JOURNAL OF AEROSPACE INFORMATION SYSTEMS
Vol. 14, No. 5, May 2017

Dual Radio Autopilot System for Lightweight, Swarming Micro/Miniature Aerial Vehicles

Andrew Bingler* and Kamran Mohseni†
University of Florida, Gainesville, Florida 32611-6250

DOI: 10.2514/1.I010445

This paper presents the autonomous micro aerial vehicle pilot, a new autopilot platform weighing 6.25 g and measuring 11.3 cm^2, specifically designed for use on micro/miniature aerial vehicle mobile sensing platforms. An overview of the hardware, firmware, ground station, and validation testing used to demonstrate this autopilot as a viable research instrument for atmospheric thermodynamic sensing on micro/miniature aerial vehicles is presented. The autonomous micro aerial vehicle pilot incorporates a 16 bit 140 MHz processor, global positioning system, dual radios, inertial measurement unit, pressure sensor, humidity sensor, and temperature sensor. Through these components, the autonomous micro aerial vehicle pilot is capable of full-state feedback, vehicle state estimation, localization, and wireless networking. Notable features of this autopilot are its dual-radio configuration, providing redundancy and adaptability in communication, and the detachable sensor breakout design that allows for increased flexibility in sensor placement on a vehicle. Full-state feedback of the autopilot platform was validated through a series of bench tests. This includes a unique technique for dynamic inertial measurement unit validation performed using the present group's model positioning system as well as comparing estimated pressure values with known values at multiple heights and global positioning system values with a known path. Systemwide validation was performed through flight tests on a micro/miniature aerial vehicle airframe.

I. Introduction

MICRO/MINIATURE aerial vehicles (MAVs) are becoming increasingly popular in civil, military, and scientific fields. MAVs can be used for a variety of purposes, such as navigating compact spaces like those found in urban environments, search and rescue, and disaster assessment. As a result, research communities are constantly experimenting and pushing the bounds of MAV capabilities. Our research group is an example of such, having worked extensively with MAVs to develop cooperative control [1,2], multihop communications and sensor networking [3,4], dynamic data-driven application systems [5], and extreme weather monitoring applications [4].

Our group's previous autopilot, the Colorado PIC (CUPIC) [6], developed at the University of Colorado at Boulder and shown in Fig. 1, served as the core platform of our autonomous research activities until 2014. The CUPIC was designed, based on the capabilities available a decade ago, to be a simple autopilot equipped with a limited number of sensors while still allowing system observability for a MAV under severe weight, size, processing, and sensor capability restrictions [7]. In our group, the CUPIC was predominately flown on a 0.94-m span delta wing aircraft (shown in Fig. 2) and used to test swarm control [1] and sensor networking [3]. During the seven years since the CUPIC's last update, a number of technological advances have been made in the areas of embedded systems and microelectromechanical system (MEMS) sensors. Accordingly, we decided it was necessary to upgrade from the CUPIC's large body (6.4×2.9 cm with a mass of 19 g) and out-of-date hardware. In this process, we also achieved an equivalent full-state feedback capability for MAV control.

Currently, our group is working in scenarios such as plume detection [1,8,9], atmospheric flow sensing [10], and hurricane tracking [4]. These new research areas benefit most from the use of collaborative MAV swarms, which can enable sensor coverage over large airspaces. As such, the autopilots driving these MAVs should have the capabilities that most benefit a swarm. This includes very lightweight yet adept hardware that is capable of full-state feedback, allowing for state-feedback control (a capability unavailable in our CUPIC autopilot), along with a flexible sensor payload that is easily updated with new sensors. Furthermore, these environments may also require sensor coverage across kilometers of land, such as when tracking multiple plumes in an area. As such, a single swarm may be expected to split into multiple teams of vehicles, which may be separated from each other by significant distances. Thus, a highly customizable form of communication is required on each autopilot, so that communication within a team and within a swarm can be facilitated in a way that is both flexible and low-resource. This communication should allow for both redundancy and adaptability in bandwidth, so that potential radio failure can be accounted for and so that data payloads containing anything from three-digit humidity values to video feeds can be accommodated.

We propose a novel dual-radio autopilot, which allows for power-efficient communication to both near and far targets along with increased data bandwidths and built-in redundancy. This autopilot is also equipped with an easily upgradeable sensor breakout package to facilitate uncomplicated sensor swapping along with the hardware to enable full-state feedback, all in a package weighing slightly more than a U.S. quarter. In the following discussion, we compare capabilities of our system to other autopilots available.

The design of a MAV autopilot is commonly constrained by a number of specifications. Primarily, these autopilots are expected to be lightweight and small. The current MAV autopilot designs weigh no more than 40–50 g nor exceed 60 cm^2 (based on values from [11,12]).[‡,§] A suite of sensors capable of providing means for altitude estimation, attitude estimation, localization, and remote control are also expected.

When developing and researching a MAV autopilot, selecting the proper hardware is vital. Currently, there are already a number of commercial and open-source, off-the-shelf autopilot platforms available for use on low-weight MAVs. Commercial platforms can often provide capable hardware in a streamlined package. Micropilot's MP2128 (see footnote ‡) and Piccolo's Nano (see footnote §) are two common examples. Both

Received 24 December 2015; revision received 14 February 2017; accepted for publication 25 February 2017; published online 6 April 2017. Copyright © 2017 by the authors. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. All requests for copying and permission to reprint should be submitted to CCC at www.copyright.com; employ the ISSN 1940-3151 (print) or 2327-3097 (online) to initiate your request. See also AIAA Rights and Permissions www.aiaa.org/randp.

*Graduate Research Assistant, Department of Electrical and Computer Engineering, Institute for Networked Autonomous Systems.
†William P. Bushnell Endowed Chair, Department of Mechanical and Aerospace Engineering, Department of Electrical and Computer Engineering, Institute for Networked Autonomous Systems. Associate Fellow AIAA.

‡Data available online at http://store.micropilot.com/product-p/a-2128-helig2.htm [retrieved March 2017].
§Data available online at http://www.cloudcaptech.com/products/detail/piccolo-nano [retrieved March 2017].

Fig. 1 CUPIC autopilot system.

Fig. 2 Delta wing outfitted with an AMP board.

are lightweight (the MP2128 is 28 g, whereas the Nano is 29 g) and provide a processor, radio, and methods for attitude and altitude estimation. These autopilots are rigorously tested, straightforward to setup, and well suited for standard autopilot needs. However, because of commercial autopilot's proprietary nature, low-level modification can be quite difficult. The user may often have restricted access to the processor, obstructing software development. This is coupled with the problem that the majority of commercial autopilots run on closed hardware. Closed hardware is commonly difficult to modify and provides limited support for adding external components. These restrictions are less than ideal for research and development purposes.

Open-source autopilot platforms are less restrictive compared to their commercial counterparts, often providing easy access to both the hardware and software for the user to modify. The Lisa/S [11], one of the lightest and smallest autopilots available, is a good example. The Lisa/S contains an ARM processor, GPS, and state estimation hardware. It primarily runs the open-source paparazzi software [13] but is easily reprogrammed to run whatever the user chooses. Schematics are also provided for the Lisa/S, detailing the specifics of its open-source hardware to enable user modification. Another open-source option available is the well-rounded PixHawk development board [14]. The PixHawk runs on an STM processor and includes an MPU6000 accelerometer and gyroscope, along with the LSM303D accelerometer and magnetometer. The ArduPilot firmware [15] is commonly used to drive the PixHawk, but again, the user is capable of making any software or hardware changes they want, aided by the provided hardware schematics [16].

Despite their advantages, open-source autopilots are still imperfect. Open-source systems often do not include all the hardware required for a fully capable autopilot in mobile sensing applications. This can range from missing a GPS, an inertial measurement unit (IMU), or a radio. The UDB5 [17], a robust autopilot development platform, only comes equipped with a processor and IMU. The user must then add other components to the UDB5, such as a GPS or altimeter, if they wish to gain localization or altitude information.

Commercial and open-source autopilots are often quite adept and even capable enough for advanced autopilot needs on larger aircrafts. However, both types of autopilots still have challenges associated with their use on lightweight, swarm MAVs. Many of these autopilots are simply too heavy. The MP2128, the lightest commercial autopilot at 28 g, is prohibitive on a MAV airframe with a payload 25 g, such as the new generation of short wing aerodynamics modeling platform (SWAMP) MAVs developed and used by our group (see Fig. 3). Even when autopilots are appropriately lightweight, they may have had to make unhelpful sacrifices in the name of size, such as the previously mentioned 3.8 g Lisa/S, which is incapable of multinode networking and which precludes the inclusion of additional sensors.

