

---

## LAN9252 SDK 固件 API 指南

---

作者: <i>Riyas Kattukandan</i> <i>Microchip Technology Inc.</i>
--

### 1.0 简介

本文档介绍用于将 Microchip LAN9252 EtherCAT® 从器件控制器 (EtherCAT Slave Controller, ESC) 与 PIC32MX 相集成的固件 API, 以便开发人员移植基于 LAN9252 开发的 EtherCAT 从器件协议栈应用程序。

### 1.1 术语和缩写

- ETG——EtherCAT 技术协会
- ESC——EtherCAT 从器件控制器
- EVB——工程设计验证板
- HAL——硬件抽象层
- HBI——主机总线接口
- IDE——集成开发环境
- PDI——过程数据接口
- SDK——软件开发工具包
- SPI——串行协议接口
- SQI——串行四线接口
- SSC——从器件协议栈代码

### 1.2 参考资料

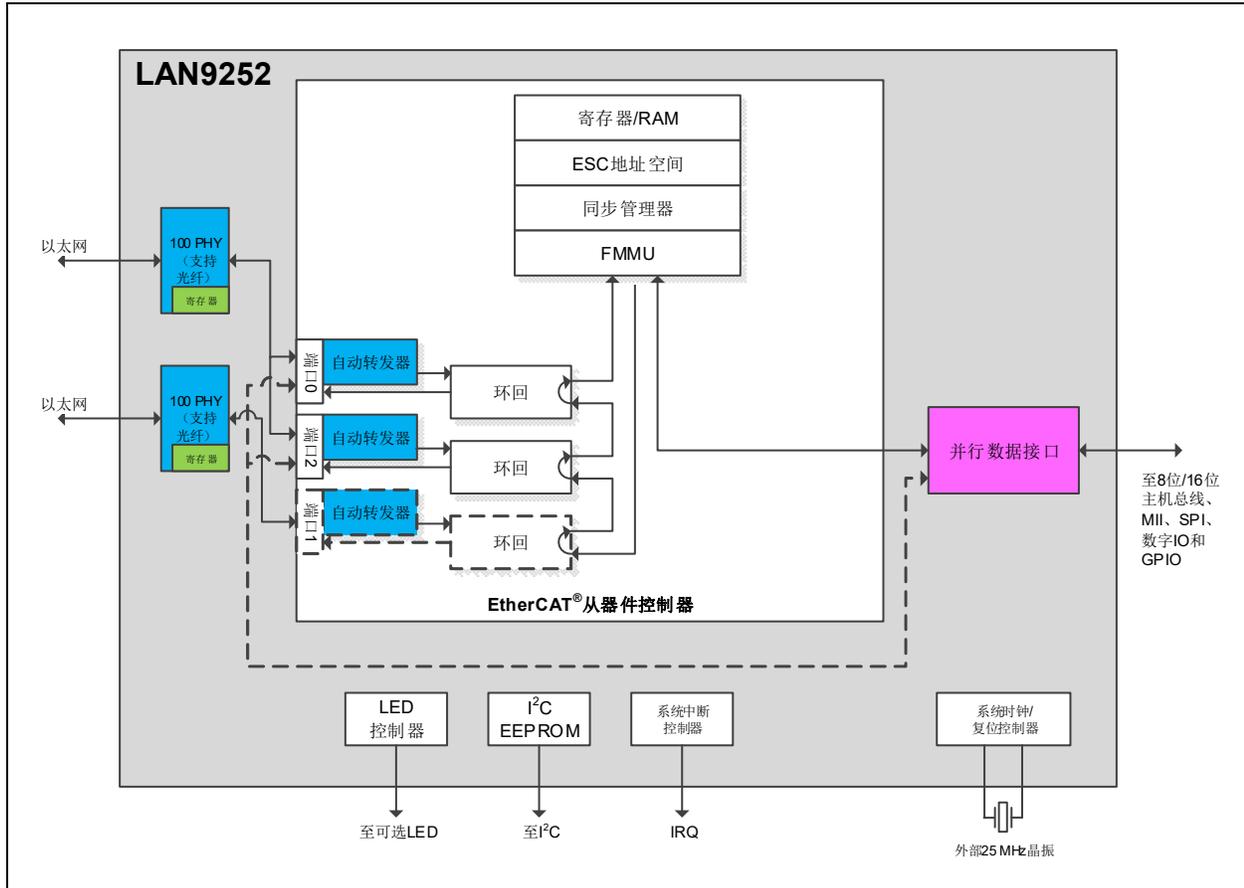
使用本应用笔记时, 应参考以下文档。请咨询您的 Microchip 代表以获取这些文档。

- *LAN9252 Data Sheet*
- AN1907——《从 Beckhoff ET1100 到 Microchip LAN9252 的移植》
- AN1916——《Microchip LAN9252 SDK 与 Beckhoff EtherCAT® SSC 集成》
- AN1995——*LAN9252 SOC Porting Guidelines*
- *EtherCAT Slave Stack Code (SSC) ET9300*——[www.ethercat.org](http://www.ethercat.org)

# AN2655

## 2.0 LAN9252 概述

图2-1: LAN9252 框图



LAN9252 是一款集成双以太网 PHY 的 2/3 端口 EtherCAT 从器件控制器，每个 PHY 包含一个全双工 100BASE-TX 收发器并支持 100 Mbps (100BASE-TX) 工作模式。

基于 LAN9252 的解决方案可采用以下模式实现：

**单片机模式：**LAN9252 通过类 SRAM 的从接口与单片机通信。凭借简单但功能强大的主机总线接口，可与大多数常见的 8 位或 16 位微处理器和单片机以及支持 8 位或 16 位外部总线的 32 位微处理器进行无胶合连接。

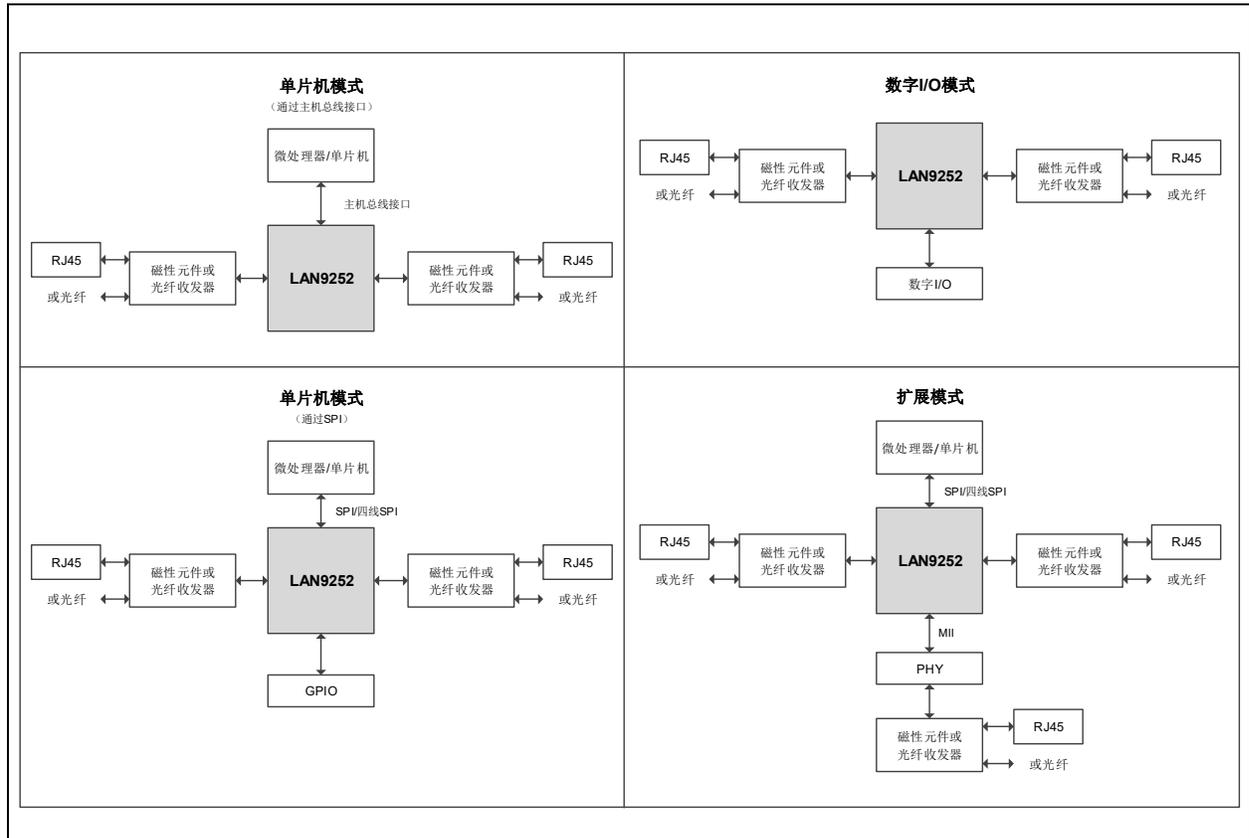
此外，该器件还可通过 SPI 或四线 SPI 进行访问，并提供最多 16 个通用输入或输出。

