
8位单片机上的CRC和存储器扫描

作者: *Vinaya Skanda*
Microchip Technology Inc.

简介

在当今的大多数应用中，数据完整性对于数据的存储和传输都是必需的。循环冗余校验（Cyclic Redundancy Check, CRC）是一种常用的错误检测技术，可用于保持存储器和通信中的数据完整性。

本技术摘要介绍了某些PIC®单片机上集成的循环冗余校验（CRC）外设和存储器扫描功能。还阐释了CRC计算方法以及不同标准的存储器扫描方法和常用的CRC多项式。在执行IEC 60730标准所需的存储器测试以支持B类认证的过程中，CRC和存储器扫描外设非常有用。

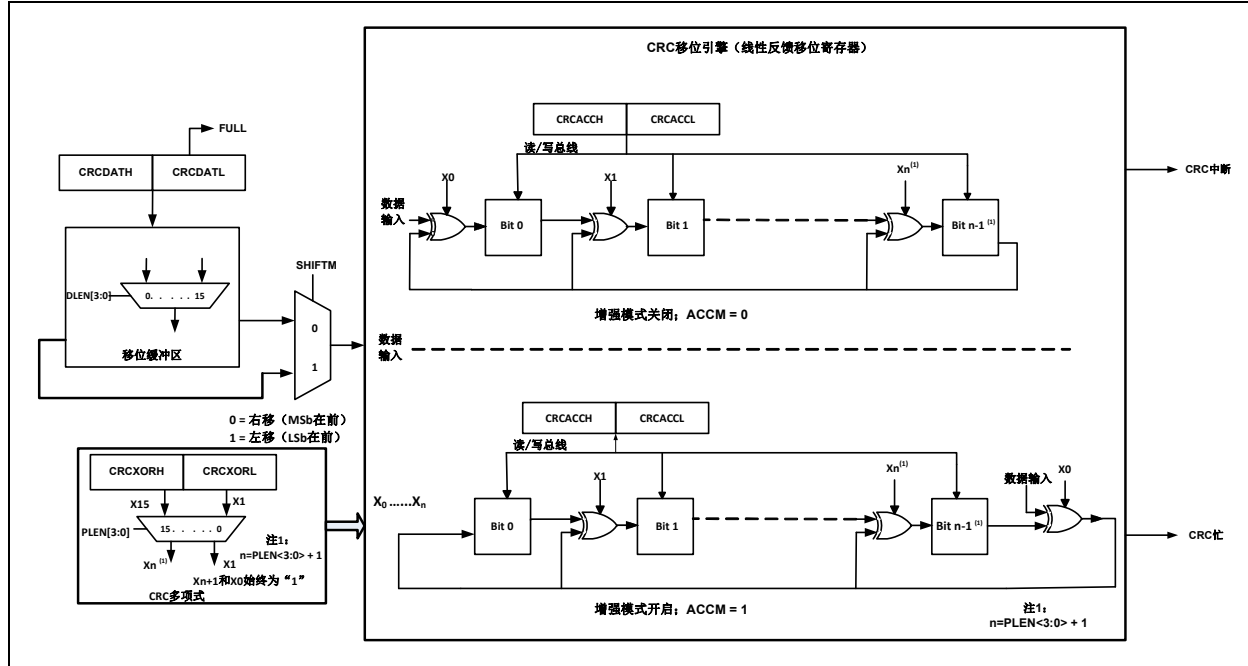
PIC单片机上的CRC外设

循环冗余校验（CRC）模块提供了可通过软件配置并由硬件实现的CRC校验和生成器。CRC算法将数据视为一个二进制数。将该数字除以另一个称为多项式的二进制数。除法的余数为CRC校验和，它会附加到该数据的最后。每个整数都可以唯一表示为一个以2为基数、系数非0即1的多项式。用虚拟变量X来解释多项式非常方便。

例如，报文110101由下面的多项式表示： $X^5 + X^4 + X^2 + 1$ ($1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$)

在通信系统中发送报文时，CRC校验和会附加到报文最后。接收器随后将报文（包括计算出的CRC）除以发送器使用的同一多项式（要实现基于多项式的CRC，发送器和接收器必须使用相同的多项式）。如果除法得出的结果为零，则视为传输成功。但是，如果除法的结果不等于零，则视为传输期间出现错误，需要针对推定的数据损坏采取纠正措施。当CRC的校验和值为n位长时，会将其称为n位CRC。

