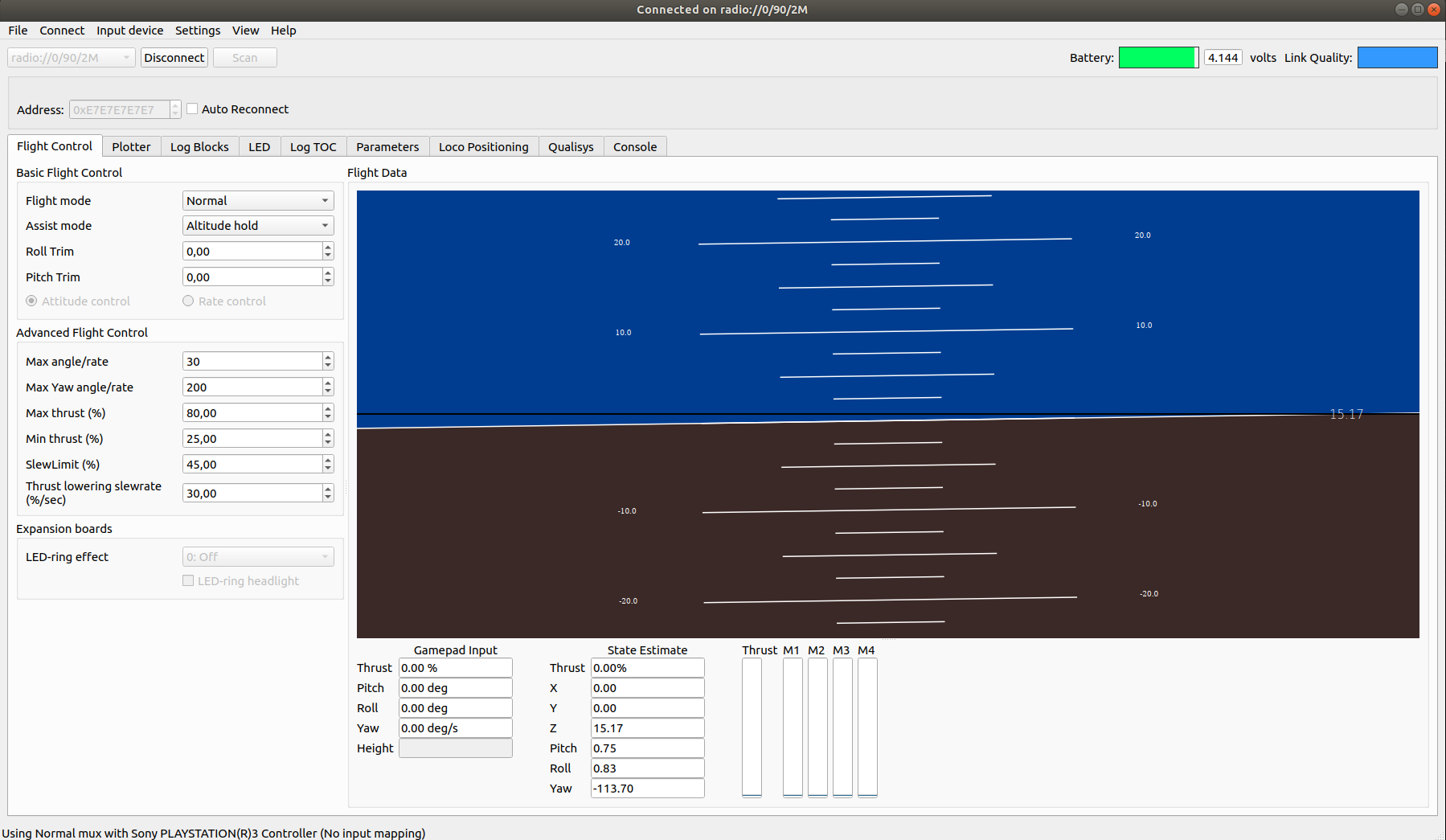
# 2.1 Crazyflie PC client基本介绍

Crazyflie PC client能够实时显示和控制Crazyflie，实现了用户界面和高级控制（例如手柄处理）的功能。与Crazyflie的通信以及控制Crazyflie的CRTP协议由cflib项目处理。



# 2.2 Crazyflie PC client安装步骤

## 2.2.1 Windows端安装

### 1）下载安装程序

·浏览器打开<https://github.com/bitcraze/crazyflie-clients-python/releases>

·下载名为cfclient-win32-install-XXX.exe的最新版本文件。

·运行安装程序

### 2）安装硬件及驱动

·将Crazyradio PA插入USB端口（将游戏控制器插入USB端口）

·安装USB驱动https://wiki.bitcraze.io/doc:crazyradio:install\_windows\_zadig

## 2.2.2 Linux端安装

### 1）设置用户权限（重要）

·打开一个终端，输入：

$ sudo groupadd plugdev

$ sudo usermod -a -G plugdev $USER

·创建99-crazyradio.rules文件，在终端中输入

$ sudo gedit /etc/udev/rules.d/99-crazyradio.rules

在99-crazyradio.rules中添加

# Crazyradio (normal operation)

SUBSYSTEM=="usb", ATTRS{idVendor}=="1915", ATTRS{idProduct}=="7777", MODE="0664", GROUP="plugdev"

