BO19E-36 Design of the ALOT PCB

A thesis by

Aleksander Waage and Sindre Jacobsen Færø

For the degree of

Electronics Engineer

Western Norway University of Applied Sciences

May 2019

Document Control

Report title BO19E-36 Designing a PCB for ALOT	Date/Version 30.05.19
	Report number: B019E-36
Author(s): Aleksander Waage Sindre Jacobsen Færø	Course: EEL16
	Number of pages including appendices 57
Supervisor at Western Norway University of Applied Sciences Simon Voigt Nesbø	Security classification: Open
Comments: We, the authors, allow the publishing of the report.	

Contracting entity: University of Bergen	Contracting entity's reference:
Kjetil Ullaland +47 957 68 437 Kjetil.Ullaland@uib.no	Shiming Yang +47 55 58 83 28 Shiming.Yang@uib.no

History

Revision	Date	Status	Performed by
1.0	29.05.2019	First complete version	AW and SJF

Abstract

Birkeland Centre for Space Science at UiB is working on a project to study terrestrial gamma-ray flashes further to get a better understanding of the relationship between earth and space. These gamma-ray flashes are the most energetic phenomenon that occurs in our atmosphere. Many uncertainties are surrounding them, and thus, this project came to be. It is the microelectronics group at UiB who is responsible for developing the hardware and firmware. This new design is based on a predecessor but is considerably upgraded.

Throughout this thesis, the details of how the PCB was designed is presented. The given specifications are the starting point of the design. These specifications form a list of what is required of the system or specific components. The design process itself is divided into three stages: choosing the components, creating the schematic, and lastly, designing the layout.

In the first stage, all the components necessary in the design need to be selected either by a given specification or the demand of the system. The two most significant components are the FPGA and the ADCs. Most of the design is based around these. The FPGA was selected by a combination of given requirements and consultation of the theses where the firmware was developed, while the ADC was selected solely by the requirements. The ADC is the biggest upgrade compared to the predecessor. The predecessor has a sampling rate of 50MSPS, while the new design has 500MSPS.

The second stage is where all the components are used to create the needed circuits. Most of what is done at this stage is researching how components external circuitry should be designed and how the components are connected. The more advanced design is the analog amplifier and filter, which is required prior to the ADC to use its range fully and to avoid the aliasing effect. A transimpedance amplifier is used to convert currents from the PMTs to voltages, while also amplifying. A fourth-order low pass filter is used to remove the unwanted frequencies. It is built by a KRC filter and an MFB filter using an ADC driver. The workings of this filter are simulated in OrCAD Capture CIS to be certain of its performance.

After the circuitry is designed, and the connections are made, comes the last stage where the physical design of the board is made. All the components are placed on a virtual board, and a trace connects the pins that are specified to be connected in the schematic. Most of the process is straightforward, but for high-speed components, there are some extra considerations to keep in mind to retain a signals integrity.

Preface

The work for this thesis was performed at the University of Bergen (UiB), between January and June of 2019. The work is part of a larger project initiated by the Birkeland Centre for Space Science (BCSS) in collaboration with the Microelectronics group at UiB. The task at hand is to create the hardware, a circuit board for signal processing, for BCSS. This project builds on two theses done at UiB and the predecessor. The development of the PCB was executed with limited experience with PCB design.

Working on a large-scale and advanced project like this, with an incredibly steep learning curve, has been a great challenge, but with all the knowledge and experience we are left with, we can safely say we are pleased with our choice. Choosing a project that seems out of one's reach in the beginning is the best way to push oneself to learn a lot.

Acknowledgments

Starting from scratch with a project like this was a rough process for two students with little experience within this field. Our contacts at UiB have been beneficial throughout the entire project. Kjetil Ullaland and Shiming Yang have both helped us stay on track and has pushed us in the right direction. The combination of Kjetil always asking the right questions, and Shiming's technical expertise made sure that the ball always kept on rolling. Shiming was quick to respond to all our questions, and in the cases where he didn't know the answer, he would do his best to figure it out. Kjetil provided useful pieces of advice on how to approach the different issues. For all of this, we are immensely grateful.

Our mentor, Simon Voigt Nesbø, has also been a great help throughout the semester. He made sure we were updated with all the upcoming dates and was always ready to answer our questions. We are grateful for having a mentor who was quick to respond and gave thorough and useful feedback.

Lastly, we would like to thank Svein-Atle Engeseth with his help in the analog design. This part of the design was complicated, and thus, getting to discuss details and get feedback from him on the analog design was indeed helpful.

Contents

1 Introduction 2
2 Specifications 3
3 Software 4
3.1 Mentor Xpedition 4
3.1.1 Library Manager 4
3.1.2 Designer 4
3.1.3 Layout and Constraint Manager 5
3.1.4 OrCAD 6
4 Design Process 7
4.1 System Architecture 7
4.2 Choosing components 8
4.2.1 FPGA and its critical components 8
4.2.2 Data acquisition 10
4.2.3 Peripherals 12
4.2.4 Others 13
4.3 Schematic 15
4.3.1 Data acquisition 15
4.3.2 Digital Logic 20
4.4 Layout 28
4.4.1 High-speed considerations 28
4.4.2 Setup 28
4.4.3 Placement 29
4.4.4 Routing 29
5 Conclusion and Outlook 30
5.1 Conclusion 30
5.2 Further work 31
5.2.1 PCB design in layout 31
5.2.2 Practical testing of Data Acquisition Stage 31
6 Appendices 31
Appendix A - Words and abbreviations 31
Appendix B – Schematic 32

1 Introduction

This bachelor's project is part of a grander project by BCSS in collaboration with the UiB. BCSS is a Norwegian Centre of Excellence that focuses on understanding Earth's relationship to space. They have divided this objective into smaller areas of focus because of its immensity. One of these areas is all about studying terrestrial gamma-ray flashes, or TGFs, which is the most energetic phenomena that occur in our atmosphere. There is a general agreement on the physical requirements for the TGFs, although there is much uncertainty around the details of the mechanism. The TGFs appear to be related to lightning and is observed during lightning storms, but why they occur and how they are related to lightning is still uncertain [1].

The ALOT project was initiated by BCSS to get a better understanding of this phenomena. In the ALOT project, also known as Airborne Lightning Observatory for TGFS, sensors will be sent up to the atmosphere by plane to gather information, for which the researchers can base their research. The Microelectronics group at UiB is responsible for designing the hardware and firmware. Two master students developed the firmware in the fall of 2018, and with this bachelor's project, the hardware is near completion. There was a precedent to this project called FEGS, or Fly's Eye GLM simulator, which launched in 2017. The hardware design for ALOT is inspired by the design of FEGS and the architecture presented in the master theses.

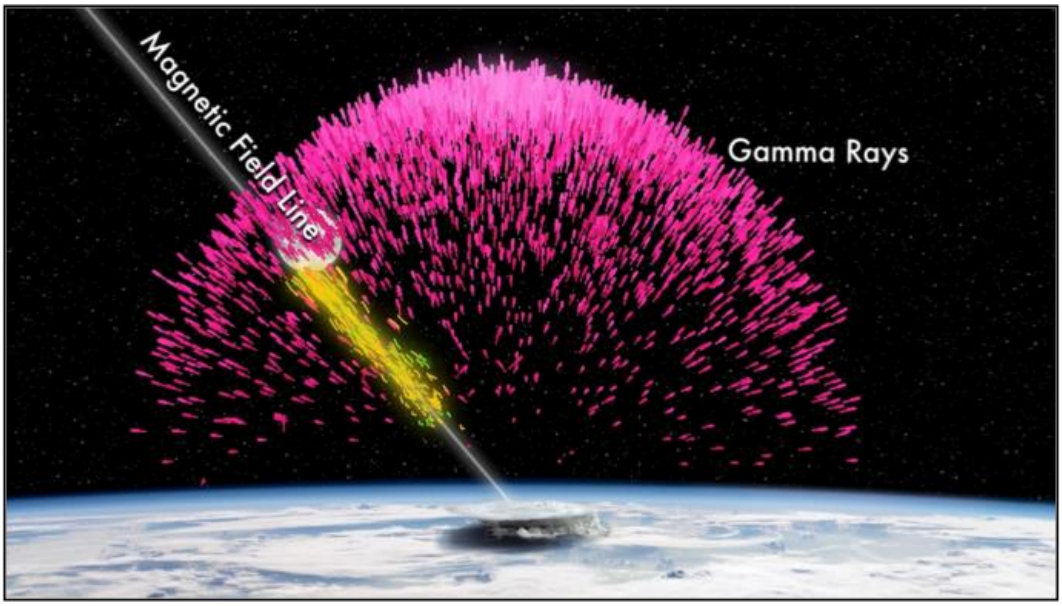


Figure 1: An illustration of a terrestrial gamma-ray flash in the atmosphere. The colors pink, yellow, and green represents gamma-rays, electrons, and positrons, respectively [20]

2 Specifications

In the architecture of the original FEGS system, the different subsystems were spread across multiple circuit boards. In contrast to the FEGS system, the current ALOT project will have an architecture where all the subsystems are implemented on a single circuit board. In addition to these physical changes, other significant modifications are also required by the given specifications; A different type of FPGA, and upgrading the ADCs to higher sampling rates.

The main functionality of the system is to process signals received from external sensors and store the outcome in memory. The unit responsible for processing these signals is FPGA based. The FPGA, in this project, is mounted directly onto the circuit board, as was done on the predecessor, but instead of using the former IGLOO-2 FPGA, an FPGA of the Zynq-7000 series is employed in this version. As mentioned in section 1, the firmware developed for the FPGA has already been completed by two master students in 2017/18. Due to the complexity of the firmware, the limited resources of a Zynq-7010 is not sufficient, and a larger device is required.

The signals received from the sensors are analog and need to be digitized for them to be usable for the FPGA. To achieve this, Analog-to-Digital Converters (ADC) are used. These ADCs are required to have sampling rates of up to 500 MSPS. This is an order of magnitude higher than the ADCs used in the FEGS system, which only ran at 50 MSPS, and will significantly improve the time resolution and frequency range of the sampled signals.

The data processed by the FPGA will be stored on removable flash memory, making it significantly more accessible. The board requires a JTAG interface and Ethernet for a computer to be able to communicate with it; which allows the board to be configured, and makes it possible to monitor it while running.

3 Software

The entirety of this project has been completed by using various tools. A PCB is fully designed using digital tools before the physical board is made. There is plenty of software to choose from to create a PCB, but in this project, the use of Mentor Xpedition was specified. For the analog circuits in this design, the simulation tool OrCAD Capture CIS is used. The next subchapters will describe these tools and explain how they are used.

3.1 Mentor Xpedition

All of the PCB design is done in Mentor Xpeditions software from Siemens, which is a software package with a plethora of tools to assist in the design process. From the multiple software tools included in this package, the ones used the most in this project is Library Manager, Designer, and Layout.

3.1.1 Library Manager

One of the first steps in the design process is usually to select the needed components. Different systems have different requirements for their components, which makes the selection process one of the more important ones. Picking a suitable component can be a challenge, and although it can always be replaced, doing so becomes more and more complicated in the later stages of the design.

Library Manager is where one collects all the different components needed in a project. It can import, export, and edit symbols and cells as needed. A symbol is a simplistic version of a component which is used in the schematic to establish a connection, while the cells are a physical representation of the components used in the layout. Creating a part with both a cell and a symbol is a time-consuming task. Thankfully, there is a more straightforward solution made for Mentor Xpedition. They have developed the tool Partquest, which is a website where one can find a plethora of components that are free to download and import to one's library. If the needed component is not available, one can request it, and within a couple of days, a professional has created both the symbol and cell for that component. This tool removes the need to spend countless hours on creating cells, which requires precise measurements as specified in the datasheet for the component.

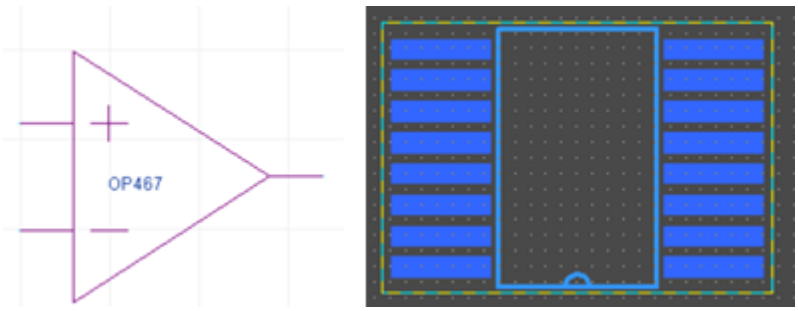


Figure 2: Symbol versus Cell of a dual opAmp IC

3.1.2 Designer

The schematic is created in the Designer software. It is a simple tool without too many functions. The main functionality of the software is to place and connect components. There are some extra functions to assist in this stage, but those are mostly to add shortcuts and to enhance the readability of the

schematic. Most of what is done in this stage comes down to research on how different components should be connected and configured.

There are two ways to connect components in Designer; either by directly connecting the pins or by creating a net which connects them. Creating a net gives the freedom to place the component somewhere else on the sheet or in a completely different sheet. When and where to use the different methods is entirely up to the creator. The goal should be to make a readable schematic, and hence, one would need to use what makes sense in the given situation.

One of the extra functionalities is for FPGA pin assignment and is called I/O optimizer. After the schematic has been created, and the components have been placed in the layout process, I/O optimizer can change where pins are connected to reduce the number of unwanted crossovers. This functionality is possible to use for single connections or for groups at a time, which also makes it possible to change where differential pairs are connected. Reducing the number of crossovers simplifies the routing process that is done in the Layout stage.

3.1.3 Layout and Constraint Manager

It is in Layout that the actual design of the PCB is created. All the components are placed on a virtual board, and a trace is made between the different pins that were specified to be connected in the schematic. This is a process that becomes more and more complex with increasing frequencies and higher densities of components. For a simple circuit, it can be as easy as placing the components and route without needing to consider any particular factors, but in a more advanced system, like the one created in this project, plenty more consideration needs to be taken. These factors are discussed further in Section 4.4.

There are plenty of details at this stage of the design. Just about everything in this process has a physical attribute that can be changed for the specific design: from the width of the traces and the diameter of the via holes, which connects traces in different layers; to the physical size of the board itself. The minimum design parameters are given by the company that will produce the PCB. For this project, it was given that the card would be ordered from Elprint.no. The minimum sizes, shown on their website, can be taken into another tool called Constraint Manager. In Constraint Manager, one can specify minimum, and typical sizes, and other parameters for the entire board, a specific net, or even specific areas on the board. Using this tool helps the designer keep within the design rules without having to change all the sizes manually.

PARAMETER	CHEAPEST		MINIMUM	
	mm	mil	mm	mil
Trace with	0.15	6	0.10	4
Clearance	0.15	6	0.10	4
Annular ring	0.5	20	0.30	12
Solder mask ring	0.2	8	0.15	6
Via pad diameter	1.0	40	0.38	15
Hole diameter	0.5	20	0.15	6

Figure 3: Design parameters from Elprint.no [13]

There are a lot of extra tools to help routing. One could use the Auto-route function to make the software automatically route the entire system, but as the design becomes more complex, the reliability of this tool drops significantly. For high-speed components, like the ADC and RAM ICs in this design, it is preferred to create the traces manually. The software can also assist during manual routing.

When routing a differential pair, it will create both the traces at once and try to keep the traces at the same length. One can also route multiple traces at ones, or even try an auto-route like function to automatically create routes for chosen connections.

3.1.4 OrCAD

OrCAD is a simulation tool that, in this project, is used to support the design of the data acquisition stage. While designing this stage, the tool is used to simulate basic analog circuitry, where only transient- and AC analysis of the circuits are done. The advantage of using such a tool is that it can simulate the inherent characteristics of components, where non-idealities like an OpAmps varying closed-loop gain and a component's noise characteristics are apparent. The tool also gives an excellent visual overview of the system's frequency response where, if necessary, specific actions can be taken to improve the characteristics of the system.

4 Design Process

To go from a list of requirements to a fully finished PCB, one must go through three different processes: Choosing components, designing the schematic, and designing the layout. The given order is how it might go in theory, but in many cases, one alternates between the different ones as one learns something new and need to adapt the design

4.1 System Architecture

Figure 4 illustrates a simple block diagram of the entire system. By considering the task the system is specifically made for, it is clear that the system is divided into three distinct stages: data acquisition, data processing, and data storage. Excluding the obvious components in these stages, there are additional components that are required for optimal functionality in the system.

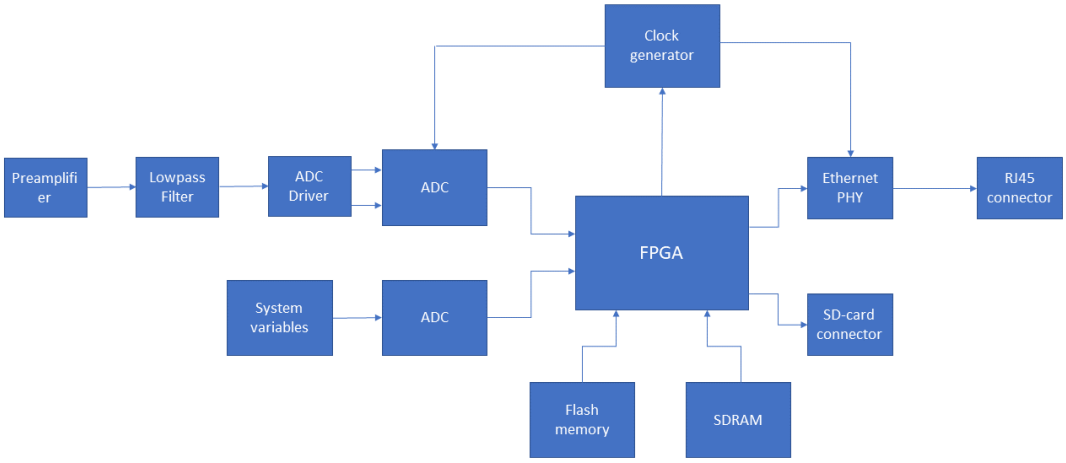


Figure 4: Block diagram for the ALOT system

The data-acquisition stage is where the analog signals, received from external sensors, are converted into digital signals for the FPGA. For this, a preamplifier is needed to amplify the signal to an amplitude the ADC can utilize its entire voltage range. This part consists of a transimpedance amplifier that converts the incoming signal from current to voltage. Before the ADC can sample this voltage, the signal must propagate through a low-pass filter. The Nyquist-Shannon sampling theorem says that the frequency of the signal should not go above half of the sampling frequency to avoid aliasing, and this indicates the starting point of the stopband. The low-pass filter is of fourth order and is made up of two second-order filters; one of which is made using the ADC driver. In addition to being a part of the filter, the ADC driver is responsible for converting the signal from single-ended to differential. Differential signals are the preferred signal type at high frequencies as it reduces the crosstalk between components. The last part of this stage is the ADC, which samples the differential input and converts it into digital signals that are sent to the FPGA using an LVDS format.

After the conversion to digital, the FPGA reads these signals and processes them. If the received signal is above a specified threshold, it is sent to the last stage where it is stored on an SD-card.

There is one additional function in the system, which is the ability to monitor and transmit system information as it is running. The monitoring is done by measuring different variables, like currents and voltages, and then transmitting them through an ethernet connection. This functionality is added to be able to make sure the system is working as it should.

4.2 Choosing components

Choosing components can be a significant challenge. There may be many requirements, thus finding one that fulfills them all can, at times, be difficult. Only a few components in this project had specific requirements. Most of the components had to be chosen by the needs of the system, which, in most cases, gave a high degree of freedom in the selection. These next couple of subchapters discuss and explain the selection of all components.

4.2.1 FPGA and its critical components

The FPGA is the largest and most complex component in this system. It is a key player in the design and configuration of several other components. There is a vast amount of specifications for FPGAs, and different systems have different needs. Which is why it is a good idea to have pointers for what to look for before one start browsing for a suitable one. This system was specified to use a Xilinx Zynq-7000 FPGA, which gives a good starting point.

