

简介

本文档介绍了如何使用带有 MPLAB® XC32 编译器和 MPLAB Harmony v3 软件开发框架的 MPLAB X IDE 配置和使用 PIC32C 和 SAM MCU 中的紧耦合存储器（Tightly Coupled Memory, TCM）功能。

TCM 为应用开发人员带来诸多优势，其性能与代码位于高速缓存中时的性能相同。

下面给出了一些在应用程序中优化 TCM 使用的建议：

- 应用程序堆栈可位于 DTCM 中
- 应用程序堆可位于 DTCM 中
- 关键变量可位于 DTCM 中
- 经常更新的变量可位于 DTCM 中
- 关键函数/程序可位于 ITCM 中
- 中断向量表的副本可位于 ITCM 中
- 中断服务程序可位于 ITCM 中

目录

简介..... 1

1. Microchip 基于 Cortex-M7 的 MCU 概述..... 3

 1.1. PIC32CZ CA 系列单片机..... 3

 1.2. SAM E70/S70/V7x 系列单片机..... 4

2. 高速缓存存储器与 TCM 概述..... 5

3. 紧耦合存储器..... 6

 3.1. 紧耦合接口单元（TCU） 6

 3.2. 访问..... 6

 3.3. 配置..... 7

4. 软件对 TCM 的影响..... 9

 4.1. PIC32CZ CA 系列..... 9

 4.2. SAM E70/S70/V7x 系列..... 11

5. TCM 配置的编程序列..... 15

 5.1. PIC32CZ CA 系列..... 15

 5.2. SAM E70/S70/V7x 系列..... 15

6. 硬件和软件要求..... 17

 6.1. 准备工作..... 17

 6.2. 应用概览..... 17

 6.3. 应用程序代码..... 17

7. 编译并编程应用程序..... 19

8. 观察 MPLAB 数据可视化器上的输出..... 22

9. 资源..... 31

10. 缩写..... 32

11. 版本历史..... 33

Microchip 信息..... 34

 商标..... 34

 法律声明..... 34

 Microchip 器件代码保护功能..... 34

1. Microchip 基于 Cortex-M7 的 MCU 概述

Microchip 的 PIC32CZ CA 系列和 SAM E70/S70/V7x 系列单片机 (MCU) 是采用 Arm Cortex-M7 内核的高性能 MCU。该系列 MCU 提供了丰富的外设, 并具有大型内置代码和数据存储器。为确保对时间敏感的代码和数据进行高速、低延时且确定性的访问, Arm Cortex-M7 内核连接到 TCM。

1.1. PIC32CZ CA 系列单片机

PIC32CZ CA 系列单片机采用高性能 Arm Cortex-M7 内核以及高级存储器架构设计。这些单片机提供最高 8 MB 的闪存和 1 MB 的 SRAM, 可满足复杂应用对存储器的严格需求。Arm Cortex-M7 内核包含一个内置纠错码 (Error Correction Code, ECC) 的专用 TCM 的接口, 可增强高速、低延时和确定性处理能力。

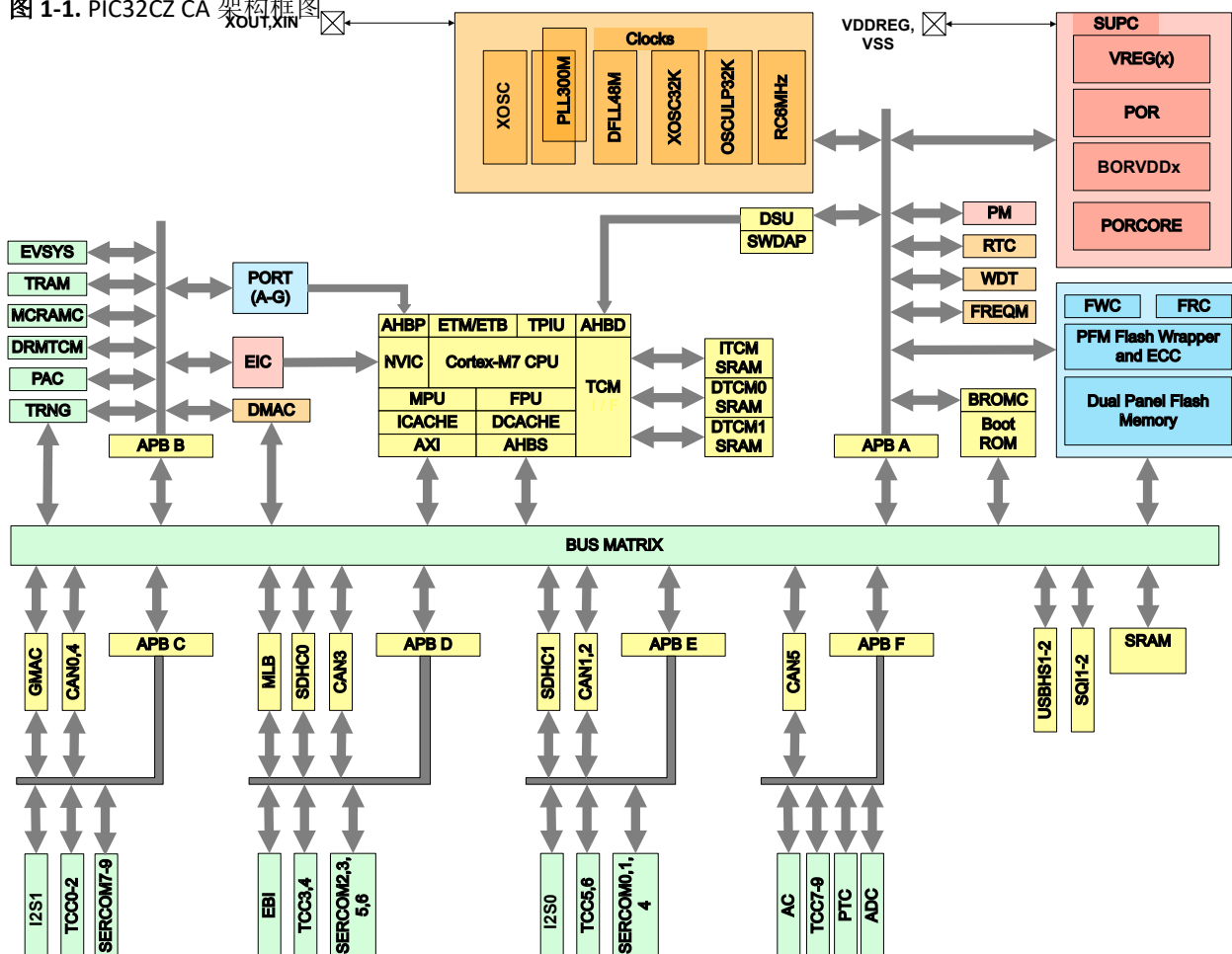
PIC32CZ CA9x 系列提供集成的硬件安全模块 (Hardware Security Module, HSM) 和多种连接功能, 是汽车和工业领域应用以及需要强大安全措施的其他领域 (如通信加密、安全引导过程、身份验证协议和安全密钥存储) 的绝佳选择。

它提供了丰富的连接功能, 例如高速 USB、CAN FD、串行四线 I/O (Serial Quad I/O, SQI)、安全数字高容量存储 (Secure Digital High Capacity, SDHC)、I²S™、媒体本地总线 (Media LB™)、外部总线接口

(External Bus Interface, EBI)、10/100/1000 以太网和串行通信接口 (SERCOM)。

这些 MCU 配备了存储器保护单元 (Memory Protection Unit, MPU) 来增强安全性, 并集成了时钟故障检测机制和故障安全内部 RC 振荡器, 可为选定的外设提供写保护以确保系统完整性。

图 1-1. PIC32CZ CA 架构框图



2. 高速缓存存储器与 TCM 概述

本章重点介绍高速缓存存储器与 TCM 之间的主要差异。

表 2-1. TCM 与高速缓存

紧耦合存储器	高速缓存存储器
紧耦合存储器通过与内核的专用连接进行访问。 在 PIC32CZ CA 和 SAM E70/S70/V7x 系列单片机中，从 Cortex-M7 内核到内部 SRAM 有两个专用连接，一个用于指令 TCM，另一个用于数据 TCM。	高速缓存存储器集成在 Cortex-M7 内核内部的 RAM 中。 在 PIC32CZ CA 和 SAM E70/S70/V7x 系列单片机中，提供了 16 KB 的指令高速缓存（I-cache）和 16 KB 的数据高速缓存（D-cache）。
紧耦合存储器是系统存储器映射的一部分，具有明确的起始地址。TCM 的大小决定了结束地址。	高速缓存存储器不是系统存储器映射的一部分。它没有物理存储器地址。
编程器可在编译时决定将内容存储在 TCM 中的位置。	控制逻辑（而非编程器）确定存储在高速缓存存储器中的内容。
TCM 可直接通过软件访问。	在程序执行期间，高速缓存中存放从存储器向 CPU 取出的指令或数据。
CPU 和 DMA 均可访问 TCM。	DMA 不能访问高速缓存。
TCM 具有确定性的访问时间。访问 TCM 的内容始终需要单周期。	高速缓存存储器用作处理器和存储器之间的中间缓冲区，可缩短存储器访问时间。访问存储器存储单元所需的周期数因高速缓存命中和高速缓存未命中情况而异。

注：TCM 内容未被高速缓存。TCM 通过总线直接连接到 Cortex-M7 内核。可采用与访问高速缓存相似的速度对其进行访问，而不受高速缓存未命中和高速缓存一致性问题的影响。

3. 紧耦合存储器

PIC32CZ CA80/90 和 SAM E70/S70/V7x 系列器件支持两个 TCM 实例：指令 TCM（Instruction TCM，ITCM）和数据 TCM（Data TCM，DTCM）。每个 TCM 的基址均由 Arm 系统地址映射固定：

- ITCM 固定在 0x00000000
- DTCM 固定在 0x20000000

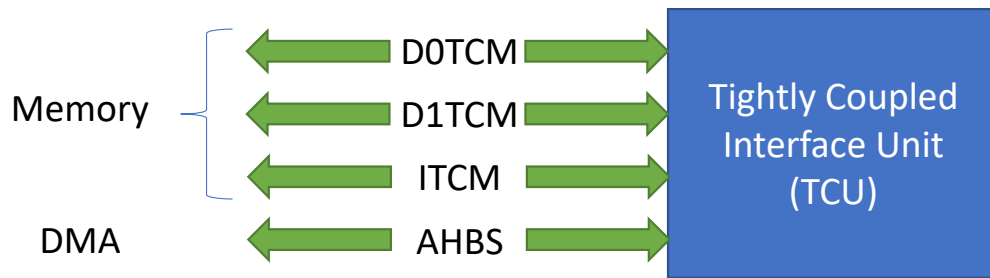
在 Arm 单片机存储器映射中，ITCM 和 DTCM 占用的地址范围低于闪存和内部 SRAM。

3.1. 紧耦合接口单元（TCU）

TCU 接口至以下部分：

- **ITCM**：一个 64 位接口
- **DTCM**：通过 2 个 32 位接口连接：
 - 双 32 位接口支持 DTCM 实现两路同时访问。可在一个周期内同时获取两个 32 位数据。
- **AHBS**：TCM 可由 DMA 通过 AHB 从模式总线接口进行访问：
 - 这样，外设（例如 GMAC）便可在无任何 CPU 干预的情况下访问 TCM 存储器。

