

[19] 中华人民共和国国家知识产权局



[12] 发明专利申请公布说明书

[21] 申请号 200710193526.6

[51] Int. Cl.

G06F 9/445 (2006.01)

G06F 9/44 (2006.01)

G06F 13/38 (2006.01)

[43] 公开日 2008 年 5 月 21 日

[11] 公开号 CN 101183314A

[22] 申请日 2007.12.11

[21] 申请号 200710193526.6

[71] 申请人 中国科学院长春光学精密机械与物理研究所

地址 130033 吉林省长春市东南湖大路 16 号

[72] 发明人 吕春雷

[74] 专利代理机构 长春菁华专利商标代理事务所
代理人 赵炳仁

权利要求书 2 页 说明书 7 页 附图 3 页

[54] 发明名称

一种实现数字信号处理器程序在线编程的方法

[57] 摘要

本发明涉及一种实现数字信号处理器程序在线编程的方法，该方法首先使用 DSP 仿真器通过 JTAG 接口对 Flash 存储器烧写用于用户程序编程数据的接收、校验及写入的内核程序；其次将数字信号处理器与计算机连接，运行用于计算机与数字信号处理器之间数据交换的在线编程应用软件，将用户程序写入 Flash 存储器；本发明烧写内核程序过程仅需一次，用户程序的写入及更新通过数字信号处理器的串口在线完成，因而用户程序能够很简捷地写入数字信号处理器外挂的 Flash 存储器中，增强了以 DSP 芯片为平台的系统可维护能力。

1、一种实现数字信号处理器程序在线编程的方法，其特征在于包括下述步骤：

(一)、使用 DSP 仿真器通过 JTAG 接口对 Flash 存储器烧写内核程序；

(二)、将数字信号处理器与计算机连接，将在线编程应用软件存入计算机；

(三)、运行在线编程应用软件，将用户程序写入 Flash 存储器；

所述的内核程序用于用户程序编程数据的接收、校验及写入；

所述的在线编程应用软件用于在线编程时计算机与数字信号处理器之间的数据交换。

2、根据权利要求 1 所述的实现数字信号处理器程序在线编程的方法，其特征在于所述的
的内核程序由编程文件接收模块、差错校验模块、接收应答模块、Flash 写入模块、Flash 擦
除模块、完整性效验模块、二次加载模块和控制模块组成；

所述的在线编程应用软件由编程文件载入模块、串口配置模块、控制管理模块组成。

3、根据权利要求 1 所述的实现数字信号处理器程序在线编程的方法，其特征在于所述
的步骤（二）中，数字信号处理器通过串行接口及异步串行通讯芯片与计算机连接；用户程
序输入计算机后，通过串口输出；再经过串行接口电平转换及异步串行通讯芯片串并转换后，
通过数字信号处理器写入 Flash 存储器。

4、根据权利要求 3 所述的实现数字信号处理器程序在线编程的方法，其特征在于串行
接口采用 MAX232；异步串行通讯芯片采用 ST16C550。

5、根据权利要求 1 所述的实现数字信号处理器程序在线编程的方法，其特征在于所述
的步骤（二）中，数字信号处理器通过串行接口、异步串行通讯芯片及可编程逻辑芯片与计
算机连接；可编程逻辑芯片作为 Flash 存储器和异步串行通讯芯片的译码选通，实现 Flash
存储器和异步串行通讯芯片与数字信号处理器的连接；用户程序输入计算机后，通过串口输
出；再经过串行接口电平转换及异步串行通讯芯片串并转换后，通过数字信号处理器写入
Flash 存储器。

6、根据权利要求 5 所述的实现数字信号处理器程序在线编程的方法，其特征在于串行
接口采用 MAX232；异步串行通讯芯片采用 ST16C550，可编程逻辑芯片采用 XILINX