# Bootloader

SUBSYSTEM=="usb", ATTRS{idVendor}=="1915", ATTRS{idProduct}=="0101", MODE="0664", GROUP="plugdev"

保存文件并退出。

·创建99-crazyflie.rules文件，在终端中输入

$ sudo gedit /etc/udev/rules.d/99-crazyflie.rules

在99-crazyflie.rules中添加

SUBSYSTEM=="usb", ATTRS{idVendor}=="0483", ATTRS{idProduct}=="5740", MODE="0664", GROUP="plugdev"

保存文件并退出。

·在终端输入以便重载udev-rules

$ sudo udevadm control --reload-rules

$ sudo udevadm trigger

完成上述操作后重新登录用户，以使其生效。

最后插上Crazyradio PA（如之前已经插上，拔插一次）。

### 2）安装Crazyflie程序库和客户端

* 安装crazyflie-lib-python库

安装相关依赖文件

$ sudo apt-get install git python3 python3-pip python3-pyqt4 python3-numpy python3-zmq  
$ sudo pip3 install pyusb==1.0.0b2  
$ sudo pip3 install pyqtgraph appdirs

安装程序库

$ mkdir ~/crazyflie

$ cd ~/crazyflie

$ git clone https://github.com/bitcraze/crazyflie-lib-python.git

$ cd crazyflie-lib-python

$ pip3 install --user -e .

* 安装crazyflie-client-python客户端

$ cd ~/crazyflie

$ git clone https://github.com/bitcraze/crazyflie-clients-python.git

$ cd crazyflie-clients-python

$ pip3 install cfclient

启动客户端

$ cd ~/crazyflie/crazyflie-clients-python

$ python3 bin/cfclient

另外，重启系统后也可以使用cfclient来运行客户端，或者使用python3 -m cfclient.gui。

细节参考：

<https://github.com/bitcraze/crazyflie-clients-python/blob/master/README.md>

# 2.3 Crazyflie PC client使用教程

## 2.3.1主界面介绍

* 窗口标题显示连接状态。
* 2.连接配置

Address：Crazyflie所分配的连接地址，默认为0xE7E7E7E7E7；

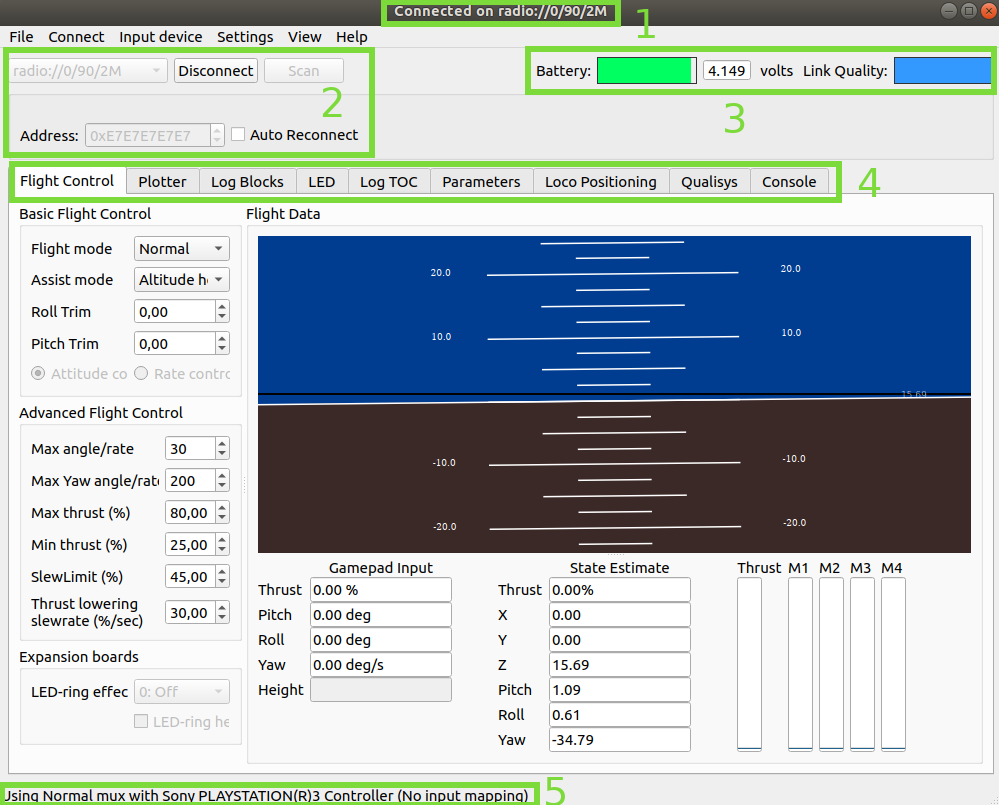
Scan：扫描选中地址Address所对应的可用Crazyflies（目前我只尝试过连接一架Crazyflie，这里官方教程所写的复数，不知道是否可以同时连接多架机）；

