Department of Electronic & Electrical Engineering

BEng/MEng in EEE/EDS/EES/EME: 19.496/EM401 Individual Project

INTERIM REPORT

- A draft e-copy of the report (as pdf file and entitled "name_interim-report") must be submitted online via MyPlace by 12h00 on Friday 01 December 2023.
- The final e-copy must be submitted (and confirmed) by Friday 08 December at 12h00. (The draft submission from 01 December will be considered as the final submission after the deadline.) No submission will be permitted after 08 December.
- Formal written feedback will be provided after the project oral in January.

Student Name:	**Registration Number:**	**Supervisor:**
Daniel Stebbings	**202118874**	Louise Crockett
Project Title: Downlink System Design for the STRATHcube Satellite Mission		

Please read the following extract from the University of Strathclyde Regulations 5.4 and 5.6.

"Essays, assignments, dissertations, project reports and other forms of written material, submitted by either individual students or groups, form important components of your assessed work leading to credit awards. It is your responsibility to ensure that such material is your/your group's own work, and not that of others. Failure to comply with this standard of academic honesty will result in penalties being imposed through reduction in marks and possible disciplinary action."

Full details of the University's Academic Dishonesty Policy and definitions of Plagiarism and Collusion are given in the University Regulations and reproduced in all Undergraduate and Graduate School Handbooks.

Declaration

By submitting the work, I am confirming that that I have read and accepted the University policy and hereby declare that this work has not been submitted for any other degree/course at this University or any other institution and that, except where reference is made to the work of other authors, the material presented is original.

A key part of project conduct and reporting is the requirement to compare current progress to that anticipated in any previous reporting stage and then highlight where changes have occurred, the reasons for the changes and the actions taken in response to such changes. As part of the interim report, students are required to reflect upon the progress of the project to date, reflect upon on the degree to which this progress has met expectations and intimate the impact that such changes have had on the planned project objectives and deliverables. Limited to this current page. Comments should be simple and clear. Not duplicate content from main body of interim report.

A. Project Objectives

The primary objective remains the creation of the downlink communication system for STRATHcube. However, due to difficulty in sourcing the TOTEM SDR in time for the project, development will be targeted towards creating an engineering model using development boards. This was foreseen during planning, and introduces only minor changes.

Additionally,the objectives stated in the Statement of Intent were quite skewed towards RTL development. After further research, more time will be required for data handling aspects such as scheduling.

B. Project Progress:

My plans for development were also quite skewed towards RTL, as such I underestimated the time required to complete research into all aspects of the system. After finding a detailed MATLAB example for DVB-S2 implementation my time was spent analysing the resource requirements, identifying potential tools and software required, and analysing system performance requirements culminating in a detailed plan for the system implementation. Overall, I'm satisfied with the volume of work I've been able complete and believe it is as expected in the Statement of Intent, albeit with a different weighting on tasks than expected.

C. Project Deliverables:

As the project target has changed from the TOTEM SDR to development boards, the final implemented system will require further development before it is ready for flight hardware, as manufacturer development files could not be sourced. Again, this was foreseen during the planning of the project and is not an issue. As all further resources are either freely available, or already owned by the university, no further change of project scope is foreseen.

1. Project Context & Background

1.1. STRATHcube

STRATHcube is a student led satellite project at the University of Strathclyde that will be launched from the International Space Station with the aim of demonstrating the use of a Passive Bistatic Radar (PBR) for in-orbit detection of space debris. This project aims to create an engineering model of the downlink communication system for the satellite using commercially available development boards in compliance with relevant standards, a key step towards the preliminary design review with the European Space Agency (ESA) Fly Your Satellite program.

