

使用 STM8 Nucleo-64 板构建温度计

引言

NUCLEO-8S208RB（围绕 STM8S208RBT6 器件构建）和 NUCLEO-8L152R8（围绕 STM8L152R8T6 器件构建）板能够评估所有 STM8S 系列和 STM8L 系列微控制器的主要功能。

本应用笔记演示了如何基于 STM8 Nucleo-64 板和 LM235 精确温度传感器构建一个简单的温度计。STM8S208RBT6 或 STM8L152R8T6 微控制器（取决于相应的板）读取温度值并通过 UART 接口进行传输。温度值显示在 PC 的终端窗口（可能基于 Windows HyperTerminal）上，该 PC 通过 RS232 或 FTDI 线连至 UART。

当 STM8 Nucleo-64 通过连至主机 PC 的 USB 线上电后，会在终端窗口上显示一条消息，提示用户输入最小和最大温度阈值。

每分钟在终端窗口上显示一次当前温度，当温度超过范围时会同时显示告警消息。

每小时会将一小时周期内的最小和最大温度记录在 MCU 数据 EEPROM 中。当任何时候按下下一个按钮时即可显示它们。

表 1. 适用产品

类型	产品编号
评估板	NUCLEO-8S208RB
	NUCLEO-8L152R8

参考文档

- 可从意法半导体网站获取：
 - *STM8 Nucleo-64 板数据摘要*（DB3591）
 - *STM8L152R8T6 Nucleo-64 板用户手册*（UM2351）
 - *STM8S208RBT6 Nucleo-64 板用户手册*（UM2364）
 - *ST232C 5 V 供电多通道 RS-232 驱动器和接收器数据手册*（DS1588）
 - *精确温度传感器数据手册*（DS0437），用于 LM235



1 先决条件

运行 STM8 Nucleo-64 板温度计演示应用所需的材料如下：

- 运行于 PC 上的终端窗口：终端模拟器软件可以是 Windows HyperTerminal（请参见第 附录 A 节 配置终端窗口）、TeraTerm Pro 或任何终端软件。
- RS232 零调制解调器线（收发线交叉连接）或 USB TTL 串行线。
- TYPE-A 转 USB mini-B 型 USB 数据线。

2 应用描述

2.1 硬件要求

本应用使用 STM8 Nucleo-64 板的板上 LED (LD2) 及相关电阻 (R1)。应用所需的外部无源元件列表如下。

表 2. 无源元件列表

元件名称	值	注释
R2	2.2 k Ω	上拉电阻
R3、R4 (可选)	100 Ω	限流电阻
C6, C7	100 nF	去抖动滤波器
按钮 1	-	标准按钮
按钮 2 (板上用户按钮)	-	标准按钮
C2、C3	100 nF	电荷泵电容
C1, C4	100 nF	输出电容
C5	100 nF	去耦电容

该应用使用一个 5V 供电的 ST232C 作为 RS232 接口的驱动接收。这个外部器件是必不可少的，因为 PC 的 COM 口工作于 12 V 标称供电，与工作于 5 V 的 STM8 UART 输入/输出不兼容。

此元件使用 SO16 封装，与 STM8 Nucleo-64 板匹配。请参考相关的数据手册以获取 ST232C 的更多信息。下表列出了封装的元件。

表 3. 封装元件列表

产品编号	元件描述	封装
ST232C (订购代码 ST232CN)	超高速超低功耗 5 V RS232 驱动器/接收器，用于 UART 5/12 V 电平转换器。	SO16
LM235	精确温度传感器 IC，工作于 -40 至 -125 °C 温度范围，具有 1 °C 的出厂精度。	SO8

2.2 应用原理图

下图显示了如何将 LM235 温度传感器、ST232C 驱动器/接收器以及按钮与 STM8 Nucleo-64 板连接。

按钮 1 和按钮 2 (板上用户按钮) 需要一个 RC 去抖动滤波器，以避免触发多次中断。按钮 1 和按钮 2 (板上用户按钮) 的去抖动滤波器包括 C7 和 C6 电容加 I/O 引脚内部上拉电阻 (约 45 k Ω)。

因为使用了 I/O 的内部上拉电阻，所以无需外部上拉电阻。

C2 和 C3 是两个电荷泵外部电容，用在 ST232C 中分别作为电压解耦器和电压逆变器。

流入 LM235 V+ 引脚的电流必须由一个电阻调节。传感器由 5 V 电源 (V_{DD}) 供电。跨越传感器的击穿电压与绝对温度直接成正比，灵敏度为 10 mV/°K。因为环境温度约为 300 °K，所以压降约为 3 V，这会为 2.2 k Ω 电阻留下 2 V，将电流调节为约 1 mA (强度用于确定数据手册中的典型值)。



