmitted over the 2-wire bus MSB first, temperature data can be written to/read from the DS1624 as either a single byte (with temperature resolution of 1°C) or as 2 bytes, the second byte containing the value of the four least significant bits of the temperature reading as shown in Figure 4. Note that the remaining 4 bits of this byte are set to all zeros.

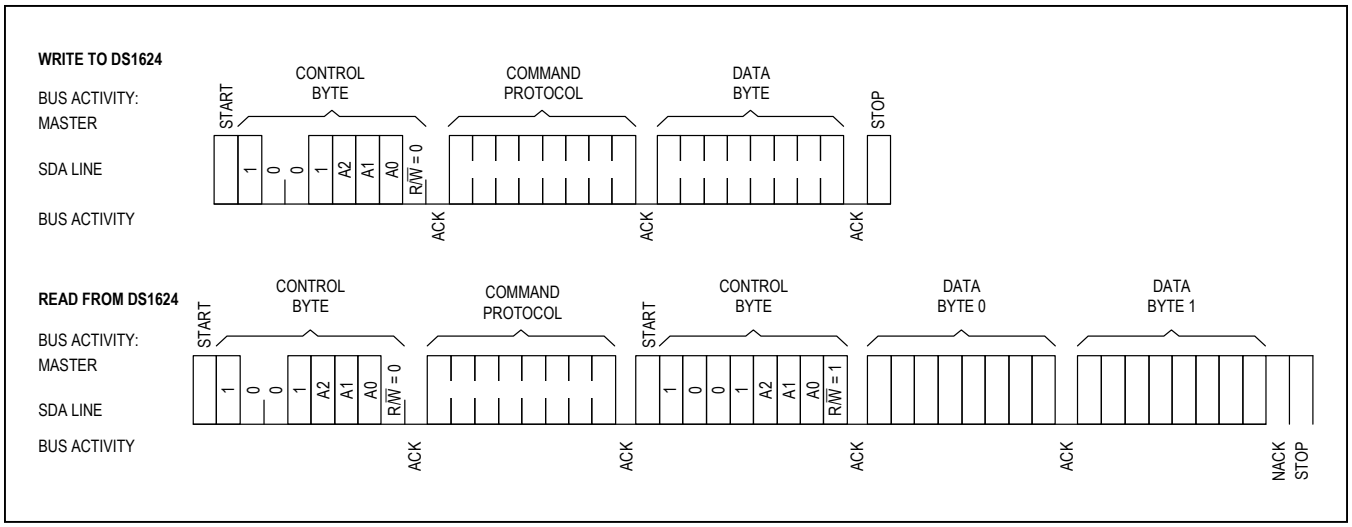


Figure 3. 2-Wire Serial Communication with DS1624

	BIT 15	BIT 14	BIT 13	BIT 12	BIT 11	BIT 10	BIT 9	BIT 8
MS BYTE	S	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	BIT 7	**BIT 6**	**BIT 5**	**BIT 4**	**BIT 3**	**BIT 2**	**BIT 1**	**BIT 0**
LS BYTE	2^{-1}	2^{-2}	2^{-3}	2^{-4}	0	0	0	0

Figure 4. Temperature Register Format

Table 1. Temperature/Data Relationships

TEMP (°C)	DIGITAL OUTPUT (BINARY)	DIGITAL OUTPUT (HEX)
+125	01111101 00000000	7D00h
+25.0625	00011001 00010000	1910h
+0.5	00000000 10000000	0080h
0	00000000 00000000	0000h
-0.5	11111111 10000000	FF80h
-25.0625	11100110 11110000	E6F0h
-55	11001001 00000000	C900h

Temperature is represented in the DS1624 in terms of a 0.0625°C LSB, yielding the following 12-bit example:

MSB								LSB							
0	0	0	1	1	0	0	1	0	0	0	1	0	0	0	0

= +25.0625°C

Operation and Control

A configuration/status register is used to determine the method of operation the DS1624 will use in a particular application as well as indicating the status of the temperature conversion operation.

The configuration register is defined as follows:

CONFIGURATION/STATUS REGISTER

DONE	0	0	0	1	0	1	1SHOT

where:

DONE = Conversion Done bit, “1” = Conversion complete, “0” = conversion in progress.

