AE2111-I: System Design – Project Reader

Work Package 1 to 3

Spacecraft Design

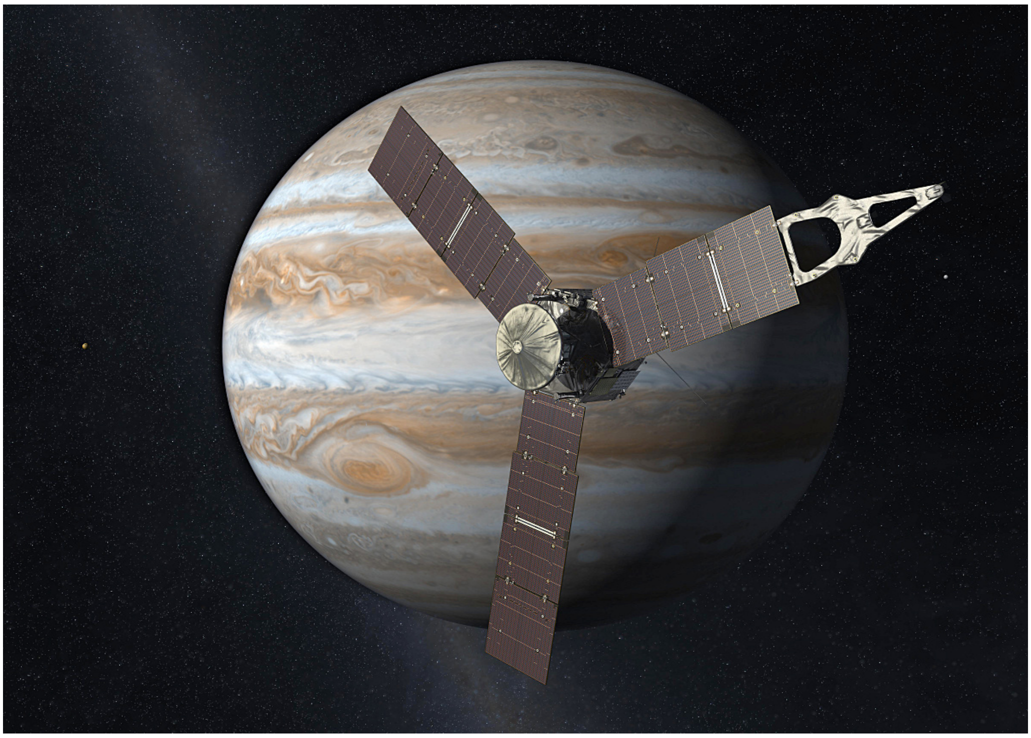


Illustration credits: NASA-JPL/Caltech

Prepared: Dr.B.V.S.Jyoti (B.V.S.Jyoti@tudelft.nl)

Delft University of Technology

Table of Contents

Abbreviations

ADCS	Attitude Determination and Control System
ECSS	European Cooperation for Space Standardization
EOL	End of Life
ESA	European Space Agency
MLI	Multi-Layer Insulation
MMOI	Mass Moment of Inertia
OSR	Optical Solar Reflector
P/L	Payload
S/C	Spacecraft
WP	Work Package
RTG	Radioisotope Thermoelectric Generator
TRL	Technology Readiness Level

1 Introduction

Welcome to the first part of the AE2111-1 system design project. This document contains a list of the space projects for the Fall 2021 course and the related Work Packages Description (the same description is valid for each 1 of the 3 different space projects). The AE2111-1 project aims at making you, the student, more familiar with the spacecraft design process. Keep in mind that, in this process, every choice you make may have a large influence on the following steps. It is therefore advised to make efficient use of all the tools you are provided with (such as software like MATLAB, Excel, CATIA, etc.). The design process often involves iterations and becomes much faster if you only need to change some parameters in a Matlab file or an Excel sheet, rather than doing all calculations repeatedly by hand.

Teamwork plays a vital role in this kind of project, and you will soon find out that good communication is the key to an efficient process. The Work Packages (WP) contain too much content to be handled by only a couple of individuals, which may have been the case in the projects of the first year. Performing tasks in parallel is paramount to a successful outcome of the project. It is advised to set up some project rules at the beginning of the project – should there be central meetings at the beginning or end of the session to discuss the progress, what time does the group collectively take a 15-minute break, are smoking breaks allowed (and if so, how long and how often), should people who are late bring cake? As advice, every group should appoint at least a chairman and a secretary on the first day of the project. The chairman will have the responsibility to lead regular short team meetings, and the secretary will take the minutes. These roles can also be regularly changed inside the team, for instance at the end of each Work Package. This way, most of the team members experience this kind of responsibility during the project. Another idea is to design a process which truly takes quality control into account. Not only do calculations need to be checked, but the writing does also – the latter may even be more important! Since this project is a group effort, everyone needs to clearly contribute towards group organization, design work, and formally presenting your findings in the form of reports and presentations.

Every group is assigned to a Teaching Assistant and your point of contact for assignment questions and clarifications. You may contact the author of the present assignment document at any time (whose name and e-mail address are indicated on pages 1 and 2).

The reader begins with an overview of the three different space projects that will be considered this year, one of which you and your peers shall be working on. Please consult Brightspace to check which space project is assigned to your group. For each Work Package 1 to 3, and sub-Work Package structure all the required deliverables are listed. A separate report shall be delivered for each Work Package (the delivery deadlines are indicated below in section 1.2, and in the Work Package descriptions). This release of the Project Reader contains Work Package 1, 2 and 3. Work Packages 4 and 5 will be released at a later stage and are the focus of the second quarter of the semester. The reader concludes with an appendix on Systems Engineering elements. You are recommended to study this appendix thoroughly; application of the tools discussed in the appendix is crucial to a successful outcome of this project.

For project rules, information on the project rules and regulations, you are referred to the general project guide, posted on Brightspace.

1.1 Use of assumptions

Throughout your project work, you will make assumptions and simplifications at various points. This is encouraged: the design you are making is ultimately a low-fidelity paper design. Furthermore, the design calculations should be finished in due time, such that they can be used in an iterative procedure. At the end, as for many engineering exercises, there is simply no time to make a 'perfect' design. Nonetheless, you should be aware of the assumptions and simplifications you are making, and document them diligently. This means, at a minimum, you should describe at least the following:

- The assumption itself.
- The result of the assumption.
- The validity of the assumption.

As an example, unrelated to this project itself, if one were to describe a stress-strain relation for the deformation of a material, then one would need to document 1) that the material elastic deflection is assumed to be proportional to the applied load; 2) the result of this is that the stress-strain relationship is linear; and 3) this assumption is valid because the deflections are small, meaning that plastic deformations are absent. If possible, it is desirable to have at least an order of magnitude estimate of the error caused by the assumption, and if applicable, whether the assumption is conservative or non-conservative (i.e., does it result in an under- or overestimate of the actual result). If it is non-conservative, would it be appropriate to use a safety factor, and if so, what do you deem to be appropriate? Finally, it is preferable to use identifiers when listing assumptions, such that it is convenient to refer to them.

1.2 Work Package reports and deadlines

The reports for each Work Package require the same format as taught in the course Technical Writing and must comply with the page limits shown in the table below. "Content" is defined as all pages from the introduction to the conclusion, thus including the introduction and conclusion itself. Appendices should *only* contain information that is non-essential to the reader to understand the report; information that is wrongly put in the appendix will result in a lower grade.

Technical drawings may be included as A3 pages; this will count as a single page. No other kind of information may be put on A3.

Note: Several CATIA drawings need to be delivered throughout the project. **As a rule, each group member shall participate in producing at least one of these drawings.**

In case Python or Matlab is used for any kind of calculation, the source code should be included in the report, such that it can be checked for plagiarism. The code does not count towards the page limit.

Page limits for the report:

	CONTENT	APPENDICES
WORK PACKAGE 1	20 pages	5 pages
WORK PACKAGE 2	45 pages	5 pages
WORK PACKAGE 3	30 pages	10 pages

Report deadlines:

	DEADLINE
WORK PACKAGE 1	**Monday, week 1.3, 13 September, 12:00 (noon) CEST**
WORK PACKAGE 2	**Monday, week 1.6, 4 October, 12:00 (noon) CEST**
WORK PACKAGE 3	**Monday, week 1.8, 18 October, 12:00 (noon) CEST**

One last remark: ensure to complete all the required deliverables in your report as a minimum. If it is not in the report, then it has not been done! The deadlines are non-negotiable.

2 References

[1] Lecture Notes for the course AE1222-II (Aerospace Design & System Engineering Elements I), TU Delft

[2] P. Fortescue, J. Stark, G. Swinerd, *Spacecraft Systems Engineering*, Wiley

[3] J. Larson, J. R. Wertz, *Space Mission Analysis and Design*, Wiley (Kluwer Academic Publishers)

[4] Lecture Notes for the course AE1110-I (Introduction to Aerospace Engineering I), TU Delft

[5] Lecture Notes for the course AE1110-II (Introduction to Aerospace Engineering II), TU Delft

[6] J. Larson et al., *Applied Space Systems Engineering*, Wiley (The Mc Graw Hill Companies, Inc.)

