

Ground Testing of the FLEXOP Demonstrator Aircraft

Sodja, Jurij; De Breuker, Roeland; Meddaikar, Yasser; Roessler, Christian; Teubl, Daniel; Gyulai, László; Toth, Szabolcs; Nagy, Mihaly; Vanek, Balin; More Authors

DOI

[10.2514/6.2020-1968](https://doi.org/10.2514/6.2020-1968)

Publication date

2020

Document Version

Final published version

Published in

AIAA Scitech 2020 Forum

Citation (APA)

Sodja, J., De Breuker, R., Meddaikar, Y., Roessler, C., Teubl, D., Gyulai, L., Toth, S., Nagy, M., Vanek, B., & More Authors (2020). Ground Testing of the FLEXOP Demonstrator Aircraft. In *AIAA Scitech 2020 Forum: 6-10 January 2020, Orlando, FL* [AIAA 2020-1968] (AIAA Scitech 2020 Forum; Vol. 1 PartF). American Institute of Aeronautics and Astronautics Inc. (AIAA). <https://doi.org/10.2514/6.2020-1968>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.



Ground Testing of the FLEXOP Demonstrator Aircraft

Jurij Sodja^{*1}, Roeland De Breuker¹, Yasser M. Meddaikar², Johannes Dillinger², Keith Soal², Yves Govers², Wolf R. Krüger², Panagiotis Georgopoulos³, Christos Koimtzoglou³, Christian Roessler⁴, Sebastian Koeberle⁴, Julius Bartasevicius⁴, Dániel Teubl⁴, László Gyulai⁵, Szabolcs Tóth⁵, Mihály Nagy⁵, Dániel Balogh⁵, Miklós Jásdi⁵, Péter Bauer⁵, and Bálint Vanek⁵

¹Faculty of Aerospace Engineering, Delft University of Technology, The Netherlands

²Institute of Aeroelasticity, German Aerospace Centre (DLR), Germany

³INASCO Hellas Co., Greece

⁴Institute of Aircraft Design, Technical University of Munich, Germany

⁵SZTAKI - Institute for Computer Science and Control, Hungary

Ground testing campaign conducted on the FLEXOP demonstrator aircraft is presented in this paper. The conducted tests are grouped in structural, flight system and integration tests. Along with the description of the test setup and test execution, the main findings and conclusions are also given. The structural tests comprise the static, ground vibration and the airworthiness test. The static and the ground vibration tests were used for structural characterisation of the manufactured wings and airframe as a whole. Assessment and calibration of the Fibre Brag strain sensing system for wing shape and load reconstruction is also presented within this context. The airworthiness test is used to demonstrate the structural integrity of the manufactured wings under specified limit loads. Within the context of the flight system tests, the main components of the on-board autopilot hardware-software system are briefly introduced including the signal data flow from the RC transmitter to the aircraft controls, the functionality of the baseline autopilot software and the communication with the ground station. All of these components are integrated into the hardware-in-the-loop environment, which is also briefly introduced along with the servo motor identification and the hardware delay measurements. The measured hardware delay was considered in the design of the baseline and flutter controllers. The flutter controllers were tested together with the baseline controller in the software-in-the-loop environment. System integration tests are presented last. In this context the airbrake, the engine, the compatibility of electronic components, the range and the taxi tests are presented.

I. Introduction

THE Flutter-free Flight Envelope Extension for Economical Performance Improvement (FLEXOP), was an European research project with the goal of developing and demonstrating passive load alleviation using aeroelastic tailoring and active flutter control methodologies. Both, passive load alleviation as well as active flutter control are to be demonstrated and assessed in flight using an unmanned aerial vehicle(UAV).¹ Various aspects of the project leading up to the testing of the aircraft have already been presented.²⁻¹⁵ In order to conduct the flight tests in a safe and efficient manner a number of ground tests were performed. The ground tests performed within FLEXOP can be divided in three categories: structural tests, flight systems tests and the integration tests.

Structural tests comprise an important segment of ground tests that were performed during the FLEXOP project. The static and ground vibration tests are fundamental in assessing and improving the fidelity of the numerical models used during the aircraft design process and in the process of developing various control laws

^{*}Senior researcher, Faculty of Aerospace Engineering, Aerospace Structures and Computational Mechanics, Kluyverweg 1, 2629 HS Delft; J.Sodja@tudelft.nl

used by the autopilot and the flight control computer. The static and the GVT tests were also important for the calibration of the strain-sensing Fibre Bragg Gratings (FBGs) installed in the wings in order to provide information on loads and structural deformations during flight testing. This information is crucial in demonstrating the passive load alleviation capabilities of the developed wings. Finally, the airworthiness test represents an important milestone among the structural tests. Successful completion of this test demonstrates structural integrity of the airframe and clears it for flight.

An important part of ground testing are also software-in-the-loop (SIL), hardware-in-the-loop (HIL) and SIL-HIL tests. SIL implements both aircraft simulation and all controllers in software and runs them together to evaluate basic functioning of controllers. If the SIL test is passed SIL-HIL test can be an intermediate step towards HIL test when part of the control is implemented on the real hardware and part of it runs in software. Finally, HIL test is the last stage before real flight testing. In this case all of the controllers run on the real hardware and only the aircraft and sensors are simulated in software. Besides introducing the autopilot hardware-software system and the HIL simulation together with servo motor identification this paper focuses on SIL evaluation of the flutter controllers designed by DLR and SZTAKI.

Finally, a significant part of ground testing is represented by the integration tests. These tests are crucial in ensuring the compatibility among all the systems integrated in the aircraft. Within this scope it is also important to demonstrate that all the systems meet all of the critical functional requirements. The most important integration tests performed within the FLEXOP project were the airbrake, and the engine test, a test of compatibility of electronic systems, and the range and taxi test.

While this paper is focused on the ground testing campaign performed within the FLEXOP project it has been presented together with papers on flight testing of aeroelastically tailored wings¹⁶ and active flutter mitigation testing performed on the FLEXOP aircraft.¹⁷

The structure of the paper reflects all the three major test groups. Section II introduces the static and the GVT test, followed by the presentation of the strain-based wing shape and load reconstruction and the airworthiness test. Next, in section III, the signal data flow, autopilot loops and telemetry system of the autopilot hardware are first introduced. Next, the servo motor identification, the HIL simulation and the measurement of the hardware delay in HIL are detailed. The results of the SIL test of two flutter controllers (DLR and SZTAKI) run together with the baseline controller are presented at the end of the section. Section IV details all the major integration tests performed within the project. First, the airbrake test and the engine test are presented, followed by the presentation of the compatibility test of electronic systems, the range and the taxi test. Finally, the most important outcomes and conclusions are summarised in section V.

II. Structural test

Within the FLEXOP project three different wing designs were developed: the reference wing, (-0), the flutter wing, (-1), and the tailored wing (-2). All three wings were subjected to the static, ground vibration and the airworthiness tests. However, processing of the data pertinent to the (-1) wing is still in progress, hence only the results pertinent to (-0) and (-2) wing are presented here. More details on the design of the (-0) and the (-2) wing are reported in Sodja et al.¹¹ and Meddaikar et al.¹⁰

II.A. Static test

Comparison of numeric and experimental results in aeroelastic problems often show significant differences. In addition to modelling assumptions, the modelled aeroelastic response can differ from the measured response due possible discrepancies between the numeric model and the physical experimental model. The main differences originate from the mass and stiffness distribution on the structural side and from the geometric deviations in shape of the physical model relative to the numeric model on the aerodynamic side. The static test was performed to assess the actual stiffness properties of the manufactured wings relative to the stiffness properties obtained during the wing design process. Within this scope, the static test results were also used together with the GVT results to update the wing structural models used in subsequent development tasks within the FlexOp project, such as derivation of flight dynamic model and aircraft aeroelastic stability assessment. Finally, the static test results were also used to calibrate the FBG sensors installed in the wings.

Therefore, the tested wings were clamped in a dedicated test stand provided by TUM. In order to achieve representative boundary conditions at the root of the wing, both port and starboard half of the wing were attached to the test stand and connected in the middle as if they were mounted on the fuselage. Prescribed

loads were then applied to both halves in a symmetric fashion.

II.A.1. Test setup

The overview of the static test setup is shown in Fig. 1. The tested wing was clamped in a dedicated test stand. Representative boundary conditions at the root of the wing were ensured by mounting both port and starboard half of the wing in the same way they were mounted on the fuselage. The test stand and the fixture details are depicted in Fig. 2.

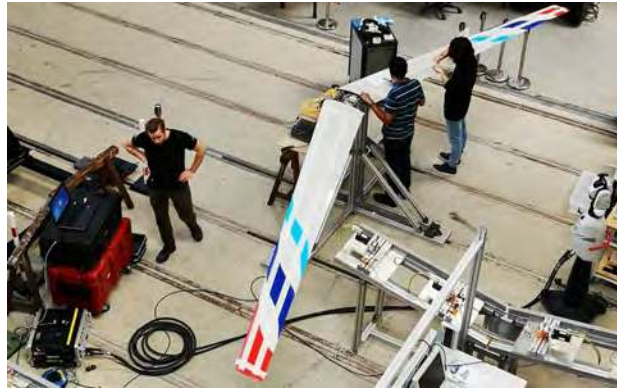
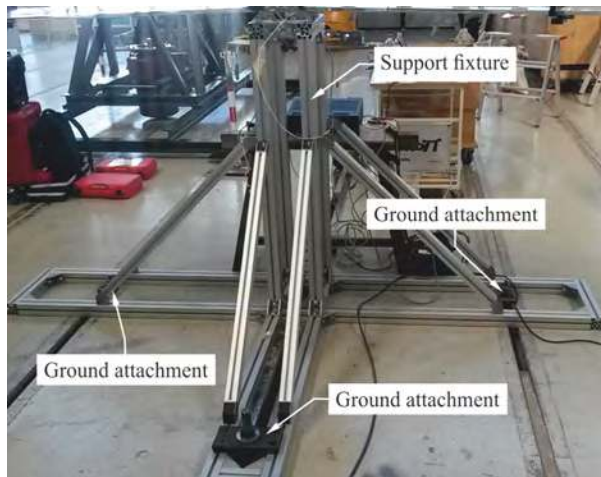
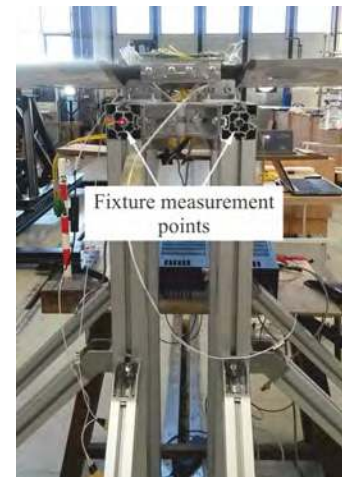


Figure 1. Static test setup



(a) Fixture detail



(b) Support fixture measurement points

Figure 2. Test stand

The wing deformation was measured in a fixed laboratory coordinate system using the laser tracking system. Hence any motion or deformation in the test stand corrupts the measured wing deformation. Rigid-body-motion of the test stand was prevented by bolting it to the ground as shown in Fig. 2a. Moreover, the deformation in the fixture points was also monitored by adding two additional measurement points on the fixture itself as shown in Fig. 2b. Finally, both port and starboard halves of the wing were measured in order to detect any asymmetry in either the test setup or in the manufactured wing halves.

The selected loads were applied to both wing halves in a symmetric fashion using load clamps such as the one shown in Fig. 3a. The load clamps were 3D printed to match the loft shape of the wing in order to uniformly introduce the applied loads over the entire contact surface of the clamp. Moreover, the load clamps were always positioned over a rib within the wing to support the wing skin and to prevent any damage due to excessive deformation in the vicinity of the clamps.

Shear and torque loads were applied by hanging weights from the load introduction point at the tip of the wing as depicted in Fig. 4. Both positive and negative shear loads were applied by flipping the wing

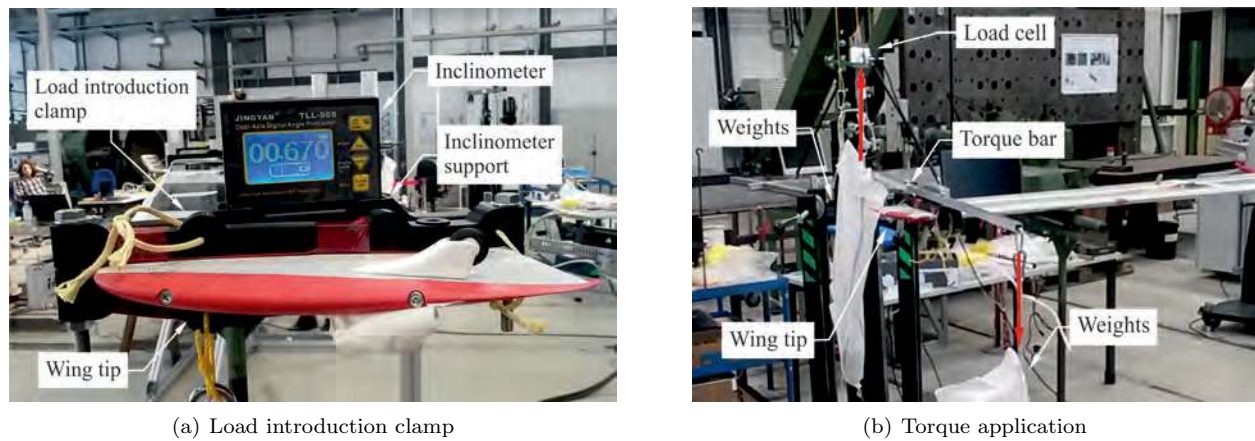


Figure 3. Load application

from upside-up to upside-down orientation. To be able to apply sufficient amount of torque, a torque bar was mounted on top of the load clamp as shown in Fig. 3b. A set of maximum shear and torque loads were determined numerically for each loading direction to prevent damaging the wing structure during the test. The maximum shear force in the wing upside-up and upside-down configuration was limited to 30 N and 50 N, respectively. The maximum torque was limited to 50 Nm regardless of wing orientation.

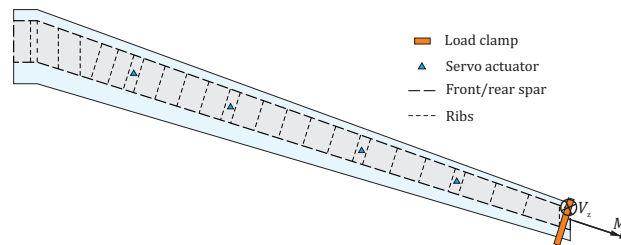


Figure 4. Static test setup

II.A.2. Results

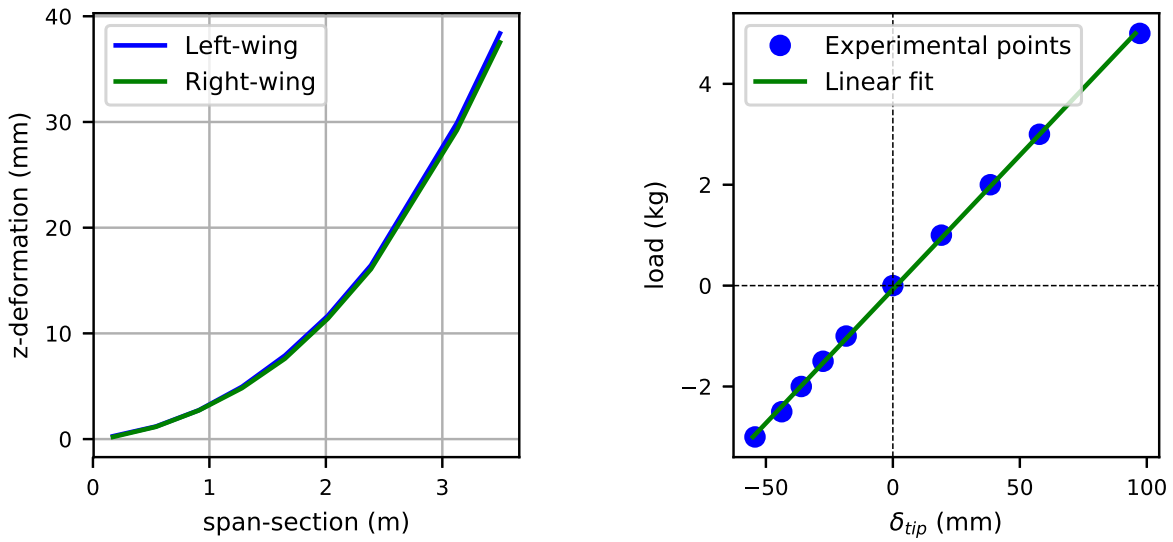
As already mentioned both wing halves were loaded and measured during the static test. Fig. 5a shows the out-of-plane deflection for wing (-0) in upside-up configuration for both wing halves. The applied tip load was 30 N. The response of both wing halves is very similar implying very small manufacturing deviations between the two wing halves. Similar level of similarity between individual wing halves was observed also in the case of (-2) wing.