Another common problem with these autopilots is that their sensors, like the IMU or altimeter, are adhered to a single relatively large board. The user may wish to place a certain sensor on another part of the vehicle for accuracy or flow access reasons. An example is that, although the main

Fig. 3 Group of SWAMP MAVs.

Table 1 Comparison of the AMP to a selection of other autopilots

Parameter	AMP	CUPIC	Lisa/S	UDB5	Pixhawk
Size, cm^2	11.3	18.2	6	25.8	40.8
Weight, g	6.25	19	3.8	11.7	38
Processor	140 MHz dsPIC33EP	40 MHz PIC18	72 MHz ARM Cortex-M3	80 MHz dsPIC33F	168 MHz STM32F427
Ram, KB	48	4	64	30	256
Flash	512 KB	128 KB	512 KB	256 KB	2 MB
Accelerometer	Three-axis MPU9250	— —	Three-axis MPU6000	Three-axis MPU6000	Three-axis LSM303D/ MPU6000
Gyroscope	Three-axis MPU9250	One-axis ADXRS Gyro	Three-axis MPU6000	Three-axis MPU6000	Three-axis L3GD20H/ MPU6000
Magnetometer	Three-axis MPU9250	— —	Three-axis HMC5883L	— —	Three-axis LSM303D
Altimeter	BMP180 barometer	— —	MS5611 barometer	— —	MS5611 barometer
GPS	Linx TM GNSS	Fastrax Itrax	U-Blox Max-7Q	— —	— —
Short-range radio	CYRF6936	— —	CYRF6936	— —	— —
Long-range radio	XBee Pro	XBee Pro	— —	— —	— —
Other sensors	SHT25 humidity sensor	— —	— —	— —	— —
Multinode networking	Yes	Yes	No	No	No
Sensor breakout	Yes	No	No	No	No

autopilot board may need to be kept in the fuselage, such as for our smaller SWAMP MAVs, the IMU would need to be adhered closer to the center of mass of the vehicle to aid in state calculations, which may be located outside of the fuselage.

Finally, commercial and open-source autopilots generally include either a single radio or require that the user attach a radio themselves. It is especially rare for any included radio to enable wireless networking (such as the XBee radio), which allows the autopilot to act as a node in a larger mesh network, facilitating interautopilot communication. For distributed MAV swarms, where redundant, flexible communication is critical, systems that seem to treat the radio as an afterthought can be nonconducive.

The autopilot presented in this paper is an attempt to address these issues. Designing our own autopilot allowed us to include all the functionality required to perform in swarm scenarios as well as meet the constraints of our group's smallest vehicle, the SWAMP MAV (shown in Fig. 3). Because of the small size of the SWAMP MAV, its payload is limited to a maximum of 25 g [including the battery, motor, servos, radio, and electronic speed controller (ESC)]. Because of this, the autopilot would need to be very lightweight. Hardware approaching this weight limit would not be acceptable because this would preclude the addition of extra components. The autopilot system would also need to be low power because energy storage is desired for propulsion. Sensor parsing, control, and telemetry are all computationally expensive operations; therefore, the autopilot would need to be powerful enough to handle these requirements, with processing power to spare for future functionality. In addition, this system must be able to function as a development tool, in order for new or upgraded capabilities to be tested without requiring a hardware revision.

With these specifications and motivations in mind, we created the autonomous MAV pilot (AMP). The AMP is a small, lightweight platform (11.3 cm^2 and 6.25 g) that is designed for research and development and provides extended capabilities in the forms of communication, sensor flexibility, and wireless networking. It includes two radios, the XBee and cypress radio frequency (CYRF) system on a chip, as well as the potential to add a third radio using the system's included development pins. This novel dual-radio configuration provides both redundancy and flexibility in communication. For instance, the XBee can be used for long-range, multinode networking, whereas the CYRF allows for low-power, point-to-point communication. Both radios can be used to broadcast to the same node simultaneously, increasing data bandwidth. The presented autopilot also divides its components between a large main board and a smaller sensor breakout, positioning environmental sensors on the breakout so as to ease sensor upgradability and increase sensor mounting options. Finally, the autopilot platform includes all the components necessary to enable full-state feedback "out-of-the-box", including a GPS, IMU, pressure, humidity, and temperature sensors.

A comparison of the autonomous micro aerial vehicle pilot (AMP) to other autopilots available on the market is presented in Table 1.

This manuscript is organized as follows. First, the specifications of the hardware used on the AMP are described in Sec. II. The firmware and code used on the AMP are described in Sec. III. This section discusses the architecture of the code as well as the drivers for the AMP's individual components and communication capabilities. Section IV describes the abilities of the ground station paired with the AMP board. Section V describes the calibration of the AMP's sensors and Sec. VI describes the tests used to validate the accuracy of the AMP's components. In Sec. VII, the results of flight testing using the AMP board are presented and discussed. Last, the conclusion is presented in Sec. VIII.

II. Hardware

The AMP's hardware (see Fig. 4) is designed to collect and parse data from a variety of sensors and receivers, make control decisions, and transmit all pertinent information to a ground station, all with computational power and functionality to spare for future capabilities. The AMP is composed of two boards: the main board and the sensor breakout board.

The AMP's main components include 1) the Microchip dsPIC33EP512GM306 digital signal controller (DSC), 2) the Bosch BMP180 pressure/temperature sensor, 3) the Sensirion SHT25 humidity/temperature sensor, 4) the InvenSense MPU9250 nine-axis IMU, 5) the Linx 10Hz TM global navigation satellite system (GNSS) GPS receiver [18], 6) the Digi Series 1 XBee-PRO radio, 7) the Cypress CYRF6936 radio, 8) an 8 MHz crystal, 9) a 12 MHz crystal, 10) a 3.3 V regulator, 11) a programming port, 12) a port to connect the sensor breakout board to the main board, 13) a development port with direct access to processor pins, and 14) ports for interfacing with external servos and motors.

These components will be described in more detail throughout the remainder of this section.

A. Main Board

The main board acts as the central hub for the AMP, connecting all the various elements of communication, sensing, and control together. The main board handles all incoming and outgoing signals and performs any required operations through the dsPIC processor.

The main board is 3.25×3 cm and includes the following components (see Figs. 5a and 5b): 1) Microchip dsPIC33EP512GM306, 2) 10 Hz Linx GPS, 3) Digi XBee radio, 4) Cypress CYRF6936 radio, 5) 8 MHz crystal, 6) 12 MHz crystal, 7) 3.3 V regulator, 8) programming port, 9) sensor breakout board port, 10) development port, 11) motor and servo ports, and 12) light-emitting diode (LED) indicators.

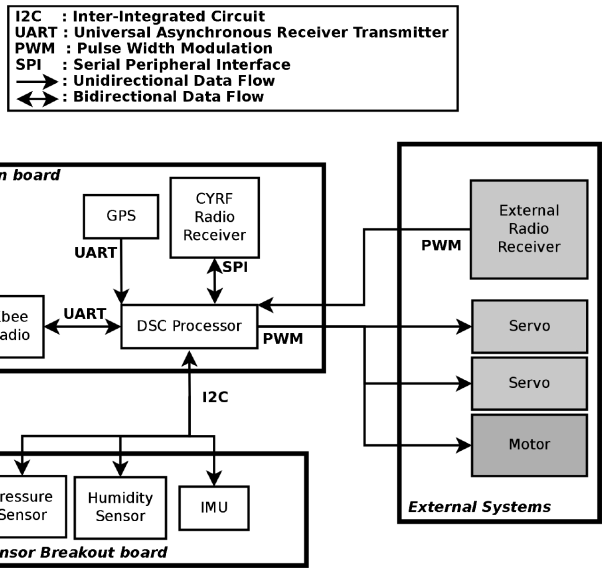
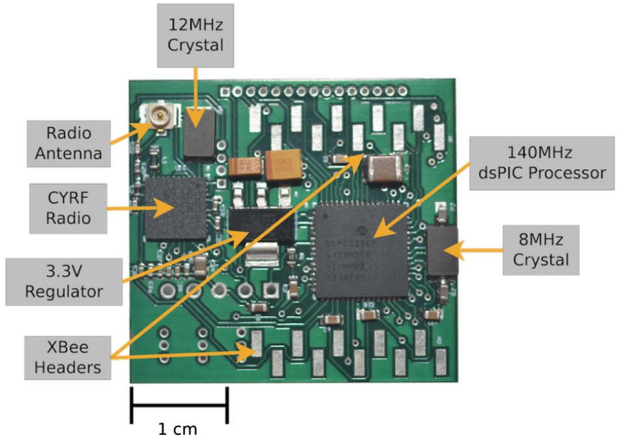
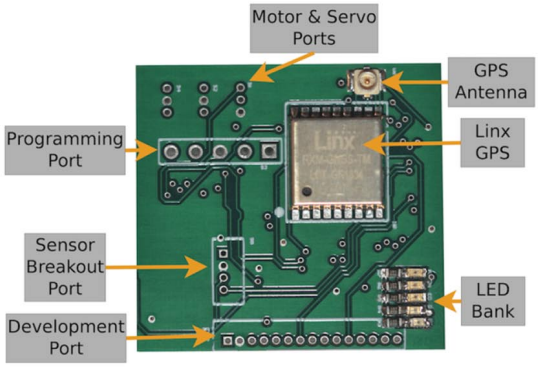


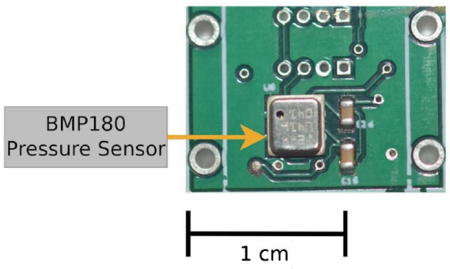
Fig. 4 Block diagram of the AMP system.