若需具体的实现，请参考 *STM8 Nucleo-64* 板数据摘要（DB3591）中提供的板原理图。

图 1. STM8L 系列应用原理图

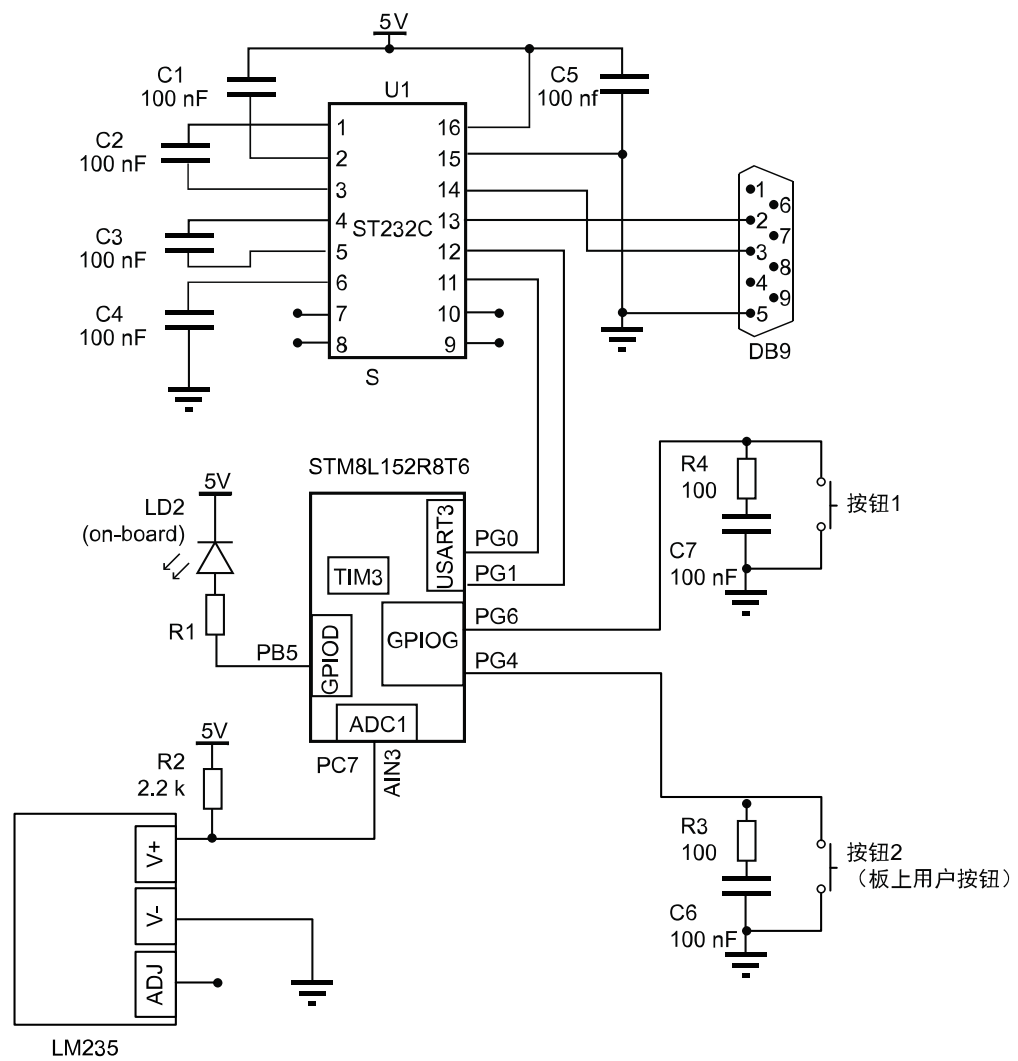
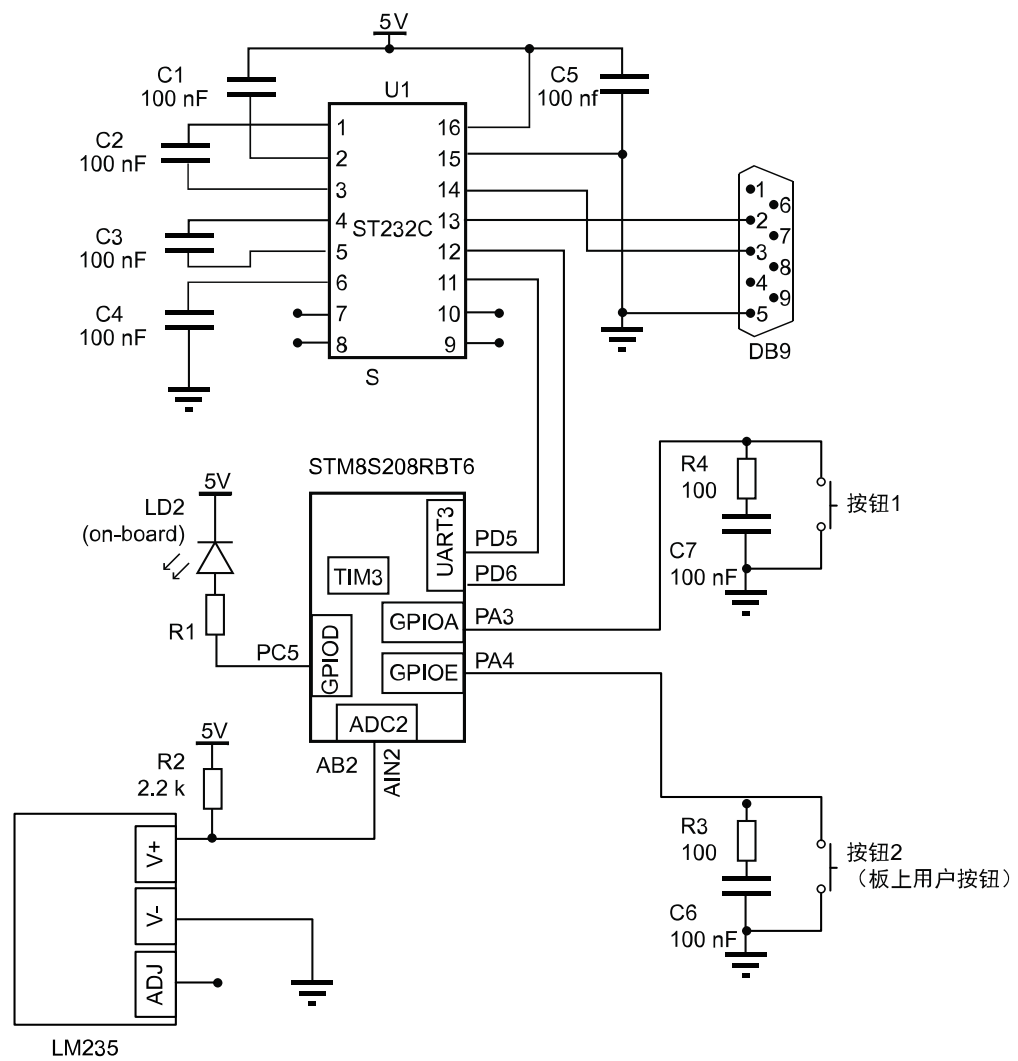


图 2. STM8S 系列应用原理图



2.3 应用原理

应用启动，消息显示在终端窗口上，提示用户输入关键的最小和最大温度阈值。

LM235 连续测量环境温度。每隔 **50 ms** 定时器中断时，由 **ADC2**（对于 **STM8S** 系列）或 **ADC1**（对于 **STM8L** 系列）转换模拟值。为提高温度测量的可靠性，会在经过 **1 秒** 后，对前 **16** 个测量的采样取平均，得到温度数据。