XC2C256。

7、根据权利要求 1 所述的实现数字信号处理器程序在线编程的方法，其特征在于所述的内核程序与用户程序在 Flash 存储器里分开存储。

8、根据权利要求 7 所述的实现数字信号处理器程序在线编程的方法，其特征在于所述的内核程序设置为写保护。

一种实现数字信号处理器程序在线编程的方法

技术领域

本发明涉及一种实现数字信号处理器程序在线编程的方法，特别涉及一种数字信号处理器外挂 Flash 芯片的在线编程方法。

技术背景

近年来，数字信号处理器（DSP）性价比的提高和开发环境的不断完善，使 DSP 在通信、计算机、控制、消费类电子产品等各个领域得到了广泛应用。对于以 DSP 芯片为平台的系统开发，除考虑系统本身的功能、性能实现，还应完善系统的可维护、可扩展能力，保证系统的可持续发展。对应于这种设计要求，开展基于 DSP 程序的在线编程技术研究就成为一项很有必要的工作。

目前，在 DSP 应用系统中，普遍采用 Flash 芯片作为 DSP 的程序存储器。一般来说，将用户程序代码烧写入外部 Flash 存储器的方法有两种：一种是使用通用编程器对 Flash 存储器进行编程；另一种是通过 DSP 的 JTAG 仿真接口对 Flash 存储器进行编程。第一种方法操作繁琐，随着贴片 Flash 的出现，逐步淡出主流设计。在线烧写 Flash 是一种简捷、有效、灵活的方法，但 JTAG 口的编程不能脱离 CCS 仿真环境、编程时需打开设备外壳等因素也很难适应现场编程的要求。

发明内容

本发明要解决的技术问题是提供一种实现数字信号处理器程序在线编程的方法，该方法使用户程序能够很简捷地写入数字信号处理器外挂的 Flash 存储器中，并且用户程序更新灵活。

本发明的实现数字信号处理器程序在线编程的方法包括下述步骤：

- （一）、使用 DSP 仿真器通过 JTAG 接口对 Flash 存储器烧写内核程序；
- （二）、将数字信号处理器与计算机连接，将在线编程应用软件存入计算机；
- （三）、运行在线编程应用软件，将用户程序编写入 Flash 存储器；

所述的内核程序用于用户程序编程数据的接收、校验及写入；

所述的在线编程应用软件用于在线编程时计算机与数字信号处理器之间的数据交换。

所述的内核程序由编程文件接收模块、差错校验模块、接收应答模块、Flash 写入模块、Flash 擦除模块、完整性效验模块、二次加载模块和控制模块组成；

所述的在线编程应用软件由编程文件载入模块、串口配置模块、控制管理模块组成。

所述的步骤(二)中，数字信号处理器通过串行接口及异步串行通讯芯片与计算机连接。用户程序输入计算机后，通过串口输出；再经过串行接口电平转换及异步串行通讯芯片串并转换后，通过数字信号处理器写入 Flash 存储器。

所述的步骤(二)中，数字信号处理器还可以通过串行接口、异步串行通讯芯片及可编程逻辑芯片与计算机连接；可编程逻辑芯片作为 Flash 存储器和异步串行通讯芯片的译码选通，实现 Flash 存储器和异步串行通讯芯片与数字信号处理器的连接。用户程序输入计算机后，通过串口输出；再经过串行接口电平转换及异步串行通讯芯片串并转换后，通过数字信号处理器写入 Flash 存储器。

所述的内核程序与用户程序在 Flash 存储器里分开存储。

所述的内核程序设置为写保护，不能随便擦除，所以在操作过程中遇到突然掉电或误操作引起的擦除、写入不成功也不影响程序再次更新。

所述的内核程序中，编程文件接收模块负责接收来自计算机的用户程序编程文件，经差错校验模块校验无误后，把收到的字节缓存到分配的内存空间，为写入 Flash 存储器做好准备，并且接收应答模块对在线编程应用软件发出应答信号；如果差错校验没有通过，则接收应答模块不做任何应答，等待在线编程应用软件重发数据包。Flash 擦除模块、Flash 写入模块对 Flash 存储器实施在线编程的具体操作，待整个编程文件接收完毕后，启用 Flash 擦除模块、Flash 写入模块完成 Flash 存储器的在线写入用户程序编程文件。写入完成后，由完整性效验模块对 Flash 存储器编程的正确性进行检查，把编程前数据的校验和编程后 Flash 存储器中读出数据的校验进行比较，确认编程是否成功。二次加载模块负责引导 DSP 用户程序的执行。控制模块是内核程序的调度者，通过对设定的在线编程等待时间的监视，控制当前是工作在接收串口信息状态，还是启用二次加载模块。

