High Precision Digital Thermometer M601/M1601/M1820/MTS01

Datasheet

(V4.4)

2022/09

Table of contents

1. Overview

M601, M1601, M1820, and MTS01 is MinYuan fourth generation of digital high-precision temperature sensor with a typical temperature measurement accuracy of 0.1℃. The temperature sensing principle is based on the characteristic relationship between the semiconductor PN-junction temperature and voltage, through small signal amplification, analog to digital conversion and digital calibration, the 16-bit digital temperature data can be easily accessed. It has strength of high accuracy and consistency, long lifetime, low power consumption, programmable configuration flexibility. The thermometer integrates an EEPROM as a storage for ROM ID of the sensors, temperature alarm threshold, temperature calibration correction values and user-defined information such as sensor node information. Each sensor IC has a unique 64-bit serial number as ROM ID, and is fully calibrated before ship out. The IC have one-wire or I2C interface for the host processor to access. One-wire interface fits for cost-effective multiple nodes access through one data bus in a long cable. The search for the IC’ s ROM ID, temperature data memory access and functional configuration can be easily controlled by one GPIO of the microprocessor. I^2C interface is suitable for high-speed board-level data communications, with the maximum interface speed up to 400kHz.

High-precision temperature measurement region can be customized upon requirement. For example, for wearable thermometer and cold chain logistics range, the in-fab calibration procedure is different. Therefore the product series is divided into classes of 0.1℃/0.5°C for different working conditions and precision requirement at different price, as shown below.

Part Number	High precision temperature range	Typical Accuracy(℃)	Package (mm)	Interface
M601	+28°C to +43°C	±0.1	DFN8(2X2X0.55)	One-wire
M1601			SOT23	One-wire
MTS01			DFN8(2.5X2.5X0.7)	One-wire and I^2C
M601Z	0℃ to +50℃	±0.1	DFN8(2X2X0.55)	One-wire
M1601Z			SOT23	One-wire
M1820Z			TO92S	One-wire
MTS01Z			DFN8(2.5X2.5X0.7)	One-wire and I^2C
M601W	+20° to +70°C	±0.1	DFN8(2X2X0.55)	One-wire
M1601W			SOT23	One-wire
M1820W			TO92S	One-wire
MTS01W			DFN8(2.5X2.5X0.7)	One-wire and I^2C
M601P	-20℃ to +30℃	±0.1	DFN8(2X2X0.55)	One-wire
M1601P			SOT23	One-wire
M1820P			TO92S	One-wire
MTS01P			DFN8(2.5X2.5X0.7)	One-wire and I^2C
M601B	0°C to +50°C	±0.5	DFN8(2X2X0.55)	One-wire

M1601B			SOT23	One-wire
M1820B			TO92S	One-wire
MTS01B			DFN8(2.5X2.5X0.7)	One-wire and I^2C

2. Features

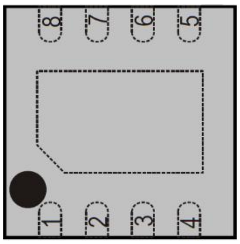
- Highest accuracy: ±0.1°C, (or ±0.5°C, accuracy in the product series)
- -70°C~+150°C working temperature range
- The standard to the series has a default 16-bit ADC output, with a maximum resolution of 0.004°C
- Configurable conversion time: 10.5ms/5.5ms/4ms
- Configurable single shot mode and periodic data acquisition mode
- User configurable alarm threshold
- 32-bit non-volatile storage space for user data
- Wide supply voltage range 1.8V-5.5V
- Typical standby power consumption is 0.1μA@3.3V, maximum peak power consumption is 0.45mA@3.3V and average temperature measurement current is 5.2 μA (@3.3V, 1 sample per second)
- Every thermometer has a unique ROM ID for easy network addressing
- One-wire and I2C interface

3. Applications

- Body and animal temperature measurement
- Electronic thermometer
- Medical Electronics
- Cold chain logistics
- Heat meter gas meter water meter
- Board level temperature monitoring
- Environmental temperature monitoring
- Smart home
- Consumer electronics
- Temperature measuring instrument

4. Package pin description

4.1. M601（DFN8，2X2X0.55mm）



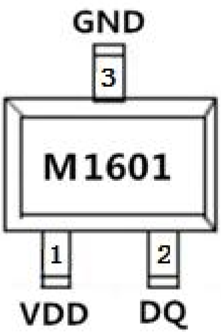
Front View

Physical Photo

NO.	Pin Name	I/O	Description
1	VDD	—	Power
2	ALERT	O	Alarm
3	DQ/SDA	IO	One-wire data line
4	NC	—	Floating
5	GND	—	Ground
6	NC	—	Floating
7	NC	—	Floating
8	NC	—	Floating
thermal pad	NC	—	Floating or grounded (1)

Note 1: When designing the circuit, it is recommended that the thermal pad is floating or grounded.
(1) If the IC is pasted on PCB, it is recommended to connect the thermal pad to ground;
(2) If the thermal pad is pasted with metal sheets for temperature detection, it is recommended to not connect the thermal pad to ground.

4.2. M1601（SOT23-3，2.9X2.8X1. 1mm）

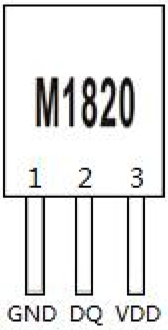


Front View

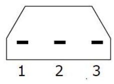
Physical Photo

No.	Pin Name	I/O	Description
1	VDD	—	Power
2	DQ	I/O	One-wire data line
3	GND	—	Ground

4.3. M1820 (TO92S)



Front View

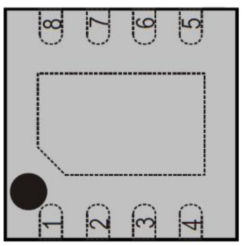


Bottom view


Physical Photo

No.	Pin Name	I/O	Description
1	GND	—	Ground
2	DQ	I/O	One-wire data line
3	VDD	—	Power

4.4. MTS01 （DFN8，2.5X2.5X0.7mm)



Front View

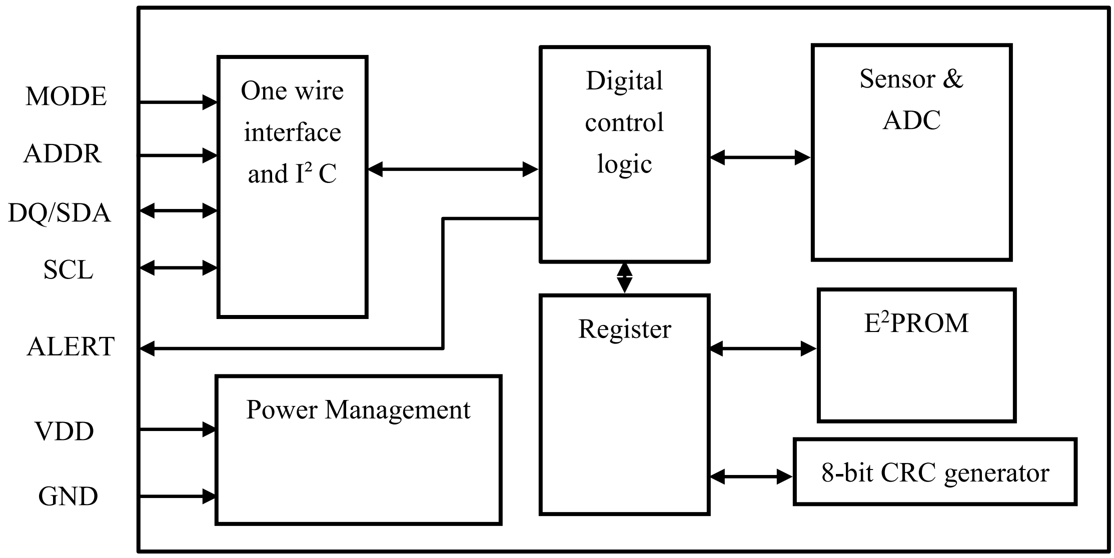
Physical Photo

No.	Pin Name	I/O	Description
1	SDA/DQ	I/O	I^2C / one-wire data line
2	ADDR	I	I^2C address selection，can’ t be floated
3	ALERT	O	Alarm
4	SCL	I/O	I^2C data line
5	VDD	—	Power

6	nRESET	I	Reset, low level is valid. When not in use, it can be floated or pulled up to VDD with resistance greater than 2K Ω
7	Mode	I	Interface mode selection, can not be suspended. When lowIn low, pin 1 is I2C SDA, pin 4 is I2C SCL; when high, pin 1 is DQ of single bus, pin 2 and pin 4 are invalid
8	GND	—	Ground
thermal pad	NC	—	Floating or grounded (1)

Note 1: When designing the circuit, it is recommended that the thermal pad is floating or grounded.
(1) If the IC is pasted on PCB, it is recommended to connect the thermal pad to ground;
(2) If the thermal pad is pasted with metal sheets for temperature detection, it is recommended to not connect the thermal pad to ground.

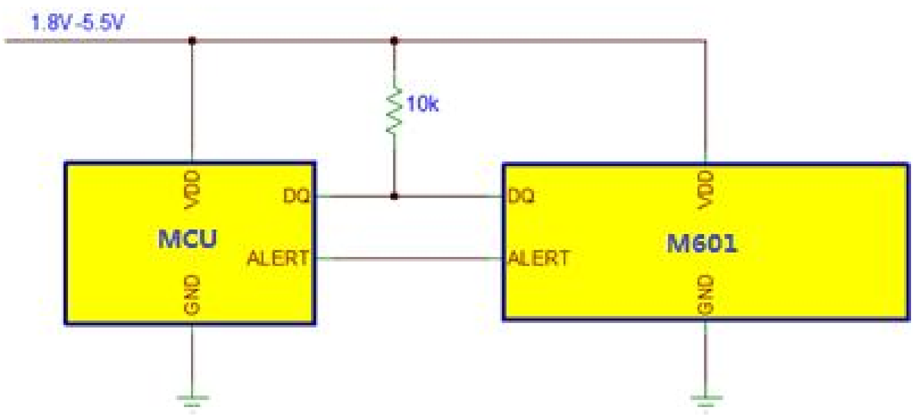
5. System Diagram



The figure above shows the schematic diagram of the temperature sensor. The 64 bit ROM stores the unique ID serial number of the device. The register contains two bytes of temperature register, which stores the digital output from the sensor. In addition, the register of alarm trigger threshold is provided by the register and the extended register. The configuration register allows the user to set the repeatability of temperature digital conversion and the frequency of continuous measurement. The status register can be used to query whether an alarm is triggered. Data can be stored in non-volatile memory, and data will not be lost when the chip is powered down.

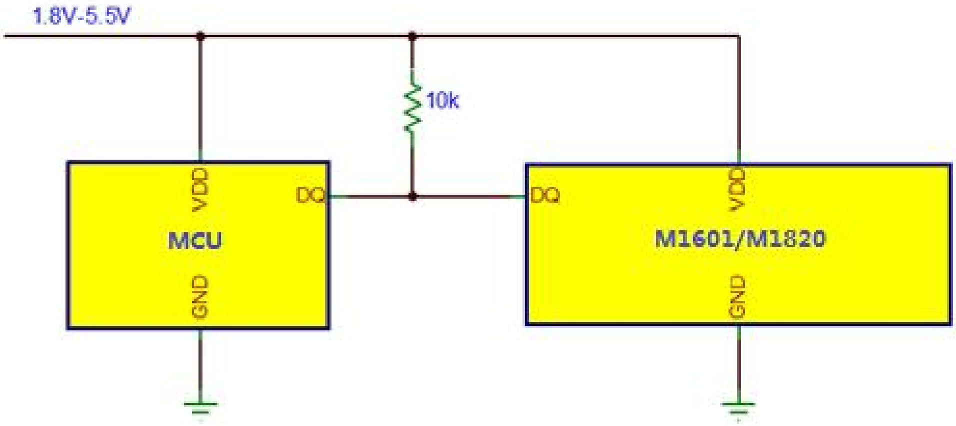
6. Typical Application

6.1. M601 one-wire typical application



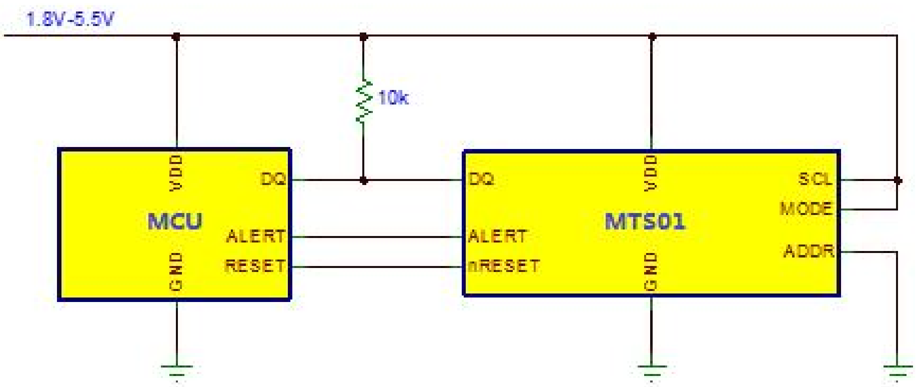
Notes:The alert pin is directly connected to the interrupt pin of the microcontroller or the electronic switch. The temperature alarm threshold is configured during the initialization of the equipment. There is no need to frequently read the software alarm flag bit during the temperature measurement process, and the real-time temperature alarm information can be obtained according to the collected alert pin level state.

