

M117 M117Z M117W M117B

±0.1°C/±0.5°C精度、16bitADC、超低功耗、I²C 接口

数字温度传感芯片

1. 概述

M117、M117Z、M117W、M117B 是数字模拟混合信号温度传感芯片，M117、M117Z、M117W 最高测温精度±0.1°C，M117B 最高测温精度±0.5°C，用户无需进行校准。

温度芯片感温原理基于 CMOS 半导体 PN 结温度与带隙电压的特性关系，经过小信号放大、模数转换、数字校准补偿后，数字总线输出，具有精度高、一致性好、测温快、功耗低、可编程配置灵活、寿命长等优点。

温度芯片内置 16-bit ADC，分辨率 0.004°C，具有 -70°C 到 +150°C 的超宽工作范围。芯片在出厂前经过 100% 的测试校准，根据温度误差特性进行校准系数的拟合，芯片内部自动进行补偿计算。芯片支持数字 I²C 通信接口、测温数据内存访问、功能配置等均可通过数字协议指令实现。I²C 接口适合高速率的板级应用场景，最高接口速度可达 400kHz。

芯片内置非易失性 E²PROM 存储单元，用于保存芯片 ID 号、高低温报警阈值、温度校准修正值以及用户自定义信息，如传感器节点编号、位置信息等。芯片另有 ALERT 报警指示引脚，便于用户扩展硬件报警应用。

2. 特性

- 最高测温精度：±0.1°C/±0.5°C
- 测温范围：-70°C ~ +150°C
- 低功耗：典型待机电流 0.1μA@3.3V，测温峰值电流 0.45mA@3.3V，测温平均电流 5.2μA (@3.3V, 1s 周期)
- 宽工作电压范围：1.8V-5.5V
- 感温分辨率：16 位输出 0.004°C
- 温度转换时间可配置：10.5ms/5.5ms/4ms
- 可配制单次/周期测量
- 用户可设置温度报警

- 32 bit 额外 E²PROM 空间用于存放用户信息
- 标准 I²C 接口

3. 应用

- 智能穿戴
- 电子体温计
- 动物体温检测
- 医疗电子
- 冷链物流、仓储
- 智能家居
- 热表气表水表
- 替代 PT100/PT1000
- 板级温度监控
- 工农业环境温度
- 智能家电
- 消费电子
- 测温仪器仪表

产品信息

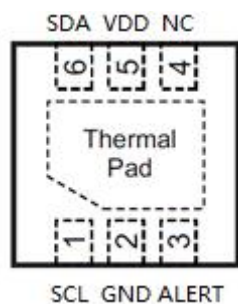
型号	最高精度区间	封装	尺寸
M117	±0.1°C@+28°C to +43°C	DFN6L	2*2*0.75mm
M117Z	±0.1°C@0°C to +50°C	DFN6L	2*2*0.75mm
M117W	±0.1°C@+20°C to +70°C	DFN6L	2*2*0.75mm
M117B	±0.5°C@0°C to +50°C	DFN6L	2*2*0.75mm

备注：其他温度区间精度特性详见“测温性能指标”。

目录

1. 概述	1
2. 特性	1
3. 应用	1
4. 封装管脚描述及实物图	3
5. 结构框图	3
6. 典型应用电路	4
7. 测温性能指标	4
8. 电气规格	6
8.1 绝对最大额定值	6
8.2 电气特性	6
8.3 交流电气特性-非易失性存储器	7
9. 运行-测量温度	7
9.1 温度输出和转换公式	7
9.2 配置寄存器和状态寄存器	8
9.3 报警	9
10. 存储系统	12
11. I ² C 总线协议	13
11.1. I ² C 命令集	13
11.2. 操作与通信	14
11.3. 上电及通信起始	14
11.4. 开始测量	14
11.5. 设定配置寄存器指令	15
11.6. 读取状态寄存器和配置寄存器指令	15
11.7. 复位状态寄存器指令	15
11.8. 单次测量模式指令	15
11.9. 单次测量模式下读取数据	16
11.10. 连续测量模式指令	17
11.11. 连续测量模式下读取数据	17
11.12. 停止连续测量模式指令	17
11.13. 复位	17
11.13.1. 接口复位	17
11.13.2. 软复位/重新初始化	17
11.13.3. 通用呼叫复位	18
11.13.4. 硬复位	18
11.14. 报警限写入和读取命令	18
11.15. 寄存器保存和恢复指令	19
11.16. I ² C 时序特性	20
12. 封装及 PCB 焊盘引脚尺寸图	21

4. 封装管脚描述及实物图

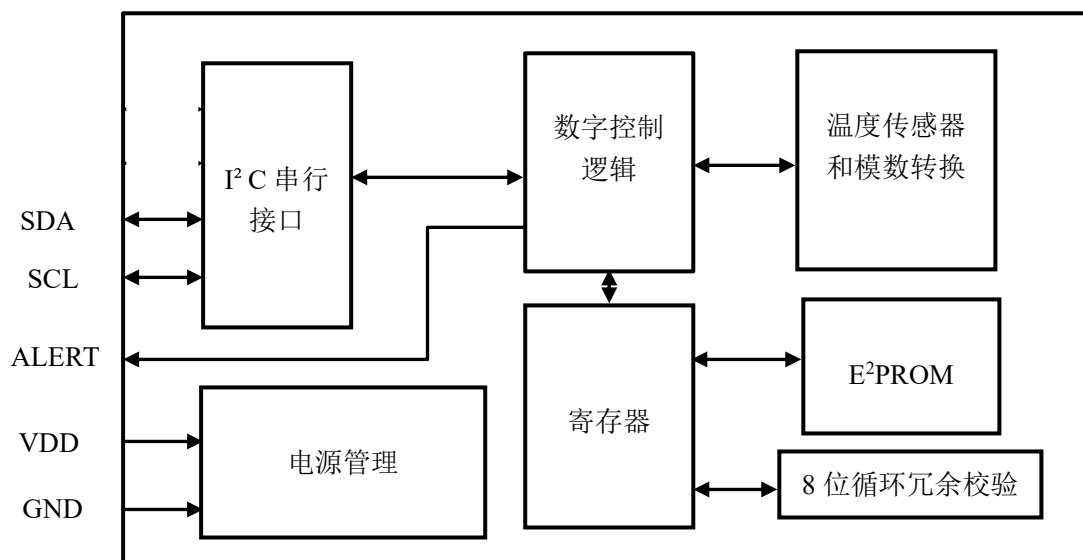


(正面透视图)

管脚编号	管脚名称	I/O	说明
1	SCL	输入/输出	I ² C 时钟线
2	GND	—	地
3	ALERT	输出	报警指示
4	NC	—	悬空
5	VDD	—	电源
6	SDA	输入/输出	I ² C 数据线
导热焊盘	NC	—	悬空或接地