在线编程应用软件由编程文件载入模块、串口配置模块、控制管理模块组成。编程文件载入模块完成用户程序编程文件的装载，根据文件大小实现文件分片和校验；串口配置模块用于配置串口端口和速率；控制管理模块主要实现在线编程时与 DSP 的交互、发送数据包等功能。

本发明通过深入研究 DSP 的上电引导机制，采用预先使用 DSP 仿真器通过 JTAG 接口对 Flash 存储器烧写内核程序来实现数字信号处理器程序在线编程。对 Flash 存储器烧写内核程序的过程仅需一次，此后用户程序的编写及更新则都通过数字信号处理器的串口在线写入，不再需要仿真器的参与，因而用户程序能够很简捷地编写入数字信号处理器外挂的 Flash 存储器中，增强了以 DSP 芯片为平台的系统可维护能力。

当需要更新程序时，在计算机上运行在线编程应用软件，设置串口端口号和波特率，发出加载指令给数字信号处理器。内核程序实际是一个可以完成 Flash 操作和 DSP 代码加载的内核。DSP 上电后或复位后，利用 DSP 固有的自举功能从 Flash 中加载内核程序到 DSP 内部并开始运行，在设定的时间内，内核程序监视串口的数据，如果检测到符合 Flash 编程数据格式的 DSP 代码，则开始接收代码并缓存到 DSP 片内，待代码全部正确接收后，启动 Flash 编程指令将代码编程到 Flash 中。如果在设定的时间内没有检测到指定格式的代码数据，则内核程序负责搬移 Flash 中的 DSP 代码开始运行。

下面结合附图和具体实施方式对本发明作进一步详细说明。

附图说明

图 1 为内核程序流程图。

图 2 为在线编程应用软件流程图。

图 3 为在线编程时软件操作时序图。

图 4 为数字信号处理器与计算机第一种连接方式示意图。图中 1 计算机，2 串行接口，3 异步串行通讯芯片，4 数字信号处理器，5 为 Flash 存储器。

图 5 为数字信号处理器与计算机第二种连接方式示意图。图中 11 计算机，12 串行接口，13 异步串行通讯芯片，14 数字信号处理器，15 为 Flash 存储器，16 可编程逻辑芯片。

具体实施方式

本发明的实现数字信号处理器程序在线编程的方法具体步骤如下：

首先，使用 DSP 仿真器通过 JTAG 接口对 Flash 存储器烧写内核程序；

其次，将数字信号处理器与计算机连接，将在线编程应用软件存入计算机；

最后，运行在线编程应用软件，将用户程序写入 Flash 存储器；

所述的内核程序流程如图 1 所示，包括下述步骤：

- a、开始；
- b、编程文件接收模块接收来自计算机的用户程序编程文件数据；
- c、差错校验模块对接收的数据进行校验；若数据正确，则把收到的字节缓存到分配的内存空间，并且接收应答模块对在线编程应用软件发出应答信号；若数据不正确，则丢弃数据，接收应答模块不做任何应答，等待在线编程应用软件重发数据包；
- d、判断用户程序编程文件数据是否接收完毕；若是则启动 Flash 擦除模块、Flash 写入模块，完成 Flash 存储器的在线写入用户程序编程文件；若否则返回步骤 b；
- e、完整性校验模块对 Flash 存储器编程的正确性进行检查，确认编程是否成功；若编程成功，则发送文件正确写入 Flash 应答信号；若不成功，则发送文件写入 Flash 失败应答信号；
- f、结束。

