IO类型手柄使用指南

IO类型手柄使用指南

版本

更新日期

更新说明

文档状态

维护责任人

V1.0

2024.5.16

新建

使用中

适用范围

控制器：SRC全系列

实现方式

通过 IO 高低电平的变化，触发模型文件配置的 trigger，调用上传的脚本使 AGV 进行开环运动。 适用设备：有线IO扩展模块，无线IO扩展模块，PLC等。

使用方法

1、上传脚本文件

c78c12cc154f7e4617c92421c4b8625.png

2、配置trigger

image.png

脚本示例

 # -- *coding: utf-8* --

 # @Date : 2022/2/24 17:41

 # @Author : huang qiangsheng

@Version : 1.0

 # @Project : 增加通过di开环发送速度的示例

import math

import time

import sys

sys.path.append("syspy")

from syspy.rbk import MoveStatus, BasicModule, ParamServer from syspy.rbkSim import SimModule

class Module(BasicModule):

```
def __init__(self, r:SimModule, args):
    super(Module, self).__init__()
    self.status = MoveStatus.RUNNING
```

self.vx = 0.5 #最大下发速度 0.5m/s

self.vy = 0.5 #最大下发速度 0.5m/s

```
    self.vw = 30/180.0 * math.pi  #最大角速度30度/s
```

self.forward_di = 1 #前进di

self.back_di = 2 #后退di

```
    self.rot_colockwise_di = 3  #顺时针旋转di
    self.rot_counterclockwise_di = 4  #逆时针旋转di
```

self.moveLeft_di = 5 #左横移di

self.moveRight_di = 6 #右横移di

```
    self.manual_stop_time = 3  #如果di信号持续 3s 没有任务就结束
    self.start_time = time.time() #开始停止的时间
```

def reset(self, r:SimModule):

```
    self.status = MoveStatus.RUNNING
```

self.start_time = time.time()

```
    self.status = MoveStatus.RUNNING

def run(self, r:SimModule,args):
```

vx = 0

vy = 0

vw = 0

DI = r.Di()

```
    nodes = DI.get('node', list())
```

for node in nodes:

node_id = node.get('id', -1)

```
        if node.get('status', False) is True \
            and node.get('forbidden', True) is False \
```

and node_id >= 0:

```
            if node_id == self.forward_di:
                vx = self.vx if vx == 0 else 0
```

elif node_id == self.back_di:

```
                vx = -self.vx if vx == 0 else 0
            if node_id == self.rot_counterclockwise_di:
                vw = self.vw if vw == 0 else 0
            elif node_id == self.rot_colockwise_di:
                vw = -self.vw if vw == 0 else 0
            if node_id == self.moveLeft_di:
                vy = self.vy if vy == 0 else 0
            elif node_id == self.moveRight_di:
                vy = -self.vy if vy == 0 else 0
    r.openSpeed(vx,vy,vw) # 下发开环速度让agv执行
```

dt = 0

```
    if vx == 0 and vy == 0 and vw == 0:
        dt = time.time() - self.start_time
        if dt >= self.manual_stop_time:
            self.status = MoveStatus.FINISHED
```

else:

self.start_time = time.time()

```
    str_log = "[OpenSendIn][{}|{}|{:.3f}]".format(vx, vw,dt)
```

r.logDebug(str_log)

return self.status

if **name** == '**main**':

import syspy.rbkSim

r = syspy.rbkSim.SimModule()

m = Module(r,None)

m.run(r, None)