

讲稿: StellarisWare 软件入门介绍

Introducing StellarisWare

大家好，欢迎收看德州仪器 StellarisWare 入门介绍

幻灯片 1:

今天我们将向您介绍德州仪器 StellarisWare 软件，让您更好地了解其应用程序。StellarisWare 是一种需要与我们 Stellaris MCU 配合使用的软件。它包括我们 Stellaris 团队自己创建的软件，或经移植与 Stellaris MCU 配合使用的第三方软件。StellarisWare 软件可分为 5 类功能程序库：

- 外设驱动库
- 图形库
- USB 程序库
- IEC 60730 文件库
- IQMath

利用已预载在 MCU ROM 上的 StellarisWare，可以轻而易举地访问常用函数，为您的设计尽力提供一个稳健的解决方案。

幻灯片 2:

StellarisWare 主要有三种的使用方法：

- 1) 开箱即用；MCU 到货开包后即可立即安装使用
- 2) 利用我们所提供的大量文档详细了解 Stellaris MCU，启动您的产品开发。
- 3) 利用 我们提供的免专利费的 TI 技术，快速进军市场

我们特有的芯片和 StellarisWare 软件并行开发的模式实践了我们尽心助您成功的一贯承诺！

幻灯片 3:

欲获得 StellarisWare，您可以在 ti.com/stellarisware 网页上免费下载，或通过器件附带的 CD 直接安装。鉴于供应商的工具更新频繁，Stellaris 团队会及时更新重建 StellarisWare 工程，向客户推出最新的软件，从而确保以供应商产品的一致，为您提供最佳的用户体验机会。

幻灯片 4:

StellarisWare 外设驱动库 是 此套软件的核心部分，提供开发中要使用的源代码和库文件，帮助您加快开发进度。StellarisWare 的使用已经得到 TI 许可证的授权许可。您可以原样使用我们的代码，或按您自己的需要对其进行修改。然而，第三方工具的许可条件有所不同，对此类问题我们可以提供支持。

幻灯片 5:

这里一个简单的代码范例。它的代码编写清晰易懂，函数名称意思明了。StellarisWare 的编码以匈牙利命名法 (Hungarian-Notation) 为标准。

这里有 2 个函数： Main (主函数) 和 UART interrupt handler 中断处理函数。 本例中的所用到的函数调用指令都取自于 DriverLib 驱动库所提供的函数。

在主函数中，

- 首先利用函数 SysCtlClockSet 设置系统时钟。
- 然后使能 UART 和 GPIOA。
- 用函数 IntMasterEnable 使能处理器中断功能。
- 设置 GPIOA，引脚 0 和 1 作为 UART 引脚使用。
- 用函数 UARTConfigSet 对 UART 进行配置，以便通过 8 个数据位（无奇偶校验）和 1 个停止位,使达到 115,200 波特率。
- 使能 UART 端口 0，使能在中断控制器层面产生中断，并且在收到字符或发生接收超时(receive timeout)的时候产生 UART0 中断。
- 最后，当数据通过 UART 实现回送后,即进入无限循环状态。
-

当 UART 接收到数据时，系统会执行 UART 中断处理程序。该程序负责将数据回送到 UART。 首先中断处理程序转换为中断状态，然后清除产生的中断。 While loop 确保字符收到，并将其回传到 UART。

使用驱动库以及其文档可以使编程变得轻松简单，我们不再需要了解 UART 模块或中断控制器的低级寄存器细节。

注意： 如要使用 _MAP，需要用#include 包含指令加载 StellarisWare inc 文件夹中的 rom_map.h 头文件。 如果 ROM 中有可用的宏(macros)，这些宏将从 ROM 进行函数调用。 否则将从闪存中寻找执行指令。

幻灯片 6:

USB 库是 StellarisWare 中比较复杂的一个程序库。此库为 USB 主控制器(Host), USB 接口(device)和 移动 USB OTG 应用程序提供支持。 StellarisWare 提供一套完整的各种操作模式下的 USB 代码示例。其中有很多示例都可以原样照搬(as-is) 地集成在终端应用程序中，例如 USB 接口 Firmware 工具软件的升级，USB 主控制器和接口以及 CDC 类上的批量存储(mass storage) 驱动。 如想修改这些示例，可借用 StellarisWare 中的源代码。 TI MCU 上都免费提供 USB 文件库。 TI 还对 USB 供应商和产品 ID 进行了再许可。这对拥有少量产品或以发展为目的的客户来说非常有益。 USB 库文件和源代码的组成类似于 StellarisWare 中的其他文件库。

幻灯片 7:

图形库（Graphic Library）可以帮助开发图形应用。 它在 2 种不同功能层面(layers)上提供支持。

- 1. 初始图形（primitives layer）
- 2. 插件（widget layer）

但图形库本身不是一个图形应用程序。但如果借用相应驱动程序，它支持多个显示器和显示控制器。鉴于图形库中还拥有低级和高级功能层面（layers of function），因此，您可以选择在最适合您需要的层面上进行编程。

幻灯片 8:

我们提供经过 IEC 60730 认证的 MCU，这种 MCU 最适用于洗衣机/烘干机、冰箱、冷冻柜和炉具等家用电器。StellarisWare extensions 为 IEC 60730 级安全要求提供支持。

幻灯片 9:

我们有两类代码示例：

- 适用于特定外设的代码示例：例如 ADC，UART 等
- 适用于特定 IDE 的代码示例，如这里的屏幕截图所示

幻灯片 10:

SimpleLink 是一个自包含独立解决方案。通过最大程度减少无线连通所需的 RF 专业知识量，来简化无线技术的开发和认证。TI SimpleLink 产品系列不断扩大，为多种无线技术（包括 Wi-Fi®、ZigBee®、和 Bluetooth）提供解决方案。

幻灯片 11:

Boot Loader(启动加载程序)与 Serial Flash Loader(串行闪存加载程序)不同。

可用于现场更新的 BootLoader 支持多个程序现场更新。对于配有 ROM 的器件，BootLoader 已预载在 ROM 中。我们有意将 Boot Loaders 长期储存在闪存中，或者用户也可自行将 BootLoader 烧写入闪存中。

在那些没有 ROM 的器件上，Flash loader 在出厂前被烧写到闪存中，并非是我们特意将其存储在闪存中。通常，在器件生产过程中利用 Flash loader 来完成的一次性零件程序烧写，无需利用 JTAG。

幻灯片 13:

- LM Flash Programmer 是一种闪存编程 utility

- 支持串行、JTAG、以太网、USB 等多种接口上的 bootloader
- 与 FTDI ICDI 和基于 Stellaris 的 ICDI 兼容
- 在 GUI 和命令行模式下均可用
- 允许在生产流程中编写脚本 (SCRIPT)

如果您使用的是基于 FTDI 或 Stellaris 的 FTDI 解决方案，如想编写 StellarisMCU 的闪存程序，请参阅 Application Note 三 “how to program flash”（如何编写闪存程序）

幻灯片 14:

我们提供了一款名为 LM Flash Programmer (LMFP)的 GUI 闪存烧写器。这是一种简易 Windows 应用程序，可通过评估和开发板上的 ICDI 来烧写 MCU 的闪存程序。您可以使用此应用程序来编写、清除和上传闪存中的信息。还有其他 utilities 可用于一次性大量清除器件上的信息，更新上面非易失性寄存器里存储的信息

这里，您可以了解到如何从硬件角度使用 LMFP。多数 Stellaris 评估板均具有内置 ICDI。在第一个示例中，您可以了解到如何直接使用 LMFP 对 Stellaris 评估板上的 StellarisMCU 进行编程。

在下方第二个示例中，您可以使用评估板的外部调试(debug-out) 功能来编写外部目标程序。这里，您可以看到 EK-LM3S811 套件用于与 20 引脚至 10 引脚 JTAG 适配器的连接，以便在其中一个 reference design (参考设计)板上编写 MCU 程序。在制造和测试过程中，这种方法特别适用于编写终端应用程序。

幻灯片 15:

ARM 生态系统的优势是支持 ARM 产品的公司与 TI 基于 ARM 的产品兼容。我们和很多公司建立了合作伙伴关系。他们的经营项目弥补了芯片供应商通常不提供的工具，如生产编程器，从而服务于客户。所有这些公司均使用 TI ARM 器件。

幻灯片 16:

- 这些是我们合作伙伴经营的主要 IDE。
- 我们提供与上述每个 IDE 配套的软件套件。
- 那么哪种 IDE 最好呢？
- 这要视情况而定。不同的 IDE，他们在功能、成本和优势方面都不尽相同。
- 如果您在 IDE 选择上有问题，请告知您所在地的 TI 或代理商的 FAE。

幻灯片 17:

这是几种适用于 Stellaris MCU 的实时操作系统(RTOS)。每个 RTOS 的功能和成本各不相同。OPEN-RTOS、SAFE-RTOS 和 FREE-RTOS 是其中几种适用于 Stellaris MCU 的常用 RTOS。

FreeRTOS 是针对嵌入式设备的实时操作系统，可移植到多个微控制器上。FreeRTOS 根据 GPL (General Public License)授权进行分配，但也存在一种特例。这种特例使得用户的专用代码在维持内核本身作为开源的同时维持闭源，从而协助 FreeRTOS 在专用应用程序中的使用。

FreeRTOS 设计小巧且易于操作。内核 (kernel)本身仅由三个或四个 C 文件构成。多数情况下使用 C 语言编程，使代码可读性强、易于移植和维护，但有些汇编函数在需要时会包含一些汇编语言函数（多数在架构特定的调度程序进程 (scheduler routines)中）。来源：<http://en.wikipedia.org/wiki/FreeRTOS>

幻灯片 18:

LM4F 评估套件是我们 M4F MCU 最新具有浮点功能的套件。它具有许多强大的功能，例如：

OLED 显示屏，休眠功能，传感器：加速度传感器 (Accelerometer)、温度传感器)，USB OTG/设备/主机、SD 卡等

除了这些功能，它还拥有一个针对 Windows 的数据记录器(datalogger)，用来将 MCU 中的数据记录到 csv 文件中。它可以帮助用户在 10 分钟或更短时间内快速入门

幻灯片 19:

如果您需要更多信息，可以参考我们此次培训的有关资源，也可以随时于我们 E2E 论坛上互动提问。谢谢您的观看。再见。

Introducing StellarisWare

Slide 1:

This PTM will introduce you to StellarisWare and provide you with a better understanding of its applications. StellarisWare is software that has been made to be used with Stellaris MCUs. It includes software that the Stellaris team has created, or third party software that has been ported to use with Stellaris MCUs. There are 5 functional areas:

Stellaris Peripheral Driver Library

Stellaris Graphics Library

Stellaris USB Library

Stellaris IEC 60730 Library

IQMath

StellarisWare has been programmed into ROM on Stellaris MCUs to provide easier access to common function calls and to be as robust of a solution as possible.

Slide 2:

StellarisWare can be used in 3 key ways:

- 1) Straight out of the box; it is ready to be used for development when you receive your MCU
- 2) To learn more about the Stellaris MCUs, so you can begin developing your product, we have provided plenty of documentation to ease use.
- 3) It can help you be fast to market, because we provide it to you royalty free.

We develop StellarisWare along with silicon. This parallel development shows our commitment in helping you succeed!

Slide 3:

You can download StellarisWare for free at ti.com/stellarisware, or install from a CD included with your device. We will provide frequent updates, when necessary. Tool vendors frequently update tool features, in order to keep up with this the Stellaris team will rebuild the projects and push out the updates to the customers, this is done with an intention to provide a fantastic user experience.

Slide 4:

DriverLibrary is a key core component of StellarisWare, which provides source code and library files for use in your development and to speed up your development time using StellarisWare. Use of StellarisWare is governed by TI license. You can use the code as is, or modify it to meet your needs. However our third party tools have different licensing conditions, we can help provide support for these types of questions.

Slide 5:

Here is a really simple example to demonstrate the power of StellarisWare

Note that the code is readable, the function names are self descriptive, and we follow a coding standard, which is Hungarian notation.

