

MAVROS 常用话题简介

一、mavros 订阅消息:

1、global_position 订阅 GPS 数据

消息名称: mavros/global_position/global

类型名称: sensor_msgs::NavSatFix.h

类型所在头文件: sensor_msgs/NavSatFix.h

常用类成员变量:

1、float64 latitude *//经*

2、float64 longitude *//纬*

3、float64 altitude *//海拔*

4、float64[9] position_covariance

5、uint8 position_covariance_type

2、imu_pub 订阅 IMU 信息

消息名称: 滤波后的 mavros/imu/data (或原始信息 mavros/imu/data_raw)

类型名称: sensor_msgs::Imu

类型所在头文件: sensor_msgs/Imu.h

常用类成员变量:

1、geometry_msgs::Quaternion orientation *//旋转四元数 (xyzw) *

2、float64[9] orientation_covariance *//方差*

3、geometry_msgs::Vector3 angular_velocity *//3 轴角速度 (xyz) *

4、float64[9] angular_velocity_covariance *//方差*

5、geometry_msgs::Vector3 linear_acceleration *//线性加速度 (xyz) *

6、float64[9] linear_acceleration_covariance *//方差*

3、local_position 订阅本地位置数据

消息名称: mavros/local_position/pose

类型名称: geometry_msgs::PoseStamped

类型所在头文件: geometry_msgs/PoseStamped.h

常用类成员变量:

1、geometry_msgs::Pose pose

4、manual_control 订阅遥控器的值

消息名称: mavros/manual_control/control

类型名称: mavros_msgs::ManualControl

类型所在头文件: mavros_msgs/ManualControl.h

5、sys_status 查询系统状态

消息名称: mavros/state

类型名称: mavros_msgs::State

类型所在头文件: mavros_msgs/State.h

6、waypoint 航点信息

消息名称: mavros/mission/waypoint

类型名称: geometry_msgs::WaypointList

类型所在头文件: mavros_msgs/WaypointList.h

二、mavros 发布消息：

1、actuator_control 控制飞控 IO 输出（混控器）

消息名称：mavros/actuator_control

类型名称：geometry_msgs::PoseStamped

类型所在头文件：mavros_msgs/Actuator_Control.h

常用类成员变量：

1、uint8 group_mix *要控制的混控器分组 1-8（control group）*

2、float32[8] controls *控制量（前四个分别是：roll、pitch、yaw、thrust）*

2、setpoint_accel 控制期望的加速度

消息名称：mavros/setpoint_accel/accel

类型名称：geometry_msgs::Vector3Stamped

类型所在的头文件：geometry_msgs/Vector3Stamped.h

常用类成员变量：

1、geometry_msgs::Vector3 vector *三轴加速度**

3、setpoint_attitude 控制期望的姿态

消息名称：mavros/setpoint_attitude/attitude

类型名称：geometry_msgs::PoseStamped

类型所在的头文件：geometry_msgs/PoseStamped.h

常用类成员变量：

1、geometry_msgs::Pose pose *三个欧拉角，或者是四元数任选其一**

4、setpoint_position 控制期望的位置（相对坐标）

消息名称：mavros/setpoint_position/local

类型名称：geometry_msgs::PoseStamped

类型所在的头文件：geometry_msgs/PoseStamped.h

常用类成员变量：

1、geometry_msgs::Pose pose //NED 坐标系下的位置（xyz），只有 position 成员变量生效

5、setpoint_velocity 控制期望的速度

消息名称：mavros/setpoint_velocity/cmd_vel

类型名称：geometry_msgs::TwistStamped

类型所在的头文件：geometry_msgs/TwistStamped.h

常用类成员变量：

1、geometry_msgs::Twist twist *//三轴速度*

2、geometry_msgs::Twist 类成员变量：

3、geometry_msgs::Vector3 linear *//三轴线性速度*

4、geometry_msgs::Vector3 angular *//三轴角速度*

6、setpoint_position 控制期望的位置（GPS 坐标）

消息名称：mavros/setpoint_position/global

类型名称：mavros_msgs::GlobalPositionTarget

类型所在的头文件：mavros_msgs/GlobalPositionTarget.h

常用类成员变量：

1、uint8 coordinate_frame *//5 为绝对 GPS 坐标系，6 为相对高度 GPS 坐标*

2、uint16 type_mask

三、mavros 服务:

1、arming Services 加解锁服务

消息名称: mavros/cmd/arming

类型名称: mavros_msgs::CommandBool

类型所在的头文件: mavros_msgs/CommandBool.h

2、模式切换消息名称: mavros/set_mode

类型名称: mavros_msgs::SetMode

类型所在的头文件: mavros_msgs/SetMode.h

南京超维空间智能科技有限公司

一、常用接收的话题详解：

1.1 系统状态

消息名称：mavros/state

类型：mavros_msgs::State

头文件：mavros_msgs/State.h

成员变量：

//这里只列PX4的，APM的用 `rosmmsg show mavros_msgs/State` 查看

```
string MODE_PX4_MANUAL=MANUAL
string MODE_PX4_ACRO=ACRO
string MODE_PX4_ALTITUDE=ALTCTL
string MODE_PX4_POSITION=POSCTL
string MODE_PX4_OFFBOARD=OFFBOARD
string MODE_PX4_STABILIZED=STABILIZED
string MODE_PX4_RATTITUDE=RATTITUDE
string MODE_PX4_MISSION=AUTO.MISSION
string MODE_PX4_LOITER=AUTO.LOITER
string MODE_PX4_RTL=AUTO.RTL
string MODE_PX4_LAND=AUTO.LAND
string MODE_PX4_RTGS=AUTO.RTGS
string MODE_PX4_READY=AUTO.READY
string MODE_PX4_TAKEOFF=AUTO.TAKEOFF
std_msgs/Header header
  uint32 seq
  time stamp
  string frame_id
bool connected //是否连接
bool armed //是否解锁
bool guided
bool manual_input
string mode //当前飞行模式
uint8 system_status //系统状态
```

1.2 GPS 数据

消息名称: `mavros/global_position/global`

类型: `sensor_msgs::NavSatFix`

头文件: `sensor_msgs/NavSatFix.h`

成员变量:

```
uint8 COVARIANCE_TYPE_UNKNOWN=0uint8
COVARIANCE_TYPE_APPROXIMATED=1uint8
COVARIANCE_TYPE_DIAGONAL_KNOWN=2uint8
COVARIANCE_TYPE_KNOWN=3std_msgs/Header header
  uint32 seq
  time stamp
  string frame_idsensor_msgs/NavSatStatus status
  int8 STATUS_NO_FIX=-1
  int8 STATUS_FIX=0
  int8 STATUS_SBAS_FIX=1
  int8 STATUS_GBAS_FIX=2
  uint16 SERVICE_GPS=1
  uint16 SERVICE_GLONASS=2
  uint16 SERVICE_COMPASS=4
  uint16 SERVICE_GALILEO=8
  int8 status
  uint16 servicefloat64 latitudefloat64 longitudefloat64 altitudefloat64[9]
position_covarianceuint8 position_covariance_type
```

1.3 本地位置

消息名称: `mavros/local_position/pose`

类型: `geometry_msgs::PoseStamped`

头文件: `geometry_msgs/PoseStamped.h`

成员变量:

std_msgs/Header header

uint32 seq

time stamp

string frame_idgeometry_msgs/Pose pose

geometry_msgs/Point position

float64 x

float64 y

float64 z

geometry_msgs/Quaternion orientation

float64 x

float64 y

float64 z

float64 w

1.4 三轴速度

消息名称: `mavros/local_position/velocity`

类型: `geometry_msgs::TwistStamped`

头文件: `geometry_msgs/TwistStamped.h`

成员变量:

`std_msgs/Header header`

`uint32 seq`

`time stamp`

`string frame_id geometry_msgs/Twist twist`

`geometry_msgs/Vector3 linear`

`float64 x`

`float64 y`

`float64 z`

`geometry_msgs/Vector3 angular`

`float64 x`

`float64 y`

`float64 z`

1.5 IMU 原始数据

消息名称: `mavros/imu/data(_raw)//data_raw` 为原始数据, `data` 为滤波后数据

类型: `sensor_msgs::Imu`

头文件: `sensor_msgs/Imu.h`

成员变量:

`std_msgs/Header header`

`uint32 seq`

`time stamp`

`string frame_id geometry_msgs/Quaternion orientation`

`float64 x`

`float64 y`

`float64 z`

`float64 w float64[9] orientation_covariance geometry_msgs/Vector3 angular_velocity`

`float64 x`

`float64 y`

`float64 z float64[9] angular_velocity_covariance geometry_msgs/Vector3`

`linear_acceleration`

`float64 x`

`float64 y`

`float64 z float64[9] linear_acceleration_covariance`

1.6 遥控器数据

消息名称: `mavros/manual_control/control`

类型: `mavros_msgs::ManualControl`

头文件: `mavros_msgs/ManualControl.h`

成员变量:

`std_msgs/Header header`

`uint32 seq`

`time stamp`

`string frame_id float32 xfloat32 yfloat32 zfloat32 ru int16 buttons`

1.7 waypoint 查询航点

消息名称: `mavros/mission/waypoints`

类型: `mavros_msgs::WaypointList`

头文件: `mavros_msgs/WaypointList.h`

成员变量:

`uint16 current_seq mavros_msgs/Waypoint[] waypoints`

`uint8 FRAME_GLOBAL=0`

`uint8 FRAME_LOCAL_NED=1`

`uint8 FRAME_MISSION=2`

`uint8 FRAME_GLOBAL_REL_ALT=3`

`uint8 FRAME_LOCAL_ENU=4`

`uint8 frame //坐标系 3: 经纬度为绝对, 高度为相对(起飞点为 0) uint16 command //`

命令 id 16: 航点 `bool is_current`

`bool autocontinue`

`float32 param1`

`float32 param2`

`float32 param3`

`float32 param4`

`float64 x_lat`

`float64 y_long`

`float64 z_alt`

二、常用发布的话题

2.1、 控制期望的位置（GPS 坐标）

2.1.1、 mavros/setpoint_raw/global

消息名称: mavros/setpoint_raw/global

类型: mavros_msgs::GlobalPositionTarget

头文件: mavros_msgs/GlobalPositionTarget.h

成员变量:

```
uint8 FRAME_GLOBAL_INT=5
uint8 FRAME_GLOBAL_REL_ALT=6
uint8 FRAME_GLOBAL_TERRAIN_ALT=11
uint16 IGNORE_LATITUDE=1 //只想控制经纬高则设置
type_mask=8+16+32+64+128+256+512+1024+2048
uint16 IGNORE_LONGITUDE=2
uint16 IGNORE_ALTITUDE=4
uint16 IGNORE_VX=8 //只想控制速度则设置
type_mask=1+2+4+64+128+256+512+1024+2048
uint16 IGNORE_VY=16uint16 IGNORE_VZ=32
uint16 IGNORE_AFX=64 //只想控制加速度则设置
type_mask=1+2+4+8+16+32+512+1024+2048
uint16 IGNORE_AFY=128
uint16 IGNORE_AFZ=256
uint16 FORCE=512 //Force in af vector flag
uint16 IGNORE_YAW=1024
uint16 IGNORE_YAW_RATE=2048std_msgs/Header header
uint32 seq
time stamp
string frame_iduint8 coordinate_frame //5 为绝对 amsl 高度, 6 为相对 amsl 高度
uint16 type_mask //用于确定控制哪些量
float64 latitude //经纬高
float64 longitude
float32 altitude //in meters, AMSL or above terraingeometry_msgs/Vector3 velocity //速度
float64 x
```

```
float64 y
float64 zgeometry_msgs/Vector3 acceleration_or_force //加速度 float64 x
float64 y
float64 zfloat32 yaw //偏航 float32 yaw_rate //偏航角速度
```

2.1.2、mavros/setpoint_position/global

消息名称: `mavros/setpoint_position/global`

类型: `geographic_msgs::GeoPoseStamped`

头文件: `geographic_msgs/GeoPoseStamped.h`

成员变量:

```
std_msgs/Header header
uint32 seq
time stamp
string frame_idgeographic_msgs/GeoPose pose
geographic_msgs/GeoPoint position
float64 latitude
float64 longitude
float64 altitude
geometry_msgs/Quaternion orientation
float64 x
float64 y
float64 z
float64 w
```

2.2、控制期望的位置

消息名称: `mavros/setpoint_position/local`

类型: `geometry_msgs::PoseStamped`

头文件: `geometry_msgs/PoseStamped.h`

成员变量:

`std_msgs/Header header`

`uint32 seq`

`time stamp`

`string frame_id` `geometry_msgs/Pose pose` //NED 坐标系下的位置(xyz), 只有 `position` 成

员变量生效 `geometry_msgs/Point position`

`float64 x`

`float64 y`

`float64 z`

`geometry_msgs/Quaternion orientation`

`float64 x`

`float64 y`

`float64 z`

`float64 w`

2.3、控制期望的速度

消息名称: `mavros/setpoint_velocity/cmd_vel_unstamped`

类型: `geometry_msgs::Twist`

头文件: `geometry_msgs/Twist.h`

成员变量:

```
geometry_msgs/Vector3 linear //三轴线速度  float64 x
float64 y
float64 zgeometry_msgs/Vector3 angular //三轴角速度  float64 x
float64 y
float64 z
```

2.4、控制期望的加速度

消息名称: `mavros/setpoint_accel/accel`

类型: `geometry_msgs::Vector3Stamped`

头文件: `geometry_msgs/Vector3Stamped.h`

成员变量:

```
std_msgs/Header header
uint32 seq
time stamp
string frame_idgeometry_msgs/Vector3 vector //三轴加速度  float64 x
float64 y
float64 z
```

