

简要的配置教程，更多的建议参考[创客基地](#)

- 查询电压
 - `odrv0.vbus_voltage`
- 置M0的限制电流为10A
 - `odrv0.axis0.motor.config.current_lim = 20`
- 配置 M0上的电流采样范围为 90A
 - `odrv0.axis0.motor.config.requested_current_range = 90`
- 电机校准电流限制值
 - `odrv0.axis0.motor.config.calibration_current = 15`
- 电机转速限制值
 - `odrv0.axis0.controller.config.vel_limit=5000`
- 功率耗散电阻的电阻值，单位为 [Ohm]
 - `odrv0.config.brake_resistance`
- 永磁电机中的磁极对数（磁钢个数除以2）
 - `odrv0.axis0.motor.config.pole_pairs`
- 设置电机类型
 - `odrv0.axis0.motor.config.motor_type`
 - `MOTOR_TYPE_HIGH_CURRENT`
 - `MOTOR_TYPE_GIMBAL`
- 配置编码器
 - `odrv0.axis0.encoder.config.cpr`
- 保存参数
 - `odrv0.save_configuration()`

配置教程2

配置参数

进行配置前建议首先执行一遍擦除配置`odrv0.erase_configuration()`并重启 ODrive 以确保配置恢复为固件默认配置

```
odrv0.erase_configuration()
```

```
odrv0.save_configuration()
```

```
odrv0.reboot()
```

1. 基本配置

- 配置功率耗散电阻阻值，我们使用的功率耗散电阻阻值为 0.5 Ohm，如果不接制动电阻或不想使用功率耗散电阻将此项配置为 0 即可

```
odrv0.config.brake_resistance = 0.5
```

- 配置电源电压低压保护阈值,当电源电压低于 8V 时将停止电机并报错，注意：8V 为极限值建议根据自己所使用的供电更精确的配置以更好地保护 ODrive 主板

```
odrv0.config.dc_bus_undervoltage_trip_level = 8
```

- 配置电源电压过压保护阈值，当电源电压高于 56V 时将停止电机并报错，注意：56V 为极限值建议根据自己所使用的供电更精确的配置以更好地保护 ODrive 主板

```
odrv0.config.dc_bus_overvoltage_trip_level = 56
```

- 配置母线电流过流保护阈值，当母线电流高于 80A 时将停止电机并报错，配置为无穷大时禁用此保护

```
odrv0.config.dc_max_positive_current = 80
```

- 配置电机制动时在母线上产生的反向电流过流保护阈值，当反向电流高于 5A 时将停止电机并报错，配置为负无穷大时禁用此保护

```
odrv0.config.dc_max_negative_current = -5
```

- 配置制动回充电流值为 0A，由于使用的开关电源进行供电所以不具备电能回收功能，所以配置为 0，如果您使用的电池供电则可以根据电池组可以承受的回充电流大小进行配置，当母线上反向回充电流高于此值时，高出的电流能量将会通过功率耗散电阻进行消耗

```
odrv0.config.max_regen_current = 0
```

2. 基本配置

- 配置电机极对数，我们所使用的电机极对数为 7

```
odrv0.axis0.motor.config.pole_pairs = 7
```

- 配置电机参数校准时的电流，此电流值在进行电机参数校准和编码器偏移校准时使用，如果设置的过小在进行编码器偏移校准时电机将没有足够的力量旋转

```
odrv0.axis0.motor.config.calibration_current = 10
```

- 配置电机参数校准时的电压，当电机的相电阻越高此值应该越高，但是如果此值过高会造成电流过大，产生过流保护错误

```
odrv0.axis0.motor.config.resistance_calib_max_voltage = 2
```

- 配置所使用的电机类型为大电流电机

```
odrv0.axis0.motor.config.motor_type = MOTOR_TYPE_HIGH_CURRENT
```

- 配置电机运行的最大电流限制

```
odrv0.axis0.motor.config.current_lim = 20
```

- 配置电机电流采样范围，注意：此值设置后需要重新启动才能生效

```
odrv0.axis0.motor.config.requested_current_range = 60
```

3. 编码器配置

- 配置电机编码器类型为增量式编码器

```
odrv0.axis0.encoder.config.mode = ENCODER_MODE_INCREMENTAL
```

- 我们使用的编码有索引信号，所以配置为启用索引信号输入

```
odrv0.axis0.encoder.config.use_index = True
```

- 配置编码器分辨率，即电机转动一圈多少个计数值

```
odrv0.axis0.encoder.config.cpr = 4000
```

- 设置编码器 PLL 带宽，一般对于高分辨率编码器 (> 4000个计数/转) 此值应该越高，这样有助于减少电机振动

```
odrv0.axis0.encoder.config.bandwidth = 3000
```

