

前 言

这篇文档的内容来源于广州周立功公司网站的技术论坛“ARM与uC/OS-II”版块。摘录了自从EasyARM2104 试验板公开销售以来约一年内（2003.7~2004.7）的大部分技术帖。

我们在论坛的众多问答中逐条查看，整理出具有代表性的 400 多篇帖子。并对原帖做了适当的修改，包括对回答的修正，以及对语言和排版的处理。

该文档内容包括 ARM 体系结构、EasyARM2100/2104 开发试验板的使用、LPC2xxx 芯片原理与使用和开发工具的使用等众多用户所关心的问题。其中有初学者的疑惑，也有实际工程应用时可能遇见的问题，它汇集了众多用户的心得经验。

我们希望该文档能起到一个抛砖引玉的作用，在读者遇见问题时，能帮助您开阔思路。所以在整理该文档时尽量摘录每一个有价值的帖子，也尽量纠正一些帖子中不妥的回答。但是该 BBS 中帖子众多，难免会遗漏一些对后来者有帮助的问答，而且在已收录的问题中有可能一些回答比较片面，甚至是错误的，我们恳请读者指正。

读者可以访问本公司网站获取更多信息。

公司网站：

<http://www.zlgmcu.com>

BBS:

<http://www.zlgmcu.com.cn/club/bbs/bbsView.asp>

编者

2004-10-10

目 录

第 1 章	体系结构.....	3
第 2 章	编译器与语言.....	6
第 3 章	操作系统.....	28
第 4 章	芯片.....	42
第 5 章	GPIO	56
第 6 章	中断.....	60
第 7 章	串口.....	88
第 8 章	I2C	95
第 9 章	SPI	99
第 10 章	定时器.....	101
第 11 章	存储器.....	104
第 12 章	JTAG	118
第 13 章	其它.....	133

第1章 体系结构

第1问:

Q: 请问在初始化 CPU 堆栈的时候一开始在执行 mov r0, LR 这句指令时处理器是什么模式?

A: 复位后的模式, 即管理模式。

第2问:

Q: 请教: MOV 中的 8 位图立即数, 是怎么一回事? 0xF0000001 是怎么来的?

A: 是循环右移, 就是一个 0—255 之间的数左移或右移偶数位的来的, 也就是这个数除以 4 一直除, 直到在 0-255 的范围内它是整数就说明是可以的!

A: 8 位数 (0-255) 循环左移或循环右移偶数位得到的, F0000001 既是 0x1F 循环右移 4 位, 符合规范, 所以是正确的。这样做是因为指令长度的限制, 不可能把 32 位立即数放在 32 位的指令中。移位偶数也是这个原因。可以看一看 ARM 体系结构 (ADS 自带的英文文档) 的相关部分。

第3问:

Q: 请教: 《ARM 微控制器基础与实战》2.2.1 节关于第 2 个操作数的描述中有这么一段: #inmed_8r 常数表达式。该常数必须对应 8 位位图, 即常熟是由一个 8 位的常数循环移位偶数位得到。

合法常量: 0x3FC、0、0xF0000000、200、0xF0000001。

非法常量: 0x1FE、511、0xFFFF、0x1010、0xF0000010。

常数表达式应用举例:

...

...

LDR R0, [R1], #-4 ; 读取 R1 地址上的存储器单元内容, 且 R1 = R1-4

针对这一段, 我的疑问:

1. 即常数是由一个 8 位的常数循环移位偶数位得到, 这句话如何理解?
2. 该常数必须对应 8 位位图, 既然是 8 位位图, 那么取值为 0-255, 怎么 0x3FC 这种超出 255 的数是合法常量呢?
3. 所举例子中, 合法常量和非法常量是怎么区分的? 如 0x3FC 合法, 而 0x1FE 却非法? 0xF0000000, 0xF0000001 都合法, 而 0xF0000010 又变成了非法?
4. 对于汇编语句 LDR R0, [R1], #-4, 是先将 R1 的值减 4 结果存入 R1, 然后读取 R1 所指单元的 值到 R0, 还是先读取 R1 到 R0, 然后再将 R1 减 4 结果存入 R1?

A: 提示, 任何常数都可用底数*2 的 n 次幂 来表示。

1. ARM 结构中, 只有 8bits 用来表示底数, 因此底数必须是 8 位位图。
2. 8 位位图循环之后得到常数, 并非只能是 8 位。
3. 0xF0000010 底数是 9 位, 不能表示。
4. LDR R0, [R1], #-4 是后索引, 即先读, 再减。

可以看一看 ARM 体系结构对相关寻址方式的说明。

第4问:

Q: 在程序移植的过程中, 什么代码段处于什么样的模式, 这可真是一个困扰人的大难题, 有没有一种标志或办法能够识别“代码段处于什么样的模式”?

A: 读取 CPSR, 任何时候都是可以读。

第5问:

Q: 为什么保护现场时, 总是保护 R0-R3, R12, 为什么不保护 R4-R11?

A: 请看一看“ARM-thumb 过程调用标准”这个文档。

第6问:

Q: 请问 mov R1, #0x00003DD0 错误: out of the range of operation 是怎么回事? 我就是想 IODIR=0x00003dd0, 汇编就是

```
LDR      R0, =IODIR
MOV      R1, #0x00003dd0
STR      R1, [R0]
```

编译时候说是超出操作范围

A: 使用 ldr, mov 的操作数为 8 位位图数。

第7问:

Q: “在 ARM7TDMI(-S) 处理器内部有 37 个用户可见的寄存器:”

问题: “用户可见”应该怎样理解? 这 37 个寄存器是否是 37 个不同的物理寄存器, 例如 R8 与 R8_fiq 应该是两个不同的物理寄存器吧?

A: 用户可见是指用户可以通过程序操作的。R8 与 R8_fiq 是两个不同的寄存器。

第8问:

Q: USR 模式、SVC 模式、IRQ 模式分别有哪些限制? ?

A: 对于外设操作限制与芯片设计有关。USR 模式不能设置 CPSR 寄存器。

用户模式下无 SPSR 寄存器, 代码可以为 ARM、Thumb。

第9问:

Q: 请问“在初始化堆栈时就决定了工作模式”是什么意思? 如何决定工作模式的?

A: 设置 CPSR 寄存器。

第10问:

Q: 请问: ARM 汇编程序设计中所指的“文字池”作何理解?

A: 可以理解为常量数组, 文字池中保存的是常量, 这些常量可以是正常的常量, 也可以是地址。

第11问:

Q: 为什么在中断向量表中不直接 LDR PC, “异常地址”。而是使用一个标号, 然有再在后面使用 DCD 定义这个标号?

A: 因为 LDR 指令只能跳到当前 PC 4kB 范围内, 而 B 指令能跳转到 32MB 范围, 而现在这样在 LDR PC, “xxxx”这条指令不远处用“xxxx”DCD 定义一个字, 而这个字里面存放最终异常服务程序的地址, 这样可以实现 4GB 全范围跳转。

Q: LDR 不是可以全空间跳转的吗? 《ARM 微控制器基础与实战》程序清单 5.3。

A: LDR 伪指令通过设置指令缓冲池才能实现全范围跳转, 而 LDR 指令则只能实现 4KB 范围跳转。

第12问:

Q: ARM7TDMI-S 和 ARM7TDMI 有何区别?

A: ARM7TDMI-S 是 ARM7TDMI 的可综合 (synthesizable) 版本 (软核)。

对应用工程师来说, 除非芯片生产厂商对 ARM7TDMI-S 进行了裁减,

否则 ARM7TDMI-S 与 ARM7TDMI 没有太大的区别, 其编程模型与 ARM7TDMI 一致。

第13问:

Q: DCD 伪指令的疑惑。

“StackUsr DCD UsrStackSize + (USR_STACK_LENGTH - 1) * 4”

这句话是什么意思? DCD 后面的程序标号或数字表达式是何意?

A: 它的内容是初始化递减堆栈的最高地址, 看《ARM 微控制器基础与实战》2.3.2 节。

第2章 编译器与语言

第14问:

Q: 00254: Unimplemented RDI message 是什么错误提示? 我的设置连接都正常, 是不是芯片烧了?

A: 是 JTAG 的问题。可以先使用 ISP 操作试试就知道了, 如果能 ISP, 说明 LPC2104 没有损坏, 还能正常运行程序。

第15问:

Q: 请教: 我在调试程序的时候在 AXD 中出现这样的提示信息:

RDI Warning 00159: could not open specified device port.

我是根据配套教程的步骤设置的。

A: 请按照光盘\easyarm_drive\readme.txt 安装驱动程序。

第16问:

Q: 我用实验程序运行经常出现下列信息! 程序不能下载到目标板。

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!

A: 1. 仿真器配置一定要正确, 即“EasyARM Configuration”设置窗口中的“FLASH”项中选择“Erase Flash when need”;

2. 向量表累加和要为 0;

3. 可以先在 RAM 调试一个程序(运行), 然后 STOP, 再使用 File->Load Image... 加载要下载到 FLASH 的调试文件。

第17问:

Q: 在 ADS 中是否可以软件调试基于 UCOS-II 的程序?

A: ADS 软件调试只能调试 ARM 的内核, 不能调试外设。但是取消 PLL 锁定检测后, 可以调试任务切换, 最终到空闲任务上。开始移植时软件仿真是最好的工具。

第18问:

Q: ARMulate 软件是干什么的? 2104 不是用 EasyJTAG.dll 来仿真吗?

A: 软件仿真只能仿真 ARM 核。

第19问:

Q: 有关 LPC2106.INC 的问题。我无法在 project 引用 lpc2106.inc 文件, 只能引用 lpc2106.h 文件, 这是什么原因? 且当我的主程序用汇编编写时, 不能引用 lpc2106.h, 用 lpc2106.inc 则无法加入 project, 请问汇编器应如何设置?

A: 不用加 2106.inc 只要该文件在你的工程文件夹中, 就可以直接在汇编程序的开始处加

“include 2106.inc”。

注意：该文件是汇编文件定义的头文件，定义内部寄存器。

第20问：

Q: 入口点是什么意思？我在使用 LPC2106 上移植 UCOS-II，每次 MAKE 时总是提示我 Image does not have an entry point，可是我是把光盘的 vetctors.s 复制过来的，而且仔细看了看，已经声明了 ENTERY，这是怎么回事？

A: 需要在 ADS 中设置入口。

第21问：

Q: 请教：如何定义不被初始化变量？

A: 让编译器不知道有这个内存地址即可。

A: 如用分散加载文件分配 RAM 故意预留一部分 RAM 不分配，用它来存您不需要初始化的东西。或者不调用编译器提供的启动代码，不过这样可能编程会麻烦一些。

第22问：

Q: 我直接通过 JTAG 口下载 EasyArm 板带的 Ext1_test 程序到 ARM 中，出现中断向量的告警：
interrupt vector is not correct arm is not running freely。

果然复位后芯片不能运行。但是我用串口下载后芯片能正常工作，中断也行的。

并且我用 JTAG 仿真的话，芯片能正常工作，中断也行的，唯独 JTAG 口下载不行。

不知道是什么原因？

A: 仿真器配置中要设置 Erase Flash when need。也可以这样试试：

1. 可以先打开一个工程在 RAM 中调试运行；
2. stop 程序；
3. 使用 File->Load Image...重新加载 Ext1_test 生成的*.axf 文件。

Q: 仿真器配置中我是设置了 Erase Flash when need，但照你说的话，那不是在 RAM 下调试吗？

在 RAM 下调试我是可以的，但是下载后出现 interrupt vector data is not correct。

我又看了几篇文章，是不是跟中断向量表的累加和不为零有关系啊？

A: 是的，是向量表的累加和不为零。

因为如果用 ISP 下载能运行，说明向量表的累加和已为零，而用 JTAG 下载不能运行的情况可能是 没有正常下载代码。先在 RAM 中调试，目的是为了后面正确下载程序到 FLASH。

第23问：

Q: 用 Scatter 怎样将某个函数或文件定位在 Flash 的某个位置？

A: 参考 <http://www.zlgmcu.com/download/downs.asp?ID=1009>

第24问:

Q: 我在仿真时遇到这样的提示:

Error, Flash is protected by user configuration!

怎么写到 flash 里面呢?

A: 看配套《ARM 微控制器基础与实战》附录一。

第25问:

Q: 我在移植实验中想到了两个问题, 如下:

1. Debug 和 Release 以及 DebugRel 有什么不同, 为什么在作 2104 移植实验时, 要用 Release?

2. 在 Release 中为什么要将 RW Base 设置为 0x40000040? 我将其设置为 0x40003000, 为什么不能工作?

A: 都只是一个问题, 内存空间的使用, 因为跑 OS 要比较大的内存空间, 所以要腾出点地方。

第26问:

Q: 请问没有 MMU 的 ARM 芯片是否支持使用 malloc() 函数动态分配内存?

A: 是否支持 malloc() 函数与芯片没有多大关系, 主要与编译器有关。

Q: 再问: 如果没有操作系统支持呢?

A: 也支持。

第27问:

Q: 在 I2C 实验程序中, 我想查看数据缓冲区 DataBuf 的值, 怎么查看?

A: watch 窗口或鼠标停留在要查看的变量名上。

Q: 我查询的是写入 DataBuf 缓冲区的值, 鼠标在上面根本就不会出现他的值, 即使在 watch 中加入,

结果也是 "name not found"。

A: 变量被优化, 调试时可以把该变量定义为全局变量查看。

第28问:

Q: 仿真软件和 2104 开发板连接不上?

DBE Warning 00041:

!An unspecified Debug Toolbox call failed

电源和开发板都连好, 错误和没接开发板一样, 驱动也安装了, 安装时按确定键时, 软件很长时间才有如上反应, 请帮忙?

A: 1. 串口是否正常?

2. 在其它操作系统 (如 98) 下或其它台式 PC 下试试。

第29问:

Q: 如何生成 32 位 hex 文件?

我在 Release Setting->ARM fromELF->Output Format 中设置为 Intel 32bit HEX, 可是好像没有生成 hex 文件?

A: 试试这种方法:

Target->Target Setting->Post Link 中选择“ARM fromELF”加上你上面设的应该不成问题。

第30问:

Q: 请问关于 settings 中 r0 base rw base 的意思?

A: ro: read only, rw: read and write.

第31问:

Q: 编译成功后的信息第一行, code, R0 data, RW data, ZI data, debug 分别代表什么?

A: R0 只读段, 即程序代码空间;

RW 可读/写段, 即数据变量空间;

ZI 清零变量段, 即需要清零初始化的数据变量空间。

第32问:

Q: 如何在 ADS 里面看任务执行的一些情况? 比如堆栈。

A: 多任务环境下的堆栈, 内存等信息需要调试软件的支持才可以实现。

ucos 下有一个统计功能的模块可以间接实现部分功能。

第33问:

Q: 请问向 flash 烧数据时出现: exceeds flash limitation 请予赐教!

A: 要写入的 flash 地址超过了范围。如果不是代码太大的问题, 可以检查 scf 文件是否正确。

第34问:

Q: 在 LPC2214 之类的芯片中如何实现数组的绝对地址定位, 比如 51 的_at_的用法。

A: *((char*)0x40000300)类似访问

Q: 谢谢, 但这样做就无须定义数组变量, 访问也不便, 还有高招吗?

A: 可以使用分散加载。

第35问:

Q: 请问 ADS 编译错误“L6221E: Execution region ER_R0 overlays with Execution region ER_ZI” 该如何解决?

A: 请用我们网站上的工程模板试一试, 最大的可能是因为你的 RELEASE 或者 DEBUG 选项里面没有正确设置, 按照参考《ARM 微控制器基础与实战》上面的设置, 是不会有这个问题的。

第36问:

Q: 请教一下: 将程序写入 flash, 再用从 JTAG 方式调试写入之后再复位程序没什么反映。看了很多以前的帖子, 说 memmap 寄存器要为 1, 我用的是一个很简单的控制 led 的例子, 改动了参数之后写入 flash 的。在这个程序的 vectors 中找不到关于 memmap 寄存器操作的部分啊, 这是怎么回事, 该怎么办呢?

A: 《ARM 微控制器基础与实战》上附录有“常见问题”, 列举了几点程序写到 FLASH 不能运行的原因。 memmap 操作可以在 target.c 中的 TargetResetInit() 函数内添加。

第37问:

Q: HEX 文件。EASY2100 配套《ARM 微控制器基础与实战》上讲: 把项目编译成 HEX 文件, 我不会呀, 咋办?

A: Target-> target settings 设置 Post-linker 并且设置 Linker-> fromELF。

Q: 再问: 我用的是光盘里的例子, 打开工程项目里是 DebugInARM。DebugInFlash。RelInFlash。不是《ARM 微控制器基础与实战》上的 DebugRel 呀, 我都照《ARM 微控制器基础与实战》上设置的, 可用 ISP 下载, 提示无法找到 HEX 文件。

A: 光盘上的例子是用专用工程模板建立, 已经设置好参数, 与默认模板不同。

第38问:

Q: EasyARM2100 开发板如何通过 JTAG 接口下载到 Flash?

我在用 EasyARM2100 开发板时 JTAG 接口不能进行 Flash 中的调试 (DebugInFlash), 但是可以在 RAM 中调试 (DebugInRAM), 且通过 ISP 编程可以写入 Flash。

通过 JTAG 接口进行 DebugInFlash 调试时, 总是出现如下提示:

```
Flash Sector 0 write failed!  
Warning interrupt vectors data is not correct!  
Program you downloaded can not run freely!
```

请问是什么原因? 我该如何办?

A: 仿真器设置选项有一个允许擦除 FLASH 的选项, 选择它。

第39问:

Q: 我发现程序在 RAM 调试时 (R0=0X40000000) OK, 但是 JTAG 下载到 FLASH (R0=0X00000000), 显示如下:

```
The session file 'c: \Documents and Settings\jan\default-1-2-0-0.ses'  
could not be loaded.
```

A: 这是 ADS 自身的问题, 请不要通过 IDE 运行 AXD, 而是通过开始菜单运行, 然后 Load 调试文件调试。

第40问:

Q: 我有几个问题想问问大家。

调试主机负责对 ARM 源程序进行编译链接，最好用什么样的高级语言对 ARM 源程序进行编译链接？ 使用调试程序（如 AXD）进行 JTAG 调试，AXD 是什么调试程序，是否是类似什么软件之类的？到哪能下载 AXD 的调试程序呢？怎么样进行 AXD 程序进行 JTAG 调试？

如何通过 JTAG 仿真器发送到目标机上呢？

对于单片机的调试，都用到了什么软件？

都个有什么作用呢？

A: 可以使用 ADS1.2, Keil C 也支持。AXD 是 ADS 的一个组件。

您可以在本公司网站下载 EasyARM2100 开发套件快速入门看一看。

第41问:

Q: EasyARM2100AXD 调试时出错，故障现象:

按照光盘上的方法设置好 ADS1.2 后打开光盘上的 expamples\gpio\c\LedDisp 的工程文件(从光盘上拷贝到硬盘，已去处只读属性)。

编译通过后，按 Debug 按键运行调试，进入 AXD。按 Go 按键，EasyARM 没有反应。再按 Stop 按键。

AXD 弹出两个确认框窗口:

“No disassembly could be read at the requested address”.

如果进入 AXD 直接按 Step 按键，也是会弹出同样的两个确认框。

软件是光盘上的，没有经过任何改动。光盘上的程序试过 4, 5 个都是这种情况。

AXD 的设置是按照光盘上的说明设置的。

EasyARM 上的 JP8 是断开的。ADS 在出现这种问题后重新安装过，故障依旧。

出现这个问题前，可以调试。只是单步的时候感觉比较慢。差不多 1-2 秒钟才能单步一次。

A: 原因找到了，是芯片被加密了，无法写入新的程序，用 FLASH ISP 清除后解决。

加密后 JTAG 完全不能控制芯片，否则可能被解密。

第42问:

Q: 我板上的 lpc2214 开始可以在线编程，只搞了两三次可以把文件写进去。但我发现 P0.14 未置低电平时也进入 ISP 模式，烧入的文件原来可以 ISP 下载到 2014 开发板中运行的。最后只能读芯片的一些 ID，载文件都不能进行。以下是对整片 ERASE 时，对串口的捕获，命令返回是 19。

Synchronized

Synchronized

OK

11059

OK

U 23130

0

P 0 14

0

E 0 14

19<-----在表中查不到该值

附表:

- 0 CMD_SUCCESS 命令被成功执行。
 - 1 INVALID_COMMAND 无效命令
 - 2 SRC_ADDR_ERROR 源地址没有以字为边界
 - 3 DST_ADDR_ERROR 目标地址的边界错误
 - 4 SRC_ADDR_NOT_MAPPED 源地址没有位于存储器映射中。计数值必须考虑可用性。
 - 5 DST_ADDR_NOT_MAPPED 目标地址没有位于到存储器映射中。计数值必须考虑到可用性。
 - 6 COUNT_ERROR 字节计数值不是 4 的倍数或是一个非法值。
 - 7 INVALID_SECTOR 扇区号无效或结束扇区号小于起始扇区号。
 - 8 SECTOR_NOT_BLANK 扇区非空。
 - 9 SECTOR_NOT_PREPARED_FOR_WRITE_OPERATION 为写操作准备扇区命令未执行。
 - 10 COMPARE_ERROR 源和目标数据不相等。
 - 11 BUSY Flash 编程硬件接口忙。
 - 12 PARAM_ERROR 参数不足或无效参数。
 - 13 ADDR_ERROR 地址没有以字为边界。
 - 14 ADDR_NOT_MAPPED 地址没有位于存储器映射中。计数值必须考虑可用性。
 - 15 CMD_LOCKED 命令被锁定。
 - 16 INVALID_CODE 解锁代码无效。
 - 17 INVALID_BAUD_RATE 无效波特率设定。
 - 18 INVALID_STOP_BIT 无效停止位设定。
 - 19 CODE_READ_PROTECTION_ENABLED (在英文手册上找到的)。
- A: 搞定了, 在 philips 网上下载最新的 ISP 软件就 OK 了, 附网址
http://www.semiconductors.philips.com/files/products/standard/microcontrollers/utilities/lpc2000_flash_utility.zip

第43问:

Q: RDI Warning 00254: Unimplemented RDI message 是什么意思?

A: 我遇到过, 可能是电源, 因为我连接了一个功耗比较大的设备, 把电压拉低了 0.2V。单独供电即可。

第44问:

Q: 关于工程模板的问题。

我在用 zlg 提供的工程模板建立工程后。能够不添加文件, 直接进行编译吗?

我试了以后 ads 提示:

Undefined Symbol main(referred from kernel.o)。这是什么道理呢?

A: 没有定义 main 函数, 只要添加了 main 函数就可以编译了。

第45问:

Q: 请教: 单步运行程序设断点的问题。

我单步运行程序在有的地方设断点，为什么老是会出来这样的问题：

RDI Warning 00148: Can't set point.

我用的是 LPC2104，不知哪里的问题？

A: EasyJTAG 只有 2 个硬件断点。可以使用 Run To Cursor 命令使程序运行到当前光标处，免去设置/取消断点操作。

第46问：

Q: 使用几次 RelInFLASH 后不能正常工作，包括其他两种方式也不能工作在 RelInFLASH 和 DeBUGinflash 时显示一个错误

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!

不知怎么回事？使用 DebugInRAM 虽然不报错，但无法正常工作。

A: 使用 ISP 擦除芯片可以解决您的问题。

第47问：

Q: 请问在没有开发板的情况下怎样设置 ADS 才能调试程序啊？也就是能够单步执行程序！谢谢！

A: 选择 option -> configure tar... -> ARMUL ， 只能仿真 ARM 核外设不能仿真。

第48问：

Q: 为什么盘中的工程会出错啊？

A: 文件的只读属性去掉了么？

第49问：

Q: 您在 2104 的《ARM 微控制器基础与实战》中提到 ucos 与应用代码分开编译，在分开编译调试成功之后，最后要将代码统一固化到芯片中。在最后一步需要注意什么？统一编译时感觉要改动很多东西，有没有什么最简单的办法？能否详细介绍一下方法？多谢！

A: 使用我们的工程模板，可在本公司网站下载。其实不分开编译也可调试。

<http://www.zlgmcu.com/tools/kaifaban/EasyARM2104.asp>

的 EasyARM2104 开发套件快速入门和 LPC210...

第50问：

Q: 请问，我的 2104 板子，通过 EasyJTAG 仿真时，在 AXD 上没有文件，而且出现这样“Error, Flash is protected by user configuration!”

的提示，这是怎么回事啊？在 AXD 应该打开什么类型的文件啊？谢谢啦!!!!

A: 在仿真器的配置窗口设置仿真器允许擦除 FLASH。

AXD->Options->configure target->configure->erase.....

第51问:

Q: 关于 arm 汇编语言跳转指令的特殊用法。有如下两条跳转指令:

beq lablef

beq lableb

其中 lable 为某段程序的标号, beq lablef 表示向前跳转到与当前指令最接近的标号 lable 处执行, 而 beq lableb 表示向后跳转到与当前指令最接近的标号 lable 处执行。

在 arm 汇编中有定义这样的用法吗?

A: 没有。

第52问:

Q: 我将一段程序从 flash 复制到了 RAM 中, 但是因为混合编程中不能直接向 PC 寄存器中赋值来实行跳转, 这样我该如何跳转到这个 RAM 中的地址呢?

A: 用函数指针。可参考 IAP 例子。

第53问:

Q: 我想详细的了解一下 ADS 开发工具中的 stack.s 、 heap.s 和 startup.s 中代码的含义, 各位高手推荐本书或给解释一下?

A: 开发套件用户指南的第 3.1.3 节有说明。

 stack.s 定义了系统模式堆栈的起始地址。

 heap.s 为初始化库函数的堆, 是按 ADS 的编译器要求编写的。

 startup.s 向量表及初始化代码, 是根据 CPU 来编写的。

第54问:

Q: 我不知道所选择的 C 文件到底是用 ARM 还是 THUMB 编译的, 请大侠指点!

A: 看配套《ARM 微控制器基础与实战》7.1、7.2、7.3 节。

第55问:

Q: 我重新建立了一个工程, 没有用模板, TEST.C 程序在编译的时候提示 OS_EVENT 没有申明, 但是我看了代码, TEST.C 中包含了 CONFIG_EX.H, 而 config_ex.h 中又包含了 include_ex.h, include_ex.h, 这两个头文件又包含了 uc0s_ii.h, 而 OS_EVENT 是在 uc0s_ii.h 中申明的, 不知道为什么会出这种错误, 能不能给我一个比较合理的目录结构?

A: 请参考我们光盘的目录结构。最好使用我们的工程模板。

第56问:

Q: 请问, 我现在做 2114 的产品开发。出现以下问题, 特向您请教! 问题:

我们开发的产品要采样, 采用周期固定。于是我们采用 timer0 进行时钟触发, 利用触发中断程序 每隔 10 毫秒采样一次并显示波形。这都能很好的进行。但是在中断程序里面对几个全局变量赋值, 跳出中断后却不能使用该全局变量。很是头疼! 请赐教!!

框架如下:

```
uint8 a; //全局变量
void __irq funtimer0()
{
    ...
    a=0; //在中断函数里面对全局变量赋值
    ...
}

void fun1()
{
    uint8 i;
    i=0;
    .....
    i=a; // 出错位置, 在调试过程中运行到当前位置, 鼠标移动到变量 a 上
        // 显示数值是 0, 但是不能传给 i, i 的值不会随该命令改变。
    .....
}
```

因此, 我在中断函数中采集到的数据放到全局数组中。采集完成关中断后在用户模式下的程序中来处理该数组, 但是该数组中的数据也不能使用! 全局数组变量占 1000 个字节, IRQ 堆栈长度是 256。

A: 全局变量用 volatile 声明。

第57问:

Q: 在 AXD 里打开 Debugger Internals, 在 Variable Name 里为什么没有 PINSEL2, ADDR, ADCR 等寄存器? 要在哪里设置?

A: 有些片内外设寄存器是不能读出, 《ARM 微控制器基础与实战》上的 P33 页有说明及处理方法(在 Memory 窗口写入相应寄存器地址)。

第58问:

Q: 请问不连 easyArm 板可不可以直接软件调试程序?

我是指如果想直接调软件部分, 能不能在 ads 或 axd 里直接调试?

A: 这和板子没关系啊, 你直接选 ARMulate 不就行了, 注意把等待 PLL 配置完那条语句屏蔽就可以啦!

第59问:

Q: AXD 里面出现 “RDI Warning 00159: Could not open specified device port”

谁能告诉我这个问题啊我一直搞不懂啊, 全部是按照《ARM 微控制器基础与实战》第 4 章上配的图做的啊, 但是就是不能仿真啊。

A: 请先按照光盘的 easyarm_drive\readme.txt 安装驱动程序。

第60问:

Q: 1. 在《ARM 微控制器基础与实战》的 2.6.3 节有说到“浮点数寄存器(F0-F7...)”, 2104 是否具有这些寄存器?

2. 如果 EasyArm 不支持浮点运算, 而我的程序需要用到浮点运算, 请问可以实现吗?

A: 1. 没有。

2. 用 C 就可以用浮点运算。

第61问:

Q: EASYARM2104 的例子程序怎么都是调用 C 写的程序, 汇编那个怎么没用的啊?

A: 按如下步骤进行处理:

1. 在项目管理窗口中删除原来的所有文件;

2. 在项目管理窗口中增加汇编文件*.S;

3. 编译链接, 调试。

第62问:

Q: 请问版主: 在 AXD 调试软件中, 我单步运行到一定时候, 我想让自己编的软件从复位处重新运行, 在 AXD 软件中, EXECUTE 栏目中有单步, 连续, 运行到光标处等功能, 但无复位功能, 如果要实现此功能, 如何操作?

A: 我也没有发现, 目前只能用重新装载的操作实现。

第63问:

Q: 在 AXD 调试中, 通过从 jtag 接口, 连接上 lpc2106, 出现如下信息:

TKSimulator for ADS, V1.2, 2003/08

Software Supplied by: ZLGMCU

ARM7TDMI-S, Little Endian

在从 file|load image... 文件, 出现两种情况:

1. 正常, 完全能调试;

2. 出现问题, 如下:

RDI Warning 00254: Unimplemented RDI message

请问, 这是错在哪里? 如何解决? 谢谢!

A: 可能是电源或其他接插件接触不良引起。

第64问:

Q: 请教几个问题:

1. 移植中底层接口里的 __swi 关键字在 SDT 中能用吗?

2. 为什么在勘误文档中时钟节拍服务子程序里去掉了开中断及关中断的宏?

3. 时钟节拍中断的优先级应该设为最高吗?

- A: 1. 不知, 请自己看一看软件自带文档。
2. 因为中断服务程序中肯定是关中断的。
3. 不必。

第65问:

Q: arm 汇编中的中括号是什么意思? 比如下面的例子中括号是什么作用呀?

```
[ PLLONSTART
    ldr    r0, =PLLCON
    ldr    r1, =((0xe8<<12)+(0x4<<4)+0x2)
    str    r1, [r0]
]
```

A: 等效于:

```
IF PLLONSTART
    ldr    r0, =PLLCON
    ldr    r1, =((0xe8<<12)+(0x4<<4)+0x2)
    str    r1, [r0]
ENDIF
```

第66问:

Q: AXD 中已经设置过了: 用了下载的“EasyJTAG_Driver_V103”, AXD 的“Choose Target”也显示了 “EasyJtag Debugger for ADS1.2 interface”。硬件的连接上: JP6 是 PCSEL, JP7 是 PCDBG。用 EasyJTAG 把 2104 和 PC 连接完毕->上电->启动 AXD。现象: 数码管能显示不断变化的数字 0--F, 但是 AXD 的连接失败, 提示“DBE Warning 00041:”。重复试了几遍, 现象相同。

请教原因以及对策。

A: 刚才把我机器的 Win2K 系统的用户属性从“PowerUser”修改为“Administrator”就没事了。可能是 ADS/AXD 安装的时候用了 Administrator 权限, 在“PowerUser”下工作有问题吧。

如果哪位老兄有类似问题, 不妨试一下这个方法。

第67问:

Q: 在 Init.s 中有这样一段:

```
Reset
    BL    InitStack           ;初始化堆栈
    BL    TargetResetInit    ;目标板基本初始化
    B     __main              ;跳转到 c 语言入口
```

谁知道__main()函数的具体内容?

假如我不需要调用库函数的话, 是否可以改为 B main?

A: 看调试是汇编代码里有__main(), 组成成分与一些链接选项有关。
可以。但是全局变量初始化不了。

第68问:

Q: 我在 C 中嵌入这样一条语句

```
__asm
{
    MRS R4, CPSR;
    STMFD SP!, {R4};
    ORR R4, R4, 0x80;
    MSR CPSR_cxsf, R4
}
```

会出新报错 “illegal write to sp” 版主和各位大侠有什么办法解决阿?

A: 嵌入汇编不能使用 SP。

第69问:

Q: 请问 ADS 如下的编译错误是什么意思?

```
L6221E:Execution region ER_RO overlays with Execution region ER_ZI
```

A: 程序段内存分配可能有问题, RO (只读), ZI (0 初始化)

第70问:

Q: 在 axd 中有没有运行程序的时间计算器? keil 中就有那样的东西。

A: 有的, 不过不是时间而是执行周期, 可以换算成时间。在 debugger internals 中, 具体可以参考帮助文档。

第71问:

Q: 我最初的 init.s 中没有加入 heap 的分配和 __user_initial_stackheap 函数, 结果程序跳到 SWI 中死循环。我把他们加入后就好了。或者不加入他们, 把 B __main 改为 B main 也可以。请问何解? 另外, 就算目标板会陷入死循环, 但是用软件仿真却可以正常运行。何解?

