基础控制_电机

2019年4月8日 19:53

速度类型

子类	说明	方法
SpeedPercent(SpeedValue)	电机最大速度的百分比	将输入的百分比转换为电机的速度
SpeedNativeUnits(SpeedValue)	每秒电机计数	返回电机实际计数
SpeedRPS(SpeedValue)	每秒旋转的圈数	返回期望转速与实际转速的比值（秒）与最大速度的乘积
SpeedRPM(SpeedValue)	每分钟旋转的圈数	返回期望转速与实际转速的比值（分）与最大速度的乘积
SpeedDPS(SpeedValue)	每秒旋转的角度	返回期望角速度与实际角速度比值（秒）与最大速度的乘积
SpeedDPM(SpeedValue)	每分钟旋转的角度	返回期望角速度与实际角速度比值（分）与最大速度的乘积

电机父类

class Motor(Device)

其子类分别为：

MediumMotor(Motor)

LargeMotor(Motor)

ActuonixL1250Motor(Motor)

ActuonixL12100Motor(Motor)

说明	方法	参数	返回值
查看连接端口	def address(self)	self	端口名
控制指令	def command(self)	self	向电机控制器发送一串指令
更改运行模式	def command(self, value)	value是使用motor类的指令字段	无
查看可执行指令	def commands(self)	self	一系列电机控制器支持的指令
查看电机每1转的计数	def count_per_rot(self)	self	电机每每一转的计数
查看电机转1米的计数	def count_per_m(self)	self	电机转一米的计数
查看该电机驱动器的名称	def driver_name(self)	self	电机驱动器名称
查看占空比	def duty_cycle(self)	self	以百分比表示的占空比[-100，100]
写入设置占空比设定值，返回当前值	def duty_cycle_sp(self)	self	当前占空比[-100, 100]
设置占空比设定值	def duty_cycle_sp(self, value)	value是期望占空比	无

查看在整个旅程中电机的计数，以此可计算总距离	def full_travel_count(self)	self	在整个旅程中电机的计数
电机的极性 如果设置为'normal'，+值正转 如果设置为'inversed'，+值反转	def polarity(self)	self	电机当前的极性
设置电机极性	def polarity(self, value)	value可选'normal'或'inversed'	无
查看电机在旋转中的位置（可看作在一条数轴上左右移动） 顺时针旋转时数值增加，逆时针旋转时数值减小	def position(self)		电机在旋转中的位置
设置电机在旋转中的位置	def position(self, value)	self， value	无
查看位置PID的比例常数	def position_p(self)	self	位置PID的比例常数
设置位置PID的比例常数	def position_p(self, value)	self，value	无
查看位置PID的积分常数	def position_i(self)	self	位置PID的积分常数
设置位置PID的积分常数	def position_i(self, value)	self，value	无
查看位置PID的导数常数	def position_d(self)	self	位置PID的导数常数
设置位置PID的导数常数	def position_d(self, value)	self，value	无
查看为以位置为基础的运行模式设置的目标位置，可转换为角度或旋转数	def position_sp(self)	self	以位置为基础的运行模式设置的目标位置，以计数的方式返回
设置以位置为基础的运行模式设置的目标位置	def position_sp(self, value)	self，valu	无
查看理论上的最大速度	def max_speed(self)	self	理论上的最大速度
读取当前每秒tacho计数，负值代表反方向旋转	def speed_sp(self)	self	当前每秒tacho计数
设置每秒tacho计数	def speed_sp(self, value)	self，value	无
读取当前提升设定值	def ramp_up_sp(self)	self	当前提升设定值，以毫秒为单位
修改提升设定值，速度将会按照该值从0-100%的最大速度设置	def ramp_up_sp(self, value)	self， value value必须是正数	无
读取当前减小设定值	def ramp_down_sp(self)	self	当前减小设定值
修改减小设定值，速度将会按照该值从0-100%的最大速度设置	def ramp_down_sp(self, value)	self， value value必须是正数	无
查看速度调节pid的比例常数	def speed_p(self)	self	速度调节的比例常数
修改速度调节pid的比例常数	def speed_p(self, value)	self， value	无
查看速度调节pid的积分常数	def speed_i(self)	self	速度调节pid的积分常数
设置速度调节pid的积分常数	def speed_i(self, value)	self，value	无
查看速度调节pid的导数常数	def speed_d(self)	self	速度调节pid的导数常数