所述的在线编程应用软件流程如图 2 所示，包括下述步骤：

- ① 开始；
- ②配置串口端口和速率；
- ③发送请求帧；
- ④若收到应答帧，则转步骤⑤；若未收到应答帧，则返回步骤③；
- ⑤发送数据帧；
- ⑥若收到应答帧，则转步骤⑦；若未收到应答帧，则返回步骤⑤；
- ⑦判断数据是否发送完毕，是则转步骤⑧；否则返回步骤⑤；
- ⑧发送文件发送完毕帧；

⑨收到应答帧，则转步骤⑩；若未收到应答帧，则返回步骤⑧；

⑩显示烧写结果。

所述的数字信号处理器与计算机可以通过第一种方式连接，如图4所示，数字信号处理器4通过异步串行通讯芯片3和串行接口2与计算机1连接。用户程序输入计算机1后，通过串口输出；再经过串行接口2电平转换及异步串行通讯芯片3串并转换后，通过数字信号处理器4写入Flash存储器5。

所述的数字信号处理器与计算机还可以通过第二种方式连接，如图5所示，数字信号处理器14通过串行接口12、异步串行通讯芯片13和可编程逻辑芯片16与计算机11连接；可编程逻辑芯片16作为Flash存储器15和异步串行通讯芯片13的译码选通，实现Flash存储器15和异步串行通讯芯片13与数字信号处理器14的连接。用户程序输入计算机11后，通过串口输出；再经过串行接口12电平转换及异步串行通讯芯片13串并转换后，通过数字信号处理器14写入Flash存储器15。

所述的内核程序与用户程序在Flash存储器里分开存储。

所述的内核程序设置为写保护，不能随便擦除，所以在操作过程中遇到突然掉电或误操作引起的擦除、写入不成功也不影响程序再次更新。

本发明的数字信号处理器可以是美国德州仪器公司的(TI)TMS320C6416 DSP，它不仅具有强大的处理内核，能够完成各种复杂的数据处理任务，同时片内硬件资源充足，可以作为设备通用的微控制器来使用。数字信号处理器外挂的Flash存储器是AMD公司生产的CMOS单电压供电Flash存储器AM29LVO33C，具有低功耗、大容量、擦写速度快等特点，内部嵌入ISP(In System Program)功能，支持在线的实时擦除和再编程操作。通过TMS320C6416与AM29LVO33C的配接，为在线编程系统提供一个通用硬件操作平台。串行接口采用MAX232；异步串行通讯芯片采用ST16C550；可编程逻辑芯片采用XILINX XC2C256。

计算机采用RS232串口标准，通过串行接口MAX232实现电平格式转换，送往ST16C550实现串并数据转换(UART)和产生串口收发中断，通过CPLD完成译码选通，实现ST16C550和AM29LVO33C与DSP的连接，DSP程序存储在Flash芯片AM29LVO33C中。这只是本

发明采用的一个硬件平台，实际上只要 DSP 能操作串口和 Flash 的硬件平台都可以使用。

为保证传输可靠性，计算机与 DSP 串行传输波特率固定设置为 57600kbps，数据格式为：1 个起始位，8 个数据位，1 个停止位。计算机端主动发起连接，在发出一帧数据后开始计时，若在两秒内没有收到 DSP 的正确响应，则重发当前帧，DSP 端被动地接收计算机发出的数据，在收到正确的一帧数据后，根据协议作出相应的响应，如果收到的数据是错误的，DSP 不做任何操作。在这里，随机码用来保证 DSP 和计算机之间的通信是针对当前连接的，每次连接的随机码是不同的。校验位是除帧头、帧尾以外每一帧数据的字节和。为了保证每一帧数据中不出现 7E、7F 帧头、帧尾标记，对数据中出现的 7E、7F 字节采用替换算法，对数据中出现的 7E 用 7D5E 两字节代替，对 7F 用 7D5F 代替，对 7D 用 7D5D 代替。

