



(12) 发明专利

(10) 授权公告号 CN 102298352 B

(45) 授权公告日 2012. 11. 28

(21) 申请号 201010210039. 8

(22) 申请日 2010. 06. 25

(73) 专利权人 中国科学院沈阳自动化研究所  
地址 110016 辽宁省沈阳市东陵区南塔街  
114 号

(72) 发明人 曾舒婷 杨志家 吕岩 谢闯  
刘志峰 段茂强

(74) 专利代理机构 沈阳科苑专利商标代理有限  
公司 21002

代理人 李晓光

(51) Int. Cl.

G06F 9/38 (2006. 01)

(56) 对比文件

CN 201716564 U, 2011. 01. 19,

CN 1731346 A, 2006. 02. 08,

Xu Meihua, et al. IP core design of PLC  
microprocessor with boolean module. 《High  
Density Microsystem Design》. 2005,

冉峰. IP 复用的 FSPLC 微处理器 SOC 设  
计. 《微电子学与计算机》. 2007, 第 24 卷 (第 11  
期),

Kyeonghoon Koo. Architectural design  
of a RISC processor for programmable  
logic controllers. 《Journal of Systems  
Architecture》. 1998,

审查员 丁匡正

权利要求书 4 页 说明书 18 页 附图 7 页

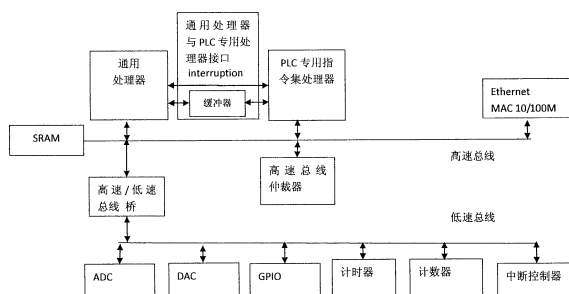
(54) 发明名称

高性能可编程控制器专用处理器体系结构及  
其实现方法

(57) 摘要

本发明涉及一种高性能可编程控制器专用处  
理器体系结构及其实现方法, 体系结构包括 PLC  
专用指令集处理器、通用处理器, 其中 PLC 专用指  
令集处理器通过 PLC 专用集处理器与通用处理器  
的接口与通用处理器相连; 实现方法按照该可编  
程控制器专用指令集处理器采用四级流水线, 包  
括取指阶段、译码阶段、执行阶段、以及回写阶段。  
本发明通过设计符合 PLC 指令特征的 PLC 专用指  
令集, 减少 PLC 处理器执行的指令数, 加快可编  
程控制器程序的执行速度, 提高了 PLC 处理器对功  
能块指令的处理性能。

CN 102298352 B



1. 一种高性能可编程控制器的处理器体系结构,其特征在于:包括 PLC 专用指令集处理器、通用处理器,其中 PLC 专用指令集处理器通过 PLC 专用集处理器与通用处理器的接口与通用处理器相连;

所述 PLC 专用指令集处理器具有以下结构:指令存储器、指令计数器、指令寄存器、指令译码器、控制单元、功能块单元、功能块寄存器组、数据存储器、寄存器组、位处理器、跳转调用指令及存取指令处理单元、I\O 数据存储器、状态寄存器,其中:

指令存储器,存储通用处理器并行编译后的指令,在指令计数器的控制下,将指令发送至指令寄存器;

指令计数器,计算下一条指令地址,实现指令地址的自增,并在跳转调用指令及存取指令处理单元的控制下,按照跳转调用指令中指令地址偏移需求,改变指令地址;在位处理器的控制下,按照可跳转位逻辑运算指令指令地址偏移需求,改变指令地址;将自增后的指令地址、执行跳转调用指令后改变的指令地址及执行可跳转位逻辑运算指令后改变的指令地址,通过指令计数器中指令地址选择,将指令存储器中对应的指令输出至指令寄存器;

指令寄存器,接收指令计数器选择的指令存储器中的指令,输出该指令至指令译码器;

指令译码器,将从指令寄存器输入的指令进行译码后,将操作码输出至位处理器、跳转调用指令及存取指令处理单元、功能块单元,将操作数地址输出至寄存器组、功能块寄存器组、数据存储器及 I\O 存储器;

控制单元,向指令计数器、指令寄存器、指令译码器发送控制指令;接收指令译码器操作码中的组码,分别对位处理器、跳转调用指令及存取指令处理单元和功能块单元进行控制;并控制与通用处理器进行中断信号的接收或发送;

功能块单元,接收控制单元的控制信号和指令译码器的操作码,与功能块寄存器组进行数据的存取操作;

功能块寄存器组,存储来自功能块单元的执行结果,根据来自跳转调用指令及存取指令处理单元的控制信号及指令译码器对其的操作数地址的选择信号,存取数据存储器的数据,或取出功能块寄存器组的数据,并通过多路选择器送至寄存器组中的累加器堆栈中;

数据存储器,通过跳转调用指令及存取指令处理单元的控制及指令译码器对其操作数地址的选择,存取寄存器组、功能块寄存器组的数据;

寄存器组,通过跳转调用指令及存取指令处理单元的控制及指令译码器传来的操作数地址,选择寄存器组中的寄存器,将该寄存器中的操作数输出至位处理器进行处理;或存取数据存储器及 I\O 数据存储器中的数据;

位处理器,通过指令译码器的操作码,对来自寄存器组的操作数以及通过多路选择器选择的来自功能块寄存器组、数据存储器、I\O 数据存储器并存储在寄存器组累加器堆栈中的数据进行相应处理,并将处理结果存储到寄存器组、状态寄存器、数据存储器或 I\O 数据存储器中;

跳转调用指令及存取指令处理单元,接收来自控制单元的控制信号,根据指令译码器传来的操作码,分别对指令计数器进行操作,对寄存器组、功能块寄存器组、数据存储器、I\O 数据存储器进行存取操作控制;

I\O 数据存储器,接收来自跳转调用指令及存取指令处理单元的控制信号以及指令译

码器的操作数地址选择信号,与寄存器组、功能块寄存器组以及 I\O 数据存储器进行数据存取;

状态寄存器,保存位处理器执行后的状态结果,并通过中断方式,将状态信号发送至通用处理器。

2. 按权利要求 1 所述的高性能可编程控制器的处理器体系结构,其特征在于:功能块单元中每类功能块并行扩展 1~16 个同类功能块单元,每个功能块具有自己的功能块寄存器组。

3. 按权利要求 1 所述的高性能可编程控制器的处理器体系结构,其特征在于:所述指令译码器译码使用的指令格式为位逻辑运算指令格式、可跳转位逻辑运算指令格式、跳转调用及存取指令格式以及功能块指令格式,其中:

位逻辑运算指令格式,第 31~30 位为组码;第 29~26 位为操作码;第 25 位用于判断当前位逻辑运算指令的操作数是否存储在累加器堆栈当中;第 24 位为条件位,用于对通用处理器运算的条件使能,其值决定于存储在状态寄存器中的状态结果;第 23~16 位,用于存储保存位逻辑运算结果的寄存器地址,其寄存器地址编码位数为 8 位;第 15~8 位,用于存储保存位逻辑运算操作数之一的寄存器地址;第 7~0 位,用于存储保存位逻辑运算另一操作数的寄存器地址。

可跳转位逻辑运算指令格式,第 31~30 位为组码;第 29~26 位为操作码;第 25 位决定是否跳转的条件位;第 24 位为条件位,用于对通用处理器运算的条件使能,其值决定于状态寄存器的状态结果;第 23~8 位,为跳转偏移位,其跳转偏移量为该可跳转位逻辑运算指令跳转到可跳转位逻辑运算指令跳转结束符或累加器堆栈中的操作数出栈并求或指令的偏移量;第 7~0 位,为存储累加器堆栈指针寄存器保存的当前累加器堆栈值的地址;

跳转调用及存取指令格式,第 31~30 位为组码;第 29~26 位为操作码;第 25 位,用于区分跳转调用指令及存取指令,当  $I = 0$  时,代表跳转调用指令,当  $I = 1$  时,代表存取指令,其中  $I$  为用于区分跳转调用指令及存取指令的一位二进制数;第 24 位,为条件位,用于基于条件位跳转偏移的跳转调用指令,其条件位的值来自于累加器堆栈当前值或功能块单元输出值或来自通用处理器的运算结果,如通用处理器执行比较指令后的比较结果;第 23~16 位,表示累加器堆栈栈顶地址或寄存器地址;第 15~0 位,表示跳转调用偏移位或操作数存取地址;

功能块指令格式,第 31~30 位为组码;第 29~26 位为操作码;第 25 位,表示复位优先位或加计数器输入的上升沿触发位;第 24 位,表示置位优先位或减计数器输入的上升沿触发位;第 23 位表示复位或时钟电平;第 22 位,表示装入加减计数器初始值使能位或计时器使能位;第 21~11 位,表示计数器当前值或计时器当前值;第 10~0 位,表示计数器予置值或计时器予置值。

4. 按权利要求 3 所述的指令译码器译码需要用到指令格式,其特征在于:所述指令译码器译码使用的指令格式中组码及操作码表示的指令集为位逻辑运算指令集、可跳转位逻辑运算指令集、跳转调用指令集、存取指令集、功能块指令集,其中:

位逻辑运算指令集包括:位与操作指令、位与非操作指令、位或操作指令、位异或操作指令以及位异或非操作指令;

可跳转位逻辑运算指令集包括:位累与操作指令、位累与非操作指令、位累或操作指

令、位累或非操作指令、可跳转位逻辑运算指令跳转结束符、累加器堆栈栈顶操作数出栈并求或指令；

跳转调用指令集包括：无条件跳转指令、符合条件跳转指令、不符合条件跳转指令、无条件功能块调用指令、符合条件功能块调用指令、不符合条件功能块调用指令、无条件从功能或功能块返回指令、符合条件功能或功能块返回指令、不符合条件功能或功能块返回指令。

存取指令集包括：

LOADQBIT：取功能块单元的布尔输出值到累加器堆栈中；

STOREQBIT：将功能块单元的布尔输出值存储到功能块寄存器组中；

LOADNQBIT：取功能块单元的输出值后取反输出到累加器堆栈中；

STORENQBIT：将功能块单元的输出值取反存储到功能块寄存器组中；

LOADMBIT：从存储器单元取操作数到寄存器组或功能块寄存器组中；

STOREMBIT：将寄存器组或功能块寄存器组的数据直接存储到存储器单元中；

LOADNMBIT：从存储器单元取操作数取反后存储到寄存器组或功能块寄存器组中；

STORENMBIT：将寄存器组或功能块寄存器组的数据取反后存储到存储器单元中；

LOADIOBIT：从 I\O 数据存储器直接取数到寄存器组中；

STOREIOBIT：将寄存器组数据直接送到 I\O 数据存储器中；

LOADNIOBIT：从 I\O 数据存储器取的数据取反后存储到寄存器组中；

STORENIOBIT：将寄存器组数据取反后送到从 I\O 数据存储器中；

LOADRMBIT：将寄存器组的内部寄存器的值压栈到累加器堆栈中；

STORERMBIT：将累加器堆栈中的值出栈到寄存器组的内部寄存器中；

LOADNRMBIT：将寄存器组的内部寄存器的值取反压栈到累加器堆栈中；

STORENRMBIT：将累加器堆栈中的值取反出栈到寄存器组的内部寄存器中；

功能块指令集包括：置位优先功能块指令、复位优先功能块指令、上升沿检测功能块指令、下降沿检测功能块指令、加计数器功能块指令、减计数器功能块指令、加减计数器功能块指令、脉冲功能块指令、接通延时功能块指令、断开延时功能块指令、实时时钟功能块指令。