The satellite will include an Alén Space TOTEM Software Defined Radio (SDR) to handle both communication and PBR systems, which consists of an AMD Zynq 7020 System on Chip (SoC) connected to an Analog Devices AD9364 RF transceiver[1]. Due to the high cost of space qualified hardware, it will be modelled by a Digilent Zedboard connected to an FMCOMMS1 AD9361 development board. This has an identical Zynq SoC with less external memory and a similar RF transceiver.[2], [3]

A summer research project investigated the downlink configuration, selecting a frequency range of 430-440MHz, creating a link budget and selecting DVB-S2 (Digital Video Broadcasting - Satellite - Second Generation) as the modulation scheme.[4] This has extremely strong Forward Error Correction (FEC) capabilities and can adapt based on channel conditions to maximise spectral efficiency with Adaptive Coding and Modulation (ACM).[5]

As this project ties into a larger mission, it was important to identify clear goals and limit scope. The primary goal is to create a detailed system definition and implementation of the downlink communication system that maximises data throughput over the duration of the mission. The detailed definition of packet structures, uplink communication system and final implementation on flight hardware are outside the scope of this project. To ensure that the final solution is suitable, systems will be designed to comply with Consultative Committee for Space Data Systems (CCSDS) standards where possible.

1.2. DVB-S2

1.2.1. Overview

The DVB-S2 standard is modular with several optional blocks and data formats depending on the application.[6] This project will use a single generic packetised stream with a fixed User Packet Length (UPL), as it includes 8 bit Cyclic Redundancy Check (CRC-8) error detection. The data stream is sliced into Data Field Length (DFL) sized chunks and a header inserted to become a Baseband Frame (BBFRAME). The FEC systems require the BBFRAME to be padded to a specific length that depends on the current ACM settings. After two coding stages, this becomes a FECFRAME which goes through constellation mapping, pilot insertion and scrambling to become a Physical Layer Frame. It is finally filtered and modulated up to the carrier frequency.

DVB-S2 uses four modulation schemes from QPSK to 32APSK, coding rates from 1/4 to 9/10 and two FECFRAME lengths. These parameters, along with DFL and UPL, are controlled by the ACM Router to adapt to channel conditions and can be changed from one BBFRAME to the next. This means that the required size of the BBFRAME changes frequently. To minimise padding, a link layer protocol can be used called Generic Stream Encapsulation (GSE) that handles the fragmentation of packets based on the current optimal DFL[7]

1.2.2. Implementation

The Zynq SoC consists of a traditional Field Programmable Gate Array (FPGA), referred to as the Programmable Logic (PL) combined with two ARM processing cores, referred to as the Processing System (PS). One of the key challenges of this project is managing PL resource usage. The flight hardware must handle uplink and downlink communications, PBR processing and existing code included by the manufacturer. A target of no more than 50% usage on any one resource was implemented to ensure everything could fit. The second key challenge is reliability, verification, and adherence to standard. To ensure this, a MATLAB example DVB-S2 implementation using HDL Coder will be modified to minimise resource requirements and integrate with the rest of the system.[8] Packet handling will be managed on the PS, including ACM, scheduling, any GSE implementation and header configuration.

1.3. Adaptive Coding and Modulation Analysis

Due to the complexity of implementing ACM in the system, an investigation into the benefits over a fixed configuration that maximised availability was conducted. The satellite orbit was simulated for the full mission duration at its initial altitude of 400km using the MATLAB Satellite Communications Toolbox.[9] The elevation of the satellite during each pass over the Glasgow based ground station was binned into 5° increments Next, an existing link budget was used to calculate the Carrier to Noise Ratio (CNR) for each elevation bin and cross-referenced to the optimimum modulation and coding rate using the DVB-S2 standard[6] with a 10dB link margin and Bit Error Rate (BER) of 10^{-7}. The maximum possible throughput was calculated for each elevation bin and multiplied by the time spent to find the total throughput for each strategy. The results are shown in Figure 1 and show a significant increase in throughput, proving that the benefits of implementing ACM are worth the added complexity.

