MICROCHIP

AN2565

无ROM Cortex®-M器件的SAM-BA®监视器

简介

SAM Boot Assistant（SAM-BA）允许使用USB或UART主机进行在系统编程（In-system Programming，ISP），而无需任何外部编程接口。通常，SAM-BA监视器在出厂时编程到ROM（如果存在）中。如果ROM不存在，默认情况下不支持SAM-BA。要在无ROM器件中支持SAM-BA，可以将SAM-BA监视器应用程序加载到闪存中。

本应用笔记重点介绍无ROM Cortex-M器件并说明SAM-BA监视器及其在这些器件上的接口。SAM-BA监视器兼容现有SAM-BA软件工具，但与其他Cortex-M器件相比有一些差异。本文档将说明这些差异。

特性

- 允许最终用户在没有编程器的情况下编程、验证和保护器件
- 允许最终用户升级应用程序固件
- 利用I/O引脚提供可配置启动条件
- 支持USB-CDC和UART接口
- 提供源代码，可根据用户需求进行定制

目录

简介.....1

特性.....1

1. SAM-BA简介.....4
   - 1.1 小程序简介.....4
   - 1.2 SAM-BA监视器.....4

2. 无ROM Cortex-M器件.....6

3. 硬件要求.....7

4. 软件实现.....8
   - 4.1 存储器映射.....8
   - 4.2 设计注意事项.....8
   - 4.3 配置.....8
   - 4.4 存储器使用情况.....9

5. 重新编译SAM-BA监视器.....10
   - 5.1 使用Atmel Studio将SAM-BA监视器移植到SAM L22的示例.....10
   - 5.2 使应用程序重新指向新的起始地址.....11
   - 5.3 为SAM-BA主机重新生成小程序.....12

6. 使用SAM-BA监视器.....14
   - 6.1 编程SAM-BA监视器.....14
   - 6.2 进入SAM-BA监视器.....14

7. 在Windows®上运行SAM-BA.....16
   - 7.1 连接SAM-BA GUI.....16
   - 7.2 闪存编程.....16
   - 7.3 脚本.....17

8. 其他解决方案.....19

9. 参考资料.....20

10. 版本历史.....21

Microchip网站.....22

变更通知客户服务.....22

客户支持.....22

Microchip器件代码保护功能.....22

法律声明.....23

商标.....23

DNV认证的质量管理体系.....24

全球销售及服务网点.....25

1. SAM-BA简介

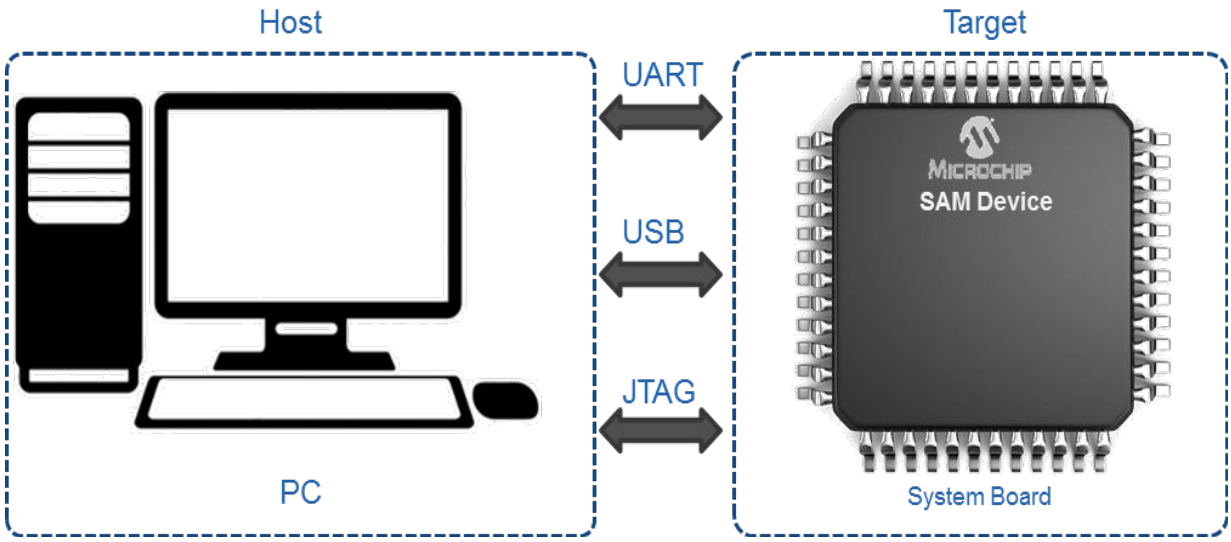
SAM Boot Assistance（SAM-BA）软件提供一套开放式工具来编程基于Microchip SAM Arm® Thumb®的单片机。这些工具基于通用动态链接库（Dynamic Linked Library，DLL）——AT91Boot_DLL，该动态链接库供SAM-BA和所有ISP工具使用。

SAM-BA用于在PC上运行的主机图形用户界面（SAM-BA GUI）与现有系统板上的目标器件（Cortex-M器件）之间建立通信。

所使用的通信协议如下：

- USB
- UART
- JTAG

图1-1. SAM-BA连接概述



1.1 小程序简介

要应答主机请求，在主机请求对特定存储器执行读/写/擦除操作时，目标必须能够执行多个功能。因此，必须将称为*applet*的函数提前加载到RAM中。“applet”就是目标上运行的一种小程序，供SAM-BA用来使能非易失性存储器的编程、低级初始化或其他外设。对于每个SAM器件而言，与芯片交互的每个存储器都有一个专用的小程序。

1.2 SAM-BA监视器

要与SAM-BA GUI进行通信，必须在目标器件上放置名为*SAM-BA监视器*的命令解释程序。一些SAM器件在出厂时已将SAM-BA监视器编程到ROM中（见图1-2），另一些器件没有SAM-BA监视器，因此必须将SAM-BA监视器加载到嵌入式闪存中（见图1-3）。后一种类型的器件即称为*无ROM*器件。

