



(12)发明专利

(10)授权公告号 CN 104572425 B

(45)授权公告日 2017.11.17

(21)申请号 201310476768.1

(22)申请日 2013.10.12

(65)同一申请的已公布的文献号

申请公布号 CN 104572425 A

(43)申请公布日 2015.04.29

(73)专利权人 上海东软载波微电子有限公司

地址 200235 上海市徐汇区龙漕路299号天  
华信息科技园2号楼A座5层

(72)发明人 王珊珊 裴巍

(74)专利代理机构 北京同立钧成知识产权代理  
有限公司 11205

代理人 刘芳

(51)Int.Cl.

G06F 11/36(2006.01)

(56)对比文件

CN 1506808 A,2004.06.23,

CN 101414278 A,2009.04.22,

US 2004083331 A1,2004.04.29,

CN 1506808 A,2004.06.23,

Rob Oshana.DSP系统的测试和调试.《百度  
文库,URL:<http://dwz.cn/55yX5t>》.2011,1-5.

审查员 高新琳

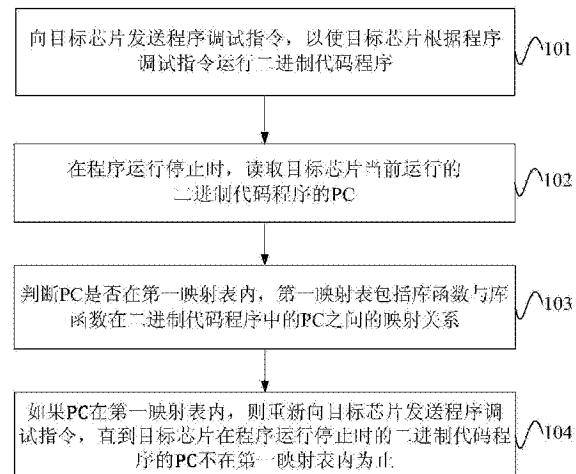
权利要求书3页 说明书8页 附图4页

(54)发明名称

程序调试方法、装置及系统

(57)摘要

本发明提供一种程序调试方法、装置及系统,该方法包括:向目标芯片发送程序调试指令,以使所述目标芯片根据所述程序调试指令运行二进制代码程序;在程序运行停止时,读取所述目标芯片当前运行的二进制代码程序的地址PC;判断所述PC是否在第一映射表内,所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;如果所述PC在所述第一映射表内,则重新向所述目标芯片发送程序调试指令,直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止。本发明解决了调试过程中程序停止在库函数对应的机器码中的问题,进而能够确定目标芯片运行的二进制机器码与源程序之间的对应关系。



1. 一种程序调试方法,其特征在于,包括:

向目标芯片发送程序调试指令,以使所述目标芯片根据所述程序调试指令运行二进制代码程序;

在程序运行停止时,读取所述目标芯片当前运行的二进制代码程序的地址PC;

判断所述PC是否在第一映射表内,所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;

如果所述PC在所述第一映射表内,则重新向所述目标芯片发送程序调试指令,直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止。

2. 根据权利要求1所述的方法,其特征在于,所述向目标芯片发送程序调试指令之前,还包括:

接收界面处理装置发送的所述第一映射表,所述第一映射表为所述界面处理装置根据编译过程中生成的调试信息生成的,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中所述程序指令与二进制代码一一对应。

3. 根据权利要求1所述的方法,其特征在于,还包括:

如果所述PC不在所述第一映射表内,则向界面处理装置发送所述PC,以使所述界面处理装置根据第二映射表,确定所述PC对应的源代码行号,所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

4. 根据权利要求1~3中任一项所述的方法,其特征在于,所述调试指令包括了单步指令、步越指令或停止运行指令。

5. 一种程序调试方法,其特征在于,包括:

对源代码进行编译,生成二进制代码和调试信息,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中所述程序指令与所述二进制代码一一对应;

根据所述程序指令与当前运行的二进制代码的PC之间的对应关系,确定所述源代码中调用的库函数对应的二进制代码的起始PC和结束PC;

根据各个库函数对应的二进制代码的起始PC和结束PC,生成第一映射表,所述第一映射表包括所述库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;

根据所述源代码行号与所述源代码对应的二进制代码的PC之间的对应关系,生成第二映射表,所述第二映射表包括所述源代码行号与所述源代码对应的二进制代码的PC之间的映射关系。

6. 根据权利要求5所述的方法,其特征在于,所述生成第一映射表之后,还包括:

向所述程序调试器发送所述第一映射表。

7. 根据权利要求6所述的方法,其特征在于,所述向程序调试器发送所述第一映射表之后,还包括:

接收所述程序调试器发送的PC;

根据所述第二映射表,确定所述PC对应的源代码行号。

8. 一种程序调试器,其特征在于,包括:

第一发送模块,用于向目标芯片发送程序调试指令,以使所述目标芯片根据所述程序

调试指令运行二进制代码程序；

读取模块，用于在程序运行停止时，读取所述目标芯片当前运行的二进制代码程序的地址PC；

判断模块，用于判断所述PC是否在第一映射表内，所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系；

第一发送模块，还用于如果所述PC在所述第一映射表内，则重新向所述目标芯片发送程序调试指令，直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止。

9. 根据权利要求8所述的程序调试器，其特征在于，所述第一发送模块还包括：

第一接收单元，用于第一发送模块向目标芯片发送程序调试指令之前，接收界面处理装置发送的所述第一映射表，所述第一映射表为所述界面处理装置根据编译过程中生成的调试信息生成的，所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系，以及程序指令与当前运行的二进制代码的PC之间的对应关系，其中所述程序指令与二进制代码一一对应。

