
利用 MPLAB® Harmony v3 在 Cortex®-M7 MCU 上实现 QSPI 的就地执行 (XIP) 功能

简介

并行数据通信接口过去一直凭借着高效的性能而备受青睐，但由于引脚数量有所增加，往往会导致设计变得较为复杂。并行接口外设的出现对于解决上述问题起到了至关重要的作用，而四通道串行外设接口 (Quad Serial Peripheral Interface, QSPI) 便是其中之一。

本文档将以基于 Arm®Cortex®-M7 的 MCU (SAM E70) 为例介绍 QSPI 的就地执行 (Execute-In-Place, XIP) 功能，并讨论如何使用 MPLAB® Harmony v3 软件框架来实现应用程序。文中将具体说明如何生成要在 QSPI 存储区域中执行的应用程序二进制文件，以及如何从 QSPI 区域中执行应用程序。

目录

简介.....	1
1. 硬件和软件要求.....	3
1.1. SAM E70 Xplained Ultra 评估工具包.....	3
1.2. MPLAB X 集成开发环境 (Integrated Development Environment, IDE) 和 XC 编译器.....	3
1.3. MPLAB Harmony v3.....	3
2. QSPI 简介.....	4
2.1. QSPI 串行存储器模式.....	4
2.2. 指令帧.....	6
3. 就地执行.....	8
3.1. 连续读模式.....	8
4. QSPI 的 MPU 配置.....	9
5. 定制链接描述文件.....	10
6. 使用 MPLAB Harmony v3 的 QSPI XIP 示例.....	11
6.1. QSPI XIP 主 MPLAB Harmony v3 应用程序.....	11
6.2. QSPI 镜像 MPLAB Harmony v3 应用程序.....	13
6.3. 十六进制镜像生成 Python 应用程序.....	14
7. 性能.....	16
8. 结论.....	17
9. 参考资料.....	18
Microchip 网站.....	19
产品变更通知服务.....	19
客户支持.....	19
Microchip 器件代码保护功能.....	19
法律声明.....	19
商标.....	20
质量管理体系.....	21
全球销售及服务网点.....	22

1. 硬件和软件要求

1.1 SAM E70 Xplained Ultra 评估工具包

SAM E70 Xplained Ultra 评估工具包是一款用于评估 SAME70 单片机（MCU）的开发工具包。SAM E70 基于 Cortex-M7，能够以 300 MHz 运行。该评估工具包包括一个板上嵌入式调试器，无需借助外部工具即可对 SAME70 进行编程或调试。该评估工具包还提供外部连接器以扩展电路板的功能，并简化定制设计的开发过程。

SAM E70 Xplained Ultra 评估工具包可从 [Microchip 直销网站](#) 获取。

1.2 MPLAB X 集成开发环境（Integrated Development Environment, IDE）和 XC 编译器

MPLAB X IDE 是一款可扩展且高度可配置的软件程序，包含多种功能强大的工具，可帮助用户发现、配置、开发、调试和验证大多数 Microchip 单片机的嵌入式设计。

MPLAB X IDE 可从 [Microchip 网站](#) 获取。本文档使用 MPLAB X IDE 版本 5.30。

MPLAB XC 编译器可从 [Microchip 网站](#) 获取。本文档使用 MPLAB XC32 版本 2.30。

1.3 MPLAB Harmony v3

MPLAB Harmony v3 是一款完全集成的嵌入式软件开发框架，能够提供灵活且可互操作的软件模块，可使用户将资源专注于创建 32 位 PIC[®] 和 SAM 单片机应用程序的开发，而无需处理器件详细信息、复杂协议和库集成等挑战。

它包括 MPLAB Harmony 配置器（MPLAB Harmony Configurator, MHC），这是一种易于使用的开发工具，带有图形用户界面（Graphical User Interface, GUI），可简化器件设置、库选择、配置和应用程序开发。MHC 可作为插件直接与 MPLAB X IDE 集成，它具有单独的 Java 可执行文件，可独立用于其他开发环境。

本文档中的示例使用以下 MPLAB Harmony v3 资源库，可以从 GitHub 下载：

- [CSP](#)（芯片支持包）
- [DEV_PACKS](#)（MPLAB Harmony v3 产品数据库）
- [MHC](#)（MPLAB Harmony v3 配置器）

或者

使用 [MPLAB Harmony v3 内容管理器](#) 下载资源库。

2. QSPI 简介

四通道 SPI 接口（QSPI）是一种同步串行接口，用于与外部器件或存储器进行通信。QSPI 具有四条数据线，除此之外与串并接口（Serial Parallel Interface, SPI）协议基本相同。由于数据通过多条线路发送，因此与标准 SPI 协议相比，带宽和性能都有所增加。

QSPI 支持单、双或四 I/O，具体取决于所选模式。

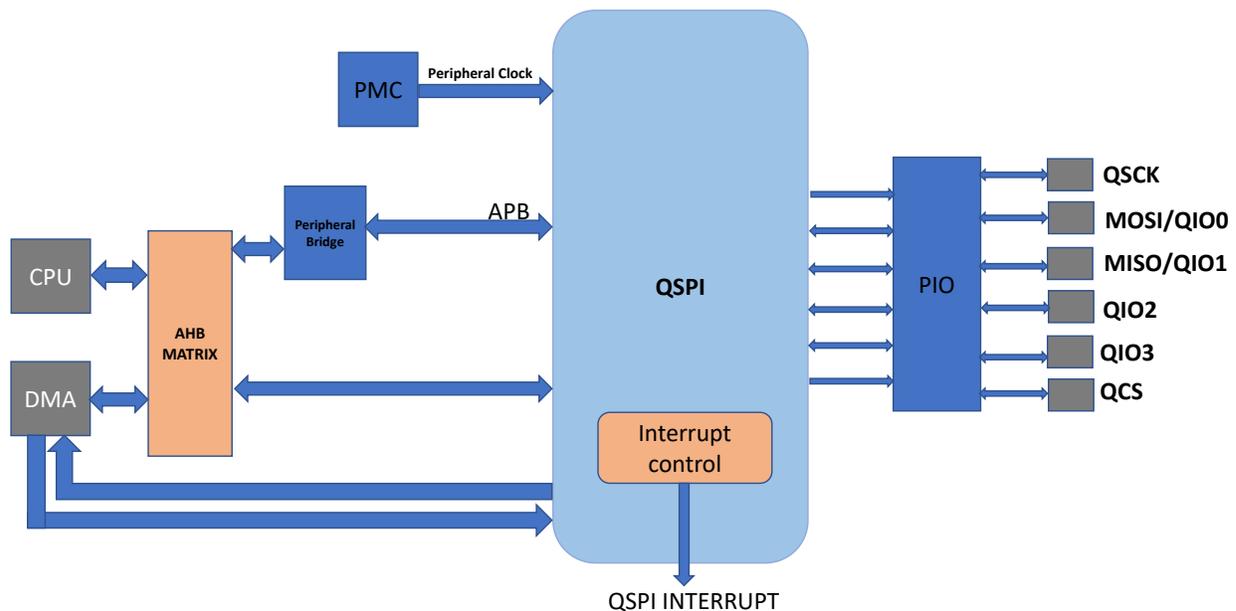
QSPI 可用于以下模式：

- **SPI 模式：** 用作常规 SPI 主模式。连接串行外设，例如 ADC、DAC、LCD 控制器、CAN 控制器和传感器。
注： 本文档仅介绍 QSPI 串行存储器模式。有关 SPI 模式下 QSPI 的操作和配置的详细说明，请参见具体器件数据手册。
- **串行存储器模式：** 连接串行闪存。

