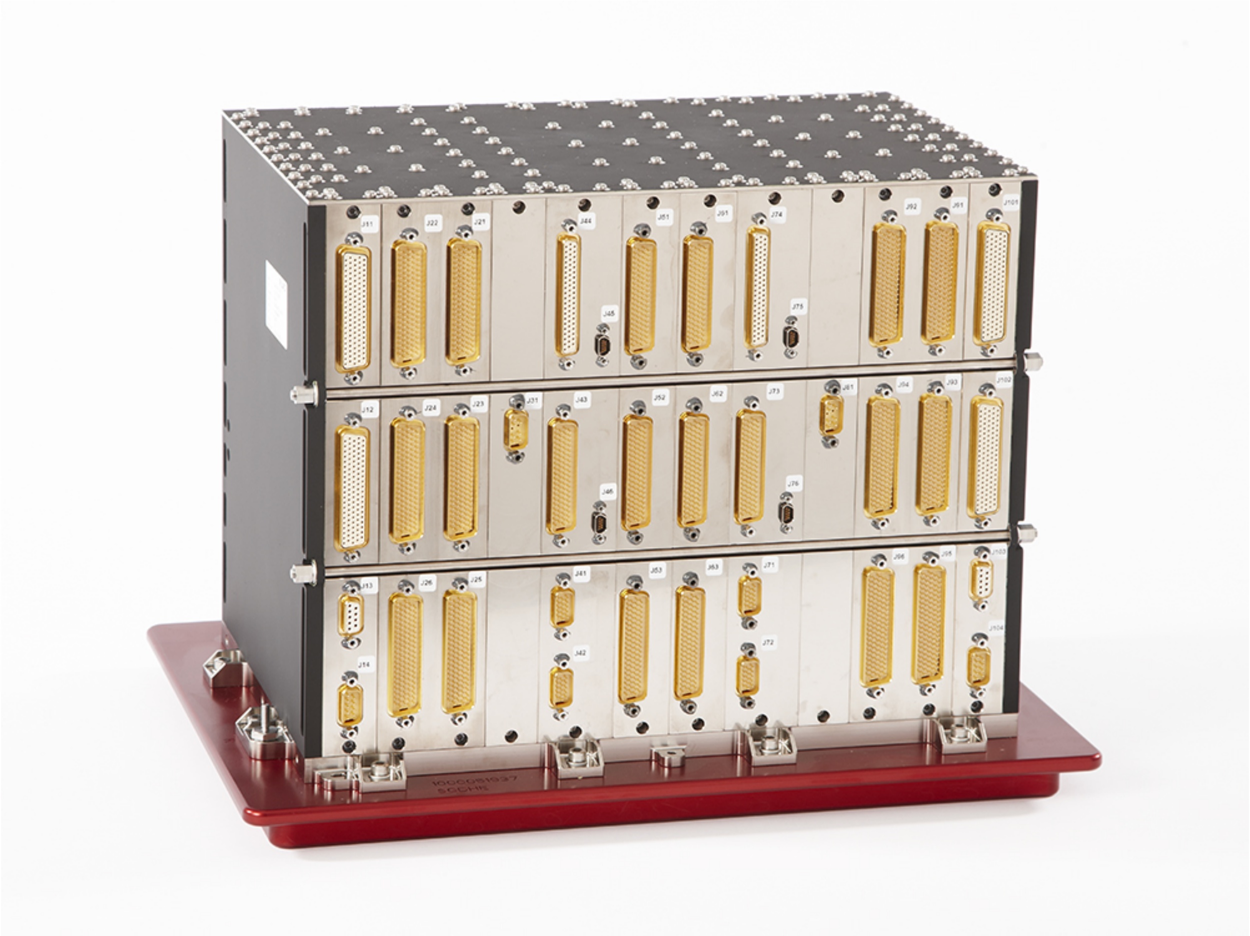

A Configurable Interface Unit for Telemetry Monitoring in Satellites

Master's thesis in Embedded Electronic System Design

FREDRIK HAGSLÄTT
FREDRIK JOHANSSON

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2018

Master's thesis 2018

A Configurable Interface Unit for Telemetry Monitoring in Satellites

FREDRIK HAGSLÄTT
FREDRIK JOHANSSON

Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
Gothenburg, Sweden 2018

A Configurable Interface Unit for Telemetry Monitoring in Satellites
FREDRIK HAGSLÄTT
FREDRIK JOHANSSON

Supervisor: Vilhelm Geijer, RUAG Space AB
Supervisor: Andreas Wramdemark, RUAG Space AB
Supervisor: Lena Peterson, Department of Computer Science and Engineering
Examiner: Per Larsson-Edefors, Department of Computer Science and Engineering

Master's Thesis 2018
Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A remote interface unit designed and manufactured by RUAG Space AB.

Typeset in LaTeX
Printed by Teknologtryck
Gothenburg, Sweden 2018

A Configurable Interface Unit for Telemetry Monitoring in Satellites
FREDRIK HAGSLÄTT
FREDRIK JOHANSSON
Department of Computer Science and Engineering
Chalmers University of Technology
University of Gothenburg

Abstract

This report presents a concept for a flexible remote interface unit for telemetry monitoring in satellites. The concept consists of a flexible hardware interface that can be configured by a microcontroller into reading several different types of sensors. An important aspect in the introduced flexibility is the presented method for filtering signals digitally, eliminating the need for slow analog low-pass filters. The feasibility of the concept is verified and its expected performance is evaluated. The performance evaluation is based on acquisition times that are affected by the hardware's configuration time, the sensor's settling times and the time needed for digital filtering. The report also summarizes the number and types of sensors on several finished satellites to evaluate how many the proposed concept can cover. Digital filtering together with the evaluation of performance and coverage shows that this concept can be further developed into a flexible design that can cover future projects without the need for major hardware changes, thus reducing development costs.

Keywords: Flexible, Configurable, RIU, Telemetry, Satellite, Microcontroller, Digital Filtering

Acknowledgements

First of all we would like to thank our supervisor at RUAG Space AB, Vilhelm Geijer, for his outstanding guidance throughout the whole project. His knowledge in the field of space engineering has boosted the progress of this project significantly. We also want to extend our gratitude to Andreas Wramdemark whose knowledge in the remote interface units designed at the company has greatly aided our understanding of the system.

The advise given by our supervisor at Chalmers, Lena Peterson, and our examiner Per Larsson-Edefors has been a great help when shaping the project and writing this report. We also want to express our appreciation to Olle Martinsson, Fredrik Öhnell, Patrik Sandin and Andreas Karlsson from RUAG Space AB for their help and guidance during the project.

Fredrik Hagslätt and Fredrik Johansson, Gothenburg, June 2018

Abbreviations

ADC	Analog-to-Digital Converter
ASIC	Application Specific Integrated Circuit
ASM	Analog Signal Monitoring
BDM	Bi-level Discrete Monitor
BSM	Bi-level Switch Monitor
ECSS	European Cooperation for Space Standardization
ESA	European Space Agency
FRIU	Flexible Remote Interface Unit
HPC	High Power Command
I/F	Interface
I/O	Input/Output
LSB	Least Significant Bit
NRE	Non-Recurring Engineering
OBC	On-Board Computer
OBDH	On-Board Data Handling
Op-amp	Operational Amplifier
PCB	Printed Circuit-Board
PnP	Plug and Play
RC filter	Resistor-Capacitor filter
RIU	Remote Interface Unit
SAS	Sun Acquisition Sensor
SPA	Space Plug and Play Architecture
TSM	Temperature Sensor Monitoring

Contents

1

Introduction

Products developed by the space industry have traditionally been custom made from the start. As in most technical fields, resources can be saved by using off-the-shelf products instead of designing ad hoc solutions [1]. This project is a step in that direction which intended to simplify a part of the design process for satellites.

Today satellites have a wide range of use, from communication and broadcasting to gathering information about the Earth for different missions. The information that is gathered is used for various research purposes, but can also serve other areas like collecting information concerning the weather and environment [2]. The European Space Agency (ESA) is planning several Earth-observation missions in the coming years. These missions will for example gather information about the winds, the amount of carbon in the forests and provide better weather-forecast possibilities [3, 4].

The satellites for different missions may differ in terms of what orbit path they are to use, the size of the satellite and the installed equipment. However, a challenge all satellites are faced with is that space is not a hospitable place and there are a number of factors that can cause malfunctions. Some examples are space debris and the constant exposure to radiation [5–7]. Because of such risks, it is important to monitor the condition and functionality of the satellites to ensure that they are operating correctly. For this purpose a number of sensors are installed on each satellite.

For the satellite to accomplish its mission, there is also other equipment installed that is to be turned on and off when needed. The task of reading all sensors and routing the control signals is handled by a remote interface unit (RIU). This project was carried out at RUAG Space AB, that designs and manufactures RIUs [8, ch. 15]. Since satellites have different tasks, sensors and equipment, the RIU needs to be tailored to each new specification, resulting in a large amount of non-recurring engineering (NRE) for each unit. Given the tight requirements on power, weight and redundancy for space applications tailoring is a costly process [8, ch. 3]. A possible solution to address these problems is to use a flexible design that covers the most standard cases of sensors and equipment. A design of the proposed type could lead to a significant reduction in NRE costs.

1.1 Aim

The main goal of this project was to increase the flexibility of the RIUs developed at RUAG Space AB today and, as a result, decrease the NRE cost. The idea is that instead of having to design new RIU hardware for each system specification, the engineer should be able to reconfigure an existing RIU unit. Such a unit, from now on called a flexible RIU (FRIU), was the focus of this project.

To achieve the main goal, a conceptual overview of a new hardware design was defined, where the intention was that it should replace parts of the current RIU designs. The idea for the hardware design was that it should have some configurable parts that then would be controlled by a microcontroller. The goal for this conceptual design was that it should act as a base for subsequent evaluations regarding the improvement on the RIU design. Where the aim for the evaluations was to give answer on aspects such as to what extent that parts of the current RIU designs could be replaced, if the concept is realistic to implement and how well it meets requirements such as timing. An additional goal for the design was that it should be a base for further investigation, either during the project if time allows or after the project's end.

1.2 Problem Description

Satellites today show a wide variety in size, application and design. Consequently there are many different combinations of systems to be controlled by the RIU. These systems can be divided into three main categories: sensors, high-power commands (HPCs) and propulsion. Sensors often outputs a voltage that is to be handled with a designated circuit (from now called a conditioning block) and converted to a digital signal via an analog-to-digital converter (ADC). HPCs are load-driving commands used for switching relays or similar loads. These commands are short pulses of varying voltage, pulse width and maximum output current. Propulsion systems send certain voltages to the equipment that move the satellite physically. The different combinations of these systems result in many different RIU designs that each requires custom development.

One type of NRE that often occurs in the RIU's development process is designing several printed circuit boards (PCB) with specific conditioning blocks designed for each kind of different input or output, limiting each input/output (I/O) pin to one specific task. A possible way to reduce this type of NRE would be to reuse old designs in newer RIUs. Because of the variation in requirements for the different RIUs, this would require a hardware that is flexible and can adapt to different quantities and combinations of inputs and outputs.