设置速度调节pid的导数常数	def speed_d(self, value)	self，value	无
查看可允许的状态，如'running'、'ramping'、'holding'	def state(self)	self	可允许的状态的列表
查看当前停止状态下的行为	def stop_action(self)	self	当前停止状态下的行为
设置停止状态下的行为	def stop_action(self, value)	self，value	无
查看停止状态下的行为列表，如 `coast`, `brake` and `hold`	def stop_actions(self)	self	停止状态下的行为列表
查看 ‘run-timed'状态下需要持续的时间	def time_sp(self)	self	‘run-timed'状态下需要持续的时间，单位毫秒
设置 ‘run-timed'状态下需要持续的时间	def time_sp(self, value)	self，value	无
在另一个command送达前持续运作	def run_forever(self, **kwargs)		无
运转到 ‘position_sp ‘设置的绝对位置	def run_to_abs_pos(self, **kwargs)		无
运转到当前位置+'position_sp' 的相对位置	def run_to_rel_pos(self, **kwargs)		无
持续运转 ‘time_sp’ 设置的时间	def run_timed(self, **kwargs)		无
根据'duty_cycle_sp'设置的占空比运转，与其他command不同，对'duty_cycle_sp'的修改即时执行	def run_direct(self, **kwargs)		无
停止一切running command	def stop(self, **kwargs)		无
所有参数回到默认设置，并停止运转	def reset(self, **kwargs)		无
查看电机是否供电	def is_running(self)		STATE_RUNNING状态，str/None
查看电机在达到稳定值前是否增加或减小	def is_ramping(self)		STATE_RAMPING状态，str/None
判断电机是停止转变还是保持固定值	def is_holding(self)		STATE_HOLDING状态，str/None
查看电机是否过载	def is_overloaded(self)		STATE_OVERLOADED状态，str/None
查看电机是否故障	def is_stalled(self)		STATE_STALLED状态，str/None
阻塞直到条件满足或者超时	def wait(self, cond, timeout=None)	用于检查I/O操作是否满足设定的cond即self.state timeout单位毫秒	T/F
阻塞直到电机状态变为 `running`、`stalled`或`holding`	def wait_until_not_moving(self, timeout=None)		无

阻塞直到状态变为 s	def wait_until(self, s, timeout=None)	s应是self.state中的状态	
阻塞直到 s 不再为当前状态(self.state)	def wait_while(self, s, timeout=None)	同上	无
控制电机根据旋转数和速度运转	def on_for_rotations(self, speed, rotations, brake=True, block=True)	speed可以是百分比或者SpeedValue对象 rotation是指旋转圈数	无
控制电机根据角度和速度运转	def on_for_degrees(self, speed, degrees, brake=True, block=True)	drgrees是旋转角度	无
控制电机根据位置和速度运转	def on_to_position(self, speed, position, brake=True, block=True)		无
控制电机根据速度、持续时间旋转	def on_for_seconds(self, speed, seconds, brake=True, block=True)		无
控制电机按照速度持续运转	def on(self, speed, brake=True, block=False)		无
控制电机停止运转	def off(self, brake=True)		无
返回当前旋转圈数	def rotations(self)		float
返回旋转角度	def degrees(self)		float

多电机控制

class MoveTank(MotorSet) # 只能控制一对电机

说明	方法	参数	返回值
控制左右电机按速度和角度运转	def on_for_degrees(self, left_speed, right_speed, degrees, brake=True, block=True)	speed可以是百分比或者SpeedValue对象	无
控制左右电机根据旋转数和速度运转	on_for_rotations(self, left_speed, right_speed, rotations, brake=True, block=True)	rotations是旋转圈数	无
控制左右电机根据速度、持续时间旋转	on_for_seconds(self, left_speed, right_speed, seconds, brake=True, block=True)		无
控制左右电机按照速度持续运转	on(self, left_speed, right_speed)		无

同步控制

class MoveSteering(MoveTank) # 同时控制一对电机，将这两个电机视作一个整体

说明	方法	参数	返回值
控制电机组按速度和角度运转	on_for_degrees(self, steering, speed, degrees, brake=True, block=True)	speed可以是百分比或者SpeedValue对象 steering属于[-100，100]，0 表示直走， 负值代表逆时针行驶、正值代表顺时针行驶，数值越大角度越大	无
控制电机组根据旋转数和速度运转	on_for_rotations(self, steering, speed, rotations, brake=True, block=True)	rotations是旋转圈数	无
控制电机组根据速度、持续时间旋转	on_for_seconds(self, steering, speed, seconds, brake=True, block=True)		无
控制电机组按照速度持续运转	on(self, steering, speed)		无
获取左右电机的速度值	get_speed_steering(self, steering, speed)		(left_speed, right_speed)

异步控制

class MoveDifferential(MoveTank)

在MoveTank基础上增加了新的agreement：

wheel_class	ev3dev2.wheel.Wheeld子类，用于获取轮子周长
wheel_distance_mm	轮间距

说明	方法	参数	返回值
走直线	on_for_distance(self, speed, distance_mm, brake=True, block=True)		无
根据轮子半径和移动距离顺时针移动	on_arc_right(self, speed, radius_mm, distance_mm, brake=True, block=True)		无

根据轮子半径和移动距离逆时针移动	on_arc_left(self, speed, radius_mm, distance_mm, brake=True, block=True)		无
根据角度右转	turn_right(self, speed, degrees, brake=True, block=True)		无
根据角度左转	turn_left(self, speed, degrees, brake=True, block=True)		无

操纵杆

class MoveJoystick(MoveTank) # 通过单个操纵杆向量控制一对电机

说明	方法	参数	返回值
将操纵杆的x、y坐标值转化为左右电机的速度百分比并驱动电机运转	on(self, x, y, radius=100.0)	x、y 是操纵杆的横坐标和纵坐标 radius为定义域半径，如x和y属于[-1, 1]，则 radius = 1	无
将操纵杆的移动角度转变为速度百分比	angle_to_speed_percentage(angle)		x和y的速度百分比