QSPI 使得系统能够使用尺寸小巧且价格低廉的高性能串行闪存，以代替尺寸较大且更为昂贵的并行闪存。

下图给出了 QSPI 的框图。

图 2-1. QSPI 框图



通过使用模式寄存器（QSPI_MR）中的 SMM 位，可让 QSPI 在 SPI 模式或串行存储器模式之间切换。QSPI 的工作时钟由内部可编程波特率发生器控制。时钟的相位和极性可在串行时钟寄存器（QSPI_SCR）中配置。下列延时可通过 QSPI_MR 进行编程。QSPI 可根据所连外设的速度和时序使用这些延时与其进行同步。

- 连续传输之间的传输延时
- 时钟和数据之间的延时
- 片选停用和激活之间的延时

2.1 QSPI 串行存储器模式

在串行存储器模式下，QSPI 用作串行闪存控制器。要激活该模式，必须将 QSPI_MR 中的 SMM 位置 1。使能后，外设设在 QSPI 存储空间 0x8000_0000 上显示为存储器映射器件。在串行存储器模式下，数据读写操作在地址 0x8000_0000 处进行。在该模式下，无法通过 QSPI_TDR 或 QSPI_RDR 传输数据。QSPI 可用于从串行闪存中读取数据，从而使 CPU 能够从中执行代码（XIP）。QSPI 还可用于发送特定命令来控制串行闪存（编程、擦除和锁定等）。在串行存储器模式下，QSPI 与以下模式兼容：

- 单位 (Single-Bit) SPI
- 双 SPI
- 四 SPI

2.1.1 单位 SPI

单位 SPI 使用两条数据线：主输出从输入 (Master Out Slave In, MOSI) 和主输入从输出 (Master In Slave Out, MISO)) 与串行闪存进行通信。

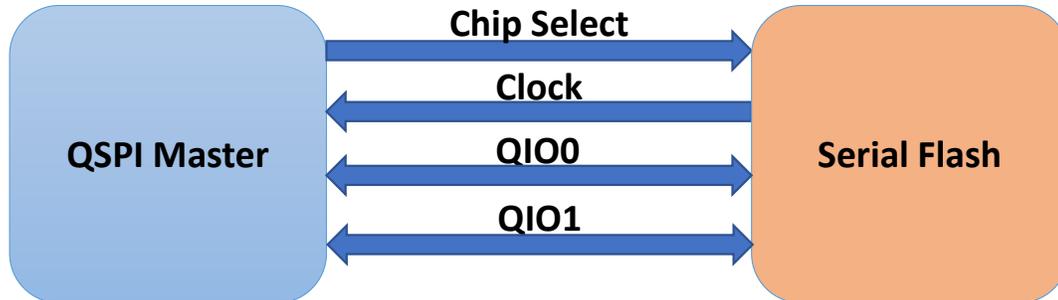
图 2-2. 单位 SPI



2.1.2 双 SPI

双 SPI 使用两条双向 QIO 线 (QIO0 和 QIO1) 与串行闪存或外部存储器进行通信。

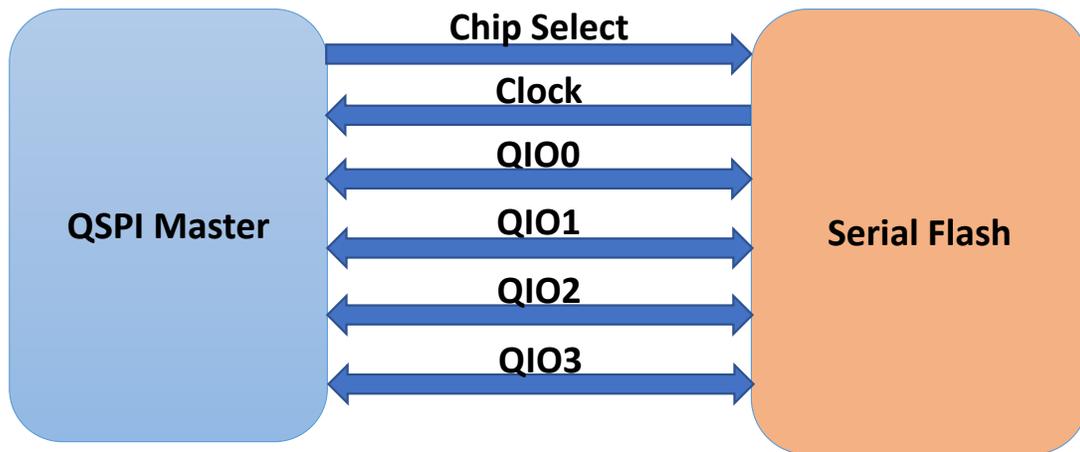
图 2-3. 双 SPI



2.1.3 四 SPI

四 SPI 模式使用四条双向 QIO 线与串行闪存或外部存储器进行通信。

图 2-4. 四 SPI



2.2 指令帧

QSPI 发送 READ、WRITE、PROGRAM、ERASE 和 LOCK 等指令来控制串行闪存。上述所有指令集均由串行闪存供应商提供。为了支持所有串行闪存，QSPI 包含了完整的指令帧寄存器（QSPI_IFR），这使其十分灵活且可兼容所有串行闪存。

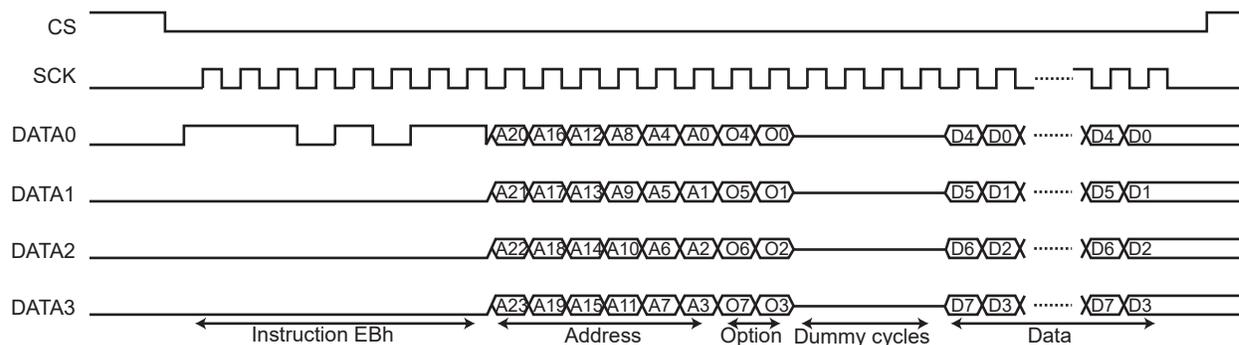
下表列出了指令帧的结构。

表 2-1. 指令帧结构

指令帧字段	说明
指令代码（大小：8 位）	串行闪存列出的指令。在某些情况下是可选的。
地址（大小：24 位或 32 位）	地址是可选的，但 READ、PROGRAM、ERASE 和 LOCK 等指令需要地址。地址长度默认为 24 位，但也可以为 32 位，用以支持大于 128 Mb（16 MB）的串行闪存。
选项代码（大小：1/2/4/8 位）	这对于为某些串行闪存器件中的 READ 指令激活 XIP 模式或连续读模式十分有用。这些模式可缩短数据读取延时。
空周期	READ 指令需要空周期。
数据字节	数据字节用于数据传输指令，例如 READ 或 PROGRAM。