6.2. M1601/M1820 one-wire typical application



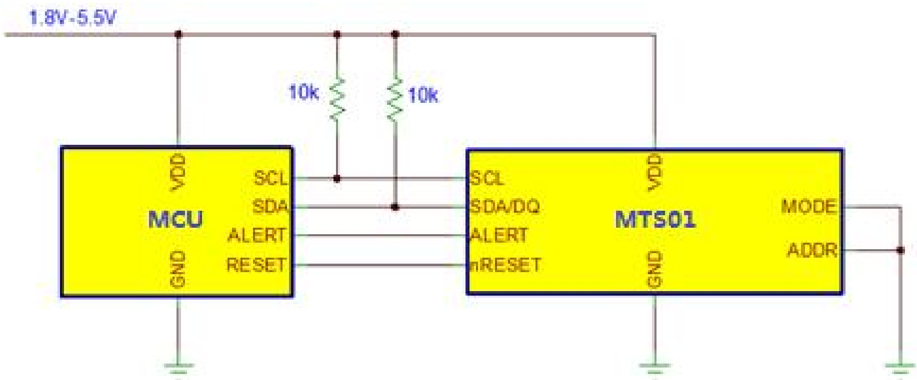
Notes1: Under long cable or multi-point driving conditions, please ensure that the power supply voltage is more than 3.3V as far as possible.
Note2: Under the condition of long cable or multi-point drive, 1K resistance is preferred for pull-up resistance.
Note3: Under the condition of 5V voltage and 1K pull-up resistance,100 PCS M601/M1601/M1820 can be connected in a single bus, and the maximum length of the cable can be up to 500m.

6.3. MTS01 one-wire typical application



Notes1: Under long cable or multi-point driving conditions, please ensure that the power supply voltage is more than 3.3V as far as possible.
Note2: Under the condition of long cable or multi-point drive, 1K resistance is preferred for pull-up resistance.
Note3: Under the condition of 5V voltage and 1K pull-up resistance,100 PCS MTS01 can be connected in a single bus, and the maximum length of the cable can be up to 500m.

6.4. MTS01 I2C typical application



7. Temperature measurement performance

Parameter	Symbol	Condition	Min	Typ	Max
Range	—	—	-70℃	—	+150℃
Accuracy tolerance	t_{ERR}	M601/M1601/MTS01	—	—	±0.1°C@+28°C to +43°C, ±0.5°C@-10℃ to+28°C&+43°Cto +60℃
		M601Z/M1601Z/M1820Z/MTS01Z	—	—	±0.1°C@0℃ to +50℃, ±0.2°C@-10℃ to 0°C& +50℃ to+60℃, ±0.5°C@-30℃ to -10℃&+60℃to+70℃
		M601W/M1601W/M1820W/MTS01W	—	—	±0.1°C@+20℃to +70℃, ±0.2°C@0℃ to+20℃&+70℃ to +75℃, ±0.5@-20℃ to 0℃ & +75℃ to +90℃
		M601P/M1601P/M1820P/MTS01P	—	—	±0.1°C@-20℃ to +30℃, ±0.2°C@-30℃to -20℃&+30℃to+40℃, ±0.5°C@-40℃to -30℃&+40℃to +70℃
		M601B/M1601B/M1820B/MTS01B	—	—	±0.5℃@0℃to 50℃
		M601/M1601/MTS01/M601Z/M1601Z/M1820Z/MTS01Z/M601W/M1601W/M1820W/MTS01W/M60	—	±1.5℃ @-55°C to +125°C	—

		1P/M1601P/M1820P/MTS01P/M601B/M1601B/M1820B/MTS01B			
		M601/M1601/MTS01/M601Z/M1601Z/M1820Z/MTS01Z/M601W/M1601W/M1820W/MTS01W/M601P/M1601P/M1820P/MTS01P/M601B/M1601B/M1820B/MTS01B	—	±2℃ @-70°C to +150°C	—
Repeatability	—	Low repeatability setting	—	0.07℃	—
		Medium repeatability setting	—	0.05℃	Note 1
		High repeatability setting	—	0.03℃	—
Resolution	—	—	—	0.004℃	—
Long-term drift	—	—	—	—	0.03°C/Y

Note 1: different output accuracy can be achieved by configuring different filter bandwidth, see table 9.2-2.

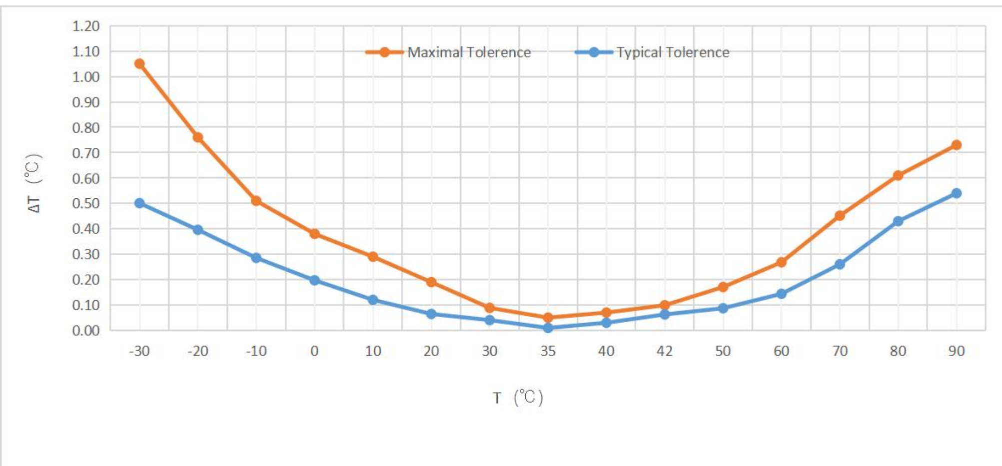


Figure 7 - 1 M601 / M1601 / MTS01 accuracy curve

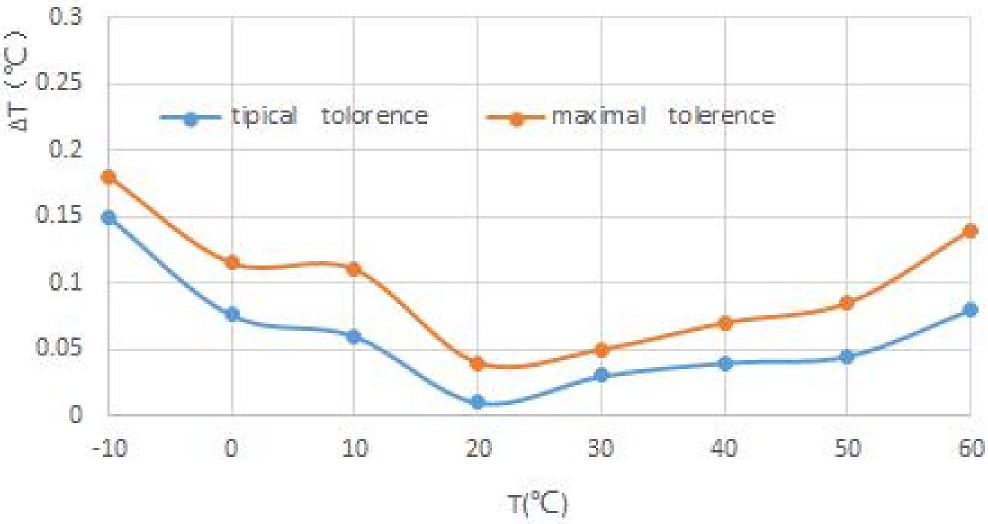


Figure 7 - 2 M601Z / M1601Z /M1820Z/ MTS01Z accuracy curve

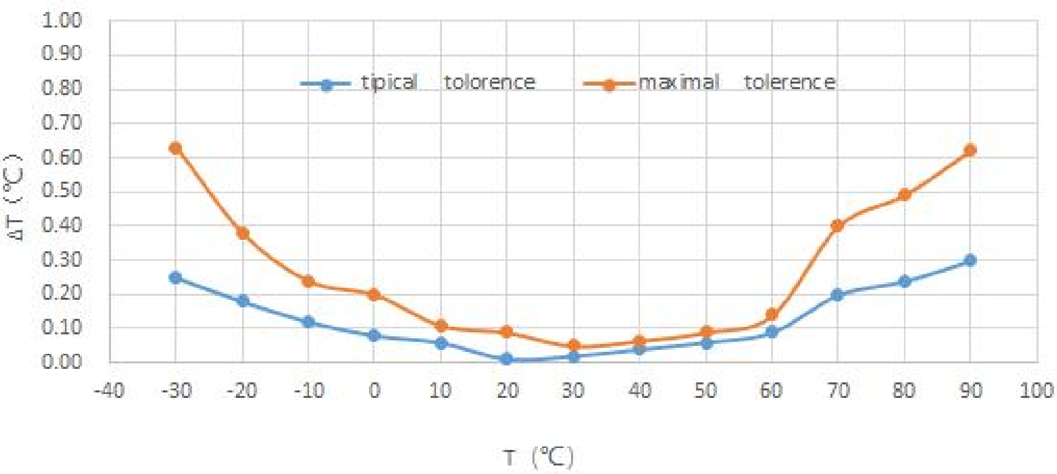


Figure 7- 3 M601W / M1601W /M1820W/ MTS01W accuracy curve

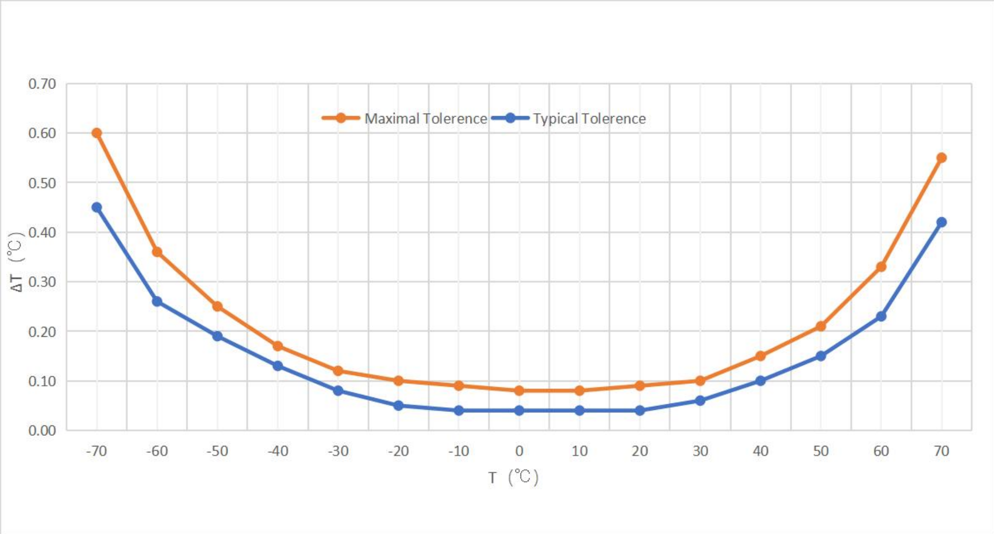


Figure 7 - 4 M601P / M1601P /M1820P/ MTS01P accuracy curve

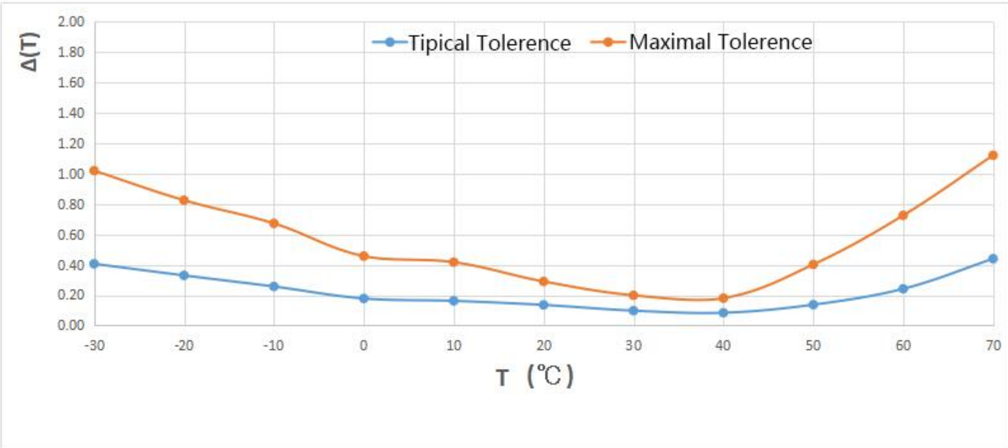


Figure 7 - 5 M601B / M1601B /M1820B/ MTS01B accuracy curve

8. Electrical Characteristics

8.1. Absolute maximum rating

These are only limit parameters and are not applicable to functional operation of the device in environments where this limit condition or higher. Long-term exposure to this extreme environment can affect device reliability.

