

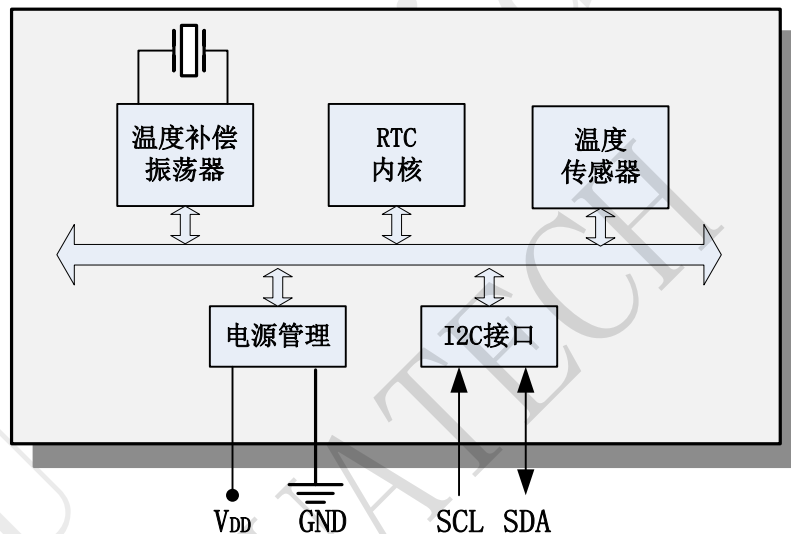


INS5710A — I²C 低功耗 RTC 实时时钟芯片

特性

- 低功耗: 0.6uA(最小值)
- 稳定度:
< 5±23ppm @ +25°C
- 内置晶体: 32.768kHz
- 内置温度传感器
- 通信接口类型: I²C 总线接口
- 电压输入范围: 1.6V ~ 5.0V
- 温度范围: -40°C ~ +85°C
- 闰年自动调整功能
- 封装尺寸: 4.9mm × 6.0mm × 1.6mm

原理框图



说明

INS5710A 是一款低功耗实时时钟芯片，内置 32.768KHz 晶振、高精度温度传感器以及温度补偿电路，自动调整时钟精度。具有 I²C 通信接口，支持日历（年，月，日，时，分，秒）功能。采用贴片 SOP8 封装，适用于三表、便携式终端及其他小型电子仪器等。

修订记录表



目录

| | | |
|-------|-----------------------------|----|
| 1 | 产品概述 | 5 |
| 2 | 原理框图 | 5 |
| 3 | 特性 | 5 |
| 4 | 管脚定义 | 6 |
| 5 | 电气特性 | 7 |
| 5.1 | 绝对参数..... | 7 |
| 5.2 | 额定工作参数..... | 7 |
| 5.3 | 频率特性..... | 7 |
| 5.4 | 直流电气特性..... | 8 |
| 5.5 | 交流特性..... | 9 |
| 6 | 用户寄存器..... | 10 |
| 6.1 | 寄存器列表..... | 10 |
| 6.2 | 寄存器详细描述..... | 12 |
| 6.2.1 | 时间..... | 12 |
| 6.2.2 | 扩展寄存器..... | 13 |
| 6.2.3 | 标志寄存器..... | 13 |
| 6.2.4 | 控制寄存器..... | 13 |
| 6.2.5 | 温度寄存器..... | 13 |
| 6.2.6 | 保留寄存器..... | 14 |
| 6.2.7 | Device ID 寄存器..... | 14 |
| 6.2.8 | 控制寄存器 1..... | 14 |
| 6.2.9 | 亚秒时间寄存器..... | 14 |
| 7 | I ² C 总线接口 | 15 |
| 7.1 | 注意事项..... | 15 |
| 7.2 | 总线地址..... | 15 |
| 7.3 | 总线协议..... | 15 |
| 7.3.1 | 写序列..... | 15 |



| | | |
|-------|------------|----|
| 7.3.2 | 读序列..... | 16 |
| 8 | 封装尺寸图..... | 18 |

DAPU Confidential
DAHUA TECH



1 产品概述

INS5710A 是一款低功耗实时时钟芯片，内置 32.768KHz 晶振、高精度温度传感器以及温度补偿电路，自动调整时钟精度。具有 I²C 通信接口，支持日历（年，月，日，时，分，秒）功能。采用贴片 SOP8 封装，适用于三表、便携式终端及其他小型电子仪器等。

