

## PENINGKATAN TAMPILAN *SOFTWARE MISSION PLANNER* MENGGUNAKAN *SCRIPT PYTHON* DALAM MEMONITOR DATA TERBANG UAV

Andreas P Adi

Peneliti Pusat Teknologi Roket, LAPAN

Email: [and741@gmail.com](mailto:and741@gmail.com)

### Abstrak

LAPAN telah mengembangkan wahana tanpa awak (UAV) RKX-200EDF/TJ sebagai sarana penguasaan teknologi kendali wahana berkecepatan tinggi, khususnya kendali terbang secara autopilot mengikuti *waypoint*. Sistem kendali yang digunakan pada wahana adalah Ardupilot, sedangkan pada *ground station* digunakan *software Mission Planner* untuk mengatur *waypoint* yang harus dilalui wahana serta memantau data-data terbang wahana. *Mission Planner* dapat menampilkan jalur terbang dan *attitude* wahana secara visual, serta menampilkan secara numerik enam data terbang yang dianggap penting oleh pengguna yang dapat dipilih dari antara banyak data terbang yang tersedia. Makalah ini membahas metode untuk menampilkan lebih banyak data terbang agar dapat dipantau secara bersamaan, yaitu menggunakan *script Python* yang dijalankan dari *Mission Planner*. Metode ini praktis karena tidak perlu mengubah *source code Mission Planner* dan pengguna dapat mengganti suatu *script* dengan *script* lain sesuai kebutuhan secara *real time* pada saat pelaksanaan uji terbang. Keterbatasan metode ini adalah tampilan datanya berukuran relatif kecil.

Kata kunci: UAV, data terbang, ardupilot, *mission planner*, *script Python*.

### Abstract

LAPAN has developed unmanned aerial vehicles RKX-200EDF/TJ as a means for mastering high speed vehicle control technology, especially autopilot flying control system by waypoints. The control system used onboard is Ardupilot, while Mission Planner software is used on ground station to set the waypoints to be followed by the vehicle and to monitor flight data. The Mission Planner can display flight path and vehicle attitude visually, and display six main flight data that can be chosen from many data that available. This paper discusses a method to display more flight data to be monitored simultaneously using Python script run from inside the Mission Planner. This method is practical because there is no need to modify Mission Planner source code, and we can change one script with another script according to our need in real time during flight test. The limitation of the method is that the data is displayed in small font size.

Keywords: UAV, flight test, ardupilot, mission planner, Python script.

## 1. PENDAHULUAN

Sejak tahun 2013 Pusat Teknologi Roket LAPAN telah merancang dan mengembangkan suatu pesawat kendali tanpa awak (UAV)[1][2][3]. Pesawat ini diberi nama RKX-200 EDF yang menggunakan mesin *electric ducted fan* (EDF) berkecepatan 200 km/jam. Dan sejak tahun 2014 juga dikembangkan RKX-200 TJ yang berbahan bakar kerosin yang berkecepatan 250 km/jam. Pesawat ini dirancang untuk dijadikan sebagai wahana terbang untuk penguasaan teknologi kendali wahana berkecepatan tinggi, khususnya kendali terbang secara autopilot mengikuti *waypoint*. Sejalan dengan itu juga dilakukan kegiatan litbang aspek-aspek terkait, seperti *launching* wahana dan visualisasi dinamika terbang wahana [4][5].

RKX-200 EDF/TJ menggunakan sistem kendali terbang berbasis Ardupilot. Ardupilot sudah dikenal di dunia pesawat tanpa awak. Merupakan sebuah sistem autopilot yang didasarkan atas “*Arduinooopen source electronic paltform*” yang digunakan untuk mengendalikan helikopter, pesawat *fix wing* dan pesawat multi rotor[6]. Ardupilot ini juga dilengkapi dengan system IMU (*inertial Measurement Unit*) dengan menggunakan akselerometer, gyroskop dan magnetometer, sehingga disamping menentukan

lokasi atau posisi pesawat, alat ini juga dapat memberikan informasi mengenai kecepatan dan gerak serta *attitude* pesawat selama terbang.