A: 堆的位置没有分配到有效的 RAM 中。

第72问:

Q: 在 ARM 汇编语言中, 对立即数的有要求, 我记得原来没有这样一说了。现在反而糊涂了, 用立即数时候, 很小心很忌讳, 不回象 51 下随心所欲的使用了, 有对立即数熟悉的朋友站出来给大家释疑。

A: 如使用 “非法数据可以使用 lrd 送到寄存器” 的方法。

首先在存储器中定义一个常量, 再 ldr 进去, 编译器是这样弄的。可以看下面语句反汇编的区别:

```
int z=0x101;
int z=0Xff;
```

第73问:

Q: 请教各位: 我在 AXD 中单步运行一段程序后, 想回到程序的开始重新运行, 除了重新加载,

还有其它方法或按键?

A: 你的程序有多大? 程序在 Flash 运行点击 Reload Current Image 也不需要 1 秒吧~?
或在代码窗口 Set PC=0?

第74问:

Q: 在《ARM 微控制器基础与实战》程序清单 6.22 的 376 页的 OSIntCtxSW_1 中:

```
LDR    R4, [R6]
ADD    SP, R4, #68
LDR    LR, [SP, #-8]
MSR    CPSR_c, #(NoInt | SVC32Mode)
MOV    SP, R4

LDMFD  SP!, {R4, R5}
LDR    R3, =0sEnterSum
STR    R4, [R3]
MSR    SPSR_cxsf, R5
LDMFD  SP!, {R0-R12, LR, PC }^
```

请问高手: LDMFD SP!, {R0-R12, LR, PC }^ 是不是恢复新任务工作寄存器和工作模式, LR 寄存器一并得到恢复, 那么 ADD 指令后面的 LDR LR, [SP, #-8] 指令是不是可以去掉?

A: 两者恢复的是两个不同处理器模式的 LR。

第75问:

Q: 既然 option 页中的 Image entry Point 填入的是调试入口地址, 那么在实际的程序运行当中它是不会覆盖代码中的 ENTRY 入口声明的, 对吗? 仅仅是为了调试的方便。

A: Image entry Point 优先。其实代码中的 ENTRY 是为了确保代码不被优化掉。

第76问:

Q: 我想使用标准 C 语言的库函数, 比如 memset 在 string.h 中有定义, 可是我直接包含 #include <>, 但是编译没错误, 运行确有错误, 应当怎么设置????

A: 请使用最新的工程模板

第77问:

Q: 软件中断是不是必须由 SWI 指令触发?

A: 可以直接用你定义成软中断的函数名啊。比如:

```
__SWI(0x12) void myswi(void); // 声明函数

__asm // 调用方式 1
{
    swi 0x12
}

myswi(); // 调用方式 2
```

第78问:

Q: 《ARM 微控制器基础与实战》程序清单 6.9 的程序是在哪里被调用的?

```
__user_initial_stackheap
    LDR r0, =bottom_of_heap
    MOV pc, lr
```

A: __main。千万别删哟，否则出大事的。

第79问:

Q: “LDR R0, =PINSELO” 中“=”是什么意思？这语句是取地址还是取地址中的内容呢？

A: 这是 LDR 伪指令，用来加载 32 位立即数或地址，LDR R0, =PINSELO 是将 PINSELO 的地址加载到 R0 中。

第80问:

Q: 我不明白“SWI 0”和“SWI 0X123456”这两条指令中的 0 和 0X123456 中有何用。有没有应用 SWI 的具体例子让我看看？

A: 那是特定的中断入口地址，见 ADS_DeveloperGuide_D.pdf 下的 swi。

第81问:

Q: SWI 的功能表吗？

A: SWI 的功能表由 swi 异常服务程序决定。很多时候由编程者自己决定。

第82问:

Q: 请问：我在实验您的原代码进行 ucos 移植时用的是例 1，但是在按照图 7.17 设置处理器的仿真器模式时 enable comms channel view 和 semihostin 项为不可激活状态。为什么，能否给一点提示？

A: 在一些仿真器上使能了它们会影响 swi 异常处理程序。

第83问:

Q: 请问“Unimplemented RDI message”这个出错提示是什么意思啊？

还有，有的时候当我用 axd load 一个 .axf 文件时，常常 loading 的没完没了，这是怎么回事啊？

A: Unimplemented RDI message: 为命令操作失败，需要重新连接。

下载不结束为出现不正常现象，请重新连接并下载。如果每次都出现该现象请联系我们的技术支持。

第84问:

Q: 各位高手好，我是 ARM 初学者我在程序调试中“LDR PC, [PC, R2]”命令执行后 PC 为什么 =0x0000000c，在这条命令执行前 PC+R2 地址上的值是 0x0000000c 吗？

A: 要使用软件仿真。0x0000000C 是预取中止了。

Q: 请问：ARM7 在初始化 CPU 堆栈时，寄存器 CPSR 和 CPSR_c 有什么关系？CPSR_c 是在那里定义的？

A: 这是 MSR 指令的语法，“_”后部分指定 CPSR 的域，请参考《ARM 微控制器基础与实战》上关于这条指令的说明。

第85问:

Q: 软中断 SWI 作底层接口的问题。以下函数为啥要通过软中断调用, 可以直接调用吗？

```
OS_TASK_SW(),
_OSStartHighTdy(),
OS_ENTER_CRITICAL(),
OS_EXIT_CRITICAL(),
```

A: 在用户模式或 Thumb 状态不能直接调用。

第86问:

Q: ARM 的一条指令是 32bit 长，但有时一个立即数也是 32bit，这是如何解释的？

A: 指令中使用的立即数需要时 8 位数移位获得，并非所有数都可以。

第87问:

Q: 请教：全局变量的值在复位（不断电）后会不会自动清零？（用光盘自带的启动代码）

A: 在 C 语言中一般会初始化 0 或用户指定的值，但这不是硬件自动的。

第88问:

Q: the setting files for *.mcp is locked! 是什么问题？

A: 文件属性只读。

第89问:

Q: 请问, 关于 printf() 在 ads 中怎么用不了? 谢谢!

A: 请用我们我们网站上下载 2104 的工程模板, 它解决了这个问题。

即使这样, 也不能真正使用 printf。您还需要自己编写一些底层函数才能使用, 详细参考 ads 自带的 ADS_CompilerGuide_D.pdf。

第90问:

Q: swi 的功能号是如何来的? 它和 LR 寄存器的值是何关系?

如《ARM 微控制器基础与实战》程序清单 6.13

```
LDREQ R0, [LR, #-4] ;BICEQ R0, R0, 0xff000000
```

A: LDREQ R0, [LR, #-4] ;用来读 SWI 的代码

BICEQ R0, R0, 0xff000000 ;是 ARM 方式进入, 取低 24bit

第91问:

Q: 请问在 TargetInit() 中函数开始会执行 srand((INT32U)TargetInit), 它是做什么用的?

A: ex1 用它来产生随机数种子的。

Q: 请问: 我建立的工程中, 所有源文件与 2104 附带的光盘例子源文件相同, 编译链接也一切正常, 可就是不能到板子上跑!

提示错误是: 向量中断有错误, 无法自由运行!!

这是怎么回事? 相同的源文件, 加在你的工程里, 正常; 加到我的工程中就出错! ?? 我创建的是 ARM 可执行映像。

A: vector.s 你自己的吗? 如果是要计算向量去的累加和了。

第92问:

Q: 请问: vectors.s 中“DCD 0xb9205f80” 的 0xb9205f80 在实际运用中需要改动吗?

A: 不需要更改, 除非改动了向量表中的指令代码。

第93问:

Q: 请问, 为什么 queue 数据队列, 《ARM 微控制器基础与实战》上给画成了环形?

A: 因为头跟尾的指针指向一个地址, 队列逻辑上是环状的。

第94问:

Q: 有没有人成功的把一个数据定义到程序空间里? 各种方法我都试了, 是不是有什么编译开关?

A: 使用 const 修饰, 定义变量时带初始化值, 要定义为全局的变量。

第95问:

Q: ads 里没有 CODE 关键词, 怎样使定义的字符串数组不占用 RAM 空间?

A: `const unsigned char string[]`

第96问:

Q: 在例程中 TIME_test 中的 TIMEOUT 当写入时提示:

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!

我不知怎样改设置, 哪位提醒一下?

A: 中断向量表的校验和不为 0, 用 AXD 看 0 地址的数据 (32 位方式), 自己加一下。注意把高于 32 位的部分去掉。

第97问:

Q: 请教各位: 我做 TIMEOUT 实验 (2104 板), Make 通过, Debug 时出现了如下错误提示:

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!

设置 Link: ARM Link

r0 base: 0x00000000

rw base: 0x40003000

image entry point: 0x00000000 其它的没变!!

而我使用 ISP 完全可以下载而且可以运行! 这可以说是设置问题, 但是我实在找不到! 怎么办??

A: 请看一看配套《ARM 微控制器基础与实战》附录 1, ISP 软件可能对它进行了处理, 而 JTAG 没有特殊处理

第98问:

Q: `*(volatile unsigned int *)` 是什么意思? 例如 `*(volatile unsigned int *)addr` 具体是什么意思?

A: 分开来看, `(volatile unsigned int *)` 就是定义一个可变的无符号整形指针, 前面的那个 `*` 就是取起内容。

第99问:

Q: 如何理解 `#define VICIRQStatus ((*(volatile unsigned long *) 0xFFFFF000))`

A: 宏定义, 参考 C 语言的书籍。

`(volatile unsigned long *) 0xFFFFF000` 将 `0xFFFFF000` 强行转换为指针, 然后 `*` (指针) 即可对此地址进行访问。

第100问:

Q: 在异常处理向量表的设置中, 为什么不直接将异常向量的入口地址写入 PC 中呢, 为什么非要用什么 DCD 这些伪指令, 到底有什么用啊?

A: 为了保证任何时候其累加和为零, 不然改一次程序就要计算一次。

第101问:

Q: 单步调试 i2cINT.C 在 ISendStr 中启动总线后, 程序怎么又跑到 vectors.o 中, 进行初始化了呢??

A: 由于 VIC 的限制, 程序不能在 0x18 停下来, 否则执行非向量中断。如果没有设置, 则非向量中断为 0, 将从 0 地址执行。

建议: 有中断时不要单步执行程序, 可以手动暂时关中断或设置断点代替单步。

(编者注: 下载最新的驱动可以解决这个问题)

第102问:

Q: 请教: 如何修改 ADS 的启动代码, 从 __main 到 main, 按道理应该有一个文件, 修改这个小文件, 就可以修改 启动代码。

A: __main 是 ADS 运行库, 最好不要修改, 要是不使用库就不要用 main() 函数, 起个别的名。直接跳转过去就行了。

详见: 《ARM 体系结构与编程》P328。

第103问:

Q: 我在 JTAG 仿真时出现的对话框, 提示:

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!

A: 参见《ARM 微控制器基础与实战》附录一第一个问题的第二部分, 选择源代码的启动代码文件夹下的 vectors.s, 并参照《ARM 微控制器基础与实战》图 7.11 设置。

第104问:

Q: 初始化代码中分配堆栈的问题

MSR CPSR_c, #0xd3 ;进入特定的处理器模式

LDR SP, StackSvc ;给当前处理器模式的堆栈指针赋值, 这只是一个值而已,
;他在接下去的 DCD 语句赋值

SvcStackSpace SPACE SVC_STACK_LEGTH * 4

这个语句开辟一个 SVC_STACK_LEGTH * 4 大小的内存, SvcStackSpace 是这块内存的标号, 我想应该就是内存的首地址, 这样应该好理解了。

StackSvc DCD SvcStackSpace + (SVC_STACK_LEGTH - 1)* 4

这个语句就是把这个首地址加上堆栈大小值。

A: 计算出堆栈指针来, 我想这个堆栈是向下生长的, 所以 SP 赋值的是堆栈地址最高的那个。

第105问:

Q: 请问启动代码中这句是什么意思?


```
__user_initial_stackheap
LDR r0, =bottom_of_heap
MOV pc, lr
;/* 分配堆空间 */
AREA Myheap, DATA, NOINIT, ALIGN=2
bottom_of_heap SPACE 256 ;库函数的堆空间
```

我的问题是：

1. 这个函数在哪里被调用？
2. 赋值给 r0 以后就完事了？ r0 起什么作用？

A: 堆和栈的分配函数，由 ADS 提供的初始化代码调用，具体参考配套《ARM 微控制器基础与实战》的相关部分。

第106问：

Q: 2104 的启动代码是否适用 2124 呢？

网站上下载的那个 easyarm2104 工程模版里面写着 arm executable imag for lpc21**, 是否说也适用 2124 呢？如果不行，哪些地方需要修改？希望能够指点一下！

A: 您可以下载 2100 的工程模板。

第107问：

Q: 我在用 ARM 的 IAP 功能，要把已经烧到 flash 中的数据写到 RAM 中，我是这么写的：

```
uint32 *q, data;
q = 0x00006000;
data = *q;
```

这是要读的 flash 的起始地址，但是第二句编译出错，我想应该是 C 编译器不许给指针直接赋值，我用汇编解决了这个问题，但是怎么用 C 来解决呢？

A: q = (uint32 *)0x00006000;

第108问：

Q: 问一个有关汇编的基础问题。

```
例程： ANDS R1, R1, #0x0400
      BEQ WAITOK
```

请问 ANDS 是如何影响标志位的，而 BEQ 判断的是那两个操作数相等时执行？

A: R1&0x0400 => R1, 若结果为 0(即 R1 为 0)，则标志位 Z=1。

当 Z=1 时，BEQ WAITOK 有效执行。

第109问：

Q: 请教：在 ADS 中怎么给某个变量确定固定物理地址？

就是类似 KEIL 中的 XDATA xxx _AT_ 0x4456 的功能。

A: 使用分散加载机制，mem_c.scf 等就是例子。

第110问:

Q: 哪里有介绍 ARM 的 C 语言编程的?

A: 其实大多数嵌入式系统的 C 语言都差不多, 可找一本写嵌入式 C 语言的书即可。

第111问:

Q: 请问在系统复位后首先执行的是否为 Boot Block 的中断向量?

然后由 boot block 里面的程序决定是执行用户程序还是 ISP 程序。

那么这时启动的 boot block 里的中断向量表和用户程序里的中断向量表是如何在 flash 里安排的?

在《ARM 微控制器基础与实战》上看到的是复位后 boot 扇区的最低 64 字节出现在 0x00000000 区域, 那用户的中断向量表应该在哪个区域?

产品中一般为 0x00000000, 在开发板中重启后实际为 boot block 的中断向量地址, 对吗?

A: 看一看 3.3.6 节。产品中一般为物理 0 地址处。

第112问:

Q: 《ARM 微控制器基础与实战》上有一段程序:

```
uint32 i;  
i = VICIRQStatus;  
i = IOSET;
```

请问: 为何先要读取 VICIRQStatus 的值, 才能读 IOSET 的值?

A: 读出 VICIRQStatus 只是为了方便观察当前 VICIRQStatus 的值, 没其它用途。

第113问:

Q: 请问处理器在什么情况下处于用户模式? 多谢!

A: 需要你去设置 CPSR 寄存器。

用户程序前台程序一般在用户模式/系统模式下运行。

第114问:

Q: 在 EINT1_LED.S 中的倒数第二行有一个单独的 B 指令, 它是什么含义? 多谢!

A: 是“B .”, 跳转到当前地址, 即死循环, 与以下代码等效:

```
HALT      B      HALT
```

第115问:

Q: 模板里到底有些什么?

A: 起动代码, 相关编译链接设置。

起动代码是用来初始化系统的程序, 如 Startup.s、target.c、stack.s 等等。

第116问:

Q: 今天试用了工程模板。使用了 ARM Executable Image for lpc21xx 建了个项目,看了下启动文件与原来的不同了,用了个《ARM 微控制器基础与实战》上的例程来作试验,用的是 time0 定时中断来亮灯的程序,用的例程的 TargetInit()和 int main(void);修改了用户堆栈和 bottom_of_heap 的长度,程序能运行,但不能产生中断,time0 和 VIC 初始化的是正确的,是什么原因?

用工程模板建立的项目文件夹的 src 中多了几个文件,其中的 mem_a.scf,mem_b.scf,mem_c.scf,怎样才能导入到项目中来?谢谢!

A: 原来 startup.s 默认是关了中断的!

第117问:

Q: 在 easyarm 2104 的配套光盘中的 IAP 例子中,定义了一个函数指针

```
void (*IAP_Entry)(uint32 param_tab[], uint32 result_tab[]);
```

我看不懂,这个函数和普通的函数定义方法不一样,尤其是(*IAP_Entry),大家教教我。

A: 这是一个指向函数的指针!!

第118问:

Q: 变量定义中 volatile 是什么意思?请大侠告诉我一下《ARM 微控制器基础与实战》上 c 语言定义变量时出现了 volatile,不知道是做什么用的。C 语言里好像没有这个语法阿!

A: 告诉编译器不要优化掉,volatile 是易变的意思。

第119问:

Q: 请问 ARM 中的一个编译方面碰到的问题。

在 ARM 地汇编语言中,有条件编译伪指令 IF..ELSE...ENDIF;

而在 C 语言中,有相应的条件编译伪指令 #IF...#ELSE...#ENDIF,在较大的程序设计中,往往需要对整个程序进行条件编译。

我要问的问题是:我在 CONFIG.H 中设置一个编译开关,对所有的 C 文件进行条件编译是有效的,因为每个 C 文件中都有一个语句: #include "config.h",我如何将该编译开关的信息传递给汇编语言文件,如 VECYORS.S?

A: 好像没有直接的方法,可以编写一个 CONFIG.INC 来管理汇编程序的配置。

第3章 操作系统

第120问:

Q: 在 SWI_Exception 的 0x40 中的 OSFunctionAddr[regs[0]] 中的 Regs[0] 指的是堆栈中的 R0 还是其它?

A: 是堆栈中的 R0。

第121问:

Q: 《ARM 微控制器基础与实战》中在 ucOS 移植中说 C 语言无法保证堆栈的结构, 请问这是什么含义?

A: 就是不能保证有哪些寄存器入栈及寄存器入栈的顺序。

第122问:

Q: 请问斑竹 OSNeedToSwapContext 在哪个文件里声明的, 我找不到?

A: 这是要删除的, 我的 OS_CPU_C.C 中函数 OSIntCtxSw:

```
/*
*****
* 函数: OSIntCtxSw.
* 描述: 中断级任务切换, 此处并不真正进行任务切换, 具体切换在 IRQ 服务程序中.
*****
*/
void OSIntCtxSw (void)
{

}
```

第123问:

Q: 我的情况如下: 我自己制作了一个硬件模块, 用的是 lpc2214, 现在需要将 ucOS-ii 移植到上面去。我用板子做 ZLG 公司提供的实验, 是可以做的, 如 led 灯等, 按道理说串口等硬件应该没有问题, 然后我就做公司提供的那三个移植实例: ex2_arm. 编译等是通过的, 生成了 hex 文件, 下载到 flash 中, 运行 easyarm, 但是没有任何显示, 不知道是什么问题, 应该是硬件上呢, 还是其他的?

A: 这主要是 RAM 的问题, 请用我们的工程模板, 并选择在 FLASH 中调试, 如果编译通过, 则一般行, 否则请减少任务堆栈的大小。

第124问:

Q: 下载了工程模板以后, 将工程模板目录下的文件都拷贝到了 stationary 目录下, 然后建立了一个 ARM Executable Image for UCOSII(for lpc21xx)的工程, 在 PROJECT 目录下的文件结构如下:

```
test.c
includes.h

src->config.h, heap.s, includes.h, irq.s, lpc2294.h,
      lpc2106.h, mem_a.scf, os_cfg.s, stack.s, startup.s
      target.c, target.h, ucos_ii.c, ucos_ii.h,
arm->os_cpu.h, os_cup_a.s, os_cup_c.c
uCOS->ucos 源文件
arm_pc->pc.h, pc.c
```

编译可以通过，但是 MAKE 时提示：

```
ERROR: Execution region IRAM overlaps with Execution region STACKS.
```

如果我表述得不是很清楚的话，那么可否告诉我工程模板到底怎么操作，怎样才能把例子在 ARM2104 上跑起来。

A: RAM 占用太大，请在 flash 中调试，并减少任务堆栈的大小。

第125问：

Q: 我改了以下 SCF 文件，把堆栈的值设置大了一些，但是还是出现以前的老问题，那就是程序跑到

```
TargetResetInit() 函数处后就跳到取数据终止的异常中断去了。
```

A: 不能把堆栈设置到内部 RAM 之外。

第126问：

Q: 新旧任务级的切换是不是必须在管理模式下切换？切换后新任务的运行是不是必须在管理模式下？

A: 在配套《ARM 微控制器基础与实战》里的任务切换是通过 SWI 实现的。SWI 会使 CPU 进入管理模式。同时 CPU 自动将任务的 CPSR 保存到管理模式的 SPSR 中。

SWI 程序还保存了一些寄存器到管理模式的堆栈中。然后任务切换程序会在管理模式和系统模式间来回切换，目的是把保存在管理模式堆栈的寄存器内容拷贝到任务的堆栈中（注意：任务是运行在用户模式或系统模式的，这两个模式使用同一个 SP，这时的系统模式的 SP 指向的就是任务的堆栈），也会将管理模式的 SPSR 拷贝到任务堆栈（注意管理模式的 SPSR 就是在执行 SWI 时 CPU 自动保存的任务运行时的 CPSR，它记录了任务运行的 CPU 模式和其他的运行状态信息）。最后把任务模式 SP 保存到任务的任务控制块中（注意此时 CPU 已经是运行在系统模式下的了）。以上大致就是 uC/OS-II 中所说的保存任务运行环境的工作。

任务的恢复运行就比较好理解了。

接着前面的说，CPU 还在系统模式下。程序从任务控制块中得到任务的堆栈指针。

（OSIntCtxSw_1 的第一句）将这个指针的值加上一定的偏移量保存到 SP 中。为什么要加上一定的偏移量呢？因为这时任务的堆栈中除了保存任务运行时的数据外还保存了任务的上下文，即任务切换时保存的 CPU 寄存器等内容。而我们恢复任务等一下是在管理模式下进行的，用的是管理模式的堆栈指针。等恢复任务后 CPU 就开始执行任务

代码了。我们已经没有机会再修改系统模式下的 SP 值了。所以在这个时候就应该对 SP 进行调整。调整的大小就应该是所保存的上下文占用的大小。从堆栈中恢复 LR 的值。同样的道理，管理模式和系统模式使用不同的 LR，在切换到管理模式之前，必须先要在系统模式下恢复任务运行时的 LR。（OSIntCtxSw_1 的第三句）接下来就切换到管理模式了。

（OSIntCtxSw_1 的第四句）使管理模式 SP 指向任务堆栈（第五句）注意这是没调整过的任务 SP，也就是保存了上下文环境的 SP。

接下来的事情就真的很简单了。从堆栈中恢复出保存的东西。先是 OSEnterSum，然后是任务的 CPSR（注意任务的 CPSR 就是进入 SWI 时由 CPU 自动保存到管理模式的 SPSR 中的，所以此时也是恢复到管理模式的 SPSR 中）最后一句就是典型的中断返回语句了。恢复寄存器，恢复 PC，加了一个 ^ 号意味着同时将 SPSR 拷贝到 CPSR 中。

现在是回答你的问题的时候了。

切换前任务是运行在什么环境下的（包括 CPU 模式，所有的寄存器），任务切换后任务就是运行在 什么环境下的。对任务来说，什么也没改变。只有 PC 值指向了下一条指令，哈哈。

第127问：

Q：请教：在 rtos51 解释的概念里的“信号量”比较难懂，“消息队列”可以理解为任务之间互相传递的参数，但“信号量”怎样理解呢？具体一点，谢谢！

A：信号量是 60 年代中期 Edgser dijkstra 发明的，它实际上是一种程序间的约定机制，这种约定决定那个程序（任务）可以执行。在多任务内核中普遍使用信号量用于：

1. 控制共享资源的使用权（满足互斥条件）；
2. 标志某事件的发生；
3. 使两个任务的行为同步。

信号量像是通行证，且通行证的数目是有限的。任务要运行下去，要先拿到通行证。如果信号量 已被别的任务占用，该任务只得被挂起，直到信号量被当前使用者释放掉。信号量的值可以是 0 到 255 或 0 到 65535，或 0 到 4294967295，取决于信号量规约机制使用的是 8 位、16 位还是 32 位。到底是几位，实际上是取决于用的那种内核。根据信号量的值，内核跟踪那些等待信号量的任务。

一般地说，对信号量只能实施三种操作：初始化，也可称作建立；等信号也可称作挂起；给信号或发信号。信号量初始化时要给信号量赋初值，等待信号量的任务表应清为空。想要得到信号量的任务执行等待操作。如果该信号量有效（即信号量值大于 0），则信号量值减 1，任务得以继续运行。如果信号量的值为 0，等待信号量的任务就被列入等待信号量任务表。多数内核允许用户定义等待超时，如果等待时间超过了某一设定值时，该信号量还是无效，则等待信号量的任务进入就绪态准备运行，并返回出错代码（指出发生了等待超时错误）。任务以发信号操作释放信号量。如果没有任务在等待信号量，信号量的值仅仅是简单地加 1。如果有任务在等待该信号量，那么就会有一个任务进入就绪态，信号量的值也就不加 1。于是通行证给了等待信号量的诸任务中的一个任务。至于给了那个任务，要看内核是如何调度的。收到信号量的任务可能是以下两者之一：

1. 等待信号量任务中优先级最高的任务；
2. 最早开始等待信号量的那个任务，即按先进先出的原则（FIFO）。

有的内核有选择项，允许用户在信号量初始化时选定上述两种方法中的一种。但 Small RTOS51 只支持优先级法。如果进入就绪态的任务比当前运行的任务优先级高（假设，是当前任务释放的信号量激活了比自己优先级高的任务）。则内核做任务切换（假设，使用的是占先式内核），高优先级的任务开始运行。当前任务被挂起。直到又变成就绪态中优先级最高任务。

第128问:

Q: 我用了两个串口，但当我把程序做大的时候，发现在这个任务里所建立的邮箱失败，我只用一个串口时是没有这种事情的。

```
Uart0ReviceMbox = OSMboxCreate((void *)0 );    /* 建立邮箱 */
if (Uart0ReviceMbox == NULL)
{
    while (1);
}
```

我想问一下有几种可能会造成这种分配失败？我的程序存储器用量如下：

```
Total R0 size      22168
Total RW size       7956
Total ROM size      22168
```

A: 在 OS_CFG.H 中定义最大事件数，太小的话，分配会失败的。

第129问:

Q: 在周立功的原例子中，系统代码事先烧到 ROM 中，应用代码在 RAM 中运行。我想知道，（RAM）应用代码调用系统函数，那么 2104 是如何把 ROM 中的系统函数的地址传递给 RAM，也就是应用代码和系统代码是如何关联在一起的？我猜是通过 swi（软中断），应用代码通过 swi 调用系统代码，但是 swi 中断程序中，是如何获得烧到 RAM 中的系统函数的地址哪？百思不得其解。

A: 您看一看 romcode 工程中 swi 的 0x40 和 0x41 功能，及 Os_call.c、Os_call_arm.s、Usr_call_arm.s。

第130问:

Q: 16k 的 RAM，128 的 FLASH ROM 跑 uc/os-II 够吗？再跑点应用程序够吗？

A: 取决于您的应用程序对 RAM 的需求，及代码量。一般程序可以。

第131问:

Q: 利用 lpc2104 开发应用程序时，在 Os_call_arm.s、Os_call.c 两个文件中，那几个函数需要在汇编中实现，那几个需要在.c 文件中实现，为什么？谢谢各位大虾！

A: 4 个以上参数用 c 实现。因为通过寄存器只能传递 4 个参数。

第132问:

Q: 如果禁止 SmallRTOS51 进行中断嵌套管理 (#define EN_OS_INT_ENTER 0), 是否还需要设置中断优先级寄存器 IP, 使得所有中断的优先级都相同 (全高或全低)?

A: 受 SmallRTOS51 管理的设置为最低优先级, 不受 SmallRTOS51 管理的设置其它优先级较好。

第133问:

Q: 在 OS_CPU_S.S 文件中有如下一段代码, 其中有两句代码的作用不是很明白, 请解说一下。

```
OSIntCtxSw_1
    LDR    R4, [R6]          ;获取新任务堆栈指针
    ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    ;这两行代码有何作用呢?
    ADD    SP, R4, #68      ;17 寄存器 CPSR, OsEnterSum, R0-R12, LR, SP
    LDR    LR, [SP, #-8]
    ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
    MSR    CPSR_c, #(NoInt | SVC32Mode) ;进入管理模式
    MOV    SP, R4           ;设置堆栈指针

    LDMFD SP!, {R4, R5}    ;CPSR, OsEnterSum
                                ;恢复新任务的 OsEnterSum

    LDR    R3, =OsEnterSum
    STR    R4, [R3]

    MSR    SPSR_cxsf, R5    ;恢复 CPSR
    LDMFD SP!, {R0-R12, LR, PC }^ ;运行新任务
```

A: 这两条指令是在系统模式下恢复用户模式下的 LR; 可以参考配套《ARM 微控制器基础与实战》6.4.9 节。

第134问:

Q: 我将《ARM 微控制器基础与实战》上 uC/OS-II 的第一章例 2 移植例在 ARM2104 开发版上实现例 2 中的 TEST.C 中的 “#define TASK_STK_SIZE 512” 改为 “...256”, 程序编译后, 数据量低于 16K, 能正常运行, 但运行时 “Total Stack” 不是 256 全是 1024。另 UC/OS-II 书上, 第一章例 2 每一个任务的堆栈都是 512, 为什么书上 11 页 “Total Stack” 是 “624, 1024, 1024, 1024, 1024, 1024, 1024”, 请老师指点, 万分感谢!

A: 显示的是字节而定义的是字。

第135问:

Q: 可不可以把所有的任务和相关信号量 (二值) 在一个任务里面一起创建呢? 我现在是这样做的, 不知是不是这方面的原因? 而且全部信号量是这样创建的:

```
XX1 = OSSemCreate(0);
```



```
XX2 = OSSemCreate(0);  
XX3 = OSSemCreate(0);
```

一共创建了十个信号量，以前程序代码版本是分开写的执行起来没有问题，不知现在为什么这样写就会出现小问题呢？这样的问题和我的任务和信号里书写位置有关系吗？

A: 必须在使用信号量前创建信号量。

第136问:

Q: UCOS 在中断处理完后调用 OSIntExit(), 该函数将判断是不是要进行任务切换, 如果是则调用 OSIntCtxSw() 切换任务, 然后才恢复寄存器, 中断返回。这样说来, 在中断返回前, 已经切换到别的任务去了, 在再次回到被中断的任务前, 是不是一直没有中断返回?

A: 如果进行任务调度, 则不会执行 OSIntCtxSw() 后的语句, 相当于中断已退出。

第137问:

Q: 最近我在学习使用 SMALL RTOS 时发现一个 BUG。

任务在调用系统等待函数 OSWait(K_TMO | K_SIG, x) 后, 不能在延时 x 个 ticks 后被唤醒。

我分析了一下原因如下:

```
在 OS_CORE.C uint8 OSWait(uint8 typ, uint8 ticks) small 中,  
case (K_TMO | K_SIG): 中执行了函数 OSTaskSuspend(OSTaskID);  
( OS_CORE.C 第 549 行) 而此函数将延时值改为了 0!
```

解决办法: 将此函数改用另一个函数 OS_TaskSuspend(TaskID);
并在其后加入任务切换函数 OSSched();
便能工作正常。

A: 对, 这是疏忽, 谢谢。您的更改是对的。

第138问:

Q: 在文件 Os_cpu_a.s 中的函数 OSIntCtxSw 中有如下语句:

```
.....  
(1) LDR    R4, [R6]  
(2) ADD    SP, R4, #68  
    ; 17 寄存器 CPSR, OsEnterSum, R0-R12, LR, SP  
(3) LDR    LR, [SP, # --8]  
(4) MSR    CPSR_c, #(NoInt | SVC32Mode)    ;进入管理模式  
(5) MOV    SP, R4  
.....  
(6) LDMFD  SP!, {R0-R12, LR, PC }^    ;运行新任务
```

《ARM 微控制器基础与实战》404 页中解释: 只所以要 (1) (2) 是因为 “OSTCBHighRdy
—>OSTCBStkPtr 保存的是任务栈位置, 而寄存器恢复后堆栈指针并不指向这, 所以要调

整新任务堆栈指针。”可是将堆栈指针调整到“新任务入栈的其它数据（见《ARM 微控制器基础与实战》图 6.3）”处后，除了取出了 LR 以外并没有做什么，而当执行了（4）后又重新将堆栈指针指向了栈顶（即第（5）句）。我认为可以将（2）省去，直接将（3）改为“LDR LR, [SP, # 60]”，而当执行完（6）后，堆栈指针就自动指向了“新任务入栈的其它数据”处。这样修改后我运行了 EX1_arm 例子，正常。

请问这利用（3）进行堆栈指针调整的原因是什么？不调整可以吗？谢谢。

A: 注意不同模式有不同的 SP 指针。您这样做会造成内存泄漏，长时间运行会耗尽堆栈而使程序崩溃。

第139问:

Q: 不明白在光盘上 ROMCODE/SRC/下的 os_call.c 做什么用。

A: 是应用程序与事先固化到 flash 中的 ucossii 接口的一些代码。在本例中为应用程序如何调用事先固化到 flash 中的 ucossii 的 OSFlagPend 等函数的接口代码。请看配套《ARM 微控制器基础与实战》的 7.4.3 节的第 10 点。

第140问:

Q: 请问在 OS_CORE.C 中的常数数组 OSUnMapTbl[] 是做什么用的？

A: 用来计算优先级的，查表计算比较快。

第141问:

Q: ucoss 的中断嵌套层数是否受到初始化时分配的 IRQ 堆栈大小的限制？我的理解是每一次中断需入栈的寄存器有 R0-R3, R12, LR, SPSR, 共 7 个，如果想达到 8 层嵌套的话，堆栈长度 IRQ_STACK_LENGTH 应设为 56，不知这样理解对不对。

A: 是，但中断至少占用 8 个字，因为 c 语言的中断处理函数会将一些数据压入堆栈。

要达到 8 级嵌套需要的堆栈长度与具体的代码有关。

（编者注：新移植代码已有改变）

第142问:

Q: 请问：核心定时器中断不进入可能因为……？

我的程序在运行一段时间以后，核心定时器中断即操作系统用的 Timer0 不能进入，查发现 CPSR 的 I 位为 1，请教这可能是哪个原因？

A: 估计为开关中断次数不匹配造成。

第143问:

Q: 请问：GetOSPprioCur() 函数应怎样调用？它是一个内核函数吗？

为什么我在内核和任务分别编译时正常，而合到一起编译时它告警为未定义呢？

同时我已给您发了一个邮件，请教如何将分别编译的程序合在一起编译的方法，请指教！

A: 这是我自己编写的函数，其实就是返回 OSPrioCur 的值，请参考 MyFunction.c。

第144问:

Q: 关于不受 uc/os 控制的中断: 在 ARM 板中, 非屏蔽中断可以挂起正在执行的 uc/os 任务, 除了不能使用 uc/os 中的函数外, 它的执行工序是不是和受管理的中断一样, 先挂起当前任务, 再保存 CPU 寄存, 然后再执行中断 ISR, 完成后, 内核脱离, 寄存器恢复, 最后任务调度。

另外, 《ARM 微控制器基础与实战》中提到不受管理的中断, 它的工作是否和非屏蔽中断一样, 也可以在 uc/os 正执行时发生中断。如果不同, 它们是怎样工作?