2 原理框图

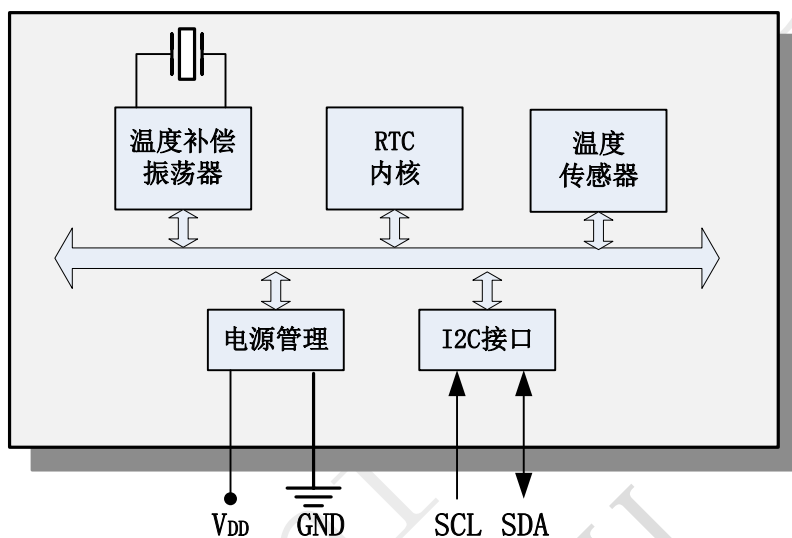


图 1 原理框图

3 特性

- 低功耗：0.6uA(最小值)
- 高稳定度：
 $< 5 \pm 23\text{ppm} @ +25^\circ\text{C}$
- 内置晶体：32.768kHz
- 内置温度传感器
- 通信接口类型：I²C 总线接口
- 电压输入范围：1.6V ~ 5.0V
- 温度范围：-40°C ~ +85°C
- 闰年自动调整功能



- 封装尺寸: 4.9mm × 6.0mm × 1.6mm

4 管脚定义

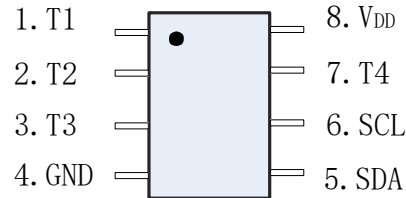


表1. 管脚定义

| 管脚号 | 管脚名称 | I/O 方向 | 说明 |
|-----|-----------------|--------|-----------------------|
| 1 | T1 | | 厂家测试, 必须悬空 |
| 2 | T2 | | 厂家测试, 必须悬空 |
| 3 | T3 | | 厂家测试, 必须悬空 |
| 4 | GND | - | 电源地 |
| 5 | SDA | In/Out | I ² C 数据信号 |
| 6 | SCL | In | I ² C 时钟信号 |
| 7 | T4 | | 厂家测试, 必须悬空 |
| 8 | V _{DD} | - | 主电源输入 |



5 电气特性

5.1 绝对参数

表2. 绝对参数

| 参数 | 记号 | 数值 | | | 单位 | 备注 |
|----------|-------------------|---------|-----|-----|----|-------------|
| | | 最小值 | 典型值 | 最大值 | | |
| 主电源 | V _{DD} | -0.3 | | 5.5 | V | |
| I/O 输入电压 | V _{IN} | GND-0.3 | | 5.5 | V | SCL, SDA 输入 |
| I/O 输出电压 | V _{OUT1} | GND-0.3 | | 5.5 | V | SDA 输出 |
| 储存温度 | T _{STG} | -55 | | 125 | °C | |

5.2 额定工作参数

表3. 额定工作参数

| 参数 | 记号 | 数值 | | | 单位 | 备注 |
|-----------|------------------|-----|-----|-----|----|----|
| | | 最小值 | 典型值 | 最大值 | | |
| 主电源(常规模式) | V _{DD} | 2.5 | 3.0 | 5.0 | V | |
| 主电源(时间保持) | V _{DD} | 1.6 | 3.0 | 5.0 | V | * |
| 工作温度 | T _{OPR} | -40 | 25 | 85 | °C | |