之后，将结果数据与当前最低和最高温度阈值进行比较，如有需要可修改阈值。若温度低于用户定义的低阈值或者超过高阈值，则 **LD2** 开启。

最后计算的平均温度会在终端窗口上每分钟显示一次，当有相关的关键温度告警时也会同时显示。

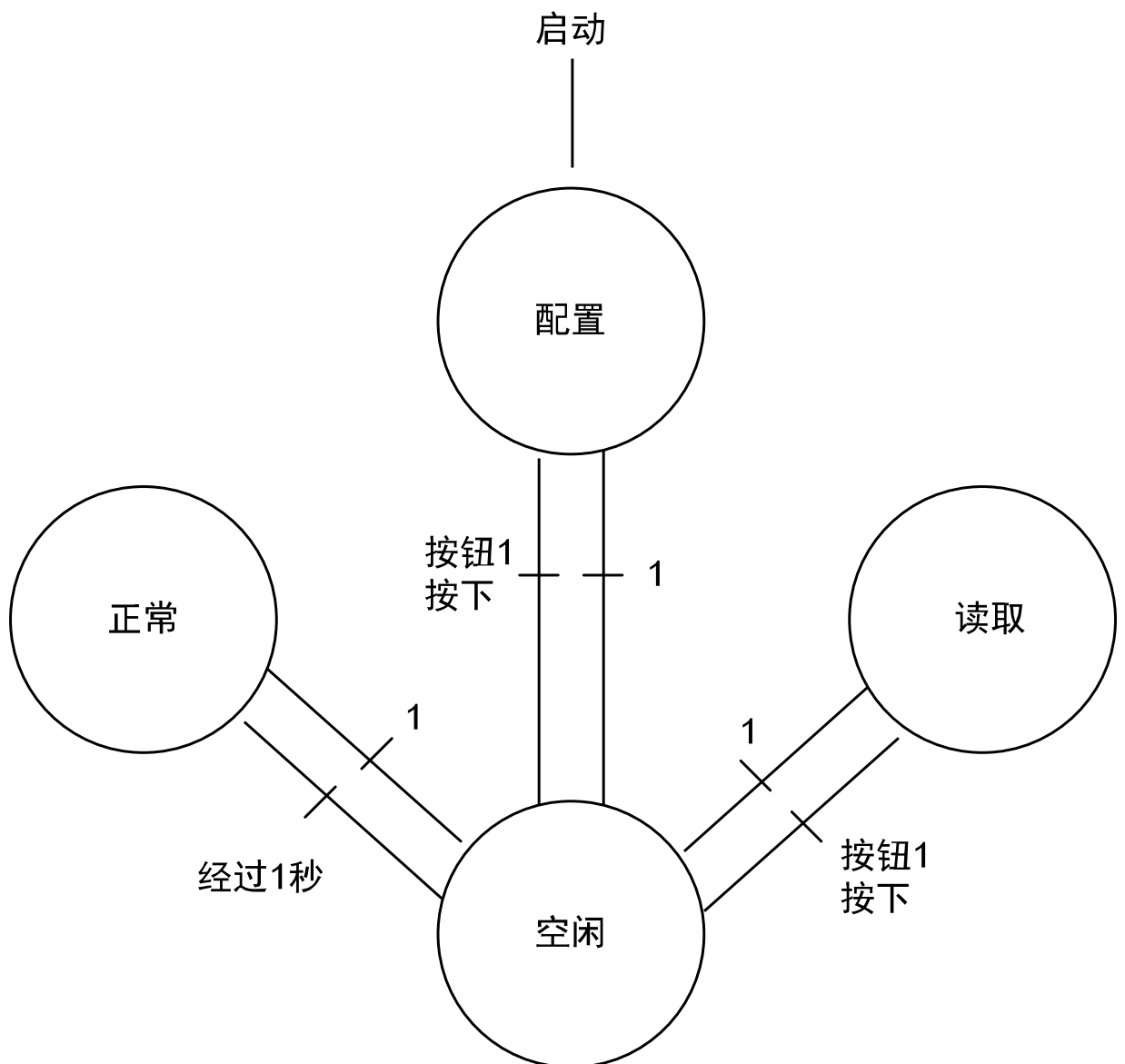
每隔一小时，会将该小时内的最小和最大温度记录在数据 **EEPROM** 中，并显示在终端窗口上。

按下按钮 **1** 会提示用户输入新的温度阈值。

按下按钮 **2**（板上用户按钮）会触发终端窗口上显示数据 **EEPROM** 中储存的所有最小和最大温度记录。**LD2** 关闭。

下图显示了应用状态图。

图 3. 应用状态机





下表说明了每次转换时应用所执行的行为。

表 4. 应用典型行为

应用状态	LED 状态	进入条件	动作
状态 0 空闲	-	默认状态	每隔 50 ms: 转换由 LM235 提供的模拟温度。
状态 1 配置	-	在启动时或按下按钮 1 时	消息显示在终端窗口上。 在第一次或进行更新时, 提示用户输入最小和最大温度阈值。
状态 2 读取	LD2 关闭	按下按钮 2	从数据 EEPROM 读取最小和最大温度值并显示在终端窗口上。
状态 3 Normal	若温度超过范围, 则 LD2 开启	每隔 1 秒 每隔 1 分钟 每隔 1 小时	<i>每隔 1 秒:</i> 从 16 个采样计算平均温度。如果需要, 更新当前的最小和最大温度值。 <i>每隔 1 分钟:</i> 在终端窗口上显示当前温度值和关键温度告警 (若有)。 <i>每隔 1 小时:</i> 将前一小时的最小和最大温度储存在数据 EEPROM 中。

通过 `State_Machine()` 函数, 根据定时器和外部事件管理该控制执行过程的算法 (请参见第 3.3.3 节 状态机流程图)。

2.4 启动应用

若需显示终端窗口, 您可以运行配置好的基于 Windows HyperTerminal (COM1 端口) 的 `thermometer.h` 终端, 或者如第 附录 A 节 配置终端窗口中的说明创建一个终端。