图1: CRC框图



CRC模块通过实现线性反馈移位寄存器（Linear Feedback Shift Register, LFSR）来执行CRC计算所需的多项式除法，如图1所示。CRC模块最多可接受17位（n+1）生成器多项式。在任一（n+1）位多项式中， X^n 和 X^0 都始终为1。将（n+1）位多项式装入CRCXOR<15:1>寄存器时，应仅输入 X^{n-1} 至 X^0 。PLEN<3:0>位用于指定多项式的长度。装入该长度时，应将（n+1）位多项式的MSB和LSB排除。因此应装入n-1。

例如，采用CRC-16-ANSI时，为17位多项式 $X^{16} + X^{15} + X^2 + 1$ ， $n = 16$ 。

因此，PLEN<3:0> = $n - 1 = 15$

CRCXOR<15:1> = 0b100000000000010

（CRCXOR<0>未实现，因为CRC硬件将 X^{16} 和 X^0 视为1）

因此，CRCXOR可以是0x8005或0x8004。CRC引擎将其读为0x8005，但当CPU使用软件读取CRCXOR时，会将其读为0x8004。

数据需要按照相同的顺序装入CRCDATH和CRCDATL寄存器。数据装入CRCDATL寄存器后，整个16位数据会移入移位缓冲区，这是一个可读寄存器。DLEN位确定数据宽度（CRCDATH和CRCDATL寄存器中的有效位）。只将由DLEN位确定的数据移至移位缓冲区。假如SHIFTM位置1，数据会从LSb开始移位，反之数据会从MSb开始移位。

零扩充

对于要完成的CRC计算，需要通过XOR门对全部数据位进行移位，这需要在其末尾添加零，零的个数等于多项式的长度。这称为零扩充。之前，此过程必须使用软件完成。现在，可利用扩充模式在硬件中完成零扩充，从而降低软件开销。通过将ACCM位置1，可以使能扩充模式。

初始CRC值（种子值）

这是提供固定起点的初始值，可以从该起点开始处理数据位。许多CRC算法将其种子值初始化为非零值。在实际应用中，许多数据字符串可能都以一长串的0开头，而CRC计算会忽略第一个值为1的位前的任意个0。

要避免此问题，在计算n位CRC之前，始终从前n位与非零值的n位字符串作逻辑异或运算开始。

种子值可在两种CRC计算方法中使用，例如：

直接方法

将种子值直接装入CRCACC<15:0>寄存器。

非直接方法

将种子值视为初始数据值并将实际数据流作为前缀，CRCACC<15:0>寄存器装入0x0000。

BUSY位指示CRC计算是否完成。BUSY位为0后，可以从CRCACC<15:0>寄存器读取校验值。CRCIF标志也将置1。

PIC 单片机上的存储器扫描模块

程序存储器扫描模块可与CRC模块结合使用，在程序存储器地址范围内执行CRC计算。CRC模块与存储器扫描模块配合使用可以实现更快速的CRC计算。存储器扫描模块可以自动为CRC模块提供数据。这种存储器扫描测试有助于为通过IEC 60730标准的B类认证提供支持。

有四种可能的存储器扫描操作模式：

突发：CPU操作停止，直到整个存储器扫描完成。该模式可提供最高的存储器扫描吞吐量。

并发：SCANGO位置1后立即开始扫描。存储器访问期间CPU操作停止。

CPU在每次访问后恢复执行。

触发：等待至独立触发时钟的上升沿开始扫描。存储器访问期间CPU操作停止。

CPU在每次访问后恢复执行。

窥视：等待CPU无需访问NVM的指令周期（例如跳转指令），并将该周期用于进行自己的NVM访问。该模式可提供最低吞吐量。CPU操作不受影响。

将数据置于程序存储器中时，存储器扫描器会自动将数据提供给CRC模块。在窥视模式下使用存储器扫描的优势在于，可在使用CRC对数据进行错误检查时释放CPU资源以执行其他任务。