* 注：在上电起振时间内，必须提供 2.5V 以上电压确保振荡电路稳定起振。

5.3 频率特性

表4. 频率特性

| 参数 | 符号 | 数值 | | | 单位 | 备注 |
|---------|------------------|------|-----|-----|-----|--|
| | | 最小值 | 典型值 | 最大值 | | |
| 频率稳定度 | $\Delta f_1/f$ | 5±23 | | | ppm | V _{DD} =3.0V; @25°C |
| 温度稳定度 | $\Delta f_2/f$ | -120 | | +10 | ppm | -20°C~+70°C; V _{DD} =3.0V; 参考+25°C 频率 |
| 起振时间 | t _{STA} | | | 1 | s | @25°C |
| 年老化 | f _a | | | ±3 | ppm | |
| 温度传感器精度 | T _{emp} | | | ±5 | °C | V _{DD} =3.0V |



5.4 直流电气特性

表5. 直流特性

| 参数 | 符号 | 数值 | | | 单位 | 备注 | |
|--------|-----------|--------------|-----|--------------|----|-------------------------------------|---------------------------------------|
| | | 最小值 | 典型值 | 最大值 | | | |
| 平均电流 1 | I_{DD1} | 0.91 | | 5.1 | uA | $V_{DD}=5.0V$ | $f_{SCL}=0Hz, /INT = V_{DD};$ 补偿间隔 2s |
| 平均电流 2 | I_{DD2} | 0.6 | | 4.9 | | $V_{DD}=3.0V$ | |
| 输入高电平 | V_{IH} | $0.8*V_{DD}$ | | 5.0 | V | SCL, SDA 脚 | |
| 输入低电平 | V_{IL} | GND-0.3 | | $0.2*V_{DD}$ | V | | |
| 输出低电平 | V_{OL1} | GND | | GND+0.4 | V | $V_{DD} \geq 3.0V, I_{OL} = 3mA$ | SDA 脚 |
| 输入漏电流 | I_{LK} | -0.1 | | 0.1 | uA | SDA, SCL 脚, $V_{IN} = V_{DD}$ 或 GND | |
| 输出漏电流 | I_{OZ} | -0.1 | | 0.1 | uA | SDA 脚, $V_{IN} = V_{DD}$ 或 GND | |



5.5 交流特性

表6. 交流特性

V_{DD} =2.5V~5.0V; Ta=-40℃~+85℃

| 参数 | 符号 | 数值 | | | 单位 |
|-----------------|----------------------|-----|-----|-----|-----|
| | | 最小值 | 典型值 | 最大值 | |
| SCL 时钟频率 | f _{SCL} | | | 400 | kHz |
| SCL 低电平时间 | t _{LOW} | 1.3 | | | us |
| SCL 高电平时间 | t _{HIGH} | 0.6 | | | us |
| 开始条件保持时间 | t _{HD: STA} | 0.6 | | | us |
| 开始条件建立时间 | t _{SU: STA} | 0.6 | | | us |
| 停止条件建立时间 | t _{SU: STO} | 0.6 | | | us |
| 从停止到开始的恢复时长 | t _{RCV} | 1.3 | | | us |
| 数据建立时间 | t _{SU: DAT} | 100 | | | ns |
| 数据保持时间 | t _{HD: DAT} | 0 | | | ns |
| SCL, SDA 输入上升时间 | t _r | | | 0.4 | us |
| SCL, SDA 输入下降时间 | t _f | | | 0.4 | us |