图 3-1. 紧耦合接口单元



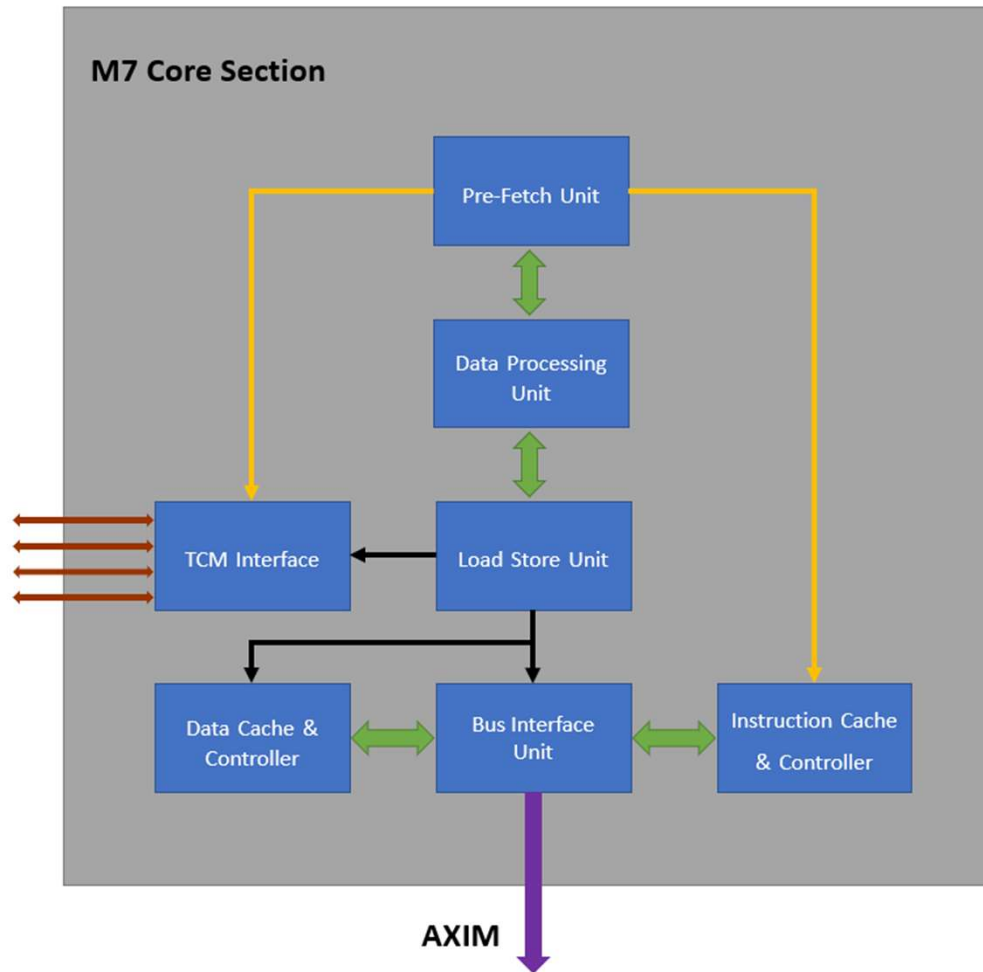
3.2. 访问

对于每个 TCM 接口，读/写请求由以下内核接口触发：

- 装载存储单元（Load Store Unit，LSU）
- 预取单元（Pre-fetch Unit，PFU）
- 系统 AHBS 接口（AHBS）
- 调试单元（DGU）
- 总线接口单元（Bus Interface Unit，BIU）

TCM 包含仲裁逻辑，可对各种接口的 TCM 访问请求仲裁。PFU 只能读取 TCM，而其他接口既可读取也可写入 TCM。

图 3-2. TCM 接口的 Cortex®-M7 架构



3.3. 配置

3.3.1. PIC32CZ CA 系列单片机

PIC32CZ CA 系列单片机具有总容量为 256 KB 的集成 TCM，平均分配给 128 KB 的指令 TCM（ITCM）和 128 KB 的数据 TCM（DTCM）。ITCM 和 DTCM 的大小是固定的，并通过 MPLAB Harmony v3 使用 MPLAB 代码配置器（MPLAB Code Configurator，MCC）进行默认配置。

注：

- 以处理器速度（HCLK）访问 TCM。在 PIC32CZ CA 系列中，HCLK 最高可达 300 MHz，而无任何等待状态。来自内核的所有访问均配置为单周期访问。因此，TCM 访问的行为具有确定性。
- 可通过 Cortex-M7 的系统控制块中的寄存器配置来明确使能或禁止 TCM。

3.3.2. SAM E70/S70/V7x 系列单片机

TCM 可以通过软件设置为四种不同配置之一。可以通过对 GPNVM bit (8:7) 进行编程来更改配置，其值如下表中所示。多种 TCM 配置为应用开发人员带来了灵活性。

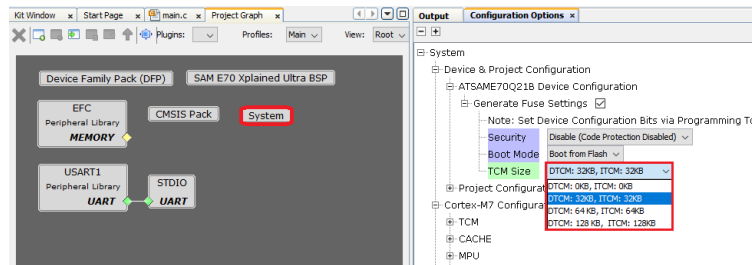
表 3-1. TCM 配置（单位为 KB）

ITCM (KB)	DTCM (KB)	可用 SRAM (KB)		GPNVM bit (8:7)
		384 KB 型号	256 KB 型号	
0	0	384	256	0
32	32	320	192	1
64	64	256	128	2
128	128	128	0	3

要调整 ITCM 和 DTCM 的大小，用户需要导航到 MCC 项目图窗口并选择系统模块。根据配置选项中指定的参数配置 ITCM 和 DTCM 设置。

注：PIC32CZ CA70/MC70 系列单片机共享与 SAM E/S/V 系列类似的 TCM 配置。

图 3-3. ITCM 和 DTCM 配置



注：

- 没有显式或额外的存储器 TCM。它是单片机的内部 SRAM 的一部分，即 TCM 和 SRAM。GPNVM 位控制用作 TCM 的存储空间。
- 以处理器速度（HCLK）访问 TCM。对于 SAM E70/S70/V7x 系列，HCLK 最高可达 300 MHz，而无任何等待状态。来自内核的所有访问均配置为单周期访问。因此，TCM 访问的行为具有确定性。
- 对于整个内部 SRAM 的其余部分（不用作 TCM），可在最高 150 MHz 的主时钟（MCK）频率下进行访问。
- 可通过 Cortex-M7 的系统控制块中的寄存器配置来明确使能或禁止 TCM。

4. 软件对 TCM 的影响

4.1. PIC32CZ CA 系列

4.1.1. 复位后的地址映射

PIC32C 系列器件具有引导 ROM，该 ROM 负责管理配置的加载并为安全引导提供信任根。引导 ROM 完成后，它会准备让 CPU 从保存向量表的 BFM 基址开始执行。

根据用户配置中的 FSEQ 的 SEQNUM (15:0) 和引导用户配置中的 BROM_BSEQ 的 SEQNUM (15:0)，相应的闪存面板被映射为最低存储器区域，具体如下。

引导闪存 (BFM)：

- 如果 BFM 面板 1 的序列号 > 面板 2，则将面板 1 映射到 BFM 的起始位置，后跟面板 2。
- 如果 BFM 面板 2 的序列号 > 面板 1，则将面板 2 映射到 BFM 的起始位置，后跟面板 1。

闪存程序存储器 (PFM)：

- 如果 PFM 面板 1 的序列号 > 面板 2，则将面板 1 映射到 PFM 的起始位置，后跟面板 2。
- 如果 PFM 面板 2 的序列号 > 面板 1，则将面板 2 映射到 PFM 的起始位置，后跟面板 1。

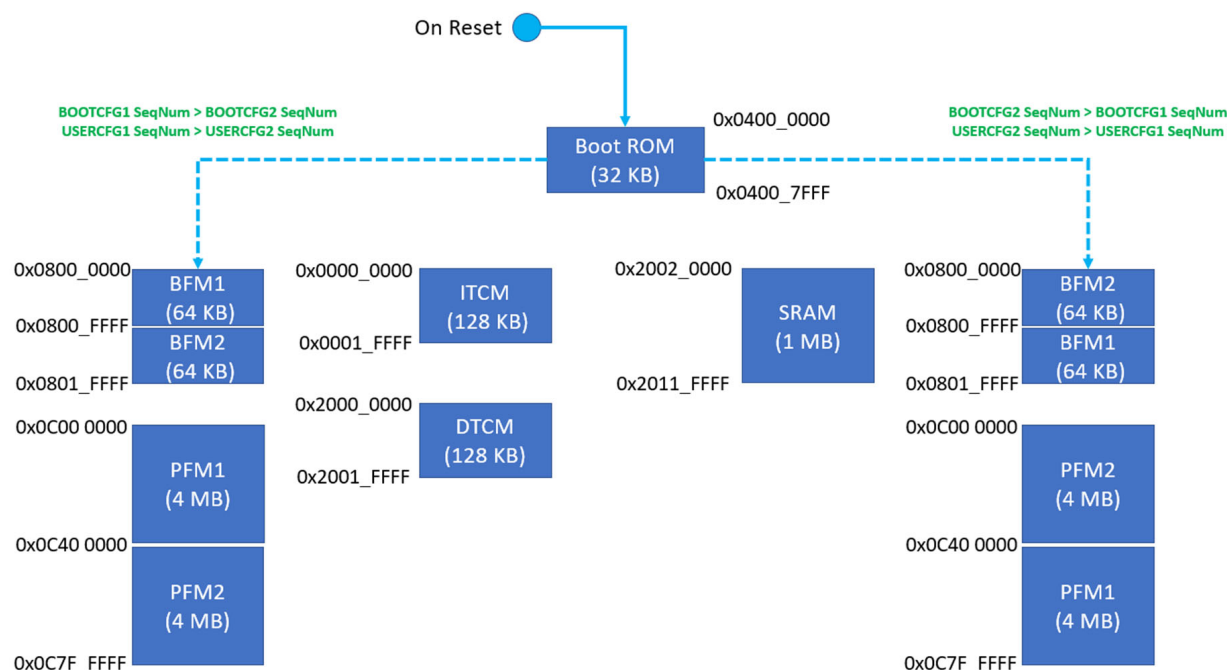
在完全擦除的单片机中，闪存配置的值如下。

- 用户 1 配置中的 FSEQ 的 SEQNUM (15:0) 设置为 0x0001
- 用户 2 配置中的 FSEQ 的 SEQNUM (15:0) 设置为 0x0000
- 引导用户 1 配置中的 BROM_BSEQ 的 SEQNUM (15:0) 设置为 0x0001
- 启动用户 2 配置中的 BROM_BSEQ 的 SEQNUM (15:0) 设置为 0x0000

示例： [PIC32CZ8110CA90208](#) 单片机具有 8 MB 闪存和 1 MB 内部 SRAM。

考虑这样一种情况：USERCFG1_FSEQ -> SEQNUM [15:0] 大于 USERCFG2_FSEQ -> SEQNUM (15:0)，且 BOOTCFG1_BROM_BSEQ -> SEQNUM (15:0) 大于 BOOTCFG2_BROM_BSEQ -> SEQNUM (15:0)。可用引导闪存 (Boot Flash Memory, BFM) 与闪存程序存储器 (Program Flash Memory, PFM) 之间不存在交换。