最好是象受管理中断一样说明一下它的工作时序!

A: 必须比受管理的中断的优先级高, 编写方法与没有 OS 时中断的编写方法一样。

第145问:

Q: 不受 uc/os-ii 管理的中断和受管理的中断是否具有相同的响应方式, 即不受管理的中断在中断 uc/os 正执行的任务后, 是不是也是先挂起当前的任务, 保存 CPU 寄存器, 再执行中断子程序, 恢复 CPU 寄存器, 进行最优先级任务的调度。

A: 不受 uc/os-ii 管理, uc/os-ii 都不知道, 还有什么任务调度?

Q: 我知道不受管理的中断不能调用 uc/os 的函数, 但《ARM 微控制器基础与实战》上说: FIQ 不受 uc/os 管理, 但可以用来执行紧急任务, 就是说在 uc/os 运行时, 不受 uc/os 管理的中断还是可以发生的, 它是把整个 uc/os 操作系统中止, 还是只是中止 uc/os 正在执行的任务。不受管理的中断完成后, 接着执行什么?

A: 可以发生, 中断整个 RTOS。

Q: 我看了您在 lpc210x 上的移植代码, 你在说明中说: “如果您想通过软件仿真, 请将 target.c 中的第 51 行屏蔽, 这样就可以看到任务逐个切换, 最后将进入空闲任务。”我照这做了, 但是在单步或者设断点执行时会产生异常, 原因是未定义指令 OS_ENTER_CRITICAL() 引起的, 经过编译的函数都会变成蓝色, 但这个函数还是黑色, 因为它实际上是一个软中断, 请教您如何调试才能看到任务逐个切换? 谢谢!

A: 1. 屏蔽的那一行是死等锁相环锁定, 软件仿真时是没有锁相环的。

2. 黑色是正常的, 因为没有定义成函数。

3. 因为你的 AXD -> OPTION -> 配置处理器中的 VECTOR CATCH 中的 S 选中了, 所以 AXD 将你的软件中断当成了异常给捕捉了, 取消即可。

第146问:

Q: 请教 ucos2 的源代码中经常碰到: return((void*)0) 是什么意思?

A: 返回空指针。

第147问:

Q: 这是个什么错误, 怎么改啊?

OsMemPut 是个函数名称

Error : L6200E: Symbol OSMemPut multiply defined (by uCOS_II.o and Os_mem.o).

A: 这是重复定义错误。请不要把 uCOS_II.C 添加到你的工程中。

第148问:

Q: 我想请教一下在配套光盘中有没有 uc0sII.h 文件, 怎么我总是找不到的?
如没有能否提供下载地方? 谢谢!

A: 北航出版的《嵌入式实时操作系统 uC/OS-II》第 2 版就带有 uC/OS 2.52 源代码,
还可以到其它网站上找找。

第149问:

Q: 我想问一下一个任务中的子函数的局部变量算不算进堆栈的容量。

我发现我在用邮箱传递一个较大的值时, 是在一个任务的函数中声明了 `uint8 byte[500]`, 可在传递时值发送了变化, 只好用的外部变量了, 最后搞得邮箱只起了信号量的作用, 所以我觉得局部变量虽然是从堆栈中取得数据, 最后在任务切换时, 把局部变量也保存到堆栈中, 不知道我这种理解是否正确?

A: 算。局部变量在函数退出后(不管任务是否切换过)被释放。

第150问:

Q: 我用 UCOS 在 EASYARM 上的一个程序内编写了 5 个任务, 前 4 个任务调试后工作正常, 在编译第 5 个任务时, 提示为:

Execution region ER_RO overlaps with Execution region ER_ZI.

编译还提示:

TOTAL ROM SIZE (CODE +RO DATA +RW DATA) 8624

我曾经试图修改“`#define TASK_STK_SIZE 64`”也不管用, 请问, 如何解决?

A: 问题已找到, 代码超过 16K。上面的信息会在 zi 段或者 rw 段与 ro 段(代码段)发生重叠时发生。

第151问:

Q: 请教 uc/os 移植问题?

在 2104 上面能进行移植吗? 如果不行的话, 需要买 2106?

A: 你要在 `os_cfg.h` 中把不用的内部调用都裁减掉, 就可以做一个很小的内核, 就是不裁减你就是 14k 左右, 写在 FLASH 里一样也能运行。

第152问:

Q: 请教信号量的概念问题。

我准备用信号量来编写一个 ARM 程序, 但我没有完全理解信号量的概念。

例如: 现在有 4 个任务: TASK1, TASK2, TASK3 和 TASK4, 任务的要求是: TASK1 和 TASK2 之间需有信号量传递信息; TASK3 和 TASK4 之间需也有信号量传递信息。

假定 TASK1 和 TASK2 之中已建立了一个信号量为

```
RandomSem = OSemCreate(1);
```

我要问的问题是: TASK3 和 TASK4 之中的信号量是否需重新建立一个? 例如, 加一条

```
RandomSem1 = OSSemCreate(1);?
```

换一个问法：即若 TASK1 和 TASK2 之间需有信号量传递信息；

TASK3 和 TASK4 之间需也有信号量传递信息，我只需在运行这 4 个任务之前的初始化程序中有一条指令即可：RandomSem = OSSemCreate(1);?

A: 用两个信号量，你第一个说法对了。

第153问:

Q: 各位：在 UCOS 多任务中有一个任务结构如下：

```
void TASK(*pdata)
{
    uint8 i;
    任务循环前的一些指令集 A;
    while(1)
    {
        任务循环;
    }
}
```

我要问的是：与该任务有关的初始化，如串口初始化，是否不用放在整个程序的开始处，而作为指令集 A 的一部分，也可达到同样效果？

A: 启动和初始化代码放在一个文件里主要是为了程序的模块化，不过代码少的话倒是可以放在一个函数里完成。

A: uc0s 的事件标志使用注意点：

请在事件标志结构定义前加__packed，否则由于字节对齐的问题会产生没有反应的 bug！

第154问:

Q: 移植中碰到的问题！

我把周工《ARM 微控制器基础与实战》上的范例 1 移植到 ARM 上，但任务没有跑起来，请做过移植的朋友帮帮我吧！

我是把代码生成 HEX 文件全部下到 FLASH 中，不知道这样对不对啊？软件需要怎么设置呢？

A: 先检查 ARM Linker 的设置是否正确？参照《ARM 微控制器基础与实战》P420 设置 release 的参数。

A: 感觉运行 ex1 还是比较容易的，毕竟斑竹都已经把代码写好调试好了，你可以不必完全把它拷贝下来，可以自己建一个 project，然后一个个文件添加，这样比较容易弄懂哪些文件都是来干嘛的，哪些还需要自己改写。推荐在建立 project 时可以把相应文件分组，我就是把 uC/OS2.52 文件放在一个组下，需要自己改写的(如 OS_CPU_A.S)放在一个组内，启动代码放在一个组内，自己的代码(如 main，自己的任务)放在一个组内，这样比较分明些，个人建议。

第155问:

Q: 看过操作系统固化之后，有点疑问：

把操作系统和用户代码分开真的能节约 RAM 空间吗？只是在用户代码定义的起始地址在 0x40000000 的情况下(即调试时)才会节约吧？一般情况下我觉得如果要真正使用的话都应该是把程序起始地址定义到 00000000 上的,这样的话才有,不然一掉电程序就没了。这样理解对吗？

A: 是的, 参考合并在一起的例程 <http://www.zlgmcu.com/download/downs.asp?ID=861>。

第156问:

Q: 任务间的数据传输除邮箱等外是否可以建立一个全局变量在任务间传递数据！

A: 可以, 但要注意重入问题和代码优化问题, 最好使用 volatile 修饰变量, 如果不能一次读写完毕则需要加上开关中断的代码。

第157问:

Q: 请问: uc/os 任务堆栈问题。

在 ucos 移植的程序 EX1_arm 中:

```
#define TASK_STK_SIZE 128 /* Size of each task's stacks (# of WORDs) */
#define N_TASKS 10 /* Number of identical tasks */
```

请问, 任务堆栈设为什么设这么大？

A: 可以变小, 但要在复杂任务中应保证够用。

第158问:

Q: 在看 uc/os-II 的书时, 关于 ucos-II 中任务切换的几种情况的问题。

1. 在一个时钟周期内, 至少所有的任务都要运行一遍, 对吗？

2. 任务切换发生的三种情况:

- 任务 A 主动放弃 CPU 的主动权, 利用 OSTimeDly() 延时, 进行任务切换;
- 中断时钟周期发生时, 在任务就绪表中寻找最高优先权的任务, 如果当前任务不是最高优先级, 发生任务切换, 否则仍执行原任务;
- 发生 irq、fiq、软中断、取指错误和取数据错误五种异常模式时, 直接发生任务切换, 中断处理结束后, 在任务就绪表中寻找最高优先权的任务, 如果当前任务不是最高优先级, 发生任务切换, 否则仍执行原任务。

A: 1. 不一定, 如等信号时 OSPend(x, 0, &err)时。

2. 只有在 IRQ、FIQ 和软中断受 OS 管理时才能进行任务切换, 同理取指, 取数异常在启动阶段的死循环根本谈不上任务切换。

第159问:

Q: 请问: 在移植 UC/OS 中的软件中断汇编接口程序中取功能号码。原程序为:

如果是在 THUMB 状态, 则为 LDR R0, [LR, #-2]

如果是在 ARM 状态, 则为 LDR R0, [LR, #-4]、

在清除 R0 中的最高两位, R0 的值就为功能号。

为什么是从 LR 中取值呢? LR 的值是进入中断时候保存的 PC 值吗? 迷惑!

A: 执行 SWI 指令后, 处理器进入管理模式, LR_svc 中保存返回地址, 顺藤摸瓜, 根据 LR_svc

就可以取得 SWI 语句，也就取得了中断号。

第160问：

Q: 请问及各位高手：OSIntCtxSw() 函数分别在 includes.h 和 OS_CPU_A.s 中有定义，区别在哪？

程序 OSIntExit 中调用该函数的时候是不是都是去调用了 includes.h 中的宏定义的那个？

两者各自的应用范围在哪儿？我看过一个移植实例上只有一个 OSIntCtxSw 函数。请指教！

A: includes.h 中的宏 OSIntCtxSw 在 C 中被调用。

第161问：

Q: 问一个堆栈指针的问题

在 OSIntCtxSw_1, 获取新任务堆栈指针。

```
LDR    R4, [R6]
ADD    SP, R4, #68                ;17 寄存器
CPSR, OsEnterSum, R0-R12, LR, SP
LDR    LR, [SP, #-8]
MSR    CPSR_c, #(NoInt | SVC32Mode) ;进入管理模式
MOV    SP, R4                    ;设置堆栈指针

LDMFD  SP!, {R4, R5}             ;CPSR, OsEnterSum
                                           ;恢复新任务的 OsEnterSum
.....
```

我想问一下：LDR LR, [SP, #-8]，这一行程序中，为什么堆栈指针要减去 8 个字节的值？

A: 这是调整 SP 的指针，使其指向栈中的 LR。看一看配套《ARM 微控制器基础与实战》的图 6.3。

Q: ucOS 中断丢失。

在 ucOS 在每秒切换 200 次时正常，但在 1000 次时中断丢失？可能是哪里的问题？

A: 问题已经找到，是随 2104 中的 ucOS 不支持中断嵌套。在不嵌套时 4000 次/秒也是正常的。

第162问：

Q: uC/OS-II 能在 2104 上和用户程序一起编译吗？一定得分开编译吗？

《ARM 微控制器基础与实战》上介绍的移植方法上是分开编译的，我想合并在一起调试，这样就不必用软件中断去寻找系统函数的入口地址。要做到这样，是不是只需把 OS_TASK_STAT_EN 设置为 1，OS_SELF_EN 为 0？

A: 可以, 网站上有例子 <http://www.zlmcu.com/tools/kaifaban/EasyARM2104.asp>.

第163问:

Q: 请教:我运行 OSInit(); 函数,程序死在 SoftwareInterrupt B SoftwareInterrupt 困惑呀!

A: uc/osii 的启动代码与普通的启动代码不同,您使用的是普通的启动代码。

第164问:

Q: 请问: 在 UC/OS-II 目录中的 README.TXT 下, 有的例子说: “仅 lpc2106 有足够的 RAM 可以在 RAM 中放入所用代码。”那么, 如果我使用 LPC2104, 在 FLASH 中存放运行代码是否能正常运行? 谢谢!

A: 主要看数据是否超出 16k, 如果超出, 请减少任务占用的堆栈。

第165问:

Q: 请问: 在 UCOS-II\ex2_arm 中, 我编译时出现错误提示如下:

```
ERROR: L6221E: execution region ER_R0 overlaps with Execution region ER_ZI.  
如何解决? 而我在 UCOS-II\ex1_arm 中编译。运行都正常, 谢谢!
```

A: 定义任务堆栈小一点, 如 “#define TASK_STK_SIZE 128”

第166问:

Q: uCOS-II\EX2_arm 我什么也不动, 用 ADS 打开, 可是编译的时候出错, 是很多文件找不到。

比如 ucos_ii.h 就找不到。

```
#include "..\..\source\ucos_ii.h"
```

可是我的光盘里没有 source 这个文件夹啊, 还有很多 uCOS-II 文件也找不到, 是不是我的光盘少了东东? 站上有得下载吗?

A: 要将 uC/OS-II(v2.52)源代程序放到 source 目录下, 北航出版的 uC/OS-II 书(第二版)带有。

第167问:

Q: OSIntNesting (嵌套层数) 和 OsEnterSum (关中断计数器), 有什么区别吗?

我个人认为这两个变量相同, 不知为什么要定义两个变量?

A: OsEnterSum 是关中断的计数器, 目的是实现临界段嵌套。

OSIntNesting 是中断嵌套层数, 目的是实现中断嵌套。

第168问:

Q: 为何我 EX1_FLASH 移植的时候总出现如下错误提示, 请问是哪里出错了?

```
Error : C2933E: type disagreement for 'OSTaskStkInit'
```


Os_cpu_c.c line 70

A: 是调用 OSTaskStkInit 时类型不一致导致, 好好看一下 C 语言。

第169问:

Q: 请问: 在 SWI 软中断指令中, LR 中放的是异常模式下的返回地址, 而这个地址的低 8 位和低 24 位分别是 thumb 和 ARM 指令下的立即数, 这种对应的关系是如何来的?

还有执行 THUMB 和 ARM 指令, LR 中放的地址最后的位应是 0 和 00, 那么为什么会有

```
swi 01
```

```
swi 03
```

这样的立即数呢?

A: 您的理解不对, 可以看一看 os_cpu_s.s 这个文件。

第170问:

Q: 2104、2119 中都不能用 UCOSII 的 flag 的问题。

在模板中 ARM Executable Image for UCOSII (for lpc21xx) 加入这个程序文件。实际上程序运行到 KeyFlagGrp = OSFlagCreate(KeyFlags, KeyFlagErr); 时就进入 Startup.s 的取数据终止, 但在没模板的情况下是没问题的, 请问哪位在模板中用过 UCOSII 的 flag?

A: 字节对齐问题, 请在事件标志结构定义前加 __packed。

第171问:

Q: 我请教您一个问题, 在 2104 里写程序的时候用 malloc() 来分配内存单元的时候, 经常出现在程序中被分配的单元数值在任务切换的过程中被改变的情况, 不知道您碰到过没有?

是如何解决的? 谢谢! 用 ucos 中的内存分配函数就不回出现这种问题。

A: 请在网上下载最新的模板, 其中启动代码有更新。

另外, 如果分配太大的空间可能会造成程序跑飞。

第172问:

Q: 在 uC/OS 中进入临界代码区时, 只是关 swi 中断, 还是将所有的中断都关断。

A: 要关所有调用了 OS 系统服务函数的中断。一般是 IRQ 中断。

第4章 芯片

第173问:

Q: 2100 实验板的问题。

使用几次 Re1InFLASH 后不能正常工作, 包括其他两种方式也不能工作在 Re1InFLASH 和 DeBUGInflash 时显示一个错误

```
Warning! interrupt vectors data is not correct!
```

```
Program you downloaded can not run freely!
```

不知怎么回事? 使用 DebugInRAM 虽然不报错, 但无法正常工作。

A: 可能是启用了加密功能引起的, 使用 ISP 擦除 FLASH。

第174问:

Q: 很有意思, 自从买了 2100 实验板一直在用 DebugInRAM, 今天想试试 DebugInFLASH, 后来就选了后者进行调试, 奇怪的现象发生了, 以后再进行 DebugInFLASH 之前确切的说是进入 AXD 在运行之前都显示着我第一次用 DebugInFLASH 时下装的程序, 一运行就可以显示当前程序的结果, 每次如此, 只要不运行其他程序, 实验板上的 LED 就保持第一次下载的程序的状态, RESET 和重新上电都不行, 是不是我的这个程序把开机的 DEMO 替换掉了, 哪里可以下载到那个 DEMO, 我想再装回去。

A: DebugInFLASH 就已经把程序下载到 FLASH 了, 原先出厂的程序已被你的程序覆盖。

第175问:

Q: 我将其按照 intle 32 bit hex 编译后将 hex 文件通过 ISP 下载到 2104 开发板上为什么没有反映? 后来再下载 C 语言编写的同样功能的文件, 前几次可以运行, 后来又下载了别的程序后就又不能运行了, 请问这是什么问题? 多谢!

A: 光盘上的 EINT1_LED.S 只适合于在 RAM 中调试, 如果要下载到 FLASH 中运行, 需要加入向量表(且要求向量表累加和为 0)。

Reset

```
LDR    PC, ResetAddr
LDR    PC, UndefinedAddr
LDR    PC, SWI_Addr
LDR    PC, PrefetchAddr
LDR    PC, DataAbortAddr
DCD    0xb9205f80
LDR    PC, [PC, #-0xff0]
LDR    PC, FIQ_Addr
```

```
ResetAddr    DCD    MAIN
UndefinedAddr DCD    Undefined
SWI_Addr     DCD    SoftwareInterrupt
PrefetchAddr DCD    PrefetchAbort
DataAbortAddr DCD    DataAbort
```

Nouse	DCD	0
IRQ_Addr	DCD	0
FIQ_Addr	DCD	FIQ_Handler
MAIN	...	

第176问:

Q: 手册上只有 I1.8, 而没有 I3.3, 无法计算片子功耗。

A: 3.3V 与外设相关。芯片本身在 3.3V 中的消耗可以忽略不计。

第177问:

Q: 通过查看数据手册 LPC2119 的接地有三种, 分别是 Vss (0V 电压参考点), Vssa (模拟地), Vssa_p11 (p11 模拟地), 三种的电压都为 0V, 但为了降低噪声和出错几率需要隔离, 请问如何隔离? 电源方面: 有两种供电电压, 一种是 1.8V (内核), 一种 3.3V (I/O 口) 分别存在隔离问题, 请问如何解决? 吾乃新手, 还忘老手不吝赐教, 谢谢!

A: 与普通的多种地布线类似 (它们有多少种方法就有多少种方法), 最终这些地线还是要接到一起。

第178问:

Q: 请问 LPC2119 的电源芯片选哪款较为合适? 有没有推荐的匹配?

A: 与 2104 一样, 可选 SPX1117。

第179问:

Q: 我现在数字电路除 LPC2114 使用 3.3V 电压外, 其余的大部分是 5V 的。我想先将 LPC2114 连接到 CPLD, 再连接到其它电路 (CPLD 的 I/O 口可以输出或输入 5V) 不知道行不行。谢谢。

A: 我觉的简单的电路可以用 LVC 芯片, 必要时可以用 3V 的 CPLD。

第180问:

Q: 请问: 你们公司的 ARM 芯片相对于 象 44BOX 等等一系列的 ARM 处理器的优点在那里? 好象 LPC2104 还比较的贵啊! 现在我正在考虑选择 ARM 芯片的问题, 望回答。谢谢

A: 我们的 LPC210X 内部有 FLASH 和 RAM, LPC211X 为工业级适用于工控领域, 并且是全球唯一可加密的 ARM。

以整个系统成本来说, LPC ARM 内部有 FLASH 和 RAM, 低功耗, 可加密性价比更高。具有 ARM 的性能, 单片机的特点, 在许多应用场合比其它的 ARM 更优越。

第181问:

Q: 我的 2214 板 I/O 口和中断均能正常工作, 今天准备调显示器, 显示器接在 CS2 上, 8 位数据线, 通过查资料, 得 cs2 的地址空间为 8200 0000, 于是我将命令口地址定为 0x8200 0001, 数据口地址定义为 0x8200 0000, 对啦, 数据命令切换线在 A0 上, 在显示器初始化前, 对 CS2 进行了配置, 即 BCFG2=0XCFFF FFFF, 即除修改成 8 位数据线外, 其它均不变, 对显示初始, 定时送显示器, 但是测量 CS2 口线, 根本不出现低电平, 更不用说数据线啦, 即好像外部总线根本没有工作。

之后根据 2214 的模板, 从新设置了参数, 结果写信号出不来, 片选, 读地址, 数据均有。

```
PINSEL2=0x0f814910;
```

```
BCFG2=0xcfffffff; // CS2, 8 位数据线,
```

请问, 配置外部总线还需要其它寄存器吗, 由于手上没有这方面的资料, 请给予帮助。

谢谢!

A: 注意 BCFG2 的写保护位。

第182问:

Q: 请问: 关于 LPC2214 的 BOOT0 和 BOOT1 的用法。

前提: 我使用内部 FLASH 作为程序存储器, 利用外部总线 CS0 扩展一片并口的 12 位 AD, CS1 扩展并口的液晶, CS2 扩展一片 RAM 芯片。

我的想法: 将 BOOT0 和 BOOT1 上拉, 系统上电启动后, 检测到 11, 那么从内部 FLASH 启动, 之后我可以设置管脚的选择寄存器, 使能 CS0 和 CS1 和外部总线, 那么我就可以随时的利用它的地址访问我的外部扩展的设备。

请问我的想法对吗, 并请指点一下我应该注意的相关的问题。

A: 正确。

第183问:

Q: 请问: 2214 外扩 SST39VF320 在 CS1 上为什么读 ID 不正确? 程序如下:

```
//设置 PINSEL2 位
```

```
#define P1_26_31_Debug 4 // JTAG D2
```

```
#define P1_16_25_Track 8 // 跟踪 D3
```

```
#define P2_Data_Bus 0x10 // D0-15 I/O (D5 D4=01)
```

```
#define P3_27_WE 0x100 // 写信号 D8
```

```
#define P3_26_CS1 0x800 // D11
```

```
#define P3_25_CS2 0x4000
```

```
#define P3_24_CS3 0x10000
```

```
#define P3_0_23_ADDR 0x0f800000 // 地址总线
```

```
//PINSEL0=P0_0_15_Set;
```

```
#define PIN2Set (P1_26_31_Debug+P1_16_25_Track+P2_Data_Bus+\n P3_27_WE+P3_26_CS1+P3_25_CS2+P3_24_CS3+P3_0_23_ADDR)
```

```
#define SST39VF320_Com_Add1 (*((volatile unsigned long *) 0x81005555))
```

```
#define SST39VF320_Com_Add2 (*((volatile unsigned long *) 0x81002aaa))
```

```
#define SST39VF320_ID1_Add (*((volatile unsigned long *) 0x81000000))
```

```
#define SST39VF320_ID2_Add (*(volatile unsigned long *) 0x81000001)

void InitGPIO()
{
    PINSEL2=PIN2Set;
    BCFG1=0x10000000;    // 16 位
}

void Check320(void)
{
    SST39VF320_Com_Add1=0xaa;
    SST39VF320_Com_Add2=0x55;
    SST39VF320_Com_Add1=0x90;
    // Delay_150_Nano_Seconds();
    // Read the product ID from 39VF320
    SST_id1=SST39VF320_ID1_Add;    // 读出 ID 不对
    SST_id2=SST39VF320_ID2_Add;    // 读出 ID 不对
    SST39VF320_Com_Add1=0xaa;
    SST39VF320_Com_Add2=0x55;
    SST39VF320_Com_Add1=0xF0;
}
```

A: CPU 的 A0 能不能接, 另外地址也要相应的变化 (针对 FLASH 的地址和针对 CPU 的地址不一样)。

第184问:

Q: 请问 1.8V 电源和 3.3V 电源的跳线要同时供电吗? 分别是给谁供电的?

A: 一个是给核心 (内核) 供电, 一个是给 I/O 口供电。

第185问:

Q: 1. ARM 的 P0.1 (WR)、P0.4 (RD) 为何不直接和以太网控制芯片的 IOWB、IORB 相连? NETCS 起什么作用?

2. 网卡控制芯片的 INTO 为何要接一个反相器和 ARM 的 P0.7 相连?

3. 我能否直接连接到 ARM 的 P0.16 (外部中断 0 输入) 口呢?

4. 8 位数据线和 5 位地址线怎么实现分时复用?

A: 1. 使用片选选择, NETCS 就是片选信号。

2. 因为 8019 的中断是输出高电平, 为了适应所选芯片, 所以加反相器。

3. 如果所选芯片支持低电平或跳变触发, 就可以不用加反相器而直接与外部中断输入相连。

4. 加锁存器。

第186问:

Q: 目前 210x 的各功能均试验完毕, 现在想知道的是其 IAP 功能是否可以向 PSD 产品那样实现远程的程序更新? 如果可以, 那么当程序代码写入 FLASH 后如何让程序转到更新后的程序? PSD 是通过寄存器来完成转换的, 那么 210x 如何实现呢? 比如 210x 的程序是从 0 地址开始执行的, 假如原代码长 2FFFH, 现在更新后的程序长 3500H, 那么在 IAP 时只能将代码写到非 00000000~00002FFFH 区域 FLASH 中, 写完后该如何让当前程序停止而转到更新后的程序呢? PSD 产品是因为有 2 块存储器, 在执行当前存储器中程序时可对另块存储器写操作, 写完后在将执行权交给另块存储器。期待您的回复!

A: 如果, 你的程序较小(64k), 你就把它当作两块独立的 FLASH(多个扇区)。
写入其中一块, 另一块空着...
程序开始执行的时候, 判断哪一块是最新的, 跳转过去就行了!
一旦从外界收到“更新程序”的命令, 检验数据, 更新另外一块, 就行了!
接收完毕, 烧写结束, 写入最新程序标志。立即跳过去。
下一次开机, 执行最新的。
这个方法, 比较安全。可以防止中途通讯结束, 复位等。
如果你的程序比较大, 有点悬……至少, 在某些干扰环境下, 不要那么做。

第187问:

Q: 开发板实验中 delay_ns(uint32 dly) 在上述条件下是延时 dly 秒吗? 如是则执行速度似乎并不快? 怎么估计出来的呢?

A: 如果程序和数据均在片内 RAM 中, 这样计算:

1. 一般指令需 1 个主时钟时。
2. 每次跳转增加 3 个主时钟时间 (也许是 2 个, 需要确认)。
3. 从 RAM 中取一个操作数多增加一个主时钟时间, 以此类推。
4. 保存一个结果到 RAM 中多增加一个主时钟时间, 以此类推。
5. 访问片内外设, 增加一个外设时钟时间。

注意伪指令 ldr rn, =x 需要从 ram 中取一个操作数。

第188问:

Q: 请问如果 lpc2104 和 5V 的芯片字节连接, 中间不接小电阻, 会怎样? 对系统有没有很大的影响?

A: 不安全而已。

第189问:

Q: 2104 的功耗最大在多少, (考虑外设全部工作, 功率损耗)?

EASY ARM 板上提供的电源最大输出电流好象只有 300mA 哦。

A: 芯片本身功耗最大值小于 70mA (120 摄氏度, 核 2.1V 供电, 运行于 60MHz)。一般情况小于 50mA。

IO 口耗电 (即 3.3V 耗电) 与其它电路密切相关, 实质是其它电路在消耗电源 (ARM 本身对 3.3V 的消耗是极小的)。

第190问:

Q: LPC2114 替换 LPC2104 的话, 我的程序还需要做哪些变动?

哪些内部专用寄存器的地址是否变动?

还有, 我原来设计电路用的是 LPC2104 的次要 JTAG 口, 现在换到 LPC2114 上来, 需要怎么改电路?

A: 程序无需改动, JTAG 设计时在 RTCK 接一个 4.7k 的下拉电阻即可。

第191问:

Q: 有一问题请教, 在市场上买的 ARM 核单片机的程序用 ADS1.2 建立项目, 添加初始化程序 vectors.s、init.s、target.c 和 target.h 后再写自己的主程序, 编译烧录到 ARM 中即可投入使用了吗? 而有的 ARM 嵌入式系统中还有 bootloader, 那是不是利用 ARM 开发的产品都要 bootloader, 而 bootloader 是通过什么软件编译的?

我所买的 easyarm 开发板的 lpc2104 中有 bootloader 吗?

A: bootloader 是用来下载程序到 FLASH 中的, LPC210x 具有 ISP 功能, 可直接使用 ISP 下载程序, 不需要自己编写 bootloader。

第192问:

Q: 我想用 IAP 把数据存到 2104 的 FLASH 里, 可我怎么读出来呢?

A: 定义一指针, 指向你用 IAP 写数据的那个区域, 就可以读了。

第193问:

Q: 当 PLLCFG 设置为 0x05 时的效果和设置为 0x25 时一样, 但晶振是 11.0592MHz 的, 如果设为 0x05, Fcco 的范围不在 156M-320MHz, 为什么还可以用, 而且和其它没有区别?

A: 手册说明的是保守值。实际的芯片的 Fcco 可能在更大的范围可以工作。

第194问:

Q: WDT 复位后从地址 0 开始执行吗? 运行程序 5.40 后, WDT 复位后并不是从地址 0 开始执行, 却跑去执行

[0x07806808] dcd 0x07806808,

并一直执行此语句。如何才能从地址 0 开始执行?

A: WDT 复位后是从 0 开始执行程序。可在复位处设置一些外部现象 (如 LED 闪烁), 然后脱机运行。

第195问:

Q: 我运行光盘上原始的 IAP 程序例子, 使用它的 project 是正常的, 但我自己重新建一个目录和 project, 参数设置按照原来工程的设置, 编译出来的 axf 文件就是会在运行到第一个 SelSector(1, 1)时无法继续下去, 我自己的 axf 文件和光盘上的一样大 16kB, 究竟是怎么一回事? 有一处设置 Language Settings 里的 ATPCS 项有什么意义? 其他

工程好像没有选择这一项。

A: IAP 是 THUMB 指令, 如果用户程序是 ARM 指令, 所以调用时需要使用 BX 指令。则 C 编译器的 ATPCS 项要设置 “ARM/Thumb interwork...”。

第196问:

Q: LPC2104 具有 6 个 PWM, 可以应用与电机控制, 可是却没有计数器进行电机转速的 PID 闭环控制, 这个情况是不是 LPC2106 的一个缺陷呢? 如果不是, 那应该怎么样才能进行外部脉冲的计数? 在不用外加计数器的情况下! 很是困惑!