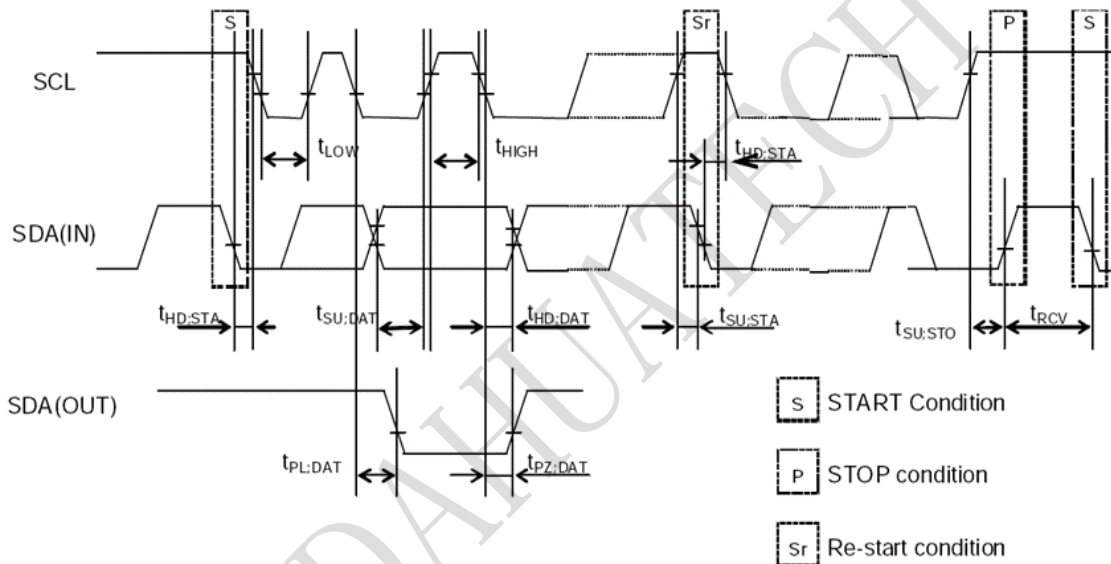


图 2 I²C 时序图



6 用户寄存器

6.1 寄存器列表

地址 0x00~0x0F: 基本时间和日历寄存器。

地址 0x10~0x1F: 扩展寄存器组 1。

地址 0x20~0x30: 扩展寄存器组 2。

* 注: 0x10~0x16 与 0x00~0x06 完全相同, 0x1B~0x1F 与 0x0B~0x0F 完全相同。

表7. 基本时间和日历寄存器列表

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 读/写 |
|------|--------------------|-----------------|-----------------|-----------------|-----------------|-----------------|------|------|-------|-----|
| 0x00 | SEC | ○ | BCD 码, 秒十位, 0-5 | | | BCD 码, 秒个位, 0-9 | | | | R/W |
| 0x01 | MIN | ○ | BCD 码, 分十位, 0-5 | | | BCD 码, 分个位, 0-9 | | | | R/W |
| 0x02 | HOUR | ○ | ○ | BCD 码, 时十位, 0-2 | | BCD 码, 时个位, 0-9 | | | | R/W |
| 0x03 | WEEK | ○ | 6 | 5 | 4 | 3 | 2 | 1 | 0 | R/W |
| 0x04 | DAY | ○ | ○ | BCD 码, 日十位, 0-3 | | BCD 码, 日个位, 0-9 | | | | R/W |
| 0x05 | MONTH | ○ | ○ | ○ | BCD 码, 月十位, 0-1 | BCD 码, 月个位, 0-9 | | | | R/W |
| 0x06 | YEAR | BCD 码, 年十位, 0-9 | | | BCD 码, 年个位, 0-9 | | | | R/W | |
| 0x07 | RAM | ● | ● | ● | ● | ● | ● | ● | ● | R/W |
| 0x08 | RSV | 保留位 | | | | | | | | R |
| 0x09 | RSV | 保留位 | | | | | | | | R |
| 0x0A | RSV | 保留位 | | | | | | | | R |
| 0x0B | RSV | 保留位 | | | | | | | | R |
| 0x0C | RSV | 保留位 | | | | | | | | R |
| 0x0D | Extension Register | TEST | 保留位 | 保留位 | 保留位 | ● | ● | 保留位 | 保留位 | R/W |
| 0x0E | Flag Register | ○ | ○ | 保留位 | 保留位 | 保留位 | ○ | VLF | 保留位 | R/W |
| 0x0F | Control Register | CSEL [1] | CSEL [0] | 保留位 | 保留位 | 保留位 | ○ | ○ | RESET | R/W |

