



Microchip 2018财年 第一季度财报

2018 财年第一季度 (2017年4月1日至2017年6月30日) 财报

公司动态

MICROCHIP发布2018财年第一季度财报，宣布净销售额和收入创纪录

2018 财年第一季度 (2017年4月1日至2017年6月30日) 财报：

按照通用会计准则 (GAAP)：

净销售为9.721亿美元，环比上涨7.7%，同比上涨21.6%。

按照非通用会计准则 (Non-GAAP)：

净销售为9.721亿美元，环比上涨7.7%，同比上涨15.2%。我司于2017年6月5日更新的净销售额预期值为9.433亿美元至9.568亿美元。

按照GAAP准则：

毛利率为60.1%；营业利润创纪录，为2.216亿美元；净利润创纪录，为1.706亿美元；摊薄后每股收益创纪录，为70美分。我司于2017年6月5日更新的EPS 预期值为62美分至64美分。

按照Non-GAAP准则：

毛利率为60.4%；营业利润创纪录，为3.643亿美元；净利润创纪录，为3.191亿美元；摊薄后每股收益创纪录，为1.31美元，环比上涨12.9%，同比上涨56%。我司于2017年6月5日更新的EPS预期值为1.22美元至1.26美元。

经营活动的现金流创纪录，为3.45亿美元。

季度股息创纪录，为每股36.2美分。

Microchip首席执行官Steve Sanghi先生说：“我们第一季度的财报非常强劲，代表着Microchip 2.0倡议带动起来的有机增长在加速。Microchip 2.0将Microchip与前期收购公司的产品、技术、系统和员工的力量结合起来，使得我们通过将多款产品应用在一块电路板中，向客户提供完整的系统解决方案，从而推动最终应用的开发。”

Microchip总裁兼首席运营官Ganesh Moorthy先生说：

“我们的单片机业务在第一季度的业绩非常迅猛，收益环比增长了9.5%，创下了新纪录。我们看到有客户仍在继续使用Atmel的传统器件，充分体现了他们对Microchip经营这些产品系列的信任。正因如此，我们看到更多的设计即将投产和增产。我们也看到了我们的“设计”漏斗的持续增长。预计随着时间的推移，这些设计步入生产阶段，将推动未来增长。”

Moorthy先生补充道：“我们模拟产品的收益在这一季度环比增长3.7%，同样也创了纪录。我们凭借强大的完整系统解决方案，向Microchip 2.0转型，帮助我们成功地发掘更多商机，在各种客户的设计和应用中配合使用Microchip丰富的模拟产品和Atmel的单片机和微处理器。加以时日，随着这些成功的设计投产，这些努力终将获得回报。”

详细数据请查看
Microchip官网发布
的财报。





活动花絮

电源技术研讨会2017

电源技术专题研讨会于9月7日在上海成功举行。始于2002年，211C举办的电源专题研讨会是伴随着国内电源行业的成长而发展起来的；十六载中，它见证了电源技术的发展，成为了一个电源技术展示和创意交汇的平台。今年Microchip在研讨会中紧扣电源技术发展趋势的主旨，探讨工程师在研究开发过程中所面临的问题。Microchip资深应用工程师徐乃洲及郜俊以《数字电源新趋势与设计》为题演讲、展示最新电源技术、与参会者进行更深入的技术交流互动。现场观众反应踊跃，气氛热烈。



现场照片：

<https://goo.gl/RyYeQB>



Microchip 2017嵌入式解决方案研讨会（中国）中国最后四场研讨会正式落下了帷幕，来年再会！

Microchip本年度的嵌入式解决方案研讨会（ESS）已经落下了帷幕。我司的工程师们和与会者们在北中国最后四场（北京站、西安站、天津站和沈阳站）的研讨会互动频繁。现场设有Microchip的产品演示及开发工具特价发售，反响热烈！



接下来还有更多不同的技术培训，请密切留意Microchip的网站！

www.microchip.com.cn/newcommunity/index.php?m=Training&a=index&id=82





最新活动

Microchip中文社区 全新改版正式上线

炎热的夏季已渐渐远去，秋，便是收获的季节。在金秋十月的10日，我们迎来了焕然一新的“Microchip中文社区”网站（www.microchip.com.cn）。改版后的网站栏目更加清晰直观，您可以更方便地查阅到所需讯息；界面支持移动设备，您可以随时随地进行访问；增设了许多新的栏目，您能更便捷地了解Microchip在中国区的各类资讯及活动。欢迎大家奔走相告，踊跃访问！

访问全新的“Microchip中文社区”网站
www.microchip.com.cn



Microchip 2017 技术精英年会（中国/台湾）

11月于成都、杭州、台北、深圳和台中举行
2017年技术精英年会（Microchip Annual Strategic Technical Engineering Review – MASTERS Conference）— 这一专为全国嵌入式控制工程师提供的技术培训盛宴，将于11月在大中华区5个城市举行为期2至3日的培训。根据不同地区的需求，今年我们开设26至30个课程不等（包括动手实验课程），主题丰富，并由Microchip应用与设计工程师亲自授课。无论是技术新手，还是资深工程师，各个技术水平的与会者都可从我们提供的课程中找到自己感兴趣并适合自己的课题。课程完成后您将获得开发新设计所需的全部信息，以助您的产品更快上市。



参加培训的费用包括：MASTERS课程的听课费（可自选课程）、一套完整的课程资料、microchipDIRECT开发工具优惠券、时尚电脑背包等等。把握最后机会报名，我们期待在2017年中国和台湾技术精英年会与您相遇！

*凡参加台湾技术精英年会者，均可免费出席其中一场Microchip嵌入式解决方案研讨会2017(秋)!

