

计算机科学与技术学院 自助智能系统和机器人 课程实验报告

实验题目：机械臂的 SDK 概况		学号：201600301304
日期：2019.6.2	班级：2016 级人工智能	姓名：贾乘兴
Email: chengxingjia_0229@foxmail.com		
实验目的： 了解机械臂的 SDK 概况		
实验软件和硬件环境： DobotMagician 机械臂 Windows10		
实验原理和方法： <h3>3.1 SDK 概况</h3> <p>SDK（软件开发工具包）是一些软件工程师为特定的软件包、软件框架、硬件平台、操作系统等建立应用软件时的开发工具的集合。越疆魔术师机械臂不仅为用户提供了 Dobot Studio 软件进行控制，还为 Dobot Magician 机械臂的二次开发者提供了 Dobot 常用的 API 和搭建开发环境，用户在官网上可以获取 API 接口文档和 DobotDemo，对于通用桌面系统，越疆官方已经向 Dobot 二次开发者提供了动态链接库。二次开发者只要直接调用动态链接库即可控制 Dobot，而不需进行通讯协议相关的开发工作。</p>		

3.2 API 函数介绍

API（应用程序编程接口）是一些预先定义的函数，目的是提供应用程序与开发人员基于某软件或硬件得以访问一组例程的能力，而又无需访问源码，或理解内部工作机制的细节。API 函数的存在使得程序员可以直接调用某一些函数或者方法，其中部分的 API 函数在 Dobot Studio 的脚本控制界面左侧菜单栏已列出，如图 3.1 所示。

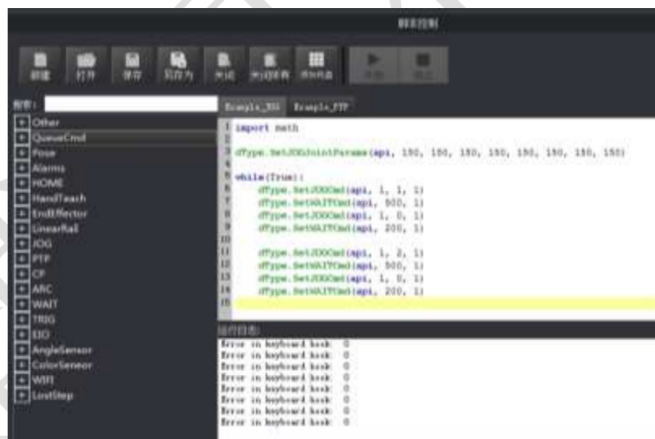


图 3.1 Dobot Studio 脚本控制界面

下面分类介绍 API 函数及其参数含义。

3.2.1 指令队列控制

(1) `GetQueuedCmdCurrentIndex (api)`: 控制器队列计数索引。

其中 `api` 是 Doobot 库的对象，不可修改。

25

在 Dobot 控制器指令队列机制中，有一个 64 位内部计数索引。当控制器每执行完一条命令时，该计数器将自动加一。通过该内部索引，可以查询控制器已经执行了多少队列指令，以及当前已经执行到哪条指令（指示运行进度时）。

(2) `SetQueuedCmdForceStopExec(api)`: 强制停止指令队列运行。

无论指令队列是否正在运行一条指令，控制器都会强制其停止运行。

(3) `SetQueuedCmdStartExec (api)`: 启动指令队列运行。

(4) `SetQueuedCmdStopExec(api)`: 停止指令队列运行。

若当前指令队列正在运行一条指令，则其将会在这条指令运行完成后，停止指令队列运行。

(5) `SetQueuedCmdStartDownload()`: 启动指令队列下载。

(6) `SetQueuedCmdStopDownload(void)`: 完成指令队列下载。

(7) `SetQueuedCmdClear(void)`: 清空指令队列。

3.2.2 实时位姿

(1) GetPose(api): 获取实时位姿。

(2) ResetPose(): 重设姿态。

当角度传感器损坏, 必须借助外部的角度测量手段或者角度传感器精度太差, 借助外部的角度直接/间接测量手段确认其精确值, 需要重设姿态。

(3) GetKinematics(api): 获取运动学参数。

3.2.3 ALARM 功能

(1) GetAlarmsState(api, maxLen=1000): 获取系统报警状态。

其中, maxLen: Incoming external buffer length to avoid overflow(输入外部缓冲区长度以避免溢出)。

(2) ClearAllAlarmsState(api): 清除系统所有报警。

3.2.4 HOME 功能

(1) SetHOMEParams(api, x, y, z, r, isQueued=0): 设置回零参数。

其中, x: X 轴坐标; y: Y 轴坐标; z: Z 轴坐标; r: R 末端坐标; isQueued: 表示队列模式开关状态。1 是使用队列模式, 0 反之。

(2) GetHOMEParams(api): 获取回零参数。

(3) SetHOMECmd(api): 执行回零功能。

实验步骤: (不要求罗列完整源代码)

3.3 使用 API 函数控制机械臂示例

实验目的:

本实验以机械臂在笛卡尔空间中按照给定的坐标实现点对点（Point to Point, PTP）点动。

实验原理:

使用 Dobot Studio 官方软件中自带的 API 函数来控制机械臂。这里涉及到的 API 函数如下:

- (1) dType.SetPTPCCommonParams(api, velocityRatio, accelerationRatio, isQueued=0): 设置点位公共参数。
- (2) dType.GetPose(api): 获取机械臂实时位姿。
- (3) dType.SetPTPCmd(api, ptpMode, x, y, z, rHead, isQueued=0): 执行 PTP 运动指令。

实验步骤:

- (1) 首先完成机械臂的软、硬件连接, 具体步骤参考 2.3 节。
- (2) 在 Dobot Studio 软件界面中选择“脚本控制”, 打开如图 3.2 所示界面。



图 3.2 Dobot Studio 脚本控制界面

- (3) 进入脚本控制模块后点击“新建”, 新建一个脚本工程, 可取名为“TEST1”, 并点击“确认”, 如图 3.3 所示。



图 3.3 新建脚本工程

- (4) 脚本框左边有各类 API 函数, 单击 API 函数上的“?”字样, 会出现函数的使用说明, 双击你的

API 函数便可显示左侧脚本程序输入框中，如图 3.5 所示。



图 3.4 Dobot Studio 脚本 API 函数说明

(5) 利用 python 编程语言，编写自己的脚本程序，代码具体示例如下。

```
import math

#设置点位公共参数

dType.SetPTPCommonParams(api, 100, 100)

#设置变量

moveX=0;moveY=0;moveZ=10;moveFlag=-1

#获取机械臂实时位姿

pos = dType.GetPose(api)

x = pos[0]

y = pos[1]

z = pos[2]

rHead = pos[3]

#点动路线与设定循环次数

while(True):

    moveFlag *= -1

    for i in range(5):

        dType.SetPTPCmd(api, dType.PTPMode.PTPMOVLXYZMode, x+moveX, y+moveY, z+moveZ, rHead, 1)

        moveX += 10 * moveFlag

        dType.SetPTPCmd(api, dType.PTPMode.PTPMOVLXYZMode, x+moveX, y+moveY, z+moveZ, rHead, 1)

        dType.SetPTPCmd(api, dType.PTPMode.PTPMOVLXYZMode, x+moveX, y+moveY, z, rHead, 1)
```

(6) 程序写好后，单击“开始”按钮，将程序编译，下载到机械臂中。机械臂即按照程序预定的轨迹进行运动。

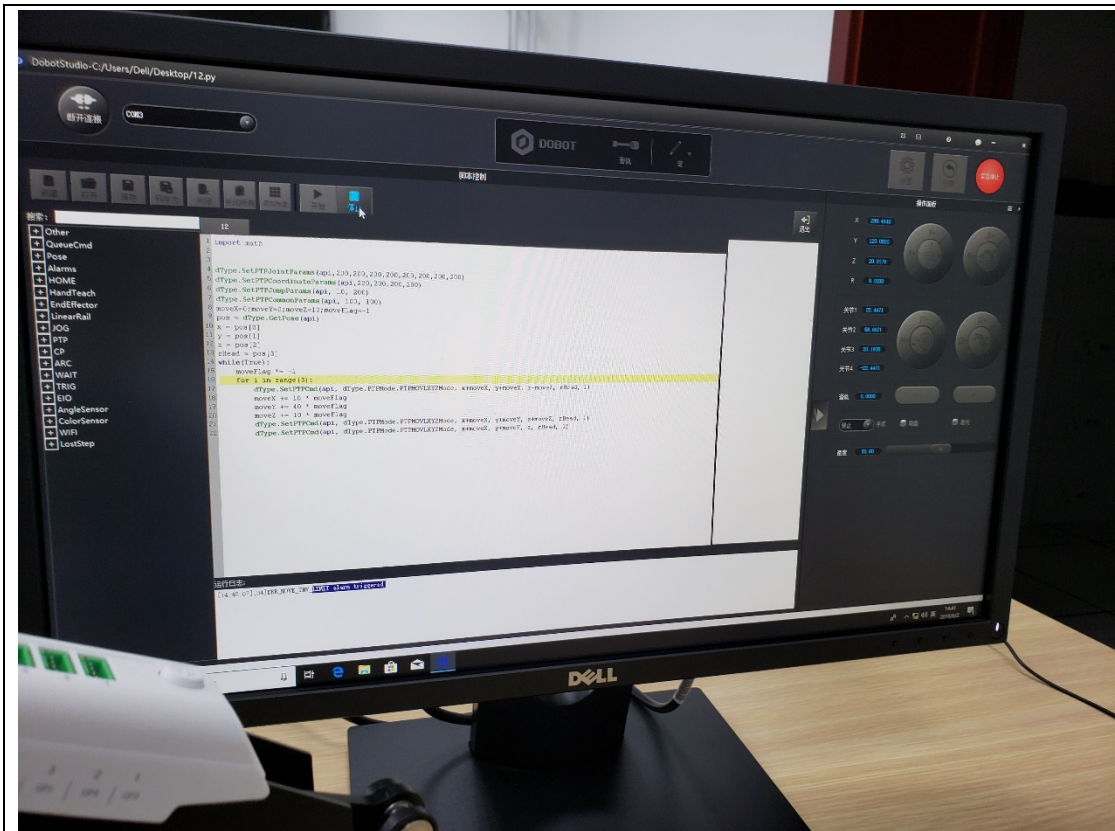
实验结果:

你会观察到机械臂在从预定义的初始位置处沿 X 轴和 Z 轴方向循环运动。





代码如下：



结论分析与体会：

接触了脚本编写指令，对控制机器人的方式有了更多的了解
使用脚本编写就更加的灵活自由了