5. 一种高性能可编程控制器专用指令集处理器体系结构的实现方法，其特征在于：按照该可编程控制器专用指令集处理器采用四级流水线，包括以下步骤：

取指阶段：依据指令计数器计算的指令地址，从指令存储器取指令至指令寄存器当中；

译码阶段：从指令寄存器取出指令送入指令译码器进行译码，通过指令中操作数地址，取出操作数；并依据指令中组码，将相应的操作码及操作数送至对应的处理单元中；

执行阶段：依据指令译码器发出的操作码及操作数，位处理器、功能块单元、跳转调用指令及存取指令处理单元给予相应处理；如执行的指令不是可编程控制器专用指令集处理器的指令时，则由控制单元向通用处理器发出中断请求，该可编程控制器专用指令集处理器可继续执行与通用处理器指令无依赖关系的指令，当通用处理器向可编程控制器专用指令集处理器发送中断请求时，可编程控制器专用指令集处理器将保存通用处理器的执行结果到状态寄存器中；将存储器访问阶段与执行阶段，合为执行阶段；存储器访问由跳转调

用指令及存取指令处理单元执行；

回写阶段：将可编程控制器专用指令集处理器的结果，依据存取指令类型写回累加器堆栈、内部寄存器、数据存储内部寄存器或 I\O 存储器当中。

6. 按权利要求 5 所述的高性能可编程控制器专用指令集处理器体系结构的实现方法，其特征在于：所述位处理器处理位逻辑运算指令及可跳转位逻辑运算指令：=

当位处理器执行位逻辑运算指令时，从累加器堆栈栈顶和寄存器组中取操作数后，进行位逻辑运算操作，将运算结果存储到累加器堆栈栈顶或寄存器组中；

执行完成后存储位处理器的运算结果。

7. 按权利要求 6 所述的高性能可编程控制器专用指令集处理器体系结构的实现方法，其特征在于：

当位处理器执行可跳转位逻辑运算指令时，按可跳转位逻辑运算指令的执行过程执行，执行完成后存储位处理器的运算结果；

所述可跳转位逻辑运算指令的执行过程为：

由位处理器判断该可跳转位逻辑指令为位累与操作指令、位累与非操作指令、位累或操作指令还是位累或非操作指令；

若为位累与操作指令或位累与非操作指令，且累加器堆栈的栈顶值为 0，则执行与操作或与非操作，将执行结果存储在累加器堆栈的栈顶，并设置指令格式中跳转条件位为 1，并设置跳转偏移量值为可跳转位逻辑运算指令跳转结束符与该可跳转位逻辑运算指令的偏移量，执行跳转；一次执行过程结束。

8. 按权利要求 7 所述的高性能可编程控制器专用指令集处理器体系结构的实现方法，其特征在于：

如果不满足“位累与操作指令或位累与非操作指令，且累加器堆栈的栈顶值为 0”条件，则不执行跳转，只执行位累与操作或位累与非操作，并设置指令格式中跳转条件位为 0。

9. 按权利要求 7 所述的高性能可编程控制器专用指令集处理器体系结构的实现方法，其特征在于：若为位累或操作指令或位累或非操作指令，且累加器堆栈的栈顶值为 1，则执行位累或操作或位累或非操作，将执行结果存储在累加器堆栈的栈顶，并设置指令格式中跳转条件位为 1，设置跳转偏移量值为可跳转位逻辑运算跳转结束符指令或累加器堆栈中的操作数出栈并求或指令到该位累或操作指令的偏移量；执行跳转；一次执行过程结束。

10. 按权利要求 9 所述的高性能可编程控制器专用指令集处理器体系结构的实现方法，其特征在于：

如果不满足“位累或操作指令或位累或非操作指令，且累加器堆栈的栈顶值为“1”条件，则不执行跳转，只执行位累或操作或位累或非操作，并设置指令格式中跳转条件位为 0。

11. 按权利要求 7 所述的高性能可编程控制器专用指令集处理器体系结构的实现方法，其特征在于：

所述可跳转位逻辑运算指令中的位累与操作指令、位累与非操作指令、位累或操作指令、位累或非操作指令，其执行过程与累加器堆栈的值相关，为区别于位逻辑运算指令中的与操作指令、与非操作指令、或操作指令、或非操作指令而命名。

## 高性能可编程控制器专用处理器体系结构及其实现方法

### 技术领域

[0001] 本发明涉及一种应用于自动控制领域中的处理器,具体的说是一种高性能可编程控制器的处理器体系结构及其实现方法。

### 背景技术

[0002] 可编程控制器 (PLC) 是以微处理器为核心,把自动化技术、通信技术融为一体的新型工业自动控制装置。可编程控制器以其可靠性高、抗干扰能力强、实时性强、控制程序可变、体积小、功耗低、使用方便、易维护等特点,广泛应用于工业行业的各个领域,如钢铁、石油、化工、电力、建材、机械制造、汽车、轻纺、交通运输等。

[0003] 随着可编程控制器国际标准 IEC\_61131-3 的颁布,可编程控制器在系统结构上,从传统的单机向多处理器发展;在控制系统组态上,从单机控制系统到分布式及过程控制系统发展;在编程语言上,呈多样化及简易化发展,创造了符合控制要求及通信能力的编程环境。由于信息技术的迅猛发展,可编程控制器对工业网络通信能力及实时性要求越来越高。这不仅要求可编程控制器通信能力的提高,而且需求可编程控制器的核心处理器提高其处理性能。

[0004] 现有自主研发的可编程控制器微处理器,大多采用通用微处理器或者通用处理器和布尔协处理器相结合的形式。通用处理器主要面向字节、字的处理,而可编程控制器指令主要面向比特位操作,占可编程控制器指令使用频率的 69%。这就迫切需要推出适合于可编程控制器处理特征的处理器。

### 发明内容

[0005] 针对现有技术中存在的处理器速度不能充分满足工业自动控制领域有关实时性的要求,本发明要解决的技术问题是提供一种能够很好的满足实时性要求的高性能可编程控制器的处理器体系结构及其实现方法。

[0006] 为解决上述技术问题,本发明采用的技术方案是:

[0007] 本发明一种高性能可编程控制器的处理器体系结构,包括 PLC 专用指令集处理器、通用处理器,其中 PLC 专用指令集处理器通过 PLC 专用集处理器与通用处理器的接口与通用处理器相连;

[0008] 所述 PLC 专用指令集处理器具有以下结构:指令存储器、指令计数器、指令寄存器、指令译码器、控制单元、功能块单元、功能块寄存器组、数据存储器、寄存器组、位处理器、跳转调用指令及存取指令处理单元、I\O 数据存储器、状态寄存器,其中:

[0009] 指令存储器,存储通用处理器并行编译后的指令,在指令计数器的控制下,将指令发送至指令寄存器;

[0010] 指令计数器,计算下一条指令地址,实现指令地址的自增,并在跳转调用指令及存取指令处理单元的控制下,按照跳转调用指令中指令地址偏移需求,改变指令地址;在位处理器的控制下,按照可跳转位逻辑运算指令指令地址偏移需求,改变指令地址;将自增后

的指令地址、执行跳转调用指令后改变的指令地址及执行可跳转位逻辑运算指令后改变的指令地址,通过指令计数器中指令地址选择,将指令存储器中对应的指令输出至指令寄存器;

[0011] 指令寄存器,接收指令计数器选择的指令存储器中的指令,输出该指令至指令译码器;

[0012] 指令译码器,将从指令寄存器输入的指令进行译码后,将操作码输出至位处理器、跳转调用指令及存取指令处理单元、功能块单元,将操作数地址输出至寄存器组、功能块寄存器组、数据存储器及 I\O 存储器;

[0013] 控制单元,向指令计数器、指令寄存器、指令译码器发送控制指令;接收指令译码器操作码中的组码,分别对位处理器、跳转调用指令及存取指令处理单元和功能块单元进行控制;并控制与通用处理器进行中断信号的接收或发送;

[0014] 功能块单元,接收控制单元的控制信号和指令译码器的操作码,与功能块寄存器组进行数据的存取操作;

[0015] 功能块寄存器组,存储来自功能块单元的执行结果,根据来自跳转调用指令及存取指令处理单元的控制信号及指令译码器对其的操作数地址的选择信号,存取数据存储器的数据,或取出功能块寄存器组的数据,并通过多路选择器送至寄存器组中的累加器堆栈中;

[0016] 数据存储器,通过跳转调用指令及存取指令处理单元的控制及指令译码器对其操作数地址的选择,存取寄存器组、功能块寄存器组的数据;

[0017] 寄存器组,通过跳转调用指令及存取指令处理单元的控制及指令译码器传来的操作数地址,选择寄存器组中的寄存器,将该寄存器中的操作数输出至位处理器进行处理;或存取数据存储器及 I\O 数据存储器中的数据;

[0018] 位处理器,通过指令译码器的操作码,对来自寄存器组的操作数以及通过多路选择器选择的来自功能块寄存器组、数据存储器、I\O 数据存储器并存储在寄存器组累加器堆栈中的数据进行相应处理,并将处理结果存储到寄存器组、状态寄存器、数据存储器或 I\O 数据存储器中;

[0019] 跳转调用指令及存取指令处理单元,接收来自控制单元的控制信号,根据指令译码器传来的操作码,分别对指令计数器进行操作,对寄存器组、功能块寄存器组、数据存储器、I\O 数据存储器进行存取操作控制;

[0020] I\O 数据存储器,接收来自跳转调用指令及存取指令处理单元的控制信号以及指令译码器的操作数地址选择信号,与寄存器组、功能块寄存器组以及 I\O 数据存储器进行数据存取;

[0021] 状态寄存器,保存位处理器执行后的状态结果,并通过中断方式,将状态信号发送至通用处理器。

[0022] 功能块单元中每类功能块并行扩展 1~16 个同类功能块单元,每个功能块具有自己的功能块寄存器组。

[0023] 所述指令译码器译码使用的指令格式为位逻辑运算指令格式、可跳转位逻辑运算指令格式、跳转调用及存取指令格式以及功能块指令格式,其中:

[0024] 位逻辑运算指令格式,第 31~30 位为组码;第 29~26 位为操作码;第 25 位用于

判断当前位逻辑运算指令的操作数是否存储在累加器堆栈当中；第 24 位为条件位，用于对通用处理器运算的条件使能，其值决定于存储在状态寄存器中的状态结果；第 23 ~ 16 位，用于存储保存位逻辑运算结果的寄存器地址，其寄存器地址编码位数为 8 位；第 15 ~ 8 位，用于存储保存位逻辑运算操作数之一的寄存器地址；第 7 ~ 0 位，用于存储保存位逻辑运算另一操作数的寄存器地址。

[0025] 可跳转位逻辑运算指令格式，第 31 ~ 30 位为组码；第 29 ~ 26 位为操作码；第 25 位决定是否跳转的条件位；第 24 位为条件位，用于对通用处理器运算的条件使能，其值决定于状态寄存器的状态结果；第 23 ~ 8 位，为跳转偏移位，其跳转偏移量为该可跳转位逻辑运算指令跳转到可跳转位逻辑运算指令跳转结束符或累加器堆栈中的操作数出栈并求或指令的偏移量；第 7 ~ 0 位，为存储累加器堆栈指针寄存器保存的当前累加器堆栈值的地址；

[0026] 跳转调用及存取指令格式，第 31 ~ 30 位为组码；第 29 ~ 26 位为操作码；第 25 位，用于区分跳转调用指令及存取指令，当  $I = 0$  时，代表跳转调用指令，当  $I = 1$  时，代表存取指令；第 24 位，为条件位，用于基于条件位跳转偏移的跳转调用指令，其条件位的值来自于累加器堆栈当前值或功能块单元输出值或来自通用处理器的运算结果，如通用处理器执行比较指令后的比较结果；第 23 ~ 16 位，表示累加器堆栈栈顶地址或寄存器地址；第 15 ~ 0 位，表示跳转调用偏移位或操作数存取地址；

[0027] 功能块指令格式，第 31 ~ 30 位为组码；第 29 ~ 26 位为操作码；第 25 位，表示复位优先位或加计数器输入的上升沿触发位；第 24 位，表示置位优先位或减计数器输入的上升沿触发位；第 23 位表示复位或时钟电平；第 22 位，表示装入加减计数器初始值使能位或计时器使能位；第 21 ~ 11 位，表示计数器当前值或计时器当前值；第 10 ~ 0 位，表示计数器予置值或计时器予置值。