A: 捕获算周期。

第197问:

Q: ISP 通讯的问题。

自己做的板子, 使用 ISP 下载程序, 提示无法通讯。板子上使用了跳线使得 44 脚和地连通, 也通过 max3232 进行电平转换了。还有复位电路感觉也一切正常。不知道为什么就是无法通讯。好晕啊, 各位成功下载的大侠们指点一下吧。试了好几天了, 都没法成功。

A: 你确保了 max3232 过来的电平正确了吗? 如果正确了, 还要确保通信方向是否弄反, 最后注意复位信号是否正确。

A: 搞定了, 原来是 max3232 的问题, 现在已经可以下载程序了!

第198问:

Q: 如何用 ISP 软件擦除 FLASH 中的内容啊?

我在调试 UART0 串口程序的时候出现问题, 上网看到很多文章都建议要先擦除 FLASH 中已有的内容, 于是用 ISP 软件进行擦除。但是在使用该软件过程中, 按到很多钮都说: 串口不能正常工作。

这是什么原因啊??? 请教高手正确使用 ISP 软件的方法及注意事项。

A: 看看你的跳线是不是按要求接好了!!

第199问:

Q: EasyArm2104 开发板上的 ISP 下载的时候通讯不上。请问, 要跳 JP6 吗?

A: 您可以自己试一下:

1. 是不是接的 UART0 口?
2. 系统晶振 (XATL.Freq) 是不是 11059.2kHz?
3. 串口选择是否正确?
4. 要求你复位的时候你是否有复位?
5. ISP 使能跳线使用 JP1 (JP6 和 JP7 保持主 JTAG 调试状态)!

第200问:

Q: 请教高手, lpc2104 和 lpc2114 外部晶振接法?

原来我用 LPC2104 做的板子, 外部晶振接到 X1 时不可用。后来接到 X2 上时好象可以用, 但这与资料上的说法好象有点出入。现在我想换 LPC2114 改版, 但不想再次改版了。故在此请教外部晶振接法?

A: 使用外部无源晶振, 晶振两个管脚无需连接, 与 51 单片机基本一致; 时钟信号由 X1 输入。

第201问:

Q: 请问有没有工业级的 21**系列?

A: lpc2114 等芯片就是工业级的。

第202问:

Q: 读 LPC2119/2129/2194/2292/2294 使用指南之迷惑。

1. 对于 2292/2294, 开放了外部总线, 程序应该可以放到外部, 外部和内部的存储空间地址不连续, 如何保证程序超过 128K 后自动跳转到外部执行程序?
2. VIC 部分, VIC 通道号 17#和 18#都是 EINT2, 是什么意思? 难道一个中断源占 2 个号? EINT3 的 VIC 通道号是多少?
3. 我购买的芯片是否内部已经有了 Boot 装载程序?

A: 1. 使用分散加载。
2. 可能是笔误。
3. 有。

第203问:

Q: 请问用 LPC2104 扩展 D12 有没有问题?

A: 可以使用模拟总线, 速度比 51 快几倍。

第204问:

Q: 请教: LPC2114 的 V3 能否接 2.8V?

系统有多个外设, 有一个使用 $2.8V \pm 5\%$ 的电平接口, 其它使用 3.3V, 那么 LPC2114 的 V3 怎样接?

V3 接 2.8V 电压基准是否可以, 口线与其它 3.3V 的接口能否直接相连?

A: 建议不要超出手册说明的范围使用芯片。

第205问:

Q: 我自己做的板子, 把 2104 的 44 脚 (EINT1) 接地, 再利用 LPC210XISP 软件通过串口 0 往芯片里写程序, 为何按了复位键以后连芯片的 ID 都不能读到?

A: 应该检查如下几点:

1. 确保你的芯片在工作。一般可以查看芯片的几个电源是否都正确, 查看晶体是否有波形, 最好还可以看看各电源的纹波是否严重。
2. 确保电平转换芯片的正确工作, 以及各输入输出的正确。这个方向最容易弄反。
3. 查看复位端的电平是否正确, 复位按键正常。
4. 确保 44 脚接地。

第206问:

Q: 请教如何通过 2104 的串口 1 来往芯片里写程序?

A: 使用 ISP 功能, 不能用串口 1, 只能用 UART0。

第207问:

Q: 请教 ARM2104 配套《ARM 微控制器基础与实战》上第一个实验的问题。

在 ADS1.2 中编译以下程序后进入 AXD 进行调试成功。

程序稍加修改, 使 LED1..LED4 同时闪烁, 在 ADS1.2 中编译以下程序进入 AXD 调试成功。

问题: 两个程序的软件延时部分相同, 但两个程序使 LED 的闪烁间隔时间不同, 不知是何原因?

A: 可以先使用 ISP 将芯片 FLASH 擦除试试, 可能是 FLASH 程序设置 PLL 的缘故。

第208问:

Q: 1. 《ARM 微控制器基础与实战》上说 32 字节的向量表的累加和必须为 0 才可脱机运行, 是否需要每次修改程序后修改第六个保留向量的值 (因为地址有改变)。

2. 《ARM 微控制器基础与实战》上说用 `ldr pc, resetaddr` 指令代替 `b` 指令可全空间跳转, 但是我看指令说明 `ldr` 指令的地址只能是当前地址的 $\pm 4k$ 呀!

A: 1. 不用修改。

2. 请看书。

第209问:

Q: 关于 2119 的 can 的通讯问题, 就是 2119 的波特率的设置与 SJA1000 有什么不同?

A: 基本上一致, 将 ARM7 的 Fvpb 与 SJA1000 的晶振对比:

BRP、SJW、TSEG1、TSEG2 均是同样的设置方法, 需要计算, 可参照 SJA1000 中位定时参数的计算方法。

第210问:

Q: 请问各位高手。新买的 ARM 芯片是不是直接安装上去就可以直接下载程序工作了, 还需要有什么别的设置吗。

A: LPC2000 系列有片内 FLASH 的都可以直接 ISP, 不过要保证硬件正常。

第211问:

Q: 2119 实时时钟电池怎么供电? 所有的电源引脚都要供电吗?

A: 没有独立的电池供电引脚, 以后会推出 RTC 独立晶振和备份电源接口的型号。

不太适合电池供电, 除此以外, 功能还是很强的。

第212问:

Q: 在调试实验程序的时候, 把低功耗掉电模式实验代码 PDRUN 编译后下载到 2104 FLASH 以后, 程序正常运行了。可是再 DEBUG 程序时, 在 AXD 下总是提示错误:

Error: flash is user configured protected

我能明白这个是因为 2104 一直处于掉电模式的原因, 可是如何解除这种状态, 把现在 FLASH 里的这段程序擦除掉啊?

A: 通过 ISP 软件擦除。

第213问:

Q: 请教关于“IAP 实验的问题”

《ARM 微控制器基础与实战》359 页的 IAP 实验是用汇编写的, 然而在光盘上 IAPtest 实验的程序是用 C 编写, 其中提到本实验是调用 IAP 服务程序。请问:

1. IAP 服务程序放在了什么地方 (汇编部分)?
2. 我想向其中写入 10 字节数据, 而不是诸如 256 个字节, 是否可以, 如何实现?
3. 在主程序中定义一个数组, 如何把此数组中的数据固定在特定地址中阿, 然后我好进行 RamToFlash。

A: 1. 在 BOOT 区, 入口地址为 0x7ffffff0, IAP 服务程序为 THUMB 指令。

2. 先读取原数据到 RAM 中。如果只需要保存少量数据, 且经常要更改, 最好使用外部 E2PROM。

3. 可能不好实现, 参考 ADS 帮助手册吧。

第214问:

Q: 请问 LPC2104 系统为什么采用 11.0592Mhz 的晶振, 是在定时上有优势还是其他原因。

A: 选用 11.0592MHz 只是为了得到精确的通信波特率, 串口通信的可靠性高。

第215问:

Q: 请教: lpc2104 和 RTL8019AS 的接口问题。

以前在 C8051F 上做, 高低地址是固定的, 容易接口, RTL8019 映射地址也容易算出。现在 lpc2104 是 32 位的, 32 个地址线和 32 个数据线, 和外围器件接口时带到很困惑。不知有没有高手看过周立功网站的提供的 lpc2104+RTL8019AS 的接口电路, SA0-SA4 和 lpc2104 的哪个脚接在一起? 请指点由一个 573 来控制地址与数据总线复用现在有点明白虚拟总线了, 加上一个地址锁存器, 和 51 机的就很相似了。

A: 对, 为了兼容 51。

第216问:

Q: 开发板上的 74HC125 的 U5C、U5D 起什么作用?

A: 去抖动。

第217问:

Q: 请教: 在 2100 开发板的 ADC 实验 (P119 面) 中为什么每次 AD 转换都要启动两次?

A: 切换通道后, 丢弃第一次 ADC 转换值。

第218问:

Q: 如果我的 AD 转换需要的满量程是 0V~5V, 是不是把 AD 部分的电源即 V3A 由现在的 3.3V 改为 5V 供电就可以了?

A: 把输入的电压用电阻分压就可以了。

第219问:

Q: 疑问: 2104 开发板上用了 MAX708, 其已有高/低电平复位输出, 为什么还要用 74HC125 呢?

A: 为了使手动与 JTAG 都可以对芯片复位。

第220问:

Q: 请教高手关于自己做板子的问题。

我自己做了一个 2104 的板子, 第一次上电可以通过串口顺利地把 .hex 文件从串口写入 flash, 然后再次上电, 程序就不执行了, 并且再次用周立功公司提供的 ISP 软件却无法访问该芯片。同样地程序我放到周立功地 2104 的开放板用同样的步骤, 它却能正常运行, 我的板子基本上是按《ARM 微控制器基础与实战》上的图制作的, 请指点我的问题可能的原因在哪里? 请教自己做板子容易出错和注意的地方。

A: 重新上电试试。

P0.14 口是否已接为低电平?

测一下晶振是否起振?

Q: 晶振测过已经起振, 运行自己的程序不是要把 P0.14 口断开吗? 用 ISP 软件下载程序的时候才把 P0.14 接地是不是?

A: 需要 ISP 时 P0.14 口接地;

需要运行用户程序时, P0.14 口要接一个上拉电阻, 因为 P0.14 内部无上拉(作为输入时)。

第221问:

Q: 请问 2104 的复位电路我不用 MAX708 和 74HC125, 做成象 51 单片机那样的复位电路可以吗?

A: 做成这样是为了保证 JTAG 接口和按钮、上电都可以复位芯片!

实现线与逻辑, 如果你不需要调试只是生产的话, 完全可以那样设计复位电路。

第222问:

Q: 关于 REMAP

请问 remap 的时候只要给 MEMMAP 赋值就好了吗? 具体的中断向量表的映射系统自动给你生成?

在启动代码中关于 remap 我只看到赋值, 而在《ARM 微控制器基础与实战》上写了一段重映射的代码示例 P280 程序清单 5.2。

如果系统自动 remap 的话，那么映射到的地址空间我们没法改动了？
代码中，通过 `B label1` 或 `bl Label2` 进行跳转时，要不要进行栈的操作？

如果调用的是 c 语言函数，除了传递必要的参数，有没有保存寄存器呢？

A: 中断向量表不是系统自动生成的，是由用户编写。Remap 通过改变 MEMMAP 的值实现。
请看一下“ARM-Thumb 过程调用标准”(ATPCS.pdf) 这篇文档。安装 ADS 后就有。

第223问:

Q: 请问: lpc2104 运行速度?

ARM7TDMI (-S) 能提供 0.9MIPS/MHz 的指令执行速度，lpc2104 cclk 为 60MHz 时指令执行速度应为 $0.9 \times 60 = 54$ MIPS，这样得到一条指令的执行速度大概为 0.018us。可我用示波器观察计算得出一条指令的执行时间远远大于这个数值，大概为 0.1us-0.8us (可能由于流水线的影响，指令执行时间的差别很大)。这是怎么回事呀？

A: 这是一个在存储器带宽足够时的平均值。如果程序和数据均在片内 RAM 中，这样计算

1. 一般指令需 1 个主时钟时间
2. 每次跳转增加 3 个主时钟时间 (也许是 2 个，需要确认)
3. 从 RAM 中取一个操作数多增加一个主时钟时间，以次类推
4. 保存一个结果到 RAM 中多增加一个主时钟时间，以次类推
5. 访问片内外设，增加一个外设时钟时间

注意伪指令 `ldr rn, =x` 需要从 ram 中取一个操作数。

在 flash 中如果 MAM 配置为最优，平均速度与在 ram 中相差无几。

第224问:

Q: 请教斑竹关于 211x 和 22xx 系列 P1.16-P1.25 这些引脚的作用？

另外在开发板上除了 JTAG 口外，上述引脚是否用到？

在设计自己的系统时，如果将 P1.16-P1.25 这些脚作为普通输入输出用，是否会影响系统的调试？

A: ETM 跟踪端口。

一般 JTAG 仿真器不使用这些引脚，所以设计自己的系统时用作 I/O 即可。不会影响调试。

第225问:

Q: 请问，请问评估板上的 S-1131B 哪里能订购到，或者有什么替换型号。谢谢

A: SPX1117M3_1.8 和 SPX1117M3_3.3。

<http://www.zlgmcu.com/Sipex/power/SPX1117.asp>。

第226问:

Q: 关于 P2104 的驱动能力!

我正在用 2104 外接 LCD 其中用 P0 口模拟总线 我的 LCD 电压是 5V 的，而 2104 是 3.3V 的，能否直接相接？我没接时还有信号输出但接上之后就什么信号都没有了，请问是何缘故？是否需要电平转换芯片或驱动芯片？上拉电阻能否满足要求？

A: 中间串一个小电阻试一试。我们接过多种总线器件都没有问题。

第227问:

Q: 问一下, lpc2104 的 I/O 的驱动能力?

单个 I/O 可以走多大电流?

全体 I/O 可以走多大电流?

A: 数据手册上有, Ioh, Iol。

第228问:

Q: 2119 最高频率可达 60MHz, 在频率比较高的时候应该使用几层的电路板, 有没有确切的规定?

A: 不知道什么时候有人就把 PCB 的层数与频率划上了一个必然的关系。60MHz 很高吗?

看看你的调频收音机, FM 波段范围是 88—108MHz, 内部本振频率可达 118.7MHz, 1 层板! 看看电视机里头的电路板, 高频盒内一般最多就 2 层板(还是一层的居多)它跑多高的频率? 自己查查 UHF 的范围吧, 记得保持冷静因为没有什么值得去惊讶的! 再有几年前拆过一个 GIGA 的游戏机, 音频部分的 PCB 是 4 层板(独立的小板), 小日本的东西元器件密密麻麻的, 看看这个它又能是跑多高的频率? 况且现在数字电路的电平容差相对于模拟电路的至少有一个数量级的差别。在高频模拟电路下, 多一块覆铜可能就使电路的性能下降或工作点异常, 例如它可能使电感特性器件的 Q 值下降、产生涡流损耗等。很多时候增加 PCB 的层数决不是单纯因为系统速度, 你所说的 60MHz 也只是芯片内部的系统时钟, 难道你的 IO 也要作为时钟源使用吗? 在 ARM 中采用 PLL 技术使得外部时钟可以下降到一个较低的水平, 则可以抑制部分由时钟电路产生的射频干扰, 有较好的 EMC、EMI 特性, 一个合理的 PCB layout 可能比盲目增加 PCB 层数的方法提高系统的稳定性来得更合理和有效! 看你是想做精品还是想做产品或其他, 这就要综合多方因素考虑了。

无可否认, 增加 PCB 的层数后比较容易的处理一些由于电源寄生干扰产生的问题, 适当配合 layout 可以满足一些速度(时序)要求苛刻的电路。由于现在元器件体积日渐小型化, PCB 上元器件密度大, 很多时候单面或双面的 layout 实现已经非常困难, 增加 PCB 层数是为了更好的使设计变得合理和可靠。

第229问:

Q: ARM 抗干扰能力怎样, 是否低压器件一定比高压器件抗干扰能力弱? 谢谢!

A: 我们用自己的开发板打群脉冲到最高都正常工作(程序没有任何抗干扰措施)。

第230问:

Q: 我刚刚设计完了一块 LPC2106 的电路板。在用 EASYARM 仿真器通过 JTAG 接口调试没有能够使用, 但是通过 JTAG 口下载到 FLASH 后上电复位却不能正常工作, 手动复位也不行。同样的程序在购买的实验板上完全可以使用。不知道是怎么回事。我初步怀疑是复位电路问题, 但不知道怎么解决。

A: 问题解决了! 只要将 P0.14 上拉到高电平就可以了。

第231问:

Q: VPB 是什么意思? 英文全称是什么?

A: VPB (VLSI Peripheral Bus)

第232问:

Q: 请问大侠们, 2104 怎样与 5V 的外设连接, 多谢!!

下一个产品想用 2104, 可是用惯了 51, 不知怎样与 5V 外设连接, 多谢!!

A: 可以考虑使用 3.3V 至 5V 的缓冲器件如 4245, 最简单的方法就串上个小电阻。

第233问:

Q: 请教大家: LPC2104 芯片中的 PWM 有中断功能, 请问输出置 PWM 匹配通道的中断标志有什么意义, 有什么作用?

A: 举个例子, 如果 使用 PWM 滤波 生成正弦波, 有中断, 就方便很多!

第5章 GPIO

第234问:

Q: 我是在 LPC2119 与光隔相连时遇到的这个问题, 以前用的片子 IO 口都是 5V 电压, 由于 2119 是

3.3V 所以和以前系统中采用的高速光耦相连时就涉及到两个问题:

接受光耦送来的信号 2119 可否承受?

还有 2119 输出的信号能否驱动光耦?

搞 CAN 的朋友可否同样遇到此问题, 我觉得可以解决这个问题, 但不敢肯定, 我想通过分析光耦内部结构可以解决这个问题, 有经验的朋友谈谈吧

A: 可以将 CAN 引脚当成普通 IO 口对待, 就不会存在疑惑了。

第235问:

Q: LPC2119 的 IO 口输出电压是多少, 我在数据手册上没有找到?

还有它的 IO 口带不带上拉电阻?

A: GPIO 输出可输出高电平电压约为 3.3V, 手册上的 Voh。

GPIO 作输入时没有内部上拉。

第236问:

Q: 请教:LPC2214 的 P2,P3 作为普通 IO 时, 怎么设置输入输出方向? 没有找到它们的 IODIR!

A: 与 P0 一样, 请下载工程模板, 内有头文件。

第237问:

Q: 请问: LPC2214 的 GPIO 的管脚在输入时没有上拉, 在输出时有没有上拉?

好像在资料里说的都没有上拉。

A: P0.2、P0.3 是 I2C 接口, 是开漏输出, 所以要外接上拉电阻, 其它 GPIO 不用接。

第238问:

Q: LPC2104 实验板接上 JTAG 线时, P0.22 变为 0? 我在测试键盘时发现的, 一拿下 JTAG 插头, 一切正常。

A: 使用主 JTAG 调试时, P0.22 为 ETM 跟踪调试口, 不能作 GPIO 用。

第239问:

Q: 请教 LPC2214 的 P2/P3 口做 I/O 口扩展而不是 EMC 怎样访问? 手册上没有找到, 只有 P0/P1 的访问方法。

A: 同 P0 口一样。请下载“EasyARM2100 开发套件快速入门和 L...”, 里面的头文件有其地址的定义。

第240问:

Q: GPIO 哪些内部有上拉, 哪些没有?

A: 《ARM 微控制器基础与实战》中说的明白, 都没有上拉电阻的, 不过是推挽输出的, 输入 0、1 都没问题, 放心!

第241问:

Q: 因为看到 LPC2114 外围电源用的是 3.3V 供电, 而我想用它的 I2C 接口去控制后面用 5V 供电的芯片, 因此想问一下是否可以直接接上拉电阻和 5V 相连? 如果不能是不是只能加电平转换了? 谢谢!

A: 可以。

第242问:

Q: 当外部有数据送到 IO 口时, 是不是通过读寄存器 IOPIN 就可以得到其 IO 的当前状态呢?

A: 你说的对, 但是要将你要用的 IO 口设为 GPIO (PINSELO\PINSEL1) 方式, 并将其设为输入方式 (IODIR), 就可以了。不过要是你用的 IO 口不连续的话, 要将输入的管脚状态处理一下才是你要的字节或是字数据。我就是用这种方式扩展外部 CAN 总线器件的。

第243问:

Q: 当用用 GPIO 时, 输出电流是多大呢? 急用!

A: 数据手册上有, $I_{oh} = 4mA$ 。

第244问:

Q: 请教: 2104 的 IOPIN 这个寄存器为什么老是一个值。

我的 IO 不断变化, 他就是不变, 不知为什么?

A: 我认为可能是 IODIR (管脚方向寄存器) 和 PINSELO、PINSEL1 (管脚功能寄存器) 中的一个寄存器没设置好吧。

第245问:

Q: 可以直接把+5V RAM/FIFO 的数据线和 2104 连接吗 (模拟总线时)?

A: 我做过外接 RAM 的实验, RAM 输入的直接连, 输出到 ARM 的串电阻。

第246问:

Q: 我用 2104 的时候, 怎么设置 IO 口, 比如例子上有一个: #define LEDCON 0x00002000。

为什么设置 P0.13 的时候要定义为 0x00002000, 是怎么计算的。谢了?

A: 右移 13 位 ($1 \ll 13$)。

第247问:

Q: 请问高电平输出电压问题?

我想使用 T6963c 的液晶控制器, 他的接口电压最低为 vdd-2.2, 即为 2.8V, 不知 LPC 的 GPIO 串 100 欧姆电阻是否能达到这个要求? 谢谢!

A: 如果为 COMS 电路肯定可以, 其它电路可以做试验。不行的话可以不串电阻。

第248问:

Q: 我买了一套 EasyARM 2104 开发板, 现在想给他加一个 12232 的液晶屏, 但屏幕就是显示乱码, 察看发现当

```
PINSEL0 = 0x00000000;
```

```
PINSEL1 = 0x00000000;
```

```
IODIR = 0xffffffff;
```

```
IOSET = 0xFFFFFFFF;
```

此时, 回读 IOPIN 的值却是 0X0001FFFF, 请问怎么回事?

```
IOCLR = 0xFFFFFFFF;
```

此时, 回读 IOPIN 的值是 0X00000000 正常

当 IODIR = 0x00000000 时, 回读 IOPIN 的值却是 0X000XXXXX 值不定, 请问又是怎么回事?

六个按键, 同时按下后, IOPIN 的值只有三个位变化, 是不是我的板子坏了?

A: 使用主要 JTAG 调试, 高 15 各 IO 口被占用。

第249问:

Q: 请问 LPC2104 的 32 个 I/O 口读入数据怎么设计?

想给它接一个显示屏, 我用 P0.0-p0.7 作为数据线 P0.8 作 RD, P0.9 作 WR, 我不知道怎样读入数据, 按照资料上说的 IOPIN 反映外部引脚的状态, 可是我观察 IOPIN 和引脚上的电平不对应。请大家指点。

A: 1. 相应的 I/O 要设置为 GPIO, 即设置 PINSEL0/1;

2. 要设置为输入模式, 即设置 IODIR;

3. 参考 LPC2104 的模拟总线软件包 <http://www.zlgmcu.com/download/downs.asp?ID=959>。

注意: I2C 接口不是推挽输出, 而是开漏输出, 做 GPIO 时需要接上拉电阻。

第250问:

Q: 关于 2119 的接口电平的问题

1. 好像有人说 2119 的 GPIO 可以工作在 5V, 那么我是否可以将 2119 开发板的 GPIO 的工作电源改为 5V? 因为我的应用项目中外部 IC 都是 5V 供电的。

2. 如果是只能工作在 3.3V, 那么请你们给出和 5V 供电的 IC 接口时的标准的接法, 即输入和输出口线的上拉电阻和限流电阻等。

A: 1. 芯片数据手册参数表有说明(Vi), GPIO 可以承受 5V 电压, 但芯片电源 V3 不能接 5V。

2. 与外围 5V 器件接口时, 一般在口线上串接一个小电阻即可直接连接。(50--470 欧) 如果电平不能匹配, 则需要电平转换芯片。

第251问:

Q: 请问一下, 我使用 2104 的 IO 口与 P0.5 口直接相连, 希望将 P0.5 口置为低电平, 但此时 P0.5 口的电压却为 2.45V 左右, 请问为什么, 该如何解决?

A: 将 JP9 断开试一试, 估计您将这个 IO 口设置为输出了。

第6章 中断

第252问:

Q: LPC210x 怎么样才能进行外部脉冲记数啊?

A: 使用外部中断功能实现。

第253问:

Q: 问一下, lpc2104 中, Boot Block 中断向量与从 0x00000000 开始的 32 字节中断向量一样不?

假设不一样的话 Boot Block 中断程序是定义在 Boot Block 中么?

A: 从 0x00000000 开始的 32 字节中断向量的指令译码有些是一样的, 这可以通过反汇编看。Boot Block 中断程序不是定义在 Boot Block 中, 而只是在其中放一条跳转指令。

第254问:

Q: 我的板子开始调试时, 在 FLASH 和 RAM 中调试都没有问题, 后来在 RAM 中调试 I2C 中断时 (用 LPC2200 的模板调试), 发现不能产生中断, 后来我就改为 FLASH 中调试, 就出现 RDI Warning 00254 的错误提示, 所以我问 2 个问题:

1. 我的 I2C 为什么不能中断: 可是例子里的可以, 看不出区别?
2. 是不是片子坏了, 因为我的板子在 RAM 里调试好的, 写了一下 FLASH 就不行了, 出现上面的提示?

A: CPSR 的 I 位是否设置正确? 在 InitStack 最后或 InitStack 之后的任何地方用程序设置 “MSR CPSR_c, #0x5f”。

第255问:

S: 关于中断处理应该注意的问题

前几天做中断试验, 一直进不了中断, 今天在 zhenyu 的帮助下, 终于可以进入中断了! 下面是看资料时整理的使用的一般过程, 希望对刚接触飞利浦 ARM 的有所帮助, 也希望大虾们能提一些自己的看法。

以下过程主要参考 ZLG 翻译的 ARM 资料和 LPC2104 英文资料以及 ARM 公司专门介绍 VIC 的资料 (ARM PrimeCell Vectored Interrupt Controller)整理。

1. 确定中断的类型, 是 FIQ 还是 IRQ, 如果是 IRQ, 还要进一步区分是向量 IRQ 还是非向量 IRQ。
2. 如果是 FIQ, 就会有 FIQ 的中断服务程序 (ISR)。应该在中断向量 0x1C 处放置转向 FIQ 的 ISR 的语句。对于 FIQ 的 ISR, 如果只有一个中断设置为 FIQ, 没有特殊要求, 像一般的函数一样就行了。如果有多个中断设置为 FI, 则要在 ISR 中读取 VICFIQStatus, 判断是哪一个中断发出的中断请求, 并做相应的处理。判断中断源的顺序, 就是设置为 FIQ 的中断的优先级。为了减少中断反应时间, 一般建议使一个中断设置为 FIQ。
3. 如果设置为非向量 IRQ, 在 VICDefVectAddr 中设置非向量中断服务程序的地址。在非向量中断服务程序中, 需要读取 VICIRQStatus 的值, 判断是谁请求了中断, 并做

处理。

4. 如果设置为向量 IRQ, 则
 - 1) 确定该中断使用的 Slot (设置相应的 VICVectCnt1X(X:0-15));
 - 2) 在对应的向量地址寄存器 (VICVectAddrX(X:0-15)) 中写入该中断的服务程序地址。
需要注意: 在该中断的服务程序最后, 需要向 VICVectAddr 执行写操作来复位 VIC。
5. 需要注意的其他问题:
 - 1) 允许了 VIC 中断 (设置 VICIntEnable) 还需要允许 IRQ 中断 (CPSR 寄存器的 I 位) 才能进入中断 (我就是没有设置 I 位才进不了中断的, 是 zhenyu 帮我搞定的!)
 - 2) 如果对 VIC 有兴趣可以看看 ARM 公司的 ARM PrimeCell Vectored Interrupt Controller (PL192) Technical Reference Manual。

第256问:

S: 还说中断

1. 无论是向量中断还是非向量中断, 都需要在中断返回前向 VICVectAddr 执行写操作。
2. 中断函数必须使用关键词 __irq, 否则可能返回错误! (至于 ucossii 中不需要, 是因为具有汇编接口代码。)

第257问:

Q: 利用模板生成的程序, 下面的程序错在哪里, 怎么不能进中断?

```
#include "LPC2294.h"

#define LED1 0x200
#define LED2 0x400
#define LED3 0x800
#define LED4 0x1000

//typedef unsigned char uchar;
//typedef unsigned int uint;

void __irq eint0(void);
void __irq irqISR(void);

int main(void)
{
    int temp;

    PINSEL0=0xa0000000;
    PINSEL1=0x01;

    IODIR=LED1|LED2|LED3|LED4;
```

```
IOSET=LED1|LED2|LED3|LED4; //for test
//IOCLR=LED1|LED2|LED3|LED4;

VICDefVectAddr=(int)irqISR;

VICVectCntl0=0x2e;
VICVectAddr0=(int)eint0;

VICVectCntl1=0x2f;
VICVectAddr1=(int)eint0;

VICVectCntl2=0x30;
VICVectAddr2=(int)eint0;

//VICVectAddr=0;
VICIntSelect=0x00000000;
VICIntEnable=0x0001c000;
EXTINT=0x07;

while(1)
{
    temp=0;
}

void __irq eint0(void)
{
    int temp;

    temp=VICIRQStatus;

    temp=IOPIN;
    if((temp&LED1)==0)
    {
        IOSET=LED1;
    }
    else
    {
        IOCLR=LED1;
    }
}
```

```
    }

    while((EXTINT&0x07)!=0)
    {
        EXTINT=0x07;
    }

    VICVectAddr=0;
    return;
}

void __irq irqISR(void)
{
    int temp;

    temp=VICIRQStatus;

    return;
}
```

A: 在 Startup.s 中有如一段设置堆栈的汇编，在其中找到下面的语句：

```
MSR    CPSR_c, #0xdf
```

然后将其改为：

```
MSR    CPSR_c, #0x5f
```

第258问：

Q: 我用的是你们公司的“ARM 微控制器基础与实战”一书，开发板是 EASYARM2104，我试的程序是书上 303 页的“外部中断 1 实验的 C 语言代码”。我花了一天时间了，反正直接用光盘上程序是不行。是不是要修改什么地方，比如 VECTORS.S 文件里的东西。因为我初学，对堆栈哪设置的代码看不太懂，所以我没法在 VECTORS.S 里找到哪错了，但感觉是这里有问题！你们的 VECTORS.S 文件里有两行这个代码，

```
MSR    CPSR_c, #0x1f
MOV    PC, R0
```

我去掉反而调试时可以进入 main，不去单步跟踪就卡在 InitStack 函数里了。请问到底该怎么改？

A: 这可能是因为 CPU 已产生了中断标志(如芯片内部固化有程序，并打开了某项中断)，在 InitStack 中一旦打开 IRQ 中断允许，CPU 即进入 IRQ 中断，就进不到 main 函数了。可以先用 ISP 软件将芯片全部擦除，再进行调试。

第259问：

Q: 各位大哥, 我用 EASYARM 开发板, LPC2104 芯片。例子是 ARM 微控制器基础与实战一书上的 IRQ 中断试验。303 页那个! 我把源程序直接输入, 运行, 然后按板子上的 KEY1 键, 但中断没有产生! 请问是不是书中例子程序有误, 关于 IRQ 的中断初始化设置, 是不是要在 Vectors.s 文件中进行设置, 该如何设置?

是不是要仿照下面这样设置! 还缺些什么?

```

;快速中断
IRQ_Handler
    STMFD    SP!, {R0-R3, LR}
    BL      IRQ_Exception      // 跳到这个中断函数处
    LDMFD    SP!, {R0-R3, LR}
    SUBS    PC, LR, #4
    
```

如果是, 哪中断函数执行完了, 又该如何返回。请教了!

A: 程序在产品光盘上的“实验程序\VIC_test\EINT1_test” Vectors.s 文件中要打开 IRQ 中断允许 (InitStack 子程序):

```

.....
MSR    CPSR_c, #0x1f
MOV    PC, R0
    
```

第260问:

Q: 在 LPC2200 工程模板中的“IRQ.S”的作用是什么, IRQ 中断不是可以通过 VIC 来控制就可以了?

A: VIC 没有管理中断嵌套功能。IRQ.S 是实现中断嵌套而提供的程序, 主要是进行现场的保护及恢复。

第261问:

Q: 请问 2114 的中断初始化程序这一段为什么执行到 enable 之后就自动进入非向量中断处理程序呢?

```

/*****
*中断初始化函数
*****/
void Vic_Init(void)
{
    PINSEL0 |= 0xa0000000;          // P0.14, P0.15 设置为 EINT1, EINT2
    PINSEL0 &= 0xfffffff;
    PINSEL1 |= 0x00000001;          // P0.16 设置为 EINT0;
    PINSEL1 &= 0xfffffff;

    VICIntSelect = 0x00008000;      // EINT1 设置为 FIQ 模式, 其它为 IRQ
模式

    VICDefVectAddr = (int)IRQ_Exception; // 非向量中断入口地址
    
```



```
VICVectCntl0 = 0x2E; // EINT0 设置为向量中断, 对应 slot0
VICVectAddr0 = (int)IRQ_Vect_Eint0; // Eint0 对应的中断入口地址

VICVectCntl1 = 0x2d; // RTC 中断设在 slot1
VICVectAddr1 = (int)IRQ_Vect_Rtc;

VICIntEnable = 0x1e000; // 使能 EINT0, EINT1, EINT2, RTC

EXTINT = 0x7; // 清外部中断标志
ILR = 0x1; // 清 RTC 中断标志

}
```

A: 请不要单步执行!

