



Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich



Institut für  
Technische Informatik und  
Kommunikationsnetze

# Event-based Geophone Platform with Co-detection

Master Thesis

**Akos Pasztor**

<https://akospasztor.com>

Computer Engineering and Networks Laboratory  
Department of Information Technology and Electrical Engineering  
ETH Zürich

**Advisors:**

Dr. Jan Beutel  
Matthias Meyer

**Professor:**

Prof. Dr. Lothar Thiele

May 31, 2018

*This page intentionally left blank.*

# Abstract

Between 6% and 8% of Switzerland's surface area is unstable. The increasing number of landslides and rockfalls resulted in disasters emphasized the necessity of early detection. Consequently, micro-seismic monitoring has become more important than ever. Current sensing systems are costly and most of them rely on regular interaction and maintenance, which prevents long-term autonomous operation. This thesis presents a state-of-the-art, networked, event-based micro-seismic sensing platform with co-detection, wireless sensor network integration, ultra-low power consumption and comprehensive remote management with the guarantee of fully autonomous operation for several years. This work covers the entire platform development process from detailed planning all the way to production, including complete hardware design, prototyping, embedded software development, network integration and extensive platform evaluation.

*This page intentionally left blank.*

# Acknowledgements

I would like to express my sincere gratitude to my supervisor, Dr. Jan Beutel for the continuous support of my master thesis. His guidance, motivation and broad knowledge helped me throughout my work. I would also thank Matthias Meyer for co-advising my thesis. Beside my advisors, a special thank goes to Reto Da Forno and Tonio Gsell who were constructively challenging me, providing professional support and overall being great colleagues. I would like to thank the Computer Engineering Group for providing me a personal workplace with access to laboratories and equipment. My sincere thank goes to Prof. Dr. Lothar Thiele, without whom this project would not have been possible.

*This page intentionally left blank.*

# Contents

<b>Abstract</b>	<b>iii</b>
<b>Acknowledgements</b>	<b>v</b>
<b>Contents</b>	<b>vii</b>
<b>List of Figures</b>	<b>xi</b>
<b>List of Tables</b>	<b>xv</b>
<b>Abbreviations</b>	<b>xvii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Contribution . . . . .	2
1.3 Outline . . . . .	2
<b>2 Background</b>	<b>3</b>
2.1 Wireless Sensor Networks . . . . .	3
2.2 PermaSense Consortium . . . . .	3
2.2.1 PermaSense Network Architecture . . . . .	4
2.3 Dual Processor Platform . . . . .	5
2.3.1 BOLT - A Stateful Processor Interconnect . . . . .	5
2.3.2 eLWB - Event-Based Low-Power Wireless Bus . . . . .	6
2.3.3 DPP Communication Board . . . . .	6
2.4 Seismic Recording Systems . . . . .	7
2.4.1 Micro-Seismic Monitoring . . . . .	8
2.4.2 Sensor Technologies . . . . .	8
<b>3 System Specification</b>	<b>11</b>
3.1 Pilot Study . . . . .	11
3.1.1 Setup . . . . .	11
3.1.2 Experiment Evaluation . . . . .	13
3.2 System Requirements . . . . .	14
3.2.1 Sensor Requirements . . . . .	14
3.2.2 Functional Requirements . . . . .	14
3.2.3 Network Requirements . . . . .	15
3.2.4 Mechanical Requirements . . . . .	15
3.2.5 From Planning to Production . . . . .	15

<b>4</b>	<b>Triggering and Co-detection</b>	<b>17</b>
4.1	Concept . . . . .	17
4.2	Triggering Methodologies . . . . .	18
4.2.1	Triggering in Analog Domain . . . . .	18
4.2.2	Triggering in Digital Domain . . . . .	20
4.3	Initial Triggering Circuit . . . . .	20
4.4	Implementation of Triggering Circuit . . . . .	21
4.4.1	Processing Unit . . . . .	24
4.4.2	External Components . . . . .	24
4.4.3	Power Supply . . . . .	25
4.4.4	Printed Circuit Board Design . . . . .	25
4.4.5	Signal Paths . . . . .	27
4.5	Evaluation of Triggering Circuit . . . . .	27
4.5.1	Embedded Scenario . . . . .	27
4.5.2	Mixed Scenario 1 . . . . .	28
4.5.3	Mixed Scenario 2 . . . . .	29
4.5.4	External Scenario . . . . .	30
4.5.5	Comparing Results . . . . .	31
4.6	Improved Triggering Circuit . . . . .	31
4.6.1	Dual-Side Triggering . . . . .	32
4.6.2	Multi-Stage Triggering . . . . .	33
4.6.3	Realization of the Improved Triggering Circuit . . . . .	35
4.7	Co-detection . . . . .	36
<b>5</b>	<b>Hardware Design</b>	<b>37</b>
5.1	System Architecture . . . . .	37
5.2	Microcontroller Subsystem . . . . .	39
5.2.1	The Microcontroller . . . . .	39
5.2.2	Microcontroller Circuit . . . . .	40
5.2.3	Storage . . . . .	41
5.2.4	Miscellaneous Components . . . . .	42
5.3	Triggering Subsystem . . . . .	42
5.4	Digitizer Subsystem . . . . .	44
5.4.1	Analog-to-Digital Converter . . . . .	44
5.4.2	Voltage Reference . . . . .	45
5.4.3	Signal Path . . . . .	46
5.5	Sensor Subsystem . . . . .	47
5.5.1	Inertial Measurement Unit . . . . .	47
5.5.2	Temperature and Humidity Sensor . . . . .	48
5.5.3	Battery Voltage Monitoring . . . . .	48
5.6	Communication and Networking Subsystem . . . . .	50
5.7	Power Supply Subsystem . . . . .	52
5.7.1	Battery . . . . .	53
5.7.2	Voltage Regulator . . . . .	54



---

5.8	Ultra-Low Power Hardware Design . . . . .	55
5.9	Mechanics . . . . .	56
5.9.1	Enclosure . . . . .	56
5.9.2	Geophone Clamp . . . . .	56
5.10	Prototyping . . . . .	57
<b>6</b>	<b>Software Design</b> . . . . .	<b>61</b>
6.1	Major Software Features . . . . .	61
6.1.1	Modified RTOS . . . . .	61
6.1.2	Precise Time Synchronization . . . . .	61
6.1.3	Fast Wakeup . . . . .	62
6.1.4	Extensive Logging . . . . .	62
6.1.5	Flexible Configuration . . . . .	62
6.1.6	Network Integration . . . . .	63
6.1.7	Self-Test . . . . .	63
6.1.8	Self-Monitoring . . . . .	63
6.1.9	Production Hardware Test . . . . .	63
6.1.10	Connectivity . . . . .	64
6.1.11	Customizable Bootloader . . . . .	64
6.2	Software Components . . . . .	64
6.2.1	Low-Level Drivers . . . . .	64
6.2.2	Middleware . . . . .	65
6.2.3	Custom Application Libraries . . . . .	66
6.2.4	Geophone Bootloader . . . . .	67
6.3	Software Architecture . . . . .	69
6.3.1	Initialization Sequence . . . . .	69
6.3.2	RTOS Tasks . . . . .	71
6.3.3	Interrupts . . . . .	76
6.4	Ultra-Low Power Software Design . . . . .	77
6.5	Configuration . . . . .	79
6.6	GSN Integration . . . . .	80
6.6.1	Network Messages . . . . .	80
6.6.2	Network Commands . . . . .	82
<b>7</b>	<b>Evaluation</b> . . . . .	<b>85</b>
7.1	Analog Frontend . . . . .	85
7.1.1	Stage Gains . . . . .	85
7.1.2	Frequency Responses . . . . .	87
7.2	Triggering Attributes . . . . .	90
7.2.1	Threshold Voltages . . . . .	90
7.2.2	Trigger Propagation . . . . .	91
7.3	Wakeup Time Analysis . . . . .	92
7.4	Power Analysis . . . . .	92
7.4.1	Consumption of Components in Low Power State . . . . .	93

---

7.4.2	Measurement Scenarios . . . . .	95
7.4.3	Battery Life Estimation . . . . .	101
7.5	Triggering and Load Test . . . . .	102
7.6	Temperature Test . . . . .	103
7.7	Software Performance Analysis . . . . .	104
7.8	Field Test . . . . .	108
7.8.1	Test Setup . . . . .	108
7.8.2	Test Scenarios . . . . .	109
7.8.3	Test Evaluation . . . . .	110
7.9	Production . . . . .	111
7.10	Geophone Platform Datasheet . . . . .	111
<b>8</b>	<b>Conclusion</b>	<b>115</b>
8.1	Media Coverage . . . . .	115
8.2	Outlook . . . . .	116
	<b>Bibliography</b>	<b>117</b>
<b>A</b>	<b>Repository Organization</b>	<b>121</b>
<b>B</b>	<b>Geophone Schematics</b>	<b>123</b>
<b>C</b>	<b>Geophone PCB</b>	<b>131</b>
<b>D</b>	<b>Additional Power Traces</b>	<b>141</b>
<b>E</b>	<b>RTOS Traces</b>	<b>145</b>
<b>F</b>	<b>Field Test Data</b>	<b>151</b>

# List of Figures

2.1	PermaSense architecture . . . . .	4
2.2	DPP communication board . . . . .	6
2.3	General seismic recording system architecture . . . . .	7
2.4	Geophone structure . . . . .	8
3.1	Preliminary hardware . . . . .	12
3.2	Deployment of preliminary hardware . . . . .	12
4.1	Analog signal path . . . . .	17
4.2	Amplitude triggering . . . . .	19
4.3	Amplitude triggering block diagram . . . . .	20
4.4	Triggering circuit implementation . . . . .	23
4.5	PCB design of triggering circuit board . . . . .	26
4.6	Photo of triggering circuit board . . . . .	26
4.7	Triggering: embedded scenario . . . . .	27
4.8	Triggering: mixed scenario with external amplifier . . . . .	28
4.9	Triggering: mixed scenario with external amplifier and DAC . . . . .	29
4.10	Triggering: external scenario . . . . .	30
4.11	Dual-side triggering . . . . .	32
4.12	Multi-stage triggering . . . . .	33
4.13	Improved triggering circuit schematics . . . . .	35
5.1	System architecture . . . . .	37
5.2	System core - microcontroller domain . . . . .	39
5.3	Triggering domain . . . . .	43
5.4	Digitizer domain . . . . .	46
5.5	Sensor domain . . . . .	47
5.6	Voltage sensing circuit . . . . .	49
5.7	Communication domain . . . . .	51
5.8	Power supply subsystem . . . . .	52
5.9	Custom-designed, 3D-printed geophone clamp . . . . .	57
5.10	Empty printed circuit boards . . . . .	58
5.11	Prototype . . . . .	58
5.12	Development . . . . .	59
6.1	Microcontroller flash organization . . . . .	68
6.2	Software architecture . . . . .	70
6.3	Software initialization sequence . . . . .	71

6.4	RTOS task priorities . . . . .	72
6.5	Logging commands and message structure . . . . .	75
6.6	Interrupts and their priorities . . . . .	77
6.7	Node info packet . . . . .	80
6.8	App health packet . . . . .	81
6.9	Geophone acquisition packet . . . . .	81
6.10	Inertial measurement unit data packet . . . . .	82
6.11	Network command structure . . . . .	82
6.12	Sending command from GSN backend . . . . .	83
7.1	Single-stage gain . . . . .	86
7.2	Dual-stage gain . . . . .	87
7.3	Frequency response of single stage . . . . .	89
7.4	Frequency response of dual stage . . . . .	90
7.5	Trigger propagation . . . . .	91
7.6	Triggering and 3-second recording without COM board . . . . .	96
7.7	Triggering and 3-second recording with COM board . . . . .	98
7.8	Periodic generation of status and health data . . . . .	99
7.9	Network command reception and execution . . . . .	100
7.10	Radio transmission . . . . .	101
7.11	Visualization of geophone samples . . . . .	103
7.12	Temperature test . . . . .	104
7.13	Geophone samples during temperature test . . . . .	104
7.14	Trace of a 3-second recording . . . . .	105
7.15	Beginning of a 3-second recording trace . . . . .	106
7.16	End of a 3-second recording trace . . . . .	107
7.17	Trace of a periodic health wakeup . . . . .	107
7.18	Field test location . . . . .	108
7.19	Data visualization of a rock thrown from 2 meters . . . . .	110
7.20	Data visualization of shoveling . . . . .	111
8.1	Video featuring the geophone platform on SRF . . . . .	116
A.1	Repository organization . . . . .	121
D.1	Triggering and 3s recording without COM board, without IMU . . . . .	141
D.2	Triggering and 3s recording with COM board, without IMU . . . . .	142
D.3	Power trace of a radio beacon . . . . .	143
E.1	Trace of a 3-second recording . . . . .	146
E.2	Beginning of a 3-second recording trace . . . . .	147
E.3	End of a 3-second recording trace . . . . .	148
E.4	Trace of a periodic health wakeup . . . . .	149
F.1	Data visualization of initialization and enclosure screwing . . . . .	151

---

F.2	Data visualization of manual synchronization with hammer . . . . .	151
F.3	Visualization of a heavy rock thrown from 5m distance into the pit .	152
F.4	Visualization of a heavy rock thrown next to node into the pit . . .	152
F.5	Rock rolling down the pit after hitting the slope . . . . .	153

*This page intentionally left blank.*

# List of Tables

4.1	Consumption of embedded scenario . . . . .	28
4.2	Consumption of mixed scenario 1 . . . . .	29
4.3	Consumption of mixed scenario 2 . . . . .	30
4.4	Consumption of external scenario . . . . .	31
5.1	Switching versus linear regulators . . . . .	54
7.1	Wakeup times . . . . .	93
7.2	Summary of power analysis in low power state . . . . .	95
7.3	Configuration parameters for 1 <sup>st</sup> scenario . . . . .	95
7.4	Configuration parameters for 2 <sup>nd</sup> scenario . . . . .	97
7.5	Electrical characteristics . . . . .	112
7.6	Storage parameters . . . . .	112
7.7	Sensor parameters . . . . .	113
7.8	Configuration parameters . . . . .	113
7.9	Network parameters . . . . .	114
7.10	Connectivity . . . . .	114
7.11	Mechanical parameters . . . . .	114

*This page intentionally left blank.*



# Abbreviations

<b>ADC</b>	Analog-to-Digital Converter
<b>API</b>	Application Programming Interface
<b>CMSIS</b>	Cortex Microcontroller Software Interface Standard
<b>DAC</b>	Digital-to-Analog Converter
<b>DPP</b>	Dual-Processor Platform
<b>ESD</b>	Electronic Static Discharge
<b>GPIO</b>	General Purpose Input/Output
<b>IMU</b>	Inertial Measurement Unit
<b>IoT</b>	Internet of Things
<b>ISR</b>	Interrupt Service Routine
<b>IT</b>	Interrupt
<b>LDO</b>	Low-Dropout
<b>LPM</b>	Low Power Mode
<b>MCU</b>	Microcontroller Unit
<b>OpAmp</b>	Operational Amplifier
<b>OTA</b>	Over-The-Air
<b>PCB</b>	Printed Circuit Board
<b>PGA</b>	Programmable Gain Amplifier
<b>RTC</b>	Real Time Clock
<b>SMU</b>	Source Measure Unit
<b>SNR</b>	Signal-to-Noise Ratio
<b>SoC</b>	System-on-Chip
<b>SPI</b>	Serial Peripheral Interface
<b>UART</b>	Universal Asynchronous Receiver/Transmitter
<b>ULP</b>	Ultra-Low Power
<b>WSN</b>	Wireless Sensor Network

*This page intentionally left blank.*

# Introduction

---

It is estimated between 6% and 8% of Switzerland's surface area is unstable, as recently reported by the Swiss National Platform for Natural Hazards. In addition to climate change, several factors affect the stability of land surfaces and rock walls in the areas of mountain permafrost. Consequently, special attention must be paid to the increasing number of hazardous activities, including landslides, debris flows, rock falls and rock avalanches. These commotions are often anticipated by increased micro-seismic activities. Henceforth, accurately monitoring micro-seismic activities is crucial in order to predict disasters and save lives.

## 1.1 Motivation

Micro-seismic monitoring is a recent research area evolved from traditional earthquake seismology. Micro-seismic activities manifest in usually weak signals, thus sophisticated, high-quality engineering is required to detect and record usable micro-seismic data. Current approaches are based on continuous sampling with non-autonomous systems that require periodic, physical interaction. These devices are often heavy and bulky, they tend to have high energy consumption, furthermore their long-term operation is costly.

This thesis presents a state-of-the-art, networked, event-based micro-seismic sensing platform with co-detection, wireless sensor network integration, ultra-low energy consumption and comprehensive remote management together with the guarantee of fully autonomous operation for several years.

## 1.2 Contribution

The contribution of this thesis can be summarized as follows:

1. The preliminary experiment performed by the Department of Geology was evaluated in detail.
2. A state-of-the-art micro-seismic triggering evaluation circuit was designed, assembled and evaluated.
3. The platform specification was set up based on the results of the preliminary experiment and the evaluation of the triggering circuit.
4. The platform hardware was designed focusing on the system requirements, furthermore three prototypes were assembled and debugged.
5. The developed embedded software features a modified real-time operating system kernel enhanced for fail-safe and ultra-low power operation.
6. The platform was integrated into an existing wireless sensor network.
7. Extensive system evaluation was conducted with detailed power analysis.
8. Launched and supervised the mass-production of 50 pieces.

## 1.3 Outline

Chapter 2 presents the background of the research and used technologies. Chapter 3 describes the system specification. In Chapter 4 the design of a state-of-the-art triggering circuit is presented in detail. Chapter 5 describes the complete platform hardware design. The developed embedded software is presented in Chapter 6. Extensive platform evaluation and analysis results are demonstrated in Chapter 7. Finally, Chapter 8 contains the summary of this thesis.

The appendix contains the platform design files together with the evaluation results. Furthermore, the digital project repository contains all used material for this work, including literature, discussion transcripts, CAD projects, embedded software source code, analysis results, evaluation data, and presentation slides.

# Background

---

This chapter presents the thesis background, focusing on wireless sensor networks (WSN) and the PermaSense WSN and the dual-processor platform (DPP) with BOLT. Furthermore, this chapter introduces the micro-seismic sensing in general.

## 2.1 Wireless Sensor Networks

A traditional wireless sensor network consists of individual devices called nodes with wireless communication capability. The nodes are organized to form a wireless network. Generally, the nodes are responsible for sensing and recording data which is then transmitted to the network wirelessly. Typically the data packets are collected by dedicated sink nodes which communicate directly or indirectly with the network backend. The backend usually consists of storage units and databases where the packets are stored. Users can connect to the backend generally via internet by using computers and smartphone applications, where data collected from the nodes can be queried and visualized, furthermore commands can be issued remotely to change the configuration and behavior of the nodes.

## 2.2 PermaSense Consortium

The PermaSense consortium comprises multiple research projects from Swiss universities and companies, focusing on environmental research of climate, cryosphere, geomorphodynamics and the connection between them. As a result, multiple wireless sensing systems have been developed specifically for autonomous, high-alpine environment [1]. The PermaSense system focuses on ultra-low power and



## 2.3 Dual Processor Platform

The Dual Processor Platform (DPP) is a novel system architecture focusing on limited component interference and composable construction [2]. In event-triggered systems, concurrently active components may interfere with each other regarding computation, memory and peripherals. Therefore, the limitation of possible resource interferences is crucial in event-based, ultra-low power systems. Additionally, the properties, including adaptability, responsiveness and energy efficiency of each event-triggered module must be preserved. Consequently, different components have to be interconnected without affecting their properties.

In order to meet these requirements in the case of a traditional wireless sensor node, the tasks related to sensing and processing are mapped to a dedicated application processor and the wireless communication is handled by a dedicated communication processor. The two processors are interconnected with BOLT.

### 2.3.1 BOLT - A Stateful Processor Interconnect

BOLT is a state-of-the-art, ultra-low power processor interconnect with bi-directional, asynchronous message transfer and predictable run-time behavior [3]. By using BOLT, two processors can be decoupled with regard to time, clock and power domains, enabling them to concurrently execute their assigned tasks without resource interference.

The realization of BOLT is a piece of integrated hardware and software that interconnects the application and the communication processor. In BOLT, dedicated communication peripherals, including control and data channels are assigned to each processor, therefore both processors can transfer messages simultaneously. BOLT is implemented with a 16-bit MSP430FR5969 microcontroller [4]. This MCU is equipped with a ferro-electric random access memory (FRAM), which enables preserving context even when the power is completely switched off. The messages are stored in two FIFO queues. One serves messages from the application to the communication processor and the other serves messages from the communication processor to the application processor. BOLT consumes  $1.1mW$  in active mode and  $1.3\mu W$  in sleep mode when there is no communication request from either of the processors.

### 2.3.2 eLWB - Event-Based Low-Power Wireless Bus

The Event-based Low-power Wireless Bus (eLWB) is based on the open-source Low-power Wireless Bus (LWB) [5]. The LWB represents a logical shared bus for multi-hop wireless topologies using time-slotted Glossy-floods [6]. The LWB uses periodic communication rounds which consist of scheduling, contention, data and acknowledgement slots. The Low-power Wireless Bus is greatly suitable for periodic transmissions with slowly changing traffic demands, however the LWB performs less optimally in event-triggered wireless applications. Consequently, a modified version called Event-based Low-power Wireless Bus (eLWB) is proposed in [2]. The eLWB reduces the wakeup time and supports simultaneous allocations of aperiodic and variable bandwidth to the wireless nodes. The eLWB supports high-precision time synchronization and on-demand event dissemination, while providing highly reliable operation.

### 2.3.3 DPP Communication Board

The DPP communication board is a standardized module equipped with BOLT and a TI CC430 radio system-on-chip (SoC) [7]. The COM board can be connected to an application board with a board-to-board connector with standardized pinout, thus enabling multiple types of COM boards to be attached to application boards. The DPP COM board has a compact size of 44 x 24 mm, enabling easy integration to various applications.

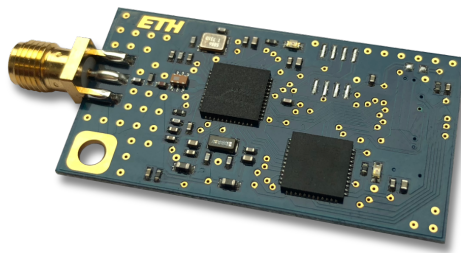


Figure 2.2: DPP communication board

The DPP communication boards running the eLWB communication protocol are used in the geophone platform. Additionally, the platform completely sup-



ports the newly developed communication boards equipped with LoRa<sup>1</sup> transceivers.

## 2.4 Seismic Recording Systems

Seismic activity can be detected by sensors that are capable of measuring ground movement. As technology evolved, analog solutions have been replaced with modern, digital systems. These modern seismic recording systems are required to measure, digitize and take samples of seismic sensors, store the recordings to storage units and operate autonomously [8].

A general seismic recording system architecture is depicted in Figure 2.3.

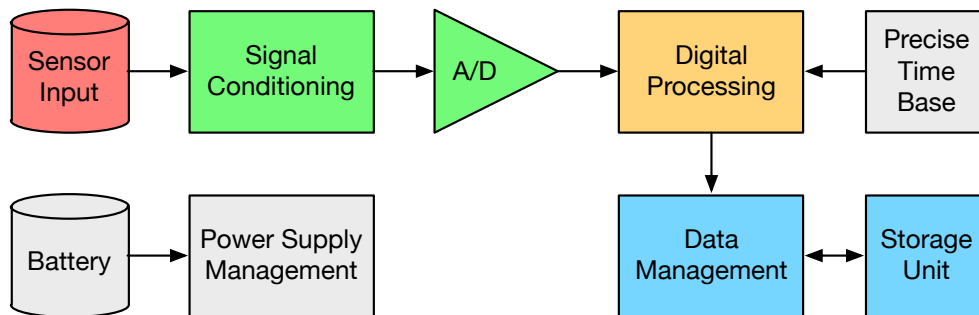


Figure 2.3: General seismic recording system architecture

First, the signal coming from the seismic sensor needs to be conditioned. This may include pre-amplification and active or passive signal filtering. Then the conditioned signal is digitized using a suitable analog-to-digital converter (ADC) unit. Afterwards, the signal is processed in the digital domain and the samples are stored on a storage unit. The samples must be logged with accurate timestamps, thus precise time has to be provided and maintained with regular time synchronization. A modern recording system is compact and portable, therefore it is required to be battery-powered. Consequently, a recording system needs to be energy-efficient without sacrificing recording quality and performance.

<sup>1</sup> LoRa - <https://en.wikipedia.org/wiki/LoRa>

### 2.4.1 Micro-Seismic Monitoring

Micro-seismic monitoring has evolved from traditional earthquake seismology, focusing on detecting micro-earthquakes caused by human industrial activity or natural events, including rock and glacier movements. It uses passive methods in contrast to traditional seismic technologies. Detecting micro-seismic activities is a challenging task, therefore it requires precise and sophisticated sensing system.

### 2.4.2 Sensor Technologies

Several methods can be used for detecting micro-seismic activities. The most popular approaches use geophones, accelerometers, piezoelectric sensors and broadband seismometers. However, all methods have their own performance limitations as described in [9].

#### Geophones

Geophones are passive analog devices equipped with a spring-mounted mass which moves within a wire coil and a magnet. As the magnet moves relative to the wire, the changing magnetic field induces voltage. The generated voltage is proportional to the ground velocity. Analog geophones are highly sensitive devices which can detect distant seismic activities, thus making them a suitable choice for micro-seismic recording systems.

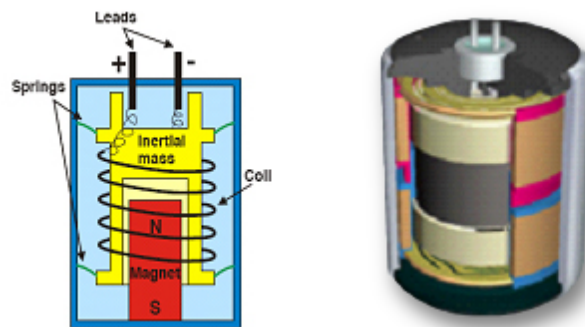


Figure 2.4: Geophone structure

### **Accelerometers**

Generally, accelerometers convert mechanical motions into electrical signals using commonly piezoelectric, piezoresistive and capacitive components. Piezoelectric accelerometers usually employ piezoceramics or single crystals. They operate mostly in higher frequency ranges, thus they are preferred devices for detecting high impact shocks. Micro-electro-mechanical systems (MEMS) technology enables the fabrication of tiny, low-cost capacitive accelerometers. They operate based on the capacitance change in the mass caused by acceleration. In contrast to geophones, accelerometers respond proportionally to acceleration. Additionally, they have less sensitivity and higher noise level which limit their use for detecting only stronger motions.

### **Broadband Seismometers**

Although broadband seismographs can record a broad range of frequencies, they are not suitable for portable micro-seismic recording systems. They are often large, bulky, furthermore they require complex, sophisticated electronics. Additionally, broadband seismometers are extremely sensitive to the environment. Uneven changes in surrounding pressure can easily affect air density, resulting in unwanted, spurious signals. Therefore, these devices have to be sealed in their enclosure, making development and manufacturing cumbersome.

*This page intentionally left blank.*

# System Specification

---

This chapter presents the system specification and the requirements of the geophone platform, based on a pilot study.

## 3.1 Pilot Study

A preliminary experiment was carried out during the summer of 2017 by the Department of Geography at University of Zürich (UZH).

### 3.1.1 Setup

A single, initial initial measurement device is built with off-the-shelf components. Up to six SM-6 geophones from ION [10] can be connected externally to the preliminary device. These geophones are suitable for single-direction measurements with a maximum tilt angle of  $25^\circ$ . They have  $375\Omega$  standard coil resistance and  $28.8V/m/s$  open-circuit sensitivity. Each of the geophones are enclosed in a small waterproof barrel with a pre-amplification circuit and the barrels are connected to the main device enclosure with  $25m$  long cables.

The main enclosure contains a baseboard where further signal conditioning and amplification is performed. Additionally, a rudimentary triggering circuit is incorporated to the baseboard. The circuit features a symmetrical triggering threshold that can be adjusted manually with a single potentiometer. The baseboard is connected to an Arduino M0 module<sup>1</sup> which contains an ARM Cortex-M0+ core microcontroller from Atmel. The embedded ADC peripheral of the MCU is used for digitizing the signals coming from the geophones. Additionally, an extension shield is connected to the Arduino board. This board

---

<sup>1</sup> Arduino M0 R3 - <https://store.arduino.cc/arduino-m0>



Figure 3.1: Preliminary hardware

contains a microSD card slot for data storage and a GPS module from SkyTraq which provides precise time for the system. The device is powered by a  $6VDC$ ,  $12Ah$  sealed lead battery from Ultracell.

The preliminary measurement device was deployed to the Swiss mountains. The barrels containing the geophones were installed and placed around the main device enclosure, with a maximum distance of 20 meters. The sensors were covered by large stones to provide protection from rain and environmental effects.

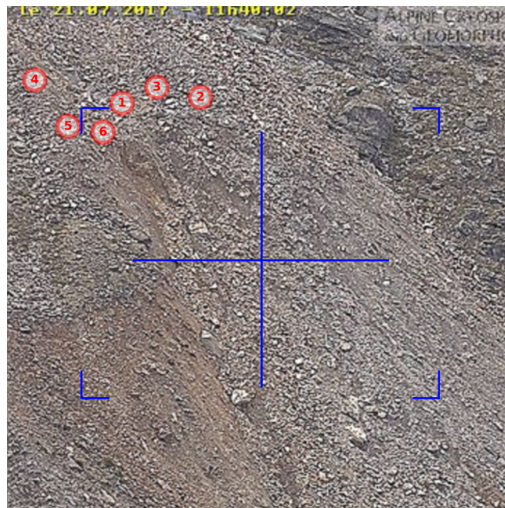


Figure 3.2: Deployment of preliminary hardware

### 3.1.2 Experiment Evaluation

The device is designed to run continuously, without utilizing any low-power methodologies to save energy. After a connected sensor triggers, i.e. the measured signal amplitude is greater than the manually set threshold, the device starts recording data from all sensors and it logs the collected data to the SD card. The waveforms are recorded for a fixed, one second duration, provided none of the geophone signals exceed the triggering threshold. If the amplitude of a signal exceeds the threshold during this one second interval, the recording is extended for another second. All signals are digitized with the built-in ADC of the microcontroller and the samples are stored on the SD card. The device time is synchronized to the GPS time once per day, at midnight. The measured power dissipation of the device is  $212.5mW$  in idle mode. Furthermore, the average consumption raises significantly as the number of recordings increase.

Although the overall operation of the preliminary device was satisfactory during the preliminary test, the system has several shortcomings:

- Energy consumption: even if the device remains in idle mode and does not perform recordings, the equipped battery completely discharges approximately after 15 days of operation.
- Triggering: the triggering threshold has to be set manually with a potentiometer which does not guarantee precise threshold values, furthermore external effects (e.g. vibrations, shocks) can accidentally re-configure the presets. Additionally, the device is unable to provide independent configurability of positive and negative thresholds and the microcontroller cannot differentiate if the trigger is caused by a signal exceeding the positive or negative threshold.
- Sampling frequency: the sampling frequency is fixed at  $250Hz$ .
- Recording quality: the built-in, 12-bit ADC of the microcontroller does not have sufficient precision and SNR to provide high-quality data recordings for post-processing and advanced research.
- Configurability: the configuration parameters cannot be reconfigured on-the-fly and most of the values are hard-coded into the firmware.
- Scalability: the device can record signals coming from maximum of 6 geo-

phones. If more signals are required to be recorded, multiple devices have to be installed and there are no synchronization or communication possibilities exist between devices.

- **Robustness:** the system is not weather- and waterproof. Furthermore, the inside circuitry and components are not suitable for harsh, high-alpine environment.
- **Portability:** the device is heavy and the installation is cumbersome.

## 3.2 System Requirements

After evaluating the shortcomings of the preliminary measurement device, the system requirements of an entirely new, state-of-the-art micro-seismic sensing platform were drafted together with the scientist from the Department of Geology.

### 3.2.1 Sensor Requirements

The geophone platform has to detect and record *micro-seismic activities* using geophones and inertial measurement units. Each device should contain a high-sensitivity, *omnidirectional geophone* sensor. Furthermore, each device must employ an *inertial measurement unit*, thus the platform is capable of triggering from multiple sources.

### 3.2.2 Functional Requirements

The geophone platform should feature a state-of-the-art, ultra-low power triggering circuitry that can perform *independent, dual-side triggering* with remotely configurable, high sensitivity range. Each device should consume as *low energy* as possible when no activity is detected. The devices need to operate *autonomously for several years* supplied from a single battery. Upon triggering, each node must wake up in the order of milliseconds and must start recording *high-quality samples* from the geophone. In addition to geophone samples, the devices have to perform regular, periodic system and *status measurements*. The gathered information must be *logged hierarchically* to the appropriate subfolders on the



microSD card formatted with FAT file system. Furthermore, each device should be capable of performing *activity detection* with an inertial measurement unit.

### 3.2.3 Network Requirements

The devices have to operate seamlessly not only independently but also part of a *wireless sensor network*. When connected to the network, measurement, status and event *packets should be transmitted* and forwarded to the backend. The devices must *process commands* received from the backend. Each configuration parameter must be *remotely configurable* for each node independently. Furthermore, the nodes must be capable of sub-millisecond time-synchronization to the network time.

### 3.2.4 Mechanical Requirements

The device should be *robust*, fully functional in harsh, *high-alpine environment* autonomously, without interruption. The nodes must be enclosed in a standard *metal enclosure*. The geophone sensor should be located inside the enclosure and *fixed tightly to the bottom* to provide effective transfer of micro-seismic activities from the environment to the sensor. The enclosure should feature a *metal mounting plate* in order to install the nodes onto the desired surfaces.

### 3.2.5 From Planning to Production

Based upon the initial system requirements, the entire platform was meticulously designed. The development involved diligent hardware design, including circuit and PCB design, prototyping and debugging. Furthermore, a robust, fail-safe embedded software was developed with customized RTOS featuring advanced tracing and ultra-low power support. The following chapters present the entire development process in detail.

*This page intentionally left blank.*

# Triggering and Co-detection

---

In this chapter, the triggering and co-detection of seismic signals is introduced in detail. Various triggering methods are investigated and compared to each other from the perspective of efficiency, energy consumption and their practicability. An initial triggering circuit is introduced in detail, alongside with its realization and evaluation. Finally, a more advanced, improved triggering circuit is presented which is used in the implementation of the geophone platform.

## 4.1 Concept

In order to record meaningful and high-quality micro-seismic measurement data, a sophisticated circuitry is needed alongside the path from the analog sensor until the signal is completely digitized. This often comprises precise signal conditioning and digitization circuitry. More specifically, it involves amplifier circuits, active filtering circuits, high-performance Analog-to-Digital Converter (ADC) with high Signal-to-Noise Ratio (SNR) and more.

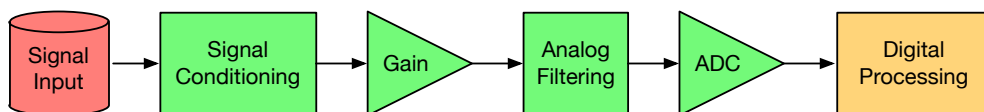


Figure 4.1: Analog signal path

Naturally, a recording device can only record data when all its required components are switched on and active. Because it is unpredictable when happens a sudden increase in seismic activities – i.e. when to turn on data recording, or when the seismic activities decrease – i.e. when to turn off data recording, the recording device needs to be always on and active in order to record all the meaningful micro-seismic activities, without missing a possibly life-saver event.

However, having the device recording constantly raises two serious issues. First, the high-quality digitization process has high energy consumption in the order of several tens or hundreds of milliwatts which would completely drain an average D-cell battery within days. Secondly, high-precision data recording requires large storage capacity. For instance, a 24-bit ADC with 1000 samples per second generates more than 7.7 GB of data over 30 days of continuous operation. In order to overcome these problems, the recording device has to be switched on immediately as micro-seismic activities increase, and it needs to be switched off after the relaxation of seismic activities to minimize energy consumption and unnecessary data recording. The process of detecting a change in seismic-activities and deciding whether it is necessary to wake up and start recording is called triggering.

## 4.2 Triggering Methodologies

To perform successful triggering, the signal from the sensor needs to be conditioned and effectively compared to a pre-defined sample (or samples). If the comparison shows salient signal parameters, the high-precision recording subsystem needs to be turned on and the signal data has to be recorded. Nevertheless, the triggering subsystem - which consists of signal conditioning and signal comparison - has to be always on, even while other subsystems are completely turned off. Therefore, the ultimate goal is to develop an always-on triggering subsystem that can effectively detect increases in micro-seismic activities with minimal energy consumption, furthermore the extent of comparison – i.e. the trigger sensitivity can be configured without making any physical changes to the device.

There are several different concepts for triggering, based on the domains that the triggering is performed in: triggering in analog or digital domain and triggering in analog or frequency domain.

### 4.2.1 Triggering in Analog Domain

Triggering in analog domain implies that the signal comparison and the wakeup decision is carried out entirely in analog domain, without involving signal digitization. Performing triggering to an analog signal can be either implemented by

comparing the amplitude of the signal to defined threshold values – this is carried out in time domain, or by checking the frequency components of the signal – this is carried out in frequency domain.

Triggering in time domain, performed as comparisons to specific amplitude threshold values can be realized with minimum circuitry and with the lowest power consumption. The disadvantage of this method is that the comparison, therefore the triggering is based on entirely on whether the momentarily signal amplitude exceeds the threshold (or thresholds), and beside the instantaneous amplitude value, there is no information about other signal parameters, e.g. frequency components, rise and fall times, etc.

