



(12)发明专利申请

(10)申请公布号 CN 106997285 A

(43)申请公布日 2017.08.01

(21)申请号 201610833829.9

(22)申请日 2016.09.20

(71)申请人 遨博(北京)智能科技有限公司

地址 102300 北京市门头沟区莲石湖西路
98号石龙阳光大厦5号楼301A1

(72)发明人 黄真 潘毅飞

(74)专利代理机构 北京柏杉松知识产权代理事
务所(普通合伙) 11413

代理人 项京 马敬

(51)Int.Cl.

G06F 9/30(2006.01)

G06F 9/44(2006.01)

G06F 9/45(2006.01)

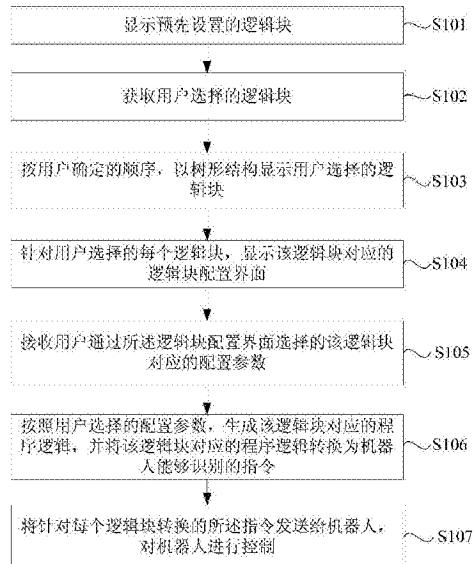
权利要求书2页 说明书14页 附图6页

(54)发明名称

一种控制机器人的方法及装置

(57)摘要

本发明实施例提供了一种控制机器人的方法及装置，所述方法包括：显示预先设置的逻辑块；获取用户选择的逻辑块；按用户确定的顺序，以树形结构显示用户选择的逻辑块；针对用户选择的每个逻辑块，显示该逻辑块对应的逻辑块配置界面；接收用户通过所述逻辑块配置界面选择的该逻辑块对应的配置参数；按照用户选择的配置参数，生成该逻辑块对应的程序逻辑，并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令；将针对每个逻辑块转换的所述指令发送给机器人，对机器人进行控制。应用本发明实施例，简化了控制机器人的方法，使得非专业人员在短时间内掌握控制机器人的方法。



1. 一种控制机器人的方法,其特征在于,包括:
 - 显示预先设置的逻辑块;其中,每个逻辑块对应一条控制逻辑语句;
 - 获取用户选择的逻辑块;
 - 按用户确定的顺序,以树形结构显示用户选择的逻辑块;
 - 针对用户选择的每个逻辑块,显示该逻辑块对应的逻辑块配置界面;
 - 接收用户通过所述逻辑块配置界面选择的该逻辑块对应的配置参数;
 - 按照用户选择的配置参数,生成该逻辑块对应的程序逻辑,并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令;
 - 将针对每个逻辑块转换的所述指令发送给机器人,对机器人进行控制。
2. 根据权利要求1所述的方法,其特征在于,所述逻辑块根据预设条件被划分为基础条件逻辑块和高级条件逻辑块,其中,所述预设条件包括所述逻辑块被选择的频率或者编辑所述逻辑块对应的控制逻辑语句的难易程度;
 - 所述显示预先设置的逻辑块的步骤,包括:
 - 按用户指令显示编程界面;
 - 当检测到所述编程界面中的基础条件逻辑块选项按钮被按下,则在所述编程界面中显示基础条件逻辑块;
 - 当检测到所述编程界面中的高级条件逻辑块选项按钮被按下,则在所述编程界面中显示高级条件逻辑块。
3. 根据权利要求2所述的方法,其特征在于,所述按用户确定的顺序,以树形结构显示用户选择的逻辑块的步骤,包括:
 - 按用户选择逻辑块的顺序,在所述编程界面的第一预设区域中,以树形结构由上至下显示用户选择的逻辑块;
 - 当检测到所述编程界面中的任一个逻辑块编辑按钮被按下,则执行该逻辑块编辑按钮对应的编辑操作,在所述树形结构中增加逻辑块或删除逻辑块在树形结构中的位置;所述逻辑块编辑按钮为:增加按钮或删除按钮。
4. 根据权利要求2所述的方法,其特征在于,所述针对用户选择的每个逻辑块,显示该逻辑块对应的逻辑块配置界面的步骤,包括:
 - 在所述编程界面的第二预设区域中,显示逻辑块配置界面;所述逻辑块配置界面中包含该逻辑块所有可选的参数,和/或该逻辑块的参数输入框。
5. 根据权利要求2所述的方法,其特征在于,所述按照用户选择的配置参数,生成该逻辑块对应的程序逻辑,并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令的步骤,包括:
 - 当检测到所述逻辑块配置界面中的确认按钮被按下,则按照用户选择的配置参数,生成该逻辑块对应的XML语言源代码;
 - 将该逻辑块对应的XML语言源代码转换为机器人能够识别的脚本语言代码。
6. 根据权利要求2所述的方法,其特征在于:所述的编程界面集成于机器人示教器软件中。
7. 一种控制机器人的装置,其特征在于,包括:
 - 第一显示单元,用于显示预先设置的逻辑块;其中,每个逻辑块对应一条控制逻辑语

句；

获取单元，用于获取用户选择的逻辑块；

第二显示单元，用于按用户确定的顺序，以树形结构显示用户选择的逻辑块；

第三显示单元，用于针对用户选择的每个逻辑块，显示该逻辑块对应的逻辑块配置界面；

接收单元，用于接收用户通过所述逻辑块配置界面选择的该逻辑块对应的配置参数；

转换单元，用于按照用户选择的配置参数，生成该逻辑块对应的程序逻辑，并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令；

控制单元，用于将针对每个逻辑块转换的所述指令发送给机器人，对机器人进行控制。

8. 根据权利要求7所述的装置，其特征在于，所述装置还包括：

划分单元，用于将所述逻辑块根据预设条件被划分为基础条件逻辑块和高级条件逻辑块，其中，所述预设条件包括所述逻辑块被选择的频率或者编辑所述逻辑块对应的控制逻辑语句的难易程度；

所述第一显示单元，包括：

第一显示子单元，用于按用户指令显示编程界面；

第二显示子单元，用于当检测到所述编程界面中的基础条件逻辑块选项按钮被按下，则在所述编程界面中显示基础条件逻辑块；

第三显示子单元，用于当检测到所述编程界面中的高级条件逻辑块选项按钮被按下，则在所述编程界面中显示高级条件逻辑块。

9. 根据权利要求8所述的方法，其特征在于，所述第二显示单元，包括：

第四显示子单元，用于按用户选择逻辑块的顺序，在所述编程界面的第一预设区域中，以树形结构由上至下显示用户选择的逻辑块；

第五显示子单元，用于当检测到所述编程界面中的任一个逻辑块编辑按钮被按下，则执行该逻辑块编辑按钮对应的编辑操作，在所述树形结构中增加逻辑块或删除逻辑块在树形结构中的位置；所述逻辑块编辑按钮为：增加按钮或删除按钮。

10. 根据权利要求8所述的方法，其特征在于，所述第三显示单元具体用于：

在所述编程界面的第二预设区域中，显示逻辑块配置界面；所述逻辑块配置界面中包含该逻辑块所有可选的参数，和/或该逻辑块的参数输入框。

11. 根据权利要求8所述的方法，其特征在于，所述转换单元，包括：

第一转换单元，用于当检测到所述逻辑块配置界面中的确认按钮被按下，则按照用户选择的配置参数，生成该逻辑块对应的XML语言源代码；

第二转换单元，用于将该逻辑块对应的XML语言源代码转换为机器人能够识别的脚本语言代码。

一种控制机器人的方法及装置

技术领域

[0001] 本发明涉及机器人技术领域,特别是涉及一种控制机器人的方法及装置。

背景技术

[0002] 目前,对传统机器人的控制,需要专业的技术人员编写复杂抽象的程序来实现对机器人的控制,为了简化对机器人的控制难度,主流的机器人大部分采用过程语言(即不需要编译,直接可以运行)对机器人进行控制。虽然过程语言与以前的复杂抽象的程序相比较,降低了部分难度,可是过程语言依旧难以被非专业人员掌握使用,所以出现了示教器来控制机器人的模式。

[0003] 现有技术大多采用示教器结合类过程语言的方式来控制机器人,具体控制机器人过程主要包括两个步骤:

[0004] 第一,用户在示教器的指导下,用类过程语言来编写机器人的控制逻辑。

[0005] 第二,示教器将用户用类过程语言编写的控制逻辑转换为解释性过程语言的指令发送给机器人,从而实现对机器人的控制。

[0006] 可见,当用户对机器人进行控制时,仍然需要用户用类过程语言来编写机器人的控制逻辑。这样的控制方式比较复杂,对用户本身的要求比较高,非专业人员难以通过短时间的学习,来掌握控制机器人的方法。