- 配置电机进行编码器索引校准时电机开环转动的相关参数

```
odrv0.axis0.config.calibration_lockin.current = 10
```

```
odrv0.axis0.config.calibration_lockin.ramp_time = 0.4
```

```
odrv0.axis0.config.calibration_lockin.ramp_distance = 3.1415927410125732
```

```
odrv0.axis0.config.calibration_lockin.accel = 20
```

```
odrv0.axis0.config.calibration_lockin.vel = 40
```

- current 开环运行时的电流，单位为 [A]，根据负载调整，以电机能够正常转动为准
- ramp_time 电流爬升时间，表示电流从零爬升到设定的 current 值所需要的时间
- ramp_distance 电流爬升时电角度转动距离，单位为 [rad]，配合 ramp_time 参数来缓慢锁定转子相位，可以通过调整 ramp_time、ramp_distance 使编码器索引校准启动的更平顺
- accel 转速爬升的加速度，单位为 [rad/s^2]
- vel 编码器索引校准的运行速度，单位为 [rad/s]

4. 控制器配置

- 配置电机控制模式，此处我们设置为位置模式

```
odrv0.axis0.controller.config.control_mode=CONTROL_MODE_POSITION_CONTROL
```

- 配置电机最大转速，单位为 [turn/s]，例如：此处我们配置为 50 转/秒

```
odrv0.axis0.controller.config.vel_limit = 50
```

- 配置控制器控制增益，pos_gain 为位置环增益，vel_gain 和 vel_integrator_gain 为速度环增益

```
odrv0.axis0.controller.config.pos_gain = 30
```

```
odrv0.axis0.controller.config.vel_gain = 0.1
```

```
odrv0.axis0.controller.config.vel_integrator_gain = 0.03
```

5. 梯形加减速配置

- 配置梯形轨迹模式下电机匀速时的转速为 30 转/秒

```
odrv0.axis0.trap_traj.config.vel_limit = 30
```

- 配置梯形轨迹模式下电机加速时的加速度，单位为 $[\text{turn/s}^2]$ ，例如：此处我们配置为 5,表示电机从静止加速到 5 转/秒 需要一秒钟的时间

```
odrv0.axis0.traj_traj.config.accel_limit = 5
```

- 配置梯形轨迹模式下电机减速时的加速度，单位为 $[\text{turn/s}^2]$ ，例如：此处我们配置为 5，表示电机从 5 转/秒减速到静止需要一秒钟的时间

```
odrv0.axis0.traj_traj.config.decel_limit = 5
```

- 将输入模式配置为梯形轨迹模式

```
odrv0.axis0.controller.config.input_mode = INPUT_MODE_TRAP_TRAJ
```

6. 保存配置

```
odrv0.save_configuration()
```

```
odrv0.reboot()
```

校准工作

1. 电机校准

- 进行电机参数校准，校准过程将测量电机相电阻、电机相电感，并根据相电阻相电感自动生成电流环 PI 增益，当听到电机发出 ‘哔’ 声后表示电机参数测量操作已完成

```
odrv0.axis0.requested_state = AXIS_STATE_MOTOR_CALIBRATION
```

- 列出状态信息，如果返回状态没有错误则电机校准OK，可以继续下面步骤，如果出现错误则要根据错误信息分析原因，然后输入 `dump_errors(odrv0, True)` 对错误进行清除后重新尝试

```
dump_errors(odrv0)
```

```
dump_errors(odrv0, True)
```

- 将电机 `pre_calibrated` 设置为 `True`，表示电机已校准下次重新启动后可以直接使用本次校准的结果

```
odrv0.axis0.motor.config.pre_calibrated = True
```

2. 编码器校准

- 进行电机编码器索引校准，启动后电机将朝着一个方向开环旋转直到检测到索引信号脉冲时停止

```
odrv0.axis0.requested_state = AXIS_STATE_ENCODER_INDEX_SEARCH
```

- 如果想要指定编码器索引校准时电机的转动方向可以通过改变以下三个参数的正负号来实现：

```
odrv0.axis0.config.calibration_lockin.ramp_distance = -3.1415927410125732
```

```
odrv0.axis0.config.calibration_lockin.accel = -20
```

```
odrv0.axis0.config.calibration_lockin.vel = -40
```