图1-2. 带ROM的SAM系统板

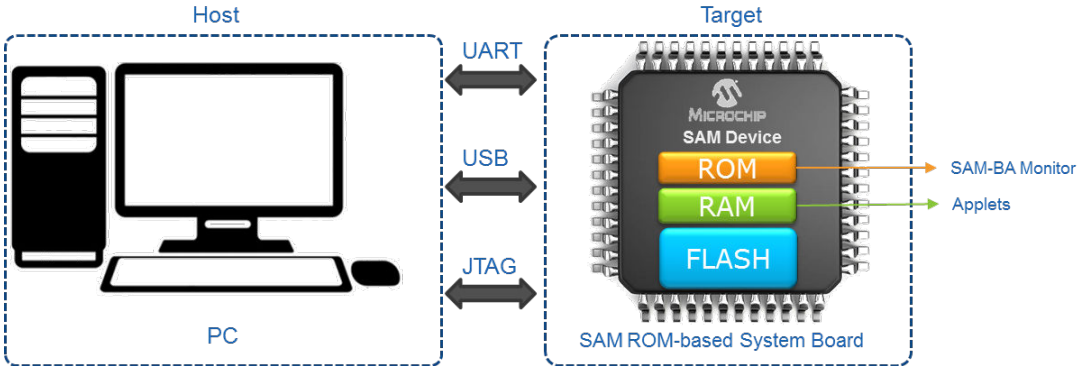
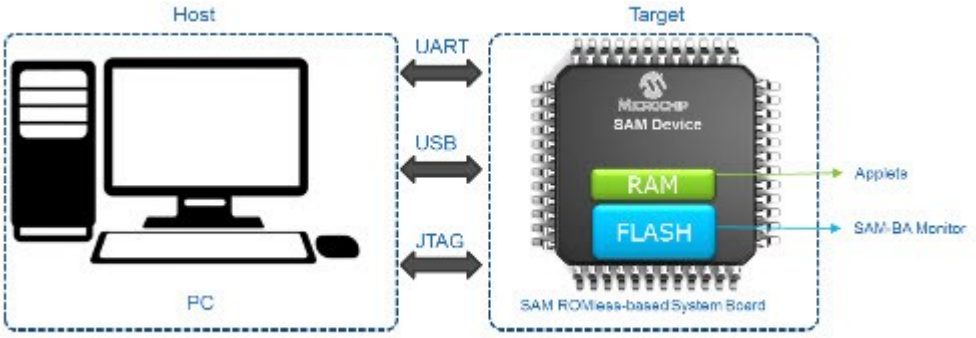


图1-3. 不带ROM的SAM系统板



SAM-BA监视器为现场编程片上闪存提供了一种简单的方法。SAM-BA监视器支持USB-CDC和UART通信。SAM-BA将持续检测UART和/或USB接口的启动条件。当检测到启动条件时，SAM-BA监视器会进入无限循环，等待SAM-BA GUI命令。

USB接口的启动条件是枚举完成。当SAM-BA监视器检测到主机枚举完成时，它将开始等待SAM-BA命令。

UART接口的启动条件为“#”。当SAM-BA监视器接收到该字符时，它将开始等待SAM-BA命令。

2. 无ROM Cortex-M器件

下表列出了*未*将SAM-BA监视器编程到ROM中（即未进行出厂编程，SAM-BA监视器位于闪存中）的现有Cortex-M器件。Microchip为专用XPlained Pro板提供了SAM-BA监视器示例，不过SAM-BA监视器还是一款开源工具，可以轻松移植到无ROM产品系列的任何器件中。

表2-1. 无ROM Cortex®-M器件

产品系列	Xplained Pro板（板上器件）	内核
SAM L22	SAML22 Xplained Pro（SAML22N18A）	Cortex-M0+
SAM L21	SAML21 Xplained Pro（SAML21J18B）	Cortex-M0+
SAM D21	SAMD21 Xplained Pro（SAMD21J18A）	Cortex-M0+
SAM C21	SAMC21 Xplained Pro（SAMC21J18A）	Cortex-M0+
SAM D5x/E5x	SAME54 Xplained Pro（SAME54P20A）	Cortex-M4

3. 硬件要求

SAM-BA监视器使用通用I/O（自举程序进入引脚）来确定SAM-BA监视器启动还是执行应用程序闪存。

下表列出了相应接口的信号及其说明。

表 3-1. SAM-BA监视器UART模式连接

信号名称	说明
UART RXD	串行通信（115200-8-N-1）接收线路
UART TXD	串行通信（115200-8-N-1）发送线路
自举程序进入引脚	用于确定执行情况（即，执行应用程序或侦听SAM-BA主机命令）的GPIO引脚

表3-2. SAM-BA监视器USB模式连接

信号名称	说明
USB D+	与主机的USB连接
USB D-	与主机的USB连接
自举程序进入引脚	用于确定执行情况（即，执行应用程序或侦听SAM-BA主机命令）的GPIO引脚

注：有关使能串行或USB通信的任何器件特定要求的信息，请参见相应器件的数据手册和/或原理图清单部分。

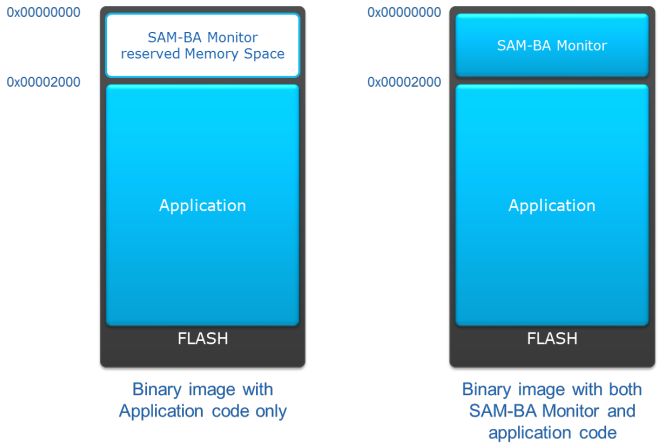
4. 软件实现

本应用笔记随附基于Atmel Studio的示例SAM-BA监视器应用程序。提供完整的源代码，并可根据用户要求轻松移植到IAR/KEIL。

4.1 存储器映射

需要为SAM-BA监视器应用程序预留闪存空间。此空间从0x00000000到应用程序起始地址（0x00002000），可确保不同编译器组合（即，GCC/IAR和UART，和/或USB-CDC ）之间的一致性。

图4-1. 器件存储器映射



4.2 设计注意事项

以下是该实现的设计注意事项：

- UART和USB驱动程序支持无ROM Cortex-M器件中使用的大多数外设
- 在本应用笔记中，必须使用Atmel Studio 7将SAM-BA监视器编程到器件中
- 用于开发和测试的工具版本是最新版本的Atmel Studio 7和SAM-BA GUI 2.xx
- 使用ARM/GNU编译器6.2.1执行编译，并进行优化 -Og, STACK_SIZE=0x280, --specs=nano.specs

4.3 配置

SAM-BA监视器应用程序支持根据用户需求进行定制的配置。以下是定制的相应配置：

- UART和/或USB通信接口：
  - 在使用UART时，器件上为SERCOM实例
- 控制SAM-BA监视器执行的自举程序进入引脚
- 应用程序起始地址，更改应用程序起始地址需要
  - 为SAM-BA主机重新生成小程序