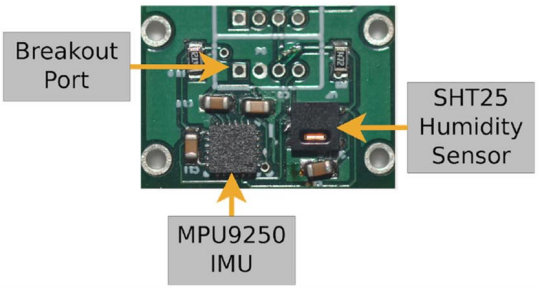
a) Bottom side of the main board. The processor, CYRF radio, radio antenna, XBee headers, voltage regulator, and crystal oscillators are indicated.



b) Top side of the main board. The GPS, GPS antenna, LEDs, and ports are indicated.



c) Bottom side of the sensor breakout board. The pressure sensor is indicated.



d) Top side of the sensor breakout board. The humidity sensor, IMU, and breakout port are indicated.

Fig. 5 Multiple views of the AMP board.

A development port is included in the main board, connecting a number of dsPIC peripherals to user accessible pins, as seen in Fig. 5b. This port allows the AMP to be used as a rapid development tool, providing access to the interintegrated circuit (I2C) and serial peripheral interface (SPI) communication, pulse-width modulation (PWM) output, or analog-to-digital conversion (ADC) features of the dsPIC. As a result, future components, such as new or upgraded sensors, can be added without requiring a new revision of the printed circuit board.

The main board also carries the responsibility of voltage regulation. The AMP can support a supply voltage range of 5–15 V. This source can be either from an attached ESC or through dedicated voltage pins.

B. Sensor Breakout Board

The sensor breakout board is designed to house the majority of MEMS sensors used by the AMP. The sensor breakout board is 1.4×1.1 cm and includes the following components (see Figs. 5c and 5d): 1) BMP180 pressure/temperature sensor, 2) SHT25 humidity/temperature sensor, and 3) MPU9250 nine-axis IMU.

Limiting the sensor placement to the sensor breakout board provides a number of benefits. Because the pressure and humidity sensors are sensitive to ambient temperature, separating them from the main board allows interference from major heat producing components (such as the 3.3 V regulator, XBee, or GPS) to be kept to a minimum. The small size of the sensor breakout also creates more options with regard to placement of the board on or within the body of a vehicle, as compared to the larger main board. Four mounting holes included on the sensor breakout (as seen in Figs. 5c and 5d) allow the board to be easily affixed to the vehicle. These features allow the sensor breakout board to be placed so that any environmental requirements for the included sensors can be met. For example, the SHT25 requires exposure to the airflow of which the humidity is to be tested. Thus, the sensor breakout board can be affixed to the exterior of a vehicle to meet this requirement. Further, having a sensor board separate from the main board simplifies future sensor expansion. Assuming any new sensors communicate over the I2C protocol, the sensor breakout board can be updated with a variety of sensor configurations without requiring any revisions to the main board.

C. Processor

The AMP's dsPIC33EP512GM306 processor, referred to as the dsPIC, was chosen after a careful review of processors available on the market. When first choosing a processor, the selection was narrowed down to the three categories commonly used in embedded systems: 1) microcontrollers, 2) digital signal processors (DSPs), and 3) digital signal controllers. The benefits and limitations of each category are discussed next.

Microcontrollers are processors outfitted with a wide variety of peripherals as well as sizable amounts of program memory and RAM. As such, microcontrollers are often treated as computers on a chip. Microcontrollers are used for a variety of functions, but most commonly for control applications. However, they often rely on hardware designed for simple mathematical calculations. Because of this, microcontrollers are not suited for a system expected to run computationally demanding programs, such as the AMP.

DSPs are processors with an architecture optimized for signal/data processing. DSPs are commonly used for applications such as image processing or audio compression and are designed to perform complex mathematical operations quickly and efficiently. Nonetheless, many DSPs can lack useful peripherals, such as PWM or ADC modules, and often incorporate less onboard memory compared to microcontrollers. Because of this, DSPs are usually considered unsuitable for control applications and thus were not chosen for use on the AMP.

DSCs combine aspects from both microcontrollers and DSPs. DSCs offer many of the control oriented aspects of microcontrollers, such as fast interrupt handling, a variety of peripherals, and a considerable supply of onboard memory. Similarly, DSCs also provide the data processing capabilities of a DSP through large accumulators and barrel shifters [19]. Because of this computational power and versatility, the DSC category of processor was chosen for use on the AMP.

From the available DSCs on the market, we chose Microchip's dsPIC line for its range of models, performance specifications, and strong documentation. From there, we selected the dsPIC33EP512GM306 for its large memory, processor speed, and available peripherals.

The dsPIC processor is highly capable despite not having the power of a full-fledged DSP. For example, the dsPIC is capable of processing vision requirements for applications such as determining optic flow or edge detection [20–22]. The processor can also handle a user's high-level filtering needs, such as for an extended Kalman filter with rates of 100 Hz and over [22–24]. With this processor, our group has had experience running a feedforward, lead, proportional-integral, and two proportional-derivative controllers in parallel at 100 Hz without encroaching on processor time needed for other functions.

Specifications for the dsPIC are listed in Table 2.

D. Inertial Measurement Unit

The IMU is another important component to consider when building an autopilot. The IMU is used to calculate the roll, pitch, and yaw of the vehicle, allowing the control system to determine orientation in reference to a fixed earth frame. A nine-axis IMU was preferred because it allows for more accurate flight dynamics. Because of its proficient specifications, Invensense's MPU9250 was chosen. The MPU9250 is a nine-axis IMU, composed of a three-axis gyroscope, accelerometer, and magnetometer in a small $3 \times 3 \times 1$ mm package.

E. Global Positioning System

GPS is used to provide geographic coordinates for the AMP, via latitude and longitude information. Chosen for its low current draw of 20 mA, the AMP uses the Linx TM GNSS as its GPS receiver. Horizontal accuracy for the Linx is listed as ± 2.5 m, whereas vertical accuracy can be anywhere from ± 4 to ± 10 m.

F. XBee Radio

For communication with the AMP, Digi's XBee radio was chosen, specifically the Series 1 PRO model. This radio features a line-of-sight range of up to 1.5 km and communicates over a universal asynchronous receiver/transmitter (UART) connection. One of the primary benefits of using XBees is their support for wireless networking between multiple radios.

Table 2 dsPIC specifications

Parameter	Value
Operating voltage	3.3 V
Number of pins	64
Architecture	16 bit
Flash memory	512 KB
RAM	48 KB
CPU speed	140 MHz
PWM outputs	12
Timers	21
UART modules	4
SPI modules	3
I2C modules	2

G. CYRF6936 Radio

For communicating with and operating the vehicle, we included Cypress' 2.4 GHz CYRF6936 radio. This radio features a wireless range of 100 m and an operating current of only 21 mA.

H. Pressure Sensor

The BMP180 is a digital, atmospheric pressure sensor. The BMP180 was chosen because it is capable of providing altitude measurements accurate up to ± 1 m with a resolution of 17 cm. The BMP180 also includes a temperature sensor used for pressure calibration.

I. Humidity Sensor

The SHT25 digital humidity sensor is also included on the AMP. The SHT25 is capable of providing humidity information with an accuracy of $\pm 1.8\%$ relative humidity (RH) along with a current consumption of only 30 μA. A temperature sensor is also included within the SHT25 package for humidity calibration.

III. Firmware

The AMP is programmed in C, using Microchip's MPLAB integrated development environment [25]. Microchip's PICKIT 3 [26] and ICD 3 [27] programmers are used to program the dsPIC.

A. Architecture

The firmware architecture of the AMP is designed to treat each sensor as independently as possible, in an effort to keep the system modular. Inputs and outputs of components are handled using function calls, keeping code overlap to a minimum. Because of this, the firmware is easily revised as hardware changes are made.