[7] H. J. Kramer, *Observation of the Earth and Its Environment: Survey of Missions and Sensors*, Springer-Verlag Berlin

[8] European Space Agency ECSS-E-30 Standard Part 2A, *Space Engineering / Mechanical – Part 2: Structural*, ESA Publications Division

2.1 Note on the References

In principle, Refs. [1], [2] and [3] in the above list include all the general information needed for completing the project. When additional information is needed, you can consult the other References or the World Wide Web in particular.

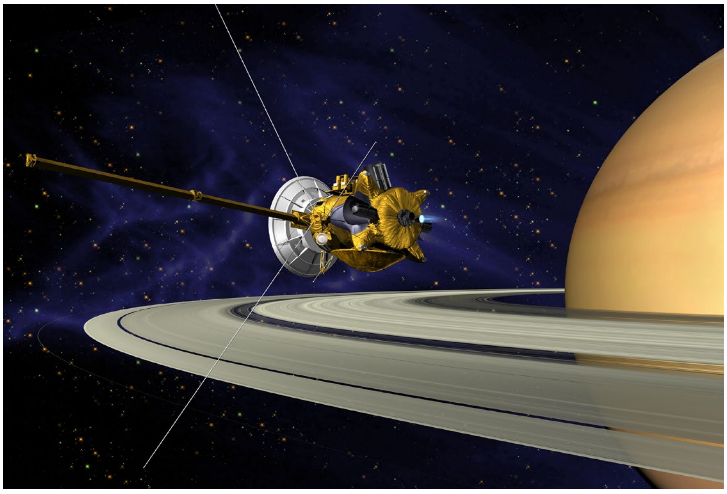
3 Description of the space projects

This section provides a general description of the mission objectives and top-level requirements for the three space projects proposed in the framework of the Fall 2021 course. These objectives and requirements shall be taken into account when you define your mission profile and your detailed list of driving requirements during Work Package 1.

Each one of the three proposed projects is represented by a satellite orbiting around an outer planet of our Solar System for scientific observation purposes. This satellite also carries a probe that will separate from it and orbit one of its selected moons. Your group will be randomly assigned to ***only one*** of these projects and will work exclusively on it. Note that the Work Package descriptions provided in the following sections are common to all the projects, but work needs to be done in a different way depending on the project to which you are assigned, due to the different mission orbit, objectives, and needs.

Note: the spacecraft part of the course AE2111-II (Aerospace Design and Systems Engineering Elements II), offered in September 2021, includes topics that may be helpful for a better understanding of the objectives and the work to be done for this project. It is therefore suggested to attend the lectures of that course and to use its study material as a reference.

3.1 Space Project 1: Saturn-Phoebe Orbiter



In the figure: Artist's impression of the Cassini-Huygens spacecraft (www.jpl.nasa.gov).

Mission Objectives

The mission objective is to conduct a scientific and geological study of Saturn and its moons using a spacecraft orbiting the planet. Following the Cassini-Huygens mission, this mission aims to carry out a detailed study of Saturn's surface, plasma environment, and magnetosphere by remote-sensing and optical imaging techniques. This spacecraft is also used as a transport vehicle for a moon orbiter. This orbiter will study and characterize the atmosphere and surface of Phoebe, one of Saturn's moons. The moon orbiter will communicate with the vehicle to receive commands and send data throughout its mission. The vehicle relays the data sent by the moon orbiter back to Earth. Saturn orbiter payload, thus include: a moon orbiter; a high-resolution camera (for capturing details as small as 1 m); a magnetometer (for characterizing the Saturn's magnetic field); plasma spectrometer (to investigate the plasma and solar wind in the Saturn's ionosphere) and an optical imaging instrument (to produce a high-resolution atlas of the Saturn's surface).

Top-Level Requirements

- Moon orbiter (to be treated as one of the payloads)
  - Mass**: 250 kg**
  - Size (H x L x B): **2.5 m x 2 m x 1 m**
- Total Payload Mass: **500 kg**
- Total Payload Size: **5 m x 5 m x 4 m**
- Total Payload Required Power (max; incl. power required to communicate with the moon orbiter): **550 W**
- Payload Operational Temperature Range: **180K – 300K**
- Delta-V required by the vehicle for manoeuvring from transfer orbit to final mission orbit (including all margins): **7 km/s**
- Mission Orbit: **elliptical, at 300 km minimum altitude above the Saturn surface, 51^{o} inclination.**
- Mission Duration**: at least 5 years in Saturn orbit**
- Launch Date: **2032**

3.2 Space Project 2: Uranus-Miranda Orbiter

In the figure: Artist's impression of the Voyager 2 spacecraft (www.jpl.nasa.gov).

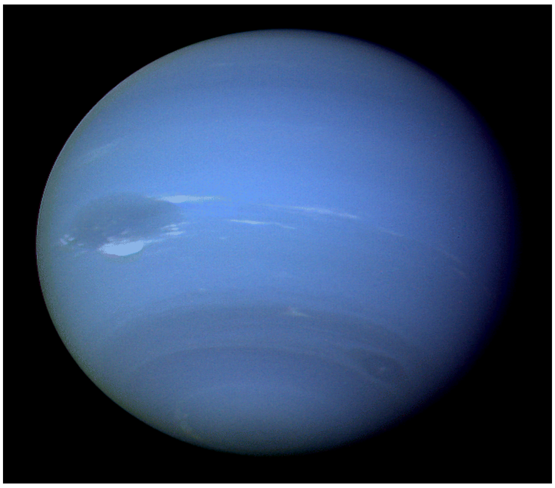
Mission Objectives

The mission aims to observe and characterize Uranus, the "ice giant" of our Solar System and its moon, Miranda. The scientific tasks include generating the chemical and mineralogical map of the upper atmosphere, an analysis of Uranus's icy mantle that surrounds a rock and ice core, a detailed characterization of surface and atmospheric temperatures, investigation of the radiation environment. This spacecraft is also used as a transport vehicle for a moon orbiter. This orbiter will study and characterize the atmosphere and surface of Miranda, one of Uranus's moons. The moon orbiter will communicate with the vehicle to receive commands and send data throughout its mission. The vehicle relays the data sent by the moon orbiter back to Earth. Uranus orbiter payload thus includes a moon orbiter, a thermal camera (for mapping the temperature profiles), a high-resolution camera (to capture details as small as 1 m), a multi-wavelength spectrometer (to analyse the icy mantle), and a radiation monitor (to characterize the radiation environment).

Top-Level Requirements

- Moon orbiter (to be treated as one of the payloads)
  - Mass: **150 kg**
  - Size (H x L x B): **2 m x 1 m x 1 m**
- Total Payload Mass: **300 kg**
- Total Payload Size: **4 m x 3 m x 2 m**
- Total Payload Required Power (max; incl. power required to communicate with the moon orbiter): **350 W**
- Payload Operational Temperature Range: **180K – 300K**
- Delta-V for manoeuvring from transfer orbit to final mission orbit (including all margins): **8 km/s**
- Mission Orbit: **circular, 200 km altitude above Uranus surface, 90^o inclination (polar)**
- Mission Duration: **at least 4 years in Uranus orbit**
- Launch Date: **2032**

3.3 Space Project 3: Neptune-Triton Orbiter



In the figure: Image of Neptune taken by Voyager 2 (http://voyager.jpl.nasa.gov/).

Mission Objectives

The mission objective is to perform scientific observation of Neptune and its moon, Triton. The scientific goals include: to characterize and map the atmosphere, weather, and its inner core, analysis of atmospheric temperatures, and investigation of its magnetic field and radiation belt. This spacecraft is also used as a transport vehicle for a moon orbiter. This orbiter will study and characterize the atmosphere and surface of Triton, one of Neptune's moons. The moon orbiter will communicate with the vehicle to receive commands and send data throughout its mission. The vehicle relays the data sent by the moon orbiter back to Earth. Neptune orbiter payload thus includes a moon orbiter, a magnetometer (for characterizing the Neptune's magnetic field), a high-resolution camera (to capture details as small as 1 m), imaging spectrograph (to determine the composition of atmospheric gases), multi-wavelength spectrometer (to analyse the weather patterns), and a radiation monitor (to characterize the radiation environment).

Top-Level Requirements

- Moon orbiter (to be treated as one of the payloads)
  - Mass: **200 kg**
  - Size (H x L x B): **2.5 m x 1.5 m x 1 m**
- Total Payload Mass: **250 kg**
- Total Payload Size: **3 m x 3 m x 1.5 m**
- Total Payload Required Power (max; incl. power required to communicate with the moon orbiter): **280 W**
- Payload Operational Temperature Range: **180K – 300K**
- Delta-V for manoeuvring from transfer orbit to final mission orbit (including margins): **10 km/s**
- Mission Orbit: **circular, 250 km altitude above Neptune surface, 0^o inclination (equatorial)**
- Mission Duration: **at least 3 years in Neptune orbit**
- Launch Date: **2032**

4 Work Package 1: Initial Sizing

This Work Package deals with the mission functional analysis, the definition of the spacecraft requirements and the mission profile, and the initial sizing of your spacecraft.