存储器扫描期间的中断交互：如果INTM位置1，则中断的优先级更高，从而延迟存储器扫描。如果INTM位未置1，则扫描器的优先级将高于中断，从而增加中断响应延时。

CRC和存储器扫描外设的配置

一些常用的CRC生成器多项式如下：

表1：常用的CRC生成器多项式

名称	多项式	多项式表示	种子值
SDLC (CRC-16- CCITT)	$X^{16} + X^{12} + X^5 + 1$	0x1021	0xFFFF
CRC-16- ANSI	$X^{16} + X^{15} + X^2 + 1$	0x8005	0
CRC12	$X^{12} + X^{11} + X^3 + X^2 + X + 1$	0x80F	0

配置CRC

本节以代码段的形式介绍了CRC的配置顺序。

例1: 直接模式CRC的CRC-CCITT标准代码段

```
unsigned char data_buffer[] = {0x31,0x32,0x33,0x34,0x35,0x36,0x37,0x38,0x39};
CRCCON0bits.EN = 1; // enable the CRC module
CRCACC = 0xFFFF; //seed a starting CRC value
CRCXOR = 0x1021; //CRC generator polynomial for CRC-CCITT
CRCCON1bits.DLEN = 0b0111; //length of the data word = 8 bits
CRCCON1bits.PLEN = 0b1111; //order of the polynomial = 16
CRCCON0bits.ACCM = 1; //Direct CRC algorithm, zero augmentation not required
CRCCON0bits.SHIFTM = 0; // MSb of the data will be shifted first
CRCCON0bits.CRCGO = 1; //start the CRC module
data_size = sizeof(data_buffer); // data_buffer is the array containing data
for (data_index=0; data_index < data_size; data_index++ )
{
    while (CRCCON0bits.CRCFULL); // wait until the data registers are full
    if (CRCCON1bits.DLEN <8){ // data has < 8 bits; load data in CRCDATL
        CRCDATL = data_buffer[data_index];}
    else { // data has > 8 bits; load CRCDATAH register first and then CRCDATL
        CRCDATH = (data_buffer[data_index] & 0xFF00)>>8;
        CRCDATL = (data_buffer[data_index] & 0xFF);    }
}
while(CRCCON0bits.BUSY); // check if CRC calculation is completed
crc = CRCACC; // read the CRC check value ( = 0x29B1 for given data sequence)
```

配置存储器扫描

根据 IEC 60730 标准, 要支持 B 类认证过程, 需要在系统运行期间定期执行存储器测试。系统启动时, 会对存储在程序存储器 (闪存) 中数据的 CRC 校验和进行计算, 并且该校验和可用作参考校验和。可将其存储到闪存中。运行期间, 可以定期调用 CRC 和存储器扫描功能。可将运行期间计算的校验和与参考校验和进行比较。如果两个校验和相等, 则视为已通过存储器测试。

例2: 非直接模式CRC采用CRC-CCITT标准的存储器扫描代码段

```

SCANCON0bits.EN = 1; // enable the Memory Scan module
SCANCON0bits.MODE = 0; // Concurrent mode
SCANCON0bits.INTM = 0; //Interrupts do not prevent NVM access
SCANLADR = STARTADR; // load beginning location of the memory to be scanned
SCANHADR = STOPADR; //load the end location of the memory to be scanned

CRCCON0bits.EN = 1; // enable the CRC module
CRCACC = 0xFFFF; //seed a starting CRC value
CRCXOR = 0x1021; //CRC generator polynomial for CRC-CCITT
CRCCON1bits.DLEN = 0b0111; //length of the data word = 8 bits
CRCCON1bits.PLEN = 0b1111; //order of the polynomial = 16
CRCCON0bits.ACCM = 1; //zeroes equal to order of polynomial will be augmented
CRCCON0bits.SHIFTM = 0; // MSb of the data will be shifted first
CRCCON0bits.CRCGO = 1; //start the CRC module
SCANCON0bits.SCANGO = 1; //Begin the scan
while(SCANCON0bits.SCANGO); // wait for scanner to finish pushing data in to
//the CRC data registers
while(CRCCON0bits.BUSY); //check if CRC calculation is completed
crc = CRCACC; // read the CRC check value

```

将MPLAB®代码配置器（MCC）用于CRC和存储器扫描模块

MPLAB®代码配置器是一款用于MPLAB X IDE的插件工具，它根据在GUI中做出的设置和选择来生成用于控制和驱动CRC和存储器扫描模块的驱动程序，如图2所示。

