

# 山东大学计算机科学与技术学院

## 机器人学导论 课程实验报告

学号： 201700301042	姓名：陈佳睿	班级：17 人工智能
实验题目：机械臂的信息检测和报警		
实验目的：熟悉并掌握 dobot Magician 机械臂规划运动、限位三种报警类型的信息检测		
硬件环境： 越疆机械臂		
软件环境： Dobot studio		
<p>实验步骤与内容：</p> <p>首先，了解了机械臂的报警类型 分别是公共报警、规划报警、运动报警、超速报警、限位报警，大致知道了每一个索引对应的错误是什么意思。 接下来看了信息检测与报警的函数 1)dType.GetAlarmsState(api, maxlen=1000) 功能:获取系统报警状态 参数:api:缺省 maxlen:为避免溢出 传入的外部缓冲区的长度 返回:list[0]:用于接收各报警位 list[1]:警报所占字节 (2)dType.ClearAllAlarmsState(api) 功能:清除系统所有报警 参数:api:缺省 返回:无</p> <p>实验中，我们调用各种类型的报警索引，发出模拟的故障指令 来使得机械臂产生规划参数错误、逆解算限位、运动至极限等故障，通过获取系统报警状态的 api 函数来确定具体报警类型 首先在代码中，我使用 ALAsetdefalut 函数来给每一个报警类型指定一个对应的数字 方便标识错误。 使用 a = [100, 100, 100, 100]为圆弧上任意一点坐标 将 b = a, 此时圆弧半径规划为无穷，导致规划圆弧输入参数报警</p>		

```

31 #规划报警的故障模拟 圆弧插补参数错误
32 a=[100,100,100,100]
33 b=a
34 dType.SetARCCmd(api, a, b, 1)
35
36 #运动报警的故障模拟 规划逆解算限位
37 #pos=dType.GetPose(api)
38 #dType.SetPTPCoordinateParams(api, 20, 20, 20, 20, 1)
39 #dType.SetPTPCmd(api, 3, pos[0]+1000, pos[1], pos[2], pos[3], 1)
40
41 #限位报警的故障模拟 关节1负向限位
42 #dType.SetJOGJointParams(api, 10, 10, 10, 10, 10, 10, 10, 10, 1)
43 #dType.SetJOGCmd(api, 1, 2, 1)
44
45 #获取系统报警状态
46 ala=dType.GetAlarmsState(api, maxlen=1000)
47
48 #计算报警索引并打印 print(' 报警所占总字节数:',ala[1] , ' 检测结果如下:')
49 for i in range(1000):
50     if ala[0][i]!=0:
51         err=1
52         index=i*8+int(log(ala[0][i],2))
53         print(' 检测到报警 索引为:0x%x'%index,ALA[index])
54         break
55     else:
56         pass
57 if(i==999):
58     print(' 没有检测到报警\n')
59

```

#### 运行日志:

```

[14:39:34] 检测到[14:39:55] 检测到报警 索引为:0x14 规划圆弧输入参数报警
报警 索引为:0x14 规划圆弧输入参数报警
[14:39:35][20]ERR_PLAN_ARC_INPUT_PARAM alarm triggered

```

#### 第二个实验

我们需要将前一个实验对于圆弧差补错误的相关代码注释掉

接下来，使用 Getpose 函数获取位姿

使用 SetPTPCoordinateParams() 函数进行坐标轴点设置

使用 SetPTPCmd () 函数让机械臂执行 PTP 运动指令

因为 Pos[0]+1000 超出关节限位值，机械臂不执行任何动作，发出规划逆解算限位报警

```

32 #运动报警的故障模拟 规划逆解算限位
33 pos=dType.GetPose(api)
34 dType.SetPTPCoordinateParams(api, 20, 20, 20, 20, 1)
35 dType.SetPTPCmd(api, 3, pos[0]+1000, pos[1], pos[2], pos[3], 1)
36
37 #限位报警的故障模拟 关节1负向限位
38 #dType.SetJOGJointParams(api, 10, 10, 10, 10, 10, 10, 10, 10, 1)
39 #dType.SetJOGCmd(api, 1, 2, 1)
40
41 #获取系统报警状态
42 ala=dType.GetAlarmsState(api, maxLen=1000)
43
44 #计算报警索引并打印
45 print(' 报警所占总字节数:',ala[1] , ' 检测结果如下:')
46 for i in range(1000):
47     if ala[0][i]!=0:
48         err=1
49         index=i*8+int(log(ala[0][i],2))
50         print(' 检测到报警 索引为:0x%x'%index,ALA[index])
51         break
52     else:
53         pass
54     if(i==999):
55         print(' 没有检测到报警\n')
56

```

运行日志:

```

[14:59:38] 报警所占总字节数: 16 检测结果如下:
[14:59:38] 检测到报警 索引为:0x12 规划逆解算限位
[14:59:38][18]ERR_PLAN_INV_LIMIT alarm triggered

```

### 第三个实验

使用 SetJOGJointParams() 设定关节点位参数

使用 SetJOGCmd() 执行点动指令

```

31
32 #运动报警的故障模拟 规划逆解算限位
33 #pos=dType.GetPose(api)
34 #dType.SetPTPCoordinateParams(api, 20, 20, 20, 20, 1)
35 #dType.SetPTPCmd(api, 3, pos[0]+1000, pos[1], pos[2], pos[3], 1)
36
37 #限位报警的故障模拟 关节1负向限位
38 dType.SetJOGJointParams(api, 10, 10, 10, 10, 10, 10, 10, 10, 1)
39 dType.SetJOGCmd(api, 1, 2, 1)
40
41 #获取系统报警状态
42 ala=dType.GetAlarmsState(api, maxLen=1000)
43
44 #计算报警索引并打印
45 print(' 报警所占总字节数:',ala[1] , ' 检测结果如下:')
46 for i in range(1000):
47     if ala[0][i]!=0:
48         err=1
49         index=i*8+int(log(ala[0][i],2))
50         print(' 检测到报警 索引为:0x%x'%index,ALA[index])
51         break
52     else:
53         pass
54     if(i==999):
55         print(' 没有检测到报警\n')
56

```

运行日志:

```

[15:08:03][65]ERR_LIMIT_AXIS1_NEG alarm triggered
[15:08:03][65]ERR_LIMIT_AXIS1_NEG alarm triggered
[15:08:04][65]ERR_LIMIT_AXIS1_NEG alarm triggered
[15:08:04][65]ERR_LIMIT_AXIS1_NEG alarm triggered
[15:08:05] 报警所占总字节数: 16 检测结果如下:
[15:08:05] 检测到报警 索引为:0x41 关节1负向限位报警

```

结论分析与体会:

这个实验相对于可以让机械臂移动的实验来说,比较抽象,但是让我认识到了对于机械臂和机器人这类物理存在界限和极限的物体,出错报警是很有必要的。