

Application Note

TMP6x 过采样



Nicole Khoury

摘要

本文档可帮助用户熟悉对 TMP6x 线性热敏电阻系列进行过采样的优势、该系列对恒温器技术的应用以及在校准和精度方面的优势。

TMP6x 热敏电阻具有高线性度，并在整个温度范围内具有始终如一的灵敏度，25°C 时的 TCR 为 6400ppm/°C。TMP6x 在整个 -40° 至 +125°C 工作范围内具有 0.2% 的典型 TCR 容差，可抵抗外部变化并具有强大的性能。发动机管理和电机监控系统经常利用极端温度，与在温度范围内表现出非线性行为的传统 NTC 热敏电阻相比，TMP6x 器件的线性度可在高温下提供更好的动态范围。对于恒温器和其他楼宇自动化系统，工作范围更窄。对于这些室温应用，过采样可以提高精度并带来更好的性能。本文档侧重于介绍室温应用以及 TMP6x 器件的性能如何超越 NTC 热敏电阻。

内容

1 引言.....	2
2 过采样.....	3
2.1 方法 1.....	3
2.2 方法 2.....	4
2.3 奈奎斯特速率.....	5
2.4 抖动.....	5
2.5 分辨率.....	6
3 参考文献.....	7

插图清单

图 1-1. 采用 X1SON 封装的 TMP61.....	2
图 2-1. FIFO 平均法 1.....	3
图 2-2. FIFO 平均法 2.....	4

表格清单

表 2-1. 比较求平均值之前的 TMP61 和 NTC 热敏电阻.....	6
表 2-2. 采用过采样方法 1 后 TMP61 和 NTC 的分辨率和误差.....	6

商标

所有商标均为其各自所有者的财产。

1 引言

更高精度的温度检测可允许系统在更接近热限值的情况下运行，从而提高性能和效率。在恒温器中，技术正在改变人们与环境互动和控制环境的方式，节能和效率多年来一直是设计迭代的主要驱动因素。

TMP6x 硅基热敏电阻的高线性度有助于在高温下实现更好的动态范围，但在室温下实现较低动态范围。但是，这种线性度可以实现 NTC 热敏电阻无法做到的软件改进，并且在实施后，TMP6x 器件可以实现更高的精度和分辨率，而不受动态范围的影响。除了 TMP6x 系列能够实现的更短响应时间、更低漂移和易用性之外，这还允许在超出室温的情况下实现更高精度。

NTC 热敏电阻在 25°C 时往往具有良好的精度，因为这是非常常见的校准点。除此之外，从 NTC 分压器获取的温度变得不太准确。更复杂的多点校准成本高且耗时，因为它们需要保温。TMP6x 器件的线性度允许单点校准，无需成本高昂的保温即可完成。然后，可以使用过采样方法在软件中进一步提高最终精度。通常，每八次过采样会将 ADC 的分辨率提高两位。本应用手册稍后将讨论不同的过采样方法以及应用实例。



图 1-1. 采用 X1SON 封装的 TMP61

2 过采样

过采样可以提高 ADC 分辨率，而无需求助于更昂贵、分辨率更高的 ADC。系统分辨率可能会受到 ADC 的限制，而过采样有助于在整个温度范围内有效提高 TMP61 的精度。

这对于室温应用特别有用，因为环境已经过优化。除了更高的精度和测量分辨率外，过采样还可以提高信噪比。温度值存储在一个数组中，因为它们是根据 ADC 位计算得出的。按照先入先出 (FIFO) 序列，随着新样片的添加，样片会在数组中移动。下面讨论的方法可用于温度转换中使用的各种值，包括温度、ADC 位值、分压器电压或计算得出的电阻。但是，本文档中的代码示例是对计算得出的温度求平均。

2.1 方法 1

第一种过采样方法会考虑整个周期的运行平均值。这可能更适合恒温器和其他环境监测系统，因为在求平均之前等待一个完整的周期可能会更好。这可以防止传递温度变化对系统响应产生重大影响。