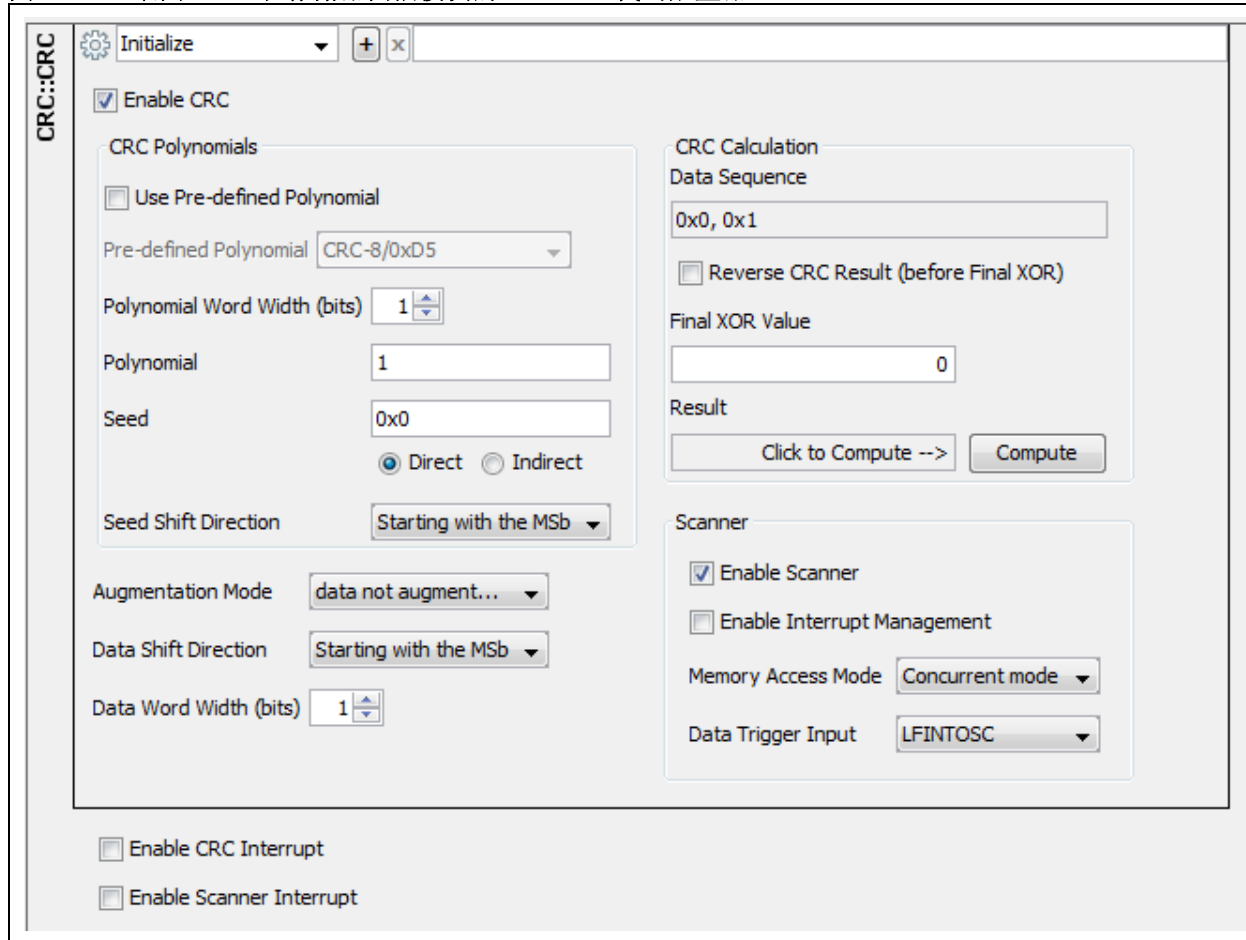
单击Generate Code（生成代码）按钮后，MPLAB X中的项目会生成下表2列出的有关CRC和存储器扫描模块的所有API。