图 4-1. 复位时的地址“0”映射



4.1.2. ITCM 处于地址 0x0 处

在 PIC32CZ CA 器件中，根据 ARMv7-M 架构，ITCM RAM 区域映射到 0x0000 0000。

仅当设置了位于 0xE000EF90 的 ITCM 控制寄存器 (ITCMCR) 位“0”时，才能访问映射到 ITCM 的地址 0x0。

ITCMCR 寄存器的 bit [0] 可用于使能或禁止 ITCM。

- 当该位置 1 时，Cortex-M7 内核必须通过 ITCM 总线（而非 AXIM 接口）取 ITCM 范围内（在 0x0 与 ITCM 的最后一个地址之间）的所有映射指令。
- 闪存仍具备从 0x0C00 0000 开始的物理存储器地址。
- 在任何复位情况下，ITCMCR 寄存器的位“0”都会清零。
- 向量表和启动代码可位于引导闪存和闪存程序存储器中。

Cortex-M7 内核在复位后通过 AXI 总线执行所有指令访问，但不会访问 TCM 接口，请参见图 3-2. [TCM 接口的 Cortex-M7 架构](#)。但是，将 ITCMCR 配置为使能 ITCM 后，会在地址 0 处通过 TCM 接口（而非 AXI 总线）对 ITCM 进行进一步访问。

4.1.3. 映射 ITCM 和向量表重定位 (VTOR) 寄存器

VTOR 的默认值为 BFM 的基址。如[复位后的地址映射](#)中所述，向量表通常位于同一存储器的起始位置，即地址 0x080_0000 处。在执行期间触发的任何中断均包含从 VTOR 偏移量中正确提取的中断服务程序。

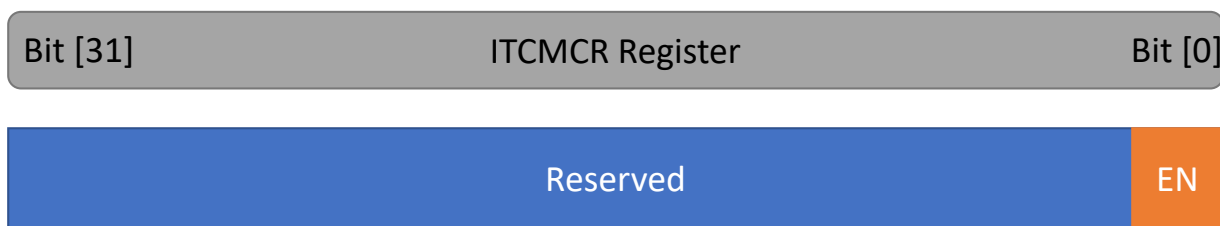
下面给出了重新映射 VTOR 寄存器时的两种可能配置：

- 用闪存中的中断向量表地址写入 VTOR 寄存器。
- 启动后将向量表复制到 ITCM 中，并用此地址更新 VTOR 寄存器。

注：用户还可将任何关键中断程序复制或定位到 ITCM，以提高程序的访问速度并减少服务中断的延时。为了始终以最快速度对时间敏感型事件进行中断响应，向量表和时间敏感型事件处理程序必须位于 ITCM 中。

在更改 VTOR 寄存器（在地址 0xE000ED08 处）后，使能 ITCMCR 寄存器的 bit 0（地址 0xE000EF90 处）。复位时的地址“0”映射一节中介绍了存储器映射。

图 4-2. ITCMCR 寄存器



4.1.4. DTCM

DTCM 大小与 ITCM 大小相同。默认情况下，DTCM 处于使能状态。

如果需要，可使用 DTCMCR 寄存器中的位 0 来禁止或使能 DTCM。该存储器在系统中保持未使用状态。有关更多信息，请参见 *TCM 配置*。

4.1.5. 软件编译

不应动态更改 TCM。链接描述文件中使用的存储器映射必须与 TCM 的 GPNVM 配置匹配。原因在于 TCM 内部的代码或数据并非做镜像处理，而是通过启动代码实际复制到存储器中，且副本程序必须获知 TCM 的边界。

4.2. SAM E70/S70/V7x 系列

4.2.1. 地址 0x0

GPNVM 的 bit “1” 用于选择在地址 0x0000 0000 处镜像的存储器。

- **0:** 内部 ROM: 位于 0x0080 0000 的 ROM 内容将在地址 0 处镜像
- **1:** 内部闪存: 位于 0x0040 0000 的闪存内容将在地址 0 处镜像

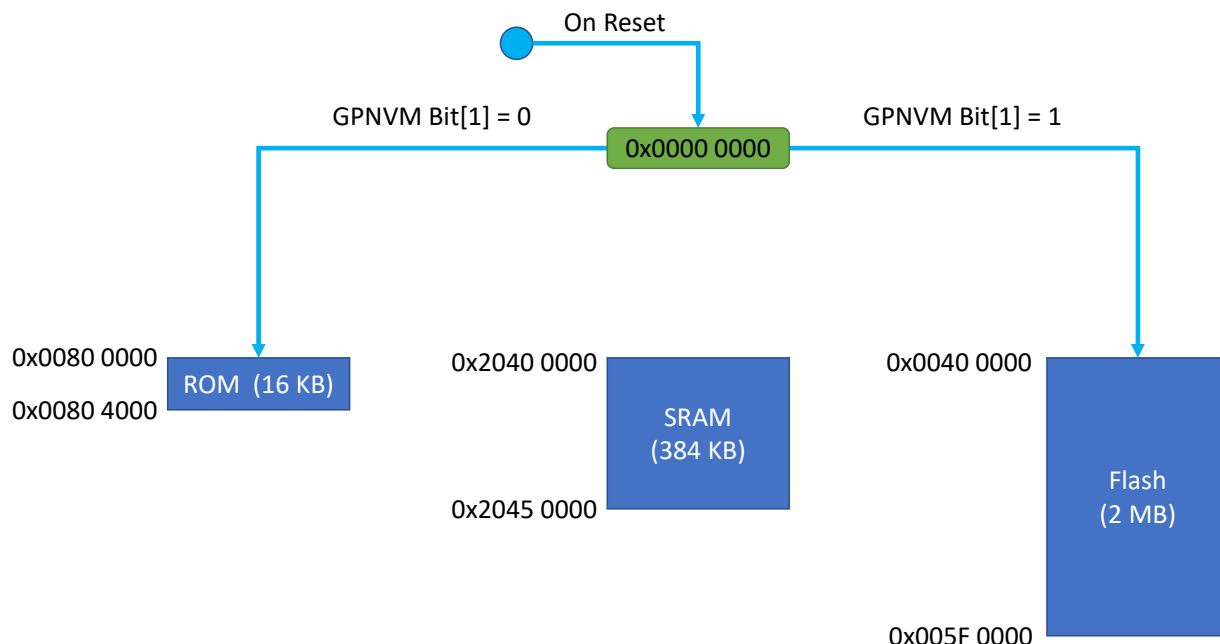
在完全擦除的单片机中，此 GPNVM 位的值设置为 0。该位值可使 ROM 中的引导加载程序代码执行并编程闪存。在闪存中编写有效的应用程序后，GPNVM 的 bit 1 应设置为 1。

注: PIC32CZ CA70/MC70 系列单片机共享与 SAM E/S/V 系列类似的 TCM 配置。

示例: [ATSAME70Q21](#) 单片机，2 MB 闪存，384 KB 内部 SRAM。

考虑 GPNVM (8:7) = b'00 的情况。其中，ITCM = 0 KB；DTCM = 0 KB。可用的 SRAM 空间未变少。

图 4-3. 复位且 GPNVM (8:7) = b'00 时的地址 “0” 映射



4.2.2. 地址 0x0 处的 ITCM

在 SAM E70 中，根据 ARMv7-M 架构，ITCM RAM 区域也映射到 0x0000 0000。

只有以下两个条件均满足时，才能访问映射到 ITCM 的地址 0x0。

- GPNVM (8:7)位设置指示非零 TCM。
- 0X E000EF90 处 ITCM 控制寄存器 (ITCM Control Register, ITCMCR) 的 bit 0 置 1。

ITCMCR 寄存器的 bit [0]可用于使能或禁止 ITCM。

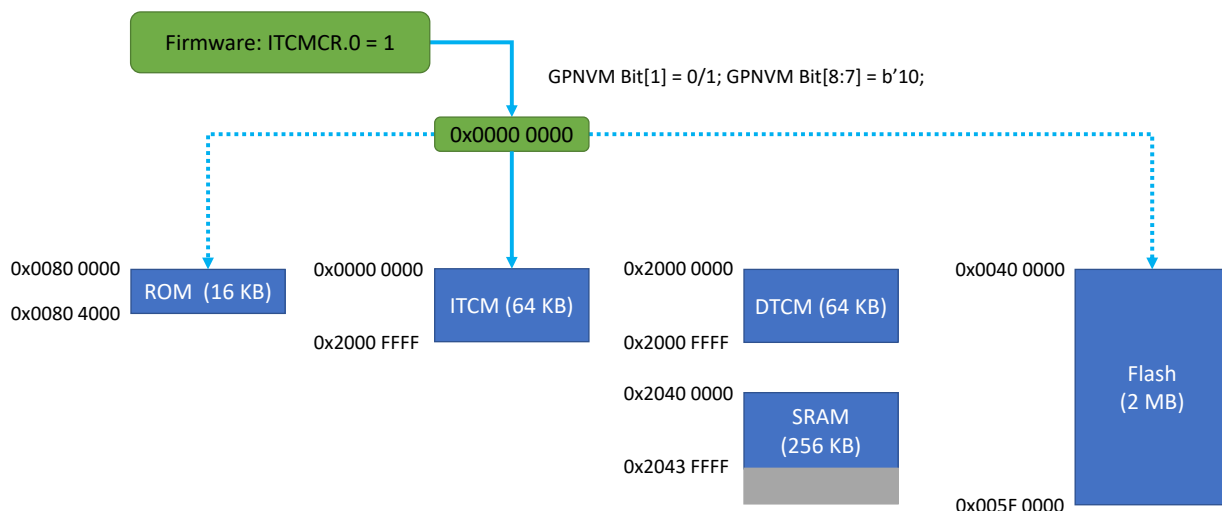
- 当该位置 1 时，Cortex-M7 内核必须通过 ITCM 总线（而非 AXIM 接口）取 ITCM 范围内（在 0x0 与 ITCM 的最后一个地址之间）的所有映射指令。
- 闪存仍具备从 0x0040 0000 开始的物理存储器地址。
- 复位时，ITCMCR 的 bit 0 将清零：无论 GPNVM 位的配置如何，向量表和启动代码均可位于 ROM 或闪存中。

Cortex-M7 内核在复位后通过 AXI 总线执行所有指令访问，但不会访问 TCM 接口，请参见图 3-2.TCM 接口的 Cortex®-M7 架构。因此，即使 GPNVM bit(8:7)非零，复位时的地址 0x0 也会映射到闪存或 ROM。但是，将 ITCMCR 配置为使能 ITCM 后，会在地址 0 处通过 TCM 接口（而非 AXI 总线）对 ITCM 进行进一步访问。

示例：SAMV71Q21 单片机，2 MB 闪存，384 KB 内部 SRAM。

考虑 GPNVM (8:7) = b'10 的情况。其中 ITCM = 64 KB，DTCM = 64 KB。内部 SRAM 减少 128 KB。可用 SRAM 为 256 KB 而非 384 KB。

图 4-4. ITCM 配置中的地址“0”映射



4.2.3. 映射 ITCM 和向量表重定位 (VTOR) 寄存器

VTOR 的默认值为 0x0000_0000。按照地址 0x0 中所述，在不使用 ITCM 的情况下，该地址 0x0000_0000 会镜像闪存的启动。向量表通常位于相同的起始位置，即地址 0x0040_0000 处。在执行期间触发的任何中断均包含从 VTOR 偏移量中正确提取的中断服务程序。