10. 根据权利要求8所述的程序调试器，其特征在于，还包括：

第二发送模块，用于如果所述PC不在所述第一映射表内，则向界面处理装置发送所述PC，以使所述界面处理装置根据第二映射表，确定所述PC对应的源代码行号，所述第二映射表包括所述源代码行号与所述源代码对应的二进制代码的PC之间的映射关系。

11. 一种界面处理装置，其特征在于，包括：

编译模块，用于对源代码进行编译，生成二进制代码和调试信息，所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系，以及程序指令与当前运行的二进制代码的PC之间的对应关系，其中所述程序指令与二进制代码一一对应；

确定模块，用于根据所述程序指令与当前运行的二进制代码的PC之间的对应关系，确定所述源代码中调用的库函数对应的二进制代码的起始PC和结束PC；

第一生成模块，用于根据各个库函数对应的二进制代码的起始PC和结束PC，生成第一映射表，所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系；

第二生成模块，用于根据所述源代码行号与所述源代码对应的二进制代码的PC之间的对应关系，生成第二映射表，所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

12. 根据权利要求11所述的界面处理装置，其特征在于，还包括：

第三发送模块，用于在第一生成模块生成第一映射表之后，向程序调试器发送所述第一映射表。

13. 根据权利要求12所述的界面处理装置，其特征在于，所述第三发送模块还包括：

第二接收单元，用于第三模块向程序调试器发送所述第一映射表之后，接收所述程序调试器发送的PC；

确定单元，用于根据所述第二映射表，确定所述PC对应的源代码行号。

14. 一种程序调试系统，其特征在于，包括：

如权利要求8-10任一项所述的程序调试器和权利要求11-13任一项所述的界面处理装

置。

## 程序调试方法、装置及系统

### 技术领域

[0001] 本发明涉及一种程序调试技术,特别涉及一种防止程序停止在库函数中的程序调试方法、装置及系统。

### 背景技术

[0002] C语言调试开发环境包括界面处理装置和硬件调试器部分,其中界面处理装置包括了代码编写、分析、编译和调试等功能,界面处理装置与硬件调试器通过串口或USB等端口相连实现相互通信,硬件调试器连接用户实际应用系统,用于对目标芯片进行实时调试及监控,实现基本的调试命令。

[0003] 调试时,硬件调试器提供了运行、单步、停止、断点等功能实现对目标芯片的运行控制,目标芯片中运行的是C编译器产生的二进制机器码,其中二进制机器码包括了用户自己编写的源程序对应的二进制机器码和编译器内部提供的库函数程序对应的二进制机器码,源程序通过调用库函数实现相应功能,C编译器提供的库函数一般集成在工具链中,不以用户源代码形式出现,当目标芯片执行完单步、步越命令或接收到停止运行命令时,目标芯片停止运行并返回当前的程序计数器(Program counter,简称:PC),界面处理装置根据PC找到所对应的源程序行号信息并显示当前程序运行的源码位置。

[0004] 然而,现有技术在执行完单步、步越命令或接收到停止运行命令之后,时常出现无法确定目标芯片运行的二进制机器码与源程序之间的对应关系,不利于用户调试。

### 发明内容

[0005] 本发明提供一种程序调试方法、装置及系统,解决了调试过程中程序停止在库函数对应的机器码中的问题,进而能够确定目标芯片运行的二进制机器码与源程序之间的对应关系。

[0006] 第一个方面,本发明实施例提供一种程序调试方法,包括:

[0007] 向目标芯片发送程序调试指令,以使所述目标芯片根据所述程序调试指令运行二进制代码程序;

[0008] 在程序运行停止时,读取所述目标芯片当前运行的二进制代码程序的PC;

[0009] 判断所述PC是否在第一映射表内,所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;

[0010] 如果所述PC在所述第一映射表内,则重新向所述目标芯片发送程序调试指令,直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止。

[0011] 进一步的,所述向目标芯片发送程序调试指令之前,还包括:

[0012] 接收界面处理装置发送的所述第一映射表,所述第一映射表为所述界面处理装置根据编译过程中生成的调试信息生成的,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中所述程序指令与二进制代码一一对应。

- [0013] 进一步的,还包括:
- [0014] 如果所述PC不在所述第一映射表内,则向界面处理装置发送所述PC,以使所述界面处理装置根据第二映射表,确定所述PC对应的源代码行号,所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。
- [0015] 可选的,在上述所有实施例提供的一种程序调试方法中,所述调试指令包括了单步指令、步越指令或停止运行指令。
- [0016] 第二个方面,本发明实施例提供一种程序调试方法,包括:
- [0017] 对源代码进行编译,生成二进制代码和调试信息,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中程序指令与二进制代码一一对应;
- [0018] 根据程序指令与当前运行的二进制代码的PC之间的对应关系,确定所述源代码中调用的库函数对应的二进制代码的起始PC和结束PC;
- [0019] 根据各个库函数对应的二进制代码的起始PC和结束PC,生成第一映射表,所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;
- [0020] 根据源代码行号与源代码对应的二进制代码的PC之间的对应关系,生成第二映射表,所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。
- [0021] 进一步的,所述生成第一映射表之后,还包括:
- [0022] 向程序调试器发送所述第一映射表。
- [0023] 进一步的,所述向程序调试器发送所述第一映射表之后,还包括:
- [0024] 接收所述程序调试器发送的PC;
- [0025] 根据所述第二映射表,确定所述PC对应的源代码行号。
- [0026] 第三个方面,本发明实施例提供一种程序调试器,包括:
- [0027] 第一发送模块,用于向目标芯片发送程序调试指令,以使所述目标芯片根据所述程序调试指令运行二进制代码程序;
- [0028] 读取模块,用于在程序运行停止时,读取所述目标芯片当前运行的二进制代码程序的PC;
- [0029] 判断模块,用于判断所述PC是否在第一映射表内,所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;
- [0030] 第一发送模块,还用于如果所述PC在所述第一映射表内,则重新向所述目标芯片发送程序调试指令,直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止。
- [0031] 进一步的,所述第一发送模块还包括:
- [0032] 第一接收单元,用于第一发送模块向目标芯片发送程序调试指令之前,接收界面处理装置发送的所述第一映射表,所述第一映射表为所述界面处理装置根据编译过程中生成的调试信息生成的,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中所述程序指令与二进制代码一一对应。
- [0033] 进一步的,还包括:
- [0034] 第二发送模块,用于如果所述PC不在所述第一映射表内,则向界面处理装置发送