在线报名：

中国技术精英年会：
www.microchip.com/cm



台湾技术精英年会：
www.microchip.com/tm



最新活动

工业自动化展 Industrial Automation Show (IAS)

工业自动化展将于11月7至11日在上海国家会展中心举行。工业自动化展是覆盖了工业自动化，电气系统，机器人技术与IT信息解决方案的国际性展会Microchip是首次参加此展览，届时将展出最新的以太网控制自动化技术和电机控制等解决方案。诚邀您的客户莅临Microchip位于A022号的展位参观！

观众预登记：

www.industrial-automation-show.com



大学计划

第七届“微芯杯”电子设计大赛颁奖典礼暨校园专题技术讲座圆满结束

第七届“微芯杯”电子设计大赛颁奖典礼暨校园专题技术讲座在电子科技大学隆重举行。本届大赛报名参赛队伍为162支，参赛队员达400余人，得到全校同学的广泛响，最终提交作品30件，其中获奖作品18件。

近年来，在Microchip和电子科技大学航空航天学院的大力支持下，“微芯杯”电子设计大赛已经在电子科技大学成功举办了7次，累计超过3000名在校本科生参与了大赛。比赛的影响力、知名度、参与度、赛题特色、奖励金额均是校内顶尖水平，积累了足够的经验也培养了一批热衷于电子设计并具有创新能力及动手能力的高素质大学生人才，为发掘中国未来IT人才提供强有力的平台。颁奖仪式由航空航天学院首席教授徐利梅教授致开幕辞，她代表电子科

技大学衷心感谢Microchip一直以来对学院的倾力支持和帮助。

其间，Microchip应用工程师经理袁野先生为大家带来题为“8、16、32位单片机技术现状及展望”的技术讲座，介绍了单片机的发展历程和实际应用中的相关问题。

其他大学计划



1. 宣传资料

- UPD360宣传页
- SCH322X宣传页
- BM62/64宣传页
- 32位单片机系列宣传册
- PIC16F19197系列单片机宣传页

2. 数据手册

- KSZ8081MNX/RNB数据手册
- USB5807数据手册
- MIC45404数据手册
- PIC18(L)F67K40 28/44引脚XLP系列数据手册
- LAN7850数据手册
- PIC32MX1XX/2XX系列数据手册
- BM70/71数据手册

3. 参考手册

- dsPIC33/PIC24 FRM – 12位高速多SAR A/D转换器 (ADC)
- dsPIC33/PIC24 FRM – 可配置逻辑单元 (CLC)

4. 用户指南

- Microchip调试器 (MDB) 用户指南
- 适用于MPLAB® X IDE的MPLAB REAL ICE™在线仿真器用户指南
- HV98101 230VAC离线式LED驱动器评估板用户指南

5. 应用笔记

- AN2059 – LIN基础知识和8位PIC®单片机上实现的MCC LIN协议栈库
- AN2290 – 8位PIC®单片机的位分裂增强型UART
- AN2375 – 使用CIP实现斜率补偿的优势
- TB3156 – 8位PIC®单片机上的通用异步收发器 (UART)
- TB3157 – 8位PIC®单片机上的串行外设接口 (SPI) 通信
- TB3158 – 8位PIC®单片机上具有时钟切换功能的数控振荡器
- TB3163 – 8位PIC®单片机上的存储器访问分区

6. 其他文档

- dsPIC33EPXXXGS70X/80X系列闪存编程规范
- PIC24FJ256GA/GB110到PIC24FJ1024GA/GB610的移植和性能增强指南

阅读更多最新文档:



最新视频

- 1) MIC4605 BLDC吊扇演示
(主讲人: Miguel Mendoza (中文字幕))
- 2) 反盗版和配件安全加密解决方案
(主讲人: 沈宁 Echo Shen)
- 3) AVR® Insights专辑
 - 第1集 — AVR® 存储器
 - 第2集 — 边写边读存储器
 - 第3集 — 端口
 - 第4集 — 休眠模式
 - 第5集 — 熔丝
 - 第6集 — USART
 - 第7集 — 看门狗定时器
 - 第8集 — 省电寄存器
 - 第9集 — 双缓冲寄存器
 - 第10集 — 优化C代码
 - 第11集 — ADC工作模式
 - 第12集 — SPI
 - 第14集 — QTouch®防水演示

观看更多最新视频:





产品综述

Microchip推出两款具有多种连接接口的SAM单片机系列新品 SAM D5x 和 E5x MCU不但性能强大还增强了安全功能

Microchip宣布推出SAM D5x和SAM E5x单片机 (MCU) 系列产品。这些32位MCU系列新品提供各种连接接口，不但性能强大，还具有基于硬件的可靠的安全特性，适合于多种应用。

SAM D5/E5 单片机将ARM® Cortex® - M4处理器的性能与浮点单元 (FPU) 集于一身。这避免了使用中央处理单元 (CPU)，提高了系统能效，支持低功耗平台上的进程密集型应用。运行速度高达120 MHz的D5x和E5xMCU具有支持纠错码 (ECC) 的1MB双区闪存，不用中断系统运行就很容易进行现场更新。此外，这些系列的SRAM高达256 KB，也支持ECC，为医疗设备和服务器系统等任务关键应用提供了必要支持。