WP 1.1		**WP 1.2**		**WP1.3**		**WP1.4**
Functional Analysis	➔	Spacecraft Requirements	➔	Mission Profile	➔	Initial Sizing

Timeline: **Weeks 1.1 and 1.2** (2-weeks total duration)

Deadline for Report Delivery: **Monday, week 1.3, 13 September, 12:00 (noon) CEST**

Content and appendices limited to: 20 pages (content) plus 5 pages (appendices)

Each report shall include, as a minimum, all the Deliverables listed in the sub-WP descriptions provided below (each deliverable is identified by a unique code, in the form *Dx.y.z*). The names and student numbers of the authors shall be present on the cover page. The report shall also include a work division table, indicating which group members have contributed to each one of the tasks.

Deliver the printed report or electronic version to the Teaching Assistant assigned to your group.

4.1 WP 1.1 Functional Analysis

Tasks

A. Search for 5 existing (or planned) similar missions and identify for these missions:
   a. the mission objectives and, if possible, the associated mission requirements;
   b. the main elements that make up the mission and their main function(s);
   c. the main performance and design characteristics of these elements.

B. Define the mission elements with which the design process will interact (e.g.: mission operations, ground system, launch system, etc.).

C. List the main objectives to be achieved in the design process (related to, e.g., payload capability, performance, structure, etc.). Mention at least 10 objectives.

D. For each one of the identified design objectives, list the main functionalities that the design must provide in order to accomplish it.

E. List all the design parameters that are already available (e.g., from the project description provided in Section 3) and the ones which still need to be identified.

Deliverables

D1.1.1. Comparative table including all the relevant information collected for the reference missions identified under Task A above.

D1.1.2. Block diagram showing the outcomes of Task B above (hint: you can put the item to be designed in the middle, and the relevant mission elements around it).

D1.1.3. Numbered lists obtained from Tasks C, D and E above (note: lists should be as complete as it is possible at this stage of the design, and a clear justification for each listed item shall be provided).

4.2 WP 1.2 Spacecraft Requirements

Tasks

A. Familiarize with the scientific objective of the mission and collect all the relevant information on the characteristics of the outer planet and its moon (e.g., gravitation coefficient, radius, atmospheric density, orbital period, eclipse periods, solar intensity received, albedo, etc.).

B. Search for at least 5 existing (or planned) spacecraft to be used as a reference for your design, i.e. for which the S/C specifications and characteristics are similar to the ones expected in your mission. Collect information on the design of the identified reference spacecraft (e.g., mass and power budgets, size, structure configuration, payload characteristics, orbital parameters, etc.).

C. List a detailed set of requirements for your design, taking into account the top-level mission requirements (from the project description provided in Section 3) and the characteristics of the identified reference spacecraft. Remember: the requirements shall be SMART (Specific, Measurable, Achievable, Realistic, Time-bound)!! Whereas possible express them as numbers, preferably in the form of a range of values.

D. From the complete list of requirements, identify at least **10 *driving requirements*** for your design, i.e., requirements that will play a major role in the design process. Provide an adequate justification for your choice.

Deliverables

D1.2.1. Table including all the relevant data on the characteristics of the outer planet and its moon.

D1.2.2. Comparative table including all the relevant design data collected for the reference spacecraft identified under Task B above.

D1.2.3. Detailed list of design requirements, with a clear indication of the driving requirements.

4.3 WP 1.3 Mission Profile

Tasks

A. Using methods based on statistical data (as, for instance, the ones presented in Ref. [1]), make a first vehicle level estimation of the spacecraft dry mass, power, size, reliability, and cost. Hint: you can use the reference spacecraft characteristics found in the framework of WP 1.2 to evaluate the accuracy of the methods you have adopted, or even to develop your own estimation method.

B. Preliminarily define the orbital parameters foreseen for your mission, also based on the orbit requirements provided in the project description in Section 3.

C. Estimate the total Δv required by the mission and based on this; the total propellant mass needed (note: this value depends on the type of engines/thrusters you use!! If this has not been decided yet in detail, you may still have several open options for the propellant mass at this stage…). Include in your estimation all the relevant mission phases (e.g., launch, transfer, orbit corrections, station keeping, etc.). Don't forget to take into account proper margins!! Hint: you can use the data referred to similar missions, found in the framework of WP 1.1, as a baseline for your estimation. Note that the Δv needed for moving from the interplanetary transfer orbit to the final mission orbit is already provided, as a top-level requirement, in the project description in Section 3. The detailed mission profile of the moon orbiter such as the Δv required by it to reach the moon, is not needed.

D. Choose an adequate launcher for your mission and, consequently, indicate the launch location, sketch the ascent profile, estimate the maximum launch loads and the required minimum longitudinal and transversal natural frequencies, and identify the payload adapters available for use and their characteristics.

E. Based on the information obtained from the previous Tasks, generate a detailed mission profile and a rough timeline for the mission. Note: Include the moon orbiter's separation stage in the total mission timeline.

Deliverables

D1.3.1. Vehicle-level estimation of the spacecraft dry mass, propellant mass, power, size, reliability, and cost.

D1.3.2. Complete mission profile (including a description of the launcher requirements and characteristics) and preliminary mission timeline up to EOL (End of Life).

D1.3.3. Graphical sketch of the spacecraft trajectory during all the phases of the mission, including launch, ascent, separation, parking orbit(s) and final orbit. Note: Moon orbiter separation stage is optional for this sketch.

D1.3.4. Table showing the orbital parameters characterizing the spacecraft final orbit.

D1.3.5. Estimated total Δv required by the mission, including a table showing its breakdown into all the contributions related to different mission phases and operations. Note: the moon orbiter has its own propulsion system on board so no need to calculate the Δv for its mission profile.

4.4 WP 1.4 Initial Sizing

Tasks

A. Using methods based on statistical data and/or empirical relationships (as, for instance, the ones presented in Refs. [1] and [2]), define the preliminary mass and power budgets for the spacecraft, i.e., the mass and power distribution among all the relevant subsystems. Pay particular attention to the definition of proper design margins in your budgets.

B. Based on the information gathered so far, sketch at least three different preliminary architectures of the spacecraft, illustrating the configuration and conceptual layout of all components (e.g., antennae, solar arrays, batteries, nuclear generator, payload units, external skin). Draw at least one of these preliminary sketches as a 3-view drawing (American projection) in CATIA, with outer dimensioning (including centrelines are a good practise). for the preliminary architectures and the CATIA drawing, treat the moon orbiter as a payload.

C. For each one of the proposed preliminary architectures, estimate the vehicle MMOI (Mass Moment of Inertia) both for the deployed and un-deployed state. No need to estimate the MMOI of the moon orbiter.

Deliverables

D1.4.1. Tables showing the preliminary mass and power budgets for the spacecraft.

D1.4.2. Sketches of different optional preliminary spacecraft architectures.

D1.4.3. CATIA 3-view drawing (American projection) of at least one preliminary spacecraft architecture.

D1.4.4. Estimation of the vehicle MMOI in the deployed and un-deployed state.

5 Work Package 2: Spacecraft Design

This Work Package deals with the detailed design of the spacecraft subsystems, the revision of the mass and power budgets, and the definition of the final spacecraft architecture.

WP 2.1		**WP 2.2**		**WP2.3**
Subsystems Design	➔	Revised Budgets	➔	Spacecraft Architecture

Timeline: **Weeks 1.3, 1.4 and 1.5** (3-weeks total duration)

Deadline for Report Delivery: **Monday, week 1.6, 4 October, 12:00 (noon) CEST**

Content and appendices limited to: 45 pages (content) plus 5 pages (appendices)

Each report shall include, as a minimum, all the Deliverables listed in the sub-WP descriptions provided below (each deliverable is identified by a unique code, in the form *Dx.y.z*). The names and student numbers of the authors shall be present on the cover page. The report shall also include a work division table, indicating which group members have contributed to each one of the tasks.

Deliver the printed report or electronic version to the Teaching Assistant assigned to your group.

5.1 WP 2.1 Subsystems Design

Tasks

Note that all the Tasks below, related to different subsystems of the spacecraft, need to be carried out in parallel, because several key requirements and characteristics of the different subsystems are significantly coupled each other. Hint: divide the whole group into sub-groups (or individuals), each one working on a different subsystem, and plan regular "system engineering meetings" (e.g. daily) to exchange with the other sub-groups the information related to the work done on each subsystem.

A. Attitude Determination and Control System (ADCS)

1. Define the key subsystem requirements and functions (e.g., spacecraft stabilization and station keeping, precise orientation of the payload instruments, orientation of the solar arrays, etc.).
2. Evaluate whether different modes of operation of the ADCS are needed or not.
3. Evaluate the external and internal disturbances (forces and torques) acting on the spacecraft.
4. Define the type, number and size of attitude sensors and attitude control actuators needed by the spacecraft (note: consider sufficient redundancies!!)
5. Select the sensors and actuators for the spacecraft among commercially existing models, and dimension your subsystem based on the characteristics of the selected components.
6. Develop a block diagram that identifies the various functions that need to be performed by the ADCS software.