下图给出了典型的 QSPI 模式指令帧。

图 2-5. QSPI 指令帧



2.2.1 指令帧配置

必须基于要发送到外部闪存的控制命令来配置指令帧。有关所支持命令的列表，请参见相应外部闪存的数据手册。需要配置的寄存器如下：

- 指令帧寄存器 (QSPI_IFR)
- 指令地址寄存器 (QSPI_IAR)
- 指令代码寄存器 (QSPI_ICR)

2.2.1.1 指令帧寄存器

必须基于要发送的命令写入 QSPI_IFR。如果指令帧不包含任何数据，则写入该寄存器将触发通过 QSPI 传输指令。如果指令帧包含数据，则将通过 QSPI 存储空间中的第一次数据访问来传输指令帧。QSPI_IFR 包含以下可配置位域：

表 2-2. 指令帧寄存器

位域[bit]	说明
WIDTH [2:0]	配置必须使用哪些数据通道（单位、双或四）发送指令代码、地址、选项代码以及传输数据。
INSTEN [4]	使能后可发送指令代码。
ADDREN [5]	使能后可在指令代码后发送地址。
OPTEN [6]	使能后可在地址后发送选项代码。
DATAEN [7]	使能后可在 READ 或 Program 指令期间接收/发送数据。
OPTL [9:8]	选项代码的长度。该长度必须与 WIDTH 配置一致。
ADDRL [10]	地址长度（24 位或 32 位）。
TFRTYP [13:12]	要执行的数据传输的类型。
CRM [14]	使能连续读模式。
NBDUM [20:16]	读取串行闪存时要添加的空周期数。

传输类型 (TFRTYP)

- **TFRTYP = 0:** 读取串行存储器，例如 JEDEC-ID 或串行存储器状态寄存器，但不读取存储在存储器中的数据。
- **TFRTYP = 1:** 读取串行存储器数据。第一个指令帧的地址是 QSPI 存储空间 (0x80000000) 中第一次读访问的地址。对于非顺序读访问，将与上一次系统读访问一起发送新的指令帧。
- **TFRTYP = 2:** 写入串行存储器，例如配置寄存器或状态寄存器，但不写入 QSPI 空间中的存储器数据
- **TFRTYP = 3:** 写入串行存储空间

注：当 TFRTYP = 0/2/3 时：指令帧中发送的地址是第一次系统总线访问的地址。QSPI 不使用下一次访问的地址。

2.2.1.2 指令地址寄存器

如果指令帧只包含一个地址而没有数据，则必须将要发送的地址写入指令地址寄存器 (QSPI_IAR)。例如，BLOCK ERASE 命令只需要地址，不需要任何数据。当存在数据时，指令的地址由 QSPI 存储空间中的数据访问地址定义，而不是由 QSPI_IAR 定义。

2.2.1.3 指令代码寄存器

如果指令帧包含指令代码和/或选项代码，则必须配置指令代码寄存器 (QSPI_ICR) 中的 INST 和 OPT 位域。

2.2.1.4 指令帧结束

如果未使能数据传输，则当 QSPI_SR 中的 INSTRE 标志置 1 时，表示指令帧结束。如果使能了数据传输，用户必须通过将 QSPI_CR 寄存器中的 LASTXFR 位置 1 来指示 QSPI 存储空间中已完成数据传输。当 QSPI_SR 中的 INSTRE 标志置 1 时，表示指令帧结束。

3. 就地执行

就地执行（XIP）是一种直接从串行闪存执行代码的方法，无需将代码复制到 RAM。串行闪存被视为 MCU 存储器地址映射中的另一个存储器。

通过将 QSPI 配置为四 SPI 串行存储器模式即可使用 XIP。由于四 SPI 模式使用四条线传输数据，这使得系统能够使用尺寸小巧且价格低廉的高性能串行闪存，以代替尺寸较大且更为昂贵的闪存。之所以能够在串行闪存上实现 XIP，原因在于 QSPI 能够读取随机地址的数据，从而允许 CPU 直接从中执行代码。

在 SAME70 器件中，通过将 QSPI 配置为连续读模式（QSPI_IFR 寄存器中的 CRM 位）、将 QSPI_IFR 寄存器中的 DATAEN 和 TFRTP 置 1 并将指令发送到串行闪存来使能 XIP。

在 XIP 模式下，从串行闪存执行的代码指令将映射到 QSPI 存储空间（0x80000000）。

3.1 连续读模式

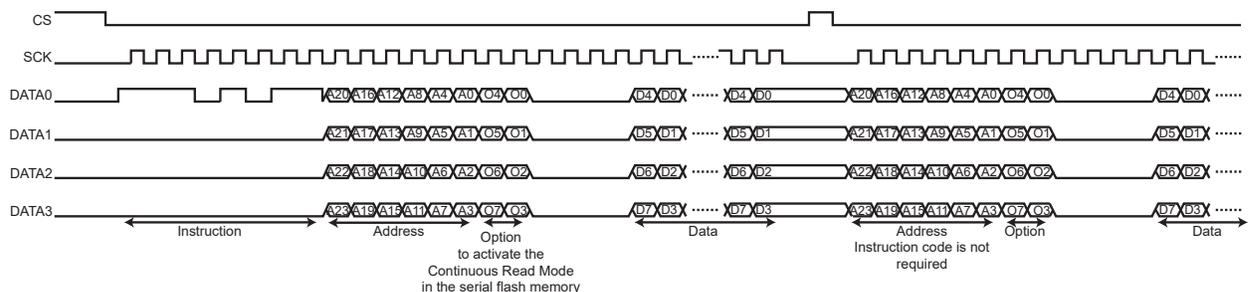
QSPI 支持在某些串行闪存中实现的连续读模式。从存储器读取数据时（TFRTP = 1），使用连续读模式。系统总线读访问的地址通常是非顺序的，这会导致许多指令帧具有相同的指令代码。通过特定的选项代码在串行闪存中激活连续读模式时，指令代码将存储在存储器中。对于后续指令帧，不需要指令代码，因为存储器会使用先前存储的指令代码。通过禁止发送指令代码，连续读模式可缩短数据访问时间并降低指令开销。

QSPI 和串行闪存均必须使能连续读模式。QSPI 中的使能方式是将 QSPI_IFR 中的 CRM 位置 1（TFRTP 位域值必须等于 1）。串行闪存中的使能方式是发送特定的选项代码。

注： 如果串行闪存不支持连续读模式，则无法实现 XIP。

下图显示了 QSPI 连续读模式。

图 3-1. QSPI 连续读模式



4. QSPI 的 MPU 配置

Cortex-M7 处理器有一个存储器保护单元，可根据权限和访问规则将存储器映射划分为多个区域。这有助于提供精细的存储器控制，允许应用程序使用多级权限，以任务为单位对代码、数据和堆栈进行分离与保护。