1.3 Limitations

Given the RIU designs that were investigated in the start of this project, it was a possibility that the FRIU would include mixed-signal application specific integrated circuits (ASIC). Designing and evaluating ASICs is beyond what can be handled in a time period as short as this project. If the design would have an ASIC that only handled digital signals, its functionality could have been tested on a field programmable gate array (FPGA), but this project lacked resources to evaluate a mixed signal ASIC design. Therefore it was decided in the start of the project that if parts of the design were implemented on a mixed-signal ASIC, they would not be designed in detail. Their architecture would instead be described in block schematics and if necessary, circuit diagrams, and the detailed designing would be focused more on the parts of the system that can be evaluated in a valuable way.

A significant part of the RIU is to handle control signals for the propulsion system of satellites. However this part requires extensive and well-defined hardware and is subject to large variations among the different designs. Hence it was decided to limit the project to the interfaces concerning sensors and HPCs, excluding the propulsion interfaces from the design.

A general challenge in this field is the effects that the space radiation environment has on electronics [6]. Accumulated damage from radiation can limit the systems endurance and individual high-energy protons can disrupt operations. Another important factor in space applications is power efficiency since the solar panels on a satellite generate limited power. All residual power will be converted into heat which might be problematic to dispose of, this is another reason why a low power consumption is an important factor. These three aspects were accounted for to a certain extent but they were not our main focus. The goals were more focused on the actual functionality and we therefore limited our scope, leaving the adaptation for space to a possible future project.

1.4 Report Outline

This project started with an initial study period where we researched all the systems that the FRIU should handle and how RUAG Space AB implements them in today's solution. The information that was gained through this period is presented in chapter 2 where we present the RIU, its most important tasks and the challenges in making it more flexible. Chapter 3 summarizes similar projects that have been carried out before this one, and compares them to our problem.

After gaining knowledge about the RIU and looking for solutions in related work, we defined a concept on which the rest of the project is built on. This concept is presented in chapter 4 along with explanations of how we verified its feasibility and evaluated its performance. During the phase where we verified its feasibility, we made several necessary additions to the design to ensure that all the systems could maintain their functionality. Therefore in chapter 5 where the results of the feasibility verification and performance evaluations are presented, we also present a more detailed version of the concept which includes the added changes. Chapter 6 discusses the implications of the results presented in chapter 5, and the project is concluded in chapter 7.

2
Remote Interface Unit

This chapter presents what a RIU is, technical information about its most significant parts, and some of the challenges in making it flexible.

Every satellite uses some type of on-board computer (OBC) that controls the implemented systems and maintains the desired status of the satellite, among other tasks. The systems and sensors that an OBC has to communicate with mostly use analog signals. To enable this communication there needs to be an interface between the OBC and the equipment. Given that there are hundreds of interfaces needed for each satellite, they are implemented in a separate unit, the RIU, that interprets the information and transmits it to the OBC over some type of communication bus.

The OBC interacts with the RIU by sending a command that is to be carried out or by requesting data from a specific sensor or system. Examples of commands or requests are acquisition of sensor statuses, to switch on systems and relays or to control the propulsion of the satellite. If the RIU is to interact with all the sensors and systems, it needs hardware adapted to each one.

Fig. 2.1 shows an overview of the RIU's different functional parts. The Control FPGA and the data buses handle the communication system of the RIU and are described in section 2.1. The telemetry module, which handles the physical interfaces for the sensors, is described in section 2.2 and the HPC module is explained in section 2.3. The propulsion module is not in the scope of this project but is still briefly discussed in section 2.4.

The figures shown in this section are designed to show function, not exact implementation. Some systems will for example need low-pass filters to reduce noise, or redundancy stages in the final implementation.

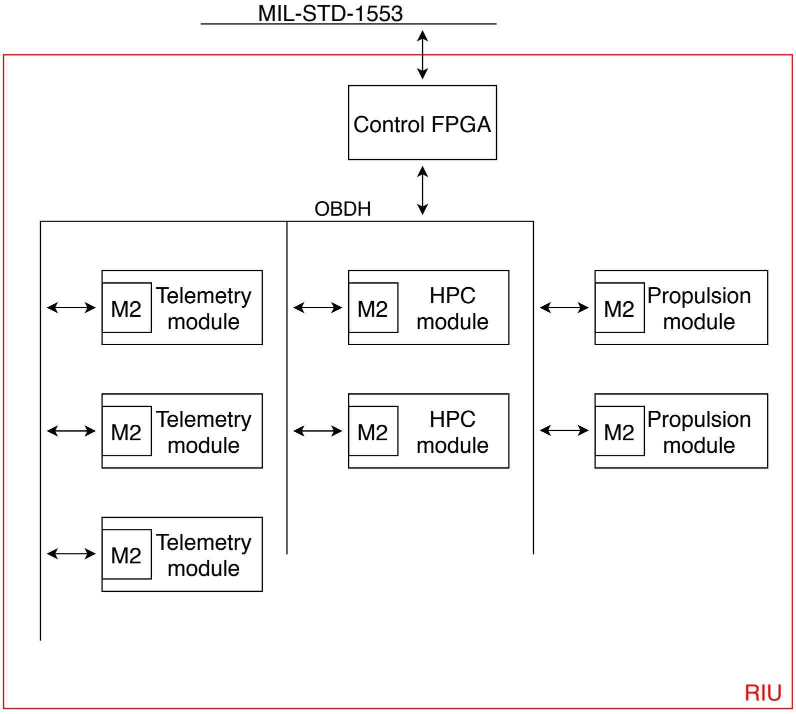

Figure 2.1: Functional block diagram for an RIU. The control FPGA gets orders from the OBC via the MIL-STD-1553 data bus and sends them to the different parts of the RIU via an OBDH bus. The different modules handle the circuits to all the inputs and outputs, performs the issued orders and sends the information back to the FPGA over the OBDH bus.

2.1 The RIU's Communication System

The RIUs designed at RUAG Space AB today usually have an FPGA as a central communication device. An RIU receives the commands from the OBC using a MIL-STD-1553 data bus and communicates with the rest of the RIU with an on-board data handling (OBDH) data bus. MIL-STD-1553 is usually used by the OBC, meaning that the RIUs often are tied to using this type of bus. The OBDH bus on the other hand is used solely to enable communication with the (by RUAG Space AB) commonly used ASIC called M2.

The M2 ASIC can read a number of digital and analog signals and send digital control signals to equipment. These features combined with the fact that it is radiation hardened makes it well adjusted for data acquisition systems in space applications. The downside is that the rest of the system has to be designed around the M2 ASIC which can limit the possibility for designing more flexible systems.

2.2 Telemetry Monitoring

An RIU frequently receives signals to acquire information about the condition of the satellite. There are several systems that generate voltage levels, both single ended and differential. There are also sensors that use other quantities, such as resistance or current as the measured parameter. These signals are in today's solution transmitted through a circuit that converts them to voltages before they are read. These kind of conversions are typically performed for sensors such as thermistors and sun sensors.

2.2.1 Analog Signal Monitor Interface

The analog signal monitor (ASM) interfaces are specifically designed to read voltages within certain ranges with as good accuracy as possible. The voltage is read using an ADC, sometimes after being amplified or attenuated by an operational amplifier (Op-amp), like in Fig. 2.2, to fit to the voltage range of the ADC. These voltage ranges are typically included in one of the following intervals: [0 V;2.5 V], [0 V;5 V], [−5 V;5 V], [−6 V;6 V], [−10 V;10 V]. Since the voltage range varies, different op-amp gains are needed to achieve the best possible accuracy. The method for programmable op-amps that Catunda et al. [9] propose could possibly introduce some extra flexibility in the RIU. There is also an option to route the signals to different op-amps depending on the chosen range.

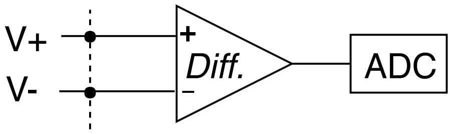

Figure 2.2: General circuit for analog voltage acquisition. The voltage difference of the two input pins is amplified or attenuated by an op-amp to fit the range of the ADC before it is measured.

2.2.2 Temperature Sensor Monitor Interface

The measurement of the temperature sensor monitor (TSM) interfaces is performed by calculating the resistance of a thermistor that is externally connected to the RIU. This is done with a circuit similar to the one presented in Fig. 2.3. To measure this resistance the thermistor is connected in series with a reference resistance to form a voltage divider. A set reference voltage is applied to the circuit and the ADC is connected to read the voltage in the middle. Since the reference voltage and reference resistance are known, the voltage read by the ADC can be used to calculate the resistance of the thermistor.

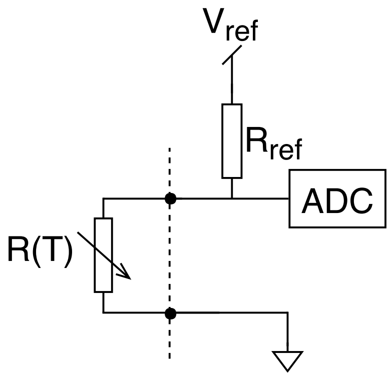

Figure 2.3: General circuit for a temperature sensor. The resistor R_{ref} and the thermistor $R(T)$ form a voltage divider. The voltage measured by the ADC along with the voltage V_{ref} is used to calculate the resistance $R(T)$, which is then translated to the corresponding temperature.

To achieve accurate measurements, the reference resistance is chosen based on the given resistance range for the sensor. Furthermore, the reference resistance is chosen based on which section of the temperature range that requires the highest accuracy. The section with the highest accuracy requirements is most often in the middle of the temperature range, according to specifications for previous RIUs designed by RUAG Space AB.

2.2.3 Sun Acquisition Sensor Interface

The sun acquisition sensors (SAS) are photo diodes that under different exposures of light yield a corresponding current. To measure this current, it is lead through a shunt resistor to create a voltage drop for the ADC to read. The general circuit used for this conversion is presented in Fig. 2.4. The current generated by a photo diode is affected by its load, and it is important that the shunt resistor has a low enough resistance for the photo diode to function as expected. At the same time it needs a resistance high enough to create a measurable voltage drop. The used resistance varies depending on the exact type of sensor, but is usually in the order of one Ohm.

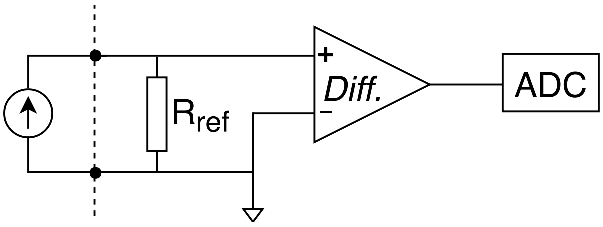

Figure 2.4: General circuit for sun sensor acquisition. The sensor generates a current that is lead through the resistor R_{ref}, creating a voltage drop. The voltage drop is then amplified with an op-amp and measured with an ADC.