发明内容

[0007] 本发明实施例的目的在于提供一种控制机器人的方法及装置,以简化控制机器人的操作,降低对用户的要求。具体技术方案如下:

[0008] 一方面,本发明实施例公开了一种控制机器人的方法,包括:

[0009] 显示预先设置的逻辑块;其中,每个逻辑块对应一条控制逻辑语句;

[0010] 获取用户选择的逻辑块;

[0011] 按用户确定的顺序,以树形结构显示用户选择的逻辑块;

[0012] 针对用户选择的每个逻辑块,显示该逻辑块对应的逻辑块配置界面;

[0013] 接收用户通过所述逻辑块配置界面选择的该逻辑块对应的配置参数;

[0014] 按照用户选择的配置参数,生成该逻辑块对应的程序逻辑,并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令;

[0015] 将针对每个逻辑块转换的所述指令发送给机器人,对机器人进行控制。

[0016] 较佳的,所述逻辑块根据预设条件被划分为基础条件逻辑块和高级条件逻辑块,其中,所述预设条件包括所述逻辑块被选择的频率或者编辑所述逻辑块对应的控制逻辑语句的难易程度;

[0017] 所述显示预先设置的逻辑块的步骤,包括:

[0018] 按用户指令显示编程界面;

[0019] 当检测到所述编程界面中的基础条件逻辑块选项按钮被按下,则在所述编程界面

中显示基础条件逻辑块；

[0020] 当检测到所述编程界面中的高级条件逻辑块选项按钮被按下，则在所述编程界面中显示高级条件逻辑块。

[0021] 较佳的，所述按用户确定的顺序，以树形结构显示用户选择的逻辑块的步骤，包括：

[0022] 按用户选择逻辑块的顺序，在所述编程界面的第一预设区域中，以树形结构由上至下显示用户选择的逻辑块；

[0023] 当检测到所述编程界面中的任一个逻辑块编辑按钮被按下，则执行该逻辑块编辑按钮对应的编辑操作，在所述树形结构中增加逻辑块或删除逻辑块在树形结构中的位置；所述逻辑块编辑按钮为：增加按钮或删除按钮。

[0024] 较佳的，所述针对用户选择的每个逻辑块，显示该逻辑块对应的逻辑块配置界面的步骤，包括：

[0025] 在所述编程界面的第二预设区域中，显示逻辑块配置界面；所述逻辑块配置界面中包含该逻辑块所有可选的参数，和/或该逻辑块的参数输入框。

[0026] 较佳的，所述按照用户选择的配置参数，生成该逻辑块对应的程序逻辑，并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令的步骤，包括：

[0027] 当检测到所述逻辑块配置界面中的确认按钮被按下，则按照用户选择的配置参数，生成该逻辑块对应的XML语言源代码；

[0028] 将该逻辑块对应的XML语言源代码转换为机器人能够识别的脚本语言代码。

[0029] 较佳的，所述的编程界面集成于机器人示教器软件中。

[0030] 另一方面，本发明实施例还公开了一种控制机器人的装置，包括：

[0031] 第一显示单元，用于显示预先设置的逻辑块；其中，每个逻辑块对应一条控制逻辑语句；

[0032] 获取单元，用于获取用户选择的逻辑块；

[0033] 第二显示单元，用于按用户确定的顺序，以树形结构显示用户选择的逻辑块；

[0034] 第三显示单元，用于针对用户选择的每个逻辑块，显示该逻辑块对应的逻辑块配置界面；

[0035] 接收单元，用于接收用户通过所述逻辑块配置界面选择的该逻辑块对应的配置参数；

[0036] 转换单元，用于按照用户选择的配置参数，生成该逻辑块对应的程序逻辑，并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令；

[0037] 控制单元，用于将针对每个逻辑块转换的所述指令发送给机器人，对机器人进行控制。

[0038] 较佳的，所述装置还包括：

[0039] 划分单元，用于将所述逻辑块根据预设条件被划分为基础条件逻辑块和高级条件逻辑块，其中，所述预设条件包括所述逻辑块被选择的频率或者编辑所述逻辑块对应的控制逻辑语句的难易程度；

[0040] 所述第一显示单元，包括：

[0041] 第一显示子单元，用于按用户指令显示编程界面；

[0042] 第二显示子单元,用于当检测到所述编程界面中的基础条件逻辑块选项按钮被按下,则在所述编程界面中显示基础条件逻辑块;

[0043] 第三显示子单元,用于当检测到所述编程界面中的高级条件逻辑块选项按钮被按下,则在所述编程界面中显示高级条件逻辑块。

[0044] 较佳的,所述第二显示单元,包括:

[0045] 第四显示子单元,用于按用户选择逻辑块的顺序,在所述编程界面的第一预设区域中,以树形结构由上至下显示用户选择的逻辑块;

[0046] 第五显示子单元,用于当检测到所述编程界面中的任一个逻辑块编辑按钮被按下,则执行该逻辑块编辑按钮对应的编辑操作,在所述树形结构中增加逻辑块或删除逻辑块在树形结构中的位置;所述逻辑块编辑按钮为:增加按钮或删除按钮。

[0047] 较佳的,所述第三显示单元具体用于:

[0048] 在所述编程界面的第二预设区域中,显示逻辑块配置界面;所述逻辑块配置界面中包含该逻辑块所有可选的参数,和/或该逻辑块的参数输入框。

[0049] 较佳的,所述转换单元,包括:

[0050] 第一转换单元,用于当检测到所述逻辑块配置界面中的确认按钮被按下,则按照用户选择的配置参数,生成该逻辑块对应的XML语言源代码;

[0051] 第二转换单元,用于将该逻辑块对应的XML语言源代码转换为机器人能够识别的脚本语言代码。

[0052] 本发明实施例提供的控制机器人的方法,在获得用户从显示的逻辑块中选择的逻辑块后,将该逻辑块按照用户选择逻辑块的顺序,添加到树形结构,并在每个逻辑块的配置界面,选择每个逻辑块对应的配置参数,生成该逻辑块对应的程序逻辑,进一步将程序逻辑转换为机器人能够识别的指令,通过该指令就可以实现对机器人的控制。应用本发明实施例,用户只需要从显示的逻辑块中,选择出需要的逻辑块,再通过逻辑块配置界面选择各个逻辑块的配置参数,就能够实现对机器人的控制,因此,简化了控制机器人的方法,使得非专业人员在短时间内掌握控制机器人的方法。实施本发明的任一产品或方法必不一定需要同时达到以上所述的所有优点。

附图说明

[0053] 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简单地介绍,显而易见地,下面描述中的附图仅仅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的前提下,还可以根据这些附图获得其他的附图。

[0054] 图1为本发明实施例提供的控制机器人的方法的流程图;

[0055] 图2为本发明实施例提供的控制机器人的方法中,选择基础条件逻辑块中Loop逻辑块的界面图;

[0056] 图3为本发明实施例提供的控制机器人的方法中,用户选择的基础条件逻辑块生成的树形结构的界面图;

[0057] 图4为本发明实施例提供的控制机器人的方法中,选择高级条件逻辑块的界面图;

[0058] 图5为本发明实施例提供的控制机器人的方法中,基础条件逻辑块中的Loop逻辑

块的配置界面图；

[0059] 图6为本发明实施例提供的控制机器人的方法中，基础条件逻辑块中的Set逻辑块的配置界面图；

[0060] 图7为本发明实施例提供的控制机器人的方法中，基础条件逻辑块中的If逻辑块的配置界面图；

[0061] 图8为本发明实施例提供的控制机器人的方法中，基础条件逻辑块中的Move逻辑块的配置界面图；

[0062] 图9为本发明实施例提供的控制机器人的方法中，基础条件逻辑块中的Waypoint逻辑块的配置界面图；

[0063] 图10为本发明实施例提供的控制机器人的装置的结构示意图。

具体实施方式

[0064] 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，所描述的实施例仅仅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做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。

[0065] 本发明实施例提供了一种控制机器人的方法及装置，以简化控制机器人的操作，降低对用户的要求。

[0066] 需要说明的是，本发明实施例，从显示的逻辑块中，获得用户选择的逻辑块后，按照用户选择的逻辑块的顺序，依次将所选择的逻辑块，添加到树形结构，针对用户选择的每个逻辑块，显示该逻辑块对应的配置界面，在该配置界面选择该逻辑块对应的配置参数，得到该逻辑块对应的程序逻辑，进一步将程序逻辑转换为机器人能够识别的指令，这样，通过所述指令，就可以实现对机器人的控制。本发明实施例的控制机器人的方法及装置，简化了控制机器人的方法，使得非专业人员在短时间内掌握控制机器人的方法。