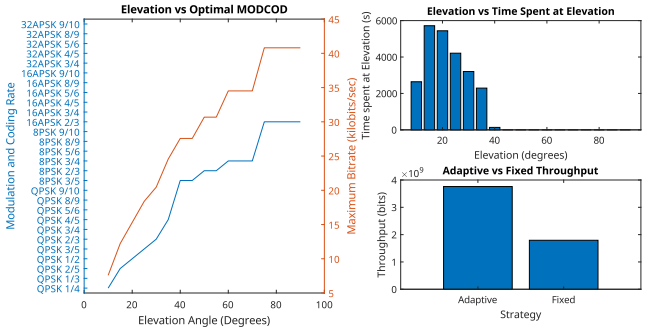


Figure 1: Left - Elevation vs modulation and coding rate (Blue), modulation and Coding rate to throughput (red).
Top Right - Time spent in each elevation bin.
Bottom Right - Throughput vs strategy.

1.4. Packet Handling

1.4.1. Interfacing and Formatting

When designing the system, it was important to balance implementation complexity with future integration of the final system as the onboard data handling on the satellite is an area that has not been fully developed. For this purpose it was decided to design all systems to be compatible with XML Telemetric and Command Exchange (XTCE) formatted packets. XTCE was defined by the CCSDS in order to unify the interpretation of data between space systems and is widely supported.[10] This will allow generic packets to be defined for testing without restricting future development.

Project packet definitions will be managed with a NASA tool called Core Flight System (CFS) Command and Data Dictionary (CCDD).[11] This can be used to generate C headers for parsing, another option being a Python library called “Space Packet Parser”.[12] The Python approach may be simpler to implement, but the C headers will be more efficient, and easier to integrate with the rest of the system.

The TOTEM SDR is connected to the rest of the satellite through two bus interfaces: Controller Area Network (CAN) and Inter-Integrated Circuit (I2C).[1] There is also an Ethernet interface for ground testing and Linux Inter-Process Communication (IPC) for the PBR system running on the PS. The system will be designed to accept packets from all of these sources in parallel.

1.4.2. ACM Routing and PL Interface

A previous dissertation project investigated the scheduling and compression requirements of the downlink system.[13] This identified a need to prioritise housekeeping packets over PBR data. The research also selected the POCKET lossless compression algorithm and created a Python implementation in order to reduce the data footprint of packets.

The DVB-S2 implementation guidelines define methods for controlling quality of service for several types of input stream, including transmission priority and changing the ACM policy.[14] This could allow the use of a lower speed, low BER target for housekeeping data and a lower reliability target but higher speed for PBR data to increase total throughput. The flight implementation of the ACM Router will utilise information received from uplinked data, which must be synthesised for this project based on expected channel conditions.

GSE may be used to handle the varying DFL due to ACM, however the standard defines several features required for Internet Protocol that may not be necessary for this application.[7] To reduce complexity, the system may implement only a subset of the standard.

The ACM Router will interface with the PL using “libiio”, a library for interfacing with devices that use the Linux Industrial Input Output subsystem.[15] Created by Analog Devices, this library is well documented and simplifies the creation of drivers.

1.5. System Architecture

The resulting system will have a structure as shown in Figure 2, with three main blocks on the PS, most of which connected through buffers.

The first is the system interface, which collates packets from each external interface buffer and forwards them for parsing.

The Packet Parser will read each packet to identify its priority level, placing them accordingly in buffered queues.

The ACM Router will select packets based on their priority according to its scheduling policy. It will also choose ACM settings based on the current reported channel conditions and encapsulate packets accordingly using GSE. The packets and ACM command information will be passed to the relevant registers using the PL Interface.

This data will go through modulation on the PL and be converted into In-phase and Quadrature (IQ) data for the transceiver. Control of the physical device will be through MATLAB HDL Coder blocks available from the Analog Devices that interface with the transceiver over a Low Voltage Digital Signalling (LVDS) link.[16] The resulting RF signal output can then be connected to an SDR for analysis.