- 列出状态信息，如果返回状态没有错误则编码器索引校准OK，可以继续进行下面步骤，如果出现错误则要根据错误信息分析原因，然后输入 `dump_errors(odrv0, True)` 对错误进行清除后重新尝试

```
dump_errors(odrv0)
```

```
dump_errors(odrv0, True)
```

- 进行电机编码器偏移校准，启动后电机将朝着一个方向开环旋转然后反方向旋转然后停止，开环旋转时的电流大小为 `odrv0.axis0.motor.config.calibration_current`

```
odrv0.axis0.requested_state = AXIS_STATE_ENCODER_OFFSET_CALIBRATION
```

- 列出状态信息，当得到上述返回状态没有错误则表明编码器偏移校准OK，可以继续进行下面步骤，如果出现错误则要根据错误信息分析原因，然后输入 `dump_errors(odrv0, True)` 对错误进行清除后重新尝试

```
dump_errors(odrv0)
```

```
dump_errors(odrv0, True)
```

- 将编码器 `pre_calibrated` 设置为 `True`，表示编码器已校准下次重新启动后可以直接使用本次校准的结果，注意：由于我们使用的并非绝对值编码器而是增量编码器，所以每次重新启动后需要进行编码器索引校准，如果您使用的编码器没有索引信号 (只有 AB) 则每次重新启动后需要进行编码器偏移校准

```
odrv0.axis0.encoder.config.pre_calibrated = True
```

3. 保存电机校准结果

```
odrv0.save_configuration()
```

```
odrv0.reboot()
```

控制电机运行

- 重新启动后进行编码器索引校准。注意：由于我们使用的并非绝对值编码器而是增量编码器，所以每次重新启动后需要进行编码器索引校准，如果您使用的编码器没有索引信号 (只有 AB) 则每次重新启动后需要进行编码器偏移校准

```
odrv0.axis0.requested_state = AXIS_STATE_ENCODER_INDEX_SEARCH
```

- 进入闭环运行模式，此时电机将维持在当前位置，如果电机产生震动则说明控制增益设置的不合适需要适当调低，当尝试用手转动电机轴时如果电机反应很迟钝反作用力很弱则说明需要适当调高控制增益，参考 [3.4 控制器配置](#3.4 控制器配置)

```
odrv0.axis0.requested_state = AXIS_STATE_CLOSED_LOOP_CONTROL
```

- 控制电机运行到 50 圈位置，此过程电机将经历加速、匀速、减速三个过程，最终停在我们命令它移动到的位置。如果想反方向转动，位置值改为负数即可，此值也可以为小数，如：`input_pos = 0.5`，则电机正向转动到180°机械角度

```
odrv0.axis0.controller.input_pos = 50
```

- 如果想释放电机，发送此命令即可让 ODrive 进入待机模式，此时电机可以被自由转动

```
odrv0.axis0.requested_state = AXIS_STATE_IDLE
```

避免每次重新启动后手动编码器索引校准、进入闭环控制

- 每次重新启动后必须要进行编码器索引校准，然后进入闭环控制后才能向电机发送运行指令，这两个步骤可以通过配置让 ODrive 自动替我们完成

```
odrv0.axis0.config.startup_encoder_index_search = True
```

```
odrv0.axis0.config.startup_closed_loop_control = True
```

```
odrv0.save_configuration()
```