Untuk mengetahui kondisi pesawat selama terbang, informasi yang dikirim Ardupilot dikirim ke stasiun darat. *Mission Planner* adalah salah satu *software* yang dapat digunakan untuk menampilkan data dari Ardupilot[7]. Sistem Ardupilot -*Mission Planner* merupakan sistem kendali wahana terbang dan aplikasi *ground station* yang bersifat *open source* dan banyak digunakan[8][9]. Ardupilot digunakan untuk mengatur kendali dan stabilisasi wahana, sedangkan *Mission Planner* digunakan untuk mengatur waypoint yang harus dilalui wahana dan memantau data-data uji terbang dari Ardupilot. Gambar 1-1 memperlihatkan contoh tampilan dari *Mission Planner*.



Gambar 1-1. Tampilan *Mission Planner* dalam memonitor data terbang

Sebagaimana terlihat pada gambar tersebut, *Mission Planner* dapat menampilkan jalur terbang dan *attitude* wahana secara visual, serta menampilkan secara numerik enam data terbang yang dianggap penting oleh pengguna pada kotak kiri bawah. Keenam data tersebut dapat dipilih dari antara banyak data terbang yang tersedia di *Mission Planner*.

Makalah ini membahas metode untuk menampilkan lebih banyak data terbang agar dapat dipantau secara bersamaan sesuai kebutuhan. Pertama-tama akan dipaparkan lebih jauh tentang data yang ada pada *Mission Planner*, kemudian dibahas teknik menampilkan data lebih lengkap menggunakan *script Python* dan contoh penggunaannya, serta kelebihan dan kekurangan metode ini.

## 2. TAMPILAN DATA *MISSION PLANNER*

*Software Mission Planner* digunakan untuk merencanakan uji terbang (mengatur *waypoint* yang hendak dilintasi wahana) serta menerima data terbang wahana yang dikirim dari *Ardupilot* dan menampilkannya secara visual dengan baik.

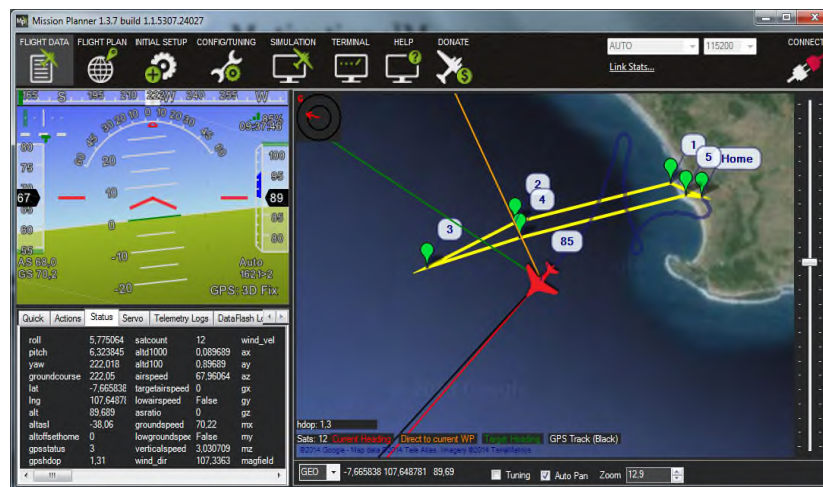
Seperti terlihat pada Gambar 1, pada kotak kanan layar *Mission Planner* terlihat peta wilayah uji terbang, titik-titik *waypoint* yang hendak dilintasi wahana, serta jalur lintasan terbang *riil* dari wahana. Pada kotak kiri atas terlihat tampilan simulasi pandangan dari kokpit wahana, yang dengan jelas menggambarkan *attitude* wahana. Pada kotak kiri bawah terlihat tampilan 6 (enam) data terbang yang utama, antara lain *altitude* (ketinggian), *ground speed* (kecepatan), *distance to waypoint* (jarak ke *waypoint* berikutnya), *yaw* (geleng), *vertical speed* (kecepatan vertikal) dan *distance to mav* (jarak ke pangkalan).