SAM E70 器件使用 MPU 管理最多 16 个区域，适用于安全关键型应用。下表总结了 Cortex-M7 中可用的 MPU 属性。

表 4-1. MPU 属性

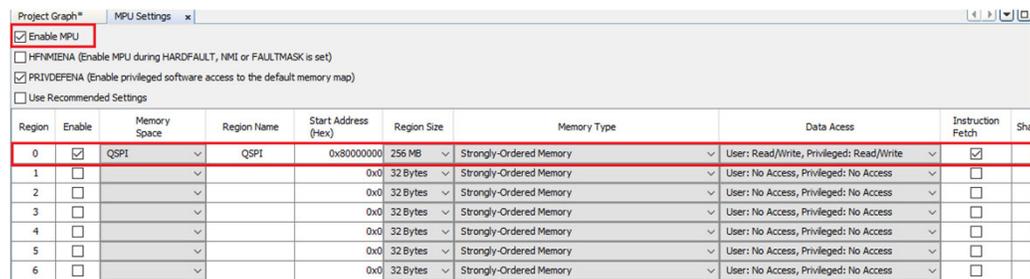
存储器类型	共用性	属性	说明
强排序	不适用	不适用	所有访问按程序顺序进行。不支持并发访问，必须等到当前访问完成之后才能进行下一次访问。
器件	共用	不适用	所有访问按程序顺序进行。存储器映射的外设由多个主器件共用。
	非共用	不适用	所有访问按程序顺序进行。存储器映射的外设由单个主器件使用。
常规	共用	不可高速缓存 可高速缓存的直写 可高速缓存的回写	常规存储器由多个主器件共用。
	非共用	不可高速缓存 可高速缓存的直写 可高速缓存的回写	常规存储器由单个主器件使用。

当 Cortex-M7 处理器访问 QSPI 进行编程操作时，必须在 Cortex-M7 存储器保护单元（Memory Protection Unit, MPU）中定义 QSPI 存储空间。

对于编程操作，必须在 MPU 中使用属性“器件”或“强排序”定义 QSPI 存储空间。对于取操作或读操作，必须在 MPU 中使用属性“常规”定义 QSPI 存储空间，以便对内部高速缓存善加利用。

下图使用 MPLAB Harmony v3 配置器中的“MPU Settings”（MPU 设置）窗口显示了 QSPI 存储区域的 MPU 配置。

图 4-1. MPLAB Harmony v3 MPU 设置



5. 定制链接描述文件

要从 QSPI 存储空间执行，需要将应用程序链接到 QSPI 地址空间。链接器文件在链接中起着重要作用。

在编译器生成目标文件后，必须根据目标器件的存储器映射正确链接这些文件。所有目标文件均使用相对寻址，最终地址映射在链接时执行。链接器将输入文件（目标文件格式）组合成一个输出文件（可执行文件）。每个编译器的链接器文件各不相同。链接器利用链接描述文件或命令文件将不同的代码段和数据段放入适当的存储器中。

默认的链接器文件具有用于闪存和 SRAM 区域的存储段。此外，为了能够将应用程序链接到 QSPI 存储空间，必须为 QSPI 区域定义一个具有相应地址和长度的存储段。定义存储区域后，链接描述文件便可以指示链接器将特定的输出段放入该存储区域。

下图给出了修改后的定制链接器文件，用于将应用程序链接到 QSPI 地址空间。

图 5-1. 定制链接器 ID 文件

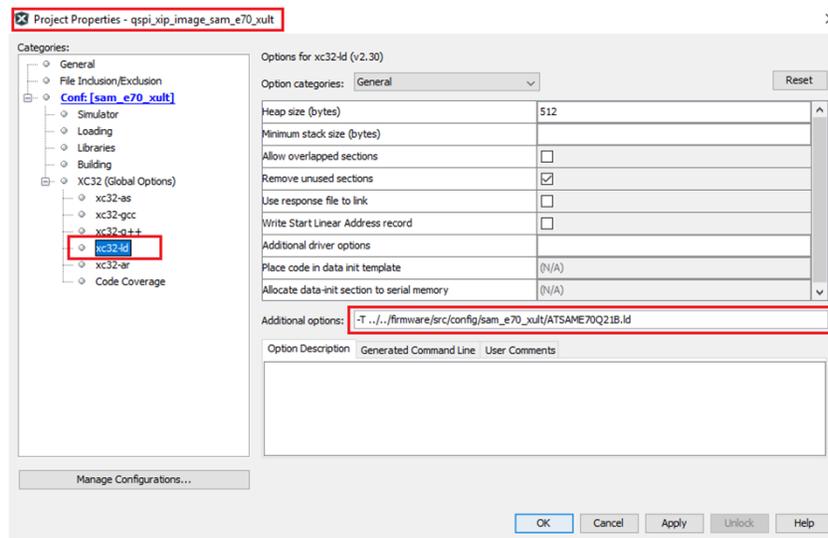
```

46 #define XC32_RESET_HANDLER_NAME Reset_Handler
47 #endif /* __XC32_RESET_HANDLER_NAME */
48
49 /* Set the entry point in the ELF file. Once the entry point is in the ELF
50 * file, you can then use the --write-sla option to xc32-bin2hex to place
51 * the address into the hex file using the SLA field (RECTYPE 5). This hex
52 * record may be useful for a bootloader that needs to determine the entry
53 * point to the application.
54 */
55 ENTRY(__XC32_RESET_HANDLER_NAME)
56
57 /*-----
58 * Memory-Region Macro Definitions
59 * The XC32 linker preprocesses linker scripts. You may define these
60 * macros in the MPLAB X project properties or on the command line when
61 * calling the linker via the xc32-gcc shell.
62 *-----*/
63
64 #ifndef ROM_ORIGIN
65 # define ROM_ORIGIN 0x80000000 Address changed to QSPI memory address
66 #endif
67 #ifndef ROM_LENGTH
68 # define ROM_LENGTH 0x20000000 Length of the QSPI memory space
69 #elif (ROM_LENGTH > 0x20000000)
70 # error ROM_LENGTH is greater than the max size of 0x20000000
71 #endif
72