**扩展模式：**当该器件处于 SPI 或四线 SPI 模式时，可启用第三个网络端口来提供额外的 MII 端口。该端口既可以连接到外部 PHY 来实现星型或树型网络拓扑，也可以连接到另一个 LAN9252 来构建四端口解决方案。该端口可配置为上行或下行方向。

**数字 I/O 模式：**对于没有单片机的简单数字模块，LAN9252 可工作在数字 I/O 模式下，该模式下可通过 EtherCAT 主器件控制或监视 16 个数字信号。此外，还提供 6 个控制信号。

图2-2给出了这些工作模式的框图。

图2-2: LAN9252工作模式



从器件协议栈代码（Slave Stack Code, SSC）适用于单片机模式和扩展模式，这两种模式下可使用微处理器/单片机过程数据接口（Process Data Interface, PDI）。

SDK 1.3仅支持32位架构。而对于8位和16位架构，则需要根据编译器的要求更改源代码。

片上系统（System On Chip, SoC）MPU/MC可使用不同类型的PDI（例如HBI、SPI或SQI）访问LAN9252。LAN9252数据手册中定义了可使用PDI访问的ESC寄存器。

## 2.1 LAN9252 寄存器分类

LAN9252中有两种类型的寄存器：

- 可直接访问的寄存器（LAN9252控制和状态寄存器（Control and Status Register, CSR））

**表2-1： 系统控制和状态寄存器**

地址	寄存器名称（符号）
000h-01Ch	EtherCAT <sup>®</sup> 过程RAM读数据FIFO（ECAT_PRAM_RD_DATA）
020h-03Ch	EtherCAT过程RAM写数据FIFO（ECAT_PRAM_WR_DATA）
050h	芯片ID和版本（ID_REV）
054h	中断配置寄存器（IRQ_CFG）
058h	中断状态寄存器（INT_STS）
05Ch	中断允许寄存器（INT_EN）
064h	字节顺序测试寄存器（BYTE_TEST）
074h	硬件配置寄存器（HW_CFG）
084h	功耗管理控制寄存器（PMT_CTRL）
08Ch	通用定时器配置寄存器（GPT_CFG）
090h	通用定时器计数寄存器（GPT_CNT）
09Ch	自由运行25 MHz计数器寄存器（FREE_RUN）
<b>复位寄存器</b>	
1F8h	复位控制寄存器（RESET_CTL）
<b>EtherCAT 寄存器</b>	
300h	EtherCAT CSR接口数据寄存器（ECAT_CSR_DATA）
304h	EtherCAT CSR接口命令寄存器（ECAT_CSR_CMD）
308h	EtherCAT过程RAM读地址和长度寄存器（ECAT_PRAM_RD_ADDR_LEN）
30Ch	EtherCAT过程RAM读命令寄存器（ECAT_PRAM_RD_CMD）
310h	EtherCAT过程RAM写地址和长度寄存器（ECAT_PRAM_WR_ADDR_LEN）
314h	EtherCAT过程RAM写命令寄存器（ECAT_PRAM_WR_CMD）

- 可间接访问的寄存器——EtherCAT状态和控制寄存器（ESC）。所有EtherCAT内核寄存器均归为该组。应用程序可使用LAN9252 CSR寄存器读写EtherCAT内核寄存器。

**注：** 有关每个寄存器的更详细说明，请参见LAN9252数据手册。

## 2.2 中断

所有中断都必须在硬件初始化期间配置。由于SSC在中断现场和轮询模式下都能访问ESC寄存器，因此必须保护ECAT\_CSR\_CMD和ECAT\_CSR\_DATA寄存器，否则可能损坏SSC状态机程序。

例如，如果在轮询模式下读取0x120（ESC）寄存器期间触发中断，则一旦将0x120（地址）更新到ECAT\_CSR\_CMD寄存器后，从中断程序执行的任何ESC访问（读/写）都会立即改写ECAT\_CSR\_CMD寄存器（某个其他地址，例如0x220），进而覆盖ECAT\_CSR\_DATA寄存器。

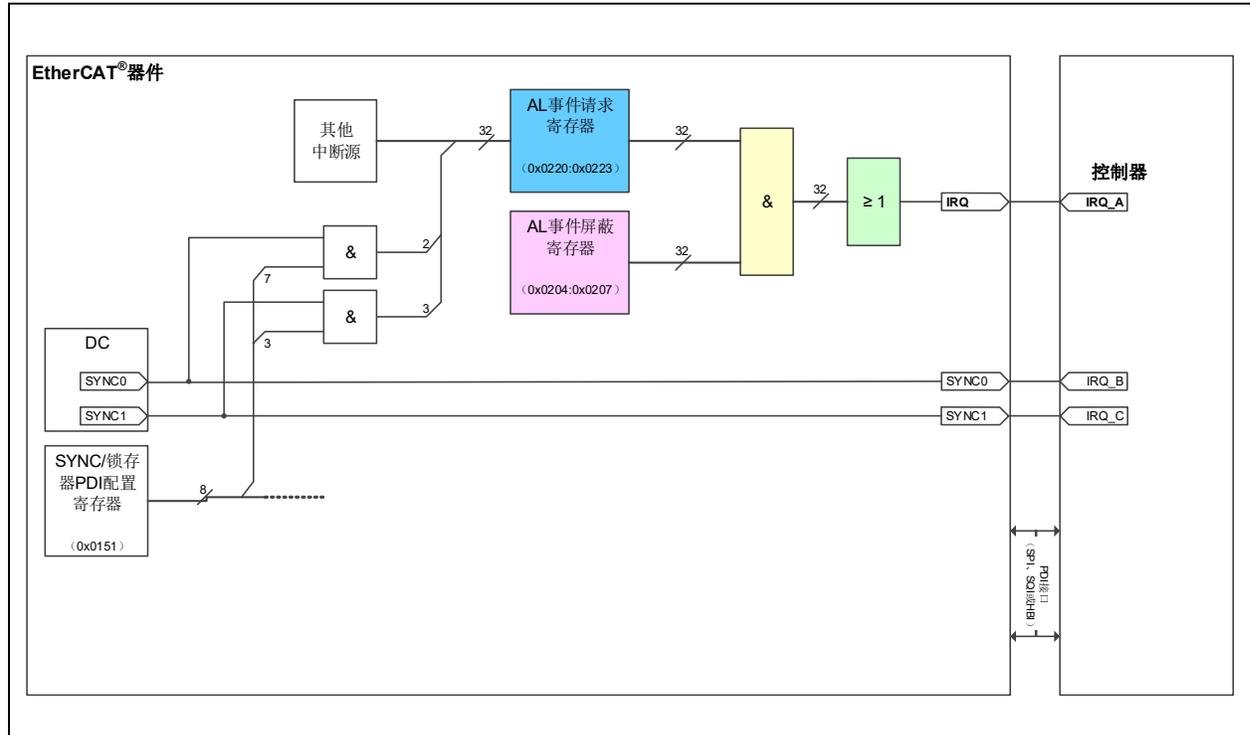
访问任何ESC寄存器时，可通过禁止中断程序来避免因触发中断而损坏数据。不过，读/写EtherCAT内核寄存器后，可通过中断线（GPIO）来监视因禁止中断而丢失的任何中断。

必须根据主机SOC/单片机修改SDK 1.x中的中断处理程序。

## 2.2.1 PDI 中断

可编程的系统中断由各种器件子模块在内部生成，并且可配置为通过 IRQ 中断输出引脚产生单个外部主机中断。凭借主机中断的可编程特性，用户可以根据应用要求优化性能。IRQ 中断的缓冲区类型、极性和置为无效间隔都可以修改。可以将 IRQ 中断配置为漏极开路输出，以便与其他器件共用该中断。所有内部中断均可屏蔽，并且能够触发 IRQ 中断。

图2-3: 功能中断机制



如果 SOC 上运行的应用程序需要 AL 事件中断，则应将 IRQ 线连接到单片机输入中断。可使用中断配置寄存器（IRQ\_CFG, 0x54）和中断允许寄存器（INT\_EN, 0x5C）来配置 IRQ。有关更多详细信息，请参见 LAN9252 数据手册。