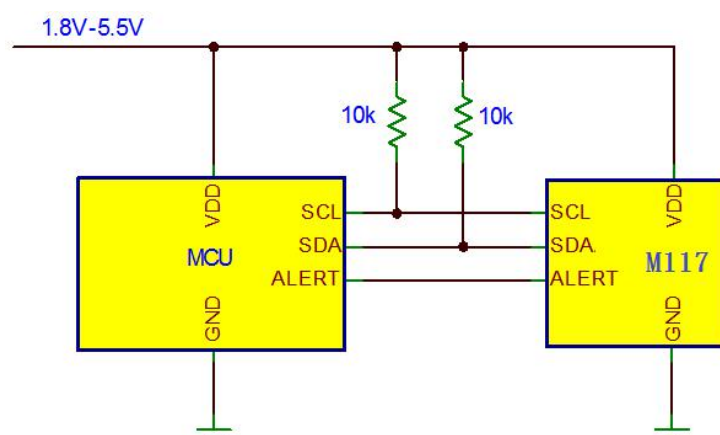
备注：导热焊盘的导热设计建议可参见 FAQ。

5. 结构框图



温度传感器的原理框图见上图。暂存器包含了两个字节的温度寄存器，存储来自于传感器的数字输出。另外，暂存器和扩展暂存器提供了报警触发阈值寄存器。配置寄存器允许用户设定温度转换重复性和连续测量频率。状态寄存器可以查询报警状态。数据可存入非易失性单元，芯片掉电时数据不会丢失。

6. 典型应用电路



备注：1、M117/ M117Z/M117W I²C 地址固定为 0x44；
2、M117B I²C 地址固定为 0x45。

7. 测温性能指标

参数	符号	条件	最小	典型	最大
测温范围	—	—	-70℃	—	+150℃
温度误差	t _{ERR}	M117	—	—	±0.1℃@+28℃ to +43℃ ±0.5℃@-10℃ to +60℃
		M117Z	—	—	±0.1℃@0℃ to +50℃ ±0.2℃@-10℃ to +60℃ ±0.5℃@-30℃ to +70℃
		M117W	—	—	±0.1℃@+20℃ to +70℃ ±0.2℃@0℃ to +75℃ ±0.5@-20℃ to +90℃
		M117B	—	—	±0.5℃@0℃ to +50℃
		M117 M117Z M117W M117B	—	±1.5℃@-55℃ to +125℃ ±2℃@-70℃ to +150℃	— —
重复性	—	低重复性设置	—	0.07℃	—
		中重复性设置	—	0.05℃	—
		高重复性设置	—	0.03℃	—
分辨率	—	—	—	0.004℃	—
响应时间	τ ₆₃	—	—	—	2s
长期漂移	—	—	—	—	0.03℃/年