示例：

```
odrv0.erase_configuration()
odrv0.save_configuration()
odrv0.reboot()
odrv0.config.brake_resistance = 0.5
odrv0.config.dc_bus_undervoltage_trip_level = 8
odrv0.config.dc_bus_overvoltage_trip_level = 56
odrv0.config.dc_max_positive_current = 40
odrv0.config.dc_max_negative_current = -5
odrv0.config.max_regen_current = 0

odrv0.axis0.motor.config.pole_pairs = 7
odrv0.axis0.motor.config.calibration_current = 10
odrv0.axis0.motor.config.resistance_calib_max_voltage = 2
odrv0.axis0.motor.config.motor_type = MOTOR_TYPE_HIGH_CURRENT
odrv0.axis0.motor.config.current_lim = 25
odrv0.axis0.motor.config.requested_current_range = 60
odrv0.axis0.encoder.config.mode = ENCODER_MODE_INCREMENTAL
odrv0.axis0.encoder.config.use_index = True
odrv0.axis0.encoder.config.cpr = 4000
odrv0.axis0.encoder.config.bandwidth = 3000

odrv0.axis0.config.calibration_lockin.current = 10
odrv0.axis0.config.calibration_lockin.ramp_time = 0.4
odrv0.axis0.config.calibration_lockin.ramp_distance = 3.1415927410125732
odrv0.axis0.config.calibration_lockin.accel = 20
odrv0.axis0.config.calibration_lockin.vel = 40

odrv0.axis0.controller.config.control_mode=CONTROL_MODE_POSITION_CONTROL
odrv0.axis0.controller.config.vel_limit = 200
odrv0.axis0.controller.config.pos_gain = 35
odrv0.axis0.controller.config.vel_gain = 0.1
odrv0.axis0.controller.config.vel_integrator_gain = 0.0075

odrv0.axis0.trap_traj.config.vel_limit = 30
odrv0.axis0.trap_traj.config.accel_limit = 5
odrv0.axis0.trap_traj.config.decel_limit = 5
```

```

odrv0.axis0.controller.config.input_mode = INPUT_MODE_TRAP_TRAJ

odrv0.erase_configuration()
odrv0.save_configuration()
odrv0.reboot()
odrv0.config.brake_resistance = 0.5
odrv0.config.dc_bus_undervoltage_trip_level = 8
odrv0.config.dc_bus_overvoltage_trip_level = 56
odrv0.config.dc_max_positive_current = 40
odrv0.config.dc_max_negative_current = -5
odrv0.config.max_regen_current = 0

odrv0.axis1.motor.config.pole_pairs = 7
odrv0.axis1.motor.config.calibration_current = 10
odrv0.axis1.motor.config.resistance_calib_max_voltage = 2
odrv0.axis1.motor.config.motor_type = MOTOR_TYPE_HIGH_CURRENT
odrv0.axis1.motor.config.current_lim = 25
odrv0.axis1.motor.config.requested_current_range = 60
odrv0.axis1.encoder.config.mode = ENCODER_MODE_INCREMENTAL
odrv0.axis1.encoder.config.use_index = True
odrv0.axis1.encoder.config.cpr = 4000
odrv0.axis1.encoder.config.bandwidth = 3000

odrv0.axis1.config.calibration_lockin.current = 10
odrv0.axis1.config.calibration_lockin.ramp_time = 0.4
odrv0.axis1.config.calibration_lockin.ramp_distance = 3.1415927410125732
odrv0.axis1.config.calibration_lockin.accel = 20
odrv0.axis1.config.calibration_lockin.vel = 40

odrv0.axis1.controller.config.control_mode=CONTROL_MODE_POSITION_CONTROL
odrv0.axis1.controller.config.vel_limit = 200
odrv0.axis1.controller.config.pos_gain = 35
odrv0.axis1.controller.config.vel_gain = 0.1
odrv0.axis1.controller.config.vel_integrator_gain = 0.015

odrv0.axis0.trap_traj.config.vel_limit = 30
odrv0.axis0.trap_traj.config.accel_limit = 10
odrv0.axis0.trap_traj.config.decel_limit = 10
odrv0.axis0.controller.config.input_mode = INPUT_MODE_TRAP_TRAJ

```

- odrv0.axis0.motor.config.current_lim = 35
- odrv0.axis0.motor.config.requested_current_range = 60
- odrv0.axis0.motor.config.calibration_current = 6
- odrv0.axis0.controller.config.vel_limit=500
- odrv0.config.brake_resistance = 0.5

- `odrv0.axis0.motor.config.pole_pairs = 7`
- `odrv0.axis0.motor.config.motor_type = MOTOR_TYPE_HIGH_CURRENT`
- `odrv0.axis0.encoder.config.cpr = 4000`
- `odrv0.save_configuration()`
- `odrv0.reboot()`
- `odrv0.axis0.requested_state = AXIS_STATE_FULL_CALIBRATION_SEQUENCE`
- `odrv0.axis0.requested_state = AXIS_STATE_CLOSED_LOOP_CONTROL`
- `odrv0.axis0.controller.input_pos = 50`
- `odrv0.axis1.motor.config.current_lim = 35`
- `odrv0.axis1.motor.config.requested_current_range = 60`
- `odrv0.axis1.motor.config.calibration_current = 6`
- `odrv0.axis1.controller.config.vel_limit=500`
- `odrv0.config.brake_resistance = 0.5`
- `odrv0.axis1.motor.config.pole_pairs = 7`
- `odrv0.axis1.motor.config.motor_type = MOTOR_TYPE_HIGH_CURRENT`
- `odrv0.axis1.encoder.config.cpr = 4000`
- `odrv0.save_configuration()`
- `odrv0.reboot()`
- `odrv0.axis1.requested_state = AXIS_STATE_FULL_CALIBRATION_SEQUENCE`
- `odrv0.axis1.requested_state = AXIS_STATE_CLOSED_LOOP_CONTROL`