```

下图给出了用于使能定制链接描述文件的 MPLAB X IDE 链接器配置选项。

图 5-2. 链接到项目的链接器文件

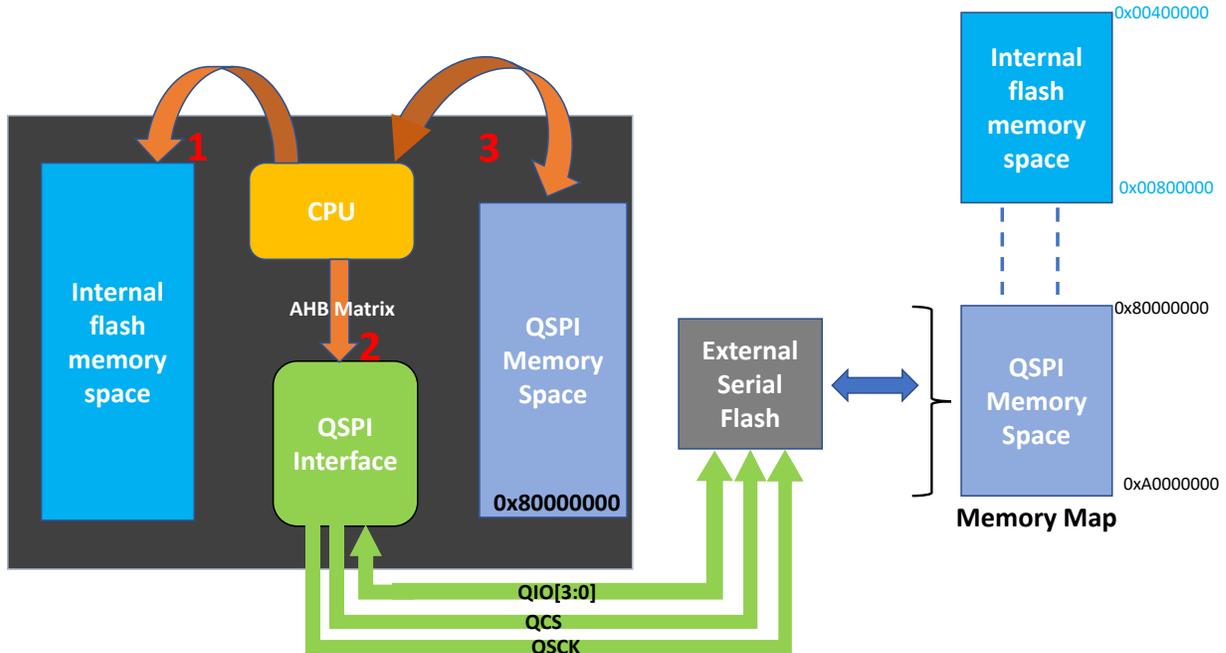


注： 要启用定制链接描述文件，应在 MPLAB X IDE 中导航到 Project Properties（项目属性），然后选择 xc32-ld 选项。

6. 使用 MPLAB Harmony v3 的 QSPI XIP 示例

要使用 MPLAB Harmony v3 实现 QSPI XIP，请参见 1. 硬件和软件要求。SAM E70 Xplained Ultra 板包含与 QSPI 线连接的 4 MB QSPI 闪存（SST26VF032BA）。有关用于与串行闪存通信的命令和指令，请参见 SST26VF032BA 数据手册。

图 6-1. QSPI 应用框图



1. CPU 从内部 MCU 闪存开始执行，并将 QSPI 初始化为串行存储器模式。
2. 使用生成的二进制文件对外部串行闪存进行编程以通过 QSPI 从 QSPI 存储区域运行，并将 QSPI 外设配置为在连续读模式下运行。
3. 在串行存储器模式下，映射到 0x80000000 的串行闪存显示为 CPU 的另一个地址映射的存储器。CPU 从 QSPI 存储区域开始执行。

QSPI XIP 示例包含以下三个应用程序：

- “QSPI XIP 主” MPLAB Harmony v3 应用程序
- “QSPI 镜像” MPLAB Harmony v3 应用程序
- “十六进制镜像生成” Python 应用程序

6.1 QSPI XIP 主 MPLAB Harmony v3 应用程序

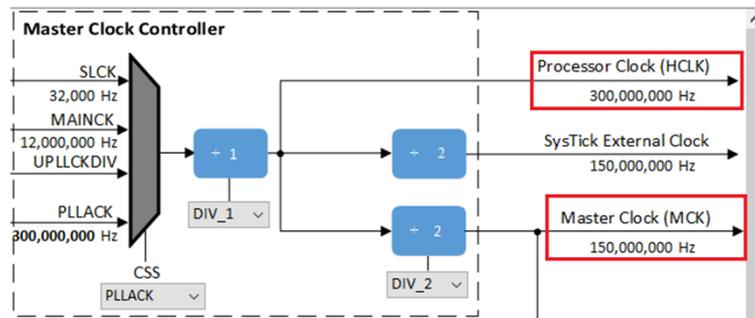
QSPI XIP 主应用程序用作镜像加载程序，以从串行闪存编程和执行。该应用程序：

- 将 QSPI 配置为加载生成的镜像（QSPI 镜像）。
- 将 QSPI 配置为连续读模式以从串行闪存执行代码。
- 包括头文件（xip_image_pattern_hex.h），其中包含以十六进制值表示的“QSPI 镜像”的固件，这些十六进制值在用户运行定制描述文件时自动写入头文件。

QSPI XIP 主应用程序的开发顺序和流程如下：

1. 为 QSPI 存储区域配置 MPU。QSPI 必须按照 [MPLAB Harmony v3 MPU 设置](#) 所示配置为“强排序”才能进行编程操作。
2. 从 MHC Clock Configuration（MHC 时钟配置）窗口验证主器件时钟和处理器时钟。

图 6-2. QSPI XIP 主时钟配置



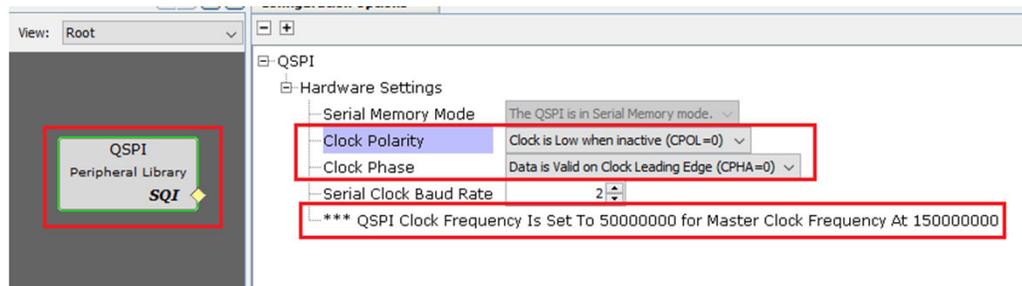
- 从 MHC Pin Settings (MHC 引脚设置) 窗口配置 QSPI 外设的 GPIO 引脚 (QSCK、QCS 和 QIO [3:0])。

图 6-3. QSPI XIP 主 QSPI 引脚配置

Pin Number	Pin ID	Custom Name	Function	Direction	Latch	Open Drain	PIO Interrupt	Pull Up	Pull Down	Glitch/Debounce Filter	Drive
64	PA11	QSPI_QCS	QSPI_QCS	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
68	PA12	QSPI_QIO1	QSPI_QIO1	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
42	PA13	QSPI_QIO0	QSPI_QIO0	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
51	PA14	QSPI_QSCK	QSPI_QSCK	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
25	PA17	QSPI_QIO2	QSPI_QIO2	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
2	PD31	QSPI_QIO3	QSPI_QIO3	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low

- 为 QSPI 配置时钟和极性。

图 6-4. QSPI XIP 主 QSPI 配置



- 在应用程序中使能四 SPI 模式以提升性能 (对于串行闪存 SST26VF032BA, 发送命令 “0x38”)。
- 通过执行相应的 ERASE 命令擦除串行存储器 (对于串行闪存 SST26VF032BA, 发送命令 “0xD8”)。该应用程序提供 APP_BulkErase 或 APP_Erase API 函数来执行擦除。
- 使用包含从 QSPI 镜像二进制文件提取的十六进制值的输入缓冲区发送 “页编程” 命令 (对于串行闪存 SST26VF032BA, 命令为 “0x02”) (在下一节中说明)。
- 在四 SPI 模式下从 QSPI 存储器读取前 32 个字节, 以提取 QSPI 镜像应用程序的堆栈指针和复位处理程序地址。
- 通过使用 API APP_MemoryReadContinuous() 使能 “连续读模式” 以进入 XIP。
- 从 QSPI 存储器读取数据后, 使用输入缓冲区验证读取的内容。
- 如果验证通过 (即写入 QSPI 的数据和从 QSPI 读取的数据匹配), 则配置 QSPI 镜像应用程序 (在步骤 8 中提取的 QSPI 闪存中编程) 的堆栈指针和复位处理程序。