2.2.4 Bi-level Switch Monitor Interface

The Bi-level switch monitor (BSM) interfaces check if a relay or an optocoupler is open or closed, using the circuit presented in Fig. 2.5. If the relay is closed, the ADC will be connected to ground and the voltage will be zero. If the relay is open, there will be no voltage drop over the resistor and the ADC will read the reference voltage. The optocoupler on the other hand will not be completely digital like the relay, but will instead act like a resistor with a varying resistance. In this case the circuit functions as a voltage divider that measures the resistance like in the TSM measurements in section 2.2. The measured value is then processed in the software, where a certain threshold resistance will define the optocoupler as opened or closed.

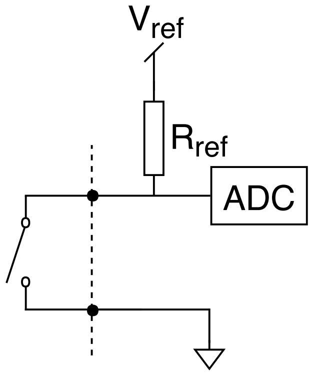

Figure 2.5: General circuit relay status acquisition. When the switch is closed, the ADC will be grounded and read 0 V. When the switch is open, the ADC will read V_{ref}.

2.2.5 Bi-level Discrete Monitor Interface

The Bi-level discrete monitor (BDM) interfaces are for equipment that use a high or low voltage where the status level is to be determined, but the exact voltage is irrelevant as long as it is known if it exceeds a certain threshold level or not. Hence these signals can be classified as digital signals, but are handled as analog voltage signals by the hardware. These signals are often used to read status signals from various systems.

2.2.6 Serial Digital Interfaces

The serial digital interfaces are used to serially send and receive information to and from equipment on the satellite. This information is typically transmitted by using an 16-bit bi-directional serial digital (BSD) interface or universal asynchronous receiver/transmitter (UART). BSD uses 5 signals: GATE_WRITE, GATE_READ, DATA_CLK_OUT, DATA_OUT and DATA_IN to transfer data where UART instead uses two signals, TX and RX. Both BSD and UART are transmitted over a standard balanced digital link (SBDL) which means that two circuits are used for each signal, where a positive differential voltage represents logic "0" and a negative represents logic "1" for the signal.

2.3 High Power Commands

Some equipment is turned on, off or is controlled by pulse commands produced by the RIU. These commands are short voltage-pulses of varying amplitude, pulse width and maximum output current that require pulse shapers and current limiters to ensure that a set of standard commands can be issued. A challenge with making these commands flexible is sharing interfaces and multiplexers with low-power interfaces due to the difference in voltage.

The company's solution for HPC has some built-in flexibility, to save space and resources, that enables one HPC system to distribute its output to one among several different output pins. All these pins however are dedicated to HPC only, at design time. Another drawback with today's solutions is that since they are designed for the individual missions, they can only produce the commands that those missions require and do not necessarily translate to other satellites. A general overview of today's solution is presented in Fig. 2.6, where we can see an example with two different HPC possibilities. The M2 ASIC controls which of the latching current limiters (LCL) to use, what pulse length for the pulse shaper to create, and which output to route the command to. The input voltage is fixed and the LCLs are designed for their respective maximum currents, and can not be adjusted to other values.

This project's ideal flexible solution for HPC would be that all of the individual I/O pins can be programmed to handle either HPC or any other interface. Furthermore, the voltage amplitude, pulse width and maximum output current should be able to be set to any value within their respective interval instead of having certain pre-decided levels. This way the hardware design could be reused in future satellites with changes only in the software's instruction list. It would also open up for the possibility to manage HPC pulses with unexpected requirements different from those used so far.

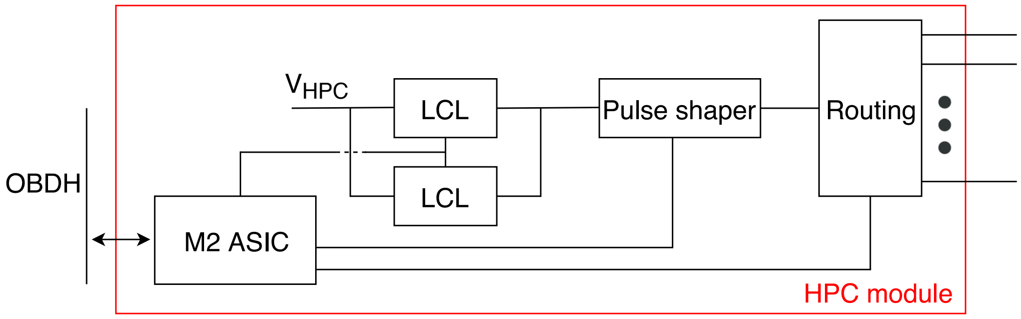

Figure 2.6: General overview of a HPC-module design. The M2 ASIC controls which current limiter to use, how long a pulse to create and which output to route the command to.

2.4 Propulsion

An important part of the RIU's functionality is to control and monitor the propulsion systems of the satellite. The propulsion systems themselves vary from mission to mission based on the orbit path used and the size, mass and lifetime of the spacecraft [8, ch.11]. Additionally, these systems are extensive in terms of components used, so the conditioning blocks required are both large and varied.

As mentioned earlier, the propulsion part of the RIU is not within the scope of this project due to the time it would take to define a universally flexible solution. However, there are parts regarding the control and monitoring of the propulsion systems that coincide with the analog-voltage acquisition or HPC. The final design definition will therefore include some interfaces related to propulsion, but not specifically designed for it.

2.5 Conditioning Blocks

All types of equipment described above need circuits designed specifically to condition their signals for the system to achieve the desired function. These conditioning blocks will most often adjust a voltage to fit the range of an ADC, translate a resistance or current into a readable voltage, or convert the RIU's central communication device's instructions to the correct type of output signals, for example HPC.

Fig. 2.7 shows a conceptual overview of today's telemetry-module designs. Every different kind of sensor uses two pins to connect to the RIU. Every pair of I/O pins is dedicated to a certain conditioning block, and can therefore only handle the type of signals that said conditioning block is designed for. This part of the design, which prevents any adaptability of the pins, is the main reason why the RIUs lack flexibility. This inflexible design method compels the customer to decide on the exact distribution of equipment on the satellite before the design process can begin. A more flexible solution could mean that the product can be configured later in the manufacturing process, giving the customer a more versatile experience. It could also open up the possibility for the same RIU design to be used in several different satellites.

2.6 Summary

The RIU handles all the signals that pass from the satellite's sensors to the OBC and from the OBC to various equipment. All these actions are performed when requested by the OBC. Today's solution uses an FPGA that communicates via a data bus to several M2 ASICs, which handles a number of conditioning blocks each. These conditioning blocks typically adjust a voltage level from the input or produce a certain electrical command for the output.

The fact that every pin has its own dedicated conditioning block is the main reason for the RIU's insufficient adaptability. As a result, every new RIU needs its hardware re-designed for the I/O budget to match the product specification.

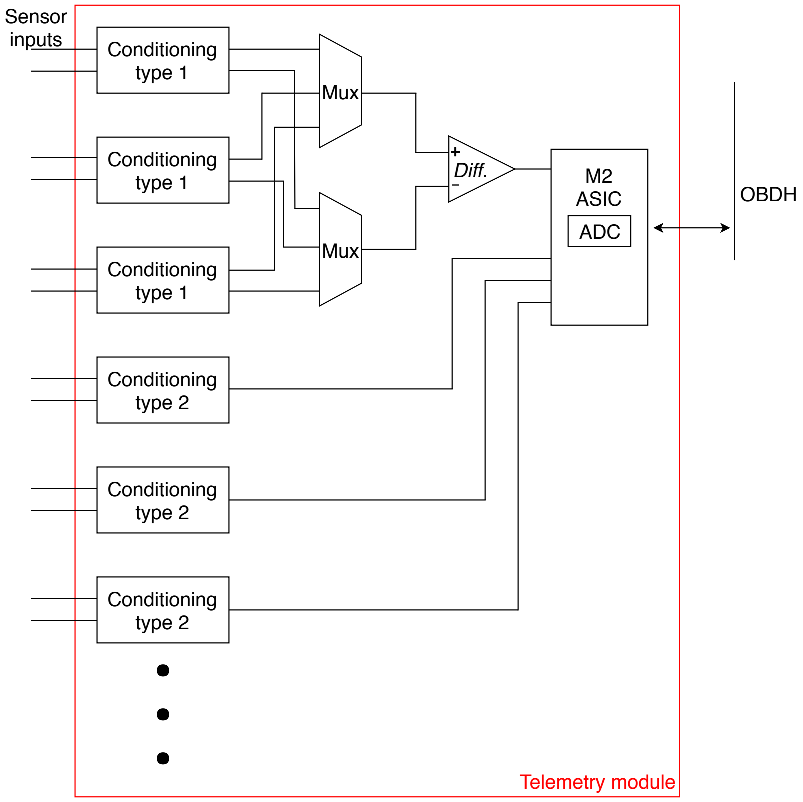

Figure 2.7: Conceptual overview of a today's telemetry-module designs. The input signals are conditioned by their own dedicated circuits before they are read by the ADC in the M2 ASIC.

3

Related Work

In this chapter we present the results of other projects that have faced similar problems. The chapter starts by introducing a solution regarding standardized sensors and then moves towards versatile interfaces that are investigated in the avionics field. Lastly we present our conclusions on programmable analog arrays and some interesting products that are available for off-the-shelf purchase.

The reduction of NRE costs and development times for spacecrafts has been an important issue for several years. During 2004, Air Force Research Laboratory (AFRL) in the USA, started the development of Space Plug and Play Architecture (SPA) as part of their goal to reduce NRE costs and development times for spacecrafts [10–12]. The idea behind SPA is to use standardized interfaces for both hardware and software so that communication between them is simplified. This technology requires sensors and systems that use a plug-and-play (PnP) interface in order for them to be connected through the five subnets supported by SPA, which are I^2C, USB, SpaceWire, Optical, and local UDP sockets.

SPA is a good step on the way towards the PnP space applications that this project was looking for, but it does not fulfill all needs. The RIUs manufactured by RUAG Space AB do not use PnP interfaces or communication protocols compatible with the five subnets of SPA. It is likely possible to design a FRIU that handles all the standards of SPA, but this solution would still have one significant difference in approach, making it incompatible with our project. Since this project was performed in collaboration with RUAG Space AB, it is important that the solution is compatible with the customers of the company. The solution that SPA offers has the philosophical difference that it requires the customer to adapt to the manufacturer, unlike our and RUAG Space AB's desired solution which preferably should enable the manufacturer to adapt to the customer needs.