应用启动时, 会提示用户输入最小和最大温度阈值。温度阈值必须在 -40 至 +125 °C 范围内, 表示如下:

- 正温度值: "+XXX"。例如, 将 25 °C 输入为 "+025"。
- 负温度值: "-XXX"。例如, 将 -5 °C 输入为 "-005"。

图 4. 启动时的 STM8S 终端窗口

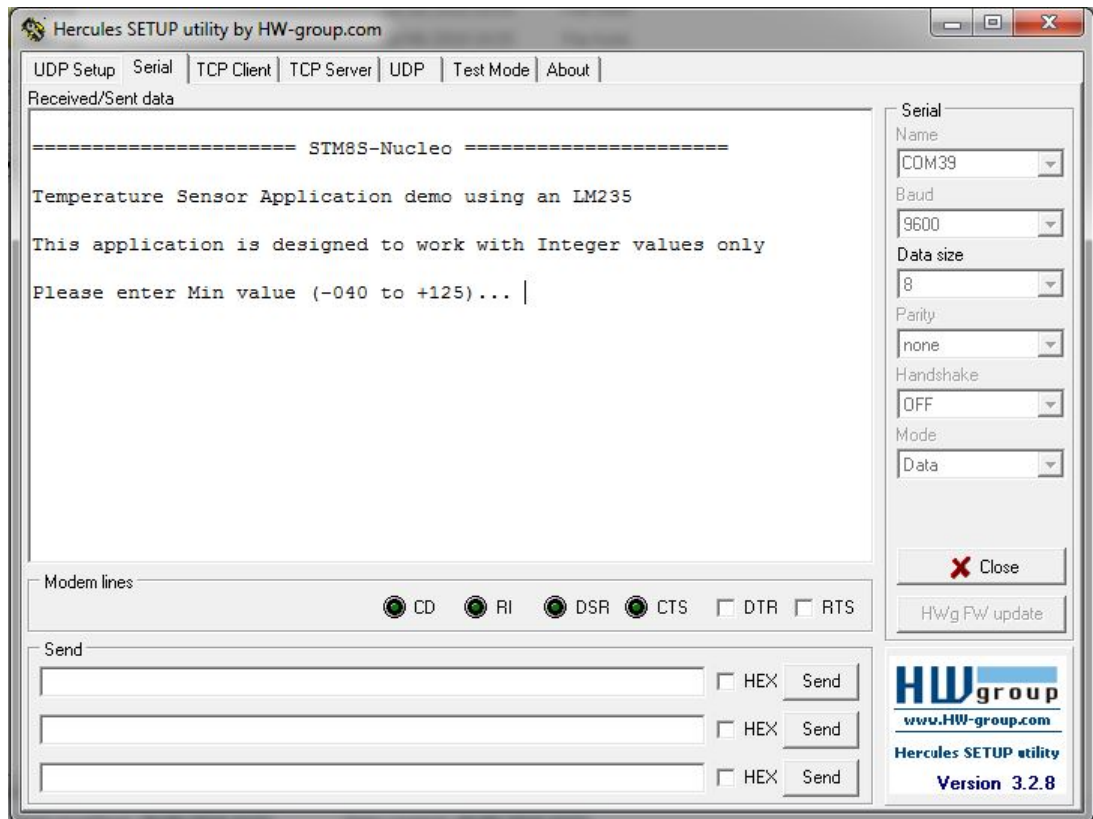
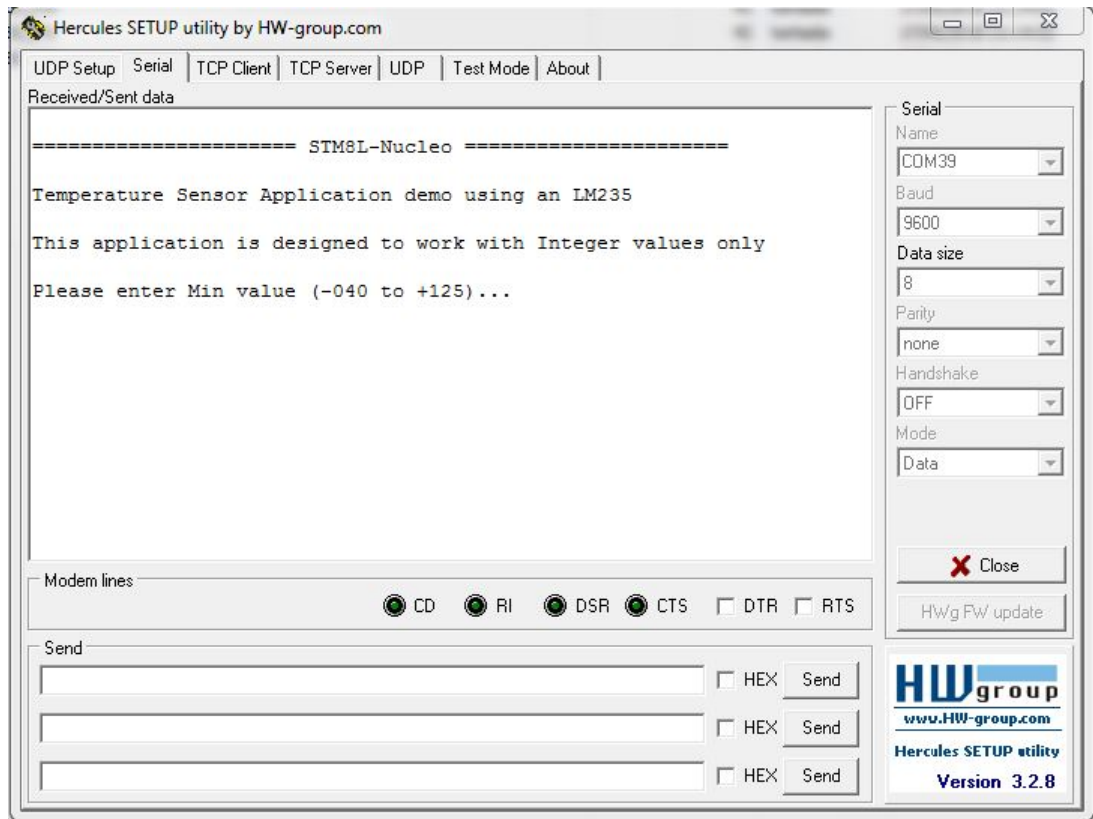