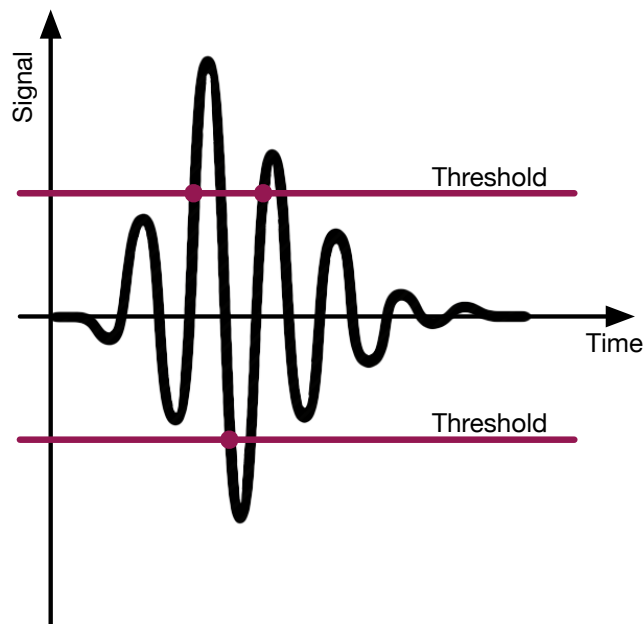


Figure 4.2: Amplitude triggering

One can perform triggering by inspecting the frequency components of the signal. This method has the advantage of having more precise information about signal and its parameters, therefore more accurate and sophisticated triggering can be implemented. However, realizing this method is more difficult and requires complex circuitry, resulting in increased energy consumption.

### 4.2.2 Triggering in Digital Domain

Triggering in digital domain denotes that the analog signal is first converted to a digital signal (i.e. digitized), then the comparison is carried out in digital domain. As in analog domain, the triggering in digital domain can be performed by either investigating amplitude or frequency parameters of the signal. The digital implementation can result in more sophisticated triggering, it is simpler to realize and requires fewer components. In fact, this method only needs a single ADC and a processing unit (often combined into a single chip). However, this method has the drawback of high power consumption, because not only digitization itself is a power-hungry task, but also additionally performing triggering in frequency-domain, i.e. performing continuous Fourier transform [11], results in significant current drain.

Considering the geophone platform system requirements and all the advantages and disadvantages of each triggering methods, one can imply that the most appropriate solution is the amplitude-triggering method performed in the analog domain. As a result, a state-of-the-art, efficient, ultra-low power consumption, high dynamic range, on-the-fly configurable triggering method is presented.

## 4.3 Initial Triggering Circuit

The goal is to build a triggering circuit that has the main functionalities of precise triggering with high dynamic range and on-the-fly configurable threshold parameters while maintaining low energy consumption. In order to achieve these requirement, the input signal needs to be amplified and compared to a threshold, i.e. to a reference value.

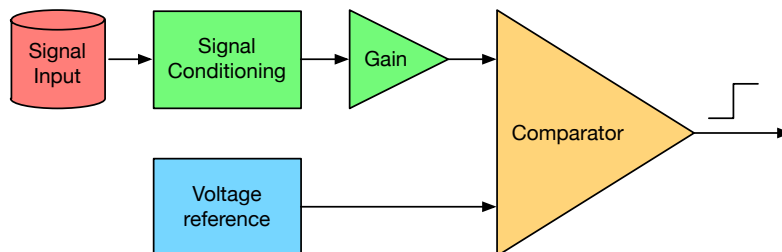


Figure 4.3: Amplitude triggering block diagram

The signal comparison itself is straightforward: a single comparator does the job perfectly, because comparators in general consume little power, furthermore almost every microcontroller has built-in comparators. On the other hand, there are different methods to make the threshold parameters configurable. First method is to make the gain of amplification variable and make the reference values fixed. The second method is to fix the amplification gain to a pre-defined value and make the reference variable. Third solution is to combine the previously mentioned two.

Implementing the first method is simple and straightforward: there are ready-made amplifier chips with built-in gain called Variable Gain Amplifiers (VGA). For instance, Analog Devices offers both analog control VGAs and digitally controlled VGAs<sup>1</sup>. As an alternative, one can make a traditional operational amplifier with variable gain by employing digital potentiometers<sup>2</sup>. Beside that digital potentiometers generally have poor temperature coefficients and tolerances, implementing a variable gain amplifier circuit with low-power components still has its power consumption in the range of milliwatts and tens of milliwatts, therefore ruling out the above described first and third methods.

Considering the second method, a variable reference has to be established. This can be implemented by using a digital potentiometer, multiplexing fixed voltage references or using a Digital-to-Analog Converter (DAC). The simplest among all is to use a DAC for multiple reasons: controlling is simple, straightforward to build, nowadays even precise DAC chips tend have low power consumption and finally most of the microcontrollers have built-in DAC. Therefore, a fixed gain operational amplifier combined with precise but low-energy consumption DAC is used in the initial triggering circuit.

## 4.4 Implementation of Triggering Circuit

The circuit realization of the above described method is presented in Figure 4.4. The incoming geophone signal first goes through basic signal filtering. Two

---

<sup>1</sup> Analog Devices: Variable Gain Amplifiers - <http://www.analog.com/en/products/amplifiers/variable-gain-amplifiers.html>

<sup>2</sup> Wikipedia: Digital potentiometers - [https://en.wikipedia.org/wiki/Digital\\_potentiometer](https://en.wikipedia.org/wiki/Digital_potentiometer)

barrier diodes function as ESD<sup>1</sup> protection and they protect the signal from transients [12]. Then the signal goes through passive filtering. The line bypass capacitors  $C_1$  and  $C_2$  suppress the common-mode noise. The across-the-line capacitor  $C_3$  suppresses the differential mode noise.

After filtering, the signal is amplified by an operational amplifier operated in differential mode. A differential amplifier amplifies the voltage applied between its inputs. The configuration is shown in Figure 4.4. The following formula gives the output voltage as a function depending on the voltage applied between the inputs:

$$V_{out} = -V_1 \left( \frac{R_3}{R_1} \right) + V_2 \left( \frac{R_4}{R_2 + R_4} \right) \left( \frac{R_1 + R_3}{R_1} \right) \quad (4.1)$$

If the resistors used “symmetrically”, i.e.  $R_1 = R_2$  and  $R_3 = R_4$ , Equation (4.1) simplifies to the following expression:

$$V_{out} = \frac{R_3}{R_1} (V_2 - V_1) \quad (4.2)$$

where  $(V_2 - V_1)$  denotes the input voltage applied between the inputs of the amplifier.

The amplification gain of the differential amplifier used in the triggering circuit is initially set to 10, because at the time of implementation, no information was given about specific voltage levels of previously recorded micro-seismic data and how to set the gain to achieve satisfactory results. To compensate this problem and provide flexible evaluation, multiple components connected in parallel are added to the amplifier and its feedback circuit. Thus, the components affecting the amplification gain can be simply changed or stacked to achieve the best results. Furthermore, the amplifier with the  $R_3$  resistor and  $C_4$  capacitor in the feedback loop act as an active low-pass filter, with the cutting frequency of:

$$f_c = \frac{1}{2\pi R_3 C_4} \approx 1kHz \quad (4.3)$$

where  $R_3 = 280k\Omega$  and  $C_4 = 560pF$ .

---

<sup>1</sup> Electronic Static Discharge



The reference voltage, acting as the triggering threshold, is produced by a low-power digital-to-analog converter with voltage output. The digital interface of the DAC is connected to the microcontroller, henceforth the threshold voltage can be changed on-the-fly. The reference signal and the amplified sensor signal is compared with a low-power comparator with built-in hysteresis to provide clean switching. The output of the comparator serves as a wakeup interrupt for the microcontroller: when the input signal exceeds the threshold value produced by the DAC, the comparator issues an interrupt to the microcontroller. The MCU then wakes up and starts the recording procedure.

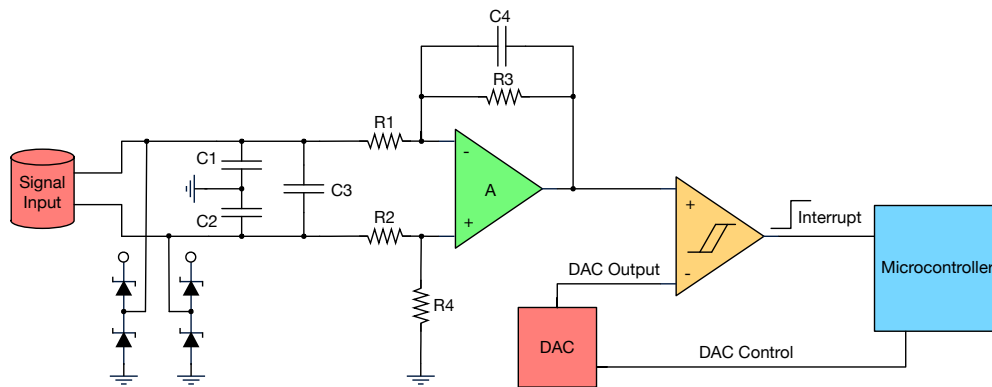


Figure 4.4: Triggering circuit implementation

The implementation of the circuit can be carried out in several ways. Increasing number of the microcontroller manufacturers embed not only digital but also analog peripherals in the microcontroller chips, including operational amplifiers, comparators, and DACs. Using the built-in peripherals can greatly simplify the design and cost of the triggering system, however their dedicated, external counterparts often have better parameters. Furthermore, using external components require more space, increase design complexity and costs. The goal is to design a triggering circuit that fulfills all triggering requirements, moreover it provides the possibility to evaluate all combinations of using embedded and external components.

The initial triggering circuit is realized with the following components:

- Processing unit with embedded amplifier, comparator and DAC
- Signal filtering circuit

- External operational amplifier
- External digital-to-analog converter
- External comparator
- Power supply circuit

#### 4.4.1 Processing Unit

The processing unit is an ultra-low power, 32-bit Cortex-M4 core microcontroller from STMicroelectronics, specifically the STM32L476VG model [13]. This microcontroller is a state-of-the-art MCU with lots of features, specifically targeted for ultra-low power applications with tons of built-in peripherals, including multiple operational amplifiers, comparators and DAC units. Furthermore, it features an advanced peripheral interconnect matrix, thus most of the embedded peripherals can be internally interconnected.

#### 4.4.2 External Components

The triggering circuit includes an external operational amplifier, an external digital-to-analog converter and an external comparator. Furthermore, the circuit features a signal protection and passive filtering circuit composing of diodes and low-pass filters.

#### Operational Amplifier

Several operational amplifiers from numerous manufacturers, including Analog Devices, Maxim Integrated and Texas Instruments are considered and compared to each other based on their performance and energy consumption. The detailed comparison can be found in the project repository. Ultimately, the ADA4051 micro-power, zero-drift rail-to-rail operational amplifier from Analog Devices is selected for the triggering circuit [14].

### **Digital-to-Analog Converter**

Numerous digital-to-analog converters from different manufacturers are compared to each other. The detailed comparison can be found in the project repository. Based on their performance and energy consumption values, the MAX5530 12-bit, ultra-low power, voltage-output DAC from Maxim Integrated is used [15].

### **Comparator**

Comparators in general consume little power, thus any comparator with sufficient internal hysteresis and sub-microwatt energy consumption is a suitable choice. The MAX919 nanopower, beyond-the-rails comparator from Maxim Integrated is selected for the triggering circuit [16].

#### **4.4.3 Power Supply**

The main objectives of the power supply design are to provide stable, noise-free system voltage with minimal quiescent current, to provide the possibility to precisely measure energy consumption of the circuit both with and without taking the consumption of DC/DC conversion into account, and finally to have the ability to easily power the circuit with a single USB cable.

The LP5907 low-quiescent current, low-noise low-dropout regulator is selected from Texas Instruments [17] to provide stabilized system voltage for the whole circuit. The system voltage is chosen to be 3.3V DC. Additional breakout pins are added for connecting external power supply and for performing precise energy consumption measurements.

#### **4.4.4 Printed Circuit Board Design**

The proposed triggering circuit is designed in Altium Designer. The layout of the Printed Circuit Board (PCB) is drawn in a way that components can be easily soldered and replaced by hand to perform experiments and achieve the best evaluation results. All major connection points, i.e. inputs and outputs, have a breakout pin to interconnect them manually with other components.

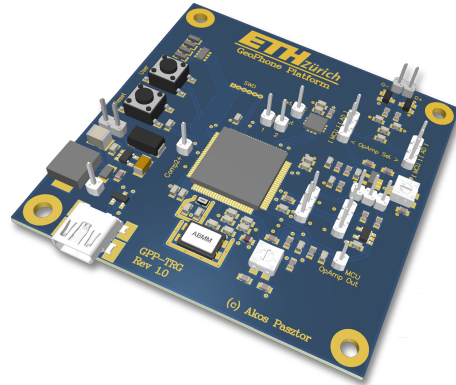


Figure 4.5: PCB design of triggering circuit board

After designing the triggering circuit board, five pieces were manufactured. All the components were soldered by hand. The design and manufacturing files with the bill of materials can be found in the project repository.

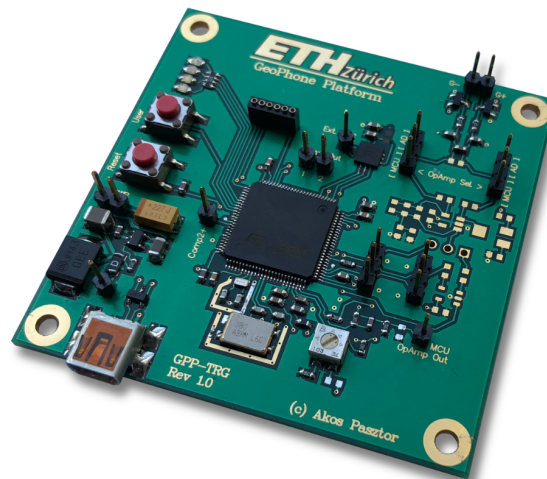


Figure 4.6: Photo of triggering circuit board

### 4.4.5 Signal Paths

The designed triggering circuit board provides several combinations of using embedded, external or both types of components for triggering:

1. Embedded: internal OpAmp, internal DAC, internal comparator
2. Mixed: external OpAmp, internal DAC, internal comparator
3. Mixed: external OpAmp, internal DAC, external comparator
4. External: external OpAmp, external DAC, external comparator

The ultimate goal is to evaluate and compare all signal paths to each other with respect to power consumption and signal quality.

## 4.5 Evaluation of Triggering Circuit

For precise evaluation, all printed circuit boards have different assembly, focusing on the individual signal paths. The printed circuit boards were assembled sequentially, i.e. assembling subsystems after subsystems, thus providing accurate power analysis of each sub-circuit.

### 4.5.1 Embedded Scenario

In this scenario, only the microcontroller, its required external components (e.g. programming connector, crystals and passives) and the required external components for the embedded operational amplifier are soldered onto the board.

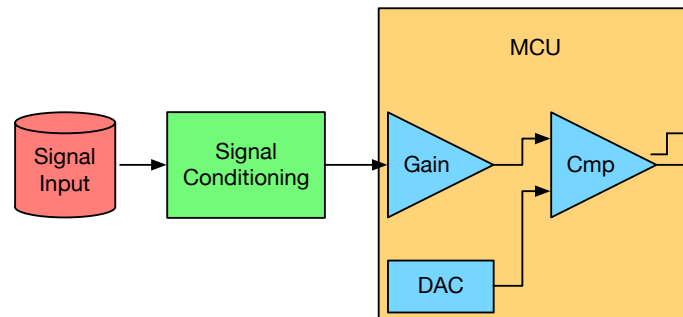


Figure 4.7: Triggering: embedded scenario

Table 4.1 summarizes the results of power analysis of this scenario. The lowest low-power mode of the microcontroller that allows all required peripherals to be operational for this scenario is the STOP 1 mode. Additionally, the RTC remains active and driven by LSE<sup>1</sup> when the low-power mode is activated. Unless otherwise stated, all measurements are performed with 3.3 VDC system voltage and 23°C ambient temperature.

MCU State	OpAmp	DAC	Comparator	Consumption
STOP 1 with RTC	-	-	-	$7.3\mu A$
STOP 1 with RTC	-	internal	-	$98.1\mu A$
STOP 1 with RTC	-	internal	internal	$99.2\mu A$
STOP 1 with RTC	internal	internal	internal	$140.5\mu A$

Table 4.1: Consumption of embedded scenario

Conclusively, this scenario requires overall  $140.5\mu A$  current to remain always-on and trigger when seismic activity is detected. All consumption values are compared to the datasheet values [13], and the results closely coincide with the typical values.

#### 4.5.2 Mixed Scenario 1

In this scenario, only the microcontroller, its required external components (e.g. programming connector, crystals and passives) and the external operational amplifier with its required components are soldered onto the board.

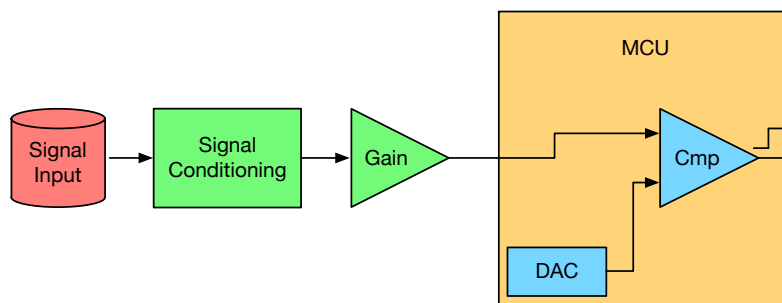


Figure 4.8: Triggering: mixed scenario with external amplifier

<sup>1</sup> Low-Speed External Oscillator

As in the embedded scenario, the lowest power mode that the microcontroller can enter while the built-in comparator and the digital-to-analog are still operational is the STOP 1 mode. Table 4.2 summarizes the results of this scenario. All measurements are performed with 3.3 VDC system voltage and 23°C ambient temperature, unless otherwise noted.

MCU State	OpAmp	DAC	Comparator	Consumption
STOP 1 with RTC	-	-	-	$7.3\mu A$
STOP 1 with RTC	external	-	-	$23.3\mu A$
STOP 1 with RTC	external	internal	internal	$115.2\mu A$

Table 4.2: Consumption of mixed scenario 1

To conclude, this scenario requires overall  $115.2\mu A$  current to be always-on and trigger when there is seismic activity detected. By comparing the results to the nominal consumption values, one can verify that the measured values match with the datasheet values.

### 4.5.3 Mixed Scenario 2

Mixed Scenario 2 differs from Mixed Scenario 1 only that this scenario uses the external comparator instead of the embedded comparator of the microcontroller. Both the embedded and the external comparator share similar parameters, however the external comparator has slightly less power consumption while having smaller offset error.

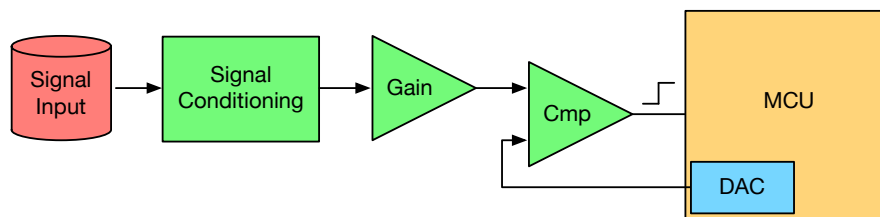


Figure 4.9: Triggering: mixed scenario with external amplifier and DAC

Table 4.3 summarizes the results of this scenario. Unless otherwise noted, all measurements are performed with 3.3 VDC system voltage and 23°C ambient temperature.

MCU State	OpAmp	DAC	Comparator	Consumption
STOP 1 with RTC	-	-	-	$7.3\mu A$
STOP 1 with RTC	external	-	external	$22.8\mu A$
STOP 1 with RTC	external	internal	external	$114.8\mu A$

Table 4.3: Consumption of mixed scenario 2

This scenario requires overall  $114.8\mu A$  current to remain always-on and to trigger when seismic activity is detected.

#### 4.5.4 External Scenario

In this scenario, all external triggering components (operational amplifier, digital-to-analog converter and comparator) together with the microcontroller are soldered to the board.

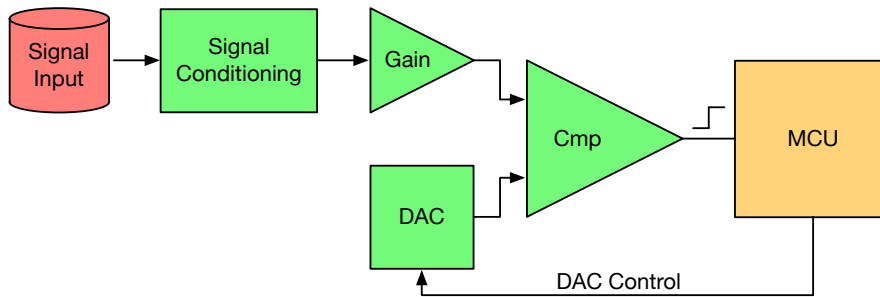


Figure 4.10: Triggering: external scenario

Because none of the analog peripherals of the microcontroller required for triggering is being used for this scenario, the lowest power mode that the MCU can enter is STOP 2. STOP 2 mode has lower energy consumption than STOP 1, while still maintaining fast wakeup and full RAM and register retention. Additionally, the RTC remains active and driven by the LSE when STOP 2 mode is active. All measurements are performed with 3.3 VDC system voltage and  $23^{\circ}\text{C}$  ambient temperature, unless otherwise stated. The resulting consumption values are summarized in Table 4.4.

In conclusion, this scenario overall requires as low as  $22.9\mu A$  current to remain always-on and trigger when there is seismic activity detected.



MCU State	OpAmp	DAC	Comparator	Consumption
STOP 2 with RTC	-	-	-	$3.1\mu A$
STOP 2 with RTC	external	-	-	$19.1\mu A$
STOP 2 with RTC	external	external	-	$22.4\mu A$
STOP 2 with RTC	external	external	external	$22.9\mu A$

Table 4.4: Consumption of external scenario

#### 4.5.5 Comparing Results

Despite the fact that one can build a triggering circuit solely using a single microcontroller with embedded peripherals, aiming for high signal and overall triggering qualities with the lowest power-consumption suggests using external components. Based on the results of the initial triggering circuit evaluation, the energy consumption of the system can be reduced at least by a factor of 4 by using external peripherals. Certainly, there are disadvantages of using external components, for instance increased circuit complexity, layout space, bill of materials and overall increase in manufacturing costs. However, the benefits of using external peripherals are ultimately higher compared to the embedded or mixed solutions.

## 4.6 Improved Triggering Circuit

The initial triggering circuit evaluation showed remarkably great results. However, at the time of the development of the initial triggering circuit, there was no information present about what types of seismic signals should the circuit be tuned for, furthermore what types of signal, triggering and recording parameters would help ongoing researches in geoscience. In other words, what should be the amplification gain value and the desired triggering threshold values. After several discussions with partners from geoscience, two new features were added to the original triggering circuit. The resulting improved triggering circuit is incorporated in the hardware design of the geophone platform.

### 4.6.1 Dual-Side Triggering

The first requested feature was to differentiate and independently trigger to both positive and negative signal threshold values. In order to implement this feature called *dual-side triggering*, two changes are made.

First, the output of the operational amplifier is set to mid-range, i.e. half of the system voltage to provide the highest dynamic range for both sides. Thus, when the input signal is zero, the output is set to  $\frac{V_{sys}}{2}$ . Consequently, when the input signal changes, the amplified signal oscillates around this voltage. This feature is implemented by simply adding a voltage divider to set the DC bias of the amplifier. Depending on the resistor values, the voltage divider slightly increases the static power dissipation of the circuit.

Secondly, two separate reference voltages need to be established. The simplest method is to employ two DACs with two comparators, which eventually issue two separate interrupts: positive wakeup trigger and negative wakeup trigger interrupts. By issuing two dedicated interrupts for wakeup, the microcontroller can automatically distinguish the type of wakeup, because the wakeup interrupt lines automatically denote that the signal exceeded either the positive threshold or the negative threshold.

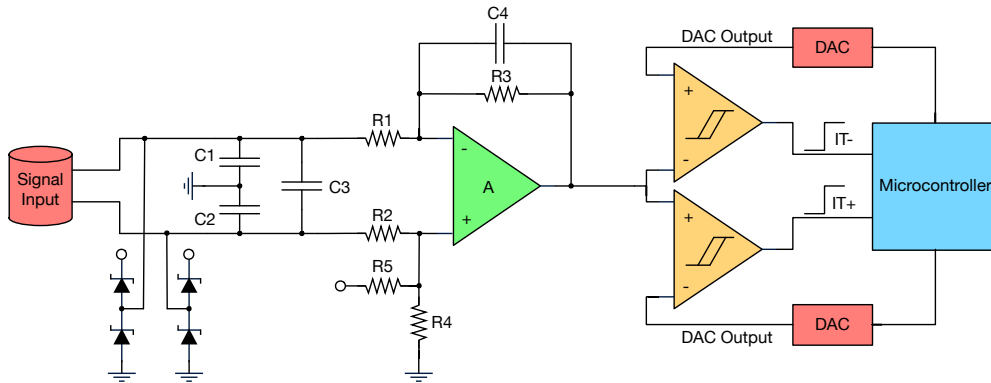


Figure 4.11: Dual-side triggering

Implementing this feature results in a slight increase of energy consumption because of the additional comparator and the digital-to-analog converter. The additional current drain caused by the two extra component is less than  $4\mu A$ . On the other hand, the complexity of the circuit design and the required layout

space does not increase, because both the DAC and the comparator are offered in packages that incorporate two of each. The dual-output version of the MAX5530 digital-to-analog converter is the MAX5532 model [18]. The dual-channel version of the MAX919 comparator is the MAX9019 variant [19].

### 4.6.2 Multi-Stage Triggering

By using a single-stage amplification circuit, the signal range of the triggering is limited to the specific gain settings. Furthermore, the amplification gain of an operational amplifier cannot be increased indefinitely, especially with sensitive, ultra-low power OpAmps. Because there has not been a precise consensus about the requirements of necessary gain and dynamic range parameters, a multi-stage amplification with dynamic on/off switching feature is implemented. This solution requires using an additional operational amplifier that can be turned on and off by the microcontroller and an analog multiplexer to choose which output signal of the OpAmps should be fed to the comparators. Figure 4.12 depicts the implementation of a multi-stage amplification circuit with on/off switching feature.

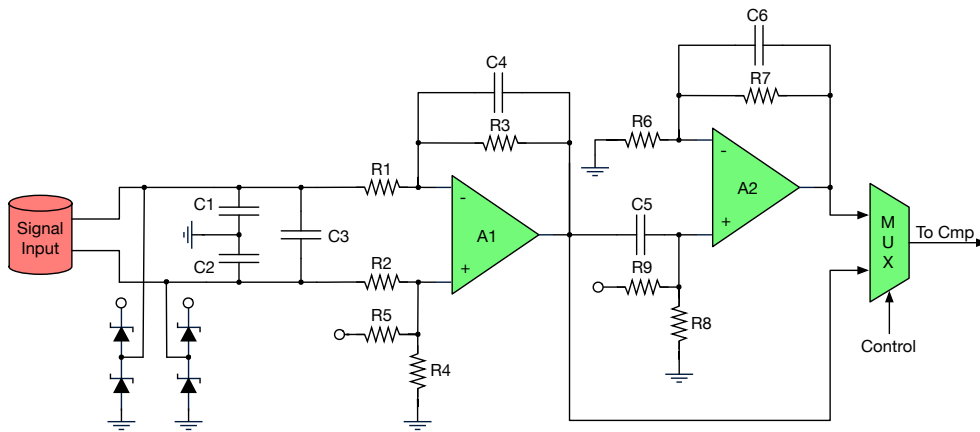


Figure 4.12: Multi-stage triggering

The first stage outputs an amplified signal biased to  $\frac{V_{sys}}{2}$  VDC. The second stage incorporates the same amplifier used in single-ended mode. Upon designing such dual-stage amplification circuit, the following two important considerations must be kept in mind.

First, the second amplifier must be used in non-inverting configuration, because regardless of which stage is used for amplification, the output signals must have the same polarity for appropriate positive and negative threshold comparison, especially in the case when positive and negative references have a non-symmetric configuration.

Secondly, regardless of the used stages, the output signal that is compared to the reference values needs to swing around  $\frac{V_{sys}}{2}$  VDC. This implies that both amplifiers must have an appropriate DC bias settings that fulfills this requirement. Because the first amplifier is used in differential configuration and the second is used in non-inverting single-ended configuration, the most straightforward solution is to remove the DC component of the signal produced by the offset configuration of the first amplifier, perform AC coupling [20] on the signal and set the DC bias of the second stage appropriately. AC coupling, or known as DC-blocking, can be realized with a single capacitor, also known as a *DC-blocking capacitor*. Considering an AC-coupled, non-inverting amplifier circuit with DC bias setting as shown in Figure 4.12, the  $C_5$  capacitor and the voltage divider resistors ( $R_8$  and  $R_9$ ) formulate a high-pass filter [21]. A high-pass filter lets frequencies higher than the cutoff frequency pass and suppresses the lower frequency components of a signal. The cutoff or sometimes called break frequency of a high-pass filter can be calculated with the following formula:

$$f_c = \frac{1}{2\pi RC} \quad (4.4)$$

where  $C$  represents the capacitance of the AC coupling capacitor ( $C_5$ ) and  $R = R_8 \times R_9 = \frac{R_8 R_9}{R_8 + R_9}$ .

Naturally, the DC component itself should be completely removed from the signal, however the goal is to keep the low frequency components of the input signal. Therefore, the cutoff frequency should be set as low as possible without affecting signal quality and response time.

When the second amplification stage is not needed, it can be turned off with a single transistor or FET<sup>1</sup> controlled by the microcontroller. Depending on which stage is being used, the appropriate output should be taken and fed to the

<sup>1</sup> Field-Effect Transistor - [https://en.wikipedia.org/wiki/Field-effect\\_transistor](https://en.wikipedia.org/wiki/Field-effect_transistor)

comparator. Therefore, a low-power, 2:1 analog multiplexer<sup>1</sup> is used to select the appropriate output from the active stage. To automatically perform correct output signal selection, the control signal that operates the FET is wired directly to the input selector pin of the multiplexer. Thus, the microcontroller can perform both the on/off switching of the FET and the selection of the appropriate output with a single control line.

### 4.6.3 Realization of the Improved Triggering Circuit

Putting the pieces together, the complete design of the improved triggering circuit is shown in Figure 4.13. This advanced triggering system is incorporated in the design of the geophone platform, described in Chapter 5.

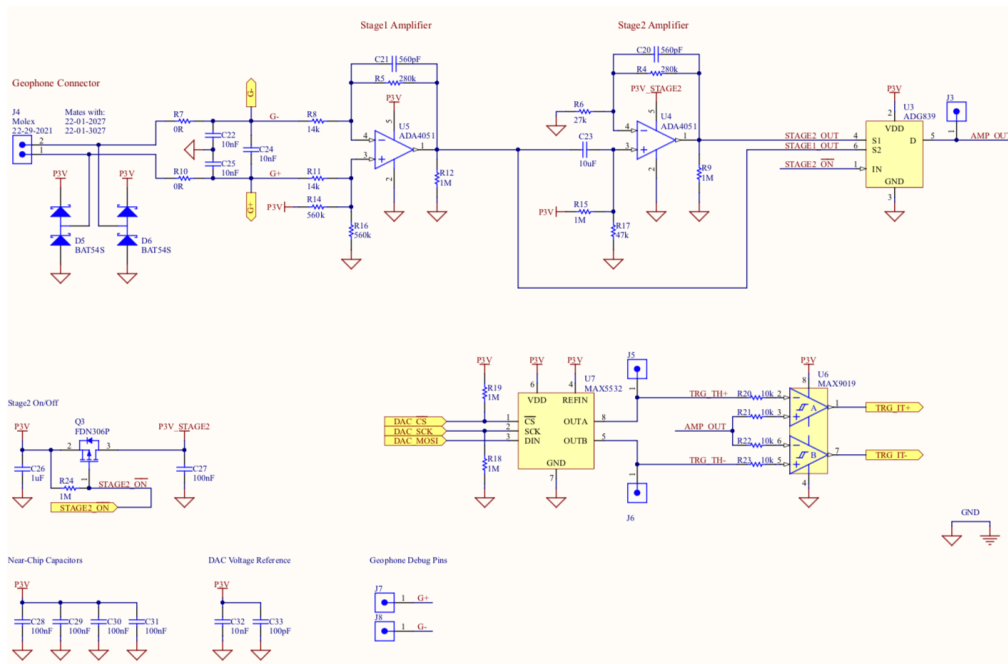


Figure 4.13: Improved triggering circuit schematics

<sup>1</sup> Multiplexer - <https://en.wikipedia.org/wiki/Multiplexer>

## 4.7 Co-detection

The main method of detecting micro-seismic activities is to observe the incoming signal from the geophone sensor. However, the efficiency of detection and triggering can be increased by adding multiple types of sensors to the platform that can detect seismic activities.

In addition to geophone sensors, Inertial Measurement Units (IMU) can be used for movement detection. Nowadays, advanced inertial measurement units require little power to operate and they offer free fall detection, magnetic field detection and movement detection with wide configurability options. Some devices even provide automatic increase in sampling rate when specific activities are detected.

The geophone platform – apart from performing triggering to geophone signals – incorporates the LSM303C advanced inertial measurement unit from STMicroelectronics [22]. This ultra-compact module has a built-in 3-axis accelerometer and a 3-axis magnetometer. It features ultra-low power modes, configurable interrupt generators and it can dynamically increase and decrease sampling rates according to the current activity to provide efficient power dissipation.

# Hardware Design

---

In this chapter, the detailed hardware design of the geophone platform is presented. First the system architecture is introduced, then the key components and circuitry are described in detail. The hardware, including the schematics and printed circuit board is entirely self-designed.

## 5.1 System Architecture

The system architecture of the geophone platform consists of six domains: the microcontroller subsystem, advanced triggering circuit, high performance digitizer unit, sensor subsystem, communication module and efficient power supply circuit.

The system architecture is depicted in Figure 5.1.

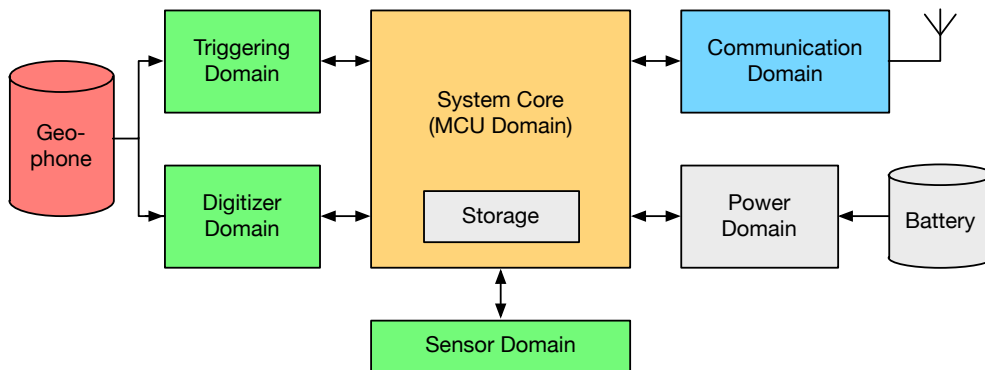


Figure 5.1: System architecture

The *microcontroller subsystem* serves as the core of the platform. It consists of a best-in-class, ultra-low power, 32-bit, ARM Cortex-M4 core microcontroller and its required external components (e.g. external crystals, programming in-

terface, etc.). The subsystem also comprises a microSD card which functions as the storage unit; and other miscellaneous components, including a buzzer, push-button, reed relay and LEDs. The microcontroller is also responsible for switching on and off all connected components and subsystems to minimize energy consumption.

The *triggering subsystem* is the realization of the improved triggering circuit described thoroughly in Section 4.6. It features an always-on, dual-side, multi-stage triggering circuit with ultra-low power consumption. It consists of signal filtering, two operational amplifiers, a dual-channel digital-to-analog converter and a dual-channel comparator.

The *digitizer subsystem* is responsible for high-precision analog-to-digital conversion of the geophone signal. It features a 24-bit delta-sigma ADC module with high SNR and built-in Programmable Gain Amplifier (PGA), together with an ultra-low noise, high-precision voltage reference. The digitizer subsystem can be entirely switched off by the microcontroller when the system is in sleep state.

The *sensor subsystem* consists of an 6-DOF inertial measurement unit, a temperature and humidity sensor and a battery voltage monitoring circuit. All sensors have separate power rails, therefore each sensor can be efficiently power-gated.

A state-of-the art communication board called *DPP COM board* represents the *communication subsystem*. The DPP COM board features an ultra-low power radio SoC and BOLT. Section 2.3 describes the Dual Processor Platform and the COM board in detail.

The *power subsystem* provides stable system voltage across the hardware. The main power source is a D-cell military-grade battery. The devices can also be supplied via USB with a single USB cable. Furthermore, the hardware provides additional connectors to attach an external power supply to precisely measure current consumption of the system.

All subsystems, circuits and the entire printed circuit board is designed appropriately for noise-resilience, reduced electromagnetic interference, increased electromagnetic immunity; and to provide overall stable, reliable operation. The entire hardware design is carried out based on previous practical experiences and hardware development guides [23].



## 5.2 Microcontroller Subsystem

In this section, the core circuit and its components are described in detail.

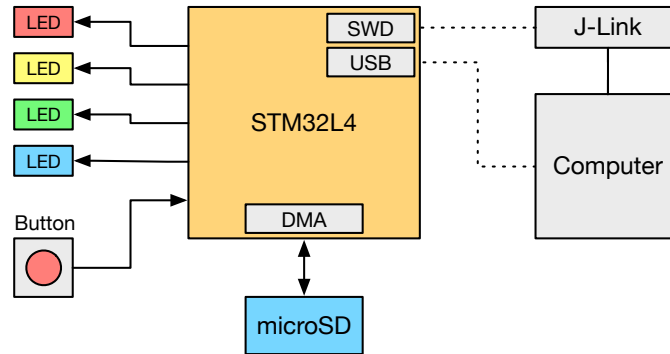


Figure 5.2: System core - microcontroller domain

### 5.2.1 The Microcontroller

An ultra-low power, 32-bit ARM Cortex-M4 core microcontroller from STMicroelectronics is selected as the processing unit of the geophone platform. The selection is based on meticulous research and comparison of microcontrollers from various manufacturers. The detailed comparison can be found in the project repository.

STMicroelectronics is a leader in electronics and semiconductor, including microcontroller manufacturing, and it is the largest semiconductor chip maker in Europe [24]. ST's STM32L4 series microcontrollers are best-in-class, 32-bit Cortex-M4 core microcontrollers specifically designed for ultra-low power applications. The STM32L496VG variant of the STM32L4 series is chosen as the core processing unit for the geophone platform.

#### STM32L496VG

The STM32L496VG microcontroller has great parameters that perfectly match the system requirements of the geophone platform. It features up to 80MHz core clock, advanced bus and interconnect matrix, 9 low-power modes, including 3uA current drain in STOP 2 mode with RTC on and full RAM retention; 1MB flash,

320kB continuous SRAM, RTC with calendar features and backup registers, various hardware timers, numerous embedded digital and analog peripherals and communication interfaces with Direct Memory Access (DMA), USB OTG 2.0 full-speed and more. For the full specifications, please refer to the STM32L496 datasheet [25].

