

LM75 温度测试

1 简介

本教程介绍使用温度传感器 LM75 来进行温度测试，例程中主要介绍温度传感器的 FPGA 驱动和数码管显示。

2 实验原理

2.1 LM75 原理介绍

LM75A 是一个高速 I2C 接口的温度传感器，可以在 $-55^{\circ}\text{C} \sim +125^{\circ}\text{C}$ 的温度范围内将温度直接转换为数字信号，并可实现 0.125°C 的精度。控制器可以通过 I2C 总线直接读取其内部寄存器中的数据，并可通过 I2C 对 4 个数据寄存器进行操作，以设置成不同的工作模式。LM75A 有 3 个可选的逻辑地址管脚，使得同一总线上可同时连接 8 个器件而不发生地址冲突。LM75A 可配置成不同的工作模式。它可设置成在正常工作模式下周期性地对环境温度进行监控，或进入关断模式来将器件功耗降至最低。OS 输出有 2 种可选的工作模式：OS 比较器模式和 OS 中断模式，OS 输出可选择高电平或低电平有效。正常工作模式下，当器件上电时，OS 工作在比较器模式，温度阈值为 80°C ，滞后阈值为 75°C 。低功耗设计，工作电流典型值为 $250\mu\text{A}$ ，掉电模式为 $3.5\mu\text{A}$ ；宽工作电压范围： $2.8\text{V} \sim 5.5\text{V}$ 。LM75 管脚说明如下图：



a) 温度寄存器 Temp（地址 0x00）

温度寄存器是一个只读寄存器，包含 2 个 8 位的数据字节，由一个高数据字节（MS）和一个低数据字节（LS）组成。这两个字节中只有 11 位用来存放分辨率为 0.125°C 的 Temp 数据（以二进制补码数据的形式），如下表所示。对于 8 位的 I2C 总线来说，只要从 LM75A 的“00 地址”连续读两个字节即可（温度的高 8 位在前）。

温度寄存器

Temp MS 字节								Temp LS 字节							
MSB							LSB	MSB							LSB
B7	B6	B5	B4	B3	B2	B1	B0	B7	B6	B5	B4	B3	B2	B1	B0
Temp 数据 (11 位)											未使用				
MSB										LSB					
D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	X	X	X	X	X

根据11位的Temp数据来计算Temp值的方法:

1. 若D10=0, 温度值(℃) = +Temp数据) × 0.125℃;
2. 若D10=1, 温度值(℃) = -Temp数据的二进制补码) × 0.125℃。

下表给出了一些 Temp 数据和温度值的例子。

Temp 表

Temp 数据			温度值
11 位二进制数 (补码)	3 位十六进制值	十进制值	℃
0111 1111 000	3F8h	1016	+127.000℃
0111 1110 111	3F7h	1015	+126.875℃
0111 1110 001	3F1h	1009	+126.125℃
0111 1101 000	3E8h	1000	+125.000℃
0001 1001 000	0C8h	200	+25.000℃
0000 0000 001	001h	1	+0.125℃
0000 0000 000	00h	0	0.000℃
1111 1111 111	7FFh	-1	-0.125℃
1110 0111 000	738h	-200	-25.000℃
1100 1001 001	649h	-439	-54.875℃
1100 1001 000	648h	-440	-55.000℃

b) 配置寄存器 (地址 0x01)

配置寄存器为 8 位可读写寄存器, 其位功能分配如下表所示。

配置寄存器位功能

B7	B6	B5	B4	B3	B2	B1	B0
保留			OS 故障队列		OS 极性	OS 比较/中断	关断

B7-B5: 保留，默认为 0。

B4-B3: 用来编程 OS 故障队列。

00 到 11 代表的值为 1、2、4、6，默认值为 0。

B2: 用来选择 OS 极性。

B2=0, OS 低电平有效（默认）；

B2=1, OS 高电平有效。

B1: 选择 OS 工作模式。

B1=0, 配置成比较器模式，直接控制外围电路；

B1=1, OS 控制输出功能配置成中断模式，以通知 MCU 进行相应处理。

B0: 选择器件工作模式。

B0=0, LM75A 处于正常工作模式（默认）；

B0=1, LM75A 进入关断模式。

c) 滞后寄存器 **Thyst** (0x02)

滞后寄存器是读/写寄存器，也称为设定点寄存器，提供了温度控制范围的下限温度。每次转换结束后，Temp 数据（取其高 9 位）将会与存放在该寄存器中的数据相比较，当环境温度低于此温度的时候，LM75A 将根据当前模式（比较、中断）控制 OS 引脚作出相应反应。该寄存器都包含 2 个 8 位的数据字节，但 2 个字节中，只有 9 位用来存储设定点数据（分辨率为 0.5℃ 的二进制补码），其数据格式如下表所示，默认为 75℃。