Connect：与列表URI（图中为radio://0/90/2M/0xE7E7E7E7E7）所指定的Crazyflie建立连接；

Disconnect：与当前Crazyflie取消连接；

Auto Reconnect：连接断开后自动重连。

* 3.Crazyflie电池Battery和连接质量Link Quality。
* 4.具有特定功能的选项(详情见后文)。
* 5. 选用的输入接口（手柄）。



## 2.3.2手柄控制飞行基本步骤

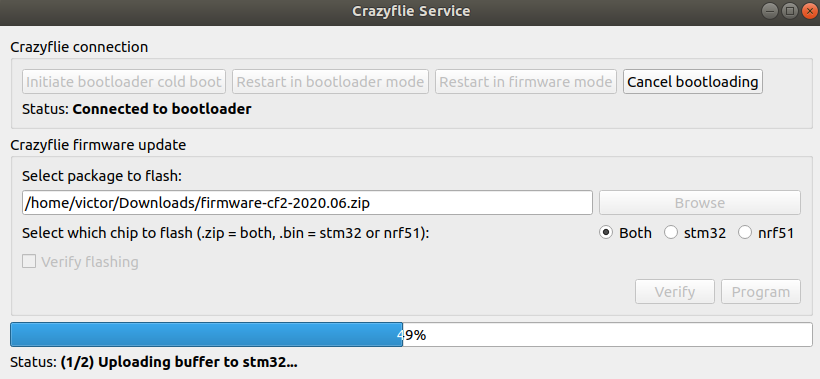
使用手柄操控Crazyflie飞行的基本步骤如下：

* 打开Crazyflie PC client
* 插入手柄和Crazyradio (PA)
* 使用前确认手柄正常工作，且油门推杆thrust为0。手柄各项数值的变化应该在飞行数据框中可见
* 修改Address为目标Crazyflie的Address，默认为0xE7E7E7E7E7
* 点击Scan，搜寻是否有可用的Crazyflie，等待一会儿
* 在Connect按钮左侧的下拉菜单中选择要连接的crazyflie。连接到URI，例如:radio://0/90/2M
* 客户端现在将连接到crazyflie并进行握手
* 握手结束后就可以手动飞行了。最棘手的部分是推力杆的控制，自己把握吧…