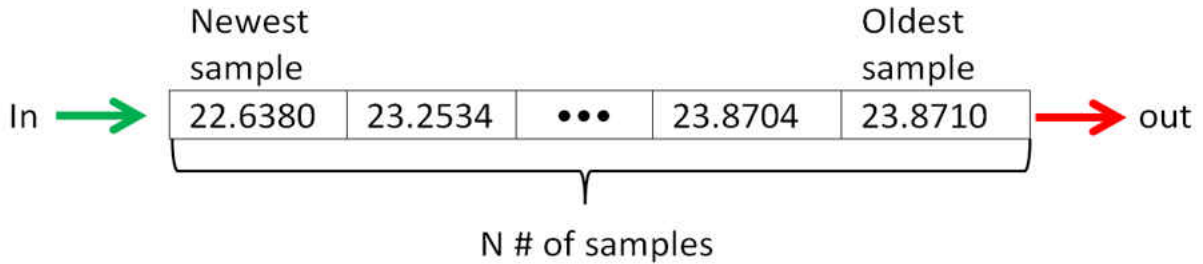


图 2-1. FIFO 平均法 1

2.1.1 方法 1 示例代码

```

// (1) Method one will read the ADC and average the last N values as set in "#define Tmp_1_length"
// FIFO setup of the temporary arrays and define the filter depth (samples to average)
#define Tmp_1_length 16 // Sample length for the averaging filter for oversampling
Float Tmp_1_array[Tmp_1_length]; // The FIFO arrays for averaging the ADC value

float ADC_AVG = 0; // This is the averaged ADC value over (x) samples
float ADC_Value = 0; // This is the most recent ADC value captured
int i = 0; // set to 0
float sum_array_1 = 0; // set to 0

void FIFO_AVG(void)
{
    // FIFO to average thermistor temperature
    i = 0; // reset to 0
    sum_array_1 = 0; // reset to 0
    for (i = 0; i < Tmp_1_Length - 1; i++) // shift the array as a FIFO and drop the last data
value
    {
        Tmp_1_array[i] = Tmp_1_array[i+1]; // makes all the arra indexes equal to the number
after them
    }
    Tmp_1_array[Tmp_1_length - 1] = ADC_Value; // add the new value to the beginning of the
array
    for (i = 0; i < Tmp_1_length; i++) // sum the array
    {
        sum_array_1 += Tmp_1_array[i]; // add all of the array elements
    }
    ADC_AVG = sum_array_1 / Tmp_1_length; // divide the sum of the array to get an average
}

// Read the ADC and place the bit value into ACD_Value
// Call the ADC_AVG function to get the last ADC value added and averaged into the array
FIFO_AVG();
// The ADC average value will be placed into ADC_AVG register

```

2.2 方法 2

另一种过采样方法涉及在每个周期对数组元素连续求平均值。这对于需要连续监控的系统（例如电机）是有益的。如果电机开始过热，则宜进行超快速采样并对采样值求均值，这样就可以密切监控所有温度变化。

