

ArduPilot 源码深度解析

来延涛

2016-5-26

目录

- ArduPilot架构变化及设计思想
- 编译及启动过程
- ArduPilot调试开发步骤
- 导航控制制导过程分析
- PX4Flow工程解析及图像定位



ArduPilot 架构变化及设计思想

APMrover2	APMrover2: fixed build from AP_Menu changes	4 years ago
ArduBoat	Merge the changes from APM_Camera branch into ArduCopter	4 years ago
ArduCopter	ArduCopter: update firmware version to 2.8.1	4 years ago
ArduPlane	Update ArduCopter and ArduPlane dependencies on AP_Menu	4 years ago
Tools	ArduPPM: redundancy mode	4 years ago
apo	Added AP_Board to APO	5 years ago
archive	more directories for the archive	5 years ago
cmake	Worked on ArduCopter cmake options.	4 years ago
libraries	AP_OpticalFlow: fix example sketch to work with modified AP_OpticalFlow...	4 years ago
gltignore	Mission Planner Cleanup	4 years ago
project	ArduPlane cmake build working.	5 years ago
COPYING.txt	cmake overhaul, cleaned up readme, added license	5 years ago
Doxyfile.in	Working on doxygen support.	5 years ago
README.txt	Switched to project specific cmake lists.	4 years ago
reformat.sh	better commit message for uncrustify	4 years ago
uncrustify.cpp.cfg	fixes to preprocessor retab settings	4 years ago
uncrustify_headers.cfg	small changes to uncrustify_headers.cft	4 years ago



ArduCopter-2.8.1

github	GitHub: update issue template	3 days ago
APMrover2	Rover: moved EK2 to a new parameter index	17 hours ago
AntennaTracker	Tracker: fix formatting and gps based alt difference	a day ago
ArduCopter	Copter: fixed ESC calibration on PixRacer	17 hours ago
ArduPlane	Plane: moved EK2 to a new parameter index	17 hours ago
Tools	ci: increase ccache size	an hour ago
benchmarks	waf: add glenclark Waf tool	6 months ago
docs	docs: Fix typos	12 days ago
libraries	HAL_PX4: prevent rate changes once oneshot is setup	24 minutes ago
mk	mk: remove support for flymaple	2 days ago
modules	PX4Firmware: submodule update	17 hours ago
tests	tests: add macro for printing test parameter	9 days ago
editorconfig	all: Change the editorconfig so that it won't want to reformat	2 years ago
glttributes	Revert "glttributes: automatically clean up newlines in source files"	a year ago
gltignore	gltignore: ignore mav param in the entire tree	2 months ago
gltmodules	git: changed URLs for modules	2 months ago
pydevproject	AP_Relay: add -1 Disabled to list of param values	2 years ago
travis.yml	travis: set cron env var	12 hours ago
COPYING.txt	cmake overhaul, cleaned up readme, added license	5 years ago
Doxyfile.in	cft: Fix typos	12 days ago
Makefile	build: allow -j on top level makefile	7 months ago
Makefile.waf	waf: Makefile.waf: use check-all command instead of -alltests	4 months ago
README-WAF.md	waf: add --debug configuration option	16 days ago
README.md	AP_HAL_FLYMAPLE: remove hal	2 days ago
Vagrantfile	Fix vagrant build issues with submodules.	11 months ago
eclipse.cproject	Eclipse: template project files	10 months ago
eclipse.project	Eclipse: template project files	10 months ago
reformat.sh	reformat: remove mention to .pde files	16 days ago
uncrustify.cpp.cfg	cft: Fix typos	12 days ago
uncrustify_headers.cfg	cft: Fix typos	12 days ago
2866522b	waf: print build summary	6 hours ago

ArduCopter-master

工程结构变化:

- 增加modules文件夹，引用其他开源工程（Nuttx、PX4Firmware）
- 去掉archive文件夹，改为在library下增加ap_hal层提高可移植性
- 主分支master中剥离对mega2560的支持，改为单独的分支