第262问:

Q: 2104 中断处理程序结束后, 要对向量地址寄存器写零, 通知 vic 中断结束。是不是所有的中断程序, 包括 fiq、向量 irq、非向量 irq 都要这样做?

A: 非向量好象不需要这样做。

第263问:

Q: 以下程序为什么“VICVectCntl0 = 0x24;”就能说明是 t0 的中断? 而且我的程序运行的时候为什么不向中断程序跳转?

```
TOPR = 99; // 设置定时器分频为 100 分频, 得 110592Hz
TOMCR = 0x03; // 匹配通道 0 匹配中断并复位 TOTC
TOMR0 = 110592/5; // 比较值
TOTCR = 0x03; // 启动并复位 TOTC
TOTCR = 0x01;

/* 设置定时器 0 中断 IRQ */
VICIntSelect = 0x00000000; // 所有中断通道设置为 IRQ 中断
VICVectCntl0 = 0x24; // 定时器 0 中断通道分配最高优先级(向量控制器
0)
VICVectAddr0 = (uint32)IRQ_t0; // 设置中断服务程序地址向量
VICIntEnable = 0x00000010; // 使能定时器 0 中断
```

A: 看相关寄存器的说明及《ARM 微控制器基础与实战》3.5.5 节。不能中断可能是内核没有允许中断。

第264问:

Q: 请大侠们谈一谈向量中断与非向量中断的区别!

A: 区别是向量有大小有方向, 非向量只有大小没有方向。

具体说就是向量中断有唯一的向量地址、优先级别、中断服务, 非向量中断共用一个向

量地址、中断服务，在共用的中断服务中需要读中断状态寄存器来区分非向量中断源。

第265问：

Q: 我本来想在移植好的 UC/OS II 中把外部中断也加上，但是当我按外部键进入外部中断程序后，发现时钟中断以后就不发生了，而外部中断都可随时进入。在脱离 UC/OS II 的情况下，也把时钟中断和外部中断加在一个工程中，现象是一样的。

A: 应该是您的程序的问题，主要可能是没有考虑中断重入。请试一试我们的工程模板

第266问：

Q: 我购买了 easy arm 开发板, 在学习过程中遇到了一些问题, 请教大侠:
在《ARM 微控制器基础与实战》程序清单 5.24, 设置定时器 0 中断 IRQ, 其中
VICVectCntl0 = 0x24; 为什么说是为定时器 0 分配最高优先级?
VICIntEnable = 0x00000010; 为什么说是为定时器 0 使能中断?

A: 定时器 0 为 VIC 通道 4, “VICVectCnt0 = 0x24” 的意思是为 4 号 VIC 通道(定时器 0) 分配为 IRQ slot0, 并使能。同时由于定时器 0 为 VIC 通道 4, 所以将 VICIntEnable 第 4 位设为 1, 即使能使定时器 0 中断。

详细情况可看《ARM 微控制器基础与实战》149 页及 150 页 VICVECTCNTL0 寄存器和 151 页中断源。

第267问：

Q: 如果程序有一个间隔为 10ms 的节拍中断程序，但中断程序的执行时间超过了 10ms，会出现什么结果？

A: 时钟节拍不准，造成延时、超时时间加长。应该避免这种情况，可把中断的一些处理移到任务中处理。

第268问：

Q: RTC 时钟设置正确，用查询方式可读到报警中断标志，但进不了中断，程序已 REMAP。

```
AREA Vect, CODE, READONLY
ENTRY ;程序入口

Vectors

LDR PC, ResetAddr
LDR PC, UndefinedAddr
LDR PC, SWI_Addr
LDR PC, PrefetchAddr
LDR PC, DataAbortAddr
DCD 0 ;0xb9205f80
LDR PC, [PC, #-0xff0]
LDR PC, FIQ_Addr

ResetAddr DCD Reset
```

```

UndefinedAddr      DCD      Undefined
SWI_Addr           DCD      SoftwareInterrupt
PrefetchAddr       DCD      PrefetchAbort
DataAbortAddr      DCD      DataAbort
Nouse              DCD      0
IRQ_Addr           DCD      INTRTC          ;IRQ_Handler
FIQ_Addr           DCD      FIQ_Handler

```

; 中断服务程序

```

Reset
Undefined
SoftwareInterrupt
PrefetchAbort
DataAbort
FIQ_Handler

```

B STARTC

STARTC ; 进行系统初始化

```

MSR      CPSR_c, #0xD3
LDR      SP, =0x40003000
MSR      CPSR_c, #0xD2
LDR      SP, =0x40003100
MSR      CPSR_c, #0xD1
LDR      SP, =0x40003200
MSR      CPSR_c, #0x10
LDR      SP, =0x40003300
NOP

```

```

LDR      R0, =VIC_BASE_ADDR      ;init VIC
MOV      R1, #0x2000
STR      R1, [R0, #0x0C]         ;RTC=IRQ
LDR      R0, =VICVectCnt10
MOV      R1, #0x2F
STR      R1, [R0]                ;FIRST
LDR      R1, =INTRTC
LDR      R0, =VICVectAddr0
STR      R1, [R0]                ;addr
LDR      R0, =VICIntEnable
MOV      R1, #0x2000
STR      R1, [R0]

```

.....

; 下面是 RTC 中断服务程序

INTRTC

```

        STMFD    SP!, {R0-R7, LR}
        NOP
        LDMFD    SP!, {R0-R7, LR}
        SUBS    PC, LR, #4
    
```

A: 设置问题:

1. 这里 VICIntSelect=0x2000, 即把 RTC 中断设置为 FIQ 中断了(为 0 才是 IRQ 中断)。

```

        ...
        LDR    R0, =VIC_BASE_ADDR    ;init VIC
        MOV    R1, #0x2000
        STR    R1, [R0, #0x0C]      ;RTC=IRQ
        ...
    
```

2. 应设置 VICVectCnt10 = 0x2d。

```

        ...
        LDR    R0, =VICVectCnt10
        MOV    R1, #0x2F
        STR    R1, [R0]            ;FIRST
        ...
    
```

第269问:

Q: 例程 TIMESCIIR 改成用向量中断方式实现 RTC 程序跑不起是什么问题?

改动部份如下:

```

void __irq IRQ_Exception(void)
{
    if(rtc==0)
    {
        IOCLR = LED1CON;           // 点亮 LED1
        ILR = 0x01;                // 清除中断标志
    }
    else
    {
        IOSET = LED1CON;          // 熄灭 LED1
        ILR = 0x01;                // 清除中断标志
    }

    rtc = ~rtc;
}

void RTCIni(void)
{
    PREINT = FPCLK / 32768 - 1;    // 设置基准时钟分频器
    PREFRAC = FPCLK - (FPCLK / 32768) * 32768;

    YEAR = 2003;                  // 初化年
    MONTH = 6;                    // 初化月
}
    
```

```

DOM    = 20;                // 初始化日
HOUR   = 00;
MIN    = 00;
SEC    = 00;

CIIR   = 0x01;              // 设置秒值的增量产生一次中断

/* 设置实时时钟中断程序 */
VICIntSelect = 0x00;        // 所有中断通道设置为 IRQ 中断
VICVectCnt11 = 0x2d;        // 中断通道分配次高优先级(向量
控制器 0)
VICVectAddr1 = (uint32)IRQ_Exception; // 设置中断服务程序地址向量
VICIntEnable = 0x00002000;    // 使能实时时钟中断
CCR = 0x01;                 // 启动 RTC
}

```

;引入的外部标号在这声明

```

IMPORT  FIQ_Exception
IMPORT  Reset
IMPORT  IRQ_Exception

```

;快速中断异常处理程序

;给外部使用的标号在这声明

```

EXPORT  InitStack
EXPORT  Vectors

```

CODE32

AREA Startup, CODE, READONLY

;中断向量表

Vectors

```

LDR    PC, ResetAddr
LDR    PC, UndefinedAddr
LDR    PC, SWI_Addr
LDR    PC, PrefetchAddr
LDR    PC, DataAbortAddr
DCD    0xb9205f80
LDR    PC, [PC, #-0xff0]
LDR    PC, FIQ_Addr

```

```

ResetAddr      DCD    Reset
UndefinedAddr   DCD    Undefined
SWI_Addr       DCD    SoftwareInterrupt
PrefetchAddr   DCD    PrefetchAbort

```

```
DataAbortAddr    DCD    DataAbort
Nouse            DCD    0
IRQ_Addr         DCD    IRQ_Exception
FIQ_Addr         DCD    FIQ_Handler
```

```
uint8 rtc;
int main(void)
{
    PINSEL0 = 0x00000000;           // 管脚连接设置
    PINSEL1 = 0x55400000;
    IODIR = LED1CON;

    RTCIni();                       // 初始化 RTC
    rtc=0;
    while(1)
    {
        rtc=rtc;
    }

    return(0);
}
```

- A: 1. 要在 VECTOR.S 文件中打开 IRQ 中断。如“MSR CPSR_c, #0x5F”
 2. 使用向量中断时，void __irq IRQ_Exception(void)函数中要设置“VICVectAddr = 0x00”。

第270问:

Q: 这样设置可以产生发送行为,却不能产生中断,为什么不行呢?

```
void __irq IRQ_UART0(void)
{
    ...
}

PINSEL0 &= 0xffffffff;
PINSEL0 |= 0x00000005;

UOFCR = 0x01;
UOIER = 0x03;

UOLCR = 0x80;
tmp = (Fpclk >> 6)/a_mode->ubps;
UODLM = tmp >> 8;
UODLL = tmp & 0xff;
```

```
tc = a_mode->data_bit;
tc |= a_mode->stop_bit;
tc |= a_mode->parity_bit;
UOLCR = tc;
```

```
VICIntSelect = 0x00000000; // 设置所有通道为 IRQ 中断
VICVectCntl0 = 0x26; // UART0 中断通道分配到 IRQ slot 0, 即优先级
最高
VICVectAddr0 = (int)IRQ_UART0; // 设置 UART0 向量地址
VICIntEnable = 0x00000040; // 使能 UART0 中断
A: 要在 VECTOR.S 文件中打开 IRQ 中断。如 “MSR CPSR_c, #0x5F”
```

第271问:

Q: 我用开发板自带的一个程序 Debug 调试, 然后 AXD 提示:

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!

我不知道其中中断向量的地址是如何算出来的, 也就是说程序中的异常向量所指向的位置我不清楚。我买的《ARM 微控制器基础与实战》中提到这儿有一句话解释了下面的程序: 在保留的异常向量位置填数据 0xb9205f80, 是为了使向量表中所有的数据 32 位累加和为 0。(没明白, 0xb9205f80 是如何计算出来的!)

Vectors

```
LDR    PC, ResetAddr
LDR    PC, UndefinedAddr
LDR    PC, SWI_Addr
LDR    PC, PrefetchAddr
LDR    PC, DataAbortAddr
DCD    0xb9205f80
LDR    PC, [PC, #-0xff0]
LDR    PC, FIQ_Addr
```

```
ResetAddr      DCD    Reset
UndefinedAddr  DCD    Undefined
SWI_Addr       DCD    SoftwareInterrupt
PrefetchAddr   DCD    PrefetchAbort
DataAbortAddr  DCD    DataAbort
Nouse          DCD    0
IRQ_Addr       DCD    0
FIQ_Addr       DCD    FIQ_Handler
```

A: 从 mem 窗口看这些地址是什么数 (32 位方式), 然后加起来。

第272问:

Q: 我将光盘里的 EINT1 中的:

```

config, init. s,
LPC2106. h
target. c
target. h
vectors. s
eint1_led. c

```

加入项目中,设置的编译环境为: debug, Release DebugRel 中的 R0 base 为 0x40000100; Rw base 为: 0x40003000; Image entry point 为 0x40000100。编译通过, 调试, f5 出现第一个默认断点, 在全速执行, 程序在 while(1) ; 循环, 按 k1 键 led4 不闪烁, 我想是没有进入中断, 但我不知道这是为什么, 另外我将 r0base 设为 0x40000100 是认为 0x40000000 到 0x40000100 之间要放置 remap 后的中断向量表, 不知道这样做对否。

A: 如果设置 robase 设为 0x40000100, 那么你要把向量表的代码复制到 0x40000000, 然后再 REMAP 操作。另处用我们的 EasyJTAG 暂时不能单步调试中断, 否则会破坏 VIC 的状态。请用先设置断点, 然后全速执行的方式调试。特别注意不能在 0x18 处设置断点 (别的仿真器也不能在这设置断点, 否则破坏 VIC 的状态)。

第273问:

Q: 贵公司网站上提供的可固化的中断程序中, 是将中断向量表先拷贝到 0x40000000, 再 REMAP 到 FLASH 的底部 (我的理解)。我认为既然是固化到 FLASH 里, 把完成上述功能的程序去掉不是也可以吗?

即下面的程序段:

; 实现 REMAP 操作

```

REMAPS  MOV      R2, #0x40000000    ; 复制中断向量代码, 设置目标地址
        LDR      R1, =Vectors      ; 设置源地址
        LDMIA   R1!, {R3-R10}     ; 共复制 16 个字 (64 字节)
        STMIA   R2!, {R3-R10}
        LDMIA   R1!, {R3-R10}
        STMIA   R2!, {R3-R10}

        LDR      R2, =MEMMAP      ; REMAP 操作
        MOV      R1, #0x02
        STR      R1, [R2]

```

A: 可以去掉, 只是这样做可以动态改变向量表。

第274问:

Q: 在《ARM 微控制器基础与实践》中的 157 页, 在 0x0000018 处加 LDR PC, [PC, #-0xFF0] 指令, 为什么就会将 VICVectAddr 寄存器中保存的地址装入 PC?

A: like this:

```

VICVectAddr = 0xFFFFF030
0x00000018+0x08-0xFF0 = 0xFFFFF030
therefore PC = [FFFFFF030]

```


A: 为什么不能进入中断, init. s, target. c 都是贵公司的提供的启动代码!!!
向量的初始化为:

Vectors

```
LDR    PC, ResetAddr
LDR    PC, UndefinedAddr
LDR    PC, SWI_Addr
LDR    PC, PrefetchAddr
LDR    PC, DataAbortAddr
DCD    0xB8A06F58
LDR    PC, [PC, #-0xff0]
LDR    PC, FIQ_Addr
```

```
ResetAddr      DCD    Reset
UndefinedAddr  DCD    Undefined
SWI_Addr       DCD    SoftwareInterrupt
PrefetchAddr   DCD    PrefetchAbort
DataAbortAddr  DCD    DataAbort
Nouse          DCD    0
IRQ_Addr       DCD    0
FIQ_Addr       DCD    FIQ_Handler
```

Undefined

```
B      Undefined
```

SoftwareInterrupt

```
B      SoftwareInterrupt
```

PrefetchAbort

```
B      PrefetchAbort
```

DataAbort

```
B      DataAbort
```

FIQ_Handler

```
STMFD  SP!, {R0-R3, LR}
BL     FIQ_Exception
LDMFD  SP!, {R0-R3, LR}
SUBS   PC, LR, #4
```

主程序如下:

```
void __irq IRQ_Handler(void)
{
}
```

```
void __irq Timer1_Handler(void)
{
}
int main(void)
{
    .
    .
    .
    TOPR = 99;
    TOMCR = 0x03;
    TOMRO = 110592/2;
    TOTCR = 0x03;

    T1PR = 99;
    T1MCR = 0x03;
    T1MRO = 70000;
    T1TCR = 0x03;

    TOTCR = 0x01;
    T1TCR = 0x01;
    VICIntSelect = 0x00000000;
    VICIntEnable = 0x00000030;
    VICVectCnt10 = 0x00000022;
    VICVectAddr0 = (int)IRQ_Handler;
    VICVectCnt11 = 0x0000002f;
    VICVectAddr1 = (int)Timer1_Handler;

    while(1);
}
```

A: the Problem may be the channel selection

VICVectCnt10 = 0x00000022; may be VICVectCnt10 = 0x00000024;

VICVectCnt11 = 0x0000002f; may be VICVectCnt11 = 0x00000025;

第275问:

Q: 断点如果设置在启动时钟节拍之后, 在断点处停顿后在继续运行, 程序运行就异常了, 这是什么原因?

A: 不是不中断, 而是时钟走过头, 还需很长时间才中断。target.c 如下修改即可避免:

TOMCR = 0x01; 改为 TOMCR = 0x03;

函数 Timer0 改为

```
void Timer0(void)
```

```
{
```

```
    TOIR = 0x01;
```

```
    // TOMRO += (Fpclk / OS_TICKS_PER_SEC);    // 删除此句
```

```
VICVectAddr = 0; // 通知中断控制器中断结束
OSTimeTick();
}
```

第276问:

Q: 请教寄存器 VICVectAddr 和 VICVectAddr0~15 的使用区别, 看文档没有明白, 谢谢!

A: VICVectAddr 在发生中断时, 存放有服务程序的地址 (来自 VICDefVectAddr 或 VICVectAddr0~15); 而 VICVectAddr0~15 是存放各个向量中断服务程序地址的寄存器, 当发生向量中断时, 相应的地址会自动装载到 VICVectAddr 中。

第277问:

Q: 在一个程序中同时编入两个中断时, 不知道如何将一个中断的优先级设定高于另一个中断。

例如: 外部中断和定时器的中断。

A: VICVectCnt10~15 优先级递减, 参看实验程序 IRQ2_test。

第278问:

Q: 还有在 AXD 环境下, 单步执行怎么起不了作用, 会跳到 void IRQ_Exception(void) 里面呢?

A: 有中断时不能单步调试, 否则 VIC 工作异常。

第279问:

Q: 我想问向量表累加和是如何计算的?

A: 前 8 个字的 32 位机器码相加。

第280问:

Q: 我把程序下载到 FLASH 中发现 MEMMAP 的值是 01, 可是我把程序下载到 RAM 时发现 MEMMAP 的值是 0; 不过程序都能运行。我这个程序没有中断, 是不是如果我的程序中有中断而且在 RAM 中运行时就必须修改 MEMMAP 同时拷贝向量表到 0x40000000 呢? 如果程序中没有中断并在 RAM 中运行时就可以不管 MEMMAP 的值了阿?

A: 是的。

第281问:

Q: 为什么在 EINT1 中断实验程序中, 中断函数里没有 VICVectAddr=0x00; 是因为它是非向量中断吗?

A: 因为它根本没有使用 VIC 的 VICVectAddr 来判断中断服务程序入口。

第282问:

Q: 当我作向量 IRQ 中断实验时, 将 EINT0 设为最高优先级, EINT1 次之。我在 IRQ_Eint1(void) 函数中加入一条 while(1) 指令, 想作一下优先级差别的实验。我先让 EINT1 中断, 这时进入 IRQ_Eint1(void) 函数中, 并产生死循环, 然后我按下 KEY3 想产生 EINT0 中断, 结果却没反应, 我已经将 EINT0 的优先级设成了最高, 为什么不进入 EINT0 中断函数?

A: 有没有允许中断? 进入中断后处理器自动设置 cpsr 中的中断允许位, 禁止中断。

第283问:

Q: 执行 FIQ 代码时, 操作系统在做什么呢? 当执行 FIQ 中断代码时, 需要注意什么呢? 要对相关寄存器进行入栈吗? 那执行完了以后, 又怎么返回到 OS 里执行任务呢? FIQ 可不可以调用 OS 函数? IRQ 呢?

- A: 1. FIQ 服务程序中断了 OS 啊。
2. FIQ 中断服务不在 OS 管理范围内, 所以不能访问任何与 OS 相关的函数及变量。
3. FIQ 其实就是普通的 FIQ 中断服务函数而已, 与普通的一样处理。
4. 受 OS 管理的 IRQ 可以调用 OS 的系统功能函数。

第284问:

Q: 请问, 在 vectors.s 程序中中断向量操作如下:

```
Vectors
        LDR    PC, ResetAddr
        LDR    PC, UndefinedAddr
        LDR    PC, SWI_Addr
        LDR    PC, PrefetchAddr
        LDR    PC, DataAbortAddr
        DCD    0xb9205f80
        LDR    PC, [PC, #-0xff0]
        LDR    PC, FIQ_Addr
ResetAddr      DCD    Reset
UndefinedAddr  DCD    Undefined
SWI_Addr       DCD    SoftwareInterrupt
PrefetchAddr  DCD    PrefetchAbort
DataAbortAddr DCD    DataAbort
Nouse         DCD    0
IRQ_Addr      DCD    0
FIQ_Addr      DCD    FIQ_Handler
;未定义指令
Undefined
        B     Undefined
;软中断
SoftwareInterrupt
        B     SoftwareInterrupt
PrefetchAbort
```

```

        B        PrefetchAbort
;取数据中止
DataAbort
        B        DataAbort
;快速中断
FIQ_Handler
        STMFD    SP!, {R0-R3, LR}
        BL       FIQ_Exception
        LDMFD    SP!, {R0-R3, LR}
        SUBS     PC, LR, #4
    
```

现我将它改为

```

Vectors
        LDR      PC, =Reset
        LDR      PC, =Undefined
        LDR      PC, =SoftwareInterrupt
        LDR      PC, =PrefetchAbort
        LDR      PC, =DataAbort
        DCD      0xb9205f80
        LDR      PC, [PC, #-0xff0]
        LDR      PC, =FIQ_Handler
;未定义指令
Undefined
        B        Undefined
;软中断
SoftwareInterrupt
        B        SoftwareInterrupt
PrefetchAbort
        B        PrefetchAbort
;取数据中止
DataAbort
        B        DataAbort
;快速中断
FIQ_Handler
        STMFD    SP!, {R0-R3, LR}
        BL       FIQ_Exception
        LDMFD    SP!, {R0-R3, LR}
        SUBS     PC, LR, #4
    
```

程序编译可以通过，用 AXD 调试时报告不能自由运行，可以仿真运行。

按我的理解 DCD 前面为地址（也就是指针），DCD 后面为 4 个字节的内容（指针的值），

```

        LDR      PC, ResetAddr
        LDR      PC, =Reset,
    
```

都是采用间接寻址，也就是将 Reset 的地址放进 PC，为什么“LDR PC, ResetAddr”地址从中间转一道就对了而“LDR PC, =Reset”就不对，既然不对为什么还可以仿真，而

脱机运行就不行？

A: 因为异常向量表累加和不为 0。详细看一看 5.1.3 节或 6.3 节。我记得以前也有一个类似的帖子

可以参考。

第285问:

Q: 是不是 FIQ 与 IRQ 的使用上没有什么区别？

FIQ 自己独有的寄存器有什么用？

我是否可以把 INT1, UART1 等中断设为 FIQ？

FIQ 的中断服务程序应该定位到什么位置？

A: 没有本质的区别。

FIQ 拥有独立的 R8_fiq~R12_fiq, FIQ 中断时, 其它模式相应的寄存器就可不入栈, 从而加快 FIQ 的处理。多个中断源共用还不如单独用 IRQ。因为 FIQ 比较适合在人命关天的情况下用。

第286问:

Q: 这是 VECTORS.s 中的一段, 里面的 REMAP 操作都做了些什么? 好像是保护一类的操作。

REMAP 和 MEMMAP 各是什么含义? 如果我要同时使用两个外部中断, 应当怎样写这一段?

```
REMAPS MOV R2, #0x40000000
```

```
LDR R1, =Vectors
```

```
LDMIA R1!, {R3-R10}
```

```
STMIA R2!, {R3-R10}
```

```
LDMIA R1!, {R3-R10}
```

```
STMIA R2!, {R3-R10}
```

```
LDR R2, =MEMMAP
```

```
MOV R1, #0x02
```

```
STR R1, [R2]
```

```
MSR CPSR_c, #0x1f
```

```
MOV PC, R0
```

A: REMAP 是重映射的意思, MEMMAP 是存储器映射控制寄存器。

MEMMAP = 0b01, 不映射, 程序由 FLASH 开始。

MEMMAP = 0b10, 程序从 RAM 开始, 如果你的程序在 RAM 中, 必须使用这种方式。

如同时使用两个中断, 这一段也不用改。

第287问:

Q: 特定的异常出现时, 处理器进入相应的异常模式。如出现 IRQ 就进入 IRQ 模式, 那出现什么情况进入管理模式呢?

A: 复位或 SWI 中断。

第288问:

Q: 请问什么时候需要通过软件中断进入管理模式?

A: 如果你的程序是在用户模式下运行, 那就可以通过软中断进入特权模式, 修改 CPSR 寄存器。如开关 IRQ 中断(CPSR 的 I 位)。

第289问:

Q: 软件中断是用软件置位的方法产生中断请求, 比如定时器 0 中断请求。用软件中断的方法可以产生 IRQ, FIQ 中断, 对吗?

A: VIC 中的软件中断是这样, 而 swi 产生的异常, 英文也为软件中断, 它是不一样的。

第290问:

Q: 我现在在看 ads 编译器文档中的 swi 部分, 我理解软中断就是一种预期发生的中断而一般中断是不可预期的, 对吗?

A: 可以这样说。

第291问:

Q: 为什么以下的程序进不了中断?

```
/******  
*修改说明 加中断  
*修改目标 我是想每发一个字符就让 LED 变换一次, 现在结果是  
*          仍正常发送字符, 但没有中断效果, 为什么呢?  
*          哪位帮看看! 先谢谢了!  
*****/  
#include "config.h"  
#define HC595_CS 0x00000100  
#define LEDCON 0x00000400 // LED1~LED4 设置为输出  
void DelayNS(uint32 dly)  
{  
    uint32 i;  
    for(; dly > 0; dly--)  
        for(i = 0; i < 50000; i++);  
}  
/******  
*初始化 SPI 接口  
*****/  
void MSpiIni()  
{  
    SPI_SPCCR = 0x52; // 设置 SPI 时钟分频  
    SPI_SPCR = 0xB0; // 配置 MSTR = 1 CPOL = 1 CPHA = 0 LSBF = 0  
}  
  
/******
```

```

*功能 向 SPI 总线发送数据
*****/
uint8 MSendData(uint8 data)
{
    IOCLR = HC595_CS;          // 片选
    SPI_SPDR = data;
    while(0 ==(SPI_SPSR & 0x80)); // 等待 SPIF 置位 即等待发送数据完毕
    IOSET = HC595_CS;
    return(SPI_SPDR);
}
/*****/
*功能 SPI 中断
*修改日期 2003-10-17
*****/
void __irq IRQ_SPIInt(void)
{
    uint32 i ;

    i = VICIRQStatus;        // 读取状态
    i = IOSET;
    i = i & LEDCON;
    if((i & LEDCON) == 0)
    {
        IOSET = LEDCON;
    }
    else
    {
        IOCLR = LEDCON;
    }
    SPI_SPINT = 0x01;        // 清除中断
}

uint8 const DISP_TAB[16] = {0xc0, 0xf9, 0xa4, 0xb0, 0x99, 0x92,
    0x82, 0xf8, 0x80, 0x90, 0x88, 0x83, 0xc6, 0xa1, 0x86, 0x8e};

int main(void)
{
    uint8 rcv_data;
    uint8 i;
    PINSEL0 = 0x00005500;
    PINSEL1 = 0x00000000;
    IODIR = HC595_CS | LEDCON;
    IOSET = LEDCON;          // 让 LED 初始熄灭
    //打开 SPINT 中断 不设置向量控制器 即使用非向量 IRQ

```



```
VICIntSelect = 0x00000000;
VICIntEnable = 0x00000400;           // 使能 SPI 中断

MSpiIni();
while(1)
{
    for(i = 0; i < 16; i++)
    {
        rcv_data = MSendData(DISP_TAB[i]); // 发送数据
        DelayNS(50);
    }
}

return(0);
}
```

- A: 1. 调试时停止程序, 检查 SPINT 中断标志位是否置位;
2. CPSR 寄存器中的 I 位是否为 0;
3. 检查 IRQ 向量入口设置是否正确。

第292问:

Q: 关于“单步调试”的问题。在调试关于 uc0s 移植的例子时, 老是掉到了 IRQ 中断去了, 请问, 怎么样才能够实现比较好的单步调试。

A: 用断点代替。以后升级可能支持直接使用断点。

(注: 新驱动已经解决)

第293问:

Q: 麻烦请教 EasyArm 实验板中断问题

为什么在向量中断后(《ARM 微控制器基础与实战》中例子 IRQ2), 把 VICVectAddr 赋值为 0x00; 别的值可以吗?

0x00 不是 Reset 地址吗? 而且单步跟踪中断程序会跑飞掉! 试了一下感觉 VICVectAddr 赋值为任何值都能跑通。这是什么原因呢? 而且当单步跟踪到“LDR PC, [PC, #-0xff0]”后, 程序就跑飞了!

A: 看一看配套《ARM 微控制器基础与实战》3.5.6 节。在有中断时不能单步调试, 原因可以搜索一下以前的帖子。

“这是因为如果程序在 0x00000018 停下来时 VIC 将状态混乱, 寄存器 VICVectAddr 的值不再是需要的值而是寄存器 VICDefVectAddr 的值。而寄存器 VICDefVectAddr 默认的值为 0。”

第294问:

Q: [请教] 《ARM 微控制器基础与实战》程序 5.19 关于 IRQ_Eint0 和 IRQ_Eint1 的问题
在 EINT01.C 定义了两个中断服务程序, IRQ_Eint0() 和 IRQ_Eint1(), 可是在 Vectors.s

中只声明了 `IMPORT IRQ_Eint1`, 好像 `IRQ_Eint0` 没用到, 另外 `IRQ_Addr DCD 0`, 怎么感觉 `IRQ_Eint1` 都没用到? 还是由于在 `IRQ_Eint0()` 和 `IRQ_Eint1()` 有了 `VICVectAddr=0x00`, 就可以这样写了?

A: 因为都是向量中断, 所以一旦产生中断程序就会从向量表里的地址减去 `0xff0` 就到 `VICVectAddr` 去了。

第295问:

Q: 使用的是一块开发板, 无外挂存储器件。

1. 使用其光盘上的程序, `entry` 为 `0x4000 0000`, `R0` `0x4000 0000`, `RW` `0x4000 3000`, 这种情况下使用 JTAG 口可以进行调试。
2. 后来想把程序下载到 flash 里面, 于是把上述 3 个地址改为 `0x00000000` `0x00000000` `0x00003000`, 使用生成的 bin 文件下载(也不知道下载成功了没), 然后 reset, 板子怎么也跑不起来, 又改回到 A 状态, 跟踪 PC, 发现一开始就到一个未知位置, 所有寄存器都是 `ffffff`。
3. 将 1 中的地址使用重新 make 了一下, 发现再下载提示: “interrupt vector” 不对。板子怎么都起不来。

我想如果不是地址配错了, 是不是芯片坏了啊?

A: 参考《ARM 微控制器基础与实战》图 7.11 设置, 用 flash 下载时注意中断向量表不能错, 见《ARM 微控制器基础与实战》附录第一个问题的回答。

第296问:

Q: 中断处理:

串口和中断的寄存器太多了, 我用中断接受数据然后发送, 进入中断该读那些串口和中断 (IRQ) 寄存器来关中断和开中断, 才能正确返回中断响应下一次的中断?

A: 数据手册的表 74 有说明, 接收数据中断需要读取 `UORBR`, 以清除相应的中断标志。

<http://www.zlgmcu.com/download/downs.asp?ID=743>

中断返回前需要对 `VICVectAddr` 写 0。

第297问:

Q: 麻烦谁回答一下关于中断的问题。

我是用串口 0 中断方式来接受数据的, 可是进入中断后, 能接受到第一个字节, 以后发送的字节都没有收到, 是不是第一次接收完退出中断时没有开中断? 我设置 `VICVectAddr` 为 00, 这算不算是开中断?

A: `VICVectAddr=0` 只是清除 VIC 的中断逻辑。中断服务函数要用 `__IRQ` 修饰, 以便中断服务能正确返回。可参考外部中断的例子来编程。

第298问:

Q: 《ARM 微控制器基础与实战》390 页上的一句话, 倒数第二行 “在调试时一个中断还没有响应就再次载入程序时, 向量中断控制器会出现状态错误...”。

请问：什么情况下会出现“在调试时一个中断还没有响应”，“就再次载入程序”？

A: 您先前调试的程序可能已经暂停（通过断点等），可能硬件还会产生中断，此时您再次调入调试文件可能产生问题。

第299问:

Q: 关于串口中断处理

系统的两个串口都用到拉，采用 UCOS 周工提供的驱动。串口 0 接 GPRS，串口 1 接 485 的设备。要想两个串口的数据都能不丢失，程序应如何处理？

UCOS 的串口驱动中的中断函数能写多少代码。我想将一些数据接受、判断放在中断中处理如何？

Q: 给两个串口再分配一个大的 FIFO，提高速率。采用邮箱管理。

第300问:

Q: 关于 LPC210X 中断向量的问题

1. VICVectAddrx 寄存器与 VICVectCntlx 寄存器是否一一对应？还是无关？如：VICVectAddr0 与 VICVectCntl0 是否有直接关系？

2. 各个外设的 VICVectCntl 值是否固定？

如：VICIntCntlx=0x29：是否只是分配 IIC 通道到 IRQ slotx？

VICIntCntly=0x26：是否只是分配 UART0 通道到 IRQ sloty？

若有固定的对应关系，请问各外设的通道值如何得到呢？

3. 各个外设的中断使能值是否固定？

如：VICIntEnable=0x0200：是否只是使能 IIC 中断呢？

VICIntEnable=0x0040：是否只是使能 UART0 的中断呢？

若有固定使能值，请问各外设的中断使能值如何获取呢？

4. VICVectAddr 寄存器是否只是为了中断正确返回而设？就是在每个中断调用的服务程序结束都应该读取 VICVectAddr 寄存器以便正确返回？它的存在还有没有其他特别的作用？

A: 1. VICVectAddrx 寄存器与 VICVectCntlx 寄存器是一一对应的，他们对应着同一个通道 slotx 的地址和控制字。

2. 关于各个外设的对应的中断通道，看一下 210x 的 datasheet，上面有详细的列表说明。

3. VICVectAddr 寄存器可以自动装取最高优先级的响亮中断入口，没有向量中断就装载 VICDefVectAddr（里面存放着非向量中断的入口）。这些在 datasheet 中都有详细说明，建议看一下。

第301问:

Q: 我在用 2104 开发板调试中断的时候，将 3 个外部中断分别设置为 FIQ，非向量 IRQ 和向量 IRQ 模式，用 ESAYJTAG 调试的时候有时成功，但有时会运行几次就停止响应。而且在单步的时候发现每次发生向量 IRQ 中断的时候，程序总是跳到非向量中断中运行，请问这是怎么回事？

A: 1. 可能是中断处理不当。是否将 VICVectAddr 设置为 0？

2. 不能单步运行读取 VICVectAddr 寄存器, 否则读取到的值为非向量中断的地址值。

Q: 谢谢, 不过我的 VICVectAddr 在中断处理后已经设置为 0 了。我在单步调试的时候 VICVectAddr

寄存器老是装非向量中断的入口, 这是为什么呀?