Parameter	Value	Unit
Supply voltage	-0.3 to 6	V

Pin-to-ground voltage	-0.3 to 6	V
Input current	±100	mA
working temperature	-70 to 150	℃
Storage temperature	-70 to 150	℃
Welding temperature	Refer to IPC/JEDEC J-STD-020	
ESD HBM (Human body discharge mode)	±8	kV

8.2. DC Electric Characteristics

Parameter	Symbol	Condition	Min	Typ	Max	Unit	Note
Voltage							
Supply voltage	VDD		1.8	3.3	5.5	V	
Supply voltage swing rate change	VDD slew	—	—	—	20	V /ms	The voltage change between VDDmin and VDDmax on DD line should be slower than the maximum voltage swing rate, and a faster voltage swing rate may cause reset.
Supply current	IDD	Standby	—	0.1	1	uA	The current when no measurement
		Periodic measurement mode	—	55	—	uA	No sleep in periodic measurement mode
		Peak	—	447	—	uA	
		Average	—	5.2	—	uA	Single mode, high repeatability, one measurement per second
Waiting time after powered-up	t_{PU}	—	—	2	—	ms	

Digital input/output							
Input logic low	VIL	SCL, DQ/SDA	—	—	0.3*VDD	V	
Input logic high	VIH	SCL, DQ/SDA	0.7*VDD	—	—	V	
Output logic low	VOL	IOL = -3 mA	—	—	0.4	V	
Input leakage current	IIN	—	-0.1	—	0.1	uA	
Alarm output drive strength	IOH	—	—	1.5* VDD	—	mA	
Pull up resistor	Rup		1	10	100	kΩ	

8.3. AC Electrical Characteristics - Non-Volatile Memory

-55℃ to +125℃; V_{DD}=1.8V to 5.5V

Parameter	Symbol	Condition	Min	Typ	Max	Unit
Non-volatile write period	t_{WR}	—	—	—	40	ms
E2PROM endurance	N_{EEWR}	-55°C to +55°C	50000	—	—	times
E2PROM data retention	t_{EEDR}	-55°C to +55°C	—	10	—	year

9. Operation - Measuring Temperature

The thermometer is powered up in a low-power idle state. To initiate temperature measurement and analog-to-digital conversion, the host must issue the Convert T command. After the conversion, the generated temperature data is stored in the 2-byte temperature register. The conversion time is related to the repeatability setting. The higher the repeatability, the longer the conversion time. After power on, the default configuration is low repeatability. Refer to status register and configuration register for repeatability setting and time characteristic table for conversion time.

9.1. Temperature output and conversion formula

The digital output of temperature is 16bit signed binary complement, the resolution(LSB) is 1 /

256 ℃, and S is the sign bit. Data stored in temp_ lsb and Temp_msb.

Table 9.1 Temperature register format

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
LS Byte	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}
	bit15	**bit14**	**bit13**	**bit12**	**bit11**	**bit10**	**bit9**	**bit8**
MS Byte	S	2^{6}	2^{5}	2^{4}	2^{3}	2^{2}	2^{1}	2^{0}

The conversion relationship with Celsius is:

$$T\left[{}^{\circ}\mathrm{C}\right] = 40 + \frac{S_T}{256}$$

For example, 40℃ corresponds to register value 0x 00 00
150℃ corresponds to register value 0x 6E 00
-70℃ corresponds to register value 0x 92 00

9.2. Configuration register and status register

MPS, number of measurement per second is set for continuous measurement interval. when the MPS is set to any configuration except for a single time, the default continuous measurement mode is set. When the host sends the temperature conversion instruction convert T, the IC automatically starts the temperature conversion at the fixed time interval. Without sending the temperature measurement instruction again, it regularly reads the temporary memory temperature data according to the measurement interval. Repeatability directly affects the transition time of measured temperature; modifying the corresponding repeatability needs to wait enough corresponding time after temperature measurement conversion.

Table 9.2- 1 Configuration（CFG）

Bit	Content	Default Value
7	1：Alarm Function ON 0：Alarm Function OFF	‘0’

6:5	Reserved	'00'
4:2	MPS, Periodic measurement frequency configuration 000: single measurement 001: 0.5 times per second 010: once per second 011: 2 times per second 100: 4 times per second 101: 10 times per second	'000' (Note 1)
1:0	Repeatability setting 00: low repeatability 01: medium repeatability 10: High repeatability	'00' (Note 2)

Note 1: The default periodic measurement configuration is single measurement, and the measurement frequency can be configured as required. The fastest speed is about 133 samples/second under low repeatability, 111 samples/second under medium repeatability and 70 samples/second under high repeatability.

Another way to achieve the periodic measurement is to use the single measurement in a cyclical control function, the steps are shown as follows:

i. Calculate the duration of each measurement under the required measurement frequency

ii. Use duration of each measurement minus the combination of conversion time and one-wire command time, to get the delay value

iii. Put the delay value at the end of the cyclical function to achieve the required measurement frequency.

Note 2: Repeatability and conversion time are a direct compromise relationship. The higher the repeatability, the longer the conversion time; The lower the repeatability, the shorter the conversion time. See table 9.2-2 below.

Table 9.2- 2 Conversion time and repeatability settings

Repeatability		**Precision**	**Conversion time t_{CONV}**
0	0	low	4ms
0	1	medium	5.5ms
1	0	high	10.5ms

The status register contains the status of the measurement command, the alarm status, and the status of the last command execution and the last write sequence. See table 9.2-3 for the content description of status register.

Table 9.2- 3 Status Register (status)

Bit	**Description**			**Default**
7:6	Reserved			'00'
5	I^2C write data CRC	0: Correct	1: Wrong	'0'
4	I^2C Command Status	0: Wrong	1: Correct	'0'
3	System reset	0: Not detected	1: Detected	'0'

	detection			
2	Temperature alarm tracking	0：Temperature alarm not triggered	1：temperature alarm triggered	'0'
1	Reserved			'0'
0	Cycle measurement command status	0：No measurement	1：Measuring temperature	'0'

9.3. Alarm

The alarm mode allows monitoring of ambient temperature conditions through programmable thresholds. When the threshold is reached, the output level of the dedicated alert pin will change. In addition, the status register bit has a special bit to indicate the alarm status. Use ALERT pin to control a switch. Or it can be connected to the interrupt pin of the microcontroller. After the sensor gives an alarm, the microcontroller can wake up from sleep mode and perform certain operations.

The alarm mode is activated as long as the sensor is measuring. Deactivate the alarm mode by setting the minimum setting to a value greater than or equal to the maximum setting (TlSet>= ThSet).

The alarm threshold can be set by corresponding instructions (see I2C command set in section 11.1. And single bus instruction set in table 10.5). Different thresholds are shown in the figure below.

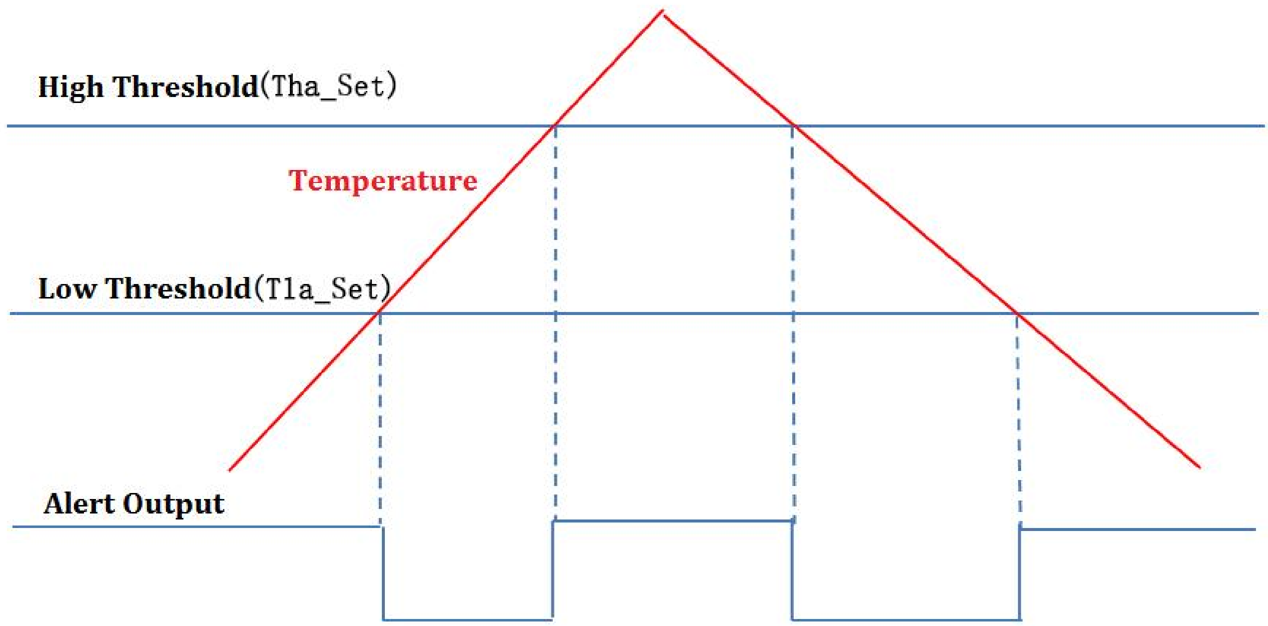


Figure 9.3- 1 Different thresholds for alarm mode

The alarm threshold is stored in a simplified format, that is, only the most effective 9 bits are stored, and compared with the most effective 9 bits of the 16 bit standard output to determine whether the alarm conditions have been met. See the figure below. Therefore, the alarm threshold has a resolution different from the measured value. The resolution of temperature alarm threshold is Δ t ≈ 0.5 ° C. Note that data is always measured and stored in 16 bit format. The simplified data format is only used to determine whether the alarm conditions are met.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
High 9 bits of temperature measured									X	X	X	X	X	X	X

Fig. 9.3-2 Relevant data bits of alarm threshold

The requirement of alarm threshold limit setting is described hereunder:

ThSet>ThClear>TlClear>TlSet>=40 或 TlSet<TlClear<ThClear<ThSet<40

Example of calculating alarm threshold No.1:

1) Select the alarm threshold value of temperature (e.g. high limit trigger threshold: ThSet= 60°C, ThClear=55℃, TlClear=45℃, low limit trigger threshold: TlSet=40°C)
2) Use the following formula

$$S_T = (T - 40) \times 256$$

3) Convert the Thset to a 16 bit binary value:

ThSet = 0001'0100'0000'0000

4) Delete the lower 7 bits of Thset, and add 7 "0" before it

5) ThSet=00000000001'0100'~~0000'0000~~=0000'0000'0010'1000

6) Tha_ Set_ msb=0000'0000, Tha_ set_ lsb=0010'1000
7) Tha_Clear_msb=0000'0000, Tha_Clear_lsb=0001'1110
8) Tla_Clear_msb=0000'0000, Tla_Clear_lsb=0000'1010
9) Tla_Set_msb=0000'0000, Tla_Set_lsb=0000'0000

Example of calculating alarm threshold No.2:

1) Select the alarm threshold value of temperature (e.g. high limit trigger threshold: ThSet = 39°C, ThClear=34℃, TlClear=30℃, low limit trigger threshold: TlSet = 25°C)
2) Use the following formula

$$S_T = (T - 40) \times 256$$

3) Convert the Thset to a 16 bit binary value:

ThSet = 1111'1111'1000'0000

4) Delete the lower 7 bits of Thset, and add 7 "0" before it
5)ThSet=00000001111'1111'~~0000'0000~~=0000'0001'1111'1110
6)Tha_Set_msb=0000'0001, Tha_Set_lsb=1111'1110
7)Tha_Clear_msb=0000'0001, Tha_Clear_lsb=1111'0100
8)Tla_Clear_msb=0000'0001, Tla_Clear_lsb=1110'1100
9)Tla_Set_msb=0000'0001, Tla_Set_lsb=1110'0010

Example of disabling alarm threshold No.3:

1) Select the alarm threshold value of temperature (e.g. high limit trigger threshold: ThSet = 38.5°C，ThClear=34℃，TlClear=30℃，low limit trigger threshold: TlSet = 40°C)
2) Use the following formula

$$S_T = (T - 40) \times 256$$

3) Convert the ThSet to a 16 bit binary value:

ThSet = 1111'1110'1000'0000

4) Delete the lower 7 bits of Thset, and add 7 "0" before it
5)ThSet=00000001111'1110'1~~000'0000~~=0000'0001'1111'1101
6)Tha_Set_msb=0000'0001, Tha_Set_lsb=1111'1101
7)Tha_Clear_msb=0000'0001, Tha_Clear_lsb=1111'0100
8)Tla_Clear_msb=0000'0001, Tla_Clear_lsb=1110'1100
9)Tla_Set_msb=0000'0000, Tla_Set_lsb=0000'0000
10) （TlSet > ThSet ），the alarm function is disabled.

9.4. Interface

The thermometer provides two serial communication interfaces: one-wire bus and I2C.
One-wire bus communication is realized by a control signal line. The control line needs to be equipped with a weak pull-up resistor, so that all devices are connected to the bus through three state or open drain port (DQ pin). In this bus system, the single-chip microcomputer (host) identifies and addresses the devices on the bus through the unique 64 bit code of each device. Because each device has a unique code, the number of devices that theoretically hang on the bus

and can be addressed is infinite. One-wire bus protocol, including detailed instruction and slot description, is described in detail in the chapter of single bus system.

I2C protocol is described in detail in I2C Bus Chapter.

10. Memory

10.1. 64-bit ROM code

Each thermometer contains a unique 64 bit code stored in ROM. The lowest 8-bit ROM code contains the one-wire bus serial code of the sensor: 28h. The next 48 bits contain a unique sequence code. The highest 8 bits contain the cyclic redundancy check code generated from the previous 56 bit ROM code. The 64 bit ROM coding and the related ROM function control logic make the thermometer use the one-wire bus protocol as a single bus device. The one-wire bus protocol is detailed in the one-wire bus system chapter.