## 2.2.2 DC —— SYNC0 和 SYNC1

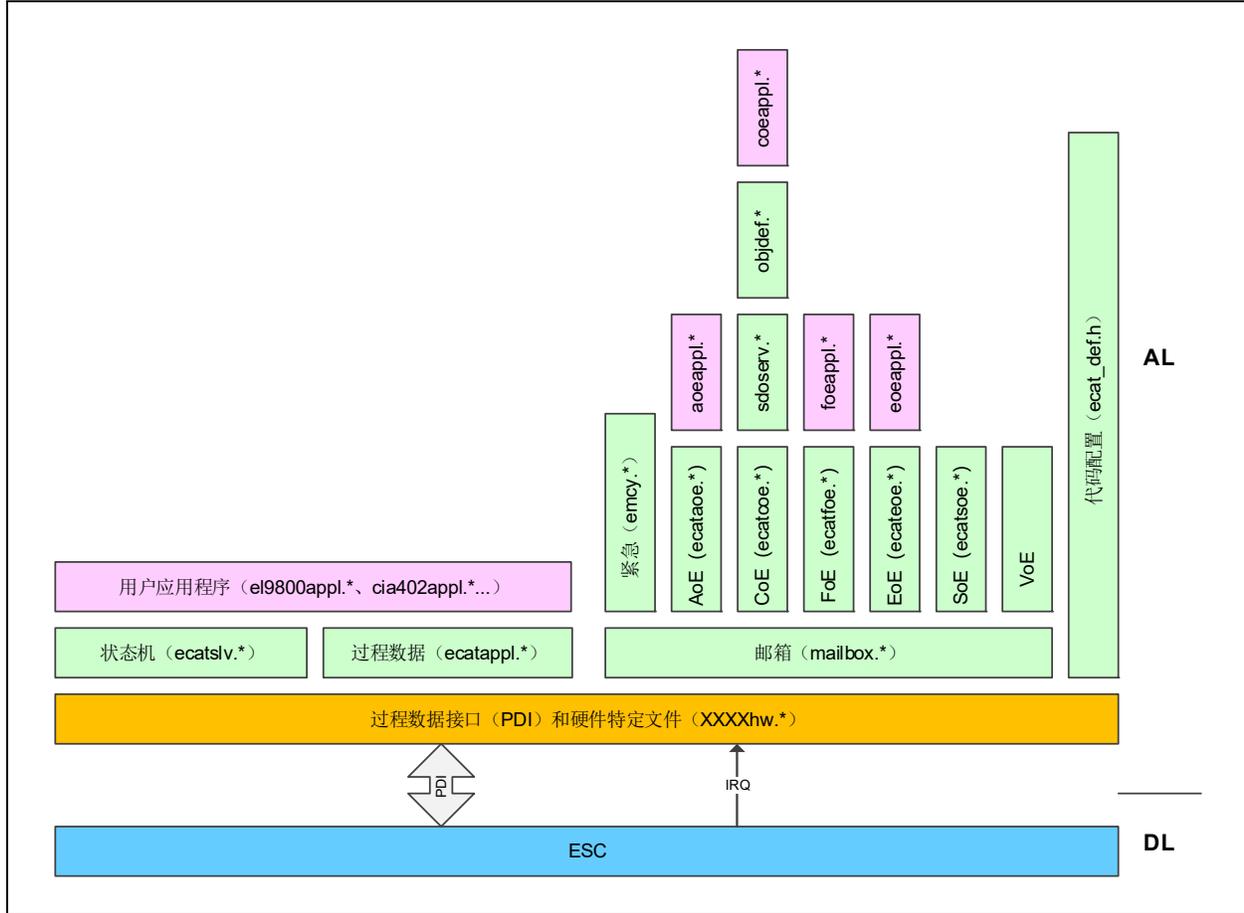
如果 SOC 上运行的应用程序需要分布式时钟（Distributed Clock, DC），则应将 SYNC0 和 SYNC1 连接到单片机的中断线。有关 SYNC0 和 SYNC1 的配置，请参见 LAN9252 数据手册。

## 2.2.3 定时器

SSC 中有一个变量每 1 ms 计数一次，具体可使用定时器中断或轮询方法来实现。生成 SSC 之前，可在 SSC 工具中选择中断或轮询模式。

## 3.0 LAN9252 SDK 1.X

图3-1: SSC概述



**注:** 有关SSC工具的更多信息, 请参见EtherCAT从器件协议栈代码(SSC)ET9300网页([www.ethercat.org](http://www.ethercat.org))。

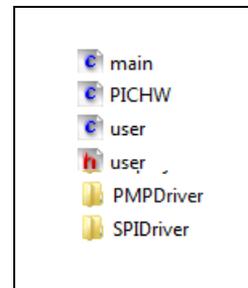
PDI和硬件特定文件是LAN9252 SDK 1.x的一部分。用户应用程序、aoeappl、coeappl、foeappl和eoeappl必须根据用户应用来定义。SDK 1.3中的PDI接口基于PIC32MX795F512L。因此, 对于任何其他SOC, 必须修改这些文件。

SDK 1.3具有以下两个根文件夹:

- **Common**——可在此处复制SSC工具生成的文件。LAN9252\_HW包含根据ET9300定义的通用PDI API, 可轻松移植到任何其他架构。
- **PIC32**——该文件夹包含与平台相关的API。对于PIC32MX795F512L之外的控制器, 必须根据使用的SOC移植所有API。

PMPDriver包含HBI驱动程序(PIC32MX PMP驱动程序)文件, 而SPIDriver包含与PIC32MX SPI相关API有关的API。

PICHW包含根据ET9300定义的平台相关API。



## 4.0 LAN9252 硬件抽象层

根据 ET9300 定义了以下函数来集成 SSC:

- UINT8 HW\_Init(void);
- void HW\_Release(void);
- UINT16 HW\_GetALEventRegister(void);
- UINT16 HW\_GetALEventRegister\_Isr(void);
- void HW\_ResetALEventMask(UINT16 intMask);
- void HW\_SetALEventMask(UINT16 intMask);
- void HW\_EscRead(MEM\_ADDR \* pData, UINT16 Address, UINT16 Len);
- void HW\_EscReadIsr(MEM\_ADDR \*pData, UINT16 Address, UINT16 Len);
- void HW\_EscWrite(MEM\_ADDR \*pData, UINT16 Address, UINT16 Len);
- void HW\_EscWriteIsr(MEM\_ADDR \*pData, UINT16 Address, UINT16 Len);
- void HW\_EscReadMbxMem(MEM\_ADDR \*pData, UINT16 Address, UINT16 Len)
- void HW\_EscWriteMbxMem(MEM\_ADDR \*pData, UINT16 Address, UINT16 Len)
- void HW\_DisableSyncManChannel(UINT8 channel);
- void HW\_EnableSyncManChannel(UINT8 channel);
- TSYNCMAN ESCMEM \*HW\_GetSyncMan(UINT8 channel);
- UINT16 MainInit(void)
- void MainLoop(void)
- IRQ、SYNC0 和 SYNC1 的中断
- 定时器中断

## 5.0 LAN9252 SDK API

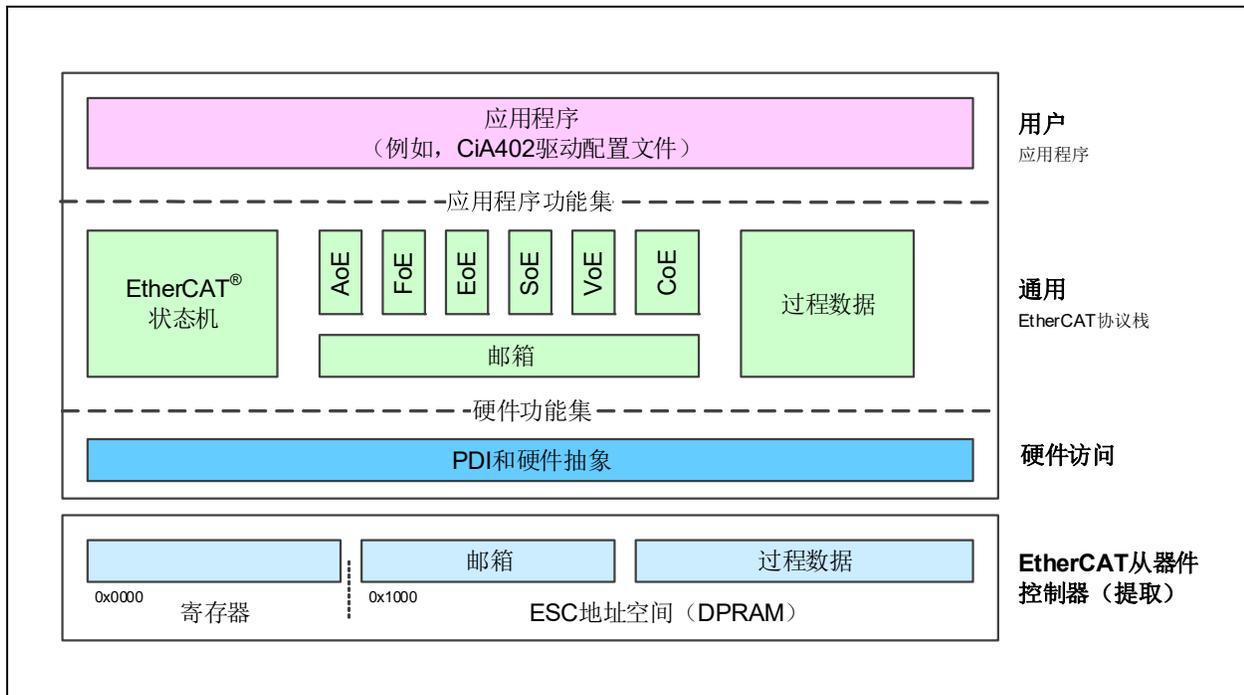
有关生成SSC代码的更多详细信息，请参见ETG提供的ET9300应用笔记和SSC工具。SSC提供了预定义宏来创建相应设置，以便为配套的ESC和SOC生成源代码。表5-1列出了一些宏示例。