[0028] 所述指令译码器译码使用的指令格式中组码及操作码表示的指令集为位逻辑运算指令集、可跳转位逻辑运算指令集、跳转调用指令集、存取指令集、功能块指令集，其中：

[0029] 位逻辑运算指令集包括：位与操作指令、位与非操作指令、位或操作指令、位异或操作指令以及位异或非操作指令；

[0030] 可跳转位逻辑运算指令集包括：位累与操作指令、位累与非操作指令、位累或操作指令、位累或非操作指令、可跳转位逻辑运算指令跳转结束符、累加器堆栈栈顶操作数出栈并求或指令；

[0031] 跳转调用指令集包括：无条件跳转指令、符合条件跳转指令、不符合条件跳转指令、无条件功能块调用指令、符合条件功能块调用指令、不符合条件功能块调用指令、无条件从功能或功能块返回指令、符合条件功能或功能块返回指令、不符合条件功能或功能块返回指令。

[0032] 存取指令集包括：

[0033] LOADQBIT：取功能块单元的布尔输出值到累加器堆栈中；

[0034] STOREQBIT：将功能块单元的布尔输出值存储到功能块寄存器组中；

[0035] LOADNQBITE：取功能块单元的输出值后取反输出到累加器堆栈中；

[0036] STORENQBITE：将功能块单元的输出值取反存储到功能块寄存器组中；

[0037] LOADMBIT：从存储器单元取操作数到寄存器组或功能块寄存器组中；



- [0038] STOREMBIT :将寄存器组或功能块寄存器组的数据直接存储到存储器单元中；
- [0039] LOADNMBIT :从存储器单元取操作数取反后存储到寄存器组或功能块寄存器组中；
- [0040] STORENMBIT :将寄存器组或功能块寄存器组的数据取反后存储到存储器单元中；
- [0041] LOADIOBIT :从 I\O 数据存储器直接取数到寄存器组中；
- [0042] STOREIOBIT :将寄存器组数据直接送到 I\O 数据存储器中；
- [0043] LOADNIOBIT :从 I\O 数据存储器取的数据取反后存储到寄存器组中；
- [0044] STORENIOBIT :将寄存器组数据取反后送到从 I\O 数据存储器中；
- [0045] LOADRMBIT :将寄存器组的内部寄存器的值压栈到累加器堆栈中；
- [0046] STORERMBIT :将累加器堆栈中的值出栈到寄存器组的内部寄存器中；
- [0047] LOADNRMBIT :将寄存器组的内部寄存器的值取反压栈到累加器堆栈中；
- [0048] STORENRMBIT :将累加器堆栈中的值取反出栈到寄存器组的内部寄存器中；
- [0049] 功能块指令集包括：置位优先功能块指令、复位优先功能块指令、上升沿检测功能块指令、下降沿检测功能块指令、加计数器功能块指令、减计数器功能块指令、加减计数器功能块指令、脉冲功能块指令、接通延时功能块指令、断开延时功能块指令、实时时钟功能块指令。
- [0050] 本发明一种高性能可编程控制器专用指令集处理器体系结构的实现方法，按照该可编程控制器专用指令集处理器采用四级流水线，包括以下步骤：
- [0051] 取指阶段：依据指令计数器计算的指令地址，从指令存储器取指令至 指令寄存器当中；
- [0052] 译码阶段：从指令寄存器取出指令送入指令译码器进行译码，通过指令中操作数地址，取出操作数；并依据指令中组码，将相应的操作码及操作数送至对应的处理单元中；
- [0053] 执行阶段：依据指令译码器发出的操作码及操作数，位处理器、功能块单元、跳转调用指令及存取指令处理单元给予相应处理；如执行的指令不是可编程控制器专用指令集处理器的指令时，则由控制单元向通用处理器发出中断请求，该可编程控制器专用指令集处理器可继续执行与通用处理器指令无依赖关系的指令，当通用处理器向可编程控制器专用指令集处理器发送中断请求时，可编程控制器专用指令集处理器将保存通用处理器的执行结果到状态寄存器中；将存储器访问阶段与执行阶段，合为执行阶段；存储器访问由跳转调用指令及存取指令处理单元执行；
- [0054] 回写阶段：将可编程控制器专用指令集处理器的结果，依据存取指令类型写回累加器堆栈、内部寄存器、数据存储器内部寄存器或 I\O 存储器当中。
- [0055] 所述位处理器处理位逻辑运算指令及可跳转位逻辑运算指令：
- [0056] 当位处理器执行位逻辑运算指令时，从累加器堆栈栈顶和寄存器组中取操作数后，进行位逻辑运算操作，将运算结果存储到累加器堆栈栈顶或寄存器组中；
- [0057] 执行完成后存储位处理器的运算结果。
- [0058] 当位处理器执行可跳转位逻辑运算指令时，按可跳转位逻辑运算指令的执行过程执行，执行完成后存储位处理器的运算结果；
- [0059] 所述可跳转位逻辑运算指令的执行过程为：
- [0060] 由位处理器判断该可跳转位逻辑指令为位累与操作指令、位累与非操作指令、位

累或操作指令还是位累或非操作指令；

[0061] 若为位累与操作指令或位累与非操作指令，且累加器堆栈的栈顶值为 0，则执行与操作或与非操作，将执行结果存储在累加器堆栈的栈顶，并设置指令格式中跳转条件位为 1，并设置跳转偏移量值为可跳转位逻辑运算指令跳转结束符与该可跳转位逻辑运算指令的偏移量，执行跳转；一次执行过程结束。

[0062] 如果不满足“位累与操作指令或位累与非操作指令，且累加器堆栈的栈顶值为 0”条件，则不执行跳转，只执行位累与操作或位累与非操作，并设置指令格式中跳转条件位为 0。

[0063] 若为位累或操作指令或位累或非操作指令，且累加器堆栈的栈顶值为 1，则执行位累或操作或位累或非操作，将执行结果存储在累加器堆栈的栈顶，并设置指令格式中跳转条件位为 1，设置跳转偏移量值为可跳转位逻辑运算跳转结束符指令或累加器堆栈中的操作数出栈并求或指令到该位累或操作指令的偏移量；执行跳转；一次执行过程结束。

[0064] 如果不满足“位累或操作指令或位累或非操作指令，且累加器堆栈的栈顶值为“1”条件，则不执行跳转，只执行位累或操作或位累或非操作，并设置指令格式中跳转条件位为 0。

[0065] 所述可跳转位逻辑运算指令中的位累与操作指令、位累与非操作指令、位累或操作指令、位累或非操作指令，其执行过程与累加器堆栈的值相关，为区别于位逻辑运算指令中的与操作指令、与非操作指令、或操作指令、或非操作指令而命名。

[0066] 本发明具有以下有益效果及优点：

[0067] 1. 本发明通过设计符合 PLC 指令特征的 PLC 专用指令集，减少 PLC 处理器执行的指令数，加快可编程控制器程序的执行速度。

[0068] 2. 本发明通过设计嵌入在 PLC 专用指令集处理器中的功能块单元及功能块寄存器组，提高了 PLC 处理器对功能块指令的处理性能。

[0069] 3. 本发明通过设计 PLC 专用指令集处理器中符合 PLC 高频数据访问特征的寄存器组、数据存储器和 I/O 数据存储器的直接存取方式，加快了 PLC 数据的存取速度。

[0070] 4. 本发明通过设计针对 PLC 大多采用位数据类型的特征，对寄存器组、数据存储器和按位编址，提高了 PLC 专用指令集处理器对位数据存取的速度，相应地提高了 PLC 专用指令集处理器的处理性能。

## 附图说明

[0071] 图 1 为本发明中可编程控制器专用处理器的体系结构框图；

[0072] 图 2 为本发明中可编程控制器专用指令集处理器的体系结构框图；

[0073] 图 3 为本发明中可编程控制器专用指令集处理器的位处理器框图；

[0074] 图 4 为本发明中可编程控制器专用指令集处理器的位处理器执行过程图；

[0075] 图 5 为本发明中可编程控制器专用指令集处理器的可跳转位逻辑运算流程图；

[0076] 图 6 为本发明中可编程控制器专用指令集处理器的指令计数器；

[0077] 图 7 为本发明中可编程控制器专用指令集处理器工作流程图。

## 具体实施方式

[0078] 本高性能可编程控制器专用处理器,拥有符合可编程控制器特征的专用指令集,并能够实现硬件加速。通过分析可编程控制器指令,设计出专用的可编程控制器指令集,和其相对应的可编程控制器专用处理器的体系结构,使其处理器能够快速处理可编程控制器程序,从而提高了可编程控制器处理性能。

[0079] 如图 1 所示,可编程控制器处理器所在片上系统体系结构包括:在高速总线上的可编程控制器处理器、Ethernet MAC 10/100M 及高速总线仲裁器;并通过高速/低速总线桥,连接低速总线上的外设,如 AD、DA 转换器,通用型输入输出接口,计时器,计数器及中断控制器。其中该可编程控制器处理器包括:通用处理器与 PLC 专用指令集处理器,其接口主要通过中断方式,利用缓冲器实现数据交换。其可编程控制器处理器实施过程如下:通用处理器通过可编程控制器并行编译器,将编译好的数据存储在 SRAM 中。待 PLC 专用指令集处理器初始化后,将 SRAM 中 PLC 专用指令集处理器的相关数据传输到 PLC 专用指令集处理器中的指令 RAM 和数据 RAM 中。当 PLC 专用指令集处理器执行到非 PLC 专用指令集中的指令时,向通用处理器发出中断,并将数据通过缓冲器发送给通用处理器。若 PLC 专用指令集处理器中的下一条指令与通用处理器将要执行的指令无关时,两处理器可并行执行;否则 PLC 专用指令集处理器将等待通用处理器执行结果。待通用处理器执行完成后,向 PLC 专用指令集处理器发出中断,并将数据传输到缓冲器中。

[0080] 如图 2 所示,所述 PLC 专用指令集处理器具有以下结构:指令存储器、指令计数器、指令寄存器、指令译码器、控制单元、功能块单元、功能块寄存器组、数据存储器、寄存器组、位处理器、跳转调用指令及存取指令处理单元、I\O 数据存储器、状态寄存器,其中:

[0081] 指令存储器,存储通用处理器并行编译后的指令,在指令计数器的控制下,将指令发送至指令寄存器;

[0082] 指令计数器,计算下一条指令地址,实现指令地址的自增,并在跳转调用指令及存取指令处理单元的控制下,按照跳转调用指令中指令地址偏移需求,改变指令地址;在位处理器的控制下,按照可跳转位逻辑运算指令指令地址偏移需求,改变指令地址;将自增后的指令地址、执行跳转调用指令后改变的指令地址及执行可跳转位逻辑运算指令后改变的指令地址,通过指令计数器中指令地址选择,将指令存储器中对应的指令输出至指令寄存器;

[0083] 指令寄存器,接收指令计数器选择的指令存储器中的指令,输出该指令至指令译码器;

[0084] 指令译码器,将从指令寄存器输入的指令进行译码后,将操作码输出至位处理器、跳转调用指令及存取指令处理单元、功能块单元,将操作数地址输出至寄存器组、功能块寄存器组、数据存储器及 I\O 存储器;

[0085] 控制单元,向指令计数器、指令寄存器、指令译码器发送控制指令;接收指令译码器操作码中的组码,分别对位处理器、跳转调用指令及存取指令处理单元和功能块单元进行控制;并控制与通用处理器进行中断信号的接收或发送;

[0086] 功能块单元,接收控制单元的控制信号和指令译码器的操作码,与功能块寄存器组进行数据的存取操作;

[0087] 功能块寄存器组,存储来自功能块单元的执行结果,根据来自跳转调用指令及存取指令处理单元的控制信号及指令译码器对其的操作数地址的选择信号,存取数据存储器

的数据,或取出功能块寄存器组的数据,并通 过多路选择器送至寄存器组中的累加器堆栈中;