A: 这应该是 VIC 的问题, 不能在 LDR PC, [PC, #-0xff0] (读 VICVectAddr) 时停下来。

第302问:

Q: 2119CAN 中断进不去, 程序是照着 LPCARM-CAN 用户指南写的 CAN1 中断通道号用 25、26; CAN2 用 27、28。

A: philips 的文档搞错了, 关于 LPC2119 ARM 的 CAN 中断通道号, 已有更新的资料, 见数据手册(中)第 60 页。

第303问:

Q: 如何在 FLASH 中捕捉中断?

外部中断试验程序, 在 RAM 中一切正常, 我想让程序运行在 flash 中, 我将 REMAP 一段去掉, 又将设置中断向量表使其和设为 0, RO_BASE: 0x00000000; point: 0x00000000; 为什么还是不能进入中断处理程序, 而 LED4 复位后一直都是亮的? 如何在 FLASH 中捕捉中断?

A: 可下载到 FLASH 运行的 EINT1 例子: <http://www.zlgmcu.com/download/downs.asp?ID=944> 在中断处理程序入口设置断点。

第304问:

Q: 2104 怎样软件复位?

我在程序里调用了函数 Reset(), 发现程序好像是回到了 init.s 的 Reset 的位置, 但停止运行了。

请问怎样正确地使用软件中断。

A: 我是这样做的: Reset() 在一个外部中断 eint0 的中断服务程序里, 我希望当我按下 key1 时系统复位重新开始运行。我试验的时候已经把程序下到 flash 里了。

A: 可以打开 WDT 功能, 使芯片真正复位。

Q: 我的意思不是要用看门狗, 就是要在中断复位程序里给 2104 软复位。

实际上是要在断点返回的时候把指针改为 0x00。但是我想直接调用 Reset() 函数, 但试了试好像不行, 应该是我不会用吧。有人知道怎么操作么?

A: 在中断使能 WDT, 并设置时间为最短, 应该比较好的方法。

A: 用程序跳转实现的话, 可能先要关中断, 复位 VIC 中断逻辑.....

第305问:

Q: 有关向量中断的问题!

我的程序有两个向量中断, 一个是串口中断, 一个是定时器 0 中断。串口的优先级高于定时器的。程序执行之后, 刚开始正常, 没一会就不能响应串口中断了。为什么?

A: 1. 中断处理完后要 VICVectAddr = 0x00;

2. 中断处理中要将中断标志清零。如串口接收中断中，可能是接收数据可用中断，或字符接收超时中断。

第306问:

Q: RS485 多点通讯。

LPC2000ARM 系列实现 RS485 多点通讯时没有硬件地址自动识别，如果定位在工业仪表及控制等 高档单片机领域单 CAN 是不够的，毕竟 RS485 现在应用太广泛了，各种 PLC、工控机等。系统工程集成商是不会像高技术开发的那样追求新技术，适用方便才是真理，所以走到 CAN 等现场总线还需要一个过程。建议菲利普在此单片机系列加此功能，不要说软件可以实现地址识别，也不要谈加加外围芯片，要不要谈什么竞争力。

A: LPC 系列 ARM 芯片内置的 UART (16C550) 功能很强。很多采用 51 的仪表也是软件识别地址，因为要和 PC 机和其他设备（如：PLC）通讯。PC、PLC 用第 9 位很不方便。

三菱、西门子等的 PLC 基本上不支持第 9 位。

另外，很多变频器厂家都支持的 USS 协议也不支持第 9 位。

ARM 本身有两个串行口，很容易变成两个 485 口

第307问:

Q: 配套光盘上的串口收发程序（就是发 8 个字节的数据，然后单片机自动返回发送的 8 个数据），开始调试时很正常，但是过了几天发现收发数据不一样了，而且返回的数据还回掉帧，很是郁闷。重启了电脑再试有可以了。但是今天用着用着的收发数据又不一样了（程序没有改动，一直用的是光盘上的原版）。后来发现把串口（计算机上的）波特率由 9600 提高到 38400 后居然收发数据恢复了正常。我用的串口调试软件是 51bbs 上提供的，后来用了很多种串口调试软件都是一样的结果。望大虾们解答？

A: 应该是波特率的问题，先将 LPC2104 片内 FLASH 擦除掉（使用 ISP 软件擦除）。

第308问:

S: EasyARM2104 学习板串口波特率不对的解决方法：

如果用户的 LPC2104 中有程序，则可能会使部分试验代码的时钟计算错误（可能导致时序过快）。主要是由于 flash 中的程序改变系统时钟和外设时钟，而受影响的程序没有初始化系统时钟和外设时钟，使用硬件复位默认值计算引起。

例如，受影响的代码中涉及到串口的代码可能会使波特率计算错误。

受影响的代码有：

\software 下的所有代码

\实验程序\Wdt_test

\实验程序\Rtc_test

\实验程序\I2c_test

\实验程序\UART_TEST

解决方法：

1、删除 Flash 中代码。

2、以\software\disptimer 的代码为例说明:

1) sendrtc.c 中第 12 行

```
#define FPCLK          2764800L
```

改为

```
#define FPCLK Fpclk
```

2) 取消 target.c 中 127 行到 159 行被注释了的代码。

这种方法可能对有些代码无效

3、自己初始化 PLL 和 VPB 分频器到复位默认值, 代码如下

```
VPBDIV = 0;
```

```
PLLCON = 0;
```

```
PLLFEED = 0xaa;
```

```
PLLFEED = 0x55;
```

注意这一段代码应该在一进入主程序就调用。

第309问:

Q: 我有些糊涂: 进入中断时, 在 PC 跳到 VectAddr 前, 有没有入栈操作?

如果有肯定是保存到了中断模式堆栈中了, 什么时候出栈哪?

A: 没有, 具体请看 2104 配套第一章, 或 ARM 体系结构。

第310问:

Q: 向量中断可以嵌套么?

lpc2104 的片子, 向量中断似乎不能嵌套。

我用 eint0, eint1 做实验, 当一个中断不断发生时, 另外一个(优先级高)不能发生。

A: 参考例子: <http://www.zlgmcu.com/download/downs.asp?ID=1114>。

第311问:

Q: EXTINT 的用法

资料上说 lpc2104 的外部中断寄存器 extint 的低三位与 eint2~0 对应, 为 1 时表示外部中断有效, 可以向其写 1 清零标志位。

这句话是否指一旦软件向其写 1, 则相应标志位会变为 0?

《ARM 微控制器基础与实战》上有如下一段程序:

```
while((EXTINT&0x01)!=0)
{
    EXTINT=0x01;
}
```

这句话是否指如果 eint0 有效, 则清标志位, 使其无效?

A: 在外部中断无效的情况下, 写入 1 可以清中断标识位, 如果外部中断一直有效, 则写入

1 不能清除中断标志。

第312问:

Q: 结束 ISR 时, VICVectAddr 寄存器必须写 0 吗? 写其它的值可以吗?

A: 可以。

第7章 串口

第313问:

Q: 我做了一块 arm 的应用控制板, 可是, 串口 0 用查询的方式能收到数据, 而用中断方式不可以 (我的串口程序是对的, 在 arm2104 开发板上测试通过), 我做的板子在刚开始时好使, 串口 0 可以接受中断, 可用了一段时间, 就出现上述情况, 我已经换了几块单片机了, 可每次都是用一段时间就不行了, 别的控制都没问题。希望斑竹给提个建议, 是我的片子坏了么?

A: 检查 ISP 引脚有没有上拉!

第314问:

Q: 我在调试 easyarm2104 时遇到了下列问题:

我用 ads1.2 编译并运行基础实验里的 UART 实验, 然后打开 EASYARM 软件, 结果没有显示, 但我用另外一个串口调试助手, 能收到开发板发过来的字符 “HELLO WORLD!” 这是怎么回事? 谢谢!

A: 波特率设置要正确, 打开串口发送/接收窗口观察。可看一看置顶的帖子 “EasyARM2104 学习板串口波特率不对的解决方法”。

第315问:

Q: 以下是我的一些设置:

```
U1FCR = 0x05;           // 使能 FIFO, 并设置触发点为 1 字节
U1IER = 0x03;           // 允许 RBR、THRE 中断
```

```
/* 设置中断允许 */
```

```
VICIntSelect = 0x00000000; // 设置所有通道为 IRQ 中断
VICVectCntl1 = 0x27;       // UART1 中断通道分配到 IRQ slot 0
VICVectAddr1 = (int)IRQ_UART1; // 设置 UART1 向量地址
VICIntEnable = 0x00000080; // 使能 UART1 中断
```

我用查询方式发送和中断方式接受能正常工作, 但是我把发送也用中断方式时, 发送中断却没有产生。看资料说明似乎还要加一些初始化条件, 但不是很明白, 请各位大侠指点一下还要加些什么?

A: 我终于知道原因了, 原来在触发发送中断之前要先发送字符触发发送中断。

第316问:

Q: 请问: 使用 UART 的 FIFO 时, 如果是接收超时中断, 如何知道 FIFO 中接受到多少字符?

A: 读取 UOLSR 寄存器的 RBR 位来判断。

第317问:

Q: EasyARM 软件窗口在使用时, 只发现窗口下面的 ‘接收字节数’ 是变化的, 而 LED 和万

年历、键盘都不起作用。运行程序为光盘中 DISPTIMER1 目录下的例子。请问这是什么原因？

A: 应该是波特率不对。请先将芯片内部 FLASH 的程序擦除掉(使用 ISP 软件操作)。

第318问:

Q: 在 UART0 中的 UOFCR 的第三位 (DMA 模式选择) 所说的单传送模式和多传送模式是什么意思？

A: 由于 cpu 不支持 DMA, 所以不要使用该方式。可以参考 16550 标准的手册。

第319问:

Q: 我的程序如下, 当我测试的时候, 每一秒发送一串字符, 大多数时候都能正确返回, 可以有时候会丢失几个字符串, 比如送出 1234, 返回的是 124, 为什么?

```
// 串口中断函数
void __irq IRQ_UART0(void)
{
    if( 0x04==(U0IIR&0x0F) )
        rcv_new = 1;
    rcv_buf[rcv_ptr] = UORBR;
    rcv_ptr = (rcv_ptr+1) & MAX_PTR_C;
    VICVectAddr = 0x00;
}

// 发送字符函数
void ISendBuf(void)
{
    uint8 i;

    for(i=0; i<rcv_ptr; i++)
        SendByte(rcv_buf[i]);
    while( (UOLSR&0x20)==0 );
}

// 主函数
if(1==rcv_new)
{
    ISendBuf();
    rcv_new = 0;
    rcv_ptr = 0;
}
```

A: rcv_ptr 在两个程序中都被改变, 中断干扰了发送。

第320问:

Q: 《ARM 微控制器基础与实战》表 3.65 上说: 当 UORBR 已经有新的字符就绪而 UORBR FIFO 已满时, UOLSR1 置位, 此时 UORBRFIFO 不会被覆盖, UORSR 中的字符将丢失。那我如果通过 UART 接收了一个数据后, 转而去执行一段很长的程序, 然后再通过 UART 连续接收数据时, 接收到的是紧接着上一个数据被发送过来的连续数据吗? 还是中间的某个数据可能已经丢失? 我的程序如下, 想接收所有数据 (一个都不能丢失), 帮我看看程序对吗?

```

UOFCR = 0x01;

while(1)
{
    while((UOLSR && 0x01)!=0x01);
    data = UORBR;                                // 读取 FIFO 的数据

    if(data == 1)
    {
        for(i=0; i<5; i++)
        {
            while((UOLSR && 0x01)!=0x01);
            xy[i] = UORBR;                        // 读取 FIFO 的数据
        }

        ..... //在这里要执行很长一段代码
    }
}

```

A: 肯定是紧接着你刚才的数据啊, 如果你接收前面一个字符的时候 FIFO 还没满的话。

第321问:

Q: 在 DATARET.C 实验中, 为什么不能发送接收 1 个或 9 个数?

1. 我把程序稍微改动一下, 想发送接收 1 个数, 但没有反应, 还是必须输入 8 个数, 这是为什么?

```

.....
uint8 rcv_buf[1];                                // UART0 数据接收缓冲区
.....
void __irq IRQ_UART0(void)
{
    .....
    for(i=0; i<1; i++)
    { rcv_buf[i] = UORBR;                        // 读取 FIFO 的数据, 并清除中断标志
    }

    VICVectAddr = 0x00;                          // 中断处理结束
}

```

```
.....  
void ISendBuf(void)  
{ uint8 i;  
  
    for(i=0; i<1; i++) SendByte(rcv_buf[i]);  
    while( (UOLSR&0x20)==0 );           // 等待数据发送  
}  
.....
```

2. 我把上面程序中的 1 改为 9 想发送接收 9 个数，输入 31 32 33 34 35 36 37 38 39 后，按发送，结果却是 31 32 33 34 35 36 37 38 38，怎么会把 39 错发成 38 呢？

A: 1. 设置 UOFCR 寄存器为 0x01。

2. FIFO 控制器触发点只有 1、4、8、14 字节，没有 9 字节(参考 FIFO 控制寄存器 UOFCR 说明)。

第322问:

Q: LPC2104 开发板的串口 SP3238 接 JP5 的信号，除接受和发送外，与 LPC2104 的连接均不对，均是输出接输出，输入接输入，使得此串口既不是 DTE 接口，又不是 DCE 接口，而且容易烧坏芯片，现在我开发的设备连接 MODEM，但 MODEM 信号都不对，给我的开发工作造成了很大困难，不知应如何解决？

A: 将 JP5 全跳开，从那引信号到自己的 max3243 上。

第323问:

Q: 请问：我在程序中可以动态改变 FIFO 的触发深度吗？就是改变 UOFCR 的 6~7 位。

A: 具体行不行没有试验过，不过没有必要动态设置，因为还有字符间隔中断。

第324问:

Q: 请教 LPC21xx 片子的 uart1 串口如果不用它的 modem 功能，只是用它的普通串口，其接线是否和 UART0 完全一样？

A: 完全兼容。

第325问:

Q: 做 UART-TEST/SENDSTR 实验时，现象都正常，但为何看不到寄存器 UOTHR 中数据随着发送字符的不同而改变呢???!!!!!

A: 这个寄存器不可读。

第326问:

Q: UART 接口波特率问题:

在调试有关 easyarm.exe 的实验，如果程序中 UART0 的初始化为 9600，发现 easyarm 在

9600*4 的波特率下才能正常接收。为什么？

我使用的是 lpc2104，参考《ARM 微控制器基础与实战》中的 easyarm 应用例程就有上述现象发生。

A: 要先将片内 FLASH 的程序擦除，因为它会设置 PLL，而调试程序的波特率设置值是按无 PLL 的系统时钟计算的。

第327问:

Q: 看到串口 0 中断处，好像要使能 FIFO 须将 U0FCR[0]置位，而串口只有 FIFO16 字节收发这种工作方式，不知我这样理解对不对？

看了示例 1 中的 target.c，发现对于 U0FCR 的处理是 U0FCR=0X00，好像除此以外并无对其的操作，可是程序是正常的，这是为何呢？是不是不用设置 FIFO 串口依然可以工作（像 51 中用一个寄存器中存放收到的值）？

A: 1. 是的。Rx FIFO 可以设置(中断)触发点。
2. 对。使用 UART 时，FIFO 不是必须的。

第328问:

Q: 请教关于 2104 波特率的于心问题

1. 现在看到《ARM 微控制器基础与实战》上的的串行通信的波特率都是 1200, 2400, 4800, 9600 等，是否波特率只能是这几个呢？能不能自己设定波特率为 200, 4000, 10000bps 等？
2. lpc2104 中，如果 Fpclk=10MHz，除数锁存默认为 1，那么计算出的波特率为： $10\text{MHz}/16=625000\text{bps}$ ，这么大的波特率或者更大的波特率，芯片是否支持呢？我们在实际中能不能用更大些的波特率呢？

A: 1. 当然可以，设置好 UxDLL、UxDLM 寄存器即可。
2. 自己测试一下。从芯片数据手册可知，IO 的上升沿/下降沿的时间典型值为 10ns。

第329问:

Q: 为何串口有些寄存器都是映射在同一地址上？

A: 根据读写属性来区分的。

第330问:

Q: 请教 max3238 的 T1IN~T5IN 要不要加下拉电阻？

我看文档好像不用加，但是 2104 板子上加了 10K 的下拉，到底应不应该加呢？

A: 连接 MODEM，使用 SP3243E。

<http://www.zlgmcu.com/Sipex/RS232.asp>。

第331问:

Q: 请问，发送器空 (TEMT) 与发送保持寄存器空 (THRE) 这两位有什么区别？

.. \Arm_pc\PC.C 中第 76 行语句

```
while((UOLSR & 0x00000020)==0);
```

A: 表示如果线状态寄存器 UOLSR 的发送保持寄存器空位 (THRE) 为 1 时就执行它下面的语句。

Q: 为什么不是发送器空位 (TEMT) 为 1 时执行它下面的语句呢?

《ARM 微控制器基础与实战》上利用查询方式发送数据时通过判断 TEMT 位是否为 1 来决定数据是否已经发送出去? 这两位有什么区别?

A: 发送的时机不一样而已。

第332问:

Q: UART 溢出错误。

所有《ARM 微控制器基础与实战》上的例子未提及 UART 的错误处理。我发现当发生溢出错误时, 可以通过读 RBR 和 LSR 将溢出错误去掉, 但是再次接收到字符如果读 LSR 仍然有溢出错误。请问, 问题可能出现在哪里? 谢谢!

A: 可以参考以下 uc0sii 中的 UART0 的例子。

第333问:

Q: 关于使用+5V 的 MAX202。

请问, 我现在使用 2114, 但是我希望使用 5V 供电的 Max202, 从 Datasheet 上看到 I/O 可以承受+5V, 不知道我直接将 RXD\TXD 连接到 2114 上是否可以正常工作。

A: 可在其间串一个小电阻。

第334问:

Q: 请教: 为何 2104 的 UART 不能向上位机发送二进制的 00H?

在调试中发现, LPC2104 的 UART0 不能向上位机 (PC 机) 发送二进制的 00H, 但是从 01H~0FFH 都能成功发送到上位机。另外能从上位机发送 00H 到下位机。请问这是什么原因, 如何解决?

A: 可以, 我们测试过 N 次。这一般是上位机过虑了 00H 这个字符。

第335问:

Q: 问一下有关 2119 开发板 UART0 的一个问题。

简而言之, 设置了串口中断触发字节数为 8 个。发送的时候采用的是直接写 U0THR, 如果我要发送 16 字节的数据, 在串口调试助手上显示的是前 8 个字节, 然后程序就跳飞了。或者我第一次发小于 16 个字节的数, 比如发了 15 个字节, 他会显示前 8 字节, 等我再发一个字节, 他会把先前的 7 个字节加上这个字节一起发送出来。这是什么原因?

还有, 一般我通过串口向 2119 发送数据, 不能确定多少字节数, 如果设定为 1 个字节触发是不是可行? 需要注意什么吗? 小弟在此多谢高手回复了。

A: 可能是中断处理程序的问题。可以直接使用 examples\UART\C\DataRet\DataRet.mcp 工程。

可以设置为接收 1 字节触发, 中断处理程序要简单点。

Q: 谢谢回答。不过那个 C 的源码是循环 8 次读接收缓存器。我想问一下, 《ARM 微控制器基

础与实战》上说，如果设置触发 10 字节，现在有 105 字节数据要发，他会响应 10 个 RDA 中断，然后响应 5 个 CTI 中断。请问，这个 CTI 中断如何使用？也需要中断的入口地址吗？

- A: 1. 中断程序读取一字节数据即可。
2. 这只是一个举例说明，串口会响应 10 个 RDA 中断(10*10=100 字节数据)，剩下的 5 个数据将会使串口产生 CTI 中断（因为没有到达触发点）。真正的 UART0/1 没有 10 字节的触发点。
3. CTI、RDA、RLS、THRE 中断均是同一个中断通道，可以说是同一个入口地址。在中断程序中通过读 UOIIR 或 UIIIR 判断是什么中断。

第336问:

Q: 请问 FIFO 是什么意思？

A: First-in First-out, 先进先出。

Q: 那在串行通讯中 FIFO 是什么意思呀？什么叫 FIFO 使能？

A: 串行芯片一般有几个 FIFO 寄存器，你可以理解为缓存。可以用，可以不用。想用就打开使能。

第337问:

Q: 怎样检测串口发送数据？

比如说在 UART0 实验中发送的数据是字符串“Hello World!”那么怎样才能显示呢？

我自己的想法是“把串口线接上，打开下载的那个 EasyARM-C.exe 并且设置好，运行程序就可以显示了”。可是却不对，只能观察到向串口发送的数据量，其他的什么都没有了，还有那个万年历也不显示啊！哪位能告诉我 EasyARM-C.exe 是干什么用的吗？谢谢了……

A: 使用其它串口监视软件。EasyARM-C.exe 是有协议的，看配套《ARM 微控制器基础与实战》5.3 节。

第8章 I2C

第338问:

Q: I2C 存储器 cat24wc02 的写操作寻址问题。

通过你们给的程序可以看出: cat24wc02 在写操作时, 可以有子地址, 也可以没有子地址, 那么器件时怎么区分第一个发送的数据是子地址还是第一个数据?

A: 可以看看 CAT24WC02 的资料, 在对其写操作时, 从地址之后的第一个字节数据为器件子地址 (即存储地址)。

总线操作: 起动 + 从地址(如 0xA0) + 存储地址 suba + 数据 + 结束。

第339问:

Q: 你们的板子附带的程序中关于 I2C 的中断程序中有这么一段:

请问是不是子地址可以发送也可以不发送啊? 发送的话他怎么区分你发送的是子地址还是数据啊? 谢谢

```
case 0x18:                // 已发送 SLA+W, 并已接收应答
    if(0==I2C_suba_en)    // 无子地址, 则直接发送数据
    {
        if(I2C_num>0)
        {
            I2DAT = *I2C_buf++;
            I2CONCLR = 0x28;
            I2C_num--;
        }
        else
        {
            I2CONSET = 0x10; // 无数据发送, 结束总线
            I2CONCLR = 0x28;
            I2C_end = 1;     // 设置总线操作结束标志
        }
        break;
    }

    if(1==I2C_suba_en)    // 发送子地址
    {
        I2DAT = I2C_suba;
        I2CONCLR = 0x28;
    }

    if(2==I2C_suba_en)
    {
        I2DAT = I2C_suba;
        I2CONCLR = 0x28;
```

```
I2C_suba_en = 0;    // 子地址已处理
}
break;
```

A: 这与所使用的 I2C 器件有关, 如 24wc02 是有子地址的, 即存储地址; 如 PCF8574 是没有子地址的。指定子地址读操作时, 写完子地址后要重新启动总线。指定子地址写, 实际上写的第一个数据就是子地址 (发送从地址后的第一个字节数据)。详细参考 I2C 的资料或书籍。

第340问:

Q: 各位在使用 I2C 的时候有没有发现 I2CONSET 和 I2CONCLR 总是同步变化? 而且对 SI 的清零总好象没有执行一样。

A: 是的, 在单步调试的时候, SI 是不会清零的。

第341问:

Q: 这个判断语句中的两者有何区别? 发的数据似乎一样的吧?

```
case 0x08:    // 已发送起始条件
    if(1==I2C_suba_en) I2DAT = I2C_sla & 0xFE;
    else I2DAT=I2C_sla;
    I2CONCLR = 0x28;
    break;
```

A: 因为指定地址读操作时 (调用 IRcvStr 函数), 设置 I2C_sla 的值为读操作的地址, 即 d0 位为 1, 所以要 I2DAT = I2C_sla & 0xFE。这样就可以设置子地址, 然后重新启动总线后进行读操作。

第342问:

Q: 在 WREEPROM.S 中, 从 CAT24WC02 读出数据, 为什么发送的器件地址还是#CAT24WC02 (#0xA0)?

在芯片资料中不是说, 当读芯片时 R/W 位应置 1 吗? 那么地址应该是#0xA1 才对啊, 是不是?

A: 读子程序内有加 1 操作。

第343问:

Q: 我想用查询的方式使用 IIC 总线, 但总是不大成功, 写从没成功过, 读倒有时候还能读出, 难道不能查询 SI 方式实现 IIC, 一定要用中断? 还是, 我有些地方会弄错概念了, 希望大家指点一下, 特别需要注意什么?

- A: 1. 总线速度不要过高, 可使用 100KHz。
2. 注意 24 系列 E2PROM 的写周期和页翻转问题。
3. RCVBYTE_A 子程序已勘误

<http://www.zlgmcu.com/tools/kaifaban/EasyARM2104/ARMbook-error.pdf>

第344问:

Q: 7.2 节原文: 当从机不能响应从机地址时, 例如它正在执行一些实时函数不能接收或发送, 从机必须使数据线保持高电平。主机然后产生一个停止条件终止传输或者产生重复起始条件开始新的传输。

问题: 产生停止条件应该是把 SDA 线从低电平到高电平切换。从机使数据线保持高电平, 那么主机如何产生停止条件呢?

A: 你首先要了解什么是“线与”逻辑。

第345问:

Q: 在程序清单 3.23 中,

```
MOV R1, #14
STR RI, [R0, #0X10] ; I2SCLH=14
STR R1, [R0, #0X14] ; I2SCLL=14
```

这样设置后 I2C 的时钟频率是多少, 如何对着表 3.91、3.92 查找?

A: 直接按计算公式计算, 如 $F_{pclk} = 11.0592\text{MHz}/4 = 2.7648\text{MHz}$,
则位频率 = $2.7648 * 1000000 / (14 + 14) = 98742$, 约 100KHz。

第346问:

Q: 关于 EEPROM 芯片 24C01 的 I2C 驱动的问题:

当数据小于 16 个时, 写入和读出的数据一样, 没有出错。

当数据大于 16 个时, 写入和读出的数据不一样。

A: 超过 16 个字节, 写入的字节就会冲掉前面的字节, IIC 操作分页写和字节写, 如果用页写, 就存在每一个页的字节数问题。

Q: 那就是说, 超过 16 的页地址要相应地增加?

比如写第 17 开始要在页地址设 1, 是否这样? 谢谢!

A: 对, 你如果采用页写的方式, 那么你要注意超过一页的字节数后, 就应把多余的字节写入下一页中, 写多余的字节前应给出下一页的页地址。如果采用单字节写就可以连续写入, 只要不超过器件容量。

第347问:

Q: 请教 I2C 总线的一个问题, 《基础与实战》上关于 I2C 总线的中断方式有些地方不明白, P328 页, 下往上数第 3 行,

```
case 0x08:
if(1==I2C_suba_en)
I2DAT = I2C_s1a&0xFE; //指定子地址读时,先写入地址
```

请问: 为什么在读从器件时, 要将 I2DAT 的最后一位清 0, 其中是 I2C_s1a 是从器件地址, 如果是为了获取 AA 应答, 该清的是 I2CONSET, 使用 I2CONCLR 清, 为何要做“与”操作?

A: 注释已说明了, 指定子地址读, 需要先写入子地址, 即:

起动总线->从地址(写)->子地址->重新起动总线->从地址(读)->接收数据... ->结束总

线。

第348问:

Q: 求助 I2C 状态表

A: <http://www.zlgmcu.com/download/downs.asp?ID=570>

第9章 SPI

第349问:

Q: 我现在将 ARM7 开发板通过 SPI 与板外的 A/D 通讯, 但只能发送, 接收不正常, 偶尔能得到一个正确数据。SPI 的接收和发送都检测 SPIF 标志吗? 发送后是不是需要专门清除 SPIF?

另外, 请问有没有 SPI 的更详细的资料。

A: 是要每次 SPIF 置位后都要读取 SPSR 寄存器, 清零该位, 然后才能访问 SPI 数据寄存器。

第350问:

Q: 今天用 LPC2104 驱动一个 SPI 接口的 LCD, LCD 的控制器是 SAMSUNG 的 KS0713 看它的手册说是串行数据在时钟的上升延驱动数据。可是在用 LPC2104 驱动它的时候发现 CPOL=0, CPHA=0; CPOL=1, CPHA=0; CPOL=1, CPHA=1 都能驱动 LCD, 只有 CPOL=0, CPHA=1 不能驱动。可是给我的感觉这个配置是最合理的阿! 因为在这个配置《ARM 微控制器基础与实战》上说第一个数据和其他数据都是上升延驱动阿! 请大家指教。

还有就是我没有看明白 CPOL 和 CPHA 到底是什么作用, 书上关于它们的时序图我也没看明白, 斑竹能不能给解释一下啊? 谢谢!

A: 我看 KS0713 资料后认为 CPOL=1, CPHA=1 最合适 CPOL 和 CPHA 的作用就是《ARM 微控制器基础与实战》上说的。不过, 就像我先前说的, 在 lpc2104 驱动数据的边沿, KS0713 正好不能采样数据, 应该在相反的边沿采样数据。

第351问:

Q: 使用 SPI 要用四个管脚: SCK、SSEL、MISO、MOSI, 可是给的原理图上的 SPI 用了 SCK、MISO、MOSI 和 TxD1, 而 SSEL 用在了蜂鸣电路, 是笔误还是什么原因? 急!

A: SSEL 是从机模式下的选择线。在主机模式下时, SSEL 不输出片选信号, 且不能为低电平, 所以只能使用其它口线 (P0. 8) 输出片选。

第352问:

Q: 2119 的 SPI 为什么没有输出

```
PINSEL0 = 0x00005500; // 设置 SPI 管脚连接
```

用的是 2104 的程序, 串口输出正常, 将引脚作为 PIO 输出正常。

A: 注意做主控时 SSEL 不能变为低电平, 即它不能再做 GPIO 使用。

第353问:

Q: SPI 通信应用问题求教

在与 ADE7754 进行 SPI 通信时采用硬件 SPI, 不知道是什么原因通信不正常, 方法如下: LPC2104(3.3V)的 SPI 输出线与 ADE7754(5V)的信号线串接了一个 30 欧的电阻后直接连接。

A: 使用硬件 SPI 时, SSEL 管脚接一上拉电阻, 保证为 SPI 主机。管脚配置如:

```
PINSEL0 = 0x00005500;    // 设置 SPI 管脚连接
```

第354问:

Q: SPI 的四根口线, 可否如下这么设置?

P0.4 SCK P0.5 GPIO P0.6 MOSI P0.7 GPIO

也就是只用 SPI 的两根, 另外的两根当 GPIO 用, 不知道可否这样?

A: P0.7 要设置为 SSEL, 且要外接上拉电阻, 不能作 GPIO, 片选控制用其它口线。

第355问:

Q: SPINT 中断是在什么情况下产生, SPIF 是数据传输完成标志吗? 接收数据完后是不是也会为 1?

A: SPIF、MODF 置位。配套《ARM 微控制器基础与实战》关于寄存器 SPCR 的说明有介绍。

Q: 谢谢答复。另外, SPI 主机方在接收数据时, SCLK 是不是应该有信号?

A: SPI 是同时收发的, 时钟肯定有。

第356问:

Q: 让数码管显示数字, 要是手碰到电路板板上的某些元件, 数码管就变成乱码, 反正是手对电路有干扰。

A: 数码管是 SPI 接口芯片驱动。SPI 总线在悬浮时很容易受干扰, 表现为用手摸变成乱码。

第10章 定时器

第357问:

Q: 请问, lpc2106 的定时器能作为外部计数器吗?