备注：重复性设置中，通过配置不同滤波带宽，实现不同输出精度，参见表 11.10-2。

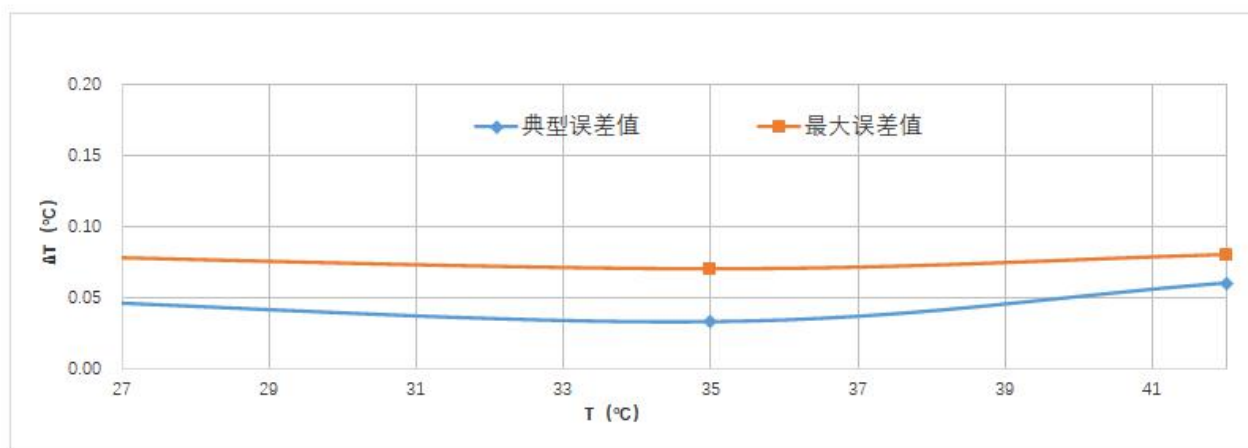


图 7-1 M117 精度误差曲线

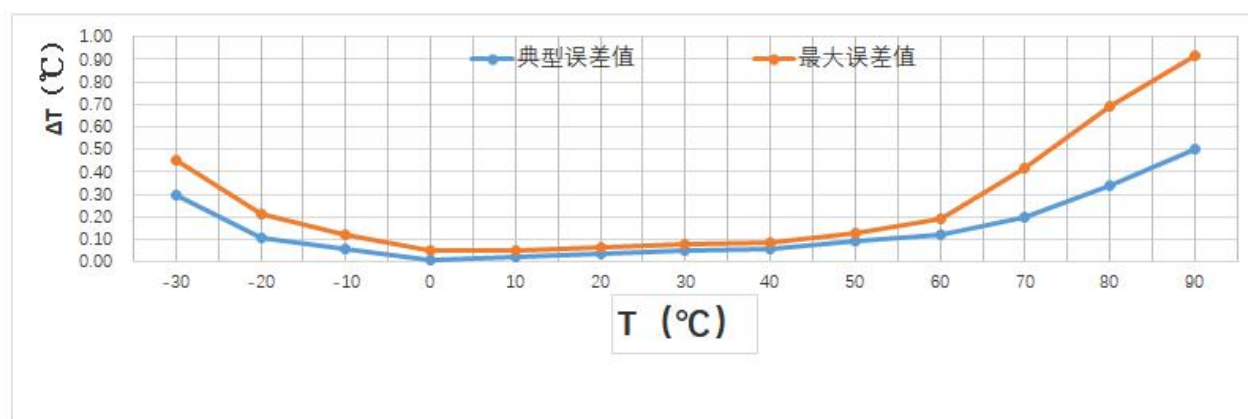


图 7-2 M117Z 精度误差曲线

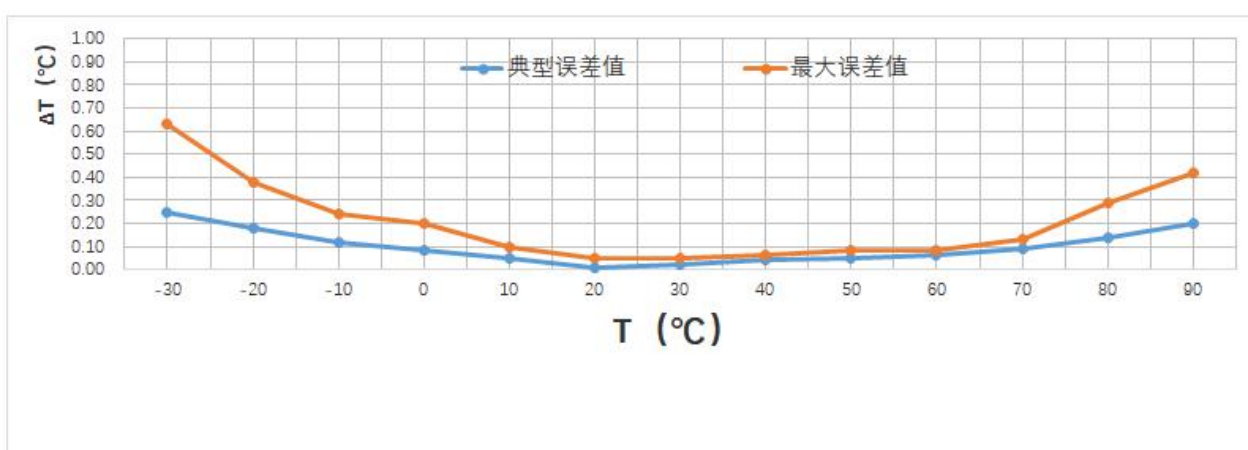


图 7-3 M117W 精度误差曲线

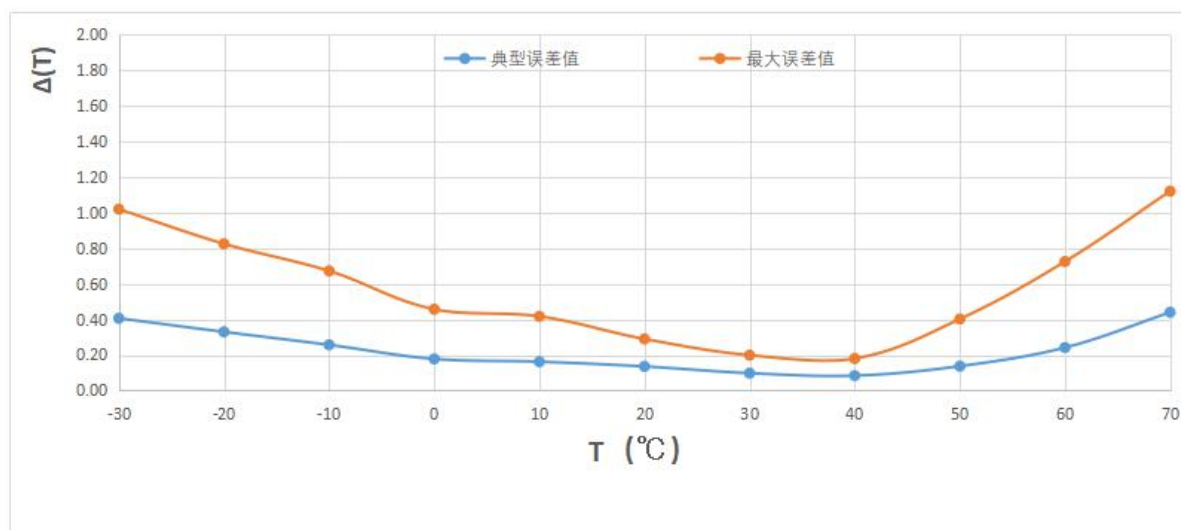


图 7-4 M117B 精度误差曲线

8. 电气规格

8.1 绝对最大额定值

以下为极限参数，对于器件在此极限条件或高于此极限条件的环境中的功能运行，本规格书并不适用。请注意长期暴露于此极限环境会影响器件的可靠性。

参数	额定值	单位
供电电压 VDD	-0.3 to 6	V
引脚上的最大电压	-0.3 to 6	V
引脚上的输入电流	±100	mA
运行温度范围	-70 to 150	°C
存储温度范围	-70 to 150	°C
焊接温度	参考 IPC/JEDEC J-STD-020 规范	
ESD HBM (人体放电模式)	±8	kV

8.2 电气特性

参数	符号	条件	最小	典型	最大	单位	备注
电源							
供电电压	VDD		1.8	3.3	5.5	V	
电源压摆率变化	VDD slew	—	—	—	20	V /ms	VDD 线上 DDmin 和 VDDmax 之间的电

							压变化应该比最大压摆率慢，更快的压摆率可能导致复位。
供电电流	IDD	待机	—	0.1	1	uA	传感器在单次测量模式下不进行测量时的电流
		周期测量模式	—	55	—	uA	传感器在周期测量模式下不进入睡眠
		测量峰值	—	447	—	uA	
		平均	—	5.2	—	uA	单次模式, 高重复性, 1 次测量/s
数字输入/输出							
输入逻辑低	VIL	SCL, SDA	—	—	0.3*VDD	V	
输入逻辑高	VIH	SCL, SDA	0.7*VDD	—	—	V	
输出低电平电压	VOL	IOL = -3 mA	—	—	0.4	V	
输入漏电流	IIN	—	-0.1	—	0.1	uA	
报警输出驱动强度	IOH	—	—	1.5* VDD	—	mA	
上拉电阻	Rup		1	10	100	kΩ	

8.3 交流电气特性-非易失性存储器

-55°C到+125°C; $V_{DD}=1.8V$ 到 5.5V

参数	符号	条件	最低	典型	最大	单位
非易失存储写周期	t_{WR}	—	—	—	40	ms
E ² PROM 写次数	N_{EEWR}	-55°C 到 +55°C	50000	—	—	次
E ² PROM 数据保留	t_{EEDR}	-55°C 到 +55°C	—	10	—	年

9. 运行-测量温度

传感器上电后进入空闲状态，如要启动温度测量，主机必须发出 Convert Temperature 指令。经过转换时间后，产生的 16 位温度数据被存储在暂存器的 2 个字节的温度寄存器中。转换时间和重复性设置相关，重复性越高，转换时间越长。上电后默认是高重复性配置。重复性设置参见“状态寄存器和配置寄存器”，转换时间参见“时间特性表”。

9.1 温度输出和转换公式

温度数字输出为 16bit 有符号的二进制补码，最低位 LSB 分辨率为 1/256 °C，S 为符号位。数据存放在暂存器 Temp_lsb, Temp_msb 中。

表 9.1 温度寄存器格式

	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
低字节	2 ⁻¹	2 ⁻²	2 ⁻³	2 ⁻⁴	2 ⁻⁵	2 ⁻⁶	2 ⁻⁷	2 ⁻⁸
	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8
高字节	S	2 ⁶	2 ⁵	2 ⁴	2 ³	2 ²	2 ¹	2 ⁰

和摄氏度的转换关系为:

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

例如, 40 °C 对应寄存器值

0x 00 00

150 °C 对应寄存器值

0x 6E 00

-70 °C 对应寄存器值

0x 92 00

9.2 配置寄存器和状态寄存器

配置寄存器包含报警功能, I²C 时钟延展, 周期测量频率, 以及重复性等设置位。该配置寄存器只能设定 I²C 模式下的连续测量功能配置, 具体配置寄存器的内容描述见下表。

表 9.2-1 配置寄存器 (Cfg)

位	内容描述	默认数值
7	1: 温度报警功能开启 0: 温度报警功能关闭	'0'
6	预留	'0'
5	时钟延展功能 0: 关闭 1: 开启	'0'
4:2	每秒测量次数配置 MPS 000: 单次 001: 每秒 0.5 次 010: 每秒 1 次 011: 每秒 2 次 100: 每秒 4 次 101: 每秒 10 次	'000'
1:0	重复性设置 Repeatability 00: 低可重复性	'00'

	01: 中等可重复性 10: 高可重复性	
--	-------------------------	--

请注意分辨率和转换时间是直接折衷关系（分辨率越高，转换时间越长；分辨率越低，转换时间越短）。

表 9.2-2 转换时间和重复性设置

重复性设置		精度	转换时间 t_{CONV}
0	0	低	4ms
0	1	中	5.5ms
1	0	高	10.5ms

状态寄存器包含测量命令状态，警报状态以及最后一个命令执行和最后一个写序列的状态等信息。

表 9.2-3 状态寄存器 (Status)