图 6-5. SP 和 PC 配置

```

/* Set PC and SP of the XIP Image application programmed in QSPI Flash memory*/

/* Pointer to Reset Handler of the XIP Image Application */
__start_new = (uint32_t(*) (void)) xip_image_buffer[1];
__start_sp = xip_image_buffer[0];

__set_MSP(__start_sp);

/* Rebase the vector table base address */
SCB->VTOR = ((uint32_t) QSPIMEM_ADDR & SCB_VTOR_TBLOFF_Msk);

/* Run XIP Image Application */
__start_new();

/* Should never reach here */
break;

```

12. 执行完前面的步骤后，控制将跳转到 QSPI 镜像，并且应用程序将从 QSPI 存储区域运行。

注： 上电复位（Power-on Reset, POR）时，QSPI XIP 主应用程序将按照前面步骤中的流程执行。

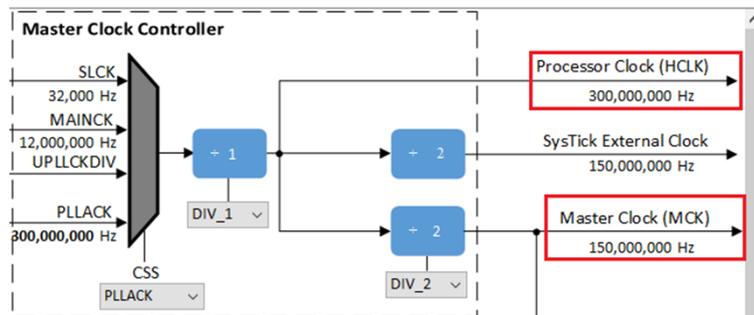
6.2 QSPI 镜像 MPLAB Harmony v3 应用程序

QSPI 镜像应用程序是一个简单的应用程序，每秒使 LED 闪烁一次。要从 QSPI 存储区域执行应用程序，必须将 QSPI 镜像应用程序的二进制文件链接到 QSPI 地址空间。QSPI 镜像应用程序使用定制链接描述文件使 LED 连续闪烁。

QSPI 镜像应用程序的开发顺序和流程如下：

1. 从 MHC Clock Configuration 窗口验证主器件时钟和处理器时钟。

图 6-6. QSPI 镜像时钟配置



2. 从 MHC Pin Settings 窗口配置 QSPI 外设的 GPIO 引脚（QSCK、QCS 和 QIO [3:0]）。

图 6-7. QSPI 镜像 QSPI 引脚配置

Pin Number	Pin ID	Custom Name	Function	Direction	Latch	Open Drain	PIO Interrupt	Pull Up	Pull Down	Glitch/Debounce Filter	Drive
64	PA11	QSPI_QCS	QSPI_QCS	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
68	PA12	QSPI_QIO1	QSPI_QIO1	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
42	PA13	QSPI_QIO0	QSPI_QIO0	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
51	PA14	QSPI_QSCK	QSPI_QSCK	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
25	PA17	QSPI_QIO2	QSPI_QIO2	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
2	PD31	QSPI_QIO3	QSPI_QIO3	n/a	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low

- 将 LED 引脚配置为 GPIO。

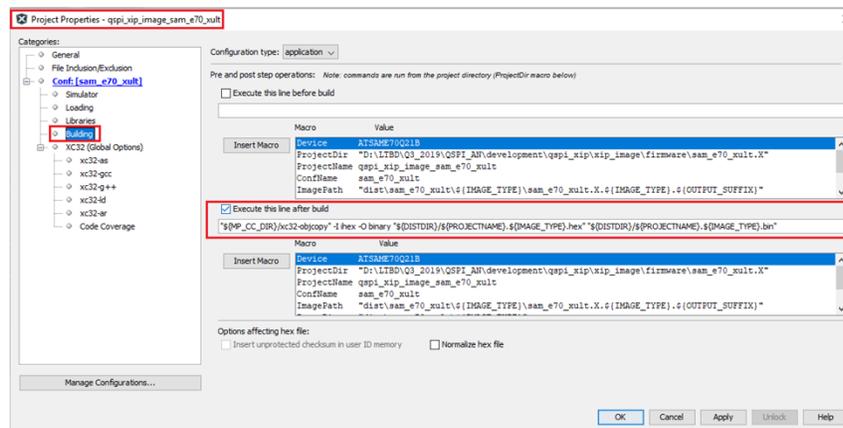
图 6-8. QSPI 镜像 LED 引脚配置

Pin Number	Pin ID	Custom Name	Function	Direction	Latch	Open Drain	PIO Interrupt	Pull Up	Pull Down	Glitch/Debounce Filter	Drive
102	PA0		Available	In	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
99	PA1		Available	In	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
93	PA2		Available	In	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
91	PA3		Available	In	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
77	PA4		Available	In	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
73	PA5	LED	GPIO	Out	Low	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
114	PA6		Available	In	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low
35	PA7		Available	In	n/a	<input type="checkbox"/>	Disabled	<input type="checkbox"/>	<input type="checkbox"/>	Disabled	Low

- 按照上文定制链接描述文件一章的说明链接定制链接描述文件。将该定制链接器文件修改为从 QSPI 存储区域运行。
- 生成一个二进制文件，Python 应用程序随后会将该文件转换为十六进制值数组。

要生成二进制格式的输出，应在 QSPI 镜像应用程序的 Project Properties 中进行更改，如下图所示。

图 6-9. QSPI 镜像二进制输出设置



注：以下代码是用于将十六进制文件转换为二进制文件的命令：

```

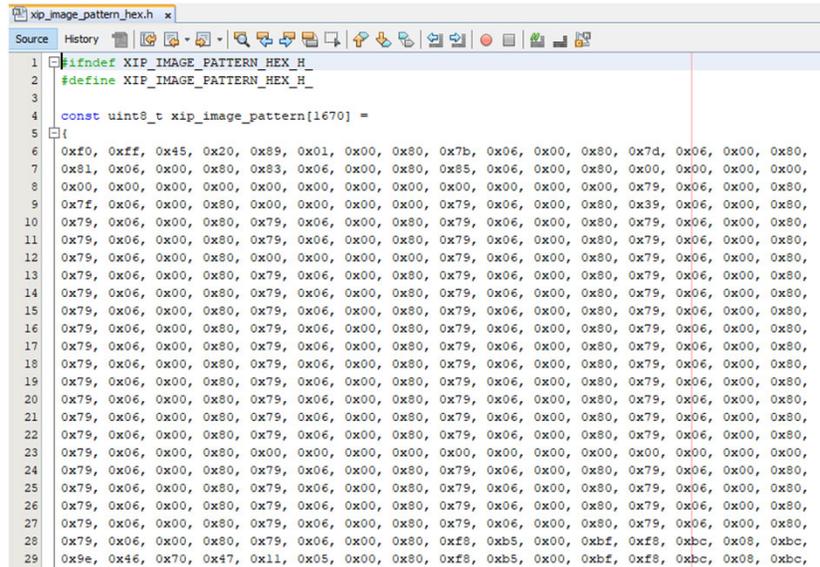
${MP_CC_DIR}/xc32-obcopy" -I ihex -O binary "${DISTDIR}/${PROJECTNAME}.${IMAGE_TYPE}.hex" ${
DISTDIR}/${PROJECTNAME}.${IMAGE_TYPE}.bin

```

6.3 十六进制镜像生成 Python 应用程序

十六进制镜像生成 Python 应用程序是一个定制 Python 描述文件，用于将二进制（QSPI 镜像）文件转换为 .hex 格式，并将其以 QSPI XIP 主应用程序中使用的十六进制值数组形式存储在头文件（xip_image_pattern_hex.h）中。