所述PC,以使所述界面处理装置根据第二映射表,确定所述PC对应的源代码行号,所述第二映射表包括所述源代码行号与源代码对应的二进制代码的PC之间的映射关系。

[0035] 第四个方面,本发明实施例提供一种界面处理装置,包括:

[0036] 编译模块,用于对源代码进行编译,生成二进制代码和调试信息,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中所述程序指令与二进制代码一一对应;

[0037] 确定模块,用于根据所述程序指令与当前运行的二进制代码的PC之间的对应关系,确定所述源代码中调用的库函数对应的二进制代码的起始PC和结束PC;

[0038] 第一生成模块,用于根据各个库函数对应的二进制代码的起始PC和结束PC,生成第一映射表,所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;

[0039] 第二生成模块,用于根据源代码行号与源代码对应的二进制代码的PC之间的对应关系,生成第二映射表,所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

[0040] 进一步的,还包括

[0041] 第三发送模块,用于在第一生成模块生成第一映射表之后,向程序调试器发送所述第一映射表。

[0042] 进一步的,所述第三发送模块还包括:

[0043] 第二接收单元,用于第三模块向程序调试器发送所述第一映射表之后,接收所述程序调试器发送的PC;

[0044] 确定单元,用于根据所述第二映射表,确定所述PC对应的源代码行号。

[0045] 第五个方面,本发明实施例提供一种程序调试系统,包括:

[0046] 如上第三个方面任一所述的程序调试器和第四个方面的任一所述的界面处理装置。

[0047] 本发明实施例提供的程序调试方法、装置及系统,通过向目标芯片发送程序调试指令,在程序停止时,读取目标芯片当前运行的二进制代码程序的PC,判断PC是否在第一映射表内,如果所述PC在所述第一映射表内,则重新向所述目标芯片发送程序调试指令,直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止,解决了调试过程中程序停止在库函数对应的机器码中的问题,进而能够确定目标芯片运行的二进制机器码与源程序之间的对应关系。

## 附图说明

[0048] 为了更清楚地说明本发明实施例或现有技术中的技术方案,下面将对实施例或现有技术描述中所需要使用的附图作一简单地介绍,显而易见地,下面描述中的附图是本发明的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动性的前提下,还可以根据这些附图获得其他的附图。

[0049] 图1为本发明程序调试方法实施例一的流程图;

[0050] 图2为本发明程序调试方法实施例二的流程图;

[0051] 图3为本发明程序调试方法实施例三的流程图;

- [0052] 图4为本发明程序调试器一实施例的结构示意图；
- [0053] 图5为本发明界面处理装置一实施例的结构示意图；
- [0054] 图6为本发明程序调试系统一实施例的结构示意图。

## 具体实施方式

[0055] 为使本发明实施例的目的、技术方案和优点更加清楚，下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例是本发明一部分实施例，而不是全部的实施例。基于本发明中的实施例，本领域普通技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0056] 图1为本发明程序调试方法实施例一的流程图。本实施例的执行主体为程序调试器，该程序调试器可以对目标芯片进行实时调试及监控，实现基本的调试命令，本实施例中程序调试器还可以对单片机进行调试。具体的，本实施例包括如下步骤：

[0057] 步骤101、向目标芯片发送程序调试指令，以使所述目标芯片根据所述程序调试指令运行二进制代码程序。

[0058] 在本实施例中，程序调试器 向目标芯片发送程序调试指令之前，程序调试器会将界面处理装置编译生成的可执行文件下载到目标芯片中，目标芯片运行的是二进制机器代码，所以可执行文件中的程序是以二进制形式表示的，该可执行文件中包含了用户自己编写的源代码对应的二进制代码以及源代码中调用的库函数对应的二进制代码，其中调用的库函数对应的二进制代码是通过链接方式连接到源代码对应的二进制代码中的。在本实施例中，程序调试器与界面处理装置通过串口、USB等方式进行数据交换，程序调试器与目标芯片通过专用调试线连接。在本实施例中，调试指令包括了单步指令、步越指令或停止运行指令，程序调试器开始调试时，接收界面处理装置发送的调试指令，程序调试器根据接收到的调试指令指示目标芯片根据调试指令运行二进制代码程序。

[0059] 步骤102、在程序运行停止时，读取所述目标芯片当前运行的二进制代码程序的PC。

[0060] 本实施例中，目标芯片根据调试指令停止运行时，程序调试器读取目标芯片当前运行的二进制代码程序对应的PC，由于目标芯片无法识别当前运行的二进制代码是用户自己编写的源代码还是源代码调用的库函数对应的二进制代码，所以目标芯片停止运行时，返回的PC可能是源代码对应的二进制代码的PC，也可能是库函数对应的二进制代码的PC，而在现有技术中，如果返回的PC是库函数对应的二进制代码的PC，即程序停止在库函数对应的二进制代码程序中，则界面处理装置根据该PC是无法找到对应的源代码行号的，也就无法显示当前运行的程序在源代码中的位置，如果返回的PC是用户自己编写的源代码对应的二进制代码的PC时，界面处理装置根据该PC能够找到对应的源代码行号信息，也就能够准确显示当前运行的程序在源代码中的位置。