ArduPilot 架构变化及设计思想

```
void loop()
{
    if (num_samples >= NUM_IMU_SAMPLES_FOR_100HZ) {
        Log_Write_Data(50, (int32_t)(timer - fast_loopTimer));
        fast_loop();
        run_50hz_loop = !run_50hz_loop;

        if (run_50hz_loop) {
            Log_Write_Data(51, (int32_t)(timer - fiftyhz_loopTimer));
            update_trig();
            calc_loiter_pitch_roll();
            update_GPS();

            // perform 10hz tasks
            // -----
            medium_loop();

            // Stuff to run at full 50hz, but after the med loops
            // -----
            fifty_hz_loop();

            counter_one_hertz++;

            // trigger our 1 hz loop
            if (counter_one_hertz >= 50) {
                super_slow_loop();
            }
            perf_mon_counter++;
            if (perf_mon_counter > 600) {
                if (g.compass_enabled) {
                    compass.accumulate();
                }
                Log_Write_Performance();
            }
        }
    } else {
        if (num_samples < NUM_IMU_SAMPLES_FOR_100HZ-1) {
            if (g.compass_enabled) {
                compass.accumulate();
            }
        }
    }
}
```

ArduCopter-2.8.1



```
void Copter::loop()
{
    ins.wait_for_sample();

    uint32_t timer = micros();

    perf_info_check_loop_time(timer - fast_loopTimer);

    G_Dt = (float)(timer - fast_loopTimer) / 1000000.0f;
    fast_loopTimer = timer;

    mainLoop_count++;

    fast_loop();

    scheduler.tick();

    uint32_t time_available = (timer + MAIN_LOOP_MICROS) - micros();
    scheduler.run(time_available);
}
```

