PIC单片机查表指令安全性分析初探*

■ 上海工程技术大学　李荣正

美国Microchip公司推出的PIC系列单片机，由于采用RISC精简指令集、哈佛总线结构、流水线指令执行方式，并且抗干扰能力强、性能价格比高等优点，深受世界各行各业的普遍欢迎。在实际外围接口方面，特别是在系统监控程序设计中，常常需要引入数据表格。由于PIC单片机并不拥有现成的查表指令，而必须由用户自己设计构造相应的功能。但在查表程序的规划中，如果稍不注意就会在程序的执行过程中产生莫名其妙的程序飞溢。本文将对这种现象进行剖析和讨论，并结合实例给出一个比较严密和安全的解决方案。

1　PIC单片机查表指令结构分析

一般查表子程序所涉及的问题主要有两个方面：即所处的起始地址和表数据结构，它们在一定程度上都可能导致程序飞溢。首先讨论常用查表子程序的结构，特别是对所涉及的要素进行分析。

查表子程序的核心语句是ADDWF　PCL，F，通过加法指令将当前程序指针用一个偏移量（W）加以修正，并以此作为下一条指令执行的方向。但根据PIC单片机特有的指令功能，若发生以PCL为目标指令执行过程，将会出现虚拟高8位PCLATH的加载过程。这个问题和PIC单片机程序存储器页选方式一起，常常给设计者带来不便。

本文以PIC16F877单片机为例，它的程序存储器（Flash）是一个具有空间为8KB×14位的存储器，其中14位为单元字节宽度。为了能完全选择8KB的程序存储器，需要合成13根地址选择线。涉及程序存储器内指令语句的选择，主要有两种途径和方式。一是当执行完以PCL为目标地址的算术逻辑运算类指令后，将可能改变下一条指令的方向。此时下一条指令13位地址的构成是这样，以PCL的运算结果为低8位，而高5位将由虚拟寄存器$PCLATH_{0-4}$装载。二是执行CALL和GOTO跳转指令，从理论上说可以跳转到程序存储器的任何地址。在其指令机器码中，操作数理应携带13位目标地址，但由于F877指令系统的机器码宽度只有14位，对应的指令操作码占3位，指令机器码将只能隐含跳转方向的低11位（$PC_{0\sim10}$）地址，是目标地址不完整参数。通过11位地址的寻址范围是2KB，即表示在当前2KB程序存储器范围内进行转移和调用子程序，不会出现什么问题。如果超出2K的范围，将要求PCLATH中的二位（$PCLATH_{3\sim4}$）预置相应的数值，由其确定程序跳转方向的高二位（加载$PC_{11\sim12}$）信息。一般把8KB程序存储器分成四个区域，每一个区域为2KB，在PIC中被称为“页”（page）面。当跳转的范围超出2KB程序存储器空间，就需要$PCLATH_{3\sim4}$作为程序存储器的页面选择。在两类指令的执行过程中，都会出现PCLATH对程序指针高5位（或2位）的装载效能，如图1和图2所示。

从以上分析可以看到，跳转到程序存储器的某一个单元指令，不管采用哪一种方式，首先必须根据该单元指令的所在地址，对PCLATH寄存器中的$PCLATH_{0\sim4}$进行预置。尽管可以通过汇编系统补充的伪指令语句，即页选指令：PAGESEL进行模糊跳转，但由此带来的问题也是始料不及的。

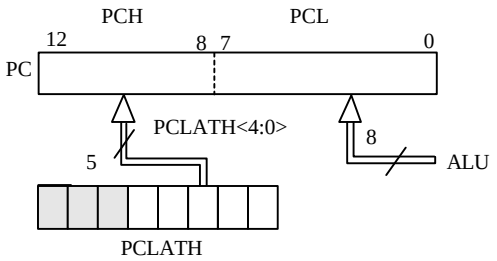

图1　执行PCL为目标地址指令

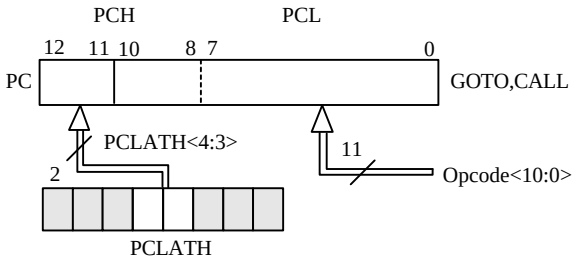

图2　执行跨页跳转和调用指令

* 本项目研究框架系上海市教育委员会基金项目（03FK16）

2 几种引起程序飞溢情况的分析

由于PIC单片机没有常用的查表指令，必须借助于特殊语句完成这一功能。然而，在使用中如果稍不留意，就很容易出现程序飞溢。在调用查表程序时应注意什么呢？ 选用下面一个程序实例进行说明。已知主程序位于页0区域，如果将查表子程序设置在0080H，对于16个查表数据是不会出现飞溢现象，子程序片段如下：

```
        ORG     0080H
CHABIAO ADDWF   PCL,F ；增加偏移量
        RETLW   00H   ；第0个数据
        RETLW   01H   ；第1个数据
        ……
        RETLW   0FH   ；第15个数据
```