[0061] 步骤103、判断所述PC是否在第一映射表内，所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系。

[0062] 本实施例中，程序调试器判断读取的PC是否在第一映射表内，第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系，具体地，所述库函数在所述二进制代码程序中的PC为库函数在所述二进制代码程序中的起始PC和结束PC的范围，

如果读取的PC在库函数对应的二进制代码的起始PC和结束PC的范围内，则执行步骤104。

[0063] 步骤104、如果所述PC在所述第一映射表内，则重新向所述目标芯片发送程序调试指令，直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止。

[0064] 本实施例中，如果读取的PC在第一映射表内，则重新向目标芯片发送调试指令，在该步骤中，重新向目标芯片发送的调试指令为单步指令或步越指令，目标芯片根据单步指令或步越指令继续执行一条指令，接着执行步骤102、步骤103，如果读取的PC还在第一映射表内，则继续执行步骤104，直到目标芯片在程序运行停止时的二进制代码程序的PC不在第一映射表内为止，如果读取的PC不在第一映射表内，即当前运行的二进制代码程序为用户自己编写的源代码程序而不是库函数对应的二进制代码，此时根据PC可以找到对应的源代码行号，进而能够显示当前运行的二进制代码在源代码中的位置。

[0065] 本发明实施例提供的程序调试方法，通过向目标芯片发送程序调试指令，在程序停止时，读取目标芯片当前运行的二进制代码程序的PC，判断PC是否在第一映射表内，如果所述PC在所述第一映射表内，则重新向所述目标芯片发送程序调试指令，直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止，解决了调试过程中程序停止在库函数对应的机器码中的问题，进而能够确定目标芯片运行的二进制机器码与源程序之间的对应关系。

[0066] 图2为本发明程序调试方法实施例二的流程图，如图2所示，本实施例的方法可以包括：

[0067] 步骤201、接收界面处理装置发送的第一映射表，所述第一映射表为界面处理装置根据编译过程中生成的调试信息生成的，调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系，以及程序指令与当前运行的二进制代码的PC之间的对应关系，其中程序指令与二进制代码是一一对应的。

[0068] 在本实施例中，程序调试器接收界面处理装置发送的第一映射表并将其保存，第一映射表是界面处理装置根据编译过程生成的调试信息中分析得到所有库函数在二进制代码中的起始PC和结束PC而生成的，其中编译生成的调试信息中包括了源代码行号与源代码对应的二进制代码的PC之间的对应关系，以及程序指令与当前运行的二进制代码的PC之间的对应关系，其中程序指令与二进制代码是一一对应的，分析程序指令，找到所有库函数对应的二进制代码的PC并将其生成第一映射表。

[0069] 步骤202、向目标芯片发送程序调试指令，以使所述目标芯片根据所述程序调试指令运行二进制代码程序。

[0070] 在本实施例中，调试指令包括了单步指令、步越指令或停止运行指令，当目标芯片接收到单步指令时，目标芯片执行完一行指令后停止运行，只有再次接收到调试指令才会继续运行程序，当目标芯片接收到步越指令时，根据步越指令不同，运行的程序会执行步入函数或者不步入函数的操作，当正在运行的程序接收到停止运行指令时，该程序会停止运行。

[0071] 步骤203、在程序运行停止时，读取所述目标芯片当前运行的二进制代码程序的PC。

[0072] 在本实施例中，目标芯片执行完单步指令、步越指令或停止运行指令而停止时，程

序调试器读取目标芯片当前运行的二进制代码程序的PC，

[0073] 步骤204、判断PC是否在第一映射表内,第一映射表包括库函数与库函数在二进制代码程序中的PC之间的映射关系。

[0074] 本实施例中,如果PC在第一映射表内,执行步骤205,否则执行步骤206。

[0075] 步骤205、如果PC在第一映射表内,则重新向所述目标芯片发送程序调试指令,直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在第一映射表内为止。

[0076] 在本实施例中,PC在第一映射表内,即该PC对应的二进制代码为调用的库函数对应的二进制代码,此时程序调试器不会将该PC发送给界面处理装置,而是重新向目标芯片发送程序调试指令,在本实施例中,重新向目标芯片发送的调试指令为单步指令或步越指令,目标芯片根据单步指令或步越指令继续执行一条指令,接着继续执行步骤203和204,直到读取的PC不在第一映射表内才停止发送调试指令。

[0077] 步骤206、如果PC不在第一映射表内,向界面处理装置发送所述PC,以使界面处理装置根据第二映射表,确定PC对应的源代码行号,第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

[0078] 在本实施例中,程序调试器读取的PC不在第一映射表内,即PC对应的二进制代码为用户自己编写的源代码对应的二进制代码而不是库函数对应的二进制代码,此时向界面处理装置发送PC,界面处理装置根据第二映射表,确定PC对应的源代码行号,第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

[0079] 本发明实施例提供的程序调试方法,通过向目标芯片发送程序调试指令,在程序停止时,读取目标芯片当前运行的二进制代码程序的PC,判断PC是否在第一映射表内,如果所述PC在所述第一映射表内,则重新向所述目标芯片发送程序调试指令,直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止,解决了调试过程中程序停止在库函数对应的机器码中的问题,进而能够确定目标芯片运行的二进制机器码与源程序之间的对应关系,而且程序调试器只有判断出读取的PC不在第一映射表内时才向界面处理装置发送PC,节省了界面处理装置与程序调试器的通信时间消耗。