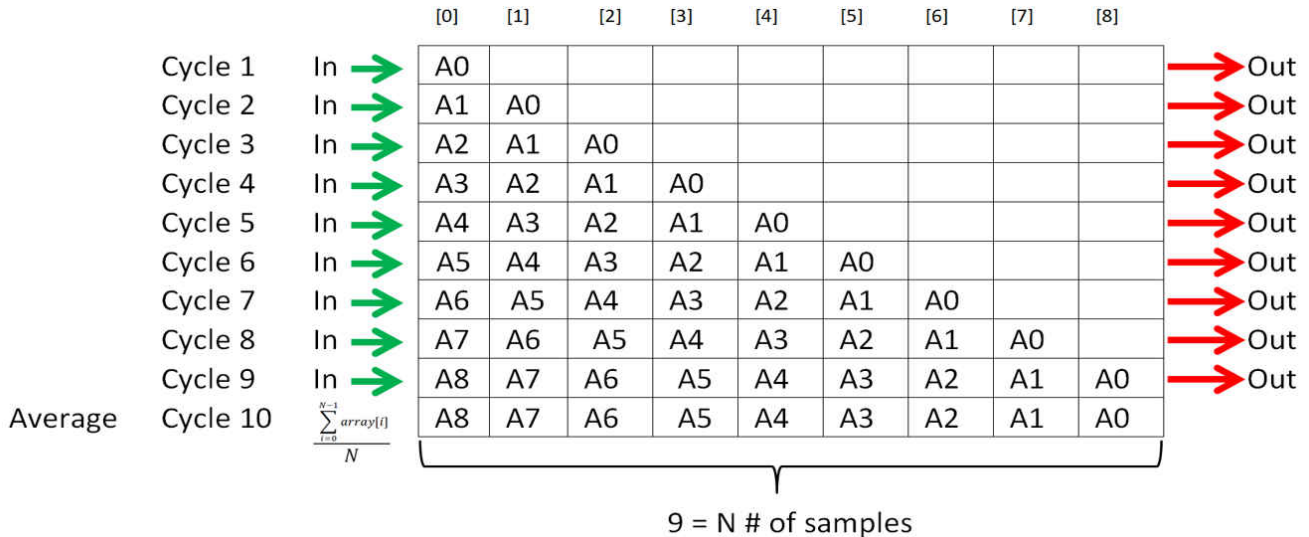


图 2-2. FIFO 平均法 2

2.2.1 方法 2 示例代码

```
// (2) Method two will read the ADC and insert that value into the FIFO N times as set in "#define
Tmp_1_length" then average the values
// FIFO setup of the temporary arrays and define the filter depth (samples to average)
#define Tmp_1_length 16 // Sample length for the averaging filter for oversampling
Float Tmp_1_array[Tmp_1_length]; // The FIFO arrays for averaging the ADC value

float ADC_AVG = 0; // This is the averaged ADC value over (x) samples
float ADC_Value = 0; // This is the most recent ADC value captured
int i = 0; // set to 0 for the array position counter
float sum_array_1 = 0; // set to 0
int j = 0; // set to 0 for the sample counter

void FIFO_AVG(void)
{
    // FIFO to average thermistor temperature
    i = 0; // reset to 0
    j = 0; // reset to 0
    sum_array_1 = 0; // reset to 0
    for (j = 0; j < Tmp_1_Length - 1; i++) // shift the array as a FIFO and drop the last data
value, add the new value to the array for (x) samples
    {
        // read the ADC and put the new value into the ADC Value register
        for (i = 0; i < Tmp_1_length - 1; i++) // shift the array as a FIFO and drop the last
data value
        {
            Tmp_1_array[i] = Tmp_1_array[i + 1]; // makes all the array indexes equal to the
number after them
        }
        Tmp_1_array[Tmp_1_length - 1] = ADC_Value; // add the new value to the beginning of
the array
        for (i = 0; i < Tmp_1_length; i++) // sum the array
        {
            sum_array_1 += Tmp_1_array[i]; // add all of the array elements
        }
        ADC_AVG = sum_array_1 / Tmp_1_length; // divide the sum of the array to get an average

// Call the FIFO_AVG routine to get N ADC values averaged
FIFO_AVG();
// The ADC average value will be placed into ADC_AVG register
```

2.3 奈奎斯特速率

在进行过采样以获得更高分辨率时需要考虑奈奎斯特速率。根据奈奎斯特定理可以发现，它是输入信号最高频分量的两倍。以高于最高频输入分量的频率进行采样被称为过采样。

2.4 抖动

抖动是故意向系统添加噪声的做法。就求平均值而言，这可以通过允许过采样来提高分辨率，从而帮助降低分辨率误差。如果这种目的性噪声被置于系统频率范围之外，就很容易将其滤除。这会导致最终测量具有更高的分辨率和更少的噪声。