The project is based on two master theses written at UiB. The students behind these theses gave a good description of the system for which they developed the firmware. They used a Xilinx Digilent Zybo SoC trainer board that contains a Z-7010 Xilinx FPGA. After all their hard work, they concluded that an upgrade for the FPGA would be necessary as the development of the ALOT project continues. Throughout all their testing using the Z-7010, they stumbled upon some problems which were caused by a limited number of LUTs and BRAM cells. This gives the lower limit for the FPGA. The University also specified that they want an FPGA with PCI-express to use at a later time. These two factors give a clear pointer for what to look for.

Figure 5 shows the different choices within the Zynq-7000 family. Taking the requirements into consideration leaves two choices: the Z-7015 or the Z-7030. The 7015 is a modest upgrade with 46200 LUTs and 3.3 Mb of BRAM, while the 7030 is a significant upgrade with 78600 LUTs and 9.3 Mb of BRAM. Figure 6 shows the pin compatibility between different packages and types. The 7015 is only backward compatible with the 7012s, while the 7030 can be forward compatible depending on the packaging. Taking the possibility of a future upgrade into consideration, the 7030 using the FBG676 or FFB676 seems to be the optimal choice.

	Device Name	Z-7007S	Z-7012S	Z-7014S	Z-7010	Z-7015	Z-7020	Z-7030	Z-7035	Z-7045	Z-7100
	Part Number	XC7Z007S	XC7Z012S	XC7Z014S	XC7Z010	XC7Z015	XC7Z020	XC7Z030	XC7Z035	XC7Z045	XC7Z100
Programmable Logic	Xilinx 7 Series Programmable Logic Equivalent	Artix®-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Artix-7 FPGA	Kintex®-7 FPGA	Kintex-7 FPGA	Kintex-7 FPGA	Kintex-7 FPGA
	Programmable Logic Cells	23K	55K	65K	28K	74K	85K	125K	275K	350K	444K
	Look-Up Tables (LUTs)	14,400	34,400	40,600	17,600	46,200	53,200	78,600	171,900	218,600	277,400
	Flip-Flops	28,800	68,800	81,200	35,200	92,400	106,400	157,200	343,800	437,200	554,800
	Block RAM (# 36 Kb Blocks)	1.8 Mb (50)	2.5 Mb (72)	3.8 Mb (107)	2.1 Mb (60)	3.3 Mb (95)	4.9 Mb (140)	9.3 Mb (265)	17.6 Mb (500)	19.2 Mb (545)	26.5 Mb (755)
	DSP Slices (18x25 MACCs)	66	120	170	80	160	220	400	900	900	2,020
	Peak DSP Performance (Symmetric FIR)	73 GMACs	131 GMACs	187 GMACs	100 GMACs	200 GMACs	276 GMACs	593 GMACs	1,334 GMACs	1,334 GMACs	2,622 GMACs
	PCI Express (Root Complex or Endpoint)[3]		Gen2 x4			Gen2 x4		Gen2 x4	Gen2 x8	Gen2 x8	Gen2 x8
	Analog Mixed Signal (AMS) / XADC	2x 12 bit, MSPS ADCs with up to 17 Differential Inputs									
	Security[2]	AES and SHA 256b for Boot Code and Programmable Logic Configuration, Decryption, and Authentication									

Figure 5: A table comparing the different Zynq-7000 FPGA [38]

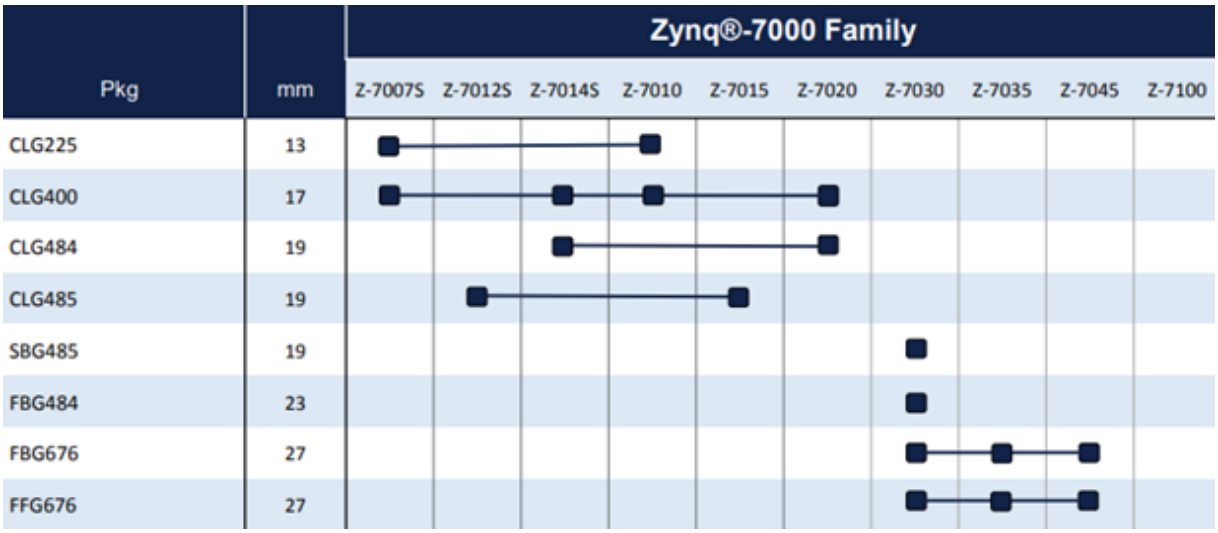


Figure 6: Pin compatibility between the different FPGA types [38]

RAM

The random-access memory is an essential component in a system architecture like this. It is used to store information the FPGA needs to have fast access to. There are different types of RAMs like DRAM and SRAM, which is dynamic and static RAM, separately. These types have different architectures and thus have different needs and specifications. The DRAM has the simplest architecture using only one capacitor and transistor for storing one bit, while the SRAM uses six transistors per bit. Because the DRAM is using a capacitor, which discharges over time, the voltage needs to be refreshed quite often. Because of this refresh process, it is a bit slower than the SRAM, but then again, the DRAM is cheaper than the SRAM as it only needs one transistor per bit. The price difference has made the DRAM the more commonly used RAM.

This system needs RAM to work optimally. There were no specifications to limit this selection. The choice was, therefore, made by looking at other designs using an FPGA from the Zynq-7000 family and choosing one that was confirmed to work with the chosen FPGA. In the end, the MT41K was selected. It is an SDRAM, which is a synchronous DRAM that differs from a regular DRAM by using pipelining to be more efficient. Pipelining is when the RAM sends the following requested information to the processor, in this case, the FPGA, before it has finished processing the previous one.

Flash

The internal memory of the FPGA is volatile, meaning that it does not keep any data when the power is cut. Or in different terms, the FPGA always starts as a blank and must be configured every time it is turned on. The flash is an external non-volatile memory that is needed to store the FPGAs configuration.

There are two factors which need to be taken into consideration for the flash in this system: the memory size and the interface. This memory is only going to store the configuration of the FPGA, and thus, it does not need to be extremely large. The Zynq-7030 has an uncompressed bitstream of about 5.8MB, which is the file containing the programming information. The boot image comes in addition to this bitstream. The size of the boot image is dependent on the chosen system. Using a more advanced system like Linux could require another several MB while using the freeRTOS would require less than 100KB. An external memory chip of 16MB is a reasonable choice as it covers the minimum but also gives the freedom to go for a more advanced option.

The interface is the setup of the connections, in this case, between the FPGA and the memory chip. For storing large amounts of data, a regular SPI interface would be a good choice, but for the boot configuration, this would not be adequate. It is typical to use either a parallel interface, where the data is sent simultaneously on parallel lines; or a Quad-SPI, which is four SPI connections in parallel. There are some benefits by choosing the QSPI; one of which is that it is the fastest configuration solution while still having a low pin count.

The Xilinx team has collected a list with different memory chips that are confirmed to work with the FPGAs in the Zynq-7000 family. The flash chosen for this system comes from that list. The S25FL127S has 16MB of memory and uses a QSPI interface which should make it a good fit.

4.2.2 Data acquisition

The data acquisition stage is composed of three substages; The Preamplifier stage that converts and amplifies the current signals received from the sensors, a fourth order Butterworth low pass filter stage that removes the undesirable signals, and the ADC driver stage that converts the analog signals to digital for the FPGA to process the signals. All of these stages have properties that require specific components: OpAmps, ADC driver, and ADC. These components are discussed in the following subsections.

OpAmps

The data acquisition stage requires two operational amplifiers, OpAmps, to achieve the desired properties. These OpAmps are used in two of the three substages: The Preamplifier stage and the filter stage. Both of the OpAmps should have a bandwidth that is roughly a magnitude higher than the 250 MHz Nyquist frequency. The reason for this relatively high bandwidth is that the closed-loop gain of the OpAmps should be a close approximation to ideal in the region of interest. By making this region ideal, there is no attenuation done by the components, which in turn makes the design of an "ideal" filter easier.

The number of available OpAmps with a bandwidth of 2.5 GHz or higher is somewhat limited, and the basis of this statement is on the bandwidth of the component alone. Two opAmp alternatives that fit the requirements for these applications are the OPA855 and THS4303.

The OPA855 has a bandwidth of 2.5 GHz and gives the closed-loop gain an excellent approximation to ideal in the region of interest. This OpAmp is specified to operate at low-noise, which is ideal for wideband TIA and voltage amplifier applications. Since Butterworth filters are voltage amplifier circuits, the component is a great candidate for both the TIA and filter stages. However, the component is decompensated, meaning it has the disadvantage of only being stable with a gain greater or equal to 7V/V, making it unfit for use in stages with unity-gain.

The THS4303 has a bandwidth of 3.0 GHz and gives the closed-loop unity gain an excellent approximation to ideal in the region of interest. This OpAmp is a wideband operational amplifier that is ideal for high-speed analog signal-processing chains, which perfectly depicts the specification requirements of the data acquisition stage. This component is a necessary candidate in the case of an amplifier being driven as unity-gain.

ADC-Driver

For the ADC to achieve optimum performance, it requires an ADC-driver in front of it. The ADC-driver receives the single-ended signals from the filter stage and converts them into differential signals. The reason for the use of differential signals as opposed to single-ended signals is to reject common-mode noise and to improve the overall performance due to balanced signaling.

The ADC-driver used in this system should have a bandwidth that is approximately equal to the 250 MHz Nyquist frequency. By doing so, the driver has an internal low-pass filter with a bandwidth close to the external filter, reducing the chance of receiving signals with frequencies above the region of interest.

The alternatives for ADC drivers that have the appropriate bandwidth for the application are the THS4504, THS4505, and AD8138. The THS4504 and THS4505 are mostly the same components, excluding the THS5404s power-down capability. Because the power-down functionality is unnecessary in the system, the THS4504 acts as a THS4505 if chosen, thereby making it a reserve candidate in case the stock of THS4505s is empty. These drivers have a bandwidth of 260 MHz while the AD8138 has a bandwidth of 320 MHz. Because the THS4505 has a bandwidth that is relatively close to the frequency of interest, as opposed to the AD8138, it works better as a filter for these frequencies.

While the usage of the THS4505 may work in theory, its low bandwidth may limit other characteristics, like the distortion, when applied in practice. Thus, switching to a 380 MHz driver, THS4503, is a

possibility as they are pin compatible, providing a substitute if the real-life testing shows certain limitations.

ADC

The component the entire project is centered around, the FPGA, requires digital signals as inputs. The signals received from the detectors, propagating through the preamplifier and filter, are analog, which needs to be digitized before reaching the FPGA. This requirement is met by placing an ADC in front of the FPGA.

The predecessor to this project, the FEGS BGO, had an ADC with a sampling frequency of 50 MSPS, while the new project has 500 MSPS. This significant increase provides a magnitude more samples per second, making it possible to collect much more information from the input signals. The disadvantage of using a 500 MSPS ADC is that there are rather few to choose from, and there are scarcely any OpAmps with a bandwidth large enough to provide sufficient properties to support the system.

Two possible alternatives with 500 MSPS are ADS5404 and ADS5407. There is very little that separates these components in terms of functionality, only in the amount of noise and other electrical characteristics. The choice to go with the provided component recommendation was made, and as such, the ADS5404 is the systems ADC.

4.2.3 Peripherals

In addition to the main components described in the sections above, there are also some peripherals that give the system additional functionalities. These are described in the sections below.

Ethernet

This system is using ethernet for the ability to monitor it remotely. There are three layers needed to be able to use ethernet: the media access controller, the physical layer, and a connector. The only given requirement for this is that it does not need a higher speed than 100Mbps.

The media access controller, also known as MAC, is the layer which either receives data that it converts into packets that can be sent onwards or vice versa. The chosen FPGA has this layer embedded in its architecture. This simplifies the process quite a lot, but it also gives some limitations. The embedded MAC uses an RGMII interface which limits the IO voltage of the physical layer.

The physical layer, or ethernet PHY, is the second layer in this three-part configuration. This component receives the packages from the MAC and converts them into signals which can be sent to the transmission lines, or vice versa. There are quite a few components that fulfill the given requirement. The one that was chosen for this project was the DP83822 because of its configurability and it being well documented. This component can be configured to use any combination of the typical voltages for IOs and power supply, and it supports the RGMII interface while only having speeds up to 100Mbps.

The last stage is the connector. The RJ45 connector is the standard when it comes to ethernet, but even this has some requirements. There should be a layer of magnetics between the PHY and the connector to isolate the transmission lines from faulty currents or voltages. Options with integrated magnetics and LEDs are possible to simplify this. The PHY also sends out signals to LEDs that signify

that everything is working as it should. The ARJC01-111002T from Abracon was chosen for this project as it fulfilled all of the needs.

SD Card

Scientific data collected by the FPGA is stored on an SD-card. For this, the system requires an SD controller and connector. The former is also one of the embedded peripherals in the FPGA, which means that the only needed part is the SD-card connector. Because there are no special requirements for this component, a standard mass-produced connector was chosen. The 693161011911 is a readily available connector suitable for this system.

Monitoring

A combination of components and circuits gives the functionality of monitoring the system. This stage has been copied from the FEGS and modified to fit this new system. An external ADC samples all the voltages and the primary current from the +5V power supply. The AD7490 has a sampling rate of 1MSPS and a 12-bit resolution, which is more than enough for this application. It can also be configured to have a maximum input voltage of 5V, which is a perfect fit as the highest voltage is also 5V in this system.

A current sensing circuit is used to get a voltage that is proportional to the current. The INA190 has a shunt resistor connected between the input pins. Everything that uses 5V is connected after this resistor, which, in turn, means that the primary current goes through the resistor and creates a small voltage difference. This difference is amplified by the instrumentation amplifier to create a readable voltage for the ADC. In FEGS the INA826 was used, but this system does not use the 12V power supply needed to get within the boundaries of its VCM range. Thus, the INA190, with a much more extensive range, was selected.

4.2.4 Others

The last couple of components needed in the system are just as important as the other components. These are used to create clock pulses for components that require one and to give power to the entire system.

Clock Generator

A big system like this, with many different components, needs multiple frequencies and thus, multiple clocks. The system has two ADC ICs requiring an LVDS clock signal of up to 500 MHz, and a 25 MHz CMOS signal for the Ethernet PHY. Instead of having a crystal or crystal oscillator for each of these, a clock generator is a good replacement. The clock generator is a component that can use PLL logic and an integrated VCO to generate a specific frequency. This functionality replaces the need for three extra crystals with only one crystal and one IC while also giving the possibility of configuring the clock frequencies as needed.

The AD9577 was chosen for this task. It has four differential outputs which could be configured as individual single-ended outputs and can generate a clock signal up to 637.5 MHz. It also supports signal formats like LVDS, which the ADC clocks require, and CMOS which the Ethernet PHY needs. The I^2C protocol is used to program the chip, which also makes this a good fit with the FPGA as it has an embedded I^2C controller.

Clocks

The system requires two crystals, in addition to the clock generator. For the FPGA to be able to configure the clock generator, it needs a clock signal which the clock generator will not be able to generate before it has been configured. The clock generator itself also needs a crystal, but at a different frequency than the FPGA. For the FPGA the ABM7 crystal oscillator which generates a CMOS signal with a frequency of 50 MHz was chosen, and for the clock generator, the CW541 crystal was chosen. The crystal oscillator is a crystal with additional components to generate a precise frequency. It is more work to handle a barebone crystal, but it gives the possibility of optimizing it for ones need. In this case, the AD9577 has a built-in oscillator circuit and therefore only requires a crystal.

Voltage translator

A frequent problem when having multiple peripherals embedded into an FPGA bank is that different components might require different IO voltages. This mix up occurred in the bank with the Ethernet MAC and SD-controller. Using a voltage translator is an easy fix for such a problem. A voltage translator, like the chosen MAX13030E, is a component that receives digital signals at a certain voltage level and transforms them into digital signals at a different level. The aforementioned translator is specifically created to be used between an SD-controller and its connector. Because the PHY needs a voltage level of 1.8V and the SD interface has the least amount of connections, it was obvious that the latter should use the voltage translator.

Power Supplies

Choosing the appropriate power supplies is crucial for the components to function. The task of choosing these supplies is the last thing that must be done when designing a circuit. This is because different components require different supply voltages and consume a different amount of power. A power supply can only provide a certain amount of current with a specific voltage at its output, and it is therefore required to know the amount of current each power supply must be able to deliver. To do so, find every required supply voltage, and cumulate all the currents each component requires from its respective supply voltage.

Based on the components mentioned in the subsections above, there are a variety of supply voltages: ±5V, 3.3V, ±2.5V, 1.8V, 1.35V, and 1.0V. There are also a couple of reference voltages that are applied: 2.5V and 0.9V. The ±5V comes from the THM 10-2421W, which is a DC/DC converter. These components are genuinely better at stepping down voltages but introduce quite a lot of noise. Both the 3.3V and 1.8V come from two separate MIC29302, which is a Low-Dropout (LDO), adjustable voltage regulator. This component can deliver enormous amounts of current (up to 3A), and the voltage output can be adjusted by merely changing the resistor values. The LP38512MR-ADJ is another LDO adjustable voltage regulator, where both the 1.35V and 1.0V require one each. This component can deliver up to half the current of MIC29302, which is still above the necessary current output for these voltages. The voltages +2.5V and -2.5V uses the LDO voltage regulators TPS73025 and TPS72325, respectively. These regulators have fixed voltages, meaning no voltage adjustment with resistors is necessary. The currents these components deliver are rather low, where the maximum output current is 200mA. The 2.5V reference voltage uses the same regulator as the 2.5V supply since the regulator has a rather good accuracy, which is more than enough for this reference. The 0.9V reference, on the other hand, uses an ISL21080DIH309Z-TK voltage reference IC. This components accuracy is superior to LDO voltage regulators, which is required for its usage in the system.

The last power supply is dedicated to RAM. The TPS51200 is a sink and source DDR termination regulator that is specifically developed to uphold the needs of RAM components. It provides both the reference voltage for the RAM ICs and FPGA and the sink voltage for the termination resistors. For a crucial component like the RAM, these voltages are required to be stable, and thus a specialized component is used.

4.3 Schematic

The second stage, as mentioned, is where the schematic is created. At this stage, all of the chosen components are connected as required by their specifications and how they are used. There are a lot of components to be connected in this system. Most components have been, more or less, designed around the FPGA, except for the analog data-acquisition stage, which requires a more advanced design procedure.

4.3.1 Data acquisition

As mentioned in section 4.2.2, the data acquisition stage is composed of three substages; Preamplifier-, filter- and ADC driver stage, shown in Figure 7, with the substages placed, respectively.