B. Component Drivers

The Linx GPS communicates over UART at 57,600 bps, or 57,600 baud, and pushes GPS messages at a rate of 10 Hz. When a message from the GPS arrives, an interrupt is triggered, and the message is parsed. A checksum is appended to the end of each GPS message, which is used for detecting any errors that may have been introduced to the message while it was transmitted.

The XBee on the AMP is used for downlinking data and receiving commands and is paired with another XBee connected to the ground station. The XBee communicates via UART at 115,200 baud and functions in either one of two modes: transparent or application programming interface (API) mode. In transparent mode, anything sent to one XBee is immediately transmitted to the paired XBee. In API mode, a metadata frame must be built around the data to be sent and a checksum calculated and appended; otherwise, the message will be rejected by the XBee and discarded. The XBee and its API mode will be discussed further in Sec. III.C.

The MPU9250, BMP180, and SHT25 all communicate over the same I2C peripheral. Commands to start desired measurements are transmitted to the corresponding component. Once the measurement is completed, an interrupt is triggered, and the values from the component are transmitted back to the processor.

C. Communication

The AMP system generates more than 30 variables that need to be saved and tracked. These include vital autopilot variables that fall into one of three categories: 1) global position (latitude and longitude and course over ground); 2) local position (tri-axis accelerometer, tri-axis gyroscope, tri-axis magnetometer, altitude, and yaw, pitch, and roll); and 3) control (control surface deflections and timestamp).

Currently, three types of telemetry packets are transmitted to the ground station: a fast packet, an intermediate packet, and a slow packet. The fast packet contains items updated frequently, such as IMU or control surface deflection values. The intermediate packet contains items updated at a slower frequency, such as altitude or GPS values. Last, slow packets contains items that are considered nonvital, such as tuning gains. Fast packets are transmitted at a frequency of 140 Hz, intermediate packets transmit at 20 Hz, and slow packets transmit at 10 Hz.

These packets are transmitted through an XBee configured to use API mode. The XBee API mode requires that a message frame around the data payload be built. After a message frame is transmitted and received by an XBee, the frame's appended checksum is examined to ensure that the message was not corrupted during transmission.

One of the most useful aspects of the XBee API mode is the potential for wireless networking [28]. By assigning addresses to multiple XBees in API mode, a complex network of autopilots can be created. In this network, XBees can specify recipients for their messages, either individuals or groups. Intermediate XBees can then be used to relay messages through multihopping, in case the destination is out of range of the source. Previous work with XBee wireless networking can be seen in [3].

IV. Ground Station

A ground station is an important part of any autopilot system, especially for the development of new and untested autopilot controllers. A ground station's design must focus on its ability to display any and all information that is relevant to the user, in a way that is quick, clear, and efficient. This is necessary because a number of flight scenarios require the ground station user to rapidly make decisions and draw conclusions based on the information displayed. A ground station should also allow for the uplink of data to the autopilot, such as control variables or flight coordinates, in a way that is intuitive for the user. Otherwise, configuring the autopilot and setting flight coordinates could be a complex and laborious process.

The ground station paired with the AMP consists of a Matlab program and an XBee Pro radio connected via USB. The various capabilities of the ground station are listed next.

A. Real-Time Data Display

During a flight, the ground station allows the user to display any variables sent down from the AMP in real time, as seen in Fig. 6. This includes the control and communication status of the vehicle (e.g., GPS lock) as well as any errors reported by the autopilot. Current GPS latitude and longitude are also displayed, with an optional Google maps overlay. Last, this display can be used to track multiple vehicles in the air, plotting separate vehicle variables on the same overlay.

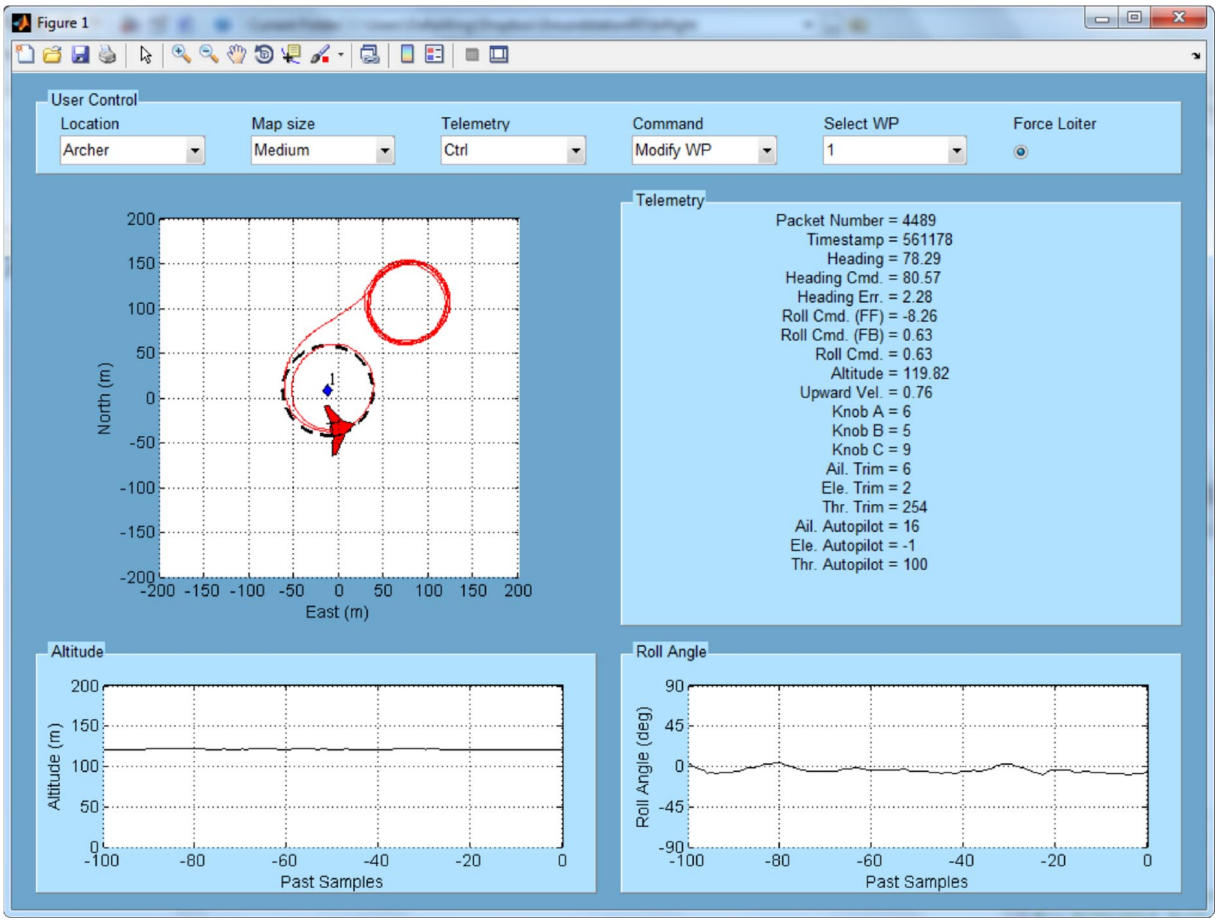


Fig. 6 Ground station GUI during a flight, currently displaying a map of the flight, telemetry received from the AMP, graphs for altitude and roll angle, and a user control section.

B. Control

The ground station allows the user to uplink a variety of control commands to the autopilot. This is especially useful when tuning a controller. In this way, the user may transmit a tuning value, observe the resulting control status variables, and then respond accordingly. Thus, the controller values for a new aircraft platform can quickly be determined, especially relevant for a multivehicle autopilot such as the AMP.

C. Waypoint Navigation

The ground station is capable of setting multiple waypoints for the autopilot, making it easy to create a flight plan.

D. Data Logging

All data received by the ground station are automatically logged in a binary file. The binary file can then be parsed and graphed for postflight study. This allows the user to, for example, pinpoint the moment of a flight failure and determine its cause.

E. Data Replay

Data logged by the ground station can also be replayed in real time, displaying the information as though the flight was taking place. This can be used to simplify the debugging of a controller or to aid in presenting a flight.

V. Sensor Calibration

Both our IMU and pressure sensor are calibrated to help ensure a stable flight. Pressure values are offset against ground pressure before a flight and are processed by a low-pass filter to help reduce noise. IMU values are individually calibrated and filtered before being used in attitude estimation.

Our magnetometer requires calibration to adjust for scale factor and any potential hard iron offsets caused by the environment using a method based on [29]. When starting this calibration, the magnetometer is rotated along all axes, and samples for the max and min values of each axis are saved. If the magnetometer is biased, then when these data are plotted, it will likely take the shape of an off-axis ellipsoid. The distances between the max and min values of each axis are then averaged, and this average value is used as the new length for each axis, forming a sphere. The center of this sphere is then shifted to the origin, correcting any offset error.