– 使应用程序重新指向新的起始地址

下表列出了示例应用程序中的配置。

表4-1. 示例应用程序中的SAM-BA配置

器件	配置	当前设置
SAM L22（SAML22N18A）	通信接口	USB和UART
	SERCOM实例	SERCOM4（PC24D和PC25D）
	自举程序进入引脚	PC01
SAM L21（SAML21J18B）	通信接口	USB和UART
	SERCOM实例	SERCOM3（PA22C和PA23C）
	自举程序进入引脚	PA02
SAM D21（SAMD21J18A）	通信接口	USB和UART
	SERCOM实例	SERCOM3（PA22C和PA23C）
	自举程序进入引脚	PA15
SAM C21（SAMC21J18A）	通信接口	USB和UART
	SERCOM实例	SERCOM4（PB10D和PB11D）
	自举程序进入引脚	PA28
SAM E54（SAME54P20A）	通信接口	USB和UART
	SERCOM实例	SERCOM2（PB25D和PB24D）
	自举程序进入引脚	PB31

4.4 存储器使用情况

SAM-BA监视器应用程序的目标是使用有限的闪存和SRAM，因此可以将其移植到32K闪存和4K SRAM等小尺寸器件中。下表说明了不同器件的闪存和SRAM使用情况。

注：最新版本的Atmel Studio，ARM/GNU编译器6.2.1，进行优化 -Og, STACK_SIZE=0x280, --specs=nano.specs。

表4-2. SAM-BA监视器的存储器使用情况

器件	USB + UART（闪存/SRAM）（以字节为单位）	仅USB（闪存/SRAM）（以字节为单位）	仅UART（闪存/SRAM）（以字节为单位）
SAML22N18A	4488/1168	3384/1160	2400/688
SAML21J18B	4508/1168	3404/1160	2420/688
SAMD21J18A	4524/1168	3436/1160	2464/688
SAME54P20A	4768/1168	3716/1160	2952/688

注：该存储器的使用情况信息可能随开发环境的变化（即，工具链变化和选项变化等）而变化。

5. 重新编译SAM-BA监视器

本应用笔记提供SAM-BA监视器的源代码，其中包括针对GCC配置的Atmel Studio项目。要将SAM-BA监视器用于其他器件或配置，必须重新编译SAM-BA监视器项目，使其具备所需变化。

使用以下步骤针对SAM L22进行重新编译。任何其他无ROM Cortex-M器件均可遵循相似的步骤。

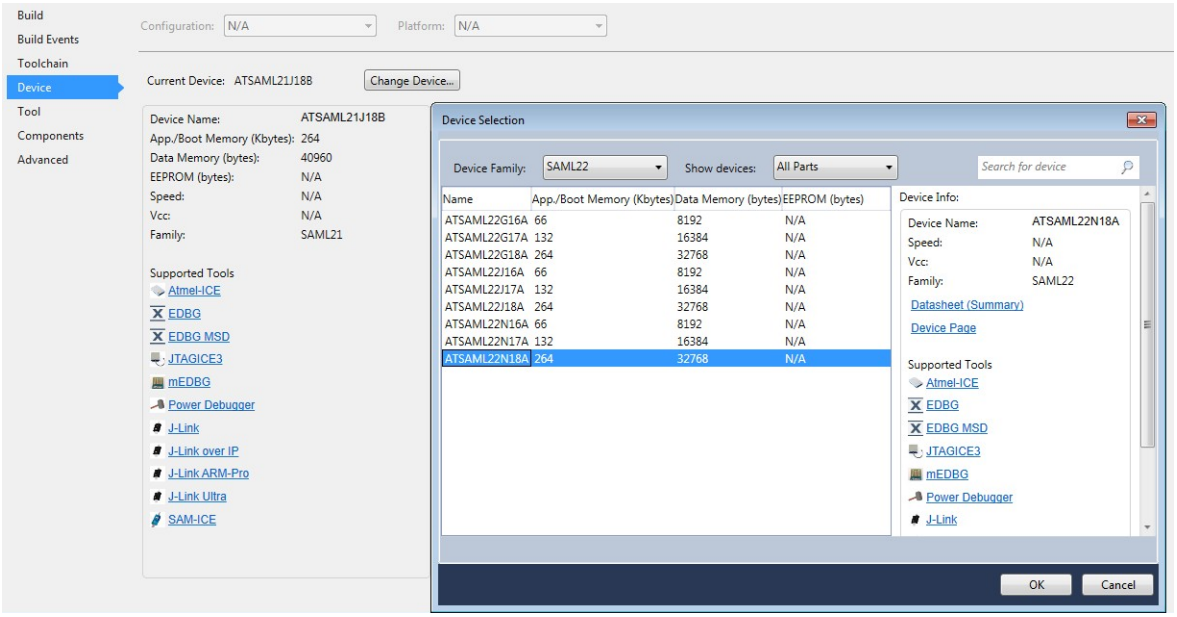
5.1 使用Atmel Studio将SAM-BA监视器移植到SAM L22的示例

1. 打开本应用笔记提供的zip压缩文件中的*samba_monitor.atsln*。
2. 创建一个与*device_config*文件夹中现有器件配置文件相似的器件特定配置文件。本例中为*device_config_saml22n18a.h*。器件定义完成后，将此新配置文件包含在*device_config\device_config.h*中。
   - 2.1. SAML22将使用以下配置：
     - 2.1.1. 通信接口——USB和UART，将`SAM_BA_INTERFACE`选为`SAM_BA_BOTH_INTERFACES`
     - 2.1.2. 默认为`APP_START_ADDRESS`（即0x00002000）
     - 2.1.3. 按照以下代码示例所示，将SAML22N18A支持添加到*device_config.h*中：

```
#elif defined (__SAML22N18A__)
#include "device_config_saml22n18a.h"
#define SAM_BA_INTERFACE   SAM_BA_BOTH_INTERFACES
#define APP_START_ADDRESS  DEFAULT_APP_START_ADDRESS
```

   - 2.2. 检查*device_config_saml22n18a.h*中的所有其他宏和嵌入函数，确保其与当前器件设置匹配。有关配置的详细信息，请参见具体器件的数据手册。

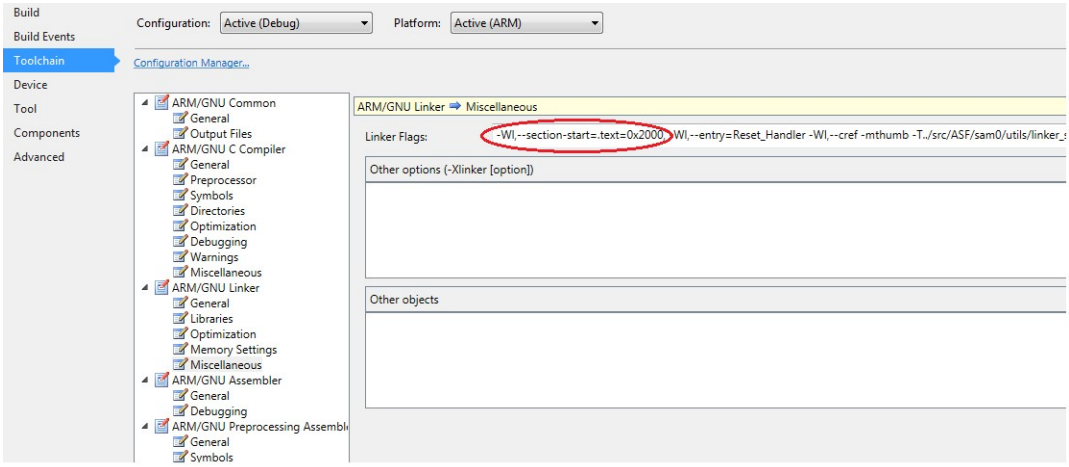
   注：本应用程序提供更新后的*device_config.h*和*device_config_saml22n18a.h*文件。
3. 通过选择*Project* > *Properties*（项目 > 属性）打开Properties（属性）部分，然后导航到：
   - 3.1. Device > *Change Device*（器件 > 更改器件）
     - 3.1.1. 选择**New Device**（新器件）。应为新器件更新*Device_Startup*文件夹。