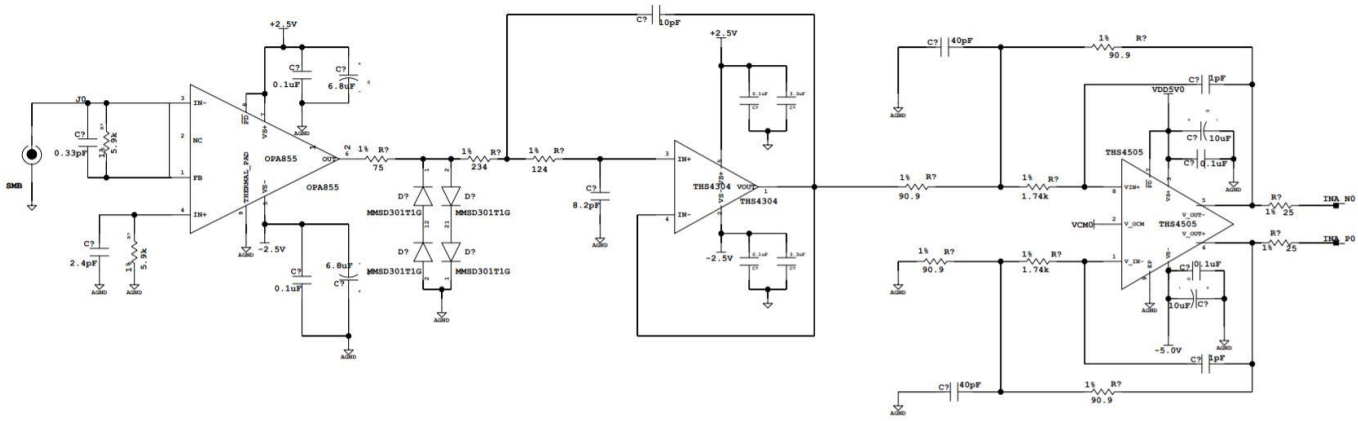


Figure 7: Amplifier, filter, and driver placed respectively from left to right.

One of the first considerations in the design of this stage is the order of the substages. By placing the preamplifier stage before the filter, the circuit achieves the best SNR. However, for this placement to be sensible, the undesirable signals from the PMTs are required to be a great deal smaller than the desired signal. The reason this condition must be met is that if large noise signals are amplified, the signal can saturate. Because the system uses scientific PMTs for the detector stage, the SNR at the output of these components is of high quality. This means that the desired signal is significantly larger than the undesirable signal, thus positioning the preamplifier first is sensible for this design.

In the following subchapters, the different substages of the data acquisition stage are explained in detail from left to right, as shown in the figure above.

Preamplifier stage

The preamplifier circuit consists of a transimpedance amplifier (TIA) that converts the incoming signals, sent by the PMTs, from currents to voltages. The very same circuit is also responsible for the amplification required to utilize the ADC resolution fully.

As mentioned in section 4.2.2, the OPA855 is suited for applications where TIAs are involved. Another prescribed application of the component is as a PMT post amplifier, making it adequate for the preamplifier circuit.

The TIA stage requires a feedback resistor that sets the gain of the circuit and converts the signals from currents to voltages. However, by placing the feedback resistor, the resistor and the total input capacitance, that is, the PMT capacitance, the input capacitance of the OPA855, and any stray capacitance form a zero in the noise gain that results in instability at higher frequencies. By placing a capacitor in parallel with the feedback resistor, a pole is formed that counteracts the zero, preventing instability from occurring. Due to the OPA855 being decompensated, it requires a gain greater than or equal to 7 to be stable. This stability can be achieved by setting the noise gain greater than or equal to 7, that is, setting $N_G = 1 + \frac{C_S}{C_F} \geq 7$, where C_S is the total input capacitance, and C_F the feedback capacitor. This equation shows that to achieve stability, $C_s \geq 6 \times C_F$. A more thorough explanation on the stability analysis of transimpedance circuits is given in [2].

Because there are certain non-idealities in the closed-loop gain of the active elements in the data acquisition stage, specific peaks, as shown in Figure 9 in the simulation subchapter, is apparent in the system's closed loop gain. These peaks are not critical, as they do not harm the system, but if not removed, the output signal experiences sudden amplification spikes in the frequency band where these peaks occur. By selecting a proper cut-off frequency on the low pass filter that is present due to the feedback resistor and capacitor, the peaks are removed.

At the output of the TIA stage, there are a few diodes that form a diode clipper. The diodes used, MMSD301T1G, have the properties necessary for usage in high-frequency applications, making them suitable for the task. These diodes have a forward voltage that is approximately 0.4V, while the ADC can utilize an amplitude of 0.6V. By placing two diodes in series, the clipping appears at 0.8V, taking advantage of the entire input range of the ADC, while protecting from voltage spikes beyond this value that may harm the components.

Filter stage

The system requires a low-pass filter, which is designed as an active fourth order filter. The filter is designed with the properties of a Butterworth filter, which, in this case, is made up of two-second order filters: a Sallen-Key filter [3], also known as KRC filter, and a Multiple feedback [4], abbreviated as MFB, filter. Here, an OpAmp, THS4304, is used to design the second order KRC filter that is placed in front of the ADC driver, THS4505, which is equipped with a second order MFB. The reason for using a THS4304 on the KRC filter as opposed to the OPA855 is because the OPA855 becomes unstable when the gain is less than 7V/V, whereas the THS4304 works best in unity-gain. Since all the gain in the data acquisition stage is placed at the preamplifier stage, the filter stage is unity gain, making the OPA855 unfit for use.

The Nyquist theorem states that signals above half the sampling frequency should not be sampled. This means that because the system samples with 500 MSPS, the most significant signal frequency that should be sampled is 250 MHz. To avoid over- or under-sampling, the filter requires a stopband that occurs at around 200 to 250 MHz. To get a stopband of this magnitude, the filter' cutoff frequency has to be relatively much lower based on the fact that a fourth order filter gives a gain roll-off of 80 dB/decade. To let practically no unwanted signaling get through in the stopband, a relatively low gain is required in this region. A rule-of-thumb is to have a maximum gain of -70 dB for it to be close to an ideal stopband. With a roll-off of 80 dB/decade and approximately -70 dB stopband gain, the cut-off frequency is around a decade lower than the stopband frequency. By a few calculations in Matlab [5], a cut-off frequency of 50 MHz gives a stopband starting a several MHz above 200 MHz.

ADC driver stage

The PMTs are sending out current pulses that travel through a single-ended system, which makes it essential to use the ADC driver' inputs in single-ended mode. To achieve single-ended mode, all that is required is to ground the inverting input channel while applying the signals to the non-inverting input. However, by doing so, it is important to be aware of the driver's supply rails. If the driver receives signals that make the output voltage go above or below the supply rails, it is essential to offset the grounded input with a voltage that is sufficiently large in the opposite direction of the larger output voltage; otherwise, clipping is achieved. If the output is above the supply rail in both directions, it is necessary to either have a more abundant supply or reduce the signal size.

The external circuitry connected to the ADC driver is a second order low pass MFB filter. The structure of an MFB filter circuit placed on a differential amplifier [6] is similar to that of an MFB filter on a basic single-output OpAmp. The difference being that instead of a single MFB from the output to inverting input, there are two identical MFBs from non-inverting output to inverting input and from inverting output to the non-inverting input, as shown in SEQ Figure * ARABIC 8. When calculating and choosing the appropriate component values for the MFBs, it is only necessary to look at one of the MFBs and mirror the values on the other.

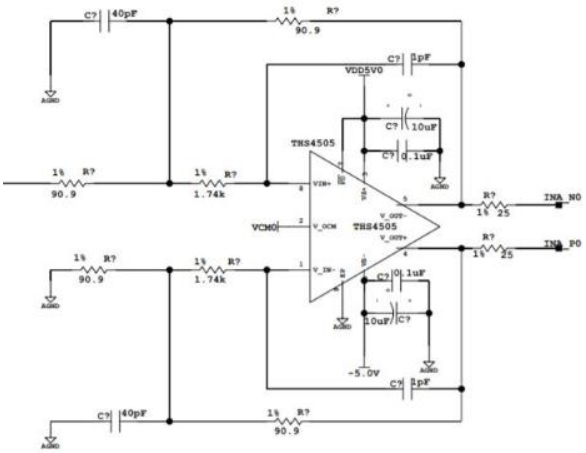


Figure 8: MFB filter placed externally on the ADC driver.

Simulation

Simulation is a way of estimating a circuit's characteristics but does not necessarily predict its detailed behavior. The simulation done in OrCAD required some improvisation as the OPA855 did not have a PSpice model. Instead, THS4304 was used to simulate an OPA855. The frequency response and the noise characteristics of THS4304 are worse than OPA855s, making it a worst-case scenario.

As mentioned in the preamplifier subchapter, the circuit experiences certain amplification peaks in the closed-loop gain due to imperfections in the active elements. In Figure 9, which shows the circuit's closed-loop gain, this statement becomes apparent. At approximately 100 MHz, a zero is located due to inherent characteristics of the THS4304. Looking at the phase plot of the system in Figure 11, a sudden deceleration in phase roll-off is apparent, meaning a zero is located at the bend. The immediate peaking due to this zero form signal spikes at regions where the gain that ordinarily declines as frequency increases, suddenly rises. By placing the pole of the TIA stage in advance of the peak, which is approximately the same frequency as the zero, a pole-zero cancellation happens, and the sudden rise fades away, as shown in Figure 10. However, because of this pole placement, the size of the feedback capacitor may compromise the OPA855s stability due to the total input capacitance not being sufficiently large. To make sure the input capacitance is large enough; an amply sized capacitor is placed in parallel with the OPA855s input to guarantee that $C_S \geq 6 \times C_F$.

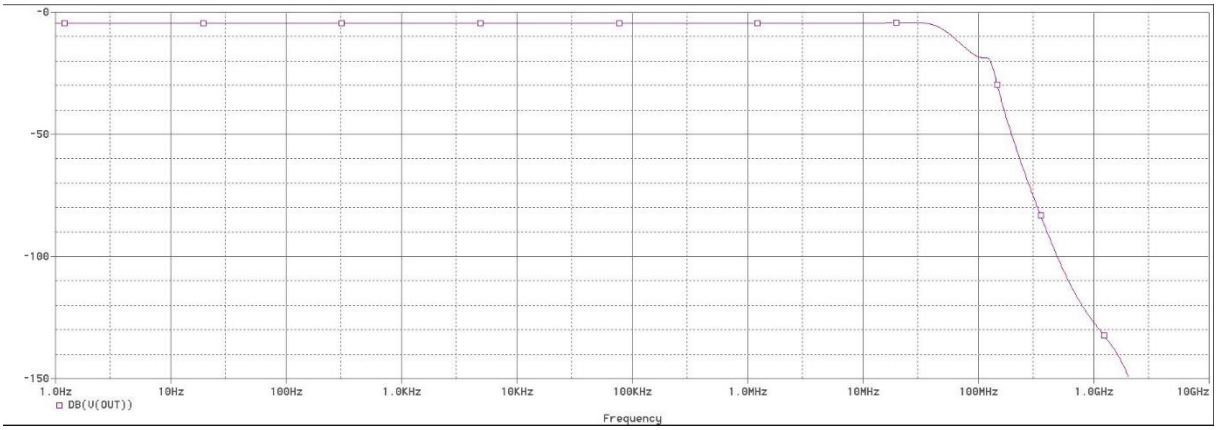


Figure 9: The system's closed-loop gain without feedback capacitor at the TIA stage.

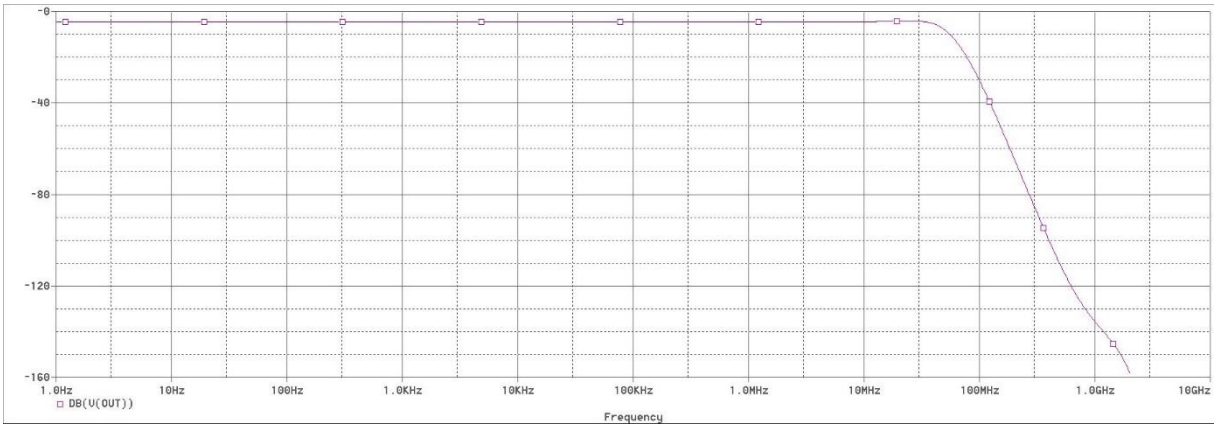


Figure 10: The system's closed-loop gain with feedback capacitor at the TIA stage, with its pole in advance of the peak.

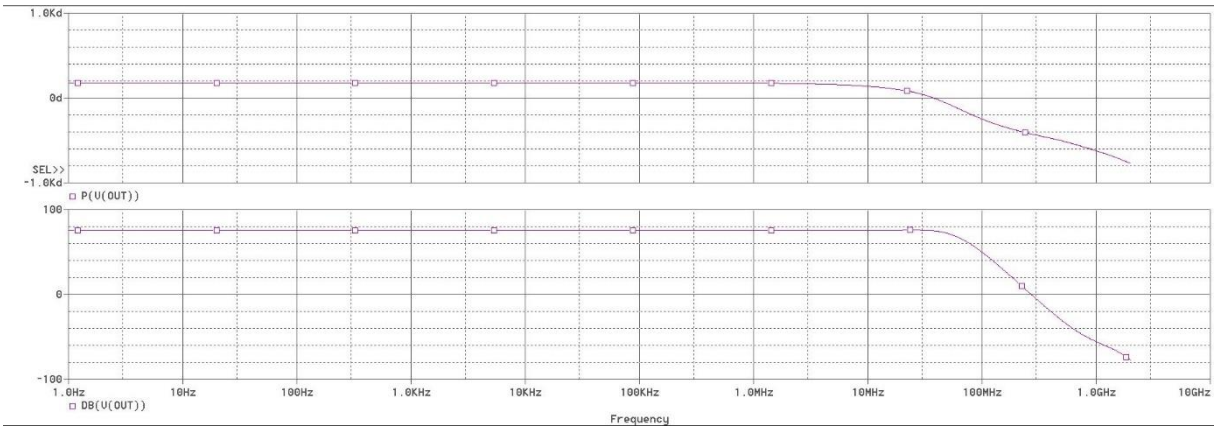


Figure 11: Simulation showing the systems bode and phase plot.

Even though the circuit may have a working bode plot, the system can still be unstable. Due to not being able to use OPA855 in the simulation, only general stability can be simulated, and not whether instability caused by the lack of noise gain presents itself. In Figure 12, a step response simulation is performed to analyze if the circuit is stable. In the graph, a sudden peak, or overshoot, at the transition edges is apparent. If this peak either oscillates indefinitely or becomes amplified as it continues to oscillate, the system is unstable. While there is an overshoot at the transition, it is quickly attenuated due to a high damping ratio [7] that rapidly removes the oscillation from the signal. Due to this rapid oscillation removal, the system quickly stabilizes itself, meaning it is a stable system.

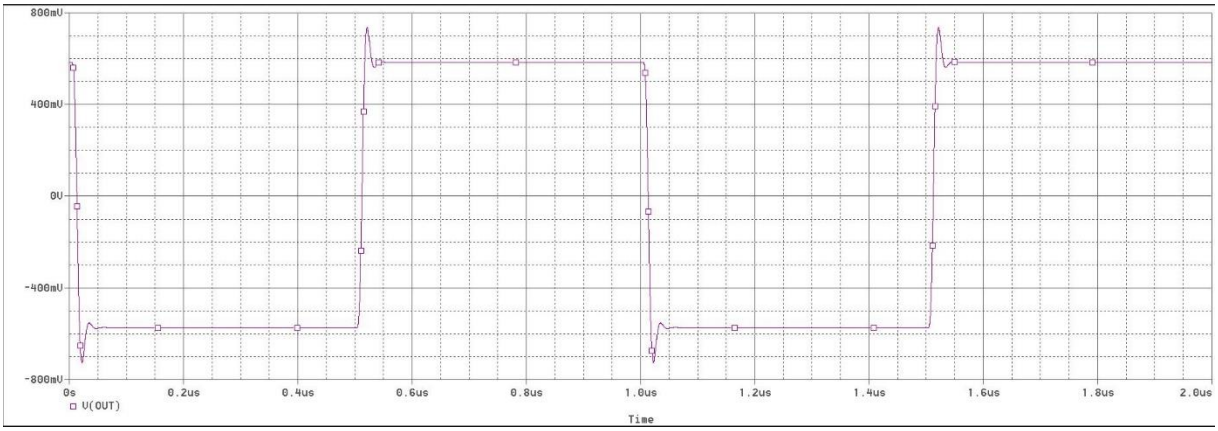


Figure 12: Simulation using step response to analyze the stability of the data acquisition stage

To be convinced that the circuit manages to maintain the signal integrity, a simulation using 1 kHz triangular pulses is performed, shown in Figure 13. This simulation is only to show that the circuit does not distort the signal at lower frequencies, with emphasis on low frequencies because the component models do not give a good prediction of the components' distortion.

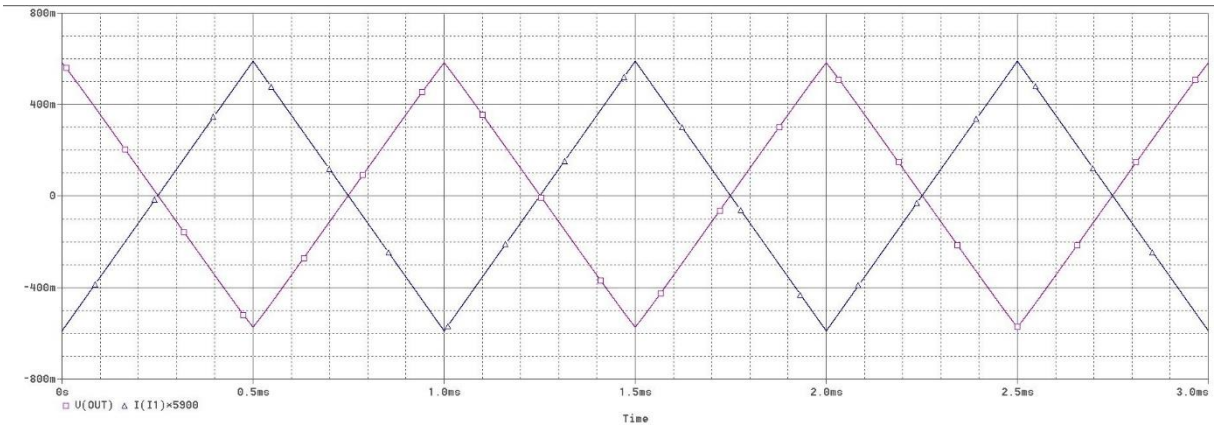


Figure 13: Simulation using triangular pulses, presenting the circuit maintaining the signal's integrity as the signal propagates from source to output.

4.3.2 Digital Logic

The FPGA is the most complex component in this project with its 676 pins, built-in peripherals, various dedicated pins, and so on. Finding which connections are needed was deduced by using a combination of the documentation for the Zynq-7000 family and other designs using a similar FPGA. As with many other components with a large number of pins, the FPGA is divided into multiple symbols. There are 13 symbols for the 676 FPGA: one for the high-speed transceiver, four for regular IOs, one for the internal ADC, two for MIOs, one dedicated for external RAM, one for configuration, and the three last are respectively ground, JTAG, and power.