忽略在电阻分压器上放置电容器是增加噪声的好方法，这通常对于求平均值足够了。

2.5 分辨率

表 2-1 比较了求平均值之前的 TMP61 和 NTC 热敏电阻，两个器件都未经校准。虽然 NTC 似乎具有更好的电压分辨率，但这不应被误解为精度。两个器件的温度误差大致相同。

表 2-1. 比较求平均值之前的 TMP61 和 NTC 热敏电阻

	TMP61	参考	NTC
	22.6368 °C	23.5468 °C	24.2259 °C
	23.2534 °C	23.5468 °C	24.1310 °C
	23.8704 °C	23.5468 °C	24.0361 °C
每 °C 的步长	0.6170 °C		0.0949 °C
	Error		Error
	0.9100 °C		0.6791 °C
	0.2934 °C		0.5842 °C
	-0.3236 °C		0.4893 °C

表 2-2 显示了使用方法 1 对 16 元素数组求平均值后 TMP61 和 NTC 的分辨率和误差。同样，虽然 NTC 的分辨率较小，但 TMP61 的温度稳定性使其从求平均值中受益匪浅。TMP61 的温度误差已经显著降低，NTC 更好的电压分辨率仍然没有提高精度。

表 2-2. 采用过采样方法 1 后 TMP61 和 NTC 的分辨率和误差

	TMP61	参考	NTC
	23.2920 °C	23.5468 °C	24.1369 °C
	23.2535 °C	23.5468 °C	24.1429 °C
	23.2149 °C	23.5468 °C	24.1488 °C
每 °C 的步长	0.0386 °C		0.0059 °C
	Error		Error
	0.2548 °C		0.5901 °C
	0.2933 °C		0.5961 °C
	0.3319 °C		0.6020 °C

3 参考文献

请参阅如下相关文档：

- 德州仪器 (TI), [热敏电阻设计工具](#)
- 德州仪器 (TI), 《降低热敏电阻在宽工作温度范围内线性化误差、容量和功率要求的方法》应用报告
- 德州仪器 (TI), 《具有 0402 和 0603 封装选项的 TMP61 $\pm 1\%$ 10k Ω 线性热敏电阻》数据表
- 德州仪器 (TI), 具有 ADC 的超小型、低成本模拟温度传感器测量电路模拟工程师电路：数据转换器

重要声明和免责声明

TI“按原样”提供技术和可靠性数据（包括数据表）、设计资源（包括参考设计）、应用或其他设计建议、网络工具、安全信息和其他资源，不保证没有瑕疵且不做任何明示或暗示的担保，包括但不限于对适销性、某特定用途方面的适用性或不侵犯任何第三方知识产权的暗示担保。

这些资源可供使用 TI 产品进行设计的熟练开发人员使用。您将自行承担以下全部责任：(1) 针对您的应用选择合适的 TI 产品，(2) 设计、验证并测试您的应用，(3) 确保您的应用满足相应标准以及任何其他功能安全、信息安全、监管或其他要求。

这些资源如有变更，恕不另行通知。TI 授权您仅可将这些资源用于研发本资源所述的 TI 产品的应用。严禁对这些资源进行其他复制或展示。您无权使用任何其他 TI 知识产权或任何第三方知识产权。您应全额赔偿因在这些资源的使用中对 TI 及其代表造成的任何索赔、损害、成本、损失和债务，TI 对此概不负责。

TI 提供的产品受 [TI 的销售条款](#) 或 [ti.com](#) 上其他适用条款/TI 产品随附的其他适用条款的约束。TI 提供这些资源并不会扩展或以其他方式更改 TI 针对 TI 产品发布的适用的担保或担保免责声明。

TI 反对并拒绝您可能提出的任何其他或不同的条款。

邮寄地址：Texas Instruments, Post Office Box 655303, Dallas, Texas 75265

Copyright © 2022，德州仪器 (TI) 公司