### 5.2.2 Microcontroller Circuit

The microcontroller is powered from the main 3.0 VDC power rail. Its analog subsystem has a separate power rail with additional filtering to provide highly stable analog supply and reference voltages for the analog peripherals.

Two crystals are connected to the microcontroller: a low-frequency, high-precision 32.768kHz quartz to provide precise clock source for the RTC peripheral, and a common 8MHz quartz for the built-in PLLs<sup>1</sup> that generate the appropriate clocks for the internal buses and peripherals. The oscillator circuits and their layouts are designed meticulously to provide stable and precise clock sources and minimize clock drift as described in [26].

### Programming and Debugging

The programming and debugging of the microcontroller is performed via the standard Serial Wire Debug (SWD) interface. SWD is an alternative to JTAG, it provides the same debugging capabilities, furthermore it requires only two wires and uses the ARM bi-directional wire protocol defined in [27]. The J-Link EDU debug probe from SEGGER is used via the SWD interface for programming and debugging the geophone platform [28].

ARM Cortex-M4 core microcontrollers use the CoreSight debug architecture which provides advanced, real-time debugging and tracing capabilities. Covering the CoreSight technology is not part of this thesis, however interested readers can refer to the entire architecture specification in [29].

There are several advanced features used for debugging and tracing the geophone platform, including the Serial Wire Viewer (SWV) and the Instrumenta-

---

<sup>1</sup> Phase-Locked Loop - [https://en.wikipedia.org/wiki/Phase-locked\\_loop](https://en.wikipedia.org/wiki/Phase-locked_loop)

tion Trace Macrocell (ITM).

Serial Wire Viewer is a real-time trace technology that uses the SWD interface in combination with a single pin located on the microcontroller: the Serial Wire Output (SWO) pin. SWV provides real-time information, including program counter values, data read and write cycles, variable and peripheral values, event counters, exception entry and return, used CPU cycles and timestamps with basically no processor overhead while the application is executing at full speed.

The Instrumentation Trace Macrocell allows to write data to the SWO pin, which can be used in many ways, including “printf-style” debugging, interrupt tracing, reporting application events, diagnostic information and real-time operating system (RTOS) tracing.

In addition, SEGGER’s Real Time Transfer (RTT) technology is used for tracing and visualizing the running RTOS in real-time. RTT combines the advantages of SWO and semihosting<sup>1</sup> at very high performance [30]. According to SEGGER, the data output performance of RTT is currently the highest available technology that can be used for transferring data to a host computer. Furthermore, RTT does not affect the real-time behavior of the target.

### 5.2.3 Storage

For storing data, including measurement recordings, events and status information, an industrial-grade microSD card with SLC NAND flash is used. Memory cards equipped with SLC NAND flash are suitable for industrial applications where frequent writes are required. The geophone platform uses microSD cards from the SC series manufactured by Panasonic [31].

To provide easy access and hot-swap of the microSD card while the hardware is running and mounted in its enclosure, a hinge-type microSD card holder is used.

The microcontroller communicates with the microSD card using the standard 4-bit SD/MMC mode with dedicated DMA channels to maximize transfer speed and minimize utilization overhead.

---

<sup>1</sup> Semihosting is a mechanism that enables code running on an ARM target to communicate and use the I/O facilities on a host computer that is running a debugger.

The platform supports microSD cards with FAT32 (or optionally exFAT) file systems. Furthermore, when the device is attached to the computer via USB, the SD card can be mounted to the PC as an external drive, thus providing simple and convenient access to the data located on the memory card without removing the card from its holder.

#### 5.2.4 Miscellaneous Components

The hardware includes additional components for user interaction. Four LEDs with different colors are placed at a visible location to indicate the most important events. A buzzer is used for providing feedback sounds when the device is fully enclosed and the LEDs are not visible. It is especially useful for indicating hardware status, command executions and giving status feedback during deployment.

Two push-buttons are located on the printed circuit board. One serves as a single user-button that allows the user to change operating modes. The other button functions as a reset button for the microcontroller. The buttons are used mainly for testing purposes.

In addition, two reed relays are located at the edges of the PCB. The relays provide reset functionality for both the microcontroller and the DPP COM board. When the device is fully enclosed, the relays can be triggered externally with a magnet, thus the appropriate modules can be reset without disassembling the enclosure.

### 5.3 Triggering Subsystem

The triggering subsystem of the geophone platform is a state-of-the-art, efficient, ultra-low power consumption, high dynamic range, on-the-fly configurable triggering circuit with dual-side triggering capability and multi-stage signal amplification. The triggering circuit and its realization is described in detail in Section 4.6.

The hardware employs an omnidirectional SM-6 geophone with high open-circuit sensitivity of  $80.0V/m/s$  and  $3.5k\Omega$  coil resistance [32]. The geophone is connected to the input of the triggering circuit.

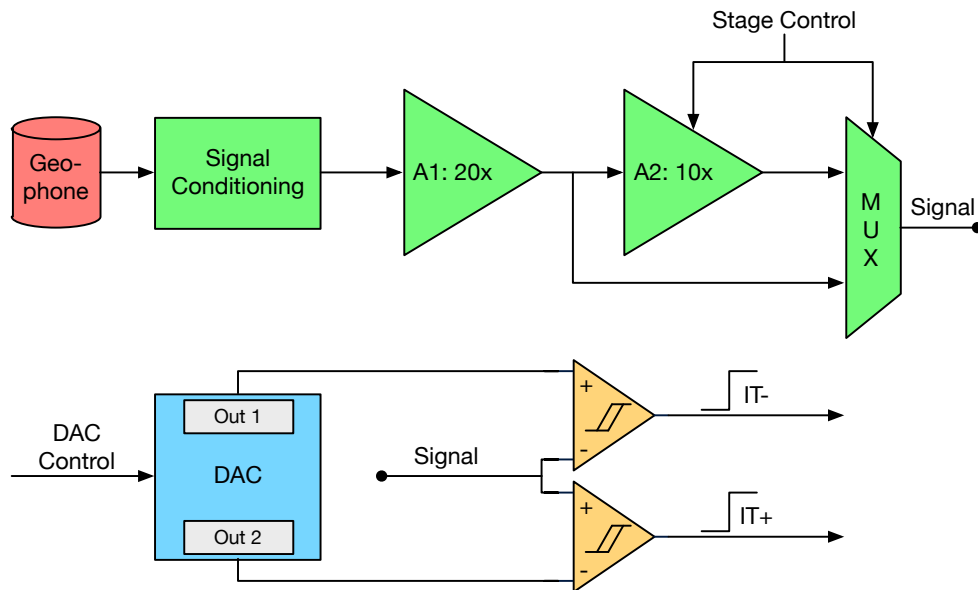


Figure 5.3: Triggering domain

The first, always-on stage has 20x amplification gain, and it has limited bandwidth. The amplifier acts as a low-pass filter with a cutoff frequency of 1kHz.

The second, switchable stage has an additional 10x amplification gain, thus resulting in a triggering circuit with a total of 200x gain. Similarly to the first stage, the second stage amplifier implements a low-pass filter with 1kHz cutoff frequency. The second stage can be switched off by the microcontroller with a single P-FET. The same line that controls the FET is wired to the input selector pin of analog multiplexer, thus the microcontroller can perform both the on/off switching of the second stage and the selection of the appropriate signal output with a single control line.

The output signal is then compared to the reference signals produced by the ultra-low power, dual-channel digital-to-analog converter. To minimize power dissipation of the system, the microcontroller pins are configured as analog inputs when the MCU is in STOP 2 mode, resulting in floating control pins of the DAC. Thus, the chip-select (CS) pin of the DAC has to be tied to VDD in order to avoid unintended communication with the DAC.

During the evaluation of the initial triggering circuit, long-term tests running

for days resulted in significant increase in energy consumption despite the fact that the system remained in sleep mode for the entire time of the tests. After thorough hardware debugging, it was discovered that the SCK line of the DAC left floating can cause severe current leakage, which appears and increase slowly only hours after the system is left untouched in sleep mode. However, this phenomenon is not mentioned in the datasheet of the DAC whatsoever. By pulling the SCK pin either to VDD or GND loosely (i.e. with a high-resistance resistor), the current leakage can be prevented, thus this issue can be resolved. It is worth mentioning that the DIN pin of the DAC is not affected by this issue.

There are several breakout pins located on the printed circuit board to provide easy measurement access to the filtered input signal, the output of the analog multiplexer and to the reference voltages produced by the DAC.

The comparator that compares the amplified signal to the reference voltages provides two independent wakeup interrupts for the microcontroller: one for crossing the positive threshold and one for exceeding the negative threshold. After detecting seismic activity and waking up the system, the two lines are sampled continuously, thus providing useful numbers about how many times the signal exceeded the thresholds during the actual recording.

## 5.4 Digitizer Subsystem

The digitizer domain consists of a high performance analog-to-digital converter and an ultra-low noise, high-precision voltage reference. This subsystem is responsible for precise and high-quality geophone data recording. The whole subsystem can be switched off to save power when there is no data recording.

### 5.4.1 Analog-to-Digital Converter

A high-precision, high-performance analog-to-digital converter is needed to convert analog signals into digital samples. Ultimately, the MAX11214 ADC from Maxim Integrated is chosen [33]. The selection is a result of extensive research and comparison of several ADCs from different manufacturers based on their qualities and parameters. The composed ADC comparison table can be found in the project repository.

The MAX11214 is a high resolution, high-performance delta-sigma analog-to-digital converter, especially targeted for instrumentation applications, high-precision portable sensors and seismic data acquisition systems. It features 24-bit resolution, wide dynamic range with 18.6-bit noise-free resolution at 1ksps, low energy consumption, high accuracy for DC measurements, single or split analog supplies, advanced filter architecture, low-noise embedded programmable gain amplifier and built-in system calibration capabilities.

### ADC Evaluation Kit

To verify the functionality of the MAX11214 module and determine whether it is suitable for the geophone platform, an evaluation kit of the ADC was tested and evaluated prior to the hardware design [34]. The MAX11214EVKIT demonstrates the features of the MAX1121X ADC family. It features proven, tested PCB design, various input sources with different signal buffering, on-board DACs, voltage references and more. Together with its advanced PC software, it provides simple and straightforward graphical interface for extensive ADC configuration, sample collection, time and frequency domain plotting, histogram plotting and more. The software also provides direct access to the registers in the ADC, thus one can easily experiment with the configuration and quickly find the most suitable ADC setup for a specific application. The evaluation kit, in combination with its software package facilitates quick hardware and software development with reduced chance of design faults.

#### 5.4.2 Voltage Reference

Considering an analog-to-digital converter, the connected reference voltage represents the maximum value that the ADC can convert. This voltage serves as a precise “meter-stick” against the analog signal. Therefore, choosing a precise and stable voltage reference is crucial to perform accurate, high-quality, noise-free sampling.

For this application, the MAX6126AASA25+ variant is selected from the MAX6126 series. This reference produces ultra-low flicker noise of  $1.3\mu V_{p-p}$  output and it has  $45nV/\sqrt{Hz}$  wideband noise. This voltage reference offers 0.02% initial accuracy with maximum  $3ppm/^\circ C$  temperature coefficient over its

operating temperature range from  $-40^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$ .

### 5.4.3 Signal Path

With the combination of the MAX11214 high-performance analog-to-digital converter and the high-precision MAX6126AASA25+ voltage reference, high-quality recordings of micro-seismic activities can be produced.

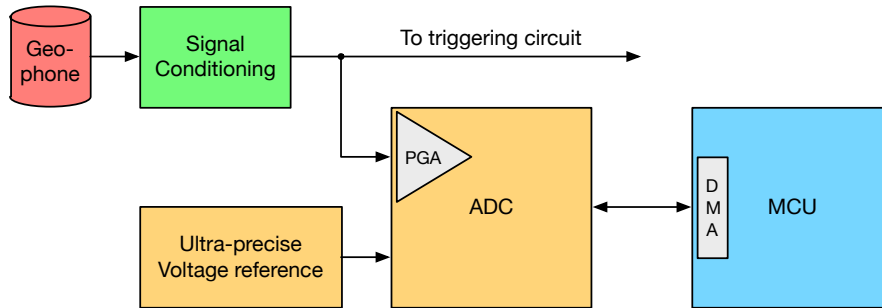


Figure 5.4: Digitizer domain

The incoming geophone signal shares the basic signal filtering for both the triggering and the digitizer circuit. However, after filtering, the signal path is divided into two separate paths: the triggering path and the digitizer path. Considering the triggering path, the signal proceeds to the triggering circuit and amplification stages described in Section 4.4. The signal on the digitizer path proceeds directly to the inputs of analog-to-digital converter. Therefore, the signal is not pre-amplified on the digitizer path.

In the case when the recorded signal needs to be amplified before the conversion, the MAX11214 ADC has a low-noise programmable gain amplifier built in. This PGA can be widely configured even remotely, on-the-fly. By using completely separate amplifiers for the triggering circuit and the digitizer circuit, the platform provides excellent flexibility to achieve the most optimized triggering setup for each field and measurement scenario while having a common conversion setup for signal recording. This construction greatly simplifies the post-processing of measurement data.



## 5.5 Sensor Subsystem

The sensor subsystem consists of an inertial measurement unit, a temperature and humidity sensor and finally a battery voltage monitoring circuit. All sensors can be independently power-gated with dedicated power-switching FETs by the microcontroller.

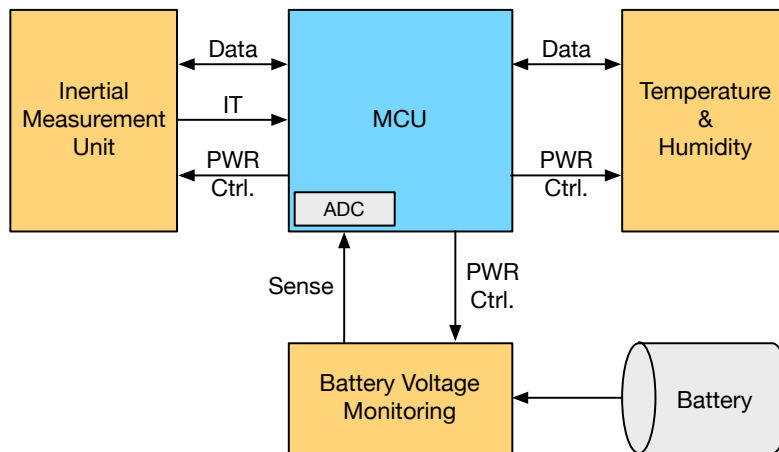


Figure 5.5: Sensor domain

### 5.5.1 Inertial Measurement Unit

Beside sampling geophone signals, the hardware employs a high-performance, low energy consumption inertial measurement unit from STMicroelectronics. The LSM303C from ST is a compact, high-performance e-compass module with 3-axis accelerometer and 3-axis magnetometer with  $\pm 2g / \pm 4g / \pm 8g$  selectable acceleration full scale and  $\pm 16gauss$  magnetic full scale [22]. It features power-down and low-power modes, two programmable interrupt generators for free-fall, motion detection and magnetic detection. Furthermore, the chip can automatically increase sampling rate when a specific activity is detected. This activity/inactivity recognition function allows reducing the power consumption of the chip, thus optimizing the current drain of the system in sleep mode.

The two dedicated interrupt pins of the chip are connected to the microcontroller which is capable of waking up the system when specific activities are detected. These two interrupt sources together with the two interrupts generated

by the triggering subsystem allow the co-detection of micro-seismic activities on the geophone platform.

The inertial measurement unit has a dedicated I2C<sup>1</sup> communication line connected to the microcontroller, furthermore the IMU has a separate power rail that can be independently switched on and off with its dedicated power-switching FET.

### 5.5.2 Temperature and Humidity Sensor

It is crucial to provide periodic temperature and humidity measurement data of each node. Therefore, the best-in-class SHT31 temperature and humidity sensor from Sensirion is incorporated in the geophone hardware [35]. The chip features fully calibrated, linearized and temperature-compensated digital output and it has typical accuracies of  $\pm 0.2^{\circ}C$  and  $\pm 2\%RH$ .

The sensor is connected to the microcontroller via a separate I2C interface. It has a dedicated alert pin used for generating interrupts based on temperature and humidity limits. The minimum and maximum alerting limits can be individually configured for both temperature and humidity measurements. Despite its usefulness, this alert feature is currently not used within this application.

The SHT31 module can be completely powered off with a dedicated power-switching FET to minimize energy consumption of the system when it is not needed.

### 5.5.3 Battery Voltage Monitoring

Although the military-grade battery (described below in Section 5.7) that powers the entire system has excellent discharge profile with constant cell voltage over time until the battery is completely depleted, it is mandatory to accurately monitor battery voltage.

In order to sense the battery voltage, the embedded analog-to-digital converter unit of the microcontroller is used. The STM32L496VG MCU features three successive approximation ADCs with 12-bit resolution, built-in calibration,

---

<sup>1</sup> I2C (Inter-Integrated Circuit) - <https://en.wikipedia.org/wiki/I2C>

5.33 Msps maximum conversion rate, single-ended and differential mode inputs and DMA controller support [25]. By using one of the embedded ADCs of the microcontroller, the required space on the printed circuit board and overall circuit complexity can be reduced, while providing sufficient accuracy for precise battery voltage measurements.

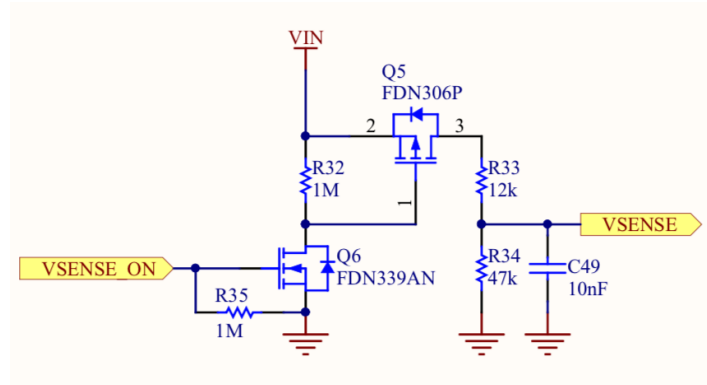


Figure 5.6: Voltage sensing circuit

The LSH 20 battery has a nominal voltage of  $3.6V$ , while the system voltage (thus the analog reference voltage for the ADC) is  $3.0V$ . Therefore, the battery voltage has to be scaled down to match the voltage range of the system. This can be done with a simple voltage divider that consists of the  $R33$  and  $R34$  resistors (see Figure 5.6):

$$V_{sense} = \frac{R_{34}}{R_{33} + R_{34}} V_{in} \quad (5.1)$$

where  $V_{sense}$  is the sampled voltage by the ADC of the microcontroller and  $V_{in}$  is the battery voltage. Knowing that the reference voltage of the ADC is  $3.0V$ , one can calculate the maximum voltage that the circuit is able to sample:

$$V_{in_{max}} = \frac{V_{sense_{max}}}{\frac{R_{34}}{R_{33} + R_{34}}} = \frac{3.0V}{\frac{47k\Omega}{12k\Omega + 47k\Omega}} \approx 3.76V \quad (5.2)$$

Therefore, the maximum battery voltage that the system can precisely measure is approximately  $3.76V$ .

Naturally, the voltage divider introduces a static power dissipation through the divider resistors. Consequently, larger resistor values are preferred in general

to minimize energy consumption of the system. However, for precise analog-to-digital conversion, the output impedance of the sampled circuit must be low, i.e. the external input impedance that the ADC “sees” on its inputs has to be significantly lower than the input impedance of the ADC. The datasheet of STM32L496VG specifies the maximum external input impedance value of  $R_{Ain} = 50k\Omega$ , therefore the values of the voltage divider resistors cannot be maximized to achieve low power dissipation.

The battery voltage is only expected to be monitored periodically, thus introducing power-gating of the voltage monitoring circuit provides a simple solution for the above described problem.

Using low-side switching (i.e. with an N-FET) of the voltage divider does not provide appropriate solution, because when the N-FET is switched off, the output of the voltage divider has the same potential as the battery, thus exceeding the reference voltage of the ADC on its input. On the other hand, by using a P-FET to perform high-side switching of the voltage divider, a new issue has to be circumvented: when the FET is switched off, the gate of the FET is pulled to its source via a resistor, resulting in increased voltage (i.e. the actual battery voltage) on the gate-controlling pin of the microcontroller. To prevent damaging the internal circuit of the microcontroller pin (unless it is a 5V-tolerant pin), an additional N-FET should be added to appropriately control the gate of the P-FET as shown in Figure 5.6. With this solution, the microcontroller can safely turn on and off the voltage monitoring circuit without risking the battery voltage appearing directly on the controlling pin of the MCU.

## 5.6 Communication and Networking Subsystem

The communication subsystem comprises a state-of-the-art communication board developed at the Computer Engineering Group of ETH Zürich [36] as part of the Dual-Processor Platform [2]. This communication board, called *DPP COM board* features a TI CC430F5147 SoC radio module together with BOLT, a stateful processor interconnect implemented on a TI MSP430FR5969 chip [3]. Assuming 3.0V system voltage, the DPP COM board drains only  $2\mu A$  in sleep mode and  $23mA$  when the radio is in listening mode. It has a compact design on a separate printed circuit board. Furthermore, the design of the geophone hardware is

fully prepared for the next generation DPP COM board, which is currently in development and it has a renewed hardware design with a completely new radio chip.

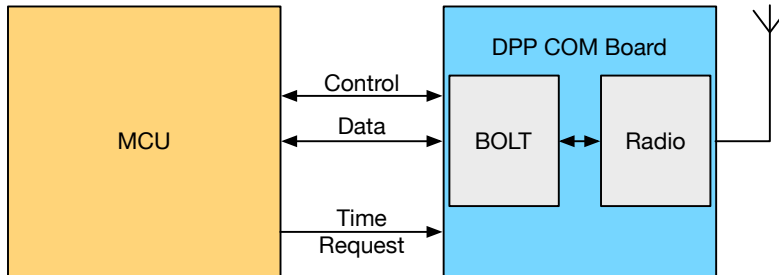


Figure 5.7: Communication domain

The DPP COM board is connected to the geophone PCB via a compact, 26-pin board-to-board connector. Not all the pins are required to be routed from the COM board to the application microcontroller, however by routing additional pins result in extended functionality. For this reason, beside the mandatory power pins and BOLT communication lines (SPI lines, ACK, IND, MODE and REQ), the TREQ and COM\_IND pins are connected. With the help of the TREQ line, the application processor is able to perform precise time synchronization, i.e. synchronize its own time to the network with sub-millisecond precision. The COM\_IND line (which is originally intended to use for debugging purposes) shows whether the communication buffer for packets sent from the application processor to the communication processor is empty. During “busy” scenarios, when many packets are generated on the application side to be sent to the network, the application processor can monitor the status of the COM\_IND line. If the pin shows constantly high-state over time, i.e. the buffer is not getting emptied by the communication processor, the wireless network might be congested. Therefore, the application processor can halt sending packets and backlog the data onto its storage unit until the quality of network service improves.

Additionally, the geophone hardware contains two connectors that are wired to the programming pins and to the serial debug communication lines of the COM board. This provides the possibility to reprogram and debug the communication board during network and protocol development when the COM board is fixed to the geophone mainboard.

One of the reed relays located on the geophone PCB is routed to the reset line of the DPP COM board, thus the communication processor together with BOLT can be reset externally with a magnet.

## 5.7 Power Supply Subsystem

The main task of the power subsystem is to provide stabilized voltage for the system with as little power dissipation as possible, supplied from a D-cell battery. In addition, the system can be supplied with a single USB cable connected to a computer or a typical mobile phone charger connected to a power-outlet.

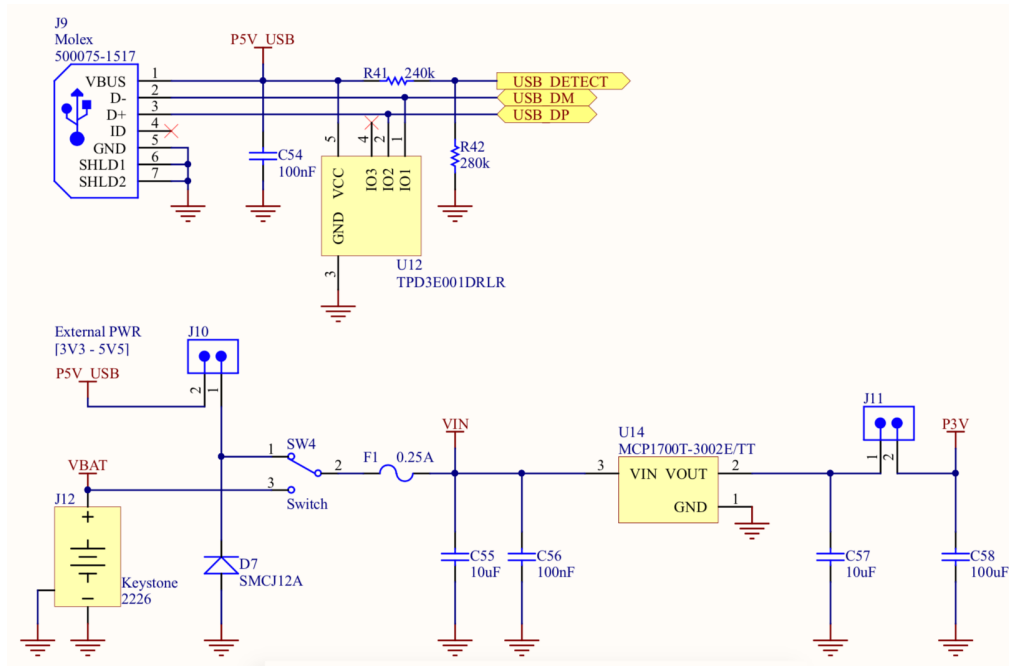


Figure 5.8: Power supply subsystem

A vertical, 90° orientation miniUSB connector is located on the geophone PCB to be able to easily attach and detach the cable even when the board is fixed in its enclosure. The USB lines, including the data lines are electrically protected against electrostatic discharges (ESD). The USB VBUS line is connected to the microcontroller via a voltage resistor, thus the application can detect when an USB cable was attached or detached. The USB data lines are routed to the microcontroller directly. The STM32L496VG microcontroller has a built-

in USB OTG peripheral with integrated transceiver and dedicated, 5V tolerant pins. Henceforth, various USB device classes are implemented on the system, including USB CDC<sup>1</sup> for serial communication used for debugging, USB MSC<sup>2</sup> for mounting the SD card as an external hard drive to the computer and USB DFU<sup>3</sup> for firmware upgrade purposes [37].

With an SPDT<sup>4</sup> side-switch, the user can choose which power source should be selected to supply the system: the “ON” state reflects that the system is supplied from the battery, regardless whether a USB cable is attached. If the switch is in the “OFF/EXT” state, the board is powered from USB (or an external power supply if present) and the system is powered down if there is no external source attached.

Additionally, there are multiple breakout pins located on the geophone printed circuit board for providing connectivity for external power supplies and source meters to perform extensive power analysis. Furthermore, the main power rail can be interrupted with a jumper just after the voltage regulator, thus a digital multimeter or ammeter can be connected in series to precisely measure the current drain of the system.

### 5.7.1 Battery

The chosen battery that powers the system is a 13.0 Ah capacity, D-cell military-grade lithium battery from Saft. The chosen LSH 20 battery has wide operating temperature range of  $-60^{\circ}\text{C}$  to  $+85^{\circ}\text{C}$ , low self-discharge rate, high drain capability and it remains stable during most of its lifetime. Its nominal voltage is 3.6V and the battery has excellent discharge profile [38]. All these excellent parameters make this type of battery an ideal choice for the geophone platform.

---

<sup>1</sup> CDC: Communications Device Class

<sup>2</sup> MSC: Mass Storage Class

<sup>3</sup> DFU: Device Firmware Upgrade

<sup>4</sup> SPDT: Single-Pole Dual-Throw Switch -

<https://learn.sparkfun.com/tutorials/switch-basics/>

### 5.7.2 Voltage Regulator

The system voltage must be set as low as possible to reduce power consumption of the overall system, however all individual components should remain fully operational. Therefore, after checking the operating conditions of each component, the ideal system voltage should be set to  $2.7V$ . However, to provide wide dynamic range for the triggering circuit and precise, widely configurable triggering thresholds, the system voltage is determined to be  $3.0V$ . The LSH 20 battery has cell voltage ranging from  $3.6V$  down to  $3.2V$  depending on load and temperature. Therefore, a suitable ultra-low operating current DC/DC regulator with high conversion efficiency is required.

In general, there are two different methods for providing stabilized  $3.0V$  system voltage from a  $3.6V$  battery. One can use a switching-mode DC/DC converter (known as buck converter) or instead, a linear voltage regulator [39, 40]. Both regulator types have advantages and disadvantages over the other and each application requires great consideration which solution is overall a better choice for that specific system. Table 5.1 summarizes the main differences between the two types of DC/DC regulators.

	<b>Switching Converter</b>	<b>Linear Regulator</b>
Efficiency <sup>1</sup>	High	Medium
Complexity	High	Low
Ripple/Noise/EMI	High	Low
Required space	Medium to high	Small
Total cost	Medium to high	Low
$V_{IN}$ range	Wide	Narrow

Table 5.1: Switching versus linear regulators

From the perspective of the geophone platform, the most important difference is that LDOs generate low electrical noise, which is a crucial requirement in a high-precision measurement device. In addition, LDOs have low complexity, they require fewer external components and they occupy small board space. On the downside, LDOs generally tend to have worse efficiency compared to

<sup>1</sup> Efficiency greatly depends on difference between input and output voltage, current load and more.



buck converters, because the voltage difference between the input and output is lowered by turning excess power into heat. On the other hand, if this voltage difference is low, LDOs have high conversion and operating efficiency.

Considering the power requirements of the geophone platform, the maximum voltage drop between the input and the output (i.e. the difference between the battery cell voltage and the system voltage) is  $3.6V - 3.0V = 0.6V$ . During the scenario when the system is in sleep mode and only the triggering circuit is active, the average current drain at  $3V$  is less than  $30\mu A$ <sup>1</sup>. Consequently, the maximum power loss equals  $0.6V * 30\mu A = 18\mu W$ .

Weighing all the advantages and disadvantages of both regulator types, the MCP1700 low-dropout linear regulator (LDO) is chosen from Microchip Technology to provide system voltage for the geophone platform. This LDO can output maximum  $250mA$  current, while it has an ultra-low operating current of  $1.6\mu A$ , low dropout voltage of  $178mV$  at maximum load and high output voltage accuracy of  $\pm 0.4\%$ . The LDO comes in different, fixed output versions ranging from  $1.2V$  to  $5.0V$ . Furthermore, it is stable with a single ceramic output capacitor and it is available in a tiny SOT-23 package.

A detailed comparison of suitable LDOs and how they compare to the MCP1700 can be found in the project repository.

## 5.8 Ultra-Low Power Hardware Design

The geophone platform is meticulously designed to have ultra-low power consumption. To achieve this, several design methods are used, including appropriate circuit design and component selection, power-gating of all components with non-continuous operating requirements, leakage minimization of always-on components and finally employing ultra-low power software design techniques.

By combining these design methodologies with previous practical experiences, the geophone platform has an outstandingly low, best-in-class energy consumption of less than  $30\mu A$  with always-on, dual-side triggering. Please refer to Section 7.4 for the detailed power analysis.

---

<sup>1</sup> Please refer to Section 7.4 for extensive power analysis.

## 5.9 Mechanics

The nodes are intended to be deployed in harsh, alpine environment, thus the hardware is enclosed in a strong metal enclosure with IP67 rating. The printed circuit board is designed to fit perfectly into the enclosure with supporting screw-holes and appropriate grounding. Every component that has to be accessed while the device is fitted in its enclosure (e.g. power switch, microSD card holder, push buttons, connectors, etc.) is designed to be easily accessible and ergonomically operable by the user. The geophone is fixed to the bottom of the box with a custom-designed clamp mechanism. The geophone nodes can be mounted to the surface with a custom-manufactured mounting plate.

### 5.9.1 Enclosure

The enclosure is chosen from the standard aluminum series from Rose Systemtechnik. It features fixing options for DIN rails and mounting plates. Furthermore, it can be used in wide temperature areas. The chosen variant is the Rose Standard 100x160x81 model. The enclosure has two additional holes drilled on one of the narrow sides: one is for an RF connector with ESD protection to interconnect the communication board with the external antenna and the other hole holds the humidity-balancer plug. The enclosure features a custom-manufactured metal mounting plate, therefore the geophone nodes can be fixed to the exact chosen location while providing excellent seismic signal transfer from the surface to the geophone sensor.

### 5.9.2 Geophone Clamp

Because the geophone sensor is located inside the enclosure, it is crucial that micro-seismic vibrations are transferred effectively from the environment to the sensor. To achieve the best results, the sensor has to be in direct contact with the bottom of the enclosure. Therefore, a custom clamping mechanism is designed to successfully fix the geophone sensor to the bottom. The base of the clamp is firmly fixed to the printed circuit board and to the enclosure. The top of the clamp is a “cap”, which features an o-ring in the inside. Six screws tighten the cap to the base, thus pressing the geophone sensor against the bottom of the

enclosure. The geophone clamps for the geophone platform are manufactured by the D-ITET Workshop<sup>1</sup> of ETH Zürich.

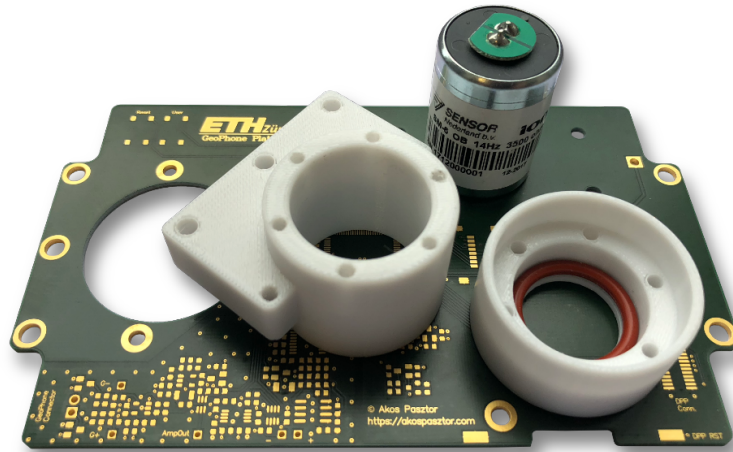


Figure 5.9: Custom-designed, 3D-printed geophone clamp

## 5.10 Prototyping

After finalizing the hardware design of the geophone platform, three prototypes were built. The design files can be found in Appendix B and Appendix C. The PCB prototypes were fabricated by PCB-POOL<sup>2</sup> in Germany. The PCBs were self-assembled and each component was soldered by hand. Hardware debugging was performed with appropriate equipment, including mixed-signal oscilloscopes, digital multimeters, signal generators, power supplies, source meters and power analyzers.

<sup>1</sup> Werkstatt D-ITET - <http://www.werkstatt.ee.ethz.ch/en/the-group.html>

<sup>2</sup> PCB-POOL - <https://de.beta-layout.com/leiterplatten/>



Figure 5.10: Empty printed circuit boards

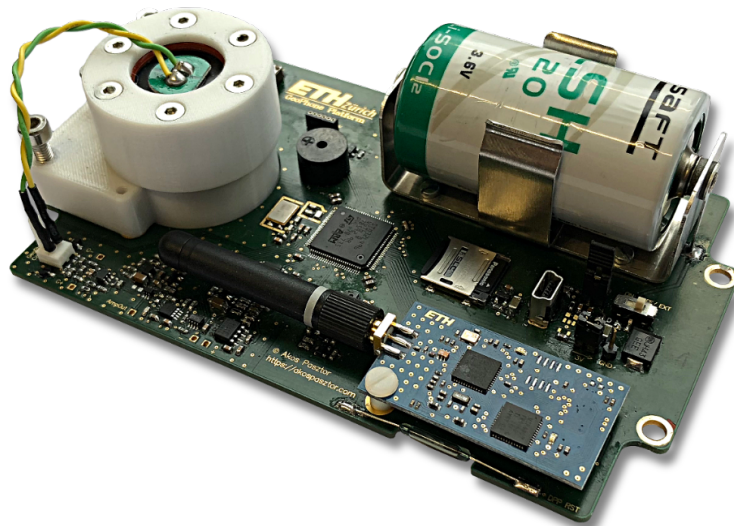


Figure 5.11: Prototype

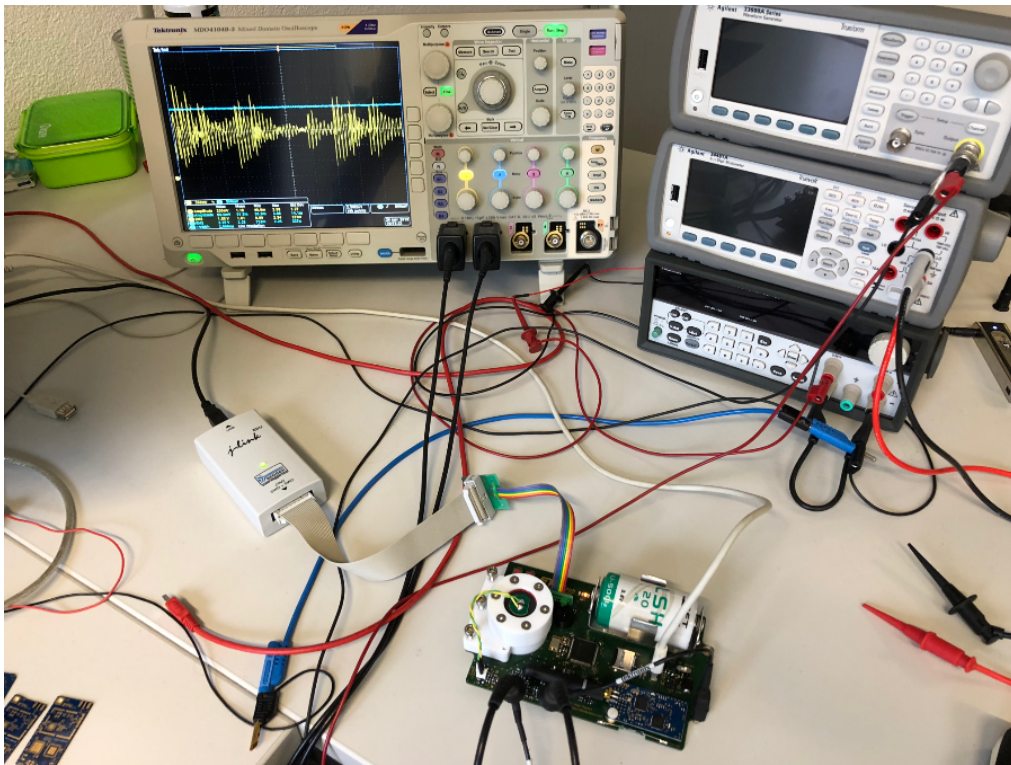


Figure 5.12: Development

*This page intentionally left blank.*

# Software Design

---

This chapter describes the software design of the geophone platform in detail. The entire software is self-written in embedded C programming language, focusing on robust and fail-safe software design; fast execution times while paying attention to producing modular and scalable code; and achieving high active performance while maintaining ultra-low power consumption in inactive state. The source code is well-documented and follows essential coding guidelines.