An example of wing tip out-of-plane deflection as a function of applied load for the port half of the wing (-0) is shown in Fig. 5b. With the loads being well within the linear regime of the structure, an expected linear relation between the displacement and loads is observed as indicated by the linear fit. Similar linear behaviour was also observed for the wing (-2).

The deformation results were used to update the finite element model (FEM). Fig. 6 shows the comparison between the initial and the updated Nastran FEM and the experimental results for a tip load of 30 N for the wing (-0). The initial FEM appears too stiff in comparison to the measured wing. Hence, the FEM was updated by applying a stiffness correction factor uniformly across the entire wing. The updated FEM model shows a very good agreement with the measured displacements. More details about the model updating approach are given by Takarics et al.¹⁷ The same updating procedure was also successfully applied to wing (-2).

II.B. Ground Vibration Testing (GVT)

The GVT is a modal test which is generally used to identify resonance frequencies, damping ratios and mode shapes of aircraft structures on the ground before the first flight of a prototype. The results are either



(a) Comparison between port and starboard wing halves for a tip load of 30 N

(b) Wing tip displacement-load curve

Figure 5. Wing (-0) out-of-plane displacement characteristics

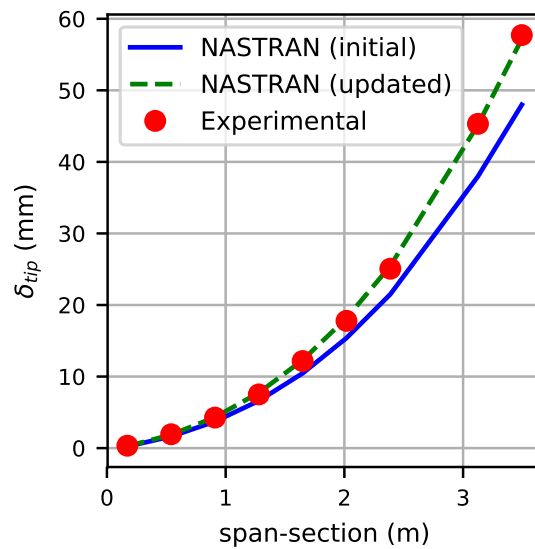


Figure 6. FEM update: comparison of the spanwise out-of-plane displacement for a tip load of 30 N

software was used to generate the input signals (random and sine sweep), perform signal processing and modal model correlation. LMS Test.Lab was used together with SCADAS III data acquisition systems to measure the data and perform modal analysis.

An extensive modal model was identified consisting of 44 modes from 0.4 Hz to 102 Hz for wing (-0) and 28 modes from 0.3 Hz to 94 Hz for wing (-2). The modal model information for the first 20 modes, provided by the GVT for the wing (-0) is shown in Fig 9. This includes mode names, frequencies, damping ratios, generalized mass, normalization point, generalized force, Mode Indicator Function (MIF), Mean Phase Deviation (MPD), Mean Phase Correlation (MPC) and the analysis run from which the main mode was selected. The orthographic projection of the mode shape number #6 2n wing bending, is shown in Fig 10. Investigations were also conducted into the non-linearity of the structure. It was found that the 1n wing inplane mode varied by 24 %, which has implications for the flight testing campaign, and should be monitored in future flight testing.


Modal Data FLEXOP							 Deutsches Zentrum DLR für Luft- und Raumfahrt e.V.			
C1_FLEXOP										
#	Name	Freq [Hz]	Damp [%]	GenMass [kgm ²]	NormPoint	GenForce [Nm]	MIF	MPD [°]	MPC [%]	AnalysisName
1	a/c_roll-a	0,418	3,29	28,4	810512+Y		749	16,5	93,2	C1_FUS_RY_B0L0_1_a
2	a/c_yaw-a	0,575	5,72	3,8	810512+Y		392	32,6	74,3	C1_FUS_RY_B0L0_1_a
3	a/c_foraft-s	0,621	0,62	390,1	820523+Z		824	20,8	86,5	C1_FUS_RZ_B0L0_1_rigidbody
4	a/c_heave-s	0,842	3,51	8,8	10501+Z		944	6,2	98,5	C1_WING_RXZ_B0L0_1_a
5	a/c_pitch-s	0,949	2,11	6,9	50101+Z		914	4,6	99,5	C1_FUS_RZ_B0L0_1_rigidbody
6	2n_wing_bend-s	3,369	2,02	3,7	10502+Z	3,96	992	0,6	100,0	C1_FUS_SZ_B1L2_1_a
7	3n_wing_bend-a	8,275	0,82	2,4	10504+Z	2,38	992	0,8	100,0	C1_WING_AXZ_B1L1_1_a
8	1n_wing_inplane-a	8,877	7,26	12,4	820512+Y	15,90	925	6,0	99,0	C1_FUS_AY_B1L2_1_a
9	4n_wing_bend-s	12,116	1,32	2,1	10504+Z	2,52	984	2,5	99,8	C1_WING_SXZ_B1L1_1_a
10	vtail_rock-a	17,316	4,56	0,4	420303+Z	1,80	972	4,5	99,6	C1_FUS_AY_B1L3_1_b
11	2n_wing_inplane-s	19,260	1,17	3,3	10501+X	1,61	958	8,4	98,9	C1_WING_SXZ_B1L1_1_b
12	5n_wing_bend-a	23,004	1,38	2,7	420303+Z	0,62	912	6,2	98,9	C1_WING_AXZ_B1L1_1_b
13	vtail_lat-a	23,238	3,33	5,3	140503+Z	5,77	794	15,2	93,6	C1_FUS_AY_B1L3_1_b
14	2n_fus_yert-s	24,883	4,09	2,4	420303+Z	4,32	793	9,4	97,7	C1_FUS_SZ_B1L2_1_b
15	6n_wing_bend-s	28,381	2,19	2,7	140503+Z	2,70	734	14,6	93,9	C1_WING_SXZ_B1L1_1_b
16	eng_lat2-a	29,070	1,78	15,0	420303+Z	1,78	848	15,6	94,2	C1_FUS_AY_B1L0_1_b
17	vtail_foraft-a	29,453	3,21	19,5	320303+Z	13,72	803	14,0	95,1	C1_FUS_AY_B1L3_1_b
18	2n_vtail_vert-s	33,006	2,73	0,5	420303+Z	2,28	924	6,9	98,8	C1_FUS_SZ_B1L2_1_c
19	3n_wing_inplane-a	40,536	0,76	5,9	230304+Z	2,76	682	31,4	78,9	C1_WING_AXZ_B1L1_1_c
20	ail1-a	40,971	1,63	0,7	210104+Z	0,21	873	12,1	96,5	C1_FUS_AY_B1L0_1_c

Figure 9. Modal model from configuration 1 [(-0) wing]

Finally testing was also conducted to validate the internal inertial measurement units (IMUs) built into the wings which were developed by SZTAKI. Twelve acceleration signals in +Z direction were recorded during all GVT testing and analysed using DLR in house algorithms and the Stochastic Subspace Identification (SSI) algorithm for Output-only Modal Analysis (OMA). The results showed a high correlation with external GVT sensors in frequency and damping ratio for the modes dominated by vertical motion of the wings. This is an important pre-cursor for on-line monitoring of flutter critical modes during FLEXOP flight testing.

II.C. Strain-based wing shape and load reconstruction

The load alleviation and flutter suppression mechanisms developed in the context of FLEXOP are to be assessed via the calculation of in-flight wing shapes and loads; the quantities of interest are vertical deflection, w , and torsional twist ϕ , as well as bending moment, M , shear force, V and torsional moment T , in terms of shapes and loads respectively. The baseline (-0) and tailored (-2) wings are equipped with Fibre Bragg Grating (FBG) strain and temperature sensors, to provide the reconstruction data.

Having established the theoretical methods for wing shape and load calculation and validated their applicability using simulation data by Roessler et al.,¹ ground testing data was collected for the calibration and validation of the methods with experimental FBG data, prior to flight testing.

II.C.1. FBG System Layout

The strain FBGs form two sensor configurations based on their positioning on the skins of the wings, along the axis (both pairs) and spars (-2 only) respectively, as shown in Fig. 11. The first configuration is designed

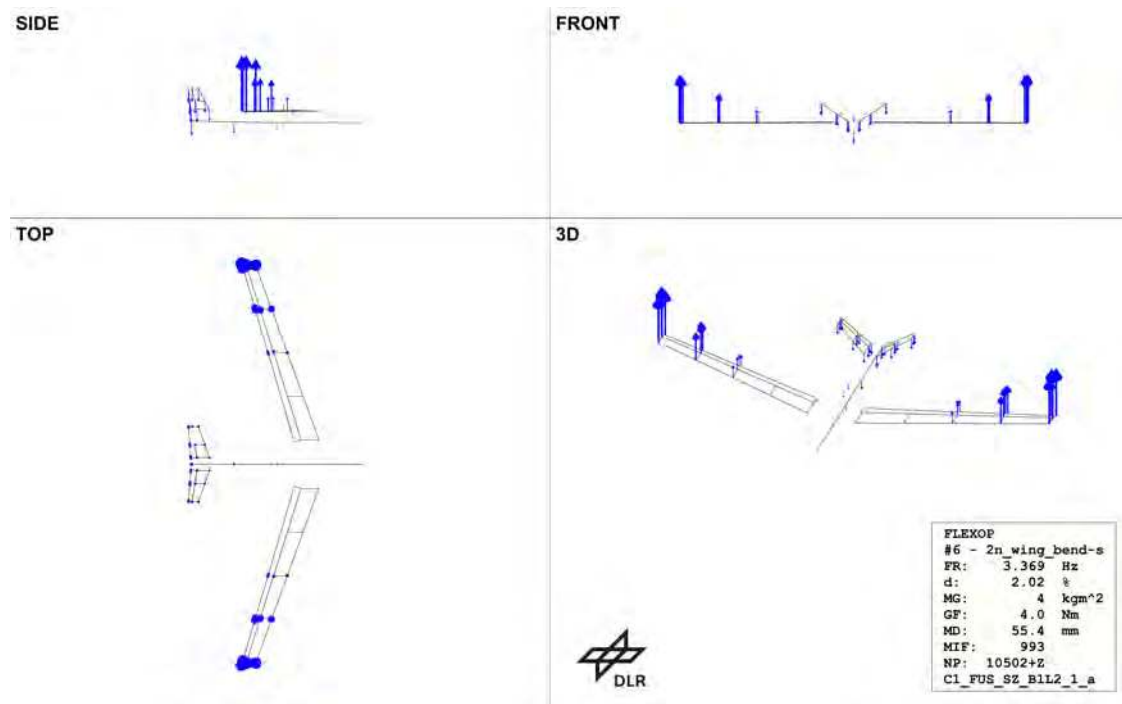


Figure 10. Mode shape for 2n wing bending

to measure both shapes and loads, using 20 rosettes at both skins. The second configuration, consisting of 8 FBGs per skin, per spar, is an alternative approach to shape sensing alone, using significantly fewer FBGs. In addition, 10 temperature FBGs per skin are included along the wing axis for correction from thermal strains.

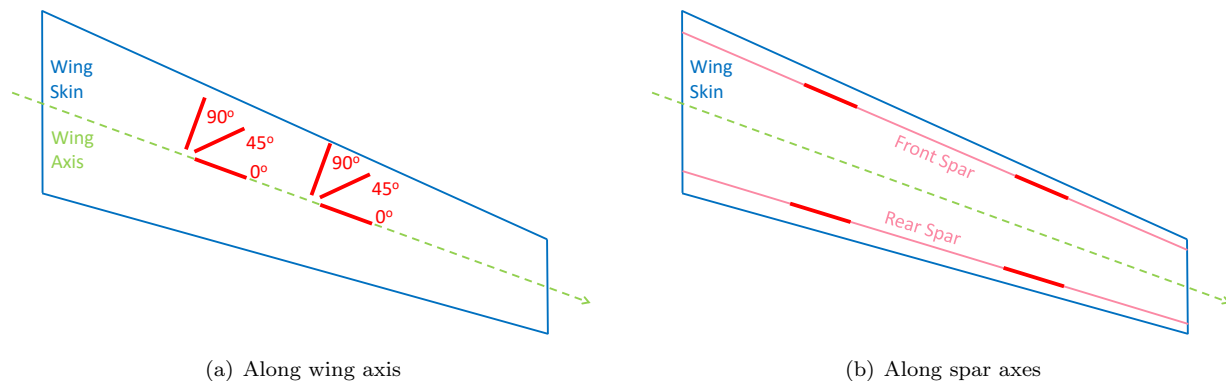


Figure 11. FBG configurations used in the (-0) and (-2) wings

II.C.2. System calibration

The static test provided the necessary calibration data for the calculation of the load and shape sensing parameters. Simulation studies indicated that a small set of load cases of pure bending and torsion applied at the wing tip provided adequate calibration data; hence, the applied load was changed incrementally, increasing from zero to the limit shear force or torsion moment and back to zero, as shown in Fig. 12 for bending and torsion.

The unforeseen presence of a significant low frequency, non-periodic “drift error”, complicated the calibration of FBGs and consequently the relevant calculations. Thus, an alternative approach was developed, linearly relating the change in applied shear force and torsional moments, ΔV and ΔT , and the resulting

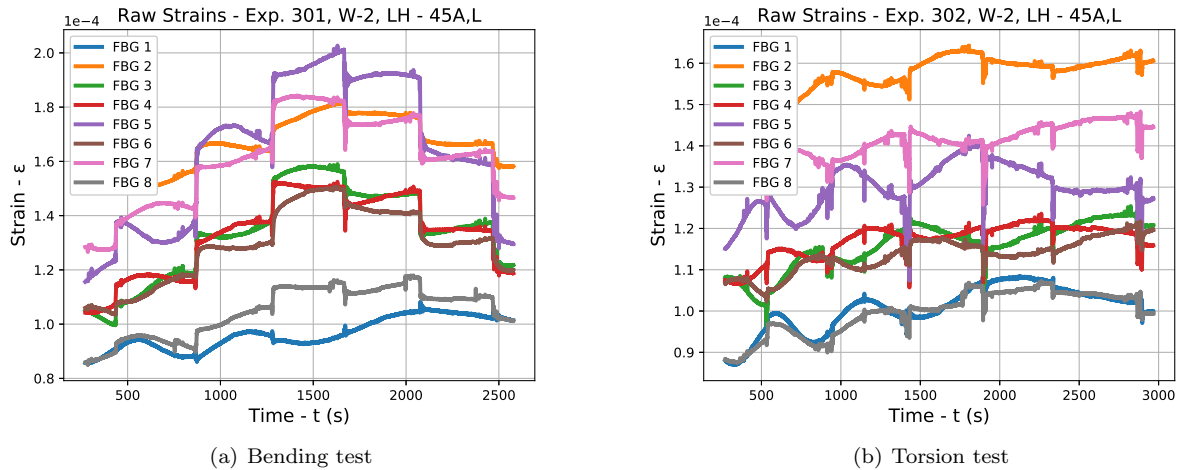


Figure 12. Raw strains measured by FBG fibre in wing (-2) during the static tests

change in strain, $\Delta\epsilon$, between load cases; this is shown graphically in Fig. 13.

