AN AUTONOMOUS DRONE PLATFORM FOR STUDENT RESEARCH PROJECTS

H. David Mathias
Department of Mathematics and Computer Science
Florida Southern College
Lakeland, FL 33801
863 680 6283
hmathias@flsouthern.edu

ABSTRACT

This paper describes a low-cost Micro Aerial Vehicle, capable of fully autonomous flight, that is well-suited as a platform for significant student research projects in a variety of areas. Autonomy is achieved via incorporation of a single board computer running Linux. While already very capable, this platform is further extensible with additional sensors such as laser rangefinders and infrared detectors. An available API aids in creation of software for vehicle control. Though we briefly outline a research project in which we used this vehicle, our focus here is on setting up the drone for use in research rather than on subsequent work.

INTRODUCTION

There can be little doubt that Micro Aerial Vehicles (MAVs), popularly known as drones, have become the subject of intense interest in many fields. Businesses large and small have identified potential commercial uses. Smaller businesses photograph homes for realtors [22] and survey courses [23] for golf clubs. In the energy sector, MAVs inspect power lines and towers [24], taking over, in part, a dangerous and time-consuming activity. In agriculture [21], drones provide farmers with views that are otherwise difficult to obtain, enabling earlier identification of citrus greening or crops that have been over or under watered. Only several days ago, at the time of this writing, Amazon released a video demonstrating the concept for its much anticipated drone delivery program. Beating Amazon to the punch, Swiss Post experimented with drone delivery in a pilot program during Summer 2015 [17].

Researchers in many areas of computer science have been similarly enamored. Path planning [11], mapping [15], learning [9], cooperation [8], computer vision [16], and more have been the subject of significant work. News stories, quite often detailing irresponsible drone use, have ensured that the technology is in the popular consciousness as well. As drone sales continue to increase, we can be sure that many of our students will have used a drone and more will be intrigued by them. As educators, we should leverage student interest in MAVs by seizing the opportunity to use them as a means to teach valuable concepts and skills.

We present a relatively inexpensive but extremely capable micro aerial vehicle into which we have integrated a single-board computer making the vehicle capable of fully autonomous flight. Extensible hardware, open source firmware, a well-supported API and an active user community help ensure that the MAV, a 3DR Iris+ quadcopter with an Odroid XU4 running Ubuntu Linux 15.04, is appropriate for significant student research projects on a variety of topics. Given that undergraduate research is among the highest of high impact practices [18], any platform that provides research opportunities, particularly one that excites students' imaginations, is compelling.

PREVIOUS WORK

Previous work with MAVs is extensive and widely varied. Restricting our focus to descriptions of flight platforms used for academic research narrows the field considerably. Eriksen, *et al.* [7] describe using a Parrot AR Drone as a research vehicle and highlight a project in which the drone is tasked with escaping from a room populated with visual markers. Engel, *et al.* [6] also use an AR Drone for camera-based navigation. While the AR Drone is inexpensive, it is less capable than the Iris+ with considerably less onboard processing.

At the other end of the spectrum, Lupashin, *et al.* [10] describe the Flying Machine Arena, a dedicated 10m x 10m x 10m indoor space populated with an array of motion sensing cameras constituting a positioning system capable of millimeter accuracy. While excellent, each of the AscTec Hummingbird drones they use approaches $5000, roughly six times the cost of the vehicle we describe here.

A STUDENT RESEARCH PLATFORM

Robotic vehicles provide many interesting and accessible research questions for undergraduate students. Investigating these problems requires a reliable, extensible, programmable vehicle with which to work. With the vehicle described here, problems in the research areas mentioned above – pathfinding, computer vision (with addition of a camera), mapping, cooperation, and others – are accessible to undergraduate students.

The 3DR Iris+, shown here with two-axis gimbal and GoPro camera. Photo credit: 3DR.