56-bit sequence	8-bit product series code
MSB	LSB

10.2. Memory organization and access

The memory organization of the sensor is shown in table 10.2-1 and table 10.2-2. The memory consists of a SRAM register and an extended non-volatile E2PROM register, which is used to store high and low alarm trigger values (th and TL), configuration register and 2-byte user programmable E2PROM. It should be noted that if the alarm function of the sensor is not used, the th and TL registers can be used as general storage. All stored instructions are described in detail in the sensor function instructions section. Register byte 0 Temp_LSB and byte 1 Temp_The MSB is a read-only temperature measurement. Bytes 2 and 3 are reserved registers. Byte 4, 5 Tha_ Set_lsb， Tla_Set_lsb is a register for setting alarm threshold. Byte 6 CFG contains configuration register data, and byte 7 status is the status register of the chip, which is explained in detail in Chapter 9.2. Byte 8 is a read-only register and a cyclic redundancy check code generated by bytes 0 to 7. The sensor generates this code through the rules described in the cyclic redundancy check code generation section. Data can be written to bytes 4, 5, 6 with the write scratchpad [0x4e] instruction. When writing the register, the data transmission must start from the lowest bit of byte 4.To verify data integrity, read the registers (via the scratchpad [0xbe] instruction) after the data write operation. When reading the register, the data transmission of single bus starts from the lowest bit of byte 0.

To permanently write data from the register to E2PROM, the host must issue the copy scratchpad [0x48] command. The data in E2PROM register will be kept when power is off, and will be loaded to the corresponding register position automatically when power is on. The data can also be reloaded at any time through the recall E2 [0xb8] instruction. Please note that the copy scratchpad [0x48] is operated by page (16 bytes). You need to make sure that all the 16 temporary contents needs to be confirmed correct and then copy uniformly.

Table 10.2- 1 Thermometer memory storage mapping - register group

Scratchpad register group							
Name	**Addr offset**	**Read**	**Write**	**E^{2}PROM**	**Copy**	**Recall**	**Reset Value**
Temp_lsb	0	Read scratchpad (0xbe)	NA	NA	NA	NA	H' 00
Temp_msb	1		NA	NA	NA	NA	H' F1
Reserved	2		NA	NA	NA	NA	H' 00
Reserved	3		NA	NA	NA	NA	H' 00
Tha_Set_lsb	4		Write scratchpad (0x4e)	0	Copy page0 (0x48)	Recall EE (0xb8)	H' 00
Tla_Set_lsb	5			1			H '00
Cfg	6			2			H' 02
Status	7		NA	NA	NA	NA	XX
Crc_src	8		NA	NA	NA	NA	XX

Table 10.2- 2 Thermometer memory storage mapping - extended registers group

Scratchpad_Ext Register group							
Name	**Addr offSet**	**Read**	**Write**	**E^{2}PROM**	**Copy**	**Recall**	**ReSet Value**
Tha_Clear_lsb	0	Read scr_ext (0xdd)	Write scr_ext (0x77)	4	Copy page0 (0x48)	Recall page0 res (0xbb)	H '00
Tla_Clear_lsb	1			5			H '00
Tha_Set_msb	2			6			H '00
Tla_Set_msb	3			7			H '00
Tha_Clear_msb	4			8			H '00
Tla_Clear_msb	5			9			H '00
Reserved	6-11			10-15			H '00
crc_scr_ext	12		NA	NA	NA	NA	XX

If larm function is disabled, the MSB of Cfg register must be set to 0. When the MSB of Cfg register is 0, registers of Tha_Set_lsb, Tla_Set_lsb, Tha_Clear_lsb, Tla_Clear_lsb, Tha_Set_msb, Tla_Set_msb, Tha_Clear_msb and Tla_Clear_msb can be used as storage registers for users. These registers can be read/written by using command Read_scratchpad(0xbe), Read_scr_ext(0xdd), Write_scratchpad(0x4e) and Write scr_ext(0x77).

11. One-wire bus

M601，M1601 and M1820 support single bus, and MTS01 supports single bus and I2C interfaces.

The one-wire bus system uses a signal line, and the host controls one or more slave devices. Sensors are always slave devices. When there is only one slave on the bus, the system is called "single point" system; when there are multiple slaves on the bus, it is called "multi point" system. All data and instructions are transmitted on a one-wire bus starting at the lowest level. The following description of a single bus system is divided into three topics: hardware configuration, transmission sequence, and one-wire bus signaling (signal type and timing).

11.1. Hardware Configuration

By definition, a one-wire has only one data line. Each device (master or slave) is connected to the data line through an open drain or tri-state port. This allows the device to "release" the data line when no data is being transferred, so that the bus can be used by other devices.

The internal equivalent circuit of the thermometer one-wire port (DQ pin) is open-drain, as shown in Figure 11.1 . A one-wire requires an external pull-up resistor of approximately 1-5KΩ; thus, the idle state of a one-wire is high. If the transfer requires a pause for any reason, the bus must remain idle until the transfer has not returned. During the recovery process, if the one-wire remains in an inactive (high) state, the recovery time between data bits can be infinitely long. If the bus is pulled low for more than 480 μs, all devices on the bus will be reset.

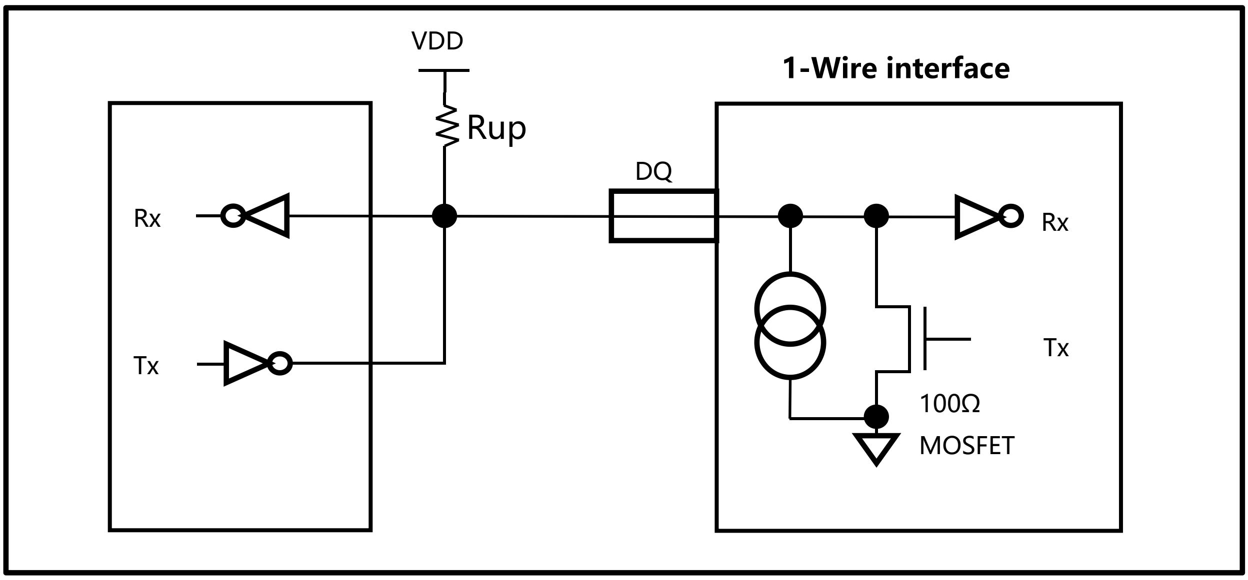


Figure 11.1 Hardware Configuration

11.2. Transfer Sequence

The transmission sequence for accessing thermometer is as follows:

Step 1 Initialize
Step 2 ROM command (follow any necessary data exchange)
Step 3 Thermometer function commands (follow any required data exchange)

The above order is very important for thermometer access, because any missing steps or incorrect order will cause thermometer not to respond. Only the Search ROM [F0h] and Alarm Search [ECh] commands can be exceptional. After these two ROM commands are issued, the host must return to step 1 in the sequence.

11.3. Cyclic redundancy check (CRC)

One-wire bus read registers and extended registers will follow cyclic redundancy check codes and change with data changes. The cyclic redundancy check code provides a data check method for the host to read data from the registers and extended registers. To verify whether the data is read correctly, the host must calculate the received data by itself and compare the data with the read cyclic redundancy check code. If the calculated cyclic redundancy check code is consistent with the read one, the data is received correctly. The comparison of cyclic redundancy checks and whether to continue operations are entirely dependent on the host. If the cyclic redundancy check of the thermometer does not match the value calculated by the bus host, there is no circuit in the sensor to prevent further execution of the command. The equivalent polynomial function of cyclic redundancy check code is:

$$CRC = X^8 + X^5 + X^4 + 1$$

The host can recalculate the cyclic redundancy check code and compare it with the cyclic redundancy check value generated by the sensor, which is generated by the polynomial generator in figure 9.4-1. The circuit consists of a shift register and a number of XOR gates. Each bit of the shift register is initialized to 0. From the lowest bit of byte 0 in the register or extended register, move one bit at a time to the shift register. Until the maximum displacement of the last byte of the register or extended register is completed, the recalculated cyclic redundancy check will be stored in the polynomial generator. Next, the cyclic redundancy check data of the sensor's register or extended register must be moved into the circuit. Here, if the recalculated cyclic redundancy check is correct, the shift register should be full of zeros.

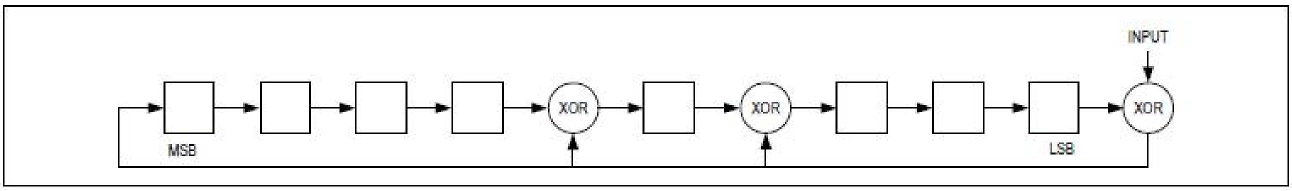


Figure11.3 cyclic redundancy check generator

The 8-bit CRC checksums sent after each data word are generated by the CRC algorithm. Its properties are shown in the following table.

Parameter	Value
Name	CRC-8/MAXIM
width	8-bit
Data Protect	W/R
Polynomial	0x31 ($x^8 + x^5 + x^4 + 1$)
Initial value	0x00
Whether each byte of the test data reversed by bit	Yes
After calculation and before exclusive or output, whether the whole data is reversed by bit	Yes
The calculation result is exclusive or to this parameter	0x00
Example	CRC (0xBEEF) = 0x76

Note: ROM ID last two bytes are 0x00 without CRC. CRC can be generated by Maxim CRC-8 polynomial above in MCU software.

11.4. Initialization

All execution (processing) through the one-wire bus begins with an initialization sequence. The initialization sequence includes a reset pulse issued by the bus controller and a presence pulse followed by a slave. There is a pulse for the host to know that the slave device is on the bus and is ready to run. The timing of reset and presence of pulses is detailed in the one-wire Signal Timing section 11.6.

11.5. ROM Commands

When the host detects the presence pulse, it can issue a ROM command. These commands operate on a unique 64-bit ROM code for each slave device and if multiple slave devices are attached to a one-wire, the host can specify the individually device. These commands also allow the host to determine what type of device is on the bus and if any device meets the alarm condition. There are 5 ROM commands, each 8 bits long. The master device must issue a suitable ROM command before issuing the thermometer function commands.

SEARCH ROM [F0h]

When a system is initialized and powered up, the host must identify the ROM encoding of all slave devices on the bus so that the host can determine the number and type of slave devices. The host recognizes the ROM code by the exclusion process, which requires the host to perform a Search ROM loop (eg, the Search ROM command follows the data exchange) repeatedly until all slave devices are identified. If there is only one slave on the bus, you can use a simple Read ROM (see below) instead of the Search ROM. After each Search ROM cycle, the host can return to step 1 (initial state) of the transfer sequence or follow a function command.

Read ROM [33h]

This command allows the host to read the 4-bit ROM code of the thermometer. This command can only be used if there is a single thermometer on the bus. If there is more than one slave on the bus, a data collision occurs when all slaves attempt to respond at the same time.

Match ROM [55h]

Matches the ROM command followed by a 64-bit ROM encoding sequence that allows the host to address a specified thermometer on a multipoint or single-point bus. Only the thermometer with exact 64-bit ROM code will respond to the function command issued by the host. Other slaves will be in idle state and wait for a reset pulse.

Skip ROM [CCh]

It is possible for the host to simultaneously address all devices on the bus without sending any ROM code. For example, the host can perform temperature conversion on all thermometer on the bus at the same time, just by issuing a Skip ROM command followed by a Convert T [44h] command.

Note that the Read Scratchpad [BEh] command can only follow the Skip ROM command when a single slave device is attached to the bus. In this case, time can be saved by allowing the host to read the slave without sending a 64-bit device ROM code. If there is more than one slave on the bus, a Skip ROM command following a Read Scratchpad command will cause a data collision because multiple devices will attempt to transmit data at the same time.

Alarm Search [ECh]

The flow chart for this command is the same as the search ROM. However, only the thermometer with the alarm flag set will respond to this command. This directive allows the host device to know if any of the thermometer has reached the temperature alarm condition during the most recent temperature conversion. After each Alarm Search cycle (eg. the Alarm Search command follows the data exchange), the host can return to step 1 (initialization) of the transmission sequence or follow a function command. The Run-Alarm Signaling section explains the operation of the alarm flags.