表8. 扩展寄存器组列表 1

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 读/写 |
|------|-------|-----------------|-----------------|-----------------|-----------------|-----------------|------|------|------|-----|
| 0x10 | SEC | ○ | BCD 码, 秒十位, 0-5 | | | BCD 码, 秒个位, 0-9 | | | | R/W |
| 0x11 | MIN | ○ | BCD 码, 分十位, 0-5 | | | BCD 码, 分个位, 0-9 | | | | R/W |
| 0x12 | HOUR | ○ | ○ | BCD 码, 时十位, 0-2 | | BCD 码, 时个位, 0-9 | | | | R/W |
| 0x13 | WEEK | ○ | 6 | 5 | 4 | 3 | 2 | 1 | 0 | R/W |
| 0x14 | DAY | ○ | ○ | BCD 码, 日十位, 0-3 | | BCD 码, 日个位, 0-9 | | | | R/W |
| 0x15 | MONTH | ○ | ○ | ○ | BCD 码, 月十位, 0-1 | BCD 码, 月个位, 0-9 | | | | R/W |
| 0x16 | YEAR | BCD 码, 年十位, 0-9 | | | BCD 码, 年个位, 0-9 | | | | R/W | |



| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 读/写 |
|------|--------------------|----------|----------|------|------|------|------|------|-------|-----|
| 0x17 | TEMP | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 | R |
| 0x18 | RSV | ○ | ○ | ○ | ○ | 保留位 | ● | ● | ● | R/W |
| 0x19 | Not use | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | R |
| 0x1A | Not use | ○ | ○ | ○ | ○ | ○ | ○ | ○ | ○ | R |
| 0x1B | RSV | 保留位 | | | | | | | | R |
| 0x1C | RSV | 保留位 | | | | | | | | R |
| 0x1D | Extension Register | TEST | 保留位 | 保留位 | 保留位 | ● | ● | 保留位 | 保留位 | R/W |
| 0x1E | Flag Register | ○ | ○ | 保留位 | 保留位 | 保留位 | ○ | VLF | 保留位 | R/W |
| 0x1F | Control Register | CSEL [1] | CSEL [0] | UIE | TIE | AIE | ○ | ○ | RESET | R/W |

表9. 扩展寄存器组列表 2

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 读/写 |
|---------|--------------------|-----------------|------|------|------|-------------|------|------|------|-----|
| 0x20 | Device ID | VendorID[3:0] | | | | Ver[3:0] | | | | R |
| 0x21 | Control Register 1 | 保留位: 确保固定为 0x8 | | | | ○ | ○ | ○ | ● | R/W |
| 0x22-26 | RSV | 保留位: 确保固定为 0x00 | | | | | | | | R |
| 0x27 | EvSubSEC&Sub SEC | 保留位 | | | | SubSEC[3:0] | | | | R |
| 0x28-30 | RSV | 保留位: 确保固定为 0x00 | | | | | | | | R/W |

注:

1. 在上电初始化 (从 0V) 或 VLF 位为 1 之后, 确保初始化所有的寄存器之后再使用 RTC。
2. 上电初始化期间, 寄存器的默认值如下:
 初始值为 0: TEST、CSEL [1]、RESET。
 初始值为 1: VLF、CSEL [0]。
 其他寄存器值为不确定值, 所以确保在使用前进行复位。
3. 标记为“○”的位, 初始化后读为 0。
4. 标记为“●”的位为 RAM, 可以用来读写任意数据。
5. 这些位只能写 0: UF、TF、AF、VLF。
6. TEST 位被厂家用于测试, 该位在写操作的时候请一定确保为“0”。保留位被厂家用于测试, 写操作的时候请一定确保按照要求固定输入。



6.2 寄存器详细描述

6.2.1 时间

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|---------|------|------|-----------------|-----------------|------|-----------------|------|------|------|------|
| 0x00/10 | SEC | ○ | BCD 码, 秒十位, 0-5 | | | BCD 码, 秒个位, 0-9 | | | | 0x25 |
| 0x01/11 | MIN | ○ | BCD 码, 分十位, 0-5 | | | BCD 码, 分个位, 0-9 | | | | 0x36 |
| 0x02/12 | HOUR | ○ | ○ | BCD 码, 时十位, 0-2 | | BCD 码, 时个位, 0-9 | | | | 0x01 |

SEC: 秒, BCD 码格式, 数值 0~59 循环递增。

MIN: 分钟, BCD 码格式, 数值 0~59 循环递增。

HOUR: 小时, BCD 码格式, 数值 0~23 循环递增。

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|---------|------|------|------|------|------|------|------|------|------|------|
| 0x03/13 | WEEK | ○ | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 0x40 |

WEEK: 周, 按 bit 指示, 对照表如下, 数值按 01h、02h、04h、08h、10h、20h、40h 循环:

表10. WEEK 寄存器值对照表

| 星期 | Data | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|----|------|------|------|------|------|------|------|------|------|
| 日 | 01h | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 一 | 02h | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 二 | 04h | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 三 | 08h | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 四 | 10h | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 五 | 20h | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 六 | 40h | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

同时只能有 1bit 置 1

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|---------|-----|------|------|-----------------|------|-----------------|------|------|------|------|
| 0x04/14 | DAY | ○ | ○ | BCD 码, 日十位, 0-3 | | BCD 码, 日个位, 0-9 | | | | 0x01 |

DAY: 日, BCD 码格式, 支持大小月、闰年 (2000~2099 年), 数值循环递增, 数值范围见下表:

表11. DAY 寄存器数值范围

| 月份 | 数值范围 |
|-----------------------|---------|
| 1, 3, 5, 7, 8, 10, 12 | 1~31 递增 |
| 4, 6, 9, 11 | 1~30 递增 |
| 2月 (平年) | 1~28 递增 |
| 2月 (闰年) | 1~29 递增 |

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|---------|-------|-----------------|------|------|-----------------|-----------------|------|------|------|------|
| 0x05/15 | MONTH | ○ | ○ | ○ | BCD 码, 月十位, 0-1 | BCD 码, 月个位, 0-9 | | | | 0x01 |
| 0x06/16 | YEAR | BCD 码, 年十位, 0-9 | | | BCD 码, 年个位, 0-9 | | | | | 0x00 |

MONTH: 月, BCD 码格式, 数值 1~12 循环递增。



YEAR: 年, BCD 码格式, 数值 0~99 循环递增。对应 2000~2099 年。

例如: 2020/01/01 Wednesday 21:18:36

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|---------|-------|------|------|------|------|------|------|------|------|
| 0x00/10 | SEC | ○ | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 0x01/11 | MIN | ○ | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| 0x02/12 | HOUR | ○ | ○ | 1 | 0 | 0 | 0 | 0 | 1 |
| 0x03/13 | WEEK | ○ | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 0x04/14 | DAY | ○ | ○ | 0 | 0 | 0 | 0 | 0 | 1 |
| 0x05/15 | MONTH | ○ | ○ | ○ | 0 | 0 | 0 | 0 | 1 |
| 0x06/16 | YEAR | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

6.2.2 扩展寄存器

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|---------|--------------------|------|------|------|------|------|------|------|------|------|
| 0x0D/1D | Extension Register | TEST | 保留位 | 保留位 | 保留位 | ● | ● | 保留位 | 保留位 | 0x02 |

TEST: 厂家测试位, 必须为“0”, 禁止用户修改。

6.2.3 标志寄存器

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|---------|---------------|------|------|------|------|------|------|------|------|------|
| 0x0E/1E | Flag Register | ○ | ○ | 保留位 | 保留位 | 保留位 | ○ | VLF | ● | 0x23 |

VLF (Voltage Low Flag): 电压低标志, 当电压低于 1.6V 时置“1”, 并一直保持为“1”直到软件写“0”。

6.2.4 控制寄存器

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|---------|------------------|----------|----------|------|------|------|------|------|-------|------|
| 0x0F/1F | Control Register | CSEL [1] | CSEL [0] | 保留位 | 保留位 | 保留位 | ○ | ○ | RESET | 0x40 |

CSEL[1], CSEL[0]: 设置温度补偿间隔, 如下:

| CSEL[1] | CSEL[0] | 温度补偿间隔 |
|---------|---------|---------|
| 0 | 0 | 0.5s |
| 0 | 1 | 2s (默认) |
| 1 | 0 | 10s |
| 1 | 1 | 30s |

RESET: 准备同步时间和定时器的起点。

6.2.5 温度寄存器

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|------|------|------|------|------|------|------|------|------|------|------|
| 0x17 | TEMP | 128 | 64 | 32 | 16 | 8 | 4 | 2 | 1 | 0xa9 |

可以读取数字化温度数据, 按如下公式计算:

$$\text{温度}[\text{°C}] = (\text{TEMP}[7:0] * 2^{-187.19}) / 3.218$$



6.2.6 保留寄存器

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|------|----------|------|------|------|------|----------|------|------|------|------|
| 0x18 | Reserved | ○ | ○ | ○ | ○ | Reserved | ● | ● | ● | 0x00 |