## 6.1 Major Software Features

The embedded software running on the geophone platform has numerous features which are outlined below.

### 6.1.1 Modified RTOS

The embedded software utilizes the FreeRTOS real time operating system (RTOS) with self-modified kernel, optimized for high active performance while maintaining ultra-low power consumption in sleep state. The kernel is modified to benefit from the tracing features described in Section 5.2.2. As a result, the running software can be debugged and visualized in real-time with negligible processor overhead.

### 6.1.2 Precise Time Synchronization

A single seismic event sensed by multiple nodes should be recorded and logged with the same timestamp, thus it is crucial that all of the nodes within the wireless network cluster provide the same timestamp at any moment. Therefore,

high-precision, sub-millisecond periodic time synchronization is implemented on the platform. The software is capable of synchronizing time with  $200\mu s$  precision.

### 6.1.3 Fast Wakeup

Upon event detection, the system needs to wake up and start recording as fast as possible. The system is designed to guarantee that the first valid sample from the analog-to-digital converter module arrives in less than  $2.7ms$  after triggering.

### 6.1.4 Extensive Logging

The recordings, including samples from the geophone sensor and data from the inertial measurement unit are logged to the microSD card. The platform supports SD cards formatted in FAT, FAT32 and exFAT file systems, furthermore it handles long file names up to 255 characters. As a result, there is no data conversion needed to extract measurement data from the SD card, the files simply can be opened and browsed directly from the SD card when it is mounted to a computer. Each seismic event is logged to a separate file which is written to the appropriate subfolder based on the current date. Each subfolder contains an acquisition file where all the status and information of the recordings are logged, furthermore the system events and health data that are recorded the same day are logged to the same subfolder. The geophone platform supports hot-swapping of microSD cards.

### 6.1.5 Flexible Configuration

The configuration parameters are stored on the microSD card. Upon system startup, the software reads and sets the configuration parameters defined for the node. The platform allows remote, on-the-fly re-configuration of application parameters individually for each node. Upon receiving commands from the network, the new parameters are written to the microSD card, thus the new parameters are loaded after system reset.



### 6.1.6 Network Integration

Acquisition data, status data and system events are transmitted through the sensor network to the backend, where the incoming data is visualized. In addition, specific commands can be issued directly from the backend, thus system parameters can be reconfigured remotely. The commands can be issued either to a specific node, or can be broadcasted to the whole network. Therefore, all nodes can set their parameters to the same value with a single command.

### 6.1.7 Self-Test

The software has a self-test functionality that checks the status of each component after power-up, and the results are logged to the SD card. If an error is detected, an appropriate event is sent to the network to indicate the failure. The execution of the self-test function can also be requested remotely, thus the test can also be performed when the application is running.

### 6.1.8 Self-Monitoring

Beside the self-test, the system periodically monitors its own status and behavior. This includes detecting increased stack watermark values for each task, out-of-range measurement values, status of SD card and more. The system is designed that it can recover from most of the errors and failures on its own, without resetting itself. In the case of a possible hard-fault, the system automatically performs a reset and continues operation.

### 6.1.9 Production Hardware Test

The software contains a separate, thorough hardware-testing unit specifically developed for production. After manufacturing and flashing the application microcontroller with the production-ready software image, the hardware can be connected to a computer via USB and the user can immediately see the results and verify the correct operation of each individual module and component with a single serial terminal, without using any debug probes. This speeds up testing after production, and possible hardware problems, including faulty components or soldering errors can be localized and detected immediately.

### 6.1.10 Connectivity

The device supports various USB device classes, specifically the USB CDC, MSC and DFU classes. The Communication Device Class (CDC) is used for serial communication, general debugging and implementing a command-line interface. The Mass Storage Class (MSC) is used for mounting the microSD card directly to the computer. Thus, when the node is connected to a computer with a USB cable, the SD card appears as an external drive for the computer, and the contents of the SD card can be browsed, furthermore the files can be opened and modified without physically removing the card from the hardware. The Device Firmware Upgrade (DFU) mode can be used for performing a complete microcontroller reprogramming. This feature is useful when the custom bootloader needs to be upgraded.

### 6.1.11 Customizable Bootloader

The geophone platform features a self-written, customizable bootloader. This highly customizable bootloader targeted for STM32 microcontrollers is documented and published at GitHub [41]. The bootloader can upgrade the application firmware directly from the SD card, verify application checksum, enable/disable write protection and more. In addition, the bootloader also supports the possibility of performing over-the-air (OTA) firmware upgrades<sup>1</sup>. The bootloader operation is described in Section 6.2.4.

## 6.2 Software Components

In this section, the employed software components are introduced in detail.

### 6.2.1 Low-Level Drivers

For fast prototyping and rapid development, the official HAL library from ST is used [42]. This driver package contains a rich set of APIs and peripheral drivers for the application layer and it offers the development of highly-portable source

---

<sup>1</sup> At the time of writing, this feature is not implemented because the wireless sensor network has not been prepared for this feature.

code between ST microcontrollers. It is well-documented and MISRA-C 2004 compliant<sup>1</sup>.

However, by solely relying on the HAL drivers results in unnecessary processing overhead and overall in non-optimized, bulky source code. Therefore, the original HAL driver libraries are heavily modified and in some cases completely rewritten to provide optimized and fast execution while preserving software portability and scalability.

In addition, the Cortex Microcontroller Software Interface Standard (CMSIS)<sup>2</sup> is used.

### 6.2.2 Middleware

The middleware used in the embedded software is listed below.

#### FreeRTOS

FreeRTOS is used as the real time operating system of the platform. As of today, it is the market-leading RTOS and de-facto standard solution for microcontrollers [43]. It is open-source and has been in development for over 14 years. Since early 2018, a modified version called Amazon FreeRTOS is available as part of Amazon Web Services (AWS)<sup>3</sup>. FreeRTOS is simple to use, reliable, portable, compact yet feature-rich, free for commercial use and has great community support.

The geophone platform is built on FreeRTOS kernel v9. The kernel contains self-made, custom modifications to provide optimized performance in active mode and ultra-low power consumption in inactive state. In addition, advanced tracing features are added for real time debugging, and it supports real time interrupt and task visualization.

---

<sup>1</sup> MISRA C - [https://en.wikipedia.org/wiki/MISRA\\_C](https://en.wikipedia.org/wiki/MISRA_C)

<sup>2</sup> CMSIS - <https://developer.arm.com/embedded/cmsis>

<sup>3</sup> AWS FreeRTOS - <https://aws.amazon.com/freertos>

## **FatFs**

All the recordings, status and system event messages are logged to the microSD card. Generally, SD cards are formatted with either FAT, FAT32 or exFAT file system. Every computer and major operating system supports these file systems<sup>1</sup>, therefore the user can read and modify the contents of the SD cards and there is no need for additional drivers in contrast with proprietary file systems. Therefore, the software of the geophone platform supports FAT, FAT32 and exFAT file systems.

FatFs is the most common FAT, FAT32 and exFAT file system module made for embedded systems [44]. The driver is completely separated from the disk I/O layer, thus it can be simply ported to the required hardware and application. The driver library is widely configurable, its components can be disabled or enabled depending on the application requirements, therefore it can be used on tiny microcontrollers with limited resources.

The FatFs library is ported to the STM32L496VG microcontroller. The disk layer is modified to use the 4-bit SD interface with DMA. It is completely integrated into the RTOS, therefore during read or write operation, the DMA offloads the processor core and performs the requested operation, while the application can perform other tasks. The DMA support can also be disabled if needed.

## **USB Device Library**

To support various USB device classes, the official USB device driver library is used from ST [45]. It supports all major USB device classes, including audio, CDC, DFU, HID, custom HID and MSC. The driver is optimized for the geophone platform and integrated into the RTOS.

### **6.2.3 Custom Application Libraries**

Various, self-developed and adopted libraries are used in the geophone platform software, which are described below.

---

<sup>1</sup> exFAT is officially supported only on Windows and MacOS, however there are third party implementations available for FreeBSD and multiple Linux distributions.

### **Lightweight Ring Buffer**

A self-developed, lightweight ring buffer library is used for inter-process communication (IPC). The library supports extensive error handling and overflows. Using ring buffers for inter-process communications provides safe data transfer and prevents data loss when a lower priority task is delayed. Furthermore, the system performance can be optimized with ring buffers. For instance, logging is performed to the SD card once per second. Writing bulk data to the SD card is generally much faster compared to writing the same amount of data in many smaller chunks. Therefore, the data that needs to be logged is collected in a ring buffer and is written to the SD card once per second.

### **BOLT Library**

A custom, RTOS-compatible library is developed for communicating with BOLT, which handles the data transfer between the application and communication processor. The software module supports reading and writing operations, precise time synchronization, command handling and error handling.

### **SEGGER RTT Library**

SEGGER's Real Time Transfer (RTT) technology is used for advanced RTOS tracing, including real-time tracing and visualization of interrupts, events and task executions. The default RTT implementation from SEGGER is modified to be compatible with the modified FreeRTOS kernel. Furthermore, the default implementation cannot handle the high number of trace events generated during software execution, therefore additional modifications are made to handle more than 7500 events per second.

#### **6.2.4 Geophone Bootloader**

The geophone platform features a self-written, customizable bootloader that is documented and published at GitHub [41]. The published bootloader demonstration is originally targeted for STM32 microcontrollers, although it can be easily customized and adjusted to all application requirements. The project has

more than 500 unique visitors and 15 unique clones on average within a two-week period.

The bootloader takes up only a tiny bit of the microcontroller memory space: it is located in the first 32 kilobytes of the flash. Upon powering or resetting the device, the bootloader automatically starts and if there is no user interaction, it checks the application firmware located in the user space, verifies its checksum and launches the application.

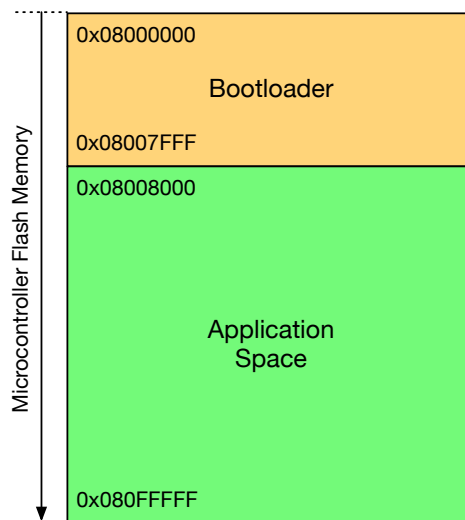


Figure 6.1: Microcontroller flash organization

If there is any interaction detected during startup, the bootloader launches the firmware upgrade sequence. First, it checks the storage unit for new firmware and checks the microcontroller flash whether it is write-protected. The bootloader is capable of disabling the write protection, however the user has to explicitly confirm this operation to do so. This prevents accidental firmware replacement. After confirming to disable the write protection of the flash, the bootloader continues with the firmware upgrade sequence. The new firmware binary is checked and verified, then the flash is erased. The flash programming can be prevented with appropriate user interaction if the user simply wants to erase the chip without programming a new firmware to it. After the erasing procedure, the application performs the actual flash programming. After programming, the flash content is checked and verified whether the programming is performed successfully. If configured, the bootloader can enable the flash write

protection after the programming sequence. Finally, the bootloader launches the new firmware.

The bootloader is also capable of performing over-the-air (OTA) firmware upgrades. However, the current state of the wireless network is not yet prepared for this operation.

The bootloader can jump to the built-in system memory of the microcontroller without toggling any external pins, i.e. the dedicated BOOT pin. The system memory boot mode provides the possibility to perform flash programming via various communication interfaces, including USART, CAN, USB, I2C, SPI, etc. When the USB cable is connected, the bootloader can jump to the system memory and put the device into DFU mode, therefore a full chip reflash can be performed via USB. This feature is useful when the bootloader itself needs to be upgraded.

## 6.3 Software Architecture

This section describes the software architecture in detail. Figure 6.2 illustrates the overview of system architecture.

The system consists of a bootloader, initialization sequence, interrupt handlers, 7 RTOS tasks, including five user tasks, an idle task and a timer service task. The following sections describe these blocks in detail.

### 6.3.1 Initialization Sequence

After power-up or software reset, the bootloader starts and launches the geophone application. The bootloader and its operation is described in detail in Section 6.2.4.

The geophone application starts with an initialization sequence. First, the hardware and peripherals are configured, including system clocks and buses, pins, ports, communication interfaces, timers, real-time clock, etc. Following the hardware initialization, the application and its parameters are configured. This includes the initialization of ring buffers, structures and reading the configuration from the SD card. If no configuration is found on the SD card, the

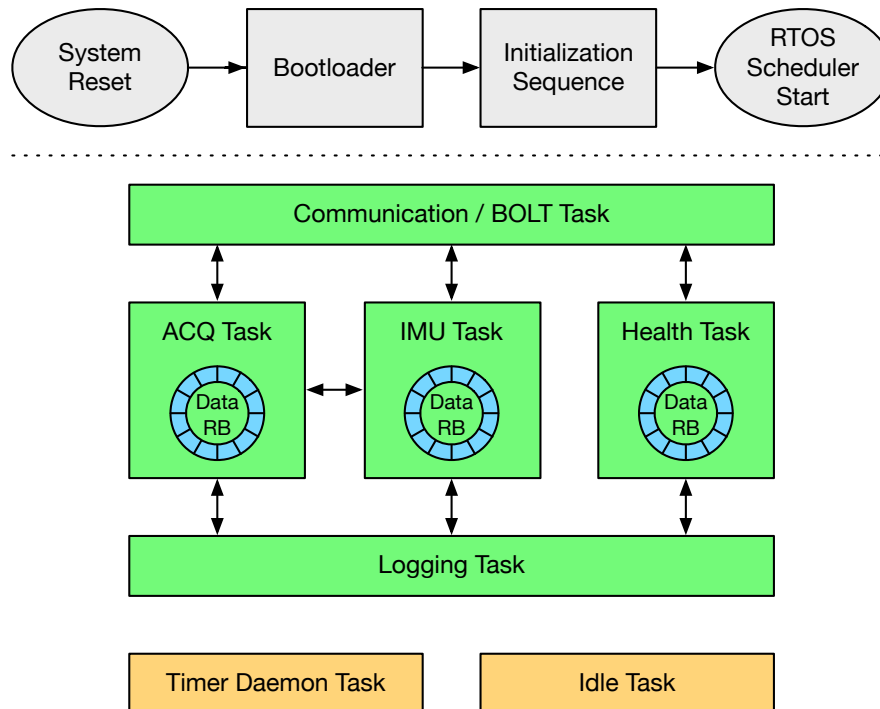


Figure 6.2: Software architecture

default configuration is used. Then a self-test is performed to verify that every major component functions properly. After self-test, the RTOS objects, including tasks, software timers, semaphores and queues are initialized, alongside with the advanced tracing features - if the firmware is compiled with the debugging features enabled.

After successful initialization, the real time operating system starts. The timer daemon task has a startup hook (callback) function which is executed only once, right after the RTOS scheduler is started. Within this callback, additional startup code can be placed that are executed when the RTOS is already running. In this callback, the following startup duties are performed:

1. Startup event is generated and logged.
2. Node info packet is generated with current information of the node.
3. Network time synchronization procedure is initiated.
4. The periodic wakeup interrupt that signals when the periodic health task



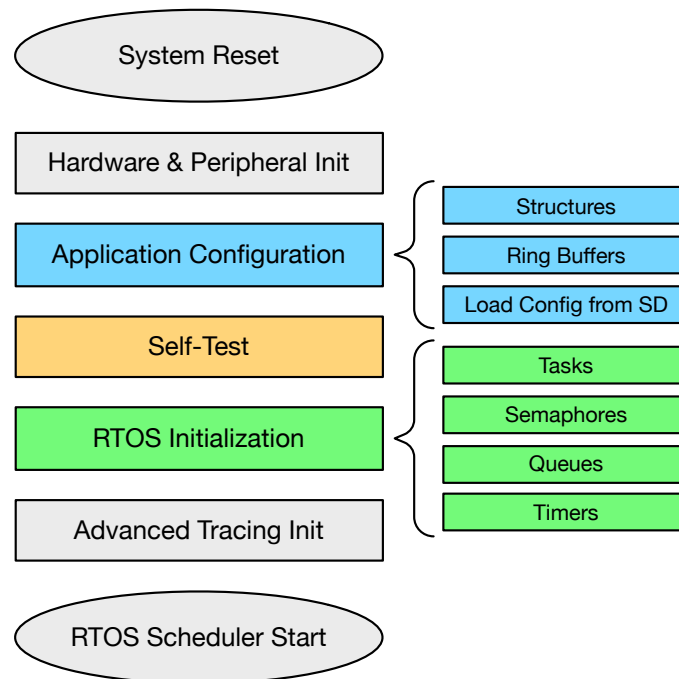


Figure 6.3: Software initialization sequence

needs to be executed is enabled.

5. Buzzer is sounded for half a second to indicate successful application launch.

### 6.3.2 RTOS Tasks

The geophone software features 7 tasks, including the RTOS timer service task and the idle task. The task priorities are depicted in Figure 6.4.

#### Acquisition Task

The acquisition task has the highest priority in the system and it is responsible for recording seismic data and controlling the lower priority IMU task which manages the data collection from the inertial measurement unit.

The task can be started directly from the triggering interrupt routines. In order to launch the ADC subsystem as fast as possible, the triggering interrupt

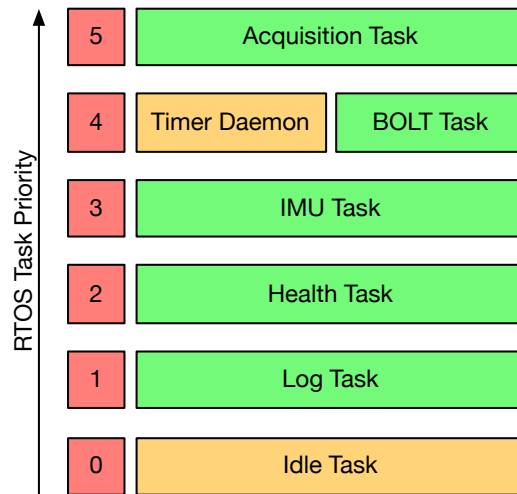


Figure 6.4: RTOS task priorities

routines turn on the digitizer subsystem and use direct-to-task notifications instead of RTOS semaphores. FreeRTOS states that unblocking an RTOS task with a direct-to-task notification is 45% faster and uses less RAM compared to binary semaphores [43].

After unblocking, the acquisition task configures the ADC module with the required parameters and launches sampling. The ADC is interfaced with the microcontroller via high-speed SPI and uses DMA to transfer measurement data. By default, the ADC generates 3000 bytes per second, thus using DMA greatly reduces the performance overhead of the microcontroller caused by the data transfer process. The data is collected to the appropriate ring buffer and logged to the microSD card once per second.

The duration of the acquisition procedure is defined by the post-trigger configuration value. The data recording lasts until there is no more interrupt generated by the triggering subsystem for the duration of the post-trigger value.

After the acquisition procedure is completed, the ADC is turned off, the IMU task is stopped and an acquisition packet is generated which is then transmitted to the sensor network.

### **BOLT Task**

The BOLT task has the second highest priority together with the timer daemon task. This task is responsible for communicating with the DPP COM board via BOLT. The task handles not only the read and write operations to BOLT, but also the entire time synchronization procedure. The execution of commands that are received from the network and affect the configuration parameters are delayed if there is already an ongoing acquisition in progress.

In the case the network gets congested and the communication module cannot send out the packets queued in BOLT, the geophone application performs packet backlogging. If the task detects that the BOLT buffer which contains packets to be sent to the network does not get emptied by the communication processor, the packets are backlogged to the storage unit. Then the application periodically checks whether the network issue has been resolved. If yes and the BOLT data buffer is empty, the software fetches the packets from the backlog and continues sending data to BOLT<sup>1</sup>.

### **IMU Task**

The inertial measurement task has the third highest priority in the system and is responsible for configuring and recording data from the inertial measurement unit. The task is operated and controlled by the acquisition task. The reason why there is a separate task for collecting IMU data is that the IMU has different timings and generally slower compared to the ADC unit. Additionally, the IMU sampling can be entirely disabled in the configuration.

### **Health Task**

The health task is a low-priority task executed periodically. It performs measurements, implements system monitoring features and collects information about the running system.

The health task checks the stack watermark of each task, including the timer daemon and idle tasks. If any of the stack watermarks exceeds 90%, a warning

---

<sup>1</sup> At the time of writing, the backlog feature has not been implemented yet

event is generated. Furthermore, the task checks the RTOS heap status and calculates the duty cycle percentage of the system. Finally, the task performs battery voltage, temperature and humidity measurements.

The default period of the health task is set to two minutes and the task is executed even when there is no seismic activity detected. The results are logged to the storage unit and a corresponding health packet containing the most important information is sent to the network.

### Logging Task

The logging task is a low-priority task that is responsible for logging data to the SD card and reading configuration from it when needed. In addition, the logging task handles all requests related to the SD card, including mounting and ejecting procedure, subfolder creation and selection, etc. This is the only task that is able to communicate with the SD card. Because of there are multiple subfolders present and different types of data need to be logged to separate files, assigning only one task that handles all requests to the SD card prevents concurrent file access and operations that can lead to data corruption or even file system crash.

Therefore, the individual tasks can issue specific commands to the logging task which blocks until there is a command pushed to its queue. The message that contains the logging command is implemented with a message structure. This structure has two important parameters: the actual command type and the corresponding data pointer. Figure 6.5 shows the implementation of the message structure together with the possible logging command types.

For instance, when the periodic health task finished with its measurements, a message structure is created with the appropriate logging command that indicates health data needs to be logged. Furthermore, the data pointer of the message structure is set to point to the corresponding data in the ring buffer that contains measurement results. Then, the message structure is pushed into the queue of the logging task. If there is no more task that is ready to be executed and has higher priority than the logging task, the logging task pops the message from its queue and logs the health data to the appropriate file on the SD card. The logging task keeps executing the commands until there is no further message left in its queue.

```

1: enum LogTypes          /* SD Card Log Types */
2: {
3:     LOGTYPE_NONE = 0,
4:     LOGTYPE_SD_MOUNT,      /* Mount SD card command */
5:     LOGTYPE_SD_EJECT,     /* Eject SD card command */
6:     LOGTYPE_SD_GETSIZE,   /* Get SD card size */
7:     LOGTYPE_SD_PREVENTCHGDIR, /* Set PreventChangeDirectory flag */
8:     LOGTYPE_SD_ALLOWCHGDIR, /* Restore PreventChangeDirectory flag */
9:     LOGTYPE_CONF_RD,     /* Read config from SD card */
10:    LOGTYPE_CONF_WR,     /* Write config to SD card */
11:    LOGTYPE_ADC_FIRST,   /* Log first data chunk from ADC */
12:    LOGTYPE_ADC_CHUNK,  /* Log ADC data chunk */
13:    LOGTYPE_ACQ,        /* Log ACQ entry */
14:    LOGTYPE_EVENT,     /* Log EVENT entry */
15:    LOGTYPE_HEALTH,    /* Log HEALTH entry */
16:    LOGTYPE_IMUSTAT,   /* Log IMU entry */
17:    LOGTYPE_IMUDATA,   /* Log IMU data */
18:    NUM_OF_LOGTYPES
19: };
20:
21: typedef struct          /* Message Struct */
22: {
23:     uint8_t type;      /* See: LogTypes or BoltMessages enums */
24:     uint8_t* data;    /* Pointer to data */
25: } Message_t;

```

Figure 6.5: Logging commands and message structure

Commands for logging health data and system events are pushed immediately to the logging queue as soon as the data is ready to be logged. However, the ADC and the IMU generates data in every millisecond when they are active and frequently logging small amount of data to the SD card is extremely ineffective. Therefore, the data generated by the ADC and the IMU is aggregated to larger chunks and stored in their appropriate ring buffers. A logging command is issued for writing the recordings to the SD card once in every second. Thus, the large amount of data is written in bulk-mode once per second. Consequently, logging recording data takes far less time and making the logging procedure highly optimized.

### Timer Service Task

The timer service (or sometimes called daemon) task together with the timer command queue is private to the kernel and cannot be accessed directly. The application can perform specific timer API calls to operate the RTOS timers. The commands are processed within the timer service task, therefore the callback function that is called upon a timer expires is always executed within the context of the timer daemon task. Thus, it is crucial that the timer daemon and its priority is configured properly and the timers used by the application are

used appropriately [43]. In addition, the startup callback function described in Section 6.3.1 is executed in the context of the timer daemon task as well.

### Idle Task

The idle task, similarly to the timer service task, is implemented by the RTOS kernel and it is created automatically before the RTOS scheduler is started. The idle task is always the lowest priority task among the tasks. Other tasks can share the lowest priority with the idle task, however in this situation, the application should be designed carefully to avoid undesirable effects [46].

When there is no task ready to run, the idle task is executed. In each cycle of the idle task, the idle task hook function is called once. The user can implement additional functionalities in this callback function which are then executed in the context of the idle task when there is no other higher priority task able to run. In the geophone application, this idle task hook function is used to perform low-priority system monitoring operations. Furthermore, the low-power functionality is implemented in this callback function described in Section 6.4.

### 6.3.3 Interrupts

In contrast with the priority convention used in FreeRTOS, ARM Cortex-M4 core microcontrollers have a priority system that lower priority values indicate higher priority interrupts. In the FreeRTOS configuration, two special priority levels have to be carefully defined [47]:

- `configKERNEL_INTERRUPT_PRIORITY`
- `configMAX_API_CALL_INTERRUPT_PRIORITY`

The `configKERNEL_INTERRUPT_PRIORITY` defines the lowest priority used by the RTOS kernel. In most of the cases it should be set to the lowest priority of the microcontroller.

By setting the `configMAX_API_CALL_INTERRUPT_PRIORITY` value above the `configKERNEL_INTERRUPT_PRIORITY`, full interrupt nesting can be achieved. Furthermore, FreeRTOS does not completely disable interrupts; it only masks interrupts with the same or lower priorities (i.e. higher priority values in Cortex-M4 core microcontrollers) even in critical sections. Consequently, interrupts

with higher priorities of the `configMAX_API_CALL_INTERRUPT_PRIORITY` value are never delayed by the RTOS kernel [43].

There are a total number of 28 interrupt service routines used and implemented in the geophone platform. Out of the 28, nine are Cortex-M4 core related interrupt handlers and three of those are used by the RTOS. These three are the *SysTick Handler*, the *PendSV Handler* and the *SVC Handler*, which are responsible for handling the RTOS ticks, the context switches and the scheduler launch respectively. The remaining 19 interrupts are peripheral interrupts, including timers, DMA transfer interrupts, communication peripheral interrupts and external interrupts. Figure 6.6 illustrates the interrupts and their priorities.

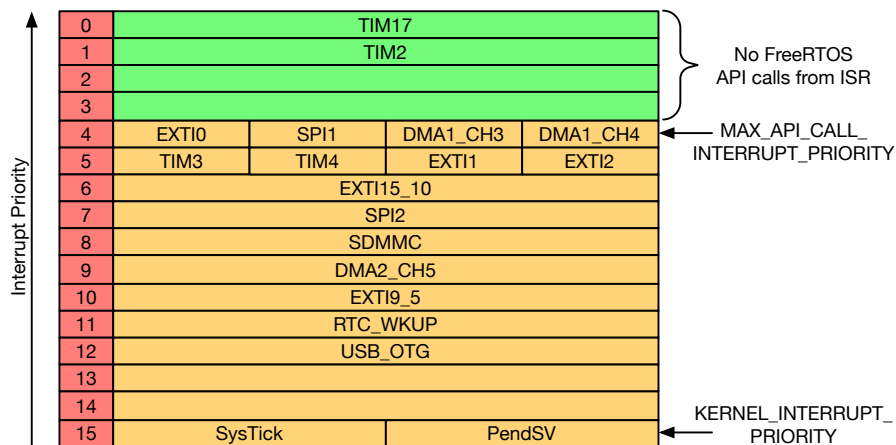


Figure 6.6: Interrupts and their priorities

## 6.4 Ultra-Low Power Software Design

The STM32L496VG microcontroller provides wide flexibility in configuration and features best-in-class power consumption among advanced Cortex-M4 core microcontrollers. In general, power and consumption optimization can be performed in two domains: the active or process phase and the inactive or sleep phase [48].

In active mode, the microcontroller can be selected to run in three different modes: “Run1” mode, “Run2” mode and “LP Run” mode. The highest processing performance can be achieved in Run1 mode, however this mode has the

highest energy consumption. After assessing the highest possible energy consumption of the system including the radio module during transmission and the SD during write sequence, the microcontroller running with maximum core frequency in Run1 mode with all peripherals enabled is only responsible for 10% of the overall consumption. Therefore, when the system is in active phase, the MCU is put in Run1 mode configured to the highest core clock frequency to minimize the time spent in active phase and to put the system in sleep mode as soon as possible.

There are 9 different low power modes of the STM32L496VG MCU as described in Section 5.2.1. The STOP 2 mode has the lowest energy consumption that still provides full RAM retention, enables wakeup events from most of the peripherals and has the RTC running without interruption. Therefore, the goal is to maximize the time spent in STOP 2 mode when there is no task to be executed in the system.

The system uses the idle task hook function to enter into STOP 2 mode. However, this has to be done with great care, because the system cannot enter into this mode when there are specific preventing conditions exist or there are still tasks running. Therefore, in each idle cycle, the application attempts to enter to STOP 2 mode by checking the following conditions:

- There are no commands received from the network to be executed.
- There is no ongoing recording of ADC samples.
- There is no ongoing recording of IMU samples.
- There is no pending time synchronization request.
- USB peripheral is inactive: both CDC and MSC classes are de-initialized, however the USB cable does not need to be detached.
- All queues that can unblock application tasks are empty.
- There is no task that is likely to be unblocked within the next 3 seconds. This is performed by checking whether the expected idle time calculated by the RTOS is greater than 3 seconds.

If all the above stated conditions are true, the system enters into STOP 2 mode. This is performed by executing the following sequence:



1. All interrupts are masked temporarily.
2. All internal and external peripherals are de-initialized and switched off.
3. The microcontroller pins are configured to analog input modes to minimize power consumption, except the pins that serve as wakeup sources.
4. RTOS tick is suspended.
5. All internal buses that are not required to operate in sleep mode are disabled.
6. External clock source (HSE) is disabled and the core is configured to use the high-speed internal oscillator (HSI) as the wakeup clock source.
7. Interrupt mask is cleared and the system is put into STOP 2 mode.

When there is an interrupt generated or a wakeup request issued in sleep mode, the system restores its original state by executing the above sequence in reverse and continues execution.

## 6.5 Configuration

Each node stores its most important configuration parameters on the microSD card. This includes the following values: the actual node ID, period of health task, period of time synchronization, operating mode, positive and negative trigger threshold values, trigger amplification gain settings, post-trigger duration, total triggering timeout duration, ADC PGA value, ADC format and IMU sampling frequency. These values are read and set upon powering on or resetting the node. Furthermore, the values on the SD card are updated after receiving new configuration values from the network.

The application firmware, including software modules and RTOS specific configuration values can be extensively configured in a separate configuration header file. In this file, more than 40 parameters can be set according to the application needs, including ADC, logging and triggering configuration, size of ring buffers, size of task stacks, queues and more. For the entire configuration list, please refer to the `config.h` file of the gophone software source code.

## 6.6 GSN Integration

With the DPP COM board connected, the node can connect to the PermaSense GSN as described in Section 2.2. The packets are forwarded to the backend, furthermore various commands that can be either broadcasted or targeted to a specific node can be sent from the backend.

### 6.6.1 Network Messages

The application processor is responsible for generating the application-related packets: the microcontroller generates the messages, encapsulates into appropriate packets and forwards it to BOLT. Eventually, the communication processor receives the messages from BOLT and sends it to the sensor network. However the communication processor alone is capable of generating packets, containing status information related to the radio communication.

The types of packets generated by the application processor can be divided into two groups. The *general packets* have the same structure across all types of sensor nodes, and all nodes should compose those packets the same way. These packets are the *node info packets* which contain information about the current node, including processor type, firmware, compile date, etc., and the *app health packets* which contain application related status information, including stack status, duty cycle, supply voltages, temperature data and more.

```

1: typedef struct
2: {
3:     uint8_t    component_id;    /* Application-specific component id */
4:     uint8_t    rst_flag;        /* Reset cause / source */
5:     uint16_t   rst_cnt;         /* Reset counter */
6:     uint8_t    mcu_desc[12];    /* MCU description, e.g. 'CC430F5147' */
7:     uint8_t    compiler_desc[4]; /* Compiler abbreviation (e.g. 'GCC') */
8:     uint32_t   compiler_ver;    /* Compiler version */
9:     uint32_t   compile_date;    /* Compilation time (seconds since 1970) */
10:    uint8_t    fw_name[8];      /* Name of the firmware/application */
11:    uint16_t   fw_ver;          /* Firmware version */
12:    uint32_t   sw_rev_id;       /* Repository revision number (GIT or SVN) */
13: } dpp_node_info_t;

```

Figure 6.7: Node info packet

Depending on the defined network and application requirements, these packets should be generated according to specific rules. In the case of the geophone platform, a node info packet has to be generated after every startup or reset,

furthermore a health packet must be generated regularly, with a few minutes of period. The health packets are generated every two minutes by default and the period can be changed individually for each node.

```

1: typedef struct
2: {
3:     uint32_t  uptime;           /* Uptime [seconds] */
4:     uint16_t  msg_cnt;         /* Number of received messages */
5:     uint16_t  core_vcc;        /* Core voltage [10-3 V] */
6:     int16_t   core_temp;       /* Core temperature [10-2 °C] */
7:     uint16_t  cpu_dc;          /* CPU duty cycle [10-2 %] */
8:     uint8_t   stack;           /* Stack watermark over the last period [%] */
9:     uint8_t   nv_mem;          /* Non-volatile memory usage [%] */
10:    uint16_t  supply_vcc;       /* Supply voltage [10-3 V] */
11:    uint16_t  supply_current;   /* Supply [10-5 A] */
12:    int16_t   temperature;      /* Temperature [10-2 °C] */
13:    uint16_t  humidity;        /* Humidity [10-2 %] */
14: } dpp_app_health_t;

```

Figure 6.8: App health packet

The other types of packets that are generated by the application processor are the *application-specific packets*. These packets are developed for a specific application. The application-specific packets are only generated by the corresponding sensor nodes.

Currently, the geophone platform has two application-specific packet types. The *acquisition packet* contains the most relevant information about the latest micro-seismic activity and recording. This packet is always transmitted after each seismic event, i.e. the acquisition procedure is completed and there is no further activity detected during the post-trigger interval. The packet structure is illustrated in Figure 6.9.

```

1: typedef struct
2: {
3:     uint64_t  start_time;       /* Timestamp of trigger */
4:     uint32_t  samples;          /* Total no. of samples */
5:     uint32_t  peak_pos_val;     /* Positive peak value of ADC */
6:     uint32_t  peak_pos_sample; /* Positive peak location (sample no.) */
7:     uint32_t  peak_neg_val;     /* Negative peak value of ADC */
8:     uint32_t  peak_neg_sample; /* Negative peak location (sample no.) */
9:     uint32_t  trg_count_pos;    /* Count of positive triggers */
10:    uint32_t  trg_count_neg;    /* Count of negative triggers */
11:    uint16_t  trg_gain;          /* Amplification gain value of TRG */
12:    uint16_t  trg_th_pos;       /* Positive TRG threshold value [mV] */
13:    uint16_t  trg_th_neg;       /* Negative TRG threshold value [mV] */
14:    uint8_t   trg_source;       /* Source of initial trigger */
15:    uint8_t   adc_pga;          /* ADC PGA value */
16: } dpp_geophone_acq_t;

```

Figure 6.9: Geophone acquisition packet

The second application-specific packet type contains the raw data from the inertial measurement unit. The structure of this packet type is depicted in Figure 6.10.

The GSN backed receives and processes the packets, converts and maps each value if needed. Furthermore, advanced queries can be submitted by defining the appropriate filters and conditions in order to request specific data and values from the backend database. Additionally, the retrieved data can be selectively visualized.

```

1: typedef struct
2: {
3:     uint16_t acc_x;      /* Accelerometer X-axis raw data */
4:     uint16_t acc_y;      /* Accelerometer Y-axis raw data */
5:     uint16_t acc_z;      /* Accelerometer Z-axis raw data */
6:     uint16_t mag_x;      /* Magnetometer X-axis raw data */
7:     uint16_t mag_y;      /* Magnetometer Y-axis raw data */
8:     uint16_t mag_z;      /* Magnetometer Z-axis raw data */
9: } dpp_imu_t;

```

Figure 6.10: Inertial measurement unit data packet

### 6.6.2 Network Commands

Various commands can be sent from the backend to the nodes. The commands can be either broadcasted or targeted to a specific node. The commands are transmitted within a special command packet. The command packet structure outlined in Figure 6.11 has two fields. The `type` field defines the command type and the `arg` field contains the corresponding arguments, for instance a new value of a configuration parameter.

```

1: typedef struct
2: {
3:     dpp_command_type_t type;      /* command ID */
4:     union
5:     {
6:         uint8_t arg[DPP_MSG_PAYLOAD_LEN - 2];      /* arguments */
7:         uint16_t arg16[(DPP_MSG_PAYLOAD_LEN - 2) / 2];
8:     };
9: } dpp_command_t;

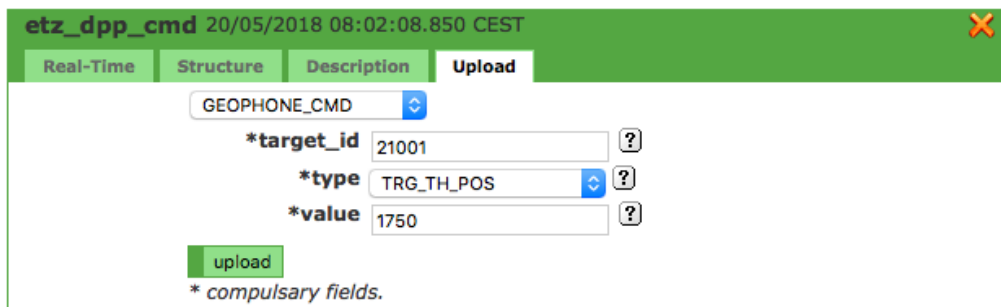
```

Figure 6.11: Network command structure

The geophone platform can receive and execute 9 different commands. These commands include general instructions, for instance to perform reset, self-test

or change operation mode. The rest of the commands are used for changing the configuration parameters of the nodes. These include the configuration of the triggering circuit parameters (e.g. the triggering thresholds, amplification stages), the ADC parameters (e.g. the PGA value, data format) and the acquisition parameters (e.g. the post-trigger duration, IMU sampling frequency and maximum timeout). The `dpp_geophone_command_type.h` file contains all commands that can be issued to the geophone nodes.