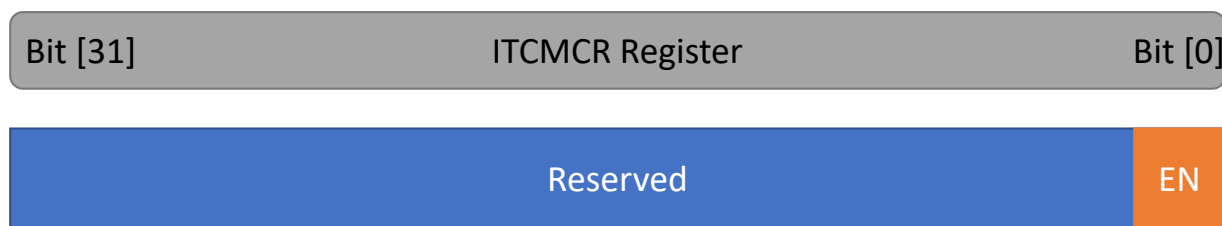
重新映射 VTOR 寄存器时，有两种可能的配置：

- 用闪存中的中断向量表地址写入 VTOR 寄存器。
- 启动后将向量表复制到 ITCM 中，并用此地址更新 VTOR。

注：用户还可将任何关键中断程序复制/定位到 ITCM，以提高程序的访问速度并减少服务中断的延时。为了始终以最快速度对时间敏感型事件进行中断响应，向量表和时间敏感型事件处理程序必须位于 ITCM 中。

在更改 VTOR 寄存器（在地址 0xE000ED08 处）后，使能 ITCMCR 寄存器的 bit 0（地址 0xE000EF90 处）。存储器映射立即更改。新存储器映射将包括 ITCM 的接口，具体如 ITCM 配置中的地址“0”映射中所述。

图 4-5. ITCMCR 寄存器



4.2.4. DTCM

DTCM 大小与 ITCM 大小相同。

基于 GPNVM (8:7)配置的非零 TCM 大小足以使能 DTCM。ITCM 在复位后必须由软件明确使能，DTCM 与之不同，如果其被配置为非零大小则会在复位时使能。

必要时，DTCMCR 的 bit 0 可用于禁止或使能 DTCM，其在系统中保持未使用状态。该存储器在系统中保持未使用状态。需要根据应用需求选择合适的 TCM 配置。有关更多信息，请参见 TCM 配置。

4.2.5. 软件编译

不应动态更改 TCM。链接描述文件中使用的存储器映射必须与 TCM 的 GPNVM 配置匹配。原因如下：

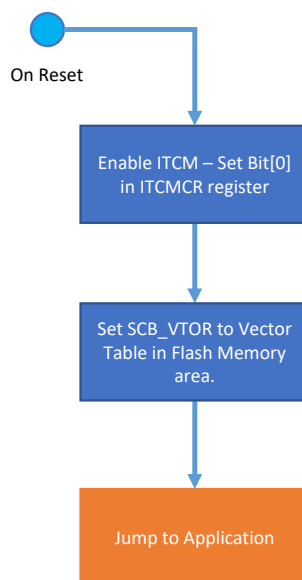
- TCM 内部的代码/数据不会被镜像，实际上会被复制到存储器的启动代码中：
 - 副本程序必须了解 TCM 的边界。
- TCM 的配置导致内部可用 SRAM 相应减少：
 - TCM 分区位于 SRAM 内部。可用 SRAM 容量取决于所选的 TCM 配置。
 - 配置 TCM 时，闪存下载程序必须清楚可用 SRAM 大小。
- 为执行更改而更改 GPNVM 配置位时，必须触发复位。

5. TCM 配置的编程序列

5.1. PIC32CZ CA 系列

PIC32CZ CA 系列中的 TCM 存储器配置如下所示。

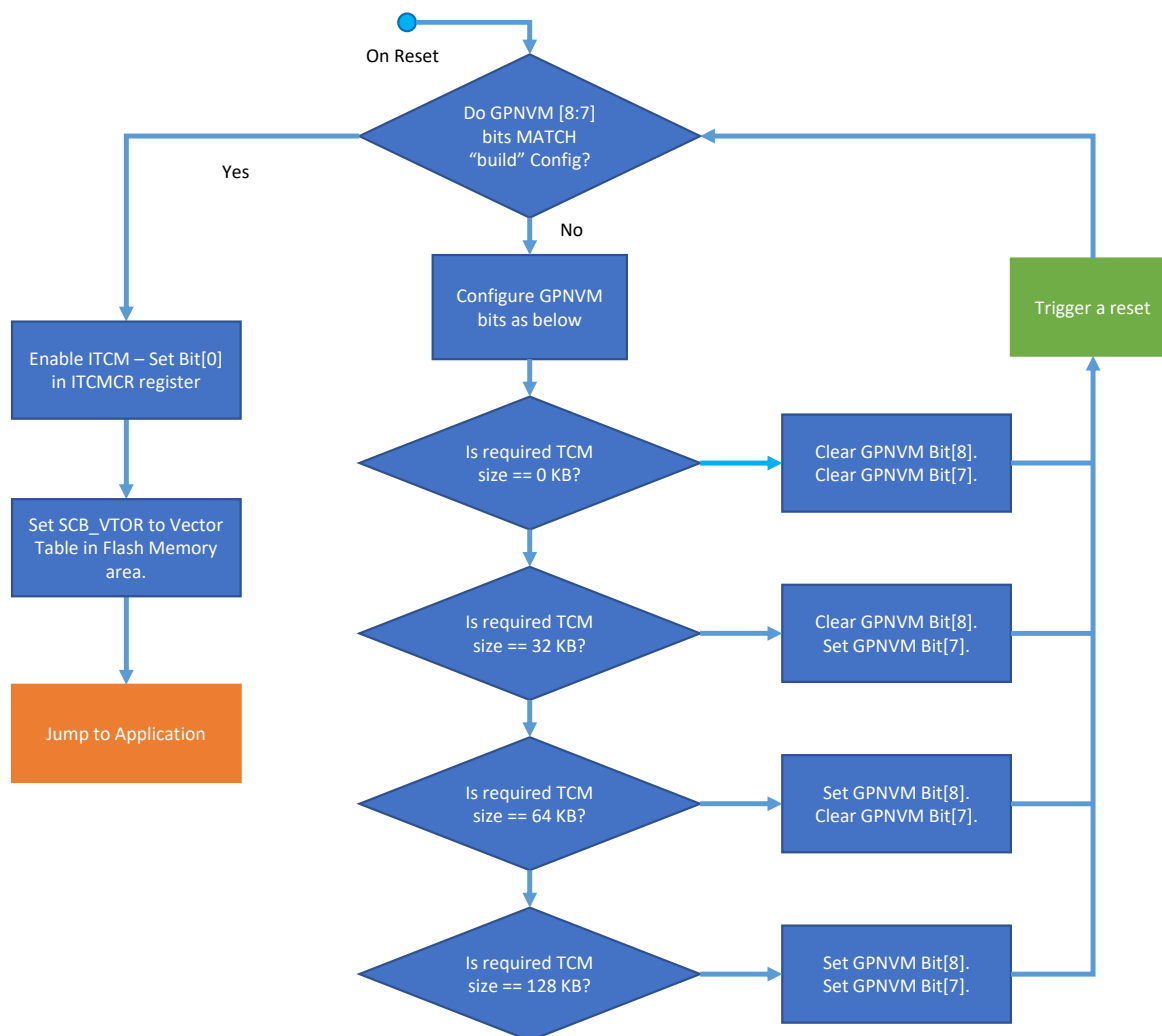
图 5-1. 软件中 PIC32CZ CA 系列的 TCM 配置



5.2. SAM E70/S70/V7x 系列

针对 TCM 配置对 GPNVM 位配置序列进行了说明。

图 5-2. 软件中 SAM E/S/V 系列的 TCM 配置



注：PIC32CZ CA70/MC70 系列单片机共享与 SAM E/S/V 系列类似的 TCM 配置。

6. 硬件和软件要求

以下各节介绍了本应用程序使用的硬件和软件前提条件并对应用程序进行了概述。

6.1. 准备工作

硬件：

- [PIC32CZ CA90 Curiosity Ultra 开发板](#)
- [SAME70 Xplained Ultra Pro 评估工具包](#)

软件：

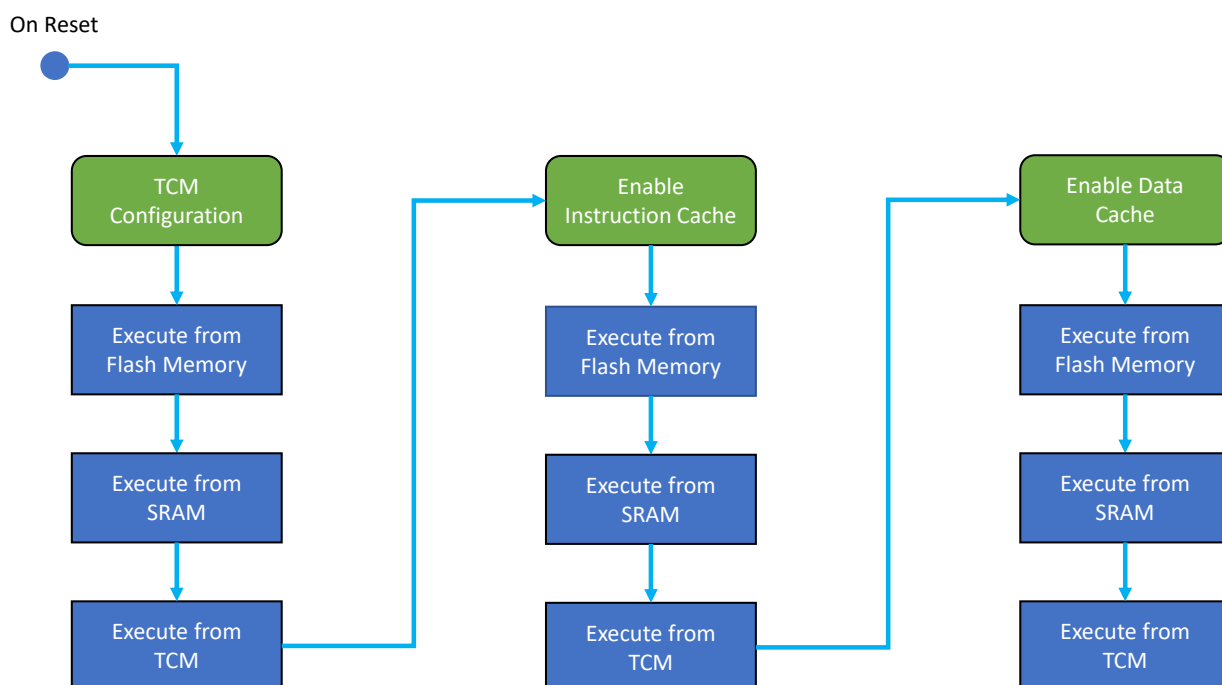
- [MPLAB X IDE v6.20](#) 或更高版本
- [MPLAB XC32 编译器 v4.45](#) 或更高版本
- [MPLAB 代码配置器 v5.5.1](#) 或更高版本
 - [csp v3.19.7](#) (MPLAB Harmony v3 芯片支持包) 或更高版本

注：本文档中讨论的应用程序使用上述 MPLAB Harmony v3 资源库。可以从 GitHub 下载这些资源库或使用 [MCC 内容管理器](#) 下载上述资源库。这些步骤也可以用于 PIC32CZ CA70/MC70 系列单片机。

6.2. 应用概览

此示例说明了 PIC32CZ CA 和 SAM E70/S70/V70/V71 系列器件中 TCM 的用法。它比较了在有或没有 I-cache 和 D-cache 的情况下从闪存、SRAM 和 TCM 执行代码所需的时间。