If we instead of flexible and standardized sensors investigate the possibility to implement versatile interfaces that can adapt to the sensors, we can look towards the avionics industry where this type of solution has been of interest. Canu et al. [13] propose an architecture for a reconfigurable interface for use in signal acquisition in avionics systems. In their case the avionic computers need to interact with various sensors and communication buses. This interaction uses dedicated interfaces, similar to the current RIU, which limits the functionality and possibilities of reuse. Canu et al. propose a versatile interface based on programmable resources.

One main difference between the RIU system and the case presented by Canu et al. [13], is that their system only handles analog voltage acquisition and does not have the same variation in signals as the RIU. A result of this limitation is that every conventional avionic interface that the authors consider contains the same functional blocks. Furthermore, they present the characteristics of the five most common inputs, which are either single-ended or differential voltage acquisition, and these use significantly higher voltages than their RIU counterparts. Due to the differences in voltage level and that it is not realizable to design a versatile interface for all of the RIUs sensors and systems this solution is not applicable in our case. However, for the voltage acquisition interfaces in the RIU it is possible to use a similar approach with programmable impedance and level adaptation if there is a need to reduce the number of conditioning blocks.

The field of programmable analog electronics provides solutions that could implement parts of the signal conditioning needed. Field programmable analog arrays (FPAA) are however mostly directed towards audio and bio-medical applications and there are no devices that we could find that are designed for space applications [14, 15]. The idea of mixed-signal programmable electronics has instead taken a step towards FPGAs, microcontroller units and programmable analog front ends (AFE). In this area Microsemi have introduced the SmartFusion device, which consists of an ARM Cortex-M3 microcontroller unit, FPGA fabric and a programmable analog front end [16]. The analog part promises analog signal conditioning blocks with voltage, current and temperature monitors which matches our needs. However the SmartFusion device is not aimed at space applications and there can be no guarantee that it has sufficient radiation protection.

Microsemi has also developed the integrated circuit LX7730, which is a radiation tolerant telemetry controller [17]. This device uses a multiplexer with 64 universal inputs and has a programmable current source that can be directed to any of the 64 universal inputs. Application notes show that a thermistor for instance, can easily be connected between two channels and use the current source on a channel connected to the thermistor in order to measure the voltage over the thermistor [17]. However this device has been investigated by RUAG earlier and even though it is promising due to its possibilities, the price range is an obstacle. In order to use this device in a flexible system that this project aims to define, a relatively large number of LX7730 circuits would be required in each RIU, making the option too expensive.

To summarize this chapter; even though no direct solution was applicable to our problem we did get some inspiration and got a clearer view of the problem at hand. Regarding standardized sensors, which would yield the possibility for flexible interfaces, they contradict the vision of the company and therefore the aim for this project. When instead looking towards versatile interfaces, we can draw inspiration on how the voltage acquisition is performed by reconfigurable interfaces but this solution is not applicable to the other interfaces. Lastly, when investigating products that are available today we can see that they either do not meet the requirements needed to be part of the solution or that they are too costly to implement.

4

Concept and Evaluation Methods

This chapter presents the initial FRIU concept that the rest of the project was built on. It then explains how we verified if the concept is feasible to implement, and what aspects that were investigated to evaluate its expected performance. During the feasibility-verification process, more detailed information about implementation suggestions were derived, and is presented in section 5.1.

4.1 The Concept

Due to the differences in most of the conditioning needed for each interface, the solutions described in related work were considered insufficient for our problem. The flexible solution that was defined in this project was instead to use analog multiplexers to enable inputs and outputs to utilize several different conditioning blocks. The multiplexers are controlled by a microcontroller which also houses the ADC used for voltage acquisitions in the different interface types.

Since propulsion was not part of the scope in this project, the FRIU was to handle the telemetry interfaces along with HPC generation and distribution. During the investigation of the multiplexer solutions, which is presented in section 5.1.2, it was discovered that it is not possible to implement a HPC solution together with the flexible I/O due to the high voltages and currents that HPC entails. Consequently the initial goal of a system with fully flexible I/Os for both telemetry and HPC interfaces is not achievable with this approach, however a FRIU that can handle the telemetry part in a flexible way is still beneficial and was from here on the new goal of the project.

Fig. 4.1 shows a conceptual design of a module that handles the telemetry measurements in the FRIU. The multiplexer network is the core of the introduced flexibility. By allowing essentially all pins on the telemetry module to be interchangeable, the multiplexers enable a single conditioning block to be used for all acquisitions of its designated type. Since all of our interfaces need two input pins each, the pins are distributed over two analog multiplexers to enable the system to make use of both input pins simultaneously. The way that the multiplexers are connected back-to-back enables a pair of input pins to be routed to any of the conditioning blocks, which in their turn can be routed through a multiplexer to the ADC. The conditioning blocks that are a part of the proposed solution are presented in section 5.1

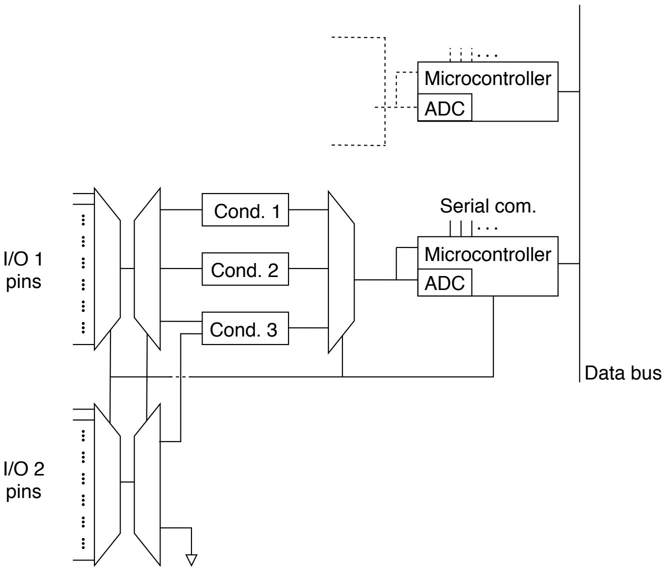

Figure 4.1: Telemetry concept

where the feasibility of the concept is evaluated.

Every telemetry module in the concept has a microcontroller that replaces the previously used M2 ASIC. The microcontroller is therefore responsible for controlling the module and communicating with the central communication unit in the RIU. The microcontroller also opens up possibilities for digital processing of the signals; this is further elaborated on in section 5.1.6. The interfaces that handle serial communication such as SDI and UART are possible to include in the multiplexer system, however this would not be practical since signals can not be received without preparing the multiplexer setting in advance. All serial communication is therefore handled by the microcontroller outside the flexible I/O system. The microcontroller can handle both BSD and UART but not SBDL due to the negative voltage levels needed, consequently interface components are still needed to enable BSD or UART over SBDL. The general purpose I/Os on the microcontroller do however still offer a degree of flexibility since they can be used for both BSD or UART depending on the loaded configuration.

4.2 Verifying Feasibility

To verify the feasibility of the concept, we designed the conditioning blocks needed for the concept to function and ensured that it is possible for a multiplexer system to support them. The conditioning blocks are heavily inspired by previous RIU designs and the circuit designs suggested by the European Cooperation for Space Standardization (ECSS) [18]. Since the conditioning blocks for past RIU designs were implemented without the flexible concept, the main task here was to ensure that they could be translated to a design where they keep their functionality when implemented with the multiplexer network. The requirements used to verify the multiplexers were extracted in the process of designing the conditioning blocks that are affected by the multiplexers characteristics. We then explored the possibility to procure multiplexers that satisfied said requirements.

A requirement in previous systems has been low-pass filtering of the input signals, where the current solution is to use resistor-capacitor (RC) filters. These RC filters are located at the inputs of the RIU which is not a possibility in the FRIU concept due to the flexibility that the FRIU is supposed to entail. If there are to be RC filters they need to be implemented together with the conditioning after the I/O multiplexer but a problem that arises is that the RC-filters include capacitors that create long settling times for the circuits. This is not a problem if the circuits are biased constantly, but our flexible concept breaks the circuit, forcing it to charge the capacitor again. As a consequence the possibility for low-pass filtering digitally with the microcontroller needs to be investigated and confirmed for the feasibility of the concept. The verification of the conditioning blocks, the multiplexers and the digital filtering is presented in section 5.1.

4.3 Evaluation of Timing and Flexibility

To evaluate the concept, flexibility and performance was considered. When evaluating the concept in terms of performance, we mainly focused on timing requirements. Regarding this aspect, there are two factors that are applicable from previous solutions and projects, communication and acquisition timings. As a reference point for communication, the OBDH bus used by the previous solutions were considered. The OBDH bus operates at a rate of 500 kbit/s nominally and each message, command or response, consists of 32 bits. These two factors translate to a time frame of 64 μs for each message, which is the requirement used for comparison with the FRIU concept.

Since the previous solutions were tied to the OBDH slots for acquisitions and that they lacked flexibility the acquisition times could could not be used as reference for the FRIU concept. We instead used typical requirements from previous projects that defined the update rate for the I/O budget as a requirement, i.e. the rate at which all of the stored values for all interfaces shall be updated with new data. This is typically 8 Hz and only limits the time for acquisitions of the interfaces, which means that the time for sending or storing the data is not included.

The time needed for configuring the multiplexers is introduced in the FRIU concept, but there is no reference point in the current solution. Consequently it will be evaluated as an addition to the total acquisition time for a sensor value. Another addition concerning the total acquisition time is brought by the digital filter implementation, which will add processing time after samples have been acquired by the ADC. One factor that persists from previous systems regarding the acquisition time is the settling time, i.e the time needed for a signal to settle to a voltage level given by the ADC resolution. In total the acquisition time for the FRIU concept will consist of configuration time, settling time and digital processing.

In addition to flexibility brought by the I/O ASIC, flexibility is added by the software of the microcontroller. In this aspect, it is difficult to compare with previous solutions or use them as a reference point, instead the possibilities added by software is presented as they are a part of the FRIU concept. These possibilities and their implications are then further discussed chapter 6. The last part of the evaluation covers a comparison of the FRIU concept with previous projects from RUAG Space AB, which evaluates how well the concept adapts to different I/O budgets. If the introduced flexibility enables the concept to handle the variations in the I/O budgets of multiple previous projects, it would indicate to what extent the concept can handle future variations.

4.4 Chapter Summary

In this chapter, a concept with configurable inputs and outputs was defined. The concept was designed to enable the system to receive different kinds of signals without the need for hardware changes. To verify if the concept was feasible to implement, conditioning blocks, multiplexers and digital filtering were investigated. The expected timing performance of the concept was evaluated, and the concepts ability to cover previous RIU projects were explored.

5

Verification and Evaluation Results

This chapter presents the result of the evaluations explained in chapter 4. First the feasibility of the concept is verified, then the improvements that the concept entails, compared to today's solution, are evaluated.

5.1 Feasibility

The feasibility verification is done in two different areas as described in section 4.2. First the conditioning blocks are redesigned to make sure that they can maintain their functionality when implemented with the multiplexers. This process resulted in requirements for the multiplexers, and possible solutions were explored. Lastly the microcontrollers ability to digitally filter the signals is explored.