B. Propulsion

1. Define the key subsystem requirements and functions (e.g., orbit insertion, orbital manoeuvres, etc.). Note: design your spacecraft assuming that the mission starts in the interplanetary transfer orbit, i.e. including the Δv needed by all the mission phases up to entering the transfer orbit is provided by the systems external to the spacecraft. ***Note that the Δv needed for moving from the interplanetary transfer orbit to the final mission orbit is already provided, as a top-level requirement, in the project description in Section 3.*** The propulsion system required by the moon orbiter is already taken into account so just treat it as a payload component.
2. Define and quantify the eventual forces to be overcome by the propulsion system (gravity, drag, solar radiation etc.).
3. Revise your estimation for the total Δv required by the mission and, based on it, select the thruster(s) and the propellants to be used (note: decide at this stage if a "low thrust" or a "high thrust" strategy has to be adopted, and select your thrusters accordingly).
4. Evaluate the total propellant mass needed by the mission. Don't forget to consider sufficient margins!!
5. List and characterize all the components of the propulsion systems; don't consider the propellant tank(s), that will be studied under the structures subsystem. If possible, select a commercially available complete propulsion system.
6. Generate a sketch showing the number and the location of the various propulsion components in the spacecraft. Include in the sketch also the major characteristics of the components shown.

C. Thermal Control

1. Define the key subsystem requirements and functions (e.g., operational temperatures of the spacecraft components, acceptable thermal gradients, etc.
2. Define the thermal environment of the spacecraft (solar intensity, planet flux, albedo, etc.). Focus should be on defining the thermal environments that are most critical for the design, i.e. those environments that lead to max/min temperature conditions.
3. Estimate the internal heat generated in the spacecraft.

4. Generate a straw-man subsystem design using only passive means of thermal control like paints, MLI (Multi-Layer Insulation) and OSR (Optical Solar Reflector).
5. Using the equations provided in Ref. [1], make a first estimation of the maximum and minimum equilibrium temperature of both the spacecraft body and the solar array (when present), in the case when only passive thermal control systems are used.
6. Based on your equilibrium temperature estimation and on the temperature requirements for the different components, define a thermal control strategy for the spacecraft and select (or design) the appropriate subsystem components needed for it. Where possible, use commercially existing components or materials.

D. <u>Power</u>

1. Define the key power subsystem requirements and, in particular, the total amount of power required by the spacecraft during all the mission phases.
2. Quantify the available solar energy, i.e. the solar intensity and the direction of solar radiation, the duration of day time and night time (eclipse) periods, the energy received from the Earth.
3. Based on your mission power requirements, design the power subsystem using two options: first, ***solar arrays only*** and ***second, RTGs only***.
    i. For the first design option, make a preliminary dimensioning of the solar arrays and, in particular, a first estimation of their size (<u>note: detailed design of the solar arrays will be done during WP 3!!</u>). You should consider, for instance, whether you would like a fixed array or an array that can be rotated over one or two axes to allow for ideal lighting conditions.
    ii. For the second design option, make a preliminary dimensioning of the RTG configuration. Consider the type of fuel, number of RTGs required, materials of the supporting structure, mounting on the spacecraft. Discuss the measures taken to ensure the safety from radiation effects to the on-board instruments and external environment.
4. Based on your preliminary designs, compare the two options in terms of pros and cons. Thus, <u>using this trade-off, select one of them and carry out the next steps</u>.
5. Based on your power requirements, evaluate the main characteristic properties of the spacecraft batteries (e.g., specific energy, energy density, capacity, mass, size, etc.).
6. Select the batteries and the power control components among commercially existing models.
7. Generate a block diagram showing the main subsystem components, the power flow and the efficiencies of power conversion.

E. <u>Structures and Mechanisms</u>

1. Define the key requirements for the structures and mechanisms subsystem.
2. Define all the loads acting on the spacecraft during its operations (note: take into account the launch loads that you have already estimated during WP 1!!). Define the safety factors to be used (you can use, for instance, the values provided by the relevant ESA ECSS standards, see Ref. [8]). Most likely the most stringent condition will be at launch
3. Preliminarily dimension and design the primary structure of your spacecraft, also choosing adequate materials for it. Based on the relevant requirements for the spacecraft and its components, decide if the structure has to be designed for stiffness and/or strength and/or internal pressure. Hint: for your analysis you may consider the primary structure to resemble a hollow cylindrical beam.
4. Based on the amount of propellant(s) needed and the required storage pressure, dimension and design the propellant tank(s), also choosing adequate materials (note: the tank(s) can eventually be used as primary structural elements!). Whereas possible, select commercially existing tanks.
5. Preliminarily dimension and design the supporting structure of the solar arrays or RTGs (if used) of your spacecraft (i.e., the mechanical structure on which the array/RTG is mounted), also choosing adequate materials for it. Based on the relevant requirements for the spacecraft and its

components, decide if the structure has to be designed for stiffness and/or strength and/or internal pressure.

6. Define the other mechanisms eventually needed by the spacecraft (e.g. for deployment, pointing, separation, gimbal, etc.) and select them among commercially existing components.

F. Payload

1. Based on the general payload requirements and instruments list provided in the project description of Section 3, choose the payload components to be used by your spacecraft among commercially available or previously used/proposed ones for other space missions. Hint: for accomplishing this task, you can also make use of the information included in Ref. [7]. *Don't forget to include the moon orbiter as one of your payload components!*
2. Define the position and interface constraints for all the payload components.

G. Other Subsystems

1. For the remaining subsystems (command and data handling, communications, navigation, etc.) define their key requirements and functions and preliminarily select their main components. Don't go too much into the details for these subsystems; just analyse their general architecture and their impact on the design of the other spacecraft subsystems. Note that the moon orbiter communicates with the vehicle to receive commands and send data throughout its mission. The vehicle relays the data sent by the moon orbiter back to Earth.

Deliverables

D2.1.1. For each one of the Tasks and sub-Tasks described above, document in detail your choices, calculations and outcomes. Provide detailed characteristics of the components you have chosen, including their mass, size and power requirements; where possible, include pictures, sketches and/or drawings of the components.

5.2 WP 2.2 Revised Budgets

Tasks

A. Using the result obtained by WP 2.1 (in particular, the requirements and characteristics of the components chosen for the subsystems), define detailed mass and power budgets for the spacecraft. Clearly indicate the margins that are available.

B. Compare the new budgets to the preliminary ones defined in the framework of WP 1 and highlight eventual changes or significant differences.

C. Verify that the mass and power budgets are compliant to ALL the mission requirements and, if necessary, re-iterate the subsystems design obtained during WP 2.1.

Deliverables

D2.2.1. Tables showing the detailed mass and power budgets for the spacecraft.

5.3 WP 2.3 Spacecraft Architecture

Tasks

A. Based on the operational requirements, the interfaces and the dimensions of the subsystems and their components, define a detailed architecture of the spacecraft including all the relevant subsystems.

B. Draft your detailed spacecraft architecture as a CATIA 3-view drawing (American projection) and compare it to the preliminary one(s) obtained in the framework of WP 1. Include a 3D isometric view with annotations, and a Bill of Materials (BOM) to fully identify each component of the larger assembly.

Deliverables

D2.3.1. CATIA 3-view drawing (American projection) of the spacecraft architecture. The drawing shall include one or more 3D isometric views and the necessary 2D views/cut-offs needed to show the position and interfaces of all subsystems. Also include the Bill of Materials (BOM).

6 Work Package 3: Subsystem Design

In WP 2.1, you were asked to design the power subsystem using two options: solar arrays and RTGs. Irrespective of your selection, this Work Package deals with the detailed sizing and design of the ***solar arrays configuration*** for your mission.

WP 3.1		WP 3.2		WP3.3		WP3.4
Preliminary Design	➔	Baseline Selection	➔	Configuration Design	➔	Subsystem Integration

Timeline: **Weeks 1.6 and 1.7** (2-weeks total duration)

Deadline for Report Delivery: **Monday, week 1.8, 18 October, 12:00 (noon) CEST**

Content and appendices limited to: 30 pages (content) plus 10 pages (appendices)

Each report shall include, as a minimum, all the Deliverables listed in the sub-WP descriptions provided below (each deliverable is identified by a unique code, in the form *Dx.y.z*). The names and student numbers of the authors shall be present on the cover page. The report shall also include a work division table, indicating which group members have contributed to each one of the tasks.

Deliver the printed report or electronic version to the Teaching Assistant assigned to your group.

6.1 WP 3.1 Preliminary Design

Tasks

A. Identify at least 5 different types of solar cells that may potentially be used in your spacecraft (hint: look into web sites of solar cell companies like Spectrolab, Emcore, AzurSpace). For each solar cell type, provide its key properties including, as a minimum: power provided in peak power point, short circuit current, open cell voltage, mass, efficiency, solar absorptivity and emissivity, surface area per cell, performance degradation over time, thermal efficiency. You should clearly indicate the conditions (in particular the cell temperature) for which the provided data are valid.

B. Make a detailed prediction of the power provided by the solar arrays throughout the spacecraft lifetime. Start as a baseline from the first-order estimation made during WP 2, and refine it by taking into account all relevant additional effects (e.g.: solar cells degradation over time, eclipse periods, variation of the solar cells efficiency with temperature, batteries efficiency, cable losses, possible misalignments of the solar arrays, etc.).

C. Define an adequate set of trade-off criteria for the selection of a solar cell, based on the identified power requirements. Perform the trade-off and select the solar cell to be used in your spacecraft.

