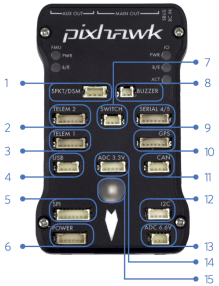
Towards An Open Instrumentation Platform: Getting The Most From MAVLink, ArduPilot, and BeagleBone

Presentation Outline

MAVs: Autopilots, Copters, and Drones

- Micro Aerial Vehicles and (RC) Copters are not necessarily Drones
  - Drones have an autopilot and are (or at least can be) autonomous
  - RC copters (even with an IMU) are generally not fully autonomous
- An autopilot has programmable firmware
  - Usually a microcontroller (typically AVR or ARM)
  - Lots of I/O ports (I2C, SPI, UARTs)
  - Open Firmware: Ardupilot, OpenPilot/LibrePilot, PX4, etc
- Open source hardware: autopilot boards and host machines
  - Ardupilot - BBBMINI Cape/BeaglePilot, PixHawk, PX/4, AUAV-X2, PXFmini, VRBRAIN 4 (supports dual firmware)
  - LibrePilot - CopterControl, CC3D, Revo/Nano, OPLink Mini
  - Autopilot board price range $20 - $200 and up
  - Embedded Linux board price range $5 - $100
  - $200 Smart Drone "Kit" Using PXFmini and RaspberryPi Zero
  - $100 DIY Smart Drone using Pi Zero and BYO PXFmini clone

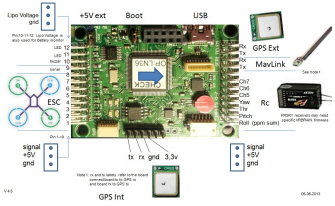
Autopilot Examples



1 Spektrum DSM receiver
2 Telemetry (radio telemetry)
3 Telemetry (on-screen display)
4 USB
5 SPI (serial peripheral interface) bus
6 Power module
7 Safety switch button
8 Buzzer
9 Serial
10 GPS module
11 CAN (controller area network) bus
12 I²C splitter or compass module
13 Analog to digital converter 6.6 V
14 Analog to digital converter 3.3 V
15 LED indicator

PixHawk Connectors (top)

CopterControl3D



VR BRAIN 4/5 Connections

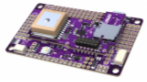
MAVLink / Ardupilot Flight Models

- Original (software) vehicle models in ArduPilot and MAVLink
  - Copter
  - Rover
  - Plane
  - Antenna tracker
- Current vehicle / flight models in MAVLink v1.0/v2.0
  - Generic micro air vehicle
  - Fixed-wing aircraft
  - Single/multi-rotor copters
  - Antenna tracker / ground control station
  - Airship
  - Free balloons, Kites
  - Rocket
  - Ground rover
  - Surface vessel, Submarine
  - Flapping wing

Ardupilot / APM

The ArduPilot family



Autopilot	ArduPilot (aka “Legacy”)	ArduPilotMega APM 1 – 1280	ArduPilotMega APM 1 – 2560	ArduPilotMega APM 2
Date of introduction	Q1 2009	Q1 2010	Q1 2011	Q4 2011
Status	Discontinued	Discontinued	Active	Active
Processors	atmega 328, attiny	atmega 1280, atmega 328	atmega 2560, atmega 328	atmega 2560, atmega 32u2, MPU-6000 DMP processor
Onboard sensors	None. External: Thermopiles or optional ArduIMU	3-axis gyro, 3-axis accel, baro, optional mag	3-axis gyro, 3-axis accel, baro, optional mag	6-axis MPU6000 (gyro+accel), baro, mag, GPS
Datalogging memory	None	2MB	2MB	4MB
Size	30x50x30mm	40x72x20mm	40x72x20mm	40x65x10mm
Assembly required	Lots!	Some soldering	Some soldering	None!

MAVLink and MAVCONN

MAVLink Protocol and Message Specification

A very lightweight, header-only message marshalling library for micro air vehicles.

- Common MAVLink Message Documentation
- MAVLink Code and Generator

There is a “common message set” containing widely used messages, which is distributed as header-only C library. If you need to add your own custom messages, you can generate the code with the MAVLink Generator (C/C++) and pymavlink (Python).

MAVCONN Aerial Middleware

The PIXHAWK MAVOS middleware / robotics toolkit is a heterogenous system consisting of an image processing and communication architecture for computer vision based flight control.

- Pixhawk MAVCONN introduction at EMAV 2009 (1st place indoor)
- Fully autonomous flight demonstrated July 2010

MAVLink Message Spec Example

MAVLINK Common Message Set

These messages define the common message set, which is the reference message set implemented by most ground control stations and autopilots.