Additionally, special commands can be issued to the communication processor. By executing these commands, the network protocol and its parameters can be adjusted remotely.



The screenshot shows a web browser window titled "etz\_dpp\_cmd" with a timestamp of "20/05/2018 08:02:08.850 CEST". The interface has a green header bar with a close button (X) on the right. Below the header, there are four tabs: "Real-Time", "Structure", "Description", and "Upload". The "Upload" tab is active. The main content area contains a form for sending a command. At the top, there is a dropdown menu labeled "GEOPHONE\_CMD". Below it, there are three compulsory fields, each with a question mark icon: "\*target\_id" with the value "21001", "\*type" with the value "TRG\_TH\_POS", and "\*value" with the value "1750". At the bottom left of the form is a green "upload" button. Below the form, there is a note: "\* compulsory fields."

Figure 6.12: Sending command from GSN backend

Figure 6.12 demonstrates how to send a command to the geophone node with the ID of 21001. The command instructs the node to re-configure its positive triggering threshold value to 1750 mV.

*This page intentionally left blank.*

# Evaluation

---

This chapter describes the evaluation of the geophone platform. The evaluation includes extensive hardware analysis, the evaluation of the triggering subsystem, load and temperature tests, comprehensive power analysis and software performance analysis.

## 7.1 Analog Frontend

This section presents the evaluation of the analog frontend of the triggering subsystem.

### 7.1.1 Stage Gains

The first stage of the triggering circuit has a nominal amplification gain of 20. This gain is determined by the R5 and R8 resistors<sup>1</sup>:

$$A_{1_{nom}} = \frac{R_5}{R_8} = \frac{280k\Omega}{14k\Omega} = 20 \quad (7.1)$$

To verify the amplification gain, a signal generator is used to produce a sine-wave with  $100mV_{p-p}$  amplitude and  $10Hz$  frequency. According to Equation (7.1), the output signal amplitude should be  $2.0V$ . The output signals of the signal generator and the first stage amplifier are connected to an oscilloscope, thus the gain of the amplifier can be measured by comparing the amplitudes of both signals.

Figure 7.1 shows the signals captured with an oscilloscope. The yellow signal is the output of the amplifier and the blue is the output of the signal generator.

---

<sup>1</sup> Please refer to the schematics included in Appendix B.

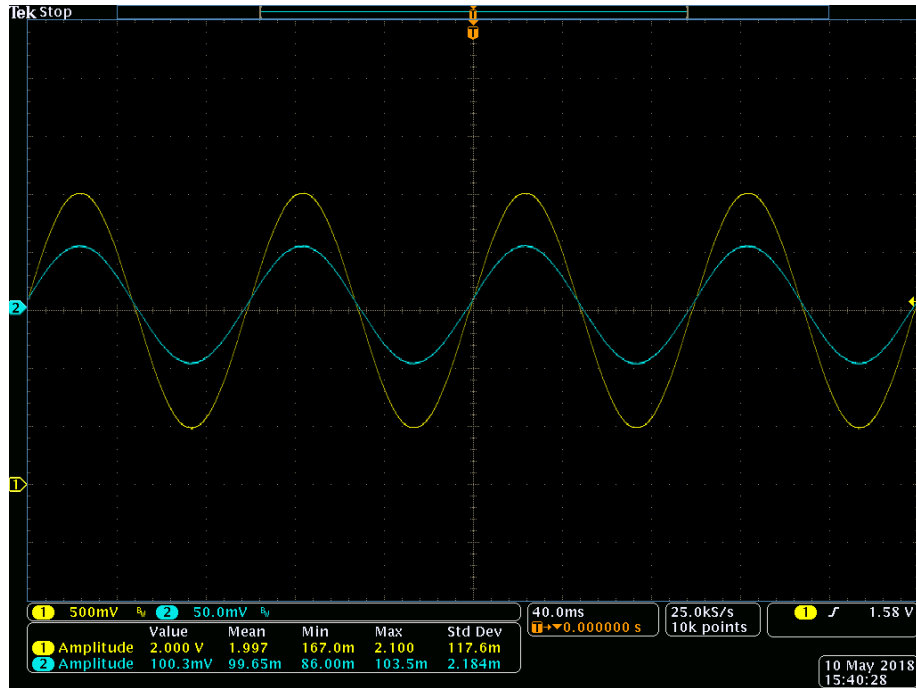


Figure 7.1: Single-stage gain

The measure function of the oscilloscope shows the amplitude of the amplified signal in the bottom-left corner. Consequently, the measured gain of the first amplification stage is:

$$A_1 = \frac{2.00V}{100.3mV} = 19.94 \quad (7.2)$$

which is within 1% error compared to the nominal value.

### Dual-Stage Amplification

When the second amplification stage is enabled, the combined nominal gain is approximately 207. This is determined by the first gain and the R4 and R6 resistors from the second stage:

$$A_{2_{nom}} = A_{1_{nom}} * \frac{R_4}{R_6} = 20 * \frac{280k\Omega}{27k\Omega} = 207.4 \quad (7.3)$$

To verify the amplification gain when both stages are enabled, a 10Hz,



$10mV_{p-p}$  sine wave is used. The full range is  $3.0V$ , therefore the amplitude of the test signal cannot be higher than  $14mV$ .



Figure 7.2: Dual-stage gain

Similarly to the previous setup, the amplitude of the output signal is compared to the original signal. Figure 7.2 shows the setup and the signals. The measured gain of the dual-stage amplification circuit is:

$$A_2 = \frac{2.040V}{10.17mV} = 200.59 \quad (7.4)$$

Based on Equation (7.2) and Equation (7.4), it can be clearly stated that the triggering subsystem of the geophone platform has perfectly tuned amplification settings to achieve 20x gain in single-stage and 200x gain in dual-stage modes.

### 7.1.2 Frequency Responses

The triggering circuit is built to amplify low-frequency signal components, i.e. seismic signals, and suppress high-frequency signals which can be interpreted as noise from the perspective of the system. In order to measure how “well” the

circuit performs these duties, the frequency response of the triggering circuit is measured. Two experiments are carried out: one with the second amplification stage disabled, and another analysis with the second stage enabled. Both stages implement an active low-pass filter with a nominal cutoff frequency of  $1kHz$  based on Equation (4.3).

The frequency response of a system can be measured by using several methods [49]. The commonly used method is to apply an impulse to the system and observe the output, i.e. the impulse response of the system. Although this method works well in simulations, in real world applications it is rather impossible to produce a perfect impulse and this method has other disadvantages. First, applying imperfect impulses result in inaccurate impulse responses, secondly a short and strong pulse may drive the system into a nonlinear domain. Therefore, the sine sweep method is used to accurately measure the frequency response of the analog frontend.

The sine sweep method involves applying and sweeping a constant-amplitude sine wave through the relevant bandwidth. The amplitude of the output is measured and compared to the input.

### Single-Stage Frequency Response

When only the first amplification stage is active and used for triggering, the total amplification gain is 20. The maximum output signal amplitude cannot exceed  $3.0V$ , therefore, a  $100mV_{p-p}$  sine wave is used as an input signal. The signal is swept over the bandwidth of  $1Hz$  to  $100kHz$  with 30 different frequency values. For each frequency, the gain of the system is calculated in decibels by Equation (7.5). The nominal amplification of the single-stage circuit is  $A = 20 * \log_{10}(20) = 26.02dB$ .

$$A = 20 * \log_{10} \left( \frac{V_{out_{p-p}}}{V_{in_{p-p}}} \right) \quad (7.5)$$

The single-stage frequency response is shown in Figure 7.3. The results indicate that the amplifier with the low-pass filter works as intended: low-frequency signals are amplified and higher frequency signals are attenuated. The  $-3dB$  signal attenuation is around  $900Hz$ , i.e. this is the cutoff frequency of the filter.

The attenuation for  $1kHz$  signals is  $-4.7dB$  and  $10kHz$  signals are attenuated with  $-17dB$  compared to the nominal amplification gain.

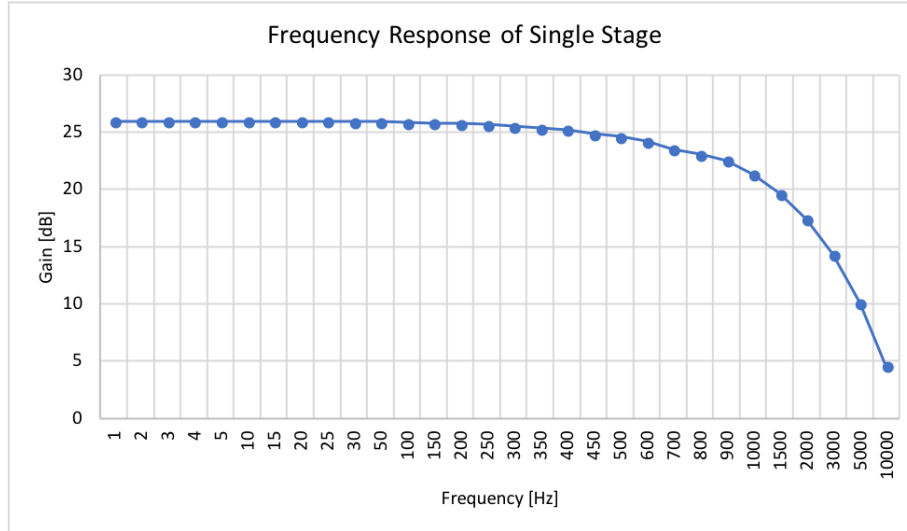


Figure 7.3: Frequency response of single stage

### Dual-Stage Frequency Response

When the second amplification stage is enabled, the total amplification gain is 200 – which equals  $A = 20 * \log_{10}(200) = 46.02dB$  according to Equation (7.5). This setup represents two low-pass filters cascaded together, therefore the system can be recognized as a second-order low-pass filter. Similarly to the previous analysis, the maximum output signal amplitude cannot exceed  $3.0V$ , therefore, a  $10mV_{p-p}$  sine wave is used as an input signal. The sine wave is swept over the same bandwidth, i.e. from  $1Hz$  to  $10kHz$  with 30 frequency measurement points.

In contrast to a first-order low-pass filter, a second-order filter has a  $-6dB$  gain at the corner frequency and the stop band roll-off is  $-40dB/decade$  which is twice the value of a first-order filter.

The dual-stage frequency response is shown in Figure 7.4. There is  $-6dB$  attenuation of signals with  $900Hz$  frequency. The attenuation for  $1kHz$  signals is  $-7.3dB$  and  $10kHz$  signals are attenuated with  $-41dB$  compared to the nominal amplification gain.

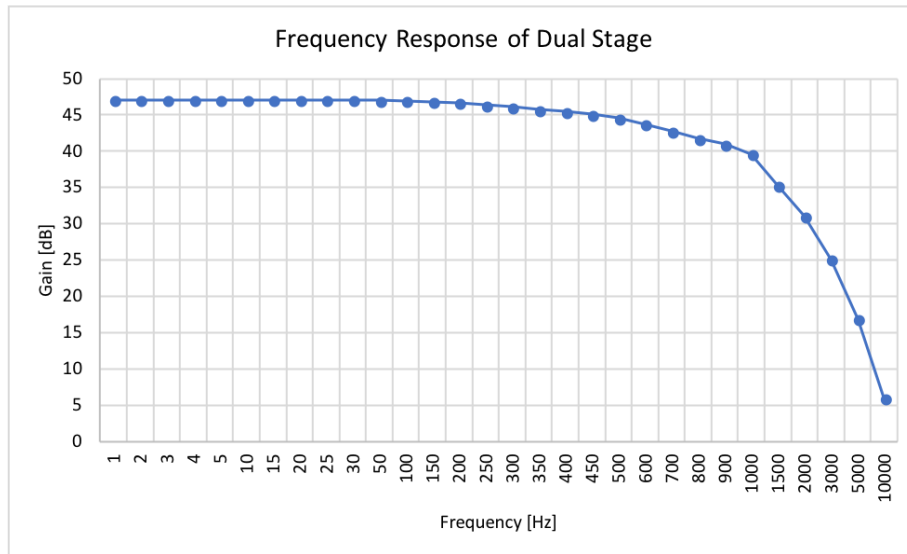


Figure 7.4: Frequency response of dual stage

Based on the above results, the triggering circuit has overall excellent signal amplification and filtering parameters both in single-stage and dual-stage modes.

## 7.2 Triggering Attributes

In this section, the attributes of the triggering circuit is evaluated, focusing on the quality analysis of threshold voltages and the trigger propagation time.

### 7.2.1 Threshold Voltages

Establishing precise trigger thresholds are crucial to achieve high-quality and accurate triggering mechanism, thus to effectively detect micro-seismic activities when the system is inactive.

The MAX5532 dual-channel digital-to-analog converter is responsible for establishing low-noise and accurate threshold voltages. Its datasheet guarantees typical offset error of  $\pm 1mV$  and maximum offset error of  $\pm 20mV$  for all input codes for the entire operating temperature range.

Extensive measurements and signal analysis confirmed that both of the threshold voltages can be set safely and precisely within the entire output range with

10mV steps in the tested temperature range of  $-25^{\circ}\text{C}$  to  $+30^{\circ}\text{C}$ .

### 7.2.2 Trigger Propagation

Trigger propagation is the duration between the point when signal crosses a threshold and the point when an interrupt is issued to the microcontroller. It is essential that this propagation delay stays within the microsecond order of magnitude, therefore the triggering circuit provides fast wakeup.

To precisely measure this value, a 10Hz square wave with  $100\text{mV}_{p-p}$  amplitude is connected to the input of the triggering circuit and the corresponding signals are observed with an oscilloscope. Figure 7.5 shows the setup of this measurement.



Figure 7.5: Trigger propagation

The figure depicts a captured event when the raising input signal exceeds the positive triggering threshold and consequently, the corresponding comparator output is driven high. The input signal is connected to the yellow-colored channel. The positive triggering threshold is connected to the blue channel and the corresponding output of the comparator is connected to the green channel. The

triggering propagation can be measured simply with the oscilloscope cursors:

$$t_{pdtrg} = t_{interrupt} - t_{trigger} = 24.16\mu s \quad (7.6)$$

### 7.3 Wakeup Time Analysis

It is crucial that as soon as there is a seismic event detected, the system wakes up, switches on the required components and starts recording samples as quickly as possible. The total required wakeup time from triggering to the first ready sample from the ADC consists of the summary of the following durations:

1. Triggering propagation
2. Microcontroller wakeup
3. Configuration of system clocks and GPIOs
4. Power on and configuration of required peripherals
5. ADC configuration and sampling launch
6. Delay until the first sample is ready and valid

The duration of each process listed above is measured individually. Table 7.1 shows the measurement results with the required total wakeup time. As a result of meticulous hardware and deliberate software design, the system guarantees that the first valid recording sample arrives in  $2.62ms$  after the triggering threshold is exceeded. Considering an input signal with  $10Hz$  frequency ( $100ms$  period), the total wakeup time required by the system (i.e. the loss of meaningful signal) is less than 3% of the signal period.

### 7.4 Power Analysis

This section covers the extensive power analysis of the geophone platform. In order to carry out precise power measurements, the N6705A DC power analyzer from Keysight Technologies is used, with a high-precision source measure unit (SMU) fitted in one of its module slots. Unless otherwise stated, the power analyzer is configured to output  $3.0V$  DC voltage and it is connected directly to

<b>Action</b>	<b>Duration</b>
Trigger propagation	24.16 $\mu s$
Microcontroller wakeup	9.2 $\mu s$
Hardware & peripheral configuration	940.0 $\mu s$
ADC configuration and sampling launch	459.12 $\mu s$
Delay until valid first sample	1187.52 $\mu s$
Total:	2.62ms

Table 7.1: Wakeup times

the system power rail, thus omitting the loss of the DC/DC converter from the measurement results.

#### 7.4.1 Consumption of Components in Low Power State

In order to successfully measure the consumption of each relevant component, the measurements were carried out in parallel with the assembly of the the three prototypes. Therefore, the consumption of each individual component could be measured systematically and the values could be compared across the prototypes. Unless otherwise noted, the consumption values indicate the average of the individual measurement values from the three prototypes.

##### Single-Stage Triggering Circuit

First the required components were soldered to the PCBs so that the triggering circuit could operate in single-stage mode. The components required for the second-stage were not soldered. Therefore, the prototypes contained only the filtering circuit, first stage amplifier, DAC and comparator. The measured average current drain of this setup was 22.8 $\mu A$ .

##### Dual-Stage Triggering Circuit

In this setup, the remaining components of the entire triggering circuit were soldered. When the second stage was disabled, the measured average consumption was 25.8 $\mu A$ . The reason why this value is higher than the value measured in the

single-stage scenario described above is the current leakage through the resistor divider which sets the DC bias of the second amplifier.

When the second stage was switched on by pulling manually the gate of the P-FET to ground, the measured current drain was  $46.9\mu A$ .

### **Always-On System**

In the next step, the microcontroller and its required external components were soldered in addition to the previous setup. The microcontroller was programmed to go to STOP2 mode with the required wakeup sources enabled. This setup contains all the components which are always-on in the system. The measured average current consumption of the system was  $29.0\mu A$  in the case of single-stage triggering and  $49.9\mu A$  in the case of dual-stage triggering.

### **DC/DC Conversion**

To evaluate the power dissipation of the power subsystem, the voltage regulator together with its external components were soldered onto the prototypes. The DC analyzer was configured to output  $3.6V_{DC}$  voltage (nominal voltage of the battery) and it was connected to the pins of the battery holder. The total measured current drain was  $30.6\mu A$  with single-stage triggering setup and  $51.7\mu A$  with dual-stage triggering setup. These results confirm that the average power dissipation of the DC/DC conversion is as low as  $1.7\mu A$  when the system is in low power state.

### **Full System**

The remaining components were expected not to have an effect on the low power state dissipation, because all the components were powered down in sleep mode. After soldering all remaining components to the prototypes, the measurements confirmed the expectations and the consumption values remained the same as presented in the sections above. Table 7.2 summarizes the results of system power analysis when the system is in inactive state.



Setup	Average Current Drain at 3.0V	
	Without DC/DC	With DC/DC
Single-stage triggering circuit	22.8 $\mu A$	n.a.
Dual-stage triggering circuit (off)	25.8 $\mu A$	n.a.
Dual-stage triggering circuit (on)	46.9 $\mu A$	n.a.
Always-on system with single-stage	29.0 $\mu A$	30.6 $\mu A$
Always-on system with dual-stage	49.9 $\mu A$	51.7 $\mu A$
Full system with single-stage	29.0 $\mu A$	30.6 $\mu A$
Full system with dual-stage	49.9 $\mu A$	51.7 $\mu A$

Table 7.2: Summary of power analysis in low power state

### 7.4.2 Measurement Scenarios

This section presents the extensive power analysis carried out for several application scenarios. A total number of eight scenarios were performed and analyzed. Appendix D contains detailed graphs and power traces of additional scenarios.

The power analysis for all scenarios was carried out with the DC power analyzer connected directly to the system power rail and the output voltage was set to 3.0V, thus all events of the scenarios could be analyzed without the effect of the DC/DC conversion.

#### Triggering and 3-second Recording without COM board

The configuration parameters of this scenario is summarized in Table 7.3. A power trace with 10-second duration was captured.

Configuration	Parameter
ADC sampling rate	1k $sps$
IMU sampling rate	10 $Hz$
Post-triggering	3 $s$
COM board	Not attached
Total duration	10 $s$

Table 7.3: Configuration parameters for 1<sup>st</sup> scenario

Figure 7.6 shows the captured power trace of this scenario. The device is initially in sleep mode, draining only  $28\mu A$  current. After gently shaking the underlying table, the node immediately wakes up, switches on and launches the ADC and IMU, then starts recording. The average consumption of the node is  $18.1mA$ . After collecting one second of samples, the device configures the SD card and starts logging the collected data. The peak current drain when the device is writing data to the storage unit is  $67.4mA$ . In parallel, the node continues recording data. Because the SD card is now initialized, the average current consumption increases from  $18.1mA$  to  $19.8mA$ . The little spikes with  $23mA$  peaks during recording are caused by the IMU. One can verify that the number of spikes matches the sampling rate of the IMU. During the 3-second post-triggering, the device does not detect any further activity and goes back to sleep mode.

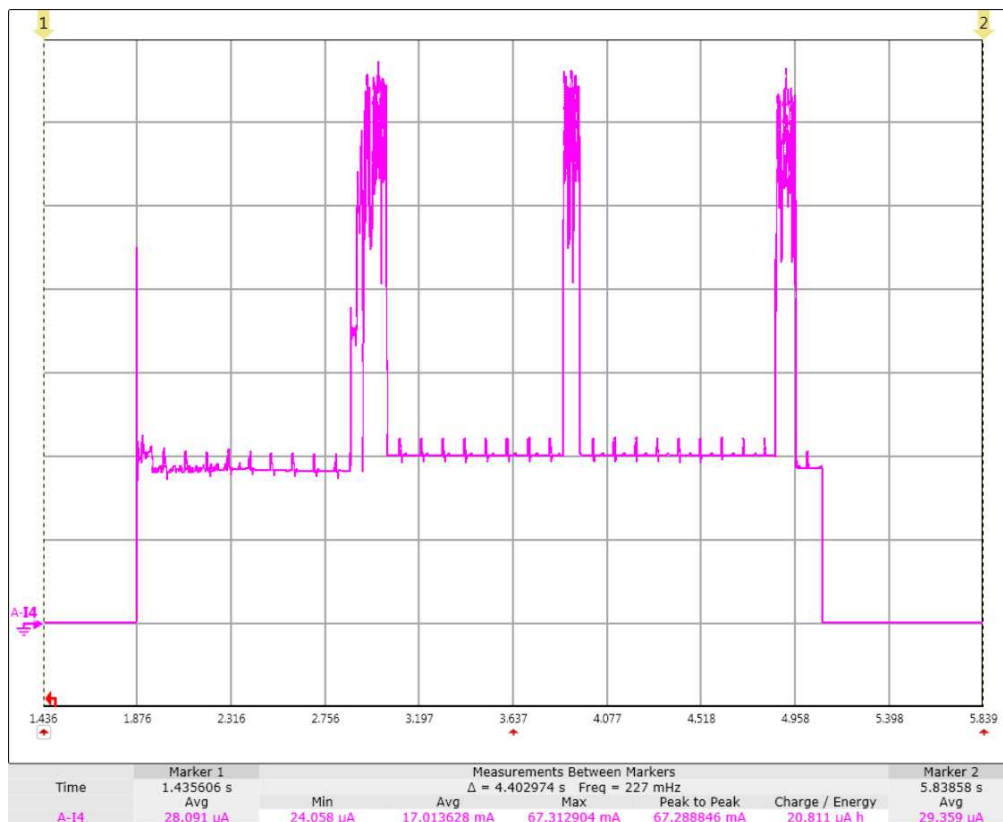


Figure 7.6: Triggering and 3-second recording without COM board

This scenario was carried out with the same analysis parameters, however

the IMU was completely disabled. During recording, the average current consumption drops to  $16.5mA$  in the first recording period and to  $18.2mA$  after the SD card has been initialized. The detailed power trace of this scenario can be found in Appendix, Figure D.1.

### Triggering and 3-second Recording with COM board

The configuration parameters of this scenario is summarized in Table 7.4. A power trace with 10-second duration was captured.

Configuration	Parameter
ADC sampling rate	$1ksp/s$
IMU sampling rate	$10Hz$
Post-triggering	$3s$
COM board	Attached
Total duration	$10s$

Table 7.4: Configuration parameters for 2<sup>nd</sup> scenario

This scenario is greatly similar to the previous configuration, except this time the COM board is attached and connected to the wireless sensor network. Figure 7.7 depicts the power trace of this scenario. The average sleep current drain is  $34\mu A$ , although current consumption during active mode remains the same<sup>1</sup>. After finishing the sampling procedure, an acquisition packet is generated with the most relevant information about the recording and forwarded to the COM board which then transmits the packet to the network. The peak current consumption during radio transmission is  $25mA$  in this scenario, however this value greatly depends on configuration and network parameters, location, signal strength, etc.

Similarly, this scenario was also performed with disabled IMU. The detailed power trace of this additional scenario can be found in Appendix, Figure D.2.

<sup>1</sup> Please note that the absolute peak current caused by writing data to the SD card varies between  $65mA$  and  $85mA$  across all scenarios depending on the data that is being logged and the actual state of SD card.

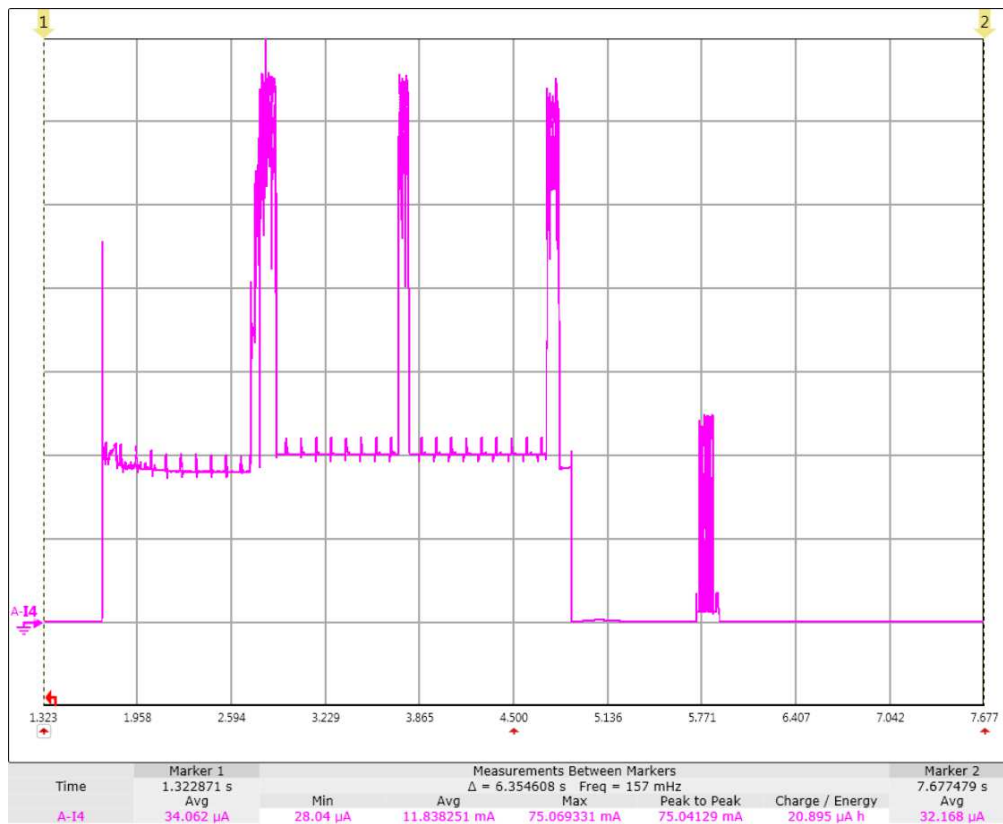


Figure 7.7: Triggering and 3-second recording with COM board

### Periodic generation of Status and Health Data

Depending on the actual node configuration, a node periodically wakes up, performs status and health data measurements, logs the collected data, then goes back to sleep. If the node is connected to the wireless sensor network, it also generates a health packet and forwards it to the network before going back to sleep. Figure 7.8 shows the power trace of this scenario without radio transmission.

The node is configured to wake up with a period of two minutes. The RTC remains operational in sleep mode, therefore it is used to generate a wakeup interrupt after every two minutes. After the interrupt, the node wakes up and performs the required operations, including temperature, humidity and battery voltage measurements. In addition, other health values are generated, including SD card status, RTOS status and more. During measurements, the average current consumption is  $35\text{mA}$ . The three peaks in the power trace with approx-

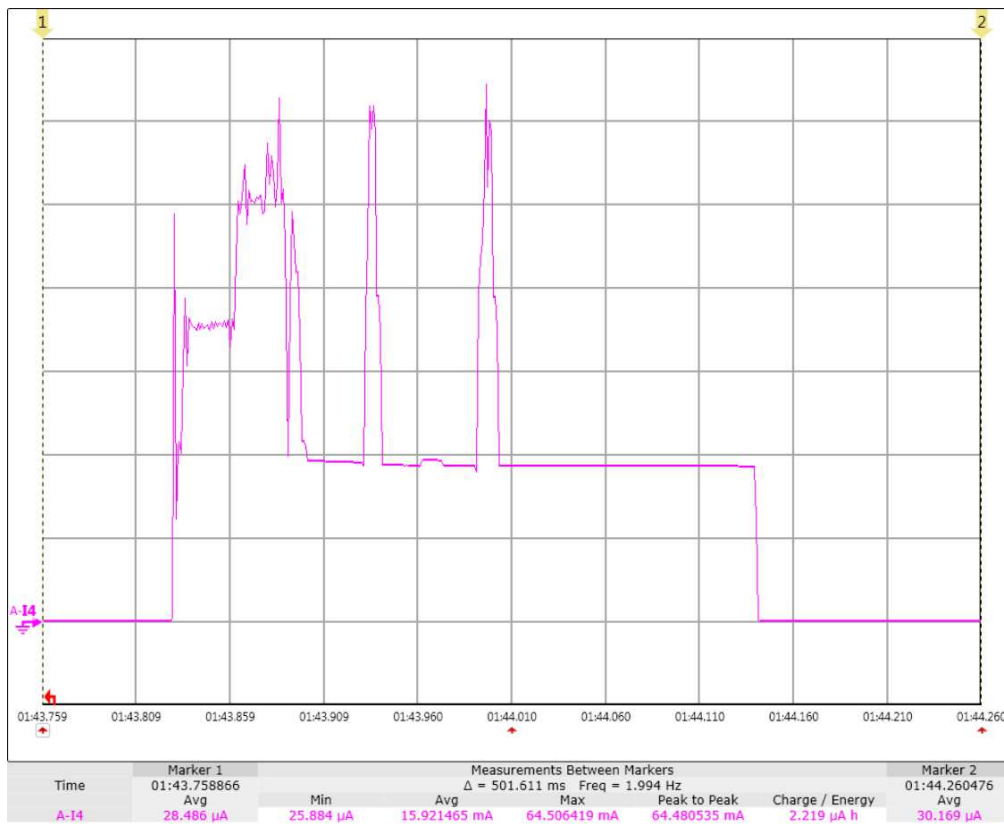


Figure 7.8: Periodic generation of status and health data

imately  $65\text{mA}$  current drain for each spike represent the SD card initialization sequence, the free SD card space retrieval and logging the health data to the SD card respectively.

### Network Command Reception and Execution

During this test scenario, a network command was sent from the backend and the command reception and execution was captured. Figure 7.9 depicts the power trace of this scenario.

The device is initially in sleep mode. The first spikes with  $22\text{mA}$  peak current drain represent the command packet reception by the communication board. Then the application microcontroller wakes up, receives the command from the communication board and executes the command which intended to reconfigure a triggering threshold parameter. Then the configuration file located on the SD

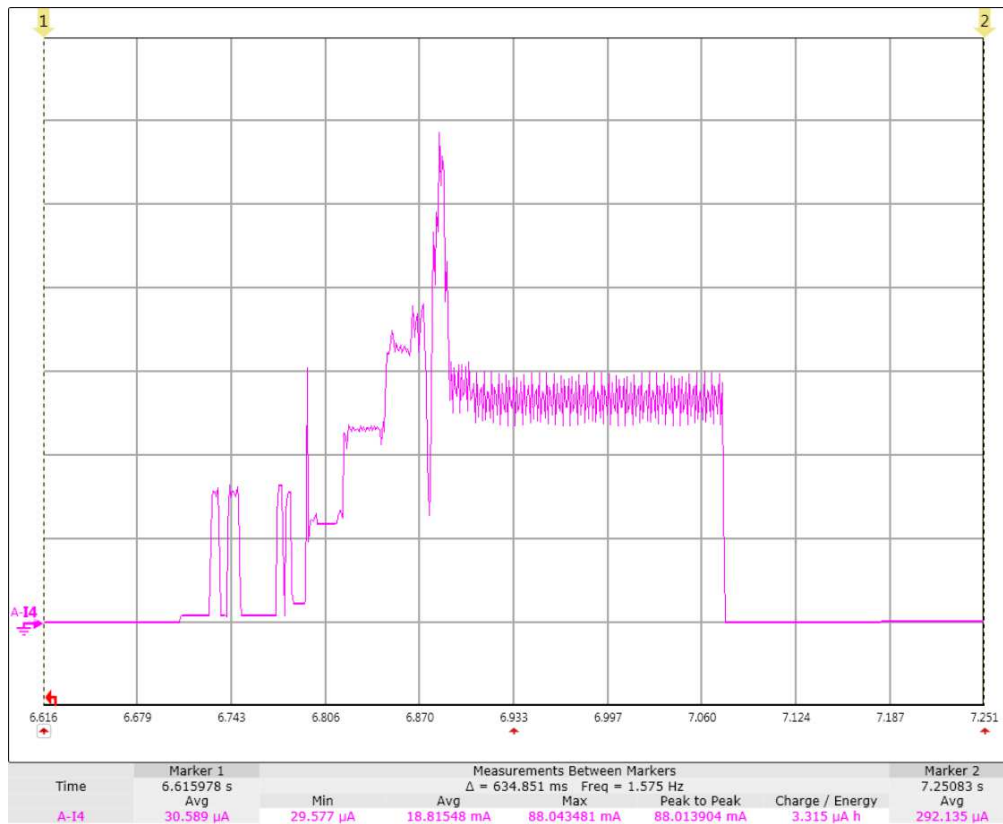


Figure 7.9: Network command reception and execution

card is updated: the spike with  $88\text{mA}$  peak current denotes the write cycle. After command execution, the buzzer is activated for  $200\text{ms}$  to give feedback about the successful execution, resulting in a high-frequency oscillation in the power trace. Finally, the node goes back to sleep mode.

### Radio Transmission

This scenario represents a packet transmission to the network. During the recorded scenario, only the radio is active in transmission mode and the rest of the system is in sleep mode. Figure 7.10 shows the power trace of a packet transmission in detail.

The transmission sequence consists of 9 events, resulting in 9 spikes in the power trace with current peaks between  $22\text{mA}$  and  $25\text{mA}$ . The 1<sup>st</sup>, 3<sup>rd</sup>, 4<sup>th</sup> and 8<sup>th</sup> spikes represent scheduling cycles in the eLWB communication protocol. The

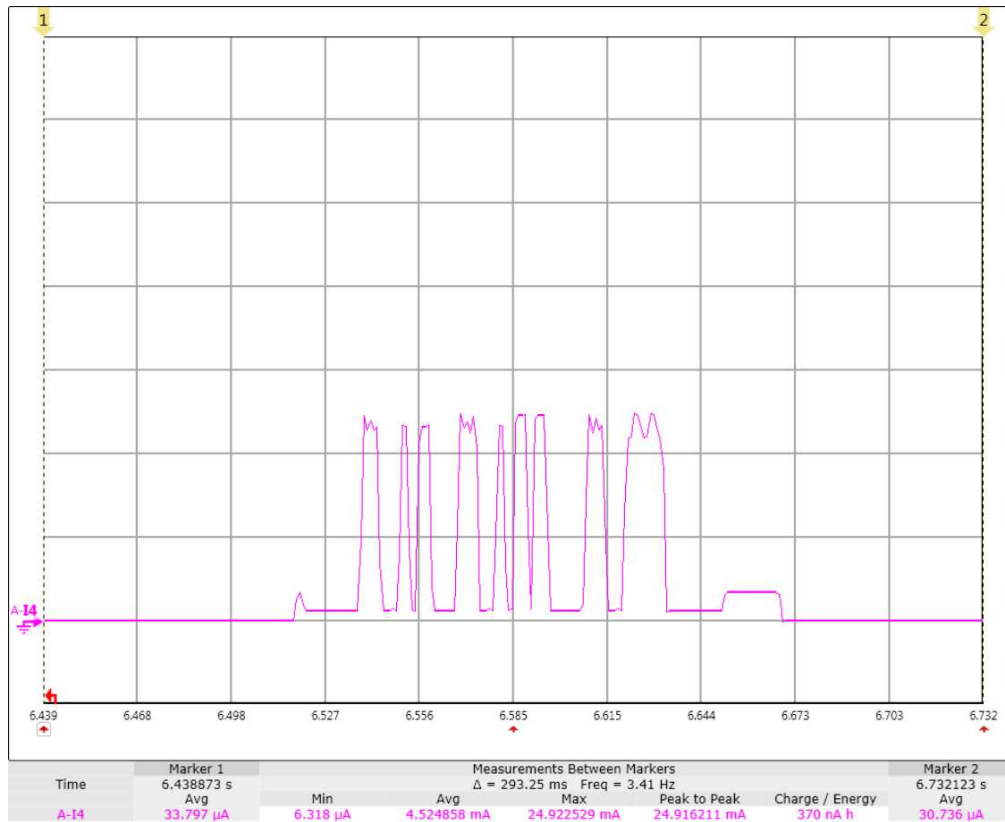


Figure 7.10: Radio transmission

2<sup>nd</sup> spike denotes the contention. The 5<sup>th</sup>, 6<sup>th</sup> and 7<sup>th</sup> spikes indicate the request slots of the nodes. The number of requests indicate the number of nodes in the network. Finally, the 9<sup>th</sup> spike represents the actual packet transmission.