表2: CRC和存储器扫描的MCC API列表

函数	说明
void CRC_Initialize(void)	根据MCC中的设置初始化CRC。
void CRC_Start(void)	将CRCCON0寄存器的CRCGO位置1，以开始移位过程。
bool CRC_8BitDataWrite(uint8_t data)	将数据写入CRCDATHL寄存器对。
uint16_t CRC_CalculatedResultGet(bool reverse, uint16_t xorValue)	读取并返回正常值或反向值。
bool CRC_IsBusy(void)	返回CRCCON0寄存器BUSY位的状态，以检查CRC计算是否完成。
void CRC_SCAN_StartScanner(void)	启动扫描过程。
void CRC_SCAN_StopScanner(void)	停止扫描过程。
void CRC_SCAN_SetAddressLimit(uint16_t startAddr, uint16_t endAddr)	将地址限制装入SCANLADRH/L和SCANHADRH/L寄存器对。
bool CRC_SCAN_HasInvalidAddressOccured(void)	检查扫描过程中是否出现无效地址。
bool CRC_SCAN_IsScannerBusy(void)	返回SCANCON0寄存器BUSY位的状态。
bool CRC_SCAN_HasScanCompleted(void)	返回SCANIF中断标志的状态。

注 1: 关于适用于CRC的最新版MCC GUI，请访问Microchip网站。

图2: 用于CRC和存储器扫描模块的MPLAB®代码配置器(MCC)



结论

在各种数字通信系统中，CRC均可用于检测数据传输过程中的错误。大多数常用的通信协议（例如USB、Modbus、CAN和以太网）都使用CRC进行错误检测。在执行IEC 60730标准所需的存储器测试以支持B类认证过程时，CRC与存储器扫描外设非常有用。本技术摘要介绍了8位PIC单片机上的CRC外设。此外，还提供了与CRC配合使用的存储器扫描模块的用法及其配置。

附录 A: 用于 CRC 和存储器扫描的示例寄存器汇总

表 A-1: 与 CRC 和存储器扫描模块相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CRCACCH	ACC<15:8>							
CRCACCL	ACC<7:0>							
CRCCON0	EN	CARGO	BUSY	ACCM	—	—	SHIFTM	FULL
CRCCON1	DLEN<3:0>				PLEN<3:0>			
CRCDATH	DATA<15:8>							
CRCDATL	DATA<7:0>							
CRCSHIFTH	SHIFT<15:8>							
CRCSHIFTL	SHIFT<7:0>							
CRCXORH	X<15:8>							
CRCXORL	X<7:1>							—
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF
PIR4	SCANIF	CRCIF	SMT2PWAIF	SMT2PRAIF	SMT2IF	SMT1PWAIF	SMT1PRAIF	SMT1IF
PIE4	SCANIE	CRCIE	SMT2PWAIE	SMT2PRAIE	SMT2IE	SMT1PWAIE	SMT1PRAIE	SMT1IE
SCANCON0	EN	SCANGO	BUSY	INVALID	INTM	—	MODE<1:0>	
SCANHADRH	HADR<15:8>							
SCANHADRL	HADR<7:0>							
SCANLADRH	LADR<15:8>							
SCANLADRL	LADR<7:0>							
SCANTRIG	—	—	—	—	—	—	TSEL<1:0>	

图注: — = 未实现单元, 读为 0。CRC 和存储器扫描模块不使用阴影单元。

* 该页提供寄存器信息。

附录 B: CRC 非直接模式下的种子转换

软件许可协议

Microchip Technology Incorporated (“公司”) 随附提供的软件旨在提供给您 (该公司的客户) 使用, 仅限于且只能在该公司制造的产品上使用。

该软件为公司和/或其供应商所有, 并受适用的版权法保护。版权所有。任何违反前述限制的使用将使其用户遭受适用法律的刑事制裁, 并承担违背此许可的条款和条件的民事责任。

该软件“按现状”提供。不提供保证, 无论是明示的、暗示的还是法定的保证。这些保证包括 (但不限于) 对出于某一特定目的应用此软件的适销性和适用性默示的保证。在任何情况下, 公司都将不会对任何原因造成的特别的、偶然的或间接的损害负责。

如果将非直接模式用于 CRC 种子, 则用户可以预先转换种子值并将其直接装入 CRCACC<15:0> 寄存器, 从而减少计算花费的时间。

例如, 采用 SDLC (CCITT) 标准时, 直接算法中使用的初始种子值为 0xFFFF。该初始值可通过代码段预先转换为 0x84CF:

用于将直接算法种子值转换为非直接算法种子值的代码段如例 B-1 所示。

例 B-1: 将直接算法种子值转换为非直接算法种子值

```
crcinit_direct = crcinit;
crc = crcinit;
for (i=0; i<order; i++) //order of CRC polynomial e.g. for CCITT order = 16
{
    bit = crc & 1;
    if (bit) crc ^= polynom;
    crc >>= 1;
    if (bit) crc |= crhighbit; //crhighbit for 16 bit CRC polynomial= 0x8000
}
crcinit_nondirect = crc;
```