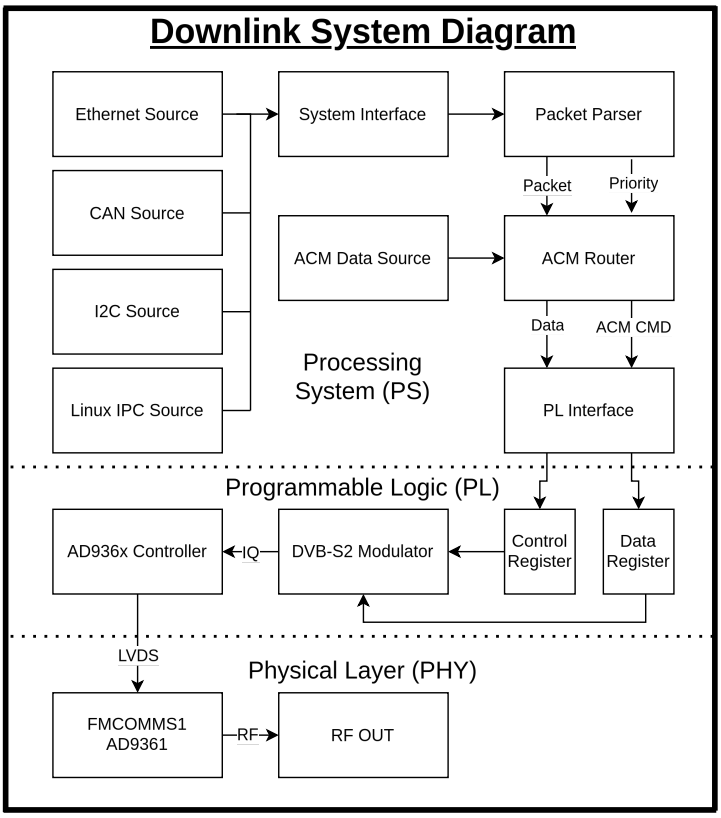
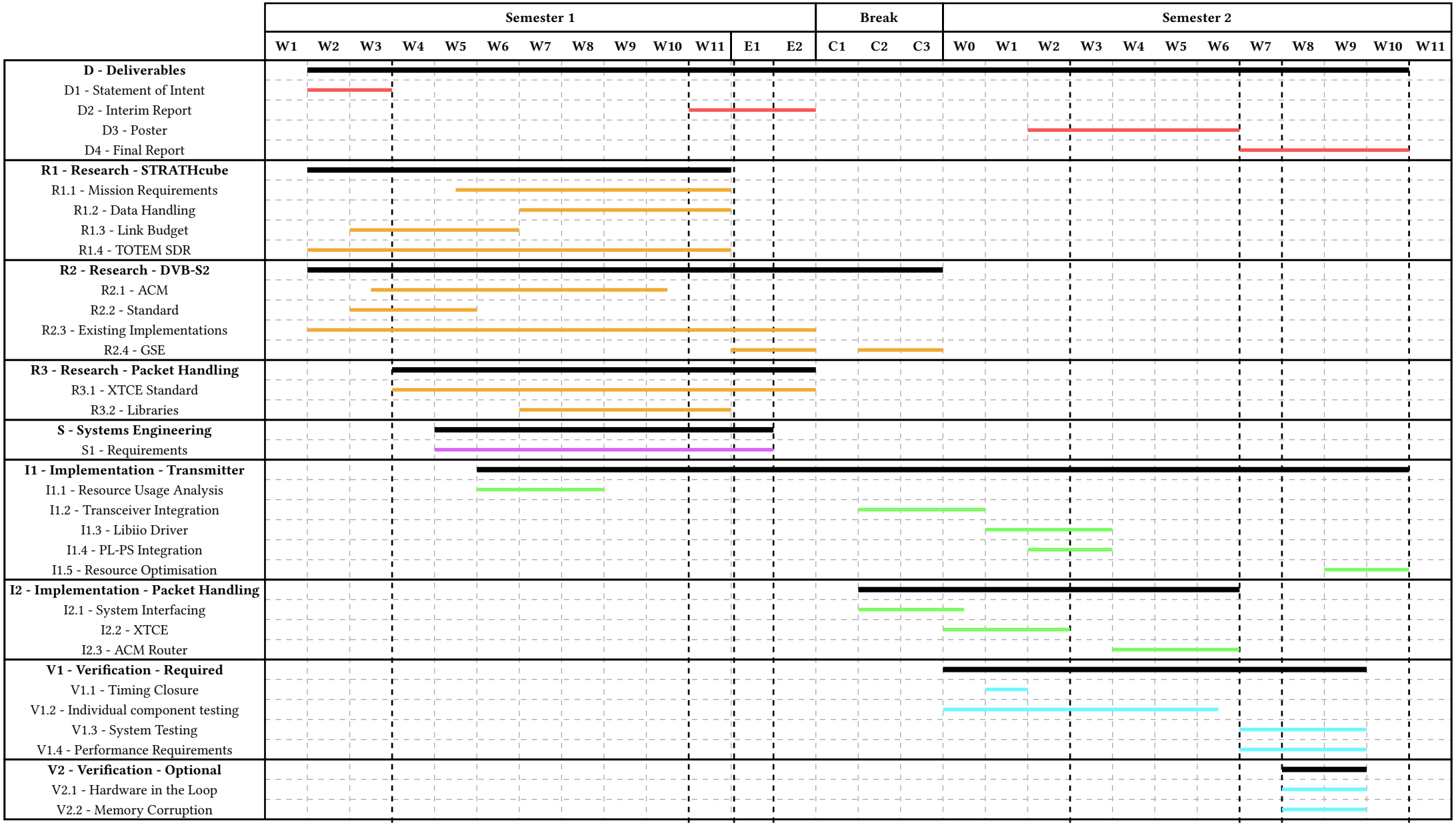