图 6-1. 流程图



6.3. 应用程序代码

为了说明如何从存储器位置执行代码，在闪存、SRAM 和 TCM 上创建了一个简单的数据传输函数。然后，借助内核周期计数器比较每次数据传输所花费的时间。使用 MPLAB XC32 编译器属性，将函数和数据缓冲区放置在不同的存储器位置。

- `__attribute__((tcm))`: 将函数或缓冲区放置在 TCM 中
- `__attribute__((ramfunc, long_call))`: 将函数或缓冲区放置在 SRAM 中

在 main.c 文件中:

- 在 DTCM 中定义并放置两个缓冲区 TCM_SrcBuff 和 TCM_DestBuff
- 在 SRAM 中定义并放置两个缓冲区 SrcBuff 和 DestBuff
- 函数 TCM_memcpy() 位于 ITCM 中
- 函数 SRAM_memcpy() 位于 SRAM 中
- 函数 flash_memcpy() 位于闪存中

7. 编译并编程应用程序

按照以下步骤对 PIC32CZ CA 和 SAM E70/S70/V70/V71 系列器件的 TCM 性能应用程序进行编程。

- 从以下链接下载 TCM 性能应用程序：
 - PIC32CZ CA90 Curiosity Ultra 评估工具包（[点击下载](#)）或从 [GitHub](#) 获取
 - SAM E70 Xplained Ultra 评估工具包（[点击下载](#)）或从 [GitHub](#) 获取
- 从 MPLAB X IDE 打开 PIC32CA CA90 TCM 性能项目（路径：<已下载路径>/*pic32cz_ca90_tcm_performance/firmware/tcm_performance_pic32cz_ca90_cult.X*）。
注：为 SAM E70 Xplained Ultra 评估工具包选择 SAM E70 TCM 性能项目（路径：<已下载路径>/*same70_tcm_performance/firmware/tcm_performance_same70_xult.X*）。
- PIC32CZ CA90 Curiosity Ultra 开发板上配有用于调试的板载调试器（PKoB4）。将 Type-A 公头转 micro-B USB 线缆连接到 micro-B DEBUG USB 端口，以便对 PIC32CZ CA90 Curiosity Ultra 评估板进行供电和调试。将外部电源（6.5V 至 14V）连接到 J100 连接器。

图 7-1. PIC32CZ CA90 Curiosity Ultra 评估板硬件设置

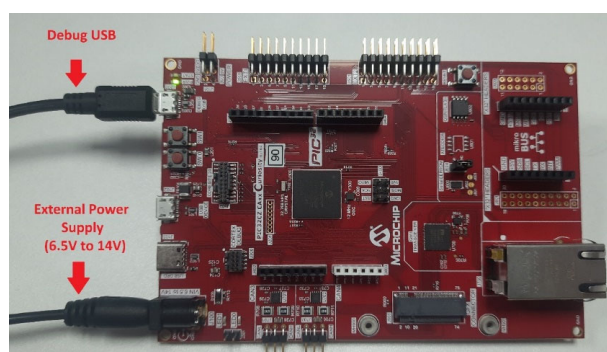
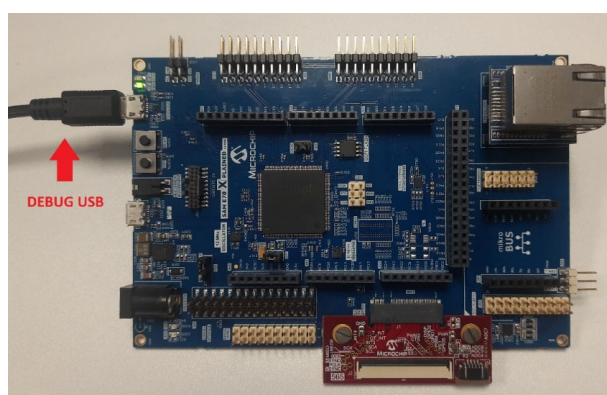


图 7-2. SAM E70 Xplained Ultra 评估工具包硬件设置

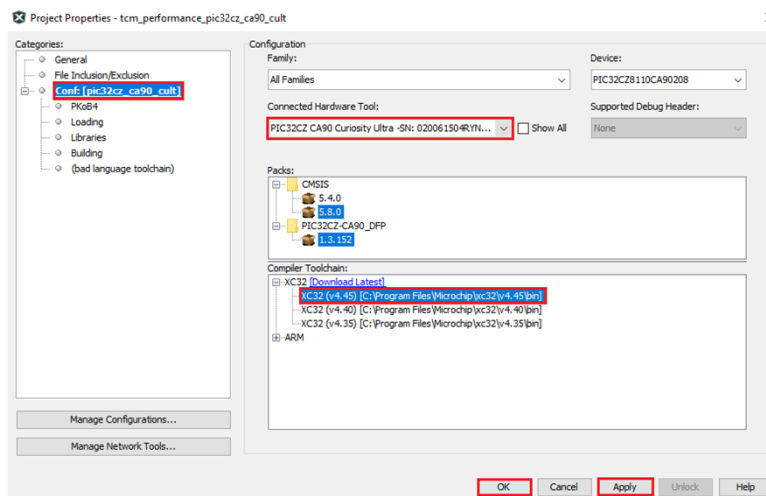


注：将 Type-A 公头转 micro-B USB 线缆连接到 micro-B DEBUG USB 端口，以便仅对 SAM E70 Xplained Ultra 评估工具包进行供电和调试。

- 在 MPLAB X IDE 的 **Project Properties**（项目属性）窗口中：
 - 在左侧的 **Categories**（类别）窗格中，选择 **Conf: [pic32cz_ca90_cult]**。
 - 在右侧的 **Configuration**（配置）属性页面中：
 - 对于已连接的硬件工具，选择 **PIC32CZ CA90 Curiosity Ultra**。

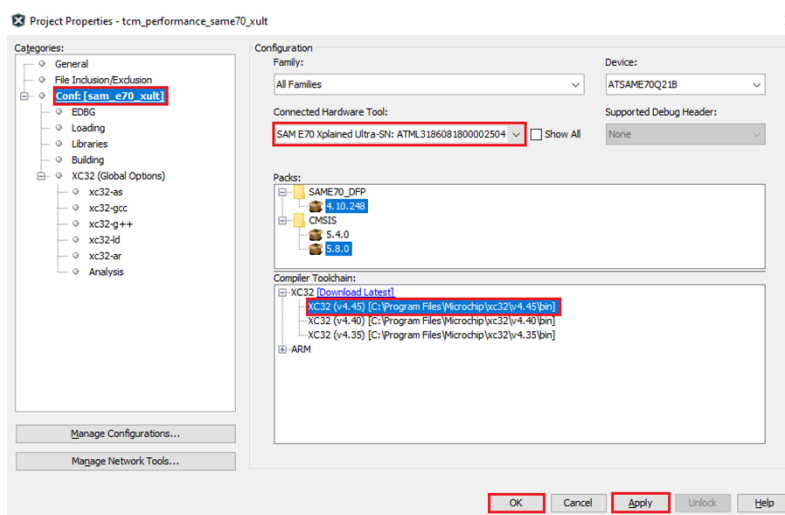
- ii. 对于编译器工具链，单击并展开 XC32，然后选择 **XC32 (V4.45)**。

图 7-3. 项目属性——PIC32CZ CA90 Curiosity Ultra 评估工具包



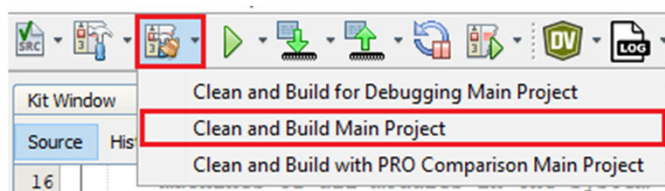
注：用户需要为 SAM E70 Xplained Ultra 评估工具包选择 Conf: [sam_e70_xult]，如下所示。

图 7-4. 项目属性——SAM E70 Xplained Ultra 评估工具包



- 单击 **Apply**（应用），然后单击 **OK**（确定）。
- 若要编译项目，请单击 **Clean and Build**（清除并编译）图标或从下拉列表中选择 **Clean and Build Main Project**（清除并编译主项目），然后验证项目是否成功编译。至此，即可实现应用程序代码。

图 7-5. 清除并编译项目



7. 单击 **Program the Device**（编程器件）图标对应用程序进行编程。

图 7-6. 编程器件



8. 观察 MPLAB 数据可视化器上的输出

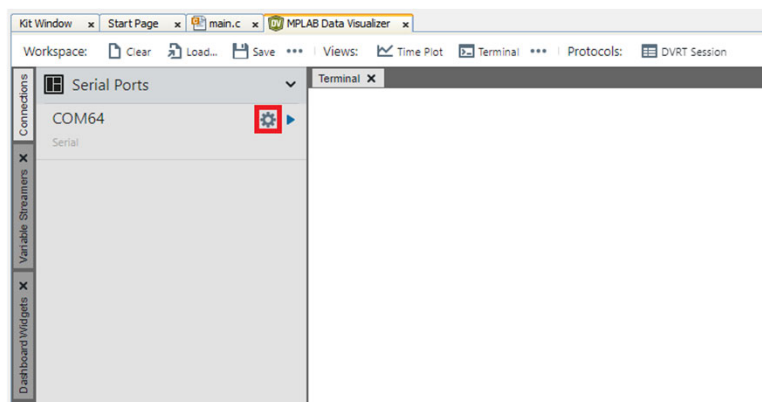
1. 构建和编程应用程序后，单击突出显示的图标打开 MPLAB 数据可视化器。

图 8-1. 启动 MPLAB 数据可视化器



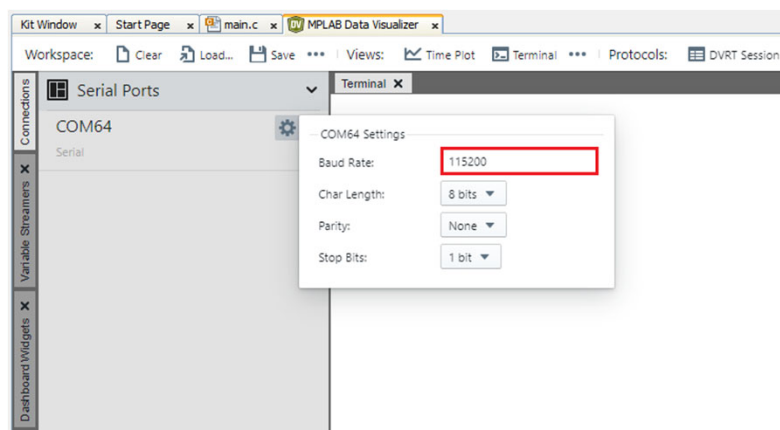
2. 要配置 PIC32CZ CA90 Curiosity Ultra 评估板的串行端口，单击下图所示的齿轮图标：

图 8-2. 串行端口设置



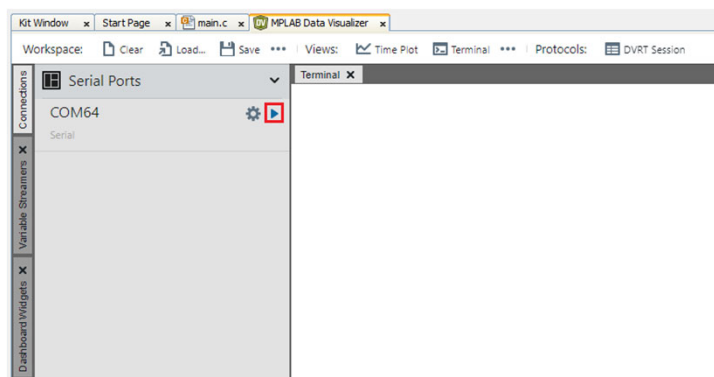
3. 在 **COM Settings**（COM 设置）属性表，在 Baud Rate（波特率）字段中输入 **115200**。