MAVLink Protocol Version

This file has protocol version: 3. The version numbers range from 1-255.

0

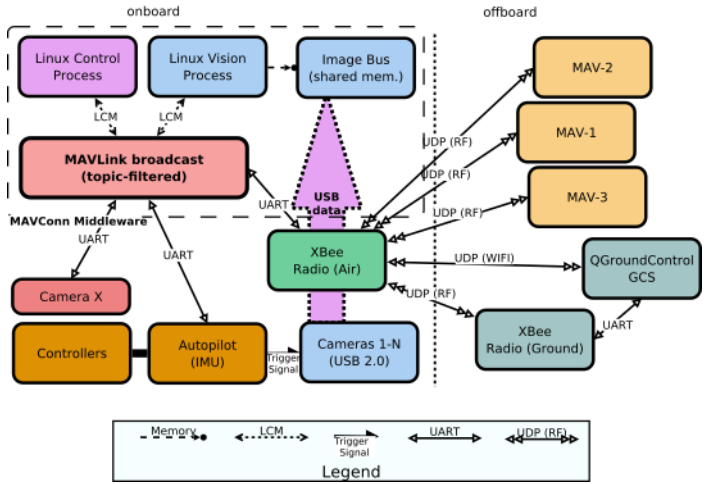
MAVLink Type Enumerations

MAV_AUTOPILOT

Micro air vehicle / autopilot classes. This identifies the individual model.

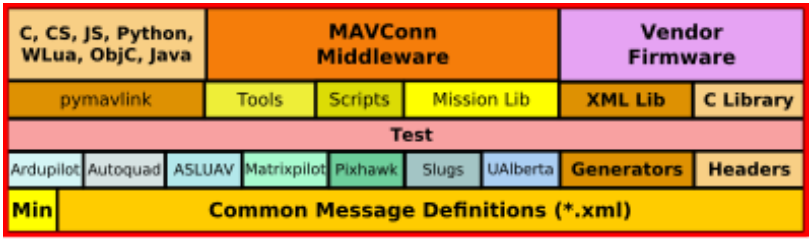
CMD ID	Field Name	Description
0	MAV_AUTOPILOT_GENERIC	Generic autopilot, full support for everything
1	MAV_AUTOPILOT_RESERVED	Reserved for future use.
2	MAV_AUTOPILOT_SLUGS	SLUGS autopilot, http://slugsuav.soe.ucsc.edu
3	MAV_AUTOPILOT_ARDUPILOTMEGA	ArduPilotMega / ArduCopter, http://diydrones.com
4	MAV_AUTOPILOT_OPENPILOT	OpenPilot, http://openpilot.org
5	MAV_AUTOPILOT_GENERIC_WAYPOINTS_ONLY	Generic autopilot only supporting simple waypoints
6	MAV_AUTOPILOT_GENERIC_WAYPOINTS_AND_SIMPLE_NAVIGATION_ONLY	Generic autopilot supporting waypoints and other simple navigation commands

MAVLink System Architecture



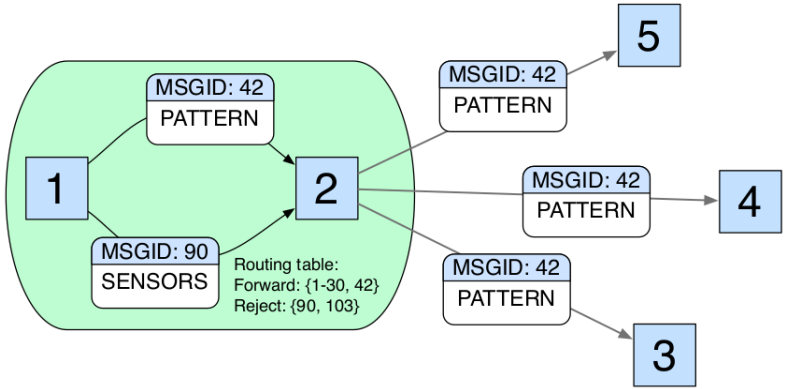
- Full MAVLink/MAVCONN autonomous vehicle configuration
- Linux host system and integrated machine vision/object tracking
- Uses multiple transport/physical layers and redundant GCS links
- Camera X is a third-party camera connected via serial link

MAVLink Software Stack



- MAVConn "middleware" typically runs on onboard Linux host
- Host connects to autopilot and onboard cameras
- MAVLink supports multiple OS/IPC Mechanisms/Interfaces
- MAVLink common messages are extended by autopilot firmware

MAVLink Message Protocol Routing



- The MAVConn message broadcast includes action, status and command messages, which can be used both for onboard and offboard communication
- Messages are identified and filtered by content instead of recipient addresses
- Routing nodes (node 2) filter messages that belong only to a particular subset (nodes 3, 4, and 5)

Ground Control Station Software

In addition to the onboard software already discussed, we also need a ground control station, typically used for everything from real-time communication and tracking to mission/vehicle configuration and flashing new autopilot firmware images.