RAM

One of the components required to make the system run optimally is the external RAM. The FPGA has a dedicated symbol just for this, which is symbol 8 shown on page 3 of the schematic. This symbol has all the pins needed to be able to connect to the RAM. The three primary connections are the data, bank address, and address, which is represented on the symbol by DQ, BA, and A. The bank address and address are used to allocate the data to different places in the RAM. **Error! Reference source not found.** illustrates how the RAM is divided into different banks, which again consists of many addresses where data can be saved. In addition to these three primary connections, there are also quite a few other vital connections. There is on-die termination, or ODT, which gives the ability to enable or disable

the built-in termination resistors on certain pins. Termination resistors are only wanted when the device is receiving data and not when it is sending data, and hence the ODT pin.

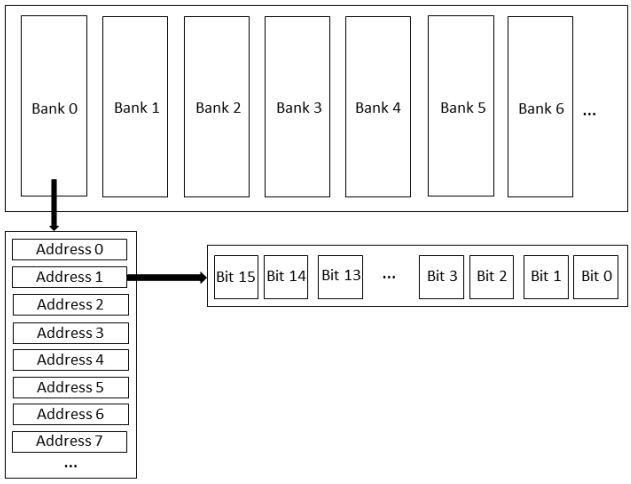


Figure 14: A simple representation of memory division in a RAM

There are two differential DQS pins on the FPGA: the UDQS and LDQS. DQS is a differential data strobe which is needed with high data rates. This system needs both the upper and lower DQS signal because it has a 32-bit data width with two RAM chips connected in parallel. In a regular synchronous system, the data is captured with a common system clock, but with higher data rates the valid data window becomes more susceptible to shifts relative to the clock due to changes of different variables like temperature. Because of this, the DQS signal is aligned with the data signals so that the data is captured by the DQS signals instead of the clock.

DM is the data masking pins. These give the ability to prevent writing to some of the bits, which can be useful if only part of the word in the RAM needs to be changed. Without this, the system would have to read the word, modify it, and then write it back to the RAM. In addition to all of these pins, there are also the general chip select, or CS, and write enable, WE, which is needed in every process using the RAM.

It is a simple process to connect the MT41K RAM to the FPGA, as both symbols use similar names for pins that should be connected. It was decided that the project will have a 32-bit data width, which is the reason why there are two of these components on page 7 of the schematic. The MT41K256M16 is a DDR3L SDRAM with a size of 4Gb, and thus by using two of them, the system has 1GB of RAM.

Flash

Another important component, which is needed to store the configuration of the device, is the flash memory that can be found on page 7 of the schematic. The chosen component is using a Quad-SPI interface to connect to symbol 6 of the FPGA on page 3 of the schematic. Because there are plenty of different interfaces for this component, the FPGA must be configured respectively to the chosen component. The 7Z030 utilizes bootstrapping pins on the same bank as the flash is connected to for this configuration. This means that at startup if the specific bootstrapping pins have a voltage over a threshold, it will be read as '1' and if it is below it is read as a '0'. In this case, using a pull-up resistor gives a '1' and using a pull-down gives a '0'. On this symbol, it is the MIO pins 2-8 which are used as bootstrapping pins. **Error! Reference source not found.** shows a table of the possible configurations. This system has been configured to use cascade mode for JTAG chain routing, Quad-SPI for the boot device, the PLLs are enabled, symbol 0 uses a voltage level of 1.8, and bank 1 uses a voltage level of 3.3V. The bootstrapping pins are multi-function pins, which is why some of them are also connected to the flash. These connections were specified in the technical reference [8] for the FPGA on page 365.

Pin-signal / Mode	MIO[8]	MIO[7]	MIO[6]	MIO[5]	MIO[4]	MIO[3]	MIO[2]
	VMODE[1]	VMODE[0]	BOOT_MODE[4]	BOOT_MODE[0]	BOOT_MODE[2]	BOOT_MODE[1]	BOOT_MODE[3]
Boot Devices							
JTAG Boot Mode; cascaded is most common[1]				0	0	0	**JTAG Chain Routing**[2] 0: Cascade mode 1: Independent mode
NOR Boot[3]				0	0	1	
NAND				0	1	0	
Quad-SPI[3]				1	0	0	
SD Card				1	1	0	
Mode for all 3 PLLs							
PLL Enabled			0	Hardware waits for PLL to lock, then executes BootROM.			
PLL Bypassed			1	Allows for a wide PS_CLK frequency range.			
MIO Bank Voltage[4]							
	Bank 1	**Bank 0**	Voltage Bank 0 includes MIO pins 0 thru 15. Voltage Bank 1 includes MIO pins 16 thru 53.				
2.5 V, 3.3 V	0	0					
1.8 V	1	1					

Figure 15: Possible configurations for the FPGA [8]

Clock generator

The symbol used for the flash also has the potentiality of using other embedded peripherals, like an I^2C controller. I^2C is a communication protocol that only requires two lines: one bidirectional for data, and one for the clock. Because the controller is already embedded in the FPGA, it will simplify the process of using this protocol. It is the clock generator that requires the I^2C protocol for it to be programmed. The clock generator, AD9577, can be found on page 8 of the schematic. This is where the two LVDS clock signals for the ADCs and the CMOS clock signal for the Ethernet PHY are generated. This component uses a power supply of 3.3V for both analog and digital, which, in turn, also means the voltage standard for the digital IO signals are 3.3V. LVDS is a standard with a set voltage level which the AD9577 conforms to, but the CMOS signals generated by it is controlled by the digital power supply, and will, therefore, have a voltage level of 3.3V. The Ethernet PHY uses a 1.8V power supply for its digital pins. Consequently, to be able to use the clock generator as the clock source for the ethernet PHY the voltage needs to be reduced, which, in this case, is as simple as using a voltage divider.

In addition to the main pins, there are also a few extra; all the pins that start with VS are connected straight to the power supply while the additional functions MAX_BW, MARGIN, and SSCG is not used and can be left floating as they have internal pulldown resistors. Figure 16 shows an example configuration for this component with all the needed capacitors.

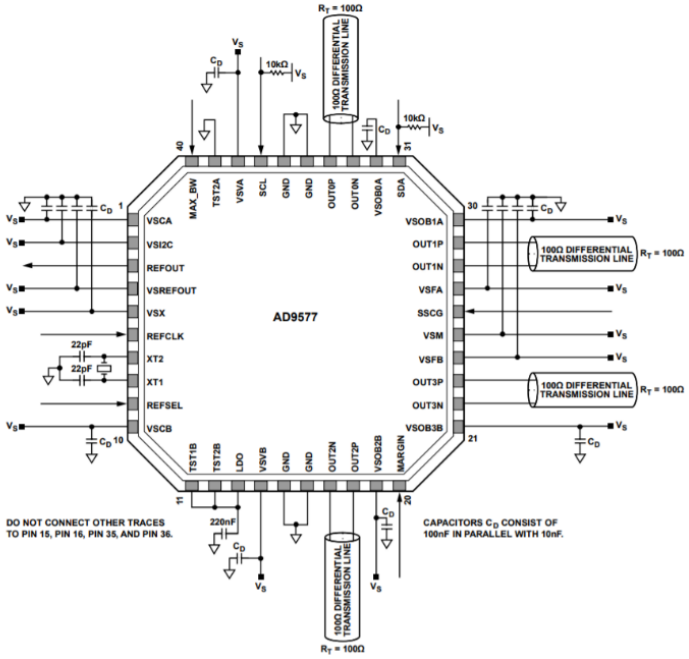


Figure 16: Example circuit of the clock generator [25]

Ethernet

Symbol 6 of the FPGA, discussed in the last couple of paragraphs, is one of the two MIO banks with embedded peripherals. The other one, symbol 7, can be found on the same page. This symbol is configured by the bootstrapping pins on symbol 6 to use a voltage level of 1.8V because the ethernet RGMII does not support an LVCMOS level of 3.3V. Like the other MIO symbol, this one also has multiple embedded peripherals available. One of these peripherals is the Ethernet MAC, which means that the Ethernet PHY is connected to this symbol. The PHY and the RJ-45 connector can be found on page 8 of the schematic. Connecting the PHY to the MAC is a simple process, just like most of the other peripherals which are connected to an embedded peripheral in the FPGA. The connections made between the PHY and the MAC is specified in the technical reference for the FPGA. In addition to making these connections, the PHY itself also requires configuration. The DP83822 uses bootstrapping pins for this, but, unlike the FPGA, it has four levels to choose from. In the datasheet on page 47, one can find a table with recommended values for the resistors, which gives a voltage level on the pin to make the specific configuration. The only setting that needs to be changed in this system is to enable the RGMII. RX_ER is the bootstrapping pin for this setting. To enable this, the RX_ER register needs to be configured for mode 3, which is done by using a voltage divider with a 6.2k and 1.96k resistor. The other default settings should be adequate for this system. By default, the LED_0 pin will be high when the link is good, but because the IOs are configured to have a voltage level of 1.8V, it will not be able to provide enough power to turn on a LED. The NDS331N is a simple MOSFET transistor that has been placed between the LED and the PHY to act as a switch. The source is connected to ground while the drain is connected to the diode in series with a resistor and a voltage source of 3.3V. Once the PHY enables the LED_0 pin, the current can flow through the transistor and turn on the LED integrated into the RJ45 connector. This connector has both LEDs and magnetics embedded in its packaging. Figure 17 illustrates how the connector with all its embedded functions is built. The Ethernet PHY is connected to the connector by the four connections: TD_M, TD_P, RD_M, and RD_P. It is on these lines data

propagates between the components. The last connections, the center taps, are connected to the same power supply as the PHY, which is 1.8V.

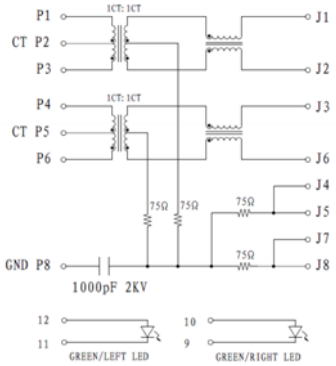


Figure 17: A schematic of the RJ-45 connector [27]

ADCs

Symbol 2 through 6 of the FPGA are dedicated for digital I/O pins and can be found on page 4. There is a total of 242 I/Os, where most of them are capable of either differential or single-ended signaling. The two first pins in bank two and three are, however, dedicated as single-ended. Many of the IOs are multipurpose pins and can thus be configured as either regular IOs or with different capabilities. There are MRCC and SRCC pins, which are multi-region clock capable and single-region clock capable pins and should be used for clock signals; VREF pins can set the threshold for the signaling standards that need one, and the last is the DQS which are strobe pins. Most of the I/O pins on symbol 4-6 are connected to the main ADCs. Where a differential pair from the ADC is connected on the FPGA was decided in the routing process to reduce crossovers and layers used.

All the I/O symbols are split into two different groups: high performance, HP; and high range, HR. The HR symbols, no. 2 and 3, support a wide range of IO standards with voltages up to 3.3V, while the HP symbols, no. 4, 5, and 6, are made to meet the performance requirements of high-speed interfaces with voltages up to 1.8V. These factors are important to keep in mind if the pins transmit a signal. Most of the IO pins used in this system are inputs using the high-speed LVDS standard, and thus the HP symbols are used. Almost all the pins on symbol 4-6 are used as inputs from the ADCs. This system uses four ADCs, which in this case means two dual ADC ICs. The ADS5404 uses a DDR LVDS output interface for its data. Using an LVDS standard gives the system a better common mode voltage tolerance, but it also requires double the number of pins as single-ended standards. About 120 pins connect the four ADCs to the FPGA. At this point in the design, it doesn't matter too much where the pins are connected, only that the differential pairs on ADCs have to be connected to the same differential pair on the FPGA. After the schematic and placement of components are done, the connections will be changed to make the best routes in Layout.

GPS and Monitoring

Two more components are using IOs, in addition to the main ADCs. The Copernicus II GPS module and the extra ADC for monitoring. The design of these circuits has been copied from the older FEGS system and connect to symbol no. 2 on page 4. The extra ADC, samples all the voltages and the main current

from the 5V supply. The current sensing circuit has also been copied from the FEGS but modified to fit with the new instrumentation amplifier.

JTAG

The last couple of symbols are for the JTAG, the extra configuration and administrative tasks, and for power. Symbol 12 is for JTAG with its four pins. The JTAG is used to program the system. There are standard connectors, and this system is using the Xilinx 14-pin interface. The connections for this connector are standardized to fit with the already created cables.

Power

There are two power banks for the FPGA: one for all the voltages and one for ground. What voltage is needed on the different pins is dependent on what is used in the FPGA. The FPGA in this system requires multiple voltage values on its power bank: 3.3V, 1.8V, 1.35V, and 1.0V. All of which are generated by the power supplies on page 13 of the schematic.

To achieve these voltages, certain ICs have to be used to convert one voltage to different voltage values. These ICs are called Low Dropout (LDO) adjustable voltage regulators and gives the possibility of stepping down a voltage - in this case, 5V - to a lower voltage. The different types of LDO adjustable voltage regulator ICs that are used to achieve these voltages are mentioned in section 4.2.4.

When designing the 3.3V and 1.8V supplies, specific resistor values, R1 and R2, has to be calculated to achieve the desired voltages. The required formula to calculate these resistors, and the power supply's circuit diagram is given in the MIC29302 datasheet and shown below. Values for the capacitors are also given in detail in the datasheet.

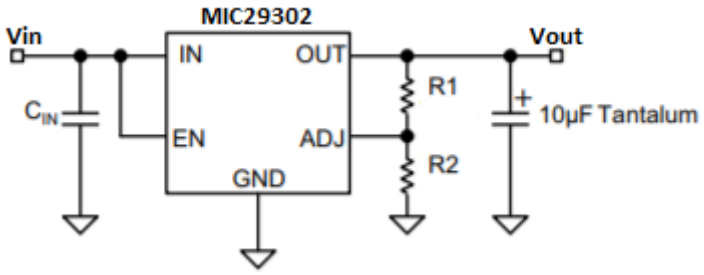


Figure 18: Circuit diagram to MIC29302, showing the required components for proper functionality.

$$\boldsymbol{R1 = R2 \times (\frac{V_{out}}{1.240} - 1)}$$

The design procedure for the 1.35V and 1.0V supplies are similar to that of 3.3V and 1.8V, the difference being that these voltages are using a different component - the LP38512-ADJ, - which uses a different reference voltage for calculating the value of R1. The information about these calculations is given in the datasheet for the specified component being used. The circuit diagram of the LP38512-ADJ is shown below.

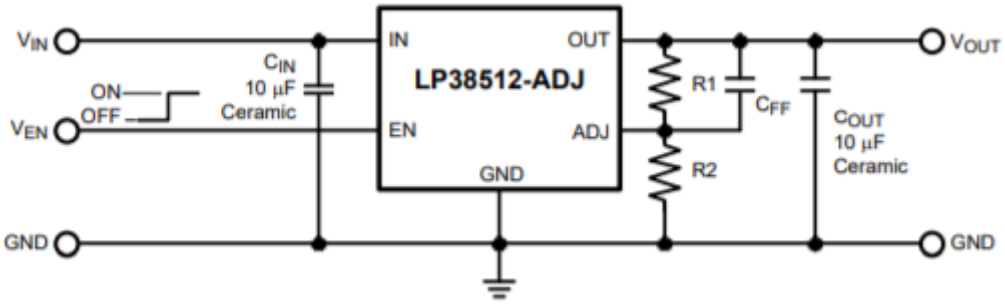


Figure 19: Circuit diagram to LP38512-ADJ, showing the required components for proper functionality.

$$V_{OUT} = V_{ADJ} \times (1 + \frac{R_1}{R_2})$$

$$C_{FF} = \frac{1}{2 \times \pi \times R_1 \times f_z}$$

The value for V_{ADJ} is given in the datasheet as 500 mV $\pm$10 mV, where values for V_{OUT}, R_1, and R_2 has to be decided. This circuit also requires an additional capacitor called C_{FF}, which is calculated using the formula given above. For optimum load transient response, the value of f_z should be between 20 kHz to 40 kHz.

Other voltages are also of necessity in the circuit; ±2.5V for the OpAmps at the preamplifier stage, a 0.9V reference voltage for the ADCs, and ±5V that supplies all of the power supplies mentioned above. Circuit diagrams for the +2.5V and 0.9V are shown below, respectively.

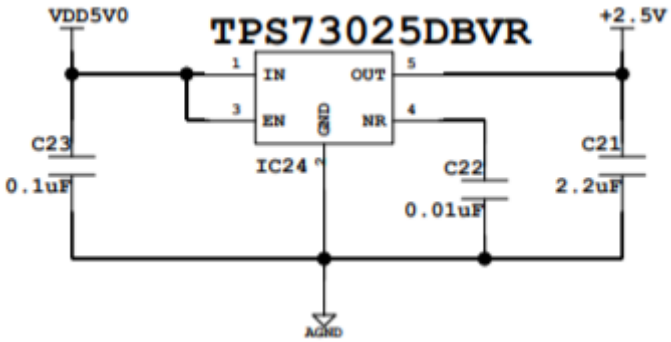


Figure 20: Circuit diagram to TPS73025DBVR, showing the required components for proper functionality.

The design of the +2.5V is given in the datasheet, where the capacitor sizes are described as well. The -2.5V uses a similar component from the TPS family, where the only difference in design is that the input voltage is -5V.

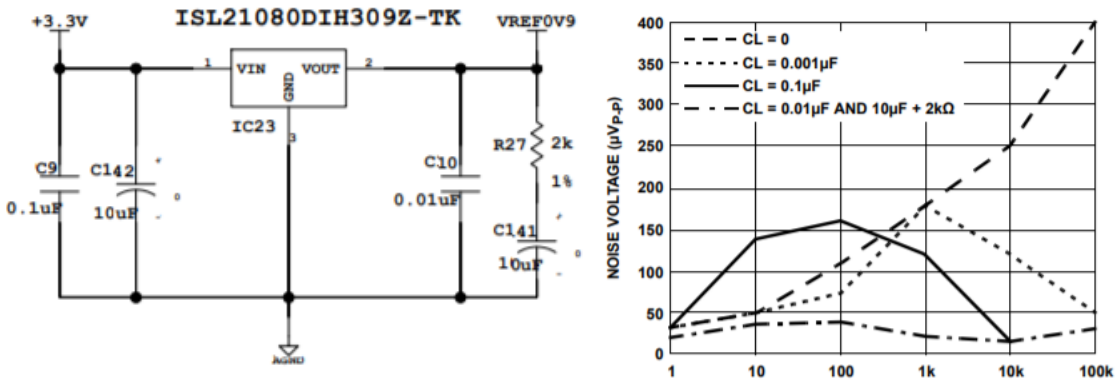


Figure 21: Circuit diagram to ISL21080DIH309Z-TK, including a noise vs. frequency graph, showing the required components for proper functionality, and how they affect the circuit's noise characteristics.

The design of the 0.9V reference voltage is given in the datasheet, where the shown graph of the noise vs. frequency presents how the different component sizes affect the circuit.

The ±5V circuit design comes from the FEGS system [9], which takes advantage of a line filter to reduce the noise fed into the power supply.

4.4 Layout

The layout is the last stage in the design process. This is where the placement and routing of components take place. As mentioned earlier, this stage can be straightforward in a simple design, but with a high-speed system like this, there are more advanced factors to take into consideration. The next couple of subchapters presents these factors and how the layout was created.

4.4.1 High-speed considerations

There are plenty of factors one might have to take into consideration in the physical design of a PCB. The importance of these factors is usually dependent on how complex the system is, or how important it is to keep a signals integrity. As the frequency increases, there is less room for imperfections, and thus something which seems negligible is now an important influence.