1SHOT = One Shot Mode. If 1SHOT is “1,” the DS1624 performs one temperature conversion upon receipt of the Start Convert T protocol. If 1SHOT is “0,” the DS1624 continuously performs temperature conversions. This bit is nonvolatile and the DS1624 is shipped with 1SHOT = “0.”

Since the configuration register is implemented in E^2, writes to the register require 10ms to complete. After issuing a command to write to the configuration register, no further accesses to the DS1624 should be made for at least 10ms.

Memory

Byte Program Mode

In this mode, the master sends addresses and one data byte to the DS1624.

Following a START condition, the device code (4-bit), the slave address (3-bit), and the R/$\overline{W}$ bit (which is logic-low) are placed onto the bus by the master. The master then sends the Access Memory protocol. This indicates to the addressed DS1624 that a byte with a word address will follow after it has generated an acknowledge bit. Therefore, the next byte transmitted by the master is the word address and will be written into the address pointer of the DS1624. After receiving the acknowledge of the DS1624, the master device transmits the data word to be written into the addressed memory location. The DS1624 acknowledges again and the master generates a STOP condition. This initiates the internal programming cycle of the DS1624. A repeated START condition, instead of a STOP condition, will abort the programming operation.

During the programming cycle the DS1624 does not acknowledge any further accesses to the device until the programming cycle is complete (no longer than 50ms.)

Page Program Mode

To program the DS1624 the master sends addresses and data to the DS1624, which is the slave. This is done by supplying a START condition followed by the 4-bit device code, the 3-bit slave address, and the R/$\overline{W}$ bit which is defined as a logic-low for a write. The master then sends the Access Memory protocol. This indicates to the addressed slave that a word address will follow. The slave outputs the acknowledge pulse to the master during the ninth clock pulse. When the word address is received by the DS1624 it is placed in the address pointer defining which memory location is to be written. The DS1624 generates an acknowledge after every 8 bits received and store them consecutively in an 8-byte RAM until a STOP condition is detected which initiates the internal programming cycle.

A repeated START condition, instead of a STOP condition, aborts the programming operation. During the programming cycle the DS1624 does not acknowledge any further accesses to the device until the programming cycle is complete (no longer than 50ms).

If more than 8 bytes are transmitted by the master, the DS1624 rolls over and overwrites the data beginning with the first received byte. This does not affect erase/write cycles of the EEPROM array and is accomplished as a result of only allowing the address register's bottom 3 bits to increment while the upper 5 bits remain unchanged. The DS1624 is capable of 20,000 writes (25,000 erase/write cycles) before EEPROM wear out can occur.

If the master generates a STOP condition after transmitting the first data word, byte programming mode is entered.

Read Mode

In this mode, the master is reading data from the DS1624 E^2 memory. The master first provides the slave address to the device with R/$\overline{W}$ set to "0." The master then sends the Access Memory protocol and, after receiving an acknowledge, then provides the word address, which is the address of the memory location at which it wishes to begin reading. Note that while this is a read operation the address pointer must first be written. During this period the DS1624 generates acknowledge bits as defined in the appropriate section.

The master now generates another START condition and transmits the slave address. This time the R/$\overline{W}$ bit is set to "1" to put the DS1624 in read mode. After the DS1624 generates the acknowledge bit it outputs the data from the addressed location on the SDA pin, increments the address pointer, and, if it receives an acknowledge from the master, transmits the next consecutive byte. This auto-increment sequence is only aborted when the master sends a STOP condition instead of an acknowledge. When the address pointer reaches the end of the 256-byte memory space (address FFh) it increments from the end of the memory back to the first location of the memory (address 00h).

Command Set

Data and control information is read from and written to the DS1624 in the format shown in Figure 3. To write to the DS1624, the master issues the slave address of the DS1624 and the R/$\overline{W}$ bit is set to 0. After receiving an acknowledge the bus master provides a command protocol. After receiving this protocol the DS1624 issues an acknowledge, and then the master can send data to the DS1624. If the DS1624 is to be read, the master must send the command protocol as before then issue a repeated START condition and the control byte again, this time with the R/$\overline{W}$ bit set to 1 to allow reading of the data from the DS1624. The command set for the DS1624 as shown in Table 2 is as follows.