As you can see, this slide shows some example code using the DriverLib peripheral drivers. This particular example utilizes the UART to echo text. All characters received on the UART are transmitted back to the UART. A PC terminal application such as HyperTerminal or TeraTerm could be used to interact with the evaluation board. The complete UART echo example is provided with DriverLib.

You can see 2 functions here: main and the UART interrupt handler. All of the function calls in this example are calls to functions provided within DriverLib.

In main, first the system clock is configured by making a call to System Control Clock Set (SysCtlClockSet). Then both the UART and GPIO A modules are enabled. Interrupts to the processor are enable by calling Interrupt Master Enable (IntMasterEnable). GPIO A pins 0 and 1 are configured to function as UART pins. Then a function call is made to configure the UART for a 115,200 baud rate using 8bits of data, no parity, and 1 stop bit. Two function calls are made to enable the UART0 interrupt at the interrupt controller level, and to enable UART0 to generate an interrupt when a character is received or when a receive timeout occurs. Finally, an infinite loop is entered while data is echoed through the UART.

The UART interrupt handler is executed when data is received by the UART and is responsible for echoing that data back out the UART. First the interrupt handler gets the interrupt status and then clears the asserted interrupts. The while loop gets characters received and transmits them back out the UART.

Using DriverLib along with its documentation makes writing such an example easy as there is no need to understand the low-level register details of the UART module or the interrupt controller.

Note: To use _MAP, please include rom_map.h from the inc folder in StellarisWare. Macros to facilitate calling functions in the ROM when they are available and in flash otherwise.

Please refer to MAP example in StellarisWare at C:\StellarisWare\examples\peripherals\rom

Slide 6:

One of the more complex library provided in the StellarisWare is the USB library. This library provides support for USB host, USB device and USB on-the-go applications. There is a full set of examples for USB in various modes of operations provided in the StellarisWare. Many of these examples can be integrated in the end applications 'as-is' such a USB device firmware upgrade, mass storage for host & device and CDC classes. If you modify these examples, the source I is provided in the StellarisWare and it can be used as a starting point for more complex applications. This library is provided royalty free with TI MCUs. TI also sub-licenses vendor & product IDs for the USB. This can be useful for customers with small number of products or for developmental purposes. The files and source in USB library are organized similarly as other libraries in StellarisWare.

Slide 7:

- Graphic Library helps developing graphics apps. It provides functionality in form of 2 layers
- 1. primitives
- 2. widgets

However it is not a graphics app. But it can be used for multiple displays & display controller if you have the driver for them. Graphics Library also has low level & high level platforms, so you can work at the level that best meets your needs.

Slide 8:

We provide MCUs that are IEC 60730 certified, this will work best in household appliances such as washers/dryers, refrigerators, freezers, and cookers/stoves. The StellarisWare extension provides support for IEC 60730 Class B Safety requirements.

Slide 9:

We have two types of examples:

- Peripheral specific: like ADC UARTs etc
- Kit specific, which is shown in the screen shot

Slide 10:

SimpleLink self-contained solutions were designed to simplify wireless development and certification by minimizing the amount of RF expertise you need to wirelessly enable a wide range of applications. TI offers SimpleLink solutions for multiple wireless technologies including Wi-Fi®, ZigBee®, 6LoWPAN, and ANT™, with an expanding portfolio to come.

Slide 11:

A Boot Loader is not the same as a Serial Flash Loader.

The BootLoader is used as a field update, supports multiple field updates and is also available in ROM, for those devices that have ROM. The Boot Loaders is meant to reside long term in flash or the user can program BootLoader in flash.

The Flash loader is programmed from the factory in flash for devices that do not have ROM and it is not meant to reside in flash. Typically it is used to program parts during production in a way that provides one time programming ability without JTAG.