5.1.1 Temperature Sensor Monitor Interface

The TSM conditioning block is shown in Fig. 5.1. When connecting multiplexers in series with the TSM interface, the introduced resistances will affect the resolution of the measurements negatively, hence it is important that the multiplexers have a resistance as low as possible. To calculate the highest tolerated resistance in the multiplexers, we first calculated at what resolution our system could measure the temperature if multiplexers with $0\,\Omega$ were implemented, and then increased the resistance until we broke the resolution requirements. The added resistance from the multiplexers will have the strongest effect on the thermistors that operate with lowest resistances, which in our case is PT100. This is therefore the thermistor that we based the multiplexer's resistance-requirements on.

The resolution of the TSM acquisition is measured by the temperature difference needed on the thermistor for the ADC to flip its least significant bit (LSB) and depends on several factors. The ADC's voltage range together with the number of bits decides what voltage difference a bit shift will represent. The resistances used in the voltage divider will affect how big a resistance change that will be needed for the voltage of a bit shift to occur, and the thermistors conversion rate will decide the temperature change needed for the corresponding resistance change. The

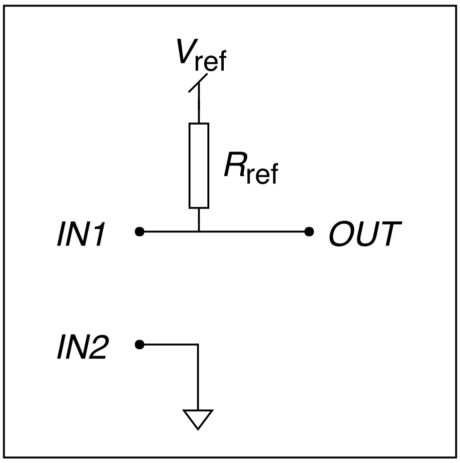

Figure 5.1: TSM conditioning block

microprocessor that we used as a baseline for this project has an ADC that operates in the voltage range of 0 V to 3.3 V using 12 bits. When using the PT100 thermistor along with the requirements on the acquisition, it operates with the temperatures −10 °C to 1000 °C corresponding to the resistances 96 Ω to 433 Ω. The requirement on resolution in this case is 0.8 K/LSB.

The critical point for resolution is at the highest resistance, 433 Ω. We therefore chose to set the reference resistance to the same value to achieve the best possible resolution at this point. The calculations were done using Matlab and Simulink, and the results are shown in Fig. 5.2. As can be seen in the graph, a 12-bit ADC is not enough to represent the thermistors values with the required resolution of 0.8 K/LSB. The microcontroller does however have the possibility to over sample the signal and achieve a higher resolution, as described later in section 5.1.6. Fig 5.2 shows that a 13-bit ADC satisfies the resolution requirement and that there is room for a multiplexer resistance of 80 Ω. The same calculations were performed for the remaining thermistor interfaces, and the conclusion was still 80 Ω. All the thermistors used in previous RIUs can be covered by the five conditioning blocks presented in Table 5.1

Table 5.1: Values for the six TSM conditioning blocks

TSM component and voltage values		
Thermistor type	$R_{ref}[\Omega]$	$V_{ref}[V]$
PT100	433	6
PT200	562	6
PT500	2210	6
PT1000	1785	6
ANY/ANF	12.1k	3.3

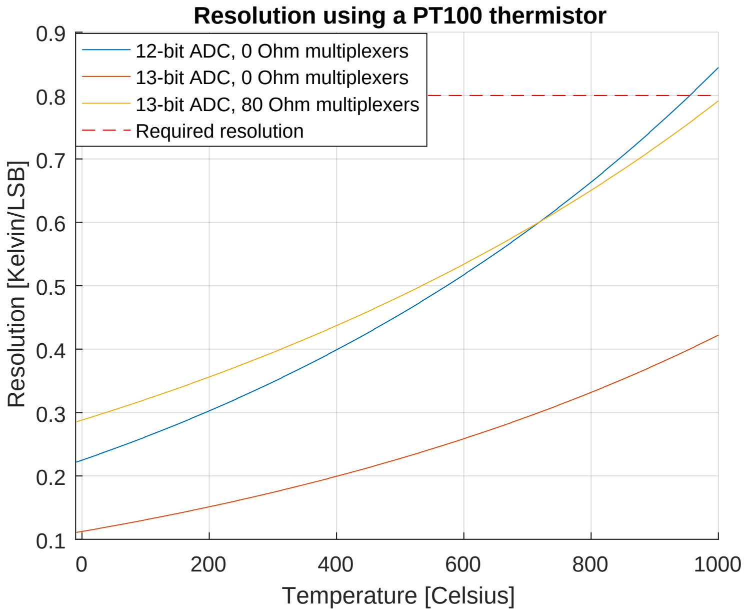

Figure 5.2: The resolution when measuring a PT100 thermistor with different settings on the ADC and multiplexer resistances

5.1.2 Multiplexers

Based on the requirement on resistance from the TSM conditioning, we set 80 Ω as the multiplexer's highest tolerated resistance. This resistance along with a voltage range of 0 V to 10 V for the ASM acquisition presented in section 2.2.1 were the requirements that we set for the verification of the multiplexer.

To reduce cost, we initially tried to stay with off-the-shelf products, but we realized that the radiation-hardened analog multiplexers available did not satisfy our requirements. The multiplexers that can handle the voltages over 10 V for the ASM system, have resistances that are too high for the TSM systems to achieve the required resolution. This could technically be solved with several multiplexers implemented in parallel, but it would need too many, making the design unnecessarily large. Since we could not find an off-the-shelf analog multiplexer that fits our requirements, we decided to investigate the possibility of implementing the multiplexer on an ASIC. This would also open up the possibility to implement the conditioning blocks on the same ASIC, as shown in Fig. 5.3. In accordance with the project's limitations we did not map the components on the silicon, but developed circuit diagrams for the conditioning blocks and ensured that the requirements on the multiplexer are realistic to implement on an ASIC. An important note here is that the goal was not to design a multiplexer to be used in the final product, but to verify that it was possible for a multiplexer to fulfill our requirements.

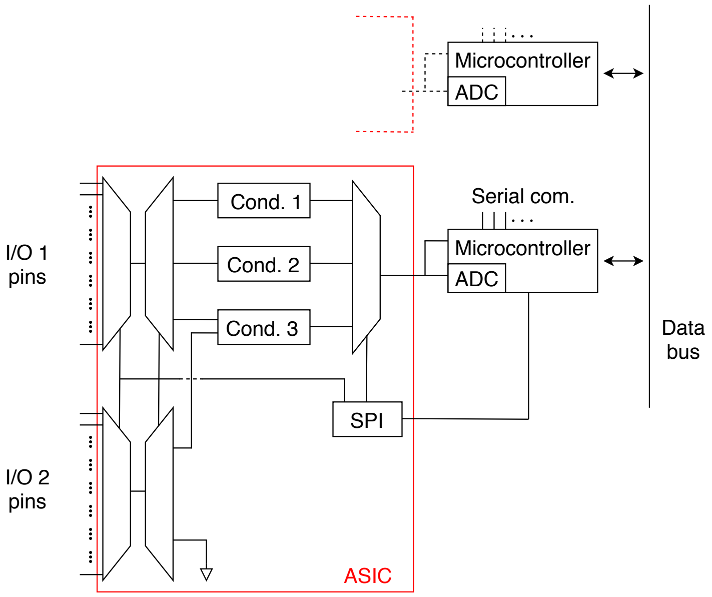

Figure 5.3: The concept after the multiplexers and conditioning blocks have been implemented on an ASIC

The technology used in mixed-signal ASICs designed by RUAG Space AB is I3T80, which is a 0.35 μm process [19]. Consequently when verifying the multiplexer on ASIC there were two conclusions that we wanted to reach. If it was possible to implement the multiplexers on an ASIC, and if it was possible to implement specifically in the I3T80 technology.

Using the transistors in the I3T80 process for the multiplexer channels would yield a resistance of around 3 Ω which is significantly lower than the off-the-shelf analog multiplexers. A downside with this ASIC technology however is that the maximum gate-source voltage of the transistors is 3.6V, which can make it problematic to achieve the required maximum drain voltage of 10 V. To achieve this function, we would have to implement a system where the gate voltages of all the multiplexer's transistors are adjusted to their individual source voltages, even on the transistors that are turned off. We decided that this system was overly complicated for the task and investigated other ASIC technologies.

A simpler solution would be to use the I3T25 technology, also a 0.35 μm process, that has transistors with similar characteristics but a maximum gate-source voltage of 12 V. This way the gate voltage can be set to a voltage slightly under 12 V when the channel is activated and to 0 V when it is deactivated, allowing the channel to manage the required voltage span. The resistance of these transistors is half of the ones in I3T80, taking the resistance down to around 1.5 Ω. It is likely that this resistance will be significantly higher when then input voltages get close to the gate voltage, which it do during the ASM acquisition. The resistance does however only affect the measured values in the TSM acquisition, which does not reach higher voltages than 3.3 V. With this information we drew the conclusion that it is possible to design a multiplexer on an ASIC with characteristics that can support the FRIU concept.

5.1.3 Analog Signal Monitor Interface

The conditioning blocks for ASM are designed in a similar way as today's solution, with a differential amplifier measuring the differential voltage of the two inputs. There is however a problem that occurs if the used ADC can only handle positive voltages and the resulting differential voltage has a negative value. This problem can be solved by implementing the extra circuitry shown in Fig.5.4. The solution includes a comparator and a switch system that together can route the highest input signal to the non-inverting input of the differential amplifier and the lowest input signal to the inverting input. Consequently the differential amplifier will output the absolute value of the differential voltage instead of negative voltage levels. This method would require an extra output for a control bit connected to a digital input on the microcontroller to indicate if the original differential voltage was positive or negative. Since the digital control signal would be routed outside of the multiplexer which carries the analog signals to the microcontroller, it would occupy an extra output pin on the ASIC.

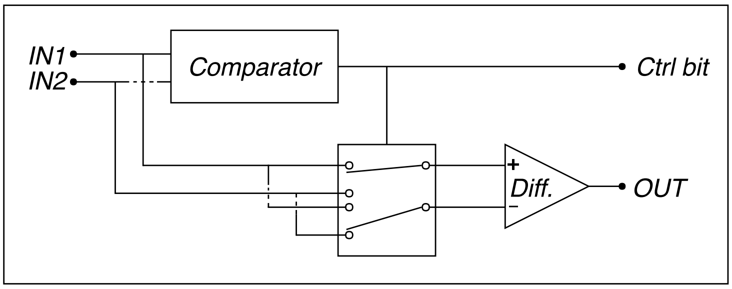

Figure 5.4: ASM conditioning block with compare function

Since the ASM interface has to handle several different voltage amplitudes, there are a corresponding number of conditioning blocks available, all with a gain adjusted to the desired voltage level. The gain is set to adapt the maximum differential voltage to the microcontrollers maximum voltage, in our case 3.3 V. We suggest the gains presented in Table 5.2, since they will cover all the versions of ASM that we have seen in previous RIUs.