Variables such as trace length, width, material, and angle are crucial to keep in mind. A signal propagates through a PCB at about half the speed of light, which is $1.5x10^8$ m/s or 15cm/ns. This means that a signal does not instantly go from one end to another. At low frequencies, this is negligible compared to how fast the signal changes, but at higher frequencies, a signal varies at more substantial speeds. Thus, in high-speed components, like the ADCs and RAM in this design, the length of the differential pairs need to be close to identical, and the length difference for the interface traces towards the FPGA must be minimal.

One of the effects that can damage the integrity of a signal is the reflection. Parts of a signal can be reflected because of sharp corners or different impedance. This reflection changes the signal and thus hurt the integrity. To avoid these reflections, one must, in general, avoid 90-degree bends, and the impedance needs to be matching throughout the trace. If a signal goes from a medium with one impedance to another medium with a different impedance, parts of the signal are reflected and can dampen the signal.

4.4.2 Setup

When setting up the parameters for the system, it is crucial to select the right properties for layer Stackup, trace impedance, the distance between traces, VIA holes, and many others. Setting these parameters was done with inspiration from online resources [10] [11]. When deciding the layer Stackup, the number of layers was of priority, and a 10-layer stackup was selected. Other factors on layer Stackup, like layer thickness, impedance characteristic, trace width, and whether it is a signal- or power-layer was copied from [12]. Layer Stackup gives standardized parameters for the properties of the PCB. To select local parameters, Constraint Manager is a useful tool. Here, the trace width, trace distances, VIA holes, and other parameters can be specified for local areas. When tracing the pins for the FPGA and ADCs, the width of the traces and VIAs have to be significantly reduced from standard size to fit between the pins. To reduce this width, a constraint can be set in Constraint Manager that can create a region surrounding the component. This region automatically adjusts VIA size and trace width when crossing the border. The commonly used PCB parameters are copied from [13], where minimum and lowest price parameters are set.

4.4.3 Placement

After the important procedure of setting all the parameters, comes another important step; placing the components. Doing this well can save much time by simplifying the routing. A common saying within PCB design is – PCB design is 90% placement and 10% routing. This may not be a precise ratio, but it indicates the importance of this stage.

There are many strategies to create a readable and good design. To what extent one can uphold these strategies is dependent on the complexity of the design. The most common strategies are: placing components to achieve the least amount of trace crossings; having the signal go from left to right by placing them according to the order of signal propagation; etc.

The placement in this design is based around these rules. It also resembles the block diagram in Figure 4.

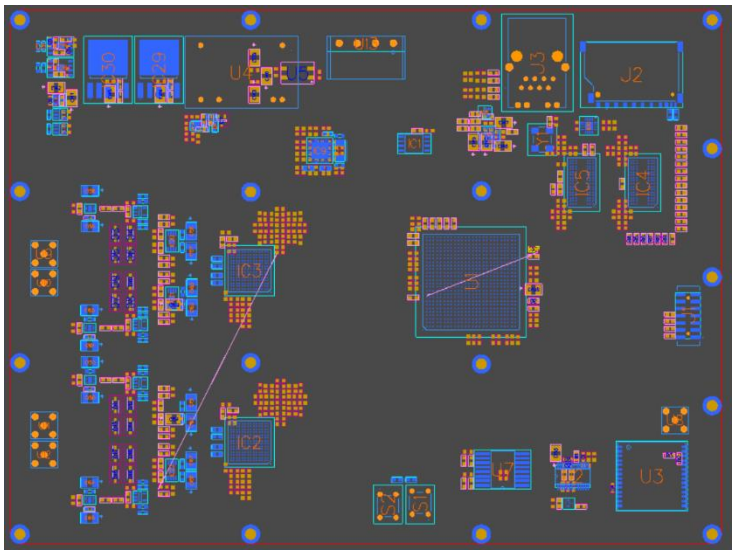


Figure 22: Component placement for ALOT

4.4.4 Routing

The routing of the layout can start once the placement of all the components has been decided. Creating traces for the well-placed data-acquisition stage is as simple as it gets, but it is a different ball game for routing the ADC and RAM ICs. The ADCs connections are not fixed and can, therefore, be changed to fit the tracing. Planning these connections was done manually before the routing. The software helps keeping the length of the differential pairs as similar as possible but getting all pairs from one IC to about the same length, requires some manual tuning afterward.

This stage is all about physically connecting all the components while trying to use the least amount of via holes, layers, and making the shortest traces possible. The autoroute function can route the entire design of this project, but it is not optimal. Using autoroute requires much tweaking in Constraint Manager to set the right parameters. In this design, auto-route uses most of the signal layers available to route the ADCs, while with manual routing, only 3 layers are needed.

5 Conclusion and Summary

With this thesis, a data acquisition-, processing-, and storage stage have been developed which allows sensors to transfer information regarding the TGFs to an FPGA in the form of raw, digitized data, which the FPGA manipulates in accordance to the firmware, and stores this newly processed data in memory. The required component upgrades in contrast to FEGS have been achieved, with the FPGA going from a Zynq-7010 to Zynq-7030, providing half a magnitude increase in programmable logic. An upgrade of one order of magnitude on the ADC's sampling rate has been achieved, and because of its differential signals the benefit of reduced crosstalk between traces. An SD card connector has been added, presenting the system with the possibility of storing data on a removable device, making the stored data more accessible. Also, the required components to establish communication with an external device through Ethernet has been added, providing a way to monitor and transmit information about the system while it is running.

The data acquisition stage has been equipped with a preamplifier that both amplifies the signals and converts signals from currents to voltages, which is required in the following substage, the filter. The filter is a fourth order Butterworth low pass filter, composed of two-second order filters; a KRC filter and an MFB filter, forming a stopband appearing at a value below the Nyquist frequency, responsible of removing unwanted signals and the possibility of aliasing from happening. Before the ADC, the component concluding the data acquisition stage resides; an ADC driver that converts the signals from single-ended to differential to take advantage of the differential inputs on the ADCs.

The use of a Zynq-7030 FPGA provides the data processing stage large quantities of programmable logic, which enables the possibility of larger firmware. By employing an FPGA, other crucial components are required for it to function, a few of them being: A clock generator providing the oscillating clock pulses that are critical in several components, the power supplies that are of paramount importance for the FPGA, and the entire system to function, and a JTAG connector, necessary to deliver the FPGA its configuration.

While the FPGA is a volatile component, in which its configuration is erased if powered down, the addition of flash memory gives the benefit of restoring the configuration if a power malfunction should occur. As the FPGA is hard at work processing information about the TGFs, it may require storing some data for later use. This data can then be stored in RAM, where the memory can easily be accessed when needed. When the data has been processed, it is essential to have a reasonably sized memory to place the data until the end of the flight. An SD card is selected as the preferred memory device, as it is removable and easily manageable. These three categories of memory form the data storage stage.

Though the components have been placed, there are still work that is required for the layout to be finished; modification of the constraints set in constraint manager to provide the proper PCB characteristics necessary in the system, and the finishing touch on the tracing between components.

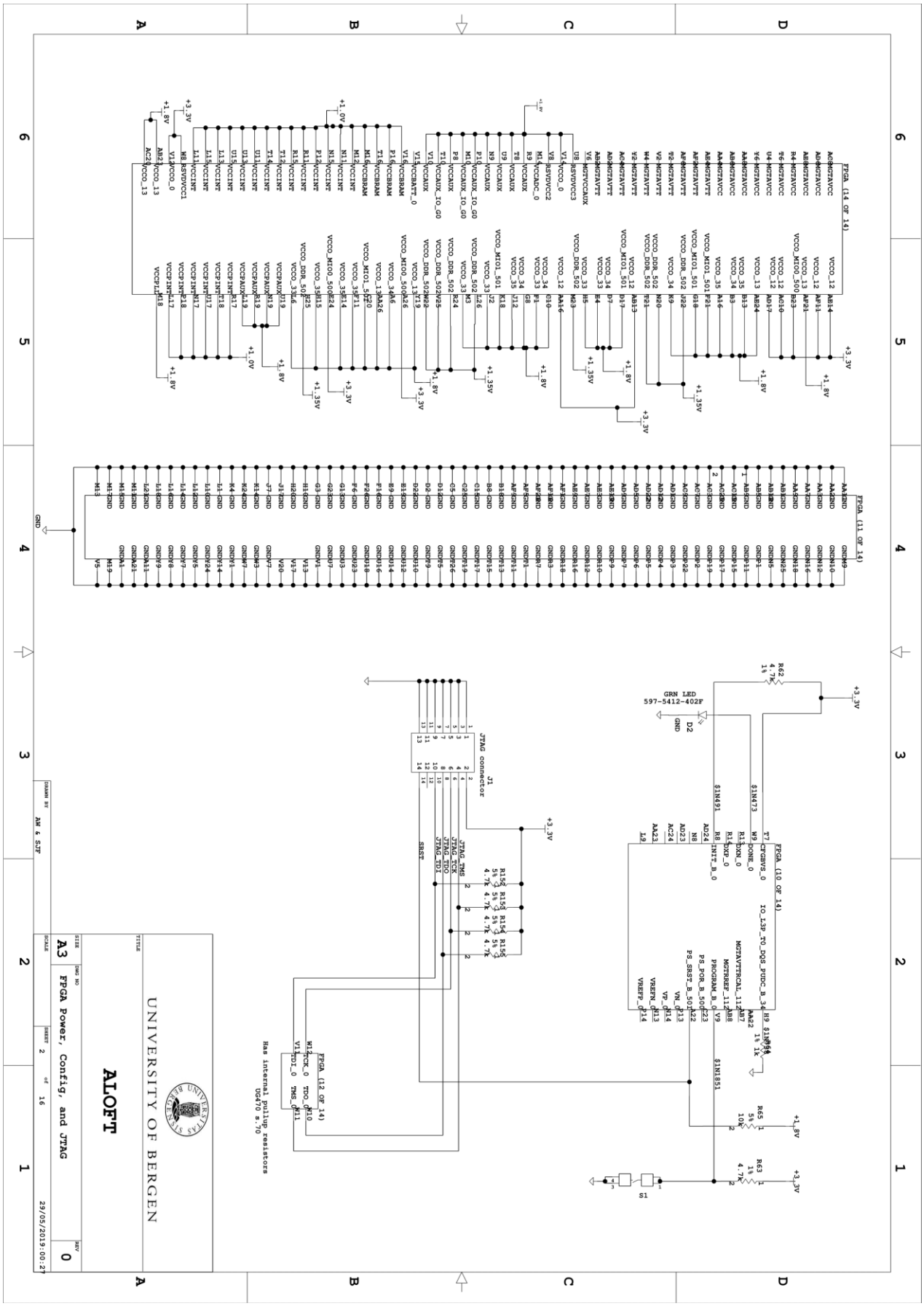
While the simulation of the acquisition stage may give out good or bad results, it does not tell how the stage may behave in reality. The reason for this is all the non-idealities that reside in an active element, and in general, the uncertainty that any component has. To confidently say if a circuit is good or bad, an actual real-life test of the components is necessary. Without this test, one can only speculate on how the circuit behaves. When the real-life testing of the data acquisition stage is accomplished and return reassuring values indicating that the circuit can be applied in the system, and the layout is finished, the board can be ordered, and functional testing can be performed.