位	内容描述			默认数值
7:6	保留			'00'
5	I ² C 写数据校验	0: 正确	1: 错误	'0'
4	I ² C 命令状态	0: 错误	1: 正确	'0'
3	系统复位检测	0: 未检出	1: 检出	'0'
2	温度报警跟踪	0: 温度报警未触发	1: 温度报警触发	'0'
1	保留			'0'
0	周期测量指令状态	0: 不测量	1: 测量温度	'0'

9.3 报警

通过可编程门限，警报模式允许监测环境温度。当达到门限时，专用 ALERT 引脚的输出电平将发生变化。此外，状态寄存器位有专门一位指示报警状态。使用 ALERT 引脚可以控制一个开关。或者可以连接到微控制器的中断引脚。在传感器发出警报后，微控制器可以从睡眠模式唤醒，然后执行指定操作。

只要传感器进行了测量操作，警报模式就会激活。通过将最小设定值设置为大于等于最大设定值 ($T_{la_Set} \geq T_{ha_Set}$) 以取消警报模式。

可以通过相应的指令设置报警门限（见 I²C 命令集）。不同的门限如下图所示：

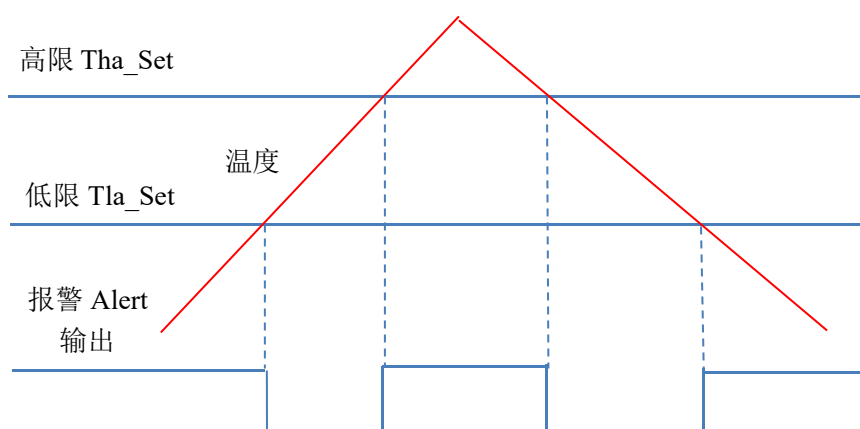


图 9.3-1 警报模式的不同门限

报警门限以精简的格式存储，即仅存储最高有效 9 位，以和 16 位标准输出的最高有效 9 位进行比较，来判断是否已满足报警条件，请参见下图。因此报警门限具有与测量值不同的分辨率。温度报警门限的分辨率为 $\Delta T \approx 0.5^\circ\text{C}$ 。请注意，数据始终以 16 位格式进行测量和存储。简化数据格式仅用于判断是否满足警报条件。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
温度输出高 9 位									X	X	X	X	X	X	X

图 9.3-2 报警门限的相关数据位

报警门限设置需要满足以下条件：

$\text{Tha_Set} > \text{Tha_Clear} > \text{Tla_Clear} > \text{Tla_Set} \geq 40$ 或 $\text{Tla_Set} < \text{Tla_Clear} < \text{Tha_Clear} < \text{Tha_Set} < 40$

下面列举三种计算温度报警门限的典型例子：

示例#1，设置 $\geq 40^\circ\text{C}$ 的报警门限

1) 设置温度报警门限值：

高限触发门限: $\text{Tha_Set} = 60^\circ\text{C}$,

高限清除门限: $\text{Tha_Clear} = 55^\circ\text{C}$,

低限清除门限: $\text{Tla_Clear} = 45^\circ\text{C}$,

低限触发门限: $\text{Tla_Set} = 40^\circ\text{C}$

2) 用下列公式：

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

3) 把 Tha_Set 限制转换为 16 位二进制值：

$$\text{Tha_Set} = 0001'0100'0000'0000$$

4) 删除 Tha_Set 的低 7 位，前面加 7 个 “0 ”：

$$\text{Tha_Set} = \underline{00000000001}0100'0000'0000 = 0000'0000'0010'1000$$

5) 由此， Tha_Set 可以配置为：

$$\text{Tha_Set_msb} = 0000'0000, \text{Tha_Set_lsb} = 0010'1000$$

6) 同样地，其他报警门限可以分别配置为：

$$\text{Tha_Clear_msb} = 0000'0000, \text{Tha_Clear_lsb} = 0001'1110$$

$$\text{Tla_Clear_msb} = 0000'0000, \text{Tla_Clear_lsb} = 0000'1010$$

$$\text{Tla_Set_msb} = 0000'0000, \text{Tla_Set_lsb} = 0000'0000$$

示例#2，设置 $< 40^\circ\text{C}$ 的报警门限

1) 设置温度报警门限值:

高限触发门限: $Tha_Set = 39^{\circ}C$,
 高限清除门限: $Tha_Clear = 34^{\circ}C$,
 低限清除门限: $Tla_Clear = 30^{\circ}C$,
 低限触发门限: $Tla_Set = 25^{\circ}C$,

2) 用下列公式:

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

3) 把 Tha_Set 限制转换为 16 位二进制值:

$$Tha_Set = 1111'1111'0000'0000$$

4) 删除 Tha_Set 的低 7 位, 前面加 7 个 "0 ":

$$Tha_Set = \underline{0000000}1111'1111'0000'0000 = 0000'0001'1111'1110$$

5) 由此, Tha_Set 可以配置为:

$$Tha_Set_msb = 0000'0001, Tha_Set_lsb = 1111'1110$$

6) 同样地, 其他报警门限可以分别配置为:

$$Tha_Clear_msb = 0000'0001, Tha_Clear_lsb = 1111'0100$$

$$Tla_Clear_msb = 0000'0001, Tla_Clear_lsb = 1110'1100$$

$$Tla_Set_msb = 0000'0001, Tla_Set_lsb = 1110'0010$$

示例#3, 设置屏蔽报警模式

1) 设置温度报警门限值:

高限触发门限: $Tha_Set = 38.5^{\circ}C$,
 高限清除门限: $Tha_Clear = 34^{\circ}C$,
 低限清除门限: $Tla_Clear = 30^{\circ}C$,
 低限触发门限: $Tla_Set = 40^{\circ}C$

2) 用下列公式:

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

3) 把 Tha_Set 限制转换为 16 位二进制值:

$$Tha_Set = 1111'1110'1000'0000$$

4) 删除 Tha_Set 的低 7 位, 前面加 7 个 "0 ":

$$Tha_Set = \underline{0000000}1111'1110'1000'0000 = 0000'0001'1111'1101$$

5) 由此, Tha_Set 可以配置为:

$$Tha_Set_msb = 0000'0001, Tha_Set_lsb = 1111'1101$$

6) 同样地, 其他报警门限可以分别配置为:

$$Tha_Clear_msb = 0000'0001, Tha_Clear_lsb = 1111'0100$$

Tla_Clear_msb = 0000'0001, Tla_Clear_lsb = 1110'1100

Tla_Set_msb = 0000'0000, Tla_Set_lsb = 0000'0000

7) 由于 Tla_Set > Tha_Set, 报警模式被屏蔽。

10. 存储系统

传感器的存储组织如表 10.1 所示。存储器包含一个 SRAM 暂存器以及拓展存储的非易失性存储 E²PROM 寄存器, 用于存储高低报警触发值 (TH 和 TL), 配置寄存器和 2 字节用户可编程 E²PROM。需要注意的是, 如果传感器的报警功能没有被使用, 则 TH 和 TL 寄存器可以用作通用存储。所有存储指令都在传感器功能指令章节有详细描述。暂存器字节 0 Temp_lsb 和字节 1 Temp_msb 是只读的温度测量结果。字节 2 和字节 3 是预留的寄存器。字节 4, 5 Tha_Set_lsb, Tla_Set_lsb 是设置报警门限的寄存器。字节 6 Cfg 是配置寄存器, 字节 7 Status 是芯片的状态寄存器, 在 9.2 章节中有详细解释。字节 8 为只读寄存器, 是字节 0 到 7 产生的循环冗余校验码。传感器通过循环冗余校验码生成章节中所述的规则来产生这个码。数据可以通过 write one byte[0x52xx] 指令写入到字节 4,5,6 中。数据传输必须始于字节 2 的最低位。为了验证数据完整性, 可以在数据写操作之后读取暂存器 (通过 read one byte [0xd2xx] 指令)。