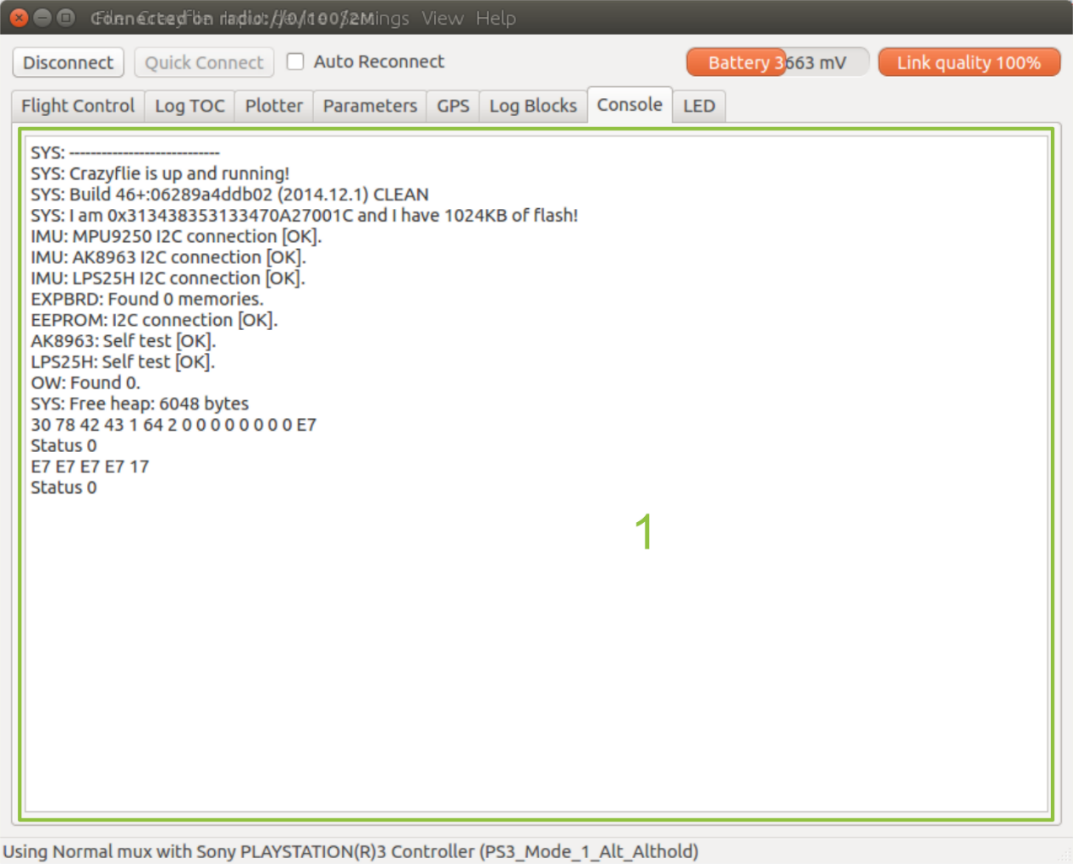
## 2.3.3固件更新

更新Crazyflie2.X固件的流程如下

* 确保Crazyflie已从客户端断开连接并关闭电源
* 点击上方菜单栏Connect，下拉菜单中选Bootloader，进入如下界面
* 按住Crazyflie开关按钮约3秒，直到蓝色LED M2开始闪烁，进入引导加载程序模式。（如果刷写了错误的nRF51固件，则可能要从未通电状态启动，重新进入固件刷写模式）
* 点击Initiate bootloader cold boot，几秒后，状态应显示为“已连接到引导加载程序”
* 点击Browser并选择需要刷写的固件（提前在主页下载好zip文件，Windows客户端可以在线刷写固件，推荐刷写最新的固件）
* 点击Program并等待，进度条将从0％变为100％两次，因为两个处理器的固件已上载到Crazyflie
* 点击Restart in firmware mode，Crazyflie会重新启动，并马上更新
* 关闭bootloader窗口



查看固件版本：View->Tabs->Console,连接到Crazyflie后查看Console的输出，如下图。



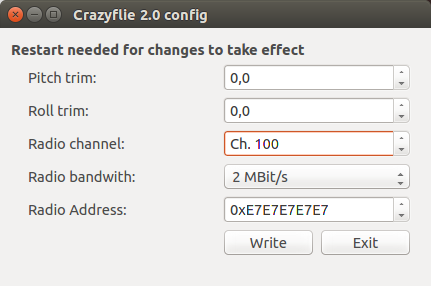
## 2.3.4固件配置

目前可以更改以下的参数，具体步骤如下

·首先根据之前教程连接上Crazyflie2.X

·点击Connect->Configure 2.X得到下图

·改写好数据后，点击Write，然后关闭



·Pitch trim / Roll trim：俯仰、滚转通道固定偏置，微调修正，基本不用改

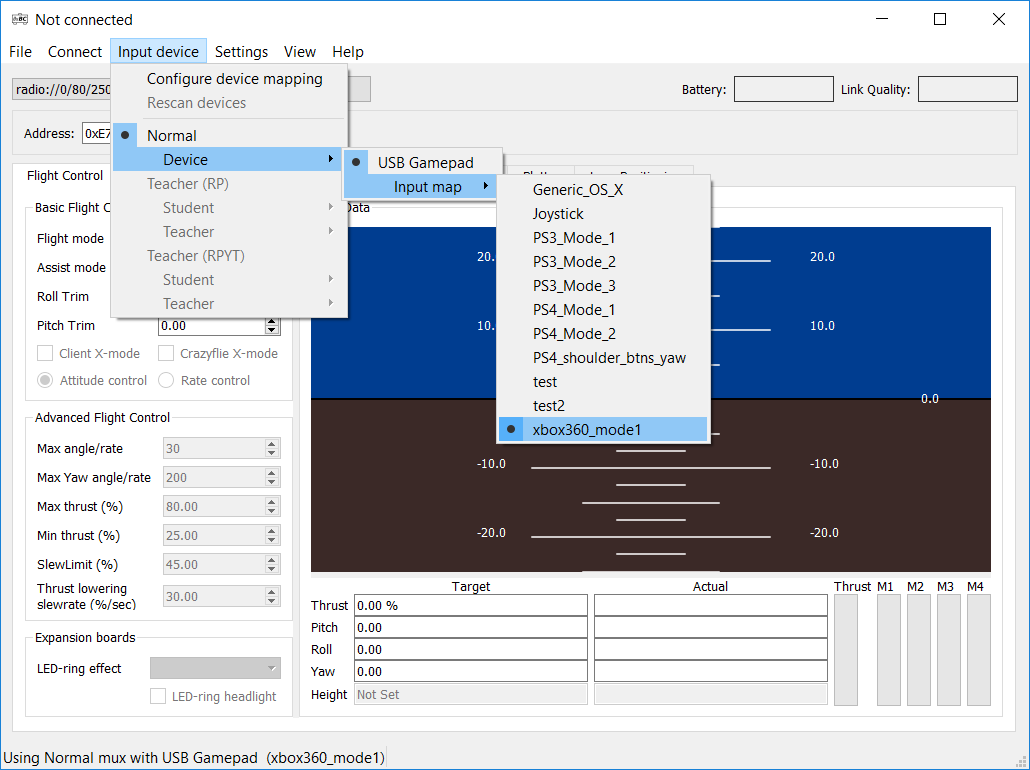
·Radio channel：可以设置为0到125之间的任意值，对应于频率从2400MHz到2525MHz。