图 5. 启动时的 STM8L 终端窗口



3 软件说明

3.1 应用使用的 STM8 外设

温度计应用软件不使用 STM8S 系列或 STM8L 系列标准固件库。它含有优化的代码，使用直接寄存器访问来控制并使用 STM8S 系列和 STM8L 系列通用外设，如下所述：

- **CLK**
时钟控制器启用，并为 CPU 和外设提供正确的时钟频率。它将 HSI 时钟配置为 16 MHz 主时钟源，将 CPU 时钟预分频因子设置为 1。
- **GPIO**
STM8S 系列和 STM8L 系列 GPIO 用于开启和关闭 LD2，以及使用如下配置与按钮连接：
 - STM8S 系列：PA3 和 PE4（板子上仅有 PE4）
 - STM8L 系列：PG6 和 PG4（板子上仅有 PG4）
- **EXTI**
配置外部中断灵敏度，对于 STM8S 系列，当在 PA3 或 PA4 上检测到下降沿时，对于 STM8L 系列，当在 PG6 或 PG4 上检测到下降沿时触发中断。
- **Flash**
将一小时内的最小和最大温度值记录在数据 EEPROM 中并显示在终端窗口上。
- **U(S)ART**
STM8 系列使用 UART3，STM8L 系列使用 USART3 与 PC 终端软件通信。它的配置如下：
 - 波特率 = 9600 波特
 - 字长度 = 8 位
 - 1 个停止位
 - 无奇偶校验
 - 启用接收和发送

提示

对于 STM8L 系列，必须禁用 USART3 CLK。

通过轮询 UART3（STM8S 系列）或 USART3（STM8L 系列）外设上的每个接收和发送操作来管理通信。

- **ADC**
STM8S 系列 ADC2 的通道 2 或 STM8L 系列 ADC1 的通道 3 用于将 LM235 发出的模拟电压转换为数字电压，微控制器用这些数字电压计算当前温度值。
- **定时器 3（TIM3）**
此外设用于生成 50-ms 的时间基准，每隔 50 ms 触发温度捕获。

3.2 排除 STM8S 系列和 STM8L 系列标准固件库

因为本应用使用优化的代码，所以必须修改 *stm8s.h* 和 *stm8l.h* 文件，不包括 STM8S 系列标准固件库和 STM8L 系列标准固件库。这一步通过注释如下的定义语句来完成：

```
#define USE_STDPERIPH_DRIVER
```

3.3 应用软件流程图

本章给出了应用软件主循环、中断功能流程图和状态机流程图概述，以及终端通信功能的一些参考。

3.3.1 主流程图

主循环代码初始化所需的特性，解锁数据 EEPROM 程序并调用需要的函数，以实现通用的应用算法。

若需最小化软件实时时钟的时间漂移（请参考定时器触发的捕获），应将 HSI 时钟用作主时钟源。

初始化阶段完成、消息显示在终端窗口上后，提示用户输入最小和最大温度阈值。因此，*State_Machine()*函数在第一次调用时直接进入配置模式（状态 = 1，请参见第 3.3.3 节 状态机流程图）。