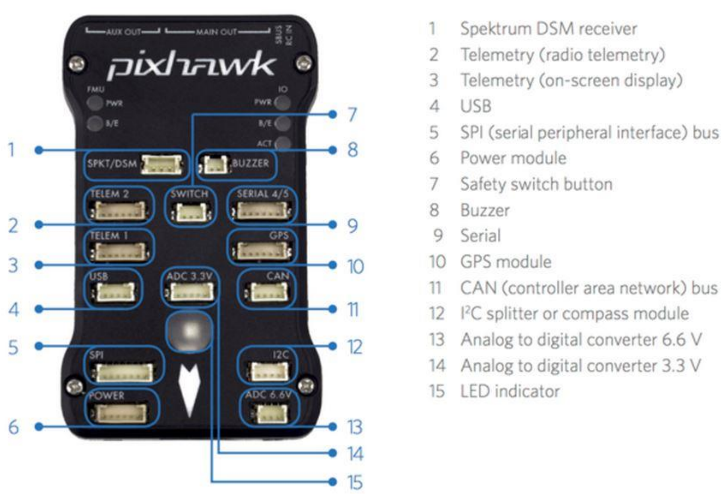
The Vehicle

The Iris+, introduced by 3DRobotics in Fall 2014, is a 550mm class (measured motor-to-motor) ready-to-fly quadcopter [1]. It is powered by a 5100mAh battery driving 4 950kV motors through a 4-in-1 electronic speed controller. It includes a Pixhawk flight control board (described below), an internal uBlox GPS with integrated 3-axis compass for navigation, and a 915MHz radio (or 433MHz, depending on country) for communication with a ground control station. Also included is a full-featured remote control transmitter with a preconfigured receiver onboard the drone. The Iris+ has an advertised flight time of 16-22 mins. As of January 2016, the Iris+ retails for $600, very reasonable for a vehicle with its capabilities.

Component	Cost (US $)
3DR Iris+ with remote control and telemetry radio	600
Odroid XU4 with 32GB eMMC, wifi dongle, and case	153
UBEC	15
USB to TTL serial converter	10
Miscellaneous cables	15
Total	793

Table 1: Component costs of the described vehicle.

Control of the Iris+ is facilitated via the Pixhawk flight control board [12]. Developed by researchers at ETH Zurich, the Pixhawk is open source and very widely adopted. It is suitable for use in copters, planes and ground-based rovers. Connectivity options are numerous, allowing for use of a wide array of external devices such as GPS, range finders and companion computers. Internally, the unit integrates a three-axis accelerometer, a three-axis gyroscope, a three-axis compass and a barometer. These sensors allow for determination of motion, orientation, heading and relative altitude, respectively.



The Pixhawk flight control board. Numerous ports allow for connection of a multitude of external sensors and devices. Photo credit: 3DR.

Software

In this section we discuss the third-party open-source software that is integral to the vehicle. It is distinct from that we have developed as part of subsequent research projects. Firmware for the Pixhawk comes in the form of two distinct, but cooperating, open source efforts: APM [3] and PX4 [14]. Both *flight stacks* have robust and supportive development communities and are updated frequently. MAVLink [13], a message-based communication protocol, underlies both efforts. In our vehicles, we use the APM flight stack.

MAVLink provides messages that allow for accessing and changing vehicle parameter values, checking vehicle status and navigation, changing, for example, vehicle position, attitude and velocity. The protocol is extensible, allowing users to define new messages for their purposes. However, the provided message set is sufficiently robust that we have not had the need to do this.

In Spring 2015, 3DR introduced DroneKit-Python, a Python API for developing MAV applications [2]. DroneKit frees researchers and developers from many of the low-level aspects of MAVLink, providing a high-level interface for connecting to, monitoring and controlling a vehicle. We currently use DroneKit Python 2.0, an update released in November 2015.