ArduCopter-master

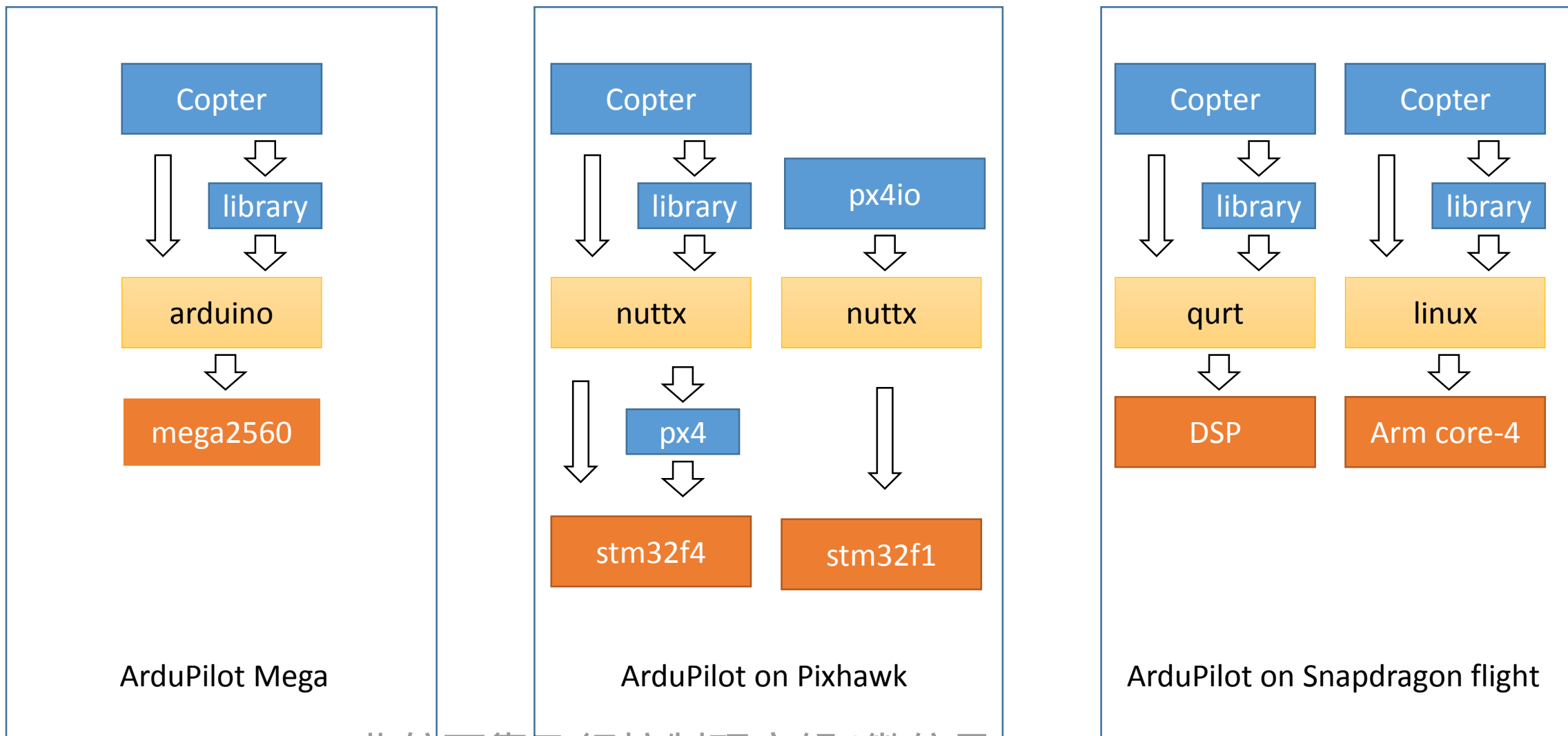


```
const AP_Scheduler::Task Copter::scheduler_tasks[] = {
    SCHED_TASK(rc_loop, 100, 130),
    SCHED_TASK(throttle_loop, 50, 75),
    SCHED_TASK(update_GPS, 50, 200),
    #if OPTFLOW == ENABLED
    SCHED_TASK(update_optical_flow, 200, 160),
    #endif
    SCHED_TASK(update_batt_compass, 10, 120),
    SCHED_TASK(read_aux_switches, 10, 50),
    SCHED_TASK(arm_motors_check, 10, 50),
    SCHED_TASK(auto_disarm_check, 10, 50),
    SCHED_TASK(auto_trim, 10, 75),
    SCHED_TASK(read_rangefinder, 20, 100),
    SCHED_TASK(update_altitude, 10, 100),
    SCHED_TASK(run_nav_updates, 50, 100),
    SCHED_TASK(update_thr_average, 100, 90),
    SCHED_TASK(three_hz_loop, 3, 75),
    SCHED_TASK(compass_accumulate, 100, 100),
    SCHED_TASK(barometer_accumulate, 50, 90),
    #if PRECISION_LANDING == ENABLED
    SCHED_TASK(update_precland, 50, 50),
    #endif
    #if FRAME_CONFIG == HELI_FRAME
    SCHED_TASK(check_dynamic_flight, 50, 75),
    #endif
    SCHED_TASK(update_notify, 50, 90),
    SCHED_TASK(one_hz_loop, 1, 100),
    SCHED_TASK(ekf_check, 10, 75),
    SCHED_TASK(landinggear_update, 10, 75),
    SCHED_TASK(lost_vehicle_check, 10, 50),
    SCHED_TASK(gcs_check_input, 400, 180),
    SCHED_TASK(gcs_send_heartbeat, 1, 110),
    SCHED_TASK(gcs_send_deferred, 50, 550),
    SCHED_TASK(gcs_data_stream_send, 50, 550),
    SCHED_TASK(update_mount, 50, 75),
    SCHED_TASK(update_trigger, 50, 75),
    SCHED_TASK(ten_hz_logging_loop, 10, 350),
    SCHED_TASK(twentyfive_hz_logging, 25, 110),
    SCHED_TASK(dataflash_periodic, 400, 300),
    SCHED_TASK(perf_update, 0.1, 75),
    SCHED_TASK(read_receiver_rssi, 10, 75),
    SCHED_TASK(rpm_update, 10, 200),
    SCHED_TASK(compass_cal_update, 100, 100),
    ...
}
```

代码结构变化

- 原主循环中通过计数实现分频的操作用scheduler类实现，即通过向task列表添加函数实现伪定时器
- 结构变的简洁但是本质未变化：
 - 顺序调用，因此一旦某个函数超时就会导致整体循环时间增加
 - 通过软计数实现分频调用，会导致整个调度过分依赖主循环运行频率的稳定性
- 这种结构导致apm on pixhawk主循环运行时间不稳定，范围约为（0.9ms~4.1ms）

ArduPilot 架构变化及设计思想



ArduPilot 架构变化及设计思想

- 为自驾仪而诞生并发展
 - 控制部分相对较弱，更强调的是mission的完成，因此直接拿来
做航拍控制效果并不好
- 一切皆文件
 - 所有设备驱动都是文件
 - 所有消息都以文件形式存在（/_obj_下）
- 分层设计
 - 控制逻辑分层
 - 功能分层
- 向上兼容
 - Scheduler这种落后的调度器一直存在
 - 主循环中循环完成绝大部分计算工作

May 2007 - Chris Anderson starts [DIYDrones.com](#) while building his [Lego mindstorm base UAV](#).