D. Based on the actual solar cell selected and its specific characteristics, define 3 different conceptual designs for the solar arrays configuration of the spacecraft (in terms of geometry, shape, size, number of cells, etc.) and draft them as CATIA drawings (you can choose the preferred drawing type that best communicates your design, e.g., use a 3-view drawing—American projection—or a 3D isometric view).

Deliverables

D3.1.1. Comparative table of the identified solar cell concepts key properties (including, eventually, pictures and plots).

D3.1.2. Detailed calculations and plots for estimating the power provided by the solar arrays during all mission phases until spacecraft's EOL.

D3.1.3. Trade-off table(s) for the selection of the solar cell used in the spacecraft.

D3.1.4. CATIA drawings of 3 different solar arrays configurations for the spacecraft.

6.2 WP 3.2 Baseline Selection

Tasks

A. Define a set of trade-off criteria and an appropriate trade-off strategy for selecting one of the 3 different solar arrays configurations; indicate and justify, in particular, the weight to be given to each criterion.

B. Analyse the 3 proposed solar array configurations in terms of at least the following aspects: total area needed to allocate the required power generation; required shape, dimensions and number of panels; materials, strength and stiffness; design ratios (power to mass, power to area, length to width); mounting on the spacecraft and orientation to the Sun.

C. Perform a trade-off of the 3 solar arrays configurations using the criteria and weights defined in Task A above. Justify, also using the results of the analysis performed for the Task B above, ALL the scores given and the final configuration chosen.

Deliverables

D3.2.1. Detailed analysis of the 3 identified solar arrays configurations with respect to the aspects listed in Task B above.

D3.2.2. Trade-off table(s) for the selection of the solar arrays configuration, including an adequate and clear explanation of all the scores given.

6.3 WP 3.3 Configuration Design

Tasks

A. Determine the size and materials of the supporting structure of the solar arrays and the position of the clamps between different panels. Base your design on the necessity of enabling a safe launch of the solar arrays structure, which translates into a requirement for its eigen-frequency to be higher than 75 Hz.

B. Design the interfaces of the solar arrays to the rest of the spacecraft. Consider in particular: the electrical interfaces; the mechanisms for orienting the panels towards the Sun; the hold-downs to be used during transport and launch.

C. Design and analyse the kinematics of the solar arrays deployment and of their pointing mechanism.

D. Make a CATIA kinematic model of the spacecraft solar arrays. Either include illustrations of different solar array positions during deployment and/or append a movie of the linkage system moving from the initial to final position).

Deliverables

D3.3.1. Complete and detailed design of: solar arrays structure; interfaces to the rest of the spacecraft; deployment and pointing mechanisms.

D3.3.2. CATIA drawings and/or kinematic model of the deployment mechanism of the spacecraft solar arrays.

6.4 WP 3.4 Subsystem Integration

Tasks

A. Briefly describe the integration of the main and secondary structure items of the solar arrays to the spacecraft, including deployment and folding of the panel stack.

B. Briefly discuss the mechanical effects of the solar arrays deployment process on the attitude of the spacecraft.

C. Describe the effects of the deployed status of the solar arrays on the attitude of the spacecraft in its final orbit in relation to the relevant disturbance forces as drag, solar wind, radiation pressure.

Deliverables

Outcomes of Tasks A, B and C above, including preliminary calculations whereas necessary.

7 Appendix A – Introduction to Systems Engineering Elements

Systems engineering elements form an important part of this project. Proper use of systems engineering (SE) is vital for the successful outcome of any design project; even a successful project that does not utilize SE will still likely have an inferior outcome to what the outcome would have been, had the project utilized SE.
It shall be noted that examples listed in this appendix often relate to spacecraft subsystems not treated in the project, or even to aircraft. Nonetheless, it should be self-evident how to translate these examples to the project itself.

7.1 Overview of systems engineering

SE is the field of engineering concerned with managing the design of complex systems. A system is a combined set of elements designed to fulfil a certain objective; in the context of this project, evidently a spacecraft is a system. It consists of a multitude of subsystems that all need to be designed simultaneously, taking into account each other's performance.
At first sight, it may seem a good idea to design each subsystem to its optimum performance, i.e. try to make each subsystem as light and cheap as possible, while maximising performance. However, such a design philosophy can grow chaotic rather soon. Consider the design of the empennage. A smaller horizontal tail will obviously be lighter, but it will also be less effective and have a higher drag coefficient during flight. Thus, the horizontal tail should be designed with certain requirements in mind, such that the interfaces with other subsystems (such as the wing positioning and the propulsion system) do not cause any issues.
Particularly in projects with a significant number of people working on it, it is not sustainable to assume that a change in the design of one subsystem can easily be accounted for in the design of other subsystems. For example, if it is only realised at a very late stage of the design that the winglets provide less drag reduction than initially thought, the engines may already have been selected, and the fairings may have already been designed. Redesigning this may prove to be very costly.
Thus, it is essential that at an early phase of the design, preferably even before the actual design started, clear requirements are set for each subsystem. For example, even though the actual drag coefficient of the complete aircraft is evidently unknown at the initial stage of the design, the designers of the landing gear system should already be able to count on a certain maximum weight of the aircraft. It is then also clear that an increase in performance of an individual subsystem may not necessarily lead to increases in the performance of other subsystems: for example, if the fuel system turns out to be lighter than expected, then the landing gear system may already have been designed and it cannot be taken into account for a reduction in weight of the landing gear system.
SE is the field concerned with managing all these interfaces, and seeing a product as a system, rather than just the sum of its elements. Although proper systems (and business) engineering considers more aspects, this project will be restrained to the following elements:

1. Functional analysis: to ensure that the product will be able to fulfil its mission, a functional analysis is performed. This analyses the functions that the solution should be able to perform (independent of the actual design of the solution).
2. Requirement analysis: a requirement analysis is performed to establish the requirements that the solution should meet to fulfil its mission. This includes requirements flow down, i.e., the development of system-wide requirements into subsystem requirements.
3. Design option generation: based on the requirements outlined previously, design options may be generated.
4. Trade-off: having selected a reasonable number of design options, a trade-off can be performed to establish which design option meets the requirements best.
5. Further design: insofar they weren't done for the trade-off, this is the part that most commonly is thought of to include the "calculations". This may include any additional iterations necessary.

6. Requirement compliance: after the design is finalised, it is verified that all requirements previously set out are met.

As can be seen above, design of a complex system is much more than just “calculations”, even if they are more enjoyable to perform than systems engineering. Indeed, doing calculations without following the proper process is bound to lead to design faults. In this project, you are thus strongly encouraged to follow the above steps carefully. Furthermore, to demonstrate that you used a system engineering approach, you are encouraged to use the same report structure for each subsystem, e.g. to have each subsystem have its own dedicated chapter, with sections dedicated to each of the above items. This ensures a consistent structure in the report, improving its quality.
Each of the above elements (except for the “Further design” phase) will now be elaborated upon.

7.2 Functional analysis

In the design of a (sub)system, the first step to perform is a functional analysis. The goal is to identify the functions that the (sub)system should perform. Figure 1 shows a functional flow diagram of the top-level functions that a certain spacecraft should perform (the functional analysis required in WP1 should go in approximately the same amount of detail) (INCOSE, 2000).
Figure 1 shows the top-level functions of the STS flight mission. Certain functions can be divided in subfunctions. For example, function 4.0 can be detailed as “4.1: Provide electric power”, “4.2: Provide attitude stabilisation”, etc. These subfunctions could then be further subdivided into even smaller functions, etc. A functional analysis for subsystems may be performed in similar fashion.
In certain cases, it is the function of a system to provide the *minimum* of something. For example, the vertical tail should naturally provide as little drag as possible, yet it would be wrong to state that a function of the vertical tail would be “Provide minimum drag”. This function could instead be better described as “Provide drag performance”. Similar wording holds for e.g., the environmental control subsystem providing power performance (Jackson, 1997).
Two things deserve special attention:

- Functions should *always* start with a verb.
- Functions should *always* include an identifier, to make it easier to cross-reference (e.g., rather than referring to a function “Transfer shuttle to OPS orbit”, one can simply refer to function 3.0).

In this project, it is not expected that you provide flowcharts for the functions, to save time. It is sufficient to present the functions in a simple table or bullet list. Furthermore, the functional analysis you perform should be to the level of detail that you are designing, and it is not expected that you go into more depth. This means that sometimes, you will indeed only have one or two functions to consider for a subsystem.