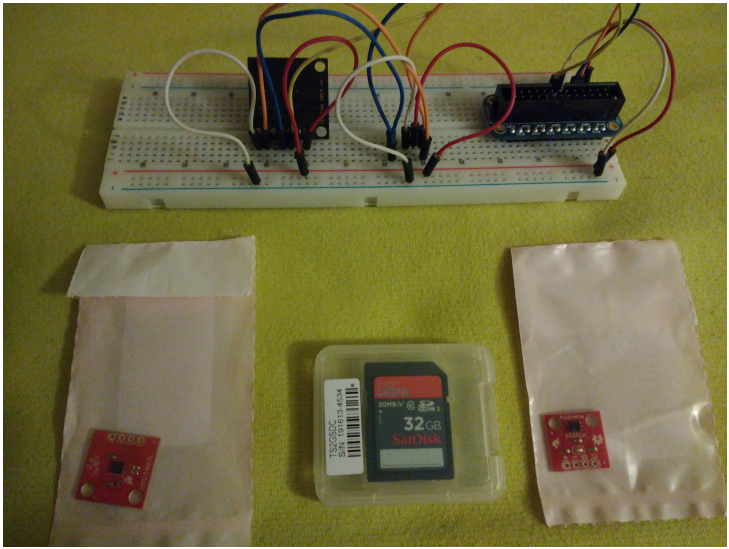
Options include:

- QGroundControl: provides full flight control and mission planning for any MAVLink enabled drone and configuration for ArduPilot or PX4 Pro powered vehicles.
- Mission Planner: Full featured and widely used open source GCS software Platform: Windows, Mac OS X (Using Mono)
- APM Planner 2.0: smaller user base and reduced feature set when compared with Mission Planner but better on Linux.
- MAVProxy: Linux GCS often used by Plane developers. Primarily a command line interface with graphical modules for map and mission editing. Written in Python, and extensible via python modules.

GPS, IMUs, Sensors, and More

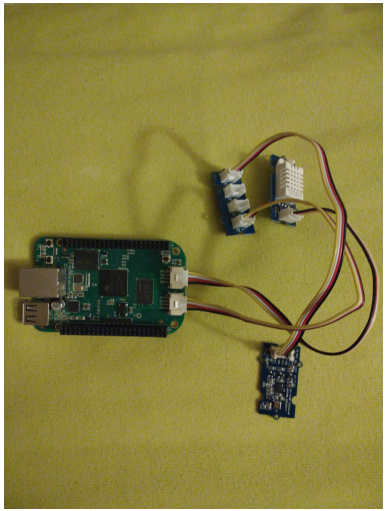
- Chips, Breakout Boards, Combo Boards, Builtin
  - Bare sensors: cheaper/smaller to integrate but requires expertise
  - Breakout boards: still cheap, convenient prototyping, SPI/I2C
  - Combo boards: multiple sensors, already integrated, watch out for compatibiity
  - SoC with integrated sensors: most convenient and least cheap, limited choice
- Many available types/models of IMU online (eg, SparkFun)
  - 1 -10 DOF models on breakout boards $10 - $100
  - Suitable for many applications (check data sheets)
  - Higher end includes multiple slave buses, high speed, onboard processing
- GPS for low-power mobile applications, typical UART interface
- Specialized ARM SoCs for robotics and vehicle applications
  - BeagleBone Green - includes Grove sensors, bus connectors
  - BeagleBone Blue - includes integrated Strawson Robotics Cape
  - Udoo Neo (Full or Extended) - builtin microcontroller, 9-DOF IMU, Wifi/BLE

Hardware Examples

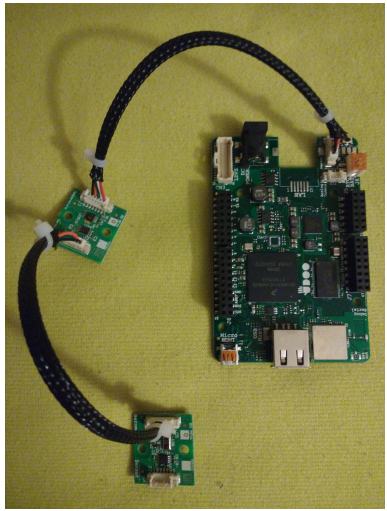


Individual Sensor Breakouts, Lightning Detector

Hardware Examples... **cont...**



BeagleBone Green with Grove Sensor Breakouts



Udoo Neo Extended with IoT Sensor Breakouts

Computer Vision, Cameras and Metadata

Still Image Example: FLIR ViewPro IR Camera

- FLIR Thermal Tiff image file - a TIFF file holding an IR image and corresponding metadata (uses extensions provided by TIFF, Exif, and XMP Specifications).