4. 重新编译项目。输出文件夹中的全新十六进制和二进制文件随时可用。

5.2 使应用程序重新指向新的起始地址

1. 打开应用程序项目，使其重新指向samba_monitor中配置的新地址。本例中为0x00002000。
   1.1. 通过选择*Project* > *Properties*打开Properties部分，然后导航到*Toolchain* > *ARM/GNU Linker* > *Miscellaneous*（工具链 > ARM/GNU链接器 > 其他）。
      1.1.1. 添加`-Wl,--section-start=.text=0x2000,`。

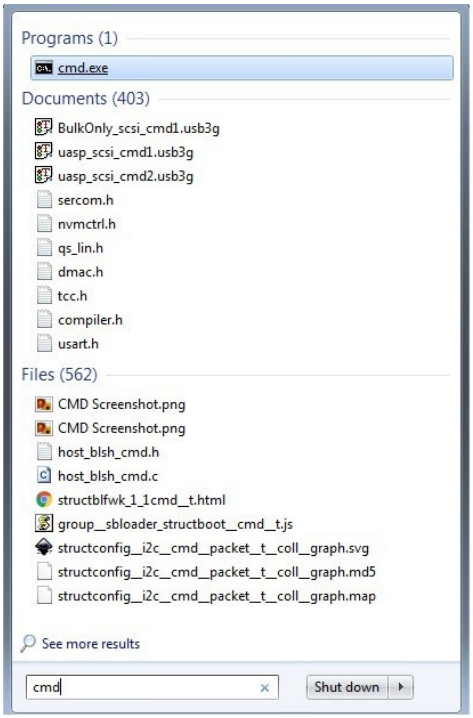


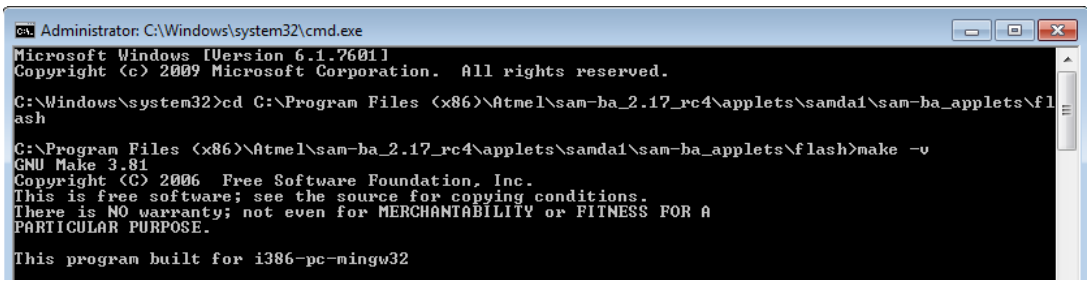
2. 对于使用相同USART或USB接口的应用程序，可能需要根据应用需求重新初始化。其他情况下，应用程序可继续使用SAM-BA监视器的配置。
3. 重新编译项目。Debug/Release（调试/发布）文件夹中的应用程序十六进制和二进制文件应随时可用。

5.3 为SAM-BA主机重新生成小程序

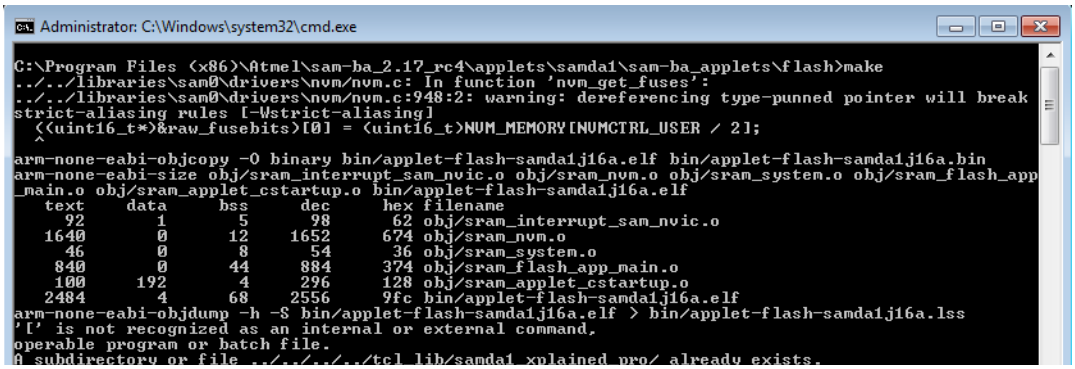
使用以下步骤更改应用程序的起始地址。有关如何为SAM-BA主机定制和重新生成小程序的完整细节，请参见http://www.microchip.com//wwwAppNotes/AppNotes.aspx?appnote=en591431。

1. 转到SAM-BA安装目录，该目录默认情况下位于*Program files (x86)/Atmel/sam-ba*。
2. 找到*applets\samda1\sam-ba_applets\flash\ flash_app_main.c*文件。
   2.1. 编辑该C文件中的MONITOR_SIZE宏以匹配定制SAM-BA监视器中的应用程序起始地址，然后保存文件。
   2.2. 以管理员权限打开CMD窗口。
      2.2.1. 在*search programs and files*（搜索程序和文件）中，输入**cmd**。
      2.2.2. 以管理员权限使用Ctrl+Shift+Enter而非Enter打开。
   2.3. 使用*cd*命令导航到*applets\samda1\sam-ba_applets\flash*文件夹。
   2.4. 在此文件夹中，输入*make –v*命令以显示make版本。如果未显示，请参见应用笔记中的**GNU Make 3.81**，可在以下网址下载：
   http://www.microchip.com//wwwAppNotes/AppNotes.aspx?appnote=en591431。





