

## ATxmega 演示代码之内部温度传感器

——基于 ATxmega64A3-EK 和 JTAGICE mkII-CN

|      |                                                                  |      |       |      |                            |
|------|------------------------------------------------------------------|------|-------|------|----------------------------|
| 文档编号 | MAN0016A_CH                                                      |      |       |      |                            |
| 文档版本 | Rev. A                                                           |      |       |      |                            |
| 文档摘要 | 详细讲述了 XMEGA64A3 的内部温度传感器的使用方法                                    |      |       |      |                            |
| 关键词  | AVR、XMEGA、ATxmega64A3-EK、USB JTAGICE mkII-CN、Temperature、内部温度传感器 |      |       |      |                            |
| 创建日期 | 2010-08-17                                                       | 创建人员 | Kiler | 审核人员 | <a href="#">Hotislandn</a> |
| 文档类型 | 公开发布/仿真器配套文件                                                     |      |       |      |                            |
| 版权信息 | <a href="#">Mcuzone</a> 原创文档，转载请注明出处                             |      |       |      |                            |

## 更新历史

| 版本     | 时间         | 更新   | 作者    |
|--------|------------|------|-------|
| Rev. A | 2010-08-17 | 初始创建 | Kiler |
|        |            |      |       |
|        |            |      |       |
|        |            |      |       |

微控电子 乐微电子  
杭州市登云路 639 号 2B143  
销售 TEL: +86-571-89908193 13957118045  
支持 TEL: 13957118045 18913989166  
FAX: +86-571-88908193  
[www.mcuzone.com](http://www.mcuzone.com) [www.atarm.com](http://www.atarm.com)

# 1. 概述

ATxmega 是 ATMEL 推出的一款全新 MCU，与之前的 AVR 相比速度更快，ADC 和 DAC 性能更佳，功耗更低，而且有多达 8 个串口，因此在工控领域有较大用途。本系列文档以 ATxmega64A3-EK 开发板和本站的 USB AVR JTAGICE mkII-CN 仿真器为平台演示 ATxmega 的一些片上外设的操作。



本文演示 64A3 的内部温度传感器操作。

## 2. 概述

XMEGA 内部有一个经过工厂校准的温度传感器。ATMEL 在出厂前对 85 摄氏度温度点进行了校准，并将校准字写在了内部 Product Signature Row。

校准值是在 85 度时温度传感器输出的电压经过 ADC 转换后的数字，采用的是内部 1V 基准。

从理论上讲，温度传感器在绝对零度（0K，即-273 摄氏度）的时候输出的电压值是 0（ADC 转换结果为 0），这样我们就可以按照 85 度的校准值计算出每摄氏度的值，即 85 度校准值代表的跨度是  $273+85=358$  摄氏度，这样可以推算出每摄氏度的间隔数值是：85 度校准值/358。

但是，实际上温度传感器并不可能在绝对零度下工作，所以以上公式存在硬伤，而且从我们的实际测试来看，如果按照以上公式计算，算出来的值有较大误差，在 20-30 度温区会有 3-10 摄氏度。

为了进一步提高精度，我们采用两点校准，第一个是 ATMEL 的 85 度出厂校准值，第二点就是室温点。我们可以参考一下温度计或者其他测温工具读取室温，一般情况下误差在 2 摄氏度之内。然后在室温下读取内部温度传感器的 ADC 转换结果（同样采用内部 1V 基准进行转换）。然后通过 85 度和室温两个温度点的差值可以算出平均每度的电压值（或 ADC 转换值）。在测量目标温度的时候，根据读取的值，减去之前记录的室温的值，再除以每度的电压值，就可以算出当时的温度与室温的差值，加上室温即是当前的温度值。

用户可以按照实际需要测量的温度范围来改进计算方法以缩短代码提高精度。

### 3. 演示代码

这里我们只演示一下如何读取内部 product Signature Row 里面的温度校准数据，并不对温度计算方法进行演示，客户可以按照实际情况进行算法优化和简化。

示例代码：

```
unsigned char SP_ReadCalibrationByte( unsigned char index )
```

```
{
    unsigned char result;

    NVM_CMD = NVM_CMD_READ_CALIB_ROW_gc;
    result = pgm_read_byte(index);

    NVM_CMD = NVM_CMD_NO_OPERATION_gc;

    return result;
}
```

```
int main(void)
```

```
{
    unsigned char temperature_h,temperature_l;
    temperature_h = SP_ReadCalibrationByte( 0x008e0200+0x2f);
    temperature_l = SP_ReadCalibrationByte( 0x008e0200+0x2e);
    while(1)
    {

    }
}
```