图 8-3. 选择波特率



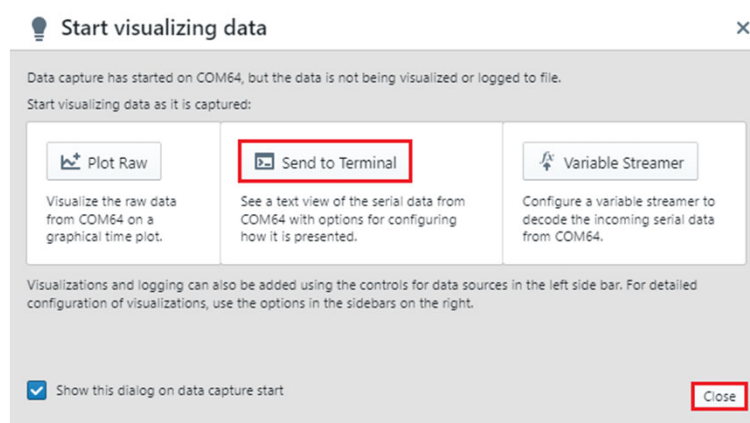
4. 通过单击 **Play**（播放）图标打开 PIC32CZ CA90 Curiosity Ultra 评估板的串行端口。

图 8-4. 打开串行端口



5. 选择 **Send to Terminal**（发送到终端）以可视化 TCM 性能测试结果，然后单击 **Close**（关闭）。

图 8-5. 选择终端选项



6. 按下评估板上的**复位**按钮以复位器件。

图 8-6. 复位 PIC32CZ CA90 Curiosity Ultra 评估板

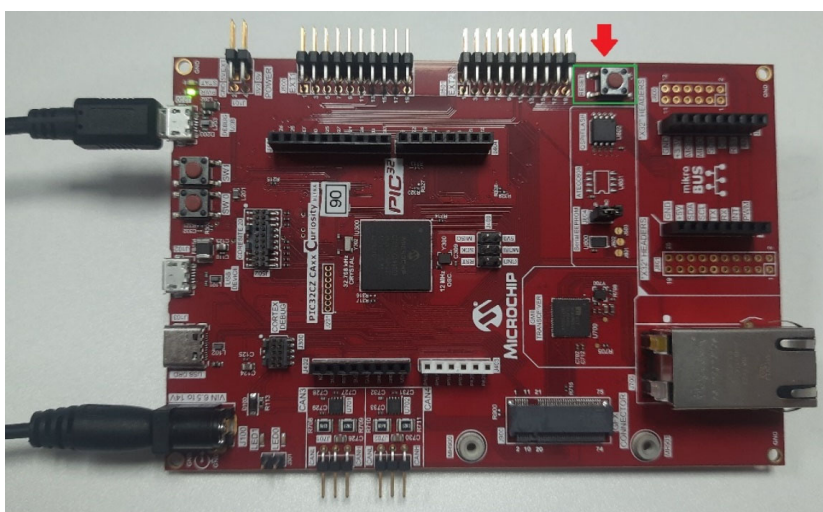
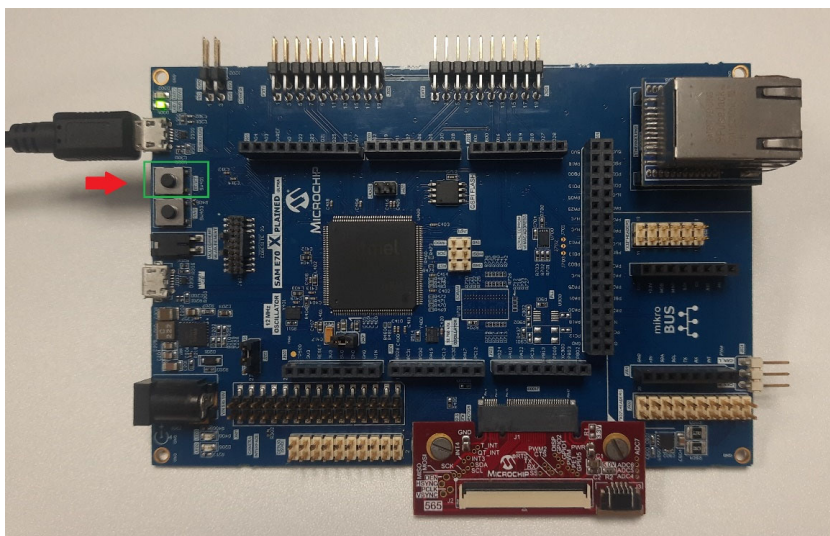


图 8-7. 复位 SAM E70 Xplained Ultra 评估工具包



7. 观察 MPLAB 数据可视化器上显示的控制台消息 **TCM Performance Demo**（TCM 性能演示）。

图 8-8. TCM 性能比较

```
Terminal X
-----
TCM Performance Demo
-----

Disabled Instruction & Data Cache
Code/Buffer - Cycles
Flash/SRAM - 1085171
Flash/DTCM - 988133
SRAM/SRAM - 955135
SRAM/DTCM - 876161
ITCM/SRAM - 916673
ITCM/DTCM - 834131

Enabled Instruction Cache
Code/Buffer - Cycles
Flash/SRAM - 1035921
Flash/DTCM - 938207
SRAM/SRAM - 944315
SRAM/DTCM - 853327
ITCM/SRAM - 898183
ITCM/DTCM - 834117

Enabled Instruction & Data Cache
Code/Buffer - Cycles
Flash/SRAM - 166569
Flash/DTCM - 163984
SRAM/SRAM - 163995
SRAM/DTCM - 163982
ITCM/SRAM - 163981
ITCM/DTCM - 163981
```

注：

a. 两个项目的优化级别均设置为“无/O0”。

b. 下表提供了针对 PIC32CZ CA 和 SAM E70/S70/V70/V71 系列器件收集的执行指标：

- [PIC32CZ CA: TCM 存储器配置——128 KB，优化——无/O0](#)
- [SAM E70/S70/V70/V71: TCM 存储器配置——32 KB，优化——速度/O0](#)

8. 下表列出了 ITCM 和 DTCM 与传统闪存/SRAM 组合相比的优势。值得注意的是，使能高速缓存时的优势很小，因为示例程序非常小且可完全装入高速缓存。然而，实际应用程序要复杂得多，并且无法完全包含在高速缓存中。对于此类应用程序，TCM 在管理关键和确定性代码段方面被证明更具优势。

表 8-1. PIC32CZ CA: TCM 配置——128 KB，优化——无/O0

配置	代码	缓冲器	周期数
指令高速缓存 = 关闭; 数据高速缓存 = 关闭;	闪存	SRAM	1085171
	闪存	DTCM	988133
	SRAM	SRAM	955135
	SRAM	DTCM	876161
	ITCM	SRAM	916673
	ITCM	DTCM	834131
指令高速缓存 = 打开; 数据高速缓存 = 关闭;	闪存	SRAM	1035921
	闪存	DTCM	938207
	SRAM	SRAM	944315
	SRAM	DTCM	853327
	ITCM	SRAM	898183
	ITCM	DTCM	834117
指令高速缓存 = 打开; 数据高速缓存 = 打开;	闪存	SRAM	166569
	闪存	DTCM	163984
	SRAM	SRAM	163995
	SRAM	DTCM	163982
	ITCM	SRAM	163981
	ITCM	DTCM	163981

表 8-2. SAM E70/S70/V70/V71: TCM 配置——32 KB，优化——无/O0

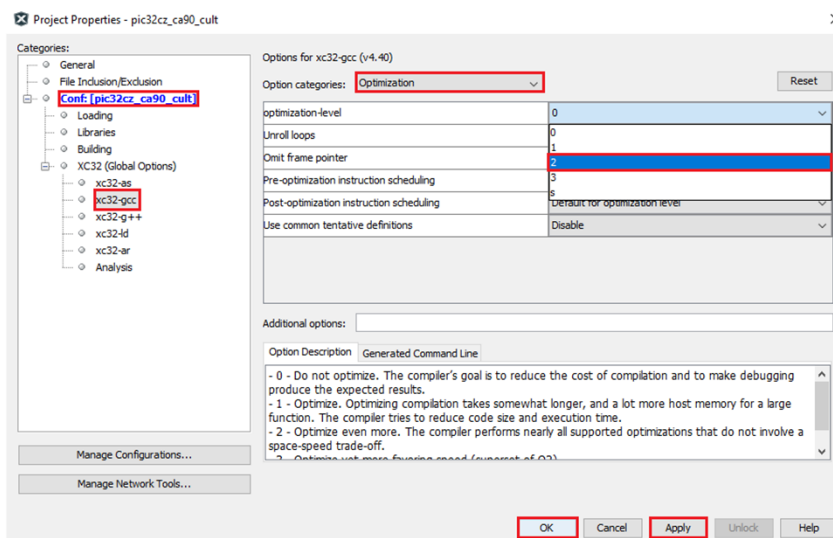
配置	代码	缓冲器	周期数
指令高速缓存 = 关闭; 数据高速缓存 = 关闭;	闪存	SRAM	1317577
	闪存	DTCM	1258257
	SRAM	SRAM	1186391
	SRAM	DTCM	1064289
	ITCM	SRAM	1039041
	ITCM	DTCM	964571
指令高速缓存 = 打开; 数据高速缓存 = 关闭;	闪存	SRAM	1266171
	闪存	DTCM	1161249
	SRAM	SRAM	1169437
	SRAM	DTCM	1051603
	ITCM	SRAM	1052657
	ITCM	DTCM	963431

表 8-2. SAM E70/S70/V70/V71: TCM 配置——32 KB, 优化——无/00 (续)

配置	代码	缓冲器	周期数
指令高速缓存 = 打开; 数据高速缓存 = 打开;	闪存	SRAM	167639
	闪存	DTCM	163973
	SRAM	SRAM	164019
	SRAM	DTCM	164034
	ITCM	SRAM	163971
	ITCM	DTCM	163973

9. 为了增强这些数据，将优化级别调整为 O2 以观察改进情况：
 - a. 在 MPLAB X IDE **Project Properties** 窗口中，选择 **Conf: [pic32cz_ca90_cult] > XC32 (Global Options) (XC32 (全局选项))**，然后选择 **xc32-gcc**。
 - b. 在右侧 **Options for xc32-gcc** (xc32-gcc 的选项) 属性表中：
 - i. 对于 Option (选项) 类别，选择 **Optimization** (优化)。
 - i. 对于 Optimization 级别，选择 **2**。

图 8-9. 项目属性



10. 单击 **Apply**，然后单击 **OK**。
11. 要重新编译并编程 PIC32CZ CA90 Curiosity Ultra 评估板，请参见 [编译并编程应用程序](#)。
- 注：**
- 下表提供了针对 PIC32CZ CA 和 SAM E70/S70/V70/V71 系列收集的执行指标：
 - [PIC32CZ CA: TCM 存储器配置——128 KB，优化——O2](#)
 - [SAM E70/S70/V70/V71: TCM 存储器配置——32 KB，优化——O2](#)
 - 上述步骤可用于更改 SAM E70 Xplained Ultra 评估工具包的优化级别。

表 8-3. PIC32CZ CA: TCM 配置——128 KB，优化——O2

配置	代码	缓冲器	周期数
指令高速缓存 = 关闭; 数据高速缓存 = 关闭;	闪存	SRAM	383503
	闪存	DTCM	385553
	SRAM	SRAM	179778
	SRAM	DTCM	134154
	ITCM	SRAM	181595
	ITCM	DTCM	133951
指令高速缓存 = 打开; 数据高速缓存 = 关闭;	闪存	SRAM	179485
	闪存	DTCM	136405
	SRAM	SRAM	179496
	SRAM	DTCM	133842
	ITCM	SRAM	181515
	ITCM	DTCM	133851
指令高速缓存 = 打开; 数据高速缓存 = 打开;	闪存	SRAM	45442
	闪存	DTCM	45203
	SRAM	SRAM	39094
	SRAM	DTCM	42650
	ITCM	SRAM	41132
	ITCM	DTCM	42640