For our gyroscope and accelerometer, the plane is held flat and steady on startup so that any detected offsets can be corrected. Afterward, the data from each sensor are processed by an eighth-order, low-pass finite impulse response filter with a 10 Hz corner frequency to reduce noise.

Last, our IMU values are processed by a Mahoney filter, which is used for estimating attitude and bias [30]. This filter was preferred over an extended Kalman filter due to its low processing requirements because this is more suitable for MAVs with limited processing capabilities.

Table 3 DYNAMO system range of amplitudes for various degrees of freedom

Axis	Minimum amplitude	Maximum amplitude	Positioning resolution
Roll, deg	−180	+180	0.2
Yaw, deg	−180	+180	0.5
Pitch, deg	−45	+45	0.2
Plunge, cm	18	39	0.2

VI. Sensor Validation

To validate the accuracy of the AMP's components, each sensor was initially given individualized tests to assess the accuracy of its readings before an integrated test of the entire AMP system was performed. The components tested include the gyroscope, accelerometer, magnetometer, pressure sensor, humidity sensor, and GPS.

A. MPU9250 Gyroscope

A validation test of the MPU9250's gyroscope was conducted using our group's model positioning system (MPS), the DYNAMO [31]. The DYNAMO, designed and built in our group, is a precise positioning system that is able to maneuver the vehicle in four degrees of freedom (roll, pitch, yaw, and plunge) inside a wind tunnel. This equipment has been used to accurately measure stability derivatives and loading on our MAV (see [32,33]). The positioning abilities of the DYNAMO are displayed in Table 3 (plunge values are taken with respect to the test section floor assuming a model parallel with the floor), and a model of the DYNAMO can be seen in Fig. 7.

The AMP was mounted to the DYNAMO (see Fig. 8), where it was swept through oscillatory profiles of roll at an amplitude of ± 10 deg and frequency of 2 Hz (the DYNAMO itself has an angular velocity accuracy of ± 2.5 deg /s at 2 Hz). The gyroscope was set to a full-scale range of ± 1000 deg /s. The peak angular velocities of the gyroscope across 40 oscillations were then averaged, and a standard deviation was calculated. This test was repeated for each of the gyroscope's axes, and the results can be viewed in Table 4. These values are within the ± 5 deg /s tolerances listed for the MPU9250.

B. MPU9250 Accelerometer

A validation test of the MPU9250's accelerometer was conducted using our group's DYNAMO MPS to sweep the IMU through oscillatory profiles of plunge at an amplitude of ± 2.54 cm and frequency of 2 Hz (the DYNAMO itself has a plunge acceleration accuracy of ± 0.03 m/s^2 at 2 Hz). The accelerometer was set to a full-scale range of $\pm 8g$. The peak accelerations of the IMU across 40 oscillations and 320 samples were then averaged, and a standard deviation was calculated. This test was repeated for each of the accelerometer's axes, and the results can be viewed in Table 5. These values are within the ± 0.59 m/s^2 tolerances listed for the MPU9250.

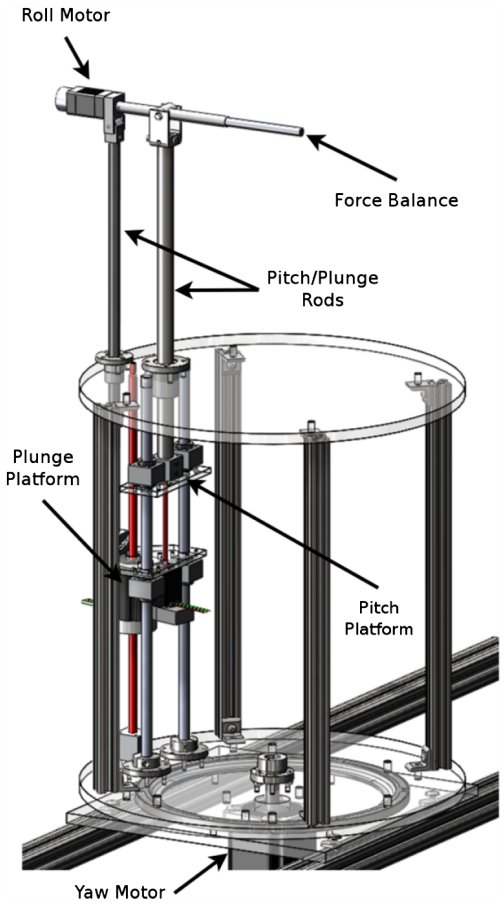


Fig. 7 Model of the DYNAMO MPS [31]. The roll motor, yaw motor, pitch and plunge rod, pitch and plunge platforms, and force balance are indicated.

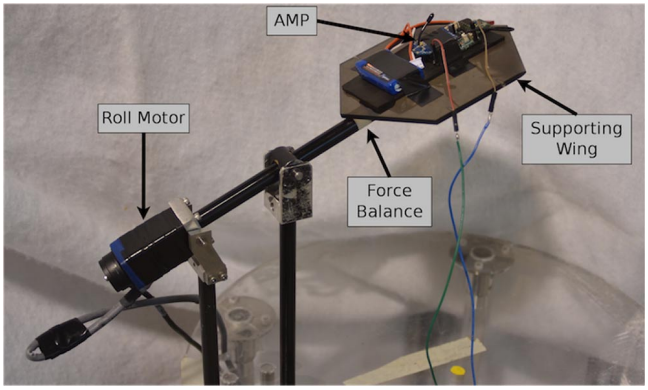


Fig. 8 AMP attached to the supporting wing of the force balance. The roll motor, AMP, supporting wing, and force balance are indicated.

C. MPU9250 Magnetometer

A validation test to determine the ability of the MPU9250's magnetometer to detect changes in angle was conducted using our group's DYNAMO MPS (itself possessing an angular position accuracy of ± 0.2 deg). The average maximum and minimum values of the three axes were first used to determine the range of the magnetic field; then, the IMU was swept through a 180 deg rotation over the length of 1 min while magnetometer values were recorded. The average and standard deviation of these values at select angles were then calculated. This test was repeated for each of the magnetometer's axes, and the results can be viewed in Table 6.

D. Linx Global Positioning System

A test was performed to validate the Linx GPS's latitude and longitude readings by carrying the GPS counterclockwise across the four sides of a rectangular parking garage. This test was repeated four times, and the sample data from each test were plotted with the path traveled, as shown in Fig. 9. Data were collected from the GPS at a rate of 10 Hz and were then compared to the latitude and longitude values of the path traveled using map data from Google. The average error was 1.79 m with a standard deviation of ± 1.35. These values are within the ± 2.5 m accuracy listed for the Linx [18]. GPS values also returned to within ± 2.5 m of the starting point after a full loop of the garage.

E. BMP180 Pressure Sensor

A validation test of the BMP180's pressure readings was performed by traversing five floors of a parking garage and collecting pressure data from the sensor on each floor for 30 s at a rate of 10 Hz. The data from each floor were then used to calculate a mean pressure and standard deviation. These values were compared against the more accurate WMR300, which lists an accuracy of ± 1 hPa. A table of the collected data can be seen in Table 7. All results were within the -4 to $+2$ hPa absolute accuracy listed for the BMP180.

F. SHT25 Humidity Sensor

To confirm the validity of the SHT25's humidity readings, the humidity values measured by the sensor were compared against three different known humidities. These humidities were created by following the procedure outlined in [34], where a known salt solution is mixed into distilled water inside a sealed container until saturated. The salt solutions used were lithium chloride, potassium chloride, and sodium chloride, each with accuracies of ± 0.31, ± 0.26, and $\pm 0.12\%$ RH, respectively. Each salt solution was allowed to settle for 24 h after mixing. Afterward, the humidities inside each container were measured by the sensor for 30 s at a sampling rate of 10 Hz. The sampled values for each test were then averaged, and a standard deviation was taken. The results of this test can be seen in Table 8. These results were within the $\pm 1.8\%$ RH accuracy listed for the sensor.