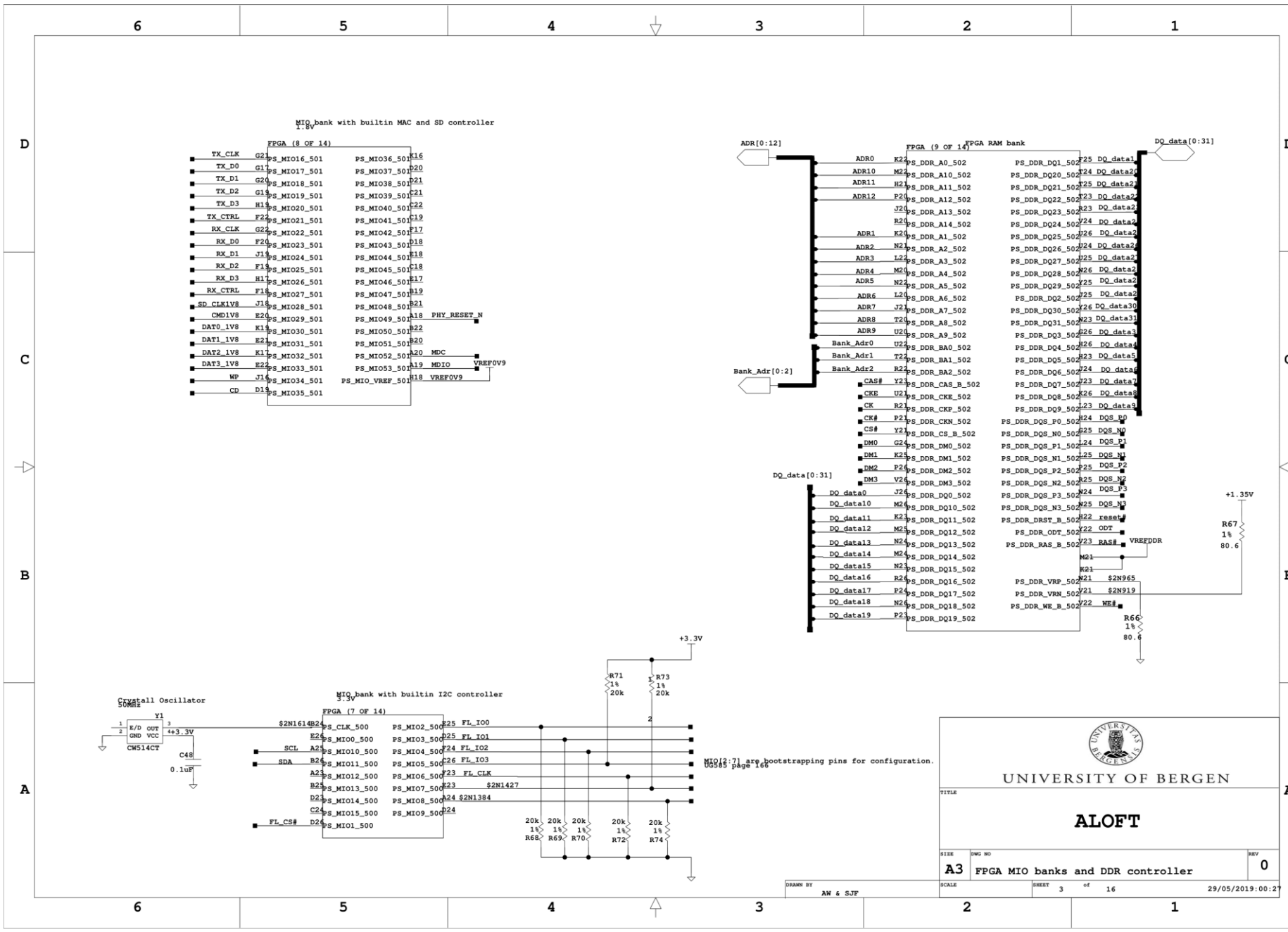
6 Appendices

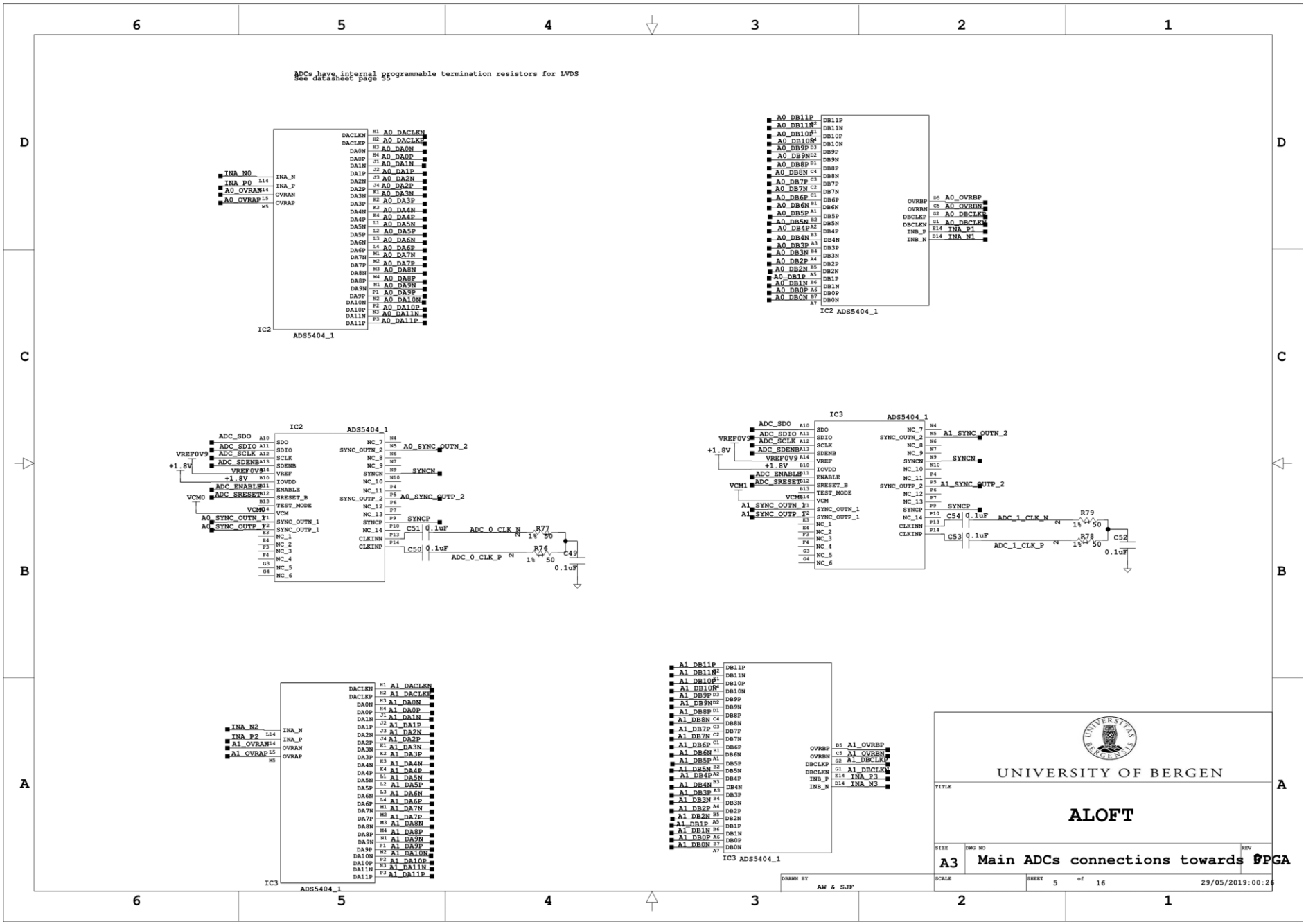
Appendix A - Words and abbreviations

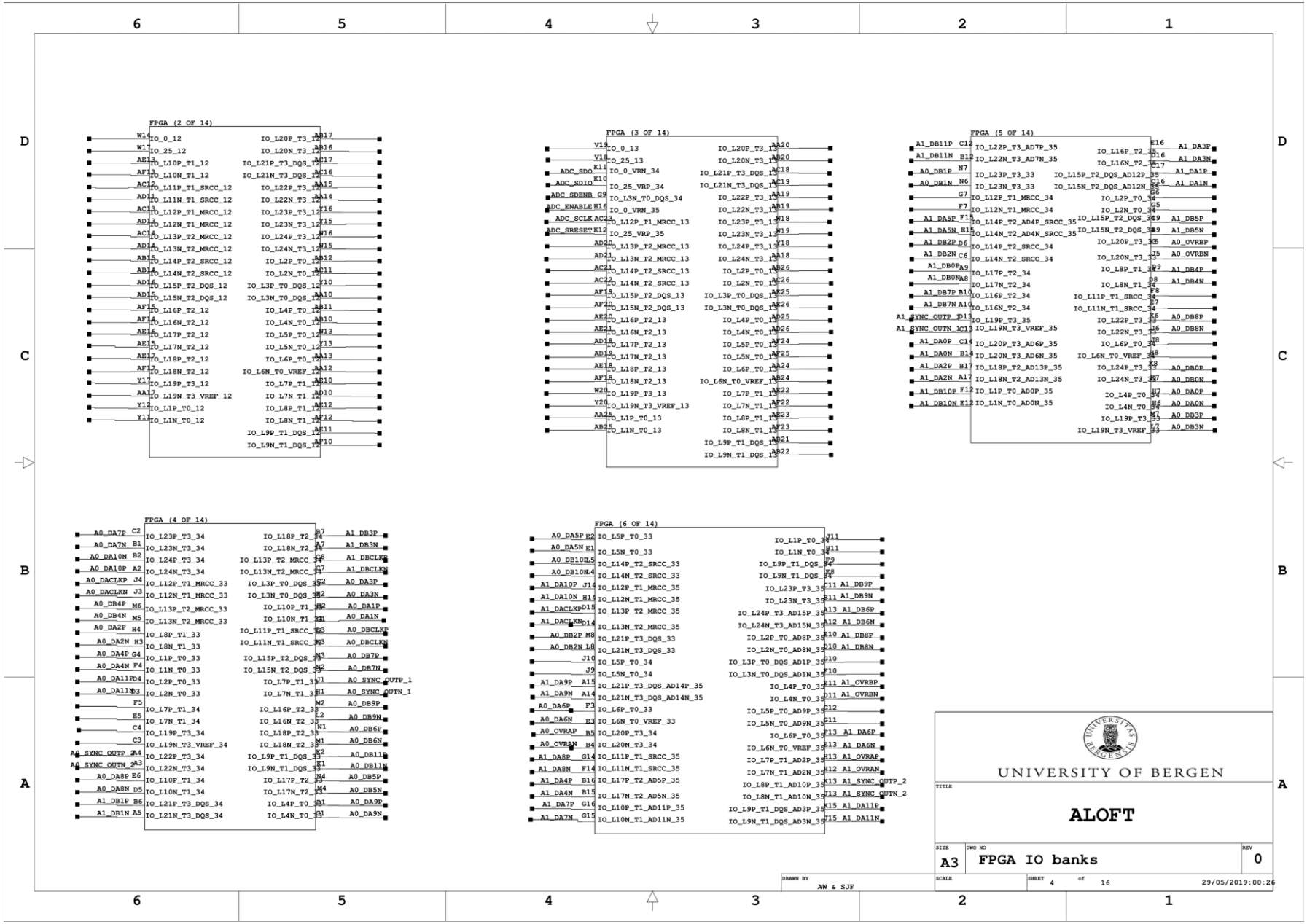
ALOT - Airborne Lightning Observatory for TGFs
ADC - Analog to Digital Converter
BCSS – Birkeland Centre for Space Science
BRAM - Block Random Access Memory
CMOS - Complementary Metal–Oxide–Semiconductor
FEGS - Fly's Eye GLM Simulator
FPGA - Field Programmable Gate Array
GLM - Geostationary Lightning Mapper
IC - Integrated Circuit
I^2C - Inter-integrated Circuit (A serial communication protocol)
I/O - Input/Output
LDO - Low-Dropout
LUT - Look-Up-Table
LVCMOS - Low-Voltage CMOS
LVDS - Low-Voltage Differential Signaling
MAC – Media access control
MSPS - Mega Samples Per Second
MFB - Multiple Feedback
OpAmp - Operational Amplifier
PCB - Printed Circuit Board
PHY – Physical layer for Ethernet
PLL - Phase-Locked Loop
PMT - Photomultiplier Tube
RAM – Random access memory
SNR – Signal to noise ratio
SoC - System on Chip
SPI - Serial Peripheral Interface
TIA - Transimpedance Amplifier
TGF - Terrestrial gamma-ray flash
UiB - University in Bergen

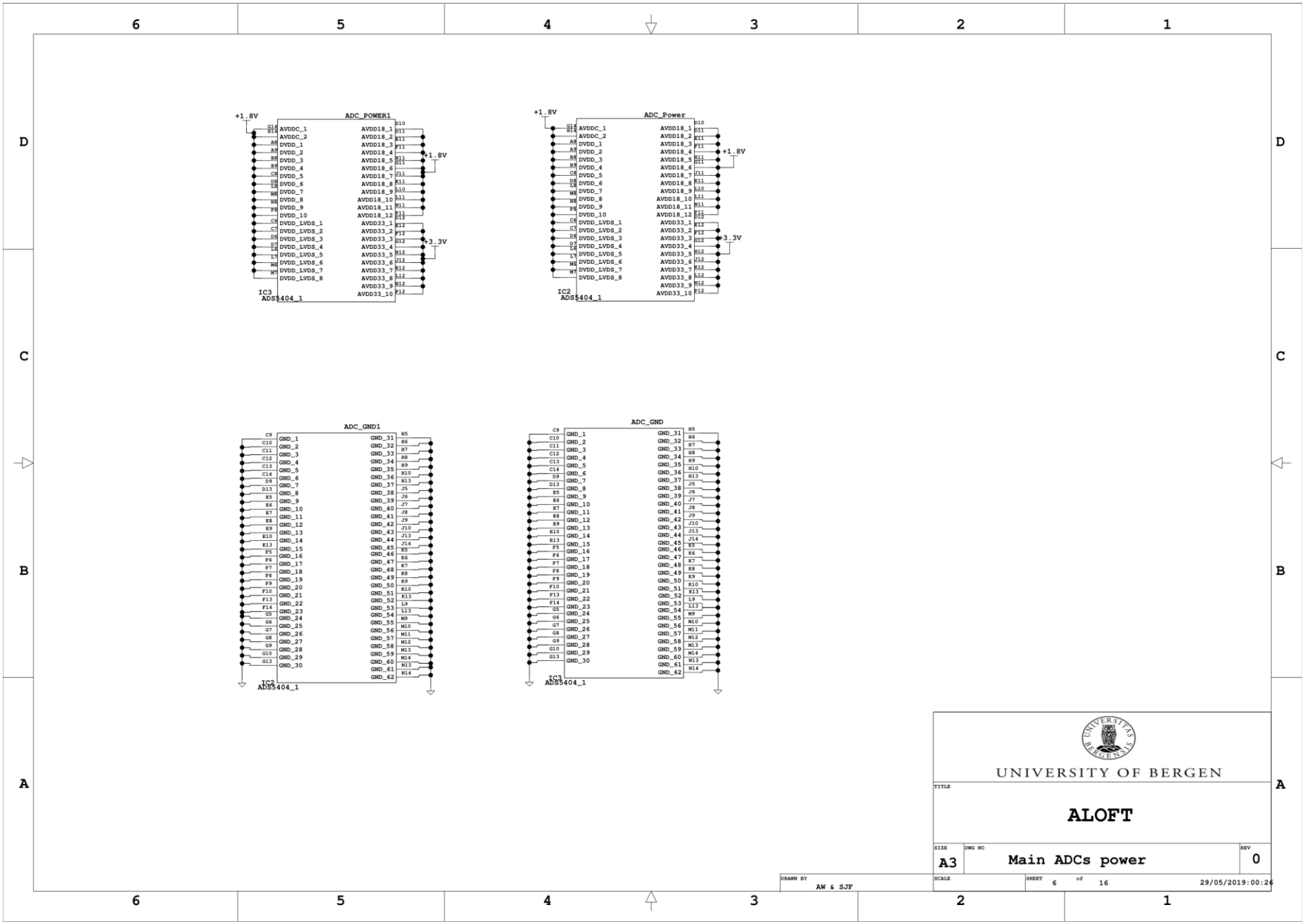
Appendix B – Schematic

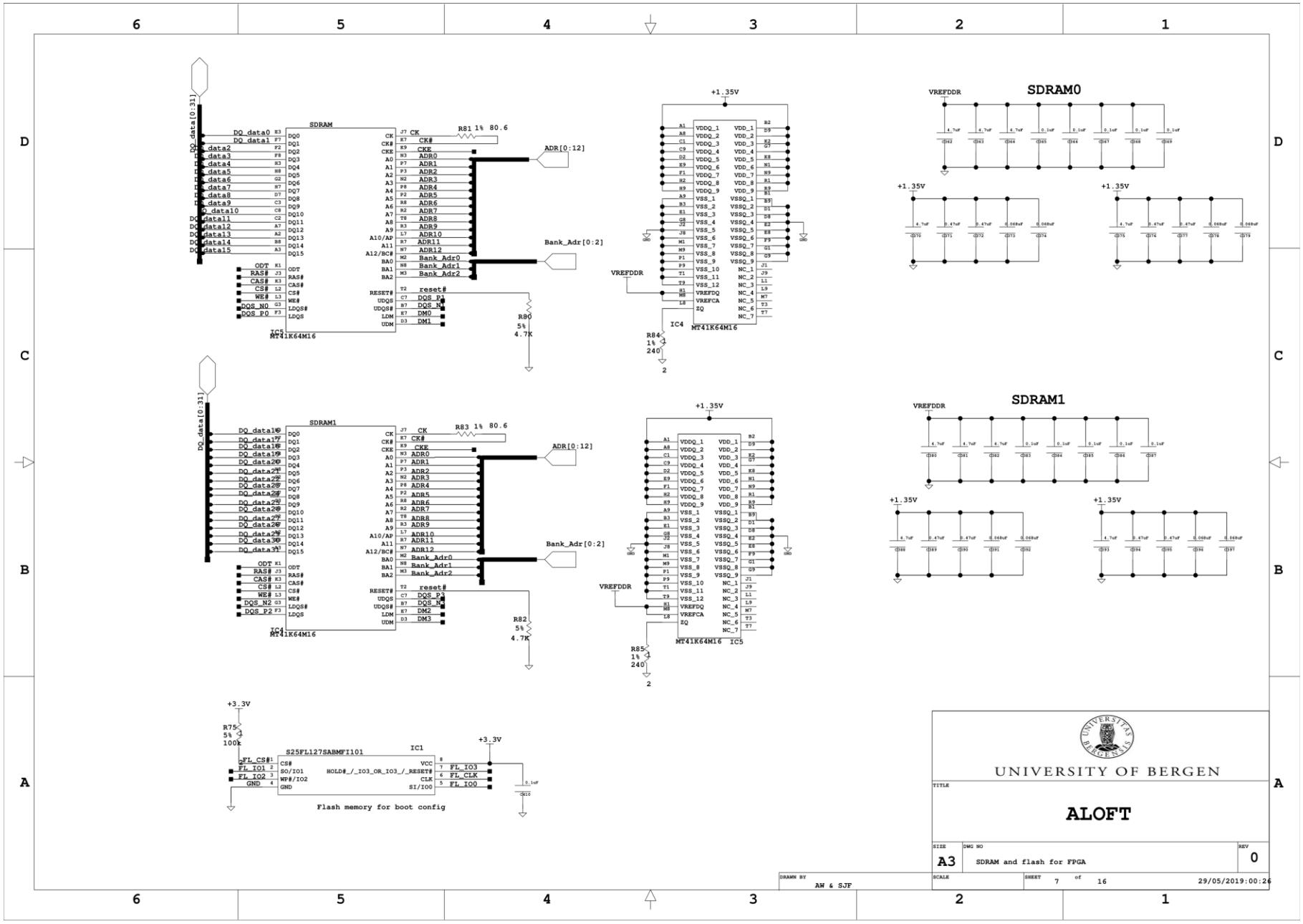


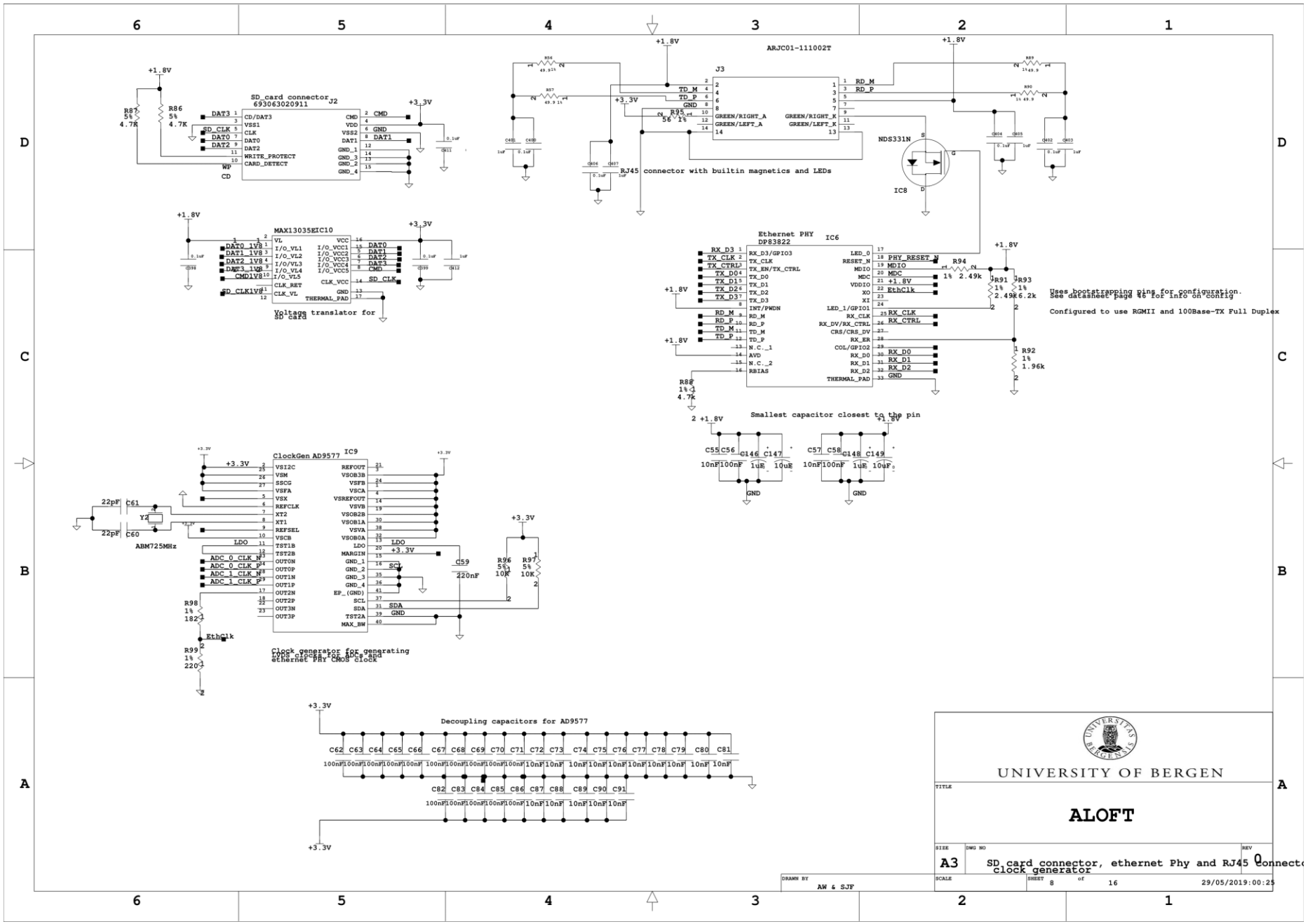

MIO bank with builtin MAC and SD controller
1.8V
FPGA (8 OF 14)
TX_CLK
TX_D0
TX_D1
TX_D2
TX_D3
TX_CTRL
RX_CLK
RX_D0
RX_D1
RX_D2
RX_D3
RX_CTRL
SD_CLK1V8
CMD1V8
DAT0_1V8
DAT1_1V8
DAT2_1V8
DAT3_1V8
WP
CD
PHY_RESET_N
MDC
MDIO
VREF0V9
ADR[0:12]
Bank_Adr[0:2]
FPGA (9 OF 14)
FPGA RAM bank
DQ_data[0:31]
CAS#
CKE
CK
CK#
CS#
DM0
DM1
DM2
DM3
VREFDDR
+1.35V
R67
80.6
R66
80.6
Crystall Oscillator
50MHz
CW514CT
C48
0.1uF
+3.3V
MIO bank with builtin I2C controller
3.3V
FPGA (7 OF 14)
SCL
SDA
FL_CS#
FL_IO0
FL_IO1
FL_IO2
FL_IO3
FL_CLK
R68
R69
R70
R71
R72
R73
R74
20k
1%
MIO[8:7] are bootstrapping pins for configuration.
UG585 page 166
UNIVERSITY OF BERGEN
TITLE
ALOFT
SIZE
A3
DWG NO
FPGA MIO banks and DDR controller
REV
0
DRAWN BY
AW & SJF
SCALE
SHEET
3
of
16
29/05/2019:00:27

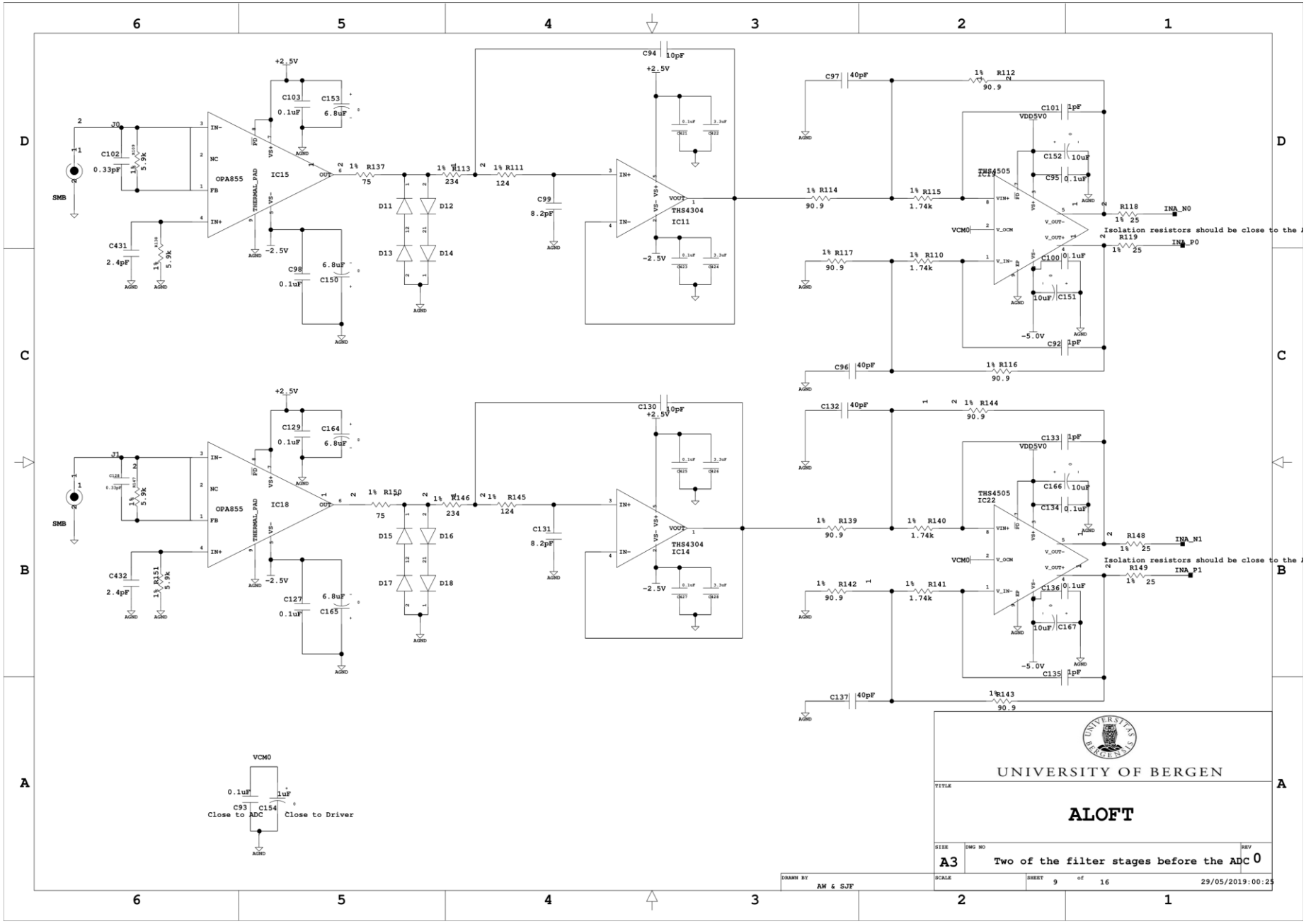

ADCs have internal programmable termination resistors for LVDS
See datasheet page 35
IC2
ADS5404_1
IC3
ADS5404_1
UNIVERSITY OF BERGEN
TITLE
ALOFT
SIZE
A3
DWG NO
Main ADCs connections towards FPGA
REV
SCALE
SHEET 5 of 16
DRAWN BY
AW & SJF
29/05/2019:00:26

