
本章教程使用CH32V103的内部温度传感器检测器件周围温度，并通过串口调试助手打印显示。

1、温度传感器简介及相关函数介绍

CH32V103内置温度传感器，连接ADC_INT16通道，通过ADC将传感器输出的电压转换成数字值来反馈器件周围温度，推荐设置采样时间是17.1us。温度传感器输出的电压随温度线性变化，由于生产差异，其线性变化的曲线斜率和偏移有所不同，所以内部温度传感器更适合于检测温度的变化，而不是测量绝对的温度。如果需要测量精确的温度，应该使用一个外置的温度传感器。

通过设置ADC_CTLR2寄存器的TSVREFE位置1，唤醒ADC内部采样通道，软件启动或者外部触发启动ADC的温度传感器通道转换，读取数据结果（mV）。其中，数字值和温度(°C)换算公式如下：

$$\text{温度}(\text{°C}) = ((V_{\text{SENSE}} - V_{25}) / \text{Avg_Slope}) + 25$$

V_{SENSE} ：温度传感器的当前输出电压；

V_{25} ：温度传感器在25°C下的电压值；

Avg_Slope ：温度与 V_{SENSE} 曲线的平均斜率（mV/°C）

参考数据手册电气特性章节中 V_{25} 和 Avg_Slope 的实际值。

注：内部温度传感器上电（TSVREFE位从0改为1）需要一个建立时间，而ADC模块上电也需要一个建立时间（ADON位从0改为1），所以为了缩短等待时间，可以同时设置ADON和TSVREFE位。

关于CH32V103内置温度传感器具体信息，可参考CH32V103应用手册。温度检测程序所用库函数在ADC教程中已介绍，在此不再赘述。

2、硬件设计

内置温度传感器属于其内部资源，无需进行硬件连接，只需进行软件设计即可。

3、软件设计

本章教程通过ADC_INT16通道读取内置温度传感器值，其程序相较于于

ADC，变化不大，增加了使能内部温度传感器和温度值计算，主要程序如下：

temdet.h文件

```
1. #ifndef __TEMDET_H
2. #define __TEMDET_H
3.
4. #include "ch32v10x_conf.h"
5.
6. void adc_Init(void);
7. u16 get_adc(u8 ch);
8. u16 get_adc_average(u8 ch,u8 times);
9. float get_temperature(void);
10.
11. #endif
12.
```

复制代码

temdet.h文件主要是相关函数的声明。

temdet.c文件

```
1. #include "temdet.h"
2.
3. void adc_Init(void)
4. {
5.     ADC_InitTypeDef ADC_InitStructure;
6.     RCC_APB2PeriphClockCmd(RCC_APB2Periph_GPIOA
7.     |RCC_APB2Periph_ADC1, ENABLE );
8.
9.     RCC_ADCCLKConfig(RCC_PCLK2_Div6);
10.    ADC_InitStructure.ADC_Mode = ADC_Mode_Independent;
11.    ADC_InitStructure.ADC_ScanConvMode = DISABLE;
12.    ADC_InitStructure.ADC_ContinuousConvMode = DISABLE;
13.    ADC_InitStructure.ADC_ExternalTrigConv =
14.    ADC_ExternalTrigConv_None;
15.    ADC_InitStructure.ADC_DataAlign = ADC_DataAlign_Right;
16.    ADC_InitStructure.ADC_NbrOfChannel = 1;
17.    ADC_Init(ADC1, &ADC_InitStructure);
18.
19.    ADC_TempSensorVrefintCmd(ENABLE);
20.
21.    ADC_Cmd(ADC1, ENABLE);
22.
23.    ADC_ResetCalibration(ADC1);
24.
25.    while(ADC_GetResetCalibrationStatus(ADC1));
26.
27.    ADC_StartCalibration(ADC1);
```

```

27. while(ADC_GetCalibrationStatus(ADC1));
28. }
29.
30. u16 get_adc(u8 ch)
31. {
32.     ADC_RegularChannelConfig(ADC1, ch, 1,
        ADC_SampleTime_239Cycles5 );
33.
34.     ADC_SoftwareStartConvCmd(ADC1, ENABLE);
35.     while(!ADC_GetFlagStatus(ADC1, ADC_FLAG_EOC ));
36.     return ADC_GetConversionValue(ADC1);
37. }
38.
39. u16 get_adc_average(u8 ch,u8 times)
40. {
41.     u32 temp_val=0;
42.     u8 t;
43.     for(t=0;t<times;t++)
44.     {
45.         temp_val+=get_adc(ch);
46.         Delay_Ms(5);
47.     }
48.     return temp_val/times;
49. }
50.
51. float get_temperature(void)
52. {
53.     u32 adcx;
54.     float temperate;
55.     adcx=get_adc_average(ADC_Channel_16,20);
56.     temperate=(float)adcx*(3.3/4096);
57.     temperate=(1.43-temperate)/0.0043+25;
58.     return temperate;
59. }
60.
61.

```

复制代码

temdet.c文件主要是内置温度传感器进行温度检测的相关配置，其配置流程和ADC差不多，主要增加了使能内部温度传感器和温度值计算，在此不再对流程进行介绍。

main.c文件

```

1. int main(void)
2. {
3.     float tem;
4.     Delay_Init();
5.     USART_Printf_Init(115200);
6.     adc_Init();

```

```

7.     while(1)
8.     {
9.         Delay_Ms(500);
10.        tem=get_temperature();
11.        printf("temp=%.2f\n",tem);
12.    }
13. }

```

复制代码

main.c文件主要进行相关函数初始化以及打印输出温度值。

4、下载验证

将编译好的程序下载到开发板并复位，串口打印情况具体如下：