[0067] 参见图1，图1为本发明实施例提供的控制机器人的方法的流程图，包括以下步骤：

[0068] S101，显示预先设置的逻辑块；其中，每个逻辑块对应一条控制逻辑语句；

[0069] 具体的，每个逻辑块对应一条控制逻辑语言，在显示每个逻辑块时，用户能够看到显示的逻辑块的名称。例如，循环(Loop)逻辑块、中断(Break)逻辑块、继续(Continue)逻辑块、条件(If/Else if/Else)逻辑块、交换(Switch)逻辑块、实例(Case)逻辑块、默认(Default)逻辑块、行注释(Line Comment)逻辑块、段注释(Block)逻辑块、移动(Move)逻辑块、路点/运动姿态(Waypoint)逻辑块、设置(Set)逻辑块、等待(Wait)逻辑块、线程(Thread)逻辑块、过程(Procedure)逻辑块、脚本(Script)逻辑块、消息(Message)逻辑块、轨迹(Track)逻辑块等。其中，条件(If/Else if/Else)逻辑块代表三个逻辑块，包括If逻辑块、Else if逻辑块以及Else逻辑块。

[0070] S102，获取用户选择的逻辑块；

[0071] 在显示的预先设置的逻辑块中，当用户选择逻辑块中的一个逻辑块时，就得到该逻辑块及该逻辑块对应的一条控制逻辑语句。

[0072] S103，按用户确定的顺序，以树形结构显示用户选择的逻辑块；

[0073] 优选的，用户在选择的逻辑块时，根据选择逻辑块的顺序，依次将选择的逻辑块添

加到树形结构。这里。需要注意的是，在用户没有选择任何逻辑块之前，树形结构是空白的，可以理解为，树形结构不存在，当用户选择一个逻辑块时，将选择的逻辑块添加到空白的树形结构，当用户选择了多个逻辑块时，将多个逻辑块按照选择逻辑块的先后顺序依次将所述多个逻辑块，添加到树形结构，从而生成包括多个逻辑块的完整的树形结构。

[0074] S104，针对用户选择的每个逻辑块，显示该逻辑块对应的逻辑块配置界面；

[0075] 可以理解的是，在所述树形结构中，包含多个用户选择的逻辑块，每个逻辑块的对应各自的逻辑块配置界面，该逻辑块配置界面用于选择该逻辑块对应的配置参数。

[0076] S105，接收用户通过所述逻辑块配置界面选择的该逻辑块对应的配置参数；

[0077] 在每个逻辑块配置界面，选择了该逻辑块对应的配置参数后，接收所述配置参数，需要注意的是，每个逻辑块对应的配置参数很多是预先编辑好的，用户根据自己的实际需求，在配置参数中，选择合适配置参数即可；另外有些逻辑块的配置参数也只需要进行简单的参数输入即可。

[0078] S106，按照用户选择的配置参数，生成该逻辑块对应的程序逻辑，并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令；

[0079] 在所生成的树形结构中，为该树形结构中的逻辑块选择配置参数，并生成树形结构中的逻辑块的对应的程序逻辑，这样，树形结构对应的程序逻辑为该树形结构中每个逻辑块对应的程序逻辑组成的。树形结构中的每个逻辑块对应的程序逻辑转换为机器人能够识别的指令，也就是将树形结构对应的程序逻辑转换为机器人能够识别的指令。

[0080] S107，将针对每个逻辑块转换的所述指令发送给机器人，对机器人进行控制。

[0081] 在将树形结构中的逻辑块对应的程序逻辑转换为机器人能够识别的指令后，将所述指令发送给机器人，机器人就会按照该指令执行动作，以实现对机器人的控制。

[0082] 需要注意的是，树形结构是根据用户选择的逻辑块生成的，所以用户选择的逻辑块不同，生成的树形结构也不同，那么树形结构对应的程序逻辑也不同，则将程序逻辑转换为机器人能够识别的指令也就不同，从而控制机器人完成不同的动作。所以，用户可以根据自己的需求，选择不同的逻辑块，生成不同的树形结构，使得将程序逻辑转换为机器人能够识别的不同指令，控制机器人完成不同的动作。

[0083] 应用上述各个实施例，从显示的逻辑块中，获得用户所选择的逻辑块后，将用户所选择的逻辑块，按照用户选择逻辑块的顺序，添加到树形结构，在每个逻辑块的配置界面，选择每个逻辑块对应的配置参数，生成该逻辑块对应的程序逻辑，将所述程序逻辑转换为机器人能够识别的指令，实现对机器人的控制。这样，简化了控制机器人的方法，使得非专业人员在短时间内掌握控制机器人的方法。

[0084] 在本发明实施例的一种可能的实现方式中，所述逻辑块根据预设条件被划分为基础条件逻辑块和高级条件逻辑块，其中，所述预设条件包括所述逻辑块被选择的频率或者编辑所述逻辑块对应的控制逻辑语句的难易程度，将逻辑块根据预设条件被划分为基础条件逻辑块和高级条件逻辑块。

[0085] 具体的，基础条件逻辑块包括：循环(Loop)逻辑块、中断(Break)逻辑块、继续(Continue)逻辑块、条件(If/Else if/Else)逻辑块、交换(Switch)逻辑块、实例(Case)逻辑块、默认(Default)逻辑块、行注释(Line Comment)逻辑块、段注释(Block)逻辑块、移动(Move)逻辑块、路点/运动姿态(Waypoint)逻辑块、设置(Set)逻辑块、等待(Wait)逻辑块。

[0086] 所述高级条件逻辑块包括:线程(Thread)逻辑块、过程(Procedure)逻辑块、脚本(Script)逻辑块、消息(Message)逻辑块、轨迹(Track)逻辑块。其中,所述条件(If/Else if/Else)逻辑块可以和交换(Switch)逻辑块、实例(Case)逻辑块、默认(Default)逻辑块三者替换,即在树形结构中,可以用Switch逻辑块、Case逻辑块以及Default逻辑块这三各逻辑块替换If逻辑块、Else if逻辑块以及Else逻辑块这三个逻辑块。

[0087] 具体的,所述显示预先设置的逻辑块的步骤,包括:按用户指令显示编程界面;当检测到所述编程界面中的基础条件逻辑块选项按钮被按下,则在所述编程界面中显示基础条件逻辑块;当检测到所述编程界面中的高级条件逻辑块选项按钮被按下,则在所述编程界面中显示高级条件逻辑块。

[0088] 参见图2,图2为本发明实施例提供的控制机器人的方法中,选择基础条件逻辑块中的Loop逻辑块的界面图。图2示出了,当检测到用户选择了基础条件逻辑块选项按钮(Basic)被按下时,在编程界面中显示基础条件(Basic Condition)逻辑块后,用户选择Basic Condition逻辑块中的Loop逻辑块,在工程项目(Project Program)下显示了Loop逻辑块的名称。

[0089] 例如,当检测到用户选择了基础条件逻辑块选项按钮(Basic)被按下时,在编程界面中显示基础条件(Basic Condition)逻辑块。当检测到用户选择了高级条件逻辑块选项按钮(Advanced)被按下时,在编程界面中显示基础条件(Advanced Condition)逻辑块。

[0090] 图2还示出了,工程按钮(Project)、子工程按钮(procedure)、条件按钮(Condition)、开发工具包按钮(Wizard)、在前增加逻辑块按钮(Add before)、在后增加逻辑块按钮(Add after)、复制按钮(Copy)、粘贴按钮(Paste)、删除按钮(Remove)、变量按钮(Variable)、脚本按钮(Script)、仿真按钮(Simulator)轨迹按钮(Track)、开始按钮(Start)、停止按钮(Stop)以及下一步按钮(Step)。

[0091] 优选的,所述按用户确定的顺序,以树形结构显示用户选择的逻辑块的步骤,包括:按用户选择逻辑块的顺序,在所述编程界面的第一预设区域中,以树形结构由上至下显示用户选择的逻辑块;当检测到所述编程界面中的任一个逻辑块编辑按钮被按下,则执行该逻辑块编辑按钮对应的编辑操作,在所述树形结构中增加逻辑块或删除逻辑块在树形结构中的位置;所述逻辑块编辑按钮为:增加按钮或删除按钮。需要注意的是,这里,所述第一预设区域为图2中的Project Program所在的区域。

[0092] 参见图3,图3为本发明实施例提供的控制机器人的方法中,用户选择的基础条件逻辑块生成的树形结构的界面图。图3示出了,在用户选择的逻辑块包括:Loop逻辑块、If逻辑块、Move逻辑块、Waypoint逻辑块以及Else逻辑块,且用户选择的顺序逻辑块的顺序是Loop逻辑块、If逻辑块、Move逻辑块、Waypoint逻辑块、Waypoint逻辑块、Else逻辑块、Move逻辑块、Waypoint逻辑块以及Waypoint逻辑块;按照用户选择逻辑块的顺序,在所述编程界面的第一预设区域中,以树形结构由上至下显示用户选择的逻辑块。