表 8-4. SAM E70/S70/V70/V71: TCM 配置——32 KB，优化——O2

配置	代码	缓冲器	周期数
指令高速缓存 = 关闭; 数据高速缓存 = 关闭;	闪存	SRAM	394969
	闪存	DTCM	397011
	SRAM	SRAM	215731
	SRAM	DTCM	159931
	ITCM	SRAM	215575
	ITCM	DTCM	159709
指令高速缓存 = 打开; 数据高速缓存 = 关闭;	闪存	SRAM	214951
	闪存	DTCM	163591
	SRAM	SRAM	216901
	SRAM	DTCM	161029
	ITCM	SRAM	216859
	ITCM	DTCM	161013
指令高速缓存 = 打开; 数据高速缓存 = 打开;	闪存	SRAM	47036
	闪存	DTCM	45180
	SRAM	SRAM	41120
	SRAM	DTCM	42636
	ITCM	SRAM	41101
	ITCM	DTCM	42633

下面各图显示了 PIC32CZ CA 和 SAM E70/S70/V70/V71 系列器件上的 TCM 性能。

图 8-10. PIC32CZ CA: 禁止指令高速缓存和数据高速缓存

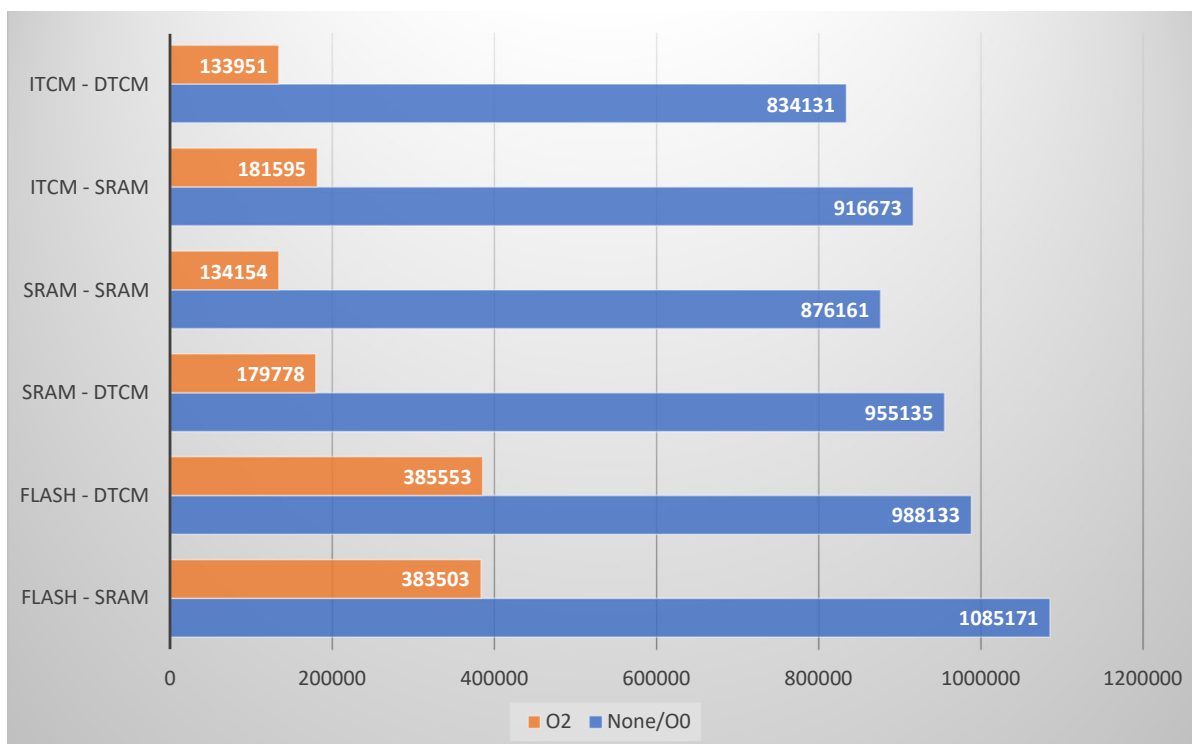


图 8-11. PIC32CZ CA: 使能指令高速缓存

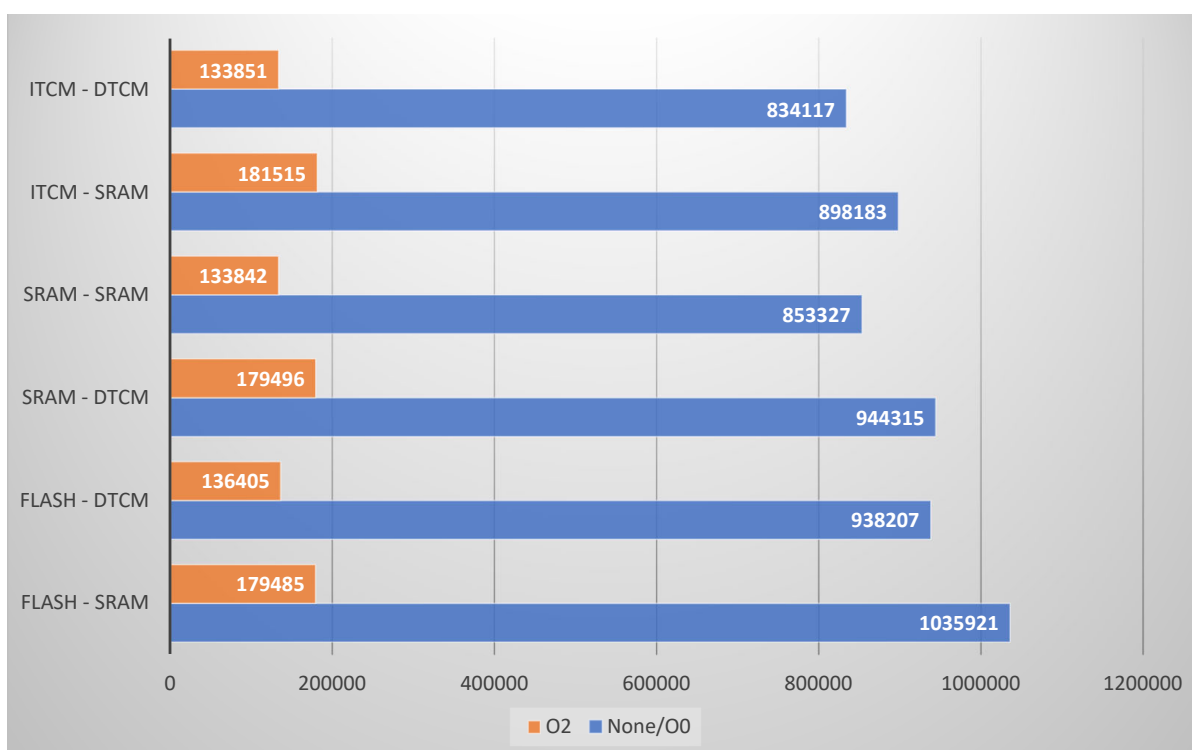


图 8-12. PIC32CZ CA: 使能指令高速缓存和数据高速缓存

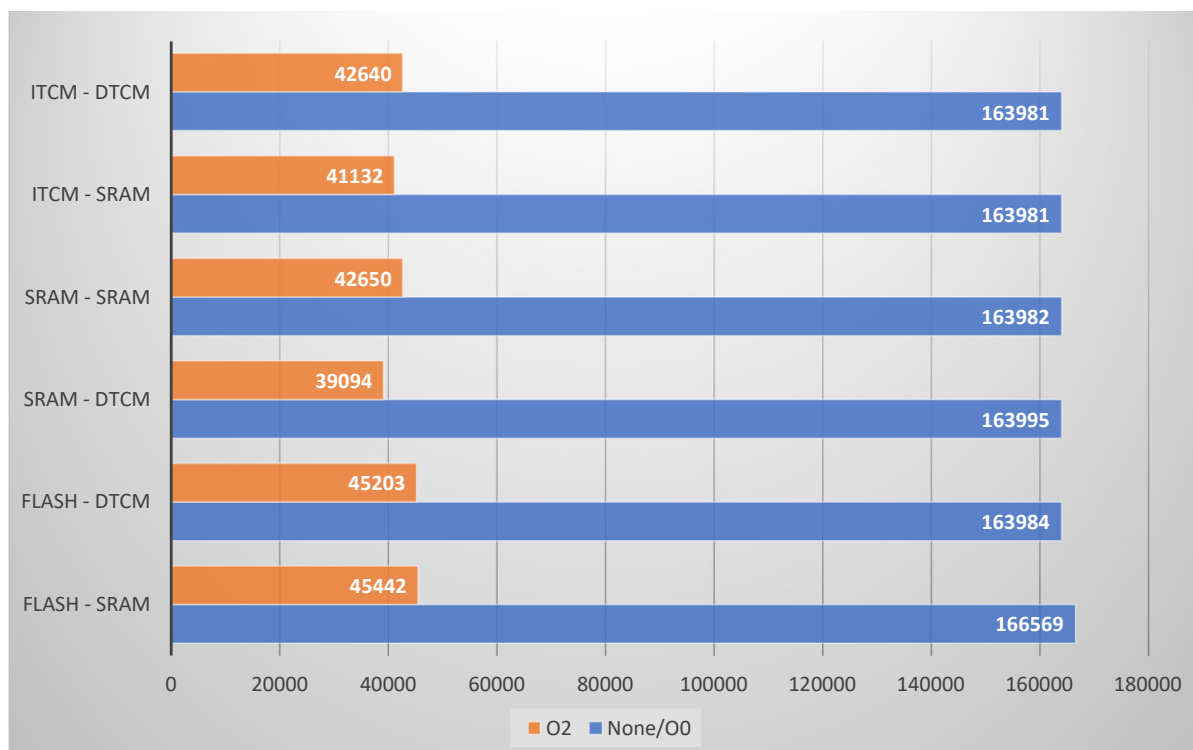


图 8-13. SAM E70/S70/V70/V71: 禁止指令高速缓存和数据高速缓存

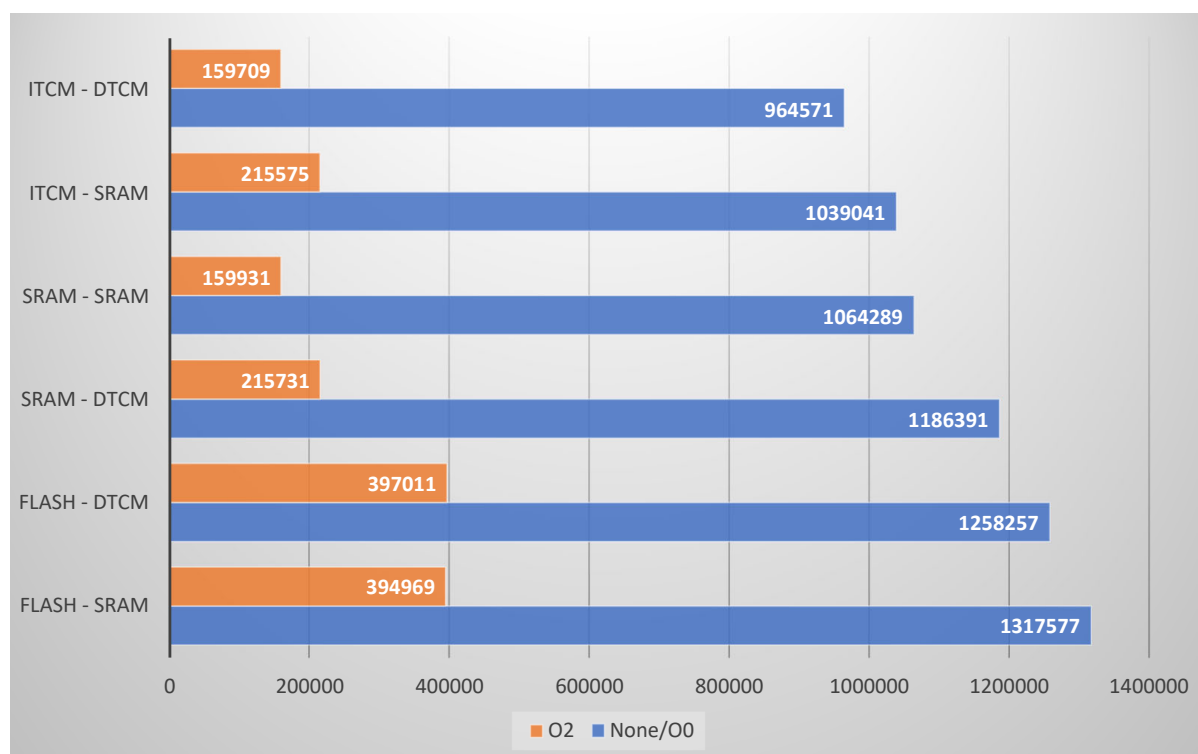


图 8-14. SAM E70/S70/V70/V71: 使能指令高速缓存

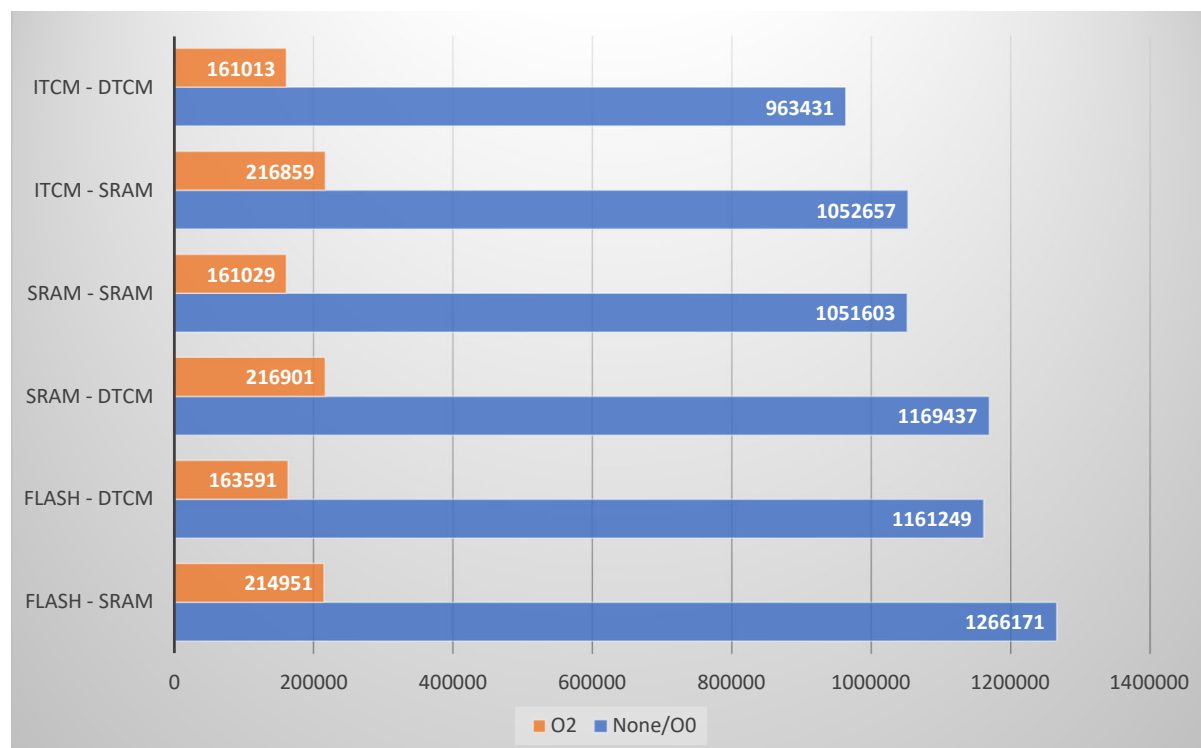
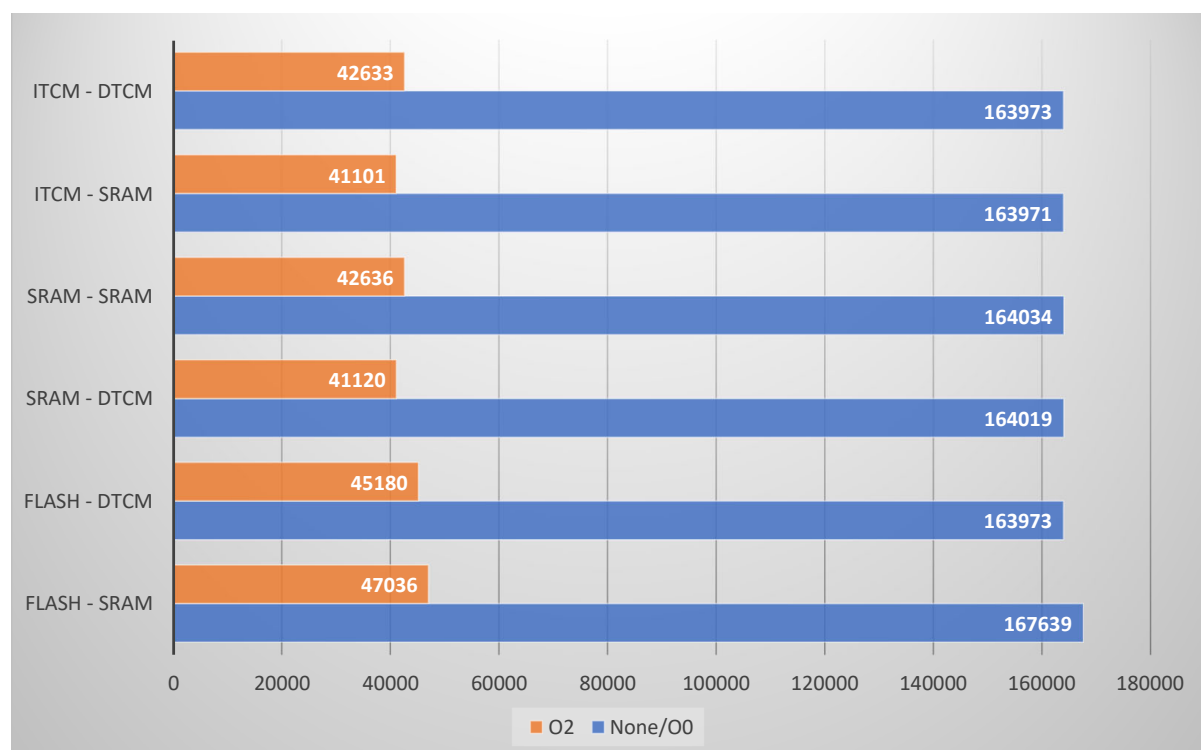


图 8-15. SAM E70/S70/V70/V71: 使能指令高速缓存和数据高速缓存



9. 资源

- [PIC32CZ CA90 Curiosity Ultra 开发板](#)
- [PIC32CZ CA80/CA90 Curiosity Ultra 用户指南](#)
- [使用 MPLAB® Harmony v3 和 MPLAB 代码配置器（MCC）在 PIC32CZ CAx 单片机上创建第一个应用程序](#)
- [SAME70 Xplained Ultra Pro 评估工具包](#)
- [SAM E70 Xplained Ultra 用户指南](#)
- [使用 MPLAB® Harmony v3 基于 SAM E70 创建您的第一个项目](#)
- 有关 MPLAB Harmony v3 的更多信息，请访问 Microchip 网站：
<https://www.microchip.com/mplab/mplab-harmony and microchipdeveloper.com/xwiki/bin/view/software-tools/harmony/>
- 有关各种应用的更多信息，请参见：
github.com/Microchip-MPLAB-Harmony/reference_apps
- 有关 32 位单片机资料和解决方案的更多信息，请参见：
<https://www.microchip.com.cn/newcommunity/Uploads/202404/6620805f4a70b.pdf>

10. 缩写

下列缩写适用于本文档：

表 10-1. 缩写

TCM	紧耦合存储器 (Tightly Coupled Memory)
ITCM	指令紧耦合存储器 (Instruction Tightly Coupled Memory)
DTCM	数据紧耦合存储器 (Data Tightly Coupled Memory)
ECC	纠错码 (Error Correction Code)
MPU	存储器保护单元 (Memory Protection Unit)