In addition to this scenario, a radio beacon is captured. The power trace of the beacon is included in Appendix, Figure D.3.

### 7.4.3 Battery Life Estimation

The device is equipped with a 13Ah military-grade battery. Considering a theoretical scenario, when the device remains entirely in sleep mode without detecting any seismic activity and without performing periodic health monitoring, the expected battery life is more than 47 years. If the periodic health monitoring is enabled with a two-minute period, the battery would last for more than 23 years.

Considering a realistic scenario, when micro-seismic activities are detected and recorded once per hour with 3-second post-triggering, furthermore the periodic health monitoring is enabled with a period of two minutes, the estimated battery life is more than 16 years. If the device is deployed to an extremely busy environment where it wakes up once a minute (i.e. more than 1400 seismic events occur per day) and performs recording for 3 seconds after each wakeup, the battery would still last for an entire year.

## 7.5 Triggering and Load Test

The triggering and the entire system operation was tested thoroughly to ensure highest product quality and robustness. Software development was carried out in parallel with extensive testing. Generally, development took place during workdays. In the evenings and during weekends, the devices were constantly running with various test configurations with logging and debugging functionalities enabled. Upon the next workday, the logs were analyzed for errors and bugs, and the software was patched with appropriate fixes.

In early development phase, the input of the devices were connected to a signal generator that produced  $10Hz$  sine wave with an amplitude higher than the triggering thresholds. Therefore, the nodes were constantly forced to trigger to the test signal and performed continuous recordings and logging. With this test method, approximately 400MB of data was generated and logged by a single node each day. This heavy load test with various parameters was running on the nodes for several weeks to ensure high quality software operation with maximum robustness.

After reaching stable and reliable operation, the nodes were placed to various locations within the ETZ<sup>1</sup> building of ETH Zürich to collect measurement data and to analyze overall platform operation and performance. These locations included busy corridors, work desks, places next to vibrating and noisy machines (e.g. refrigerator), etc. The nodes were continuously operational and they were transmitting packets to the wireless sensor network, thus the data could be immediately analyzed and visualized by accessing the network backed with a web browser.

---

<sup>1</sup> Building of the Department of Electrical Engineering.



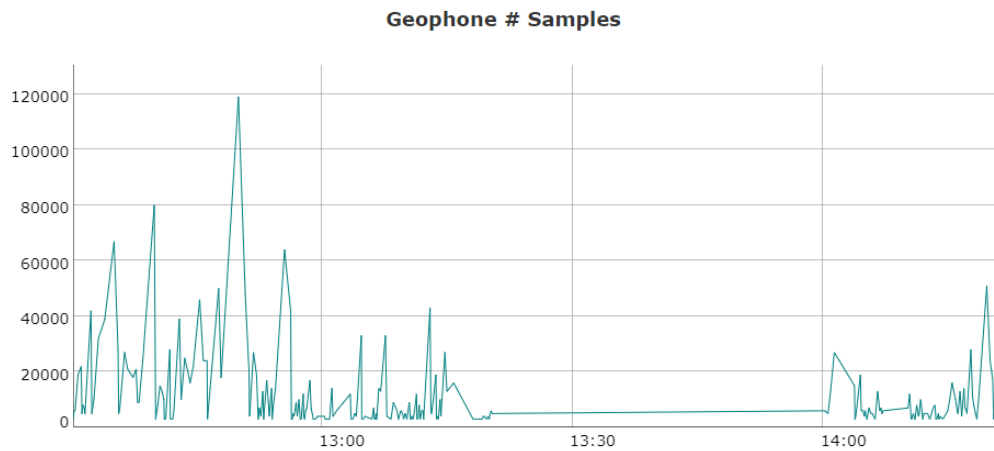


Figure 7.11: Visualization of geophone samples

For instance, Figure 7.11 depicts geophone samples recorded and visualized between 12:30 and 14:30. The node was lying on an office desk and it was configured with a sensitivity that it triggered upon human activities, e.g. typing on the keyboard or clicking with the mouse. In other words, it was a simple way to track whether someone was sitting and working at the desk. On the day of recording, the lunch break took place approximately between 13:15 and 14:00.

## 7.6 Temperature Test

To simulate operation in high-alpine environment, one of the nodes was put in a freezer set to  $-25^{\circ}\text{C}$ . The periodic health packet containing the node temperature was sent to the network every two minutes. The node was put into the fridge shortly after 13:00 and it was taken out and put on the top of the refrigerator at 15:15. Figure 7.12 shows the node temperature over time.

The node was configured to pick up the activities coming from the compressor of the freezer. Figure 7.13 depicts the activity recorded by the node, furthermore it can be observed when did the refrigerator turn on and off its compressor.

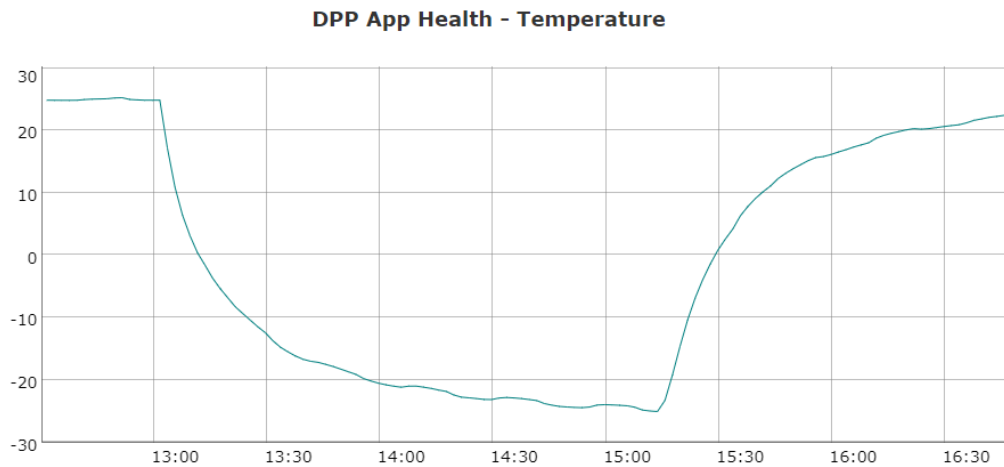


Figure 7.12: Temperature test

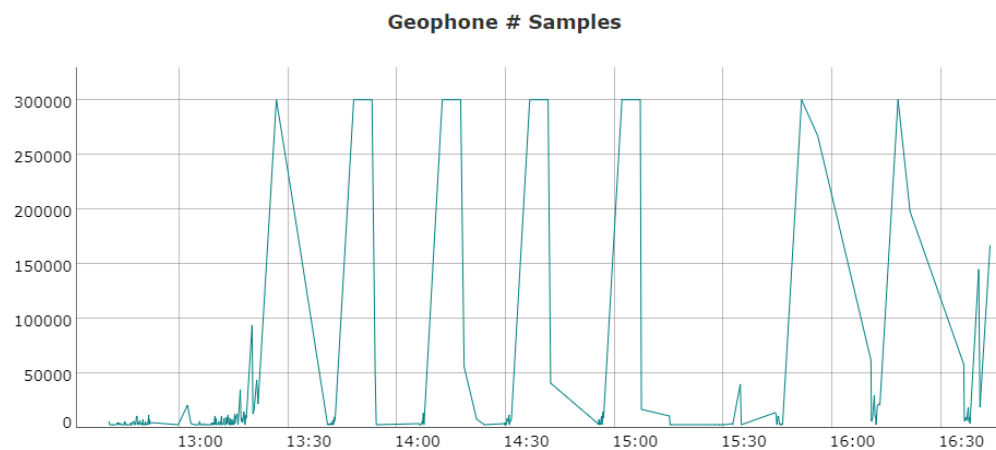


Figure 7.13: Geophone samples during temperature test

## 7.7 Software Performance Analysis

Beside using oscilloscopes and logic analyzers to accurately assess and measure system performance, the SWV real-time trace technology is used described in Section 5.2.2, in combination with the SystemView<sup>1</sup> software utility from SEGGER to measure and visualize software and RTOS performance in real time. The FreeRTOS kernel is modified to generate tracing information of interrupts

<sup>1</sup> SEGGER SystemView - <https://www.segger.com/products/development-tools/systemview/>

and tasks. When the system goes into sleep, the real-time trace output stops and it automatically continues when the node wakes up. The full-size figures of the following RTOS traces are included in Appendix E.

Figure 7.14 shows the trace of a 3-second recording. The system wakes up for a negative triggering interrupt, switches on the digitizer subsystem and the IMU sensor, performs recording for three seconds, logs the obtained data, creates a network packet, forwards it to the communication board and finally goes back to sleep.

The trace shows the relevant interrupts of the system and task executions in time. Some traces of interrupts and tasks might look continuous in the picture. This is because events with higher frequencies are visualized with continuous traces when the timeline is zoomed out. When the system is running and recording data from both the ADC and the IMU, the average frequency of trace events is more than 6500 per second. During this 3-second trace, the average interrupt frequency is approximately  $3200\text{Hz}$ , and the total load of ISRs is 3.70%. The RTOS and the geophone software is greatly optimized that the overall CPU load is less than 15% even during the busiest periods. The measured average RTT throughput is  $7.21\text{kB/s}$  with a peak throughput of  $37.78\text{kB/s}$  during the most eventful period. The FreeRTOS kernel and tracing functions are modified and optimized to support tracing of high-frequency real-time events without RTT overflows or event drops.

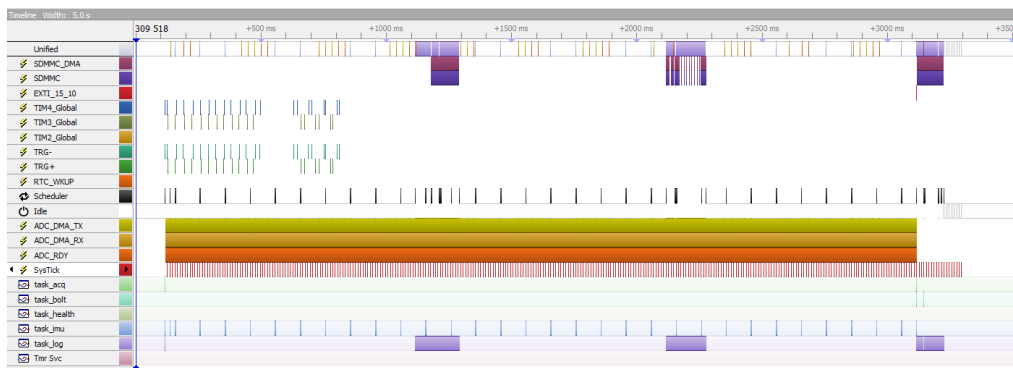


Figure 7.14: Trace of a 3-second recording

Upon zooming to the beginning of the trace, one can observe the wakeup and the procedure of launching the recording shown in Figure 7.15. The system wakes up for a negative triggering interrupt and quickly reinitializes itself. The

hardware reconfiguration takes place right in the interrupt routine to achieve the fastest wakeup time. The acquisition task is unblocked by issuing a direct-to-task notification from the interrupt service routine. The task immediately configures and launches the ADC with DMA support. In the meantime, the IMU task is unblocked which performs the configuration and sampling of the IMU. One can verify that the latency between the initial triggering interrupt and the first received and valid ADC sample is less than  $2.6\text{ms}$ .

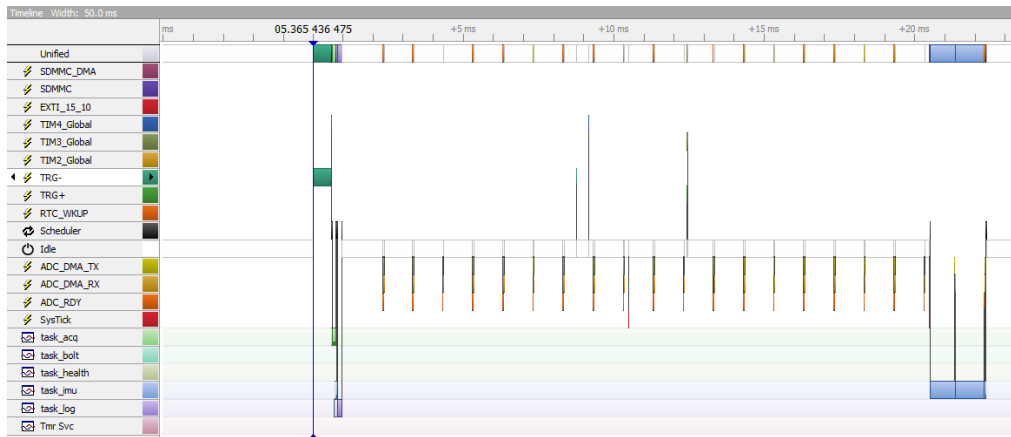


Figure 7.15: Beginning of a 3-second recording trace

Zooming to the reception of the last sample from the ADC reveals how the system stops the recording, as shown in Figure 7.16. After receiving the last sample from the ADC, the acquisition task unblocks and stops the ADC. In parallel, the IMU task finishes sampling the inertial measurement unit. A packet is generated containing the most important information about the recording and forwarded to the communication board. The acknowledgement from the COM board is handled by the `EXTI_15_10` interrupt service routine. A successful acknowledgement indicates that BOLT is ready to receive the packet from the application processor. Finally, the remaining data from the ADC and the IMU is logged to the SD card.

Periodic status packets are generated and forwarded to the network even during sleep mode, regardless of seismic activities. The RTC is configured to generate a wakeup interrupt after every two minutes by default. The node wakes up, performs the required measurements, generates and transmits a health packet to the network and goes back to sleep. The captured trace of this operation is shown in Figure 7.17.

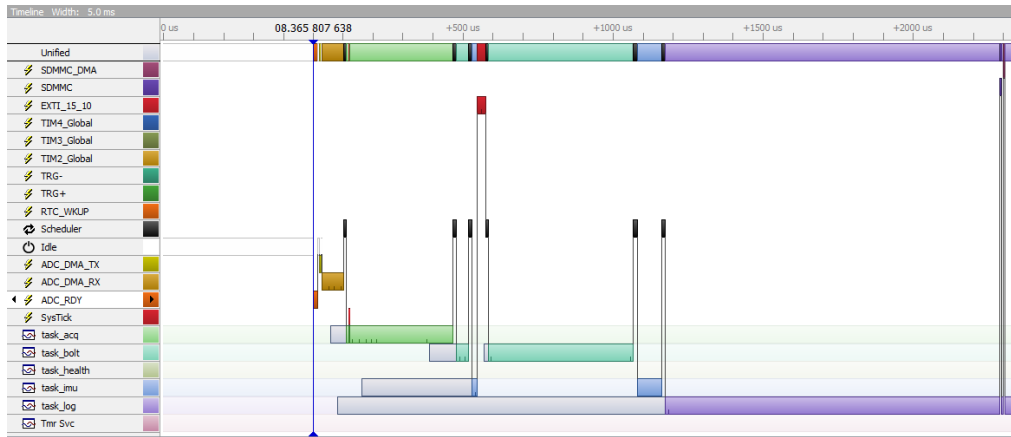


Figure 7.16: End of a 3-second recording trace

After the RTC wakeup interrupt, the system wakes up and reconfigures itself, then issues a direct-to-task notification to the health task. The health task generates an SD initialization command to the logging task and starts the measurements in parallel. After the microSD card is successfully initialized, the health task issues a command to retrieve the free space available on the SD card. After collecting all measurement and system status data, the health task generates a health packet which is forwarded to the communication board. Then the task issues a command to log the collected data to the SD card.

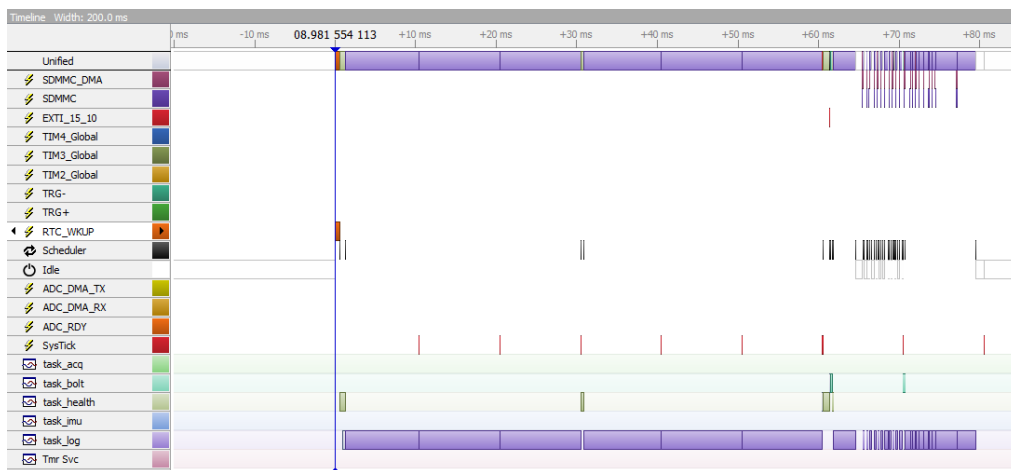


Figure 7.17: Trace of a periodic health wakeup

After completing all tasks, the system goes back to sleep. As the trace illustrates, the entire measurement procedure takes less than 80ms, including all SD

card operations which are by far the most process-demanding operations.

## 7.8 Field Test

As soon as the prototypes became operational and the software was capable of collecting and logging ADC data, an experimental field test was carried out next to a gravel pit located at a quarry site near Glattfelden, Switzerland. The exact location<sup>1</sup> is shown in Figure 7.18.



Figure 7.18: Field test location

### 7.8.1 Test Setup

At the time of the field test, the nodes were only capable of triggering to pre-configured thresholds, furthermore sampling the geophone sensor and logging

<sup>1</sup> Location coordinates: 47.558861, 8.479583

data. The RTCs of the nodes were operational, although without time synchronization support. The nodes did not support the DPP COM board at the time of the experiment.

All three prototypes were used in the experiment with the following configuration:

- One prototype with continuous sampling mode. After powering on the device, the node started sampling the geophone sensor and logging data continuously to the SD card regardless of triggering and detecting seismic activities.
- The two remaining devices were configured in triggered mode, with 3-second post-triggering, single-stage amplification and with  $1000mV$  negative and  $2000mV$  positive triggering threshold values. These values represent relatively low triggering sensitivities for both positive and negative sides.
- The ADC of each node was configured with disabled PGA and  $1ksp/s$  sampling frequency.

The nodes were laid out on the verge of a side road with approximately  $5^\circ$  grade, next to a gravel pit with  $60^\circ$  grade and with approximately 15 meters of slope length to the bottom. The nodes were placed in a line, with exactly 10 meters distance between them.

### 7.8.2 Test Scenarios

In order to provide accurate reference and synchronization between the nodes, all devices were placed next to each other and 11 strikes were performed with a hammer right next to the nodes with an approximate frequency of 1 Hz. Then various scenarios and experiments were carried out, involving the following actions:

- Hammering near the nodes from various distances.
- Shoveling gravel into the pit performed next to the nodes.
- Throwing rocks with approximately 10cm diameter into the pit, with aiming to hit the slope near the nodes.

### 7.8.3 Test Evaluation

After performing more than 10 various scenarios and experiments, the nodes were taken back and the collected data was extensively analyzed. The waveforms of the ADCs were visualized with MATLAB.

Figure 7.19 illustrates the visualization of a scenario when a rock was thrown next to the node from the distance of 2 meters.

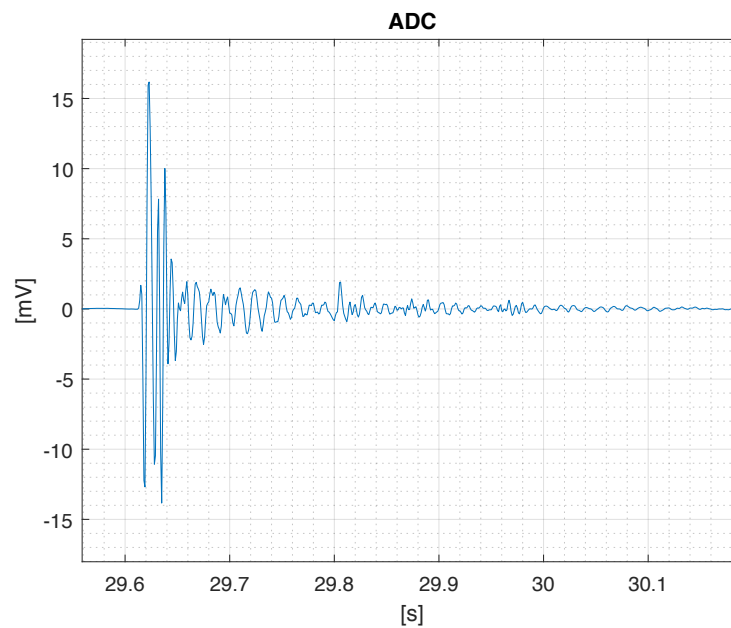


Figure 7.19: Data visualization of a rock thrown from 2 meters

Figure 7.20 shows the visualization of another scenario when intensive shoveling was performed 5 meters from the node for 30 seconds.

The visualization of other scenarios can be found in Appendix F. Furthermore, the detailed test protocol together with the data collected from all nodes during the experiment can be found in the `data/20180323/` folder within the project repository.

This field experiment confirmed that the prototypes were already capable of triggering to seismic activities and recording high-quality and low-noise geophone data while maintaining ultra-low energy consumption. The field test served as a great basis how to configure the nodes to detect different real-world activities.



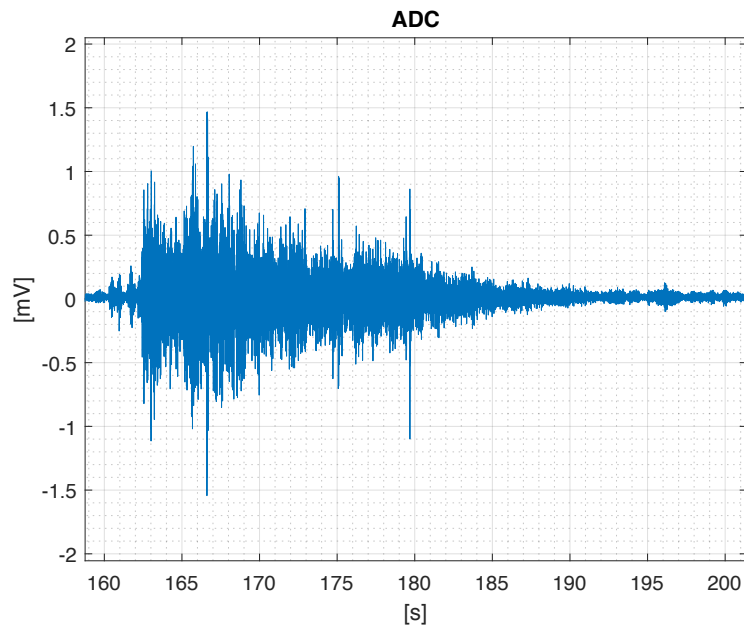


Figure 7.20: Data visualization of shoveling

## 7.9 Production

Due to the overall outstanding operation of the prototypes, including the well-designed hardware and robust software; furthermore due to the great success of various, extensive tests and experiments, including the field test described in Section 7.8, the geophone platform went into production. Minor changes were made to the printed circuit board and components used for prototyping and evaluation were removed. Currently 50 pieces are being manufactured and they are going to be deployed to the Swiss mountains later this year.

## 7.10 Geophone Platform Datasheet

The following tables summarize the parameters and characteristics of the geophone platform.

<b>Electrical characteristics</b>	
Multi-stage triggering with power management	Yes
Dual-side, independent triggering	Yes
Dynamic range for both sides	63.5dB
Smallest detectable input signal	50 $\mu$ V
ADC resolution	24 bits
Co-detection	Geophone + IMU
ADC max SNR	140dB
ADC integrated PGA	Yes
ADC PGA max. gain	128
Sleep current (single-stage)	29.0 $\mu$ A
Sleep current (dual-stage)	49.9 $\mu$ A
Average operating current during recording	20mA
Supported battery	1x D-cell
Operating temperature	-40 to +85°C

Table 7.5: Electrical characteristics

<b>Storage parameters</b>	
Supported memory card	microSD
File system support	FAT, FAT32, exFAT
LFN support	Yes
Folder and subfolder management	Yes
Volume size	Up to 2TB
Power management	Yes

Table 7.6: Storage parameters

<b>Sensor parameters</b>	
Geophone type	Omnidirectional
Geophone sensitivity	80/m/s
Temperature sensor	Yes
Humidity sensor	Yes
Battery voltage monitoring	Yes
Inertial measurement unit	Yes
Accelerometer	3-axis
Magnetometer	3-axis
Independent power management for all sensors	Yes

Table 7.7: Sensor parameters

<b>Configuration parameters</b>	
Custom node ID	Yes
Operating modes	Event-triggered, continuous
Triggering amplification	Single stage, dual stage
Triggering thresholds	0 to 3000mV, with 10mV steps
Post-triggering duration	1s to 9hrs, with 1s steps
Trigger timeout duration	1s to 9hrs, with 1s steps
ADC PGA	0 to 128
ADC format	Two's complement, offset-binary
ADC sampling rate	50sps to 1000sps
IMU sampling frequency	0 to 10sps (can be entirely disabled)
Remote on-the-fly configuration	Yes

Table 7.8: Configuration parameters

<b>Network parameters</b>	
Individual node reconfiguration	Yes
Mass reconfiguration with broadcast	Yes
Periodic node health packets	Yes
Event and notification messages	Yes
Acquisition information packets	Yes, one per event
IMU packets	Yes, multiple per event
Time synchronization	Sub-millisecond precision

Table 7.9: Network parameters

<b>Connectivity</b>	
Wireless	Sub-GHz radio module
USB	Mini-USB
Supported USB device classes	CDC, MSC, DFU
Debug	SWD + SWO

Table 7.10: Connectivity

<b>Mechanical parameters</b>	
Standard aluminum enclosure	Yes
IP rating	IP67
Mounting	Custom metal plate
Geophone clamp	Custom 3D printed
Antenna feedthrough	Yes, with ESD protection

Table 7.11: Mechanical parameters

# Conclusion

---

The thesis presents the development and implementation of a state-of-the-art, event-based geophone platform with co-detection. The work covers the complete product development, from the the beginning of setting up the system requirements all the way to the final production-state.

After introducing the general research area in Chapter 1 alongside with the related work, Chapter 2 presents the project background in detail. Chapter 3 contains the description of the system specifications based on a pilot study. Chapter 4 describes a state-of-the-art, efficient, ultra-low power consumption, high dynamic range, on-the-fly configurable triggering concept. The proposed triggering circuit is implemented and evaluated thoroughly. The hardware design of the geophone platform is described in Chapter 5. The developed embedded software based on a self-modified RTOS kernel is described in Chapter 6. Finally, a detailed evaluation of the geophone platform is presented in Chapter 7, including extensive power analysis, complete hardware evaluation, temperature test, load test, software performance analysis and a field-test.

## 8.1 Media Coverage

In May 2018, the geophone platform was featured on SRF Einstein [50]. The part of the episode, titled “Forschung extrem: Datensammeln im Permafrost” introduces the geophone platform together with the wireless sensor network deployed and operated in the Swiss mountains.

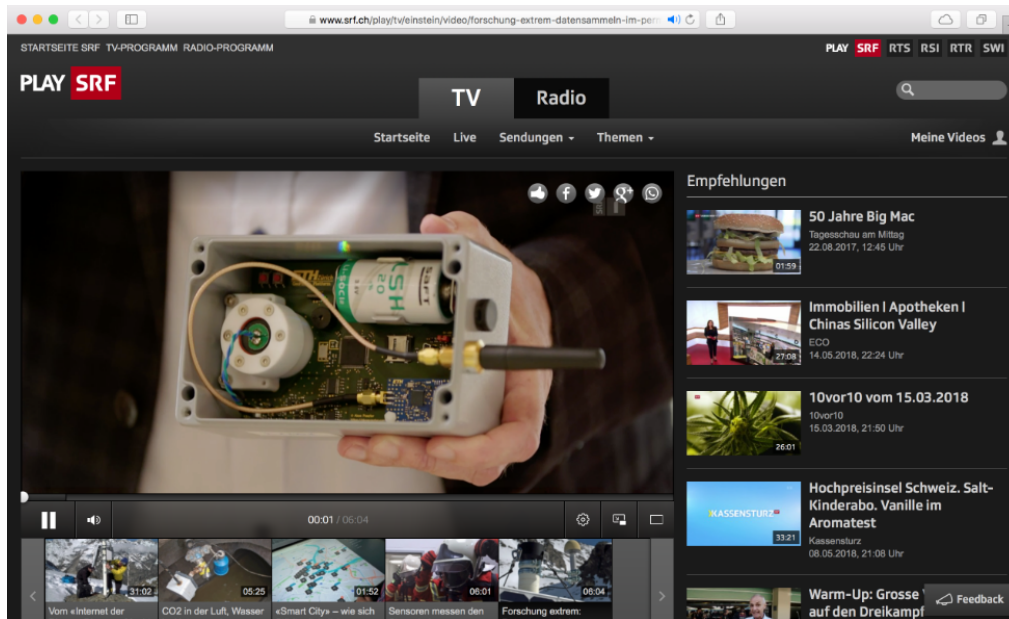


Figure 8.1: Video featuring the geophone platform on SRF

## 8.2 Outlook

Various experiments, tests and complete platform evaluation produced outstanding and greatly promising results. The designed hardware including the state-of-the-art advanced triggering system performs remarkably well in all tested scenarios. The developed embedded software runs immensely great and it is proven to be particularly robust even during unrealistically busy and demanding scenarios. Furthermore, the geophone platform perfectly integrates into the current PermaSense wireless sensor network. Conclusively, the platform went into mass-production. At the time of writing, 50 pieces are being manufactured and the devices are going to be deployed to the Swiss mountains later this year.

# Bibliography

- [1] PermaSense Consortium. <http://www.permasense.ch>.
- [2] F. Sutton, R. D. Forno, D. Gschwend, T. Gsell, R. Lim, J. Beutel, and L. Thiele, “The design of a responsive and energy-efficient event-triggered wireless sensing system,” in *Proceedings of the 14th International Conference on Embedded Wireless Systems and Networks (EWSN 2017)*, pp. 144–155, ACM, Feb 2017.
- [3] F. Sutton, M. Zimmerling, R. D. Forno, R. Lim, T. Gsell, G. Giannopoulou, F. Ferrari, J. Beutel, and L. Thiele, “Bolt: A stateful processor interconnect,” Apr 2016.
- [4] Texas Instruments, “MSP430FR59xx Microcontrollers Datasheet.” <http://www.ti.com/lit/ds/symlink/msp430fr5969.pdf>.
- [5] F. Ferrari, M. Zimmerling, L. Mottola, and L. Thiele, “Low-power wireless bus,” in *Proceedings of the 10th ACM Conference on Embedded Network Sensor Systems, SenSys '12*, pp. 1–14, ACM, 2012.
- [6] F. Ferrari, M. Zimmerling, L. Thiele, and O. Saukh, “Efficient network flooding and time synchronization with glossy,” in *Proceedings of the 10th ACM/IEEE International Conference on Information Processing in Sensor Networks*, pp. 73–84, April 2011.
- [7] Texas Instruments, “CC430F514x SoC With RF Core Datasheet.” <http://www.ti.com/lit/ds/symlink/cc430f5147.pdf>.
- [8] G. Asch, *New Manual of Seismological Observatory Practice (NMSOP)*, ch. Seismic Recording Systems, pp. 1–20. Deutsches GeoForschungsZentrum GFZ, 2009.
- [9] C. Collette, P. Carmona-Fernandez, S. Janssens, K. Artoos, M. Guinchard, and C. Hauviller, “Review of sensors for low frequency seismic vibration measurement,” Tech. Rep. ATN/Note/2011/001 (TECH), CERN, 2011.
- [10] ION, “SM-6 Geophone Datasheet.” [https://www.iongeo.com/content/documents/Resource%20Center/Brochures%20and%20Data%20Sheets/Data%20Sheets/Sensor%20Geophones/DS\\_SEN\\_121026SM6.pdf](https://www.iongeo.com/content/documents/Resource%20Center/Brochures%20and%20Data%20Sheets/Data%20Sheets/Sensor%20Geophones/DS_SEN_121026SM6.pdf).
- [11] Wikipedia contributors, “Fourier transform.” [https://en.wikipedia.org/wiki/Fourier\\_transform](https://en.wikipedia.org/wiki/Fourier_transform).

- 
- [12] P. Blanchard and B. Pelletier, “Using ESD Diodes as Voltage Clamps,” *Analog Dialogue*, vol. 49, no. 4, 2015.
- [13] STMicroelectronics, “STM32L476VG Datasheet.”  
<http://www.st.com/resource/en/datasheet/stm32l476vg.pdf>.
- [14] Analog Devices, “ADA4051-1 Datasheet.”  
[http://www.analog.com/media/en/technical-documentation/data-sheets/ADA4051-1\\_4051-2.pdf](http://www.analog.com/media/en/technical-documentation/data-sheets/ADA4051-1_4051-2.pdf).
- [15] Maxim Integrated, “MAX5530 Datasheet.”  
<https://datasheets.maximintegrated.com/en/ds/MAX5530-MAX5531.pdf>.
- [16] Maxim Integrated, “MAX919 Datasheet.”  
<https://datasheets.maximintegrated.com/en/ds/MAX917-MAX920.pdf>.
- [17] Texas Instruments, “LP5907 Datasheet.”  
<http://www.ti.com/lit/ds/symlink/lp5907.pdf>.
- [18] Maxim Integrated, “MAX5532 Datasheet.”  
<https://datasheets.maximintegrated.com/en/ds/MAX5532-MAX5535.pdf>.
- [19] Maxim Integrated, “MAX9019 Datasheet.”  
<https://datasheets.maximintegrated.com/en/ds/MAX9015-MAX9020.pdf>.
- [20] Wikipedia contributors, “Capacitive coupling.”  
[https://en.wikipedia.org/wiki/Capacitive\\_coupling](https://en.wikipedia.org/wiki/Capacitive_coupling).
- [21] Wikipedia contributors, “High-pass filters.”  
[https://en.wikipedia.org/wiki/High-pass\\_filter](https://en.wikipedia.org/wiki/High-pass_filter).
- [22] STMicroelectronics, “LSM303C Datasheet.”  
<http://www.st.com/resource/en/datasheet/lsm303c.pdf>.
- [23] STMicroelectronics, “Getting started with STM32L4 Series and STM32L4+ Series hardware development,” Tech. Rep. AN4555, 2018.
- [24] Wikipedia contributors, “STMicroelectronics.”  
<https://en.wikipedia.org/wiki/STMicroelectronics>.
- [25] STMicroelectronics, “STM32L496VG Datasheet.”  
<http://www.st.com/resource/en/datasheet/stm32l496vg.pdf>.
- [26] STMicroelectronics, “Oscillator design guide for STM8AF/AL/S and STM32 microcontrollers,” Tech. Rep. AN2867, 2017.



- [27] *ARM Debug Interface Architecture Specification*. ADIv5.0 to ADIv5.2, ARM Ltd., 2013.
- [28] SEGGER Microcontroller GmbH, “J-Link / J-Trace User Guide,” Tech. Rep. UM08001, 2018.
- [29] *ARM CoreSight Architecture Specification*. v2.0, ARM Ltd., 2013.
- [30] SEGGER, “Real Time Transfer.”  
<https://www.segger.com/products/debug-probes/j-link/technology/about-real-time-transfer/>.
- [31] Panasonic, “RP-SMSCxxDA1 Datasheet.”  
<https://na.industrial.panasonic.com/sites/default/pidsa/files/downloads/files/panasonic-sc-series-data-sheet.pdf>.
- [32] ION, “SM-6 Omnidirectional Geophone Datasheet.” [https://www.iongeo.com/content/documents/Resource%20Center/Brochures%20and%20Data%20Sheets/Data%20Sheets/Sensor%20Geophones/DS\\_SEN\\_SM6.pdf](https://www.iongeo.com/content/documents/Resource%20Center/Brochures%20and%20Data%20Sheets/Data%20Sheets/Sensor%20Geophones/DS_SEN_SM6.pdf).
- [33] Maxim Integrated, “MAX11214 Datasheet.”  
<https://datasheets.maximintegrated.com/en/ds/MAX11214.pdf>.
- [34] Maxim Integrated, “MAX11214 Evaluation Kit Datasheet.”  
<https://datasheets.maximintegrated.com/en/ds/MAX11214EVK-MAX11216EVKIT.pdf>.
- [35] Sensirion, “SHT3x-DIS Datasheet.”  
[https://www.sensirion.com/fileadmin/user\\_upload/customers/sensirion/Dokumente/0\\_Datasheets/Humidity/Sensirion\\_Humidity\\_Sensors\\_SHT3x\\_Datasheet\\_digital.pdf](https://www.sensirion.com/fileadmin/user_upload/customers/sensirion/Dokumente/0_Datasheets/Humidity/Sensirion_Humidity_Sensors_SHT3x_Datasheet_digital.pdf).
- [36] Networked Embedded Systems, ETH Zürich.  
<https://www.tec.ee.ethz.ch/research/networked-embedded-systems.html>.
- [37] USB Implementers Forum, “USB Device Class Specifications.”  
[http://www.usb.org/developers/docs/devclass\\_docs/](http://www.usb.org/developers/docs/devclass_docs/).
- [38] Saft, “LSH Lithium Batteries.”  
<https://www.saftbatteries.com/products-solutions/products/ls-lsh/>.
- [39] Wikipedia contributors, “Linear regulator.”  
[https://en.wikipedia.org/wiki/Linear\\_regulator](https://en.wikipedia.org/wiki/Linear_regulator).
- [40] Wikipedia contributors, “Buck converter.”  
[https://en.wikipedia.org/wiki/Buck\\_converter](https://en.wikipedia.org/wiki/Buck_converter).