11.6. Function commands

After the host uses a ROM command to address a desirable thermometer the host can issue one of the thermometer function commands. These commands write or read data from the thermometer scratchpad, initiate temperature conversion, and understand the power mode. The functional commands of thermometer as described below, are summarized in Table 2 and are elaborated by the flow chart of FIG 10.

CONVERT T [44h]

This command initiates a temperature conversion. After the conversion, the acquired thermal data is stored in the 2-byte temperature register in the scratchpad and then thermometer returns to the low-power idle state. The host can issue a read time slot after the Convert T command, and then thermometer will return 0 or 1 to indicate that the temperature conversion is in progress or completed.

WRITE SCRATCHPAD [4Eh]

This command allows the host to write up to 2 data to the thermometer register. The first byte is written to the TH register (storage byte 2) and the second byte is written to the TL register (storage byte 3). Data must be sent the lowest bit first. All 2 bytes must be written before the host issues a reset signal, otherwise the data may be corrupted(meaning that the host can abort writing by resetting at any time).

READ SCRATCHPAD [BEh]

This command allows the host to read the contents of the scratchpad. The data transfer begins at the lowest bit of byte 0 through the ninth byte (byte 8, CRC byte). If only part of the data in the scratchpad is needed, the host can issue a reset signal to terminate the read at any time.

WRITE SCR_EXT [77h]

This command allows the host to write up to 12 bytes of data to the sensor extension register. The second byte is written to the alarm high limit high byte Tha_Set_msb register, the third byte is written to the alarm low limit high byte Tla_Set_msb register.

READ SCR_EXT [DDh]

This command allows the host to read the contents of the extended register. The data transfer begins at the lowest bit of byte 0 through the 12th byte (byte 12, CRC byte). After all 13 bytes are read, the sensor enters the low power consumption mode by default. If only part of the data in the extend register is needed, the host can issue a reset signal to terminate the read at any time.

COPY SCRATCHPAD [48h]

This command writes the TH, TL, configuration registers and user bytes 3 and 4 (bytes 2, 3, 4, 5, 6, 7 in the scratchpad) to the E2PROM.

RECALL E^2 [B8h]

This command uses the alarm trigger value (TH and TL) from the E2PROM, configures the register, and replaces the corresponding data in bytes 2, 3, and 4 in the scratchpad. The host device can issue a read time slot following the Recall E2 command, then thermometer will indicate the status by sending back 0 or 1, which indicates that the function is in progress or the function has ended. The function is automatically executed when the power powers on, so the valid data is immediately available in the scratchpad after the device is ready.

RECALL PAGE0 RES [BBh]

The instruction loads the data of the extended register from E2PROM. The host device can issue read timeslot continuously after the recall res command is followed, and then the sensor will indicate the loading status. Transfer 0 indicates that the loading is in progress, and transfer 1 indicates that the call has ended. The loading operation is automatically executed when the device is powered on, so there is valid data in the extended register immediately after the device is powered on.

Table11.6 thermometer Function Commands

Command	Description	Value	Events after command	Remarks
Temperature transformation commands				
Convert T	Initialize temperature conversion	44h	thermometer transmits conversion status to host:When receives the read time slot, thermometer transmits a 0 while the conversion is in progress; thermometer transmits a 1 when the conversion is done..	
Memory Commands				
Read Scratchpad	Read scratchpad contents, include CRC code.	BEh	thermometer transfers up to 9 bytes to host.	1
Write Scratchpad	Writes data to scratchpad byte 2,(TH, TL)	4Eh	Host transfer scratchpad bytes 2，3, and 4 transfers to thermometer	2
Read scratchpad extend	Read the contents of all extended registers include CRC code	DDH	Sensor transfers up to 13 bytes to the host.	1
Write scratchpad extend	Write data to the extended register, including byte 2 and byte 3 （Tha_Set_msb，）（Tla_Set_msb)	77h	The host transmits 12 bytes of data from the extended register to sensor	2
Break	Stop continuous measurement mode	91h	none	
Copy Scratchpad	Copy TH, TL, config byte, and user data from scratchpad to E2PROM	48h	Write E2PROM time 40ms	
Recall E^2	Load TH, TL, config	B8h	thermometer Returns	

	byte from E2PROM to scratchpad		execution status.	
Recall page 0 res	Restore extended area of Page0	BBh	The sensor transmits the call status to the host	

N*ote:*

1) *host can interrupt data transfer by sending reset pulse.*
2) *all bytes must be written out before reset pulse.*

11.7. One-wire Signal Timing

The thermometer uses a strict one-wire communication protocol to ensure data integrity. The protocol defines several types of signaling: reset pulse, presence pulse, write 0, write 1, read 0, read 1. All signaling is initiated by the host, except for the presence pulse.

Initialization Process - Reset and Presence Pulse

All communication with the thermometer begins with an initialization sequence that contains a reset pulse from the host followed by a presence pulse from the thermometer. Figure 11 illustrates this sequence. When the thermometer sends a presence pulse in response to the reset pulse, it indicates to the host that the device is connected on the bus and is ready to run. During the initialization sequence, the master issues a reset pulse by pulling the one-wire low for at least 480 μs. The host then releases the bus into receive mode. When the bus is released, pull-up resistor pulls the bus high. When the thermometer detects this rising edge, it waits for 15μs to 60μs and then issues a presence pulse by pulling the one-wire down 60μs to 240μs.

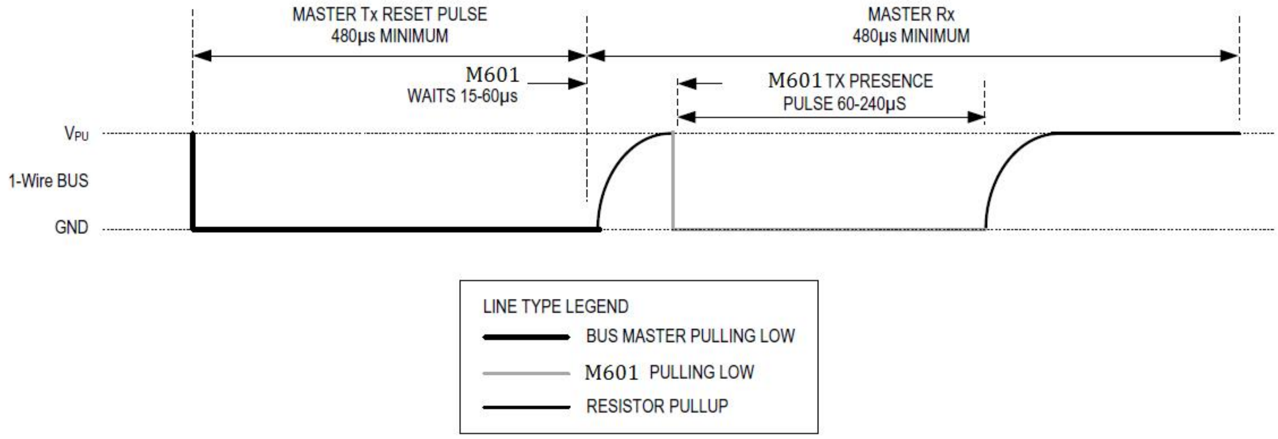


Fig. 11.7-1 Initialization

Read/Write Time Slot

The host writes data to thermometer in the write time slot and reads data from thermometer in the read time slot. Each time slot transmits one bit on one-wire.

Write Time Slot

There are two types of write time slots: "write 1" time slot and "write 0" time slot. The host writes a logic 1 to thermometer by writing 1 time slot and writes a logic 0 to thermometer by writing 0 time slot. All write time slots must last at least 60 μs and there is at least a 1 μs recovery time between the two write time slots. Both write time slots are initiated by the host pulling the one-wire low (see Figure 12).
To generate a write 1 time slot, after the one-wire is pulled low, the host must release the one-wire within 15 μs. After the bus is released, pull-up resistor pulls the bus high. To generate a write 0 time slot, after pulling the one-wire low, the host must keep the bus low at least 60 μs throughout the time slot.
After the host initiates a write time slot, the thermometer samples the one-wire for a time window of at least 15μs to 60μs. If the bus is high during this sampling time window, a 1 is written to thermometer. If the bus is low, a 0 will be written to thermometer.

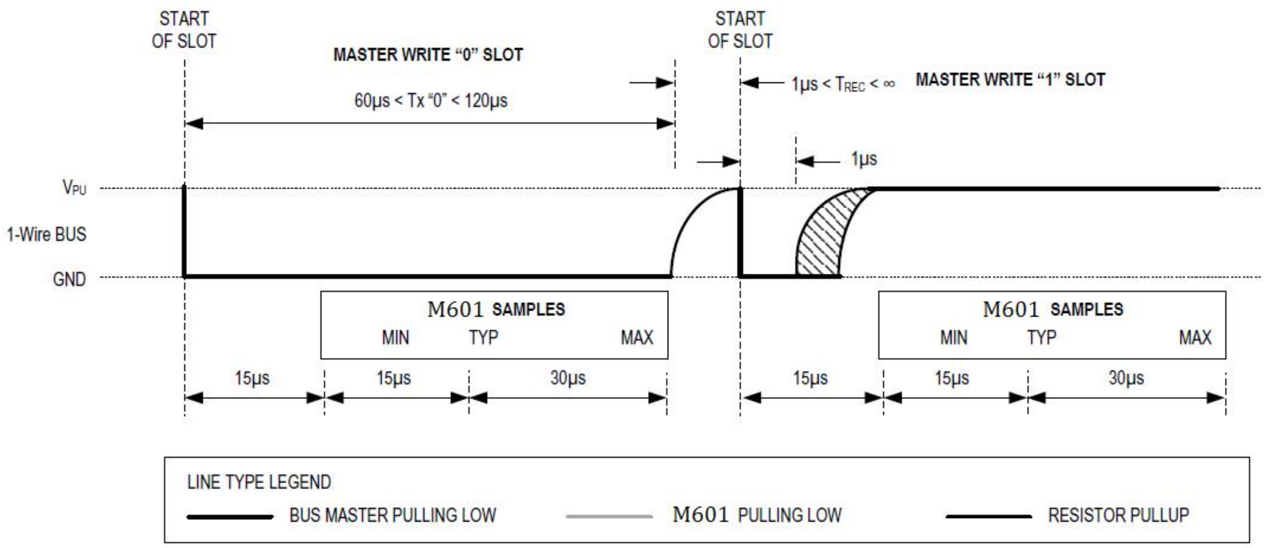


Fig 11.7-2 Write timing slot

Read Time Slot

The thermometer can only transfer data to the host during the host's issues the read time slot. After the host issues the Read Scratchpad [BEh] or Read Power Supply [B4h] command, the read time slot must be generated immediately so that the thermometer can provide the requested data. In addition, the host can generate a read time slot after issue the Convert T[44h] or Recall E2 [B8h] command to get the operating status. This part of the mechanism is explained in detail in the thermometer functional commands section. All read time slots must last at least 60 μs and the recovery time between the two write time slots is not less than 1 μs. The generation of the read time slot is achieved by the host pulling the one-wire low for at least 1 μs and then releasing the bus (see Figure 12). After the host initiates a read time slot, the thermometer will start transmitting 1 or 0 on the bus. The thermometer sends a 1 by holding the bus high and sends a 0 by pulling the bus low. When transmitting 0, thermometer will release the bus at the end of the time slot, after which the bus will be pulled back to the high idle state by the pull-up resistor. The output data of the thermometer is valid for 15μs after the falling edge of the start time slot.

Therefore, the host must release the bus and sample the bus state within 15 μs after the time slot is started. Figure 13 illustrates that the sum of t_{INIT}, t_{RC} and t_{SAMPLE} must be less than 15 μs in a read time slot. Figure 14 shows that the time margin of the system can be maximized by keeping t_{INT} and t_{RC} as short as possible and placing the host sample time at the end of the 15 μs period of the read time slot.

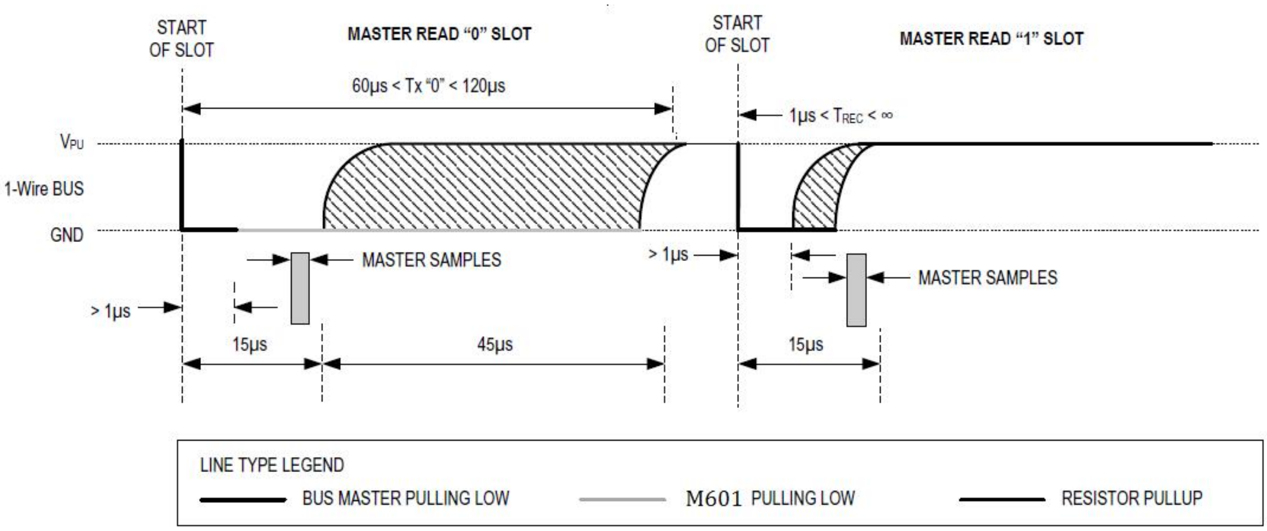


Fig 11.7-3 Read timing slot

11.8. Example Conditions 1

In this condition, for the middle cable, there will be several thermometer. The host will read a specific thermometer scratchpad after temperature conversion and then recalculates the CRC to verify the data.