[0080] 图3为本发明程序调试方法实施例三的流程图。本实施例的执行主体为界面处理装置,也可以为集成了编程、编译和调试等功能的其他集成装置,具体的,本实施例包括如下步骤:

[0081] 步骤301、对源代码进行编译,生成二进制代码和调试信息,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中程序指令与二进制代码是一一对应的。

[0082] 在本实施例中,界面处理装置对用户编写的源代码进行编译,生成目标芯片能够运行的二进制代码,在编译过程中生成二进制代码的同时,还会生成很多调试信息,然后由连接程序将调试信息整合起来生成一个调试文件,调试时,可以用该调试文件对程序进行调试,其中调试信息中包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,除此之外,还包括源代码和二进制代码的类型、变量和函数等信息。

[0083] 步骤302、根据程序指令与当前运行的二进制代码的PC之间的对应关系,确定所述源代码中调用的库函数对应的二进制代码的起始PC和结束PC。

[0084] 步骤303、根据各个库函数对应的二进制代码的起始PC和结束PC，生成第一映射表，所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系。

[0085] 本实施例中，界面处理装置分析编译过程中生成的调试信息，得到所有调用的库函数在二进制代码中的起始PC和结束PC，根据各个库函数对应的二进制代码的起始PC和结束PC生成第一映射表，界面处理装置将生成的第一映射表发送给程序调试器。

[0086] 步骤304、根据源代码行号与源代码对应的二进制代码的PC之间的对应关系，生成第二映射表，所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

[0087] 在本实施例中，界面处理装置根据调试信息，将源代码行号与源代码对应的二进制代码的PC之间的对应关系生成第二映射表，在本实施例中，界面处理装置会接收程序调试器发送的不在第一映射表内的PC，根据所述第二映射表，确定PC对应的源代码行号。

[0088] 本发明实施例提供的程序调试方法，通过界面处理装置生成二进制代码和调试信息，根据调试信息生成第一映射表并发送给程序调试器，程序调试器根据读取的PC判断是否在第一映射表内，如果所述PC在所述第一映射表内，则重新向所述目标芯片发送程序调试指令，直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止，解决了调试过程中程序停止在库函数对应的机器码中的问题，进而能够确定目标芯片运行的二进制机器码与源程序之间的对应关系。

[0089] 图4为本发明程序调试器一实施例的结构示意图。如图4所示，本实施例提供的程序调试器40包括：第一发送模块401、读取模块402和判断模块403。

[0090] 其中，第一发送模块401用于向目标芯片发送程序调试指令，以使所述目标芯片根据所述程序调试指令运行二进制代码程序；

[0091] 读取模块402用于在程序运行停止时，读取所述目标芯片当前运行的二进制代码程序的PC；

[0092] 判断模块403用于判断所述PC是否在第一映射表内，所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系；

[0093] 第一发送模块401还用于如果所述PC在所述第一映射表内，则重新向所述目标芯片发送程序调试指令，直到所述目标芯片在程序运行停止时的二进制代码程序的PC不在所述第一映射表内为止。

[0094] 本实施例的程序调试器，可以用于程序调试方法实施例一的技术方案，其实现原理和技术效果类似，此处不再赘述。

[0095] 进一步地，在图4实施例的基础上，还包括：

[0096] 第二发送模块404，用于如果所述PC不在所述第一映射表内，向界面处理装置发送所述PC，以使所述界面处理装置根据第二映射表，确定所述PC对应的源代码行号，所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

[0097] 可选地，第一发送模块401还包括：

[0098] 第一接收单元4011，用于第一发送模块向目标芯片发送程序调试指令之前，接收界面处理装置发送的所述第一映射表，所述第一映射表为所述界面处理装置根据编译过程中生成的调试信息生成的，所述调试信息包括源代码行号与源代码对应的二进制代码的PC

之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中程序指令与二进制代码是一一对应的。

[0099] 本实施例的程序调试器,可以用于程序调试方法实施例二的技术方案,其实现原理和技术效果类似,此处不再赘述。

[0100] 图5为本发明界面处理装置一实施例的结构示意图。本实施例提供的界面处理装置50包括:编译模块501、确定模块502、第一生成模块503和第二生成模块504。

[0101] 其中,编译模块501用于对源代码进行编译,生成二进制代码和调试信息,所述调试信息包括源代码行号与源代码对应的二进制代码的PC之间的对应关系,以及程序指令与当前运行的二进制代码的PC之间的对应关系,其中程序指令与二进制代码是一一对应的;

[0102] 确定模块502用于根据程序指令与当前运行的二进制代码的PC之间的对应关系,确定所述源代码中调用的库函数对应的二进制代码的起始PC和结束PC;

[0103] 第一生成模块503用于根据各个库函数对应的二进制代码的起始PC和结束PC,生成第一映射表,所述第一映射表包括库函数与所述库函数在所述二进制代码程序中的PC之间的映射关系;

[0104] 第二生成模块504用于根据源代码行号与源代码对应的二进制代码的PC之间的对应关系,生成第二映射表,所述第二映射表包括源代码行号与源代码对应的二进制代码的PC之间的映射关系。

[0105] 进一步地,界面处理装置50还包括:第三发送模块505,用于在第一生成模块生成第一映射表之后,向程序调试器发送所述第一映射表。

[0106] 可选地,第三发送模块505包括第二接收单元5051和确定单元5052。

[0107] 第二接收单元5051用于第三模块向程序调试器发送所述第一映射表之后,接收所述程序调试器发送的PC;