Slide 12:

The Stellaris LM3S9B96 MCU has an additional feature integrated in the ROM. The high-integrated RTOS, SAFERTOS from Wittenstetien is included in the ROM. This can be used as a standard operating system, or as a part of high integrity application that requires certifications like IEC61508 or FDA510. This is a 65k value free within the LM3S9B96 i.e. price for SAFERTOS is included in the price of the LM3S9B96. You won't need to buy an additional license for SAFERTOS if you purchase LM3S9B96. This can significantly reduce the time to market and development for industrial and medical applications. Innovative *Design Assurance Pack* available separately from WITTENSTEIN provides complete turnkey evidence and process documentation. Currently, SAFERTOS comes in ROM of just one device. Will plan to include in RTOS in the ROM of more devices in the future.

Slide 13:

- LM Flash Programmer is a:

- Flash prog. utility
- Supports BL over multiple interfaces: serial, jtag, ethernet, USB
- Is compatible with FTDI ICDI & Stellaris based ICDI
- Available in both GUI & command line mode
- Allows writing script in the production flow

In order to program the flash of StellarisMCU if you are using a FTDI/Stellaris based FTDI solution, please see application note of "how to program flash".

Slide 14:

The Stellaris team provides a Windows GUI flash programmer called LM Flash Programmer. This is a simple windows application which is used to program the flash of the MCU using the in-circuit debug interface (ICDI) built on the evaluation and development boards. You can use this application to program, erase and upload the contents of the flash memory. There are other utilities available for mass erasing the device and updating the contents of non-volatile registers in the device. LMFP supports JTAG and SWD.

Here you can see how to use LMFP from the hardware perspective. Most of the Stellaris evaluation boards have a ICDI built into them. In the first example, you can see how you can directly LMFP program the Stellaris MCU present on a Stellaris evaluation board using LMFP.

In the second example, you can use the debug-out feature of the evaluation board to program the external target. Here you can see that an EK-LM3S811 kit is used to connected to a 20 pin to 10 pin JTAG adapter to program the MCU on one of our reference design board. This can be particularly useful in programming end-application during manufacturing and testing processes.

Slide 15:

The benefit of ARM ecosystem is that companies that support ARM products are compatible with TI ARM products. We have partnered with a number of companies who are into the business of enabling customers by providing tools that silicon vendors do not traditionally provide, for example: production programmers. All of these companies work with TI ARM devices.

Slide 16:

- These are the major tool that we partner with.
- We have kit with each one
- Which tool is the best?
- It depends. Different tools have different features, different costs and different benefits.
- If you need recommendation/advice is selecting a tool that best meets your needs, please let a local TI/disty FAE know.
- ICDI works with all tool chains
- Some tools are based on eclipse.

Ex: Code red (it supports SWD, trace support)

CCS benefits

- supports all TI products (like DSP, MCUs etc.)
 - can use full version on our kit
 - if you want full version for development on your board, then you have to buy a license

CS: inexpensive, if you like eclipse you can use it.

Slide 17:

There are several RTOSes available for Stellaris MCUs that users may find useful. Each RTOS offers different number & types of feature and cost. OPEN-RTOS, SAFE-RTOS and FREE-RTOS are some of the popular RTOSes for Stellaris MCUS.

FreeRTOS is a real-time operating system for embedded devices, being ported to several microcontrollers. It is distributed under the GPL with an optional exception. The exception permits users' proprietary code to remain closed source while maintaining the kernel itself as open source, thereby facilitating the use of FreeRTOS in proprietary applications.

FreeRTOS is designed to be small and simple. The kernel itself consists of only three or four C files. To make the code readable, easy to port, and maintainable, it is written mostly in C, but there are a few assembly functions included where needed (mostly in architecture specific scheduler routines). Source:

<http://en.wikipedia.org/wiki/FreeRTOS>

Slide 18:

The LM4F Evaluation Kit is our latest kit with floating point capabilities with M4F MCUs. It has many great features, such as:

- OLED display
- Hibernate feature
- Sensors (Acc, temp)
- USB OTG/device/host, SD card etc.

Along with these features it also has a datalogger quick start application for Windows that lets you log the data from the MCU to a csv file. Which helps a user can get started in 10 minutes or less

Slide 19:

Thank you for listening to this PTM if you need any information we have provided resources here, and you can always ask questions on our E2E forums.