[0093] 这里,所述逻辑块编辑按钮包括:增加按钮和删除按钮;所述增加按钮包括:在前增加逻辑块按钮(Add before)和在后增加逻辑块按钮(Add after)。所述删除按钮(Remove)用于删除树形结构中选择的逻辑块以及该逻辑块下的逻辑块。例如,在所述第一预设区域中,选择Loop逻辑块,当在前增加逻辑块按钮(Add before)被按下时,在Loop逻辑块前增加一个Loop逻辑块,当在后增加逻辑块按钮(Add after)被按下时,在Loop逻辑块后

增加一个Loop逻辑块。又如,在所述第一预设区域中,选择Loop逻辑块,当删除按钮(Remove)被按下时,树形结构中的Loop逻辑块以及Loop逻辑块以下的逻辑块都被删除了。需要注意的是,所述第一预设区域为图2中的Project Program所在的区域。

[0094] 可见,用户选择的逻辑块在编程界面中的第一预设区域,以选择逻辑块的顺序,由上至下显示用户选择的逻辑块,例如,当选择Loop逻辑块时,将Loop逻辑块加入所述第一预设区域,当用户选择If逻辑块时,将If逻辑块添加至第一预设区域中的Loop逻辑块的下方。

[0095] 参见图4,图4为本发明实施例提供的控制机器人的方法中,选择高级条件逻辑块的界面图。图4示出了,选择高级条件逻辑块包括:线程(Thread)逻辑块、过程(Procedure)逻辑块、脚本(Script)逻辑块、消息(Message)逻辑块、轨迹(Track)逻辑块等。

[0096] 在本发明实施例的一种可能的实现方式中,所述针对用户选择的每个逻辑块,显示该逻辑块对应的逻辑块配置界面的步骤,包括:

[0097] 在所述编程界面的第二预设区域中,显示逻辑块配置界面;所述逻辑块配置界面中包含该逻辑块所有可选的参数,和/或该逻辑块的参数输入框。

[0098] 例如,如图5所示,在编辑界面的第二预设区域中,这里,所述第二预设区域为Loop Condition所在的区域。图5示出了,在所述第二预设区域中,Loop逻辑块的配置界面中,包含了Loop逻辑块的可选的参数,所述可选的参数包括:始终循环(loop always)、给定次数循环(loop times)以及条件循环(loop condition)。在所述第二预设区域还包括:清空按钮(Clear)、删除按钮(Remove)以及确认按钮(Confirm)。所述清空按钮用于清空所述Loop逻辑块的参数输入框中的用户输入的内容。

[0099] 另外,所述按照用户选择的配置参数,生成该逻辑块对应的程序逻辑,并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令的步骤,包括:

[0100] 当检测到所述逻辑块配置界面中的确认按钮被按下,则按照用户选择的配置参数,生成该逻辑块对应的XML语言源代码;

[0101] 例如,当检测到所述逻辑块配置界面中的确认按钮(Confirm)被按下,按照用户选择的始终循环(loop always),生成Loop逻辑块对应的XML语言源代码,或者按照用户选择的给定次数循环(loop times),生成Loop逻辑块对应的XML语言源代码,或者条件循环(loop condition)的配置参数,生成Loop逻辑块对应的XML语言源代码。

[0102] 将该逻辑块对应的XML语言源代码转换为机器人能够识别的脚本语言代码。

[0103] 这里,所述逻辑块对应的XML语言源代码为所述逻辑块对应的程序逻辑,所述机器人能够识别的脚本语言代码为所述机器人能够识别的指令。根据所述指令,就可以控制机器人。例如,在所述XML语言源代码转换为所述脚本语言代码之后,当检测到图5中的开始按钮(Start)被按下时,机器人就开始动作,当检测到停止按钮(Stop)被按下时,机器人就停止正在执行的动作,当检测到下一步按钮(Step)被按下时,机器人将执行当前动作的下一步动作。

[0104] 需要注意的是,上述各个实施例中的编程界面集成于机器人示教器软件中。

[0105] 参见图6,图6为本发明实施例提供的控制机器人的方法中,基础条件逻辑块中的Set逻辑块的配置界面图。

[0106] 例如,在编辑界面的第二预设区域中,这里,所述第二预设区域为Set Condition所在的区域。在所述第二预设区域中,Set逻辑块的配置界面中,包含了Set逻辑块的可选的

参数以及Set逻辑块的参数输入框。在所述第二预设区域还包括：清空按钮(Clear)、删除按钮(Remove)以及确认按钮(Confirm)。所述清空按钮用于清空所述逻辑块的参数输入框中的用户输入的内容。

[0107] 图7为本发明实施例提供的控制机器人的方法中，基础条件逻辑块中的If逻辑块的配置界面图。具体的，这里，所述第二预设区域为If Condition所在的区域。在所述第二预设区域中，If逻辑块的配置界面中，包含了If逻辑块的参数输入框。在所述第二预设区域还包括：清空按钮(Clear)、增加Else If逻辑块的按钮、增加Else逻辑块的按钮、删除按钮(Remove)以及确认按钮(Confirm)。所述清空按钮用于清空所述逻辑块的参数输入框中的用户输入的内容，所述增加Else If逻辑块的按钮用于在树形结构中增加Else If逻辑块，所述增加Else逻辑块的按钮用于在树形结构中增加Else逻辑块的按钮。所述删除按钮用于删除树形结构中的该If逻辑块以及该If逻辑块下的逻辑块。例如，图7中，当检测到删除按钮(Remove)被按下时，删除第一预设区域(Project Program所在的区域)中的If逻辑块以及该If逻辑块下的Move逻辑块及两个Waypoint逻辑块。

[0108] 图8为本发明实施例提供的控制机器人的方法中，基础条件逻辑块中的Move逻辑块的配置界面图。具体的，在编辑界面的第二预设区域中，这里，所述第二预设区域为Move Condition所在的区域。在所述第二预设区域中，Move逻辑块的配置界面中，包含了Move逻辑块的可选的参数。在所述第二预设区域还包括：翻转按钮(Reversal)、删除按钮(Remove)以及确认按钮(Confirm)。所述清空按钮用于清空所述逻辑块的参数输入框中的用户输入的内容。所述确认按钮用于将Move逻辑块对应的程序逻辑转换为机器人能够识别的指令。

[0109] 图9为本发明实施例提供的控制机器人的方法中，基础条件逻辑块中的Waypoint逻辑块的配置界面图。具体的，在编辑界面的第二预设区域中，这里，所述第二预设区域为Waypoint Condition所在的区域。在所述第二预设区域中，包括：在前增加逻辑块按钮(Add before)和在后增加逻辑块按钮(Add after)、拖拽按钮(Set Waypoint)以及移动按钮(Move here)。所述Add before按钮用于在树形结构中该Waypoint的逻辑块之前加一个Waypoint的逻辑块，所述Add before按钮用于在树形结构中该Waypoint的逻辑块后加一个Waypoint的逻辑块，所述Set Waypoint用于拖拽仿真图中的机器人，所述Move here按钮用于将仿真图中的机器人移动到目标位置。这里，仿真图是图9中所显示的机器人仿真图。

[0110] 下来介绍本发明实施例中的树形结构的程序逻辑(XML语言源代码)与脚本语言代码的对应关系。

[0111] 以图3中所示的树形结构为例。其中，

[0112] Loop逻辑块对应的XML语言源代码为：

```
[0113] <model logic="Loop" row_number="0" logic_mark="3" mode_mark="1">
```

```
[0114] <attribute mode="Always"/>
```

```
[0115] </model>
```

[0116] 其中，row_number="0"，对应的基础条件逻辑块显示界面中的Loop逻辑块的位置，logic_mark="3"表示Loop逻辑块用3来表示，mode_mark="1"对应配置界面中的always选项。

[0117] Loop逻辑块对应的脚本语言Script代码为：

```
[0118] setRunLine(0)
```

[0119] while (true) do

[0120] end

[0121] Set逻辑块对应的XML语言源代码为:

[0122]

```
<model logic="Set" row_number="1" logic_mark="12" mode_mark="6">
```

```
    <attribute mode="Variable">
```

```
        <Variable_Uuid>0bda7cec17f14c4e9f814b50b22a9c0d</Variable_Uuid>
```

```
        <Variable_Name>var_bool_1</Variable_Name>
```

```
        <Expression>not
```

```
            var_bool_1|0bda7cec17f14c4e9f814b50b22a9c0d</Expression>
```

```
        </attribute>
```

```
    </model>
```