Example MAVLink Interface Requirements:

MAV0001: The System Shall_MAV0001 provide a compatible MAVLink interface for querying and storing position and attitude as metadata

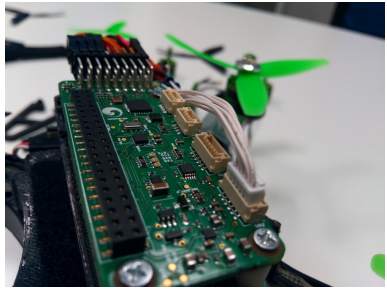
MAV0002: The System Shall_MAV0002 use only supported autopilot messages for 3D position/velocity, camera attitude (look-angle), and associated airfrme yaw/pitch/roll angles and speeds.

MAV0003: The System Shall_MAV0003 use the MAVLink common message set for generic vehicle support.

MAV0004: The System Shall_MAV0004 use the MAVConn middleware broadcast interface on systems with a Linux host.

MAV0005: The System Shall_MAV0005 use the Ardupilot serial stream interface on systems with only a Pixhawk autopilot.

Airframes and Kits



Other Hardware and Software/Firmware

- Autonomous 3D-printed Drone
  - http://tinyurl.com/3D-printed-drone
- UAVCAN (a lightweight protocol for aerospace and robotics)
  - https://github.com/uavcan
- The Robot Operating System (ROS)
  - https://github.com/ros
- MicroPython (Python for microcontrollers)
  - https://github.com/micropython/micropython
- RTIMULib2 (C++ and Python 9-dof, 10-dof and 11-dof IMU library)
  - https://github.com/Nick-Currawong/RTIMULib2
- WeatherPi
  - https://github.com/VCTLabs/weather-rpi
- DroneCode.Org UAV repositories
  - https://github.com/Dronecode

Useful (and fun) Applications

- BeagleBone NDVI Cape http://tinyurl.com/beaglebone-ndvi
  - Custom hardware and image processing board with dual cameras
- SeaSlug (marine sensor system) http://tinyurl.com/SeaSlug-pdf
  - A high-uptime, long-deployment mobile marine sensor platform
- GSoC BeagleSat https://github.com/nvisnjic/BeagleSat
  - An open source nano satellite platform toolkit (a Google Summer of Code project)
- GSoC Sonics https://github.com/Visaoni/beagle-sonic-anemometer
  - An open source anemometer using ultrasonic time of flight (a Google Summer of Code project)

Resources

Ardupilot and MAVLink

- http://copter.ardupilot.com/
- https://github.com/mavlink/mavlink
- https://github.com/mavlink/c_library
- https://github.com/mavlink/qgroundcontrol
- https://github.com/mavlink/c_uart_interface_example
- https://github.com/pixhawk/mavconn
- https://github.com/diydrones/ardupilot
- http://tinyurl.com/FLIR-TIFF-MAVLink

Additional Resources

- https://www.dronecode.org/
- https://www.librepilot.org/
- http://dev.ardupilot.com/wiki/building-px4-for-linux-with-make/
- http://copter.ardupilot.com/wiki/build-your-own-multicopter/
- http://www.instructables.com/id/DIY-Drones/

References and Specifications

Huang, Olson and Moore, Lightweight Communications and Marshalling for Low-latency Interprocess Communication. MIT CSAIL Technical Report, 2009.

Lorenz Meier, Petri Tanskanen, Lionel Heng, Gim Hee Lee, Friedrich Fraundorfer, and Marc Pollefeys. Pixhawk: A micro aerial vehicle design for autonomous flight using onboard computer vision. Autonomous Robots (AURO), 2012.

The canonical MAVLink Common Message Set is common.xml, which defines both the software interface and the MAVLINK Common Message Set documentation.

The ArduPilot "Copter" interface variant is defined in the ArduCopter GCS_Mavlink.cpp source file.

The Exif 2.3 Specification - Exchangeable image file format for digital still cameras: Exif Version 2.3, Revised on December, 2012, Camera & Imaging Products Association.

The XMP 1.0 Specification - XMP Specification, Part 1 (April, 2012), Part 2 (November 2014), Part 3 (November 2014), Adobe Developers Association.

License and Author Info

Author:	Stephen L Arnold, Principal Scientist @vctlabs.com, developer @gentoo.org
Contact:	stephen.arnold _at_ acm.org

Revision:	0.4.0
Date:	2021-01-23, 22:38 PST8PDT