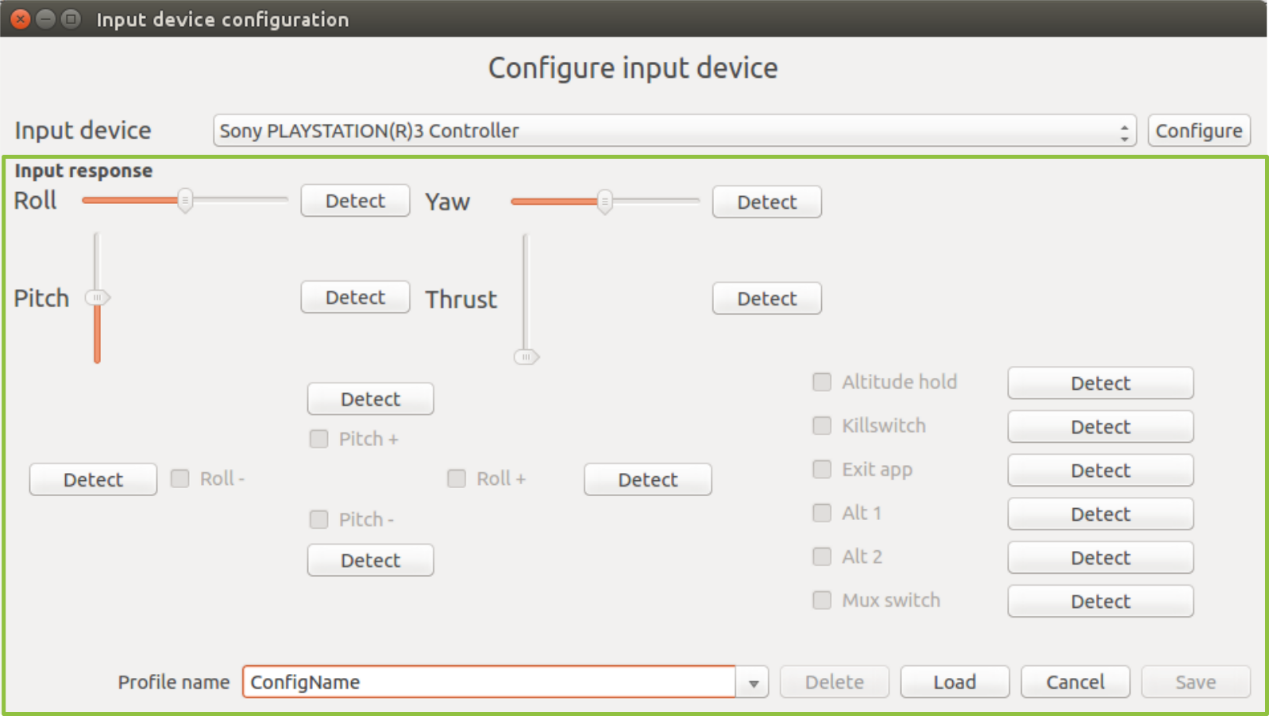
·Radio bandwidth：无线电带宽通常设置为250k、1M或2M。较低的带宽有较长的范围，但更容易与其他信号冲突。室内使用时，推荐使用1M或2Ｍ，可以减少与WiFi冲突的风险。