MCC	MPLAB® 代码配置器 (MPLAB Code Configurator)
USB	通用串行总线 (Universal Serial Bus)
I²S	内部集成电路音频总线 (Inter-IC Sound)
QSPI	四通道串行外设接口 (Quad-Serial Peripheral Interface)
SRAM	静态随机访问存储器 (Static Random Access Memory)
ROM	只读存储器 (Read Only Memory)
DMA	直接存储器访问 (Direct Memory Access)
AHBS	高级高性能总线客户端 (Advanced High-Performance Bus Client)
AXIM	高级可扩展接口管理器 (Advanced eXtensible Interface Manager)
CPU	中央处理单元 (Central Processing Unit)
TCU	紧耦合接口单元 (Tightly Coupled Interface Unit)
LSU	装载存储单元 (Load Store Unit)
PFU	预取单元 (Pre-Fetch Unit)
DGU	调试单元 (Debug Unit)
HCLK	主器件时钟 (Host Clock)
MCK	主时钟 (Main Clock)
BFM	引导闪存 (Boot Flash Memory)
PFM	闪存程序存储器 (Program Flash Memory)
ITCMCR	指令紧耦合存储器控制寄存器 (ITCMCR Instruction Tightly Coupled Memory Control Register)
VTOR	向量表偏移寄存器 (Vector Table Offset Register)
SEQNUM	序列号 (Sequence Number)
GPVM	通用非易失性存储器 (General-Purpose Non-Volatile Memory)
KB	千字节 (Kilo Bytes)

11. 版本历史

版本 A——2024 年 12 月

本文档的初始版本。

Microchip 信息

商标

“Microchip”名称和徽标、“M”徽标及其他名称、徽标和品牌均为 Microchip Technology Incorporated 或其关联公司和/或子公司在美国和/或其他国家/地区的注册商标和未注册商标（“Microchip 商标”）。有关 Microchip 商标的信息，可访问 <https://www.microchip.com/en-us/about/legal-information/microchip-trademarks>。

ISBN: 979-8-3371-1371-5

法律声明

提供本文档的中文版本仅为了便于理解。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。

本出版物及其提供的信息仅适用于 Microchip 产品，包括设计、测试以及将 Microchip 产品集成到您的应用中。以其他任何方式使用这些信息都将被视为违反条款。本出版物中的器件应用信息仅为您提供便利，将来可能会发生更新。您须自行确保应用符合您的规范。如需额外的支持，请联系当地的 Microchip 销售办事处，或访问 www.microchip.com/en-us/support/design-help/client-support-services。

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。

Microchip 器件代码保护功能

请注意以下有关 Microchip 产品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：在正常使用且符合工作规范的情况下，Microchip 系列产品非常安全。
- Microchip 注重并积极保护其知识产权。严禁任何试图破坏 Microchip 产品代码保护功能的行为，这种行为可能会违反《数字千年版权法案》（Digital Millennium Copyright Act）。
- 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。