这些新MCU拥有多个接口，提高了设计的灵活性，即使是最苛刻的连接也能满足其需求。这些系列包括一个具有就地执行 (XIP) 特性的四路串行外设接口 (QSPI)。这样，系统能够使用高性能串行闪存，与传统的引脚并行闪存相比，不但体积小而且价格低，满足了外部存储器需求。SAM D5/E5器件还具有用于数据记录的安全数字主控制器 (SDHC)、实现电容触摸功能的外设触摸控制器 (PTC)，一流的运行功耗性能 (65 uA / MHz) 适合关注高能效的应用。此外，SAM E5系列包括两个CAN-FD端口和一个支持IEEE 1588的10 / 100 Mbps以太网媒体访问控制器 (MAC)，因此非常适合工业自动化、家庭联网和其他物联网 (IoT) 应用。



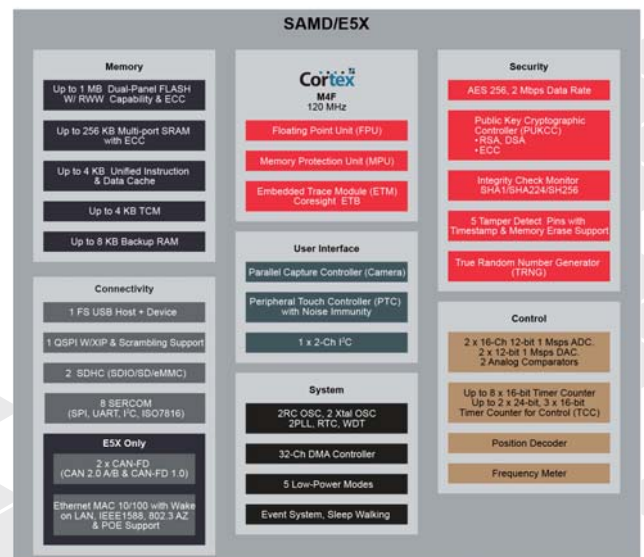
SAM D5x

SAM E5x

SAM D5x和E5x系列均提供全面的加密硬件和软件支持，使开发人员能够在开始设计时便融入安全措施。基于硬件的安全特性包括支持椭圆曲线加密 (ECC) 和RSA方案的公钥密码控制器 (PUKCC)，以及高级加密标准 (AES) 密码和安全哈希算法 (SHA)。

Microchip的MCU32业务部副总裁Rod Drake表示：应用变得越来越复杂，迫切需要有较多连接选择和灵活外设支持的高速MCU。对于寻求经济高效解决方案的开发人员而言，SAM D5/E5单片机凭借其强大的性能、丰富的接口以及内置的安全特性，能很好地满足需求。”

SAMD/E5X Block Diagram



更详细的信息，请访问：
www.microchip.com/design-centers/32-bit/sam-32-bit-mcus/sam-d-mcus



Microchip新一代在线调试器问世 拥有无与伦比的速度和灵活性 MPLAB® ICD 4的处理器速度更快 RAM容量更大



专家意见

MPLAB® Harmony 之学习篇（七） 如何使用调试信息系统服务

Microchip Technology Inc.
MCU32产品部资深应用工程师
戴仕勇

Microchip发布MPLAB® ICD 4—Microchip的PIC®单片机和dsPIC®数字信号控制器系列产品的在线编程和调试开发工具。MPLAB ICD 4 囊括了MPLAB ICD 3 调试器的所有功能，在此基础上采用更快的处理器提高了速度并增大了RAM容量。

MPLAB ICD 4速度之所以能够显著提高是因为采用了一片运行在300MHz的32位MCU。处理速度更快，还增加了2MB的高速缓存，使得该产品速度是前一代产品的两倍。



圆形MPLAB ICD 4封装在一个耐用的黑色外壳中，顶部采用拉丝铝，醒目的LED灯带用于指示调试状态。该工具具有以下特性：

- 1.2V至5.5 V更宽的目标电压范围
- 可选1 A电源（使用外部电源）
- 目标接口可选上拉/下拉选项
- 可配置接口速度，优化了编程和调试
- 具有故障检测和抗干扰能力的智能而稳健的接口
- JTAG调试功能

Microchip的MPLAB ICD 4使用方便，并通过MPLAB X集成开发环境（IDE）支持Microchip的所有PIC单片机和dsPIC数字信号控制器。当客户选择从一个PIC MCU迁移到另一个，以满足其应用需求时，这简化了客户的设计过程。

Microchip开发工具总监Rodger Richey说：“在选择调试工具时，速度和灵活性是最重要的因素。MPLAB ICD 4减少了等待时间，相应的提高了调试效率。凭借其速度、兼容性、耐用性、完整的器件支持以及屡获殊荣的MPLAB X IDE，MPLAB ICD 4再次巩固了Microchip作为嵌入式调试工具领先供应商的声誉。”

调试信息（Debug）系统服务：

嵌入式工程师经常会通过UART等打印一些调试信息，Harmony提供了标准的系统调试信息服务功能，支持调试信息的打印。这样，用户不需要再创建自己的打印接口等，节省开发时间。

Debug系统服务的消息是发给控制台（Console）系统服务的，Console系统服务可以中转给UART，USB CDC，APPIO。

该应用笔记采用大家最熟悉的UART作为调试口进行调试信息打印。

MHC配置使能调试信息（Debug）系统服务：

在利用MHC配置之前，当然我们得先了解板子使用的哪个UART，UART的TX/RX分别映射到的哪个IO口，这里以PIC32MZ EF Starter Kit开发板为例：