Table 5.2: Values for the ASM conditioning blocks

ASM conditioning blocks	
Voltage range [V]	Gain
0;0.035	94
0.005;0.05	66
-0.2;0.2	15
0;2.5	1
-5;5	0.66
0;5	0.66
-6;6	0.55
-10;10	0.33

The conditioning blocks designed for ASM can also act as an interface for the BDM acquisition. BDM uses the same voltage levels as some of the ASM interfaces, but is handled differently in the software. The gain of 94 is used in the solution for SAS that is explained in section 5.1.4.

5.1.4 Sun Acquisition Sensor Interface

If the multiplexer had a negligible resistance, the conditioning block for sun sensors could be identical to today's solution presented in Fig. 2.4. Unfortunately this is not the case in the proposed multiplexer solutions that we investigated in section 5.1.2. Instead we propose the solution presented in Fig. 5.5, which is the only interface implemented with the multiplexer system that will need adaptation on hardware level. The idea is that a set number of input pins will be mapped next to each other on the PCB with the possibility of adding a resistor between them. The signal will then be routed in a similar fashion as for the analog voltage acquisition interfaces, to differential amplifier op-amp of the correct gain and into the ADC. This would enable the designer to, very late in the production process, make a decision of how many input pin pairs that are to be dedicated to sun sensors. Before the shunt resistor is applied, the pins with this option are still as versatile as all the other pins and can handle any of the telemetry systems. The signal will then be routed to a conditioning block of the same type as for the ASM, but with a gain chosen specifically for the SAS acquisition.

The SAS interfaces implemented in the RIUs that we investigated have always been in the current ranges of [0A;10mA] or [0A;35mA]. They have always been implemented using a shunt resistor of 3.48 Ω or 1 Ω which yields a voltage drop adapted to a differential amplifier of the same gain. With our ADC using 3.3 V, we choose a gain of 94 to scale the voltage correctly.

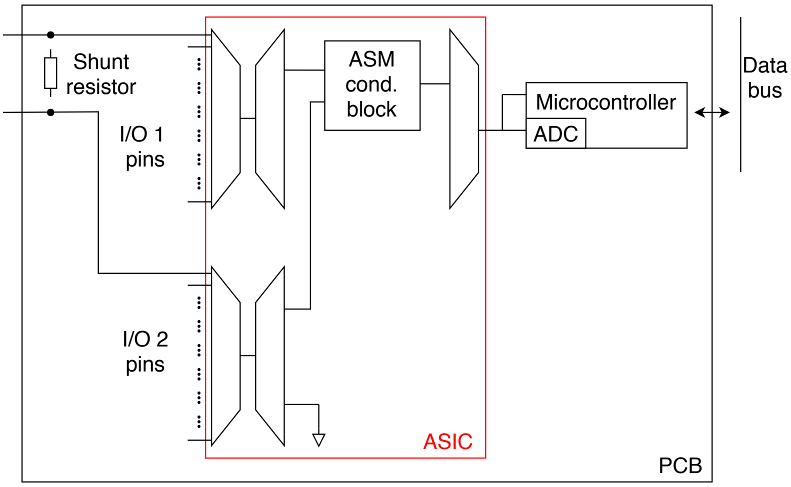

Figure 5.5: Hardware solution for including sun sensors in the multiplexers flexibility

5.1.5 Bi-level Switch Monitor Interface

The BSM interface uses a conditioning block similar to TSM but with extra voltage dividers to adjust the voltage level for the ADC. The important requirements to mention here is the allowed output current of 0.5 mA to 1 mA when the switch is closed, and the allowed voltage span of 3.7 V to 15 V. We did however design the circuit to output no more than 10 V, since this is the maximum voltage for our multiplexer solution. The conditioning block in Fig. 5.6 with the resistances $R_1 = 20k\Omega$, $R_2 = 28k\Omega$, $R_3 = 12k\Omega$ were chosen to fit these requirements.

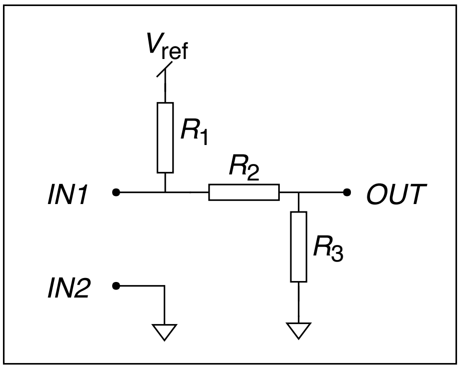

Figure 5.6: Conditioning block for BSM interface

5.1.6 Digital Filtering

A key factor that enables the introduced flexibility is the possibility to implement a low-pass filter digitally, to avoid the long settling times in analog RC filters. It is possible to configure the analog front end controller (AFEC) of the Atmel SAM v71 to oversample the signal, increasing the resolution and providing low-pass filtering at the same time.

The oversampling rates (OSR) available are 4, 16, 64 and 256 which corresponds to 13-bit, 14-bit, 15-bit and 16-bit resolution respectively. When using an OSR of 16 for example, the AFEC will take 16 consecutive samples of the signal and output the average of these. Consequently the effective sampling frequency, as seen by the core processor, will be decreased by a factor equal to the OSR, but have its resolution increased to the corresponding number of bits.

Averaging with the OSR settings available in the AFEC filters the signal with the frequency response shown in Fig. 5.7. Since the cut-off frequency is normalized to the sampling frequency for the filters, the cut-off frequency for an averaging filter will be decided by the total sampling period. In previous projects the filtering requirement has been a low-pass filter with an attenuation of 3 dB at the cut-off frequency of 350 Hz, which corresponds to a sampling period of 2.56 ms. If the AFEC is to use its built in averaging function, it is bound to a sampling frequency of 1.7 MHz. The longest sampling period that can be achieved with this sampling frequency is 150.6 μs with the averaging factor of 256, resulting in a cut-off frequency of 5.874 kHz. Consequently averaging in the AFEC does not yield the required cut-

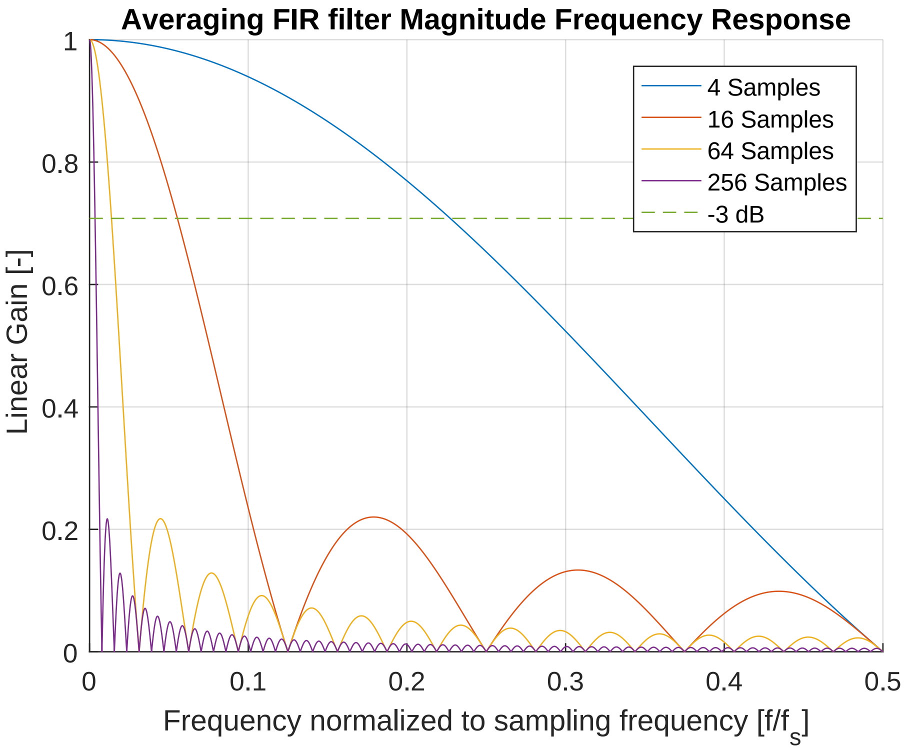

Figure 5.7: Magnitude frequency response for averaging FIR filters with different amount of samples together with a reference line corresponding to −3 dB.

off frequency, and filtering needs to be done in the core processor. As a result, there is no reason to choose an excessive OSR for the purpose of filtering.

When averaging in the core processor, the sampling frequency and number of samples per signal needs to be chosen to correspond to the sampling period of at least 2.56 ms. For evaluation purposes we chose to reduce the effective sampling frequency to 6.25 kHz and then use 16 taps in the filter. Each sample is acquired by using an OSR of 4 in the AFEC to obtain the required 13-bit resolution.

5.2 Timing Performance

The performance evaluation analyzes the time it takes to acquire and filter information from the sensors, followed by suggested software implementations for scheduling this process to improve acquisition performance. The I/O coverage is then evaluated to give an estimation of how well the concept can cover the I/O budgets for future satellite projects.

5.2.1 Acquisition Time

The microcontroller handles all the communication in the FRIU concept. Instead of using an OBDH bus, communication can be handled over controller area network (CAN) which can support data rates of up to 1 Mbit/s. With communication over CAN, the fixed time slots for command, acquisition and responses are no longer needed. Consequently the point in time when the signal has been sampled and processed is now when the response can be sent back, instead of having to wait for the next transmission slot like with OBDH. As a result, configuration time, settling time and the time for needed digital processing are now the only factors that affect the time between command and response.

When estimating the configuration time for the multiplexers, the limiting factor is the rate at which the I/O ASIC can receive data. Based on similar ASICs previously designed by RUAG, a data rate of 1 Mbit/s can achieved when transmitting over a serial link. When configuring the multiplexers, 11 bits are needed when 64 I/O pairs and 16 to 32 conditioning blocks are used. When using some kind of message control with start and stop bits the total configuration time is approximated to 15 μs.

The settling time depends heavily on the capacitance of the cable that connects the sensor to the RIU and the resistance used in the conditioning. In order to calculate the time needed for the signal to settle to an accuracy of one LSB, given a certain ADC resolution, the following equation is used

$$\text{settling time} = \tau \cdot \ln(2^N) \tag{5.1}$$

where τ is the time constant for the circuit and N is the number of bits of resolution. This equation holds when the signal has to swing between the extremes, i.e. from no charge to required level given by the resolution, which is the case with switched conditioning. TSM acquisitions have the longest settling times due to the large resistances in some thermistor types. The left circuit in Fig.5.8 shows the cable capacitance in the TSM conditioning block. When calculating the time constant, this circuit is mathematically equivalent to the right circuit, which can treated as a conventional RC-filter since the parallel resistors can be represented by one corresponding resistance. This model was used to calculate the time constants for the TSM interfaces.