2.5、 控制期望的姿态

消息名称: `mavros/setpoint_attitude/attitude`

类型: `geometry_msgs::PoseStamped`

头文件: `geometry_msgs/PoseStamped.h`

成员变量:

```
std_msgs/Header header
  uint32 seq
  time stamp
  string frame_id geometry_msgs/Pose pose //三个欧拉角，或者 4 元数任选其一
geometry_msgs/Point position
  float64 x
  float64 y
  float64 z
geometry_msgs/Quaternion orientation
  float64 x
  float64 y
  float64 z
  float64 w
```

2.6、 控制飞控 IO 输出(混控器)

消息名称: `mavros/actuator_control`

类型: `mavros_msgs::ActuatorControl`

头文件: `mavros_msgs/ActuatorControl.h`

成员变量:

```
uint8 PX4_MIX_FLIGHT_CONTROL=0 uint8
PX4_MIX_FLIGHT_CONTROL_VTOL_ALT=1 uint8 PX4_MIX_PAYLOAD=2 uint8
PX4_MIX_MANUAL_PASSTHROUGH=3 std_msgs/Header header
  uint32 seq
  time stamp
  string frame_id uint8 group_mix //要控制的混控器分组 1-8 (control group) float32[8]
controls //控制量 (前四个分别是, roll pitch yaw thrust)
```


三、常用服务

3.1、上/解锁服务

消息名称: `mavros/cmd/arming`

类型: `mavros_msgs::CommandBool`

头文件: `mavros_msgs/CommandBool.h`

请求参数

`bool value //true 解锁 false 上锁---bool successuint8 result //错误信息`

3.2、模式切换

!!!!!!!!!!!! 真实环境测试时不要用软件切换 offboard 模式!!!!!!!!!!!!

!!!!!!!!!!!! 真实环境测试时不要用软件切换 offboard 模式!!!!!!!!!!!!

!!!!!!!!!!!! 真实环境测试时不要用软件切换 offboard 模式!!!!!!!!!!!!

特别是像例程那样反复在 while 循环中设置 offboard 模式，要是控制不了机载电脑关闭该节点的话飞机就永远保持 offboard 模式了，如果发出的还是速度指令的话就祈祷别砸到人吧。所以设置电子围栏也很重要。此处建议使用我们提供的代码，可以实现遥控器接管无人机，同时针对室外无人机气压计以及 GPS 漂移问题给出了软件校准

消息名称: `mavros/set_mode`

类型: `mavros_msgs::SetMode`

头文件: `mavros_msgs/SetMode.h`

```
uint8 MAV_MODE_PREFLIGHT=0uint8
MAV_MODE_STABILIZE_DISARMED=80uint8
MAV_MODE_STABILIZE_ARMED=208
uint8 MAV_MODE_MANUAL_DISARMED=64
uint8 MAV_MODE_MANUAL_ARMED=192
uint8 MAV_MODE_GUIDED_DISARMED=88
uint8 MAV_MODE_GUIDED_ARMED=216
uint8 MAV_MODE_AUTO_DISARMED=92
uint8 MAV_MODE_AUTO_ARMED=220
uint8 MAV_MODE_TEST_DISARMED=66
uint8 MAV_MODE_TEST_ARMED=194
uint8 base_mode //基本模式，若 customs_mode 非空，则无效 string custom_mode //可用的
模式见 mavros/mavros/src/lib/uas_stringify.cpp 中的 px4（如果用的是 APM 就搜 APM）
---bool mode_sent //true 表示正确切换模式
```

以下为 PX4 可用的飞行模式

```
static const cmode_map px4_cmode_map{{
{ px4::define_mode(px4::custom_mode::MAIN_MODE_MANUAL),          "MANUAL" },
{ px4::define_mode(px4::custom_mode::MAIN_MODE_ACRO),           "ACRO" },
```

```
{ px4::define_mode(px4::custom_mode::MAIN_MODE_ALTCTL),      "ALTCTL" },
{ px4::define_mode(px4::custom_mode::MAIN_MODE_POSCTL),      "POSCTL" },
{ px4::define_mode(px4::custom_mode::MAIN_MODE_OFFBOARD),    "OFFBOARD" },
{ px4::define_mode(px4::custom_mode::MAIN_MODE_STABILIZED),  "STABILIZED" },
{ px4::define_mode(px4::custom_mode::MAIN_MODE_RATTITUDE),   "RATTITUDE" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_MISSION), "AUTO.MISSION" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_LOITER), "AUTO.LOITER" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_RTL), "AUTO.RTL" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_LAND), "AUTO.LAND" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_RTGS), "AUTO.RTGS" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_READY), "AUTO.READY" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_TAKEOFF), "AUTO.TAKEOFF" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_FOLLOW_TARGET), "AUTO.FOLLOW_TARGET" },
{ px4::define_mode_auto(px4::custom_mode::SUB_MODE_AUTO_PRECLAND), "AUTO.PRECLAND" } };
```