该寄存器为保留寄存器，bit3 建议设置为“0”。

6.2.7 Device ID 寄存器

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|------|-----------|---------------|------|------|------|----------|------|------|------|------|
| 0x20 | Device ID | VendorID[3:0] | | | | Ver[3:0] | | | | 0xd2 |

VendorID[3:0]: 厂家编码, VendorID[3:0]=1101b=Dh, 代表大普通信。

Ver[3:0]: 芯片版本号, 从 1 开始。

6.2.8 控制寄存器 1

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|------|--------------------|--------------|------|------|------|------|------|------|------|------|
| 0x21 | Control Register 1 | 保留位: 必须为 0x8 | | | | ○ | ○ | ○ | ● | 0x80 |

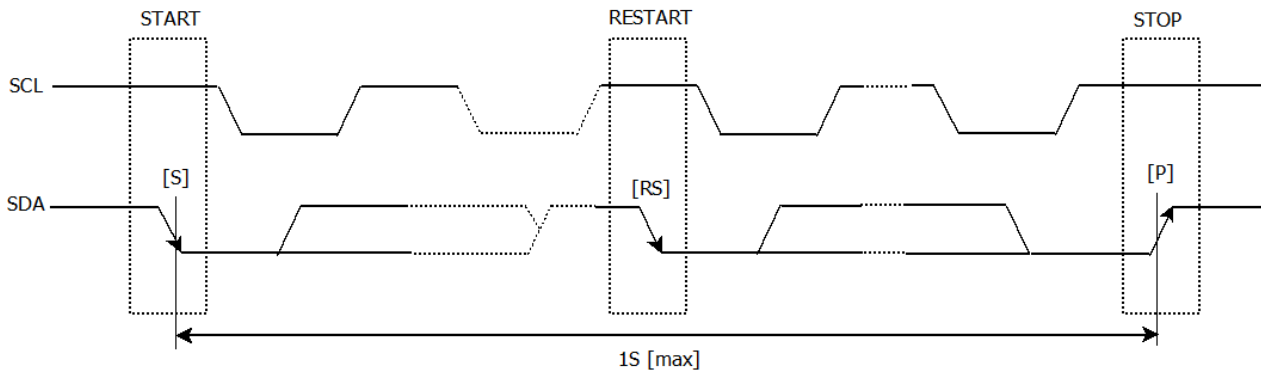
6.2.9 亚秒时间寄存器

| 地址 | 功能 | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | 默认值 |
|------|--------|------|------|------|------|-------------|------|------|------|------|
| 0x27 | SubSEC | 保留位 | | | | SubSEC[3:0] | | | | 0x00 |

SubSEC[3:0]: 时间亚秒位, 单位为 1/16s。



7 I²C 总线接口



I²C 总线接口通过 SCL、SDA 两根线作双向通信。SCL 是时钟线，SDA 是数据线。I²C 设备分为主设备和从设备，INS5710A 只能作为从设备。

7.1 注意事项

I²C 总线包含 START 命令、STOP 命令，为防止 I²C 总线挂死，从 START 命令到 STOP 命令必须在 1 秒内完成。如果超过 1 秒，INS5710A 会重置 I²C 接口。

INS5710A I²C 总线接口既支持单字节读写寄存器，也支持多字节递增访问。访问地址 0x7F 后，下一个增量地址是 0x00。

7.2 总线地址

表12. I²C 总线 Slave 地址

| Transfer data | Slave address | | | | | | | R/W |
|---------------|---------------|------|------|------|------|------|------|-----------|
| | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
| 65h (Read) | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 (Read) |
| 64h (Write) | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 (Write) |

