

MAV Downlink Advanced Users Guide 8/3/2014

This is a guide for using the MAV Downlink application with only your own hardware. In this mode, you will need an Android smartphone, a downlink receiving server, and a device that runs Mission Planner or your preferred MAV mission application. For testing purposes (or if you are clever enough), you can run the downlink receiving server and mission planner on the same computer.

Note: This guide assumes some familiarity with basic computer operations and is just an overview of how to use the MAV Downlink application for personal use. If you would like to see a more user friendly version of this application developed, please drop me a comment here:

<http://diydrones.com/profiles/blogs/apm-telemetry-using-an-android-smartphone>

What You Need Before You Begin:

- An APM 2.5 or 2.6 controller board. Clones or others might work but are untested. Pixhawk does not currently work.
- An Android smartphone that is running Android 3.1+ and has an OTG-compatible USB port.
- The MAV Downlink application installed on the aforementioned device.
<https://play.google.com/store/apps/details?id=com.appliedanalog.uav.mavdownlink>
- An OTG to USB cable. Ex: http://www.amazon.com/LIFETIME-WARRANTY-Electronics-Micro-Cable/dp/B005GGBYJ4/ref=sr_1_3?ie=UTF8&qid=1407083771
- A computer with Java 7 or greater installed. This will be referred to as the “downlink receiving server”.
- The Local MAV Mapper application downloaded on your downlink receiving server:
<http://www.applieduas.com/LocalMavMapper.zip>
- A computer, tablet or phone with a mission planning application that uses MAV installed, e.g. DroidPlanner or MissionPlanner.

Instructions

What follows is a step by step guide to connecting to your MAV-compatible autopilot over WLAN using MAV Downlink.

1. Launch “LocalMavMapper.jar” on your downlink receiving server. This can be accomplished by double clicking the file.

2. Click "Start Mapper". If you are prompted by a firewall service to allow the application permissions to access the internet, "Allow" them.

UAV Mapping Server Port:

Android Client:

Disconnected

MAV Client:

Disconnected

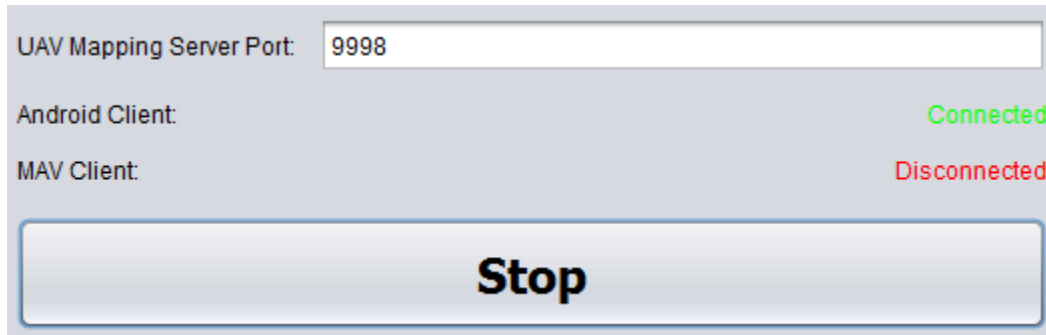
Start Mapper

3. Enable Wi-Fi on your phone. Ensure it is connecting to the same network as the downlink receiving server.
4. Launch MAV Downlink on your phone. You will be prompted to plug in your device.
5. Plug your autopilot into your phone using the OTG cable. For APM 2.6 devices, the OTG cable should be plugged in to the standard USB port on the device, NOT the telemetry port.