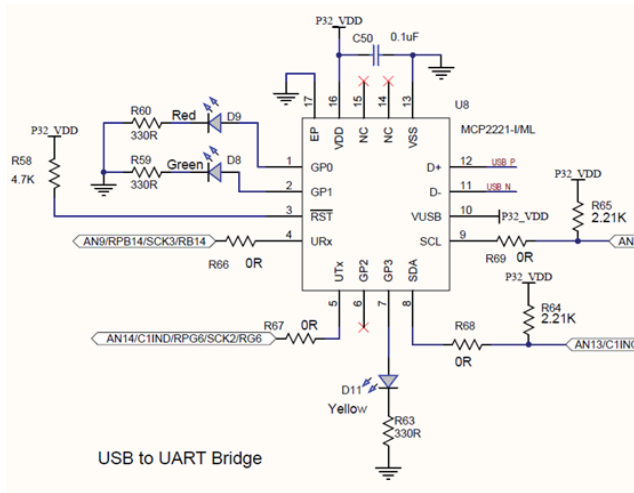


PIC32MZ with FPU Embedded Connectivity Starter Kit
(Part # DM320007)

关于MPLAB ICD 4
更详细的信息，请访问：
www.microchip.com/ICD4



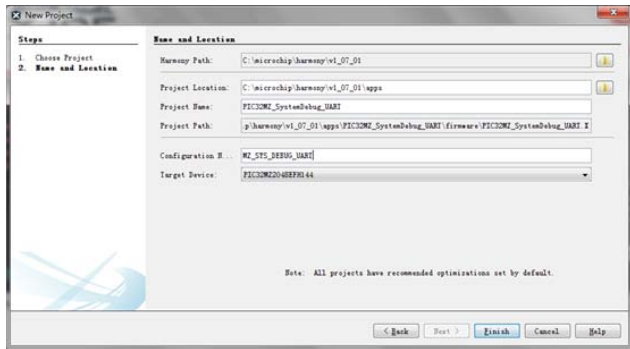
该板上有一个USB转UART芯片通过J11 (USB Mini-B) 接口连接，其UART连接如下图：



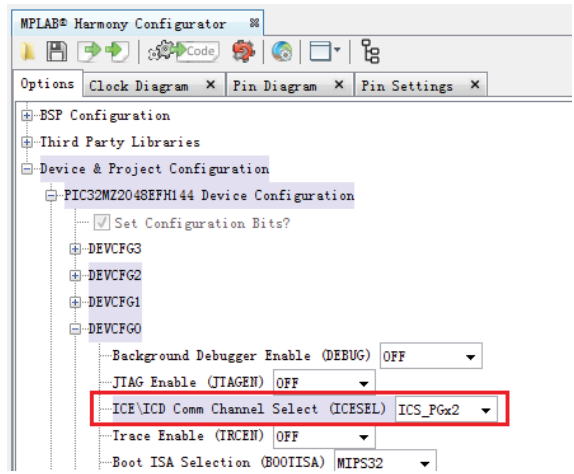
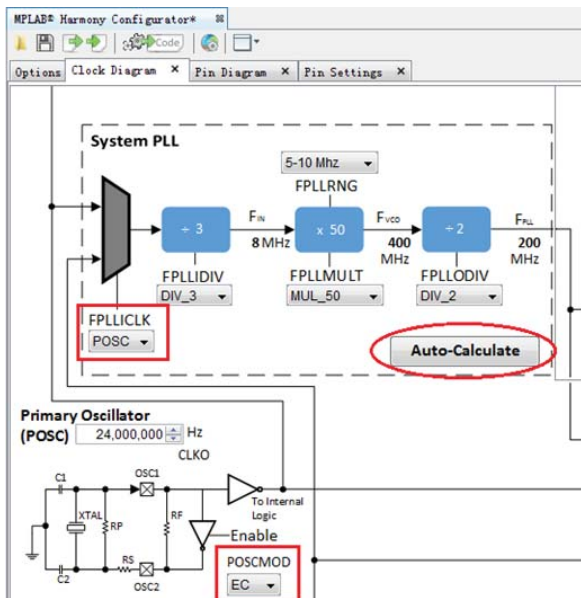
从上图可见，RPB14和RPG6分别安排给了UART的TX和RX，可以通过MZ数据手册了解到UART2可以映射到这两个IO口。