Vehicle configuration and sensor calibration are easily facilitated by ground control station software (GCS). Mission Planner (Windows only) and APM Planner (multi-platform) are both freely available. We use Mission Planner due to its ease of installation.

```
def fly_waypoint(self, waypt, index):
    print "Flying to waypoint %s..." % index
    # create the waypoint in global_relative_frame
    dest_point = LocationGlobalRelative(waypt[0], waypt[1], self.alt_0)
    # fly to the waypoint at specified velocity
    self.vehicle.simple_goto(dest_point, groundspeed=self.v_base)

    # calculate current location and distance to waypoint so
    # that we can determine when we have gotten close enough
    current_loc = self.vehicle.location.global_relative_frame
    dist_lat, dist_lon, dist = self.distance_meters(current_loc, dest_point)
    while dist > self.dist_threshold:
        time.sleep(0.25)
        current_loc = self.vehicle.location.global_relative_frame
        dist_lat, dist_lon, dist = self.distance_meters(current_loc, dest_point)

    print "    At waypoint %s..." % index
```

A function, using DroneKit Python, for flying to a waypoint specified by GPS coordinates.

An Onboard Computer

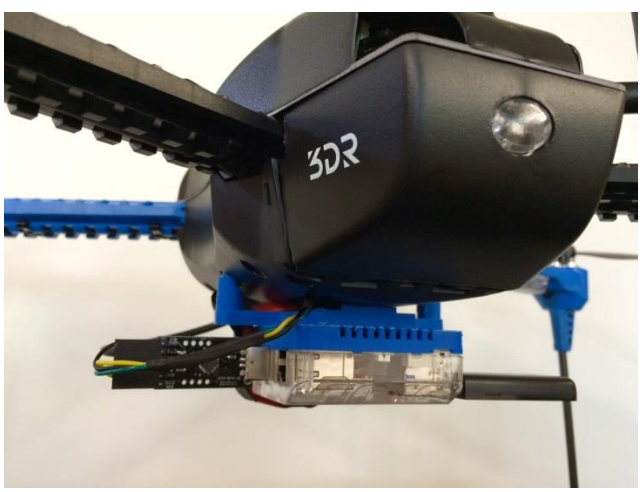
Applications to control the vehicle do not run on the Pixhawk but on a companion computer. The most straightforward implementation uses a ground-based laptop, connected to the vehicle via radio, for computationally intensive tasks. However, simplicity comes with a cost: the radio connection is low bandwidth and prone to noise. The low speed connection, 57600 baud, makes this option unsuitable for applications that require image processing, or other communication intensive computation, while noise can be problematic for mission-critical control messages.

An alternative, better suited to the task, is to incorporate a single-board computer into the drone. The serial communication channel increases bandwidth to 1.5M baud and eliminates noise. And, of course, the MAV is fully autonomous during flight, a desirable feature in applications in which communication might be difficult.

Several single board computers are appropriate for use as an onboard companion computer. We chose the Odroid XU4, due to its powerful heterogeneous 2.0 GHz ARM processor and 2 GB of RAM. As a boot drive it can use either a MicroSD card or, for increased speed, an eMMC 5.0 module. With a 32GB eMMC, wifi module, and case, the XU4 costs approximately $150.

The XU4 is capable of running Linux and Android. We use Linux 15.04 as on the ground-based laptop we used prior to adopting the Odroid. Porting our application and supporting utilities was seamless.

Incorporating the XU4 involves several steps that are detailed by APM developers[4]. First, one must mount it. There is very little space within the shell of the Iris+ so mounting on the exterior is required. Establishing communication with the Pixhawk requires a cable or interface to convert USB on the Odroid to 3.3V TTL serial for the Pixhawk. With this in place, creating a connection is straightforward. Next, it is necessary to supply power from the vehicle battery to the Odroid, requiring use of a universal battery elimination circuit (UBEC) to convert battery voltage to the 4 amps at 5 volts needed by the computer. Some soldering and drilling the drone shell are necessary.