[0088] 数据存储器,通过跳转调用指令及存取指令处理单元的控制及指令译码器对其操作数地址的选择,存取寄存器组、功能块寄存器组的数据;

[0089] 寄存器组,通过跳转调用指令及存取指令处理单元的控制及指令译码器传来的操作数地址,选择寄存器组中的寄存器,将该寄存器中的操作数输出至位处理器进行处理;或存取数据存储器及 I\O 数据存储器中的数据;

[0090] 位处理器,通过指令译码器的操作码,对来自寄存器组的操作数以及通过多路选择器选择的来自功能块寄存器组、数据存储器、I\O 数据存储器并存储在寄存器组累加器堆栈中的数据进行相应处理,并将处理结果存储到寄存器组、状态寄存器、数据存储器或 I\O 数据存储器中;

[0091] 跳转调用指令及存取指令处理单元,接收来自控制单元的控制信号,根据指令译码器传来的操作码,分别对指令计数器进行操作,对寄存器组、功能块寄存器组、数据存储器、I\O 数据存储器进行存取操作控制。

[0092] I\O 数据存储器,接收来自跳转调用指令及存取指令处理单元的控制信号以及指令译码器的操作数地址选择信号,与寄存器组、功能块寄存器组以及 I\O 数据存储器进行数据存取;

[0093] 状态寄存器,保存位处理器执行后的状态结果,并通过中断方式,将状态信号发送至通用处理器。

[0094] 为了实现 PLC 专用指令集处理器,本发明设计了符合可编程控制器特征的专用指令格式和指令集,其中指令格式为位逻辑运算指令格式、可跳转位逻辑运算指令格式、跳转调用及存取指令格式以及功能块指令格式共四种,具体为:

[0095] (1) 位逻辑运算指令格式(表 1),第 31~30 位为组码;第 29~26 位为操作码,操作码有四位,其可编程控制器指令表语言中的位逻辑运算指令有 AND、ANDN、OR、ORN、XOR、XORN,对应的可编程控制器专用指令集处理器指令集中的指令为 ANDBIT、ANDNBIT、ORBIT、ORNBIT、XORBIT、XORNBIT;第 25 位用于判断当前位逻辑运算指令的操作数是否存储在累加器堆栈当中,若不在累加器堆栈中,一般此指令为可跳转位逻辑运算指令后一个位逻辑运算指令,其运算结果将保存在累加器堆栈中;第 24 位为条件位,用于对通用处理器相关运算的条件使能,其值决定于状态寄存器的状态值;第 23~16 位,用于存储保存位逻辑运算结果的寄存器地址,其寄存器地址编码位数为 8 位;第 15~8 位,用于存储保存位逻辑运算操作数之一的寄存器地址(可以为寄存器组中的内部寄存器地址、累加器堆栈指针中保存的累加器堆栈地址);第 7~0 位,用于存储保存位逻辑运算另一操作数的寄存器地址(可以为寄存器组中的内部寄存器地址、累加器堆栈指针中保存的累加器堆栈地址)。

[0096] (2) 可跳转位逻辑运算指令格式(表 2),其组码为 01,其可编程控制器指令表语言中的指令包括:AND(、OR、ANDN(、ORN(、)、ORLD,对应的可编程控制器专用指令集处理器指令集中的指令为 ANDCRBIT、ORCRBIT、ANDNCRBIT、ORNCRBIT、CR、PUSHOR;第 25 位为跳转条件位,即该可跳转位逻辑运算指令满足跳转条件,如指令为 ANDCRBIT 时,且累加器堆栈当前值为 0 时,ANDCRBIT 指令后的位逻辑运算不用执行,并跳转到、可跳转位逻辑运算指令跳转结束符 CR 指令,其可跳转位逻辑运算流程图(图 5)表明了详细执行流程;第 24 位为条件

位,用于对通用处理器相关运算的条件使能,其值决定于状态寄存器的状态信号;第 23 ~ 8 位,为跳转偏移量,即偏移到可编程控制器专用指令集中 CR 及 PUSHOR 指令;第 7 ~ 0 位,为存储累加器堆栈指针寄存器保存的当前累加器堆栈值的地址。

[0097] (3) 跳转调用及存取指令格式(表 3),第 31 ~ 30 位为组码,其组码为 10;第 29 ~ 26 位为操作码;第 25 位,用于区分跳转调用指令及存取指令,当  $I = 0$  时,代表指令为跳转调用指令,其可编程控制器指令表语言中的指令包括 JMP、JMPC、JMPCN、CALL、CALLC、CALLCN、RET、RETC、RETCN,对应的可编程控制器专用指令集处理器指令集中的指令为 JMPBIT、JMPCBIT、JMPCNBIT、CALBIT、CALCBIT、CALCNBIT、RETBIT、RETCBIT、RETCNBIT;当  $I = 1$  时,其可编程控制器指令表语言中的指令包括:LD、LDN、ST、STN,对应的可编程控制器专用指令集处理器指令集中的指令依据存取类型分为 LOADBIT、LOADNBIT、STOREBIT、STORENBIT、LOADMBIT、STOREMBIT、LOADNMBIT、STORENMBIT、LOADIOBIT、STOREIOBIT、LOADNIOBIT、STORENIOBIT、LOADRMBIT、LOADNRMBIT、STORERMBIT、STORENRMBIT;第 24 位,为条件位,用于基于条件位跳转偏移的跳转调用指令,其条件位的值来自于累加器堆栈当前值或功能块单元输出值或来自通用处理器的运算结果,如通用处理器执行比较指令后的比较结果;第 23 ~ 16 位,表示累加器堆栈栈顶地址或寄存器地址;第 15 ~ 0 位,表示跳转调用偏移位或操作数存取地址;

[0098] (4) 功能块指令格式(表 4)。功能块分为触发器、边沿检测、计数器、定时器,它有自身的功能块单元寄存器组。第 31 ~ 30 位为组码,其指令的组码为 11;第 29 ~ 26 位为操作码,其可编程控制器指令表语言中的指令包括 SR、RS、R\_TRIG、F\_TRIG、CTU、CTD、CTUD、TP、TON、TOF、RTC,对应的可编程控制器专用指令集处理器指令集中的指令为 FB SR、FBRS、FB R\_TRIG、FB F\_TRIG、FB CTU、FB CTD、FB CTUD、FB TP、FB TON、FB TOF、FB RTC;第 25 位,表示复位优先位或加计数器输入的上升沿触发位;第 24 位,表示置位优先位或减计数器输入的上升沿触发位;第 23 位表示复位或时钟电平;第 22 位,表示装入加减计数器初始值使能位或计时器使能位;第 21 ~ 11 位,表示计数器当前值或计时器当前值;第 10 ~ 0 位,表示计数器予置值或计时器予置值。

[0099] 指令集共五种,具体如下:

[0100] (1) 位逻辑运算指令集包括:位与操作指令、位与非操作指令、位或操作指令、位异或操作指令以及位异或非操作指令;该指令集依据操作数来源分为两类:第一、操作数有一方是来自累加器堆栈,另一方来自寄存器组,如 ANDBIT CR, RS, Rm;第二、操作数两方都来自寄存器组,此位逻辑运算指令主要位于可跳转位逻辑运算后的第一个位逻辑运算,如 ANDBIT RS1, RS2, Rm。该指令集依据指令结果存储地址分为两类:第一、结果存储在累加器堆栈中,如 ANDBIT RS1, R2, CR;第二、结果存储在寄存器组的内部寄存器当中,如 ANDBIT RS1, RS2, Rm。

[0101] (2) 可跳转位逻辑运算指令集(表 6),位累与操作指令、位累与非操作指令、位累或操作指令、位累或非操作指令、可跳转位逻辑运算指令跳转结束符、累加器堆栈栈顶操作数出栈并求或指令;依据是否为位逻辑运算指令,分两类,第一类、位逻辑运算指令:ANDCRBIT、ANDNCRBIT、ORCRBIT、ORNCRBIT;第二类、可跳转位逻辑运算指令的结束符——CR 或 PUSHOR。依据符合 IEC\_61131-3 标准的可编程控制器指令表编程语言指令 -ORLD,对应的可编程控制器专用指令集指令为 PUSHOR,用于对前几项操作数输入指令 LD 或 LDN 输入到

累加器堆栈中的值进行或运算,并将运算的值输入返回累加器堆栈当中。

[0102] (3) 跳转调用指令集(表 7),包括无条件跳转指令、符合条件跳转指令、不符合条件跳转指令、无条件功能块调用指令、符合条件功能块调用指令、不符合条件功能块调用指令、无条件从功能或功能块返回指令、符合条件功能或功能块返回指令、不符合条件功能或功能块返回指令。依据是否带条件位分为两类,第一类、带条件位,包括 JMPCBIT、JMPCNBIT、CALCBIT、CALCNBIT、RETCBIT、RETCNBIT;第二类、不带条件位,包括 JMBIT、CALBIT、RETBIT。

[0103] (4) 存取指令(表 8),依据存取地址分为三类,第一类、取数地址为功能块存储单元,其指令包括:LOADQBIT、LOADNQBIT、STOREQBIT、STORENQBIT;第二类、存取数地址为存储器,其指令包括:LOADMBIT、STOREMBIT、LOADNMBIT、STORENMBIT;第三类、存取数地址为 I\O 端口,其指令包括:LOADIOBIT、STOREIOBIT、LOADNIOBIT、STORENIOBIT;第四类、存取数地址为内部寄存器,其指令包括:LOADRMBIT、LOADNRMBIT、STORERMBIT、STORENRMBIT。

[0104] 存取指令集包括:

[0105] LOADQBIT:取功能块单元的布尔输出值到累加器堆栈中;

[0106] STOREQBIT:将功能块单元的布尔输出值存储到功能块寄存器组中;

[0107] LOADNQBIT:取功能块单元的输出值后取反输出到累加器堆栈中;

[0108] STORENQBIT:将功能块单元的输出值取反存储到功能块寄存器组中;

[0109] LOADMBIT:从存储器单元取操作数到寄存器组或功能块寄存器组中;

[0110] STOREMBIT:将寄存器组或功能块寄存器组的数据直接存储到存储器单元中;

[0111] LOADNMBIT:从存储器单元取操作数取反后存储到寄存器组或功能块寄存器组中;

[0112] STORENMBIT:将寄存器组或功能块寄存器组的数据取反后存储到存储器单元中;

[0113] LOADIOBIT:从 I\O 数据存储器直接取数到寄存器组中;

[0114] STOREIOBIT:将寄存器组数据直接送到 I\O 数据存储器中;

[0115] LOADNIOBIT:从 I\O 数据存储器取的数据取反后存储到寄存器组中;

[0116] STORENIOBIT:将寄存器组数据取反后送到从 I\O 数据存储器中;

[0117] LOADRMBIT:将寄存器组的内部寄存器的值压栈到累加器堆栈中;

[0118] STORERMBIT:将累加器堆栈中的值出栈到寄存器组的内部寄存器中;

[0119] LOADNRMBIT:将寄存器组的内部寄存器的值取反压栈到累加器堆栈中;

[0120] STORENRMBIT:将累加器堆栈中的值取反出栈到寄存器组的内部寄存器中;

[0121] (5) 功能块指令集(表 9),包括置位优先功能块指令、复位优先功能块指令、上升沿检测功能块指令、下降沿检测功能块指令、加计数器功能块指令、减计数器功能块指令、加减计数器功能块指令、脉冲功能块指令、接通延时功能块指令、断开延时功能块指令、实时时钟功能块指令;按功能分为三大类,第一类、置位复位功能;第二类、边沿检测功能;第三类、计数器功能;第四类、计时器功能。

[0122] 为提高可编程控制器程序执行速度,本发明设计的可编程控制器专用指令集处理器体系结构,通过对可跳转位逻辑运算指令的处理,可加速可编程控制器专用指令集处理器对位逻辑运算指令的操作;并可在处理器内执行功能块指令,节省了对其输入输出数据的存取时间,提高了可编程控制器专用指令集处理器的实时性与精确度;其寄存器组,针对