A: 不行。

第358问:

Q: 我接着《ARM 微控制器基础与实战》上的定时器初始化实例作了, 但当执行 TOTCR=0X01 后, 也就是说我在刚启动定时器 T0 时 TOIR 中的匹配 0 中断标志就置位为 1 了。也就是说 WATE 循环成了死循环了, 到底是怎么回事, 请各位大哥帮帮我, 我在做毕设, 我是用汇编做的。

```
TOPR = 990;           // 设置定时器分频为 100 分频, 得 110592Hz
TOMCR = 0x03;         // 匹配通道 0 匹配中断并复位 TOTC
TOMRO = 11059/2;     // 比较值
TOTCR = 0x03;        // 启动并复位 TOTC
TOTCR = 0x01;
```

```
WATE:LDR R1, [R0]
      ADDS R1, R1, #1
      BNE WATE
```

A: 调试时程序停下来外设还在跑, 定时器还在不断计数, 您应该全速执行。

第359问:

Q: 《ARM 微控制器基础与实战》上定时器实验 2。这个程序其实应该是, 定时器 0 控制 led4 的闪烁, 定时器 1 控制 led3 的闪烁。定时器 0 采用的是匹配 MR0, 产生中断控制亮灭。定时器 1 采用的是匹配 MR0, 然后利用外部匹配的 MAT1.0 的翻转控制亮灭。调试的时候, 会看到 led3 亮灭的频率快, led4 慢。有意思的是, 停止调试后, led4 停止闪烁, 而 led3 仍然闪烁。我认为这是因为 led3 的亮灭都是由内部的寄存器的设定来控制, 不像 led4 是由外部程序(中断)控制, 所以, 即使停止调试, 内部的寄存器的设定值却没有发生变化, 仍然是程序中的设定值, 所以 led3 会不停闪烁。

A: 对, 定时器 0 仍在运行。

第360问:

Q: 在单步运行时, TOTC 和 TOPC 寄存器的值为何不是象仿真 8051 的 TL0 值那样按指令所需要的机器周期数加 1 或加 2?

A: 这是一个很大的问题, JTAG 不支持外围器件的仿真。也就是说定时器等不受暂停控制。

第361问:

Q: LPC2104 ARM 有无计数器? 如有计数功能, 能计外部脉冲的最高频率多大? 是否可以达

到 1MHz? 我要作的项目对能计外部脉冲的最高频率多大很重要, 将决定是否应用 ARM LPC2104, 请给予权威说明, 谢谢!

A: LPC2104 本身没有硬件计数器, 可采用外部中断来进行计数(软件实现), 可将此中断设置为 FIQ 中断。

第362问:

Q: 《ARM 微控制器基础》里有关看门狗的章节中, 我对看门狗固定装载值搞不懂, 如程序清单中 3.41WDT 初始化

```
WDTC = 0x1000
```

请问 0x1000 到底该如何换算成多少时间, 如几秒或几毫秒? 能否指教, 谢谢!

A: 与外设时钟有关, 仅一个定时器重载值而已。

第363问:

Q: 关于 LPC2104 的 5 个独立定时器。

很久以前, 看到了“PWM 建立在定时器 0/1 之上 (based on)”。

于是想当然的, 就把 LPC 的定时器当成了与其他 8 位机一样 (PWM 占用定时器)。

而且不自觉地, 把它的应用能力降低了一个档次 (只有两个定时器, 还被别的模块霸占)……但是, 当实际要用到的时候, 发现豁然开朗 (把英文的模糊, 当作自己理解错误, 不仔细阅读的借口)。

总结如下:

LPC2104 实际上有 5 个独立的定时器。定时器 0/1 说的很明白。

PWM 有独立的分频器, 独立的计数器, 独立的中断。

RTC 也一样, 也是一个独立的完整定时器。虽然只有 15 位 (计数到 32767)。

看门狗, 有固定的分频器 (4), 正常情况下, 它不应该产生中断, 但是, 可以在喂狗以后 (也可以根据差值), 使用 WDTV 进行某些事件的执行时间测量 (在下次喂狗以前)。

虽然不完整, 但也是一个独立可用的定时器。

请问我的理解正确吗?

A: 正确。

第364问:

Q: 关于 EASYARM210x 中定时器的时间问题:

在定时器实验程序 TargetInit() 中设置了 TOPR=99, …… , TOMR0=110592/2, 按照《ARM 微控制器基础与实战》上理解就是没 99+1=100 个 pclk 个周期定时计数器 TC 加 1, 直至匹配 TOMR0, 那实际上是 0.5 秒 TC 复位一次, 并产生匹配中断, 那主程序的大循环就是以这个中断标志来判别的, 那为什么 LED 的闪烁频率是 1s 呢?

A: 仔细看看程序 LED 是每 0.5 秒 状态取反。

第365问:

Q: 请帮我看一看《ARM 微控制器基础与实战》的程序清单 5.21, 多谢!

这是定时器的第一个例子, TOMCR=0x03;

意思是当满足条件时产生中断,而这个程序并不是中断方式的,我将它改为:T0MCR=0x02;
为什么就不好使了,与中断不是没关系的吗,为什么?谢谢!

A: 程序不使用中断方式但是使用中断标志位,是否产生中断标志由 T0MCR 决定。

第366问:

Q: 请问:做定时器的第二个实验时为什么当点“stop”后还是停不下来,LED3 继续闪动呢?

A: JTAG 仿真调试时,stop 是停止程序的运行,但不会停止 LPC2104 的时钟及其外设,也就是说定时器 0 没有停止,所以在匹配后输出翻转。

第11章 存储器

第367问:

Q: 请教 LPC2212 不能访问外部存储空间的问题。

我用 LPC2212 外挂了一个接口芯片, 需要以 16bits 的方式访问, 程序以内部 Flash 引导, 使用内部 RAM, 程序初始化时设置:

```
PINSEL2 = 0x0f814910
BCFG1   = 0x100fbef    // 接口芯片使用 CS1
```

但我在程序中不停读写 0x81000000 地址, 片选 CS1 始终为高电平, WE 信号也一直为高, 不知何故?

A: 自问自答: 问题原因已经找到...

原来是我为了节省功耗, 在 CONP 中误将 EMC 关闭! 希望大家不要犯同样的错误....

第368问:

Q: 我使用 LPC2200 的工程模板使用外部总线时, 我只在 LPC2294.H 中加入

```
#define M1 (*(volatile unsigned short*) 0x80000000)
#define M2 (*(volatile unsigned short*) 0x81000000)
#define M3 (*(volatile unsigned short*) 0x82000000)
#define M4 (*(volatile unsigned short*) 0x83000000)
```

然后建立一个 main.c 文件, 如下:

```
#include "config.h"
uint32 i;

void delay(void)
{
    for (i=0;i<5000;i++);
}

int main(void)
{
    while(1)
    {
        M1=5;
        delay();
        M2=6;
        delay();
        M3=7;
        delay();
        M4=8;
        delay();
    }
}
```


我单步调试，用示波器观察 CS0, CS1, WE, A0, A1。

结果：到 M1=5 之后看到一个 CS0 低，在 M2=6 之后也是 CS0 一个低，WE 看不到，A1 能看到，几个单步之后程序就飞了，请问是什么问题？

- A: LPC2200 工程模板，若使用片外 RAM 调试(DebugInExram 目标)时，程序放在 bank0 块，只要运行程序，CS0 都会有输出(因为要读程序回来执行)。工程模板中默认对 bank2、bank3 并没有配置，代码是注释的，见 startup.s 文件。几个单步之后程序就飞了，这可能与硬件电路有关。

第369问:

- Q: 我设定在片内 FLASH 中调试，在外部总线扩展 AD 和液晶，程序还没有进入 MAIN 就跳到数据中止了，不知道为什么。对应 DebugInChipFlash 目标的分散加载文件为 mem_c.scf，它使用了外部 RAM 作为变量存储区，如下：

```
ROM_LOAD 0x0
{
    ROM_EXEC 0x00000000
    {
        Startup.o (vectors, +First)
        * (+R0)
    }

    IRAM 0x40000000
    {
        Startup.o (+RW, +ZI)
    }

    ERAM 0x80000000
    {
        * (+RW, +ZI)
    }

    HEAP +0 UNINIT
    {
        heap.o (+ZI)
    }

    STACKS 0x40004000 UNINIT
    {
        stack.o (+ZI)
    }
}
```

- A: 这是分散加载的问题，如果只有片内 RAM，更改如下即可：

```
ROM_LOAD 0x0
{
```

```

ROM_EXEC 0x00000000
{
    Startup.o (vectors, +First)
    * (+R0)
}

IRAM 0x40000000
{
    Startup.o (+RW, +ZI)
    * (+RW, +ZI)
}

HEAP +0 UNINIT
{
    heap.o (+ZI)
}

STACKS 0x40004000 UNINIT
{
    stack.o (+ZI)
}
}

```

第370问:

Q: 我要使用外部总线, 请问如何配置? 这样行不行? 先在 LPC2294.H 中加入

```

#define DISP_COM (*(volatile unsigned long*) 0x81000002)
#define DISP_DAT (*(volatile unsigned short*) 0x81000000)

```

然后在 MAIN 中

```

int main(void)
{
    BCFG0=0x1200fbef;
    BCFG1=0x1000fbef;
    // PINSEL0 = 0x00000000;           // 设置所有管脚连接
    // PINSEL1 = 0x00000000;
    // PINSEL2 = 0x0b814914;
    while(1)
    {
        DISP_COM=0;
        for(i=0; i<500000; i++);
    }
}

```

我的程序使用的是 LPC2200 模板, 在 STARTUP.S 中设置的 BCFG*, 编译后用 LPC2114 的 EASYJTAG 下载, 在内部 FLASH 运行, 但在 MEMORY 中看不到数据的变化, 用示波器量 CS,

OE, WR 都看不到, 是不是模板有问题?

A: 模板没有问题。注意一下您的读写代码是否被优化掉了。

第371问:

Q: 我有一个简单的分散加载文件, 如下:

```
ROM_LOAD 0x00000800
{
    ROM_DATA1 0x00000800
    {
        p2.o (+R0)
    }

    ROM_DATA2 0x00000900
    {
        p3.o (+R0)
    }
}

ROM_LOAD 0x00000000
{
    ROM_EXEC 0x00000000
    {
        Startup.o (vectors, +First)
        *.o (+R0)
    }

    IRAM 0x40000000
    {
        Startup.o (+RW, +ZI)
        * (+RW, +ZI)
    }

    HEAP +0 UNINIT
    {
        heap.o (+ZI)
    }

    STACKS 0x40004000 UNINIT
    {
        stack.o (+ZI)
    }
}
```

我发现此文件有错误，ADS 编译不能通过，它说有两个 ROM_LOAD 区，后来我把它改成：

```
ROM_LOAD 0x00000000
{
    ROM_EXEC 0x00000000
    {
        Startup.o (vectors, +First)
        *.o (+R0)
    }

    ROM_DATA1 0x00000800
    {
        p2.o (+R0)
    }

    ROM_DATA2 0x00000900
    {
        p3.o (+R0)
    }

    IRAM 0x40000000
    {
        Startup.o (+RW, +ZI)
        * (+RW, +ZI)
    }

    HEAP +0 UNINIT
    {
        heap.o (+ZI)
    }

    STACKS 0x40004000 UNINIT
    {
        stack.o (+ZI)
    }
}
```

在 DEBUGINFLASH 时 JTAG 下载到 FLASH，可以运行，但不能选择编译产生 HEX 文件，请指点！要不两位大侠就做一个分散加载，且有数组指定位置的简单项目。急呀！我的项目已在公司立项了！

用的就是 ARM2100。

A: 改一下名字，让它们不同名即可。

第372问:

Q: 您好, 请教以下几个问题:

1. EASYARM2100 实验板中 flash 中的程序是 flash 中执行的还是在 sram 中执行的?
2. 在复位后, flash boot 中的内容映设到片内 ROM 中, 那 flash boot 中是哪些内容呀? 这些内容都在片内 ROM 中执行吗?
3. 为何在复位后, 中断向量在 flash 的底部出现, 这些是硬件自动完成的, 还是需要我自己编程?

A: 1. flash。
2. ISP、IAP 等代码, 在 flash 中执行。
3. 硬件自动完成的。

第373问:

Q: 如果要将在 flash 中特定的某个子程序映射到 ram 中运行, 改如何作? 能否举例说明?

A: 硬件不支持, 只能用软件解决。其实, 在 FLASH 中运行速度几乎没有损失, 一般不必在 RAM 中运行。

第374问:

Q: 在 ADS 中使用常量或数组, 编译后, 老把它放在 RAM 中! 看了大家发的帖, 是不是要是用分散加载才行啊! 想请问分散加载描述文件, 是不是新建一个 .s 的文件, 然后加入到工程中去啊! 还是别的, 请各位指教啊!

A: 当然不是, 看看例子里就知道了。

第375问:

Q: 我在 datasheet 中看到在取消了 FLASH 后原来的 BOOT 块被映射到了 ROM 中, 但是文件中没有写在 ROM 中有那些东西? 如果我想设置 BOOT 引脚完全在外部存储器启动, 应该怎么办? 如何设置外部存储器?

A: 2200 系列的外部启动都一样的。

第376问:

Q: BOOT 块映射的那部分 ROM 中都是什么代码? 掩膜的 BOOTLOADER? 外部扩展是不是只能用 EEPROM/SRAM? 至少 SDRAM 不能用了, 没驱动电路!

A: BOOT 块同样是 BOOTLOADER 代码, 使用手册有说明。
外部扩展不能用 SDRAM。

第377问:

Q: 分散加载的问题, 请指点!

```
//这是含汇编的原文件:  
#include "config.h"  
//#define ss ((volatile unsigned char *) 0x00000800)
```

```

extern const unsigned char pp2, pp3[20];
int main(void)
{int *pa, *pb, a, b, e;
[0xe59f3038] * ldr    r3, 0x00000234 ; = #0x00000900
[0xe59fc038]  ldr    r12, 0x00000238 ; = #0x00000800
while(1)
//ss=0x11;
{*pa=pp2;
[0xe5dc0000]  ldrb   r0, [r12, #0]
[0xe5820000]  str    r0, [r2, #0]
*pb=pp3[0];
[0xe5d30000]  ldrb   r0, [r3, #0]
[0xe5810000]  str    r0, [r1, #0]
for(e=0;e<20;e++)
[0xe3a00000]  mov    r0, #0
.....
[0xe2800001]  add    r0, r0, #1
[0xe3500014]  cmp    r0, #0x14
[0xbaffffffb] blt    0x210 ; (main + 0x1c)
{a=*pb+20;pb++;};
[0xe2811004]  add    r1, r1, #4
for(e=0;e<20;e++)b=*pa+e;
[0xe3a00000]  mov    r0, #0
[0xe2800001]  add    r0, r0, #1
[0xe3500014]  cmp    r0, #0x14
[0xbaffffffc] blt    0x224 ; (main + 0x30)
[0xeaffffff1] b     0x1fc ; (main + 0x8)
[0x00000900]  dcd   0x00000900 ....
[0x00000800]  dcd   0x00000800 ....
}
return 0;
}

```

p2.c 文件源码:

```
const int pp2=14;
```

p3.c 文件源码:

```
const unsigned char pp3[20]=
{'a','a','a','a','a','a','a','a','a','a','a',
'a','a','a','a','a','a','a','a','a','a'};
};
```

分散加载文件:

```
ROM_LOAD 0x00000000
```

```
{
  ROM_EXEC 0x00000000
  {
    Startup.o (vectors, +First)
    *.o (+R0)
  }

  ROM_DATA1 0x00000800
  {
    p2.o (+R0)
  }

  ROM_DATA2 0x00000900
  {
    p3.o (+R0)
  }

  IRAM 0x40000000
  {
    Startup.o (+RW,+ZI)
    * (+RW,+ZI)
  }

  HEAP +0 UNINIT
  {
    heap.o (+ZI)
  }

  STACKS 0x40004000 UNINIT
  {
    stack.o (+ZI)
  }
}
```

编译通过，pp2, pp3 却还没有到达指定地址，为何还不行？

A: 可以修改成如下：

```
ROM_LOAD 0x00000800
{
  ROM_DATA1 0x00000800
  {
    p2.o (+R0)
  }
}
```

```
ROM_DATA2 0x00000900
{
    p3.o (+R0)
}
}

ROM_LOAD 0x00000000
{
    ROM_EXEC 0x00000000
    {
        Startup.o (vectors, +First)
        *.o (+R0)
    }

    IRAM 0x40000000
    {
        Startup.o (+RW, +ZI)
        * (+RW, +ZI)
    }

    HEAP +0 UNINIT
    {
        heap.o (+ZI)
    }

    STACKS 0x40004000 UNINIT
    {
        stack.o (+ZI)
    }
}
```

第378问:

Q: 再请教: 如何在 ADS 中定位常量或数组?

如我定义一个常量 `const uint32 Pi=314` 到 flash 地址 `0X32000000` 一个数组 `const uint32 hzk[512]={12,24,.....}` 到 flash 地址 `0x33000000` 以上两种该如何分散加载?

A: 如果是 ADS1.2 软件, 需要使用分散加载机理实现, 以下是一个分散加载描述文件的内容。

```
ROM_LOAD 0x80000000
{
    ROM_EXEC 0x80000000
    {
        Startup.o (vectors, +First)
        * (+R0)
    }
}
```



```

    }

    IRAM 0x40000000
    {
        Startup.o (+RW,+ZI)
    }

    STACKS 0x40004000 UNINIT
    {
        stack.o (+ZI)
    }

    ERAM 0x81000000
    {
        * (+RW,+ZI)
    }

    HEAP +0 UNINIT
    {
        heap.o (+ZI)
    }
}

```

给项目增加两个文件，Pi.C 和 Hzk.C，内容如下：

```

// Pi.c
#include "config.h"
const uint32 Pi = 314;

// Hzk.C
#include "config.h"
const uint32 hzk[] = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
                      .....
                      }

```

用户程序一定要使用这两个常量。分散加载描述文件举例：

```

ROM_LOAD 0x00000000
{
    ROM_EXEC 0x00000000
    {
        Startup.o (vectors, +First)
        *.o (+R0)
    }
}

ROM_DATA1 0x00004000

```

```

    {
        Pi.o (+R0)
    }

ROM_DATA2 0x00005000
{
    Hzk.o (+R0)
}

IRAM 0x40000000
{
    Startup.o (+RW, +ZI)
    * (+RW, +ZI)
}

HEAP +0 UNINIT
{
    heap.o (+ZI)
}

STACKS 0x40004000 UNINIT
{
    stack.o (+ZI)
}
}

```

第379问:

Q: 请教大家一个问题: 我想用一 bootloader 拷贝一程序从 ROM 到 SDRAM, 然后在 SDRAM 中运行, 但是这一程序的 READONLY AND READWRITE 地址应该设为多少啊? 还有, 这是一个用 C 语言写的程序, 入口地址该怎么设置啊? 将该程序的二进制代码写入 FLASH 中后, 我可以通过跳转进入该程序, 但是我个人认为, 在 FLASH 中和 SDRAM 中各有一个程序, 它会跳转到哪一个呢? 我不知道是否存在这样的问题, 请教高手指点一下啦, 感激不尽啊!!!

A: bootloader 实现程序的 COPY 及映射工作, 其目的就是使你的程序在实际使用和仿真时一致。bootloader 采用单独的程序, 定位在 0X00, 你的程序定位在 SDRAM 中, 将链接得到的两个 BIN 文件分别烧到 FLASH, 其中你的程序烧的位置必须是固定的, 也就是必须让 bootloader 知道。

1. 同一个工程中实现 bootloader, 你需要仔细设计 bootloader 让它成为地址无关的程序段, 并通过程序判断是否执行中断向量的映射。

第380问:

- Q: 进入 axd 后, 调试程序总出现 no disassembly could be read at the address!
我的设置: ro 0x40000000, rw 0x40008000, image entry point:0x40000000, 中断向量所在模块 vectors.o startup。到底是什么原因呢?
- A: lpc2104 只有 16KRAM, RAM 的最高地址为 0x40004000, lpc2106 才能这样设置。

第381问:

- Q: 存储器加速模块起什么作用啊? 我看不明白, 哪位大虾能告诉我啊? 谢谢了!
- A: 内核很快, 但是当今世界 FLASH 很慢, 只有 TI 的 28xx 可以做到 100MIPS, 能从 FLASH 执行! 而存储器加速模块实现了“堆叠”, 能一下子读出 128 位送给内核, 内核就可以以四倍的速度执行 FLASH 里的东东。

第382问:

- Q: 我打算使用 LPC2XXX 系列 ARM, 我的系统应用中需要大量的数据存储, 大约是 10M 字节, 但是 LPC2XXX 系列没有并行的数据和地址总线接口。请问用什么方式解决好? 谢谢!
- A: lpc22xx 具有外部总线。另外, 您可以用 IO 口模拟总线, 本公司网站有其软件包。

第383问:

- Q: 请问 LPC2104 如何扩展 RAM?
- A: 用 I/O 模拟总线操作。
<http://www.zlgmcu.com/philips/arm/lpc2104.asp>
模拟总线软件包
<http://www.zlgmcu.com/download/downs.asp?ID=959>

第384问:

- Q: 我本意是在 Flash 中调试, 当 AXD 载入映像文件时, 却看到 0x40000000 中也有代码, 与 0x00000000 处一样, 为什么?
- A: 你没有把 remap 语句删去。

第385问:

- Q: 我想将一段数据(表)放在 Flash, 虽然都说没有必要固定在指定位置, 但这种情况你怎么办?
- 假如: 这段数据放的是一个电话号码, 但在应用中我的客户需要将电话号码改变。那么这里要用到 IAP, 如果不知道这个电话号码在 Flash 中的地址, 怎样改变?
- A: 您必须把它固定, 并使用指针访问。

第386问:

- Q: lpc2119 能带外部存储器吗? 怎样连接?
- 那既要有 CAN 控制器又要外部存储器怎么办?

A: 可以用 2210, CAN 可以用 SJA1000 来做。(注: 最好使用 LPC2292, 其带 CAN 控制器及外部总线)

Q: 这是个好办法, 但一个 16/32 外部总线, 如何与 8 位的 SJA100 连接? 有讲究吗?

A: 2210 可以配置成 8 位总线, 具体看用户手册。

Q: 分开的地址总线、数据总线如何与复用的总线 (AD) 相连接?

很多 32 位的 ARM 与 8 位的外设都存在这个问题!

ARM 的输出 CS 信号能不能看成是 ALE 信号?

是不是中间要加一个 (CPLD/FPGA) 这东东?

而且双向的总线选择在 CPLD/FPGA 中实现还比较麻烦, 是不是?

A: 双向总线在 CPLD 中实现不麻烦, 最好用语言来实现, 另外你还是对 8 位机的总线时序不够了解, 建议先仔细研究一下 8051 的总线读写信号、ALE 信号和 8 位总线之间的时序关系。

(编者注: 现在已经有带 CAN 控制器、带外部总线的 LPC2292 芯片。)

第387问:

Q: 请教 LPC2214 与 16 位 RAM 的接口。地址线如何连接, A0 是不是要用?

A: A0 不要, 应用手册有详细的说明。

第388问:

Q: 在 LPC2104 中有 128k 的 FLASH, 其高端 8k 是用来存放 BOOT BLOCK 还是用户自己的程序?

也就是说用户可用的空间是 128k 还是 120k? 《ARM 微控制器基础与实战》中内容我没有完全理解。

A: 用户空间 120k。高端 8k 是存 boot。

第389问:

Q: 请问 LPC2104 可以外扩 RAM 和 flash 吗? 他的管脚会不会太少了。

A: LPC2104 总线不开放, 只能使用 I/O 口模拟总线。

第390问:

Q: 请问: 如何往 2104 固定地址的 FLASH 中写数据, 又如何读?

A: 调用 IAP 写, 用 LDR 指令读。

第391问:

Q: 请教: 2200 开发板的 flash 用的是 sst39vf160, 他没有字节定位引脚, /lb, /ub, 那它是怎么进行字节定位的?

A: 读: 一次读 16 位数据, 然后由 LPC2210 屏蔽无效字节。

写: 一次写 16 位数据。

第392问:

Q: 我们要做一批控制电机设备的板卡, 想用 2214, 不知道扩什么 FLASH 和 RAM 好呢?
请推荐一下, 谢谢!

A: 我们的板上用 IS611v25616 和 39vf160, 但您的板也许需要抗干扰更好的 (看您的硬件如何设计)

第393问:

Q: 1. 为什么 LPC210X 的板子可实现 JTAG 仿真, 而 ST 公司的 PSD 产品的 JTAG 只支持编程
下载不支持仿真?

2. RO BASE 和 RW BASE 分别指的什么地址?

A: 内核设计时不一样, ARM7TDMI 设计了 DEBUG 功能, RO: Read Only, RW: Read Write。

Q: 是否就是 RO 设置的是程序开始的地址, 而 RW 设置的是 RAM 开始的地址呢?

所以在 RAM 中调试时要将 RO 设为: 0X40000000 呢?

A: 可以这么理解。因为 RAM 的擦写速度快而已。程序大了就直接在 FLASH 中调试了。10 万次, 够用了。

第12章 JTAG

第394问:

Q: 1. 主要问题是 RST 和 TRST 结构, 在 easyjtag 中是不是 OC?

2. 在 easyarm2100 板上, MAX708 和 74HC125 组成了两个类似 OC 结构, 是不是可以用 PNP 管代替?

3. 如果更简单处理 (调试比较少情况), 是不是可以直接连接 708 的 /RST 和 nRST、nTRST?

(调试时把它们分别断开, 单独上拉)。

A: 1. 不是 OC 输出。

2. 最好使用芯片。

3. 如果不使用 JTAG 调试, 直接将复位信号连到 nRST、nTRST。

第395问:

Q: 我今年初买了块贵公司的 ARM2104 学习板, 一直没有空用。昨天我接在电脑上, 通过 JTAG 接口仿真时, 发现很难正确进入仿真状态。具体现象如下:

2104 板买来时就写有程序, 一上电后, 七段数码管就依次显示 0-9, 上电时还有蜂鸣器提示。我用的软件是 ADS1.2。我是按照《ARM 微控制器基础与实战》上提示的操作步骤来练习一个例程。打开一个 PROJECT, 比如 LEDCON, 然后设置 ARM LINKER 等参数, 按照《ARM 微控制器基础与实战》上提示设置的。然后 COMPILER->MAKE->DEBUG。进入 AXD 界面进行调试。这时候就有问题了。进入 AXD 界面时, 板子好像被复位了, 蜂鸣器发出提示音, 数码管从头再重新显示 0-9。然后, 在信息提示栏出现下面的提示:

0053:Image Shorter than implied by header(has been squeezed?)。

AXD 界面内没有调入任何信息。通过 FILE/LOAD IMAGE 来调入 LEDCON.AXF, 同样没能调入任何信息。而且, 这时板子上的 CPU 仍然是激活的, 并没有处于 HALT 状态, 数码管依次显示 0-9。我在主 JTAG 和 PC 自动判断主/从 JTAG 两种状态下都试过, 结果都一样。情况大致就是这样。

我昨天晚上试了三个小时, 只有两次成功地进入 JTAG 状态。也就是说, AXD 界面上可以看到程序指令, 以 LEDCON 工程为例, 按下 GO, 就只有一个 LED 灯不停地闪烁, 数码管熄灭。

我想弄清这是为什么! 如何解决这种问题。

A: 请直接用我们的工程试一下 (拷贝到硬盘, 去掉只读属性)。

第396问:

Q: 昨天晚上我回到家后, 再打开 ADS1.2 试了一次, 我发现启动 AXD 后, 进行 CONFIGURE TARGET 时, JTAG 的驱动程序是 EASYARM.DLL, 而不是资料上提到 EASYJTAG.DLL, 我是从光盘上的 EASY_DRIVER 中拷贝的。

A: 请在我们的网站下载最新驱动试一试。

第397问:

Q: 今天用 AXD 发现有些异常,总是无法正常调试,给了提示,大家帮忙看看。

Flash Sector 0 write failed!

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!

A: 将 ISP 跳线插上试一试。

第398问:

Q: 我想自己做一块 2214 的板子,请问可不可以用 2104 的 easyjtag 来仿真?能的话要用什么版本的驱动?你们的 EasyJTAG 单独卖吗?

A: 正在开发软件之中!

第399问:

Q: EasyJTAG 是不是也是和 wiggler 一样的?

A: 不一样。

第400问:

Q: 我的 EWARM 用 easyjtag.dll,提示配置不正确或者电缆没链接好!easyjtag.dll 是不是只支持 ads1.2?

A: 理论上支持,且有用户成功过,不过我们不做这方面的技术支持。

第401问:

Q: 关于 2119 数据手册上的一句话不太明白。

上面说到“当作测试/调试接口时,GPIO/其他共用这些管脚的功能不可用”,是说调试时,与 GPIO 共用管脚的那些功能不能用,还是什么其他的意思?

A: 进行 JTAG 调试时,所用的调试口线不能做 GPIO。

第402问:

Q: P0.14, P1.20 和 P1.26 几个引脚对是否启用 JTAG 调试至关重要,根据数据手册上所说的意思他们都在复位时保持低电平可以确保进入调试跟踪,可为什么 P0.14 要下拉,另外两个要上拉?不明白!

A: P0.14 0-->进入 ISP, 1-->运行用户程序;

P1.20 0-->跟踪口使能, 1-->跟踪口禁能;

P1.26 0-->JTAG 口使能, 1-->JTAG 口禁能。

第403问:

Q: 我若不采用跳线,将 P0.14、P1.20、P1.26 都固定到低电平会有问题吗?

1. 不清楚接地电阻大小如何确定?

2. 假设我都把他们确定下来(一直保持低电平),那上电复位后将先进行 ISP 下装我的

- 程序， 接下来由于那 P0.14 不能复原，会不会接着下装程序以致于干扰调试？
3. 我看了 2100 实验板的原理图，上面有关于跟踪的线没有接，那如何跟踪呢？
 4. JTAG 的几条接线中，TDO 和 TDI 是用来传输数据的，在下载时和调试时都用这两条线进行 数据传输吗？

- A: 1. 只需要上拉电阻，跳线短接时口线为接地即可。
2. 不影响程序下载，因为 ISP 操作过程中不会对 P0.14 口操作。
3. 需要仿真器的支持，J7 就是跟踪调试接口。
4. 应该是，可以找找 JTAG 的资料看看。

第404问:

Q: 我用的是 2100 套件，已经连上 JP8 (ISP)，用 ADS 加 JATG 调试，同一个例子程序，用 debuginRAM 运行正常，用 debuginFLASH 就老是在装载程序。

A: 你的程序已经下载进去了，只是你下完了就直接运行，因此会一直在加载。请把 ISP 断开，然后按复位键运行下载的程序。

第405问:

Q: 如何通过 mluti-ice 对 2104 仿真，试过不行，请提供方法，购买的开发板仿真器坏了。

A: 可以的，一直在用 MULTI-ICE。JP6 接 PRI，JP7 接 DBGEN (EasyARM2104 试验板)。

第406问:

Q: 那天好了问了你的这个问题，现在想再确定一下，

1. 我的成品板上应该有一个 JTAG 接口吧，用它来调试和烧录程序，对吗？我就是想确定一下是否留次 JTAG 口调试烧录就够用了？
2. 如果我说的对的是对的？那这个 JTAG 口怎么引线？

A: 如果，单单烧录，最好使用 UART 进行 ISP 下载。因为如果使用 JTAG 烧录，需要主 JTAG，占用 IO 太多，走线不方便，除非你的板子够大。也就是说使用 ISP 烧录，而使用从 JTAG 调试(有些需要注意的地方)。

第407问:

Q: 我买了 2104 的板子，按照说明安装了 JTAG 的驱程并进行了选择设置后会弹出错误对话框，继续按照说明选择后对话框仍存在，不知应如何解决？

A: 在《ARM 微控制器基础与实战》的最后有常见问题，看下！

第408问:

Q: 我原买了一块 2104 开发板很好用，最近又买了块 2100 板，下了新的 JTAG 驱动，用 ADS 调 2119 也很好用。

有朋友借 2104 用，发现 2104 的 JTAG 不能用了，请版主或周工告知这两种 JTAG 是不是不能通用？或者要重装驱动？还是 2104 的 JTAG 坏了。谢谢各位。

A: 是可以使用的，要选择正确的 CPU。

第409问:

Q: 请教如何只使用 从 JTAG?

A: JP6、JP7 短路器取出, 此时只能使用从 JTAG 进行调试。

但有一点要注意, 就是想办法让 PINSEL1=0x55400000, 使从 P0.27--P0.31 配置为从 JTAG 口(可通运行 FLASH 中的程序设置, 所以要写一段代码放入 FLASH 的 00000000H 起始地址)。在仿真器配置设置中, 设置为不复位系统。在 AXD 调试环境, 打开 "Options->Configure Target...", 弹出 "Choose Target" 窗口, 在 "Target Environments" 框中选择 "EasyJTAG..." 项。点击 "Configure" 按钮, 进入 "EasyARM Configuration" 设置窗口, 在 "Options" 项中选择 Halt program。

第410问:

Q: 在 2100 套件上, 换了 EasyJTAG_Driver_V104.zip, 反而不能调试了, 套件的 CPU 为 LPC2119。

A: 应该没有问题, 直接将文件复制到 "C:\Program Files\ARM\ADsv1_2\BIN", 覆盖原先的文件。

使用片内 FLASH 调试时有时下载慢一点。

第411问:

Q: 鉴于贵公司现在没有适用于 ARM7 核 (带有双 CAN 控制器如 LPC2119) 的开发平台, 我若购买 EASYARM 显然不合适, 只能自己搭建这样一个 CAN 开发平台 (电路板), 对于 JTAG 调试我不大了解:

1. 是不是应该在我的电路板上扩展一个专供 JTAG 调试用的插槽?
2. 若是这样, 这个接口应如何搭建?
3. JTAG 下载调试器可否通用, 贵公司可否单卖 (当然要买芯片)?
4. 若欲将调试好的程序固化于芯片之中 (以前的烧录), 在这里用什么方法, 是否还要用其他装置?