Table 4 MPU9250 gyroscope accuracy

Actual	Measured X	X standard deviation	Measured Y	Y standard deviation	Measured Z	Z standard deviation
125.7 deg/s	125.1 deg/s	± 3.7	124.0 deg/s	$\pm .6$	124.8 deg/s	± 2.5

Table 5 MPU9250 accelerometer accuracy

Actual	Measured X	X standard deviation	Measured Y	Y standard deviation	Measured Z	Z standard deviation
2.00 m/s^2	2.02 m/s^2	± 0.04	2.08 m/s^2	± 0.06	1.99 m/s^2	± 0.06

Table 6 MPU9250 magnetometer accuracy

Angle, deg	Actual, mg	Measured X, mg	X standard deviation	Measured Y, mg	Y standard deviation	Measured Z, mg	Z standard deviation
0	0	4.09	± 7.8	5.15	± 7.3	5.27	± 7.5
45	122.5	122.2	± 8.9	127.88	± 8.4	131.48	± 7.8
90	245	246.65	± 7.6	241.83	± 7.3	241.61	± 7.1
135	122.5	123.67	± 8.2	125.32	± 8.5	119.66	± 8.1
180	0	6.39	± 7.8	2.12	± 7.0	0.71	± 7.2

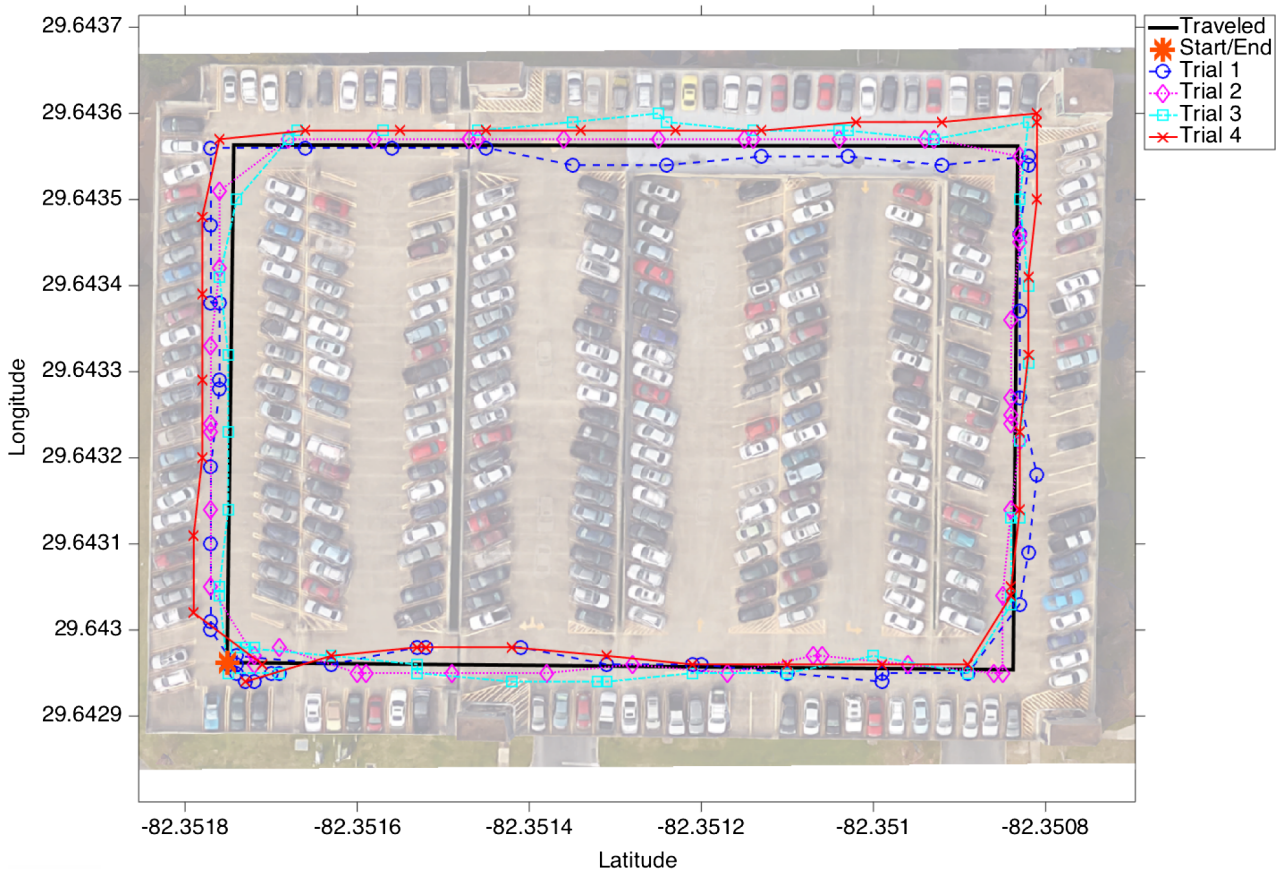


Fig. 9 Four trials of GPS values plotted with the path traveled around a University of Florida parking garage. The start/end point for all trials is indicated. Map data from Google.

G. Autonomous Micro Aerial Vehicle Pilot System

In an effort to validate that the AMP's components performed all their required functions, and thus validate the system as a whole, a number of flight tests were performed. The results of these flight tests are discussed in Sec. VII.

VII. Field-Testing Results

We have conducted numerous flight tests using the AMP on our delta wing airframe. Here, we report the performance of three cases. These flights served the purpose of both evaluating the autonomous capabilities of the AMP board as well as carrying out a full system check. The results of these flights are presented in Figs. 10–12.

During each flight, the delta wing was launched and flown by a pilot remotely until stable flight was achieved; then, the autopilot was enabled. The autopilot software and the sensor fusion algorithm used to calculate quaternion values for the autopilot can be found in [6,30], respectively.

The first flight test occurred on a clear day with wind gusts of up to 8 km/h. Three waypoints were uplinked to the AMP via the ground station and successfully navigated by the autopilot. Figure 10 shows a section of the flight during which the delta wing was circling the three waypoints in a counterclockwise loop. Figure 11 shows the control commands for roll and yaw plotted with the resulting Euler angles. Periodic yaw changes observable in Figs. 10b and 11b can be explained by the periodic nature of the waypoint navigation.

Table 7 BMP180 pressure accuracy

Floor	Actual, hPa	Measured, hPa	Standard deviation
1	1024.3	1024.3	±0.005
2	1023.9	1023.9	±0.01
3	1023.6	1023.5	±0.008
4	1023.2	1023.1	±0.01
5	1022.8	1022.7	±0.009

Table 8 SHT humidity accuracy

Salt solution	Actual, % RH	Measured, % RH	Standard deviation
LiCl	11.3	12.7	±0.10
KCl	84.3	85.9	±0.06
NaCl	75.3	75.7	±0.00

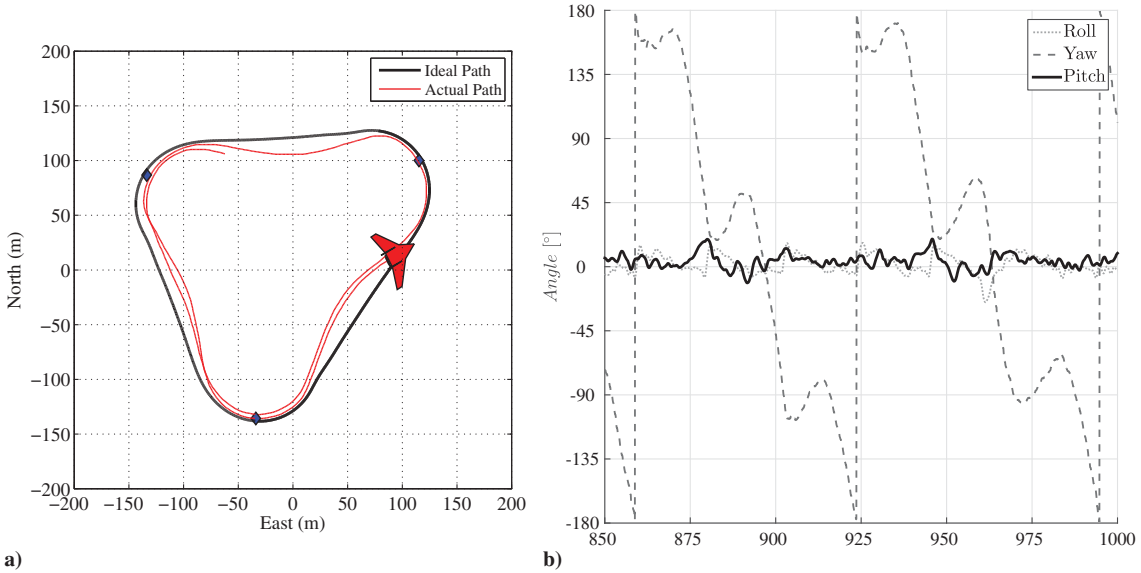


Fig. 10 Representations of a) the flight trajectory of the AMP-equipped delta wing platform compared with the reference trajectory, and b) the corresponding Euler angles of the delta wing.