接下来我们按照《MPLAB® Harmony之学习篇（三）创建一个Harmony项目》的步骤一步步的配置系统和使能调试系统服务。

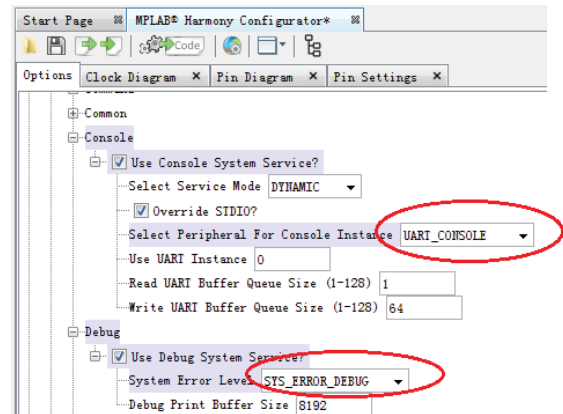
1) 新建Harmony项目：



2) 进入MHC后配置时钟、正确的ICSP调试口等：

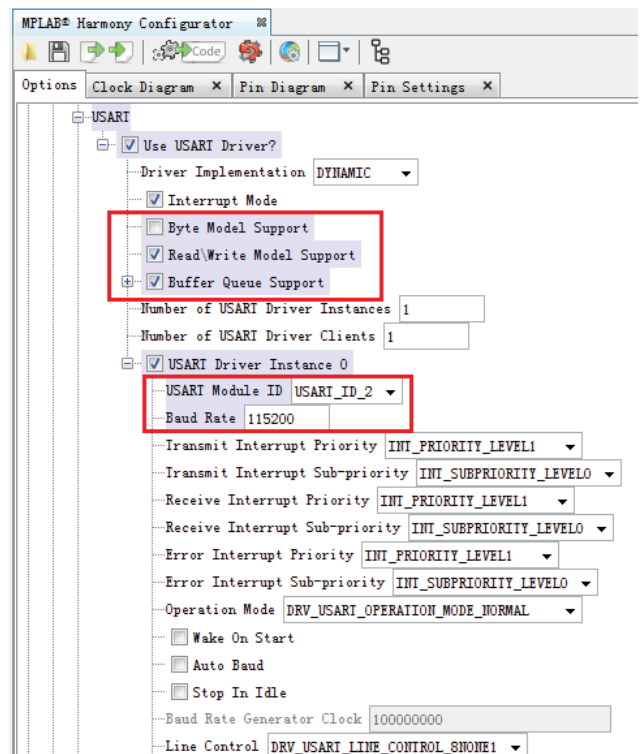


3) 使能系统服务，并在Console系统服务里选择UART：



4) 配置UART驱动：

注意：需要配置UART驱动为Buffer Queue模式，因为Console系统服务使用的是该方式；而且输入使用了数据流方式，所以还得使能“Read/Write Model Support”功能。

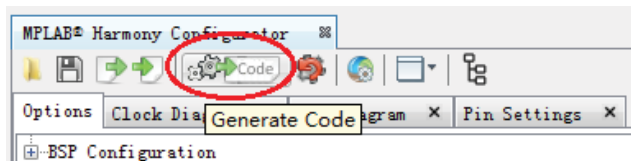


5) UART引脚PPS配置：

Output Pin Table		34	31	26	25	37	38	47	48	49	50	59	60	61
Package: LQFP		B												
Module	Function	2	3	4	5	6	7	8	9	10	11	12	13	14
Clock (OSC_TN_0)	CLKI													
	PGED1													
Debug	PGEC1													
	UZRX													
UART 2 (USART_ID_2)	UZRX													
	UZTX													

Output Pin Table		58	57	128	127	14	15	16	21	140	141	139	1	
Package: LQFP		G												
Module	Function	12	13	0	1	6	7	8	9	12	13	14	15	
Clock (OSC_TN_0)	CLKI													
	PGED1													
Debug	PGEC1													
	UZRX													
UART 2 (USART_ID_2)	UZRX													
	UZTX													

6) 生成代码：



7) 在app.c里增加如下测试代码：

```
void APP_Initialize ( void )
{
    /* Place the App state machine in its initial state. */
    appData.state = APP_STATE_INIT;

    /* TODO: Initialize your application's state machine and
    other
    * parameters.
    */
    SYS_MESSAGE("My first Test Message\r\n");
    SYS_PRINT("Test print: appData.state = %x\r\n", appData.state);
}
```

8) 编译下载测试：

如果有连接到PC，串口助手会得到你的打印信息：

My first Test Message
Test print: appData.state = 0



附注：

建议测试环境：

- MPALB-X V3.26或更新；
- XC32 V1.40或更新；
- Harmony V1.07.1或更新；

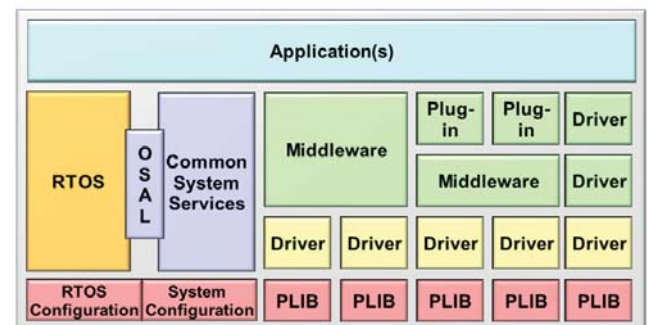
阅读更多
PIC32 Harmony教程



MPLAB® Harmony 之学习篇（八） 利用Harmony已有的 应用示例集成自己的应用

Microchip Technology Inc.
MCU32产品部资深应用工程师
王翀

Harmony框架如下图所示，接下来我们介绍一下Harmony框架下的应用示例以及如何利用已有的示例开发自己的应用程序。



Harmony框架下提供的应用示例：

Harmony下提供了很多应用程序示例。每个应用示例包含一个main函数，一个或多个独立的软件模块（设备驱动，中间件，以及系统服务）。示例展示了Harmony软件架构的各个模块是如何被上层调用的。

打开Harmony安装目录的apps文件夹（\harmonyv1_0x\apps），就可以看到如下目录结构：

名称	修改日期	类型
audio	2016/3/22 14:04	文件夹
bluetooth	2016/3/22 14:04	文件夹
bootloader	2016/3/22 14:04	文件夹
crypto	2016/3/22 14:04	文件夹
driver	2016/3/22 14:04	文件夹
examples	2016/3/22 14:04	文件夹
fs	2016/3/22 14:04	文件夹
gfx	2016/3/22 14:04	文件夹
meb_ji	2016/3/22 14:04	文件夹
programmer	2016/3/22 14:04	文件夹
rtos	2016/3/22 14:04	文件夹
tcpip	2016/3/22 14:05	文件夹
tests	2016/3/22 14:05	文件夹
usb	2016/3/22 14:05	文件夹