Figure 2: System Diagram

2. Project Timeline



3. Project Review

3.1. Project Plan

The first semester has been devoted to the research and planning of the project. The outcome of which has been a detailed analysis of each sub-block required for the final system and the identification of open source resources and relevant standards to ensure success.

There are three key milestones outlined in the project timeline. The first being the system architecture, which has been defined in Section 1.5. The second is the creation of a "System Prototype" which will demonstrate the minimum features necessary to send a packet through the system. This is targeted for Week 3 of Semester 2.

There are two key decision points, marked "DP1" and "DP2". DP1 has been reached, and marks the decision to target development hardware, as the TOTEM SDR could not be procured in time for the project. DP2 will be reached in Week 8 of Semester 2 and will mark the decision to focus on end to end testing of the system versus continuing feature development.

To ensure time is left for the final report, a design freeze will be implemented during week 9 of semester 2. This means that no major features or changes will be implemented from that point onwards to ensure that the system is finalised and tested in advance of the final report deadline.

3.2. Technical Risks

Risks	Mitigation	Likelihood	Severity	L*S
Catastrophic Dataloss	1. Use of cloud storage 2. Use of Git source control 3. Regular snapshots of work	1	5	5
Severe Illness	1. Creation of draft submissions on Myplace to avoid missing deadlines 2. Report personal circumstances on Pegasus 3. Discussion of reduction of scope with supervisor	1	4	4
Breakage of key equipment	1. Only work in designated areas 2. Use of equipment as designed, with care given for fragile connectors 3. Identification of alternatives available in the department	2	3	6
"Feature Creep"	1. Creation of requirements document 2. Regular review of work with supervisor 3. Scope freeze decided at Interim report, with options for further reduction identified in project plan	3	2	6
Difficulty in system integration	1. Regular project meetings with supervisor 2. Adherence to best practices in RTL design with continous verification and testing 3. Use of version control to track changes 4. Use of a modular design to allow verification of components 5. Use of resources from manufacturers to ensure compatibility	4	2	8
Extended learning curves of new languages and tools	1. Use of online resources 2. Use of existing knowledge of similar tools to reduce learning curve 3. Use of manufacturer resources	3	2	6
System not fit for purpose	1. Design with requirements in mind 2. Use of open-source examples to ensure compatibility and reliability 3. Use of a modular design to allow for changes and unit testing	2	3	6

3.3. Sustainability

STRATHcube's primary mission relates to a pressing issue in the space industry, space debris. The PBR will demonstrate a technology that could fill a critical resolution gap in debris detection that ground based radars cannot. The importance of space situational awareness cannot be understated, as the amount of operational satellite increases there will be a higher risk of impact with debris too small to be detected with current methods.