在本例中，选择子程序地址是0080H，子程序的调用将不会出现程序飞溢。但若改变子程序地址信息，将可能导致源程序出现莫名其妙的飞溢情况。下面分几种情况进行讨论。

2.1 目标地址高5位发生变化

查表子程序地址的选择是很讲究的，如果不加思考紧跟在主程序后面，有时也会出现程序飞溢现象，这对熟练的技术人员也可能是一件头疼的事情。假定紧跟在主程序后面的隐含地址是00F9H，那么在调用子程序时，对于16个数据查表（W=00H-0FH），有时执行正确，而有时将出现程序飞溢。实际上，主要是因为指令CHABIAO ADDWF PCL,F在执行过程中，PCL（当前指令地址的低8位）+W（偏移量）产生溢出进位所致。对于这个问题，可以有两种办法进行弥补：一个是确保CHABIAO ADDWFPCL,F的指令地址和目标转移地址的高5位一致；二是在调用CHABIAO子程序前，对于那些可能出现PCL+W（06H-0FH）溢出的情况，PCLATH必须进行预置。程序说明如下：

```
ABC  EQU    06H       ；设置符号常量
CLRF PCLATH           ；初值为零
     SUBLW  ABC       ；判断W是否≥06H
     BTFSSSTATUS，C   ；C=1，W＜06H，PCLATH为0
     INCF   PCLATH    ；C=0，W≥06H，PCLATH为1
     CALL   CHABIAO；调用CHABIAO子程序
```

表面上看源程序的调用空间并没有发生质的变化，还是位于第0页面。但其微观的指令指针却出现了不协调，作者认为正是由于程序指针的“表里不一”，最终导致出现程序飞溢。因为在CALL指令中，操作源代码携带目标地址的低11位信息，高2位表示页面情况，是应由PCLATH人为设置。但考虑到本例并未超出“0”页面，所以PCLATH理应不予变化。主要症结在于PCL+W产生溢出，PCLATH中的数据未得到及时更新，因而利用PCLATH加载后的程序指针就不再有效。譬如，当W=06H时，执行指令CHABIAO ADDWF PCL,F后，当前程序指针理应调整到00FAH+06H=0100H，但实际上此时$PCLATH_{0\sim4}$均为0，由$PCLATH_{0\sim4}$加载后形成的目标地址却是0000H，转移路径不是希望的。

2.2 程序存储器超出2K的页面范围

如果将查表子程序设置在非页0区域，根据PIC单片机宏指令规则，可以通过补充宏指令PAGESEL进行页面切换。如设计表地址ORG 0800H，当然在调用子程序以前引用PAGESEL语句，可以很顺利地完成查表任务。但若把查表子程序地址稍加改动，如改到地址ORG 0900H，情况就不一样了，程序将无法运行下去。这是为什么呢？要回答这个问题，首先必须对PAGESEL语句进行剖析。

PAGESEL语句对于初学者有很大帮助，但在使用前一定要熟悉其功能，它究竟等效于什么指令。实际上，PAGESEL仅仅承担程序存储器页面的调整功能。譬如，CHABIAO子程序位于0800H处，那么PAGESEL CHABIAO的作用是将当前程序存储器页面转到页面“1”，即替代指令BCF PCLATH，4和BSF PCLATH，3的功能。显然，通过PAGESEL CHABIAO是无法改变$PCLATH_{0\sim2}$的数值。由于查表子程序设置在ORG 0800H和ORG 0900H处，通过PAGESEL CHABIAO对PCLATH产生的结果是一样的。那么，在执行完指令CHABIAO ADDWF PCL,F后，将执行的下一条指令就出现争议了。假定此前W=2，在执行CHABIAO ADDWF PCL,F后，显然应该分别转去执行0803H处或0903H处的指令。然而，通过反复程序调试，无论如何都会转到0803H处的指令。主要原因在于当程序转到0900H时，虚拟高8位中$PCLATH_{0\sim2}$没有得到及时的更新。不要以为这个问题是由于跨页面调用程序造成的，实际上，在页面“0”同样也会有这种现象。例如，查表子程序设置在ORG 0400H和ORG 0500H处，虽然没有跨页面调用子程序，但也会出现程序飞溢。

进行专项多角度研究表明，为了有效避免这种不可预见性的错误，最有效的方法就是依照查表子程序处的单元地址对$PCLATH_{0\sim3}$预先进行人工设置。当然，还必须依托页面调整伪指令的配合（如果出现跨页调用），只要查表数据不超过256个，就可以确保不会出现飞溢现象。

2.3 查表超出256个数据范围

当查表范围超过256个数据时，即使很注意安排查表子程序的起始地址也将不可避免地发生溢出现象。此时，在关注工作寄存器W的同时，还必须留意 79

```
stir0,*ar0
ldi020h,r0
ldp@serialportx
ldi@serialportx,ar0
stir0,*ar0
bu main
```