INS5710A I²C 总线 Slave 地址是 [0110 010*]。

7.3 总线协议

本节描述基于 CPU 为 I²C 主设备，INS5710A 为 I²C 从设备。

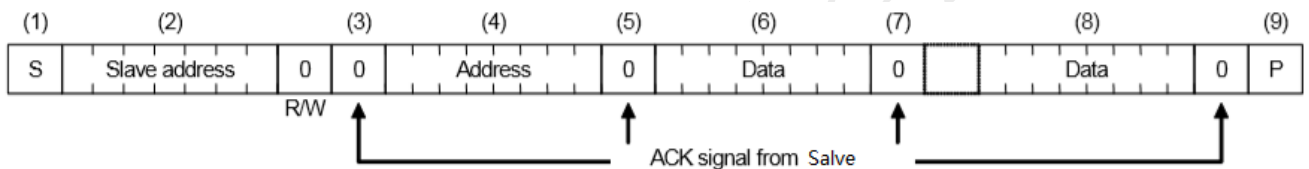
7.3.1 写序列

I²C 总线接口在写地址确定后，后续访问包含地址自增功能，即 I²C 总线接口在写一个字节数据



后，自动将后面写数据的地址自增。

- (1) CPU 发送开始[S]
- (2) CPU 发送 I²C 总线接口从地址，在 R/W 位设置为写模式
- (3) CPU 接收 I²C 总线接口的 ACK
- (4) CPU 发送写地址给 I²C 总线接口
- (5) CPU 接收 I²C 总线接口的 ACK
- (6) CPU 发送写数据给 I²C 总线接口
- (7) CPU 接收 I²C 总线接口的 ACK
- (8) 如果写多字节，重复步骤（6）和（7），地址自增
- (9) CPU 发送停止[P]



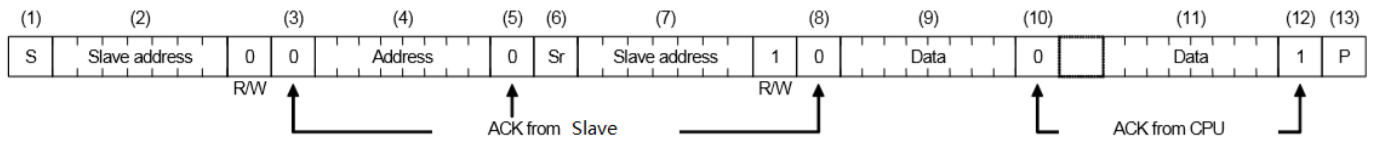
7.3.2 读序列

先用写模式写要读的寄存器地址，然后设置成读模式读取寄存器数据。

- (1) CPU 发送开始[S]
- (2) CPU 发送 I²C 总线接口从地址，在 R/W 位设置为写模式
- (3) CPU 接收 I²C 总线接口的 ACK
- (4) CPU 发送读地址给 I²C 总线接口
- (5) CPU 接收 I²C 总线接口的 ACK
- (6) CPU 发送重新开始[Sr]
- (7) CPU 发送 I²C 总线接口从地址，在 R/W 位设置为读模式
- (8) CPU 接收 I²C 总线接口的 ACK
- (9) CPU 接收 I²C 总线接口读到的数据
- (10) CPU 接收 I²C 总线接口的 ACK
- (11) 如果读多字节，重复步骤（9）和（10），地址自增
- (12) CPU 发送 ACK



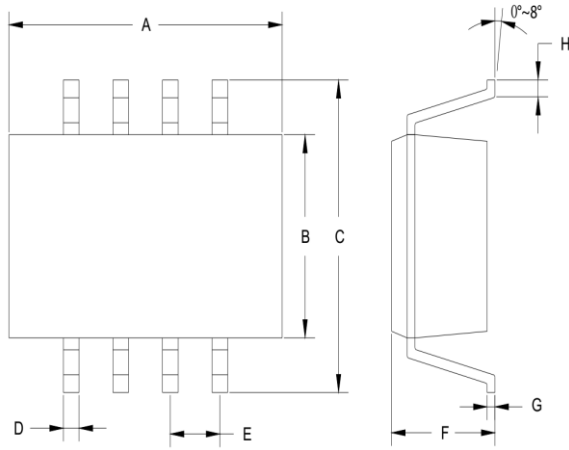
(13) CPU 发送停止[P]



DAPU Confidential
DAHUA TECH



8 封装尺寸图



| 尺寸 | 最小值 | 典型值 | 最大值 |
|----------|-------|------|-------|
| A | 4.8 | 4.9 | 5.0 |
| B | 3.8 | 3.9 | 4.0 |
| C | 5.8 | 6.0 | 6.2 |
| D | 0.356 | -- | 0.456 |
| E | -- | 1.27 | -- |
| F | 1.3 | -- | 1.6 |
| G | 0.203 | -- | 0.233 |
| H | 0.4 | 0.6 | 0.8 |

单位: mm

图 3 封装图