Host	Data (LSB first)	Command
Send	Reset pulse	Host send reset pulse
Receive	Presence pulse	Sensor respond presence pulse
Send	55h	Host send Match ROM command
Send	64bit ROM ID	Host send Match ROM command
Send	44h	Host send Convert T command
	DQ keep high	Host keep DQ high during temperature conversion
Send	Reset pulse	Host send reset pulse
Receive	Presence pulse	Sensor respond presence pulse
Send	55h	Host send Match ROM command
Send	64bit ROM ID	Host send Match ROM command
Send	BEh	Host send Read Scratchpad command
receive	9 bytes data	Host read all scratchpad data, include CRC byte.Then host calculate CRC byte for the first 8 bytes locally, and compare with the CRC code read back. Host continues if CRC match, otherwise, host repeat above steps.

11.9. Example Condition 2

In this example there is only one thermometer on the main cable. The host writes the TH, TL and configuration data to the thermometer register, then reads the scratchpad and recalculates the CRC to verify the data. The host then copies the contents of the scratchpad to the E2PROM.

Host	Data (LSB first)	Command
Send	Reset pulse	Host send reset pulse
Receive	Presence pulse	thermometer respond presence pulse
Send	CCh	Host send Skip ROM command
Send	4Eh	Host send Write Scratchpad command
Send	3 bytes data	Host send 3 bytes data to TH, TL, and configure
Send	Reset pulse	Host send reset pulse
Receive	Presence pulse	Sensor respond presence pulse
Send	CCh	Host send Skip ROM command
Send	77h	Host send Write scratchpad exten command
Send	12 bytes data	Host read all scratchpad data, include CRC byte, then host calculate CRC byte for the first 12 bytes locally, and compare with the CRC code read back. Host continues if CRC match, otherwise, host repeat above steps.
Send	Reset pulse	Host send reset pulse
Send	Presence pulse	Sensor respond presence pulse
Send	CCh	Host send Skip ROM command
Send	48h	Host send Read Scratchpad command
Send	DQ keep high	Host keep DQ high for at least 40 ms during Copying Command.

12. I^{2}C bus

12.1. I^{2}C command

Command	Function	Code
Measurement	Convert Temperature	0xCC44
Read measurement results	—	—
Set the upper threshold of alarm threshold	WRITE_ALERT_HI_SET	0x611D
Set alarm clear threshold	WRITE_ALERT_HI_UNSET	0x6116
Set alarm threshold lower threshold	WRITE_ALERT_LO_SET	0x6100
Set alarm clear threshold lower threshold	WRITE_ALERT_LO_UNSET	0x610B
Read the upper threshold of alarm threshold	READ_ALERT_HI_SET	0xE11F
Read the upper threshold of alarm clear	READ_ALERT_HI_UNSET	0xE114
Read the lower threshold of alarm set	READ_ALERT_LO_SET	0xE102
Read alarm threshold lower threshold clear	READ_ALERT_LO_UNSET	0xE109

Write configuration register	CONFIG	0x5206
Read status and configuration registers	READ_STATUS	0xF32D
Clear status register	CLEAR_STATUS	0x3041
Stop periodic measurement	BREAK	0x3093
Software reset	SOFT_RST	0x30A2
Save Page0 to EEPROM	COPY_PAGE0	0xCC48
Restore EE area	RECALL_EE	0xCCB8
Restore the residual area of Page0	RECALL_PAGE0_RES	0xCCB6

12.2. Operation and communication

The sensor supports I2C fast mode (up to 400 kHz). Clock stretch can be enabled and disabled by the corresponding user command. After sending a command to the sensor, a minimum waiting time of 1ms is required before the sensor receives another command. All sensor commands and data are mapped to a 16 bit address space. In addition, data and commands include CRC verification, which improves communication reliability. The data sent and received by the sensor always follows the 8-bit CRC checksum. In the write operation, the host must send the checksum to the slave, and the sensor can only accept the data after receiving the correct checksum. During the read operation, the main device reads and processes the checksum.

Parameter	Value
Name	CRC-8
Width	8-bit
Data Protect	W/R
Polynomial	0x31 ($x^8 + x^5 + x^4 + 1$)
Initial Value	0xFF
Whether each byte of the test data reversed by bit	No
After calculation and before exclusive or output, whether the whole data is reversed by bit	No
The calculation result is exclusive or to this parameter	0x00
Example	CRC (0xBEEF) = 0x92

Note: ROM ID last two bytes are 0x00 without CRC. CRC can be generated by Maxim CRC-8 polynomial above in MCU software.

12.3. Power on and communication starts

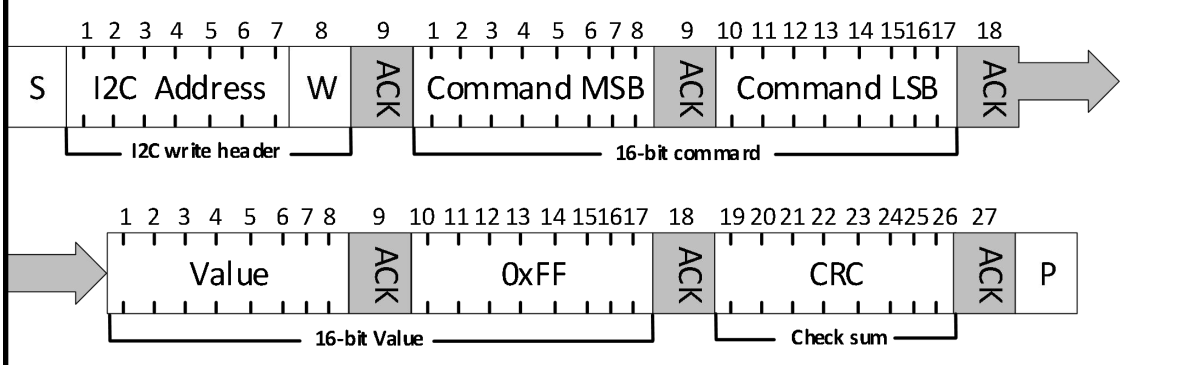
After power on, the sensor starts to power on after reaching the power on threshold voltage vpor. When the threshold voltage is reached, the sensor needs time for TPU to enter idle state. Once idle, commands can be received from the main device (microcontroller). Each transmission sequence starts with a start condition (s) and ends with a stop condition (P), as described in the I2C bus specification. Whenever the sensor is powered up but no measurement or communication is performed, it automatically goes into an idle state to save energy. The idle state cannot be controlled by the user.

12.4. Start measurement

The measurement communication sequence includes start condition, I2C writing head (7-Bit I2C device address plus 0 as the writing bit) and 16 bit measurement command. The sensor indicates the correct reception of each byte. It pulls the SDA pin down (ACK bit) after the falling edge of the 8th SCL clock to indicate reception. The complete measurement cycle is described in section 12.8. By confirming the measurement command, the sensor begins to measure the temperature. In addition, measurement repeatability and single / continuous measurement mode are set by corresponding control bits of configuration register.

12.5. Set configuration register instruction

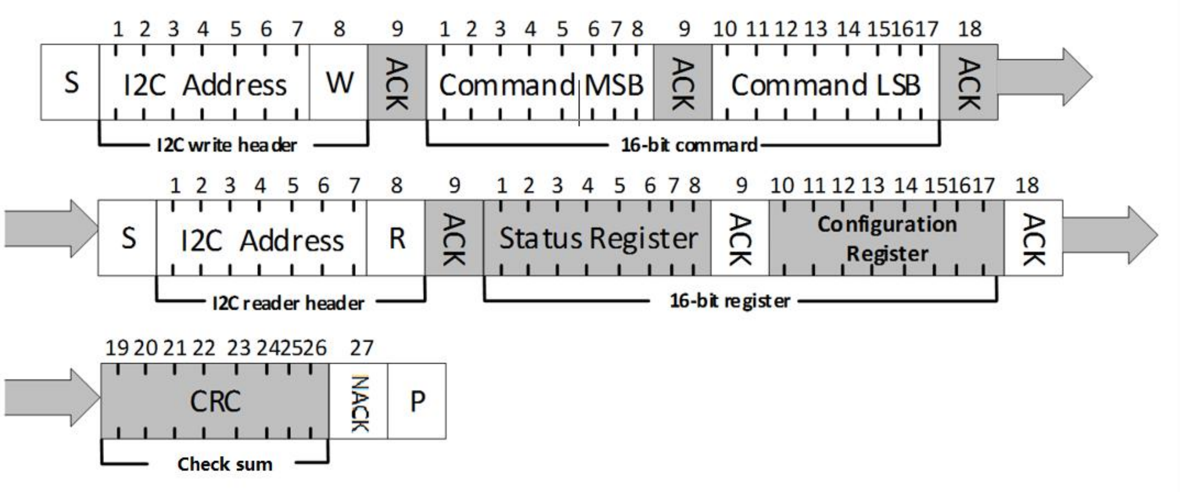
Command	Hex Code
Set configuration register	0x5206



12.6. Read status register and configuration register

The commands to read the status register and configuration register are shown in the following table.

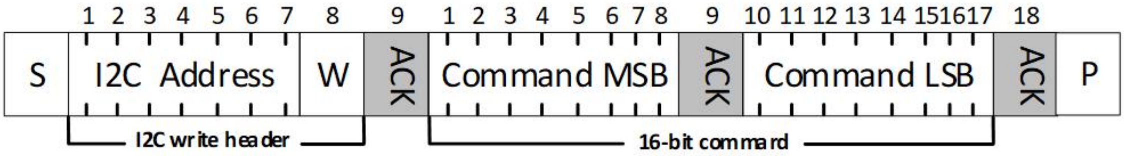
Command	Hex Code
Read status and config register	0xF32D



12.7. Reset status register

The flag (bits 3,2,1) in the status register can be cleared (set to zero) by sending the command shown in the following table.

Command	Hex Code
Reset status register	0x3041

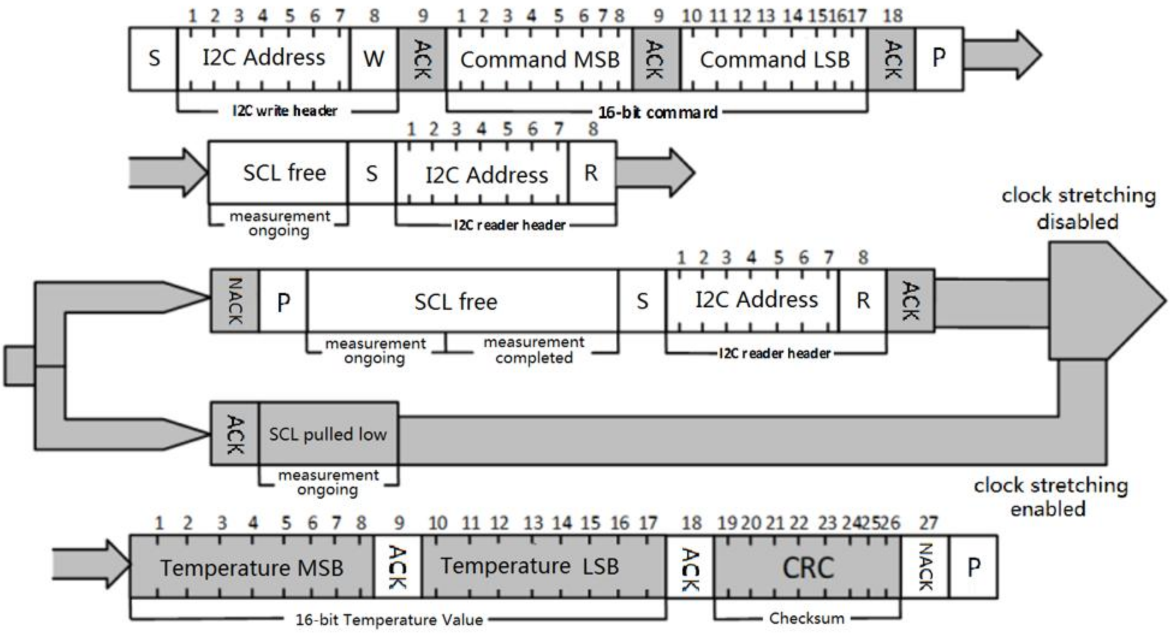


Note: the white block is controlled by the microcontroller, and the sensor response is gray block.

12.8. Single temperature measurement

In the mode, an issued measurement command triggers a temperature data acquisition. Each temperature data is a signed 16 bit binary number. During transmission, each data value always follows the CRC checksum. The 16 bit command is shown in the following table. Repeatability (low, medium and high) and clock stretch (enabled or disabled) can be achieved by modifying the configuration register. The repeatability setting affects the measurement duration, thus affecting the total energy consumption of the sensor. It is explained in Section 9.2.