表5-1: SSC预定义宏示例

宏	说明
CONTROLLER_16BIT	如果针对16位单片机编译从器件代码，则使用该设置。
CONTROLLER_32BIT	如果针对32位单片机编译从器件代码，则使用该设置。
ESC_16BIT_ACCESS	如果使用该设置，则ESC上仅执行16位对齐访问。
ESC_32BIT_ACCESS	如果使用该设置，则ESC上仅执行32位对齐访问。
MBX_16BIT_ACCESS	如果使用该设置，则从器件代码仅以16位对齐的方式访问邮箱数据。如果将邮箱数据复制到本地单片机存储器且CONTROLLER_16BIT置1，则也应设置该定义。

Microchip\_LAN9252\_SSC\_Config.xml 包含生成与LAN9252 ESC有关的SSC代码所需的预定义宏。

图5-1: SSC功能层级



SSC 由三个部分组成：

- PDI/HAL
  - 通用 LAN9252 驱动程序
  - 低层 PDI 驱动程序（特定于主机单片机）
- 通用 EtherCAT 协议栈
- 用户应用程序

通用 EtherCAT 协议栈的行为在 ETG.1000 规范 [2] 中进行了说明。通常，硬件访问实现需要支持以下功能：

- ESC 读/写访问
- 定时器源（至少 1 ms 的基本节拍）

- 每1 ms调用一次定时器处理程序（只有支持定时器中断处理时才需要；ECAT\_TIMER\_INT置1）
- 调用中断特定函数（只有支持同步时才需要）
  - PDI ISR（AL\_EVENT\_SUPPORTED置1时需要）
  - SYNC0 ISR（DC\_SUPPORTED置1时需要）

以下章节定义了HAL应提供的函数和宏（用于访问LAN9252）以及应用层应提供的函数。

## 5.1 LAN9252 驱动程序

根据ET9300应用笔记（EtherCAT从器件协议栈代码），硬件访问API需要按如下所述访问通用EtherCAT协议栈。

### 5.1.1 HW\_INIT

原型	UINT16 HW_Init(void)
参数	void
返回	初始化成功时为0 初始化期间发生错误时 > 0
说明	初始化主机控制器（PDI）并分配硬件访问所需的资源。

该函数应从从器件项目中调用，用于初始化ESC。该函数被UINT16 LAN9252\_Init(void)取代。

### 5.1.2 LAN9252\_INIT

原型	UINT16 LAN9252_Init(void)
参数	void
返回	初始化成功时为0 初始化期间发生错误时 > 0
说明	初始化主机控制器（PDI）并分配硬件访问所需的资源。

该函数应从从器件项目中调用，用于初始化ESC。调用该函数前，应初始化从器件控制器的PDI接口。

要初始化LAN9252，应读取字节顺序寄存器（0x64），直到读取到0x87654321。使能AL事件请求寄存器事件的AL事件屏蔽可映射到PDI IRQ信号（设置0x93）。

该函数初始化LAN9252 PDI接口、配置LAN9252 IRQ极性并提供函数来允许各种中断（IRQ中断、SYNC中断和定时器中断）。

- PDI\_Init\_SYNC\_Interrupts()——配置并允许SYNC中断
- PDI\_Timer\_Interrupt()——配置并允许定时器中断
- PDI\_IRQ\_Interrupt()——配置并允许IRQ中断
- PDI\_Enable\_Global\_interrupt()——全局中断

这些函数取决于从器件单片机。

请参见LAN9252数据手册的“Interrupt Registers”部分。

### 5.1.3 HW\_RELEASE

原型	void HW_Release(void)
参数	void
返回	void
说明	释放已分配的资源

如果示例应用程序停止时必须释放硬件资源，则实现该函数。

## 5.1.4 GETALEVENTREGISTER

原型	UINT16 HW_GetALEventRegister(void)
参数	void
返回	寄存器 0x220-0x221 的内容
说明	获取AL事件寄存器（0x220-0x221）的前两个字节。

**注：** 读取AL事件寄存器（0x220-0x221）时应禁止中断。

## 5.1.5 HW\_GETALEVENTREGISTER\_ISR

原型	UINT16 HW_GetALEventRegister_Isr(void)
参数	void
返回	寄存器 0x220-0x221 的内容
说明	如果需要使用特殊函数从中断服务程序中进行ESC访问，则应实现该函数，否则应将该函数定义为HW_GetALEventRegister。获取AL事件寄存器（0x220-0x221）的前两个字节。

## 5.1.6 HW\_RESETALEVENTMASK

原型	void HW_ResetALEventMask(UINT16 intMask)
参数	“intMask”中断屏蔽（禁止的中断为0）
返回	void
说明	与AL事件屏蔽寄存器（0x0204 : 0x0205）进行逻辑与运算。

## 5.1.7 HW\_SETALEVENTMASK

原型	void HW_SetALEventMask(UINT16 intMask)
参数	“intMask”中断屏蔽（允许的中断为1）
返回	void
说明	与AL事件屏蔽寄存器（0x0204 : 0x0205）进行逻辑或运算。 只有AL_EVENT_ENABLED置1时，才需要该函数。

**注：** 只有SSC版本5.10或之前的版本才需要该函数。

## 5.1.8 HW\_SETLED

原型	void HW_SetLed(UINT8 RunLed,UINT8 ErrLed)	
参数	“RunLed”	EtherCAT运行LED状态
	“ErrLed”	EtherCAT错误LED状态
返回	void	
说明	更新EtherCAT运行和错误LED（或EtherCAT状态LED）。	

**注：** LAN9252不支持错误LED；因此，需要时应通过PDI SOC使能该功能。参数RunLed将被忽略（LAN9252内置对运行LED的支持）。

## 5.1.9 HW\_RESTARTTARGET

原型	void HW_RestartTarget(void)
参数	void
返回	void
说明	复位硬件。只有BOOTSTRAPMODE_SUPPORTED置1时，才需要该函数。

## 5.1.10 HW\_DISABLESYNCMANCHANNEL

原型	void HW_DisableSyncManChannel(UINT8 channel)	
参数	“channel”	同步管理器通道
返回	void	
说明	禁止所选同步管理器通道。将相应寄存器的bit 0置1。	

找到相应“通道”的同步管理器偏移寄存器并将其禁止。

**注：** 该函数不受支持。只有SSC版本5.10或之前的版本才需要该函数。

## 5.1.11 HW\_ENABLESYNCMANCHANNEL

原型	void HW_EnableSyncManChannel(UINT8 channel)	
参数	“channel”	同步管理器通道
返回	void	
说明	使能所选同步管理器通道。复位相应0x807寄存器的bit 0。	

找到相应“通道”的同步管理器偏移寄存器并将其禁止。

**注：** 该函数不受支持。只有SSC版本5.10或之前的版本才需要该函数。

## 5.1.12 HW\_GETSYNCMAN

原型	TSYNCMAN * HW_GetSyncMan(UINT8 channel)	
参数	“channel”	同步管理器通道
返回	指向同步管理器通道说明的指针。同步管理器说明结构大小始终为8字节；TSYNCMAN的内容因支持的ESC访问而异。	
说明	从同步管理器通道获取同步管理器寄存器的内容。从(0x800 + 8*通道)地址处开始读取8字节。	

**注：** 该函数不受支持。只有SSC版本5.10或之前的版本才需要该函数。

## 5.1.13 HW\_GETTIMER

原型	UINT32 HW_GetTimer(void)	
参数	void	
返回	当前定时器值	
说明	读取硬件定时器的当前寄存器值。如果没有硬件定时器，则该函数将返回多媒体定时器的计数器值。定时器节拍值（递增次数/ms）在ECAT_TIMER_INC_P_MS中定义。	

该函数提供“PDI\_GetTimer()”函数来获取当前硬件定时器值。必须根据从器件单片机实现来定义ECAT\_TIMER\_INC\_P\_MS。SSC基于硬件时间节拍和ECAT\_TIMER\_INC\_P\_MS计算不同的超时。