• **audio目录：**

包含所有音频处理相关的应用程序示例，包括麦克风，音频解码，USB耳机等。

• **bluetooth目录：**

包含蓝牙相关的应用程序示例，包括基础蓝牙协议栈，蓝牙数据和蓝牙音频的应用示例。

• **bootloader目录：**

bootloader程序是驻留在芯片中的一段代码，它可以帮用户在不用调试/编程工具的情况下，烧写、更新芯片的应用程序。该目录下的应用示例展示了通过USB，串口，以太网来烧写应用程序的bootloader，以及MZ系列芯片的LiveUpdate特性。

• **crypto目录：**

加密程序实例。

• **driver目录：**

动态应用实例，包括I2C，USART，SPI和NVM驱动实例，其中有静态驱动也有动态驱动。

• **example目录：**

包括一些简单的示例，如所有硬件外设和部分系统服务等。

• **fs目录：**

FAT文件系统相关示例。支持的存储介质包括芯片内部Flash，SD卡和SQI Flash。

• **gfx目录：**

图形显示相关示例，包括图形原语和控件的显示示例。

• **meb_ii目录：**

MEB2是Microchip PIC32处理器配套开发板，具有蓝牙，WIFI，LCD显示等多种功能。

• **programmer目录：**

包含外部Flash读写示例，可以通过它烧写外部SPI或SQI Flash。

• **rtos目录：**

操作系统示例，包括FreeRTOS，ucOS，OpenRTOS等等。

• **tcpip目录：**

网络相关的示例，如TCP，UDP client/server示例等。

• **test目录：**

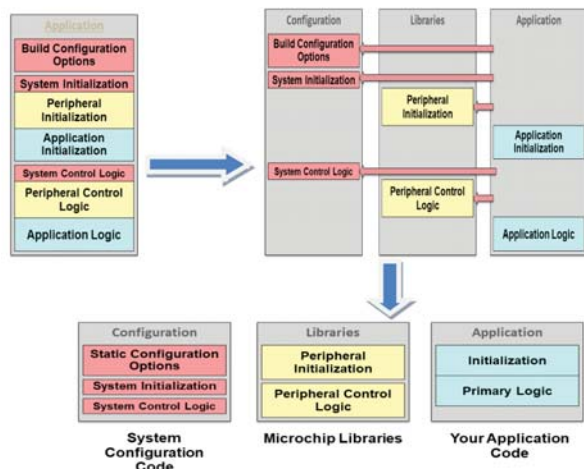
该目录下的示例用来测试操作系统或无操作系统下，各个软件库是否能正常工作。

• **usb目录：**

包含大量的USB主从设备应用实例。

理解Harmony下应用示例的架构：

接下来讲一下Harmony框架下应用的结构。我们说，Harmony下的应用程序是经过重构的。重构过程如下：



之所以Harmony下的应用要满足以上结构划分，主要是因为重构后的应用程序，软件都是模块化，易于软件功能的添加和删除。系统代码和软件库是Microchip来实现的，用户只需要编写应用代码即app.c。

一个应用软件主要有这4部分组成：

1. **App.c：**
客户业务相关代码
2. **Main.c：**
Harmony下应用程序结构
3. **System_config目录下代码：**
芯片系统相关代码
 - a) **System_exception.c:**
异常处理相关
 - b) **System_init.c：**
系统配置的外设、系统服务和其它软件建模块在这里进行初始化
 - c) **System_interrupt.c：**
系统中所有使能的中断处理函数
 - d) **System_task.c：**
各个软件模块所维护的任务
4. **Framework目录下的代码：**
主要是驱动和系统服务，也包括其它软件库如tcpip栈、RTOS等



重构过程如上图所示，其中系统代码和软件库通过MHC来添加、删除和配置。

Main函数结构如下所示：

```

system_config.h
#define APP_SMC_HW_ID          TMR_ID_3
#define SYS_CLK_FREQUENCY     0x00000200
#define APP_LED_BLINKING_RATE 0x0002625A

system_init.c
#pragma config FLLDIV = DIV_1, FLLMUL = MUL_20, FLLDIV2 = DIV_2, FRTEN = OFF
#pragma config OSC1OPNC = ON, PORSCMD = HS, PROSCEN = ON, FMO5C = PRIFLL
#pragma config ICSEL = ICS_F0X2

void SYS_Initialize( void )
{
}

system_tasks.c
void SYS_Tasks( void )
{
}

system_interrupt.c
void ISR ( _TIMER_3_VECTOR ) _ISR_TMR_3_stub( void )
{
}

int main(void)
{
    SYS_Initialize();
    while(true)
    {
        SYS_Tasks();
    }
    return (EXIT_VALUE);
}
    
```

用户在 app.c 中需要实现两个主要的函数 app_initialize 和 app_task。这两个函数分别被 SYS_Initialize 和 SYS_Tasks 所调用。

总结一下，在 harmony 框架下用户创建自己的应用只需要两步：

1. 在MHC中选择系统软件模块，驱动、系统服务、操作系统、TCPIP协议栈等；
2. 编写应用程序，app_initialize和app_task，通过API调用系统软件实现功能。

如何利用Harmony下已有的应用示例来集成用户自己的应用程序：

了解了Harmony下应用软件示例的结构，就可以利用已有的应用示例轻松集成自己的应用程序。

第一步

将多个已有例的底层软件（系统代码和软件库）通过MHC集成到一起

第二步

将多个app.c和app.h，添加到同一个工程里，并修改相应的函数和文件名为

app1.c, app2.c... appn.c

app1.h, app2.h... appn.h

app1_initialize, app1_task... appn_initialize, appn_task

其中，第一步有一个技巧，用户可以安装两个版本的MPLab X IDE，一个用来打开已有示例MHC，查看配置选项；另一个用来编辑新的工程项目。这样用户只要在自己的工程里把示例的配置选项打上勾即可。