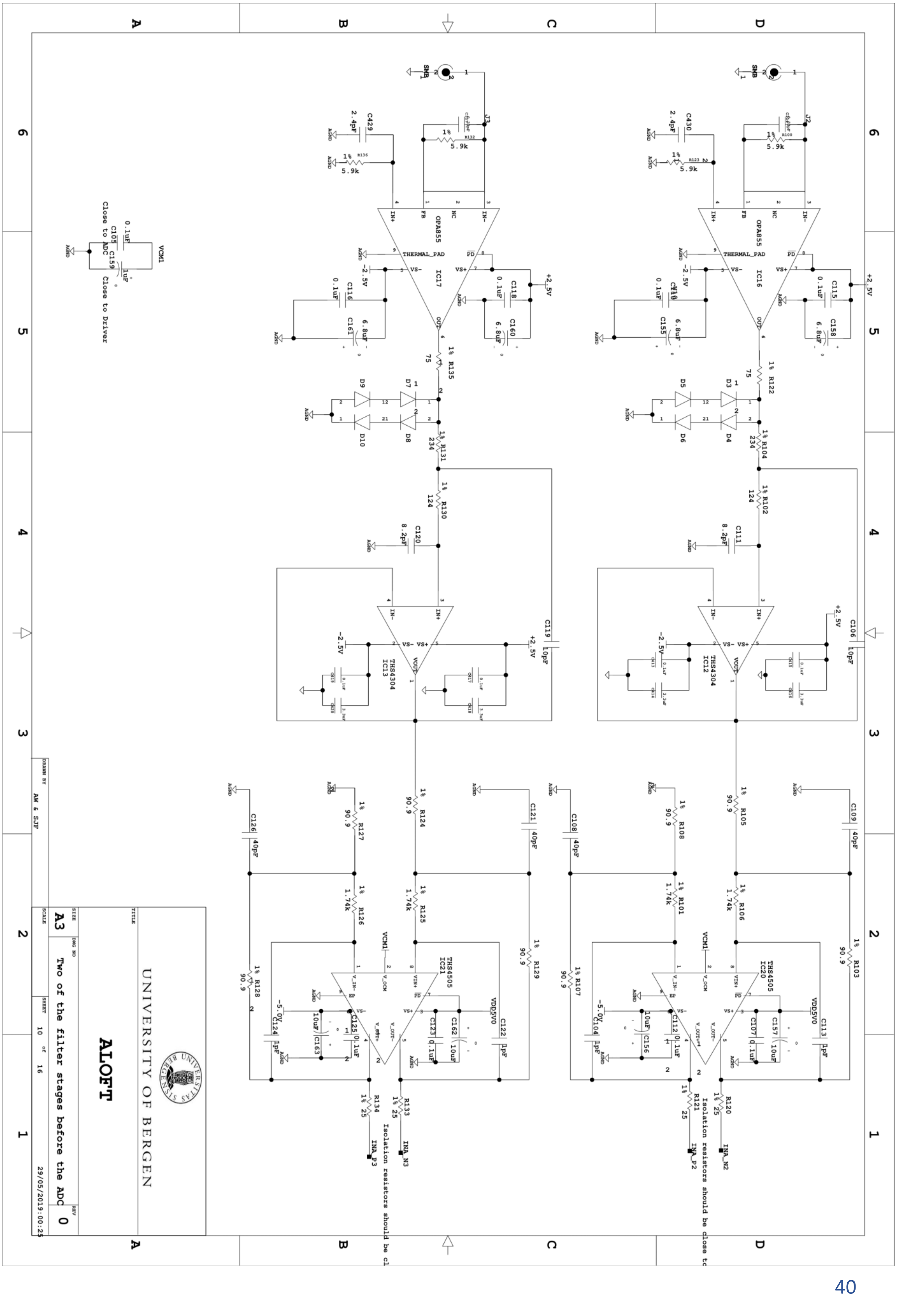

FPGA (2 OF 14)
FPGA (3 OF 14)
FPGA (5 OF 14)
FPGA (4 OF 14)
FPGA (6 OF 14)
UNIVERSITY OF BERGEN
TITLE
ALOFT
SIZE
A3
DWG NO
FPGA IO banks
REV
0
DRAWN BY
AW & SJF
SCALE
SHEET
4
of
16
29/05/2019:00:26

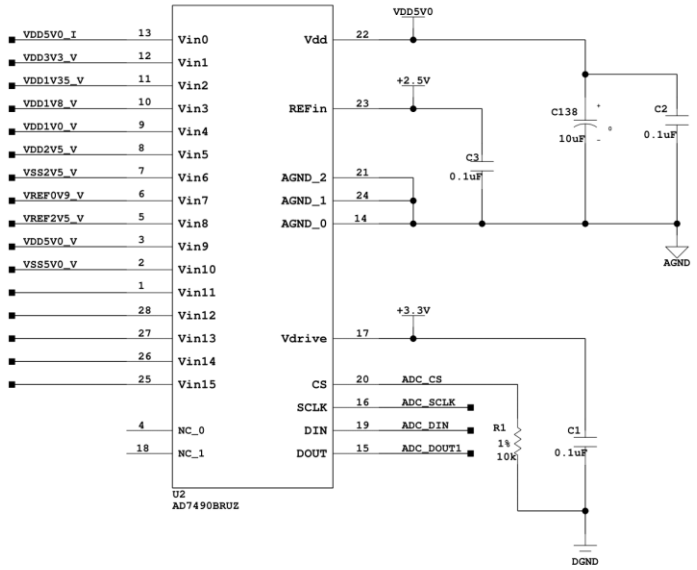

ADC_POWER1
+1.8V
+3.3V
IC3
ADS5404_1
ADC_Power
IC2
ADS5404_1
ADC_GND1
IC2
ADS5404_1
ADC_GND
IC3
ADS5404_1
UNIVERSITY OF BERGEN
TITLE
ALOFT
SIZE
A3
DWG NO
Main ADCs power
REV
0
DRAWN BY
AW & SJF
SCALE
SHEET 6 of 16
29/05/2019:00:26

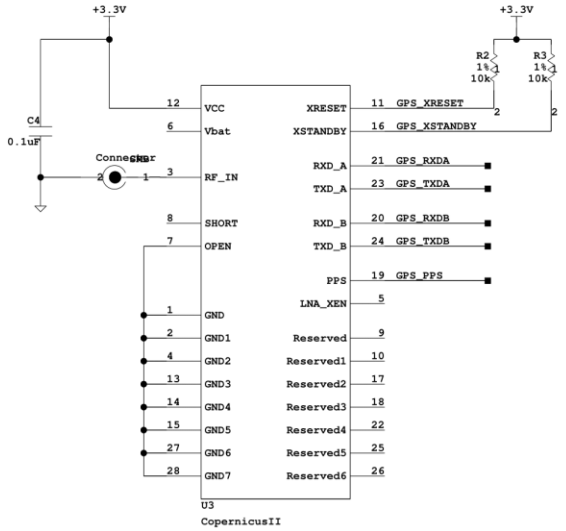

SDRAM
SDRAM1
SDRAM0
MT41K64M16
IC4
IC5
R81 1% 80.6
R83 1% 80.6
R80 5% 4.7K
R82 5% 4.7K
R84 1% 240
R85 1% 240
ADR[0:12]
Bank_Adr[0:2]
DQ_data[0:31]
+1.35V
VREFDDR
+3.3V
R75 5% 100k
S25FL127SABMFI101
IC1
Flash memory for boot config
UNIVERSITY OF BERGEN
ALOFT
SDRAM and flash for FPGA
A3
0
AW & SJF
7
16
29/05/2019:00:26


SD_card connector
693063020911
J2
MAX13035EIC10
Voltage translator for SD card
ARJC01-111002T
J3
RJ45 connector with builtin magnetics and LEDs
NDS331N
IC8
Ethernet PHY
DP83822
IC6
Uses bootstrapping pins for configuration.
See datasheet page 46 for info on config
Configured to use RGMII and 100Base-TX Full Duplex
Smallest capacitor closest to the pin
ClockGen AD9577
IC9
ABM725MHz
Clock generator for generating
100 clocks for adcs and
ethernet PHY CMOS clock
Decoupling capacitors for AD9577
UNIVERSITY OF BERGEN
ALOFT
SD card connector, ethernet Phy and RJ45 connecto
clock generator
AW & SJF
29/05/2019:00:25


UNIVERSITY OF BERGEN
ALOFT
Two of the filter stages before the ADC
A3
0
AW & SJF
9
16
29/05/2019:00:25
OPA855
THS4304
THS4505
Isolation resistors should be close to the ADC
Close to ADC
Close to Driver
INA_N0
INA_P0
INA_N1
INA_P1
VCM0
+2.5V
-2.5V
-5.0V
VDD5V0
AGND


UNIVERSITY OF BERGEN
ALOFT
Two of the filter stages before the ADC
AW & SJF
A3
10 of 16
0
29/05/2019:00:25
Close to ADC
Close to Driver
Isolation resistors should be close to





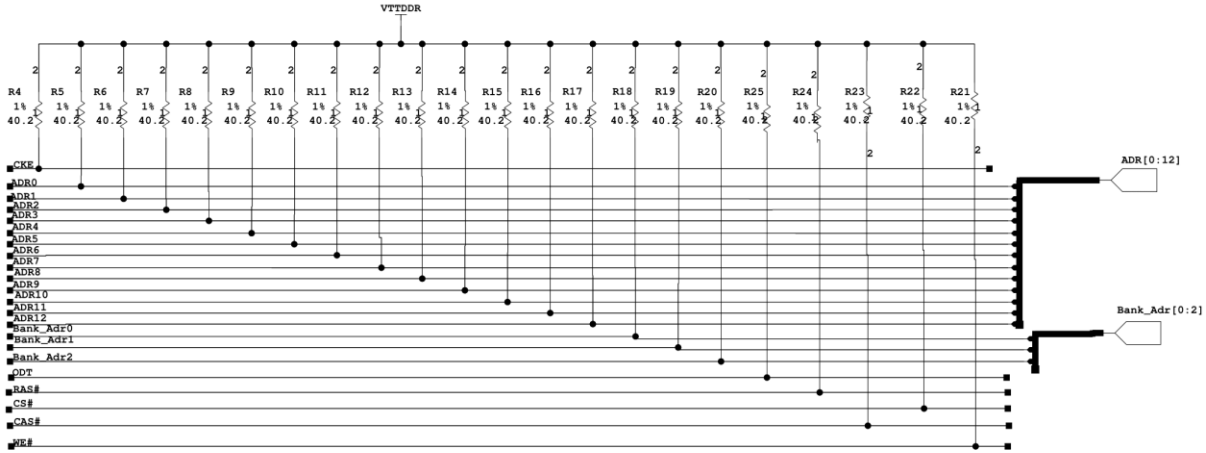
UNIVERSITY OF BERGEN

TITLE

ALOFT

SIZE	DWG NO	REV
A3	Extra ADC and GPS module	1

DRAWN BY	SCALE	SHEET			
AW & SJF		11	of	16	29/05/2019:00:25



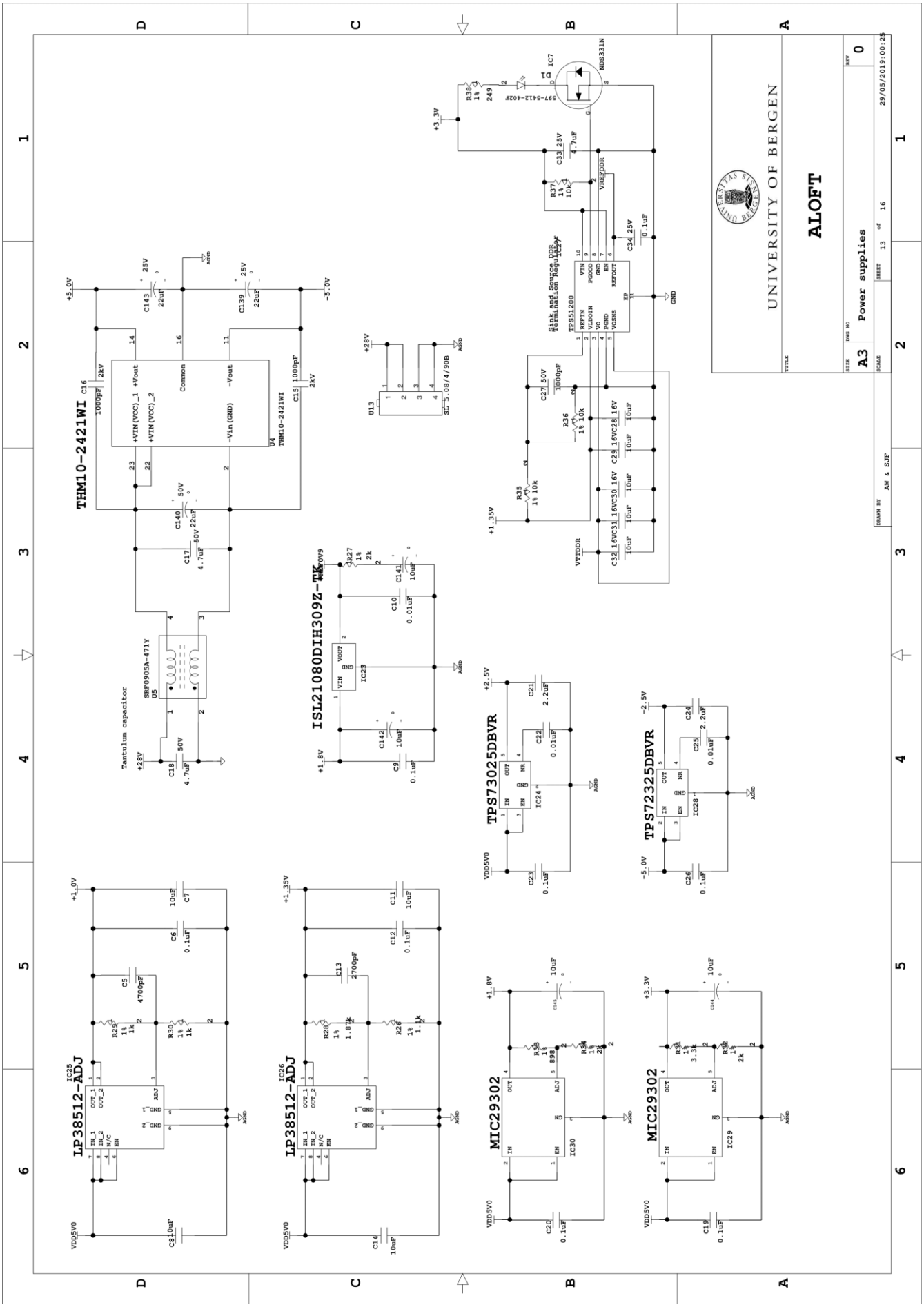
UNIVERSITY OF BERGEN

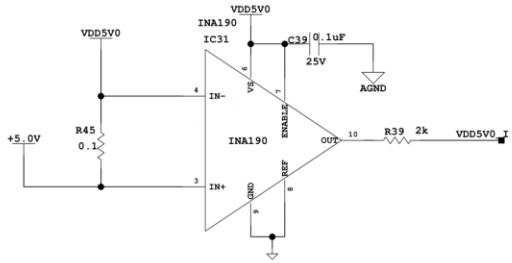
TITLE

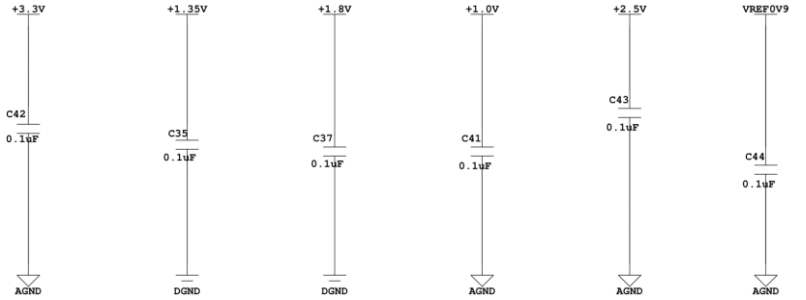
ALOFT

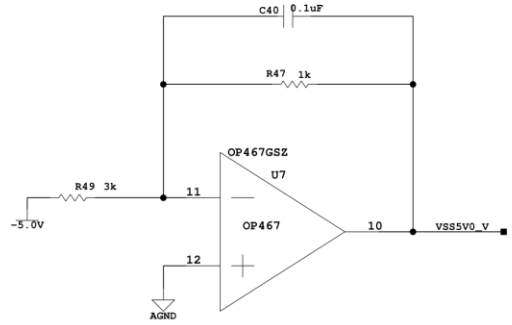
SIZE	DWG NO	REV
A3	Termination resistors for the SDRAMs	0

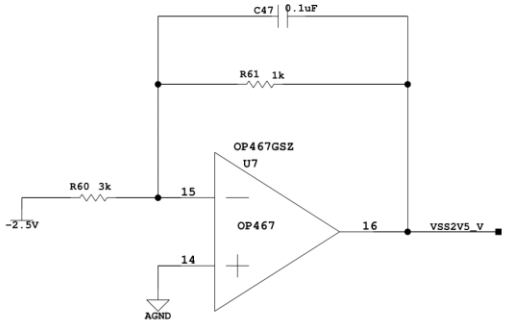
DRAWN BY	SCALE	SHEET	
AW & SJF		12 of 16	29/05/2019:00:25

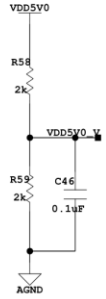

UNIVERSITY OF BERGEN
ALOFT
Power supplies
A3
AM & SJF
29/05/2019:00:25
THM10-2421WI
ISL21080DIH309Z-TK
TPS73025DBVR
TPS72325DBVR
LP38512-ADJ
MIC29302
Tantulum capacitor
Sink and Source DDR Termination Regulator
TPS51200
SI 5.08/4/90B
SERF0905A-471Y
NDS331N