A: 开发板可以直接拿来调试, 电路也可以参考。

1. 是, 如果要 JTAG 调试的话。
2. 与所使用仿真器的接口有关。可以参考 ARM 公司提出的标准 20PIN JTAG 接口。
3. EasyJTAG 支持 LPC2100, 与开发板配套销售。
4. 可以使用 ISP 下载, JTAG 下载 (EasyJTAG 支持)。

第412问:

Q: 为什么我得 easyjtag 不能用? 按照说明《ARM 微控制器基础与实战》中设置好了, 在 axd 中出现错误:

dbe warning00041:An unspecified Debug Toolbox call failed

A: 选择 Connect mode..., 再选择第三项, 确定后点击 Restart, 参考 <http://www.zlgmcu.com/download/downs.asp?ID=1074>。

第413问:

Q: 我在用 JTAG 调试时, 就是用的 ZLG 公司提供的原程序, 然后按 F5, 设置 options, 接着 load image, 出现的问题就是 AXD 一直处于 LOADING image 状态, 还有一个显示 loading image 的小窗口, 接着他就在那里不停 loading, 即使你把开发板的电源拔掉, 还是这种 loading 的状态, 好像下载程序和开发板没有关系一样, 不知道 JTAG 想把程序下载到那里, 关了重新来一次还是这样的状态。还有就是我等上电以后测板子上的 5V 电源实际是 5.5V, 测 3.3V 的电源是 3.6V, 1.8V 是 2V 了, 这是不是在正常的范围内?

A: 请按下以下步骤操作:

1. 是 EasyARM2100 开发板吗? 请按照《ARM 微控制器基础与实战》上第 2.5 章节安装驱动程序。原程序要先复制到硬盘, 并将文件的只读属性去掉。
2. 下载 V1.04 版本的 EasyJTAG 驱动程序
<http://www.zlgmcu.com/download/downs.asp?ID=1069>。
3. 开发板上电源是使用 LD0 产生的, 生产测试首先测试的就是电压, 应该没有问题, 可能是电压表不准吧。

第414问:

Q: 用从 JTAG, JP6、JP7 为空,

```
debugRel ro=0x40000000, rw=0x40003000
```

```
debug, ro=0x40000000, rw=0x40003000(40), image=0x40000000
```

```
release, ro=0x00000000, rw=0x40003000(40), image=0x00000000
```

AXD 的 Config target 设置: TP2, LPT1, errase flash when needed, halt and restart

如果直接在编译成功后, 按 F5, 会出现 vectors 错误。

如果从程序打开 AXD, load .axf 后, 出现反汇编窗口, 但是按 F10, 出现

```
no disassembly could be found !!
```

F5, 则没有反应!!

试过重装驱动和 ADX, 不行!! 斑竹说说吧!! 我都郁闷好几天了!!!!!!

A: 要先配置从 JTAG 管脚功能, 然后在配置 EASYJTAG 时使能“必要时写 FLASH”, 去除复位那个选项。

第415问:

Q: 请教 TRST 和/RST。

从周工的板上我们可以看到 TRST 和/RST 是不互连, MAX708S 出来的复位信号经过 74HC125 分别传给 TRST 和/RST。在其它 2104 的使用设计中, 该 74HC125 可不可以省略?

毕竟多了一个器件的。如果省略, 这两根线如何接? 请告知。

另外, 没有 MAX708S 的话, 708T、708R 大概也能代替吧。

A: 可以省略。

R、S、T 表示复位门槛电压的大小分别是 2.63V、2.93V、3.08V。

Q: 请问: 74HC244 是不是只在主从 JTAG 选择的时候起作用? 如果我只用从 JTAG, 还需要 74HC244 吗?

A: 是, 不需要。

第416问:

Q: 我刚刚接触 ARM, 对很多东西都不是很了解, 做了一段时间的 EASYARM, 在软件方面很有收获, 现在想做自己的板子, 但是遇到了一些硬件方面的问题, 想请教一下:

JTAG 中的 TRST、TDI、TMS、TCK、RTCK、TDO、RST, 分别与 LPC2104 中的 TRST (P0.27)、TDI (P0.30)、TMS (P0.28)、TCK (P0.29)、RTCK、TDO (P0.31)、RST 直接相连可不可以下载、调试程序! 与 EASY ARM 的 JTAG 仿真会有什么区别?

要注意哪些东西?

如果我将这几个引脚用跳线分开, 在下载程序之后, 这些引脚可不可以再用?

为什么要设立主从 JTAG 接口电路?

A: 自己做板子可以选用 LPC2114, 可加密, 有 AD, 更多 IO, 且软件兼容, 价格目前好像一样。

JTAG 的设计简单的多, 可以搜索一下以前的帖子。

第417问:

Q: 那么所谓的进入从 JTAG 调试时不能硬复位系统, 是不是指不能按下板上的 reset 键?

那复位电路岂不是没有作用了?

A: 不是, 是您的仿真器要设置为不复位芯片, 当然, 在仿真工程中肯定不能按下板上的 reset 键。

第418问:

Q: 请问, 在 EASYARM2104 中, KEY4 是不是连接了什么特殊的电路, 在 EASYARM2104 的几个按钮, 在通电时, 只有 KEY4 为低电平, 其余都为高电平, 按照电路图中所画, 他应该为高电平才对, 请您指教!

A: 使用主要 JTAG 调试占用了这些引脚, 详细看配套《ARM 微控制器基础与实战》附录 2。

第419问:

Q: 我们购买了一个在线仿真器使用的时候一直无法识别咱们的板子, 我想问一下是否是因为口线不兼容所致?

A: 使用别的仿真器上时 JP6、JP7 不能用默认值。

第420问:

Q: 2104 由于 TDO 上电时为输入状态, 所以无法采用从方式, 2212 的 TDO 上电时已经是输出状态, 所以我想, 直接采用从方式应该没有问题。

当然, 这只是预留, 万一在开发过程中想用仿真器时, 就没有必要改版啦!

另外, 我想知道 ETM 调试有什么优点, 和 JTAG 联合起来调试能给我们带来什么好处, 难道单 JTAG 调试达不到效果吗? 愿指教!

A: 2212 仅一个 JTAG。

第421问:

Q: RTCK 接 4.7k 下拉电阻则硬件进入 JTAG。当然, 如果仿真器的 RTCK 接了上拉电阻, 则这个电阻应该更小一些。

A: 您可以看一看使用手册如何关闭 JTAG 及 ETM。

第422问:

Q: 请问, 仅使用从 JTAG 接口时, 您提供的方法是通过 ISP 或并行编程器烧录一个程序到芯片中, 这个程序配置引脚连接模块, 使 P0.27~P0.31 使用第二功能。我在试验中, 先利用 ISP 在 FLASH 中的 SECTOR0 烧录了一个只含有“PINSEL1 = 0x55400000”的程序; 把程序编译到 0X00002000 的起始地址处, 想通过 AXD 把程序写到 SECTOR1 中进行调试, 我知道这中间存在一个 FLASH 的 SECTOR0 与 SECTOR1 的程序空白处。我不知道怎么办。望您能给予指点。到底怎么样才可以用您介绍的方法只是用从 JTAG 来调试程序。

A: 您的程序也可以从 0 地址开始, 只要新程序一开始就执行 PINSEL1=0x55400000 即可(如果启动代码不长, 其作为 main 函数的第一句即可)。您先前写到 flash 中的程序也有相同的要求。

第423问:

Q: 随便在配套光盘里找了个例子, 编译通过了, 可在 AXD 中就是无法调试。

选择 Target 的时候:

选择 ADP -> ARM serial/parallel driver, 随后 AXD 底部出现进度条, 过了好一会儿, AXD 告诉我无法连接目标板。

选择 ARMUL 则可以进入调试, 可是进入了死循环, 无法进一步调试。死循环代码如下:

```
void TargetResetInit(void)
{
    .....
    while((PLLSTAT & (1 << 10)) == 0); <<<-----在此地死循环
    .....
}
```

那位大侠帮忙看看是为什么, 谢谢

A: 没有装 EasyJTAG 的驱动。

第424问:

Q: 要进入 2119 的 debug 状态, RTCK 脚应该一直为低还是仅在 reset 为低的时候为低?

A: 在 reset 时为低。接一个 4.7K 的下拉电阻即可。

第425问:

Q: 看了关于主从 JTAG 方面的介绍, 有如下两个问题, 望大哥们解答:

1. 《ARM 微控制器基础与实战》上说主 JTAG 用 15 根 IO, 而我只找到 10 根, 请问另外 5 根在哪里啊?

2. 请大哥具体说明一下由仿真器自动进行 JTAG 转换的步骤, 当由 JTAG 自动转换的时候要执行那条从 JTAG 的 IO 设置指令吗?

- A: 1. 从 P0. 17 到 P0. 31 均占用。
2. 在主 JTAG 执行那条指令然后进入从 JTAG 调试。

第426问:

Q: 当系统运行之后, 不能用 break 中断, 怎么解决?

A: 应该可以的, 除非您用次要 JTAG 又没有按要求设置 PINSEL1。或是硬件故障 (如电源接触不好)。

第427问:

Q: 仔细看了附录二, 有点问题不明白, 而我想搞清一些必须的问题才好自己设计, 我不想带那么多的片子在我的 MCU 核心板上。

1. 主从 JTAG 在功能上有无区别 (除去启动条件)?
2. 学习板上的“自动选择”是如何实现的? (因为用的是标准 AXD, 对 PCPRISEC, PCDBGSEL 的控制是谁实现的) 实现的时序。
3. 我用仿真模式时, 下载代码是主还是从在工作。仿真时是主还是从在工作。

- A: 1. 除了启动条件外没有任何区别。
2. 学习板上的“自动选择”是由仿真器驱动程序实现的。
3. 均可。

第428问:

Q: 单步执行时, 在“PINSEL1 = 0x00000000;”后总是出现提示框

“No disassembly could be read at the requested address.”

A: 您是否是用从 JTAG (次要 JTAG) 调试, 如果是, 第二句改为 PINSEL1 = 0x55400000。

第429问:

Q: 请问如果只采用从 JTAG 调试, 而不是用 J6 和 J7 从主方式切换到从方式, 可以吗?

我试过先把 J6 跳到 PCSEL, J7 跳到 PCDBG, 并在 AXD 里设置 TAP2, 可以在 MEMORY 窗口看到 PINSEL1 = 0x55400000, 这时蓝色箭头指向启动代码, 取出 J6、J7 程序进入从 JTAG 调试, 一切正常, 但是如果一开始 J6、J7 就取出, 好象 EASYARM2104 这块板子无法与 AXD 通讯, 似乎必须先主后从? 难道不能直接从 JTAG 调试吗? 《ARM 微控制器基础与实战》的附录 2 最后提到了 ISP 先烧一个程序到 FLASH, 不太明白, 烧到什么地址? 而且用 ADS 调试的时候代码不是要从 JTAG 烧到 2104 的 FLASH 或 RAM 里吗? 这时好象必须是主 JTAG? 而且怎么知道执行的是先用 ISP 烧进去的代码还是后来的? 我都晕了, 还望高手赐教!

A: 进入方法:

如果使用 PC 控制主/从 JTAG:

1. JP6、JP7 分别跳到 PCSEL、PCDBG, 因为 EasyJTAG 要控制主/从 JTAG 口切换 (不用再取出)。

2. 若在 AXD 里设置 TAP2, EasyJTAG 会先使用主 JTAG 控制, 写入 PINSEL1 = 0x55400000, 后才切换到从 JTAG (控制硬件切换, 即通过 PCSEL、PCDBG 控制)。

如果一开始就取出 JP6、JP7 跳线, EasyJTAG 根本不能控制 PINSEL1 = 0x55400000。是先主后从。

第430问:

Q: 怎样使用 P0. 22、P0. 23、P0. 24?

在主 JTAG 方式下, P0. 22~24 被作为流水线状态, 不能作为 GPIO 使用。按照说明书中所讲, 在我的程序启动后, 将 PINSEL1|=0x55400000 以后, P0. 22~24 即可使用。我这样做了, P0. 22~24 好像还是不能被我作为 KEY4、KEY5、KEY6 使用。

A: JTAG 方式下对那些位的设置无效, 所以不能用

Q: 在《基础与实战》P277 上 4. 4. 4 讲, 在 JTAG 选择主从时, 可以编程实现从 JTAG, 如何理解?

A: 使用从 JTAG 方式, 需要配置管脚。

第431问:

Q: 请教: LPC2104 开发板, P0. 22 总是输出低电平, 而 P0. 23、P0. 24 总是高电平, 无论怎样都改不了他们状态, 为什么? 测试主程序如下:

```
int main(void)
{
    PINSEL0 = 0x00000000;    // 设置所有管脚连接 GPIO
    PINSEL1 = 0x00000000;
    IODIR   = 0xFFFFFFFF;   // 所有管脚都为输出
    IOSET   = 0xFFFFFFFF;   // 输出全置 1, P0. 22 仍为 0
    IOCLR   = 0xFFFFFFFF;   // p0. 23, p0. 24 仍为 1

    while(1);
    return(0);
}
```

A: JTAG 占用这些口, 调试状态不能用, 看一看配套《ARM 微控制器基础与实战》附录 2。P0. 22~P0. 26 都属于主 JTAG 接口, 只要你的 JTAG 线缆插上了, 这些管脚的电平就由 JTAG 仿真器来决定。

解决方案是在程序的开始运行 PINSEL1 = 0x55400000 将主 JTAG 切换到从 JTAG, 那么 P0. 17~P0. 26 就全都可以作为 GPIO 或者其第二功能使用。

Q: PINSEL1 = 0x55400000, 就能将主 JTAG 模式切换到从 JTAG 模式吗? 关于这个切换, 哪里有详细介绍?

A: 如果使用 PC 控制主/从 JTAG:

1. JP6、JP7 分别跳到 PCSEL、PCDBG, 因为 EasyJTAG 要控制主/从 JTAG 口切换。
2. 若在 AXD 里设置 TAP2, EasyJTAG 会先使用主 JTAG 控制, 写入 PINSEL1 = 0x55400000, 后才切换到从 JTAG (控制硬件切换, 即通过 PCSEL、PCDBG 控制)。

如果一开始就取出 JP6、JP7 跳线, EasyJTAG 根本不能控制 PINSEL1 = 0x55400000。是先主后从。

Q: 那么怎样才能仅仅使用从 JTAG 呢?

A: JP6、JP7 短路器取出, 此时只能使用从 JTAG 进行调试。

但有一点要注意, 就是想办法让 PINSEL1=0x55400000, 使从 P0. 27--P0. 31 配置为从 JTAG 口 (可通过运行 FLASH 中的程序设置, 所以要写一段代码放入 FLASH 的 00000000H 起始地址)。

在仿真器配置设置中, 设置为不复位系统。在 AXD 调试环境, 打开“Options->Configure Target...”, 弹出“Choose Target”窗口, 在“Target Environments”框中选择“EasyJTAG...”项。点击“Configure”按钮, 进入“EasyARM Configuration”设置窗口, 在“Options”项中选择 Halt program。

第432问:

Q: 关于 2119 的复位和 JTAG 连接问题。

2119 上电复位, 如果 RTCK 为低, TRACESYNC 为高, 那 P1. 16~p1. 25 作什么用呢? 是 IO 口吗?

A: 作 GPIO。。

Q: 如果 RTCK 为高, TRACESYNC 为低, 那又变成什么情况呢?

A: 理论上 P1. 16~p1. 25 用于跟踪, 而 P1. 26~p1. 31 用于 GPIO。这个配置没有意义, 因为跟踪信号必须与 JTAG 信号配合使用。

Q: 2119 和 easyARM 带的 JTAG 电缆怎么接呢? 和 2104 不一样阿。

A: 把对应的信号与 JTAG 仿真器相接即可。注意那两个引脚的电平。

Q: P1. 25 作什么用呢? 好像不属于跟踪端口。

A: P1. 25 为跟踪信号。

第433问:

Q: lpc2104 管脚里的 跟踪包位 流水线状态 有什么用啊?

A: 有些 JTAG 仿真器需要用到, 以实现 ETM 实时跟踪功能, 参考 LPC2104 的数据手册 ETM 说明部份。

Q: 在硬件上有什么要求? 只需要 P0. 14 接地就可以了吗? 是不是和 EASYARM 一样, 用一个跳线器控制 P0. 14 就行了。这样子是不是能直接通过 LPC210X ISP 软件环境进行程序下载了? 怕板子画出来不能下载程序, 那就没用了。因为采用的是从 JTAG 方式。

A: 是的, P0. 14 接地, 复位后会进入 ISP 状态, 使用 UART0 进行 ISP 操作。

第434问:

Q: 请问: 2114 的 JTAG 接口的第 17, 19 脚是悬空的吗?

A: 是。

第435问:

Q: 请问《ARM 控制器基础与实践》书中 附录 B:

JTAG 接口设计指南中,讲到用主 JTAG 口调试系统时,占用 P0.17~P0.31 共 15 个引脚,ETM 调试接口与主 JTAG 接口联合使用,ETM 调试接口与主 JTAG 接口的所有引脚将被占用。在 LPC2114 用主 JTAG 口调试系统设计时,对 ETM 有关的引脚在调试时也应当留出吗?LPC2114 制版时,直接用 P1.26~P1.31 与 EasyARM2104 开发装置的 JTAG 口直接相连是否可以实现在 ADS 下调试。

A: LPC2114 只有一个 JTAG 口,芯片复位时通过 RTCK 脚设置使能/禁止 JTAG 口,当使能 JTAG 时不会强行使能 ETM 有关的引脚(ETM 有关引脚由另一个 I/O 设置使能/禁止)。

第436问:

S: 使用 JTAG 调试目标板时,如果用户程序在调试过程中改变了 JTAG 口的配置,则可能导致 JTAG 自锁发生,现象是无法继续用 AXD 调试,马上复位或者短暂掉电重启也不行。此时需要将 JTAG 调试器断电拔下,闲置 10 分钟左右,等待调试器的电全部放光后才能再次使用。

第437问:

Q: 请问: LPC2114 制版时,直接用 P1.26~P1.31 与 EasyARM2104 开发装置的 JTAG 口直接相连是否可以实现在 ADS 下调试。

P1.26 RTCK
P1.27 TD0
P1.28 TDI
P1.29 TCK
P1.30 TMS
P1.31 TRST

A: ARM 微控制器基础与实战 P26 页 HEADER 10X2 引线对应,其中 17、19 脚不用。

第438问:

Q: 请教: lpc2104 的 JTAG 主/从模式有什么区别啊?想明白 JTAG 需要看点什么?

A: 1. 主 JTAG 只需要将 DBGSEL 接高电平,即可使能主 JTAG 口;

从 JTAG 需要 FLASH 中有可运行的程序,将 PINSEL1 设置为 0x55400000,才能使能从 JTAG 口。

2. 主 JTAG 具有 ETM 跟踪信号引脚(但 EasyJTAG 没有使用这些引脚),从 JTAG 则没有。

第439问:

Q: EasyJTAG 和其他市面上面卖的 ICE 有什么区别?

A: EasyJTAG 可以直接下载程序到 LPC2106/2105/2104 的片内 FLASH 进行仿真调试。

EasyJTAG 只能仿真 LPC 芯片。

第440问:

Q: 1. EasyJTAG 昨天晚上正常,但是今天打开就出现了如下警告,无法运行,不知为何?

RDI Warning 00254: Unimplemented RDI message

2. 还有, ZLG 的配套《ARM 微控制器基础与实战》上说: easyARM configuration 中的 flash 在需要的在 DEBUG 时要选择:

Not Erase Flash

但是我一选定, 按 OK 后在打开就又回到了:

Erase Flash When Need

不知道是为什么?

3. 在 CodeWarrior for ARM Developer Suite 中编译链接后, 再按 F5 进入 AXD, 假如在 AXD 中发现问题, 回来修改代码, 然后再编译链接, 按 F5 又进入了另外一个 AXD, 是这样的吗?

如果是这样的话那不是太麻烦了吗? 是不是需要哪里进行设置?

- A: 1. 可能使电源插座松了;
2. 应该选 Erase Flash When Need。这可能是您没有把驱动程序的文件的只读属性去掉。
3. 是, 这没有办法。

第441问:

Q: 请问, 我买的 2104 的 EasyJTAG 能直接用在 211x 上面吗?

A: 请下载升级软件!

第442问:

Q: 请问各位: 用从 JTAG 仿真是不是要事先将程序 ISP 到芯片中?

A: 是, 因为需要程序设置 PINSEL1 为 0x55400000。

第443问:

Q: 我原来的目标板是用的主 JTAG 口, 一切 OK。重新排版将主 JTAG 改成了从 JTAG 口, 其他未改。

DBOSEL 接地, 程序中 P0.27~P0.31 配置成从 JTAG。

现在程序都无法装载, 提示:

Warning: Interrupt vectors data is not correct!

我该怎么办?

A: 首先芯片要烧写有配置 PINSEL1 的程序。

第444问:

Q: 我用 2119 作了块板子, 主要是为了调 CAN, 但是 JTAG 怎么不能用呢? ADS 是否需要新的驱动呢?

还有就是我的复位电路不能用, 所以没有用复位电路, 这样对 JTAG 的连接有没有影响?

我是将 RTCK 直接加 4.7k 拉低的, 还有就是 ARM 的 REST 脚我也是直接到高电平的, 这样行不行?

A: 仿真器中有上拉电阻, 所以加 1k 拉低。您最好使用复位芯片。

第445问:

Q: 我在做自己的板子, 我将 TRST, TDI, TMS, TCK, RTCK, TDO, RST 与 LPC2114 的对应脚直接相连了, 不知道对不对? 还有! 我不知道 JTAG 的 PCPRISEC (17 口) 和 PCDBGSEL (19 口) 该怎么接?

A: 对, PCPRISEC 和 PCDBGSEL 不需要连接。

第446问:

Q: 用 JTAG 调试, 出现“RDI Warning”提示, 我查了一下 RDI 的翻译是 remote debug interface。不知为什么出现这种情况。

A: RDI Warning 估计是并口没连好的问题。

第447问:

Q: 我现在在 2214 在用的仿真器仿真 2104! 外扩 FLASH SST39VF320 不行, 不知道为什么!

A: 地址线错位连接, 即 CPU 的 A0 不要连接。

第448问:

Q: 请问, 能否用 LPC2119 试验板的 JTAG 的连接方式仿真 LPC2212 的相同的功能和完成程序下载到 FLASH 中?

A: 相同部份功能是可以的。

第449问:

Q: 请教: LPC2100 中 JTAG 仿真的问题。

我用的是 ADS1.2, 在使用 RelInFLSASH 的时候, AXD 给如下提示:

“Flash Sector 0 write failed!

Warning! interrupt vectors data is not correct!

Program you downloaded can not run freely!”

不知道如何解决。

A: 可能是 2114 的加密造成的。

第450问:

Q: 2200 开发套件是否可以通过 EasyJTAG 烧录 Flash?

如果 2210 可以, 其他的 LPC22xx 是否也可以?

A: 新的驱动程序支持烧写 EasyARM2200 的外部 FLASH, 对别的板暂时不支持。

驱动程序正在测试, 还不提供下载。

第451问:

Q: 如何在自己的板子上运行 JTAG 调试功能? 我现在有一个以 LPC2104 为核心的设计。

参考光盘上的原理图我搭建了一个 JTAG 接口电路。

请问, 如果我用通用的 ARM7TDMI (不是 ARM7TDMI-S) 的 JTAG 下载板能否调试 LPC2104? 还是只能用开发板自带的 ARMJTAG 仿真器才能调试? 我看了 Multi ICE 的文档, 一头雾水!

A: 可以, 不过在 FLASH 中调试很麻烦。

第452问:

Q: 我用 2104 的片子, 程序大后用 JTAG 好像没法调试, 不知道用仿真器能行吗?

A: JTAG 仿真是使用芯片内部的 FLASH 或 RAM 装载程序, 对于 LPC2104, 片内 FLASH 可以使用 120kB 空间。

第453问:

Q: 有几个问题向您请教:

1. 2114JTAG 接口的 RTCK 引脚要下拉 4.7k, 但是 2104 却要上拉 10k, 为什么不一样呢?
2. 2114JTAG 相连的 20 个双排针的第 17 脚, 在 2104 板子上是主从 JTAG 选择用的, 那么在 2114 中是不是就不用了, 可以悬空吗?
3. DBGSEL 引脚要加 100 欧电阻, 是限流用吗? 其他的引脚不用加吗? 还有 DBGSEL 引脚从 JTAG 出来之后, 要经过 74hc125 去抖动, 如果不这样的话是不是也可以工作?

A: 1. LPC2114 复位时通过检测 RTCK 脚, 判断是否使能 JTAG 口, 而 LPC2104 则不是这样。
2. 悬空。
3. 是(保护作用, 可以去掉)。不要也可以。

第454问:

Q: 请问现在的 JTAG 连接, 电脑这边都是用并口的吗? 有没有用 USB 接口的呢?

A: 目前市场上并口的、USB 接口的都有。

第455问:

Q: 请问 ZLG, 我前一块 2119 的板子参照 2104 在 JTAG 口前加了一块 74HC244, 下载有 ucos 的程序时, 有点问题, 没 ucos 时正常, 我买的 2100 开发板上没有 74HC244, 我们是不是也直接接 JTAG, 不用 74HC244?

A: 可以不用 74HC244?

第456问:

S: “2100 开发板与 IBM 笔记本相接总是报错”之解决法。

如 Shanyongde 所言将并口设为 “ECP”、并将其地址设为 378 即可。

我原将并口设为 “ECP” 后, 未注意地址是 3BC; 改为 378 即好。

第457问:

Q: 请问 2104 开发板中的电路如何应用到 LPC2214?

A: 注意 JTAG 口, 需要使能 LPC2214 的 JTAG 口时, RTCK 脚要接一个下拉电阻(如 4.7k)。LPC2214 只有一个 JTAG 口, 没有主、从 JTAG 之分。

还有, LPC2214 支持外部 FLASH 起动, 所以要正确配置 BOOT0 和 BOOT1 引脚(参考芯片使用手册)。

第458问:

Q: 使用从 JTAG 接口遇到问题。

我在使用从 JTAG 时总是出问题。将 JP6、JP7 跳线取掉, 进入 AXD, 选择 TAP2。一装载 Image, Debug Log 显示告警:

DBT Warning 00053: image shorter than implied by header[has it been squeezed?]. 无法进入调试状态。我已经在 FLASH 中写了个程序, 其中有一句:“PINSEL1=0x55400000”。但我将 JP6 跳线短接到 PCSEL, JP7 短接到 PCDBG。再重新进入 AXD, 选择 TAP2, 装载 Image。又一切正常。

我是这样想的---在 FLASH 中写一程序, 将 P0.27-P0.31 连接到从 JTAG (所以有 PINSEL1=0X55400000 这一句)。将 LPC2104 的 DBGSEL 引脚接地(所以将 JP7 跳线取掉)。选通 U11(所以将 JP6 跳线取掉)。复位系统, 运行 FLASH 中的程序。然后直接进入 AXD 使用从 JTAG 接口调试程序。但出现前面提到的问题。

- A: 1. 要确保 FLASH 中的程序已运行。
2. 在 AXD 中的仿真器设置中, Options 项选用 Halt program。

第459问:

Q: 怎样才能自动选择主从切换? 我在用 EASYARM 调试时, 根据《ARM 微控制器基础与实战》上将 PINSEL1 设置成从 JTAG, 而 JP6 和 JP7 分别接 PCSEL 和 PCDBG, 一直都是用主 JTAG, 只有将短路块取掉才进入从 JTAG, 怎样才能自动切换呢?

A: 使用 Config 不能改变到 Tap2。

Q: 我在配置里将 Tap1 改为 Tap2, 并点击 OK, 但是再看时它又变回 Tap1, 不知道是什么原因, 改 option 也是一样, 而且在 choose Target 那里一旦点击 OK, 则程序重载, 而修改的选择项也不存, 麻烦给点指点!

- A: 1. 使用从 JTAG 口: JP6 和 JP7 分别接 PCSEL 和 PCDBG, 在 AXD 里仿真器设置 Tap2。
2. 如果去除了文件的只读属性, 那么正常情况下是能保存的, 重新安装 ADS 试试。

第13章 其它

第460问:

Q: 问一个关于做产品的问题, 如果选用 LPC2200 系列做产品, 如何保证电磁兼容? 好像不容易过!

打算用于 POS 机, 所以抗干扰问题不是很大, 关键是辐射骚扰, 我觉得可能不太好处理! 周工有没有什么好的意见?

A: 我们用飞利浦的 ARM 和普通 51 的开发板接液晶, 用 ARM 可以承受更高的群脉冲电压, 所以 ARM 抗干扰比普通 51 好。

第461问:

Q: 2100 实验板上有一事不明, 请指教。

LPC2119 的供电引脚一共七个, V3.3A 一个, V3.3 三个, V1.8A 一个, V1.8 两个。可是为什么有那么多的接地电容, 一个供电引脚接一个不够?

还有在 V3.3 与 V3.3A 之间, V1.8 与 V1.8A 之间有跳线, 什么时候需要对它供电, 什么时候不需要?

每个供电引脚分别管辖哪些 I/O 引脚, 不用的引脚我也就不给他们供电了, 不知这样的说法对不对?

A: 把电流表串在跳线的位置就可以量电流了 (测量功耗时使用)。

第462问:

Q: 我在 <http://www.zlgmcu.com/tools/kaifaban/EasyARM2104.asp> 下载了“LPC2100/LPC2200 系列 ARM 微控制器 ISP 软件 (2004-5-31)”, 安装后一运行就出现提示:

“Run time error '383'; 'Text' property is read-only”。为什么?? 我的操作系统是 Win2000。

A: 如果安装过 LPC2106 的 ISP 软件 (早期的软件), 可能会出现冲突。需要卸载原 LPC2106 的 ISP 软件, 并在注册表中删除关联项。

第463问:

Q: 请问一开发板上 R11, R12, R13 三个电阻要不要焊上?

A: R11, R12, R13 都是要的, 100 欧电阻, 生产时都是用贴片机生产的。不用取下来。

第464问:

Q: 请问, 我想用 LPC2114 去连接一个 320240 的液晶屏, 但是一般的液晶屏都是 5V 供电的, I/O 口也是。但是 LPC 的 I/O 口电压是 3.3V 的, 不知道是否可以使用, 如果可以请指教。谢谢。

A: 可以, 连接口线上串一个小电阻 (如 220 欧) 即可。

第465问:

Q: 我前段时间买了一套 EasyARM2104 开发板, 把那本配套手册都翻烂了, 都不知道怎么才能那些例程在板子上跑起来! 真是非常郁闷! 请周立功公司的工程师给予指点!

A: 请按如下步骤操作:

1. 安装 ADS1.2。按装 ADS 提示安装, 与其它软件安装操作方法基本一致。
2. 了解 ADS1.2 集成开发环境的使用。
参考 <http://www.zlgmcu.com/download/downs.asp?ID=1009> 的文档。
3. EasyJTAG 一端连接开发板, 另一端连接 PC 机并口, 并给开发板上电, 安装 EasyJTAG 驱动程序, 参考光盘上 easyarm_drive\readme.txt 或 <http://www.zlgmcu.com/download/downs.asp?ID=1009> 的文档。
4. 将光盘上的“实验程序”目录复制到硬盘, 并把该目录下的所有文件去掉只读属性 (因为从光盘上复制到硬盘时会有只读属性)。
5. 运行 ADS1.2 集成开发环境, 用 File->Open 打开硬盘上的“实验程序\Gpio_TEST\LEDCON\LEDCON.mcp”
6. 按 F7 键编译连接项目, 按 F5 键启动 AXD, 进入调试环境。
7. 如果进入 AXD 有出错, 处理方法参考 <http://www.zlgmcu.com/download/downs.asp?ID=1009> 的文档。然后重新加载调试文件 *.axf (如“实验程序\Gpio_TEST\LEDCON\LEDCON_Data\DebugRel\LEDCON.axf”)。
8. 单步或全速运行程序, 观察结果。

第466问:

Q: 请问 ISP 里晶振不能设置 11.0592, 而是 11059。

A: 是的, 没关系的, 设成 11059 就可以了。

第467问:

S: 我研究了一下这方面的问题, 体会如下, 请大家指教

1. 如果 3.3V 器件说明可以接 5V 电平输入, 说明该端口有 ESD 保护电路, 无需担心损坏的问题, 穿一个电阻是为了更保险。
2. HCOMS 电路 (如 74hc244) 等的电源可以 2-6V, 所以可以用 3.3V 作为电源, 其电平与 LPC 的 GPIO 完全兼容 (试验板上就是这么用的)。
 1. TTL 电路的电源为 4.75-5.25V, 因此不能接 3.3V 电源供电。
 2. 若 TTL 电路 (5V 系统) 与 LPC 接口, 他们的电平在适配范围内, 可以直接连接。
 3. HCOMS 电路 (5V 系统) 与 LPC 接口, 他们的电平不适配, 不能直接连接, 必须接电平转换电路。

第468问:

Q: 求助! ARM 学习陷入困境, 好象不得门其入。

我现在用的是 2104 的开发实验板, 一个多月了, 《ARM 微控制器基础与实战》的前 5 章已经过两遍了, 现在感觉只能看懂最简单的汇编程序, 书上实验程序那四个启动文件一

直没有理解，还没有自己写过一个程序。

请问我是否可以把启动过程放一放呢？请给出你的建议或学习心得体会。

A: 启动文件需要对核心、编译器和具体芯片非常熟悉才能编写出来，可以先放一放。