以下是 ATXMEGA64A3 的存储器分配地址：

|                     |                          |
|---------------------|--------------------------|
| FLASH_BASE          | = 0x0800000              |
| EPPROM_BASE         | = 0x08C0000              |
| FUSE_BASE           | = 0x08F0020              |
| DATAMEM_BASE        | = 0x1000000              |
| APP_BASE            | = FLASH_BASE             |
| BOOT_BASE           | = FLASH_BASE + SIZE_APPL |
| PROD_SIGNATURE_BASE | = 0x008E0200             |
| USER_SIGNATURE_BASE | = 0x008E0400             |

## Mcuzone Application Notes

以下是 Product Signature Row 的内容:

### 4.23 Register Summary - Production Signature Row

| Address | Auto Load | Name        | Bit 7            | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0           | Page |
|---------|-----------|-------------|------------------|-------|-------|-------|-------|-------|-------|-----------------|------|
| +0x00   | YES       | RCOSC2M     | RCOSC2M[7:0]     |       |       |       |       |       |       |                 | 36   |
| +0x01   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x02   | YES       | RCOSC32K    | RCOSC32K[7:0]    |       |       |       |       |       |       |                 | 36   |
| +0x03   | YES       | RCOSC32M    | RCOSC32M[7:0]    |       |       |       |       |       |       |                 | 37   |
| +0x04   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x05   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x06   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x07   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x08   | NO        | LOTNUM0     | LOTNUM0[7:0]     |       |       |       |       |       |       |                 | 37   |
| +0x09   | NO        | LOTNUM1     | LOTNUM1[7:0]     |       |       |       |       |       |       |                 | 37   |
| +0x0A   | NO        | LOTNUM2     | LOTNUM2[7:0]     |       |       |       |       |       |       |                 | 37   |
| +0x0B   | NO        | LOTNUM3     | LOTNUM3[7:0]     |       |       |       |       |       |       |                 | 38   |
| +0x0C   | NO        | LOTNUM4     | LOTNUM4[7:0]     |       |       |       |       |       |       |                 | 38   |
| +0x0D   | NO        | LOTNUM5     | LOTNUM5[7:0]     |       |       |       |       |       |       |                 | 38   |
| +0x0E   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x0F   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x10   | NO        | WAFNUM      | WAFNUM[7:0]      |       |       |       |       |       |       |                 | 38   |
| +0x11   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x12   | NO        | COORDX0     | COORDX0[7:0]     |       |       |       |       |       |       |                 | 39   |
| +0x13   | NO        | COORDX1     | COORDX1[7:0]     |       |       |       |       |       |       |                 | 39   |
| +0x14   | NO        | COORDY0     | COORDY0[7:0]     |       |       |       |       |       |       |                 | 39   |
| +0x15   | NO        | COORDY1     | COORDY1[7:0]     |       |       |       |       |       |       |                 | 39   |
| +0x16   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x17   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x18   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x19   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x1A   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x1B   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x1C   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x1D   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x1E   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x20   | NO        | ADCACAL0    | ADCACAL0[7:0]    |       |       |       |       |       |       |                 | 39   |
| +0x21   | NO        | ADCACAL1    | ADCACAL1[7:0]    |       |       |       |       |       |       |                 | 40   |
| +0x22   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x23   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x24   | NO        | ADCBAL0     | ADCBAL0[7:0]     |       |       |       |       |       |       |                 | 40   |
| +0x25   | NO        | ADCBAL1     | ADCBAL1[7:0]     |       |       |       |       |       |       |                 | 40   |
| +0x26   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x27   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x28   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x29   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x2A   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x2B   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x2C   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x2D   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x2E   | NO        | TEMPSENSE0  | TEMPSENSE[7:0]   |       |       |       |       |       |       |                 | 41   |
| +0x2F   | NO        | TEMPSENSE1  |                  |       |       |       |       |       |       | TEMPSENSE[11:8] | 41   |
| +0x30   | NO        | DACAOFFCAL  | DACAOFFCAL[7:0]  |       |       |       |       |       |       |                 | 41   |
| +0x31   | NO        | DACACAINCAL | DACACAINCAL[7:0] |       |       |       |       |       |       |                 | 41   |
| +0x32   | NO        | DACBOFFCAL  | DACBOFFCAL[7:0]  |       |       |       |       |       |       |                 | 42   |
| +0x33   | NO        | DACBCAINCAL | DACBCAINCAL[7:0] |       |       |       |       |       |       |                 | 42   |
| +0x34   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x35   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x36   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x37   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x38   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x39   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| 0x3A    |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x3B   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x3C   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x3D   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |
| +0x3E   |           | Reserved    |                  |       |       |       |       |       |       |                 |      |