该可编程控制器专用指令集处理器处理开关量数据的特点,采用位编址方式,能够加快数据存取的速度;针对功能块指令数据的特征,另外设计了功能块寄存器组;数据存储器针对该可编程控制器专用指令集处理器的数据特征,分别对开关量数据及功能块数据进行编址。该可编程控制器专用指令集处理器包括:指令存储器、指令计数器、指令寄存器、指令译码器、控制单元、功能块单元、功能块寄存器组、数据存储器、寄存器组、位处理器、跳转调用指令及存取指令处理单元、I\O 数据存储器;该处理器采用四级流水线,取指、译码、执行、回写。

[0123] (1) 指令存储器,其指令存储容量设定为 64KB;因指令格式采用 32 位,则一条指令需要一个字的单元存储,该指令存储器则可存储  $2^{11}$  条指令;该指令存储器采用小端方式存储,即低地址存储低端数据,高地址存储高端数据。

[0124] (2) 指令计数器(图 6),该指令计数器中当前 PC 值分为三类:第一,当前 PC 值为下一条指令地址,即当前 PC 值=上一条指令地址+4;第二,当遇到可跳转位逻辑运算指令,则当前 PC 值=上一条指令地址+跳转偏移量;第三,当遇到跳转调用指令时,则当前 PC 值=上一条指令地址+跳转调用偏移量。其可跳转位逻辑指令的处理流程图如图 5 所示,由指令译码器向位处理器输入组码,位处理器通过组码识别是否为可跳转位逻辑指令;

[0125] (3) 控制单元,控制可编程控制专用指令集处理器的四个流水阶段,即在取指阶段,控制指令计数器的操作;在译码阶段,控制指令译码器的操作;在执行阶段,接收指令译码器操作码中的组码,分别对位处理器、跳转调用指令及存取指令处理单元和功能块单元进行控制;并控制与通用处理器进行中断信号的接收或发送;在回写阶段,控制跳转调用指令及存取指令处理单元和功能块单元中存取指令的回写操作。

[0126] (4) 功能块单元,该功能块单元主要处理功能块指令,如置位优先功能块指令、复位优先功能块指令、上升沿检测功能块指令、下降沿检测功能块指令、加计数器功能块指令、减计数器功能块指令、加减计数器功能块指令、脉冲功能块指令、接通延时功能块指令、断开延时功能块指令、实时时钟功能块指令;其功能块单元中每类功能块可并行处理 1~16 个相同功能块,符合可编程控制器程序需执行多个功能块指令的特征;功能块单元与功能块寄存器组相连,可节约数据存取时间,提高可编程控制器计时器的准确度及可编程控制器程序执行速度;该功能块单元输入信号有来自控制单元的控制信号;及来自指令译码器的信号,包括:操作码 OP[3:0],置位优先的信号 S1、复位优先信号 R1、置位信号 S、复位信号 R、时钟信号 CLK、加计数器上升沿触发信号 CU、减计数器上升沿触发信号 CD、装入加计数器 LDU 初始数使能信号、减计数器初始数使能信号 LDD、加减计数器初始数使能信号 LDUD、脉冲功能块使能信号 INTp、接通延时功能块使能信号 INTON、断开延时功能块使能信号 INTOF、实时时钟功能块使能信号 INRTC;有来自功能块寄存器组的信号,包括:计数器予置值 PV[7:0]、计时器予置值 PT[7:0]、实时时钟予置值 PDT[7:0];其功能块输出信号为输出到功能块寄存器组的信号,信号包括:当前计数器值 CV[7:0]、计时器结束时间 ET[7:0]、实时时钟当前日期和时间 CDT[7:0]。

[0127] (5) 功能块寄存器组,该寄存器组分为四类,包括触发器功能块寄存器、边沿检测功能块寄存器、计数器功能块寄存器、计时器功能块寄存器;每类功能块可最多并行存储十六个同类功能块数据;该寄存器组可存储的数据包括:各功能块的输入信号与输出信号;该寄存器组可以以位或字节的方式存取数据;当该寄存器组有一功能块寄存器数据满

时,可以请求外设功能块单元执行,并与数据存储器进行数据传输;其编址方式,如表 10 所示,共用 11 位对寄存器组进行编址,通过第 10~9 位寻址功能块类型,第 8~5 位寻址第几个功能块单元,第 4 位用于对寄存器组数据存取方式编址,第 3~0 位寻址输入输出数据。

[0128] (6) 数据存储器,其存储总容量为 64KB,分为输入存储器、输出存储器、内部存储器及功能块存储器。该存储器通过十六位编址对这四种存储器寻址。输入、输出、内部存储器主要存储位数据,功能块存储器主要存储内部及外设功能块的输入输出数据,每类功能块数据可并行存储 256 个。下面是对输入、输出及内部存储器的编址方式:第 15 位用于区别输入、输出、内部存储器和功能块存储器,其值为零是则为输入、输出、内部存储器;第 14~13 位用于对输入、输出及内部存储器编址,输入存储器编码为 00、输出存储器编码为 01、内部存储器编码为 10;第 12~3 位用于对各存储器字节寻址,第 2~0 位用于对存储器位寻址。对功能块存储器编址(表 11),第 15 位,其值为 1,表示对功能块存储器寻址;第 14~13 位用于对四类功能块的寻址,触发器编址为 00、边沿检测编址为 01、计数器编址为 10、计时器编址为 11;第 12~5 位主要对指定第几个功能块数据寻址;第 4 位主要表示对存储数据的寻址方式,如为位数据时,则对一个字节的位进行寻址;第 3~0 位表示要各输入输出数据的编址。

[0129] (7) 寄存器组,该寄存器组分为输入寄存器、内部寄存器、状态寄存器、累加器堆栈、累加器堆栈指针寄存器、指令寄存器。将寄存器组分为三类进行编址,第一类为输入、内部寄存器;第二类为状态寄存器、累加器堆栈及累加器指针寄存器;第三类为指令寄存器。对第一类寄存器编址如下:第 7~6 位表示对三类寄存器的编址,第一类寄存器编址为 00;第 5 位表示对输入、内部寄存器的编址,输入寄存器编址为 0,内部寄存器编址为 1;第 4~3 表示对字节寻址;第 2~0 表示对位寻址。对第二类寄存器编址如下:第 7~6 位,第二类寄存器编址为 01;第 5 位表示对状态寄存器、累加器堆栈的编址,累加器堆栈编址为 0,状态寄存器编址为 1;第 4~3 表示对字节寻址;第 2~0 表示对位寻址。特别地规定该第二类寄存器编址中第 4~3 位编址为 00 的字节存放累加器堆栈指针寄存器数据,即为累加器堆栈指针寄存器,则累加器堆栈指针寄存器地址为 01000000。对第三类寄存器编址如下:第 7~6 位,第三类寄存器编址为 01;第 5~0 表示对字节寻址,即可寻址 32 个字节,又指令格式为 32 位,占 4 个字节,则指令寄存器可存储八条指令;

[0130] (8) 位处理器(如图 3 所示),该位处理器特征是采用累加器堆栈的形式存储操作数,由于位处理器经常要用到前一次甚至前几次累加器堆栈中的操作数,采用符合位处理器执行特征的堆栈形式保存操作数,并通过累加器堆栈指针寄存器中的地址来区别当前操作数与前几次操作数。该位处理器处理位逻辑运算指令及可跳转位逻辑运算指令;采用累加器堆栈的形式存储位处理器所需操作数及其运算结果;其累加器堆栈栈顶值来自寄存器组、功能块寄存器组、I/O 数据存储器及位处理器运算结果;当位处理器执行位逻辑运算指令时,从累加器堆栈栈顶或寄存器组中取操作数后,进行位逻辑运算操作,将运算结果存储到累加器堆栈栈顶或寄存器组中。当位处理器执行可跳转位逻辑运算指令时,若该可跳转位逻辑运算指令为位累与运算指令、位累与非运算指令、位累或运算指令、位累或非运算指令,且满足跳转条件时,位处理器将跳转偏移量发送至指令计数器中,并执行相应的位逻辑运算,将运算结果保存在累加器堆栈栈顶。位处理器的运算结果,经跳转调用指令及存取



指令处理单元执行存取指令后,分别存储在状态寄存器、数据存储器、寄存器组中的内部寄存器以及 I\O 数据存储器当中。

[0131] 如图 4 所示,位处理器处理位逻辑运算指令及可跳转位逻辑运算指令包括以下步骤:

[0132] 位处理器处理位逻辑运算指令及可跳转位逻辑运算指令;采用累加器堆栈的形式存储位处理器所需操作数及其运算结果;其累加器堆栈栈顶值来自寄存器组、功能块寄存器组、I\O 数据存储器及位处理器运算结果;

[0133] 当位处理器执行位逻辑运算指令时,从累加器堆栈栈顶或寄存器组中取操作数后,进行位逻辑运算操作,将运算结果存储到累加器堆栈栈顶或寄存器组中;

[0134] 执行完成后存储位处理器的运算结果;

[0135] 当位处理器执行可跳转位逻辑运算指令时,按可跳转位逻辑运算指令的执行过程执行,执行完成后存储位处理器的运算结果;

[0136] 如图 5 所示,跳转位逻辑运算指令的执行过程为:

[0137] 如果是可跳转位逻辑指令,则指令译码器向位处理器输入操作码;

[0138] 由位处理器判断该可跳转位逻辑指令是否为位累与操作指令、位累与非操作指令、位累或操作指令、位累或非操作指令;

[0139] 若为位累与操作指令,且累加器堆栈的栈顶值为 0,则执行与操作,将与运算结果“0”存储在累加器堆栈的栈顶,并设置指令格式中 jump condition 跳转条件位为 1,设置 jump offset 跳转偏移量值为可跳转位逻辑运算指令跳转结束符与该位累与操作指令的偏移量;若为位累与操作指令,但不满足累加器堆栈的栈顶值为 0,则只执行与操作,将执行结果存储在累加器堆栈中,其跳转条件位为 0,不执行跳转操作。

[0140] 若为位累与非操作指令,且累加器堆栈的栈顶值为 0,则执行与非操作,将与非运算结果“1”存储在累加器堆栈的栈顶,并设置指令格式中 jumcondition 跳转条件位为 1,设置 jump offset 跳转偏移量值为可跳转位逻辑运算指令跳转结束符到该位累与非操作指令的偏移量;若为位累与非操作指令,但不满足累加器堆栈的栈顶值为 0,则只执行与非操作,将执行结果存储在累加器堆栈中,其跳转条件位为 0,不执行跳转操作。

[0141] 若为位累或操作指令,且累加器堆栈的栈顶值为 1,则执行或操作,将或运算结果“1”存储在累加器堆栈的栈顶,并设置指令格式中 jump condition 跳转条件位为 1,设置 jump offset 跳转偏移量值为可跳转位逻辑运算跳转结束符指令或累加器堆栈中的操作数出栈并求或指令到该位累或操作指令的偏移量;若为位累与操作指令,但不满足累加器堆栈的栈顶值为 1,则只执行或操作,将执行结果存储在累加器堆栈中,其跳转条件位为 0,不执行跳转操作。

[0142] 若为位累或非操作指令,且累加器堆栈的栈顶值为 1,则执行或非操作,将或非运算结果“0”存储在累加器堆栈的栈顶,并设置指令格式中 jumcondition 跳转条件位为 1,设置 jump offset 跳转偏移量值为可编程控制器可跳转位逻辑运算跳转结束符到该位累或非操作指令的偏移量;若为位累与操作指令,但不满足累加器堆栈的栈顶值为 1,则只执行或非操作,将执行结果存储在累加器堆栈中,其跳转条件位为 0,不执行跳转操作。

[0143] 如图 7 所示,本发明高性能可编程控制器专用指令集处理器体系结构的实现方法按照该可编程控制器指令集处理器的四级流水线方式,包括以下步骤:

[0144] 取指阶段:依据指令计数器计算的指令地址,从指令存储器取指令至指令寄存器当中;

[0145] 译码阶段:从指令寄存器取出指令送入指令译码器进行译码,通过指令中操作数地址,取出操作数;并依据指令中组码,将相应的操作码及操作数送至对应的处理单元中;