[0123] Set逻辑块对应的脚本语言Script代码为:

[0124] setRunLine (1)

[0125] var_bool_1=not var_bool_1

[0126] 其中, row_number="1", 对应的基础条件逻辑块显示界面中的Set逻辑块的位置, logic_mark="12" 表示Set逻辑块用12来表示, mode_mark="6" 对应配置界面中的Set Variable选项。

[0127] If逻辑块对应的XML语言源代码为:

[0128]

```
<model logic="If" row_number="2" logic_mark="4">
```

```
    <attribute mode="Expression">
```

```
        <Expression>var_bool_1|0bda7cec17f14c4e9f814b50b22a9c0d</Expression>
```

```
    </attribute>
```

```
    </model>
```

[0129] If逻辑块对应的脚本语言源代码为:

[0130] setRunLine (2)

[0131] if (var_bool_1) then

[0132] end

[0133] 其中, row_number="2", 对应的基础条件逻辑块显示界面中的If逻辑块的位置, logic_mark="4" 表示If逻辑块用4来表示, mode_mark="4" 对应配置界面中的Input condition, 0bda7cec17f14c4e9f814b50b22a9c0d 对应配置界面中的var_bool_1。

[0134] Move和Waypoint逻辑块对应的XML语言源代码为:

[0135]

```
<model logic="Move" row_number="3" logic_mark="10" mode_mark="1">
    <attribute mode="MoveJ">
        <Speed>3000</Speed>
        <Acc>6000</Acc>
        <S-Curve>0</S-Curve>
        <TrackType>0</TrackType>
        <Feature>base</Feature>
        <Tcp>tcp</Tcp>
        <EnableTcp>0</EnableTcp>

    <EnableRelativeOffset>0</EnableRelativeOffset>
        <RelativeOffsetX></RelativeOffsetX>
        <RelativeOffsetY></RelativeOffsetY>
        <RelativeOffsetZ></RelativeOffsetZ>
```

[0136]

```

<BlendRadius>0</BlendRadius>
</attribute>
<model      logic="Waypoint"      row_number="4"
logic_mark="11">
<attribute mode="Waypoint">
<Name>Waypoint 01</Name>
<Mode>0</Mode>

<FixedpPoint>0.191841,-0.098170,0.985000,0.603230,0.603230,0.368936,0.36893
6,1.097812,0.000000,0.000000,0.000000,0.000000,0.000000</FixedpPoint>
</attribute>
</model>
<model      logic="Waypoint"      row_number="5"
logic_mark="11">
<attribute mode="Waypoint">
<Name>Waypoint 02</Name>
<Mode>0</Mode>

<FixedpPoint>0.202310,0.074235,0.985000,0.404821,0.404821,0.579758,0.57975
8,1.922480,-0.000000,0.000000,0.000000,0.000000,0.000000</FixedpPoint>
</attribute>
</model>
</model>

```

[0137] Move和Waypoint逻辑块对应的脚本语言源代码为：

[0138]

```

setRunLine(3)
    set_feature("world")
    set_scurve(false)
    setRunLine(4)
    movej({62.899994,  0.000000,  0.000000,  0.000000,  0.000000,
0.000000}, 6000, 3000)
    setRunLine(5)

```

[0139]

```
movej({110.149990, 0.000000, 0.000000, 0.000000, 0.000000,  
0.000000}, 6000, 3000)
```

[0140] 其中,在"Move"row_number="3"logic_mark="10"mode_mark="1"中, row_number="3"表示Move逻辑块在对应的基础条件逻辑块显示界面中的位置, logic_mark="10"表示Move逻辑块用10来表示, mode_mark="4"对应配置界面中的Move Joint。

[0141] 在<model logic="Waypoint"row_number="4"logic_mark="11">中, row_number="4"表示Waypoint逻辑块在对应的基础条件逻辑块显示界面中的位置, logic_mark="11"表示Waypoint逻辑块用11来表示。

[0142] 在<model logic="Waypoint"row_number="5"logic_mark="11">中, row_number="5"表示Waypoint逻辑块在对应的基础条件逻辑块显示界面中的位置, logic_mark="11"表示Waypoint逻辑块用11来表示。

[0143] Else逻辑块对应的XML语言源代码为:

[0144] <model logic="Else"row_number="6"logic_mark="6">

[0145] </model>

[0146] Else逻辑块对应的脚本语言源代码为:

[0147] setRunLine (6)

[0148] else

[0149] 其中, Else"row_number="6"表示Else逻辑块在对应的基础条件逻辑块显示界面中的位置, logic_mark="6"表示Else逻辑块用6来表示。

[0150] 本发明实施例还提供了一种控制机器人的装置。

[0151] 参见图10,图10为本发明实施例提供的控制机器人的装置的结构示意图,包括:

[0152] 第一显示单元1010,用于显示预先设置的逻辑块;其中,每个逻辑块对应一条控制逻辑语句;

[0153] 获取单元1020,用于获取用户选择的逻辑块;

[0154] 第二显示单元1030,用于按用户确定的顺序,以树形结构显示用户选择的逻辑块;

[0155] 第三显示单元1040,用于针对用户选择的每个逻辑块,显示该逻辑块对应的逻辑块配置界面;

[0156] 接收单元1050,用于接收用户通过所述逻辑块配置界面选择的该逻辑块对应的配置参数;

[0157] 转换单元1060,用于按照用户选择的配置参数,生成该逻辑块对应的程序逻辑,并将该逻辑块对应的程序逻辑转换为机器人能够识别的指令;

[0158] 控制单元1070,用于将针对每个逻辑块转换的所述指令发送给机器人,对机器人进行控制。

[0159] 进一步,所述装置还包括:

[0160] 划分单元,用于将所述逻辑块根据预设条件被划分为基础条件逻辑块和高级条件逻辑块,其中,所述预设条件包括所述逻辑块被选择的频率或者编辑所述逻辑块对应的控制逻辑语句的难易程度;

[0161] 所述第一显示单元,包括:

- [0162] 第一显示子单元,用于按用户指令显示编程界面;
- [0163] 第二显示子单元,用于当检测到所述编程界面中的基础条件逻辑块选项按钮被按下,则在所述编程界面中显示基础条件逻辑块;
- [0164] 第三显示子单元,用于当检测到所述编程界面中的高级条件逻辑块选项按钮被按下,则在所述编程界面中显示高级条件逻辑块。
- [0165] 进一步,所述第二显示单元,包括:
- [0166] 第四显示子单元,用于按用户选择逻辑块的顺序,在所述编程界面的第一预设区域中,以树形结构由上至下显示用户选择的逻辑块;
- [0167] 第五显示子单元,用于当检测到所述编程界面中的任一个逻辑块编辑按钮被按下,则执行该逻辑块编辑按钮对应的编辑操作,在所述树形结构中增加逻辑块或删除逻辑块或移动逻辑块在树形结构中的位置;所述逻辑块编辑按钮为:增加按钮或删除按钮或移动按钮。
- [0168] 进一步,所述第三显示单元具体用于:
- [0169] 在所述编程界面的第二预设区域中,显示逻辑块配置界面;所述逻辑块配置界面中包含该逻辑块所有可选的参数,和/或该逻辑块的参数输入框。
- [0170] 更进一步,所述转换单元,包括:
- [0171] 第一转换单元,用于当检测到所述逻辑块配置界面中的确认按钮被按下,则按照用户选择的配置参数,生成该逻辑块对应的XML语言源代码;
- [0172] 第二转换单元,用于将该逻辑块对应的XML语言源代码转换为机器人能够识别的脚本语言代码。
- [0173] 应用上述各个实施例,从显示的逻辑块中,获得用户所选择的逻辑块,根据用户选择的逻辑块生成树形结构,在每个逻辑块对应的配置界面,选择每个逻辑块的配置参数,生成树形结构中的逻辑块对应的程序逻辑,将树形结构对应的程序逻辑转换为机器人能够识别的指令,实现对机器人的控制。这样,简化了控制机器人的方法,使得非专业人员在短时间内掌握控制机器人的方法。
- [0174] 对于装置实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。
- [0175] 需要说明的是,在本文中,诸如第一和第二等之类的关系术语仅仅用来将一个实体或者操作与另一个实体或操作区分开来,而不一定要求或者暗示这些实体或操作之间存在任何这种实际的关系或者顺序。而且,术语“包括”、“包含”或者其任何其他变体意在涵盖非排他性的包含,从而使得包括一系列要素的过程、方法、物品或者设备不仅包括那些要素,而且还包括没有明确列出的其他要素,或者是还包括为这种过程、方法、物品或者设备所固有的要素。在没有更多限制的情况下,由语句“包括一个……”限定的要素,并不排除在包括所述要素的过程、方法、物品或者设备中还存在另外的相同要素。
- [0176] 本说明书中的各个实施例均采用相关的方式描述,各个实施例之间相同相似的部分互相参见即可,每个实施例重点说明的都是与其他实施例的不同之处。尤其,对于系统实施例而言,由于其基本相似于方法实施例,所以描述的比较简单,相关之处参见方法实施例的部分说明即可。
- [0177] 以上所述仅为本发明的较佳实施例而已,并非用于限定本发明的保护范围。凡在