Enam data tersebut dapat dipilih oleh *user* dari banyak data yang ada di dalam sistem *Mission Planner*, bukan hanya data terbang yang diterima dari *Ardupilot* (*altitude*, *yaw* dan sebagainya) melainkan juga data hitungan/olahan *Mission Planner* sendiri (jarak ke waypoint, jarak ke pangkalan dan sebagainya). Data selengkapnya diperlihatkan pada Tabel 1-1.

**Tabel 1-1. Data lengkap pada *Mission Planner***

Roll	pitch	yaw	lat
lng	groundcourse	altoffsethome	gpsstatus
gpshdop	satcount	altd100	altd1000
airspeed	targetairspeed	groundspeed	verticalspeed
wind_dir	wind_vel	ax	ay
az	gx	gy	gz
mx	my	mz	failsafe
rxrssi	chx1in	chx2in	chx3in
chx4in	chx5in	chx6in	chx7in
chx8in	ch1out	ch2out	ch3out
ch4out	ch5out	ch6out	ch7out
ch8out	nav_roll	nav_pitch	nav_bearing
target_bearing	wp_dist	alt_error	ber_error
aspd_error	wpno	mode	dimbrate
tot	distTraveled	timeInAir	turnrate
radius	battery_voltage	battery_remaining	current
HomeAlt	DistToHome	press_abs	sonarrange
sonarVoltage	armed		

Dengan mengklik tab "Status" pada kotak kiri bawah data-data tersebut dapat ditampilkan di layar, seperti terlihat pada Gambar 2-1.



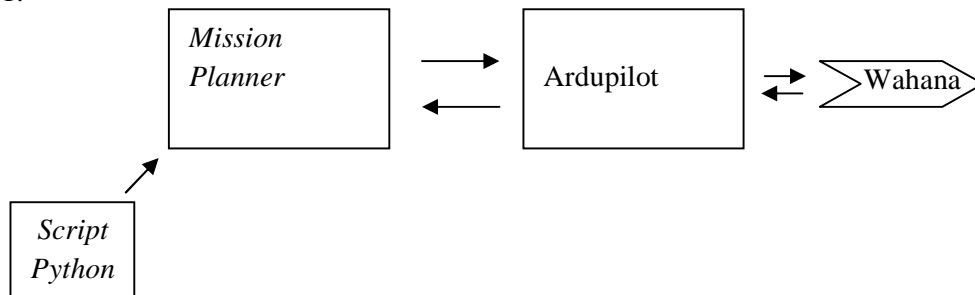
**Gambar 2-1. Tampilan *Mission Planner* untuk lebih banyak data pada kotak kiri bawah**

Pada kotak kiri bawah terlihat banyak data, dan dapat digeser (*scroll*) untuk melihat data-data lain secara bergantian. Akan tetapi kita tidak dapat memilih data apa saja yang hendak ditampilkan, sehingga tidak praktis karena tidak semua data perlu dimonitor saat uji terbang. Hanya data-data tertentu sesuai tujuan uji terbang saat itu saja yang diperlukan. Jika menggunakan tampilan standar hanya enam data yang terlihat, sedangkan pada tab Status justru terlalu banyak data yang malah sulit diamati. Untuk itu diperlukan cara untuk menampilkan lebih banyak data secara praktis, yang dapat bebas dipilih sesuai kebutuhan.

### 3. PENINGKATAN TAMPILAN MISSION PLANNER

Untuk melakukan modifikasi atau peningkatan kemampuan sistem *Mission Planner* dapat dilakukan melalui beberapa pendekatan. Pendekatan pertama, karena *Mission Planner* bersifat *open source* maka dapat dimodifikasi sesuai kebutuhan dengan mengedit *source code* program *Mission Planner*. Hal ini kurang praktis karena hasil modifikasinya hanya dapat menampilkan data itu saja, tidak dapat diganti untuk menampilkan data lain secara langsung (*real time*) pada saat uji terbang.