   2.5. 打开*cmd*窗口，运行*make*。



2.6. 更新后的二进制文件应在*bin*文件夹中生成。

2.6.1. 转到*tcl-lib\saml22_xplained_pro*文件夹，复制二进制文件。

2.6.2. 重启SAM-BA主机实用程序，使用更新后的二进制文件。

上述所有步骤均成功完成后，SAM-BA监视器和SAM-BA主机应可随时供SAM L22器件使用。

注：所有无ROM Cortex-M器件文件夹中均提供了更新后的小程序源代码、二进制文件和tcl文件。

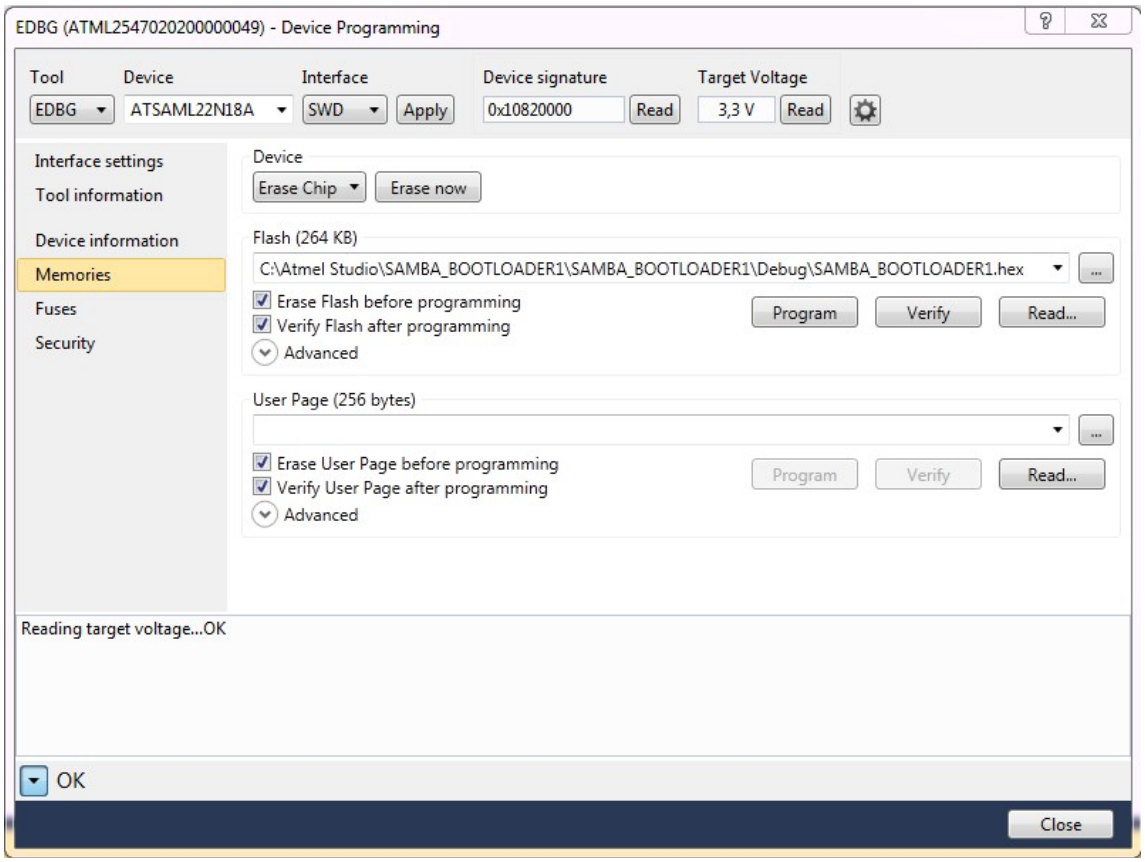
6. 使用SAM-BA监视器

6.1 编程SAM-BA监视器

可使用串行线调试（Serial Wire Debug，SWD）探头编程SAM-BA监视器：

1. 在Atmel Studio中选择*Tools* > *Device Programming*（工具 > 器件编程）。
2. 选择Tool（工具），然后单击**Apply**（应用）。
3. 从Memories（存储器）选项卡中，单击**Erase now**（立即擦除），将芯片擦除。
4. 在Flash（闪存）字段中指定SAM-BA监视器映像的路径，然后单击**Program**（编程）。

图6-1. 器件编程



5. 或者，也可以使用命令行编程工具*atprogram*编程十六进制文件。以下示例可用于编程SAM L22 Xplained板上的SAML22N18A器件。

```
atprogram -t edbg -i swd -d atsaml22n18a program -f
samd21_sam_ba_both_interfaces.hex
```