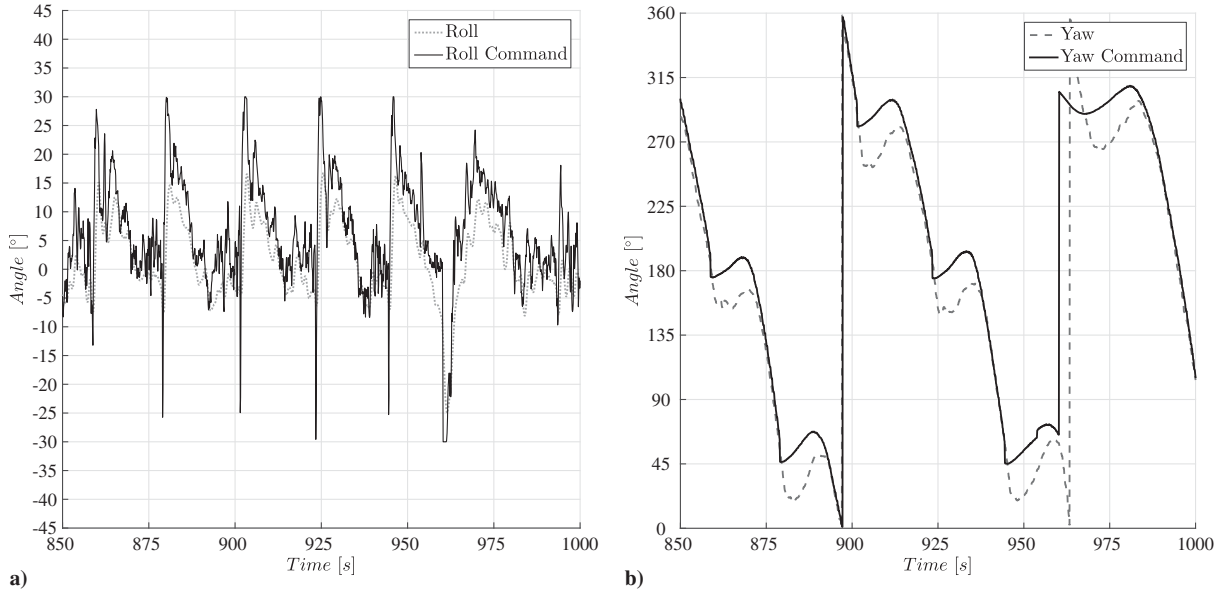


Fig. 11 Representations of a) the command roll and resulting roll against time for the flight displayed in Fig. 10, and b) the command yaw with the resulting yaw for the same flight.

The second flight test took place on a clear day with wind gusts of less than 5 km/h and was used as an opportunity to improve the tuning values for our autopilot's roll controller. No waypoints were transmitted to the autopilot; instead, various roll commands were issued, responses were observed, and roll tuning values were adjusted as needed. Figure 12a displays a section of the flight when we were sufficiently satisfied with the tuning values.

During the third flight test, a command to loiter around an uplinked waypoint was issued to the autopilot. The autopilot then flew to and circled this point, maintaining a radius of 50 m. After a number of rotations, a new loiter waypoint was issued to the autopilot, which flew to within loiter range and proceeded to circle the point as before, maintaining a 50 m radius. This flight took place on a clear day with wind gusts of up to 8 km/h, and the resulting flight data can be seen in Fig. 12b.

Based on its performance from multiple flight tests, the capability of the AMP platform was confirmed. The AMP accomplished multiple, stable, autonomous flights with waypoint navigation in dynamic atmospheric conditions using the onboard sensor package to determine altitude, attitude, and localization. The dsPIC processor demonstrated that it was capable of managing the duties of data parsing, telemetry uplink and downlink, manual control, and autonomous control. Telemetry between the ground station and AMP board proved reliable in downlinking and uplinking data and commands. The ground station proved itself to be a competent system for interacting with, and recording data from, the AMP board.

Some limitations of the autopilot were also noted during these flights. For example, some form of obstacle detection on the AMP would be very useful, decreasing the autopilots susceptibility to collisions and helping to remove the need for an operator on the ground. The system would also benefit from an airspeed sensor so that the autopilot's navigation in windy environments could be improved. Despite this, these flight tests demonstrated the AMP's qualifications as a mobile sensing platform.

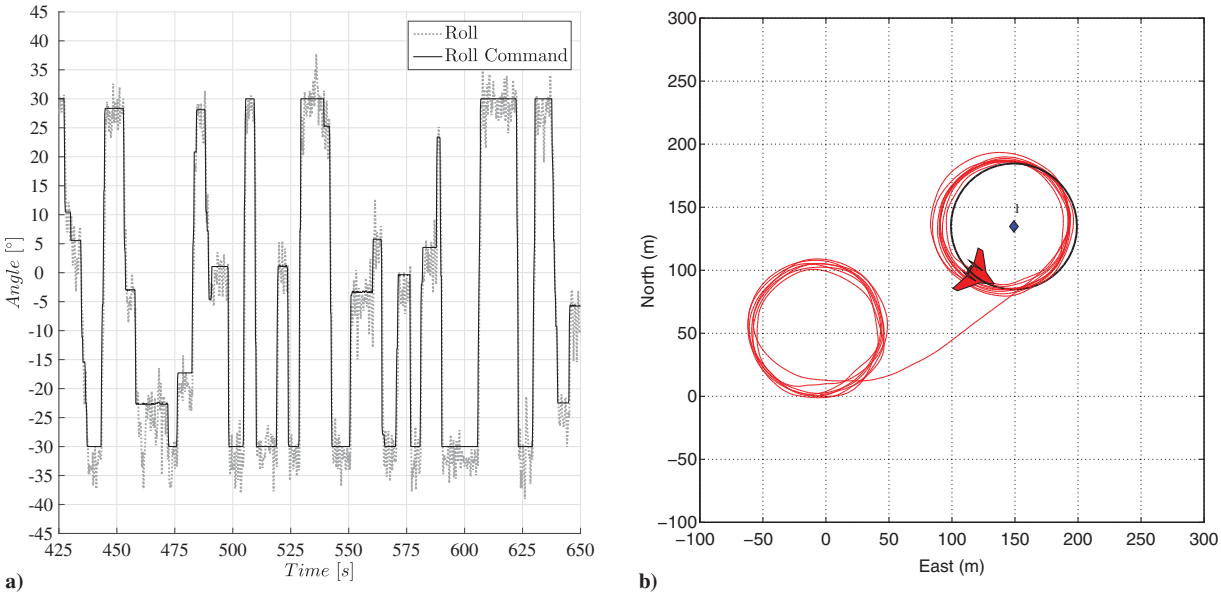


Fig. 12 Representations of a) the command roll with the resulting roll for the second flight, and b) the flight trajectory of the vehicle maneuvering through two loiter circles from the third flight.

VIII. Conclusions

A new, multivehicle autopilot designed for autonomous flight on MAVs to enable mobile, atmospheric sensing has been presented. At 6.25 g, 11.3 cm^2, and using an atmospheric thermodynamic sensing package (pressure, humidity, and temperature sensors), dual radios, GPS, and IMU, the AMP is a highly capable autopilot system designed for use on very lightweight MAVs.

The design specifications, hardware, firmware, and ground station of the AMP were detailed. The sensors used on the AMP board were each individually validated to assess their accuracy. Successful autonomous flights were performed on a 0.94-m-span MAV to illustrate the effectiveness of the AMP. From the flight results, the AMP demonstrated successful autonomous waypoint navigation, using its onboard sensors for altitude estimation, attitude estimation, and localization. Stable communication with a ground station during the flight was also confirmed because all pertinent telemetry was captured and recorded while user commands for the autopilot were successfully uplinked. These results presented the abilities of the AMP as an atmospheric mobile sensing platform, capable of autonomously navigating to a set destination and transmitting all necessary sensor data.

Future work for the AMP includes research into networking strategies between AMP boards, adding components to enable obstacle detection and wind-speed sensing, and updates to the hardware design to continue making the system as lightweight and low-power as possible.

Acknowledgments

The authors would like to thank the following for their many helpful suggestions and discussions as well as their assistance when performing tests: Matthew Silic, Richard O'Donnell, Thomas Linehan, and Sasha Zivkovic. They would also like to thank the U.S. Air Force Office of Scientific Research (ID: FA9550-13-1-0090) for their support in funding this project.