Since the thermistor resistance affects the settling time, it varies between measurements and depending on what thermistor type that is used. The longest settling times are for ANY thermistors, which can have a resistance of around 440 kΩ together with a reference resistance of around 12.1 kΩ, as stated in Table 5.1. The shortest settling times are for PT-100 thermistors, which can have a resistance of 433 Ω together with a reference resistance of the same value. Consequently the settling times can vary between 4 μs and 210 μs depending on thermistor type, when calculated with the worst-case cable capacitance of 2 nF as specified by the ECSS [18].

The remaining part of the acquisition time is digital processing, consisting of ADC sampling and filtering. As presented in section 5.1.6, the time needed for filtering is 2.56 ms. Together with the configuration time of 15 μs and settling time

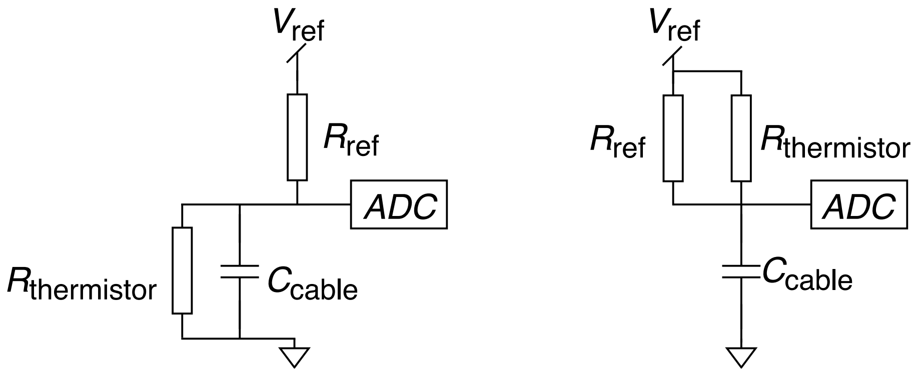

Figure 5.8: The left circuit represents the TSM acquisition circuit. The right circuit is a mathematical equivalence for the left circuit when calculating the time constant τ.

of 210 μs, the total acquisition time is 2.77 ms for the worst case.

The requirement for acquisition timing is set by the total I/O-budget's update-rate. This rate is typically 8 Hz which results in the total time of 125 ms for updating every telemetry value on the unit. The FRIU concept has 64 I/O pairs per ASIC. Since each I/O ASIC can be operated in parallel, the total time is applicable per I/O ASIC. Consequently 44 of 64, or approximately 70 %, of the acquisitions can be performed at a rate of 8Hz, if they are all of worst-case type. For an acquisition to be considered as worst-case type, it needs to be a TSM acquisition of the types ANY or ANF, where the cable capacitance exeeds 0.5 nF. The consequence of this result is further discussed in chapter 6 since there are possibilities to manage the total I/O budget with software implementations and hardware design choices.

5.2.2 Acquisition scheduling

The microcontroller enables further flexibility, with the use of software, to control the I/O ASICs and acquisitions. This aspect is difficult to evaluate with concrete results. However, in this section two methods that can increase timing performance are presented. The methods are based on the performance and specifications for the microcontroller Atmel SAM v71 that was used as a baseline in this project.

The first method is the possibility to handle several I/O ASICs with the same microcontroller. This option is enabled by the 12 ADC-channels available in the AFEC and the fact that the microcontroller operates at a significantly higher frequency than the effective sample frequency. Considering that the effective sample frequency is 6.25 kHz, there are 160 μs between each sample. With the acquisition time being less than 4 μs to obtain a resolution of 13 bits, this leaves 156 μs for the microcontroller to perform other tasks. If all 12 ADC-channels are connected to different I/O ASICs, it would yield 13.3 μs of processing time for each channel. In this time the microcontroller needs to switch ADC channels, perform the acquisition and handle the result.

The second method is to perform parallel acquisitions on the same ASIC. This method is made possible by the high operating frequency and the fact that several acquisition types have lower settling time than 160 μs. The TSM acquisitions regarding PT thermistors have settling times varying between 4 μs and 20 μs. Together with configuration of the I/O ASIC and acquisition in the ADC, the total time varies between 20 μs and 40 μs. These factors enable the microcontroller to configure the I/O ASIC to a new thermistor acquisition, wait for the signal to settle and then perform the acquisition before the first thermistor needs to be sampled again. The result is that within the 160 μs between each sample, there is a possibility of performing 4 to 8 parallel acquisitions of PT thermistors.

Considering other interface types than TSM this method shows further possibilities of increasing performance. The reason behind this is that the settling times for ASM and SAS acquisitions are negligible compared to the time for the ADC to acquire one sample. Furthermore there are interfaces that do not require filtering, for example BSM, that only needs to determine if the voltage is high or low. Consequently acquisition of these interface types may be performed in parallel to a greater extent than the TSM acquisitions. Both methods can be used to achieve the required performance, but a combination of the two could be used to save resources such as area and power usage. This approach would lead to a trade-off between parallelism on one I/O ASIC and utilizing the same microcontroller with several parallel I/O ASICs, and is discussed further in chapter 6.

5.2.3 Interface Coverage

The FRIU covers the five interface categories ASM, SAS, TSM, BSM and BDM. This means that they can be implemented by plugging them into any input on the FRIU, without the addition of extra hardware, with the exception of a simple shunt resistors for the SAS interface. The FRIU can not handle HPC or SBDL without the need for tailored extra hardware, and these are therefore excluded from this evaluation. To evaluate how well the FRIU covers the promised interfaces, a compilation of said interfaces used in previous RIUs was made, see Appendix 1. A summary of the compilation is shown in Table 5.3. This compilation was also used as a base for designing the conditioning blocks presented in section 5.1. All interfaces were however not possible to implement with the flexible approach that this project was based on. Project five has a non-recurring ASM interface type with too high voltage levels for the multiplexer, and project two has a non-recurring BDM interface type with the same problem. Aside from these two exceptions, the FRIU covers every instance of the five interface categories.

5.3 Chapter Summary

The conditioning circuits for TSM, ASM, SAS, BDM and BSM were redesigned to verify that the function is maintained after the implementation of the multiplexer network. This resulted in extra requirements for the multiplexers, which were also satisfied. The redesigning of TSM, ASM, BDM and BSM interfaces resulted in

Table 5.3: Summary of the tables in Appendix 1, which shows the FRIU concept compared to previous RIU projects and their I/O budgets for the telemetry interfaces. I/O coverage shows how many of the promised inputs that were covered and I/F coverage shows how many of the promised interfaces that were covered unrelated to the number of instances. The table also shows how many ASICs that would be needed to cover all the inputs, and WCT shows how many of the interfaces that had the possibility of being of worst-case type.

Coverage summary				
Project	I/O Coverage	I/F Coverage	ASICs	WCT
Proj.1a	100%	-/-	5	56%
Proj.1b	100%	-/-	17	46%
Proj.2	98%	12/13	13	54%
Proj.3	100%	9/9	8	58%
Proj.4	100%	7/7	13	55%
Proj.5	98%	8/9	7	53%

completely configurable I/O pins. The SAS interface's conditioning on the other hand needs a simple adaptation on hardware level, but is still able to be redesigned late in the development process.

The possibility for digital filtering is verified and a time frame needed for the required cut-off frequency is presented. This time frame is then used in the performance evaluation. The acquisition time is shown to be a challenge if 64 I/Os are to be used, but it meets the requirements with scheduling that enables parallel acquisition and filtering. This challenge can be avoided by having fewer I/Os per ASIC, where every ASIC can be run in parallel without the need for scheduling.

The concept covers the recurring interface types that had their feasibility verified in section 5.1. This result is independently of the quantity, which shows a good potential of covering the I/O budgets for future projects as well.

6
Discussion

In this chapter we first bring out an aspect that is important to consider when designing the multiplexers used in the I/O ASIC. We then discuss the digital filtering solutions and their requirements, together with possible implications that a change in filtering requirements would bring. After the parts concerning feasibility, the results performance evaluations are discussed together with several trade-off aspects that the concept brings. Lastly, suggestions on future work for the project are summarized.

In this project, we did not take the leakage current or process variations of the multiplexer into consideration, we simply explored the possibility of obtaining a multiplexer with a low enough resistance and a wide enough voltage span. If the concept is to be implemented, while not necessary for evaluation on a conceptual level, these factors need to be carefully analyzed.

When defining the digital filter, the only requirement was the attenuation of 3 dB at 350 Hz. This requirement gives no information about the stop-band attenuation or any other aspects such as the nature of the noise. With additional information regarding the noise that needs to be filtered out, it is possible that harder or softer requirements for the filtering can be obtained. These additional requirements may help us in deciding how many samples that need to be averaged in order to suppress the noise sufficiently. Increasing the number of samples needed will limit the possibilities of parallel acquisitions on the same I/O ASIC. If the additional information regarding noise instead leads to a decreased number of samples needed, this may only further improve the parallel possibilities.

Additional filtering requirements might also result in the averaging filter not being suitable in terms of performance. In this case there are still possibilities to implement a more advanced filter in the microcontroller. Depending on filter type and the coefficients needed, it may reduce the performance of the concept if more processing time is needed to perform the filtering.

The acquisition times vary for different interfaces and if every I/O is of worst-case type, an I/O ASIC with 64 input pairs will not meet the requirements of an update rate of 8 Hz. An simple solution would be to implement I/O ASICs that have 44 input pairs or fewer, however the flexibility brought by the microcontroller shows possibilities of another approach. Given that not all acquisitions are of the worst-case type, it is possible to use I/O ASICs with 64 input pairs, with some restrictions in design choices. Since the acquisitions with shorter settling times can be performed in parallel, it is possible to partition the interfaces so that both parallel and consecutive acquisitions types are performed on the same I/O ASIC. The result of a partition like this would enable a microcontroller to perform all 64 acquisitions needed within the given time frame.

The method with parallel acquisition scheduling would yield a design with a high number of I/O pairs on each ASIC, resulting in fewer I/O ASICs being required. The method that controls multiple I/O ASICs with each microcontroller uses fewer microcontrollers, but limits the possibility for more I/O pins per ASIC. If one of the methods is implemented, it limits the possibility to utilize the other one, introducing a trade-off. However, by compromising, it is possible to achieve a combination of the two methods, resulting in a design that is more optimized in terms of power and area usage.

When comparing the FRIU concept with the I/O budgets of previous projects, as described in section 5.2.3, only two projects were not covered in full. Based on this, the concept shows good possibilities of covering the telemetry parts in future RIU projects. The interfaces that the concept cannot cover, all handled voltage levels higher than 10 V; hence if there is an indication of needing interfaces above this voltage in the future, the multiplexer solution of the concept might need to be revisited and designed to handle a higher voltage level. Worth noting about the two interface types that the FRIU failed to handle, is that they were unique in both cases, and the concept still covers all the recurring interfaces.