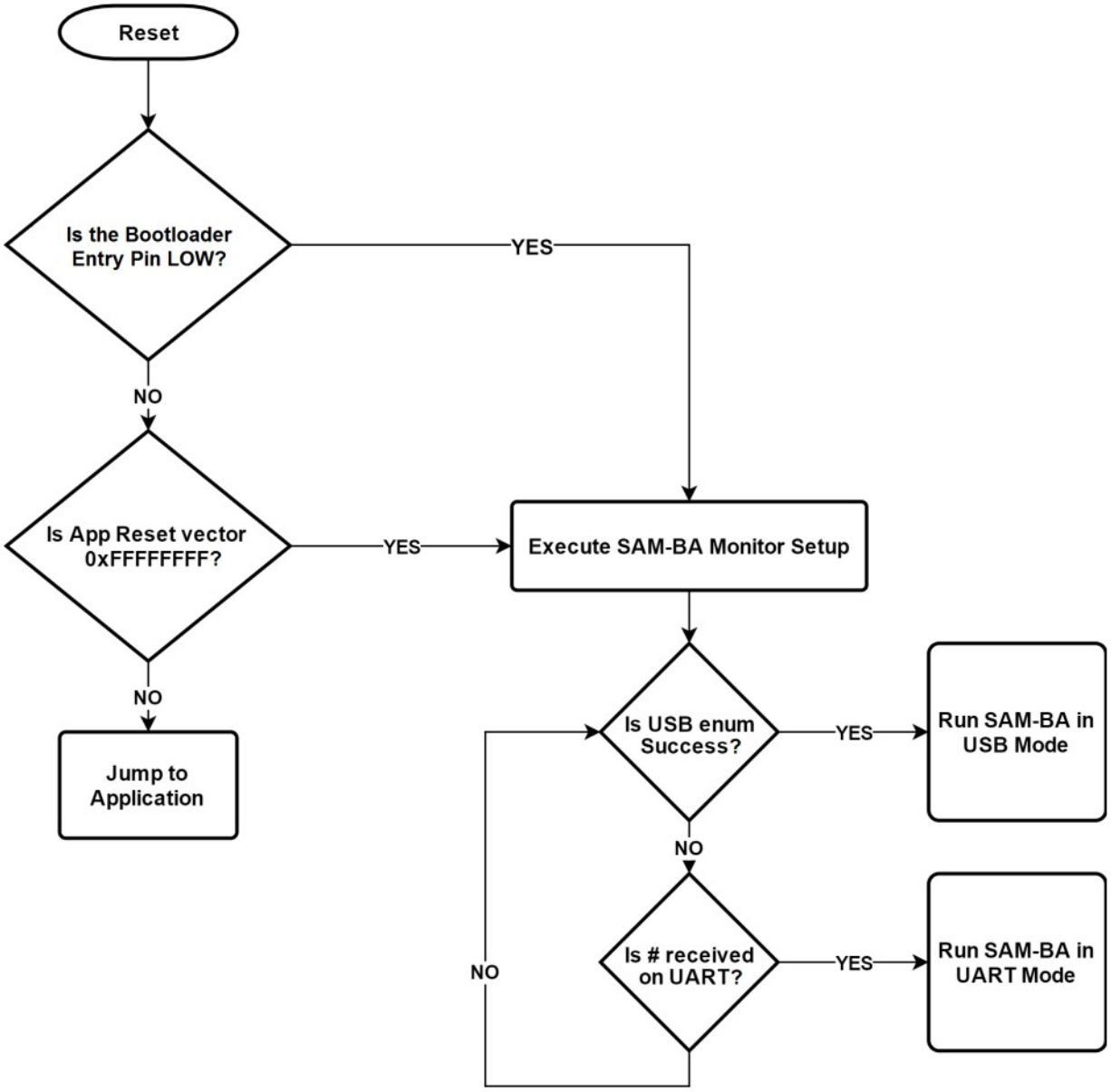
6.2 进入SAM-BA监视器

可以通过以下任何一种条件请求激活SAM-BA自举程序：

外部条件：当器件从复位条件释放时，将自举程序进入引脚拉低。通常的用法是使用可由用户访问的按钮作为SAM-BA监视器触发器。用户必须在器件上电或复位时按住按钮。

内部条件：在擦除器件后或应用程序复位向量（@APP_START_ADDRESS + 4）为空时（0xFFFFFFFF）。

图6-2. SAM-BA监视器引导过程



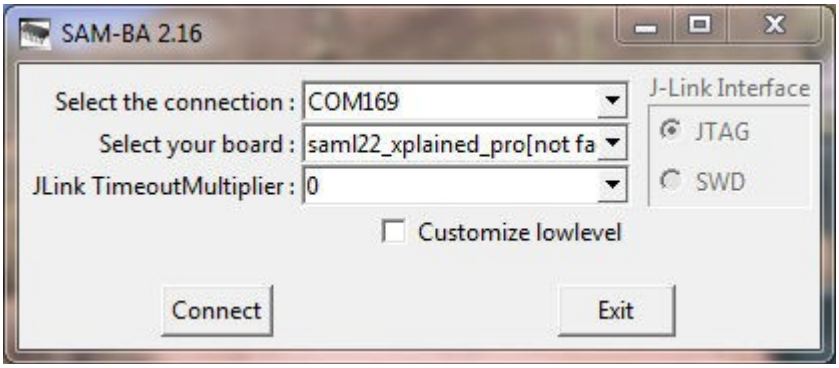
7. 在Windows®上运行SAM-BA

本章将介绍在Windows PC上运行SAM-BA GUI的基本步骤。有关完整细节，请参见SAM-BA用户指南（*<samba-install-directory>\doc\sam-ba user guide.pdf*）。