Table 2. DS1624 Command Set

INSTRUCTION	DESCRIPTION	PROTOCOL	2-WIRE BUS DATA AFTER ISSUING PROTOCOL
TEMPERATURE CONVERSION COMMANDS			
Read Temperature	Reads last converted temperature value from temperature register	AAh	<read 2 bytes data>
Start Convert T	Initiates temperature conversion (Note 1)	EEh	idle
Stop Convert T	Halts temperature conversion (Note 1)	22h	idle
THERMOSTAT COMMANDS			
Access Memory	Reads or writes to 256-byte EEPROM memory (Note 2)	17h	<write data>
Access Config	Reads or writes configuration data to configuration register (Note 2)	ACh	<write data>

Note 1: In continuous conversion mode, a Stop Convert T command halta continuous conversion. To restart, the Start Convert T command must be issued. In one-shot mode, a Start Convert T command must be issued for every temperature reading desired.

Note 2: Writing to the E^2 typically requires 10ms at room temperature. After issuing a write command, no further reads or writes should be requested for at least 10ms.

Access Memory [17h]

This command instructs the DS1624 to access its E^2 memory. After issuing this command, the next data byte is the value of the word address to be accessed. See the *Memory* section for detailed explanations of the use of this protocol and data format following it.

Access Config [ACh]

If R/$\overline{W}$ is “0”, this command writes to the configuration register. After issuing this command, the next data byte is the value to be written into the configuration register. If R/$\overline{W}$ is “1,” the next data byte read is the value stored in the configuration register.

Read Temperature [AAh]

This command reads the last temperature conversion result. The DS1624 sends 2 bytes in the format described earlier, which are the contents of this register.

Start Convert T [EEh]

This command begins a temperature conversion. No further data is required. In one-shot mode, the temperature conversion is performed and then the DS1624 remain sidle. In continuous mode, this command initiates continuous conversions.

Stop Convert T [22h]

This command stops temperature conversion. No further data is required. This command can be used to halt a DS1624 in continuous conversion mode. After issuing this command, the current temperature measurement is completed, then the DS1624 remains idle until a Start Convert T is issued to resume continuous operation.

During the programming cycle, the DS1624 does not acknowledge any further accesses to the device until the programming cycle is complete (no longer than 50ms).

Memory Function Example

BUS MASTER MODE	DS1624 MODE	DATA (MSB FIRST)	COMMENTS
{Command protocol for configuration register} {Start here}			
Tx	Rx	START	Bus master initiates a START condition.
Tx	Rx	<cadr,0>	Bus master sends DS1624 address; R/$\overline{W}$ = 0.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	ACh	Bus master sends Access Config command protocol.
Rx	Tx	ACK	DS1624 generates acknowledge bit (Note 1).
Tx	Rx	00h	Bus master sets up DS1624 for continuous conversion.
Rx	Tx	ACK	DS1624 generates acknowledge bit (Notes 2, 4).
Tx	Rx	STOP	Bus master initiates the STOP condition.
{Command protocol for Start Convert T} {Start here}			
Tx	Rx	START	Bus master initiates a START condition.
Tx	Rx	<cadr,0>	Bus master sends DS1624 address; R/$\overline{W}$ = 0.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	EEh	Bus master sends Start Convert T command protocol.
Rx	Tx	ACK	DS1624 generates acknowledge bit (Note 1).
Tx	Rx	STOP	Bus master initiates the STOP condition.
{Command protocol for reading the Temperature} {Start here}			
Tx	Rx	START	Bus master initiates a START condition.
Tx	Rx	<cadr,0>	Bus master sends DS1624 address; R/$\overline{W}$ = 0.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	AAh	Bus master sends Read Temp command protocol.
Rx	Tx	ACK	DS1624 generates acknowledge bit (Note 1).
Tx	Rx	START	Bus master initiates a repeated START condition.
Tx	Rx	<cadr,1>	Bus Master sends DS1624 address; R/$\overline{W}$ = 1.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Rx	Tx	<data>	DS1624 sends the MSB byte of Temperature.
Tx	Rx	ACK	Bus master generates acknowledge bit.
Rx	Tx	<data>	DS1624 sends the LSB byte of Temperature.
Tx	Rx	NACK	Bus master sends not-acknowledge bit.
Tx	Rx	STOP	Bus master initiates the STOP condition.
{Command protocol for writing to EEPROM} {Start here}			
Tx	Rx	START	Bus master initiates a START condition.
Tx	Rx	<cadr,0>	Bus master sends DS1624 address; R/$\overline{W}$ = 0.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	17h	Bus master sends Access Memory command protocol.
Rx	Tx	ACK	DS1624 generates acknowledge bit (Note 1).