[0146] 执行阶段:依据指令译码器发出的操作码及操作数,位处理器、功能块单元、跳转调用指令及存取指令处理单元给予相应处理;如执行的指令不是可编程控制器专用指令集处理器的指令时,则由控制单元向通用处理器发出中断请求,该可编程控制器专用指令集处理器可继续执行与通用处理器指令无依赖关系的指令,当通用处理器向可编程控制器专用指令集处理器发送中断请求时,可编程控制器专用指令集处理器将保存通用处理器的执行结果到状态寄存器中;另外存储器访问阶段与执行阶段合为一个流水线阶段,由跳转调用指令及存取指令处理单元执行存储器访问;

[0147] 回写阶段:将可编程控制器专用指令集处理器的结果,依据存取指令类型写回累加器堆栈、内部寄存器、数据存储器内部寄存器或 I\O 存储器当中。

[0148]

GP	OP	CRYN	condition	Rm\ CR	CR	Rs2\Rm
组码 2 位 (00)	操作 码 4 位	1 位	1 位	Address 占 1 字节	Address 1 字节	Address 占 1 字节
31...30	29...26	25	24	23...16	15.....8	7 .....0

[0149] 表 1

[0150]

GP	OP	Jump condition	condition	Jump offset	CR
组码 2 位 (01)	操作码 4 位	1 位	1 位	16 位	Address 占 1 字节
31...30	29...26	25	24	23 .....8	7 .....0

[0151] 表 2

[0152]

GP	OP	i	Condition	CR/Rs	Mem\IO \fun Address
组码 2 位 (10)	操 作 码 4 位	I=0(表示跳 转调用指令) I=1(表示存 取指令)	条件位	CR 或寄存器地址 (8 位)	跳转调用偏移位或数据 存储器、I/O 数据存储器的 存取地址(16 位)以 及功能块寄存器地址(11 位)
31...30	29...26	25	24	23.....16	15.....0

[0153] 表 3

[0154]

GP	OP	R1/CU	S1/CD	R/CLK	S/LD/IN	Rd1	Rs1
组码 2位 (11)	操作码 4位	复位优先/ 加计数器 输入的上 升沿触发	置位优先/ 减计数器 输入的上 升沿触发	时钟	装入加减 计数器初 始数使能/ 计时器使 能	CV\CDT\ET 计数器当前 值 INT\当前 的日期和时 间 DT	PV\PT\PDT\ 予置计数器、 计时器值 INT
31..30	29...26	25	24	23	22	21.....11	10.....0

[0155] 表 4

[0156]

操作符	操作码编 码	描述	举例
ANDBIT	000000	位与操作	ANDBIT CR, RS, Rm ANDBIT RS1, RS2, Rm/* 将该指令结果保 存在累加器堆栈中 */ ANDBIT RS1, RS2, CR/* 该指令为可跳转 位逻辑指令后的位逻辑运算指令,其计 算结果 保存在累加器堆栈中 */ ANDBIT RS1, CR/* 操作数来自累加器堆 栈,结果保存在累加器 堆栈中 */
ANDNBIT	000001	位与非操作	ANDNBIT CR, RS, Rm ANDNBIT RS1, RS2, Rm ANDNBIT RS1, RS2, CR/* 该指令为可跳转 位 逻辑指令后的位逻辑运算指令,其计 算结果保存在累加器堆栈中 */ ANDNBIT RS1, CR
ORBIT	000010	位或操作	ORBIT CR, RS, Rm ORBIT RS1, RS2, Rm ORBIT RS1, RS2, CR/* 该指令为可跳转位 逻辑 指令后的位逻辑运算指令,其计算 结果保存在累加器堆栈中 */ ORBIT RS1, CR/* 操作数 来自累加器堆 栈,结果保存在累加器堆栈中 */
ORNBIT	000011	位或非操作	ORNBIT CR, RS, Rm ORNBIT RS1, RS2, Rm ORNBIT RS1, RS2, CR/* 该指令为可跳转 位 逻辑指令后的位逻辑运算指令,其计 算结果保存在累加器堆栈中 */ ORNBIT RS1, CR
XORBIT	000100	异或操作	XORBIT CR, RS, Rm XORBIT RS1, RS2, Rm XORBIT RS1, RS2, CR/* 该指令为可跳转 位逻 辑指令后的位逻辑运算指令,其计 算结果保存在累加器堆栈中 */ XORBIT RS1, CR/* 操 作数来自累加器堆 栈,结果保存在累加器堆栈中 */
XORNBIT	000101	位异或非操作	XORNBIT CR, RS, Rm XORNBIT RS1, RS2, Rm XORNBIT RS1, RS2, CR/* 该指令为可跳转 位 逻辑指令后的位逻辑运算指令,其计 算结果保存在累加器堆栈中 */ XORNBIT RS1, CR

[0157] 表 5

[0158]

操作符	操作码编码	描述	举例
ANDCRBIT	010000	位累与操作	ANDCRBIT
ANDNCRBIT	010001	位累与非操作	ANDNCRBIT
ORCRBIT	010010	位累或操作	ORCRBIT
ORNCRBIT	010011	位累或非操作	ORNCRBIT
CR	010100	ANDCRBIT 等可跳转位 逻辑运算指令跳转结 束符	CR
PUSHOR	010101	累加器堆栈栈顶操作 数出栈并求或	PUSHOR

[0159] 表 6

[0160]

操作符	操作码编码	描述	举例
JMPBIT	1000000	无条件跳转指令	JMPBIT STATE/* 无条件跳转到哪个状态*
JMPCBIT	1000010	符合条件跳转指令	JMPCBIT STATE/* 符合某个条件则跳转到哪个状态*
JMPCNBIT	1000100	不符合条件跳转指令	JMPCNBIT STATE/* 不符合某个条件则跳转到哪个状态*
CALBIT	1000110	无条件功能块调用指令	CALBIT Counter/* 无条件调用计数器功能块*/
CALCBIT	1001000	符合条件功能块调用指令	CALCBIT Counter/* 符合条件则调用计数器功能块*/
CALCNBIT	1001010	不符合条件功能块调用指令	CALCNBIT Counter/* 不符合条件则调用计数器功能块*/
RETBIT	1001100	无条件从功能块返回指令	RETBIT
RETCBIT	1001110	符合条件从功能块返回指令	RETCBIT
RETCNBIT	1010000	不符合条件从功能块返回指令	RETCNBIT

[0161] 表 7

[0162]

操作符	操作码编码	描述	举例
LOADQBIT	100001	取功能块单元的布尔输出值到累加器堆栈中	LOADQBIT Q, CR
STOREQBIT	100011	将功能块单元的输出值存储到寄存器组中；	STOREQBIT Q, [mem]/Rm
LOADNQB	100101	取功能块单元的输出值后取反到累加器堆栈中	LOADNQB Q, CR
STORENQB	100111	将功能块单元的输出值取反存储到寄存器组中；	STORENQB Q, [mem]/Rm
LOADMBIT	1001001	从存储器单元取操作数到寄存器组中	LOADMBIT [mem], Rs1/CR/* 从存储器中获取操作数*/
STOREMBIT	1001011	将寄存器数据直接存储到存储器单元中	STOREMBIT Rm/CR, [mem]/* 数据存储到存储器当中*/
LOADNMBIT	1001101	从存储器单元取操作数取反后存储到寄存器组中	LOADNMBIT [mem], CR/* 从存储器中获取操作数*/
STORENMBIT	1001111	将寄存器数据取反后存储到存储器单元中	STORENMBIT Rm/CR, [mem]/* 数据存储到存储器当中*/
LOADIOBIT	1010001	从 I\O 数据存储器直接取数到寄存器组中	LOADIOBIT [I/O], CR/* 取 IO 数据到寄存器*/
STOREIOBIT	1010011	将寄存器组数据直接送到 I\O 数据存储器中	STOREIOBIT Rm/CR, [I/O]/* 取寄存器数据到 IO 端口*/
LOADNIOBIT	1010101	从 I\O 数据存储器取的数据取反后存储到寄存器组中	LOADNIOBIT [I/O], Rs1/* 取 IO 数据到寄存器*/
STORENIOBIT	1010111	将寄存器组数据取反后送到从 I\O 数据存储器中	STORENIOBIT Rm/CR, [I/O]/* 取寄存器数据到 IO 端口*/
LOADRMBIT	1011001	将寄存器组的内部寄存器的值压栈到累加器堆栈中	LOADRMBIT Rm, CR
STORERMBIT	1011011	将累加器堆栈中的值出栈到寄存器组的内部寄存器中	STORERMBIT CR, Rm
LOADNRMBIT	1011101	将寄存器组的内部寄存器的值取反压栈到累加器堆栈中	LOADNRMBIT Rm, CR
STORENRMBIT	1011111	将累加器堆栈中的值取反出栈到寄存器组的内部寄存器中	STORENRMBIT CR, Rm

[0163] 表 8

[0164]

操作符	操作码编码	描述	举例
FB SR	110000	置位优先功能块指令	FB SR function block name.S
FB RS	110001	复位优先功能块指令	FB RS function block name.S
FB R_TRIG	110010	上升沿检测功能块指令	FB R_TRIG function block name.clk
FB F_TRIG	110011	下降沿检测功能块指令	FB F_TRIG function block name.clk
FB CTU	110100	加计数器功能块指令	FB CTU function block name.PV
FB CTD	110101	减计数器功能块指令	FB CTD function block name.PV
FB CTUD	110110	加减计数器功能块指令	FB CTUD function block name.PV
FB TP	110111	脉冲功能块指令	FB TP function block name.PT
FB TON	111000	接通延时功能块指令	FB TON function block name.PT
FB TOF	111001	断开延时功能块指令	FB TOF function block name.PT
FB RTC	111010	实时时钟功能块指令	FB RTC function block name.PT

[0165] 表 9

[0166]

功能块类型编址(2位)	功能块单元数编址(4位)	位或字节存储方式编址(1位)	寄存器编址(4位) 以位或字节为单元
触发器-00	16个触发器数据可存储在寄存器当中	0(以位方式编址)	R: 复位输入-0000 S: 置位输入-0001 R1: 复位优先-0010 S1: 置位优先-0011 Q1: 输出-0100 注: 以上为对选取的功能块单元的位单元进行编址
边沿检测-01	16个边沿检测数据可存储在寄存器当中	0(以位方式编址)	CLK: 时钟-0000 Q: 输出-0001 注: 以上为对选取的功能块单元的位单元进行编址
计数器-10	16个计数器数据可存储在寄存器当中	0(以位方式编址)	CU: 用于加计数器输入上升沿触发信号-0000 CD: 用于减计数器的输入上升沿触发信号-0001 LD: 装入(计数器)值-0010 QU: 输出(加计数器)-0011 QD: 输出(减计数器)-0100 Q: 输出-0101 注: 以上为对选取的功能块单元的位单元进行编址
		1(以字节方式编址)	PV: 予置(计数器)值-0000 注: PV 予置值需要一个字来存储 CV: 当前(计数器)值-0100
计时器-11	16个定时器数据可存储在寄存器当中	0(以位方式编址)	IN: 输入(计时器-0000) Q: 输出-0001
		1(以字节方式编址)	PT: 预设的时间值-0000 ET: 结束时间输出-0100
		1(以字节方式编址)	PDT: 预设的时期和时间值 DT-0000 CDT: 当前的日期和时间 DT-0111

[0167] 表 10

[0168]