经过编译、连接、格式转换以及写入Flash之后，系统就可以实际脱机运行了。加电一段时间后，可以通过示波器测量得到DX0端口的均匀脉冲波形，证明引导装载成功。

结 语

通过DSP对外部Flash编程，从而实现DSP的BOOTLOASDER在整个DSP嵌入式系统开发中的重要作用。如果开发者在开发之初就掌握这项技术，就会大大方便系统的调试，缩短开发时间。有感于国内TMS320C3X系列BOOTLOADER中文资料的缺乏，特作此文。由于篇幅所限，本文仅给出部分核心程序代码，需要整个工程文件和源程序的读者请发E-mail到：dreamcolin@163.com，本人愿意与大家共同交流。

参考文献

1 TMS320C3x User's Guide(literature number SPRU 031)

2 TMS320C3x General Purpose Applications User's Guide (literature number SPRU194)

3 TMS320C3x/C4x Assembly Language Tools User's Guide (literature number SPRU035)

4 TMS320C32 Data Sheet

5 Flash AM29F400B Data Sheet

6 Cypress CY7C109 Data Sheet

7 公茂忠，刘汉魁，徐殿国． Flash存储器的在系统编程及其在DSP系统中的应用． 电子技术应用， 2002(3)

8 张雄伟． DSP芯片的原理与开发应用．第3版．北京 北京航空航天大学出版社，2003

朱显新，邓启辉：硕士研究生，主要研究方向信号处理、嵌入式系统应用。黄涛，卢珞先：副教授，主要研究方向信号处理、嵌入式系统应用。

（收稿日期：2004-03-16）

69 虚拟高5位$PCLATH_{0\sim4}$数据，应及时修正刷新。当W溢出（超过256）时，确保PCLATH加1。这样就可以处理超过256个数据的查表工作。在选择表格地址的时候尽可能取低8位是00H，至少可以保证在256个数据查寻之内不会出现程序飞溢。由于引导语句CHABIAO ADDWF PCL,F本身占有一个字节，实际仅利用W偏移量可进行访问的数据严格说只能是255个。因为当偏移量W=0FFH，加上引导语句CHABIAO ADDWF PCL,F下一条地址的低8位（01H）时，已经出现溢出，已经要涉及$PCLATH_{0\sim3}$修正问题。下面给出调用查表程序之前的处理程序，假定计数工作由两个符号变量（AB0、AB1）担任，根据数值对应查找某一个参数：

```
PAGESEL CHABIAO   ; 首先调整到子程序的页域
MOVF   AB1,W      ; 高位数据修正PCLATH
ADDWF  PCLATH,F
MOVF   AB0,W      ; 高位数据确定查表位置
CALL   CHABIAO    ; 调用CHABIAO子程序
```

符号变量AB0和AB1分别表示计数的低8位和高8位，通过AB1修改寄存器$PCLATH_{0\sim2}$，从而达到调整程序地址的高位指针，满足查表超越255个数据的限制。

结 语

本文介绍一种基于PIC单片机的常用安全查表指令，提出讨论几种易于混淆、会引起程序飞溢的情况，并给出相应解决方案。这对提高PIC单片机的实用性和运行的可靠性，有一定的实用价值。为了有效避免程序飞溢，建议查表子程序尽可能设置在程序存储器低8位地址为00H处，并可依照查表子程序单元地址对$PCLATH_{0\sim2}$预先进行人工设置。对于超出256个查表数据，可以通过两个符号变量及时修正$PCLATH_{0\sim2}$，从而确定查表位置。

参考文献

1 刘启中．PIC单片机原理及应用．北京：北京航空航天大学出版社，2003

2 李荣正．PIC单片机习题与解答．北京：北京航空航天大学出版社，2003

3 李学海．PIC单片机实用教程——基础篇．北京：北京航空航天大学出版社，2002

4 李学海．PIC单片机实用教程——提高篇．北京：北京航空航天大学出版社，2002

5 刘和平．单片机原理及应用．重庆：重庆大学出版社，2002

李荣正：硕士、副教授，主要研究方向计算机控制技术。

（收稿日期：2004-02-04）

PIC单片机查表指令安全性分析初探

作者：李荣正
作者单位：上海工程技术大学
刊名：单片机与嵌入式系统应用
英文刊名：MICROCONTROLLERS & EMBEDDED SYSTEMS
年，卷(期)：2004，""(6)
被引用次数：2次

参考文献(5条)

1. 刘启中 PIC单片机原理及应用 2003
2. 李荣正 PIC单片机习题与解答 2003
3. 李学海 PIC单片机实用教程-基础篇 2002
4. 李学海 PIC单片机实用教程-提高篇 2002
5. 刘和平 单片机原理及应用 2002

引证文献(2条)

1. 王卓.杨学友.李恭 基于RGB三基色原理的手持式颜色检测仪的设计[期刊论文]-天津科技大学学报 2006(1)
2. 张光.李学仁 一种基于PIC16系列单片机的防程序飞溢编程方法[期刊论文]-现代电子技术 2005(14)