请注意以下有关 Microchip 器件代码保护功能的要点：

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

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。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。除非另外声明，在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、Microchip 徽标、dsPIC、FlashFlex、flexPWR、JukeBlox、KEELOQ、KEELOQ 徽标、Kleer、LANCheck、MediaLB、MOST、MOST 徽标、MPLAB、OptoLyzer、PIC、PICSTART、PIC³² 徽标、RightTouch、SpyNIC、SST、SST 徽标、SuperFlash 及 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

The Embedded Control Solutions Company 和 mTouch 为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、BodyCom、chipKIT、chipKIT 徽标、CodeGuard、dsPICDEM、dsPICDEM.net、ECAN、In-Circuit Serial Programming、ICSP、Inter-Chip Connectivity、KleerNet、KleerNet 徽标、MiWi、motorBench、MPASM、MPF、MPLAB Certified 徽标、MPLIB、MPLINK、MultiTRAK、NetDetach、Omniscient Code Generation、PICDEM、PICDEM.net、PICkit、PICtail、RightTouch 徽标、REAL ICE、SQI、Serial Quad I/O、Total Endurance、TSHARC、USBCheck、VariSense、ViewSpan、WiperLock、Wireless DNA 和 ZENA 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。SQTP 为 Microchip Technology Inc. 在美国的服务标记。

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

GestIC 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。

© 2016, Microchip Technology Inc. 版权所有。

ISBN: 978-1-5224-0178-0

**QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
= ISO/TS 16949 =**

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

全球销售及及服务网点

美洲

公司总部 **Corporate Office**
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277

技术支持:

<http://www.microchip.com/support>

网址: www.microchip.com

亚特兰大 Atlanta

Duluth, GA
Tel: 1-678-957-9614
Fax: 1-678-957-1455

奥斯汀 Austin, TX

Tel: 1-512-257-3370

波士顿 Boston

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

芝加哥 Chicago

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

克里夫兰 Cleveland

Independence, OH
Tel: 1-216-447-0464
Fax: 1-216-447-0643

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

洛杉矶 Los Angeles

Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

纽约 New York, NY

Tel: 1-631-435-6000

圣何塞 San Jose, CA

Tel: 1-408-735-9110

加拿大多伦多 Toronto

Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 Asia Pacific Office

Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2943-5100
Fax: 852-2401-3431

中国 - 北京

Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

中国 - 成都

Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 重庆

Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

中国 - 东莞

Tel: 86-769-8702-9880

中国 - 杭州

Tel: 86-571-8792-8115
Fax: 86-571-8792-8116

中国 - 香港特别行政区

Tel: 852-2943-5100
Fax: 852-2401-3431

中国 - 南京

Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛

Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海

Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳

Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳

Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

中国 - 武汉

Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 西安

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 厦门

Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 - 珠海

Tel: 86-756-321-0040
Fax: 86-756-321-0049

亚太地区

台湾地区 - 高雄

Tel: 886-7-213-7828

台湾地区 - 台北

Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

台湾地区 - 新竹

Tel: 886-3-5778-3666
Fax: 886-3-5770-955

澳大利亚 Australia - Sydney

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore

Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

印度 India - New Delhi

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune

Tel: 91-20-3019-1500

日本 Japan - Osaka

Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

日本 Japan - Tokyo

Tel: 81-3-6880-3770
Fax: 81-3-6880-3771

韩国 Korea - Daegu

Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul

Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang

Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 Philippines - Manila

Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore

Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok

Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels

Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark - Copenhagen

Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris

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

德国 Germany - Dusseldorf

Tel: 49-2129-3766400

德国 Germany - Karlsruhe

Tel: 49-721-625370

德国 Germany - Munich

Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan

Tel: 39-0331-742611
Fax: 39-0331-466781

意大利 Italy - Venice

Tel: 39-049-7625286

荷兰 Netherlands - Drunen

Tel: 31-416-690399
Fax: 31-416-690340

波兰 Poland - Warsaw

Tel: 48-22-3325737

西班牙 Spain - Madrid

Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

瑞典 Sweden - Stockholm

Tel: 46-8-5090-4654

英国 UK - Wokingham

Tel: 44-118-921-5800
Fax: 44-118-921-5820