Sep 2008 - Jordi builds traditional helicopter UAV able to [fly autonomously](#) and [wins the first Sparkfun AVC competition](#).



Jordi's traditional helicopter

2009 - Chris Anderson & Jordi Munoz found 3D Robotics

May 2009 - [First ArduPilot board](#) (using [thermopiles](#)) released by Jordi/3DRobotics

Nov 2009 - [ardupilot code respository](#) created by Jordi

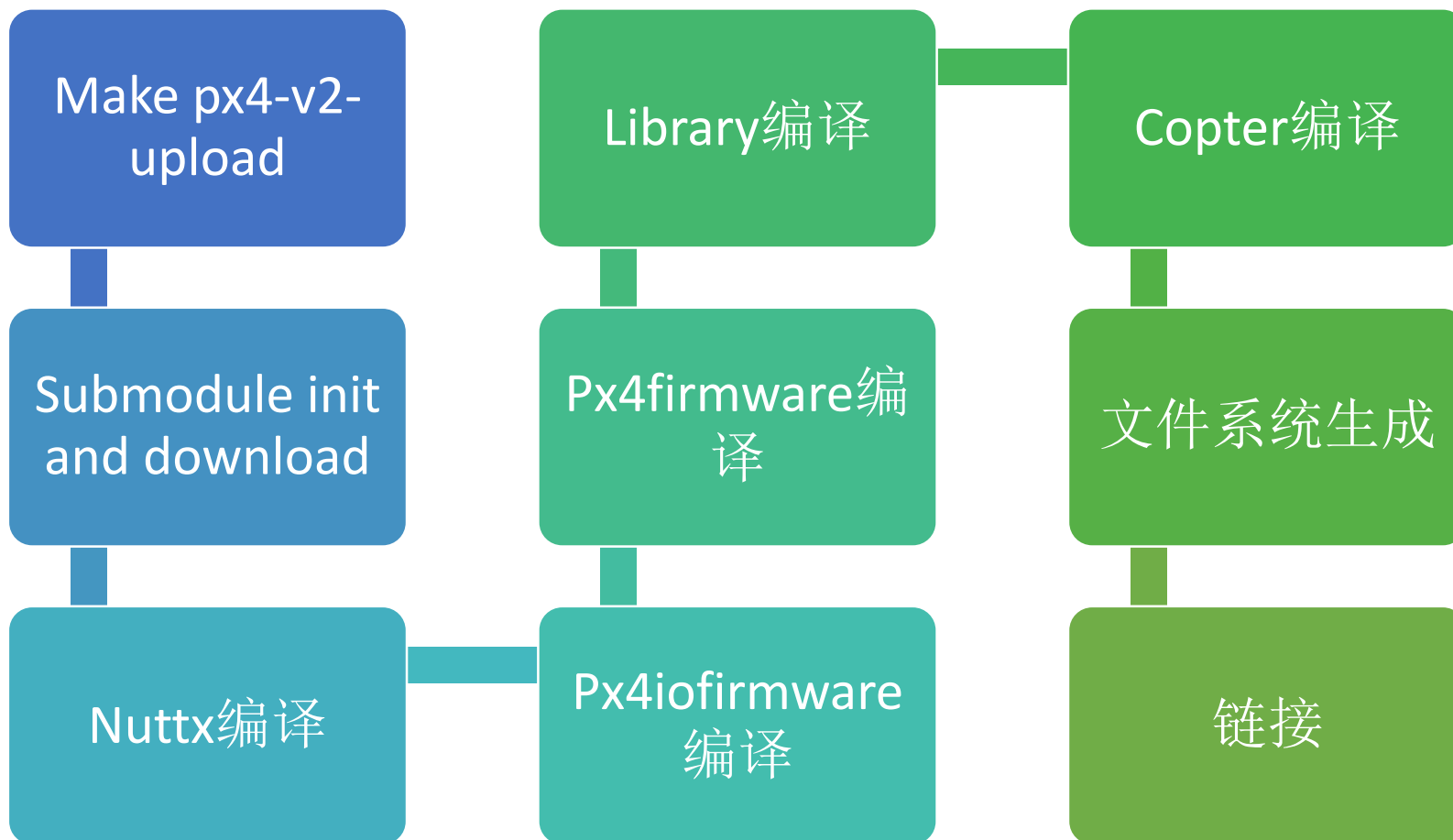
Nov 2009 - first version of ArduIMU written by Jordi, Doug Weibel, Jose Julio using DCM from William Premerlani