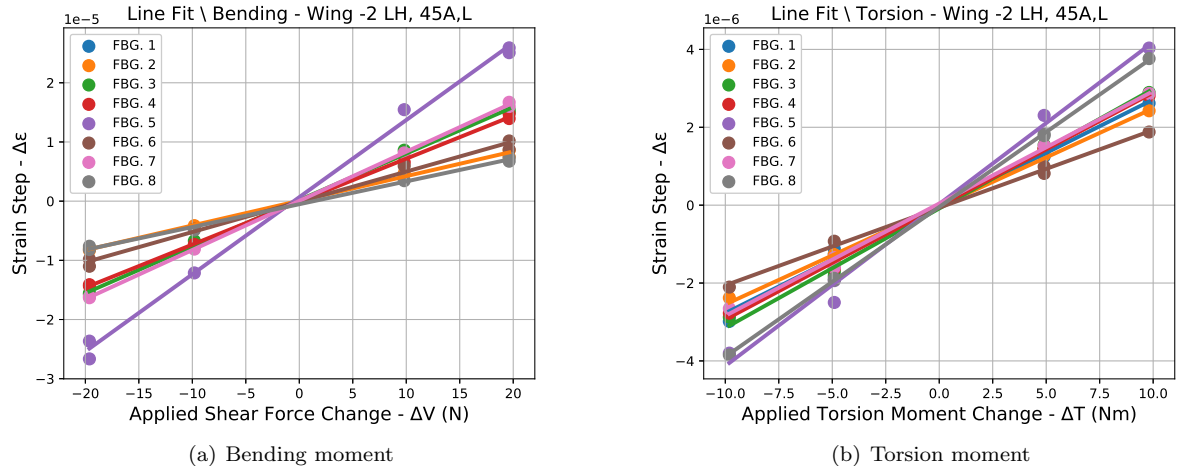


Figure 13. Linear fit between applied loads and resulting strain in a fibre

The fitted lines were used in the calculation of the load and shape sensing parameters. Linear operators relating M and T to local strains were calculated directly; V can be calculated as the derivative of M along the wing axis. Unlike loads, w and ϕ are related to the complete strain distribution, simulation studies showing that an extensive set of calibration load cases is required to capture the shape sensing parameters. Nevertheless, linear conversion operators between simulated and experimental strains were calculated. This allowed the execution of a simulated static test and the calculation of the shape sensing parameters using simulated strains; much like a coordinate transformation, the conversion operators can be used to pair the parameters with experimental strains.

II.C.3. Validation

The 2.5 g airworthiness test case provided useful strain data, emulating aerodynamic loads. Therefore, the accuracy of the calculated wing shapes and loads in this load case was deemed indicative of the capabilities of the methods with regards to aerodynamic loading during flight. The validation results are presented in Fig. 14 for both FBG configurations of the left hand, (-2) wing.

It is evident that bending related quantities such as M , V and w are calculated accurately using the axis FBG system; V being the derivative of M is more sensitive to measurement noise, but can still be found with

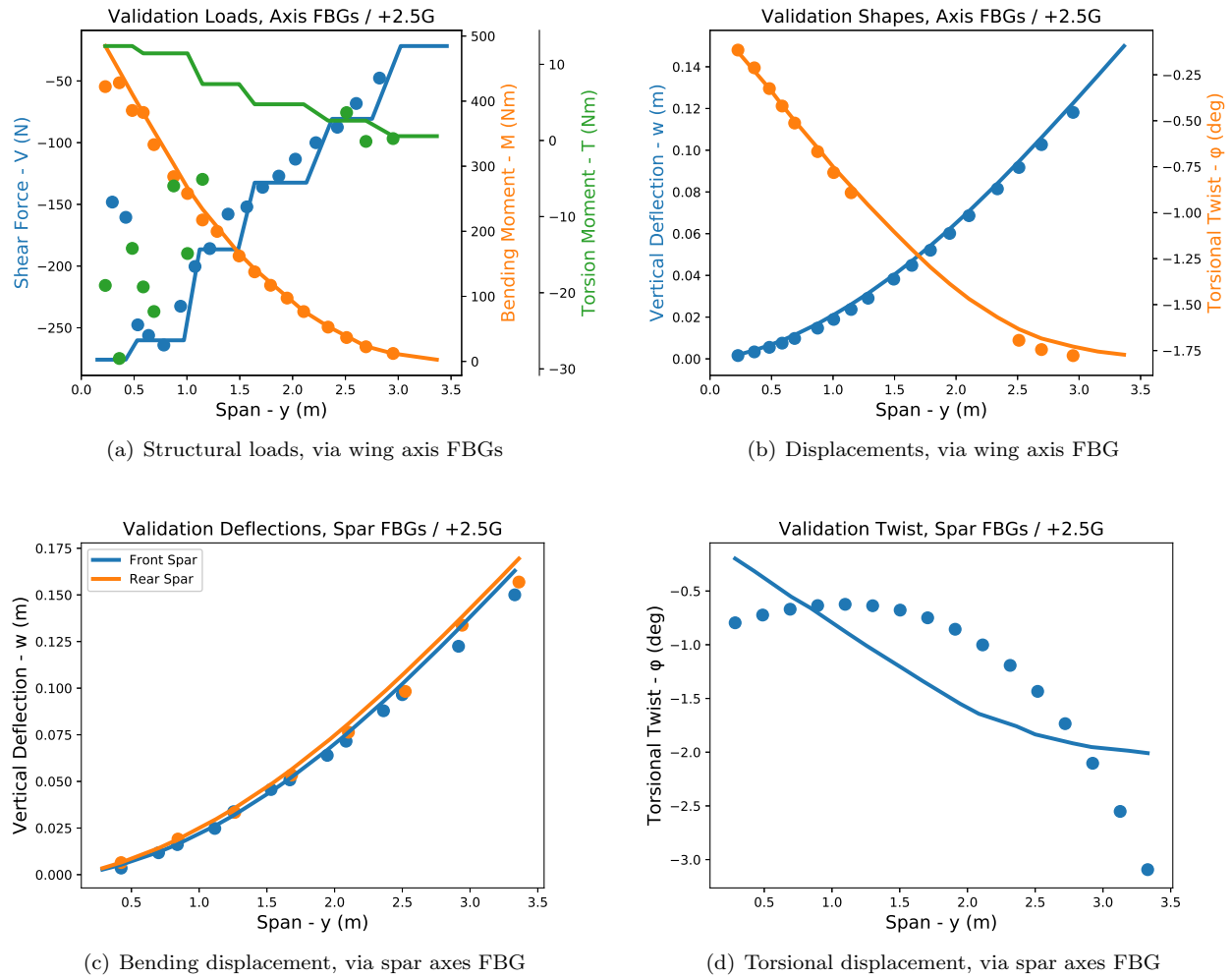


Figure 14. Validation results at 2.5 g loads; reference values (lines) versus FBG reconstructed values (dots)

good accuracy. However, in the spar FBG system, the use of single FBG per measurement location causes increased sensitivity to measurement noise and also a poorer calculation of the operator relating simulated and experimental strains. This results to a less accurate calculation of each spar's deflection shape w and consequently a poorer reconstruction of the wing's twist ϕ .

On the other hand, the calculation of torsion related quantities proved more complicated. Being of significantly smaller order than M , the calculation of T was very sensitive to measurement noise, as also foreseen by simulation studies. The calculation of ϕ was performed in two very different ways for each FBG configuration, yielding different results. In the case of the axis FBGs, the fact that ϕ is coupled to bending due to the swept geometry of the wings in conjunction with the use of the complete strain distribution rather than local strains alone, results in a more stable mathematical operation and hence an accurate calculation. On the contrary, the spar FBG configuration utilizes the individual w distributions of the two spars and their respective difference to calculate ϕ ; though theoretically possible for very accurate w distributions, this method is quite unstable for the small order of ϕ , up to 2 deg, making it less effective.

The GVT provided a set of dynamic strain data, to assess the capabilities of the measurement system under flight conditions. Specifically, it was used as sample data for the development of the processing methods necessary to bring raw FBG strains to a form compatible with the load and shape sensing parameters. Given the lack of reference data for the loads associated with the GVT, the results cannot be utilised effectively for validation purposes. Nevertheless, the calculated wing shapes and loads corresponding to the 1st bending mode of the left hand (-0) wing are presented in Fig. 15, confirming the ability of the system and of the developed methods to capture dynamic results, ultimately validating their use during flight testing.

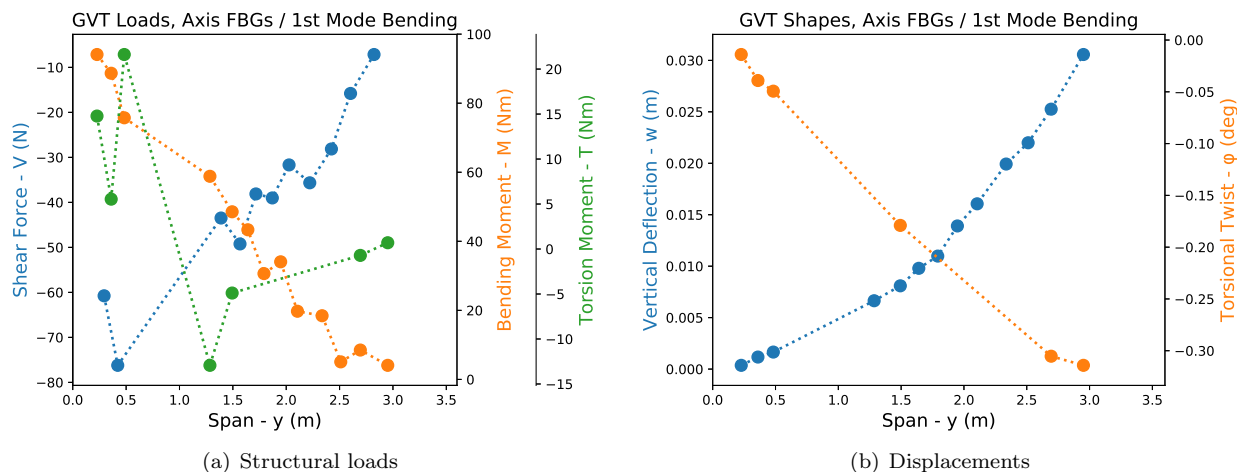


Figure 15. FBG reconstructed peak loads and displacements during the GVT test pertinent the 1st Bending Mode

II.D. Airworthiness test

In addition to the static tests for model validation, a structural airworthiness test is performed in order to confirm that the wing can sustain the design limit loads, namely 5g and -2g. The test-stand with the unloaded and the loaded wing is shown in Fig. 16. The equivalent limit loads are introduced on the wings by hanging sand bags at six span-wise locations on both wing-halves. In the case of the -2g loads, the wings are flipped over before applying the loads.

The equivalent loads are derived by imposing the deflected shape of the wings under the limit loads to the FE model and extracting back the reaction forces. The deflected shape is obtained beforehand from an aeroelastic trim analysis. This way the loads to be applied, the resulting shear force and the bending moment along the span are recovered, as shown for example in Fig. 17 for a 1g equivalent load. One can observe that the equivalent shear force changes in discrete steps due to the fact that the forces are applied at discrete locations along the span of the wing. Nevertheless, the equivalent bending moment follows the bending moment in flight very faithfully. The airworthiness test shows that the wings can sustain the limit loads without any detectable structural damage and is therefore deemed safe to operate from a structural stand-point.

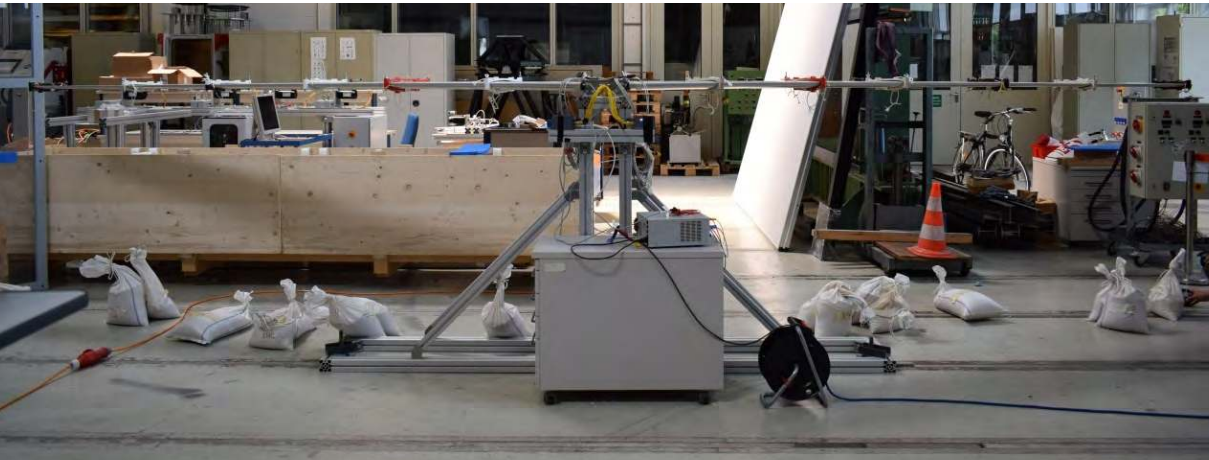


Figure 16. Wings mounted on test-stand for structural airworthiness test - unloaded (above) and loaded (below)

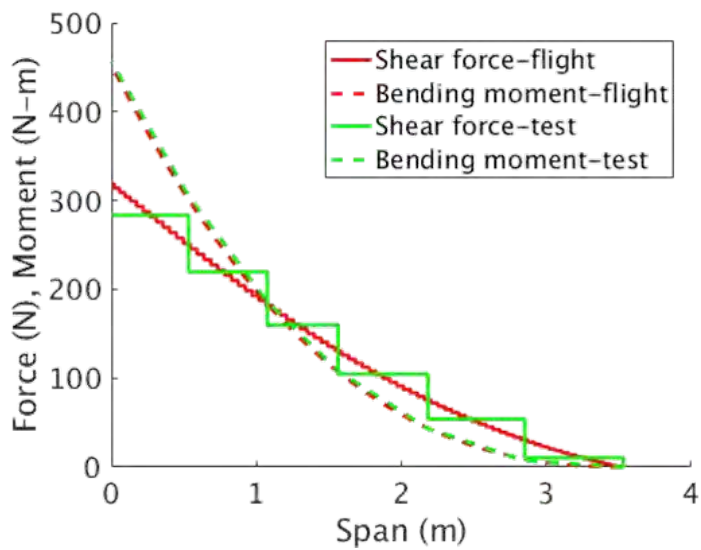


Figure 17. Derivation of 1g - equivalent loads for static testing

III. Flight systems test

III.A. Overview of signal data flow

Signals from pilot and autopilot reference sources are going through various types of transformations before getting to the actuators. Furthermore, controllers, actuator drivers and communications are in different unit and scale. Many of these data also necessary to be understandable for human eyes on the ground control station (GCS). The data flows through the transmitter and the FCC to the actuators and additionally through GCS to human supervisor.

All the following transformations inside the FCC are organized around the autopilot control loops. The embracing transformations convert the reference signals between the implementation – with low-level units and physical limitations – and high level control with convenient SI units. Therefore autopilot development and testing becomes easier and safer.

The main logic and the interfaces of the FCC are shown in Fig. 18. The unit which can select the source of reference signals called RX-MUX. It has two transmitters to choose from (Fig. 19), and the autopilot also can be used (Fig. 20). The flightHAT unit is an interface panel used for connecting sensors, GCS, RX-MUX units and a Raspberry Pi to each other. Raspberry Pi runs a real time kernel with three threads using 5 ms sampling time yielding system frequency of 200 Hz.