Pendekatan kedua adalah menggunakan script Python. Hal ini dimungkinkan karena *Mission Planner* memiliki fitur dapat menjalankan *script Python* [10]. Kita cukup menuliskan *script/program* yang perlu dijalankan oleh *Mission Planner* untuk menampilkan data-data yang dikehendaki. Selain tidak perlu memodifikasi *source code* program *Mission Planner*, pendekatan ini juga praktis karena kita dapat mengganti suatu *script* dengan *script* yang lain untuk menampilkan data yang berbeda, pada saat uji terbang secara *real time*). Skema penggunaan *Mission Planner* dengan *script Python* ditunjukkan pada Gambar 3-1.



Gambar 3-1. Skema pemrograman *Mission Planner* dengan script Python

Keluaran dari *script Python* muncul dalam suatu jendela kecil di luar jendela *Mission Planner* sehingga dapat diletakkan secara bebas sesuai keinginan, seperti ditunjukkan pada Gambar 3-2. Terlihat di sebelah kiri bawah adalah jendela yang menampilkan beberapa data tambahan yang dipanggil melalui *script Python*. Enam data utama tetap terlihat jelas di kotak *Mission Planner*, sedangkan data-data tambahan keluaran *script Python* ukurannya relatif kecil. Akan tetapi jika ditayangkan menggunakan proyektor akan terlihat cukup jelas.



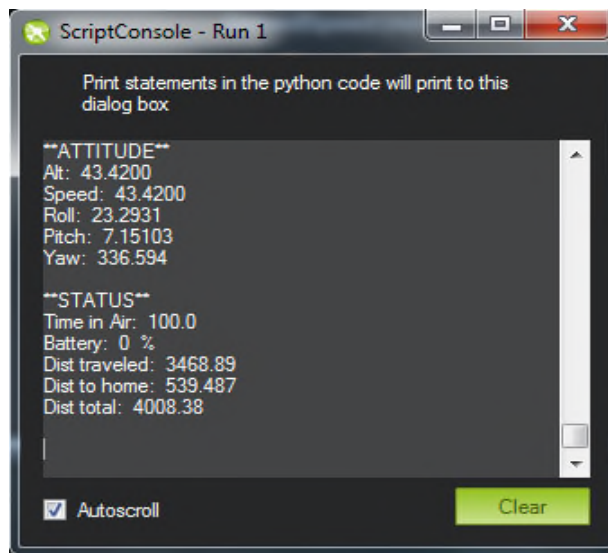
Keluaran script Python

Gambar 3-2. Tampilan jendela keluaran script Python

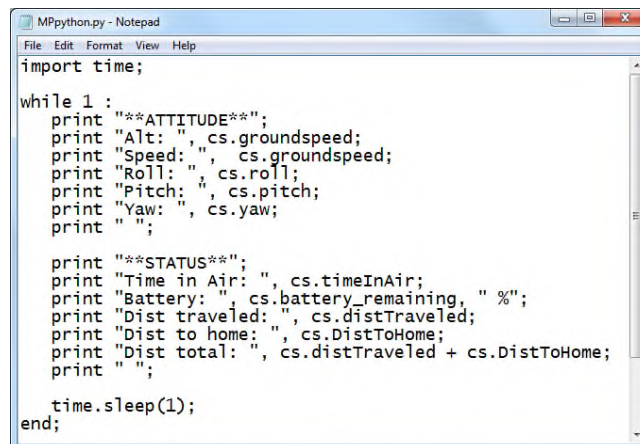


Pada contoh di atas ditampilkan satu set data-data: waktu terbang, koordinat (*atitude dan longitude*), percepatan horizontal, status baterai, kekuatan sinyal radio dan jarak ke pangkalan. Data-data ini dapat di-*update* setiap selang waktu tertentu (misalnya setiap 1 detik).