- 
- [41] A. Pasztor, *Customizable Bootloader for STM32 microcontrollers*. GitHub, <https://github.com/akospasztor/stm32-bootloader>, 2018.
- [42] STMicroelectronics, “Description of STM32L4/L4+ HAL and low-layer drivers,” Tech. Rep. UM1884, 2018.
- [43] R. Barry, *Mastering the FreeRTOS Real Time Kernel*. Real Time Engineers Ltd., 2016.
- [44] ChaN, “FatFs Module Application Note,” tech. rep., <http://elm-chan.org/fsw/ff/doc/appnote.html>, 2018.
- [45] STMicroelectronics, “STM32Cube USB device library,” Tech. Rep. UM1734, 2015.
- [46] R. Barry, *The FreeRTOS Reference Manual*. Real Time Engineers Ltd., 2016.
- [47] A. Pasztor, “Towards Low-Power, Timing-Predictable Medical Monitoring.” Semester Thesis, 2016.
- [48] STMicroelectronics, “Optimizing power and performances with STM32L4 series microcontrollers,” Tech. Rep. AN4746, 2015.
- [49] Wikipedia contributors, “Frequency response.” [https://en.wikipedia.org/wiki/Frequency\\_response](https://en.wikipedia.org/wiki/Frequency_response).
- [50] SRF Einstein, “Forschung extrem: Datensammeln im Permafrost.” <https://www.srf.ch/play/tv/einstein/video/forschung-extrem-datensammeln-im-permafrost?id=7cf5e8b6-f872-4c35-9e74-6b46e3160ccf>, 2018.

# Repository Organization

---

The project repository contains 12 folders. The `code` directory contains the geophone software source code. The `data` and `evaluation` folders contain measurement, field test and evaluation data. The used literature, including datasheets, articles and technical notes can be found in the `literature` directory. The `mechanics` folder contains the mechanical design files of the geophone platform. The hardware design files are stored in the `pcb` directory. The thesis presentation and report files can be found in `presentation` and `report` folders respectively.

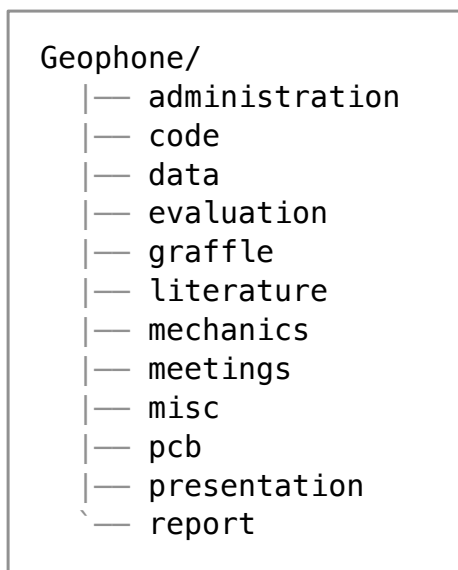


Figure A.1: Repository organization

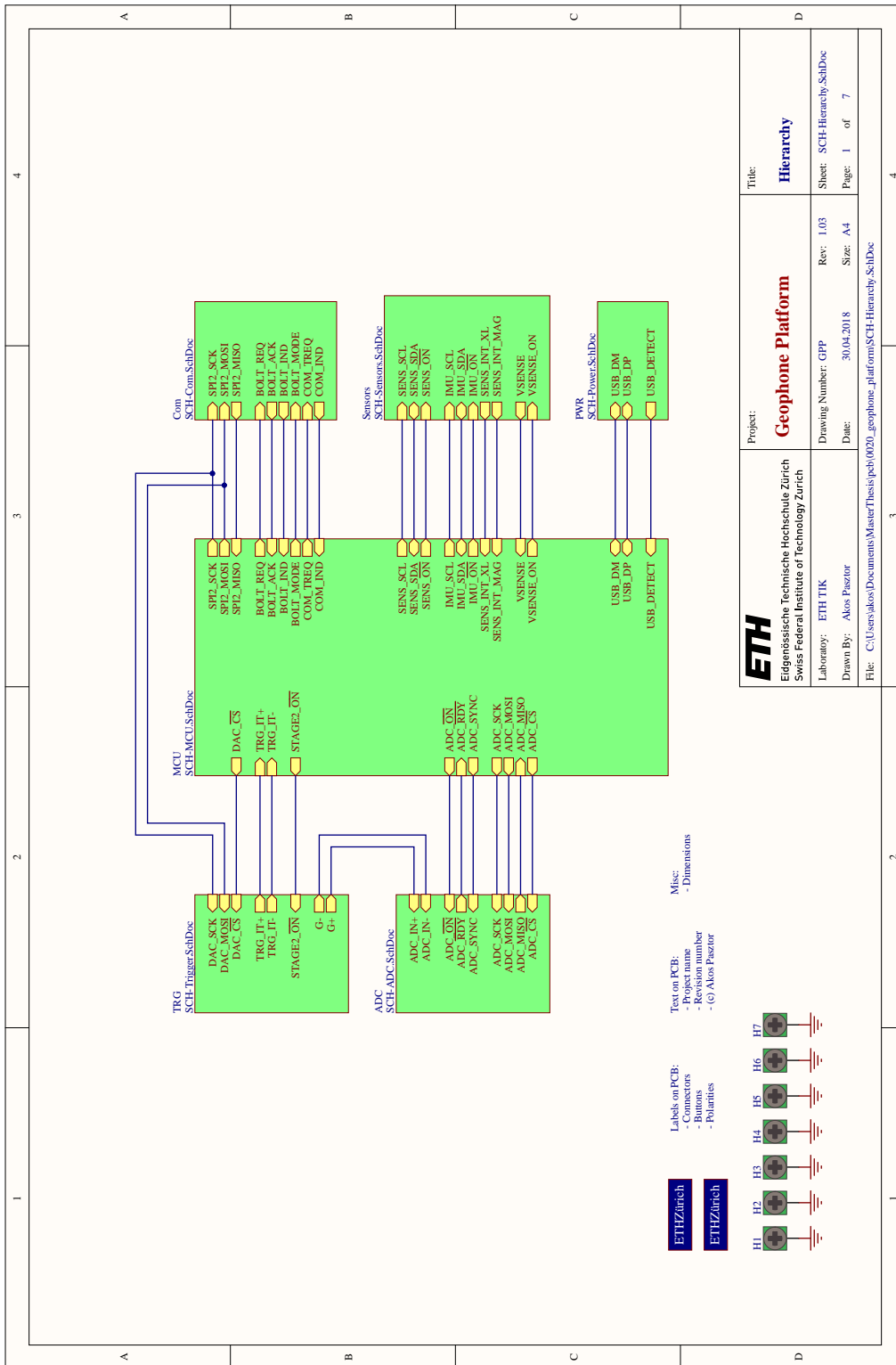
*This page intentionally left blank.*

## APPENDIX B

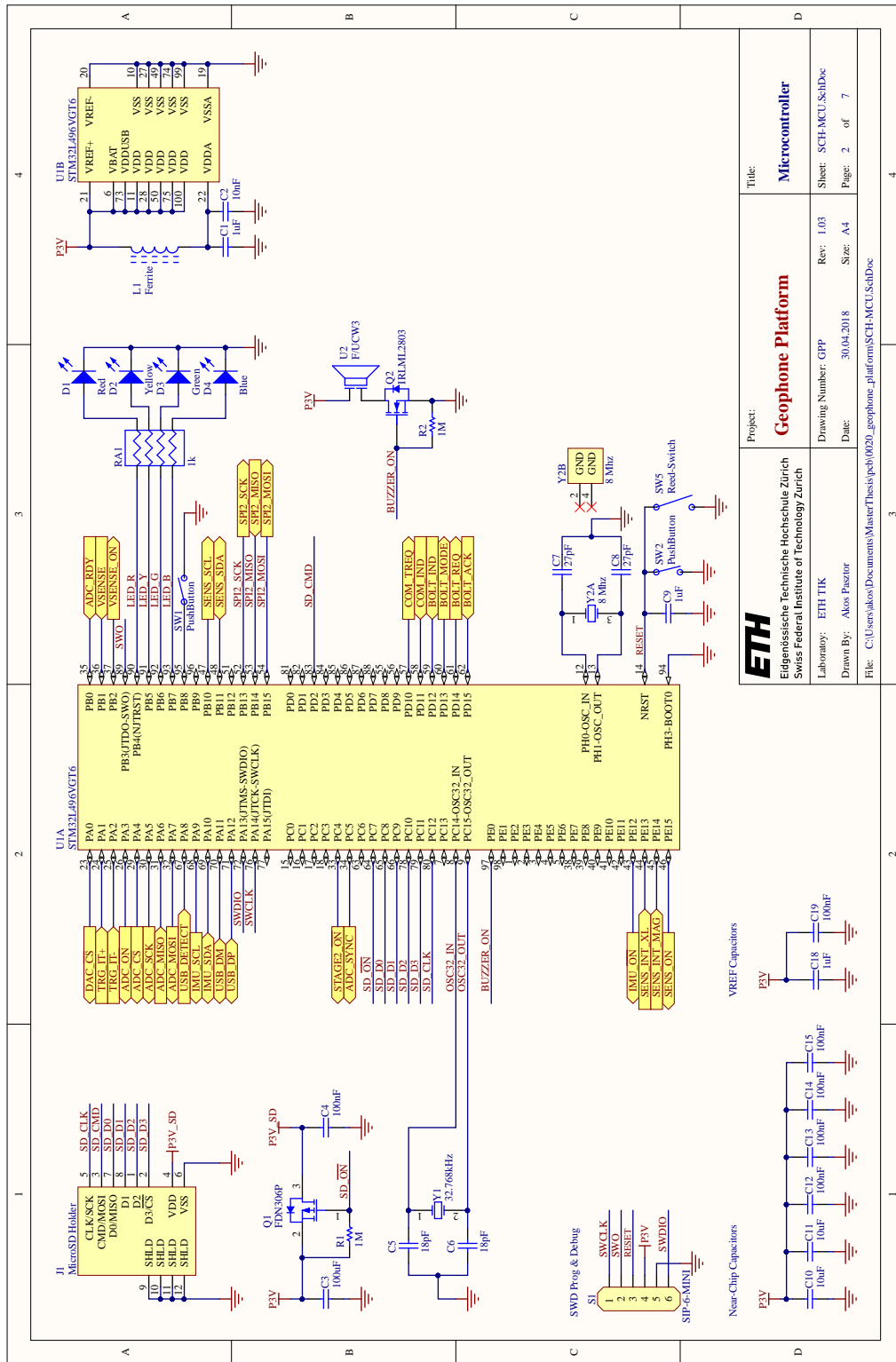
# Geophone Schematics

---

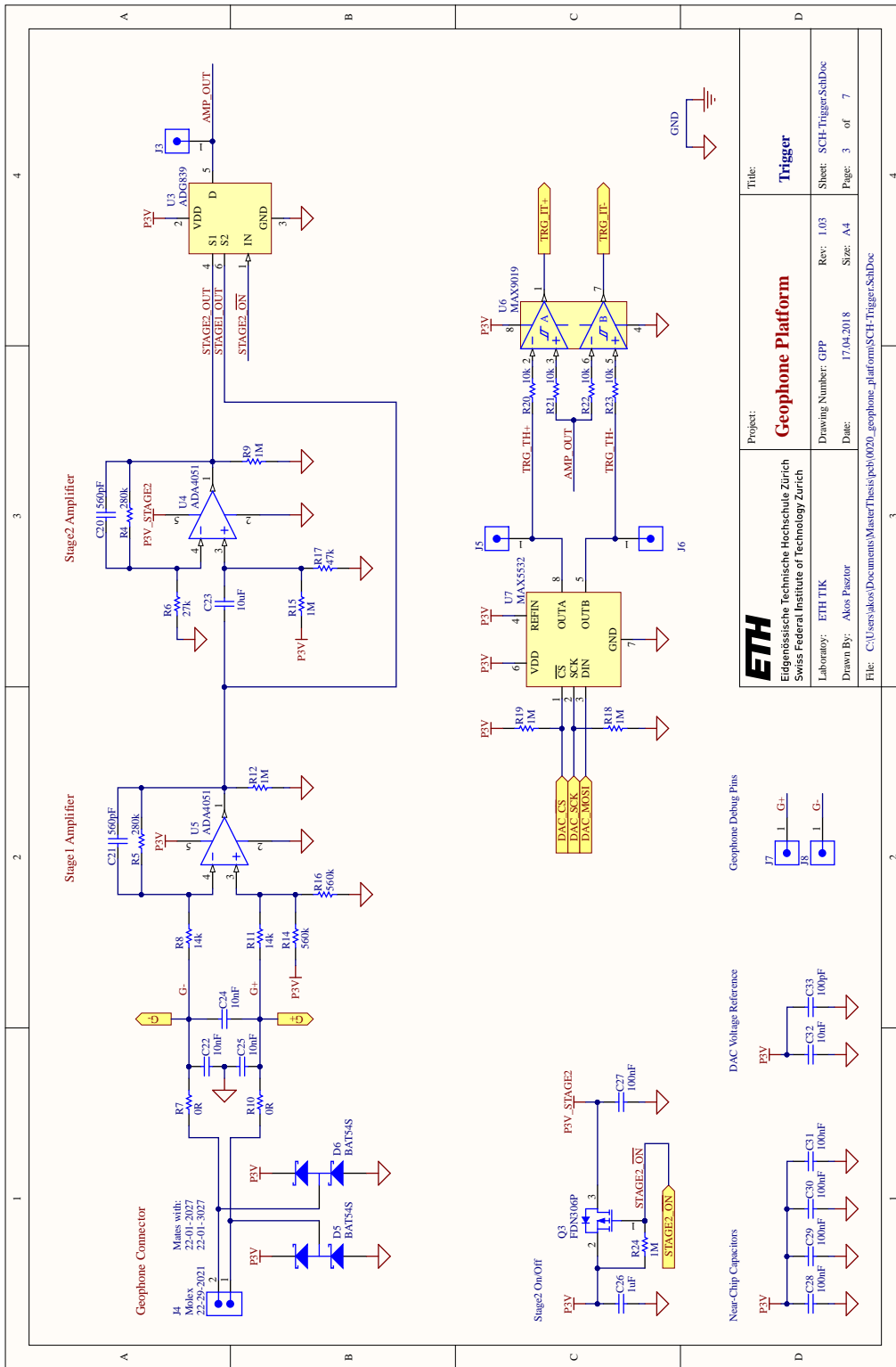
The circuit schematics of the geophone platform is included in the following pages.



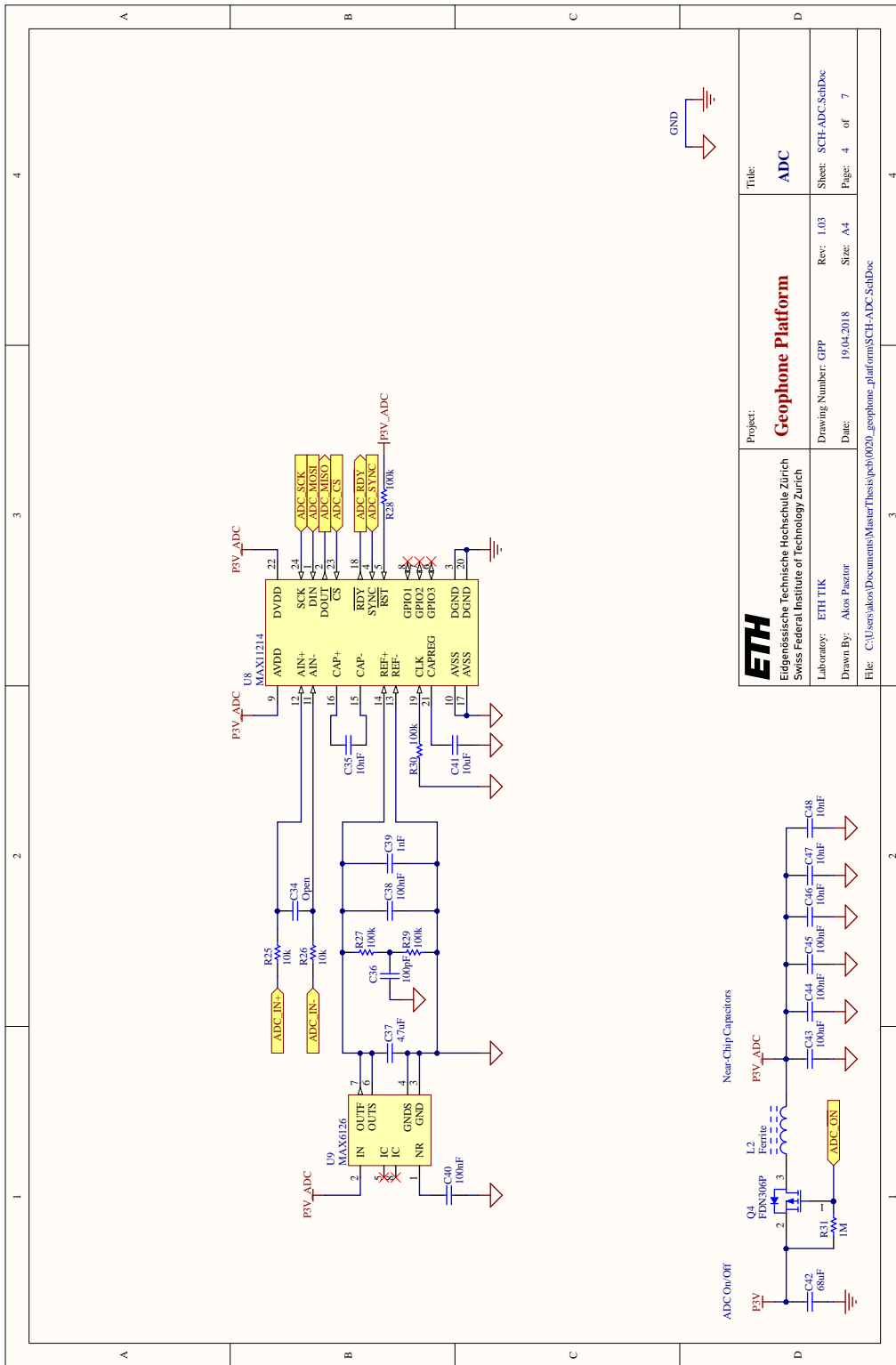
Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zürich		Project: <b>Geophone Platform</b> Title: <b>Hierarchy</b>	
Laboratory: ETH TIK	Drawing Number: GPP	Rev: 1.03	Sheet: SCH-Hierarchy.SchDoc
Drawn By: Akos Pasztor	Date: 30.04.2018	Size: A4	Page: 1 of 7
File: C:\Users\akos\Documents\MasterThesis\pcb\0020_geophone_platform\SCH-Hierarchy.SchDoc			




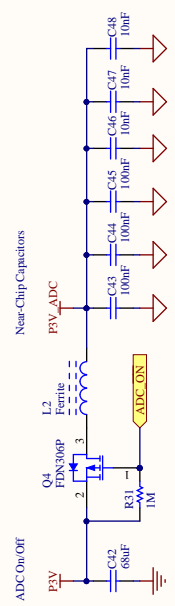
<b>ETH</b> Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zürich		<b>Geophone Platform</b>		<b>Microcontroller</b>	
Project: ETH TIK Laboratory: Alos Pasztor		Drawing Number: GPP Date: 30.04.2018		Rev: 1.03 Size: A4	
File: C:\Users\alos\Documents\MasterThesis\peb\0020_geophone_platform\SCH-MCU.SchDoc		Sheet: SCH-MCU.SchDoc		Page: 2 of 7	

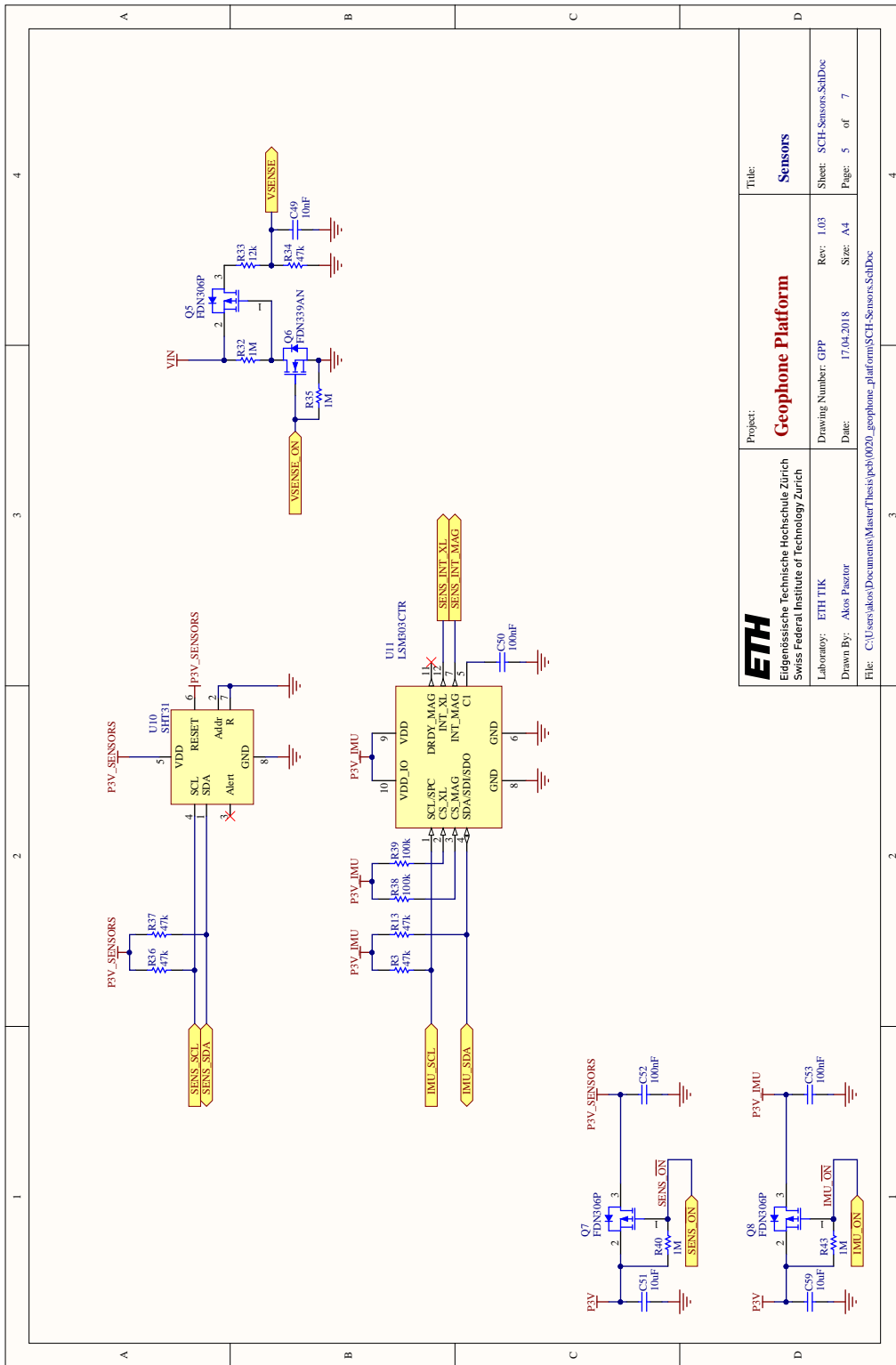





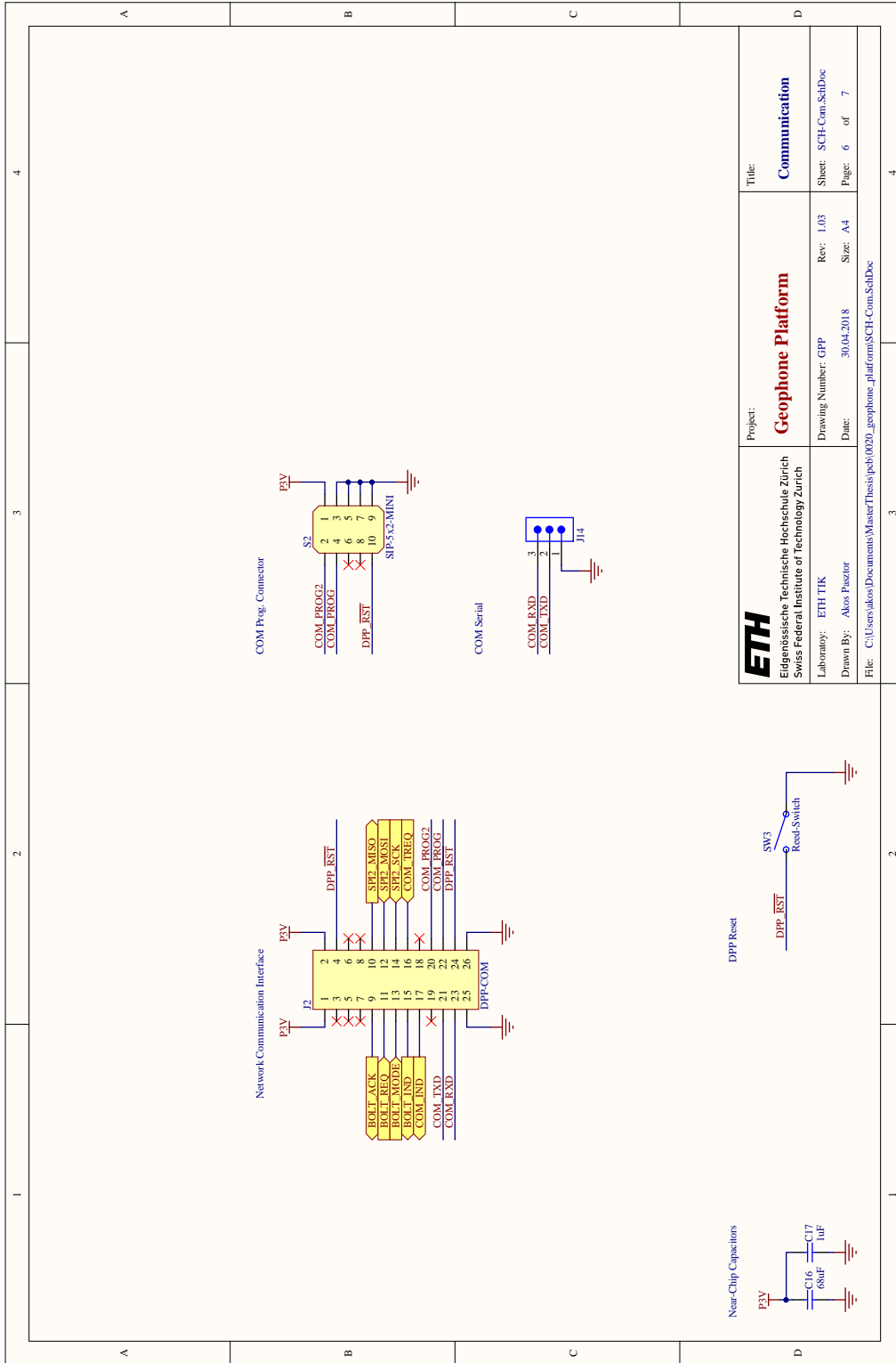


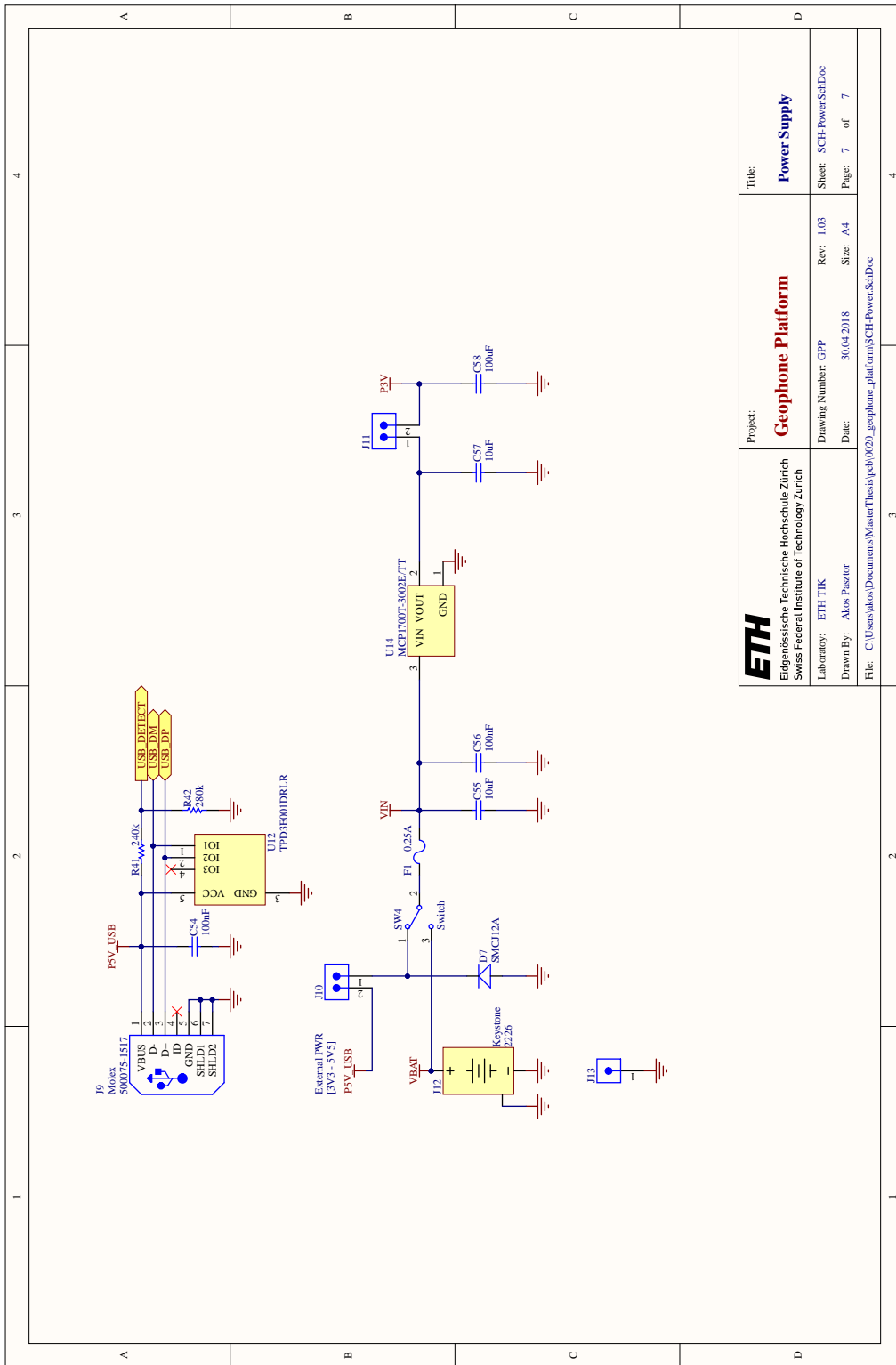
 Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zürich		Project: <b>Geophone Platform</b>	Title: <b>ADC</b>
Laboratory: ETH TIK Drawn By: Alos Pasztor	Drawing Number: GPP Date: 19/04/2018	Rev.: 1.03 Size: A4	Sheet: SCH-ADC-SchDoc Page: 4 of 7
File: C:\Users\alos\Documents\MasterThesis\peb\0020_geophone_platform\SCH-ADC-SchDoc			






 Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zürich	Project: <b>Geophone Platform</b>	Title: <b>Sensors</b>
	Drawing Number: GPP	Rev: 1.03
	Date: 17.04.2018	Size: A4
Laboratory: ETH TIK	Drawn By: Akos Pasztor	Sheet: SCH-Sensors_SchDoc Page: 5 of 7
File: C:\Users\akos\Documents\MasterThesis\peb\0020_geophone_platform\SCH-Sensors_SchDoc		





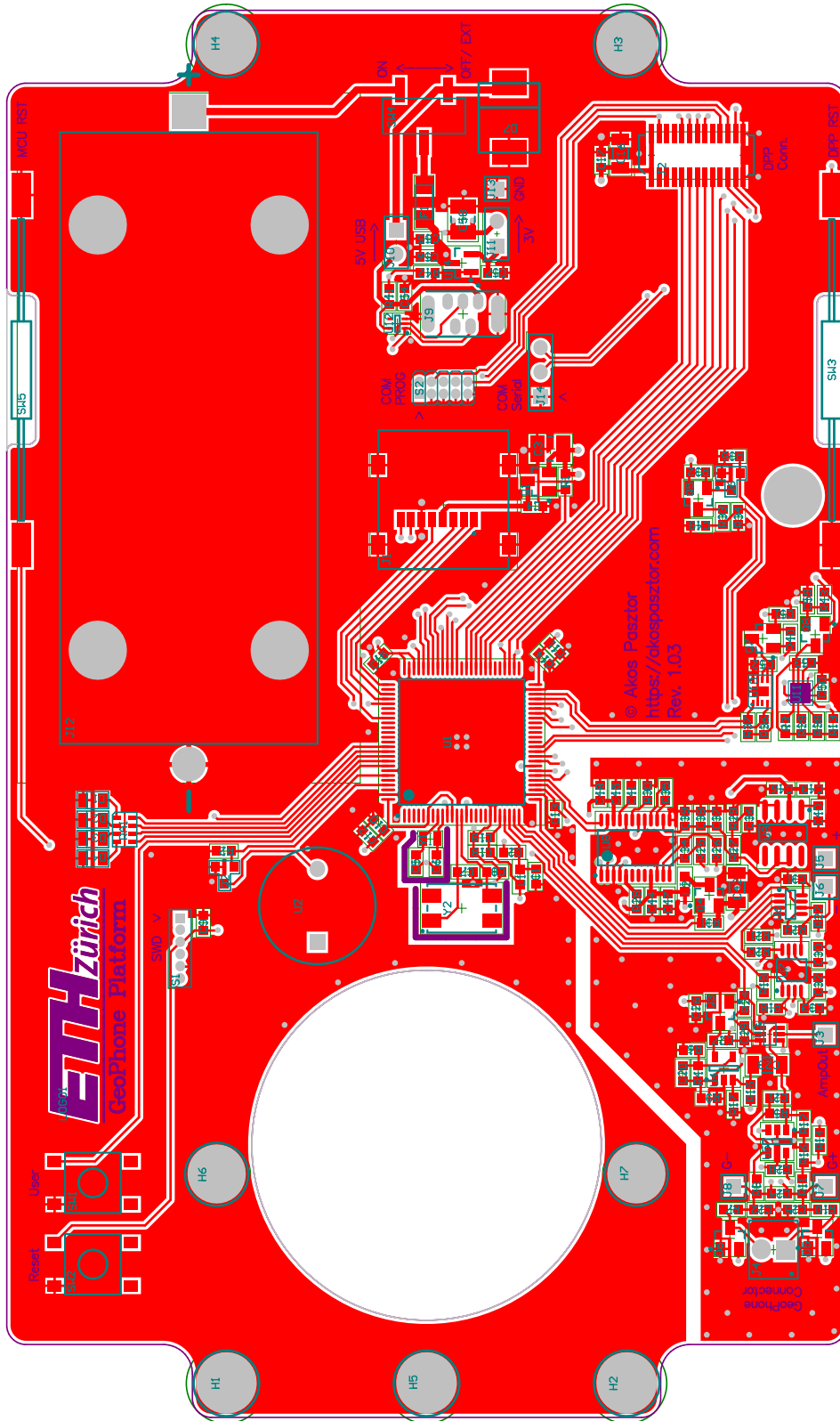
 Eidgenössische Technische Hochschule Zürich Swiss Federal Institute of Technology Zürich	Project:	<b>Geophone Platform</b>	Title:	<b>Power Supply</b>
	Laboratory: ETH TIK	Drawing Number: GPP	Rev: 1.03	Sheet: SCH-PowerSchDoc
Drawn By: Akos Pasztor	Date: 30.04.2018	Size: A4	Page: 7	of 7
File: C:\Users\akos\Documents\MasterThesis\peb\0020_geophone_platform\SCH-PowerSchDoc				