下图显示了应用软件主流程图。

图 6. 主流程图





3.3.2 中断功能流程图

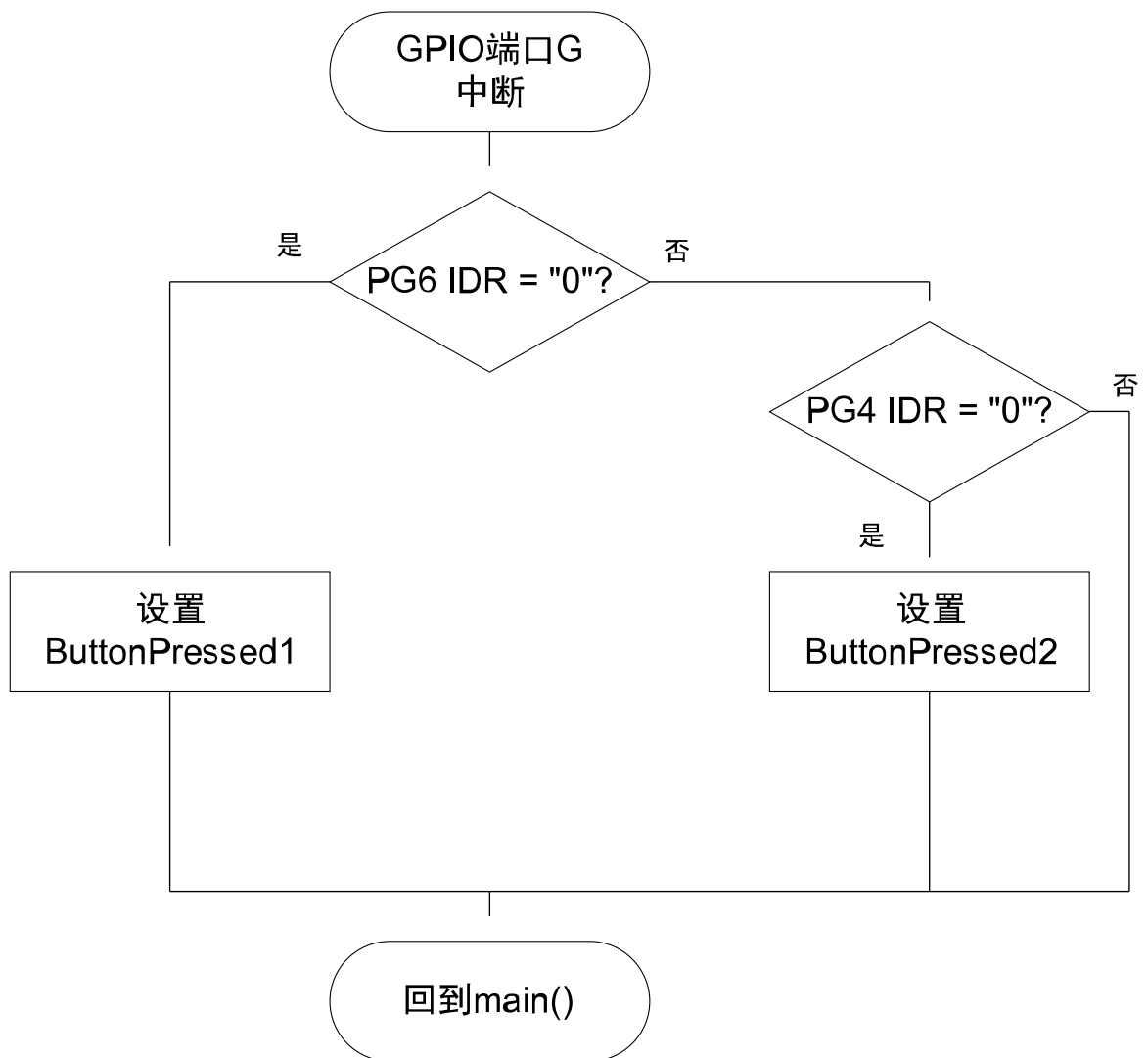
STM8L 系列的按钮捕获

每次按下按钮 1 或按钮 2（板上用户按钮）时，会触发中断，调用 PORTG_IRQhandler()函数。PORTG_IRQhandler()程序会识别测试端口 G 输入寄存器按下了哪个按钮，并相应设置 ButtonPressed1 或 ButtonPressed2 标志（请参见第 3.3.3 节 状态机流程图）。这些标志会在应用状态机中触发状态变化。



下图显示了 PORTG_IRQhandler()函数流程图。

图 7. STM8L 系列: PORTG_IRQhandler()函数流程图



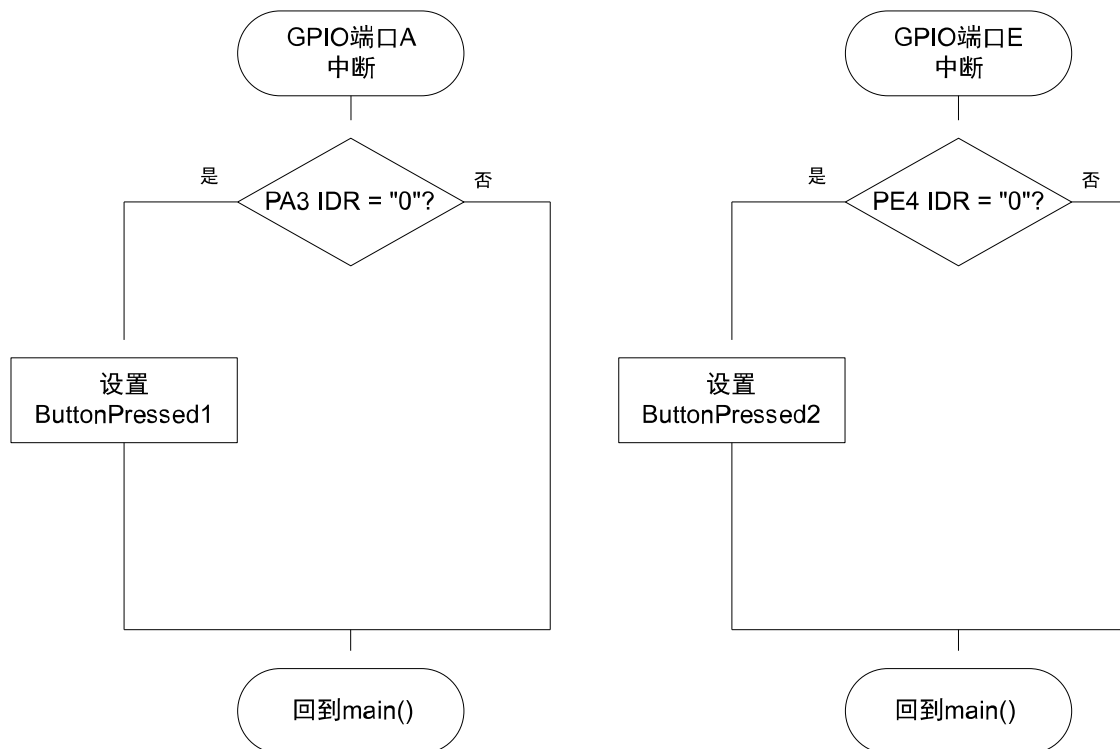
**STM8S 系列的按钮捕获**

每次按下按钮 1 时，会触发中断，调用 `PORTA_IRQHandler()` 函数。每次按下按钮 2 时，会触发中断，调用 `PORTE_IRQHandler()` 函数。程序相应设置 `ButtonPressed1` 或 `ButtonPressed2` 标志。这些标志会在应用状态机中触发状态变化。