要把数据从暂存器永久性写入 E²PROM, 主机必须发出 Copy Scratchpad [0xCC48] 指令。E²PROM 寄存器中的数据掉电会保持, 上电时会自动加载到相应的暂存器位置。数据也可以通过 Recall E2 [0xCCb8] 指令随时重新加载。请注意, Copy Scratchpad [0xCC48]是按页 (16 字节) 操作的, 需要确认 16 个暂存的内容都正确后统一拷贝。

表 10.1 温度传感器存储映射

Name	Logic Addr	Read	Write	E ² PROM	Copy	Recall	Reset Value
Temp_lsb	00	Read one byte (0xd2--)	NA	NA	NA	NA	H' 01
Temp_msb	01		NA	NA	NA	NA	H' F1
Reserved	02		Write one	NA	NA	NA	H' 00
Reserved	03		byte	NA	NA	NA	H' 80
Tha_Set_lsb	04	--为当前字节的逻辑地址	(0x52--),	0	Copy Scratch	Recall	H' 00
Tla_Set_lsb	05		--为当前字节的逻辑地址	1		EE (0xCC	H '00

Cfg	06		NA	2	hpad	B8)	H' 02
Status	07		NA	NA	(0xC	Recall	XX
Crc_src	NA		NA	NA	C48)	Page0	XX
Tha_Clear_Isb	08		Write one	4		Res	H '00
Tla_Clear_Isb	09		byte	5		(0xCC	H '00
Tha_Set_msb	0A		(0x52--),	6		BB)	H '00
Tla_Set_msb	0B		--为当前字	7			H '00
Tha_Clear_msb	0C		节的逻辑地	8			H '00
Tla_Clear_msb	0D		址	9			H '00
Reserved	0E~13			10-15			H '00
Crc_scr_ext	NA		NA	NA	NA	NA	XX

11. I²C 总线协议

11.1. I²C 命令集

命令名	功能	代码
测量温度	Convert Temperature	0xCC44
读测量结果	—	—
设置报警门限上阈值	WRITE_ALERT_HI_SET	0x611d
设置报警清除门限上阈值	WRITE_ALERT_HI_UNSET	0x6116
设置报警清除门限下阈值	WRITE_ALERT_LO_SET	0x6100
设置报警门限下阈值	WRITE_ALERT_LO_UNSET	0x610b
读取报警门限上阈值	READ_ALERT_HI_SET	0xe11f
读取报警清除门限上阈值	READ_ALERT_HI_UNSET	0xe114
读取报警清除门限下阈值	READ_ALERT_LO_SET	0xe102
读取报警门限下阈值	READ_ALERT_LO_UNSET	0xe109
写配置寄存器	CONFIG	0x5206
读取状态和配置寄存器	READ_STATUS	0xf32d
清除状态寄存器	CLEAR_STATUS	0x3041

中止周期测量	BREAK	0x3093
软件复位	SOFT_RST	0x30a2
保存 Page0 到 EEPROM	COPY_PAGE0	0xcc48
恢复 EE 区域	RECALL_EE	0xccb8
恢复 Page0 的扩展区域	RECALL_PAGE0_RES	0xccb6

11.2. 操作与通信

传感器支持 I²C 快速模式（频率可达 400 kHz）。可以通过相应的用户命令启用和禁用时钟延展。在向传感器发送命令之后，在传感器接收另一个命令之前，需要 1ms 的最小等待时间。所有传感器命令和数据都映射到 16 位地址空间。此外，数据和命令包含 CRC 校验，这提高了通信可靠性。传感器发送和接收的数据总是跟随 8 位 CRC 校验和。在写操作时，主机必须向从机发送校验和，只有收到正确的校验和，传感器才接受数据。在读取操作时，由主设备读取并处理校验和。

表 11.2 I²C 循环冗余配置

属性	数值
名称	CRC-8
位宽	8 位
保护数据	读/写
多项式	$0x31 (x^8 + x^5 + x^4 + 1)$
初始数值	0xFF
待测数据的每个字节是否按位反转	否
在计算之后，异或输出之前，整个数据是否按位反转	否
计算结果与此参数异或	0x00
示例	CRC (0xBEEF) = 0x92

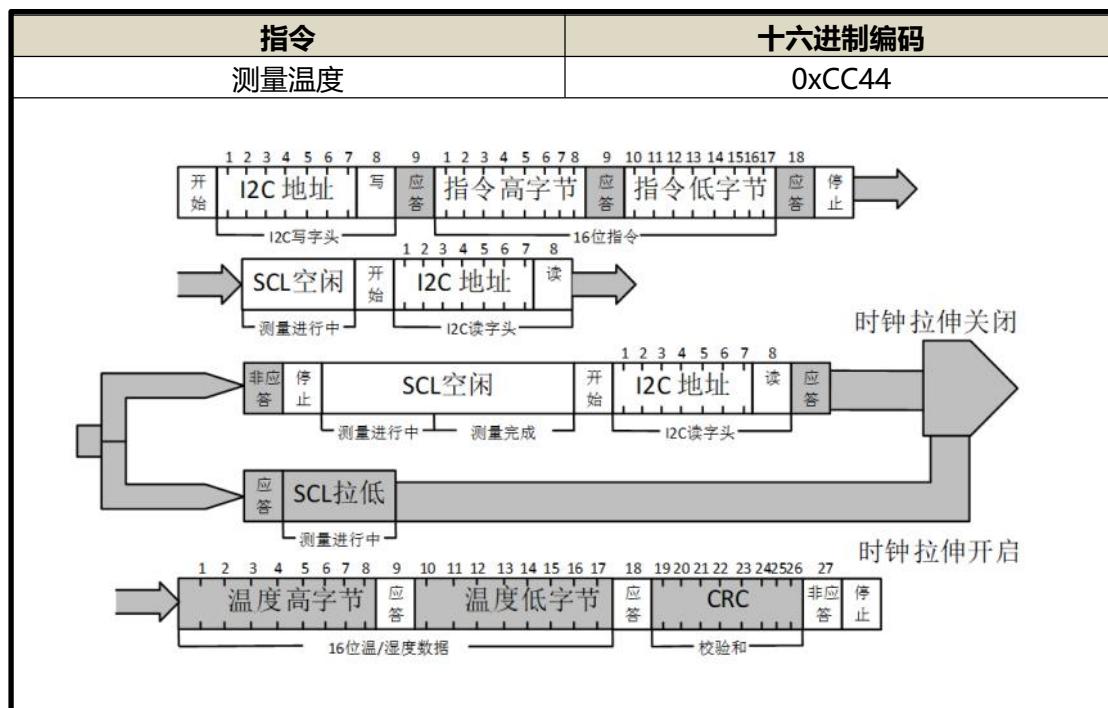
11.3. 上电及通信起始

传感器在达到上电阈值电压 VPOR 后开始上电。在达到该阈值电压后，传感器需要时间 t_{PU} 进入空闲状态。一旦进入空闲状态，就可以从主设备（微控制器）接收命令。每个传输序列以 START 条件 (S) 开始，以 STOP 条件 (P) 结束，如 I²C 总线规范中所述。无论何时传感器通电，但未执行测量或通信，它都会自动进入空闲状态以节省能量。该空闲状态不能由用户控制。