[0108] 确定单元5052用于根据所述第二映射表,确定所述PC对应的源代码行号。

[0109] 本实施例的界面处理装置,可以用于程序调试方法实施例三的技术方案,其实现原理和技术效果类似,此处不再赘述。

[0110] 图6为本发明程序调试系统一实施例的结构示意图,如图6所示,本实施例的系统包括:程序调试器40及界面处理装置50,其中,程序调试器40可以采用程序调试器一实施例的结构,其对应地,可以执行程序调试方法实施例一和二的技术方案;界面处理装置50可以采用界面处理装置一实施例的结构,其对应地,可以执行程序调试方法实施例三的技术方案,其实现原理和技术效果类似,此处不再赘述。

[0111] 本领域普通技术人员可以理解:实现上述各方法实施例的全部或部分步骤可以通过程序指令相关的硬件来完成。前述的程序可以存储于一计算机可读取存储介质中。该程序在执行时,执行包括上述各方法实施例的步骤;而前述的存储介质包括:ROM、RAM、磁碟或者光盘等各种可以存储程序代码的介质。

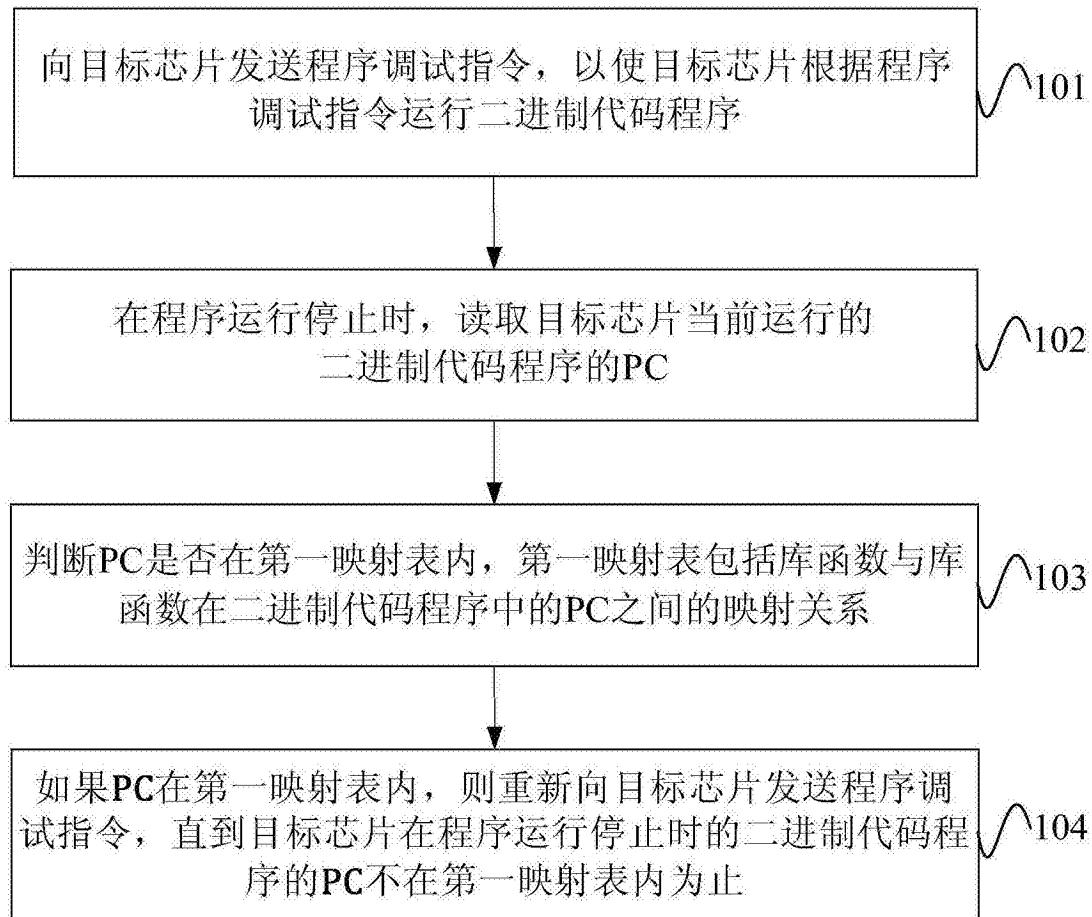