Kita dapat mengganti keluaran tersebut dengan satu set data-data lain, dengan cara menuliskan *script Python* lain, dan memanggil/menjalankannya dari dalam *Mission Planner* pada saat diperlukan. Sebagai contoh jika hendak menampilkan kelompok data-data: *attitude* (*altitude, speed, roll, pitch, yaw*) dan status (waktu terbang, sisa battery, jarak ditempuh, jarak ke pangkalan, total jarak) seperti Gambar 3-3, maka kita perlu menuliskan *script Python* seperti pada Gambar 3-4.



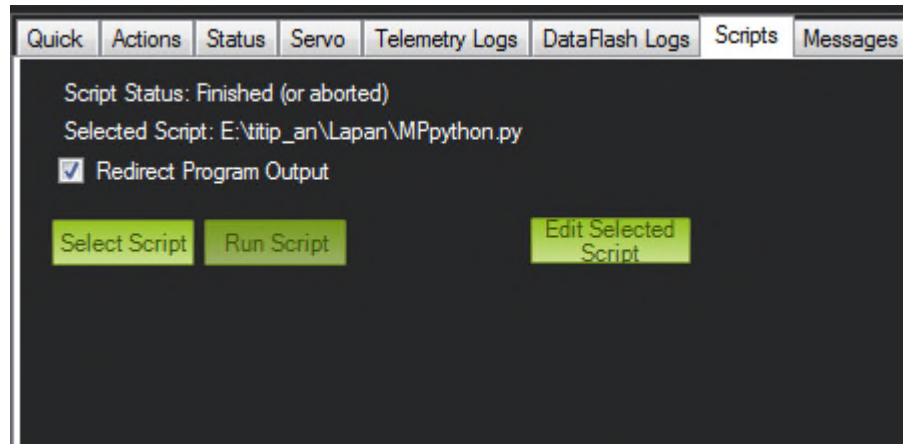
Gambar 3-3. Contoh tampilan data-data tambahan yang dipanggil dari *script Python*



Gambar 3-4. Contoh *script Python*

*Script Python* perlu disimpan dan diberi nama yang sesuai, misalnya *MPpython.py*, *Mppython2.py* dan sebagainya. Untuk dapat menjalankan *script Python* pada *Mission Planner*, pertama-tama kita perlu me-load *script* tersebut.

- Pada layar *Mission Planner* bagian kiri bawah terdapat tab-tab "Quick" "Actions" "Status" dan sebagainya (lihat Gambar 3-5), pilih tab "Scripts".
- Klik tombol "Select Script" lalu pilih file yang berisi *script Python*, misalnya file *MPpython.py*.
- Kemudian klik "Run Script". Maka akan muncul window kecil yang menampilkan jendela keluaran dari *script Python* yang sedang dieksekusi, seperti pada Gambar 3-3 di atas.



Gambar 3-5. Panel pada *Mission Planner* untuk memilih dan menjalankan *script Python*

#### 4. KESIMPULAN

Telah dibahas metode untuk menampilkan lebih banyak data terbang pada *software Mission Planner* untuk dipantau secara bersamaan sesuai kebutuhan. Pendekatan yang diajukan yaitu menggunakan *script Python* yang dijalankan dari dalam *Mission Planner*. Metode ini praktis karena tidak perlu mengubah *source code Mission Planner* dan pengguna dapat mengganti suatu *script* dengan *script* lain sesuai kebutuhan secara real time pada saat pelaksanaan uji terbang. Keterbatasan metode ini adalah tampilan datanya berukuran relatif kecil. Perlu dilakukan penelitian lanjut untuk mengatur ukuran tampilan data agar dapat terbaca lebih jelas.

#### UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih sebesar-besarnya kepada Ir. Errya Satrya atas diskusi yang berharga. Juga kepada Kepala Pustekroket atas fasilitas litbang yang disediakan.

#### PERNYATAAN PENULIS

Penulis dengan ini menyatakan bahwa seluruh isi menjadi tanggung jawab penulis.