11.4. 开始测量

测量通信序列包括 START 条件，I²C 写字头（7 位 I²C 器件地址加 0 作为写位）和 16 位测量命令。传感器指示每个字节的正确接收。它在第 8 个 SCL 时钟的下降沿之后将 SDA 引脚拉低（ACK 位）以指示接收。通过确认测量命令，传感器开始测量温度。此外，测量重复性和单次/连续测量模式由配置寄存器相应控制位设定。

传输期间，每个数据值始终跟随 CRC 校验和。下表中显示了 16 位命令。重复性（低，中和高）和时钟延展（启用或禁用）可以通过修改配置寄存器实现。重复性设置影响测量持续时间，从而影响传感器的总能量消耗。



备注：第一个“SCL 空闲”块表示最小等待时间为 1ms，白色块由微控制器控制，传感器响应为灰色块。

11.9. 单次测量模式下读取数据

传感器完成测量后，主机可以通过发送开始（START）条件，然后发送 I²C 读取头来读取测量结果。在读取温度指令发出后，传感器将确认读字头的接收并发送两个字节的数 据（温度），然后是一个字节的 CRC 校验和。每个字节必须由主机确认，并具有应答（ACK）条件，以便传感器继续发送数据。如果传感器在任何数据字节后没有收到来自主机的应答（ACK），它将不会继续发送数据。

在收到温度数据的校验和后，应发送非应答（NACK）和停止条件。如果对后续数据不感兴趣，则 I²C 主设备可以在任何数据字节之后以非应答（NACK）条件中止读传输。例如不读测量结果第二字节或 CRC 字节，以节省时间。

时钟延展模式不开启

当关闭时钟延展时，发出测量命令后，如果温度测量尚未结束，传感器会响应非应答（NACK）的读字头。

时钟延展模式开启

时钟延展模式通过设定配置寄存器第 5 位为 1 开启。当开启时钟延展时，发出测量命令后，传感器通过应答 (ACK) 响应读字头，然后拉低 SCL 线，直到测量完成。一旦测量完成，传感器就会释放 SCL 线并发送测量结果。

11.10. 连续测量模式指令

传感器连续测量模式通过设定配置寄存器第 2、3、4 位开启。它们在重复性（低，中和高）和数据采集频率（每秒 0.5, 1, 2, 4 和 10 次测量）方面有所不同。在此模式下无法选择时钟延展。数据采集频率和重复性设置会影响传感器的测量持续时间和电流消耗，具体配置见 9.2 章节。

11.11. 连续测量模式下读取数据

在连续测量模式下，主机可以通过发送开始（START）条件，然后发送 I²C 读取头来读取测量结果。

11.12. 停止连续测量模式指令

可以使用下表中所示的停止命令停止连续测量模式。收到停止命令后，传感器将停止正在进行的测量但不会进入单次测量模式。重新发送一遍测量指令后会再次进入连续测量模式。如果想切换至单次测量模式，则需对配置寄存器相关控制位进行设定。

指令									十六进制编码																																																																																					
停止连续测量									0x3093																																																																																					
<table><tr><td></td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td><td>10</td><td>11</td><td>12</td><td>13</td><td>14</td><td>15</td><td>16</td><td>17</td><td>18</td></tr><tr><td>开始</td><td colspan="7">I2C 地址</td><td>写</td><td>应答</td><td colspan="8">指令高字节</td><td>应答</td><td colspan="8">指令低字节</td><td>应答</td><td>停止</td></tr><tr><td colspan="8">I2C写字头</td><td colspan="10">16位指令</td><td colspan="2"></td></tr></table>																			1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	开始	I2C 地址							写	应答	指令高字节								应答	指令低字节								应答	停止	I2C写字头								16位指令											
	1	2	3	4	5	6	7	8	9	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18																																																																			
开始	I2C 地址							写	应答	指令高字节								应答	指令低字节								应答	停止																																																																		
I2C写字头								16位指令																																																																																						

备注：白色块由微控制器控制，传感器响应为灰色块。

11.13. 复位

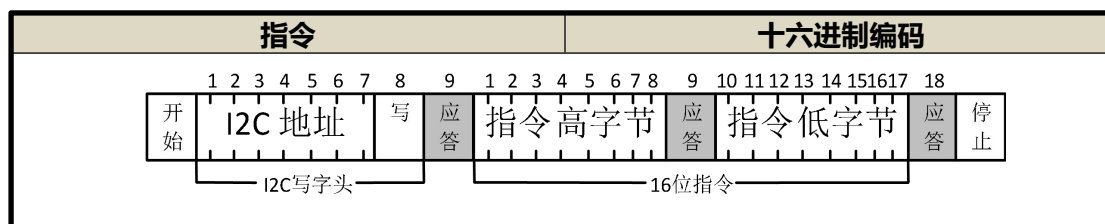
通过发出软复位命令，可以在传感器内部生成系统复位信号。此外，在上电起始阶段，芯片内部会自动生成系统复位信号。在复位过程中，传感器不会处理命令。

11.13.1. 接口复位

如果与设备的通信出现异常，以下信号序列将重置串行接口：在 SDA 保持高电平时，将 SCL 切换九次或更多次。此方法必须在下一个命令之前跟随传输启动序列。此序列仅重置接口，状态寄存器保留其内容。

11.13.2. 软复位/重新初始化

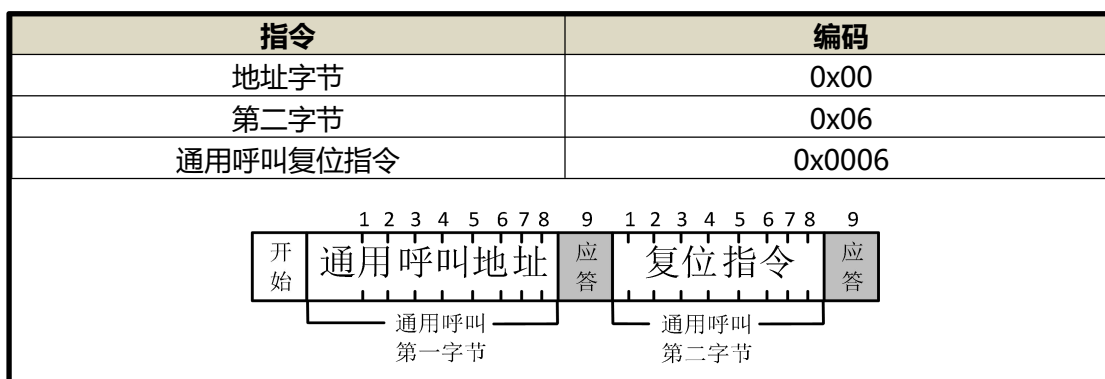
传感器提供软复位机制，可在不断电的情况下强制系统进入明确定义的状态。当系统处于空闲状态时，可以将软复位命令发送到传感器。这会触发传感器重置其系统控制器并从内存重新加载校准数据。为了启动软复位过程，应发送下表中所示的命令。