The mounted Odroid XU4. Serial connection (left) and wifi dongle (right) are visible.

In the lab, one can easily connect a keyboard, mouse and monitor to the Odroid to facilitate logging in and launching applications. In the field, when flying, this is not a viable option. It is possible to have the application launch on startup, however, because the software will, presumably, fly the drone, this is a less safe procedure. It will also require restarting the computer for each subsequent flight.

Instead, we chose to establish a wifi connection to the Odroid. We can then use ssh from a ground-based laptop to login to the onboard computer to launch our software. As an additional benefit, this also provides a channel for communicating system status and progress to the ground.

An inexpensive USB wifi dongle is the only hardware required to enable this option. Setup requires establishing a wifi access point on the Odroid. Again, APM developers have detailed the procedure [5]. Joining this network makes it possible to login to the onboard computer. Not only does this allow starting applications to initiate flight but also editing software in the field to make adjustments.

RESULTS

The focus in this paper is creation of the augmented Iris+ flight platform. Of course, the purpose of developing that platform is to use it subsequently for research projects. The first such project we have undertaken using this vehicle involves utilizing genetic algorithms to plan a path, through known obstacles [11]. In addition to other contributions, our work appears to be the first in which results are tested with actual flight rather than simulation.

The path planning application runs noticeably more slowly on the Odroid than on a ground-based laptop but as this processing occurs prior to takeoff, any delay is not operationally significant. The increased reliability of the improved communication between computer and Pixhawk outweighs the increase in preprocessing time. It is worth noting that genetic algorithm pathfinding is distinct from developing the flight platform and is not required for flying the vehicle. It provides one method of navigation. Other projects will control the drone very differently.

Flight results with our first project have been very positive. Even with the attached XU4 and its accessories, the Iris+ is stable and responsive. We used the same vehicle for numerous flights prior to incorporating the onboard computer. While these missions were largely successful, we sometimes experienced communication problems likely attributable to a noisy radio connection or low bandwidth. With the Odroid incorporated, communication with the controlling computer is via a fast and reliable serial interface.

CONCLUSIONS

For a total cost of less than $800 (see Table 1), we have constructed a vehicle capable of serving as the testbed for significant research projects that provide students with valuable, formative experiences. The Iris+, with its Pixhawk flight controller, is extensible and easily modified as well as highly programmable. The Odroid XU4 provides sufficient power to make it viable for running the CPU intensive applications, such as image processing, that many projects will require. We have plans for further projects that include computer vision and integrating a recently mounted Lidar-Lite laser range finder into a new, multi-stage navigation system.

Setting up this MAV, as outlined here, could make an interesting project for one or two students, however, in our case, students were only tangentially involved. Instead, several students are very significantly involved in undergraduate research using this drone, and others, as testbeds for research projects involving software development for problems such as those listed in an earlier section of this paper. One such project is mentioned above and others are just beginning.

We think it important to mention that there are many considerations when using aerial vehicles with students. Primary among these is safety, which requires establishing and adhering to a flight protocol. Other considerations include institutional policies, insurance, and Federal Aviation Administration policies. For example, vehicles used for research cannot be registered via the FAA website. Instead, registration is via the traditional, and more complex, paper forms [19]. For those for whom flying drones is not practical or possible, most of the techniques and some of the hardware used in this work (such as the Pixhawk and Odroid) can also be used in a ground-based vehicle.

ACKNOWLEDGEMENTS
The author thanks 3DR for providing photographs of the Iris+ and Pixhawk.