下面我们来实现一个例子，展示代码合并的过程。例子是将 usart_basic (c:\microchip\harmony\v1_06_02\apps\examples\peripheral\usart\uart_basic) 和 msd_basic (c:\microchip\harmony\v1_06_02\apps\usb\host\msd_basic) 两个工程集成起来。使新的应用同时拥有USART通信和USB存储功能。

环境准备：

软件

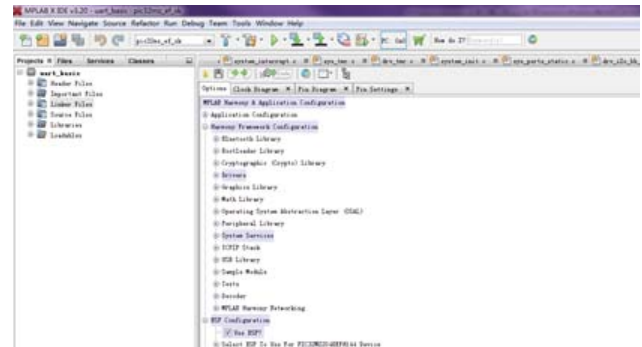
Harmony v1.06.02
MPLAB X v3.20 和 MPLAB v3.26

硬件

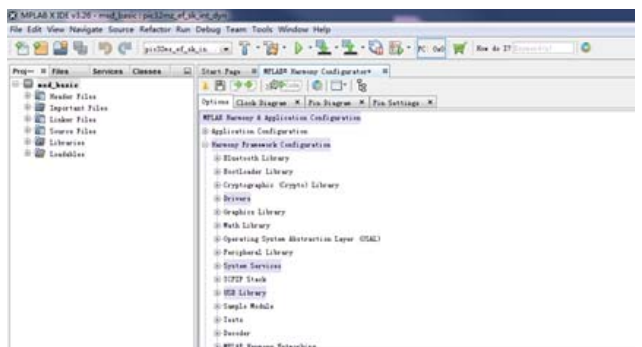
PIC32MZ EF Starter Kit

集成步骤：

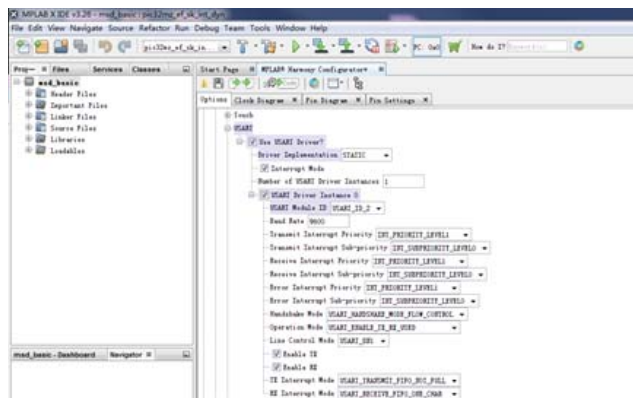
1. 用MPLAB X v3.20打开usart_basic工程

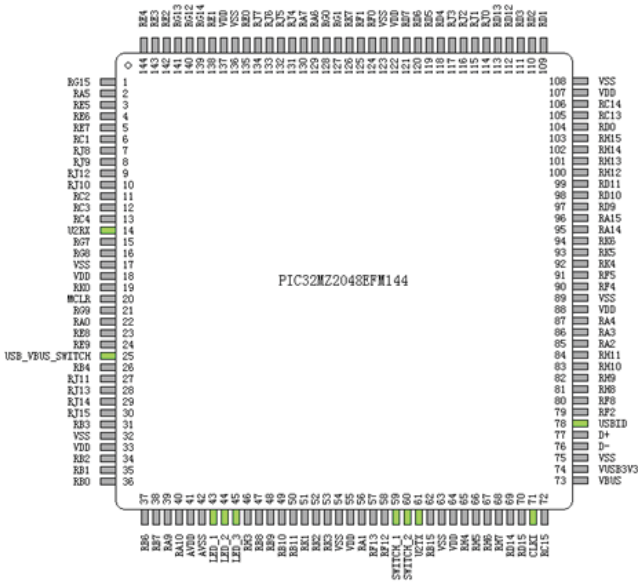


2. 用MPLAB X v3.26打开msd_basic工程，作为基线工即最终的工程)

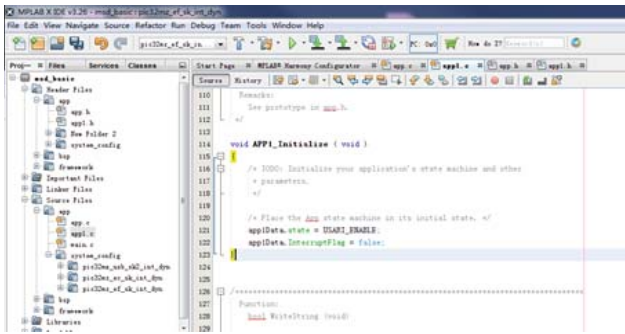


3. 将usart_basic工程的配置项添加到msd_basic工程中去，包括USART2 pin的配置，生成底层驱动





4. 将usart_basic工程中的app.c和app.h改名为app1.c和app1.h，然后添加到msd_basic工程中去。并将这两个文件中所有的app字段用app1代替。如appData改为app1Data