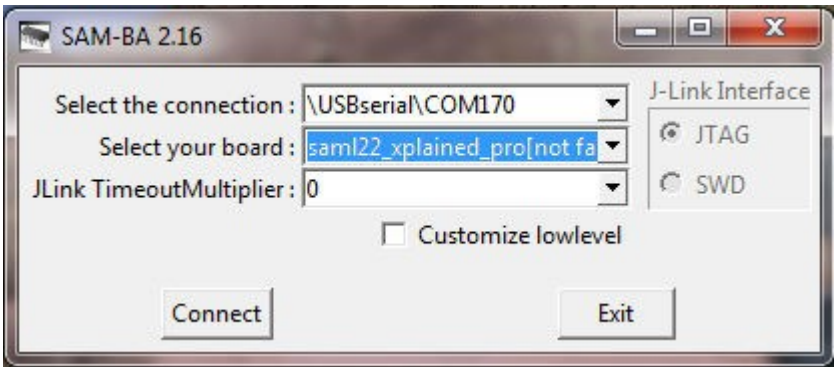
7.1 连接SAM-BA GUI

要通过UART接口使用SAM-BA监视器，需通过检测到的COM端口或选定的串行端口将硬件连接到PC。

注：对于SAM L22 Xplained Pro，UART接口使用*DEBUG USB*端口进行输出。



要通过USB接口使用SAM-BA监视器，需通过*TARGET USB*端口将硬件连接到PC。



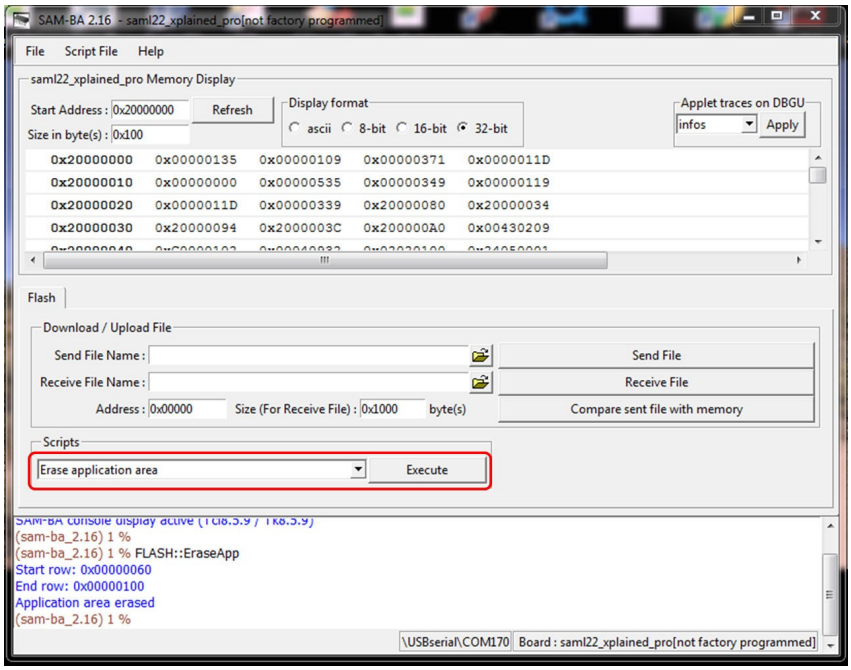
单击**Connect**（连接），与器件建立连接。

7.2 闪存编程

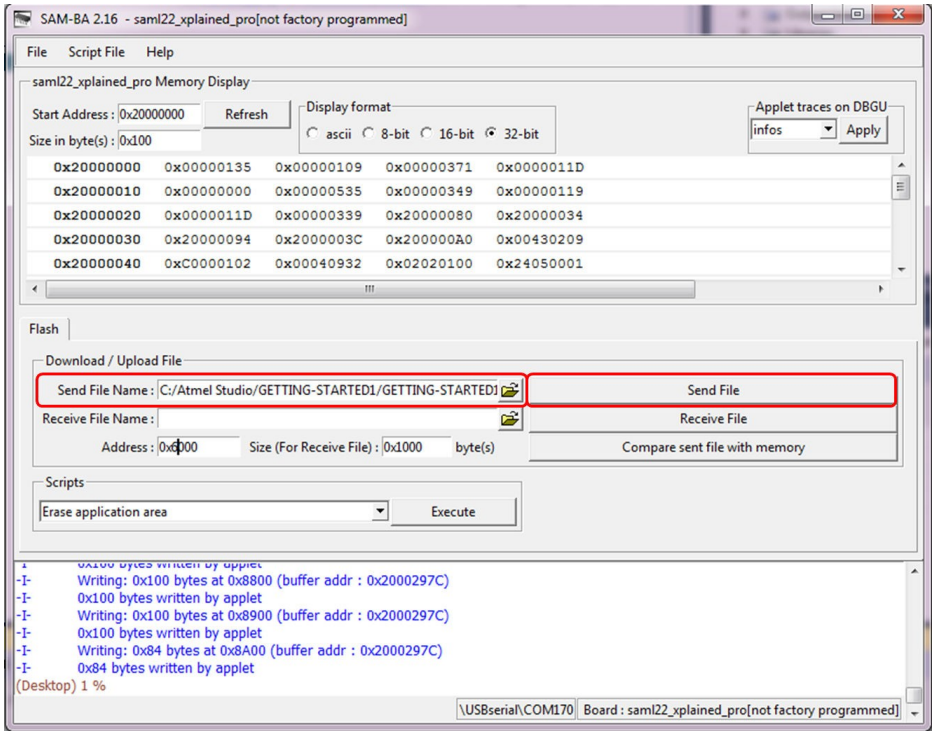
成功连接到器件后，将显示以下SAM-BA屏幕，该屏幕支持与器件进行各种交互。要升级应用程序，用户需擦除现有应用程序，然后下载或上传文件。使用Flash选项卡加载闪存的内容。将程序上传到闪存时，起始地址必须与SAM-BA监视器和小程序中的配置值相匹配（在SAM-BA监视器示例中为0x2000）；否则，传输过程将中止。

闪存编程

1. 选择*Erase application area*（擦除应用程序区域）脚本擦除应用程序，然后单击**Execute**（执行）。



2. 选择要下载到器件闪存的文件，然后将*Address*（地址）更改为**0x02000**。



3. 单击**Send File**（发送文件）。

7.3 脚本

SAM-BA主机提供以下预定义脚本。

表7-1. 预定义脚本

脚本名称	说明
置1安全位	置1安全位可保护器件安全。有关详细信息，请参见具体器件数据手册中的“NVMCTRL”一章。
读取安全位	读取当前的安全状态
擦除应用程序区域	擦除所有应用程序代码（不会擦除SAM-BA监视器区域）。
使应用程序失效	擦除应用程序的第一页。
读取熔丝	返回熔丝设置的值。有关详细信息，请参见具体器件数据手册中的“NVM用户行映射”部分。
读取锁定熔丝	读取当前锁定设置
读取器件ID	读取器件标识寄存器
设置锁定位[0:15]	设置指定的锁定位可防止擦除闪存区域。有关详细信息，请参见具体器件数据手册中的“NVMCTRL”一章。
全部解锁	解锁全部闪存区域。

8. 其他解决方案

SAM-BA监视器还提供了多种其他选项。下表列出了不同的选项和用例。

表8-1.

SAM-BA监视器选项	说明
基于ASF的SAM-BA监视器	SAM-BA监视器使用Atmel软件框架（Atmel Software Framework，ASF）进行开发，由于它负责为器件上的各种SERCOM或USB实例或变体移植驱动程序，因此可轻松移植到各种器件。 由于ASF的原因，其尺寸稍大一些。（采用ASF 3.32时为24 KB）。
基于ROM的SAM-BA监视器	在带有ROM的一些器件（例如Cortex-M3/M4/M7系列SAM器件）中，SAM-BA将加载到ROM中。它将基于复位时的GPNVM位设置进行激活。有关详细信息，请参见具体器件数据手册。
传统IAR项目	本应用笔记的zip文件夹中还包括以下传统项目： • Atmel-42238-UART-based-SAM-BA-Bootloader-for-SAM-D20_AP-Note_AT04189.zip • Atmel-42366-SAM-BA-Bootloader-for-SAM-D21_ApplicationNote_AT07175.zip

9. 参考资料

- Atmel-42438-SAM-BA-Overview-and-Customization-Process_ApplicationNote_AT09423.pdf
- Atmel-42728-Using-SAM-BA-for-Linux-on-SMART-ARM-based-Microcontrollers_ApplicationNotes_AT15004.pdf
- SAM-BA用户指南：*<samba_install_dir>\doc\sam-ba user guide.pdf*

10. 版本历史

表10-1. 版本A——11/2017

说明
本文档的初始版本。

Microchip网站

Microchip网站（http://www.microchip.com/）为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的互联网浏览器即可访问，网站提供以下信息：

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及归档软件
- **一般技术支持**——常见问题（FAQ）、技术支持请求、在线讨论组以及Microchip顾问计划成员名单
- **Microchip业务**——产品选型和订购指南、最新Microchip新闻稿、研讨会和活动安排表、Microchip销售办事处、代理商以及工厂代表列表

变更通知客户服务

Microchip的变更通知客户服务有助于客户了解Microchip产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时，收到电子邮件通知。

欲注册，请登录Microchip网站http://www.microchip.com/。在“支持”（Support）下，点击“变更通知客户”（Customer Change Notification）服务后按照注册说明完成注册。

客户支持

Microchip产品的用户可通过以下渠道获得帮助：

- 代理商或代表
- 当地销售办事处
- 应用工程师（FAE）
- 技术支持

客户应联系其代理商、代表或应用工程师（FAE）寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过以下网站获得技术支持：http://www.microchip.com/support

Microchip器件代码保护功能

请注意以下有关Microchip器件代码保护功能的要点：

- Microchip的产品均达到Microchip数据手册中所述的技术指标。
- Microchip确信：在正常使用的情况下，Microchip系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以Microchip数据手册中规定的操作规范来使用Microchip产品的。这样做的人极可能侵犯了知识产权。
- Microchip愿意与关心代码完整性的客户合作。

- Microchip或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展中。Microchip承诺将不断改进产品的代码保护功能。任何试图破坏Microchip代码保护功能的行为均可视为违反了《数字器件千年版权法案（Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

法律声明

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。Microchip对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。