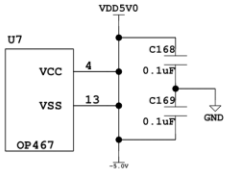












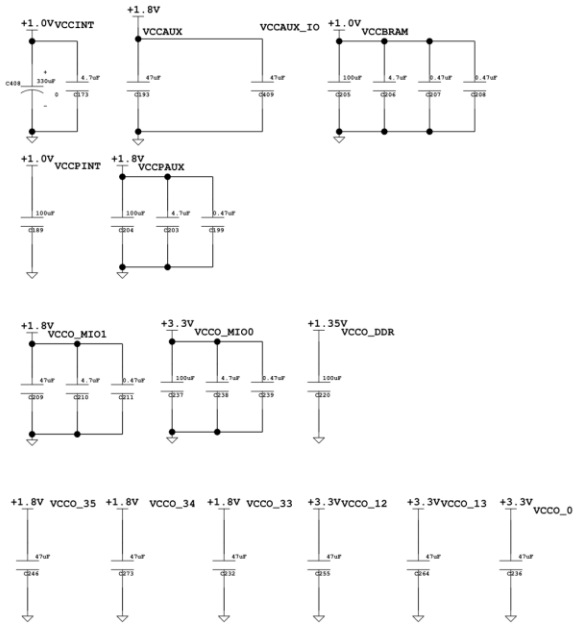
UNIVERSITY OF BERGEN

TITLE

ALOFT

SIZE	DWG NO	REV
A3	System variables	1

DRAWN BY	SCALE	SHEET	
AW & SJF		14 of 16	29/05/2019:00:25



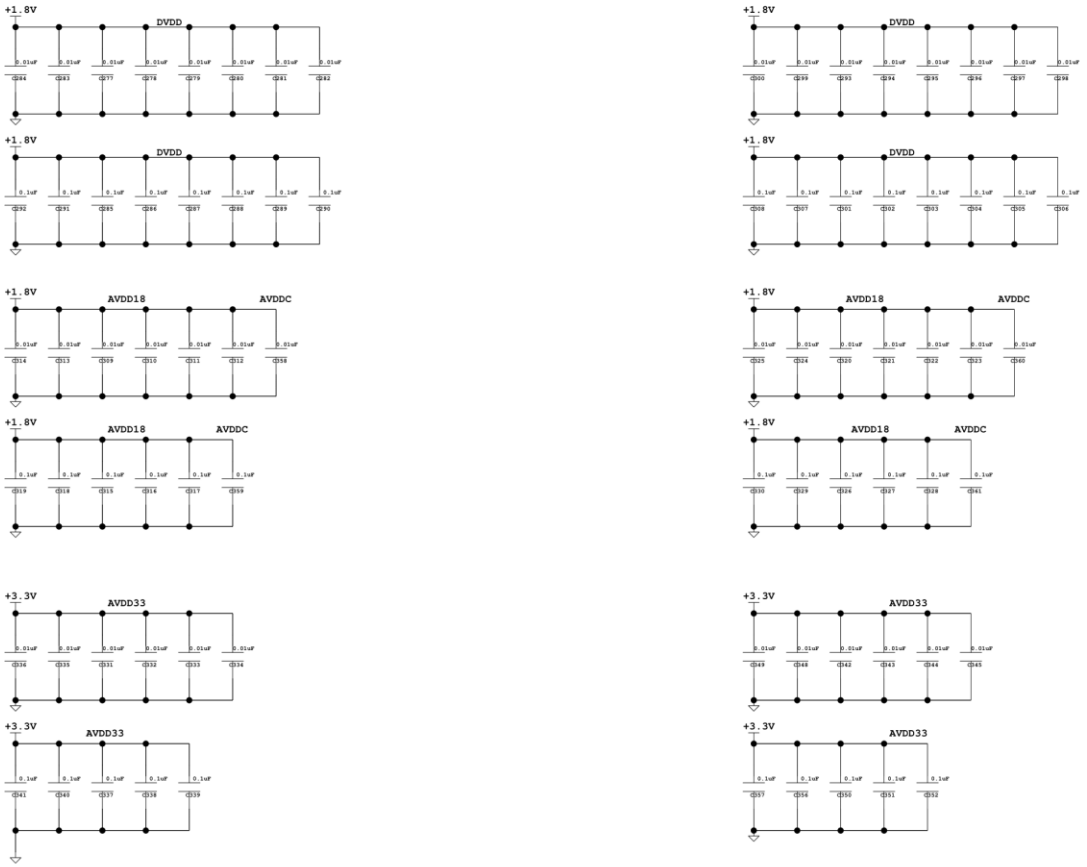
UNIVERSITY OF BERGEN

TITLE

ALOFT

SIZE	DWG NO	REV
A3	Decoupling Capacitors for the FPGA	0

DRAWN BY	SCALE	SHEET	
AW & SJF		15 of 16	29/05/2019:00:28



UNIVERSITY OF BERGEN

TITLE: **ALOFT**		
SIZE: A3	DWG NO: Decoupling Capacitors for the ADCs	REV: 0
DRAWN BY: AW & SJF	SCALE:	SHEET: 16 of 16

29/05/2019:00:25

Appendix C - Technical Explanations

C.1 Interfaces

RGMII

RGMII [14] [15] is one of several interfaces with the possibility of connecting the Ethernet MAC to the Ethernet PHY. This interface takes advantage of Double Data Rate, in which both the rising and falling edge of a clock pulse is used. RGMII significantly reduces the pin count between the MAC and PHY, saving up space on the board.

Quad-SPI

SPI [14] is convenient in short distance communication and utilizes synchronous serial communication. Synchronous serial communication uses a clock signal to synchronize the data that is being sent from transmitter to receiver. A general SPI has four pins; Serial Clock (SCLK), Master Output Slave Input (MOSI), Master Input Slave Output (MISO) and Slave Select (SS). Here, SCLK provides the synchronization of data, MISO and MOSI are for sending and receiving data, and SS decides which slave the master is communicating with. Quad-SPI is an extension of the general SPI, where four bidirectional data lines are used instead of two unidirectional lines. Here, MOSI and MISO are bidirectional, called DQ0 and DQ1, respectively, with two additional pins, DQ2 and DQ3, added to the interface.

I^2C

I^2C [15] is a synchronous serial communication protocol that is commonly used to attach low-speed peripheral ICs to processors. I^2C has the possibility of communication between multiple masters and slaves, where an address byte decides whether a unit receives or transmit data. The interface uses two pins to accomplish data transfers. These pins are named Serial Data (SDA) and Serial Clock (SCL), responsible for the data transfer and synchronization, respectively.

SDIO

SDIO is made explicitly for SD memory cards and provides an interface that makes data transfer between a processor and an SD memory card possible.

C.2 Signaling

LVDS

Low-Voltage Differential Signaling [16] [17], or LVDS, is a high-ranking technical standard used in a multitude of data communication standards. LVDS consists of two differential wires, a driver delivering a current of ±3.5mA, and a receiver with a 100Ω termination resistor, as shown in **Error! Reference source not found.** A). The impedance on the transmission line must closely match the termination impedance to avoid harmful signal reflections that decrease signal quality. When the signal propagates across the transmission line, it sees ideally infinite impedance in the receiver, making the current flow in the path of least resistance, namely the termination resistor. When the current flows through this resistor, by ohms law, the voltage across the resistor, which gives the voltage difference between the inverting and non-inverting inputs on the receiver, is 350mV. This voltage difference is shown in **Error! Reference source not found.** B), where the signal is offset to 1.2V to have the signal centered between its outer limits of 0 and 2.4V. Due to LVDS' low voltage swing, less power is required, and quicker transition times are possible, making higher data rates achievable. Even though LVDS requires twice as

many wires for transmission over the same number of channels as single-ended, it can achieve much higher speeds, making fewer channels necessary to transmit the same amount of data. Other factors that make LVDS so popular is its noise immunity and its reduced amount of noise emission compared to other technical standards. While the wires may be susceptible to environmental noise, the differential property of LVDS rejects this due to the noise being induced in approximately equal amounts on both wires, which is subtracted out at the receiver, giving it immunity from environmental noise. Its reduced noise emission is because of the same amounts of current travel in opposite directions in the wires, creating equal but opposite electromagnetic fields around the wires that cancel each other out. This cancellation reduces the strength of the electromagnetic fields that are present in the wires, producing less electromagnetic noise that may corrupt data transfer on other lines.

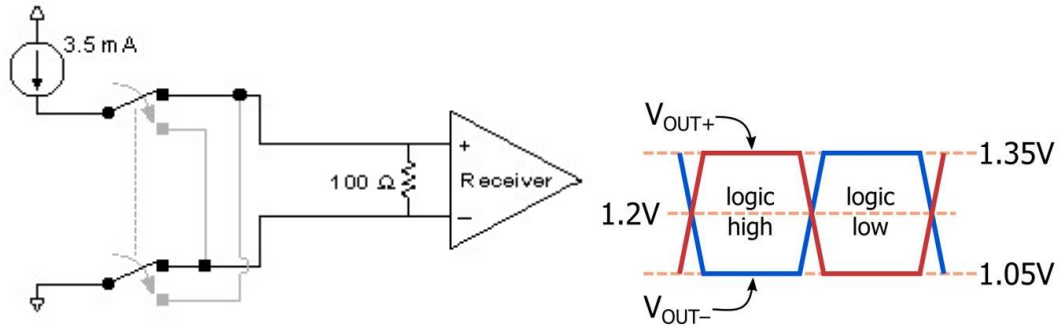


Figure 23: A) Simplification of LVDS driver and receiver. B) Differential signal.

CMOS

CMOS [18] is of the single-ended output families and is suitable for lower frequency clock sources and short trace lengths [18]. A direct connection between transmitter and receiver can be established, but often a 20-50Ω resistor is placed between to reduce signal reflections that decrease signal quality. There is a multitude of CMOS types: High-Speed CMOS, or HCMOS; Low Voltage CMOS, or LVCMOS; Advanced CMOS, or ACMOS; etc.

Single-ended vs. Differential

Single-ended signals [19] transmit signals across a single wire and measures the voltage difference between the wire and ground. A wire is susceptible to environmental noise like electromagnetic fields, that is added to the signal propagating through the wire. The received signal may contain large amounts of noise that may compromise crucial information. A differential signal, however, adds another wire to each channel, where measurements of voltage differences are between wires instead. Differential signaling is shown in **Error! Reference source not found.**.**Error! Reference source not found.** The wires are just as susceptible to noise as for single-ended, but the noise is induced in approximately the same amount on both wires, which is removed when measuring the difference between the wires. This removal of noise due to measuring the voltage difference between wires makes the signaling immune to environmental noise.

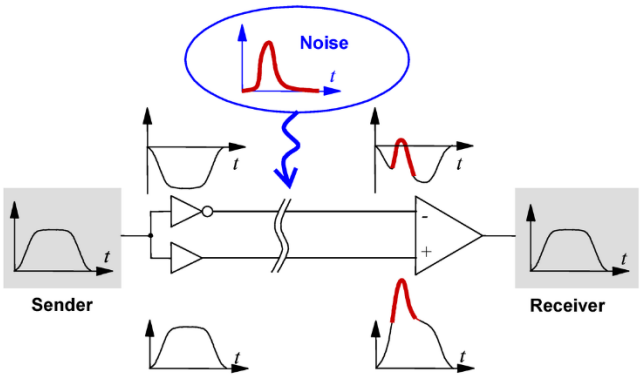


Figure 24: Differential signal propagating from the transmitter to the receiver.

DDR

Double data rate is when data is sampled on both the rising and falling edge of a clock pulse. This means that double the amount of data can effectively be transmitted or received. DDR can be used in both single-ended or differential signaling.

Figure list

Figure 1: An illustration of a terrestrial gamma-ray flash in the atmosphere. The colors pink, yellow, and green represents gamma-rays, electrons, and positrons, respectively [9]..... 2
Figure 2: Symbol versus Cell of a dual opAmp IC 4
Figure 3: Design parameters from Elprint.no [7] .....**Error! Bookmark not defined.**
Figure 4: Block diagram for the ALOT system 7
Figure 5: Pin compatibility between the different FPGA typesFigure 6: Pin compatibility between the different FPGA types [33].....**Error! Bookmark not defined.**
Figure 7: A table comparing the different Zynq-7000 FPGAs [33] .....**Error! Bookmark not defined.**
Figure 8: Amplifier, filter, and driver placed respectively from left to right. 15
Figure 9: MFB filter placed externally on the ADC driver. 17
Figure 10: The system's closed-loop gain without feedback capacitor at the TIA stage. 18
Figure 11: The system's closed-loop gain with feedback capacitor at the TIA stage, with its pole in advance of the peak. 18
Figure 12: Simulation showing the systems bode and phase plot. 19
Figure 13: Simulation using step response to analyze the stability of the data acquisition stage 19
Figure 14: Simulation using triangular pulses, presenting the circuit maintaining the signal's integrity as the signal propagates from source to output. 20
Figure 15: A simple representation of memory division in a RAM .....**Error! Bookmark not defined.**
Figure 16: Possible configurations for the FPGA [3].....**Error! Bookmark not defined.**
Figure 17: Example circuit of the clock generator [20] 23
Figure 18: A schematic of the RJ-45 connector [22] 24
Figure 19: Circuit diagram to MIC29302, showing the required components for proper functionality. 25
Figure 20: Circuit diagram to LP38512-ADJ, showing the required components for proper functionality. 26
Figure 21: Circuit diagram to TPS73025DBVR, showing the required components for proper functionality. 26
Figure 22: Circuit diagram to ISL21080DIH309Z-TK, including a noise vs. frequency graph, showing the required components for proper functionality, and how they affect the circuit's noise characteristics. 27
Figure 23: Component placement for ALOT..... 29
Figure 24: A) Simplification of LVDS driver and receiver. B) Differential signal. 48
Figure 25: Differential signal propagating from the transmitter to the receiver. 49

Bibliography

[1] Wkipedia, "Terrestrial Gamma-ray Flash," 5 March 2019. [Online]. Available: https://en.wikipedia.org/wiki/Terrestrial_gamma-ray_flash. [Accessed April 2019].

[2] X. Ramus, "Transimpedance Considerations For High-Speed Amplifiers," November 2009. [Online]. Available: http://www.ti.com/lit/an/sboa122/sboa122.pdf. [Accessed 24 May 2019].

[3] ElectronicsTutorials, "Second Order Filters," [Online]. Available: https://www.electronics-tutorials.ws/filter/second-order-filters.html. [Accessed 2019].

[4] M. Steffes, "Design Methodology for MFB Filters in ADC Interface," February 2006. [Online]. Available: http://www.ti.com/lit/an/sboa114/sboa114.pdf. [Accessed 2019].

[5] Mathworks, "Butterworth Filter Order and Cutoff Frequency," [Online]. Available: https://se.mathworks.com/help/signal/ref/buttord.html. [Accessed 2019].

[6] J. Karki, "Fully-Differential Amplifiers," September 2016. [Online]. Available: http://www.ti.com/lit/an/sloa054e/sloa054e.pdf. [Accessed April 2019].

[7] Wikipedia, "Damping Ratio," 2019. [Online]. Available: https://en.wikipedia.org/wiki/Damping_ratio. [Accessed 2019].

[8] Xilinx, "Zynq-7000 SoC Technical Reference Manual," 01 July 2018. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug585-Zynq-7000-TRM.pdf. [Accessed May 2019].

[9] S. Yang, *UiB BGO RCU board -- FEGS,* 2017.

[10] Embedded Processor Applications, "High-Speed Interface Layout Guidelines," October 2018. [Online]. Available: http://www.ti.com/lit/an/spraar7h/spraar7h.pdf. [Accessed 2019].

[11] Texas Instruments, "High Speed PCB Layout Techniques," [Online]. Available: http://www.ti.com/lit/ml/slyp173/slyp173.pdf. [Accessed May 2019].

[12] B. Olney, "Multilayer PCB Stackup Planning," [Online]. Available: https://smtnet.com/library/files/upload/Multilayer-PCB-Stackup.pdf. [Accessed April 2019].

[13] Elrpint, "Kapabiliteter," [Online]. Available: https://elprint.no/products/pcb/capabilities. [Accessed May 2019].

[14] Wikipedia, "Serial Peripheral Interface," 17 May 2019. [Online]. Available: https://en.wikipedia.org/wiki/Serial_Peripheral_Interface#Quad_SPI. [Accessed 28 May 2019].

[15] Wikipedia, "I2C," 22 May 2019. [Online]. Available: https://en.wikipedia.org/wiki/I%C2%B2C. [Accessed 28 May 2019].

[16] Wikipedia, "Low Voltage Differential Signaling," 9 April 2019. [Online]. Available: https://en.wikipedia.org/wiki/Low-voltage_differential_signaling. [Accessed 28 May 2019].

[17] National Instruments, "Understanding Lvds For Digital Test Systems," 24 May 2019. [Online]. Available: http://www.ni.com/en-my/innovations/white-papers/06/understanding-lvds-for-digital-test-systems.html. [Accessed 28 May 2019].

[18] Vectron, "Signal Types and Terminations," [Online]. Available: https://www.vectron.com/products/literature_library/Signal_Types_and_Terminations.pdf. [Accessed 29 May 2019].

[19] Siements, "Single Ended vs Differential Inputs," 7 November 2016. [Online]. Available: https://community.plm.automation.siemens.com/t5/Testing-Knowledge-Base/Single-Ended-vs-Differential-Inputs/ta-p/355022. [Accessed 28 May 2019].

[20] NASA, "Terrestrial Gamma-ray Flashes," 2011. [Online]. Available: https://svs.gsfc.nasa.gov/10706.

[21] Wikipedia, "Media Independent Interface," 14 April 2019. [Online]. Available: https://en.wikipedia.org/wiki/Media-independent_interface#Reduced_gigabit_media-independent_interface. [Accessed 28 May 2019].

[22] Cadence, "Reduced Gigabit Media Independent Interface (RGMII)," May 2004. [Online]. Available: https://opencores.org/websvn/filedetails?repname=ethernet_tri_mode&path=%2Fethernet_tri_mode%2Ftags%2Farelease%2Fdoc%2FRGMII_DataSheet.pdf. [Accessed 28 May 2019].

[23] Texas Instruments, "ADS5404," January 2014. [Online]. Available: http://www.ti.com/lit/ds/slas945b/slas945b.pdf. [Accessed May 2019].

[24] Abracon LLC, "ABM7," 08 June 2015. [Online]. Available: https://abracon.com/Resonators/abm7.pdf. [Accessed April 2019].

[25] Analog Devices, "AD9577," August 2016. [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ad9577.pdf. [Accessed May 2019].

[26] Analog Devices, "AD7490," [Online]. Available: https://www.analog.com/media/en/technical-documentation/data-sheets/ad7490.pdf. [Accessed Mai 2019].

[27] Abracon, "RJ45 Connector," 23 12 2013. [Online]. Available: https://abracon.com/Magnetics/lan/ARJC01-111002T.PDF. [Accessed May 2019].

[28] Trimble, "Copernicus," July 2011. [Online]. Available: http://www.dpie.com/manuals/gps/CopernicusIIRefMan_1A_68340-06-ENGa.pdf. [Accessed April 2019].

[29] Connor Winfield, "Surface Mount CMOS Clock Oscillator Series," 07 September 2011. [Online]. Available: http://www.conwin.com/datasheets/sm/sm126.pdf. [Accessed April 2019].

[30] Texas Instruments, "DP83822," March 2019. [Online]. Available: https://www.ti.com/lit/ds/symlink/dp83822i.pdf. [Accessed May 2019].

[31] Texas Instruments, "INA190," September 2018. [Online]. Available: https://www.ti.com/lit/ds/symlink/ina190.pdf. [Accessed May 2019].

[32] Micron, "DDR3L SDRAM," September 2013. [Online]. Available: https://www.micron.com/-/media/Documents/Products/Data%20Sheet/DRAM/DDR3/4Gb_1_35V_DDR3L.pdf. [Accessed May 2019].

[33] Texas Instruments, "OPA855," October 2018. [Online]. Available: http://www.ti.com/lit/ds/symlink/opa855.pdf. [Accessed May 2019].

[34] Cypress, "SPI Flash Memory," 30 April 2019. [Online]. Available: https://www.cypress.com/file/177961/download. [Accessed May 2019].

[35] Texas Instruments, "THS4303," January 2005. [Online]. Available: https://www.ti.com/lit/ds/symlink/ths4303.pdf. [Accessed May 2019].

[36] Texas Instruments, "THS4505," May 2008. [Online]. Available: http://www.ti.com/lit/ds/symlink/ths4504.pdf. [Accessed May 2019].

[37] Maxim, "High Speed Logic-Level Translator," January 2007. [Online]. Available: https://datasheets.maximintegrated.com/en/ds/MAX13030E-MAX13035E.pdf. [Accessed March 2019].

[38] Xilinx, "Selection Guide," [Online]. Available: https://www.xilinx.com/support/documentation/selection-guides/zynq-7000-product-selection-guide.pdf. [Accessed 2019].

[39] Pcbcart, "Layer Stackup," [Online]. Available: https://www.pcbcart.com/pcb-capability/layer-stackup.html. [Accessed 2019].