图1

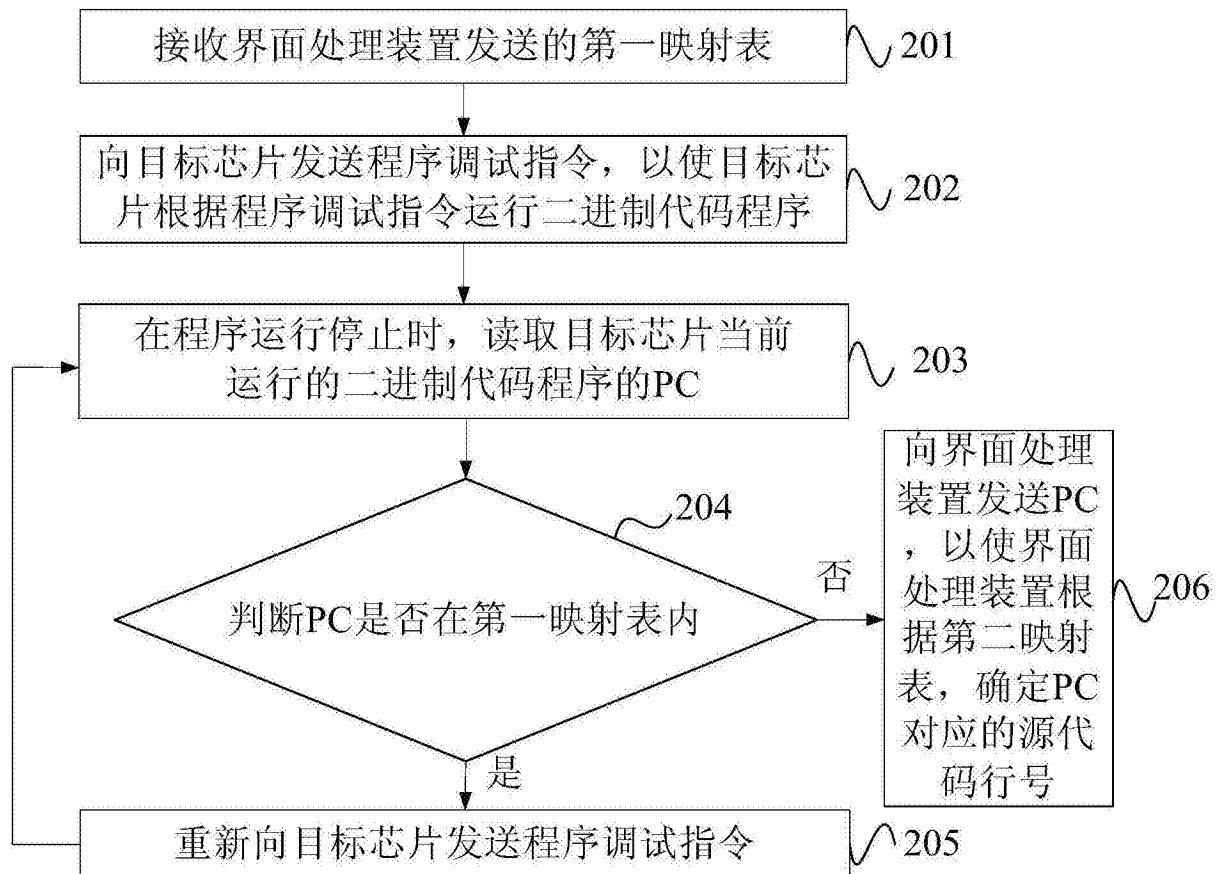


图2

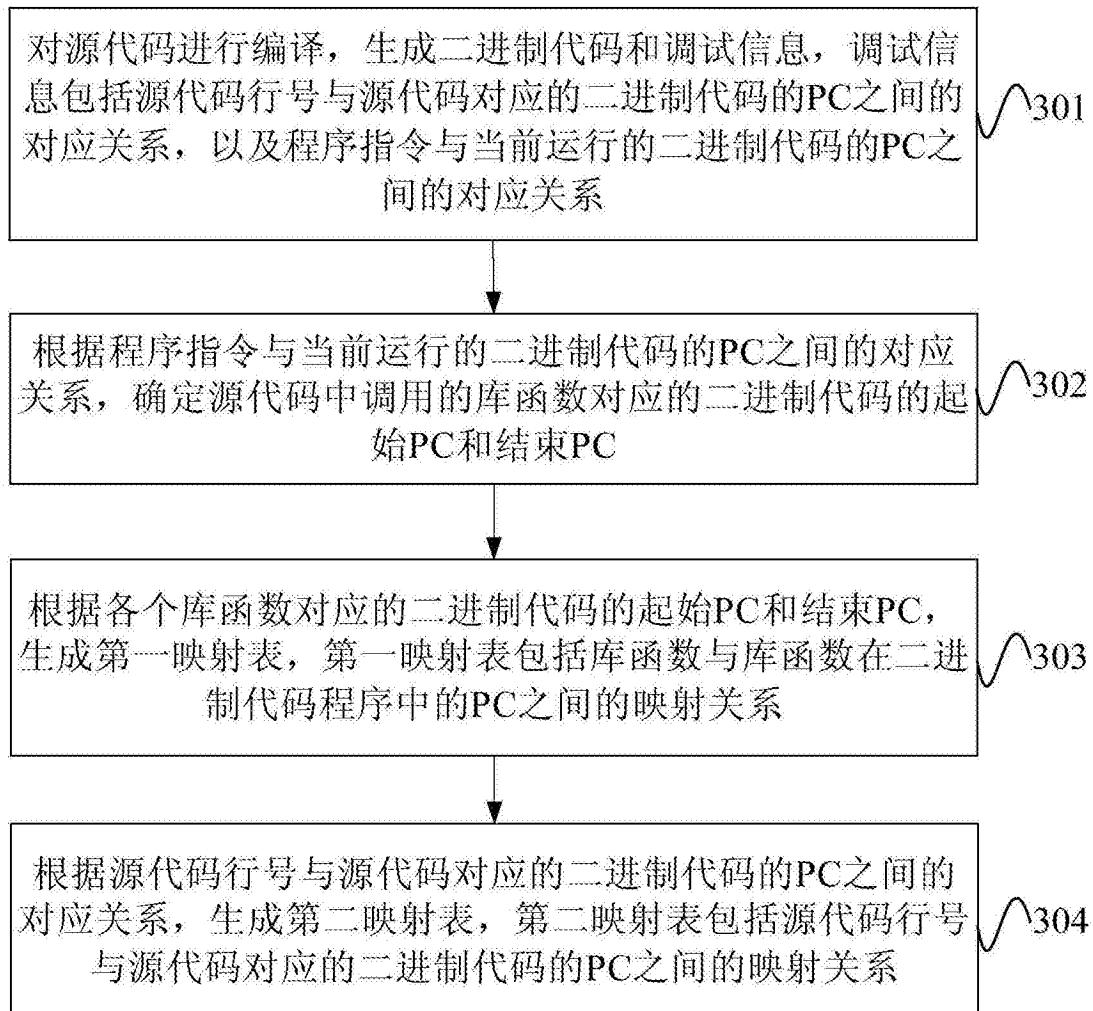


图3

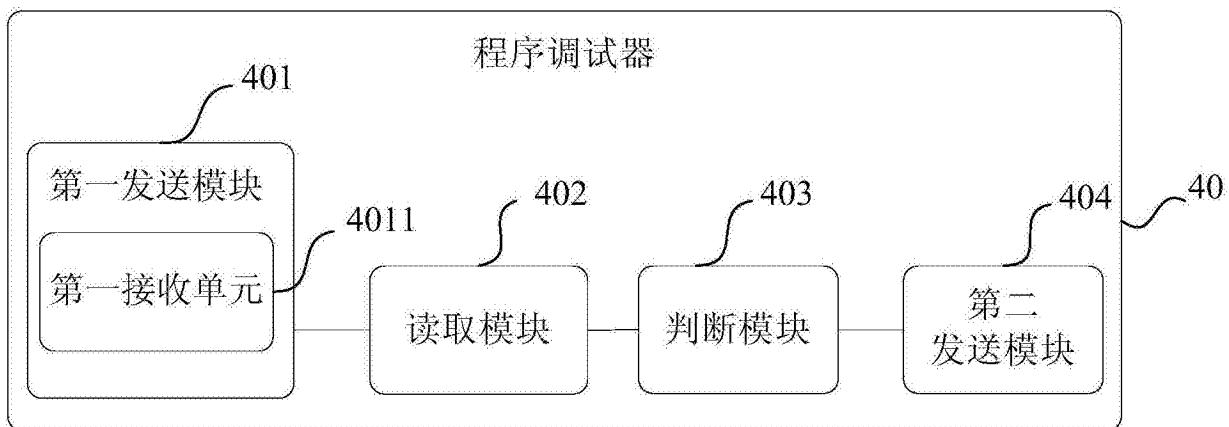


图4

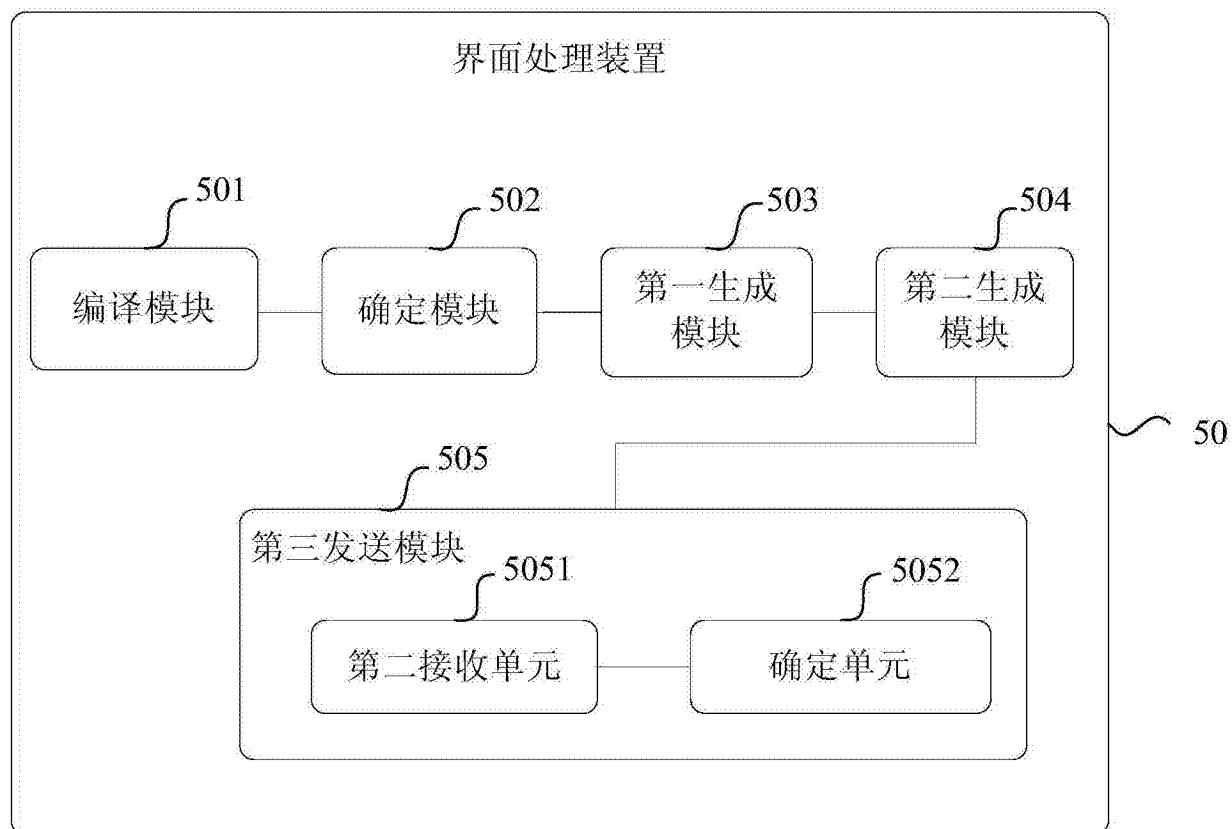


图5

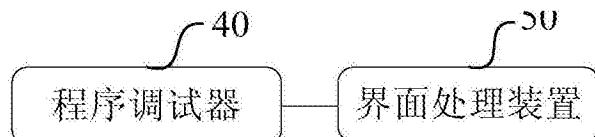


图6