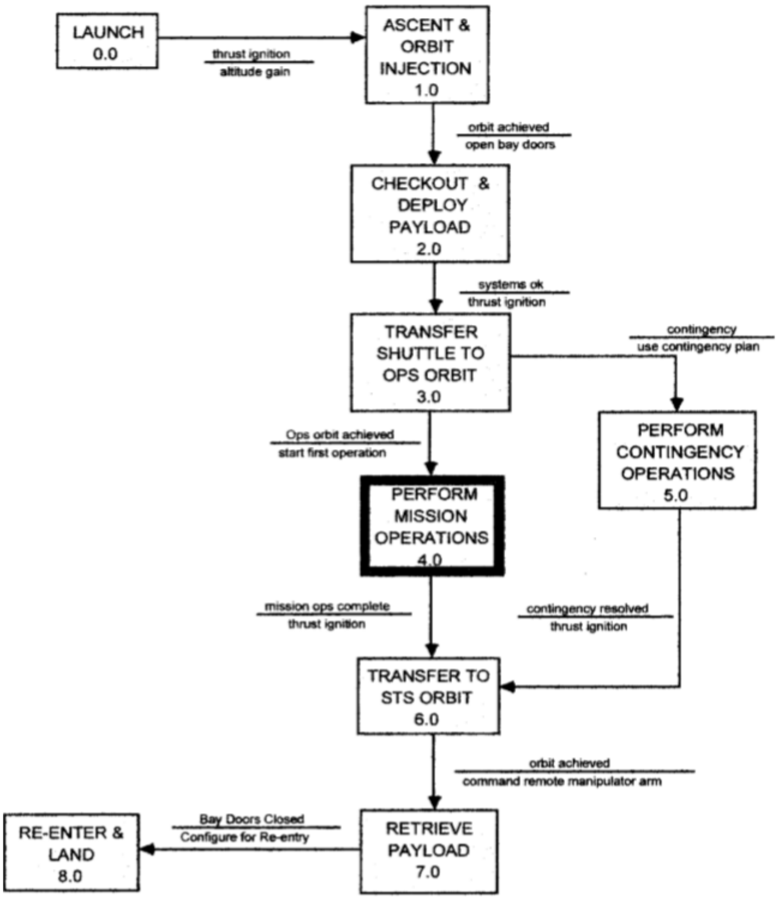


Figure 1: Functional flow diagram of STS flight mission.

7.3 Requirement analysis

The functional analysis can then be used to establish the functional requirements. Functional requirements describe what the system shall *do*. An example of a functional requirement would be

> *REQ-MIS-01: The spacecraft shall examine the atmospheric composition of Saturn.*

It should be noted that each functional requirement should follow from functions identified in the functional analysis. Functional requirements ought to be completed with non-functional requirements however, to specify how well a system should perform a certain task. Following the above system requirement, examples of corresponding non-functional requirements would be

> *REQ-MIS-01-01: The spacecraft shall be able to establish the presence of molecules in the atmosphere with mass up to at least 500 amu.*
>
> *REQ-MIS-01-02: The spacecraft shall be able to establish the distribution of particles in the atmosphere with a 2-sigma error of at most 2%.*

Additional requirements also follow from laws and regulations. In the case of aircraft, the CS, FAA and ICAO requirements are obvious sources of many requirements. Yet another common source of requirements is the currently existing infrastructure. For example, existing communication systems may impose requirements on the communication system that can be used aboard an aircraft.

7.3.1 Requirement flow down

After defining system requirements, they can be translated into system elements. The system is decomposed into subsystems, with each subsystem responsible for a major function of the system. The subsystems are allocated requirements in various manners:

- Requirements are allocated unmodified to a lower-level element. For example, a system requirement that the spacecraft shall have a minimum natural frequency of 26 Hz may be directly allocated to the structure subsystem.
- The requirement is budgeted over lower-level elements. For example, a maximum power consumption is distributed over various subsystems, such that the system requirement is met.
- The requirement is modified into a new requirement. Functional requirements are often changed into a new requirement. Consider a system requirement that a newly designed launcher should be able to send a 1000 kg payload to LEO. For the propulsion subsystem, this would impose a subsystem requirement on the ΔV that would be required. By assuming a certain specific impulse for the fuel (even without selecting the fuel yet) and a certain structural coefficient, the fuel mass can be estimated. By assuming the fuel density, a fuel tank volume may also be estimated. Four subsystem requirements are then quickly identified: for the fuel subsystem, there is now a required minimum specific impulse and maximum density. For the fuel tank subsystem, there is now a minimum required fuel tank volume. Finally, as there is now a maximum allowed structural coefficient, an estimate for the dry mass is available, and thus a mass budget can be made for the various subsystems (i.e., each subsystem gets allocated a certain amount of the total dry mass), imposing a requirement on the mass for each subsystem.

  Thus, the system requirement would be e.g.

  REQ-SYS-01: The launcher shall be able to launch a 1000 kg payload into a LEO.

  Meanwhile, the subsystem requirements for the propulsion system would be e.g.

  REQ-SYS-PROP-01: The propulsion system shall be able to deliver 7800 m/s of ΔV.

  In turn, this would flow down to the fuel subsystem (FUEL) and, fuel tank subsystem (FT) as

  REQ-SYS-PROP-FUEL-01: The fuel shall have a specific impulse of at least 300 s.
  REQ-SYS-PROP-FUEL-02: The fuel shall have a density of at most 2100 kg/m^3 *of* ΔV.
  REQ-SYS-PROP-FUEL-03: The fuel system shall have a dry mass of at most 50 kg of ΔV.
  REQ-SYS-PROP-FT-01: The fuel tank shall have a volume of at least 780 m^3.
  REQ-SYS-PROP-FT-02: The fuel tank shall have a mass of at most 2000 kg.

  It goes without saying that in time, this results in more and more subsystem requirements, including requirements with regards to interfaces between subsystems. For example, it is evident that there will be a requirement regarding the cooling (to be done by the thermal control system) for the fuel system, based on the selected fuel.

It is obvious from the above that requirement flow down is an essential step in the design of a system. Even without having selected or designing the fuel yet, the designers of the fuel tank system can already start with designing their system. Although in a small-scale project, it is still possible to have sufficient communication within the design team that everything can be sized simultaneously, in a large project with hundreds or thousands of requirements, this is not possible anymore, hence providing the need for a good use of systems engineering tools, such as requirements.

At the same time, it goes without saying that assuming the initial values of what is reasonable requires careful consideration, to ensure that the system is feasible to be designed (e.g. not selecting an impossibly high specific impulse) and not overdesigned (e.g. assuming a very high fuel density).

7.3.2 Mission requirements

System requirements are not the highest-level requirements, either. Although not further treated in this project, system requirements formally originate from stakeholder requirements. A sample stakeholder requirement relating to the Saturn mission would be

REQ-STH-01: The spacecraft shall study the effect of Saturn's rings on the atmosphere of Saturn.

Such a stakeholder requirement comes from a party who affects or can be affected by the system (a stakeholder). In this case, the above stakeholder requirement could have come from a research institute that wanted to investigate Saturn. Stakeholders can be broad however; regulatory agencies, product users, citizens living close by are all examples of stakeholders.
In turn, stakeholder requirements originate from mission statements, which originate from needs and opportunities. However, these will not be discussed in more detail here.

7.3.3 Requirements on requirements

Formulating requirements is a careful task and they should thus adhere to a list of requirements themselves. Requirements on requirements include (as a minimum):

- *Each requirement shall use 'shall'*. Sentences including "shall" are, by convention, recognised as requirements. Use of words like "should", "must", "has to" are not universally accepted (also not in this project).
- *Each requirement shall use a unique identifier*. In the above, the identifiers are e.g., REQ-STH-01. They are used so that it's easier to refer to the requirement. There is no fixed way in which identifiers are structured and this is up to the developer of the product. Nonetheless, it's recommended to use a systematic approach for your identifiers.
- *Each requirement shall be unambiguous.* A requirement should not be able to be interpreted in any other way.
- *Each requirement shall include only one requirement.* E.g., a requirement "The drone shall have a range of at least 2500 km and an endurance of at least 50 hours", should be split up in two requirements.
- *Each requirement shall be verifiable*. This means that it should be possible to check whether the requirement is met. A requirement "The aircraft shall look cool" is not verifiable as it is subjective whether the aircraft looks cool.
- *The content of each requirement shall not include the actual reason for the requirement*. An explanation of why the requirement exists should be present in separate documentation (e.g., by writing it after the requirement in a bullet list (using text formatting to distinguish the requirement from the documentation), or as separate column in a table of requirements). However, a requirement such as "The HLDs shall provide an increase in maximum lift coefficient of 0.7 in take-off condition to decrease the stall speed to 75 m/s." does not comply with the above requirement, as the "to decrease the stall speed to 75 m/s". should be in separate documentation. Providing documentation for each requirement is quintessential, as it allows for traceability of the requirement.
- *The set of requirements shall be complete*. Meeting all requirements should mean that the design is able to execute its mission successfully. For example, missing a requirement on the cost of a spacecraft will likely result in a spacecraft that is prohibitively expensive. Not only system requirements need to be complete, but obviously also subsystem requirements ought also to be complete.
- *Each requirement shall not enter the design space*. If one is to design an ADCS (attitude determination & control system) for a new spacecraft, one should not start the design process by setting a requirement "The reaction wheels should be able to provide a torque of at least 0.1 Nm", as this would imply from the beginning that reaction wheels would be used (which would be a solution to the problem that is being designed for). Instead, one first needs to consider whether reaction wheels are the preferred attitude control system; different attitude control systems (e.g., thrusters) may impose different requirements.

  Assuming a priori that reaction wheels are to be used would limit the design freedom. In other words, although the set of requirements should be *complete* (see above bullet), it should also be as small as possible, to maintain freedom in design. Nonetheless, from the moment that it is decided that reaction wheels are chosen, it is important to set requirements on the design of the reaction wheels.