·Radio address：分配给Crazyflie的通信地址，需要区分清楚每架Crazyflie对应的地址。

## 2.3.5手柄设置

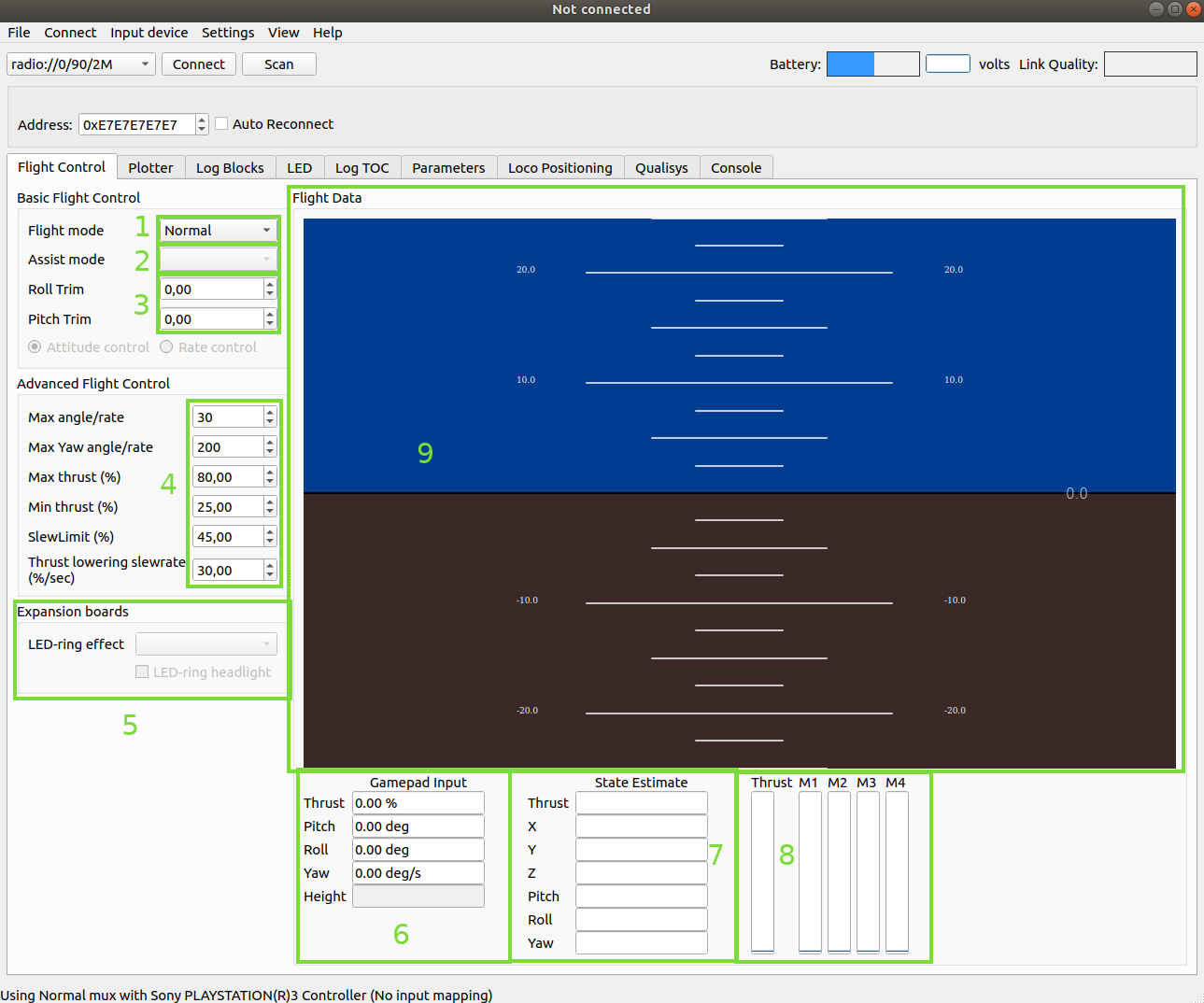
通过输入设备控制所连接的Crazyflie，这里的输入设备通常是手柄（Xbox和PS3/4等），但任何有至少4个模拟轴的输入设备都可以。当前输入设备和映射可以在Input device中设置。