**注：** 如果不支持定时器中断（ECAT\_TIMER\_INT = 0）且要计算总线周期时间，则需要该函数。

## 5.1.14 HW\_CLEARTIMER

原型	void HW_ClearTimer(void)	
参数	void	
返回	void	
说明	清零硬件定时器值。	

该函数提供“PDI\_ClearTimer”函数来清零硬件定时器值。

## 5.1.15 HW\_EEPROMRELOAD

只有支持EEPROM仿真时，才需要该函数。LAN9252不支持EEPROM仿真。

# AN2655

## 5.1.16 读/写访问

EtherCAT CSR提供对EtherCAT内核中各个参数的寄存器级访问。LAN9252相关的寄存器可按照访问方法分为两类：直接访问和间接访问。

可直接访问的寄存器（EtherCAT CSR和过程数据RAM访问寄存器）是主系统CSR（LAN9252）的一部分。这类寄存器提供数据或命令寄存器来访问间接EtherCAT内核寄存器。有关更多详细信息，请参见LAN9252数据手册的“EtherCAT CSR and Process Data RAM Access Registers (Directly Addressable)”部分。

可间接访问的EtherCAT内核寄存器位于EtherCAT内核中，必须通过EtherCAT CSR接口数据寄存器（ECAT\_CSR\_DATA）和EtherCAT CSR接口命令寄存器（ECAT\_CSR\_CMD）间接访问。可间接访问的EtherCAT内核CSR提供对EtherCAT内核中多个可配置参数的完全访问。可间接访问的EtherCAT内核CSR在地址0h至0FFFh处进行访问，相关详细信息请参见LAN9252数据手册的“EtherCAT Core CSR Registers (Indirectly Addressable)”部分。

可以通过ECAT\_CSR\_DATA和ECAT\_CSR\_CMD从1000h处开始间接访问EtherCAT内核过程数据RAM。

此外，还可以使用EtherCAT过程RAM读取数据FIFO（ECAT\_PRAM\_RD\_DATA）和EtherCAT过程RAM写入数据FIFO（ECAT\_PRAM\_WR\_DATA）更高效地访问EtherCAT内核过程数据RAM。该方法使用单个命令和少量状态读取操作就可以通过FIFO机制传输多个DWORD。

有关更多详细信息，请参见LAN9252数据手册的“ETHERCAT PROCESS RAM READS”和“ETHERCAT PROCESS RAM WRITES”部分。

### 5.1.16.1 HW\_EscRead

原型	void HW_EscRead(MEM_ADDR *pData, UINT16 Address, UINT16 Len)	
参数	“pData”	指向本地目标缓冲区的指针。指针的类型取决于主机控制器架构（在ecat_def.h或SSC工具中指定）。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅基于8位/16位或32位ESC访问使用有效的地址（在ecat_def.h或SSC工具中指定）。
	“Len”	访问大小（以字节为单位）
返回	void	
说明	读取EtherCAT从器件控制器。该函数用于访问ESC寄存器和DPRAM区域。	

该API提供PDIReadReg()函数来访问ESC内核寄存器和过程RAM。该API的实现取决于PDI接口和从器件单片机访问LAN9252寄存器的方法（例如，DMA或访问数据缓冲区）。

**注：** 要保护CSR命令和数据寄存器（从中断程序执行的读/写事件会覆盖CSR命令和数据寄存器），必须在访问任何ESC存储器前禁止中断。

## 5.1.16.2 HW\_EscReadIsr

原型	void HW_EscReadIsr(MEM_ADDR *pData, UINT16 Address, UINT16 Len)	
参数	“pData”	指向本地目标缓冲区的指针。指针的类型取决于主机控制器架构（在 <code>ecat_def.h</code> 或 <code>SSC</code> 工具中指定）。
	“Address”	EtherCAT 从器件控制器地址。指定在 ESC 存储区内的偏移量（以字节为单位）。仅基于 8 位/16 位或 32 位 ESC 访问使用有效的地址（在 <code>ecat_def.h</code> 或 <code>SSC</code> 工具中指定）。
	“Len”	访问大小（以字节为单位）
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行 ESC 访问，则应实现该函数，否则应将该函数定义为 <code>HW_EscRead</code> 。读取 EtherCAT 从器件控制器。该函数用于访问 ESC 寄存器和 DPRAM 区域。	

## 5.1.16.3 HW\_EscReadDWord

原型	void HW_EscReadDWord(UINT32 DWordValue, UINT16 Address)	
参数	“DWordValue”	将存储寄存器值的本地 32 位变量。
	“Address”	EtherCAT 从器件控制器地址。指定在 ESC 存储区内的偏移量（以字节为单位）。仅使用有效的 32 位地址。
返回	void	
说明	从 EtherCAT 从器件控制器的指定地址处读取两个字。	

## 5.1.16.4 HW\_EscReadDWordIsr

原型	void HW_EscReadDWordIsr(UINT32 DWordValue, UINT16 Address)	
参数	“DWordValue”	将存储寄存器值的本地 32 位变量。
	“Address”	EtherCAT 从器件控制器地址。指定在 ESC 存储区内的偏移量（以字节为单位）。仅使用有效的 32 位地址。
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行 ESC 访问，则应实现该函数，否则应将该函数定义为 <code>HW_EscReadWord</code> 。从 EtherCAT 从器件控制器的指定地址处读取两个字。	

## 5.1.16.5 HW\_EscReadWordIsr

原型	void HW_EscReadWordIsr(UINT16 WordValue, UINT16 Address)	
参数	“WordValue”	将存储寄存器值的本地 16 位变量。
	“Address”	EtherCAT 从器件控制器地址。指定在 ESC 存储区内的偏移量（以字节为单位）。仅使用有效的 16 位地址。
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行 ESC 访问，则应实现该函数，否则应将该函数定义为 <code>HW_EscReadWord</code> 。从 EtherCAT 从器件控制器的指定地址处读取一个字。	

**注：** 只有 `ESC_32_BIT_ACCESS` 未置 1 时才需要。

# AN2655

## 5.1.16.6 HW\_EscReadByte

原型	void HW_EscReadByte(UINT8 ByteValue, UINT16 Address)	
参数	“ByteValue”	将存储寄存器值的本地8位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。
返回	void	
说明	从EtherCAT从器件控制器读取一个字节。	

**注：** 只有ESC\_16BIT\_ACCESS和ESC\_32BIT\_ACCESS未置1时才需要。

## 5.1.16.7 HW\_EscReadByteIsr

原型	void EscReadByteIsr(UINT8 ByteValue, UINT16 Address)	
参数	“ByteValue”	将存储寄存器值的本地8位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行ESC访问，则应实现该函数，否则应将该函数定义为HW_EscReadByte。从EtherCAT从器件控制器读取一个字节。	

**注：** 只有ESC\_16BIT\_ACCESS和ESC\_32BIT\_ACCESS未置1时才需要。

## 5.1.16.8 HW\_EscReadMbxMem

原型	void HW_EscReadMbxMem(MEM_ADDR *pData, UINT16 Address, UINT16 Len)	
参数	“pData”	指向本地目标邮箱缓冲区的指针。指针的类型取决于主机控制器架构（在ecat_def.h或SSC工具中指定）。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅基于8位/16位或32位ESC访问使用有效的地址（在ecat_def.h或SSC工具中指定）。
	“Len”	访问大小（以字节为单位）
返回	void	
说明	从ESC读取数据，然后复制到从器件邮箱存储器中。如果本地邮箱存储器也位于应用程序存储器中，则该函数相当于HW_EscRead。	

## 5.1.16.9 HW\_EscWrite

原型	void HW_EscWrite(MEM_ADDR *pData, UINT16 Address, UINT16 Len)	
参数	“pData”	指向本地源缓冲区的指针。指针的类型取决于主机控制器架构（在ecat_def.h或SSC工具中指定）。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅基于8位/16位或32位ESC访问使用有效的地址（在ecat_def.h或SSC工具中指定）。
	“Len”	访问大小（以字节为单位）
返回	void	
说明	写入EtherCAT从器件控制器。该函数用于访问ESC寄存器和DPRAM区域。	

该API提供PDIWriteReg()函数来访问ESC内核寄存器和过程RAM。该API的实现取决于PDI接口和从器件单片机访问LAN9252寄存器的方法（例如，DMA或直接访问数据缓冲区）。

**注：** 要保护CSR命令和数据寄存器（从中断程序执行的读/写事件会覆盖CSR命令和数据寄存器），必须在访问任何ESC存储器前禁止中断。

## 5.1.16.10 HW\_EscWritelsr

原型	void HW_EscWritelsr(MEM_ADDR *pData, UINT16 Address, UINT16 Len)	
参数	“pData”	指向本地源缓冲区的指针。指针的类型取决于主机控制器架构（在ecat_def.h或SSC工具中指定）。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅基于8位/16位或32位ESC访问使用有效的地址（在ecat_def.h或SSC工具中指定）。
	“Len”	访问大小（以字节为单位）
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行ESC访问，则应实现该函数，否则应将该函数定义为HW_EscWrite。写入EtherCAT从器件控制器。该函数用于访问ESC寄存器和DPRAM区域。	