存储器区域 编址 (1 位)	功能块类型编 址(2 位)	功能块单元 数编址 (8 位)	位或字节存储 方式编址(1 位)	寄存器编址 (4 位) 以位或字节为单元
特殊功能块 编码-1	触发器-00	256 个触发器数据可存储在寄存器当中	0 (以位方式编址)	R: 复位输入-0000 S: 置位输入-0001 R1: 复位优先-0010 S1: 置位优先-0011 Q1: 输出-0100 注: 以上为对选取的功能块单元的位单元进行编址
	边沿检测-01	256 个边沿检测数据可存储在寄存器当中	0 (以位方式编址)	CLK: 时钟-0000 Q: 输出-0001 注: 以上为对选取的功能块单元的位单元进行编址
	计数器-10	256 个计数器数据可存储在寄存器当中	0 (以位方式编址)	CU: 用于加计数器输入上升沿触发信号-0000 CD: 用于减计数器的输入上升沿触发信号-0001 LD: 装入 (计数器) 值-0010 QU: 输出 (加计数器) -0011 QD: 输出 (减计数器) -0100 Q: 输出-0101 注: 以上为对选取的功能块单元的位单元进行编址
			1 (以字节方式编址)	PV: 予置 (计数器) 值-0000 注: PV 予置值需要一个字来存储 CV: 当前 (计数器) 值-0100
	计时器-11	256 个定时器数据可存储在寄存器当中	0 (以位方式编址)	IN: 输入 (计时器-0000) Q: 输出-0001
			1 (以字节方式编址)	PT: 预设的时间值-0000 ET: 结束时间输出-0100
			1 (以字节方式编址)	PDT: 预设的时期和时间值 DT-0000 CDT: 当前的日期和时间 DT-0111