Nov 2009 - Feb 2010 - ArduPilot rewritten from scratch (v2.5) by Jason and includes interrupt driven RC input, RC throttle failsafe, RTL, Loiter, Circle, Crosstrack correction, decent stabilization, Fly-By-Wire, system events, 4 channel RC output, and 2-way telemetry. ([Post](#))

Dec 2009 - [first IMU based autonomous plane mission](#) flown by Doug with modified ArduPilot v2.4.

ardupilot官网history部分截图，可以看出整个ardupilot发展都是为参加飞行任务比赛而不断发展

编译及启动过程



编译及启动过程

NuttX初始化

内核初始化：系统资源、进程调度器、文件系统、console初始化

Init进程（nsh进程：初始化bin文件系统、打开console口接受指令）

Init进程解析rcS

挂载USB

初始化LED

挂载SD驱动

判断是否执行rcAPM（检测/fs/microsd/APM/nostart）

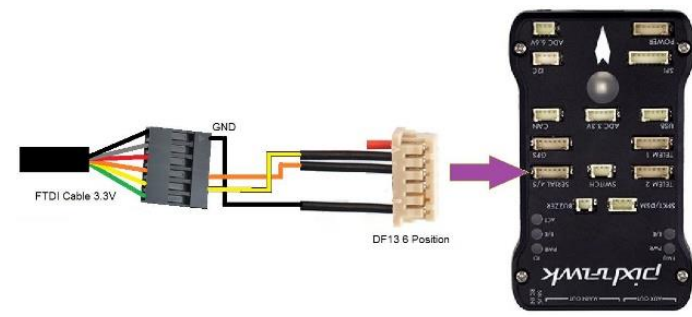
Init进程解析rcAPM

Px4io.bin更新升级（/etc/px4io/px4io.bin）

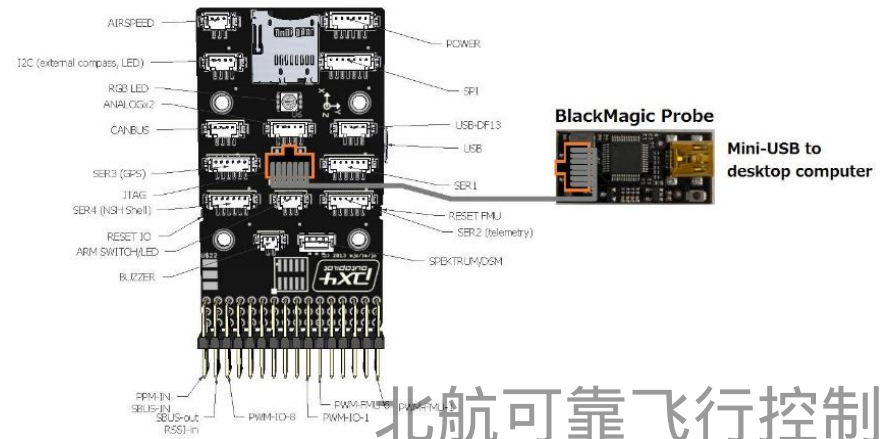
驱动初始化（mpu6000、mag、baro、GPS、mavlink、uORB、px4io）

Arducopter执行（初始化类、生成4个pthread、进入loop循环）

ArduPilot 调试开发步骤



调试手段	特点
Serial5	最方便、速率快、可调试nuttx、driver、copter、library等，最强大的工具：printf
JTAG	适合Nuttx系统都无法启动，调试最底层与功能无关的代码
GDB	适合Nuttx系统都无法启动，调试最底层与功能无关的代码
MAVLINK	速率慢、上手容易，适合结合到地面站观看实时绘图
LOG	速率最快，无延迟，但是只能离线，适合系统辨识、调试控制
SITL	适合开发除导航控制之外的功能代码，不需要硬件