## 5.1.16.11 HW\_EscWriteDWord

原型	void HW_EscWriteDWord(UINT32 DWordValue, UINT16 Address)	
参数	“DWordValue”	包含要写入ESC存储区的数据的本地32位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅使用有效的32位地址。
返回	void	
说明	向EtherCAT从器件控制器写入一个字。	

# AN2655

## 5.1.16.12 HW\_EscWriteDWordIsr

原型	void HW_EscWriteDWordIsr(UINT32 DWordValue, UINT16 Address)	
参数	“DWordValue”	包含要写入ESC存储区的数据的本地32位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅使用有效的32位地址。
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行ESC访问，则应实现该函数，否则应将该函数定义为HW_EscWriteWord。向EtherCAT从器件控制器写入两个字。	

## 5.1.16.13 HW\_EscWriteWordIsr

原型	void HW_EscWriteWordIsr(UINT16 WordValue, UINT16 Address)	
参数	“WordValue”	包含要写入ESC存储区的数据的本地16位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅使用有效的16位地址。
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行ESC访问，则应实现该函数，否则应将该函数定义为HW_EscWriteWord。向EtherCAT从器件控制器写入一个字。	

**注：** 只有ESC\_32BIT\_ACCESS未置1时才需要。

## 5.1.16.14 HW\_EscWriteWord

原型	void HW_EscWriteWordIsr(UINT16 WordValue, UINT16 Address)	
参数	“WordValue”	包含要写入ESC存储区的数据的本地16位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅使用有效的16位地址。
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行ESC访问，则应实现该函数，否则应将该函数定义为HW_EscWriteWord。向EtherCAT从器件控制器写入一个字。	

**注：** 只有ESC\_32BIT\_ACCESS未置1时才需要。

## 5.1.16.15 HW\_EscWriteByte

原型	void HW_EscWriteByte(UINT8 ByteValue, UINT16 Address)	
参数	“ByteValue”	包含要写入ESC存储区的数据的本地8位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。
返回	void	
说明	向EtherCAT从器件控制器写入一个字节。	

## 5.1.16.16 HW\_EscWriteByteIsr

原型	void HW_EscWriteByteIsr(UINT8 ByteValue, UINT16 Address)	
参数	“ByteValue”	包含要写入ESC存储区的数据的本地8位变量。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。
返回	void	
说明	如果需要使用特殊函数从中断服务程序中进行ESC访问，则应实现该函数，否则应将该函数定义为HW_EscWriteByte。向EtherCAT从器件控制器写入一个字节。	

只有禁止ESC\_16BIT\_ACCESS和ESC\_32BIT\_ACCESS时才定义。

## 5.1.16.17 HW\_EscWriteMbxMem

原型	void HW_EscWriteMbxMem(MEM_ADDR *pData, UINT16 Address, UINT16 Len)	
参数	“pData”	指向本地源邮箱缓冲区的指针。指针的类型取决于主机控制器架构（在ecat_def.h或SSC工具中指定）。
	“Address”	EtherCAT从器件控制器地址。指定在ESC存储区内的偏移量（以字节为单位）。仅基于8位/16位或32位ESC访问使用有效的地址（在ecat_def.h或SSC工具中指定）。
	“Len”	访问大小（以字节为单位）
返回	Void	
说明	将从器件邮箱存储器的数据写入ESC存储器。如果本地邮箱存储器也位于应用程序存储器中，则该函数相当于HW_EscWrite。	

# AN2655

## 5.1.17 应用程序

本节中的所有API均与最终应用程序相关。

### 5.1.17.1 APPL\_Application

原型	void APPL_Application(void)
参数	Void
返回	void
说明	该函数可通过同步ISR调用，如果未激活同步，则可从主循环调用。

### 5.1.17.2 APPL\_GetDeviceID

原型	UINT16 APPL_GetDeviceID(void)
参数	Void
返回	写入AL状态代码寄存器的显式器件ID。
说明	如果主器件请求显式器件ID，则调用该函数。只有从器件支持处理显式器件ID (EXPLICIT_DEVICE_ID) 时才需要。

### 5.1.17.3 pAPPL\_EEPROM\_Read

原型: UINT16(\* pAPPL\_EEPROM\_Read)(UINT32 wordaddr)

注: LAN9252 不支持EEPROM仿真。

### 5.1.17.4 pAPPL\_EEPROM\_Write

原型: UINT16(\* pAPPL\_EEPROM\_Write)(UINT32 wordaddr)

注: LAN9252 不支持EEPROM仿真。

### 5.1.17.5 pAPPL\_EEPROM\_Reload

原型: UINT16(\* pAPPL\_EEPROM\_Reload)(void)

注: LAN9252 不支持EEPROM仿真。

### 5.1.17.6 APPL\_StartMailboxHandler

原型	UINT16 APPL_StartMailboxHandler(void)
参数	Void
返回	请参见通用ESM返回代码说明。
说明	在状态从INIT切换到PREOP或从INIT切换到BOOT期间调用该函数。

### 5.1.17.7 APPL\_StopMailboxHandler

原型	UINT16 APPL_StopMailboxHandler(void)
参数	Void
返回	请参见通用ESM返回代码说明。
说明	在状态从PREOP切换到INIT或从BOOT切换到INIT期间调用该函数。

### 5.1.17.8 APPL\_StartInputHandler

原型	UINT16 APPL_StartInputHandler(UINT16 *pIntMask)
参数	寄存器0x204的pIntMask值 (AL事件屏蔽)
返回	请参见通用ESM返回代码说明。
说明	在状态从PREOP切换到SAFEOP期间调用该函数 (即使没有输入过程数据也是如此)。

## 5.1.17.9 APPL\_StopInputHandler

原型	UINT16 APPL_StopInputHandler(void)
参数	Void
返回	请参见通用ESM返回代码说明。
说明	在状态从SAFEOP切换到PREOP期间调用该函数（即使没有输入过程数据也是如此）。

## 5.1.17.10 APPL\_StartOutputHandler

原型	UINT16 APPL_StartOutputHandler(void)
参数	Void
返回	请参见通用ESM返回代码说明。
说明	在状态从SAFEOP切换到OP期间调用该函数（即使没有输出过程数据也是如此）。

## 5.1.17.11 APPL\_StopOutputHandler

原型	UINT16 APPL_StopOutputHandler(void)
参数	Void
返回	请参见通用ESM返回代码说明。
说明	在状态从OP切换到SAFEOP期间调用该函数（即使没有输出过程数据也是如此）。

## 5.1.17.12 APPL\_GenerateMapping

原型	UINT16 APPL_GenerateMapping(UINT16 *pInputSize, UINT16 *pOutputSize)
参数	指向存储过程数据大小的两个16位变量的指针。 pInputSize: 输入过程数据（从器件 -> 主器件） pOutputSize: 输出过程数据（主器件 -> 从器件）
返回	请参见通用ESM返回代码说明。
说明	当EtherCAT主器件请求从PREOP切换到SAFEOP时，将调用该函数。该函数计算过程数据大小（以字节为单位）。检查同步管理器设置和处理通用过程数据时需要使用这些计算值。

## 5.1.17.13 APPL\_AckErrorInd

原型	Void APPL_AckErrorInd(UINT16 stateTrans)
参数	stateTrans: 指示当前状态切换
返回	Void
说明	当主器件应答错误时，将调用该函数。

## 5.1.17.14 APPL\_InputMapping

原型	void APPL_InputMapping(UINT16 *pData)
参数	指向输入过程数据的pData指针
返回	Void
说明	该函数在应用程序调用后调用，以将输入过程数据映射到通用协议栈。通用协议栈将数据复制到SM缓冲区中。

**注：** pData 包含主器件的输入缓冲区内容。

# AN2655

## 5.1.17.15 APPL\_OutputMapping

原型	void APPL_OutputMapping(UINT16 *pData)
参数	指向输出过程数据的pData指针
返回	Void
说明	该函数在应用程序调用前调用，以获取输出过程数据。

**注：** pData包含主器件的输出缓冲区内容。

## 5.1.18 变量

### 5.1.18.1 ApplicationObjDic

名称	ApplicationObjDic
类型	TOBJECT结构的数组
说明	只有从器件支持CAN over EtherCAT (CoE)时才需要。变量在应用程序头文件中定义。该数组包含应用程序特定的对象，其中最后一个元素的索引为0xFFFF。

### 5.1.18.2 pEEPROM

名称	pEEPROM
类型	UINT8 *
说明	指向EEPROM缓冲区的指针。只有使能EEPROM仿真(ESC_EEPROM_EMULATION = 1)时才需要。该指针在ecatappl.h中定义，并由应用程序在上电期间(调用MainInit()前)设置。EEPROM缓冲区的大小由ESC_EEPROM_SIZE(默认为2048)设置定义。

**注：** LAN9252不支持EEPROM仿真。

## 5.2 PDI驱动程序

LAN9252提供了不同类型的过程数据接口(PDI)。请参见图2-2。

主机总线接口提供以下两个选项供用户选择：

- 索引寄存器访问
- 复用地址/数据总线

HBI支持大尾数法、小尾数法和混合字节顺序的8位/16位操作。两个过程数据RAM FIFO将HBI与EtherCAT从器件控制器相连，以便在主机CPU和EtherCAT从器件之间传输过程数据信息。