备注：白色块由微控制器控制，传感器响应为灰色块。

11.13.3. 通用呼叫复位

根据 I²C 总线规范，也可以使用“通用调用”模式生成传感器的复位。以这种方式生成的重置不是特定于设备的。支持通用呼叫模式的同一 I²C 总线上的所有设备都将执行复位。此外，此命令仅在传感器能够处理 I²C 命令时有效。适当的命令由两个字节组成，如下表所示。



备注：白色块由微控制器控制，传感器响应为灰色块。

11.13.4. 硬复位

硬复位通过关闭 VDD 引脚电源电压切然后再打开来实现。为了防止传感器通过 ESD 二极管供电，还需要移除引脚 1 (SCL) 和引脚 6 (SDA) 的外接电压。

11.14. 报警限写入和读取命令

指令	编码
读取报警高线	0xE11F
读取报警清除高线	0xE114
读取报警清除高线	0xE102
读取报警低线	0xE109

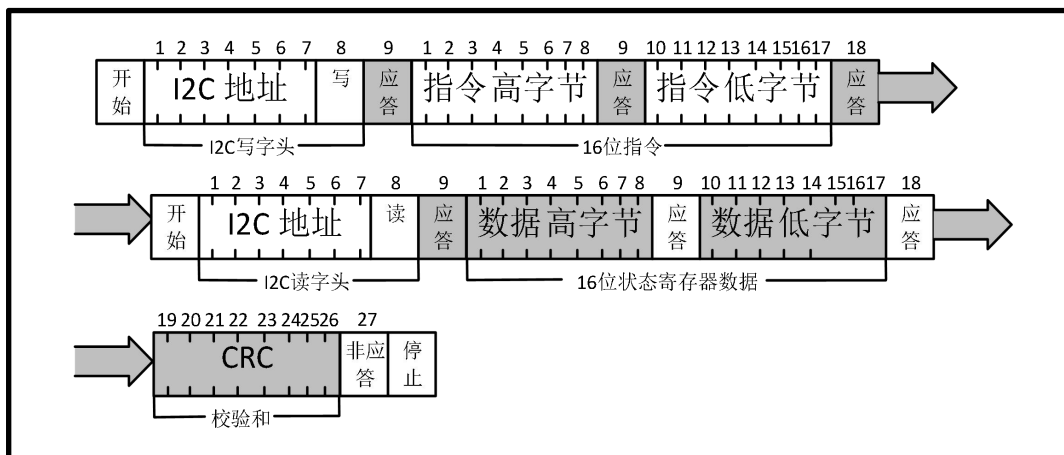


图 11.14-1 读取警报门限指令（白色块由微控制器控制，传感器响应为灰色块）

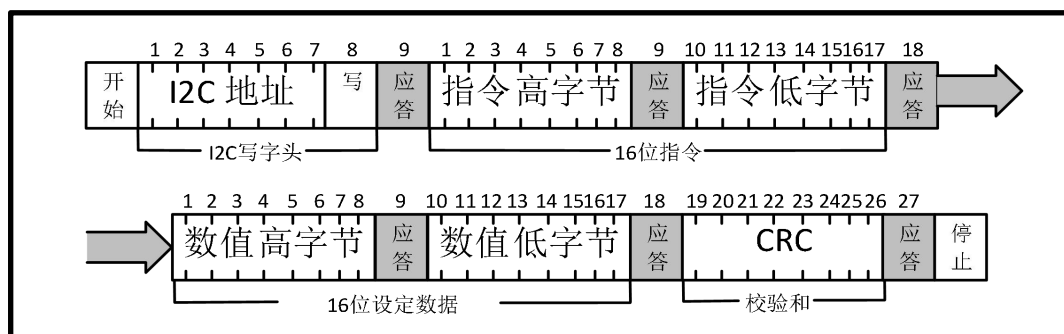


图 11.14-2 写入警报门限指令（白色块由微控制器控制，传感器响应为灰色块）

11.15. 寄存器保存和恢复指令

所有写入到寄存器的数据（工作配置和报警门限）都是暂存的。如果希望这些设置永久驻留，即失电后仍能保存，需要用复制命令将这些数据存储到 E²PROM 中。

硬复位或软复位时，系统自动加载 E²PROM 数据到寄存器中。此外，也可以用恢复指令从 E²PROM 恢复数据到寄存器中。

表 11.15 寄存器保存和恢复指令

指令	十六进制编码
保存数据到 E ² PROM	0xCC48
恢复 EE 区域	0xCCB8
恢复除 EE 之外的剩余区域	0xCCB6

11.16. I²C 时序特性

表 11.16 I²C 总线时序特性 ⁽¹⁾

参数	符号	标准模式		快速模式		单位
		最小值	最大值	最小值	最大值	
SCL 频率	f_{SCL}	0	100	0	400	kHz
SCL 低电平时间	t_{LOW}	4.7 ⁽⁷⁾	—	1.3 ⁽⁷⁾	—	μs
SCL 高电平时间	t_{HIGH}	4.0	—	0.6	—	μs
start(restart)时 SDA 拉低后 SCL 高电平的持续时间	$t_{HD;STA}$	4.0 30 ⁽⁶⁾	— —	0.6 30 ⁽⁶⁾	— —	μs
从 SCL 拉低开始到 SDA 数据发生变化的时间间隔	$t_{HD;DAT}$	5.0 0 ⁽²⁾	— 3.45 ⁽³⁾	— 0 ⁽²⁾	— 0.9 ⁽³⁾	μs μs
从 SDA 数据稳定开始到 SCL 拉高的时间间隔	$t_{SU;DAT}$	250	—	100 ⁽⁴⁾	—	ns
restart 时 SDA 拉低前 SCL 的高电平保持时间	$t_{SU;STA}$	4.7	—	0.6	—	μs
stop 时从 SCL 拉高到 SDA 拉高的时间间隔	$t_{SU;STO}$	4.0	—	0.6	—	μs
start 与 stop 的间隔时间	t_{BUF}	4.7	—	1.3	—	μs
SCL/SDA 上升沿所需时间	t_R	—	1000	20+0.1C _b ⁽⁵⁾	—	ns
SCL/SDA 下降沿所需时间	t_F	2.5	300	20+0.1C _b ⁽⁵⁾	—	ns

备注 1：所有数值都是以 VIHmin 和 VILmax 为参考。

备注 2：器件必须内部提高至少 300ns 的保持时间给 SDA。

备注 3：最大 $t_{HD;DAT}$ 只有当 SCL 低电平周期(t_{LOW})不拉伸时必须满足。

备注 4：快速器件模式 I²C 器件可以工作在标准模式，但 $t_{SU;DAT}=250ns$ 的要求必须满足。

备注 5：Cb=I²C 总线的总电容。

备注 6：对于睡眠模式开启的应用，为保证从机能够被主机成功唤醒， $t_{HD;STA}$ 的最小值与 VDD 电压成反比，当 VDD=5V 时最小值为 30us，当 VDD=3V 时最小值为 50us，当 VDD=1.8V 时最小值为 150us，具体可参见 FAQ。