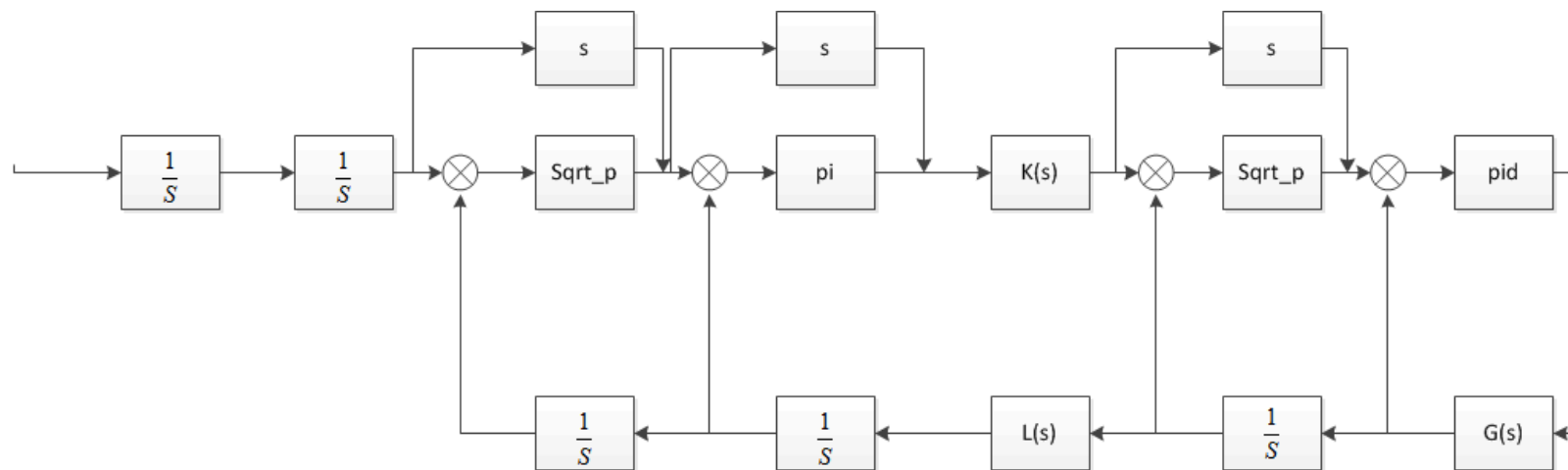


```
I- [KProcesses: 11 total, 3 running, 8 sleeping
CPU usage: 42.65% tasks, 0.52% sched, 56.84% idle
Uptime: 1438.134s total, 855.680s idle

PID COMMAND          CPU(ms) CPU(%) USED/STACK PRIO(BASE) STATE
0 Idle Task          855679 56.835  0/ 0 0 ( 0) READY
1 hpwork             31913  2.321 724/ 1592 192 (192) w:sig
2 lpwork             4998  0.343 356/ 1592 50 ( 50) w:sig
3 init              1360  0.000 1036/ 2496 100 (100) w:sem
44 <pthread>         50132  3.525 680/ 2040 181 (181) w:sem
10 px4io             14473  0.9    14473 0.945 908/ 1496 240 (240) w:sem
43 ArduCopter        404240 28.374 1268/ 8184 180 (180) w:sem
45 <pthread>         54993  3.869 612/ 2040 60 ( 60) w:sem
46 <pthread>         16711  1.117 524/ 2040 58 ( 58) w:sem
47 <pthread>          3209  0.171 444/ 1016 59 ( 59) w:sem
48 top              118  1.977 1236/ 1596 100 (100) RUN
```

导航控制制导过程分析

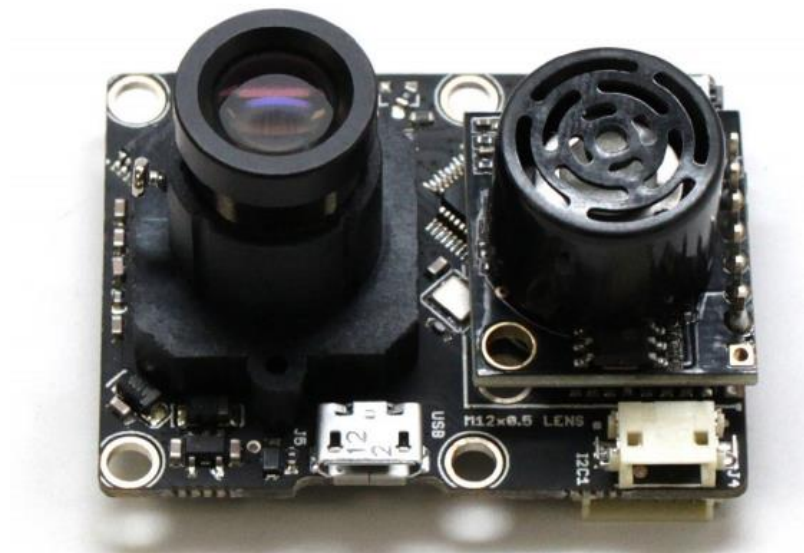
- 导航部分采用EKF，这部分代码实现是matlab代码优化之后生成的
- 控制部分采用级联P控制
 - 角速度环和速度环因为存在干扰，所以有I项
 - 角度环和位置环因为不存在干扰，所以只有P控制
 - 问题是整个控制效果严重依赖角速度环的执行效果，而实际飞行中角速度环控制效果很差
- 制导部分采用L1控制轨迹