#### DAFTAR PUSTAKA

- 1) ..., Laporan Bidang Kendali dan Telemetri Tahun 2014, Pustekroket LAPAN, Bogor.
- 2) ..., Laporan Bidang Kendali dan Telemetri Tahun 2013, Pustekroket LAPAN, Bogor.
- 3) Errya Satrya, et al, 2014, *Proses Perancangan Wahana RKX200-EDF*, Buku bunga rampai Hasil Penelitian dan Pemikiran Ilmiah tentang : Teknologi Pesawat Terbang Tanpa Awak, Roket serta Satelit, Indonesia Book Project, Jakarta.
- 4) Hakiki, et al, 2013, *Controlling X-Plane Flight Simulation Environments from Matlab for RKX-EDF Launch Scenario*, SIPTEKGAN 2013, LAPAN, Bogor.
- 5) Errya Satrya, et al, 2014, *Pengembangan Sistem Visualisasi Dinamika Terbang Wahana RKX-200EDF/TJ Menggunakan Hexapod*, SIPTEKGAN 2014, LAPAN, Bogor.

- 6) ArduPilot Dev Team, *ArduPilot Autopilot Suite*, tersedia di: <http://plane.ardupilot.com/ardupilot>, diakses Mei 2016.
- 7) ArduPilot Dev Team, *Mission Planner Home*, tersedia di: <http://ardupilot.org/planner>, diakses Mei 2016.
- 8) A. Imam, et al, 2014, *State of the Art in Rotorcraft UAVs Research*, International Journal of Engineering Science and Innovative Technology (IJESIT) Vol 3, Issue 1, January 2014.
- 9) Justice Amahah, 2019, *The Design of an Unmanned Aerial Vehicle Based on the ArduPilot*, Georgian Electronic Scientific Journal: Computer Science and Telecommunications 2009 No.5(22).
- 10) ArduPilot Dev Team, *Using Python Scripts in Mission Planner*, tersedia di: <http://ardupilot.org/planner/docs/using-python-scripts-in-mission-planner.html>, diakses Mei 2016.

## DAFTAR RIWAYAT HIDUP PENULIS



### DATA UMUM

Nama Lengkap : ANDREAS PRASETYA ADI  
Tempat & Tgl. Lahir : Surakarta, 7 April 1971  
Jenis Kelamin : Laki-laki  
Instansi Pekerjaan : LAPAN  
NIP. / NIM. : 19710407 198912 1 001  
Pangkat / Gol.Ruang : Penata / IIIC  
Jabatan Dalam Pekerjaan : Peneliti Muda  
Agama : Kristen  
Status Perkawinan : Menikah

### DATA PENDIDIKAN

SLTA : SMAN 1 Surakarta Tahun: 1989  
STRATA 1 (S.1) : Gunma Univ., Japan Tahun: 1995  
STRATA 2 (S.2) : Gunma Univ., Japan Tahun: 1997  
STRATA 3 (S.3) : Tahun:

### ALAMAT

Alamat Rumah : Regensi Melati Mas D2/15 Serpong  
Alamat Kantor / Instansi : Jln Mekarsari, Bogor  
HP. : 08129049412  
Telp. : 08129049412  
Email : and741@gmail.com

### RIWAYAT SINGKAT PENULIS



**ANDREAS PRASETYA ADI, M.Eng**, lahir di Surakarta (Jawa Tengah) pada tanggal 7 April 1971 bekerja sebagai pegawai negeri sipil di lingkungan Lembaga Penerbangan dan Antariksa Nasional (LAPAN), masuk mulai tahun 1989 sebagai karya siswa dan berdinias mulai 1997 menjadi salah satu Peneliti di satuan kerja Pusat Teknologi Roket yang berada di daerah Rumpin, Bogor. Riwayat pendidikan di Gunma *University*, Jepang, jurusan *Computer Science*, S1 lulus pada tahun 1995 dan S2 pada tahun 1997.