The result from the evaluation of the acquisition timing states that the concept can handle 70 % of acquisitions within the I/O budgets update rate. This result was based on a scenario where all acquisitions are of worst-case type. In reality this is not the case, which can be seen from the compilation of the coverage, which shows that 46 to 58 % were of worst case type. If a project has less than 70 % worst-case interfaces, this does not mean that the FRIU concept covers it by default, but instead it shows that the concept may be able to sustain the I/O budget update rate of 8 Hz. By performing parallel acquisitions of the interfaces that are not of worst-case type the concept can sustain the update rate if the I/Os are partitioned correctly, i.e. if every individual I/O ASIC in the system has a maximum of 70 % worst-case type interfaces connected to it.

The SAM v71 was used as a baseline for a microcontroller in this project. The future version that will be radiation hardened is most likely going to differ from the current version in terms of performance, which might affect the FRIU-concept's performance. To set requirements on the radiation hardened microcontroller, further investigation concerning the performance of the system is needed.

With the verification and investigations that have been performed during this project, it is possible to draw conclusions regarding the feasibility and expected performance for the concept at this stage. However, to further verify the feasibility of the concept, the leakage currents and process variations of the multiplexers have to be investigated. Additionally, the requirements on filtering together with the performance changes of the radiation hardened microcontroller would need to be explored.

7

Conclusion

This project resulted in a conceptual hardware-design that if implemented correctly, can introduce a flexibility that enables essentially any I/O pin to be configured into handling a number of different interface types. This flexibility permits the customer to make decisions late in the design process regarding the number of each interface type that is to be implemented and it gives the manufacturer the possibility to reuse the design for future projects, which can reduce NRE costs.

A key aspect to the introduced flexibility is the microcontroller, which controls the flexible hardware, handles all the digital processing and facilitates more effective acquisition-scheduling. The ability to digitally filter the signals eliminates the need for RC filters, which with their long settling times have hindered flexible RIU implementations in the past. The filtering does however also introduce some limitations for the concept, which restricts the number of interfaces that may exceed a certain settling time. If this limitation is not exceeded, the microcontroller can with effective scheduling enable the concept to meet the requirements on total acquisition times for telemetry interfaces connected to a RIU.

With the flexible I/Os, digital filtering and effective acquisition scheduling; the concept covers close to all of the recurring telemetry interfaces in previous RIU solutions. Based on the results of this project, we conclude that this concept can be further developed into an implementation that brings the flexibility needed to cover future RIU designs and reduce costs.

If this concept would be realized through a prototype it is possible to get a clearer picture on factors such as accuracy and timing performance for the measurements. Additionally, it could verify the calculations performed during this project. From here it could serve as a starting point for the way toward a final realization of the concept. If the concept shows to be reusable for multiple future missions, it is possible that the final implementation becomes a complete integrated circuit which holds the needed functionality from the microcontroller together with that of the I/O ASIC.

Bibliography

[1] ESA, “Putting Everday Computer Parts to Space Radiation Tests,” 2018. [Online]. Available: http://www.esa.int/Our_Activities/Space_Engineering_Technology/Putting_everyday_computer_parts_to_space_radiation_test [Feb. 16, 2018]

[2] C. Fransen, “Monitoring the Weather from Polar Orbit,” Noordwijk, 2006. [Online]. Available: http://esamultimedia.esa.int/docs/BR-261_MetOp.pdf [Feb. 15, 2018]

[3] ESA, “Report for Mission Selection: Biomass,” Noordwijk, 2012. [Online]. Available: http://esamultimedia.esa.int/docs/EarthObservation/SP1324-1_BIOMASSr.pdf [Feb. 14, 2018]

[4] M. Endemann, “ADM-Aeolus: the first spaceborne wind lidar,” 2006. [Online]. Available: http://esamultimedia.esa.int/docs/SP-1311_ADM-Aeolus_FINAL_low-res.pdf [Feb. 14, 2018]

[5] J. W. Howard Jr and D. M. Hardage, “Spacecraft Environments Interactions: Space Radiation and its Effects on Electronic Systems,” NASA, Tech. Rep., 1999. [Online]. Available: https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19990116210.pdf [Feb. 15, 2018]

[6] E. Stassinopoulos and J. Raymond, “The space radiation environment for electronics,” *Proceedings of the IEEE*, vol. 76, no. 11, pp. 1423–1442, 1988. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=90113 [Jan. 16, 2018]

[7] H. Klinkrad, “Space Debris,” in *Encyclopedia of Aerospace Engineering.* Chichester, UK: John Wiley & Sons, Ltd, dec 2010. [Online]. Available: http://doi.wiley.com/10.1002/9780470686652.eae325 [Feb. 16, 2018]

[8] G. L. Davis, R. R. Some, and A. a. Shapiro, *The International Handbook of Space Technology*, M. Macdonald and V. Badescu, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014. [Online]. Available: http://link.springer.com/10.1007/978-3-642-41101-4 [Jan. 16, 2018]

[9] S. Y. C. Catunda, J. F. Naviner, R. C. S. Freire, and G. A. L. Pinheiro, “Programmable gain and dc level shift analog signal conditioning circuit: Microcontroller based implementation,” *Conference Record - IEEE Instrumentation and Measurement Technology Conference*, vol. 3, pp. 1857–1861, 2005.

[10] J. Lyke, "Space-Plug-and-Play Avionics (SPA): A Three-Year Progress Report," in *AIAA Infotech@Aerospace 2007 Conference and Exhibit.* Reston, Virigina: American Institute of Aeronautics and Astronautics, may 2007. [Online]. Available: http://arc.aiaa.org/doi/10.2514/6.2007-2928 [Jan. 16, 2018]

[11] J. H. Christensen, D. B. Anderson, M. E. Greenman, and B. D. Hansen, "Scalable network approach for the space plug-and-play architecture," *IEEE Aerospace Conference Proceedings*, 2012.

[12] M. Martin, J. Summers, and J. Lyke, "AFRL plug-and-play Spacecraft Avionics Experiment (SAE)," *IEEE Aerospace Conference Proceedings*, pp. 1–6, 2012.

[13] A. Canu, P. Bénabès, D. Faura, P. Toillon, and M. Gatti, "A High Voltage Programmable Input Interface for Avionic Equipment," in *2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings.* IEEE, 2012, pp. 2464–2468.

[14] P. Falkowski and A. Malcher, "Audio signal processing based on dynamically programmable analog arrays," in *ICSES 2010 International Conference on Signals and Electronic Circuits*, Sept 2010, pp. 29–32.

[15] D. Salhi and N. Pasquino, "Functional immunity test for field programmable analog arrays (FPAA)-based biomedical application to EM radiated energy," *Proceedings - 2012 6th Asia-Pacific Conference on Environmental Electromagnetics, CEEM 2012*, pp. 343–346, 2012.

[16] Microsemi, "SmartFusion Customizable System-on-Chip (cSoC)," 2012. [Online]. Available: http://www.microsemi.com/document-portal/doc_download/131347-smartfusion-product-brief [May 3, 2018]

[17] ——, "LX7730, RAD Tolerant Telemetry Controller Datasheet," 2017. [Online]. Available: https://www.microsemi.com/document-portal/doc_download/130719-smartfusion-customizable-system-on-chip-csoc-datasheet [May 3, 2018]

[18] ECSS, "Space engineering: Spacecraft discrete interfaces," Tech. Rep., 2008. [Online]. Available: http://www.ecss.nl/wp-content/uploads/standards/ecss-e/ECSS-E-ST-50-14C31July2008.pdf [Apr 25, 2018]

[19] ON Semiconductor, "I3T80: 0.35 µm Process Technology," Tech. Rep., 2016. [Online]. Available: http://www.onsemi.com/pub/Collateral/I3T80-D.PDF [Apr 25, 2018]

A

Appendix 1

The tables in this appendix show the results of applying the FRIU concept with the conditioning blocks that were defined for it, on a compilation of I/O budgets of previous projects. It gives an approximation of how many of the telemetry systems that could be implemented if the FRIU was a finished product. The first column states the interface category, which is followed by the quantity that is covered compared with the total quantity of that category. I/F Coverage lists how many types of interfaces that are covered and included in each category.

The fourth column represents the number of I/O ASICs with 64 input pairs that is needed to cover each quantity, where the total value is rounded up since we can only implement whole ASICs. The last column state the percentage of the interfaces in that category that have the possibility to be of worst case type if they have a cable capacitance of more than 0.5 nF. Projects 1a and 1b are based on minimum and maximum values for a typical specification compiled by RUAG Space AB, and do not list the information needed for I/F coverage.

Table A.1: Coverage for the FRIU concept based on a typical specification.

Project 1a				
I/F Category	I/O Coverage	I/F Coverage	ASIC Util.	WCT
ASM	25/25	-/-	0.39	-
SAS	16/16	-/-	0.25	-
TSM	200/200	-/-	3.13	56%
BSM	25/25	-/-	0.39	-
BDM	0/0	-/-	0	-
Total	266/266	-/-	4.16 (5)	56%

Table A.2: Coverage for the FRIU concept based on a typical specification.

Project 1b				
I/F Category	I/O Coverage	I/F Coverage	ASIC Util.	WCT
ASM	125/125	-/-	1.95	-
SAS	32/32	-/-	0.50	-
TSM	750/750	-/-	11.72	46%
BSM	100/100	-/-	1.56	-
BDM	60/60	-/-	0.94	-
Total	1067/1067	-/-	16.86 (17)	46%

Table A.3: Coverage for the FRIU concept based on Project 2.

Project 2				
I/F Category	I/O Coverage	I/F Coverage	ASIC Util.	WCT
ASM	162/162	3/3	2.53	-
SAS	8/8	1/1	0.13	-
TSM	516/516	5/5	8.10	54%
BSM	72/72	2/2	1.13	-
BDM	48/60	1/2	0.75	-
Total	806/818	12/13	12.64 (13)	54%

Table A.4: Coverage for the FRIU concept based on Project 3.

Project 3				
I/F Category	I/O Coverage	I/F Coverage	ASIC Util.	WCT
ASM	96/96	3/3	1.50	-
SAS	16/16	1/1	0.25	-
TSM	284/284	3/3	4.43	58%
BSM	48/48	1/1	0.75	-
BDM	32/32	1/1	0.5	-
Total	476/476	9/9	7.43 (8)	58%

Table A.5: Coverage for the FRIU concept based on on Project 4.

Project 4				
I/F Category	I/O Coverage	I/F Coverage	ASIC Util.	WCT
ASM	61/61	2/2	0.95	-
SAS	40/40	1/1	0.63	-
TSM	535/535	3/3	8.36	55%
BSM	162/162	1/1	2.53	-
BDM	0/0	0/0	0	-
Total	798/798	7/7	12.47 (13)	55%

Table A.6: Coverage for the FRIU concept based on on Project 5.

Project 5				
I/F Category	I/O Coverage	I/F Coverage	ASIC Util.	WCT
ASM	126/134	3/4	1.97	-
SAS	8/8	1/1	0.13	-
TSM	238/238	3/3	3.72	53%
BSM	40/40	1/1	0.63	-
BDM	0/0	0/0	0	-
Total	420/428	8/9	6.45 (7)	53%