图 6-10. XIP QSPI 镜像头文件



```

1 #ifndef XIP_IMAGE_PATTERN_HEX_H
2 #define XIP_IMAGE_PATTERN_HEX_H
3
4 const uint8_t xip_image_pattern[1670] =
5 {
6 0xf0, 0xff, 0x45, 0x20, 0x89, 0x01, 0x00, 0x80, 0x7b, 0x06, 0x00, 0x80, 0x7d, 0x06, 0x00, 0x80,
7 0x81, 0x06, 0x00, 0x80, 0x83, 0x06, 0x00, 0x80, 0x85, 0x06, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00,
8 0x00, 0x79, 0x06, 0x00, 0x80,
9 0x7f, 0x06, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x79, 0x06, 0x00, 0x80, 0x39, 0x06, 0x00, 0x80,
10 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
11 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
12 0x79, 0x06, 0x00, 0x80, 0x00, 0x00, 0x00, 0x00, 0x00, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
13 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
14 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
15 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
16 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
17 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
18 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
19 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
20 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
21 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
22 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
23 0x79, 0x06, 0x00, 0x80, 0x00, 0x00,
24 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
25 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
26 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
27 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80,
28 0x79, 0x06, 0x00, 0x80, 0x79, 0x06, 0x00, 0x80, 0xf8, 0xb5, 0x00, 0xbf, 0xf8, 0xbc, 0x08, 0xbc,
29 0x9e, 0x46, 0x70, 0x47, 0x11, 0x05, 0x00, 0x80, 0xf8, 0xb5, 0x00, 0xbf, 0xf8, 0xbc, 0x08, 0xbc,

```

注：MPLAB Harmony v3 csp 资源库中提供 QSPI XIP 示例。

要运行 QSPI XIP 示例，请按照下列步骤操作：

1. 打开 MPLAB X IDE 中的 QSPI 镜像项目 (<MPLAB Harmony v3 download path>/csp/apps/qspi/qspi_xip/xip_image/firmware/sam_e70_xult.X)。
2. 使用 MPLAB X IDE 编译项目，但不进行编程。
3. 在命令提示符中使用以下命令运行 Python 描述文件 (<MPLAB Harmony v3 download path>/csp/apps/qspi/qspi_xip/xip_image_pattern_gen.py)，以从 QSPI 镜像应用程序中生成的二进制文件中提取十六进制代码，并将其以十六进制值数组的形式存储到头文件 (<MPLAB Harmony v3 download path>/csp/apps/qspi/qspi_xip/xip_main/firmware/src/config/sam_e70_xult/xip_image_pattern_hex.h) 中。

命令：python xip_image_pattern_gen.py

4. 打开 MPLAB X IDE 中的 QSPI XIP 主项目 (<MPLAB Harmony v3 download path>/csp/apps/qspi/qspi_xip/xip_main/firmware/sam_e70_xult.X)。
5. 使用 MPLAB X IDE 编译和编程应用程序。
6. 主应用程序对 QSPI 镜像进行编程，并从串行闪存执行代码。QSPI 镜像应用程序开始使 LED 连续闪烁。

7. 性能

QSPI 使得系统能够使用尺寸小巧且价格低廉的高性能串行闪存，以代替尺寸较大且更为昂贵的并行闪存。因此，QSPI 的性能起着至关重要的作用。QSPI 的性能受 QSPI 速度和闪存功能等因素的限制。

注：有关 QSPI XIP 模式下各项性能指数的详细信息，请参见 [9. 参考资料](#) 部分给出的应用笔记“Execute-In-Place with QSPI using ASF”。 “Execute-In-Place with QSPI using ASF” 文档中提及的应用程序并非使用 MPLAB Harmony v3 开发，并且文档中的各项性能指数可能会因编译器设置和优化级别而异。

8. 结论

开发 QSPI XIP 应用程序需要了解 QSPI 协议、MPU 设置和链接描述文件。MPLAB Harmony v3 提供了一款用于 32 位 SAM 和 PIC 单片机的灵活、紧凑的全集成固件开发平台。本文档介绍了如何将 QSPI 中的 XIP 模式与外部闪存配合使用。

9. 参考资料

请参见文档“Execute-In-Place (XIP) with Quad SPI Interface (QSPI) using ASF”，该文档可从以下位置下载：

http://ww1.microchip.com/downloads/en/AppNotes/Atmel-44065-Execute-in-Place-XIP-with-Quad-SPI-Interface-SAM-V7-SAM-E7-SAM-S7_Application-Note.pdf

Microchip 网站

Microchip 网站 (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。我们的网站提供以下内容：

- **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**——常见问题解答 (FAQ)、技术支持请求、在线讨论组以及 Microchip 设计伙伴计划成员名单
- **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

产品变更通知服务

Microchip 的产品变更通知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，请访问 www.microchip.com/pcn，然后按照注册说明进行操作。

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 (ESE)
- 技术支持

客户应联系其代理商、代表或 ESE 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 www.microchip.com/support 获得网上技术支持。

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规范。
- Microchip 确信：在正常使用的情况下，Microchip 系列产品非常安全。
- 目前，仍存在着用恶意、甚至是非法的方法来试图破坏代码保护功能的行为。我们确信，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。这种试图破坏代码保护功能的行为极可能侵犯 Microchip 的知识产权。
- Microchip 愿与那些注重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

提供本文档的中文版本仅为为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中提供的信息仅仅是为方便您使用 Microchip 产品或使用这些产品来进行设计。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 “按原样”提供这些信息。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，包括但不限于针对非侵权性、适销性和特定用途的适用性的暗示担保，或针对其使用情况、质量或性能的担保。

在任何情况下，对于因这些信息或使用这些信息而产生的任何间接的、特殊的、惩罚性的、偶然的或间接的损失、损害或任何类型的开销，Microchip 概不承担任何责任，即使 Microchip 已被告知可能发生损害或损害可以预见。在法律允许的最大范围内，对于因这些信息或使用这些信息而产生的所有索赔，Microchip 在任何情况下所承担的全部责任均不超出您为获得这些信息向 Microchip 直接支付的金额（如有）。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损害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任。除非另外声明，在 Microchip 知识产权保护下，不得暗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、Adaptec、AnyRate、AVR、AVR 徽标、AVR Freaks、BesTime、BitCloud、chipKIT、chipKIT 徽标、CryptoMemory、CryptoRF、dsPIC、FlashFlex、flexPWR、HELDO、IGLOO、JukeBlox、KeeLoq、Kleer、LANCheck、LinkMD、maxStylus、maXTouch、MediaLB、megaAVR、Microsemi、Microsemi 徽标、MOST、MOST 徽标、MPLAB、OptoLyzer、PackeTime、PIC、picoPower、PICSTART、PIC32 徽标、PolarFire、Prochip Designer、QTouch、SAM-BA、SenGenuity、SpyNIC、SST、SST 徽标、SuperFlash、Symmetricom、SyncServer、Tachyon、TimeSource、tinyAVR、UNI/O、Vectron 及 XMEGA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的注册商标。