本发明的精神和原则之内所作的任何修改、等同替换、改进等，均包含在本发明的保护范围内。

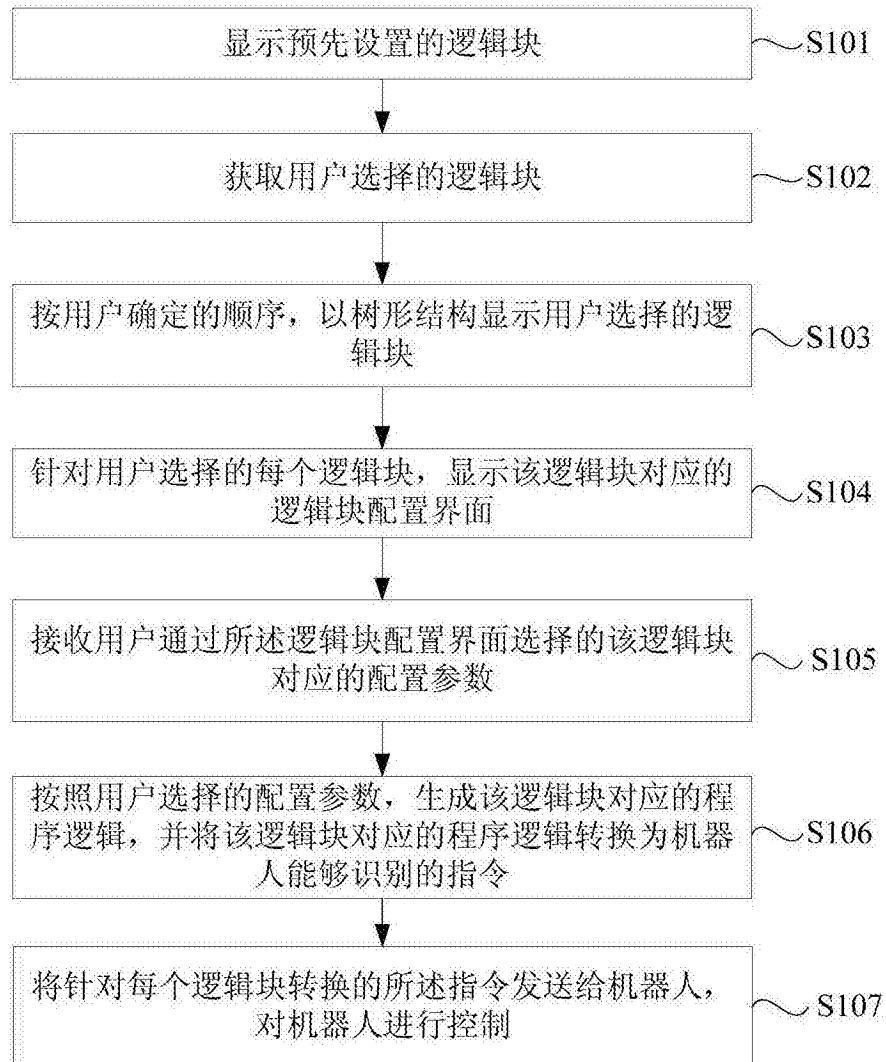


图1

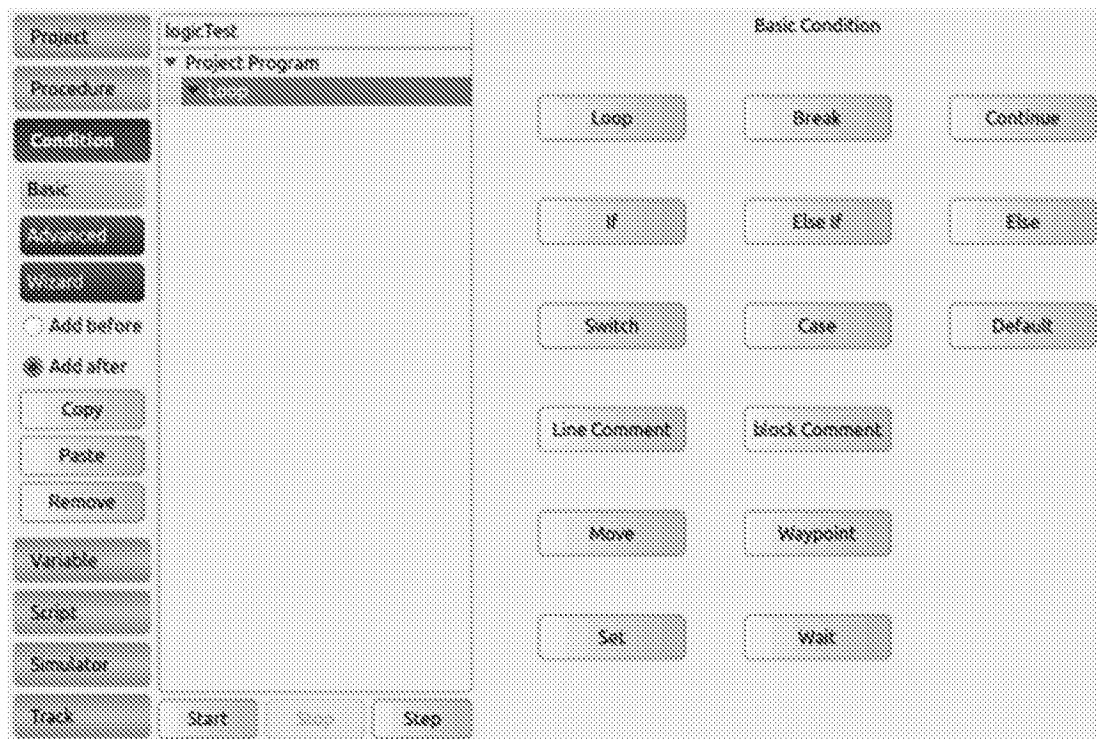


图2