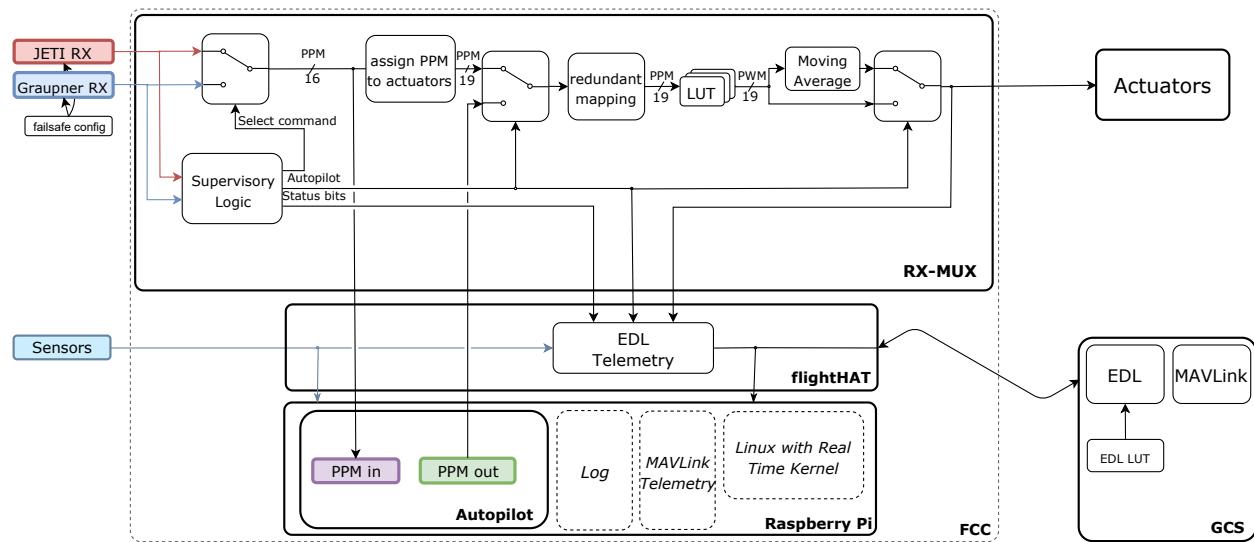


Figure 18. Main signal route in FCC

III.A.1. Radio Transmitters

The transmitter in the pilot's hand is the easiest and fastest way to modify actuator signals online. JETI transmitter has many built-in options. There are five main reference signal, each for an actuator group: like aileron, elevator, rudder, break and throttle. The reference signals are transformed with piece-wise linear gains, because up and down actuator deflection has different torque rate. Next in line is an additional trim value to precisely set actuator values for straight-and-level flight. Human mind is mostly thinking in horizontal and vertical motion, Rudder and Elevator, but these signals can't command directly a V-tail aircraft, therefore a mixer transform them to left and right tail surface position.

There are five logical switches, commanding the parachute, engine on/off, controller in command, autopilot and flight state. Autopilot and flight state are different, they are 3 state switches. Autopilot has 1 manual, and 2 autopilot state. Flight state switch has also 3 state, take off, flight, landing. And depending on its value, it's changing some ailerons symmetrically for near-surface maneuvers. All the signals are represented in milliseconds for the PPM signals. The transmitter communicates with the receiver on analog radio channels and the receiver provides the PPM signal series for the RX-MUX.

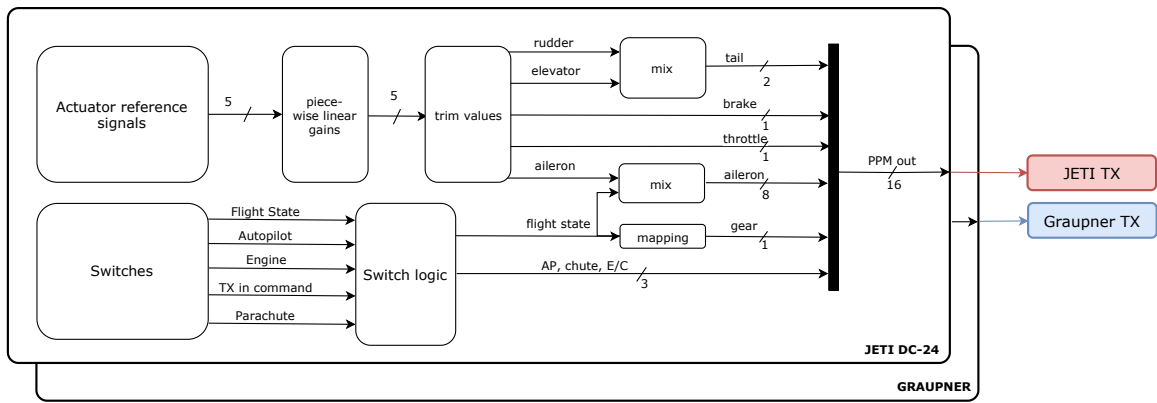


Figure 19. Reference signals from transmitter

III.A.2. Flight Control Computer

On the aircraft the servo actuators are controlled with 333Hz PWM signals. The PWM signals are created by RX-MUX units, which are located on the top of the Flight Control Computer (FCC). The PWM signals are generated based on the PPM signals of RC transmitter or the autopilot. The values of PWM duty cycles which cause the required motion of flight control surfaces are calculated using lookup tables (LUT). Lookup tables prevent the actuators from getting too high, too low or invalid reference signal from autopilot, or the human pilot and it makes possible to setup desired trajectories for each actuator, like signal inversion or linear input exponential output trajectory.

RX-MUX unit is designed to be a safety critical device, it can work independently from other parts of the FCC and give full control to the pilot in a critical situation. Controlled actuators are allocated to 3 RX-MUX units – see *redundant mapping* on Fig. 18) – and the aircraft can still be controlled by only one unit in a critical situation. Two RX-MUX units are responsible for six and one for seven servo actuators, summing 19 PWM channels together. Regarding PWM generation, the softwares running on 3 RX-MUX units are different just in their LUT values. The different LUT values symbolized with a batch consisting 3 layers in Fig. 18.

To create lookup tables, movement of control surfaces were measured. During the measurement, a device gave PWM signal to the actuators with known duty cycles. The duty cycle values which caused no deflection and maximum deflections in both directions were recorded. These minimum, middle and maximum values give the LUT and values between these are generated with linear interpolation. This PWM generation method was tested in PC first. The algorithm got different inputs inside the allowed interval, higher and lower values than minimum and maximum as well. The output values were observed and after it worked as required, the source code was put into the firmware and tested on the aircraft. During the aircraft test, the angles of flight control surfaces were measured and verified that the given PWM signals cause the required movement.

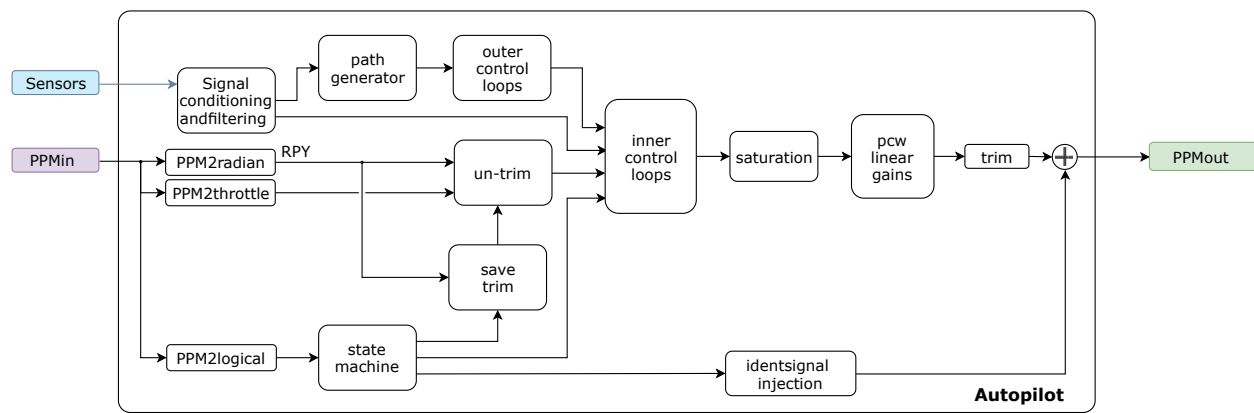


Figure 20. Autopilot signal conversions

Human maneuvers aren't stable enough for scientific flutter research. For that reason possessing an autopilot function is inevitable. Several type of autopilots have been implemented already. The autopilot loops are schematically shown in Fig. 20. The different loops for baseline trajectory tracking control are summarized below together with their inputs and outputs:

- **Guidance loop**

- *Inputs*: waypoints, reference altitude and airspeed, actual position and ground velocity
- *Outputs*: reference course angle, altitude and airspeed

- **Outer loops**

- **Course angle tracking (lateral control)**
- *Inputs*: actual and reference course angles
- *Outputs*: reference roll angle
- **Altitude tracking (longitudinal control)**
- *Inputs*: actual and reference altitude
- *Outputs*: reference pitch angle

- **Inner loops**

- **Airspeed tracking (longitudinal control)**
- *Inputs*: actual and reference airspeeds
- *Outputs*: throttle position
- **Pitch tracking (longitudinal control)**
- *Inputs*: actual and reference pitch angles, pitch rate
- *Outputs*: elevator command
- **Roll tracking (lateral control)**
- *Inputs*: actual and reference roll angles, roll rate
- *Outputs*: aileron command
- **Sideslip control (lateral control)**
- *Inputs*: angle of sideslip
- *Outputs*: rudder command

All control loops also implement saturation of in- and outputs together with anti-windup compensation and the implementation of bumpless transfer when switching from manual to auto and auto to manual modes. Details of baseline control can be found in ¹⁸.

III.B. Telemetry system

The use of telemetry services is necessary during ground testing and also in flight with the FLEXOP demonstrator aircraft. The flight staff in the ground control station (GCS) van must keep an eye on the key variables of the aircraft while it is in the air. If some value, like the temperature of a servo exceeds a threshold, or the fuel level drops down the pilots must be informed to land the aircraft. There are two wireless diagnostic tools - EDL and Mavlink - implemented on the aircraft right now. These tools make up for the telemetry of the system, so the aircraft can communicate using only these protocols.

III.B.1. EDL (Engineering Data Link)

During a flight test, a strict switch-on procedure is followed. EDL helps to verify the airworthiness of the demonstrator before every flight test: the crew can test the main subsystems before takeoff and monitor the airplane's status in the air.

The Engineering Data Link (EDL) is mainly used to receive data about the internal components of the UAV. This is implemented in MATLAB, with the help of the "GUI layout toolbox". At the beginning, the user can pick a wing configuration, since there are multiple wings built for the aircraft. This can be seen in Fig. 18 as EDL LUT and it contains scaling values for PWM to degree and SHM (servo health measurement) to degree conversion for each control surface. After selecting one, the main dialogue centre window opens, this can be seen in Fig. 21.

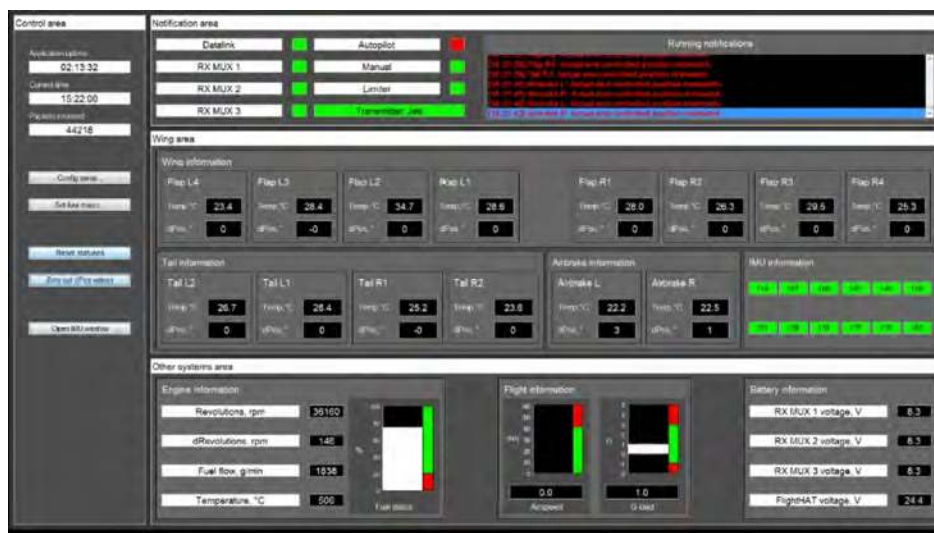


Figure 21. EDL main window during a taxi test

The window includes four main sections, namely: *Control*, *Notification*, *Wing* and *Other systems* area. The control area contains some basic information like the application uptime and received packets. Moreover, you can configure the serial port where the 3DR radio is connected, and can set the fuel mass loaded into the aircraft.

Continuing with the notifications, you can see the status of the most safety-critical components regarding the UAV. These include the Datalink itself which shows if the connection is alive, all three RX-MUXes are present, whether the control is on autopilot or on manual and the limiter (which becomes red if one of the servo's temperature or deflection is out of the safe zone). Furthermore, there is a running notifications area that acts like a textbox and contains general and warning messages from the EDL. For example, if there are no received packets for at least three seconds, or when you connect to a communication port, it displays a message. Also, when the airspeed or the g-load exceeds its safe values, a message is going to be printed here in red. On the other hand, the status of the twelve IMU sensors in the wing to measure flutter dynamics can be seen as well.

The area for other systems encompasses engine, flight and battery information. About the engine, the user can see the revolution and the differential revolution in rpm, along with the fuel flow and engine temperature. The initial fuel mass can be set before turning on the jet engine, thereby the remaining amount is estimated using the fuel flow sensor. In the flight information section, the airspeed and g-load can be examined with

handy green and red bars next to the actual display, to inform the user if either of the values exceeds the safety limit. The battery information section contains the battery voltage of each RX-MUX unit and the flightHAT.

Engineering Data Link is a two-way communication, the GCS can send configuration messages to the FCC. This occurs once per diagnostic session. The message consists of flags which can turn on or off the transmit of the corresponding subsystem. For example, it can be set, whether the ADS data or the analogue acceleration data from the IMUs should be sent or not. This way, the list of the requested variables can be easily modified.

III.B.2. Mavlink and Mission Planner

In case of the Mavlink communication, the emphasis is on the actual flight state data rather than the diagnostic data of aircraft hardware itself. Moreover, this telemetry is also two-way, so the GCS can send messages to the aircraft as well. The ground control station software in our case is Mission Planner (MP), which is an open-source program developed specifically for small unmanned vehicles. Because of the numerous features, it can be used in a wide range of fields. For example, the 3DR antennas can be configured using Mission Planner. The most frequently used part of the software is the *flight data* page, which can be seen in Fig. 22.



Figure 22. Mission Planner flight data page

As you can see, the map takes up most of the screen, with the position and orientation of the aircraft in the middle. At the top-left corner, the HUD (Head Up Display) shows the virtual horizon representing the Euler angles along with the airspeed and altitude data. These areas are mostly useful during actual flight testing. From ground test perspective, the area below the HUD is more important. Here, you can see several tabs to choose from. The default is *Quick*, where there are six configurable values displayed. Moreover, there are other useful tabs as well, like *Gauges*, *Telemetry Logs* and *Status* (which lists all the data MP can receive). As described in the EDL section, this telemetry must be implemented on the aircraft as well. Mavlink is basically the communication protocol that is used to uniformize the message structure for small UAVs. This is what MP uses, so Mavlink structured messages must be sent by the aircraft too.

During ground testing, Mavlink telemetry is mainly used to check the correct functionality of the xSens and ADS sensors. For instance, rotating the plane around different axes is going to be visible on the virtual horizon, or the GPS position can be checked on the map. Moreover, a practical feature of Mission Planner is, that all the values it receives are visible in numeric form in the *Quick* section.

III.C. Software/Hardware-in-the-loop tests of control software

Before introducing the SIL / HIL test setups aircraft servo actuator identification is discussed as this is a crucial part of aircraft modeling.

III.C.1. Servo actuator identification

Knowing the behavior of the elements of the whole aircraft model is beneficial to assemble a good simulation environment. Therefore we want to identify our servo actuators. In the wing of the aircraft there are HBL599 BLDC servo motors. These servo motors are driven by 8.4V LiPo source (on the plane) converted to a PWM signal with 5V peak-to-peak amplitude. The length of the high state means the position reference. The RX-MUX unit feeds the PWM input with a 333.33Hz PWM signal. The higher level control system has a 200Hz update frequency. These systems are not synchronized, that gives us a delay between 0-3ms with 1.5ms in average. In the RX-MUX unit signals from the control systems going through a piecewise linear transformation by LUTs. Position of the servos is measured on a potentiometer, this analogue measurement will cause some noise. Input position reference signals for servo identification were step series and triangle signals. The steps has $\pm 5^\circ$ or $\pm 25^\circ$ amplitude and a 1 sec period, based on pre-measurements this is enough to reach steady state. Low amplitude is to avoid velocity saturation, and higher amplitude is to measure the velocity limit. The same parameters are applied in triangle signals except for the 2 sec period. These test signals were evaluated without load on the servo, and without the aircraft, used only one SHM in HIL hardware environment.