REFERENCES

[1] 3D Robotics, Iris+, 2014, 3drobotics.com/iris-plus, retrieved December 13, 2015.
[2] 3D Robotics, DroneKit-Python Documentation, 2015, python.dronekit.io, retrieved December 13, 2015.
[3] Ardupilot Developers, Ardupilot Development Site, 2015, dev.ardupilot.com, retrieved December 13, 2015.
[4] Ardupilot Developers, Communicating with Odroid via MAVLink, 2014, dev.ardupilot.com/wiki/odroid-via-mavlink, retrieved December 13, 2015.
[5] Ardupilot Developers, Odroid Wifi Access Point for Sharing Files via Samba, 2014, dev.ardupilot.com/wiki/odroid-wifi-access-point-for-sharing-files-via-samba, retrieved December 13, 2015.
[6] Engel, J., Sturm, J., Cremers, D., Camera-based navigation of a low-cost quadrocopter, *Proceedings of the International Conference on Intelligent Robot Systems*, October 2012.
[7] Eriksen, C., Ming, K., Dodds, Z., Accessible aerial robotics, *Journal of Computing Sciences in Colleges*, 29 (4), 218-227, 2014.
[8] Lindsey, Q., Mellinger, D., Kumar, V., Construction with quadrotor teams, *Autonomous Robots*, 33, 323-336, 2012.

[9] Lupashin, S., Schölling, A., Sherback, M., D'Andrea, R., A simple learning strategy for high-speed quadrocopter multi-flips, *Proceedings of the IEEE International Conference on Robotics and Automation*, 1642-1648, May 2010.
[10] Lupashin, S., Hehn, M., Müller, M, Schölling, A., Sherback, M., D'Andrea, R., A platform for aerial robotics research and demonstration: The Flying Machine Arena, *Mechatronics*, 24, 41-54, 2014.
[11] Mathias, D., Ragusa, V., Pathfinding via genetic algorithm in a fully autonomous MAV, Unpublished manuscript, 2016.
[12] Meier, L., Tanskanen, P., Heng, L., Lee, G., Fraundorfer, F., Pollefeys, M., PIXHAWK: A micro aerial vehicle design for autonomous flight using onboard computer vision, *Autonomous Robots*, 33, 21-39, 2012.
[13] Meier, L., MAVLink Common Message Set, 2015, pixhawk.ethz.ch/mavlink, retrieved December 13, 2015.
[14] Pixhawk.org, PX4 Developer Documentation, 2015, pixhawk.org/dev/start, retrieved December 13, 2015.
[15] Shen, S., Michael, N., Kumar, V., Autonomous indoor 3D exploration with a micro-aerial vehicle, *Proceedings of the IEEE International Conference on Robotics and Automation*, May 2012.
[16] Shen, S., Mulgoankar, Y., Michael, N., Kumar, V., Vision-based state estimation for autonomous rotorcraft in complex environments, *Proceedings of Robotics: Science and Systems*, June 2013.
[17] Agence France-Presse, Switzerland begins postal delivery by drone, July 7, 2015, www.theguardian.com/technology/2015/jul/08/swiss-post-begins-testing-postal-delivery-by-unmanned-drone, retrieved January 17, 2016.
[18] Brownell, J. and Swaner, L., High-impact practices: Applying the learning outcomes literature to the development of successful campus practices. Peer Review, Association of American Colleges & Universities, Spring 2009.
[19] Personal communication with the Federal Aviation Administration. December 2015.
[20] Michael Oborne, Mission Planner Home, planner.ardupilot.com, retrieved January 17, 2016.
[21] Andrew Amato, AGCO uses 3DR Solo as new field mapping drone, August 2015, dronelife.com/2015/08/11/agco-uses-3dr-solo-for-new-field-mapping-drone, retrieved January 17, 2016.
[22] National Association of Realtors, Drones – Frequently Asked Questions, www.realtor.org/law-and-ethics/drones-frequently-asked-questions, retrieved January 17, 2016.
[23] Greensight Agronomics, Drone Golf Course Aerial Imaging Service, greensightag.com/surveys.pdf, retrieved January 17, 2016.
[24] The Morning Call, PPL gets approval to use drones in power line inspections, October 2015, www.mcall.com/news/local/mc-ppl-drones-20151019-story.html, retrieved January 17, 2016.