计算机端数据发送协议如下：

DSP 就绪检测

7E	01H	XXH	XXHXXHXXHXXH	XXHXXHXXHXXH	XXH	7F
帧头	命令字	随机码	代码尺寸	写入 Flash 的位置	校验位	帧尾

发送数据包

7E	02H	XXH	XXHXXHXXHXXH	XXH	7F
帧头	命令字	随机码	数据包在代码中的起始位置	数据	校验位	帧尾

文件发送完毕

7E	03H	XXH	XXH	7F
帧头	命令字	随机码	校验位	帧尾

DSP 端数据响应协议：

可以开始接收新文件

7E	31H	XXH	XXH	7F
帧头	命令字	随机码	校验位	帧尾

已经正确接收数据包

7E	32H	XXH	XXHXXHXXHXXH	XXH	7F
帧头	命令字	随机码	已正确接收字节数	校验位	帧尾

已经正确保存整个文件

7E	33H	XXH	XXH	7F
帧头	命令字	随机码	校验位	帧尾

文件正确写入 Flash

7E	34H	XXH	XXH	7F
帧头	命令字	随机码	校验位	帧尾

文件写入 Flash 失败

7E	35H	XXH	XXH	7F
帧头	命令字	随机码	校验位	帧尾

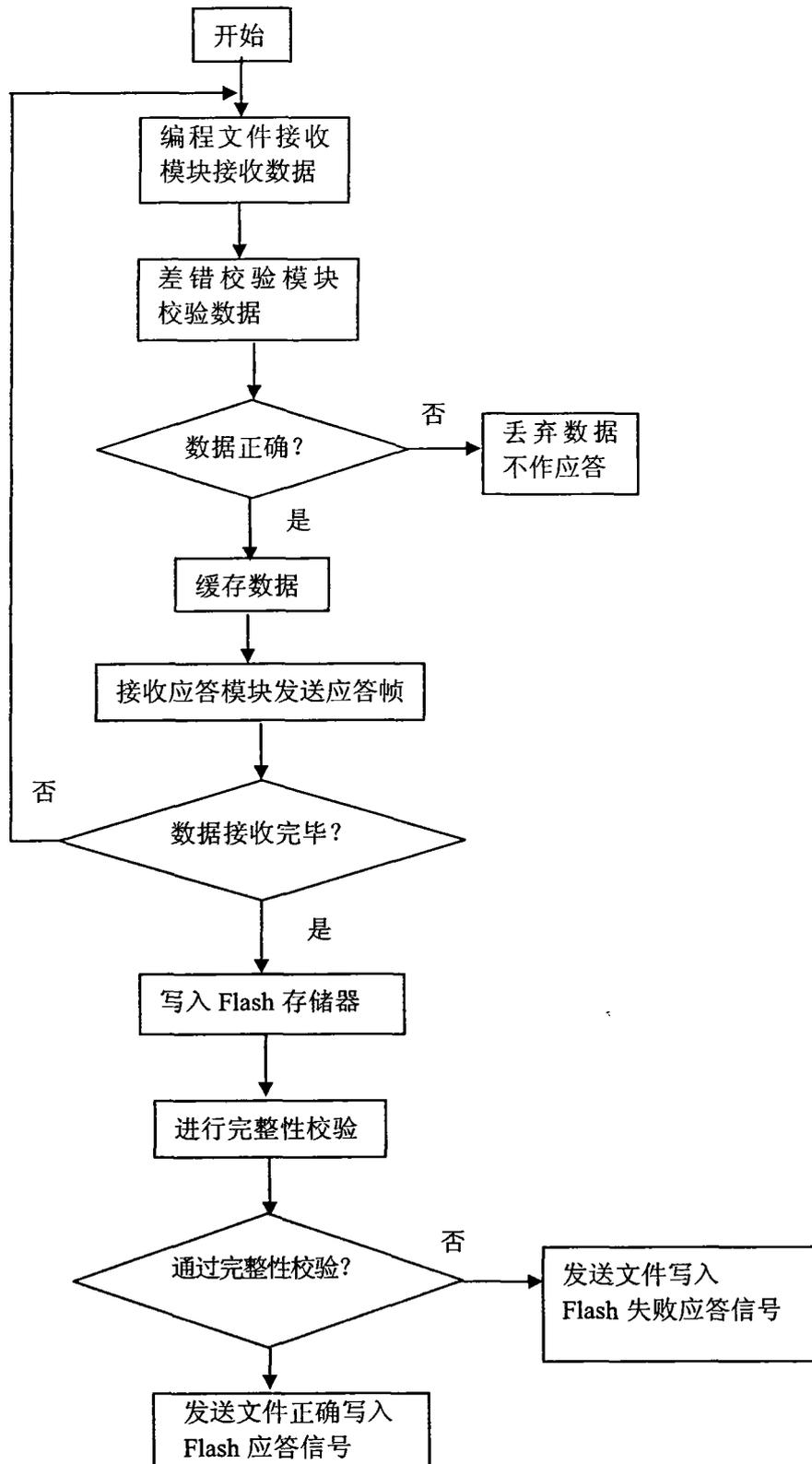


图 1

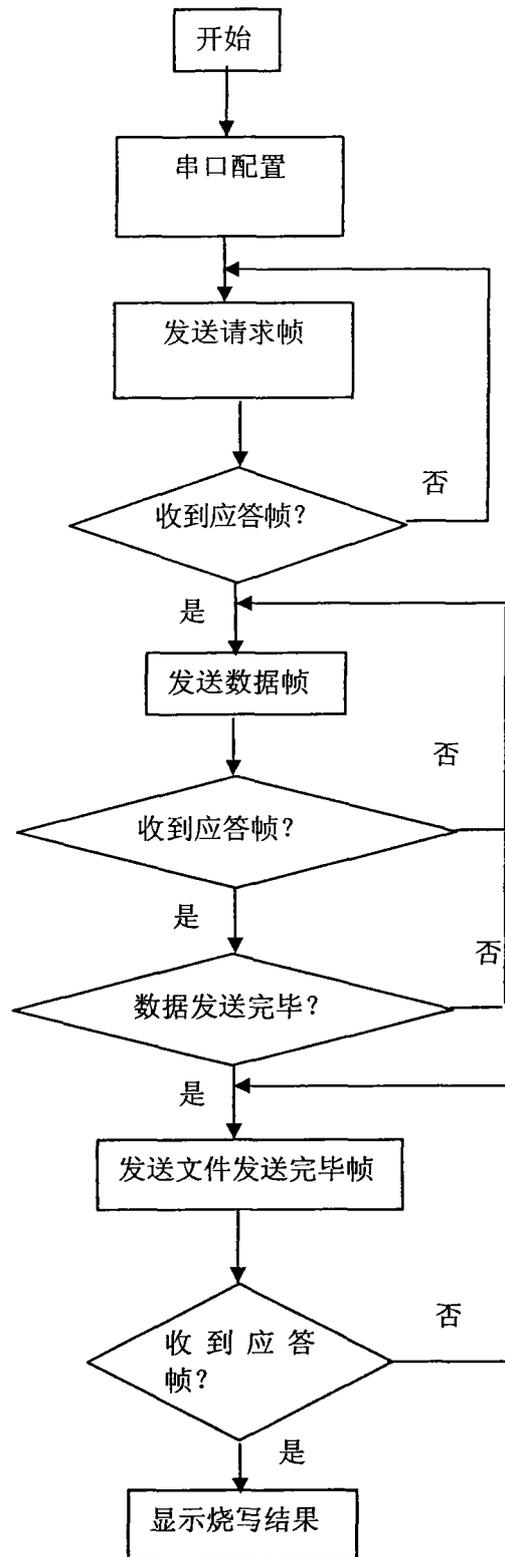
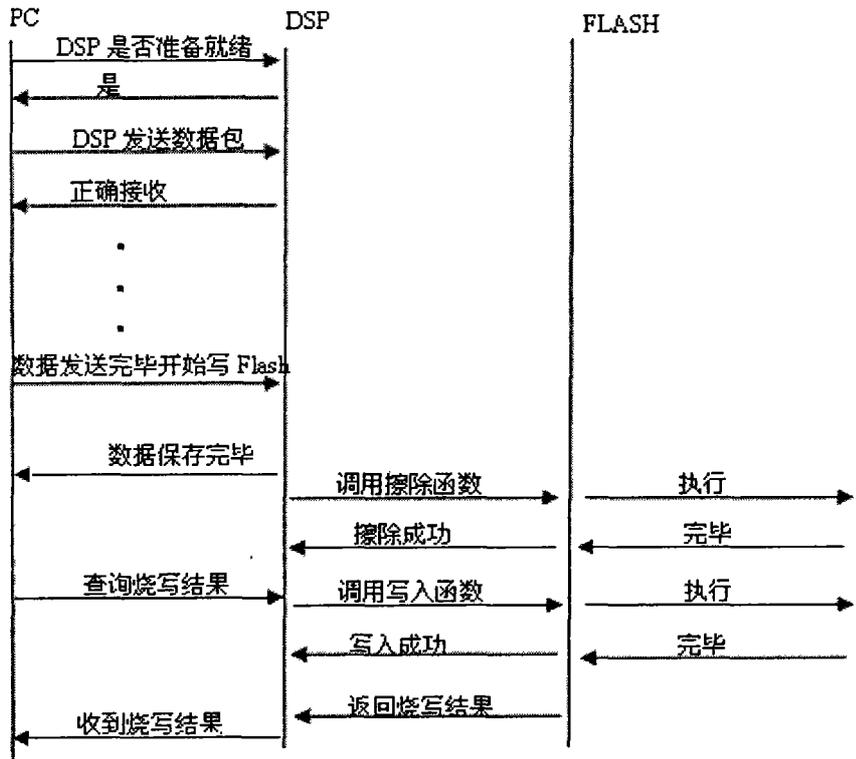