## 2.3.6飞行设置

飞行时使用的正常视图如下。



·**1.飞行模式**

Normal基本模式: 初学者使用

Advanced高级模式: 设置解锁最大角度，设置最大油门

·**2.辅助模式**（油门摇杆变为高度控制遥杆）

Altitude hold定海拔模式: 保持飞行海拔，需要气压计支持

Position hold定点模式: 保持当前位置，需要光流和TOF支持

Height hold定高模式: 保持相对高度, 触发时保持高于地面40cm，需要TOF支持

Hover悬停模式: 触发时保持高于地面 40cm，并悬停在起飞点，需要光流和TOF支持

·**3.角度修正**

Roll Trim : 翻滚角修正，用于弥补传感器水平安装误差

Pitch Trim : 俯仰角修正，用于弥补传感器水平安装误差

·**4.高级飞行设置**

Max angle最大倾角: 设置最大允许的俯仰和翻滚角度 roll/pitch

Max yaw rate最大自选速度: 设置允许的偏航速度 yaw

Max thrust最大油门: 设置最大油门

Min thrust最小油门: 设置最小油门

Slew limit回转极限: 防止油门骤降，油门低于该值时，将被平滑的接管

Slew rate回转率: 油门到回转极限时，最大的油门值

·5.设置拓展板，目前可设置Led环和头灯效果（如果环连接）

·6.从客户端发出的控制Crazyflie的目标值

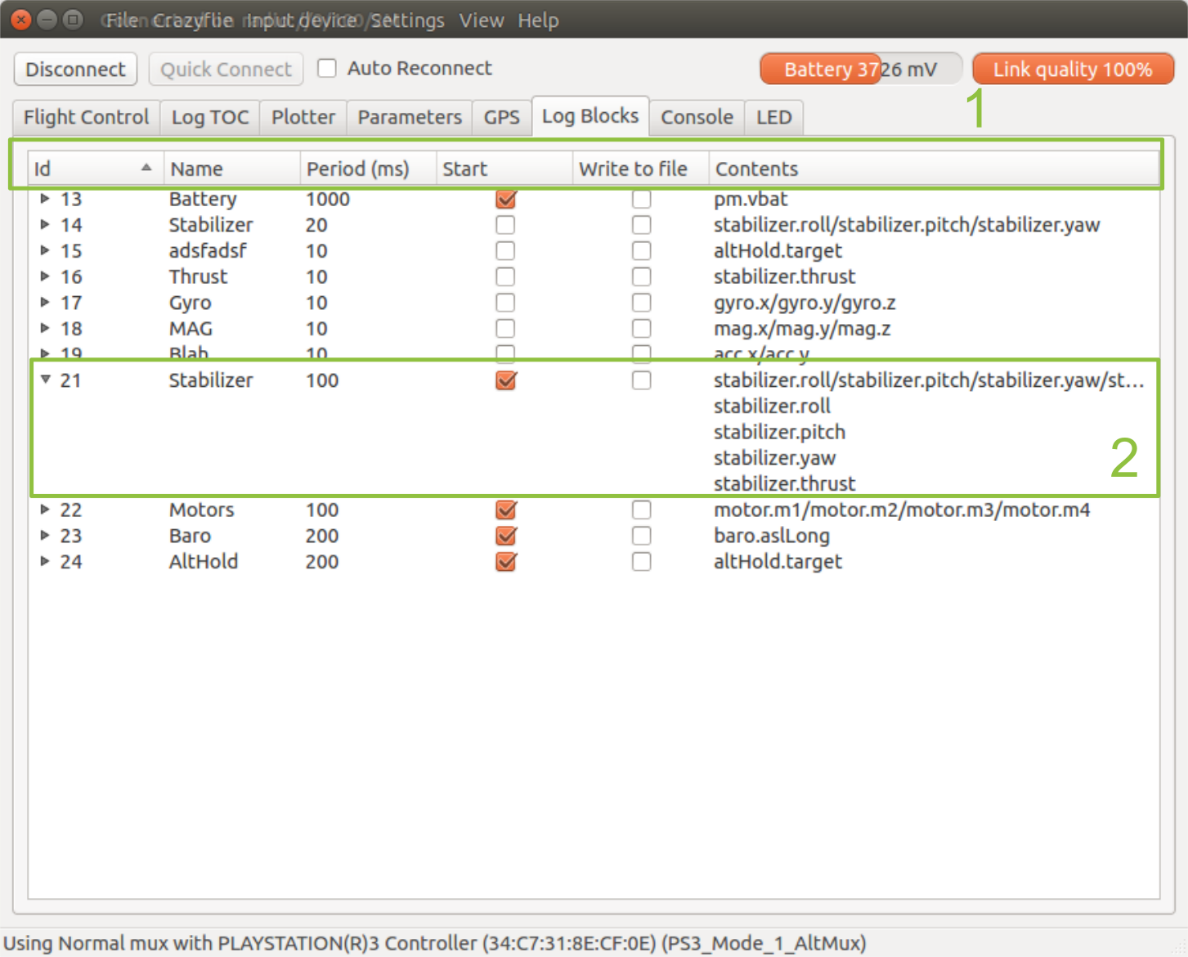