As with the functional analysis, the amount of detail you are required to go into for requirements within this project is to the same level of detail to which you are designing. For example, in this project you are never explicitly expected to deal with the noise of the aircraft, and thus you do not need to come up with requirements with regards to the noise of the aircraft as it would be considered outside the scope of the project. On the other hand, when designing the airfoil, you are expected to take into account the amount of fuel in the wing, so a requirement on the fuel tank volume is expected.

7.4 Design option generation

After setting up a complete set of requirements for the (sub)system that will be designed, the next step is to perform a trade-off between design options to select the option that will meet the requirements in the best way possible. Creativity ought to be shown in identifying possible solutions; this is the phase of the design where any breakthrough, out-of-the-box solutions can be identified. Nonetheless, the number of possible solutions should then be turned such that a reasonable number of design options remains that can be evaluated for the trade-off (3-6 design options usually suffice). These remaining design options should be complete, i.e., no obvious candidate should be omitted at this stage, but all remaining design options should at least be feasible, as it does not make sense to waste resources during the trade-off on a design that is obvious bound to fail.

7.5 Trade-off

Having determined a set of design options for which it is reasonable to assume that they will meet the requirements, it is time to perform a trade-off between them. A trade-off involves carefully evaluating the design options against a given set of trade-off criteria, to establish as objectively as possible, which design option is ‘best’. It cannot be stressed enough that a trade-off is not of the form “Option X is bad in A, and option Y is good in B, but since X is good in C, option X was chosen.”. A proper trade-off involves a systematic approach to arrive at a well-reasoned decision. It consists of four steps:

1. Determining trade-off criteria
2. Determining weights of trade-off criteria
3. Evaluating design options against trade-off criteria
4. Collecting trade-off results in a trade-off table and consequently selecting the best option.

7.5.1 Trade off-criteria

Trade-off criteria are the criteria against which the design options will be evaluated. The number of trade-off criteria should be limited, such that each criterion is of actual relevance, and no effort is wasted on aspects that are unimportant. Nonetheless, trade-off criteria should cover all important aspects of the design. All trade-off criteria should be unambiguously defined, and finally, no design aspect should be covered by multiple criteria.

7.5.2 Trade-off criteria weights

Determining the weights of trade-off criteria is often a more difficult task than coming up with the trade-off criteria. After all, this is an area where it is not possible to be completely objective; it is generally not possible to *exactly* determine the relevance of each criterion. Instead, it often relies on engineering judgement. As a result, there is no “correct” way of determining weights of trade-off criteria, however, it is important to clearly report on *why* the weights were chosen as they were, such that the reader (e.g., your manager, who was not present when the weights were selected) can easily retrace the reasoning and make adjustments if the reader deems necessary. When selecting the weights of trade-off criteria, two elements are worth bearing in mind: first, the relative importance of a criterion is strongly related to the relative importance of the requirement the criterion is based on. For example, if low weight is a critical requirement that is expected to have a large influence on the design, then a trade-off criterion related to mass will naturally have a higher weight. On the other hand, if noise regulations are so lenient that it is expected that any design option will meet the noise requirements, then it does not make sense to make noise an important criterion in the trade-off.

Secondly, the relative importance of a criterion is related to the relative difference that is expected to exist between design options. If it is expected that all design options will perform equally on a certain criterion, it is not logical to let have that criterion have a high weight.
Many ways exist to determine criteria weights, details of which can be found readily in literature. An example of such a method is one that finds its origin in Quality Function Deployment (QFD), which was introduced in the Japanese car industry to help translate customer needs into requirements. Here, it is used to translate customer needs into trade-off criteria. A snippet of a table used for such an analysis is shown in the table below. It deals with determining criteria weights for a trade-off between various system-wide car concepts (coupé, hatchback, etc.), for a newly designed automobile for driving within a city. The trade-off criteria to be used are the mass of the vehicle, the drag, the noise, production cost, and the leg space of the driver.

		Trade-off criteria				
Customer importance rating	Customer needs	Mass	Drag	Noise	Production cost	Leg space
5	Cheap	1	1		9	
4	Low fuel emissions	9	3			
3	Comfortable			3		9
1	Quiet for environment			9		
2	Looks cool		9			
2	Easy handling in turns	9				
3	High acceleration	9	1			
1	High top speed	1	9			
4	Safe	3				
Sum		99	47	18	45	27
Weights		42%	20%	7.6%	19%	11.4%

The idea of above table is straightforward. One comes up with the relevant customer needs and puts them in the second column. The trade-off criteria are noted in the columns next to it. Each customer need is given a customer importance rating, based on how important the need is to the customer. The customer needs are then related to the trade-off criteria; either a score of 0 (empty), 1, 3 or 9 is given, with higher scores given to stronger relationships. Then, for each criterion, the customer importance ratings are multiplied with the relation between the corresponding criteria and customer need; the sum of this is shown in the second-to-last row. These are then straightforwardly converted to a relative weight.
As examples, for a city car, cheapness is the most important need for a customer. This depends strongly on the production cost (so a relation of 9 is given), but also a bit on the mass and drag: higher mass and drag mean higher fuel consumption, so a lower mass and drag aid in reducing the operational cost of the vehicle (which the customer wants). Since the production cost probably dominates the cost for the customer, only a relationship strength of 1 is given. Whether the vehicle looks cool is probably of minor importance; other customer needs are simply more important. Nonetheless, it can be argued that there is a strong relation between looking cool and drag, as usually vehicles that look cooler have a lower drag. Safety is important for the customer, so it is given a customer importance rating of 4. However, based on the selected criteria, none of them relate strongly with safety. It only relates moderately with mass, as it can be argued that a lower mass would allow for more safety measures that could be implemented.
Even though the above method has its deficiencies (particularly regarding the transparency in customer importance ratings, and customer needs overlapping), it provides a significant amount of transparency in how the criteria weights are determined and allows one to shift thinking about how important criteria are to thinking about how important certain customer needs are, which is usually less abstract, obviously improving the quality of the result. Further information on QFD is readily found in literature.

7.5.3 Evaluation of design options

The design options can then be evaluated against the trade-off criteria. This is a technical task and thus will not be elaborated in much detail here. However, what should be noted is that trade-off criteria should be analysed with sufficient detail such that a meaningful conclusion can confidently be drawn. Furthermore, it is self-evident that the evaluation should be technically sound and correct.

Finally, the scoring should be sensible. Just like with the trade-off criteria weights, there is no fixed, correct way of determining a scoring method. Some suggested methods are:

- Qualitatively awarding a score. This is especially useful when there is no clear, objective way of defining a score, or it would be too much effort to find a single, clear performance measurement. For example, if risk is to be a criterion, then usually there is no single measurement of risk. Instead, one must rely on qualitative assessments of risk – high risk, medium risk, etc. This can then be converted into a numerical score, e.g., 5/5 is very low risk, 4/5 is low risk, etc., under the condition that "very low", "low", etc., are clearly defined. Care should be taken that not too many 'steps' are chosen in the scoring; using scores of 0/10, 1/10, 2/10, …, 9/10, 10/10 implies a resolution that may not exist.
- Scoring based on whether a design option meets the requirement: for example, a score of 0 may be awarded if an option does not meet the requirement and cannot be improved to meet it; 1 if an option does not meet the requirement, but it seems feasible that improvements can be made such that it does meet the requirement; 2 if an option meets the requirement; 3 if an option clearly exceeds the requirement. This has the benefit of not too greatly awarding design options that are unnecessarily overdesigned (contrary to linear scoring, see below). Design options that score a 0 somewhere may also be completely disqualified from the trade-off, meaning that it cannot win under any circumstance, even if it performs significantly better in all other aspects (bear in mind that in principle, a design that meets all requirements is unconditionally better than a design that meets all requirements minus one, regardless of relative performance differences).
- Linear scoring based on design options: best-performing design option gets 1/1, worst-performing design gets 0/1. This is especially useful if there is a single performance measurement for the criteria (e.g., for the $C_{L_{max}}$ of an airfoil, there will be a known, numerical value associated to this, contrary to the previously described risk criterion). Although this seems very intuitive, care must be taken that relative performance differences aren't exaggerated or given too little weight. For example, if all design options genuinely score approximately equally well, this method would still assign 0/1 to one concept and 1/1 to another concept, even though the relative performance difference may have been small. Similarly, if one design performs poorly, but another performs very poorly, then the one that performs poorly may still look good based on the score. Furthermore, doing better than the requirement is obviously beneficial, but above method arguably assigns overdesigned options higher scores than they should have been awarded.
- Linear scoring based on external datum points: rather than solely comparing design options to each other, one can select one or two datum points to compare the design options to (e.g., reference values based on existing designs, or the requirement). For example, if the requirement is that a certain system has an efficiency of 80%, and the design options have efficiencies of 85%, 84%, 84.5%, then one could compute a linear score on a scale of 0-1 by computing $(\eta_i - 80\%)/(85\% - 80\%)$, with η_i the efficiency of each design option. A disadvantage of this is that establishing proper datum points can be challenging.
- Non-linear scoring: non-linear functions may also be used to remove some of the drawbacks of the linear scoring methods. For example, logarithmic or exponential functions can be used to punish designs that do not meet the requirements more strongly, and that a marginal increase in performance will result in a smaller increase in score for a design option is already overdesigned than for a design option that just meets the

requirement[1]. A disadvantage is the added complexity of using mathematical functions, and the scoring method is obscured.