SPI/四线SPI从器件控制器提供低引脚数同步从接口，方便器件与主机系统之间的通信。凭借SPI/四线SPI从器件，可访问系统CSR、内部FIFO和存储器。它支持一条和多条采用递增、递减和静态寻址的寄存器读写命令。支持时钟速率最高为80 MHz的单个位、双位和四位通道。

### 5.2.1 通用PDI API

本节中的所有API均设计为与32位处理器或控制器配合使用。

#### 5.2.1.1 PDI\_Init

原型	void PDI_Init();
参数	void
返回	void
说明	初始化PDI接口

**注：** 该API应在访问PDI接口前调用，其实现取决于从器件单片机。

## 5.2.2 读/写 API

EtherCAT CSR 提供对 EtherCAT 内核中各个参数的寄存器级访问。EtherCAT 相关的寄存器可按照访问方法分为两大类：直接访问和间接访问。

LAN9252SDK 1.x 与 PIC32MX795F512L 兼容。

### 5.2.2.1 读/写可直接寻址寄存器

#### 5.2.2.1.1 *PDIWriteLAN9252DirectReg*

原型	void PDIWriteLAN9252DirectReg(UINT32 Val, UINT16 Address)	
参数	Val	DWORD 值
	Address	可直接寻址寄存器的地址
返回	void	
说明	向可直接寻址寄存器写入“Val”。	

**注：** 可使用 HBI 或 SPI/SQI 接口访问 LAN9252 寄存器。

#### 5.2.2.1.2 *PDIReadLAN9252DirectReg*

原型	UINT32 PDIReadLAN9252DirectReg(UINT16 Address)	
参数	Address	可直接寻址寄存器的地址
返回	UINT32	读取的 DWORD 值
说明	读取可直接寻址寄存器。	

**注：** 可使用 HBI 或 SPI/SQI 接口访问 LAN9252 寄存器。

### 5.2.2.2 读/写可间接寻址寄存器（EtherCAT 内核寄存器）

#### 5.2.2.2.1 *PDIReadReg*

原型	void PDIReadReg(UINT8 *ReadBuffer, UINT16 Address, UINT16 Count)	
参数	ReadBuffer	指向读缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	该函数使用 LAN9252 CSR 或 FIFO 读取 ESC 寄存器。	

**注：** 可使用 HBI 或 SPI/SQI 接口访问 LAN9252 寄存器。

#### 5.2.2.2.2 *PDIWriteReg*

原型	void PDIWriteReg(UINT8 *WriteBuffer, UINT16 Address, UINT16 Count)	
参数	WriteBuffer	指向要写入的缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用 LAN9252 CSR 或 FIFO 写入 ESC 寄存器。	

**注：** 可使用 HBI 或 SPI/SQI 接口访问 LAN9252 寄存器。

# AN2655

## 5.2.2.2.3 主机总线接口

以下 API 分别使用 PMPReadDWord 和 PMPWriteDWord 来读取和写入 LAN9252 寄存器。此类 API 必须根据 SSC 控制器中使用的主机总线接口来定义。

请参见 LAN9252 数据手册的“HBI Bus Interface”部分。

### • PMPReadRegUsingCSR

原型	void PMPReadRegUsingCSR(UINT8 *ReadBuffer, UINT16 Address, UINT8 Count)	
参数	ReadBuffer	指向读缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用 LAN9252 CSR 寄存器读取 EtherCAT 内核寄存器。	

**注：** 有关更多详细信息，请参见 LAN9252 数据手册的“EtherCAT Process RAM Reads”部分。

### • PMPReadPDRamRegister

原型	void PMPReadPDRamRegister(UINT8 *ReadBuffer, UINT16 Address, UINT16 Count)	
参数	ReadBuffer	指向读缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用 LAN9252 FIFO 读取 PDRAM。	

**注：** 请参见 LAN9252 数据手册的“EtherCAT CSR and Process Data RAM Access Registers (Directly Addressable)”部分。

### • PMPWriteRegUsingCSR

原型	void PMPWriteRegUsingCSR(UINT8 *WriteBuffer, UINT16 Address, UINT8 Count)	
参数	WriteBuffer	指向要写入的缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用 LAN9252 CSR 写入 ESC 寄存器。	

**注：** 有关更多详细信息，请参见 LAN9252 数据手册的“EtherCAT Process RAM Writes”部分。

### • PMPWritePDRamRegister

原型	void PMPWritePDRamRegister(UINT8 *WriteBuffer, UINT16 Address, UINT16 Count)	
参数	WriteBuffer	指向要写入的缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用 LAN9252 FIFO 写入 ESC 寄存器。	

**注：** 请参见 LAN9252 数据手册的“EtherCAT CSR and Process Data RAM Access Registers (Directly Addressable)”部分。

## 5.2.2.2.4 SPI

• *SPIReadRegUsingCSR*

原型	void SPIReadRegUsingCSR(UINT8 *ReadBuffer, UINT16 Address, UINT8 Count)	
参数	ReadBuffer	指向读缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用LAN9252 CSR寄存器读取EtherCAT内核寄存器。	

**注：** 有关更多详细信息，请参见LAN9252数据手册的“EtherCAT Process RAM Reads”部分。

• *SPIReadPDRamRegister*

原型	void SPIReadPDRamRegister(UINT8 *ReadBuffer, UINT16 Address, UINT16 Count)	
参数	ReadBuffer	指向读缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用LAN9252 FIFO读取PDRAM。	

**注：** 请参见LAN9252数据手册的“EtherCAT CSR and Process Data RAM Access Registers (Directly Addressable)”部分。

• *SPIWriteRegUsingCSR*

原型	void SPIWriteRegUsingCSR(UINT8 *WriteBuffer, UINT16 Address, UINT8 Count)	
参数	WriteBuffer	指向要写入的缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用LAN9252 CSR写入ESC寄存器。	

**注：** 有关更多详细信息，请参见LAN9252数据手册的“EtherCAT Process RAM Writes”部分。

• *SPIWritePDRamRegister*

原型	void PMPWritePDRamRegister(UINT8 *WriteBuffer, UINT16 Address, UINT16 Count)	
参数	WriteBuffer	指向要写入的缓冲区的指针
	Address	可直接寻址寄存器的地址
	Count	要读取的字节数
返回	void	
说明	使用LAN9252 FIFO写入ESC寄存器。	

**注：** 请参见LAN9252数据手册的“EtherCAT CSR and Process Data RAM Access Registers (Directly Addressable)”部分。

• *SPIReadDWord*

原型	UINT32 SPIReadDWord(UINT16 Address)	
参数	Address	LAN9252 CSR寄存器的地址
返回	读取的值 (DWORD)	
说明	该函数从LAN9252 CSR寄存器读取4字节数据。	

**注：** 有关更多详细信息，请参见LAN9252数据手册的“SPI/SQI Slave”部分。

# AN2655

## • SPIWriteDWord

原型	void SPIWriteDWord(UINT16 Address, UINT32 Val)	
参数	Address	LAN9252 CSR寄存器（而非ESC寄存器）的地址
	Val	4字节值
返回	void	
说明	该函数向LAN9252 CSR寄存器（而非ESC寄存器）的相应地址写入4字节数据。	

**注：** 有关更多详细信息，请参见LAN9252数据手册的“SPI/SQI Slave”部分。

## • SPIReadBurstMode

原型	UINT32 SPIReadBurstMode(UINT16 Address)	
参数	void	
返回	读取的值（DWORD）	
说明	该函数从LAN9252 CSR寄存器读取4字节数据。	

**注：** 该函数不会将CS置为有效。有关更多详细信息，请参见LAN9252数据手册的“SPI/SQI Slave”部分。

## • SPIWriteBurstMode

原型	UINT32 SPIWriteBurstMode(UINT16 Address)	
参数	Address	LAN9252 CSR寄存器的地址
返回	读取的值（DWORD）	
说明	该函数向LAN9252 CSR寄存器的相应地址写入4字节数据。	

**注：** 该函数不会将CS置为有效。有关更多详细信息，请参见LAN9252数据手册的“SPI/SQI Slave”部分。

## • SPIWriteBytes

原型	void SPIWriteBytes(UINT16 Address, UINT8 *Val, UINT8 nLength)	
参数	Address	CSR寄存器的地址
	Val	指向写缓冲区的指针
	nLength	缓冲区的长度
返回	void	
说明	该函数用于写入LAN9252 CSR寄存器。	