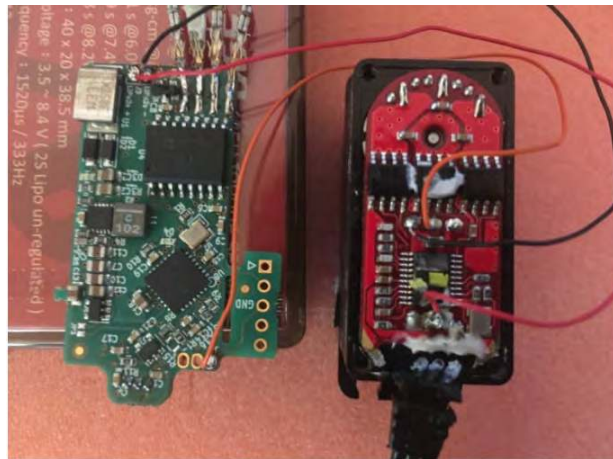


Figure 23. Measuring configuration

Because of the way of measuring the rotational position, there are several noise sources. Measurements show white noise, random high spikes, and slowly changing offset. The peak-to-peak amplitude of servo deflection is 10 degrees, and it is about 133 LSB, so $1\text{LSB} = 0.075$ degrees. We can neglect the offset because if we wait enough time, it will reach a steady state. And maximum offset was only 4 LSB, that means 0.3 degrees (see Fig. 24b). White noise (Fig. 24c) is about ± 1 LSB which is a low value and therefore tolerable. The random high spikes are really high, but fortunately they are rare (Fig. 24d).

Detecting and erasing high spikes is easy, to avoid offset, cutting out the first moving part of the signal is recommended.

After configuring the measurements the structure of the servo mathematical model should be estimated. Assuming that the inner motor itself is a linear time invariant SISO system, its model without load is:

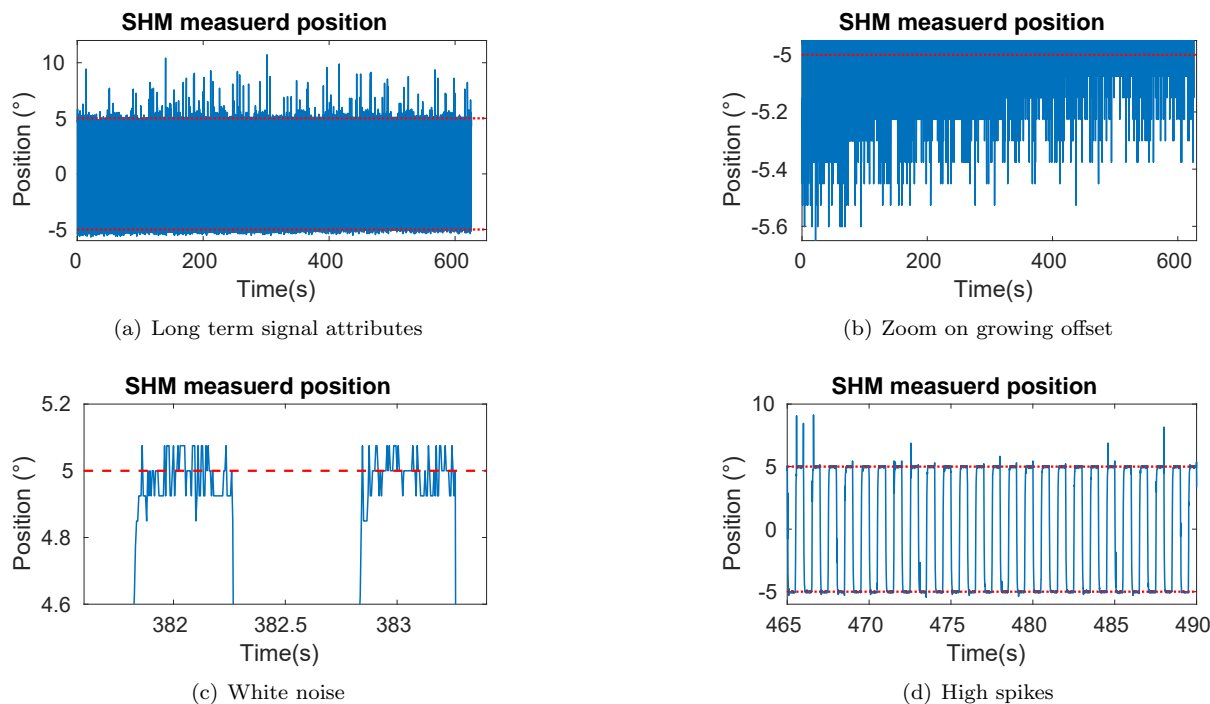


Figure 24. Output signal, with representative $\pm 5^\circ$ level lines

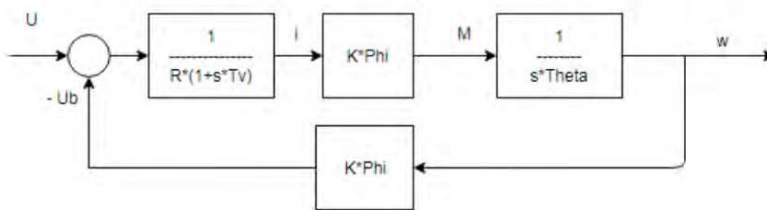


Figure 25. Motor model

Usually on DC motors they use cascade controllers: current, velocity and position. The U_b induced voltage must be compensated inside and for current controlling a PI controller is enough because the motor itself can be approximated by a 1 pole filter, therefore the numerator of PI will eliminate its pole. If the manufacturer took it easy, velocity loop uses a PI too, otherwise it is a PID. After that for position loop a P controller is enough, but sometimes they use nonlinear gains. So first we can assume a 3 pole system.

To estimate the parameters of the model first take a look on the low amplitude step responses which are the most close to a linear system. Here the output is not reaching any limit or boundary. Using built in MATLAB System Identification Toolbox for a 3 pole system, with the in- and output mentioned above lead to the identified system response in Fig. 26.

As Fig. 26 shows the response is fitting well, MATLAB says fitting is 93% but lets validate it to the triangle signals as well. As can be seen in Fig. 26, fit to the triangle signals is more than 94%, according to MATLAB, but some difference in delay is slightly sensible. The next step of validation should be to consider the higher step signal in Fig. 26. As the figure shows, the real signal is slower then the model responses so there is speed limit not considered in the model and if we fit a line to the linear part of the rising edge, through its gradient the limit can be measured.

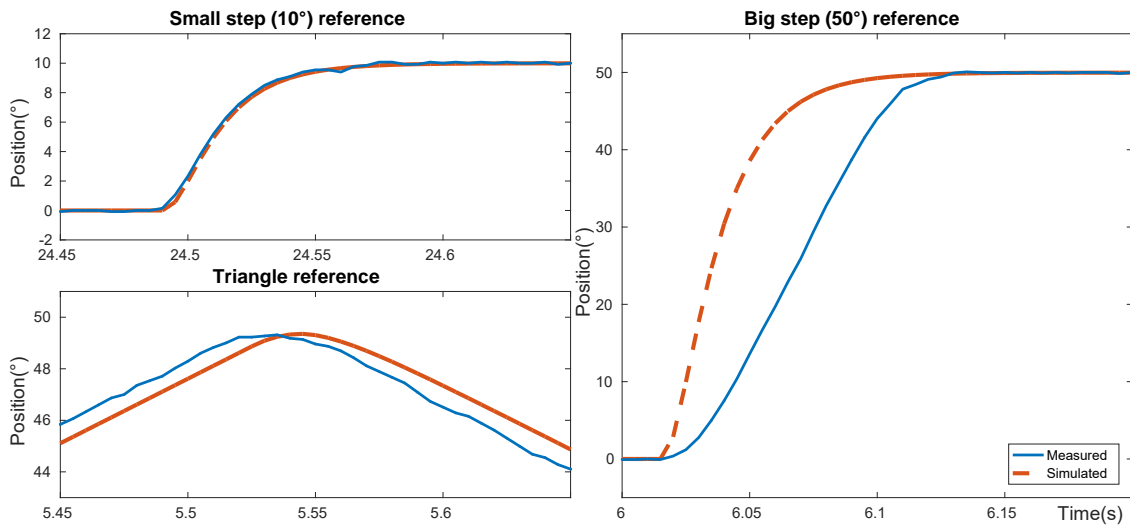


Figure 26. Response of non-saturated model

$$\begin{aligned} 25^\circ / 0.04s &= 625^\circ / s \\ \Rightarrow &0.0016rps \end{aligned} \tag{1}$$

this is 0.096 seconds at 60°, and manufacturer says its 60 degree time is around 0.08 seconds to. The LTI system from the low step identification is:

$$W_{closed} = \frac{9.581 \cdot 10^6}{s^3 + 1298 \cdot s^2 + 2.41 \cdot 10^5 \cdot s + 9.587 \cdot 10^6} \tag{2}$$

We want to reach the inside of the system, so open it up by the rule of:

$$W_{open} = \frac{W_{closed}}{1 - W_{closed}} \tag{3}$$

After evaluating the above equation

$$W_{open} = \frac{9.5805 \cdot 10^6}{(s + 1074)(s + 224.4)(s + 0.025)} \tag{4}$$

You can see, that 0.025 is close to zero enough to be a zero pole which means an integrator. It is the physical law of position equal the integral of velocity. We can put in the limitation for velocity, and the new system is:

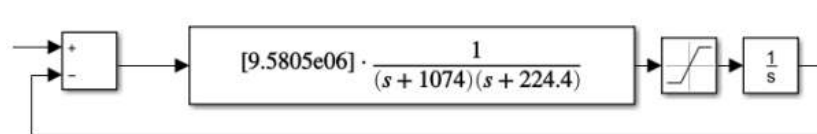


Figure 27. Transfer function model with saturated velocity (System-2)

Fig. 30a shows the answer of the new system model to a low amplitude step signal showing that saturated and non-saturated responses are the same inside the velocity limit. Fig. 30b shows the answer of the new system model to a high amplitude step signal. One can see that with saturation, the model response gets even more closer to the real world.

Opening more the loop (for the acceleration limit) and saturating was promising. Assume that there is a cascade controller in the motor with PI velocity and current loops and so their closed loop transfer function contains 2 integrators, one pole and one zero (see Fig. 28). The PI term in the velocity loop is often designed

by the rule that the geometric mean of the zero in the velocity loop and the pole of closed current loop is the cut frequency of the closed velocity loop. We can estimate the 2 pole of the closed velocity loop. The lower is the cut frequency, the higher is the pole of the closed current loop. The estimated parameters are also shown in Fig. 28.

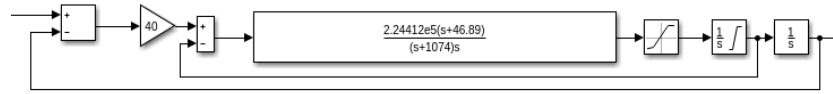


Figure 28. Inside of velocity loop (System-3)

Saturating the acceleration signal is not enough itself, we have to reach an inside state to saturate that. Therefore 2 options are available. First you can pick the integrator from the transfer function and saturate it, or second, you can open the pole as an individual filter. By choosing the second, we can do other improvements after. Now the rising edge is similar to the measured, but when approaching steady state, it is still not good enough. Measurements shows that, servo try to brake quickly in the last moments. Cascade controllers may have some nonlinear gains inside. After trying some configurations, position control loop needs an additional nonlinear integrator. To make life easier, approximate it with a switched integrator. New model reaching steady state as fast as measured, but there are a waving after. Thus velocity control loop has nonlinear gains too. Changing the integrator to a switched integrator was enough to eliminate waving. In this case, switched integrators has 2 state, on and off. 'On' is usual. in 'off' state, the integrated value is set to zero. They are switched off when absolute position error is under 0.4 degree.

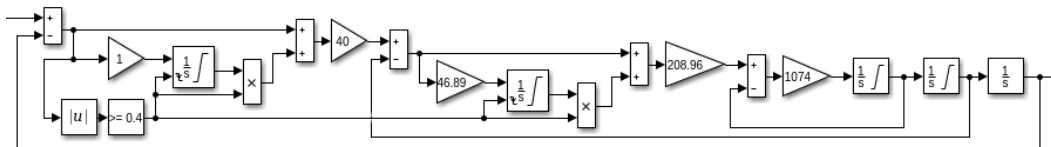


Figure 29. Piece-wise linear model (System-4)

The validation of the fourth system shows that, for low steps, the response a little differs from the previous models, for high steps it is much more better while for triangle signals it is the same. See in Fig. 30.

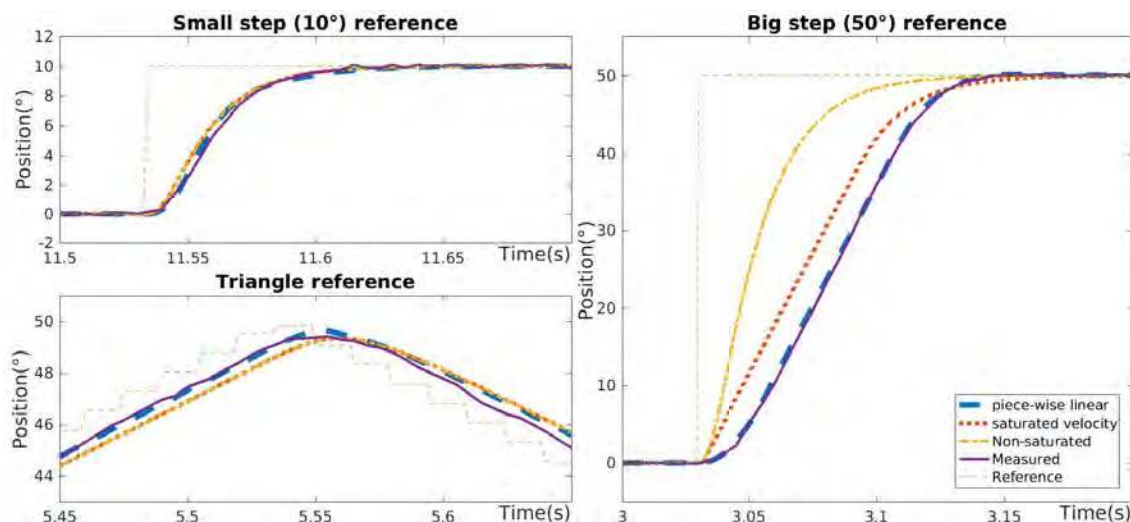


Figure 30. Responses of models

This fourth servo motor model is included into SIL and HIL simulation models of the aircraft applied for control design and evaluation.

III.C.2. SIL and HIL test configurations

After design and development its important to test the autopilot code and hardware in safe circumstances before doing real flight testing. Three different setups were created to gradually make these tests the concept is shown in Fig. 31. In the figure SIL is the so-called Software-in-the-loop simulation where the aircraft (A/C) model the baseline and flutter controllers are all run in Matlab / Simulink completely excluding FCC hardware. SIL-HIL is a hybrid SIL and Hardware-in-the-loop (HIL) simulation where part of the autopilot code (the baseline control) runs on the FCC hardware while aircraft is simulated and flutter control is done in Matlab / Simulink. The need to use this configuration will be highlighted later. Finally, HIL is the setup where all control algorithms run on the FCC hardware and only the aircraft is simulated in Matlab / Simulink. This is the last stage which should be passed before real flight testing.