Microchip对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将Microchip器件用于生命维持和/或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障Microchip免于承担法律责任，并加以赔偿。除非另外声明，否则在Microchip知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip的名称和徽标组合、Microchip徽标、AnyRate、AVR、AVR徽标、AVR Freaks、BeaconThings、BitCloud、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、Heldo、JukeBlox、KeeLoq、KeeLoq徽标、Kleer、LANCheck、LINK MD、maXStylus、maXTouch、MediaLB、megaAVR、MOST、MOST徽标、MPLAB、OptoLyzer、PIC、picoPower、PICSTART、PIC32徽标、Prochip Designer、QTouch、RightTouch、SAM-BA、SpyNIC、SST、SST徽标、SuperFlash、tinyAVR、UNI/O和XMEGA是Microchip Technology Incorporated在美国和其他国家或地区的注册商标。

ClockWorks、The Embedded Control Solutions Company、EtherSynch、Hyper Speed Control、HyperLight Load、IntelliMOS、mTouch、Precision Edge和Quiet-Wire为Microchip Technology Incorporated在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、BodyCom、chipKIT、chipKIT徽标、CodeGuard、CryptoAuthentication、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、EtherGREEN、In-Circuit Serial Programming、ICSP、Inter-Chip Connectivity、JitterBlocker、KleerNet、KleerNet徽标、Mindi、MiWi、motorBench、MPASM、MPF、MPLAB Certified徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、PureSilicon、QMatrix、RightTouch徽标、REAL ICE、Ripple Blocker、SAM-ICE、Serial Quad I/O、SMART-I.S.、SQI、SuperSwitcher、SuperSwitcher II、Total Endurance、TSHARC、USBCheck、VariSense、ViewSpan、WiperLock、Wireless DNA和ZENA为Microchip Technology Incorporated在美国和其他国家或地区的商标。

SQTP为Microchip Technology Inc.在美国的服务标记。

Silicon Storage Technology为Microchip Technology Inc.在除美国外的国家或地区的注册商标。

GestIC是Microchip Technology Inc.的子公司Microchip Technology Germany II GmbH & Co. KG在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2018，Microchip Technology Incorporated，版权所有。

ISBN：978-1-5224-3141-1

DNV认证的质量管理体系

ISO/TS 16949

Microchip位于美国亚利桑那州Chandler和Tempe与位于俄勒冈州Gresham的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了ISO/TS-16949:2009认证。Microchip的PIC® MCU和dsPIC® DSC、KEELOQ®跳码器件、串行EEPROM、单片机外设、非易失性存储器和模拟产品严格遵守公司的质量体系流程。此外，Microchip在开发系统的设计和生产方面的质量体系也已通过了ISO 9001:2000认证。

全球销售及服务网点

美洲

公司总部 Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277
技术支持：
http://www.microchip.com/support
网址：www.microchip.com

亚特兰大 Atlanta
Duluth, GA
Tel: 1-678-957-9614
Fax: 1-678-957-1455

奥斯汀 Austin, TX
Tel: 1-512-257-3370

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Novi, MI
Tel: 1-248-848-4000

休斯敦 Houston, TX
Tel: 1-281-894-5983

印第安纳波利斯 Indianapolis
Noblesville, IN
Tel: 1-317-773-8323
Fax: 1-317-773-5453
Tel: 1-317-536-2380

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608
Tel: 1-951-273-7800

罗利 Raleigh, NC
Tel: 1-919-844-7510

纽约 New York, NY
Tel: 1-631-435-6000

圣何塞 San Jose, CA
Tel: 1-408-735-9110
Tel: 1-408-436-4270

加拿大多伦多 Toronto
Tel: 1-905-695-1980
Fax: 1-905-695-2078

亚太地区

中国 - 北京
Tel: 86-10-8569-7000

中国 - 成都
Tel: 86-28-8665-5511

中国 - 重庆
Tel: 86-23-8980-9588

中国 - 东莞
Tel: 86-769-8702-9880

中国 - 广州
Tel: 86-20-8755-8029

中国 - 杭州
Tel: 86-571-8792-8115

中国 - 南京
Tel: 86-25-8473-2460

中国 - 青岛
Tel: 86-532-8502-7355

中国 - 上海
Tel: 86-21-3326-8000

中国 - 沈阳
Tel: 86-24-2334-2829

中国 - 深圳
Tel: 86-755-8864-2200

中国 - 苏州
Tel: 86-186-6233-1526

中国 - 武汉
Tel: 86-27-5980-5300

中国 - 西安
Tel: 86-29-8833-7252

中国 - 厦门
Tel: 86-592-238-8138

中国 - 香港特别行政区
Tel: 852-2943-5100

中国 - 珠海
Tel: 86-756-321-0040

台湾地区 - 高雄
Tel: 886-7-213-7830

台湾地区 - 台北
Tel: 886-2-2508-8600

台湾地区 - 新竹
Tel: 886-3-577-8366

亚太地区

澳大利亚 Australia - Sydney
Tel: 61-2-9868-6733

印度 India - Bangalore
Tel: 91-80-3090-4444

印度 India - New Delhi
Tel: 91-11-4160-8631

印度 India - Pune
Tel: 91-20-4121-0141

日本 Japan - Osaka
Tel: 81-6-6152-7160

日本 Japan - Tokyo
Tel: 81-3-6880-3770

韩国 Korea - Daegu
Tel: 82-53-744-4301

韩国 Korea - Seoul
Tel: 82-2-554-7200

马来西亚 Malaysia - Kuala Lumpur
Tel: 60-3-7651-7906

马来西亚 Malaysia - Penang
Tel: 60-4-227-8870

菲律宾 Philippines - Manila
Tel: 63-2-634-9065

新加坡 Singapore
Tel: 65-6334-8870

泰国 Thailand - Bangkok
Tel: 66-2-694-1351

越南 Vietnam - Ho Chi Minh
Tel: 84-28-5448-2100

欧洲

奥地利 Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark - Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

芬兰 Finland - Espoo
Tel: 358-9-4520-820

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Garching
Tel: 49-8931-9700

德国 Germany - Haan
Tel: 49-2129-3766400

德国 Germany - Heilbronn
Tel: 49-7131-67-3636

德国 Germany - Karlsruhe
Tel: 49-721-625370

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

德国 Germany - Rosenheim
Tel: 49-8031-354-560

以色列 Israel - Ra'anana
Tel: 972-9-744-7705

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

意大利 Italy - Padova
Tel: 39-049-7625286

荷兰 Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

挪威 Norway - Trondheim
Tel: 47-7289-7561

波兰 Poland - Warsaw
Tel: 48-22-3325737

罗马尼亚 Romania - Bucharest
Tel: 40-21-407-87-50

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

瑞典 Sweden - Gothenberg
Tel: 46-31-704-60-40

瑞典 Sweden - Stockholm
Tel: 46-8-5090-4654

英国 UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820

© 2018 Microchip Technology Inc.