下图显示了 STM8S 系列的 PORTA_IRQhandler()和 PORTE_IRQhandler()函数流程图。

图 8. STM8S 系列: PORTA_IRQHandler()和 PORTE_IRQHandler()函数流程图





定时器触发的捕获

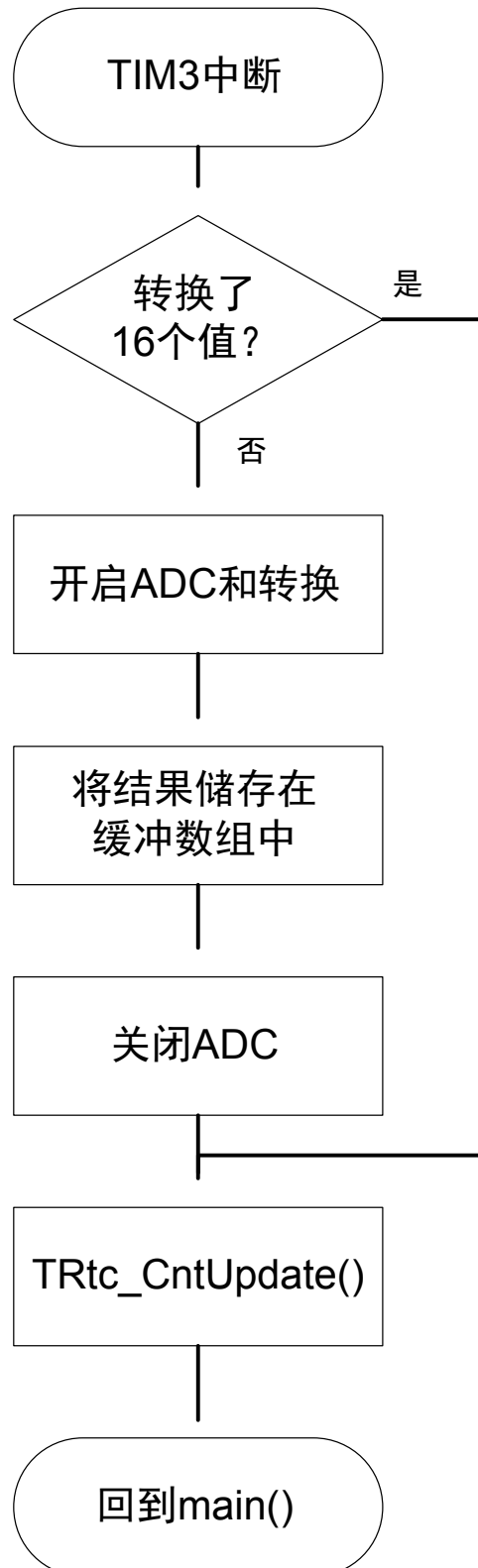
TIMER3 配置为每隔 50 ms 生成一次中断，触发温度捕获。在 ADC2 通道 2（对于 STM8S 系列）或 ADC1 通道 3（对于 STM8L 系列）的每次转换后，数字值会储存在缓冲数组中以进行进一步计算，ADC 关闭。在缓冲数组中最多能储存 16 个采样。当计算一秒周期内的平均温度时，通过简单执行右移即可除以 2。

TIM3_IRQHandler()调用 TRtc_CntUpdate()函数，模拟实时时钟。它设置标志，表示秒、分和小时。RTC 程序基于 TIM3 50 ms 时间基准。每秒时，状态机会自动切换为状态 3（普通模式）（请参见第 3.3.3 节 状态机流程图）。



下图显示了 TIM3_IRQHandler()函数流程图。

图 9. TIM3_Init()函数流程图





3.3.3 状态机流程图

State_Machine 函数实现算法，根据定时器和外部事件控制应用执行过程。状态变量的不同值表示应用模式：

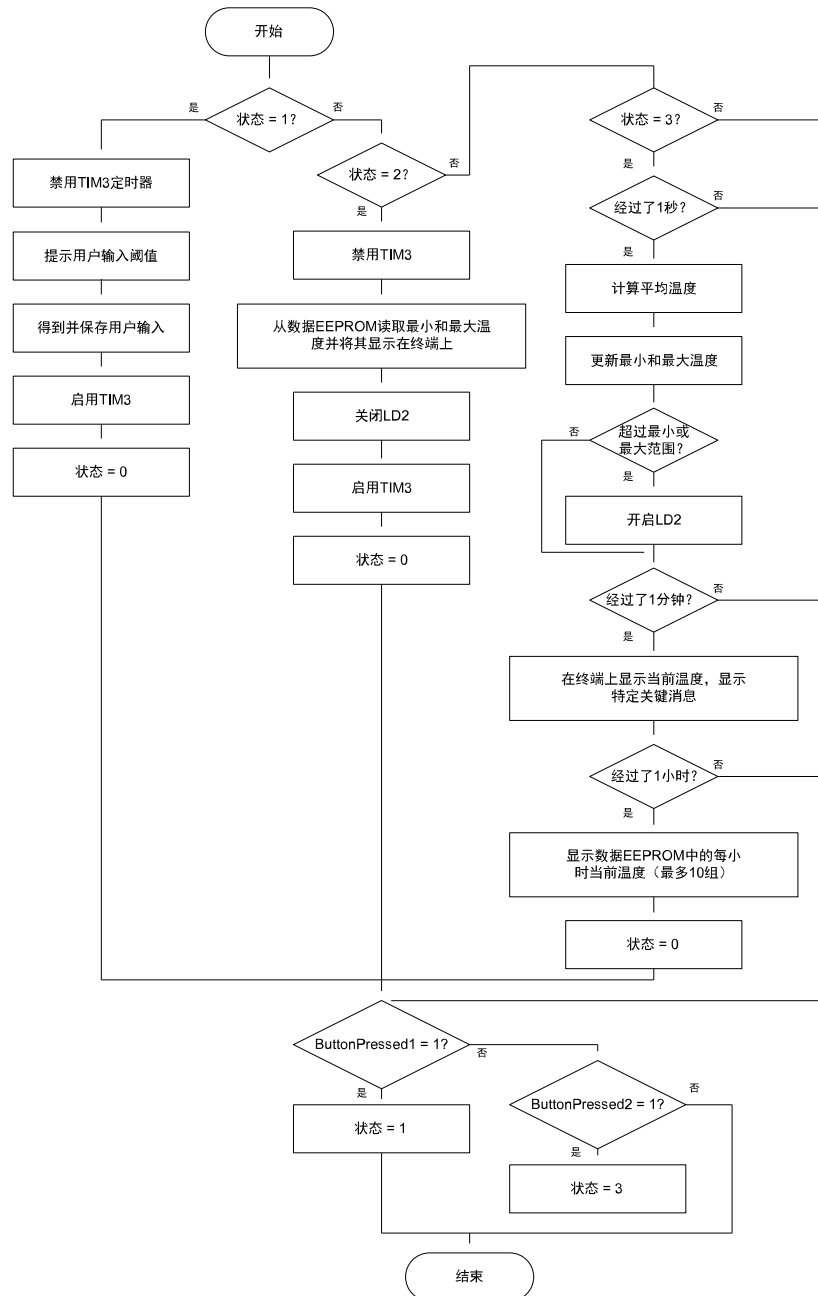
- 状态 = 0：空闲模式
这是状态机默认状态。每隔 50 ms 会对 LM235 提供的模拟温度执行转换。按下按钮 1 或按钮 2（板上用户按钮）则退出此状态。
- 状态 1 = 配置模式
在状态机配置模式中，终端消息提示用户输入最小和最大温度阈值。从终端窗口读取这些值，并以十进制格式记录。在启动时或按下按钮 1 时进入此状态。
- 状态 2 = 读取模式
在状态机读取模式中，应用从数据 EEPROM 读回最小和最大温度对，并将其和一条消息显示在终端窗口。LD2 也关闭。仅当按下按钮 2 时才会进入此状态。
- 状态 3 = 正常模式
每秒都会进入状态机正常模式。它被 TRtc_CntUpdate() 程序触发，监控经过的应用时间。每次进入此状态时，都会用缓冲数组中储存的 16 个采样计算平均温度。若此温度值超过高阈值或低于低阈值，则 LD2 开启。