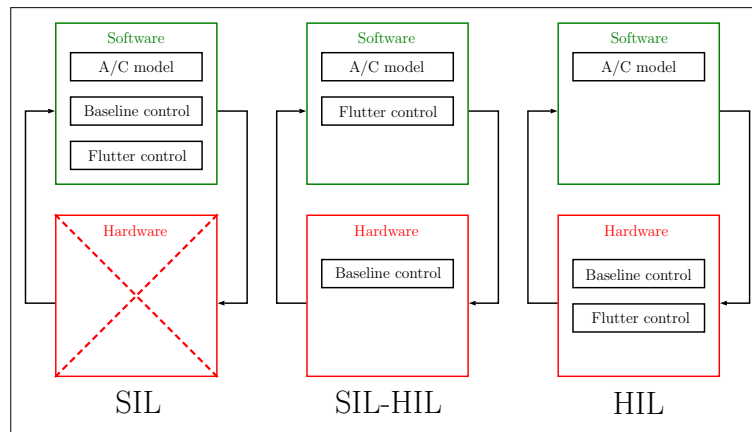


Figure 31. Three different autopilot ground test configurations

A detailed scheme of the HIL simulation is shown in Fig. 32 while its photo is shown in Fig. 33 (excluding the PC on which the Matlab simulation runs). Fig. 32 shows that the complete HIL setup consists of a Matlab / Simulink model of the aircraft together with actuator models and emulation of sensors completed with visualization and the real hardware part (the same as applied on-board the A/C). In our setup visualization is done by Flightgear software fed with position and orientation data from Matlab. The hardware part is as discussed above in Subsection III.A.

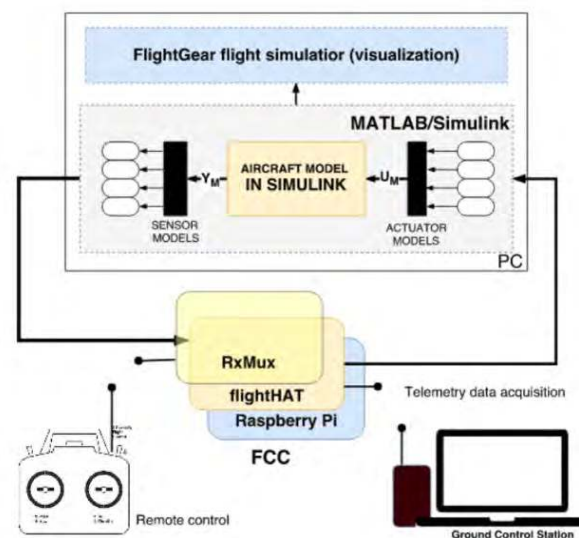


Figure 32. Overview of HIL simulation setup

After implementing the HIL setup (including implementation of aircraft simulation provided by DLR in Matlab) the delay of the hardware part should be tested to consider it in the design of the controllers as this

will be the approximate time delay on-board the aircraft also.

III.C.3. HIL delay test

The delay of the HIL setup was tested by sending a square signal to the controller and receiving it back. The time shift between the two square signals shows the approximate time delay of the hardware part. Its only approximate because includes the delay of Matlab simulation and interface hardware cards in the PC also. The result of delay measurement is shown in Fig. 34. The measured approximate delay is 20ms including the one time step delay of the discrete time controller which is 5ms. As the resolution of measurement is 5ms (sampling of controller) one can state that the hardware delay is about 15ms. This delay was considered in the design of the baseline (see¹⁸) and the flutter controllers. SZTAKI flutter controller was designed for the 4.0 model version of the aircraft provided by DLR. The design method is presented in¹⁹ and detailed in²⁰ (in hungarian). DLR flutter controller was designed for 4.0 and 5.0 models as presented in.¹⁸ The next part details the tests of the flutter controllers.

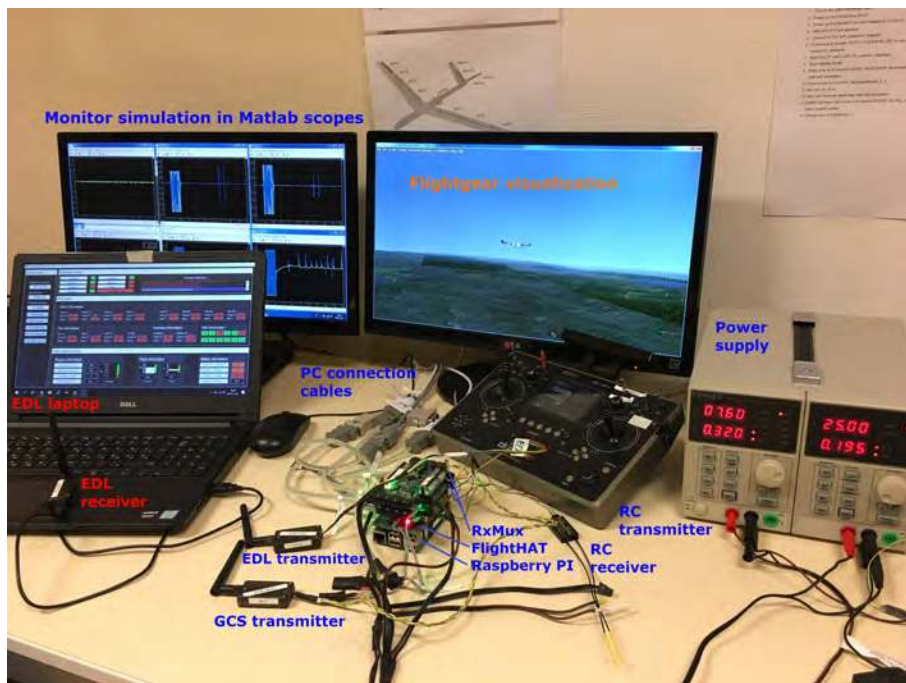


Figure 33. Picture of the HIL simulation setup at SZTAKI

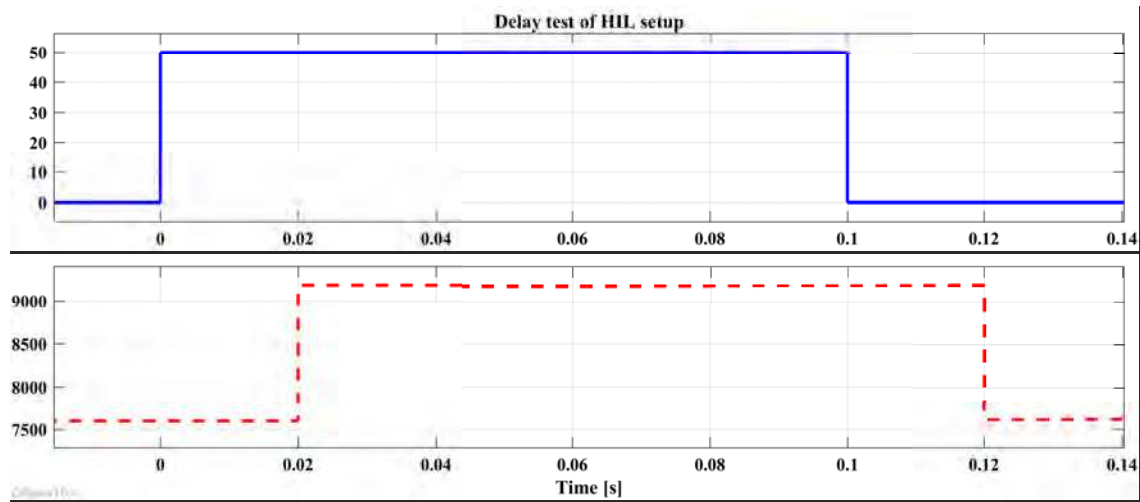


Figure 34. Delay measurement in HIL. Continuous line is the signal out, dashed line is the echoed signal from FCC

III.C.4. Implementation and basic test of flutter controllers

All of the controllers (baseline and flutter) are constructed in Matlab / Simulink as discrete time models with 5ms sampling and the executable code is built from it after SIL test runs prove the correct functionality of the control. The baseline controller implements indicated airspeed (IAS) and altitude tracking together with the tracking of a racetrack pattern where the linear parts will be the flutter test legs in real flight. This controller was extensively tested in SIL and HIL for 4.0 and 5.0 model versions and finally also real flight tested (with rigid wing). Real flight test has shown that the 5.0 model version better describes aircraft dynamics and so should be applied in any further tests.

Considering the flutter controllers the SZTAKI version has a small, time invariant state space model with 10 states, 2 inputs and 2 outputs which is easy to implement and runs in real time. On the other hand the DLR version has a linear parameter varying (LPV) state space model with 11 states, 26 inputs and 2 outputs. It is linearly interpolated between different aircraft airspeeds. As real time execution requires the reduction of the computational needs its worth to implement the linear interpolation of matrices in closed form. The two corner models of the parameter space for 30m/s and 70m/s IAS are the results of design as these are enough to cover the whole space because of linear interpolation. Denoting the related state space models as A_{30}, B_{30}, C_{30} and A_{70}, B_{70}, C_{70} (the D matrices are all zero) one can reformulate interpolation as follows:

$$\begin{aligned}
 A(V) &= A_{30} + \frac{A_{70} - A_{30}}{70 - 30}(V - 30) \\
 A(V) &= \underbrace{A_{30} - 30 \frac{A_{70} - A_{30}}{70 - 30}}_{A0} + \underbrace{\frac{A_{70} - A_{30}}{70 - 30}}_{AV} V \\
 A(V) &= A0 + AV \cdot V \\
 B(V) &= B0 + BV \cdot V \\
 C(V) &= C0 + CV \cdot V
 \end{aligned} \tag{5}$$

where V is the actual airspeed and $B0, C0, BV, CV$ are formulated similarly to the components of the $A(V)$ matrix. These matrices can be a priori calculated and so well applied in controller implementation.

Unfortunately, test of the flutter controllers in SIL on 5.0 A/C model has shown that the performance of the SZTAKI controller was unsatisfactory with 14ms delay and above, so it should be further developed before HIL and flight testing (note that HIL delay is 15ms). The only remaining possibilities were to do extensive SIL and SIL-HIL testing of the flutter controllers to compare their performance. The focus of the current work is the SIL test campaign as this is the basis of any further evaluation and so ground testing. SIL-HIL results are presented in another work also because of space constraints of the current work see.²¹

III.C.5. SIL evaluation of flutter controllers

Both DLR and SZTAKI controllers were run in SIL applying constant altitude hold (800m AMS) with constant course angle hold and a staircase IAS reference starting from 45 m/s and going up to 65 m/s in steps of 1 m/s every 15 seconds (see Fig. 36 for example). The simulation was stopped if the IAS suddenly decreased because of flutter of the wings. An open loop simulation (OLP) with only baseline controller and then closed loop simulations with both flutter controllers (together with baseline) with delay of 13 ms, 14 ms and 15 ms were run to evaluate the performance of the flutter controllers. The limit speeds and the flutter clearance are summarized in Table 1. Fail means sudden decrease of aircraft airspeed because of flutter, pass means that the airspeed was held for a period of 50 s-100 s therewith demonstrating that the flutter compensation is successful. Note that 64.5 m/s is the maximum achievable IAS of the simulated A/C model. It is interesting to observe that the DLR controller performed better with a higher delay values. This is attributed to the fact that it was designed for a 15 ms delay. Hence a too low delay can also cause performance degradation. The IAS tracking results are shown in Figs. 35 and 37 excluding the 14 ms delay cases as with DLR flutter controller the 14 ms case gives the same result as the 15 ms and with SZTAKI flutter controller the 14 ms case is not much better then the 15 ms one.

Table 1. Flutter speed results in SIL

Delay	13 ms		14 ms		15 ms	
OLP	52 m/s	N/A	N/A	N/A	N/A	N/A
DLR	64 m/s	fail	64.5 m/s	pass	64.5 m/s	pass
SZTAKI	62 m/s	fail	54 m/s	fail	53 m/s	fail

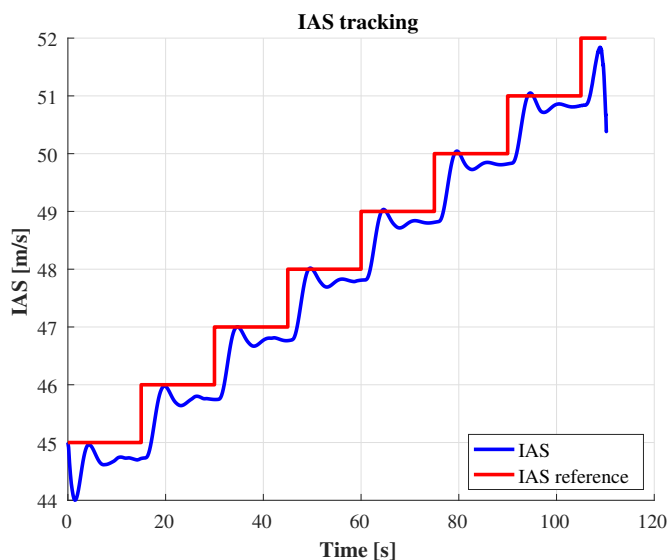


Figure 35. Baseline open loop staircase IAS tracking

Detailed control input and flutter mode energy results are collected in the Appendix A. Figs. 43 and 44 show the baseline and flutter controller input energies for the different controllers and delays. Energy here means the squares of the signals summarized for a given time interval. As the open loop flutter airspeed is about 52 m/s and the maximum airspeed is 64.5 m/s the energy calculations are done from 51 m/s to 65 m/s reference values summing up the squared signals separately for every constant reference IAS section. That's why IAS is on the horizontal axis of all the figures. Fig. 44 shows that the ruddervator energy increases as the airspeed increases and this is mainly because of the modification of elevator trim value. Otherwise the values for DLR and SZTAKI flutter control are the same, there is no significant difference. The sections going down in case of SZTAKI control are caused by the stop of the simulation after flutter control fails. From this point there is no data to sum up and this is the same in all other figures where a sudden drop to almost zero value can be seen.

Figure 43a shows that with the DLR flutter controller the baseline controller uses the same amount of control in the aileron channels for any delay and the deflections are moderate (also for aileron 3 while aileron 1 is not used in baseline control). On the other hand with the SZTAKI flutter controller also the baseline controller uses a larger amount of control with aileron deflections reaching about 4° . This can be seen in the much larger energy values in the figure. The situation is the same regarding the flutter control part (Aileron 4 in Fig. 43b) where the DLR controller uses moderate deflections while the SZTAKI controller applies much larger control energy. The figure also shows that the DLR control input decreases as the system delay approaches the value considered in flutter control design. The larger control energy input can be advantageous if it gives better damping for the flexible dynamics of wings. To check this the measured accelerations and angular rates at the wing tip (IMU6) are processed together with the modal velocities of the first modal coordinate (which is the most excited coordinate). These are all proportional with the stored energy in the wings. In case of acceleration (bending energy) and angular rate (torsion energy) signals the body accelerations and angular rates are first subtracted to get relative values and these are squared and summed as in case of the control deflections. Figs. 45a and 45b show that the SZTAKI flutter controller allows much larger accelerations and angular rates when the wing approaches flutter. This is also verified with the modal energy term (see Fig. 46) where the stored energy in the wings is much larger with SZTAKI flutter control than with DLR. In this figure the squared modal velocities are summed in the same manner as the control deflections.