·7.Crazyflie上记录的实际值

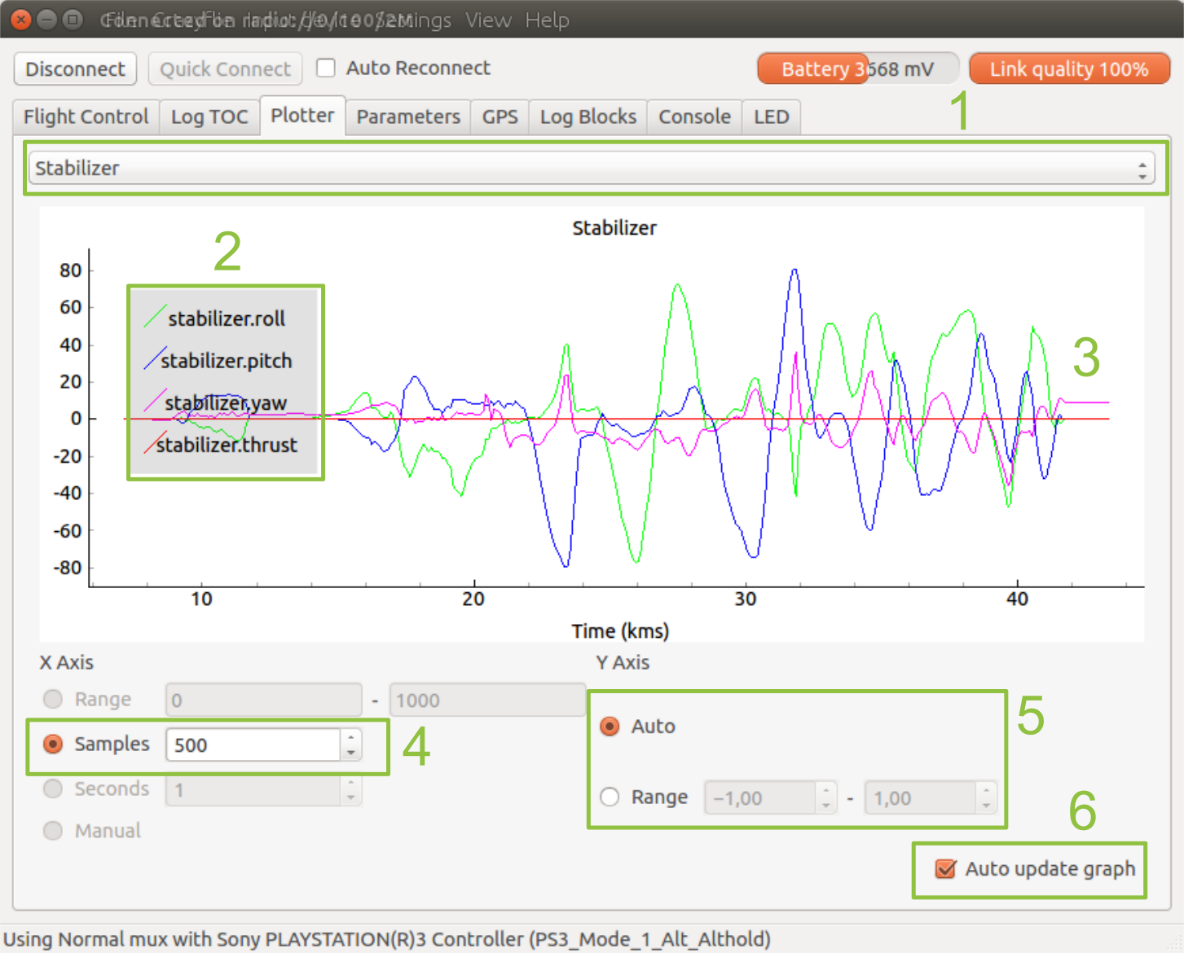
·8.推力和四个电机输出量

·9.地平线指示

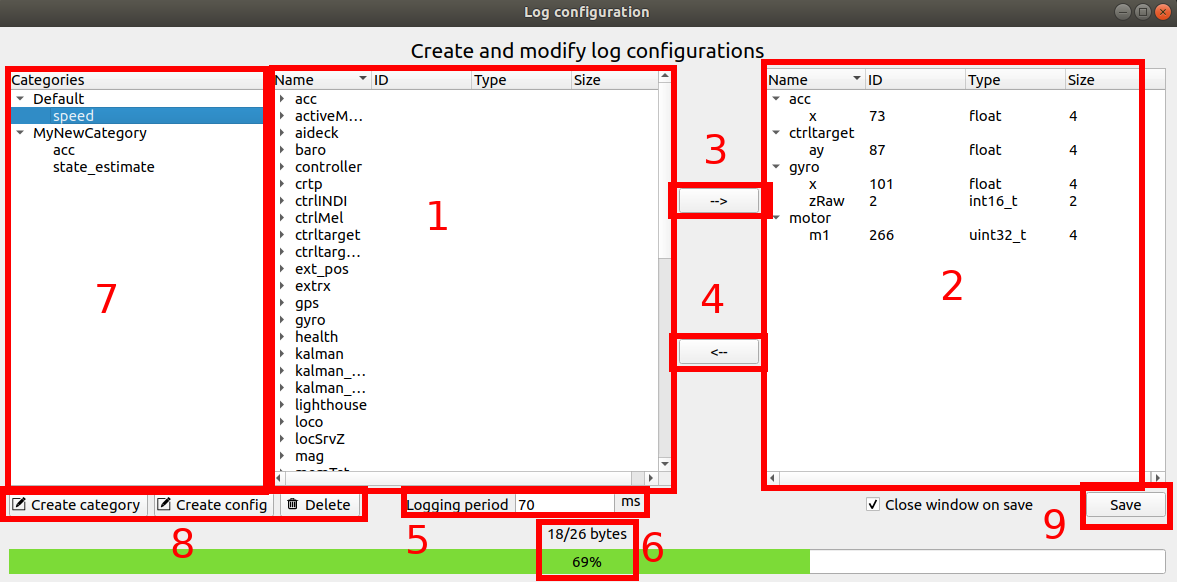
## 2.3.7日志配置

Crazyflie日志记录框架允许实时记录变量的状态，如无人机姿态和电量等信息，这些可以在客户端中实时显示出来。全部的日志变量可在log TOC中看到。变量被记录在日志块Log Blocks中，一个日志块是同时记录多个变量的列表。日志块可以自己定义，可以绘制在Plotter中且可以保存到指定的文件中。





点击Settings->Logging configuration得到如下界面



·1.在Crazyflie的日志变量列表

·2.在当前日志块中记录变量

·3.在日志块中添加选定的Crazyflie日志变量

·4.从日志块中移除选定的日志块变量

·5.日志变量采样及Crazyflie发送日志块到客户端的周期。最小周期为10ms (100Hz)，最高到2550ms，10ms步进。

·6.日志块中变量使用的字节数。

·7.文件树结构，可以对日志块进行分类。

·8.创建/删除类别或日志配置，也可右键单击类别树来实现。

·9.保存日志块配置

详见参考资料：

https://www.bitcraze.io/documentation/repository/crazyflie-clients-python/master/userguides/userguide\_client/