低/高报警温度寄存器数据格式

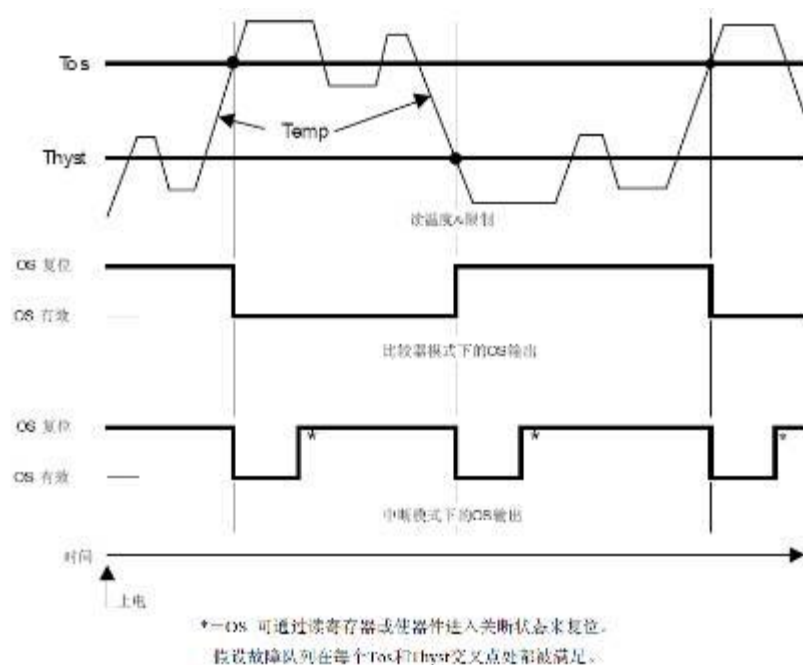
D15	D14---D8							D7	D6---D0
T8	T7	T6	T5	T4	T3	T2	T1	T0	未定义

d) 超过温关断阈值寄存器 **Tos** (0x03)

超温关断寄存器提供了温度控制范围的上限温度。每次转换结束后，Temp 数据（取其高 9 位）将会与存放在该寄存器中的数据相比较，当环境温度高于此温度的时候，LM75A 将根据当前模式（比较、中断）控制 OS 引脚作出相应反应。其数据格式如表 4 所示，默认为 80℃。

e) OS 输出

OS 输出为开漏输出口。为了观察到这个输出的状态，需要接一个外部上拉电阻，其阻值应当足够大（高达 200kΩ），以减少温度读取误差。OS 输出可通过编程配置寄存器的 B2 位设置为高或低有效。如下图所示，为 LM75A 在不同模式下 OS 引脚对温度作出的响应。OS 设为低有效。



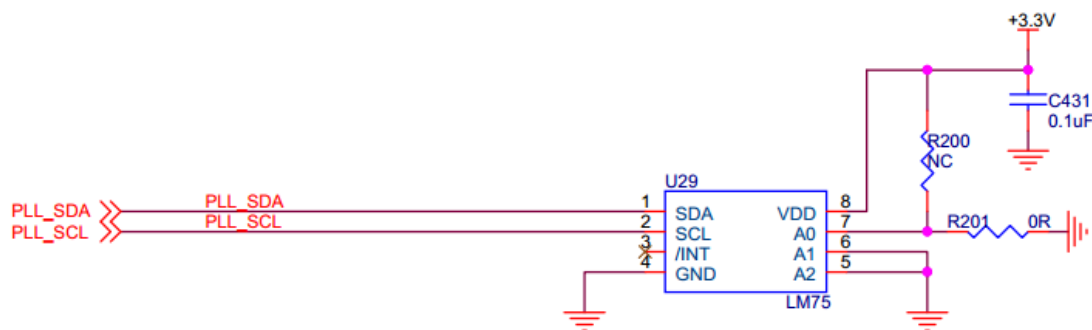
可以看出，当 LM75A 工作在比较器模式时，当温度高于 T_{OS} 时，OS 输出低电平。此时采取了降温措施，启动降温设备（如风扇），直到温度再降到 T_{Hyst} ，则停止降温，因此在这种模式下，LM75A 可以直接控制外部电路来保持环境温度；而在中断模式，则在温度高于 T_{OS} 或低于 T_{Hyst} 时产生中断。注意：在中断模式下，只有当 MCU 对 LM75A 进行读操作后，其中断信号才会消失（图中 OS 变为高电平）。

f) I2C 串行接口

在主控器的控制下，LM75A 可以通过 SCL 和 SDA 作为从器件连接到 I2C 总线上。主控器必须提供 SCL 时钟信号，可以通过 SDA 读出器件数据或将数据写入到器件中。LM75A 从地址（7 位地址）的低 3 位可由地址引脚 A2、A1 和 A0 的逻辑电平来决定。地址的高 4 位预先设置为‘1001’。下表给出了器件的完整地址，从表中可以看出，同一总线上可连接 8 个器件而不会产生地址冲突。由于输入管脚 SCL、SDA、A2-A0 内部无偏置，因此在任何应用中它们都不能悬空。

2.2 硬件原理图

如下为 AXKU040 开发板的温度传感器 LM75 部分原理图：



LM75 部分原理图

3 程序设计

程序设计比较简单，功能是 FPGA 驱动 LM75 温度传感器不断地读取温度值并送到数码管进行显示。

代码说明：

temp_test.v 是顶层模块，包含了 i2c_read_lm75，hextobcd 和 smg_interface 模块；

i2c_read_lm75.v 是 LM75 的温度读取模块，实时读取温度值；

hextobcd.v 为十六进制转 BCD 模块；

uart_send.v 为温度数据串口发送模块，每隔 1 秒在串口助手上显示温度值。

引脚名称	功能描述
sys_clk	FPGA 输入时钟 50MHz
rst_n	复位信号，低电平有效
scl	I2C 总线时钟线
sda	I2C 总线数据线
uart_rx	串口数据接收
uart_tx	串口数据发送
fan_pwm	风扇控制

4 实验现象

编译综合完成后下载程序到 AXKU040 开发板中，连接好 USB 线，可以看到串口助手不断地显示温度传感器的当前值，如下图：（串口助手的安装与使用在前面串口收发实验中已讲述）



打开串口，显示当前的温度值。