每分钟在终端窗口上显示一次当前温度，当有关键消息时也会同时显示。会在数据 EEPROM 中记录一小时周期内的最大和最小测量温度。EEPROM 中最多能储存 10 对。这意味着会保存 10 小时的测量历史。当写入第 11 小时的数据时会复位 EEPROM 地址指针，覆盖之前的数据。因此，之前 10 小时的数据会被覆盖。

当进入状态 1 和 2 时，因为退出了正常模式，所以 TIM3 计数器禁用。当退出这些状态时，会重新分配定时器寄存器，计数器再度启用，重新开始默认执行模式。

请参见下面的 State_Machine()函数流程图。

图 10. State_Machine 函数流程图



3.3.4

终端通信功能

若需终端通信功能的详细信息，请参考第 附录 A 节 配置终端窗口。

附录 A 配置终端窗口

对于所有终端类型，连至 STM8 Nucleo-64 板的终端窗口必须配置如下有效设置：

- 通信端口：COM1 或其它可用端口
- 每秒比特数：9600
- 数据位：8
- 校验位：无
- 停止位：1
- 流控制：无

为提供准备就绪的应用样例，在项目目录内提供了使用 Windows HyperTerminal 和 COM1 端口的预先配置好的终端。执行项目中的 .ht 文件即可方便地将其启动。

然而，用户也能基于 Windows HyperTerminal，使用 STM8 Nucleo-64 板建立一个新连接，并按如下步骤关联到本样例：

1. 打开 Windows HyperTerminal 应用，选择连接名称，例如“MyConnection”，点击 OK 验证。

图 11. 启动 Windows HyperTerminal



2. 选择 COM1 或您电脑上的其它可用端口，点击 OK 验证您的选择，其它字段可设为默认值。

图 12. 选择通信端口



3. 按下图所示配置通信端口属性。Windows HyperTerminal 启动，可以开始通信。

图 13. 配置连接属性



4. 若需检查通信设置：
 - 从 HyperTerminal 主菜单选择 Call > Disconnect，断开 HyperTerminal。
 - 通信结束后，转至 MyConnection Properties 菜单中的 Settings 选项卡。参数设置应如下所示。

图 14. 检查通信设置



- 最后，点击 MyConnection Properties 菜单中的 ASCII Setup，确保 ASCII 参数与下图中显示的相匹配。

图 15. ASCII 设置参数



- 关闭 MyConnection Properties 菜单，从 HyperTerminal 主菜单选择 Call > Call 重新启动通信。现在，STM8 Nucleo-64 应用已经准备好启动了。



版本历史

表 5. 文档版本历史

日期	版本	变更
2018 年 6 月 29 日	1	初始版本。

目录

1	先决条件.....	2
2	应用描述.....	3
2.1	硬件要求	3
2.2	应用原理图	3
2.3	应用原理	7
2.4	启动应用	9
3	软件说明.....	12
3.1	应用使用的 STM8 外设	12
3.2	排除 STM8S 系列和 STM8L 系列标准固件库	12
3.3	应用软件流程图	12
3.3.1	主流程图.....	12
3.3.2	中断功能流程图	15
3.3.3	状态机流程图	24
3.3.4	终端通信功能	25
附录 A	配置终端窗口.....	26
	Revision history.....	27



表一览

表 1.	适用产品.....	1
表 2.	无源元件列表	3
表 3.	封装元件列表	3
表 4.	应用典型行为	9
表 5.	文档版本历史	27



图一览

图 1.	STM8L 系列应用原理图	5
图 2.	STM8S 系列应用原理图	6
图 3.	应用状态机	8
图 4.	启动时的 STM8S 终端窗口	10
图 5.	启动时的 STM8L 终端窗口	11
图 6.	主流程图	14
图 7.	STM8L 系列: PORTG_IRQhandler()函数流程图	17
图 8.	STM8S 系列: PORTA_IRQhandler()和 PORTE_IRQhandler()函数流程图	20
图 9.	TIM3_Init()函数流程图	23
图 10.	State_Machine 函数流程图	25
图 11.	启动 Windows HyperTerminal	26
图 12.	选择通信端口	26
图 13.	配置连接属性	26
图 14.	检查通信设置	26
图 15.	ASCII 设置参数	26



重要通知 - 请仔细阅读

意法半导体公司及其子公司（“ST”）保留随时对 ST 产品和/或本文档进行变更、更正、增强、修改和改进的权利，恕不另行通知。买方在订货之前应获取关于 ST 产品的最新信息。ST 产品的销售依照订单确认时的相关 ST 销售条款。

买方自行负责对 ST 产品的选择和使用，ST 概不承担与应用协助或买方产品设计相关的任何责任。

ST 不对任何知识产权进行任何明示或默示的授权或许可。

转售的 ST 产品如有不同于此处提供的信息的规定，将导致 ST 针对该产品授予的任何保证失效。

ST 和 ST 标志是 ST 的商标。关于 ST 商标的其他信息，请访问 www.st.com/trademarks。其他所有产品或服务名称是其各自所有者的财产。

本文档中的信息取代本文档所有早期版本中提供的信息。

© 2019 STMicroelectronics - 保留所有权利