备注 7：对于睡眠模式开启的应用，若主机采用硬件 I2C 通信且无法更改 $t_{HD;STA}$ ，为保证从机能够被主机成功唤醒，正式 I2C 通信中的第 1 个 t_{LOW} （从 SCL 首次拉低到 SCL 首次拉高的时间长度）的最小值与 VDD 电压成反比，当 VDD=5V 时最小值为 30us，当 VDD=3V 时最小值为 50us，当 VDD=1.8V 时最小值为 150us，具体可参见 FAQ。

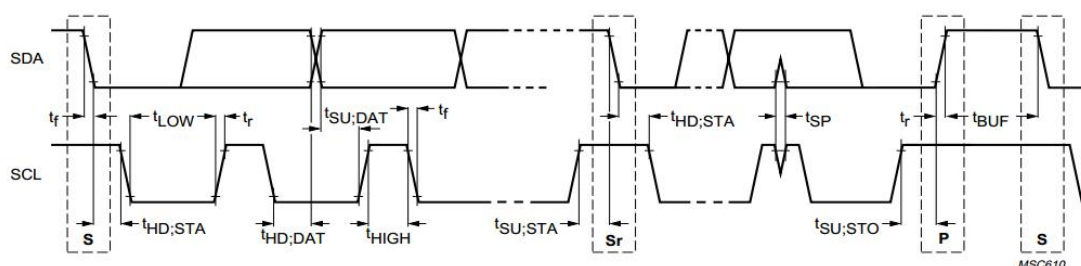
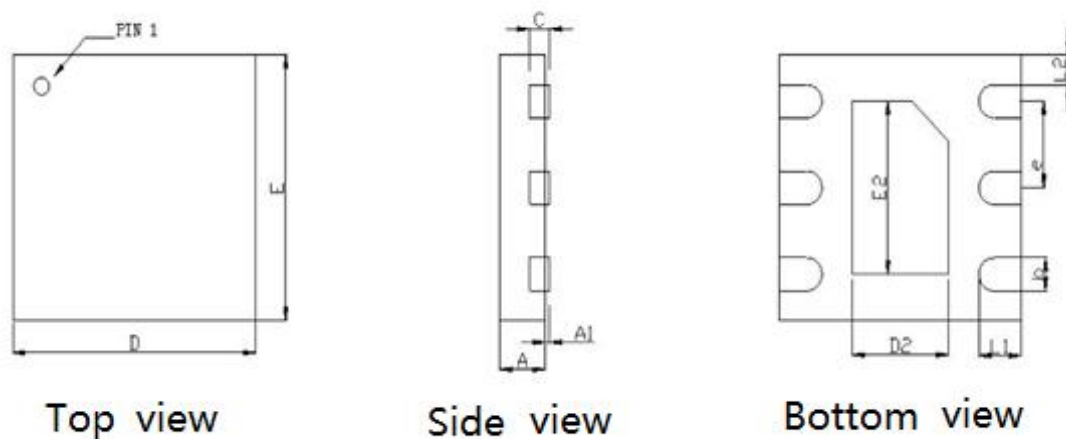


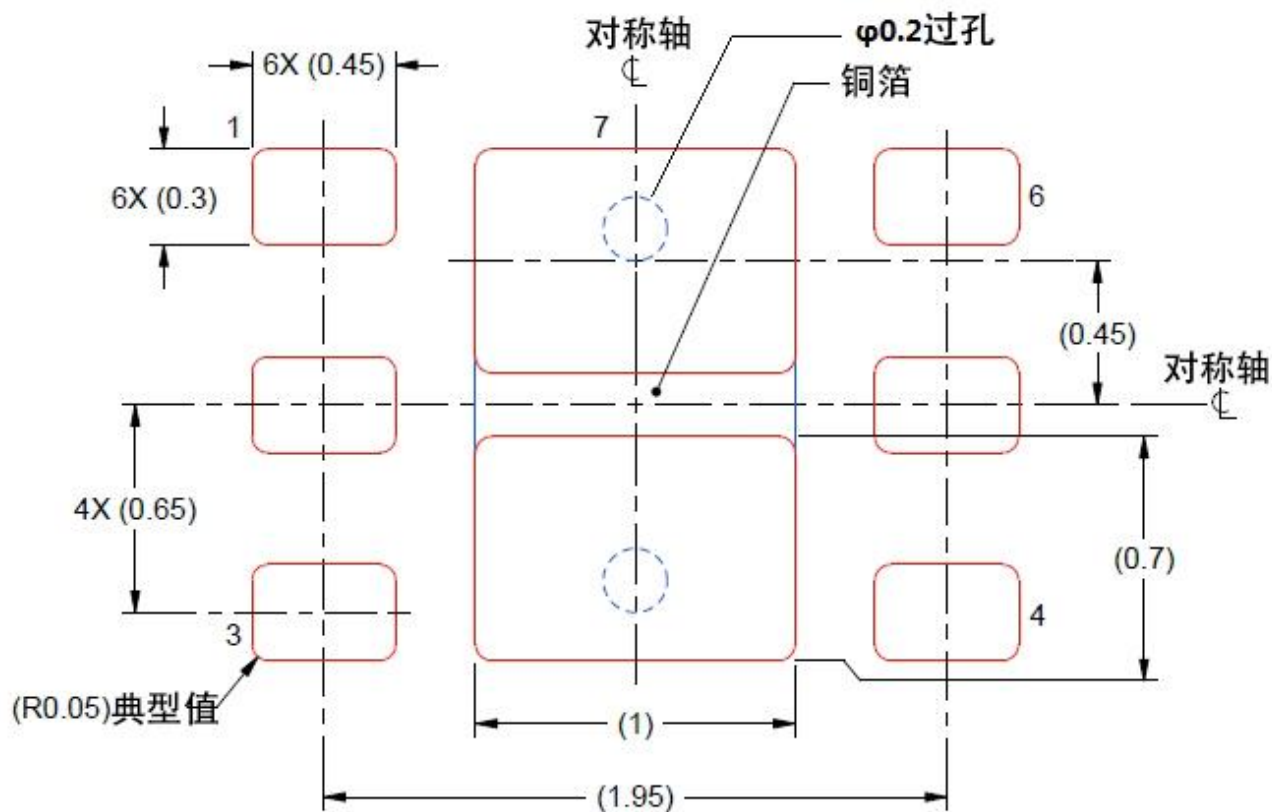
图 11.16 I²C 时序参数

12. 封装及 PCB 焊盘引脚尺寸图



Symbol	Millimeters			Inches		
	Min	Nom	Max	Min	Nom	Max
A	0.70	0.75	0.80	0.028	0.030	0.031
A1	0.00	0.02	0.02	0.000	0.001	0.002
b	0.22	0.25	0.28	0.009	0.010	0.011
C	0.20			0.008		
D	1.90	2.00	2.10	0.076	0.080	0.084
D2	0.77	0.80	0.83	0.030	0.032	0.033
E	1.90	2.00	2.10	0.076	0.080	0.084
E2	1.27	1.30	1.33	0.050	0.052	0.053
e	0.650 BSC			0.026		
L1	0.30	0.35	0.40	0.012	0.014	0.016
L2	0.100	0.150	0.200	0.004	0.006	0.008

图 12-1 封装尺寸图 DFN6L(2X2X0.75mm)



推荐焊盘尺寸
基于0.125mm厚钢网

图 12-2 PCB 焊盘引脚尺寸图

单击下面可查看定价，库存，交付和生命周期等信息

[>>Mysentech\(敏源\)](#)