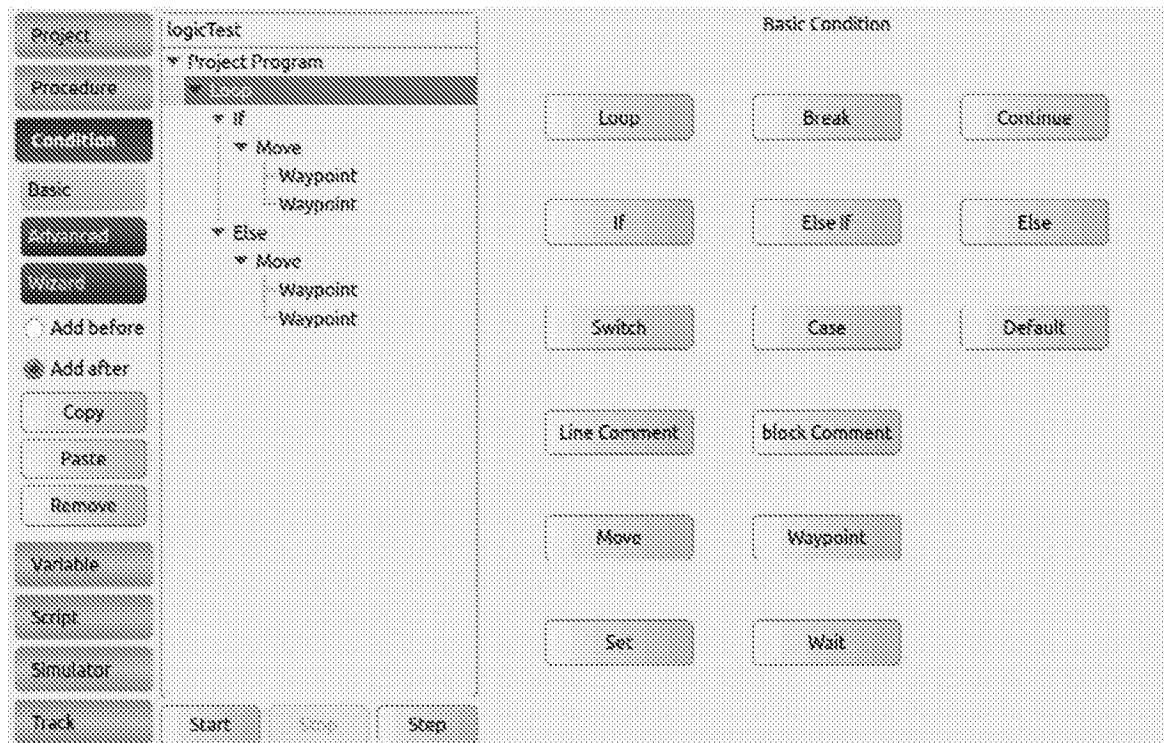


图3

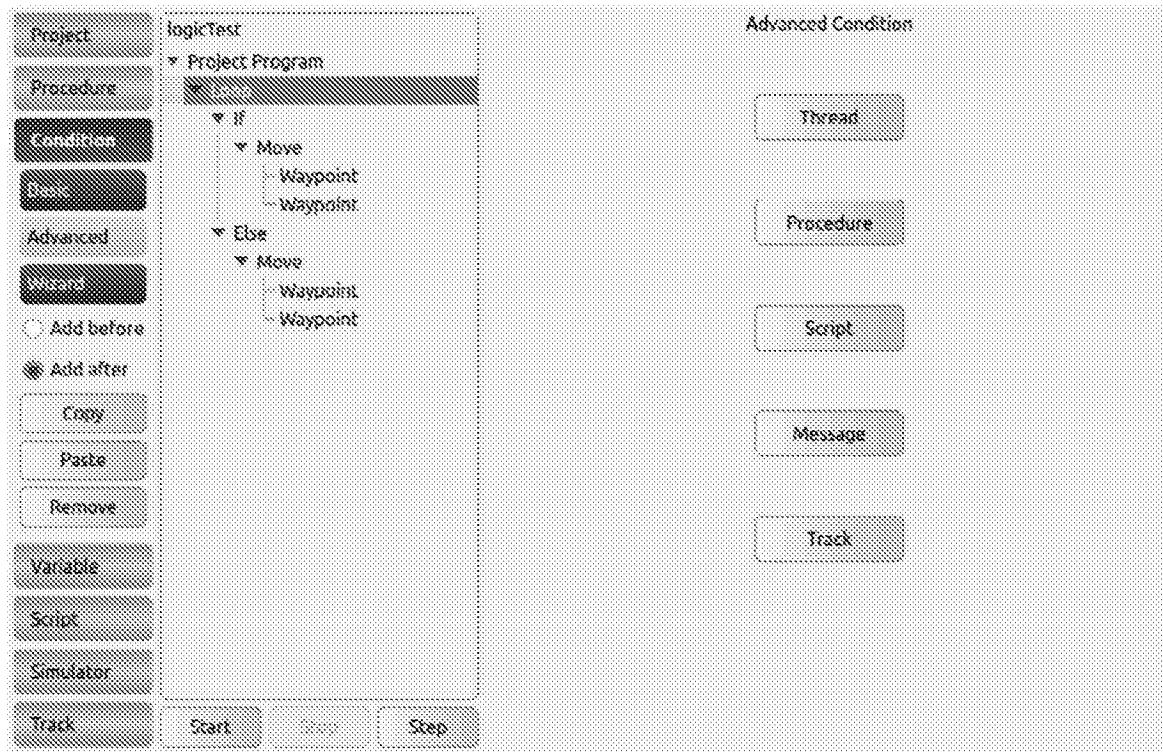


图4

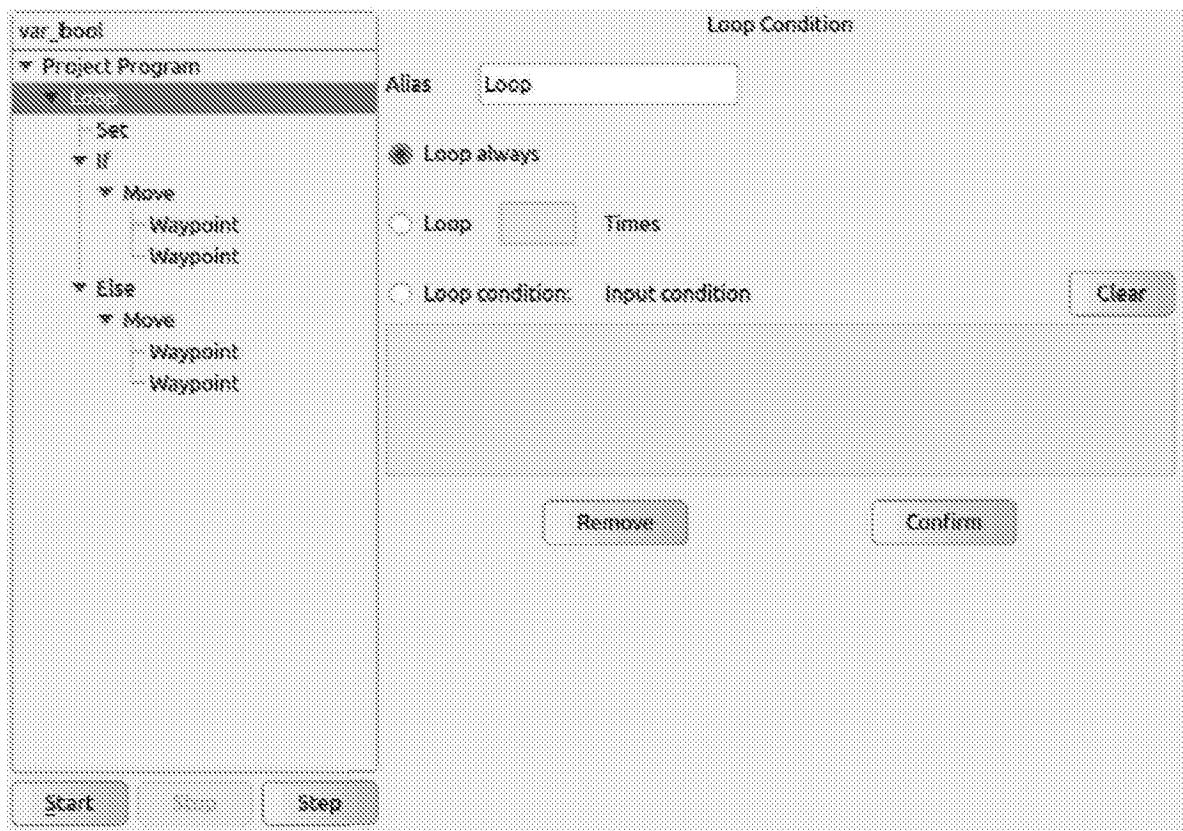


图5

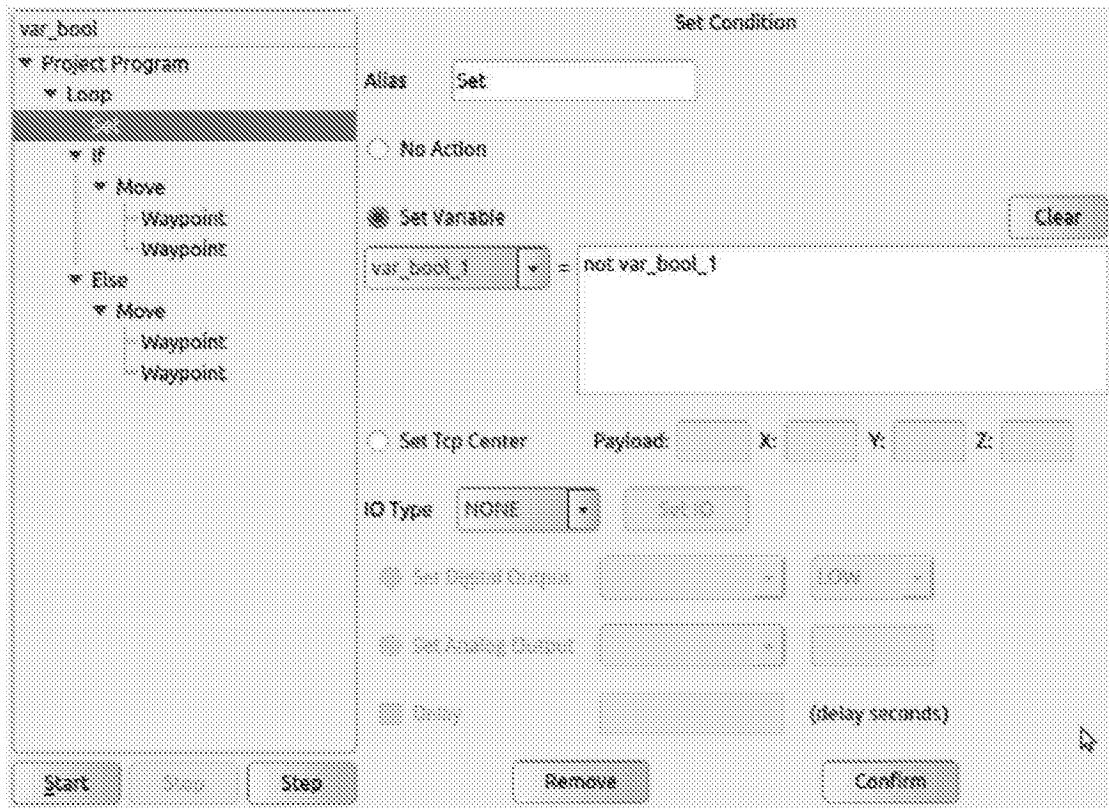