AgileSwitch、APT、ClockWorks、The Embedded Control Solutions Company、EtherSynch、FlashTec、Hyper Speed Control、HyperLight Load、IntelliMOS、Liberio、motorBench、mTouch、Powermite 3、Precision Edge、ProASIC、ProASIC Plus、ProASIC Plus 徽标、Quiet-Wire、SmartFusion、SyncWorld、Temux、TimeCesium、TimeHub、TimePictra、TimeProvider、WinPath 和 ZL 均为 Microchip Technology Incorporated 在美国的注册商标。

Adjacent Key Suppression、AKS、Analog-for-the-Digital Age、Any Capacitor、AnyIn、AnyOut、Augmented Switching、BlueSky、BodyCom、CodeGuard、CryptoAuthentication、CryptoAutomotive、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、Espresso T1S、EtherGREEN、IdealBridge、In-Circuit Serial Programming、ICSP、INICnet、Intelligent Paralleling、Inter-Chip Connectivity、JitterBlocker、maxCrypto、maxView、memBrain、Mindi、MiWi、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICKit、PICtail、PowerSmart、PureSilicon、QMatrix、REAL ICE、Ripple Blocker、RTAX、RTG4、SAM-ICE、Serial Quad I/O、simpleMAP、SimpliPHY、SmartBuffer、SMART-I.S.、storClad、SQI、SuperSwitcher、SuperSwitcher II、Switchtec、SynchroPHY、Total Endurance、TSHARC、USBCheck、VariSense、VectorBlox、VeriPHY、ViewSpan、WiperLock、XpressConnect 和 ZENA 均为 Microchip Technology Incorporated 在美国和其他国家或地区的商标。

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

Adaptec 徽标、Frequency on Demand、Silicon Storage Technology 和 Symmcom 均为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

GestIC 为 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。

© 2020, Microchip Technology Incorporated 版权所有。

ISBN: 978-1-5224-7223-0

AMBA、Arm、Arm7、Arm7TDMI、Arm9、Arm11、Artisan、big.LITTLE、Cordio、CoreLink、CoreSight、Cortex、DesignStart、DynamIQ、Jazelle、Keil、Mali、Mbed、Mbed Enabled、NEON、POP、RealView、SecurCore、Socrates、Thumb、TrustZone、ULINK、ULINK2、ULINK-ME、ULINK-PLUS、ULINKpro、µVision 和 Versatile 均为 Arm Limited（或其子公司）在美国和/或其他国家/地区的商标或注册商标。

质量管理体系

有关 Microchip 的质量管理体系的信息，请访问 www.microchip.com/quality。

全球销售及服务中心

美洲	亚太地区	亚太地区	欧洲
公司总部 2355 West Chandler Blvd. Chandler, AZ 85224-6199 电话: 480-792-7200 传真: 480-792-7277 技术支持: www.microchip.com/support 网址: www.microchip.com	澳大利亚 - 悉尼 电话: 61-2-9868-6733 中国 - 北京 电话: 86-10-8569-7000 中国 - 成都 电话: 86-28-8665-5511 中国 - 重庆 电话: 86-23-8980-9588 中国 - 东莞 电话: 86-769-8702-9880 中国 - 广州 电话: 86-20-8755-8029 中国 - 杭州 电话: 86-571-8792-8115 中国 - 香港特别行政区 电话: 852-2943-5100 中国 - 南京 电话: 86-25-8473-2460 中国 - 青岛 电话: 86-532-8502-7355 中国 - 上海 电话: 86-21-3326-8000 中国 - 沈阳 电话: 86-24-2334-2829 中国 - 深圳 电话: 86-755-8864-2200 中国 - 苏州 电话: 86-186-6233-1526 中国 - 武汉 电话: 86-27-5980-5300 中国 - 西安 电话: 86-29-8833-7252 中国 - 厦门 电话: 86-592-2388138 中国 - 珠海 电话: 86-756-3210040	印度 - 班加罗尔 电话: 91-80-3090-4444 印度 - 新德里 电话: 91-11-4160-8631 印度 - 浦那 电话: 91-20-4121-0141 日本 - 大阪 电话: 81-6-6152-7160 日本 - 东京 电话: 81-3-6880-3770 韩国 - 大邱 电话: 82-53-744-4301 韩国 - 首尔 电话: 82-2-554-7200 马来西亚 - 吉隆坡 电话: 60-3-7651-7906 马来西亚 - 槟榔屿 电话: 60-4-227-8870 菲律宾 - 马尼拉 电话: 63-2-634-9065 新加坡 电话: 65-6334-8870 台湾地区 - 新竹 电话: 886-3-577-8366 台湾地区 - 高雄 电话: 886-7-213-7830 台湾地区 - 台北 电话: 886-2-2508-8600 泰国 - 曼谷 电话: 66-2-694-1351 越南 - 胡志明市 电话: 84-28-5448-2100	奥地利 - 韦尔斯 电话: 43-7242-2244-39 传真: 43-7242-2244-393 丹麦 - 哥本哈根 电话: 45-4485-5910 传真: 45-4485-2829 芬兰 - 埃斯波 电话: 358-9-4520-820 法国 - 巴黎 电话: 33-1-69-53-63-20 传真: 33-1-69-30-90-79 德国 - 加兴 电话: 49-8931-9700 德国 - 哈恩 电话: 49-2129-3766400 德国 - 海尔布隆 电话: 49-7131-72400 德国 - 卡尔斯鲁厄 电话: 49-721-625370 德国 - 慕尼黑 电话: 49-89-627-144-0 传真: 49-89-627-144-44 德国 - 罗森海姆 电话: 49-8031-354-560 以色列 - 若那那市 电话: 972-9-744-7705 意大利 - 米兰 电话: 39-0331-742611 传真: 39-0331-466781 意大利 - 帕多瓦 电话: 39-049-7625286 荷兰 - 德卢内市 电话: 31-416-690399 传真: 31-416-690340 挪威 - 特隆赫姆 电话: 47-72884388 波兰 - 华沙 电话: 48-22-3325737 罗马尼亚 - 布加勒斯特 电话: 40-21-407-87-50 西班牙 - 马德里 电话: 34-91-708-08-90 传真: 34-91-708-08-91 瑞典 - 哥德堡 电话: 46-31-704-60-40 瑞典 - 斯德哥尔摩 电话: 46-8-5090-4654 英国 - 沃金厄姆 电话: 44-118-921-5800 传真: 44-118-921-5820
亚特兰大 德卢斯, 佐治亚州 电话: 678-957-9614 传真: 678-957-1455 奥斯汀, 德克萨斯州 电话: 512-257-3370 波士顿 韦斯特伯鲁, 马萨诸塞州 电话: 774-760-0087 传真: 774-760-0088 芝加哥 艾塔斯卡, 伊利诺伊州 电话: 630-285-0071 传真: 630-285-0075 达拉斯 阿迪森, 德克萨斯州 电话: 972-818-7423 传真: 972-818-2924 底特律 诺维, 密歇根州 电话: 248-848-4000 休斯顿, 德克萨斯州 电话: 281-894-5983 印第安纳波利斯 诺布尔斯特维尔, 印第安纳州 电话: 317-773-8323 传真: 317-773-5453 电话: 317-536-2380 洛杉矶 米慎维荷, 加利福尼亚州 电话: 949-462-9523 传真: 949-462-9608 电话: 951-273-7800 罗利, 北卡罗来纳州 电话: 919-844-7510 纽约, 纽约州 电话: 631-435-6000 圣何塞, 加利福尼亚州 电话: 408-735-9110 电话: 408-436-4270 加拿大 - 多伦多 电话: 905-695-1980 传真: 905-695-2078			