The more data that can be downlinked during the mission, the more opportunities there are to observe passing debris and verify the operation of the PBR. As such, the successful completion of this project will directly contribute to the success of the mission and an important aspect of space sustainability.

3.4. Ethical Considerations and EDI

The main ethical considerations for this project lie in compliance to relevant telecommunications regulations from the International Telecommunication Union (ITU) and Office of Communications (OFCOM). During development, all transmissions will be conducted using certified equipment and through cables to ensure that no harmful interference is caused.

The resulting source code will be made available to the public under an open-source license to ensure that the project is transparent and accessible to all. This will also allow for the project to be continued for further use in STRATHcube or as a resource for other projects.

Aditionally, space is a resource that underpins modern society and it is important to ensure that it is used responsibly. The primary and secondary missions of STRATHcube will positively contribute to space sustainability and as such will have a positive impact that will support society as a whole.

4. References

[1] "TOTEM Motherboard Datasheet." Alén Space, Apr. 01, 2024.

[2] "ZedBoard Product Brief." Avnet.

[3] "AD-FMCOMMS1-EBZ User Guide [Analog Devices Wiki]." Accessed: Dec. 11, 2024. [Online]. Available: https://wiki.analog.com/resources/eval/user-guides/ad-fmcomms1-ebz

[4] P. Rasamanickam, "Downlink Link Budget Table." 2024.

[5] M. Eroz, F.-W. Sun, and L.-N. Lee, "DVB-S2 low density parity check codes with near Shannon limit performance," *International Journal of Satellite Communications and Networking*, vol. 22, no. 3, pp. 269–279, 2004, doi: 10.1002/sat.787.

[6] "EN 302 307-1 - V1.4.1 - Digital Video Broadcasting (DVB); Second generation framing structure, channel coding and modulation systems for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 1: DVB-S2," no. EN302307-1.

[7] "TS 102 606-1 - V1.2.1 - Digital Video Broadcasting (DVB); Generic Stream Encapsulation (GSE); Part 1: Protocol."

[8] "DVB-S2 HDL Transmitter." Accessed: Dec. 11, 2024. [Online]. Available: https://uk.mathworks.com/help/satcom/ug/dvbs2-hdl-transmitter.html

[9] "Satellite Communications Toolbox." [Online]. Available: https://uk.mathworks.com/products/satellite-communications.html

[10] "XML Telemetric and Command Exchange." Feb. 2020.

[11] "nasa/CCDD." Accessed: Dec. 10, 2024. [Online]. Available: https://github.com/nasa/CCDD

[12] Gavin Medley, Michael Chambliss, and Greg Lucas, "space_packet_parser." Accessed: Dec. 03, 2024. [Online]. Available: https://github.com/medley56/space_packet_parser

[13] N. O'Neill-Berest, "Simulating STRATHcube's Data Handling and Transmission Systems," Apr. 2023.

[14] "Digital Video Broadcasting (DVB); Implementation guidelines for the second generation system for Broadcasting, Interactive Services, News Gathering and other broadband satellite applications; Part 1: DVB-S2." Accessed: Dec. 03, 2024. [Online]. Available: https://www.etsi.org/deliver/etsi_tr/102300_102399/10237601/01.02.01_60/tr_10237601v010201p.pdf

[15] "analogdevicesinc/libiio." [Online]. Available: https://github.com/analogdevicesinc/libiio

[16] "AD936x Transmitter." [Online]. Available: https://uk.mathworks.com/help/soc/ref/ad936xtransmitter.html