Command	Hex Code
Temperature measurement	0xCC44



Note: the first "SCL free" block indicates the temperature conversion time(See Table 9.2-2) , the white block is controlled by the microcontroller, and the sensor response is gray block.

12.9. Reading data in single measurement mode

After the sensor completes the measurement, the host can read the measurement results by sending the start condition and then sending the I2C reading head. After the reading temperature command is sent, the sensor will confirm the reception of the reading header and send two bytes of data (temperature), followed by a byte of CRC checksum. Each byte must be acknowledged by the host and have an ACK condition for the sensor to continue sending data. If the sensor does not receive an ACK from the host after any data bytes, it will not continue to send data.

After receiving the check sum of the temperature data, the non-acknowledge (NACK) and stop conditions shall be sent (see 12.8). If you are not interested in the subsequent data, the I2C master can suspend the read transmission after any data bytes in a Non ACK condition. For example, do not read the second byte or CRC byte of the measurement result to save time.

Clock extension mode off
When the clock extension is turned off, after the measurement command is issued, if the temperature measurement is not finished, the sensor will respond to the non response (NACK) reading header.

Clock extension mode on
The clock extension mode is enabled by setting bit 5 of the configuration register to 1. When the clock extension is turned on, after the measurement command is issued, the sensor responds to the read header through the ACK, and then pulls down the SCL line until the measurement is completed. Once the measurement is completed, the sensor will release the SCL line and send the measurement result.

12.10. Continuous temperature measurement mode

The sensor continuous measurement mode is enabled by setting bits 2, 3 and 4 of the configuration register. They differ in repeatability (low, medium, and high) and data acquisition frequency (0.5, 1, 2, 4, and 10 measurements per second). Clock stretch is off in this mode..Data acquisition frequency and repeatability settings affect the measurement duration and current consumption of the sensor. This is explained in Section 9.2.

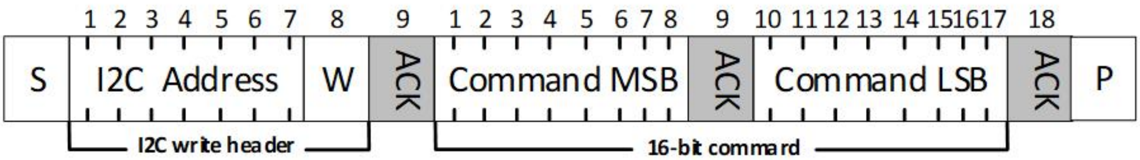
12.11. Read data in continuous measurement mode

In the continuous measurement mode, the host can read the measurement results by sending the start condition and then sending the I2C reading head.

12.12. Stop continuous measurement mode

The stop command shown in the following table to stop the continuous measurement mode. After receiving the stop command, the sensor will stop the measurement in progress but will not enter the single measurement mode. After sending the measurement command again, it will enter the continuous measurement mode again. If you want to switch to the single measurement mode, you need to set the control bits related to the configuration register.

Command	Hex Code
Stop continuous measurement	0x3093



Note: the white block is controlled by the microcontroller, and the sensor response is gray block.

12.13. Reset

The system reset signal of the sensor can be generated Internally by issuing a command of soft reset. In addition, during power on, the chip internal generation system is reset initially. During reset, the sensor does not process commands.

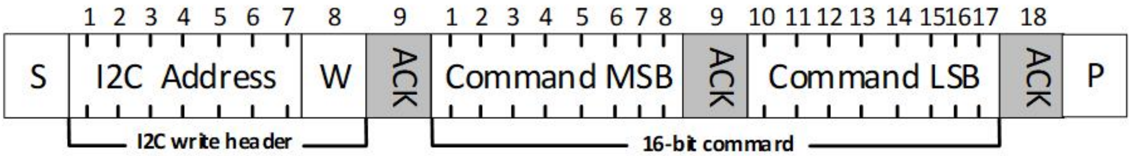
12.13.1. Reset interface

If the communication with the device is abnormal, the following signal sequence will reset the serial interface: switch the SCL nine or more times when the SDA remains high. This method must follow the transfer start sequence before the next command. This sequence only resets the interface and the status register retains its contents.

12.13.2. Soft reset/Re-initialization

The sensor provides a soft reset mechanism, which forces the system to enter a clearly defined state under the condition of continuous power supply. When the system is idle, the soft reset command can be sent to the sensor. This triggers the sensor to reset its system controller and reload calibration data from memory. In order to start the soft reset process, the commands shown in the following table should be sent.

Command	Hex Code
Software reset	0x30a2

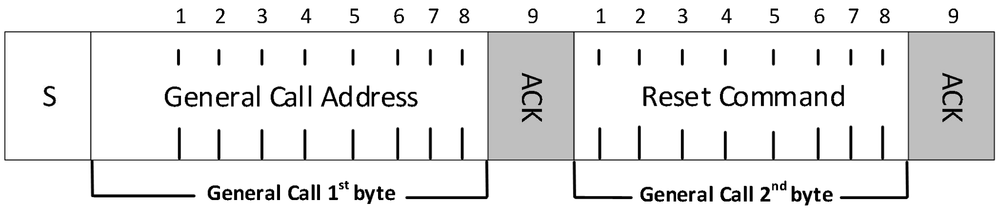


Note: the white block is controlled by the microcontroller, and the sensor response is gray block.

12.13.3. General reset

According to the I2C bus specification, the "general call" mode can also be used to generate a sensor reset. A reset generated in this way is not device specific. All devices on the same I2C bus that support universal call mode will perform a reset. In addition, this command is only valid if the sensor is able to process the I2C command. The appropriate command consists of two bytes, as shown in the following table.

Command	Code
Address Byte	0x00
The second byte	0x06
The universe reset call command	0x0006



Note: the white block is controlled by the microcontroller, and the sensor response is gray block.

12.13.4. Reset by nReset Pin

Pulling the nReset pin down, see section 4.4., will result in a reset similar to a hard reset. nReset pin is internally connected to VDD through pull-up resistor, so it is effective for low level. The nReset pin must be pulled down at least 1 μ s to produce a sensor reset.

12.13.5. Hard reset

The hard reset is realized by turning off the power supply voltage and then turning it on again. To prevent the sensor from being powered by the ESD diode, the external voltages of pins 1 (SDA), 4 (SCL) and 2 (addr) need to be removed.

12.14. Write alarm threshold and read commands

Command	Hex Code
Read alarm set upper threshold	0xE11F
Read alarm clear upper threshold	0xE114
Read alarm lower threshold	0xE102
Read alarm clear lower threshold	0xE109

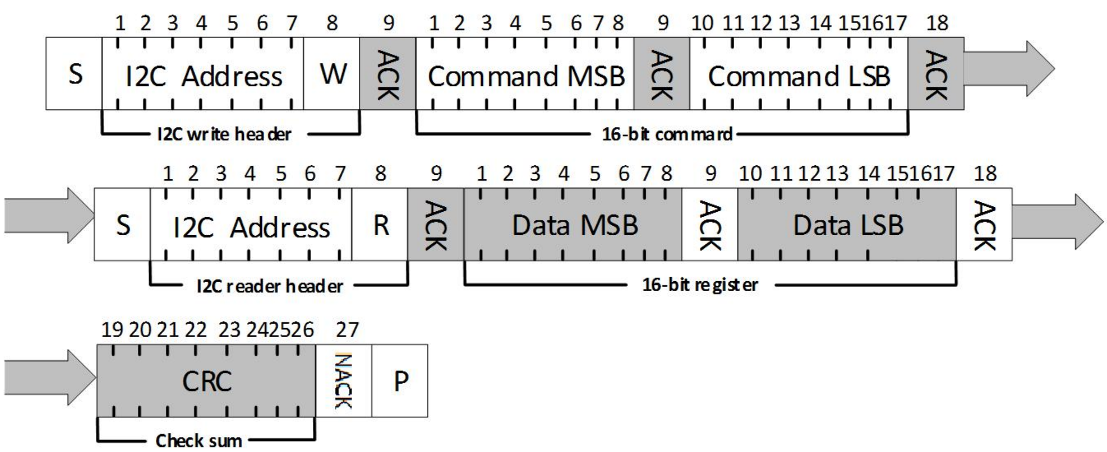


Fig. 12.14-1 Read alarm threshold command（*Note: the white block is controlled by the microcontroller, and the sensor response is gray block.*）

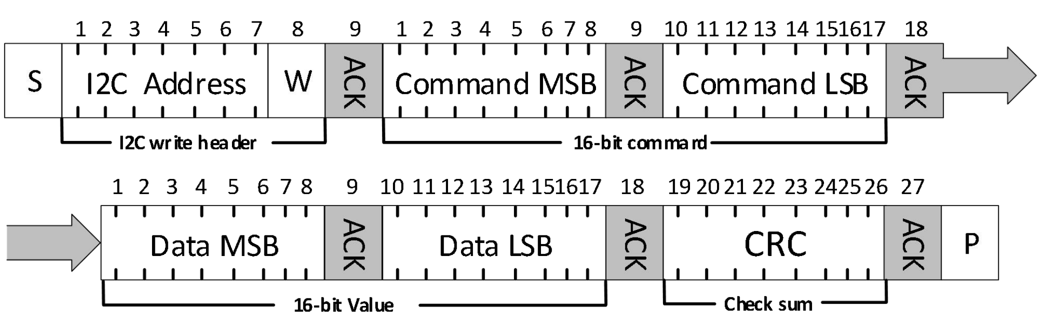


Fig. 12.14-2 Write alarm threshold command（*Note: the white block is controlled by the microcontroller, and the sensor response is gray block.*）

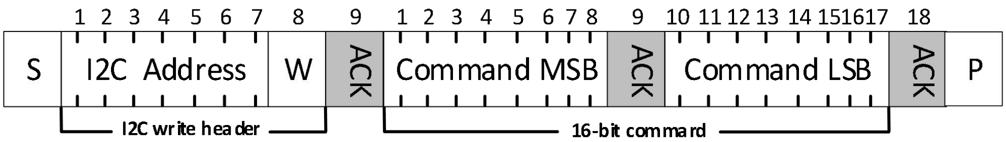
12.15. Register store and recovery commands

All data written to the register (working configuration and alarm threshold) is temporary. If you want these settings to stay permanently, that is, they can still be saved after power failure, you need to use the copy command to store these data in E2PROM.

When hard reset or soft reset, the system automatically loads E2PROM data into the register. In addition, data can be recovered from E2PROM to register with recovery instruction.

Taleb 12.15 Register store and recovery commands

Command	HEX Code
Copy to E^2PROM	0xcc48
Restore E^2PROM area	0xccb8
Restore the rest area	0xccb6



12.16. I^2C sequence characteristics

Table 12.16 I^2C bus timing Characteristics [1]

Parameter	Symbol	Standard mode		Fast mode		Unit
		Min	Max	Min	Max	
SCL frequency	f_{SCL}	0	100	0	400	kHz
SCL low level time	t_{LOW}	4.7[7]	—	1.3[7]	—	μs
SCL high level time	t_{HIGH}	4.0	—	0.6	—	μs
Time interval from SCL pull down to SDA data change	$t_{HD;STA}$	4.0 30[6]	—	0.6 30[6]	—	μs
Time interval from SDA data stabilization to SCL pull up	$t_{HD;DAT}$	5.0 0[2]	3.45[3]	— 0[2]	— 0.9[3]	μs μs
High level holding time of SDA and SCL at the beginning of restart	$t_{SU;DAT}$	250	—	100[4]	—	ns
High level holding time of SDA and SCL at the beginning of restart	$t_{SU;STA}$	4.7	—	0.6	—	μs
Time interval from SCL pull to SDA pull during stop	$t_{SU;STO}$	4.0	—	0.6	—	μs
Interval between start and stop	t_{BUF}	4.7	—	1.3	—	μs

Time required for rising edge of SCL / SDA	t_R	—	1000	$20+0.1C_b$[5]	—	ns
Time required for falling edge of SCL / SDA	t_F	2.5	300	$20+0.1C_b$[5]	—	ns

Note 1: all values are based on VIHmin and VILmax.
Note 2: the device must increase the holding time of at least 300ns internally to SDA.
Note 3: the maximum tHD;DAT must be met only when the SCL low level cycle (TLOW) is not stretched.
Note 4: fast device mode I2C device can work in standard mode, but the requirements of tSU;DAT= 250ns must be met.
Note 5: CB = total capacitance of I2C bus.
Note 6 : For application with sleep-mode enable, to guarantee the device can be waked-up successfully, the minimum value of tHD;STA is reverse proportional to VDD level. The minimum value of tHD;STA is 30us when VDD=5V, 50us when VDD=3V, and 150us when VDD=1.8V, respectively. The details are described in FAQ.
Note 7: For application with sleep-mode enable, if the master uses hardware-based I2C and the parameter tHD;STA cannot be changed, to guarantee the device can be waked-up successfully, the minimum value of the first tLOW (the time from SCL first pull-low to SCL first pull-high after I2C start) is reverse proportional to VDD level. The minimum value of tLOW is 30us when VDD=5V, 50us when VDD=3V, and 150us when VDD=1.8V, respectively. The details are described in FAQ.