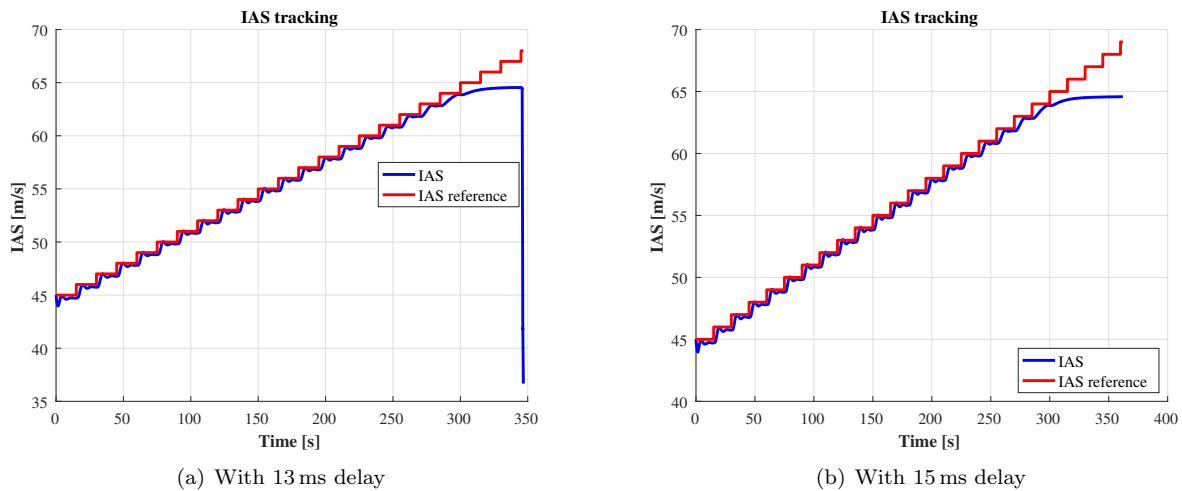


Figure 36. DLR closed loop staircase IAS tracking

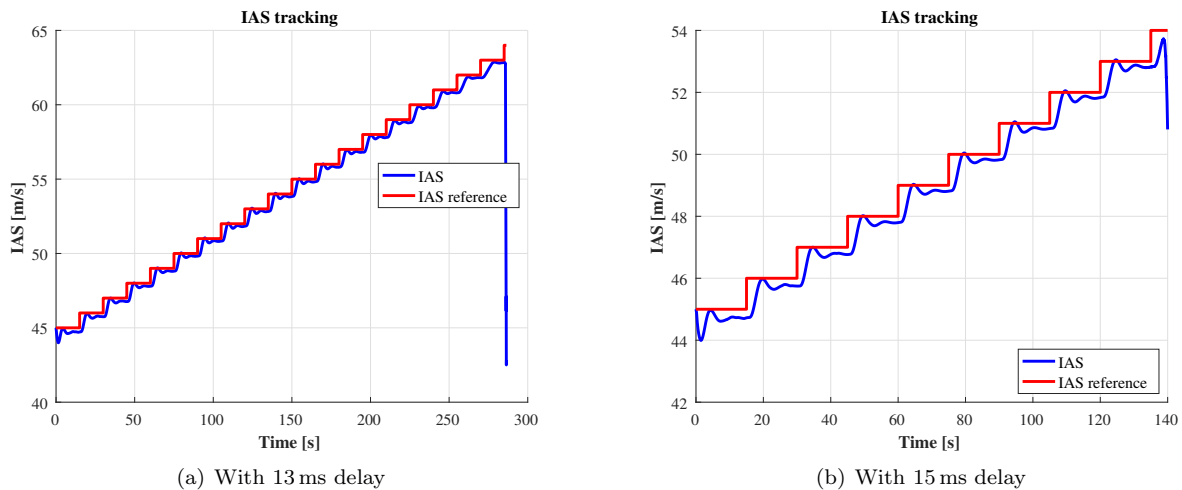


Figure 37. SZTAKI closed loop staircase IAS tracking

Summarizing the SIL test results the test shows that the baseline controller works well until wing flutter occurs. Above the 51-52 m/s flutter speed the baseline controller is incapable to stabilize the aircraft. Comparing the DLR and SZTAKI flutter controllers has led to the result that the DLR controller is capable to stabilize the wing even until the maximum airspeed of the aircraft if the delay of the system matches the delay considered in control design. On the other hand the SZTAKI controller uses much more control energy and induces much more energy in the flexible dynamics and is unable to stabilize the wing on the whole airspeed range. By the increase of system delay its capabilities become even more limited.

The next step should be to test DLR flutter controller in SIL-HIL and HIL test setups and redesign and improve SZTAKI controller and repeat SIL test after.

IV. Integration test

IV.A. Airbrake

The general set-up of the airbrake system test rig can be seen in Fig. 38a. The positions of the potentiometer and the digital pitch gauge (DPG) are also marked in Fig. 38a. The design considerations and more detailed test results can be found in the work of Sender et al.²²

The aerodynamic loads are substituted by weights with a mass m of 12.658 kg suspended from two-wire cables. Figure 38b shows the force on the airbrake in N versus the opening angle ϕ in degree. The continuous line visualizes the aerodynamic normal force F_N whereas the dotted line visualizes the perpendicular component of the force applied by the weights.

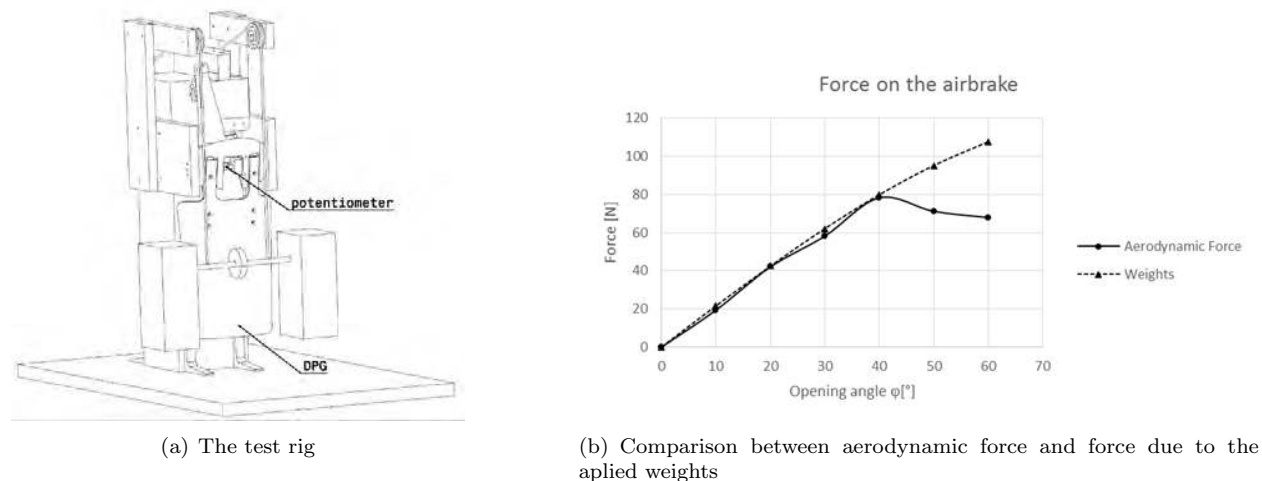


Figure 38. Airbrake test setup and assessment

IV.B. Engine

The engine was tested in the LLS Propulsion Container. The container includes a high accuracy 6 degrees of freedom load cell for force and moment measurement. Also, the custom Engine Control Unit is installed, allowing accurate logging of actual engine RPM values. The test setup for the engine itself is illustrated in Fig. 40a.

The main goal of testing was to update the simulation model for the engine. Selected engine RPM command ramps were applied to get a bigger picture of how well does the engine thrust respond to the pilot's command.

Some of the important flight phase RPM ramps were noted down and used as a testing protocol. Fig. 39 illustrates one of the test runs done in September 2016.

In addition to the testing done above, transmitter output to engine RPM mapping had to be done. The RPM response to given PWM commands shown in Fig. 40b. This test also showed that the custom flight control computer (FCC) can work reliably with the engine.

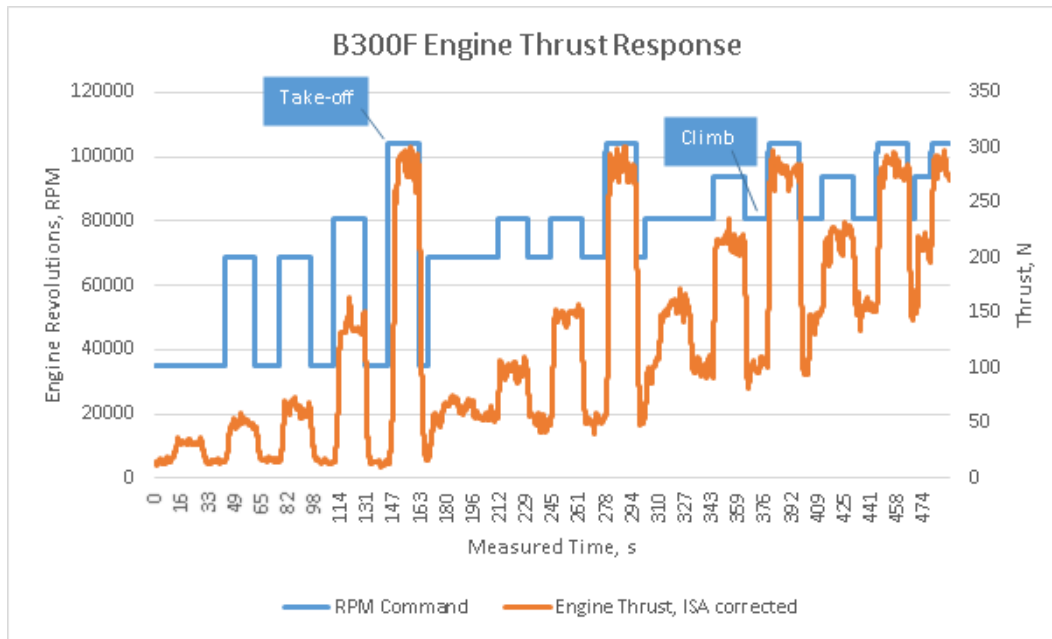
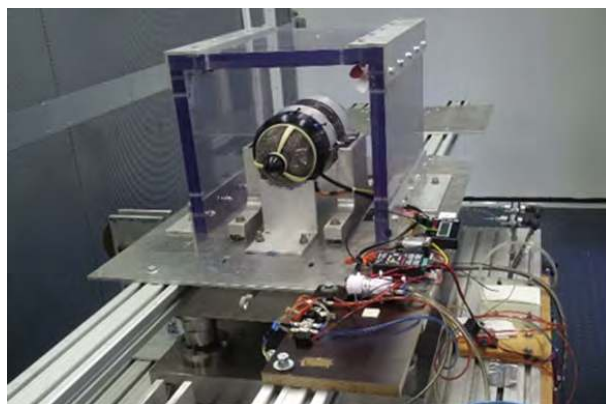
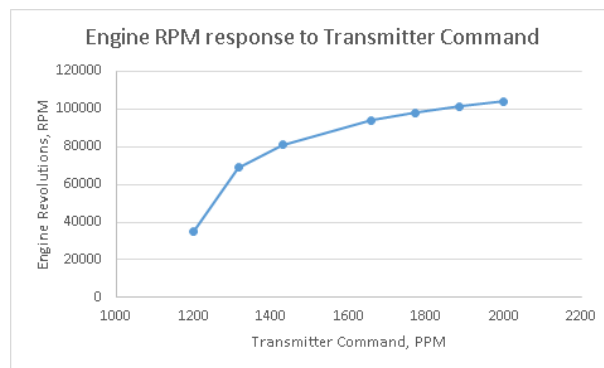


Figure 39. Engine test run on test stand with different RPM profiles



(a) B300F Engine on the LLS Engine Testbench



(b) PWM reference signal vs RPM

Figure 40. Engine test setup

IV.C. Compatibility of electric systems

The first tests and integration of the main parts of the electronic system was made on a wooden mock-up. The aim of the mock-up system was to have the electrical systems like actuators, sensors, FCC, radio control (RC) system and cabling assembled together in a way which is easy to access and modify. It showed that all the electrical components can work together smoothly.



Figure 41. Wooden mock-up bench used for electrical system compatibility testing

There was no unexpected behavior found during these tests. The system worked properly with high actuator loads, and sudden actuator movements when high current peak could cause no voltage drop on the main power system.

To see the behavior of the fully integrated system, we made range and taxi tests.

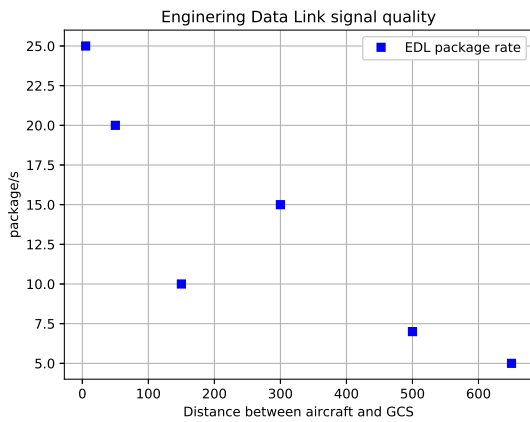
IV.D. Range test

The aim of the range test was, to gain information about telemetry and the RC system of the aircraft.

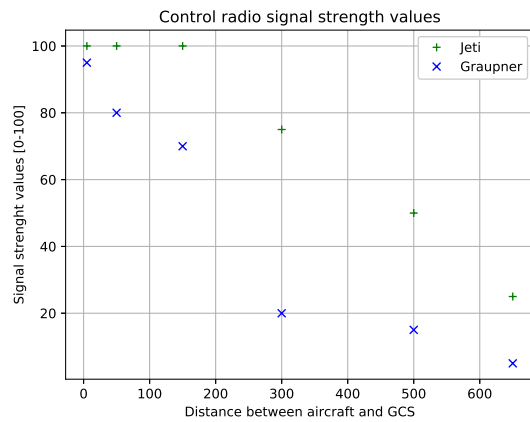
The range test is made on a clear field, in which we had a clear line of sight between the aircraft and the moving ground control station car for a distance of at least 1km. The minimum requirement for each system was to have acceptable reception at 600m range.

The aircraft was placed on a stand 1 m above the ground. The ground control station car and its antennas were placed 5 m above the ground and oriented in the same way as planned for the flight testing.

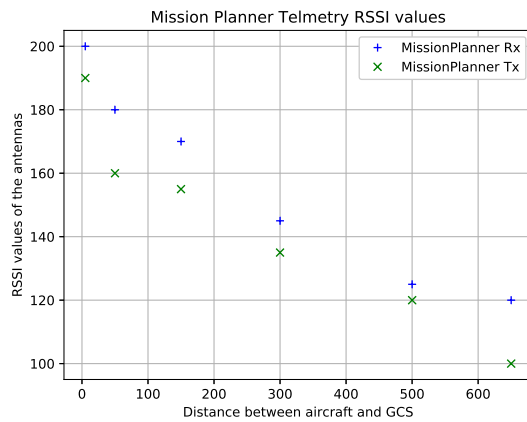
We used different measurement values to determine the quality of the radio signals. The JETI system provides Received Signal Strength Indicator (RSSI) values on its screen, for each of its antenna. The Graupner RC system provides RSSI-like values in the form of signal strength and data quality. The test results for both systems are shown in Fig. 42b. The Mission Planner provides RSSI values for both reception and transmission. The test results are shown in Fig. 42c. The EDL software gives information about the overall received packages with the test results shown in Fig. 42a. As expected, signal quality and strength values show a decreasing trend as the distance increases between the aircraft and the ground station. Although the measured values at 650m are quite low compared to the measurements at 5 – 50m, all the connections were still working. The aircraft was controllable from both transmitters, and the data flow on Mission Planner and EDL was sufficient.



(a) Engineering data link



(b) Radio control signal



(c) Mission planner telemetry

Figure 42. Range test measurement results

IV.E. Taxi test

The taxi test was aimed to gain information about the aircraft controllability on the ground and to have all the systems working together integrated in the aircraft. The taxi tests were the only tests, in which all the systems including the ground control station and the crew had to work together.

The taxi test showed poor controllability of the aircraft on the ground. The aircraft was hard to control at the speed of less than 3 m/s. The aircraft also showed a tendency to veer off course sideways at higher speeds of around 10 m/s to 15 m/s, which rendered the take-off and landing manoeuvre difficult. These two manoeuvres were therefore considered mission-critical points.

The effectiveness of the brake system was also tested during the taxi test. It was shown, that the aircraft needed the same distance to fully stop, that it required to accelerate to the given speed. It was also demonstrated that if one of the two brakes had failed, the aircraft would not veer off course at lower speeds in the range of 3 m/s to 5 m/s.

The RC and the telemetry systems were also tested during the taxi test since the radio-based systems are detrimentally affected by the proximity of the ground. The RC system was tested for the fail-safe mechanism. In the case of connection loss the engine must turn off and the parachute must be released. The telemetry system was tested for loss of data before and during the landing phase of flight the aircraft is hard to control precisely at this stage of the flight. In case of the telemetry data loss, only visual information about the aircraft is available to the pilot. Therefore both systems had to demonstrate sufficient reliability in terms of connectivity and data transmission during ground operations up to a distance of 300 m away from the pilot. Both systems were working reliably even beyond the distance of 360 m.

V. Conclusion

The paper gives a comprehensive presentation of the ground testing campaign performed within the FLEXOP project. Structural, flight system and integration tests along with the most important findings are discussed. Possibly, the most important outcome of the ground testing campaign is clearing the aircraft in either wing (-0) and wing (-2) configuration for the flight testing campaign.