**注：** 有关更多详细信息，请参见LAN9252数据手册的“SPI/SQI Slave”部分。

### 5.3 从器件控制器相关的API

以下API的定义取决于使用的SSC控制器。

#### 5.3.1 中断API

##### 5.3.1.1 PDI\_Timer\_Interrupt

原型	void PDI_Timer_Interrupt()
参数	void
返回	void
说明	该函数配置并允许TIMER中断（持续1 ms）。

##### 5.3.1.2 PDI\_IRQ\_Interrupt

原型	void PDI_IRQ_Interrupt()
参数	void
返回	void
说明	该函数配置并允许IRQ的中断。

##### 5.3.1.3 PDI\_Enable\_Global\_interrupt

原型	void PDI_Enable_Global_interrupt()
参数	void
返回	void
说明	允许从器件控制器的中断。

##### 5.3.1.4 PDI\_Disable\_Global\_Interrupt

原型	UINT32 PDI_Disable_Global_Interrupt()
参数	void
返回	中断状态的前一个状态
说明	该函数读取当前中断状态并禁止所有中断请求。

**注：** PDI\_Restore\_Global\_Interrupt使用中断状态的前一个状态来恢复中断状态。

##### 5.3.1.5 PDI\_Init\_SYNC\_Interrupts

原型	void PDI_Init_SYNC_Interrupts()
参数	void
返回	void
说明	该函数配置并允许SYNC0和SYNC1中断。

#### 5.3.2 中断处理程序

以下函数由通用SSC（在ecatappl.h中定义）提供，必须从硬件访问层调用。

##### 5.3.2.1 ECAT\_CheckTimer

原型	void ECAT_CheckTimer(void)
参数	void
返回	void
说明	该函数必须每1 ms从定时器ISR调用一次（ECAT_TIMER_INT = 1）。如果不支持定时器中断，则该函数在1 ms后自动调用（基于提供的定时器）。

**注：** 必须根据从器件控制器来实现定时器ISR。

# AN2655

## 5.3.2.2 PDI\_Isr

原型	void PDI_Isr(void)
参数	void
返回	void
说明	该函数必须从PDI ISR调用。有关PDI特定引脚的命名和中断产生逻辑，请参见LAN9252数据手册。要支持PDI中断处理，还需将AL_EVENT_ENABLED置1。

**注：** 必须根据从器件控制器来实现IRQ ISR。

## 5.3.2.3 Sync0\_Isr

原型	void Sync0_Isr(void)
参数	void
返回	void
说明	该函数必须从Sync0 ISR调用。Sync0中断由ESC的DC单元产生。默认当前不支持将Sync0信号映射到PDI中断。要支持DC同步，必须将DC_SUPPORTED置1。

**注：** 必须根据从器件控制器来实现SYNC0 ISR。

## 5.3.2.4 Sync1\_Isr

原型	void Sync1_Isr(void)
参数	void
返回	void
说明	该函数必须从Sync1 ISR调用。Sync1中断由ESC的DC单元产生。默认当前不支持将Sync1信号映射到PDI中断。要支持DC同步，必须将DC_SUPPORTED置1。

**注：** 必须根据从器件控制器来实现SYNC1 ISR。

## 5.3.3 主机总线接口

主机总线接口（Host Bus Interface, HBI）可采用索引模式或复用模式。

在复用地址/数据模式下，地址和字节顺序选择输入与数据总线共用，具体支持两种方法：最多使用16个地址或数据引脚的单阶段地址，以及仅使用低8个数据位的双阶段地址。

在索引地址模式下，使用索引和数据寄存器间接映射对器件内部寄存器和存储器的访问。所需的内部地址按特定偏移量写入器件。写入的值随后在访问相关数据寄存器地址时用作内部地址。索引或数据寄存器集共有三个，支持多线程操作，无需担心其中一个线程会损坏另一个线程设置的索引。

有关更多详细信息，请参见LAN9252数据手册的“HBI Bus Interface”部分。

### 5.3.3.1 PMPReadDWord

原型	UINT32 PMPReadDWord(UINT16 Address)
参数	Address   LAN9252 CSR寄存器的地址
返回	读取的值（DWORD）
说明	该函数从LAN9252 CSR寄存器读取4字节数据。

**注：** 如果PDI接口为主机总线接口，则必须根据从器件控制器来实现该API。该函数将在初始化从器件控制器的HBI后调用。

## 5.3.3.2 PMPWriteDWord

原型	void PMPWriteDWord(UINT16 Address, UINT32 Val)	
参数	Address	LAN9252 CSR 寄存器（而非 ESC 寄存器）的地址
	Val	4 字节值
返回	void	
说明	该函数向 LAN9252 CSR 寄存器（而非 ESC 寄存器）的相应地址写入 4 字节数据。	

**注：** 如果 PDI 接口为主机总线接口，则必须根据从器件控制器来实现该 API。该函数将在初始化从器件控制器的 HBI 后调用。

## 5.3.4 SPI

SPI/SQI 从模块提供低引脚数同步从接口，方便器件与主机系统之间进行通信。凭借 SPI/SQI 从器件，可访问系统 CSR、内部 FIFO 和存储器。它支持一条和多条采用递增、递减和静态寻址的寄存器读写命令。

有关更多详细信息，请参见 LAN9252 数据手册的“SPI/SQI SLAVE”部分。

## 5.3.4.1 SPIReadByte

原型	void SPIReadByte()	
参数	void	
返回	void	
说明	该函数从 LAN9252 CSR 寄存器读取 1 字节数据。有关如何使用 SPI 进行读取的信息，请参见 LAN9252 数据手册。	

## 5.3.4.2 SPIWriteByte

原型	void SPIWriteByte(UINT8 data)	
参数	data	要写入的数据
返回	void	
说明	该函数向 LAN9252 CSR 寄存器的相应地址写入 1 字节数据。	

## 5.3.4.3 宏

以下宏与运行 SSC 的 SOC 相关。

- CSLOW ()——该宏将 CS 线驱动为低电平。
- CSHIGH ()——该宏将 CS 线驱动为高电平。
- DISABLE\_ESC\_INT——禁止 IRQ 中断
- ENABLE\_ESC\_INT——允许 IRQ 中断
- ECAT\_TIMER\_INC\_P\_MS——定时器节拍值（递增次数/ms）

# AN2655

---

## 6.0 SSC工具配置文件

Microchip\_LAN9252\_SSC\_Config.xml 包含生成与 LAN9252 ESC 有关的 SSC 代码所需的预定义宏。该配置文件中的默认值可通过 SSC 工具或直接修改文件来更新。

**注：** 有关 SSC 工具的更多信息，请参见 EtherCAT 从器件协议栈代码（SSC）ET9300 网页（[www.ethercat.org](http://www.ethercat.org)）。

## 附录A： 版本历史

表A-1： 版本历史

版本	节/图/条目	更正
DS00002655A (2018年2月23日)	初始版本	

## MICROCHIP 网站

Microchip 网站 ([www.microchip.com](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。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

## 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，请访问 [www.microchip.com/pcn](http://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 <http://microchip.com/support> 获得网上技术支持。

---

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之中。Microchip 承诺将不断改进产品的代码保护功能。

---

提供本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。如需额外的支持, 请联系当地的 Microchip 销售办事处, 或访问 <https://www.microchip.com/en-us/support/design-help/client-supportservices>。

**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, 不得暗中或以其他方式转让任何许可证。

有关 Microchip 质量管理体系的更多信息, 请访问 [www.microchip.com/quality](http://www.microchip.com/quality)。

## 商标

Microchip 的名称和徽标组合、Microchip 徽标、Adaptec、AVR、AVR 徽标、AVR Freaks、BesTime、BitCloud、CryptoMemory、CryptoRF、dsPIC、flexPWR、HELDO、IGLOO、JukeBlox、KeeLoq、Kleer、LANCheck、LinkMD、maXStylus、maXTouch、MediaLB、megaAVR、Microsemi、Microsemi 徽标、MOST、MOST 徽标、MPLAB、OptoLyzer、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、Liberio、motorBench、mTouch、Powermite 3、Precision Edge、ProASIC、ProASIC Plus、ProASIC Plus 徽标、Quiet-Wire、SmartFusion、SyncWorld、Temux、TimeCesium、TimeHub、TimePictra、TimeProvider、TrueTime 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、Clockstudio、CodeGuard、CryptoAuthentication、CryptoAutomotive、CryptoCompanion、CryptoController、dsPICDEM、dsPICDEM.net、Dynamic Average Matching、DAM、ECAN、Espresso T1S、EtherGREEN、GridTime、IdealBridge、In-Circuit Serial Programming、ICSP、INICnet、Intelligent Paralleling、IntelliMOS、Inter-Chip Connectivity、JitterBlocker、Knob-on-Display、KoD、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、SmartHLS、SMART-I.S.、storClad、SQL、SuperSwitcher、SuperSwitcher II、Switchtec、SynchroPHY、Total Endurance、Trusted Time、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。

© 2022, Microchip Technology Incorporated 及其子公司版权所有。

ISBN: 978-1-6683-0885-1

## 全球销售及及服务网点

### 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](http://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: 45-4485-5910  
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: 49-7131-72400

**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: 47-7288-4388

**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