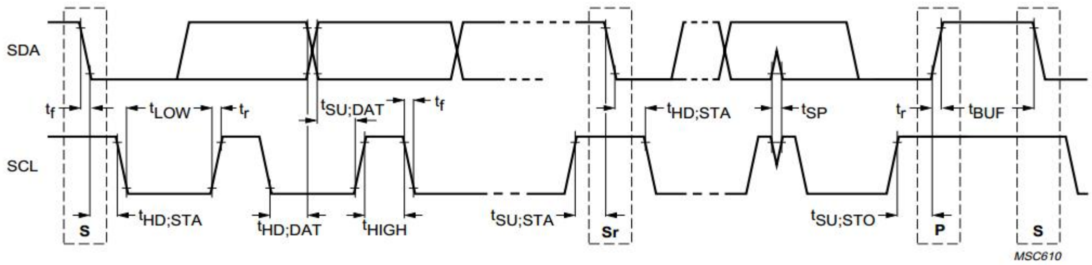
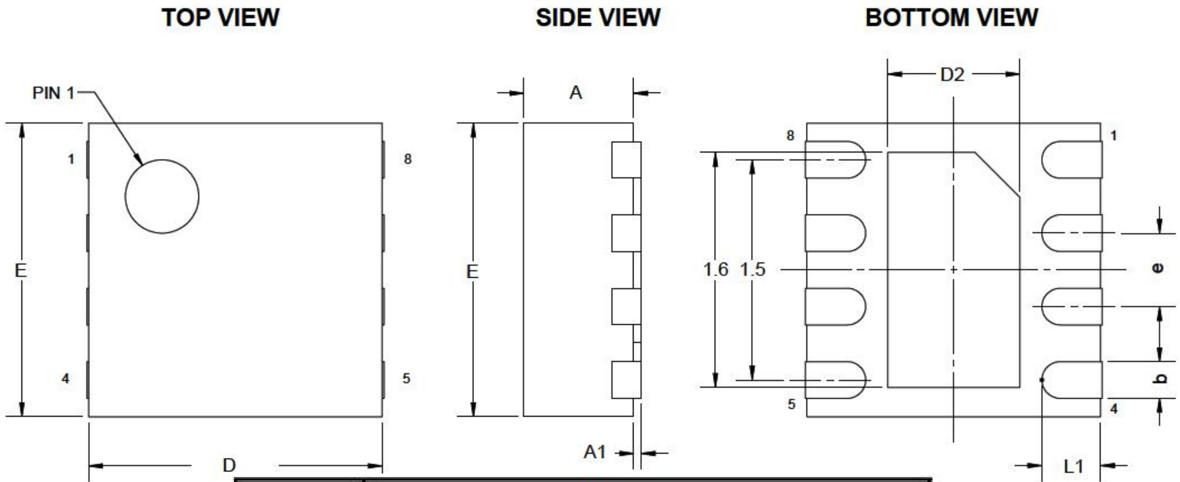


Fig. 12.16 I^{2}C sequence

13. Package

13.1. M601 package and PCB pad pin size diagram

DFN8（2X2X0.55mm）



Symbol	Millimeters		
	Min	Nom	MAX
A	0.500	0.550	0.600
A1			0.050
b	0.230	0.250	0.270
C	0.150		
D	1.900	2.000	2.100
D2	0.870	0.890	0.910
E	1.900	2.000	2.100
E2	1.480	1.500	1.520
e	0.50BSC		
L1	0.200	0.300	0.400
L2	0.105	0.125	0.145

Fig. 13.1-1 Package size diagram DFN8（2X2X0.55mm）

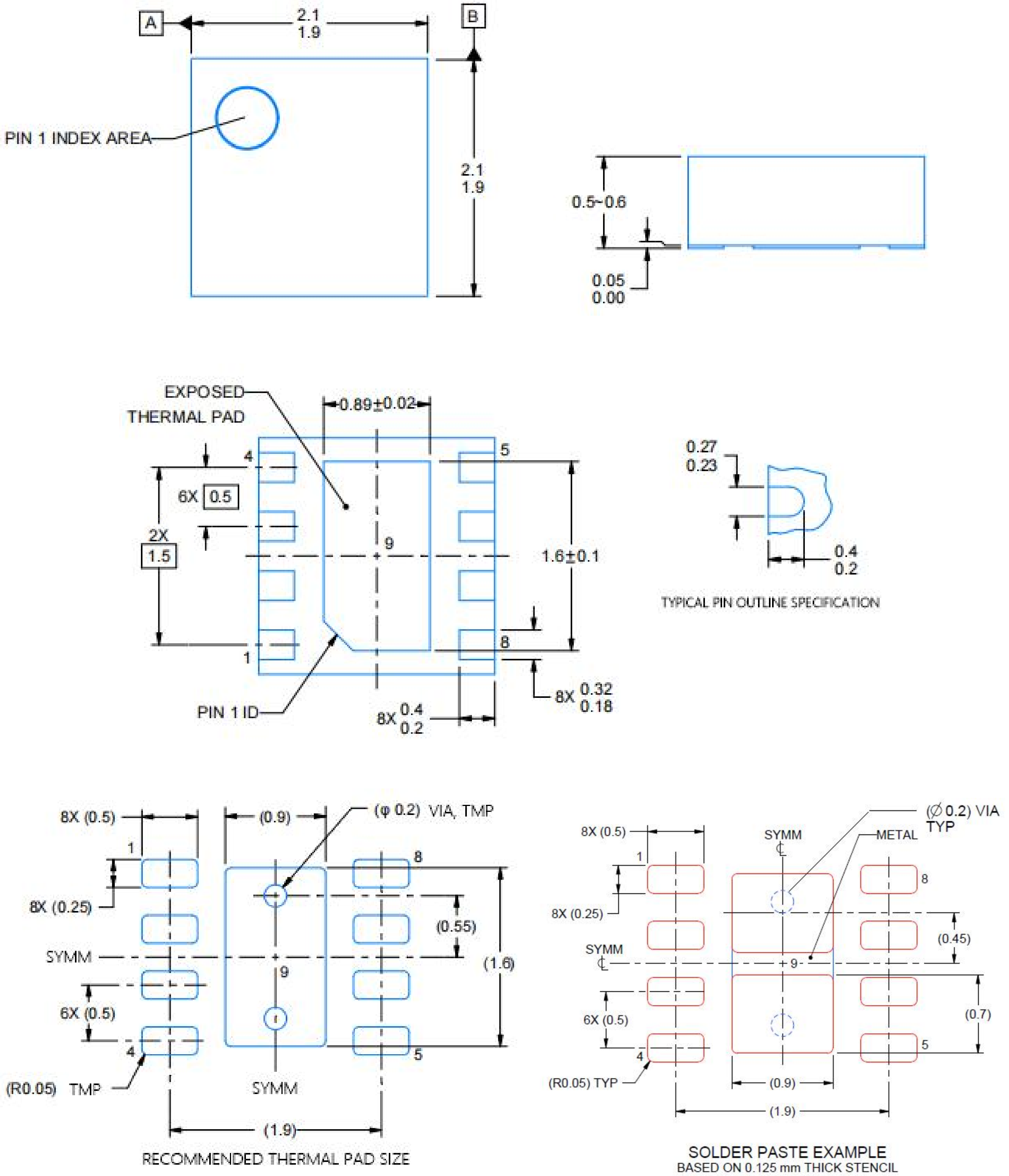
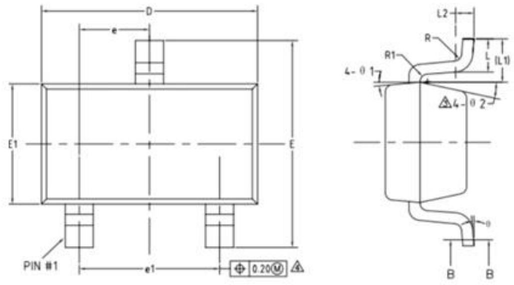
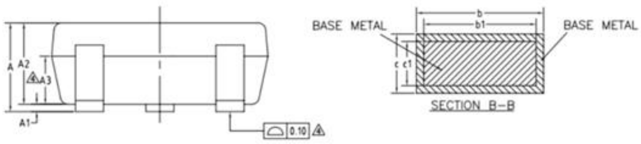


Fig. 13.1-2 PCB pad pin size diagram

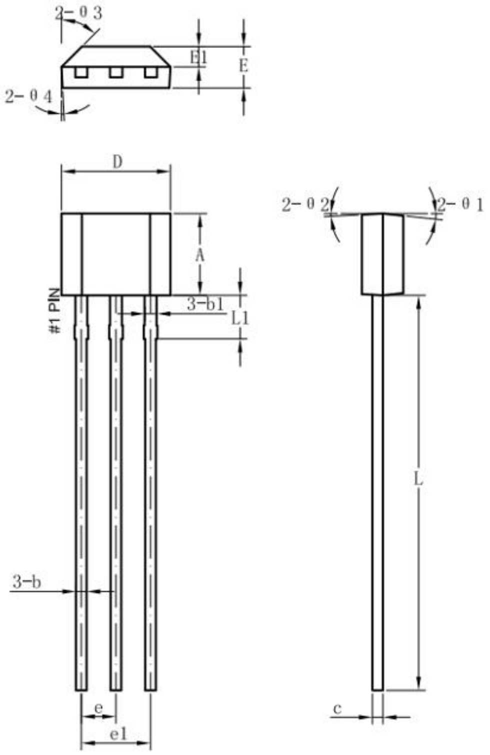
13.2. M1601 package - SOT23-3 (2.9X2.8X1.1mm)





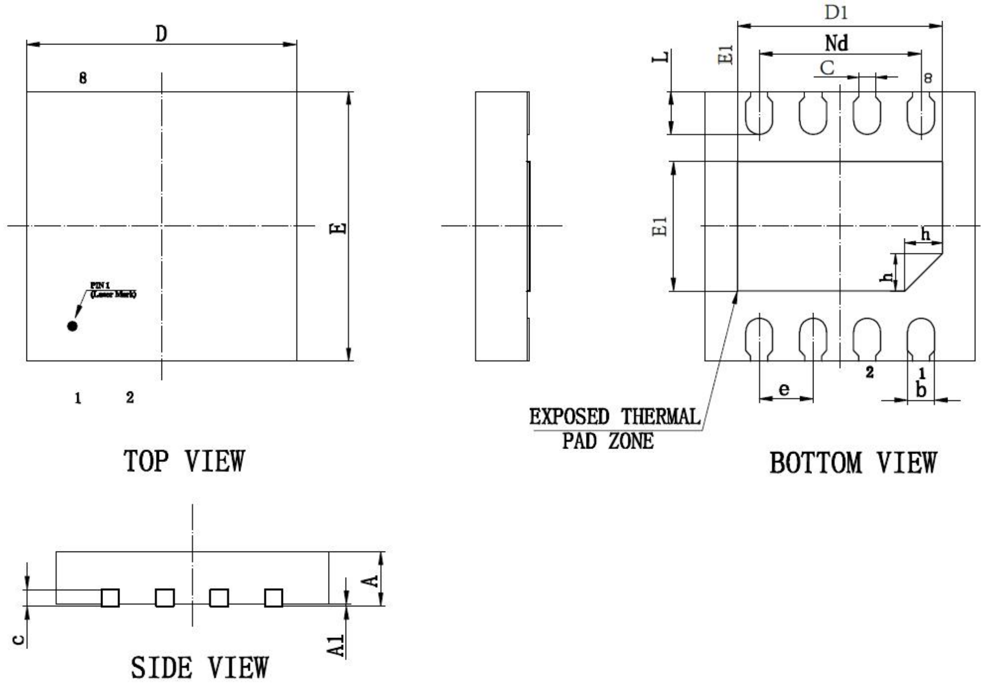
Symbol	Millimeters		
	Min	Nom	MAX
A	—	—	1.250
A1	0	0.075	0.150
A2	1.000	1.100	1.200
A3	0.600	0.650	0.700
b	0.360	0.430	0.500
b1	0.360	0.380	0.450
c	0.140	0.170	0.200
c1	0.140	0.150	0.160
D	2.826	2.926	3.026
E	2.600	2.800	3.000
E1	1.526	1.626	1.726
e	0.900	0.950	1.000
e1	1.800	1.900	2.000
L	0.350	0.450	0.600
L1	0.59REF		
L2	0.25BSC		
R	0.050	—	—
R1	0.050	—	0.200
θ	0°	—	8°
θ1	3°	5°	7°
θ2	6°	—	14°

13.3. M1820 package - TO92S



符号	单位：毫米			单位：英寸		
	最小值	典型值	最大值	最小值	典型值	最大值
A	2.900	3.000	3.100	0.114	0.118	0.122
b	0.350	0.390	0.560	0.014	0.015	0.022
b1		0.440			0.017	
c	0.360	0.380	0.510	0.014	0.015	0.020
D	3.900	4.000	4.100	0.154	0.157	0.161
E	1.420	1.520	1.620	0.056	0.060	0.064
E1		0.750			0.030	
e		1.270			0.050	
e1		2.540			0.100	
L	13.500	14.500	15.500	0.531	0.571	0.610
L1		1.600			0.063	

13.4. MTS01 package - DFN8 (2.5X2.5X0.7mm)



Symbol	Millimeters		
	Min	Nom	Max
A	0.60	0.70	0.80
A1	0.19	0.20	0.21
b	0.20	0.25	0.30
c	—	0.15	0.20
D	2.40	2.50	2.60
E	2.40	2.50	2.60
D1	1.80	1.90	2.00
E1	1.00	1.10	1.20
e	0.50		
Nd	1.50		
X	0.50	0.60	0.70
Y	1.00	1.10	1.20
L	0.30	0.35	0.40
N	8.00		