[0169] 表 11

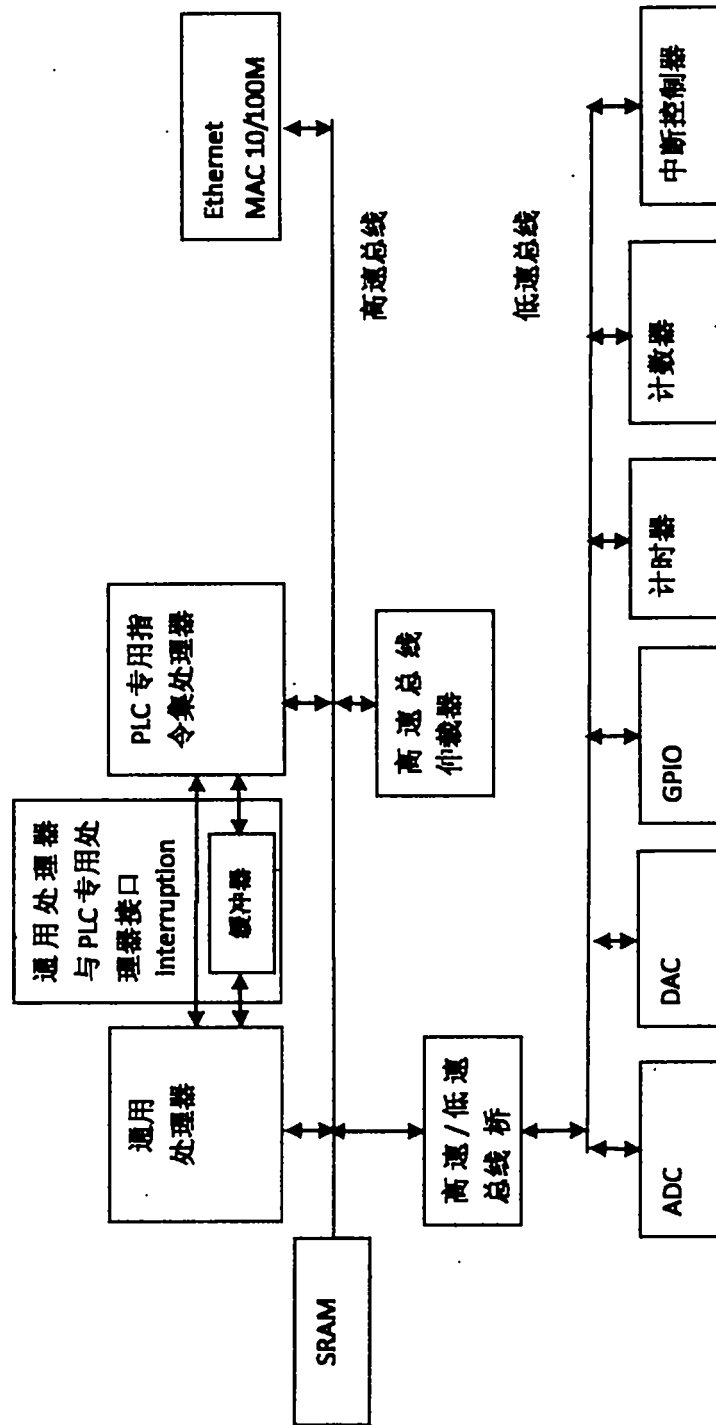


图 1



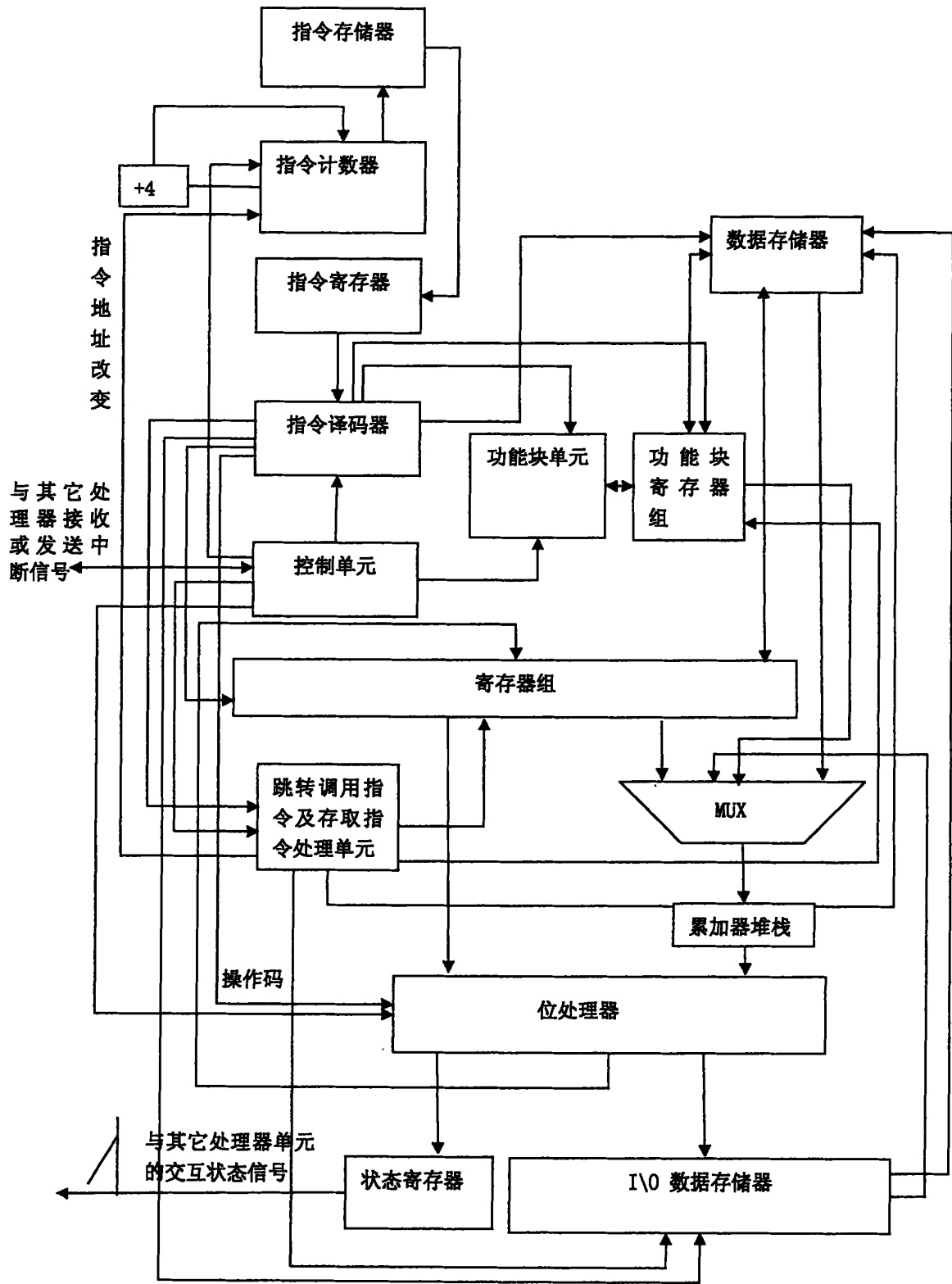


图 2

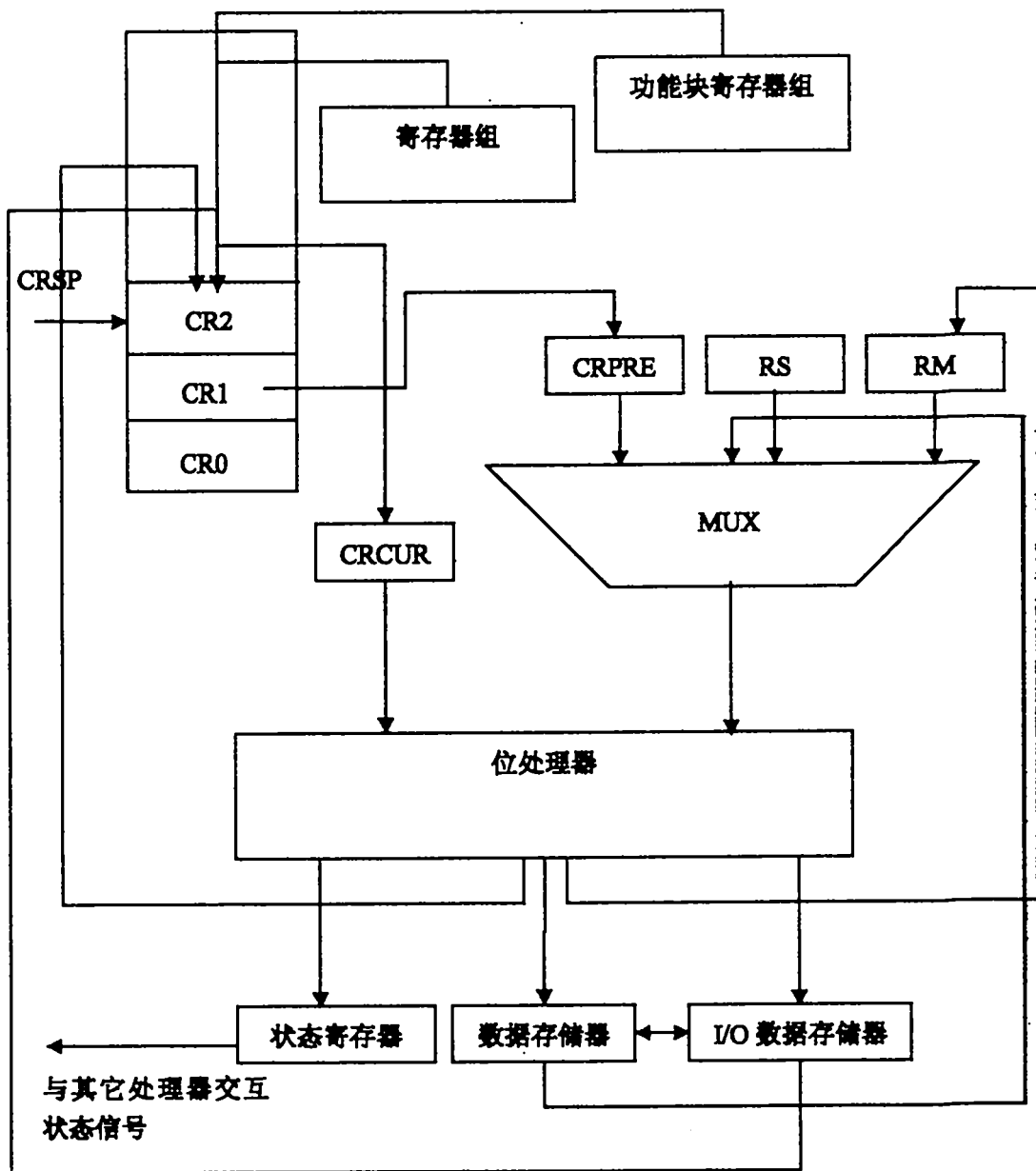


图 3

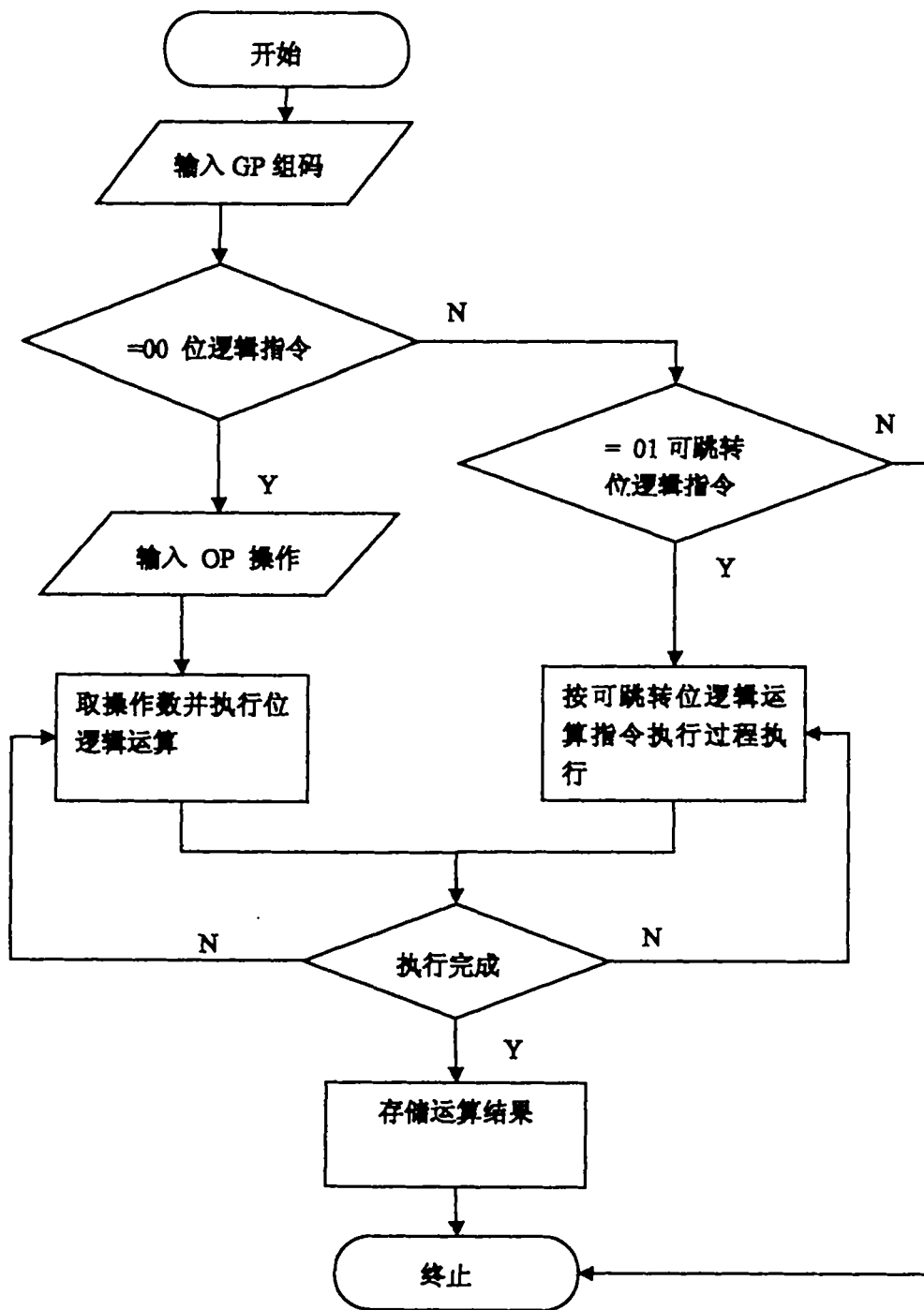


图 4

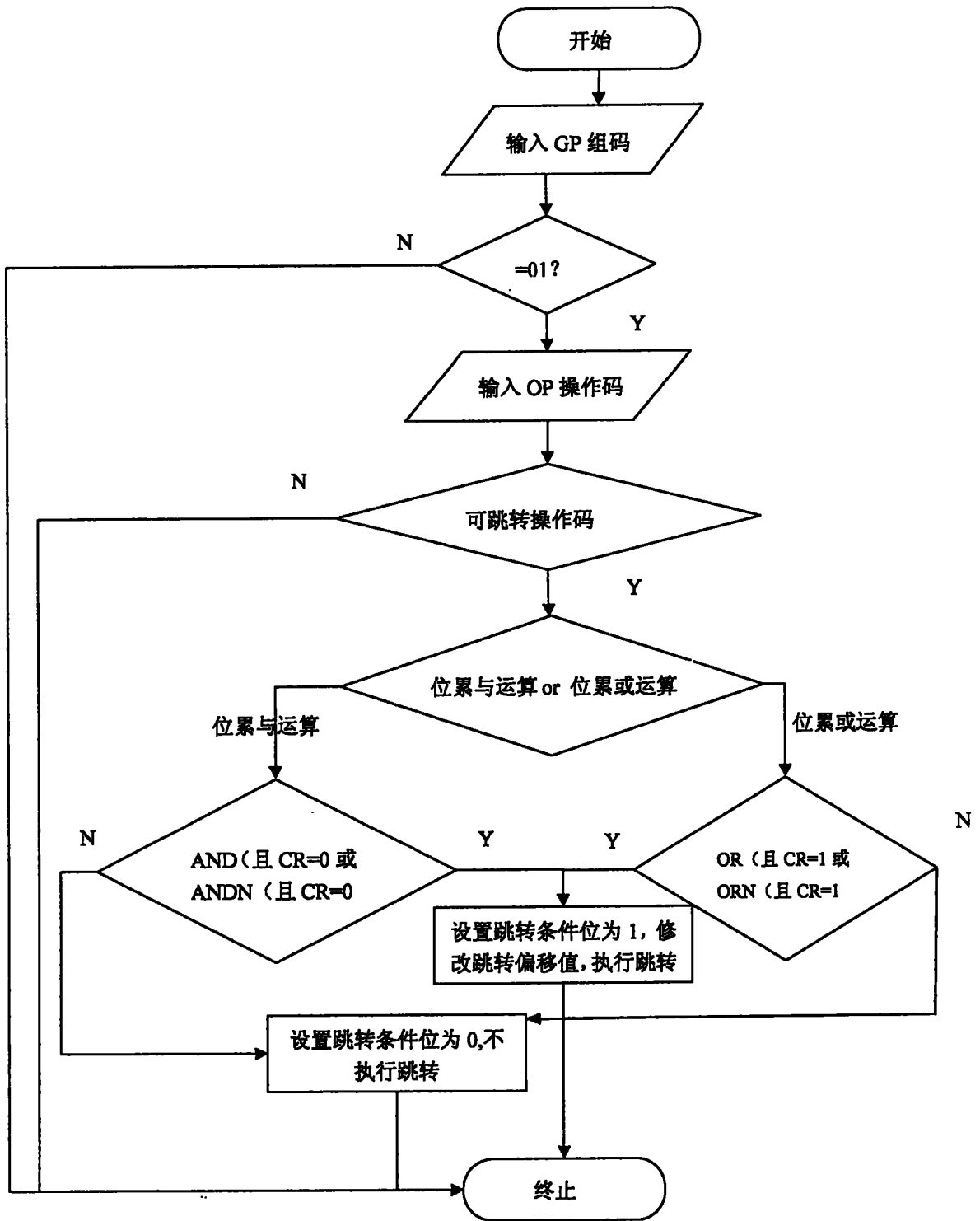


图 5

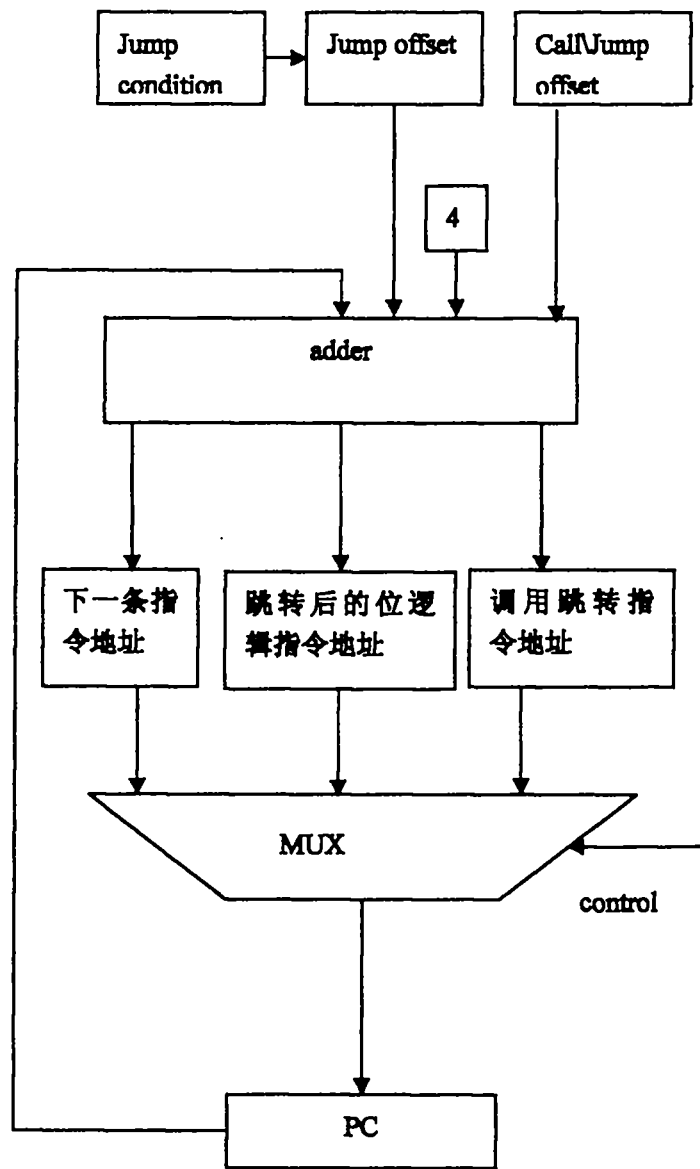


图 6

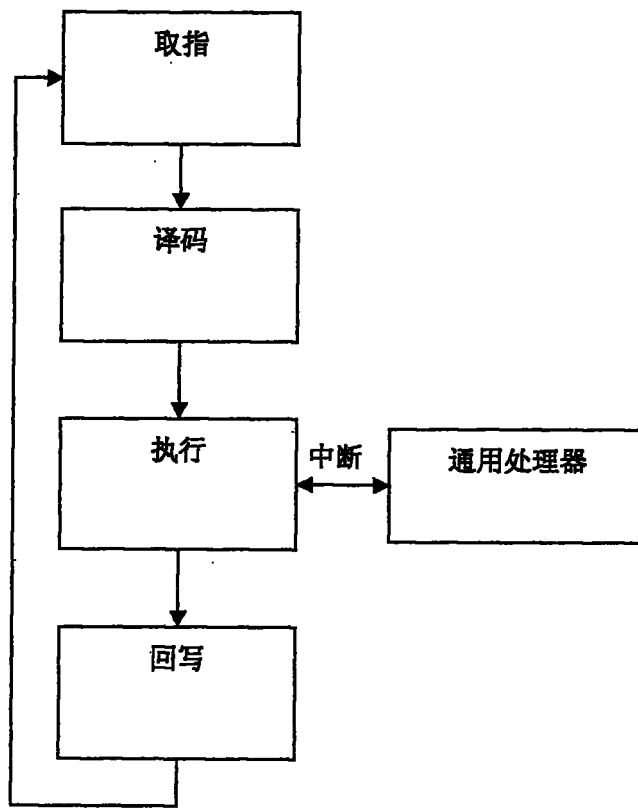


图 7