Memory Function Example (continued)

BUS MASTER MODE	DS1624 MODE	DATA (MSB FIRST)	COMMENTS
Tx	Rx	<madr>	Bus master sets the starting memory address.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	<data>	Bus master sends the first byte of data.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	<data>	Bus master sends the second byte of data.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
. . .	. . .	. . .	. . .
Tx	Rx	<data>	Bus master sends the nth byte of data (Note 3).
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	STOP	Bus master initiates the STOP condition (Notes 2, 4)
{Command protocol for reading from EEPROM} {Start here}			
Tx	Rx	START	Bus master initiates a START condition.
Tx	Rx	<cadr,0>	Bus master sends DS1624 address; R/$\overline{W}$ = 0.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	17h	Bus master sends Access Memory command protocol.
Rx	Tx	ACK	DS1624 generates acknowledge bit (Note 1).
Tx	Rx	<madr>	Bus master sends the starting memory address.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Tx	Rx	START	Bus master initiates a repeated START condition.
Tx	Rx	<cadr,1>	Bus master sends DS1624 address; R/$\overline{W}$ = 1.
Rx	Tx	ACK	DS1624 generates acknowledge bit.
Rx	Tx	<data>	DS1624 sends the first byte of data.
Tx	Rx	ACK	Bus master generates acknowledge bit.
Rx	Tx	<data>	DS1624 sends the second byte of data.
Tx	Rx	ACK	Bus master generates acknowledge bit.
. . .	. . .	. . .	. . .
Rx	Tx	<data>	DS1624 sends the nth byte of data (Note 5).
Tx	Rx	NACK	Bus master sends not-acknowledge bit.
Tx	Rx	STOP	Bus master initiates the STOP condition.

Note 1: If this protocol follows a write and the DS1624 does not acknowledge here, restart the protocol at the START here. If it does acknowledge, continue on.

Note 2: Wait for write to complete (50ms max). If DS1624 does not acknowledge the command protocol immediately following a configure register or write memory protocol, the DS1624 has not finished writing. Restart the new command protocol until the DS1624 acknowledges.

Note 3: If n is greater than eight, the last 8 bytes are the only bytes saved in memory. If the starting address is 00 and the incoming data is 00 11 22 33 44 55 66 77 88 99, the result is mem00=88 mem01=99 mem02=22 mem03=33 mem04=44 mem05=55 mem06=66 mem07=77. The data wraps around and overwrites itself.

Note 4: The STOP condition causes the DS1624 to initiate the write to EEPROM sequence. If a START condition comes instead of the STOP condition, the write is aborted. The data is not saved.

Note 5: For reading, the address is incremented. If the starting address is 04h and 30 bytes of data are read out, 21h is the final address read.

Ordering Information

PART	TOP MARK	PIN-PACKAGE
DS1624+	DS1624	8 PDIP (300 mils)
DS1624S+	DS1624S	8 SO (208 mils)
DS1624S+T&R	DS1624S	8 SO (208 mils) (2000 pieces)

Note: *All devices rated for the -55°C to +125°C operating temperature range.*

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

T&R = Tape and reel.

Package Information

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

PACKAGE TYPE	PACKAGE CODE	OUTLINE NO.	LAND PATTERN NO.
8 PDIP	P8+4	21-0043	—
8 SO	W8+2	21-0262	90-0258

Revision History

REVISION NUMBER	REVISION DATE	DESCRIPTION	PAGES CHANGED
0	5/12	Updated *Ordering Information* and *Package Information*; updated the soldering information in the *Absolute Maximum Ratings* section	1, 2, 7, 14, 18
1	12/13	Updated the *Features* and *Description*, removed the *Overview* section; replaced the *Operation—Measuring Temperature* section and Figure 4; added the *Package Thermal Characteristics* section; updated the *DC Electrical Characteristics* and *AC Electrical Characteristics* tables and related notes	1, 2, 5, 13, 14, 15
2	3/14	Updated the EEPROM Data Retention parameter in the *AC Electrical Characteristics* table	14
3	5/14	Updated the *Electrical Characteristics*	13, 14
4	1/15	Updated *Benefits and Features* sections	1
5	8/15	Updated *AC Electrical Characteristics*	3