References

[1] Shaw, A., and Mohseni, K., "A Fluid Dynamic Based Coordination of a Wireless Sensor Network of Unmanned Aerial Vehicles: 3-D Simulation and Wireless Communication Characterization," *IEEE Sensors Journal: Special Issue-on Cognitive Sensor Networks*, Vol. 11, No. 3, 2011, pp. 722–736. doi:10.1109/JSEN.2010.2064294

[2] Hodgkinson, B., Lipinski, D., Peng, L., and Mohseni, K., "High Resolution Atmospheric Sensing Using UAV Swarms," *Proceedings of the International Symposium on Distributed Autonomous Robotic Systems*, Springer, New York, Nov. 2014, pp. 31–45. doi:10.1007/978-3-642-55146-8

[3] Allred, J., Hasan, A. B., Pisano, B., Panichsakul, S., Gray, P., Han, R., Lawrence, D. A., and Mohseni, K., "SensorFlock: An Airborne Wireless Sensor Network of Micro-Air Vehicles," *Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys)*, ACM, New York, June 2007, pp. 117–129. doi:10.1145/1322263.1322275

[4] Lipinski, D., and Mohseni, K., "Micro/Miniature Aerial Vehicle Guidance for Hurricane Research," *IEEE Systems Journal*, Vol. 10, No. 3, 2015, pp. 1263–1270. doi:10.1109/JSYST.2015.2487449

[5] Peng, L., Silic, M., and Mohseni, K., "A DDDAS Plume Monitoring System with Reduced Kalman Filter," *Proceedings of the International Conference on Computational Science (ICCS)*, Vol. 51, No. 4, 2015, pp. 2533–2542.

[6] Pisano, W. J., and Lawrence, D. A., "Autonomous UAV Control Using a 3-Sensor Autopilot," *AIAA Infotech@Aerospace 2007 Conference and Exhibit*, AIAA Paper 2007-2756, May 2007. doi:10.2514/6.2007-2756

[7] Hall, J., Lawrence, D., and Mohseni, K., "Lateral Control of a Tailless Micro Aerial Vehicle," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, AIAA Paper 2006-6689, Aug. 2006. doi:10.2514/6.2006-6689

[8] Peng, L., Lipinski, D., and Mohseni, K., "Dynamic Data Driven Application System for Plume Estimation Using UAVs," *Journal of Intelligent and Robotic Systems*, Vol. 74, Nos. 1–2, 2014, pp. 421–436. doi:10.1007/s10846-013-9964-x

[9] Ravela, S., Sleder, I., and Salas, J., "Mapping Coherent Atmospheric Structures with Small Unmanned Aircraft Systems," *AIAA Infotech@Aerospace Conference*, AIAA Paper 2013-4667, Aug. 2013.

[10] Lawrence, D. A., and Balsley, B. B., "High-Resolution Atmospheric Sensing of Multiple Atmospheric Variables Using the Datahawk Small Airborne Measurement System," *Journal of Atmospheric and Oceanic Technology*, Vol. 30, No. 10, 2013, pp. 2352–2366. doi:10.1175/JTECH-D-12-00089.1

[11] Remes, B., Esden-Tempski, P., Van Tienen, F., Smeur, E., De Wagter, C., and De Croon, G., "Lisa-S 2.8 g Autopilot for GPS-Based Flight of MAVs," *Proceedings of the International Micro Air Vehicle Conference and Competition (IMAV)*, International Micro Air Vehicles (IMAVS), Delft, The Netherlands, Aug. 2014, pp. 280–285.

[12] Ratti, J., Moon, J., and Vachtsevanos, G., "Towards Low-Power, Low-Profile Avionics Architecture and Control for Micro Aerial Vehicles," *Proceedings of the IEEE Aerospace Conference*, IEEE Publ., Piscataway, NJ, March 2011, pp. 1–8. doi:10.1109/AERO.2011.5747444

[13] Brisset, P., Drouin, A., Gorraz, M., Huard, P., and Tyler, J., "The Paparazzi Solution," *Proceedings of the 2nd US-European Competition and Workshop on Micro Air Vehicles*, Ecole Nationale de l'Aviation Civile (ENAC), Sandestin, FL, Oct. 2006, pp. 1–15.

[14] Meier, L., Tanskanen, P., Heng, L., Lee, G., Fraundorfer, F., and Pollefeys, M., "PIXHAWK: A Micro Aerial Vehicle Design for Autonomous Flight Using Onboard Computer Vision," *Autonomous Robots*, Vol. 33, Nos. 1–2, 2012, pp. 21–39. doi:10.1007/s10514-012-9281-4

[15] "Ardupilot Autopilot Suite," ArduPilot, Canberra, Australia, 2016, http://ardupilot.com/ardupilot/index.html [retrieved Aug. 2016].

[16] "Pixhawk Autopilot," Lorenz Meier, Zurich, Switzerland, 2016, https://pixhawk.org/modules/pixhawk [retrieved Aug. 2016].

[17] "UDB5 – PIC UAV Development Board," Sparkfun Electronics, Niwot, CO, 2016, https://www.sparkfun.com/products/11703 [retrieved Aug. 2016].

[18] "TM Series GNSS Receiver Module Data Guide," Linx Technologies, Merlin, OR, 2015.

[19] "dsPIC Digital Signal Controllers," Microchip Technologies, Chandler, AZ, 2005, http://ww1.microchip.com/downloads/en/DeviceDoc/DS-70095K.pdf [retrieved July 2015].

[20] Liu, S., Yang, M., Steiner, A., Möckel, R., and Delbruck, T., "1 kHz 2D Visual Motion Sensor Using 20 Times 20 Silicon Retina Optical Sensor and DSP Microcontroller," *IEEE Transactions on Biomedical Circuits and Systems*, Vol. 9, No. 2, 2015, pp. 207–216. doi:10.1109/TBCAS.2015.2414881

[21] Premaratne, P., Ajaz, S., Monaragala, R., Bandara, N., and Premaratne, M., "Design and Implementation of Edge Detection Algorithm in dsPIC Embedded Processor," *Proceedings of the International Conference on Information and Automation for Sustainability (ICIAFs)*, Inst. of Electrical and Electronics Engineers, New York, 2010, pp. 8–0.

[22] Briod, A., Klaptocz, A., Zufferey, J., and Floreano, D., "The AirBurr: A Flying Robot That Can Exploit Collisions," *Proceedings of the International Conference on Complex Medical Engineering (CME)*, Inst. of Electrical and Electronics Engineers, New York, 2012, pp. 569–574.

[23] Xuan, K., "Application of Real-Time Kalman Filter with Magnetic Calibration for MEMS Sensor in Attitude Estimation," *Proceedings of the 7th International Conference on Cybernetics and Intelligent Systems (CIS) and Robotics, Automation and Mechatronics (RAM)*, Inst. of Electrical and Electronics Engineers, New York, 2015, pp. 243–247.

[24] Harms, H., Amft, O., Winkler, R., Schumm, J., Kusserow, M., and Tröster, G., "Ethos: Miniature Orientation Sensor for Wearable Human Motion Analysis," *Proceedings of Sensors*, 2010, pp. 1037–1042.

[25] "MPLAB X IDE," Microchip Technologies, Chandler, AZ, 2014, http://www.microchip.com/pagehandler/en-us/family/mplabx [retrieved June 2015].

[26] "PICkit 3 In-Circuit Debugger/Programmer User's Guide, Rev. A," Microchip, Chandler, AZ, 2013.

[27] "MPLAB ICD 3 In-Circuit Debugger User's Guide, Rev. B," Microchip, Chandler, AZ, 2014.

[28] Faludi, R., *Building Wireless Sensor Networks: With Zigbee, XBee, Arduino, and Processing*, O'Reilly Media, Boston, MA, 2003, pp. 111–160.

[29] Renaudin, V., Afzal, M., and Lachapelle, G., "Complete Triaxis Magnetometer Calibration in the Magnetic Domain," *Journal of Sensors*, Vol. 2010, Dec. 2010, pp. 1–10. doi:10.1155/2010/967245

[30] Mahony, R., Hamel, T., and Pflimlin, J., "Nonlinear Complementary Filters on the Special Orthogonal Group," *IEEE Transactions on Automatic Control*, Vol. 53, No. 5, 2008, pp. 1203–1218. doi:10.1109/TAC.2008.923738

[31] Linehan, T., Shields, M., and Mohseni, K., "Development, Characterization, and Validation of a Four Axis Wind Tunnel Positioning System," *52nd Aerospace Sciences Meeting*, AIAA Paper 2014-1308, Jan. 2014. doi:10.2514/6.2014-1308

[32] Shields, M., and Mohseni, K., "Effects of Sideslip on the Aerodynamics of Low Aspect Ratio Wings at Low Reynolds Numbers," *AIAA Journal*, Vol. 50, No. 1, 2012, pp. 85–99. doi:10.2514/1.J051151

[33] Shields, M., and Mohseni, K., "Roll Stall for Low-Aspect-Ratio Wings," *Journal of Aircraft*, Vol. 50, No. 4, 2013, pp. 1060–1069. doi:10.2514/1.C031933

[34] Greenspan, L., "Humidity Fixed Points of Binary Saturated Aqueous Solutions," *Journal of Research of the National Bureau of Standards*, Vol. 81, No. 1, 1977, pp. 89–96. doi:10.6028/jres.081A.011

E. Atkins
Associate Editor