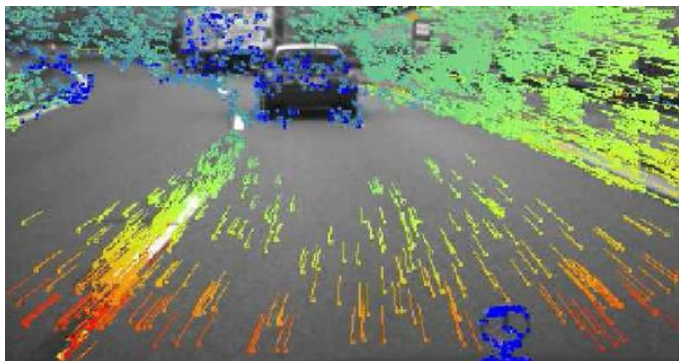
apm loiter模式下的控制结构

PX4Flow工程解析及图像定位

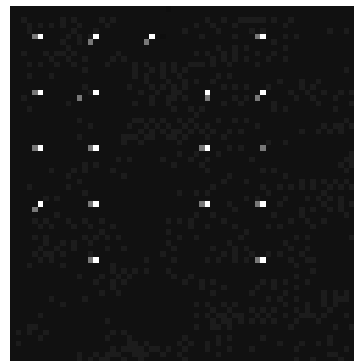
- 算法简介
- 性能分析
- 改进办法



PX4Flow工程解析及图像定位



理想中的光流：
像素足够多
计算资源足够多
纹理丰富



现实中的光流：
像素少 (64×64)
计算资源紧张 (180mhz)
纹理不清楚

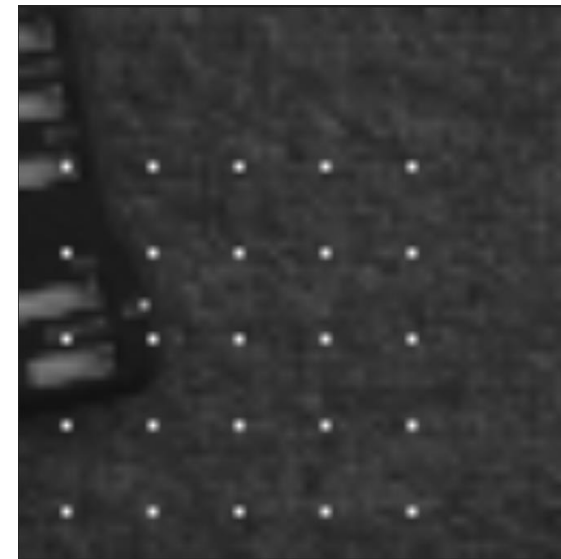
PX4Flow工程解析及图像定位



Mt9v034拍摄画面

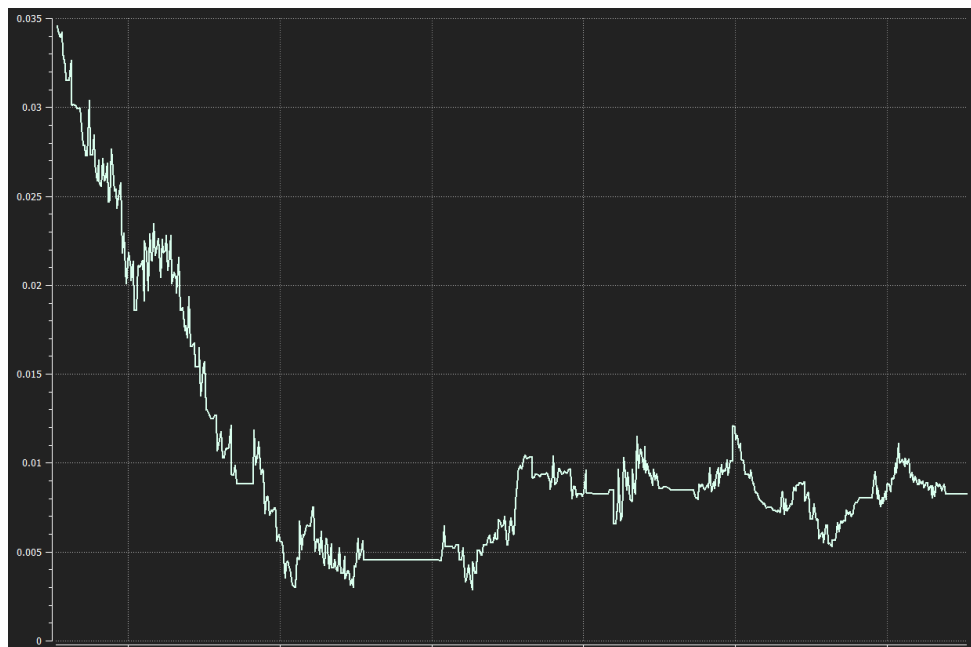


运行光流时拍摄画面

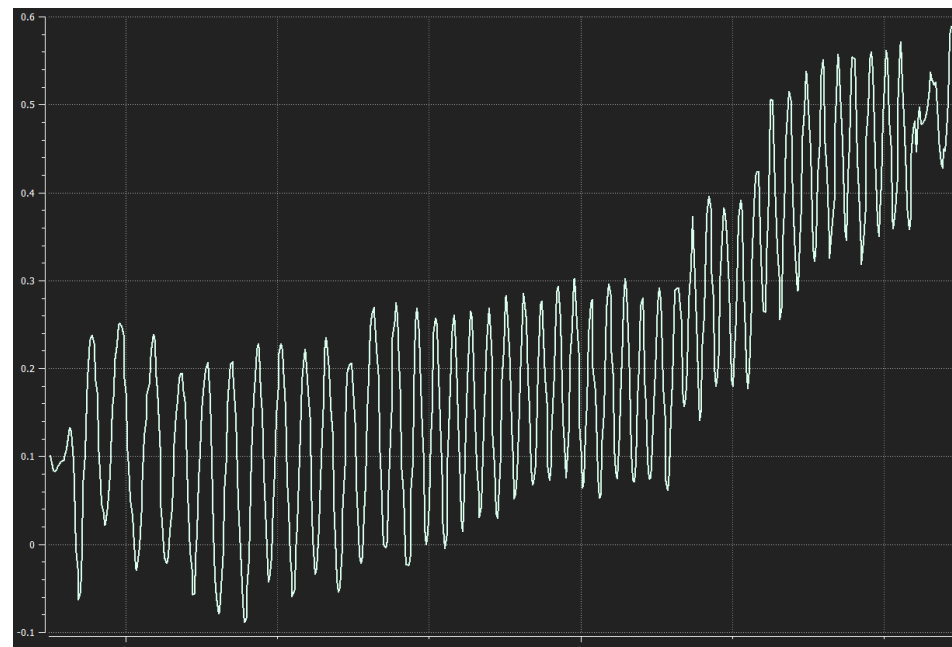


运行光流时的角点（白色点）

PX4Flow工程解析及图像定位



导航系下位置不变，姿态变化时，px4flow输出的位置（纵坐标：m），当摆动幅度小于一个阈值时漂移现象几乎消失



导航下姿态不变，位置围绕一个点水平来回运动时，px4flow输出的位置（纵坐标：m），存在明显的漂移现象

PX4Flow工程解析及图像定位

存在的问题	改进的办法
长焦镜头，对高度敏感	改为短焦镜头
视场角小的同时，算法人为裁减图像大小	不裁减图像大小，改算法减小计算量
参数固定，不能适应环境变化	改为参数自适应
对计算结果求均值	动态改变权重系数求均值
Quality指标不能代表真实结果好坏	更改Quality指标，参与上层EKF运算
Vel范围过小	改进算法、提高Vel范围
长时间存在漂移	引入特征，向上输出lpos

$$vel = \frac{pixel_size \times binning \times \Delta \times height \times freq}{focal_length}$$

提高最大分辨速度：提高搜索范围
减小最小分辨速度：减小搜索频率

光流速度计算公式

关于作者

- 来延涛，计算机专业硕士，通讯专业本科
- 感兴趣的方向：深度学习、机器学习、计算机视觉、智能控制
- 邮箱：laiyantaotao@126.com

谢谢