图6

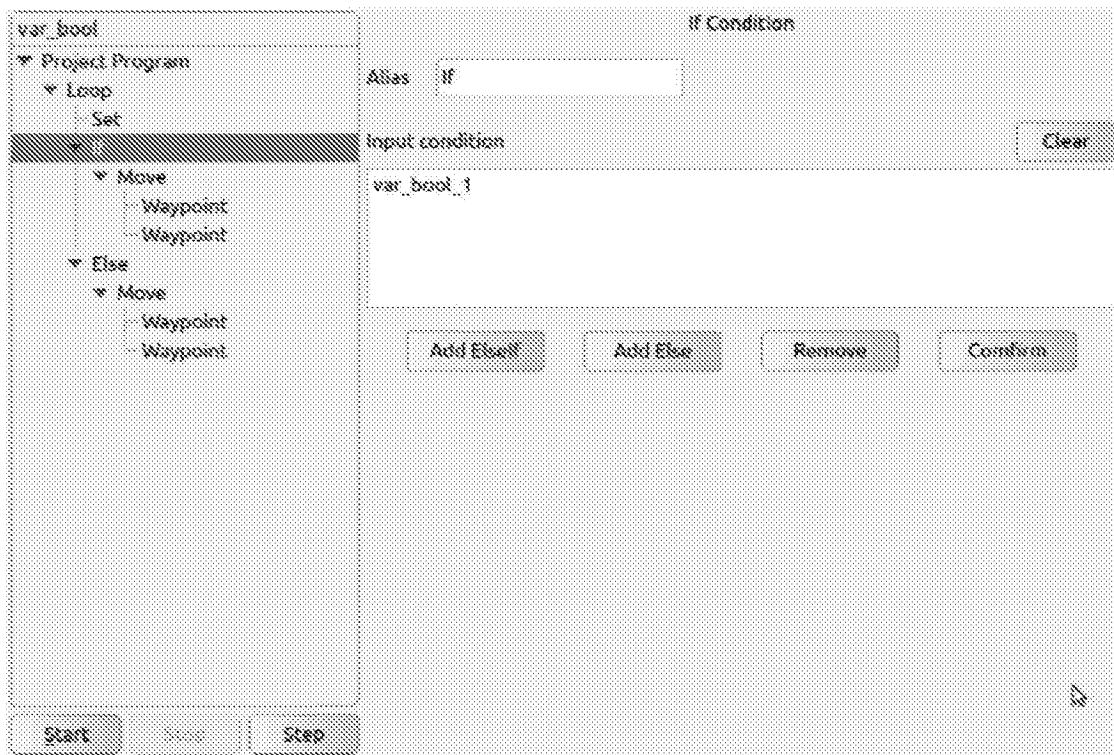


图7

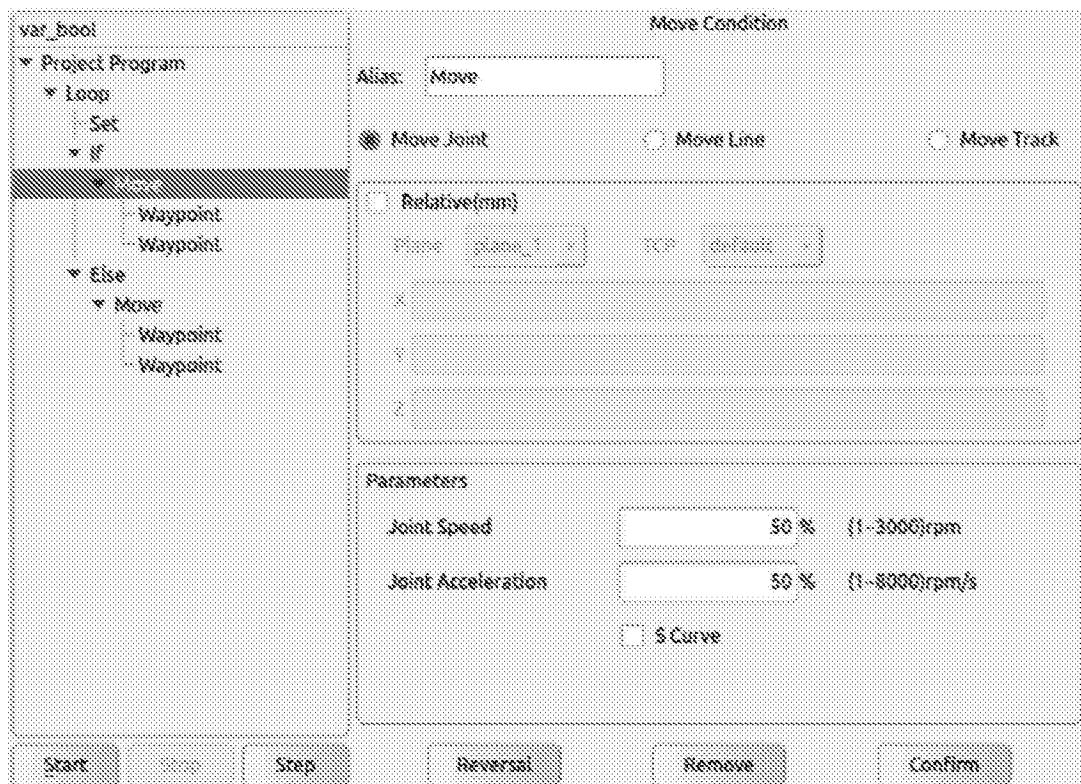


图8

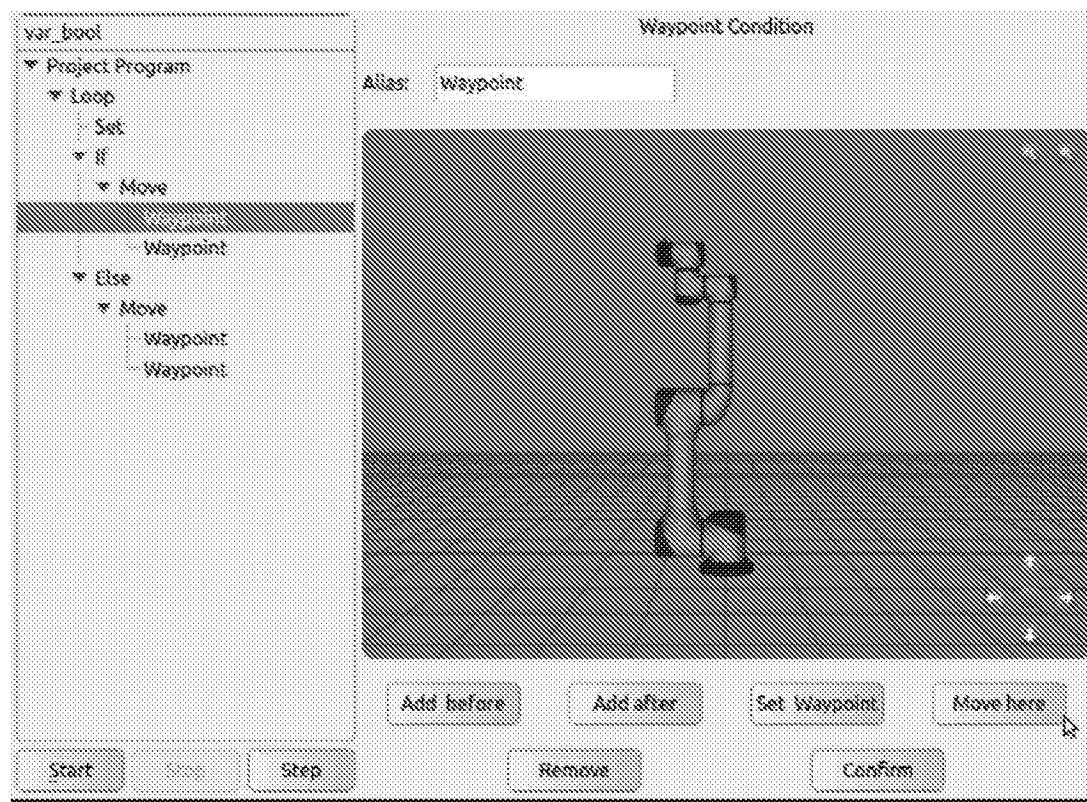


图9



图10