In terms of structural testing, the outcomes of the static and the GVT test were successfully applied in structural characterisation and model updating of the wing (-0) and wing (-2) aircraft configuration. In addition to modal properties of the various aircraft configuration, the GVT test also identified structural nonlinearities in the wing inplane bending mode which are important from the aeroelastic point of view. The results obtained during the structural tests were also successfully used for the maturation and calibration of the FBG measurement system such that it can be used in flight testing. Within this scope, the load and shape sensing parameters were determined and the ability of the FBG system to capture loads and deformations of interest was experimentally confirmed. Finally, the airworthiness test was successfully completed demonstrating the structural integrity of the manufactured wings.

Flight system testing resulted in characterisation of various flight control computer subsystems such as signal data flow, autopilot loops, telemetry subsystem and control surface actuators. Within this scope, a crucial point was the measurement of hardware time delay in the HIL test which was used in the design of baseline and flutter controllers. The designed baseline controller was extensively tested in both SIL and HIL as well as in flight test. Moreover two flutter controllers, developed by DLR and SZTAKI, were tested in SIL. The test showed that the DLR flutter controller was capable to control flutter well beyond the open loop flutter speed for all the considered hardware time delays. On the other hand, the performance of the SZTAKI flutter controller (designed for a version of the aircraft model different from the one used in the tests) degraded significantly with increasing hardware time delay. Hence, it remains to redesign the SZTAKI controller for the adequate aircraft model in the future.

Finally, the integration tests successfully demonstrated that all the critical systems in the aircraft meet all the functionality requirements and function reliably. It was demonstrated that the airbrake could deploy under anticipated aerodynamic loads. The jet engine passed all the reliability tests. In the process, the engine model was also updated. Furthermore, all the electronic systems were assembled and successfully tested for compatibility. All the radio systems were successfully tested for range and stability of the radio link during the range and taxi tests. The taxi tests were also used to assess the handling qualities and controllability of the aircraft during ground operations.

Acknowledgements

The work presented herein has been partially funded by the European Community's Horizon 2020 Programme (H2020) under the Grant Agreement 636307. The FLEXOP project (Flutter Free FLight Envelope eXpansion for ecOnomical Performance improvement) is a project funded under the topic MG-1.1-2014. involving 10 partners. The project started on June 1st 2015.

References

- ¹Roessler, C., Stahl, P., Sendner, F., Hermanutz, A., Koeberle, S., Bartasevicius, J., Rozov, V., Breitsamter, C., Hornung, M., Meddaikar, Y. M., Dillinger, J., Sodja, J., De Breuker, R., Koimtzoglou, C., Kotinis, D., and Georgopoulos, P., "Aircraft Design and Testing of FLEXOP Unmanned Flying Demonstrator to Test Load Alleviation and Flutter Suppression of High Aspect Ratio Flexible Wings," *AIAA Scitech 2019 Forum*, No. January, American Institute of Aeronautics and Astronautics, Reston, Virginia, jan 2019, pp. 1–20.
- ²Gözse, I., Péni, T., Luspay, T., and Soumelidis, A., "On the correspondence of hyperbolic geometry and system analysis," *IFAC-PapersOnLine*, Vol. 50, No. 1, 2017, pp. 9285–9290.
- ³Lipták, G., Luspay, T., Péni, T., Takarics, B., and Vanek, B., "LPV model reduction methods for aeroelastic structures," *IFAC-PapersOnLine*, Vol. 50, No. 1, 2017, pp. 6344–6349.
- ⁴Rozov, V., Hermanutz, A., Breitsamter, C., and Hornung, M., "Aeroelastic analysis of a flutter demonstrator with a very flexible high-aspect-ratio swept wing," *17th International Forum on Aeroelasticity and Structural Dynamics, IFASD 2017*, Vol. 2017-June, No. June, 2017, pp. 1–13.
- ⁵Sendner, F.-M., Stahl, P., Rößler, C., and Hornung, M., "Designing an UAV Propulsion System for Dedicated Acceleration and Deceleration Requirements," *17th AIAA Aviation Technology, Integration, and Operations Conference*, No. June, American Institute of Aeronautics and Astronautics, Reston, Virginia, jun 2017, pp. 1–13.
- ⁶Stahl, P., Sendner, F.-M., Hermanutz, A., Rößler, C., and Hornung, M., "Mission and Aircraft Design of FLEXOP Unmanned Flying Demonstrator to Test Flutter Suppression within Visual Line of Sight," *17th AIAA Aviation Technology, Integration, and Operations Conference*, No. June, American Institute of Aeronautics and Astronautics, Reston, Virginia, jun 2017, pp. 1–11.
- ⁷Binder, S., Wildschek, A., and de Breuker, R., "Aeroelastic stability analysis of the FLEXOP demonstrator using the continuous time unsteady vortex lattice method," *AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, 2018*, , No. 210049, 2018.
- ⁸Iannelli, A., Marcos, A., and Lowenberg, M., "Aeroelastic modeling and stability analysis: A robust approach to the flutter problem," *International Journal of Robust and Nonlinear Control*, Vol. 28, No. 1, 2018, pp. 342–364.
- ⁹Luspay, T., Péni, T., Gözse, I., Szabó, Z., and Vanek, B., "Model reduction for LPV systems based on approximate modal decomposition," *International Journal for Numerical Methods in Engineering*, Vol. 113, No. 6, 2018, pp. 891–909.
- ¹⁰Meddaikar, Y., Dillinger, J., Sodja, J., and De Breuker, R., "FLEXOP – Application of aeroelastic tailoring to a flying demonstrator wing," *Deutscher Luft- und Raumfahrtkongress 2018*, Friedrichshafen, Germany, 2018, pp. 1–10.
- ¹¹Sodja, J., Werter, N., and De Breuker, R., "Design of a flying demonstrator wing for manoeuvre load alleviation with cruise shape constraint," *2018 AIAA/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference*, No. 210049, American Institute of Aeronautics and Astronautics, Reston, Virginia, jan 2018.
- ¹²Wuestenhagen, M., Kier, T., Meddaikar, Y. M., Pusch, M., Ossmann, D., and Hermanutz, A., "Aeroservoelastic Modeling and Analysis of a Highly Flexible Flutter Demonstrator," *2018 Atmospheric Flight Mechanics Conference*, American Institute of Aeronautics and Astronautics, Reston, Virginia, jun 2018, pp. 1–12.
- ¹³Iannelli, A., Marcos, A., and Lowenberg, M., "Nonlinear Robust Approaches to Study Stability and Postcritical Behavior of an Aeroelastic Plant," *IEEE Transactions on Control Systems Technology*, Vol. 27, No. 2, 2019, pp. 703–716.
- ¹⁴Luspay, T., Baár, T., Teubl, D., Vanek, B., Ossmann, D., Wüstenhagen, M., Pusch, M., Kier, T., Waitman, S., Iannelli, A., Marcos, A., and Lowenberg, M., "Flight control design for a highly flexible flutter demonstrator," *AIAA Scitech 2019 Forum*, , No. January, 2019, pp. 1–29.
- ¹⁵Meddaikar, Y. M., Dillinger, J., Klimmek, T., Krueger, W., Wuestenhagen, M., Kier, T. M., Hermanutz, A., Hornung, M., Rozov, V., Breitsamter, C., Alderman, J., Takarics, B., and Vanek, B., "Aircraft Aeroservoelastic Modelling of the FLEXOP Unmanned Flying Demonstrator," *AIAA Scitech 2019 Forum*, No. January, American Institute of Aeronautics and Astronautics, Reston, Virginia, jan 2019.
- ¹⁶Roessler, C., Bartasevicius, J., Koeberle, S., Teubl, D., Hornung, M., Meddaikar, Y., Dillinger, J. K. S., Krüger, W., Wüstenhagen, M., Ossmann, D., Kier, T., Looye, G., Sodja, J., De Breuker, R., Luspay, T., Gyulai, L., Balogh, D., Vanek, B., Georgopoulos, P., and Koimtzoglou, C., "Results of an Aeroelastically Tailored Wing on the FLEXOP Demonstrator Aircraft," *AIAA Atmospheric Flight Mechanics Conference, AIAA SciTech Forum*, Orlando, Florida, 2020. (Submitted).
- ¹⁷Takarics, B., Patartics, B., Luspay, T., Bauer, P., Vanek, B., Roessler, C., Bartasevicius, J., Koeberle, S., Teubl, D., Hornung, M., Waitman, S., Marcos, A., Pusch, M., Wuestenhagen, M., Kier, T., Looye, G., and Meddaikar, M., "Active Flutter Mitigation Testing on the FLEXOP Demonstrator Aircraft," *AIAA Atmospheric Flight Mechanics Conference, AIAA SciTech Forum*, Orlando, Florida, 2020. (Submitted).
- ¹⁸Pusch, M., Ossmann, D., and Luspay, T., "Structured Control Design for a Highly Flexible Flutter Demonstrator," *Aerospace (MDPI)*, Vol. 6, No. 27, 2019.
- ¹⁹Patartics, B., Luspay, T., Péni, T., Takarics, B., Vanek, B., and Kier, T., "Parameter varying flutter suppression control for the BAH jet transport wing," *IFAC-PapersOnLine*, Vol. 50, No. 1, 2017, pp. 8163 – 8168, 20th IFAC World Congress.
- ²⁰Patartics, B., Peni, T., and Vanek, B., "Flexibilis szárnyú repülőgép aktív flatter szabályozása," *Repüléstudományi közlemények*, Vol. 30, No. 1, 2018.

²¹Patartics, B., Takarics, B., Luspay, T., Bauer, P., Vanek, B., Roessler, C., Bartasevicius, J., Koeberle, S., Teubl, D., Hornung, M., Waitman, S., Marcos, A., Pusch, M., Wuestenhagen, M., Kier, T., Looye, G., and Meddaikar, Y. M., "Active Flutter Mitigation Testing on the FLEXOP Demonstrator Aircraft," *In proc. of AIAA Scitech Forum 2020*, 2020.

²²Sendner, F.-M., Stahl, P., Rößler, C., and Hornung, M., "Design and Testing of an Electric Actuated Airbrake for Dynamic Airspeed Control of an Unmanned Aeroelastic Research Vehicle," *18th AIAA Aviation Technology, Integration, and Operations Conference 2018*, edited by T. A. I. of Aeronautics and Astronautics, The American Institute of Aeronautics and Astronautics, Atlanta, Georgia, Vereinigte Staaten von Amerika, Jun 2018, p. 10.

A. Input and wing flutter energies

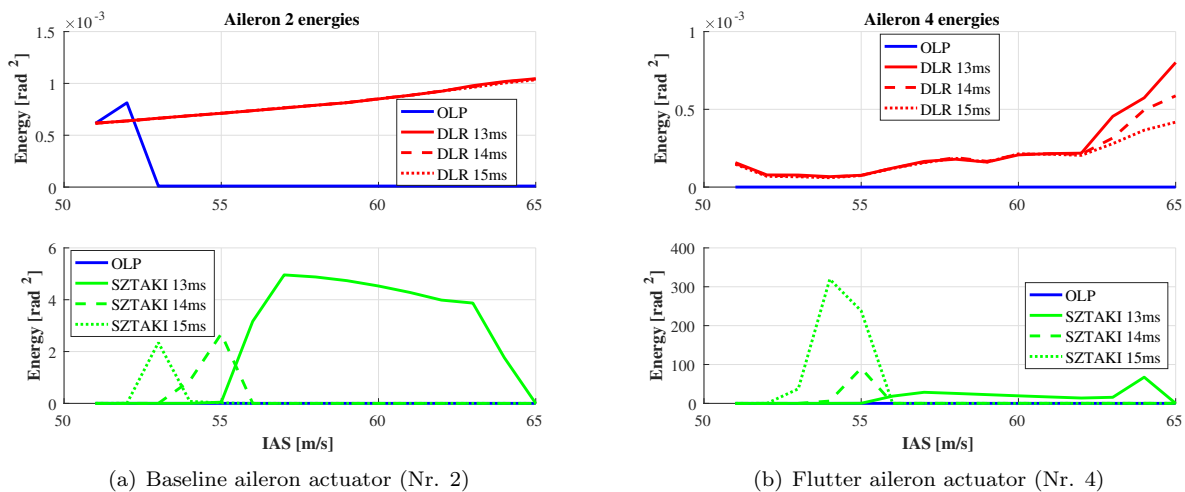


Figure 43. Energy content of aileron actuators

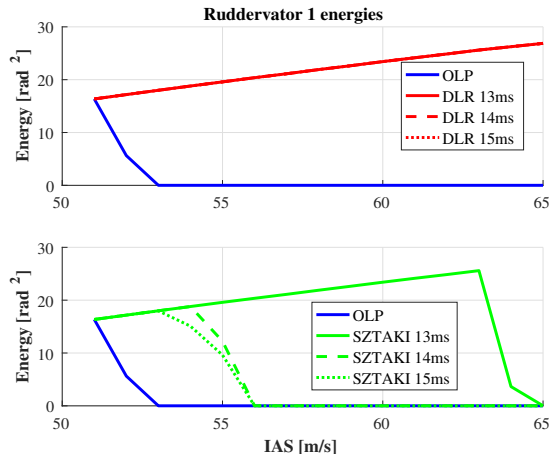


Figure 44. Energy content of baseline ruddervator actuator (Nr. 1)

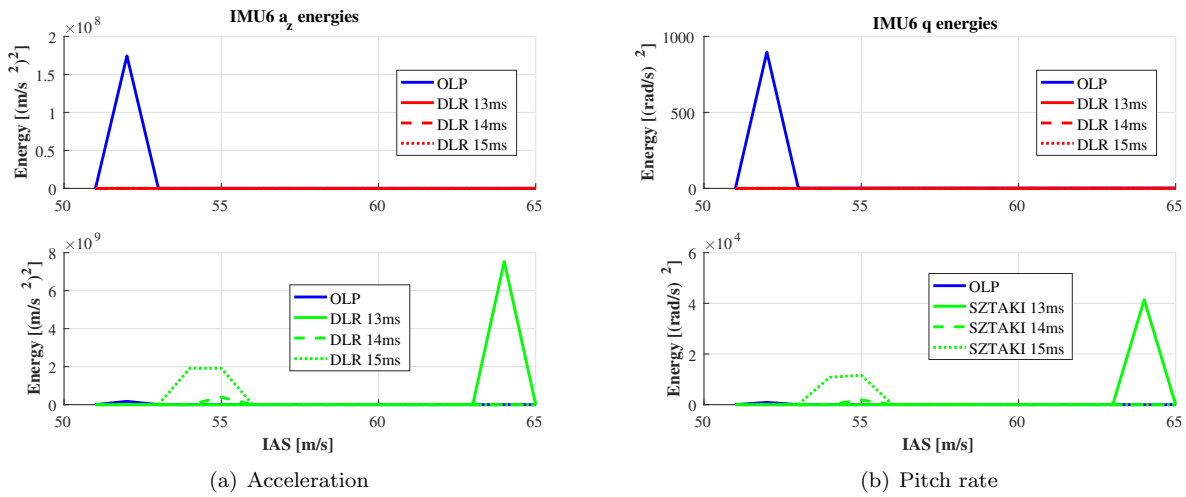


Figure 45. Energy content of wing IMU (Nr. 6)

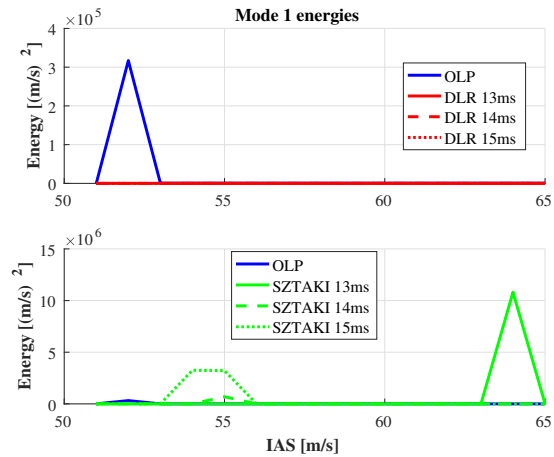


Figure 46. Energy content of wing flexible mode 1