图 2



注:PC 在发出数据后若两秒内没有收到 DSP 的响应则重发。

图 3

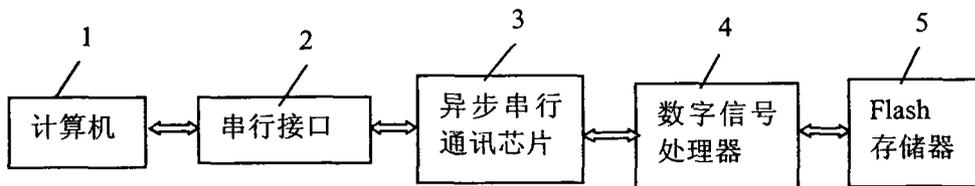


图 4

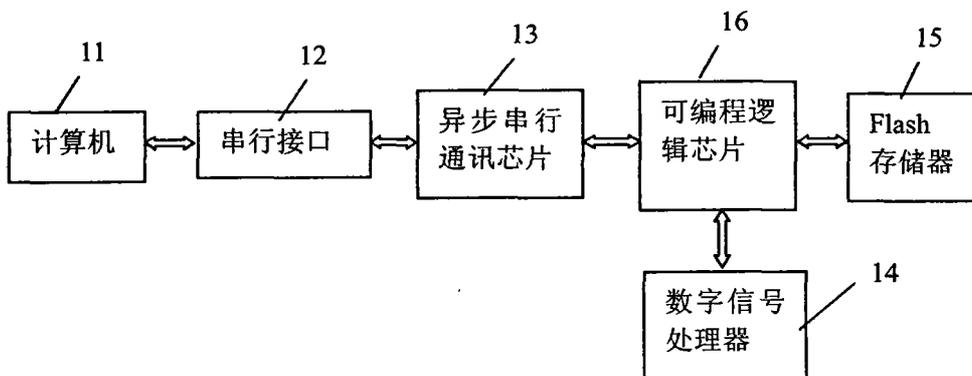


图 5