5. 添加APP1_Task和APP1_Initialize到SYS_Task和SYS_Initialize函数中去

```

void SYS_Tasks ( void )
{
    /* Maintain system services */
    SYS_DEVCON_Tasks (sysObj, sysDevcon);
    SYS_FS_Tasks ();
    SYS_TMR_Tasks (sysObj, sysTmr);

    /* Maintain Device Drivers */
    DRV_TMR_Tasks (sysObj, drvTmr0);

    /* Maintain Middleware & Other Libraries */
    /* USBSHS Driver Task Routine */
    DRV_USBSHS_Tasks (sysObj, drvUSBSHSobject);

    /* USB Host layer task routine. */
    USB_HOST_Tasks (sysObj, usbHostObject0);

    /* Maintain the application's state machine. */
    APP_Tasks ();
    APP1_Task ();
}

void SYS_Initialize ( void *data )
{
    /* Core Processor Initialization */
    SYS_CLE_Initialize ( NULL );
    sysObj.usbDevcon = SYS_DEVCON_Initialize (SYS_DEVCON_INDEX_0, G);
    SYS_DEVCON_PerformanceConfig (SYS_CLE_SystemFrequencyGet ());
    SYS_PDRM_Initialize ();

    /* Board Support Package Initialization */
    BSP_Initialize ();

    /* Initialize Drivers */
    sysObj.usbHostObject0 = USB_HOST_Initialize (SYS_MODULE_INIT);
    sysObj.drvUSBSHS = DRV_USBSHS_Initialize (DRV_USBSHS_INDEX_0);
    SYS_INT_FactorPrioritySet (INT_VECTOR_USB1, INT_PRIORITY_LEVEL4);
    SYS_INT_FactorSubprioritySet (INT_VECTOR_USB1, INT_SUBPRIORITY_1);

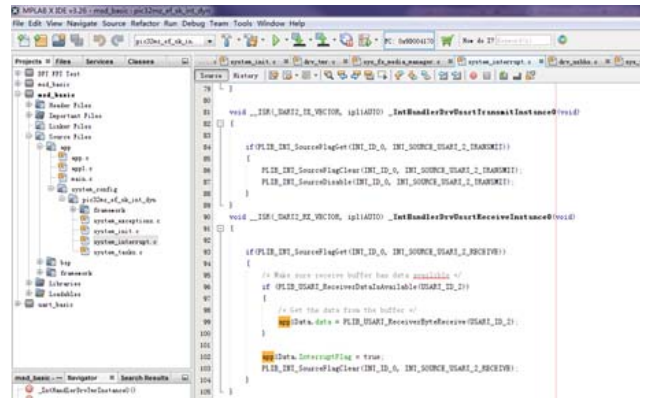
    /* See the priority of the USB DMA Interrupt */
    SYS_INT_FactorPrioritySet (INT_VECTOR_USB1_DMA, INT_PRIORITY_LEV);
    /* See Subpriority of the USB DMA Interrupt */
    SYS_INT_FactorSubprioritySet (INT_VECTOR_USB1_DMA, INT_SUBPRIORITY);

    /* Enable Global Interrupts */
    SYS_INT_Enable ();

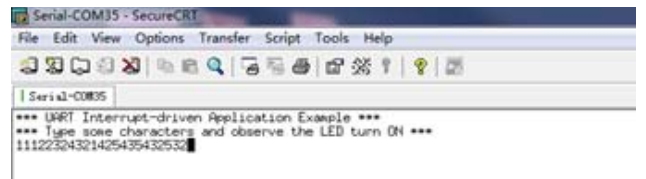
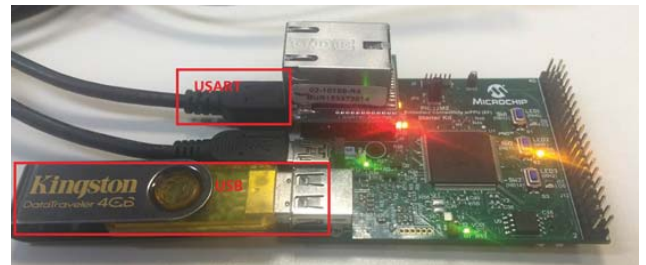
    /* Initialize the Application */
    APP_Initialize ();
    APP1_Initialize ();
}

```

6. 把USART中断处理函数拷贝到新的工程中，并把appData改为app1Data



7. 编译、下载即可调试。此时新的工程就包含了USB存储和USART通信功能



联系我们

联系电话:

- 北京 (010) 8569-7000
- 广州 (020) 8755-8029
- 青岛 (0532) 8502-7355
- 武汉 (027) 5980-5300
- 重庆 (023) 8980-9588
- 香港 +852 2943-5100
- 沈阳 (024) 2334-2829
- 西安 (029) 8833-7252
- 成都 (028) 8665-5511
- 杭州 (0571) 8792-8115
- 上海 (021) 3326-8000
- 厦门 (0592) 2388-138
- 东莞 (0769) 8702-9880
- 南京 (025) 8473-2460
- 深圳 (0755) 8864-2200
- 珠海 (0756) 3210-040

技术支持热线: 800-820-6247 (座机) 或
400-820-6247 (手机)

技术支持邮箱: china.techhelp@microchip.com

Microchip 工程师社区:
www.microchip.com.cn/community



新浪微博
weibo.com/microchiptech

騰訊微博
t.qq.com/microchiptech

microchip
DIRECT
www.microchiptech.com