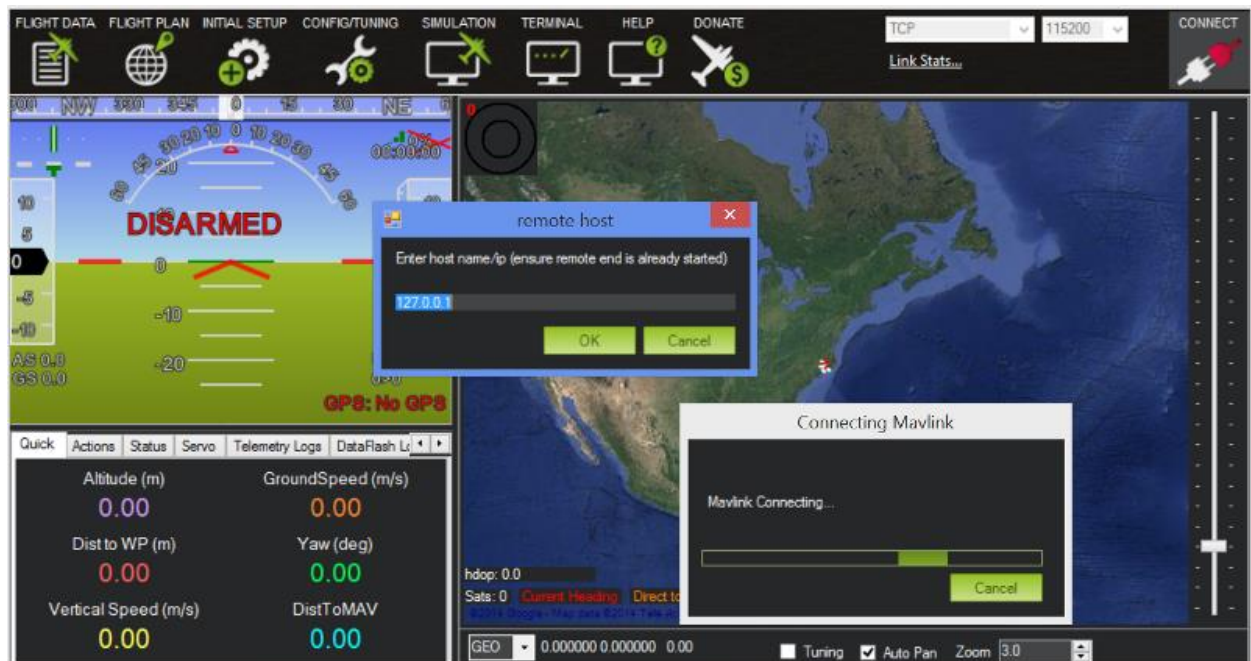


6. You will be prompted on your phone what application you want to launch to handle the USB device. Select "MAV Downlink".
7. MAV Downlink will be re-launched, this time with the "Start Downlink" button enabled.

8. In the “Server IP” field, type in the IP address for your downlink receiving server.
9. Click “Start Downlink”. The “Downlink” light should turn green immediately. On your downlink receiving server, the “Android Client” should show up as “Connected” in the downlink receiving server application as well.



10. (Mission Planner is assumed here, if you are using a different client, you will have to follow different, but similar steps)
Launch Mission Planner on your mission computer. This computer should be on the same computer network as the downlink receiving server.
11. At the top of the Mission Planner, pull down the connectivity dropdown (usually defaults to COM) and select TCP, press “Connect”.



12. When prompted to enter the IP address, enter the IP of the downlink receiving server.
13. When prompted to enter the port, enter 9998.
14. On the downlink receiving server, you should see the MAV Client button become green.
15. On the phone, you should the MAVLink icon go green and messages start to be transmitted/received.
16. On mission planner, you should connect to the autopilot.
17. Congratulations! You are connecting to your autopilot wirelessly using your phone!

Connectivity

This section seeks to answer the question “So what? Connecting to my autopilot using WiFi is useless!”

This application was designed with the intent that the “downlink receiving server” would be replaced by a centralized server sitting on the Internet somewhere. The version of the application to which this manual pertains forgoes that and lets you set up your own “downlink receiving server”. This is a universal server application and can be run anywhere, as long as it is accessible via IP from both the smartphone and the computer running Mission Planner.

For simplicity and demonstration purposes, this manual walks you through accomplishing this using a WiFi LAN, but rest assured that the creator of this application has set up the very same suite of tools to work via a 4G cellular link on both the smartphone and the mission planner device. This means that I can talk to my plane using Mission Planner in the field as long as both I and my plane have 3G or 4G coverage. On Edge this service gets pretty bad. I am not including specific instructions on how to accomplish this now, but I will give the brief of it:

- 1) Set up your home network so that all WAN traffic on ports 9999 and 9998 (or whatever you configure) gets sent to your downlink receiving server.
- 2) Find out the IP address of your home network. (Hint: go to <https://www.google.com/search?q=whats+my+ip+address>)
- 3) Use this IP address in lieu of your local LAN address when connecting to the downlink receiving server.
- 4) Bonus points: set up a no-ip forwarding service if you don't have a static IP address from your provider. I used <http://www.noip.com>

Command Line Downlink Receiving Server Program

I have also put together a simple command-line based downlink receiving server program for those that want to run this on their own dedicated server. You can download this here:

<http://www.applieduas.com/MavMapperCmd.zip>

Run this program by executing this command:

```
java -jar MavMapperCmd.jar
```

The program will run until it is forcibly killed.

Good luck guys and please let me know if you like the Application and want to see some more features here:

<http://diydrones.com/profiles/blogs/apm-telemetry-using-an-android-smartphone>

Whether or not I put more time/money into this is **entirely** dependent on your responses.