In any case, it should be clear what scoring method was used, and the chosen scoring method should be sensible.

7.5.4 Trade-off table

With all trade-off scores determined, a trade-off table summarizing the trade-off can be set up. This should provide a clear overview of the trade-off, and it should be easily identifiable from the trade-off table which design option is best. A sample trade-off table is shown in Figure 2: Example trade-off table. Colours are used to clearly indicate the relative performance of the concepts, and it is clear from the table which design option wins the trade-off.

	Pilot rating (feel of the aircraft)	Reponsiveness (Accelerations)	Manoeuvrability (Angular velocities)	Energy loss (Vertical climb)	RAMS + Risk	Cost	Total Score
Weight	20%	25%	20%	25%	5%	5%	100%
Concept 1	40.91%	68.00%	4.29%	62.50%	50.00%	44.22%	46%
Concept 2	39.39%	74.00%	0.00%	100.00%	48.33%	41.67%	56%
Concept 3	31.82%	72.00%	37.14%	62.50%	0.00%	100.00%	52%
Concept 4	36.36%	100.00%	50.00%	62.50%	0.00%	98.45%	63%
Concept 5	0.00%	45.00%	34.29%	37.50%	0.00%	98.45%	32%

Figure 2: Example of a trade-off table.

7.6 Compliance matrix

A compliance matrix is a table that shows all the requirements, with a check that the requirement is met, together with a reference to the section in the report where compliance with this requirement is demonstrated. As an example, a brief snippet of a compliance matrix for a race drone is shown in Table 1. A complete compliance matrix would cover all requirements. If a requirement is not met, the design should be revisited such that it does meet the requirement. If it is not known yet whether the requirement is met (as it would require a more detailed design), it should be indicated so, and it should be explained in the surrounding text at what stage of the design you expect to know whether that requirement is met.

Table 1: Snippet of a compliance matrix

Requirement	Satisfied?	Section
REQ-PER-1: The drone shall be able to fly for at least 10 minutes	Yes, 15 minutes	4.6
REQ-SW-3: The drone shall use onboard computation only	Yes	6.5
REQ-RAMS-2: The aggregate time required to replace all replaceable components of the drone shall not exceed 45 minutes	Not known yet	-

[1] For example, if there is a requirement that the mass should be less than 100 kg, then using an exponential function could mean that, if design A is 100 kg, design B is 95 kg, design C is 50 kg and design D is 45 kg, that design C and D have a smaller score difference than design A and B (an exponential scoring could be implemented by computing for each design option e^{m_i}, with m_i the mass of the design option, and then using a linear scoring method based on the resulting values).

7.7 Reproducibility & iterations

The previously mentioned systems engineering elements all aid in overseeing a complex project. Two aspects that will still be discussed are reproducibility and the use of iterations within the systems engineering universe.

7.7.1 Reproducibility

An aspect that is important in ensuring a successful outcome of a project is that all design activities are documented properly. This means, at a minimum, that all design activities are reproducible: a reader (who does have some technical knowledge, particularly if conventional methods are used) should be able to completely reproduce what was done and arrive at the same conclusions and design. This manifests itself in many ways. Some examples are:

- As mentioned previously, for every requirement, it is expected that it is documented where they originate from, e.g., by referring to the specific function it is related to (again, using identifiers here helps in cross-referencing). Moreover, for non-functional requirements, one is expected to document to which function it corresponds to, and why the selected numerical value was chosen. This aids later in the design process when iterations occur: consider the subsystem requirements for the launcher discussed in Section 7.3.1. Suppose one managed to select a fuel with a significantly higher specific impulse. If the requirements were clearly documented, it is then clear that the fuel tank volume requirement may thus be decreased in the next iterations.
- The above example of establishing trade-off criteria weights based on a variant of a QFD shows clearly where the trade-off criteria weights originate from. Although the result may still be imperfect, as the reader may disagree with some of the customer needs or the relations, but at least it is clear to the reader what approach was used and a fruitful discussion can be held about it to arrive at a consensus. Evidently, an argumentation of "Mass is considered very important, and thus was given a weight of 40%; cost is less important, so a weight of 20% was assigned" does not provide the same transparency.
- When reporting on calculations, it is important to not only include all the equations that are used, but also to include all relevant output that was obtained, as well as all input parameters that were used. The output and input data can for example be stored in simple tables.

Reproducibility helps in dealing with any possible design problems later in the design phase. If everything is properly documented, and there is a fault in element X, it may be more easily identified what the purpose of element X was in the first place, and whether perhaps there is another solution to it rather than just improving X. It should be noted that at the same time, design reports still should be concise. Thus, care should be taken in not reporting on things of secondary importance. In general, for this project, you may assume that the reader is knowledgeable about the topic you are working on but is not involved with your project on a day-to-day basis and thus reads the reports as a means of catching up on what you did and checking whether what you did is correct.

7.7.2 Iterations

Although the systems engineering elements above have been introduced in a purely sequential manner (first functional analysis, then requirements, then design options, then trade-off, then detailed design, then a compliance matrix), it cannot be understated that application of systems engineering is an iterative process. This was already partially discussed in Section 7.3.3, in the discussion of requirements not being allowed to enter the design space.

During the very first phase of the design, only few requirements may be imposed yet: after all, it is, strictly speaking, not clear yet at this phase of the design which subsystems will even be present; setting requirements on these subsystems would assume that the subsystem exist, limiting the design space. Compare a scenario in which one needs to design a system to reduce congestion in The Hague. It is then not clear from the beginning what kind of system this would be: it could be a new type of public transit, flying cars, simply adding more roads within the city, or any solution that has not been thought of before. Thus, one can only start with a limited number of requirements. When the system is designed on a system-wide level. The subsystems and their requirements can

then be identified and subsequently be designed. This will introduce new lower-level components, for which again requirements can be identified. Thus, the system is defined and decomposed from a top-to-bottom approach, ending up at the smallest elements of the system.
Although not part of the project, the physical versions of the smallest elements of the system are then verified that they meet their respective requirements. Then, it is verified of the sum of these elements, in the form of subsystems, meet the subsystem requirements (treating each subsystem separately, i.e., the system is not fully integrated yet; it is merely checked whether e.g., the ADCS on its own meets the requirements set on the ADCS). Obviously, during the design phase, tests to verify whether the subsystems met the requirements were already established. Finally, all subsystems are physically integrated, and the system can be validated[2]. Again, the tests that will be used to validate the system are already designed during the design of the system.
Thus, the design phase follows a top-down approach: first, the system is designed, then its subsystems, then the lower-level elements, etc. On the other hand, the integration and testing phase follows a bottom-up phase: one starts by treating the individual lower elements, and then integrates these at different levels. This is all schematically shown in the V-diagram shown in Figure 3 (Wasserman, 2014), a cornerstone of the systems engineering universe.

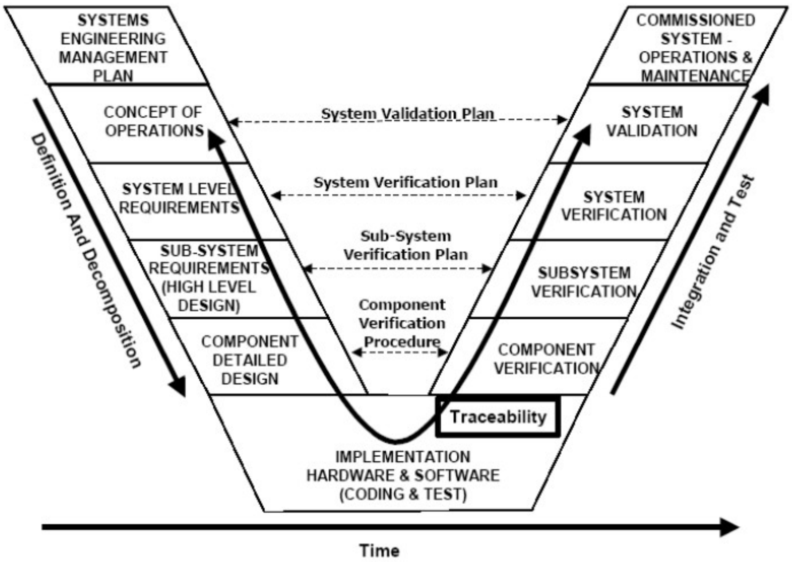


Figure 3: V-diagram.

Systems Engineering references:

INCOSE. (2000). *INCOSE Systems Engineering Handbook.* International Council on Systems Engineering.

Jackson. (1997). *Systems Engineering for Commercial Aircraft.* Ashgate Publishing Limited.

Wasserman, S. (2014, Maart 20). *Model-Based System Engineering - Beyond Spreadsheets*. Opgehaald van engineering.com: https://www.engineering.com/DesignSoftware/DesignSoftwareArticles/ArticleID/7352/Model-Based-System-Engineering--Beyond-Spreadsheets.aspx

[2] Note that checking whether a subsystem meets the subsystem requirements is called verification, but checking whether a system meets the system requirement is commonly called validation.