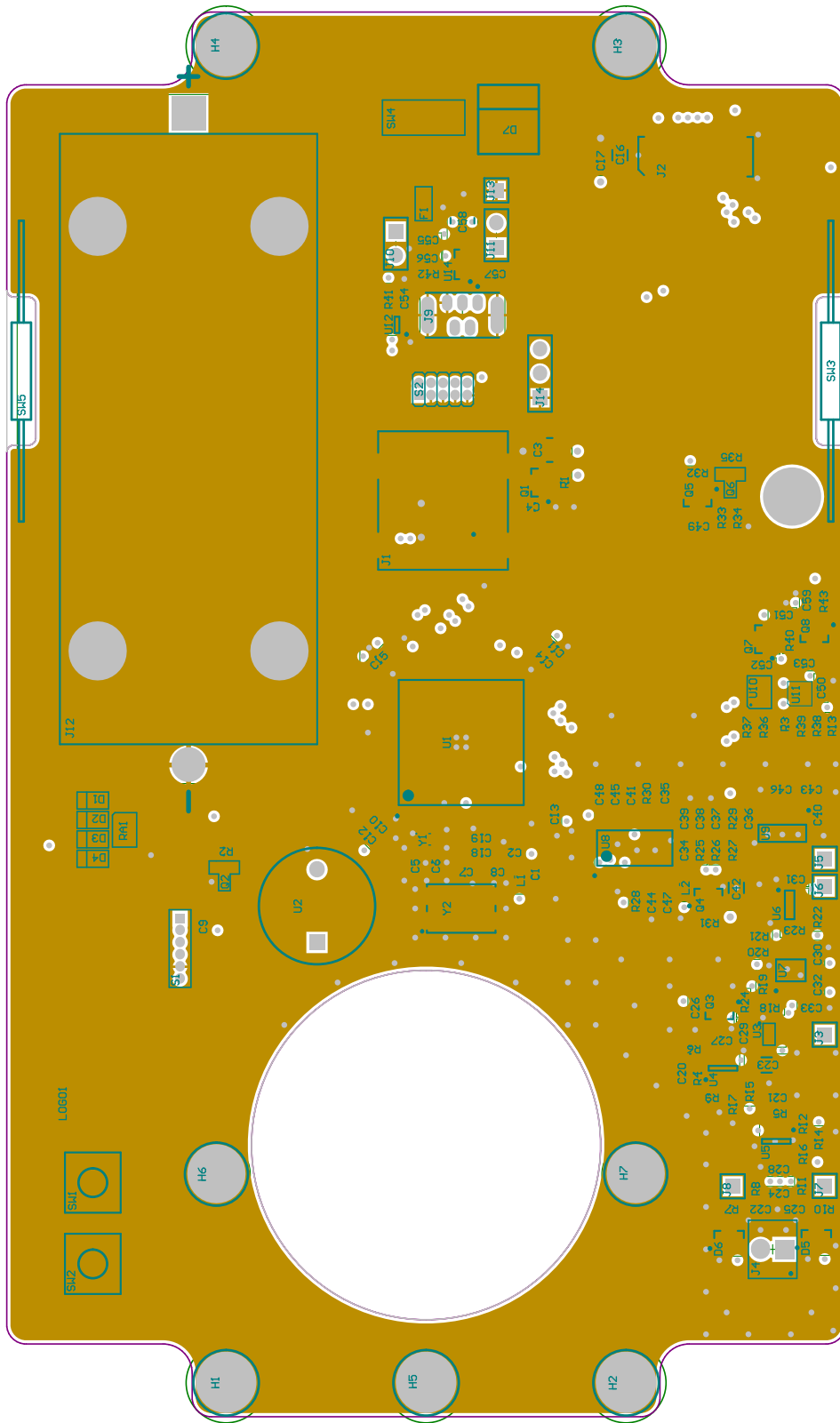
## APPENDIX C

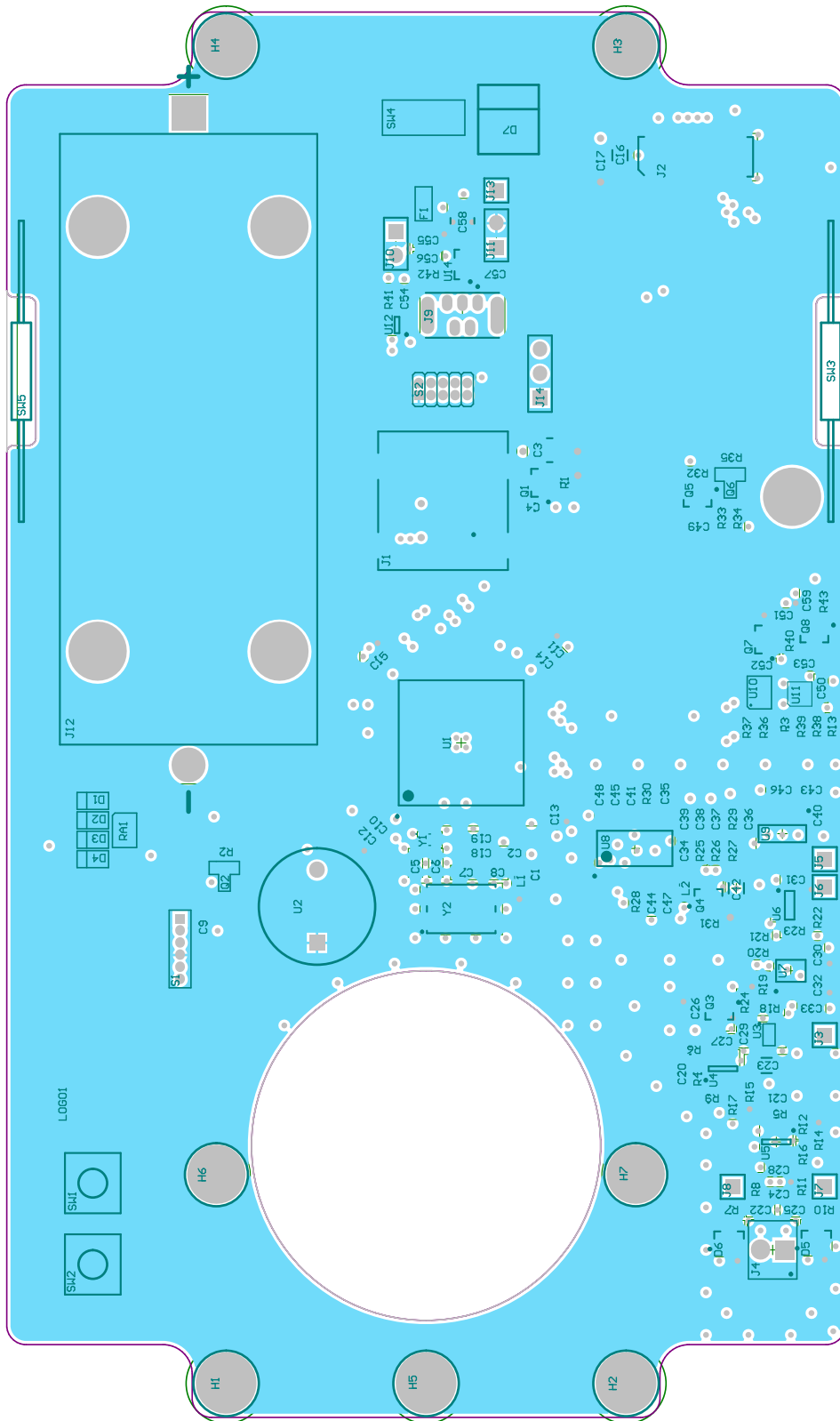
# Geophone PCB

---

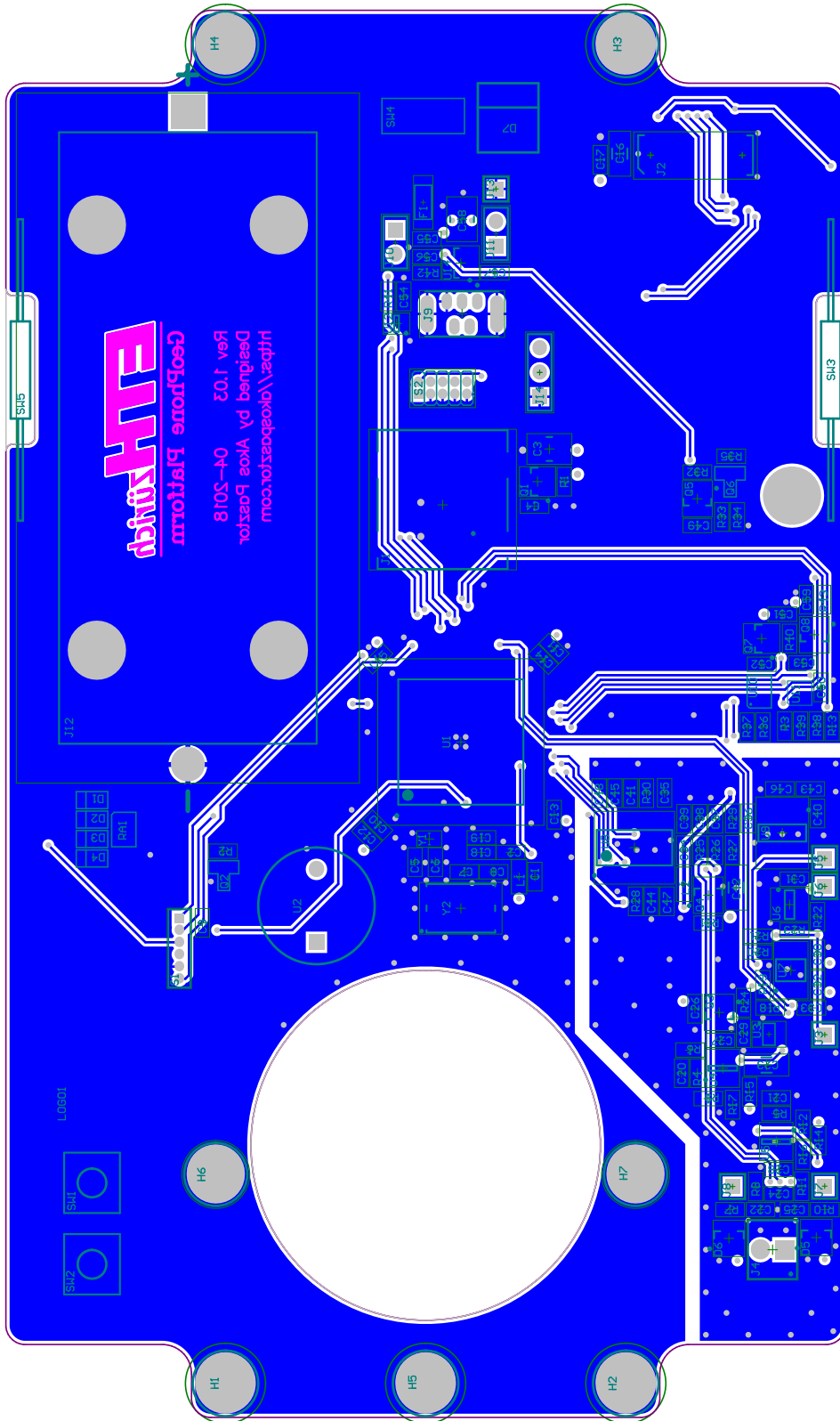
The design of the printed circuit board of the geophone platform is included in the following pages.

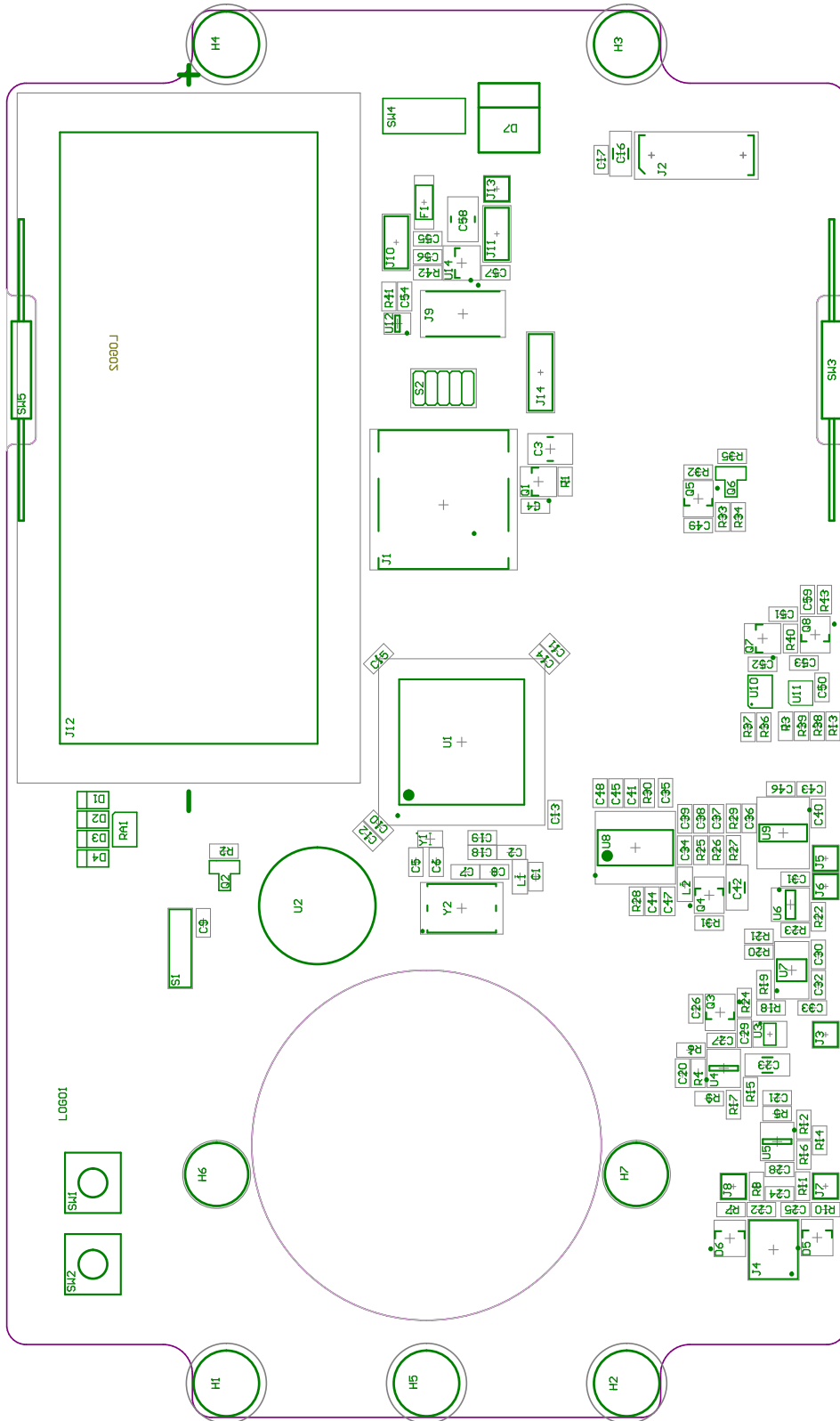


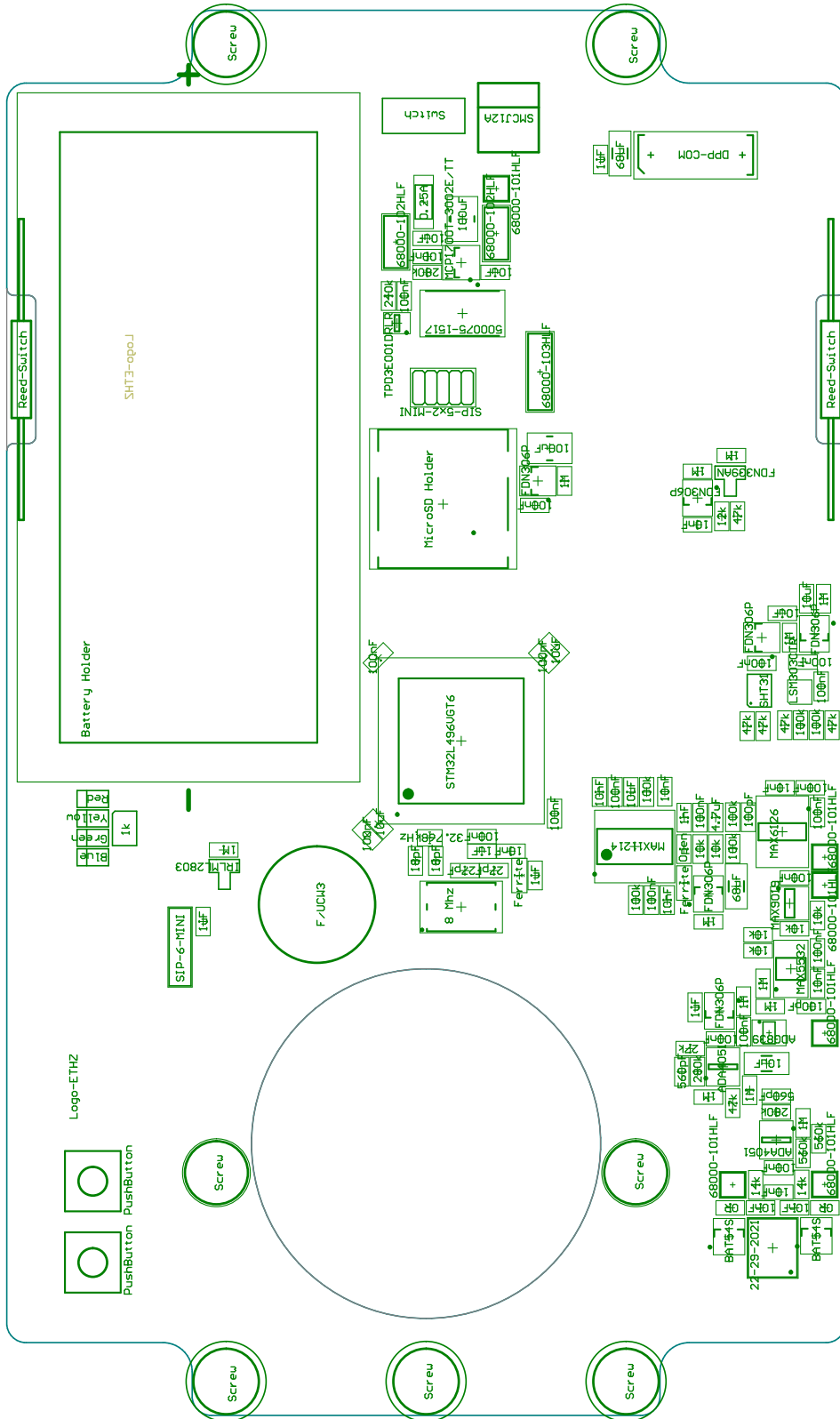


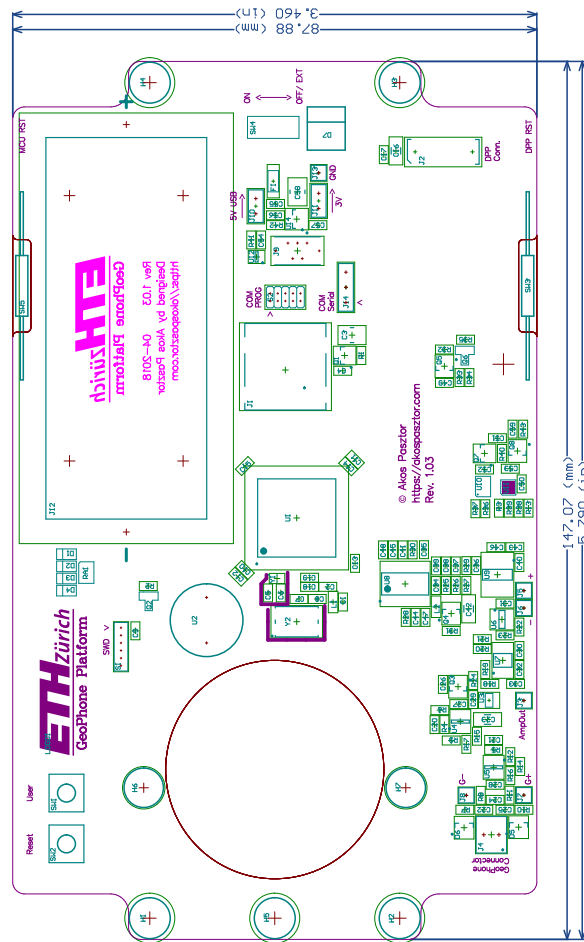


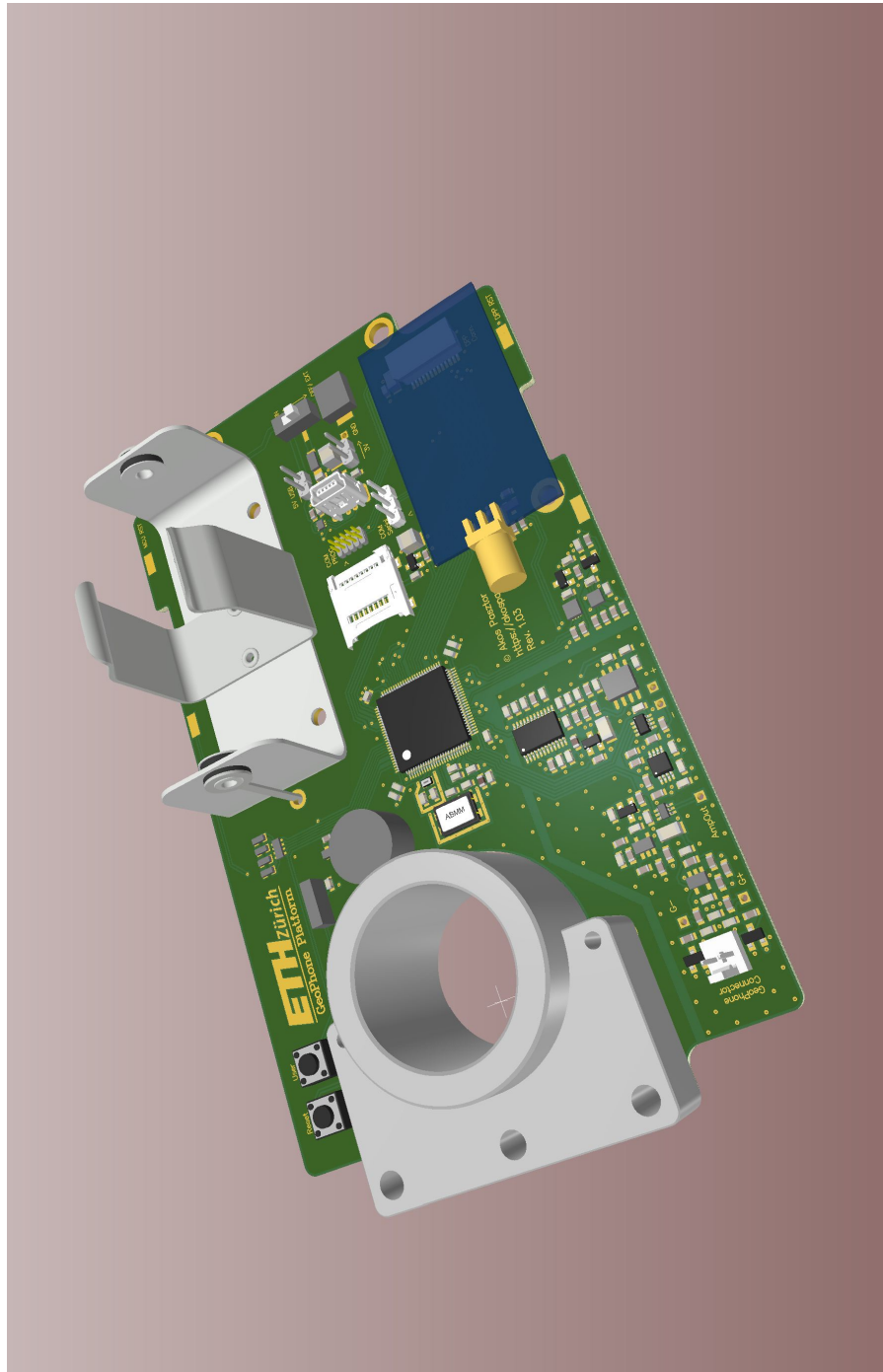












*This page intentionally left blank.*

## APPENDIX D

# Additional Power Traces

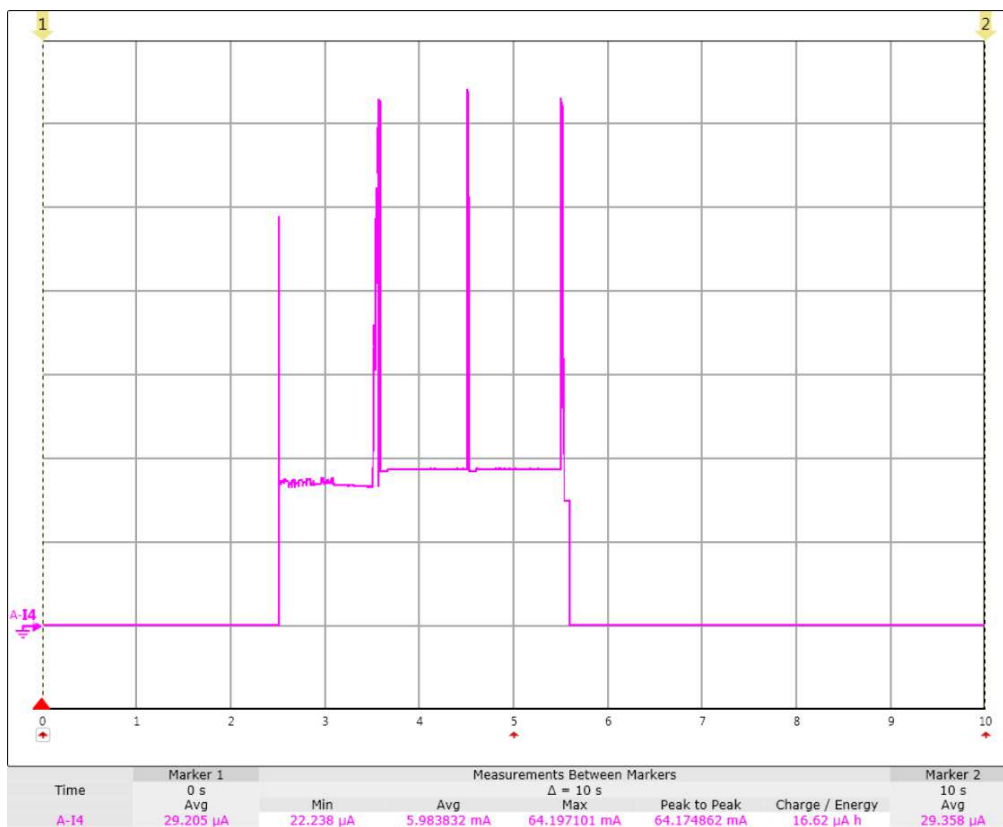


Figure D.1: Triggering and 3s recording without COM board, without IMU

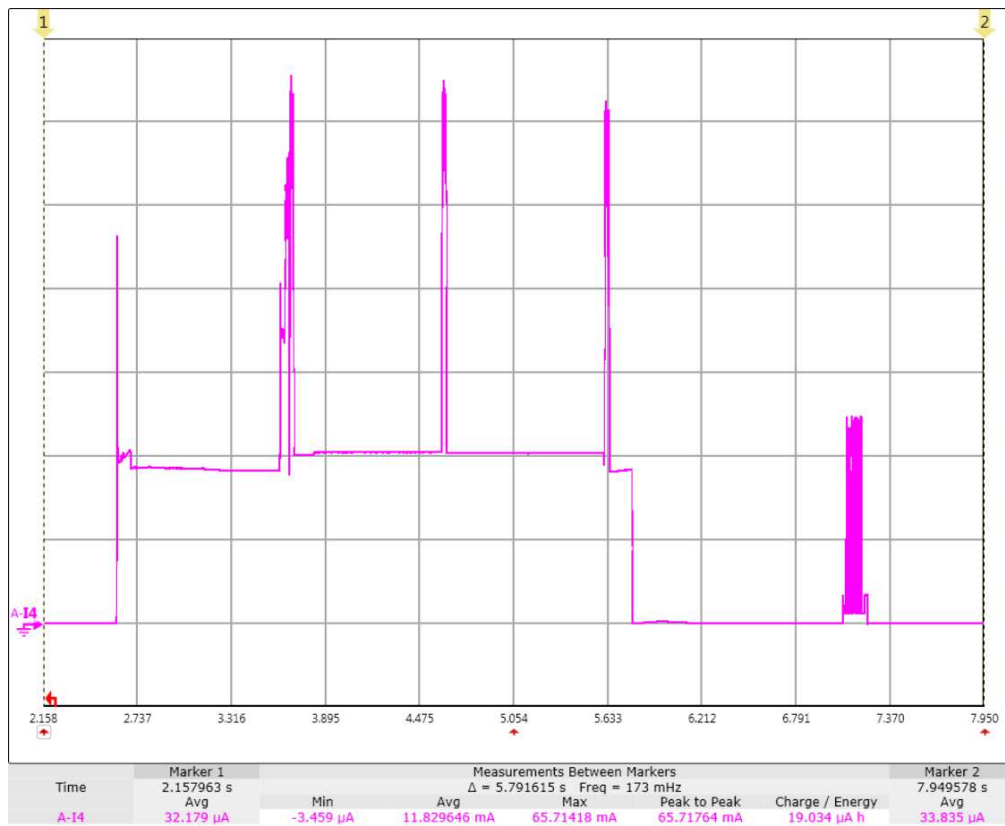


Figure D.2: Triggering and 3s recording with COM board, without IMU



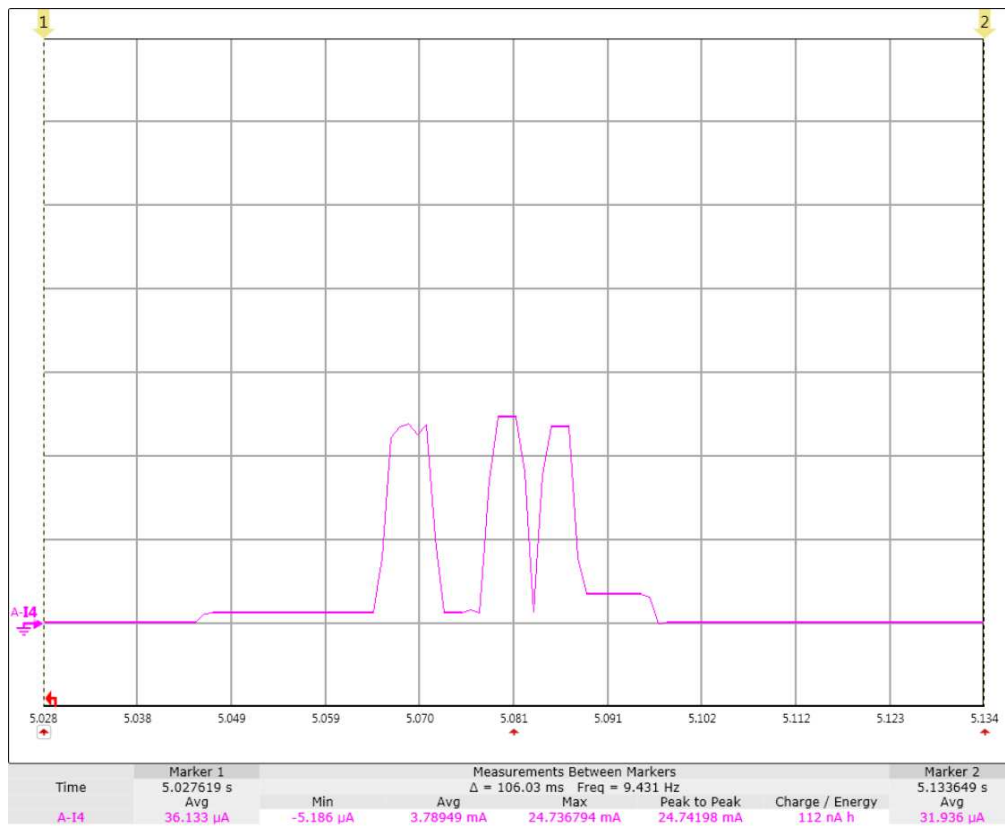


Figure D.3: Power trace of a radio beacon

*This page intentionally left blank.*

## APPENDIX E

# RTOS Traces

---

The SWV real-time trace technology is used for measuring and visualizing software and RTOS performance in real time. The following pages contain the RTOS traces recorded with the SystemView utility.

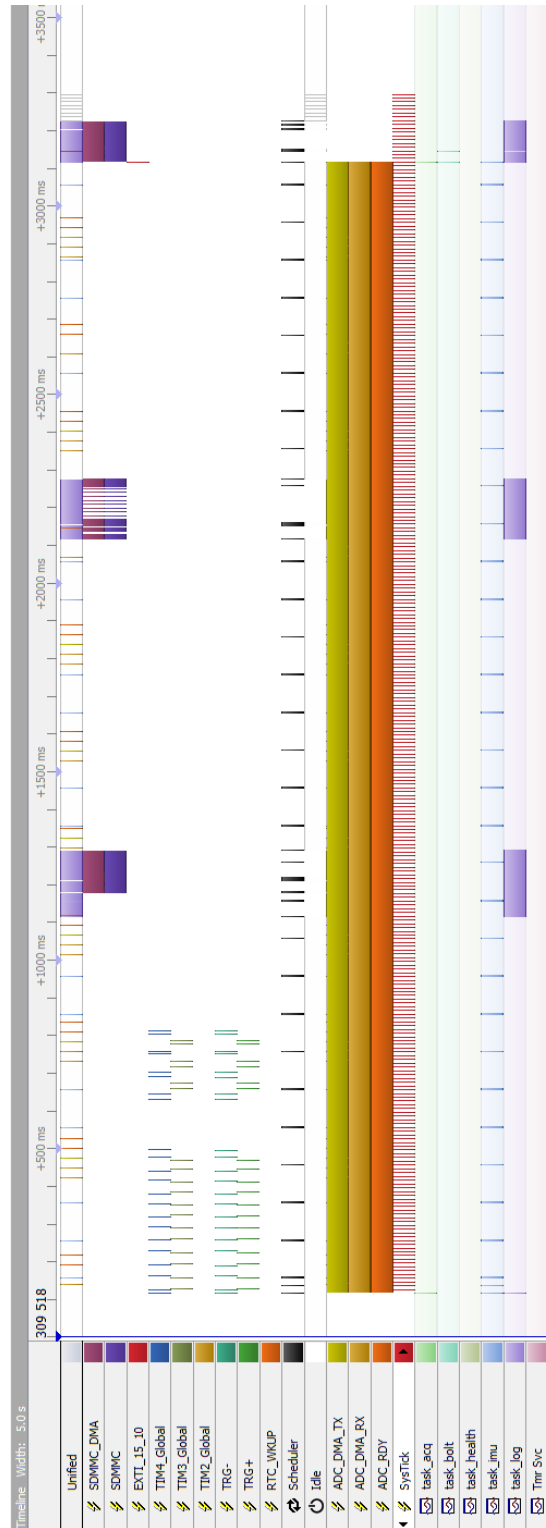


Figure E.1: Trace of a 3-second recording

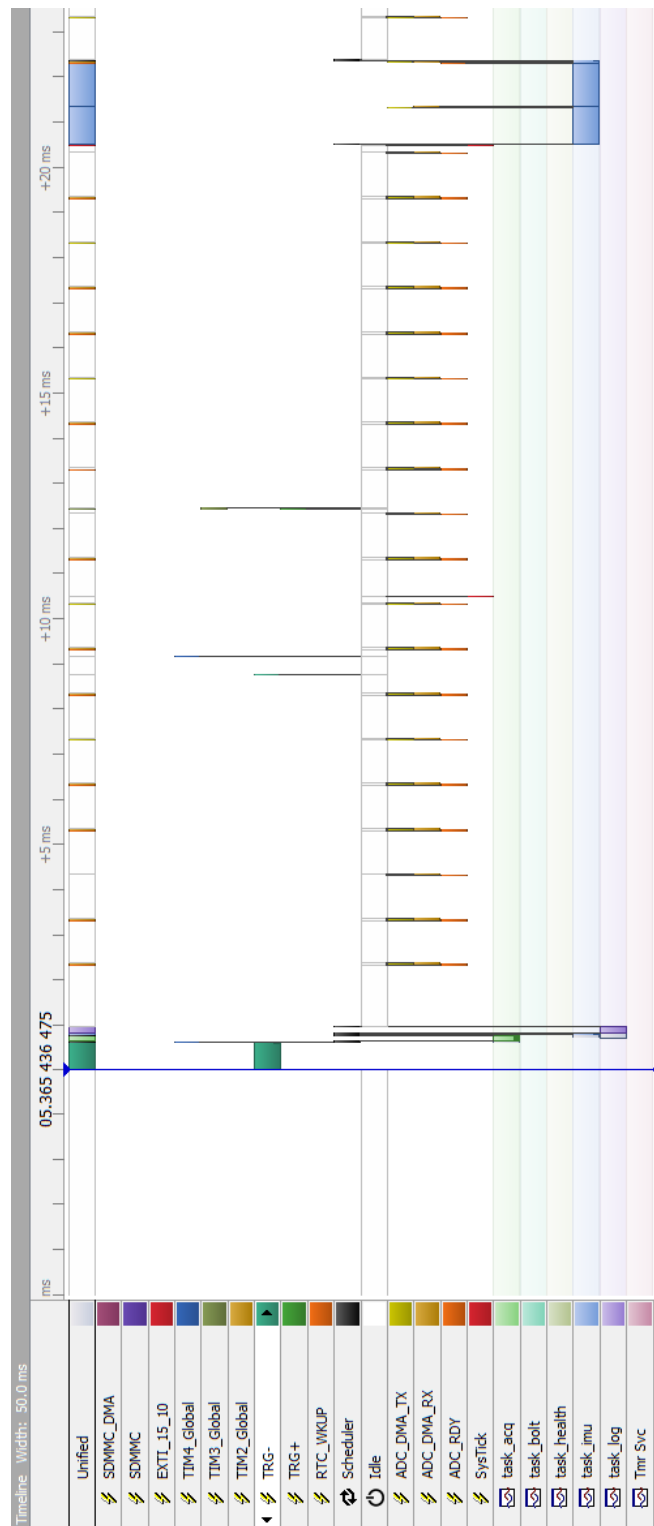


Figure E.2: Beginning of a 3-second recording trace

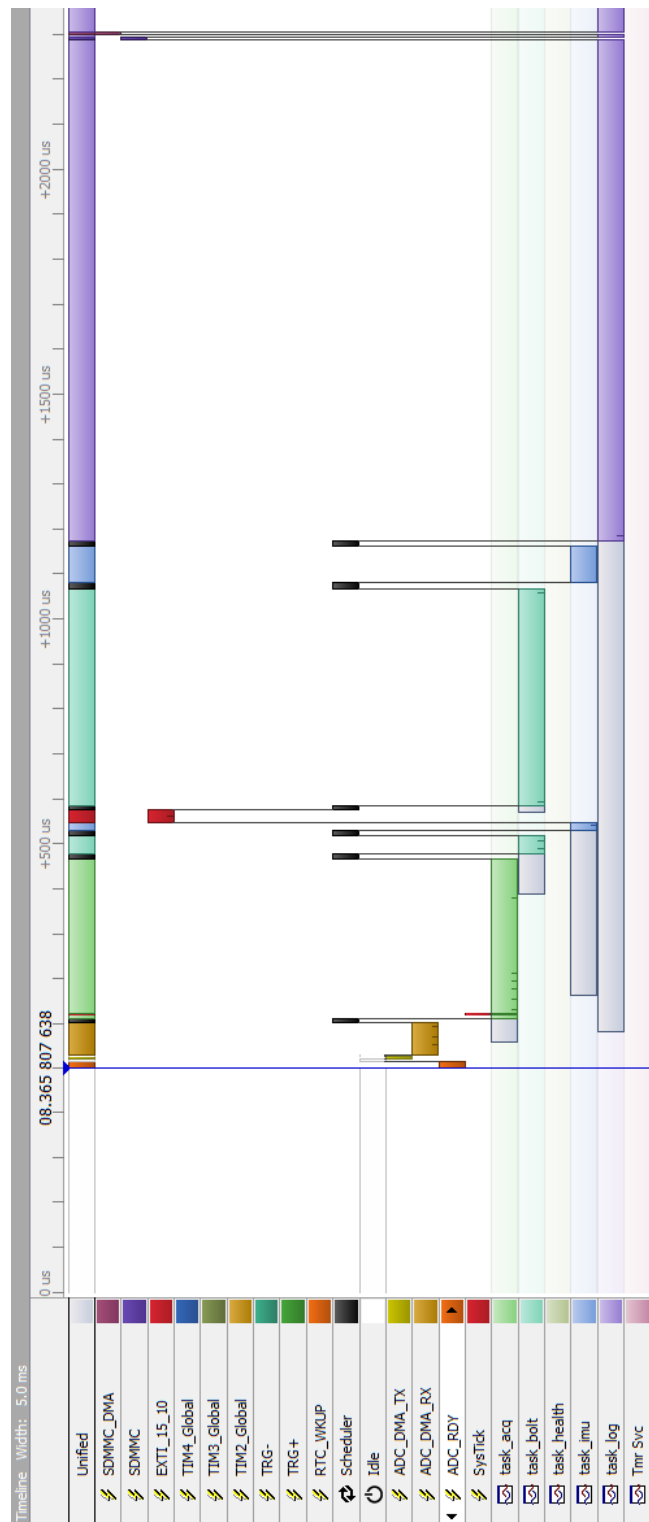


Figure E.3: End of a 3-second recording trace

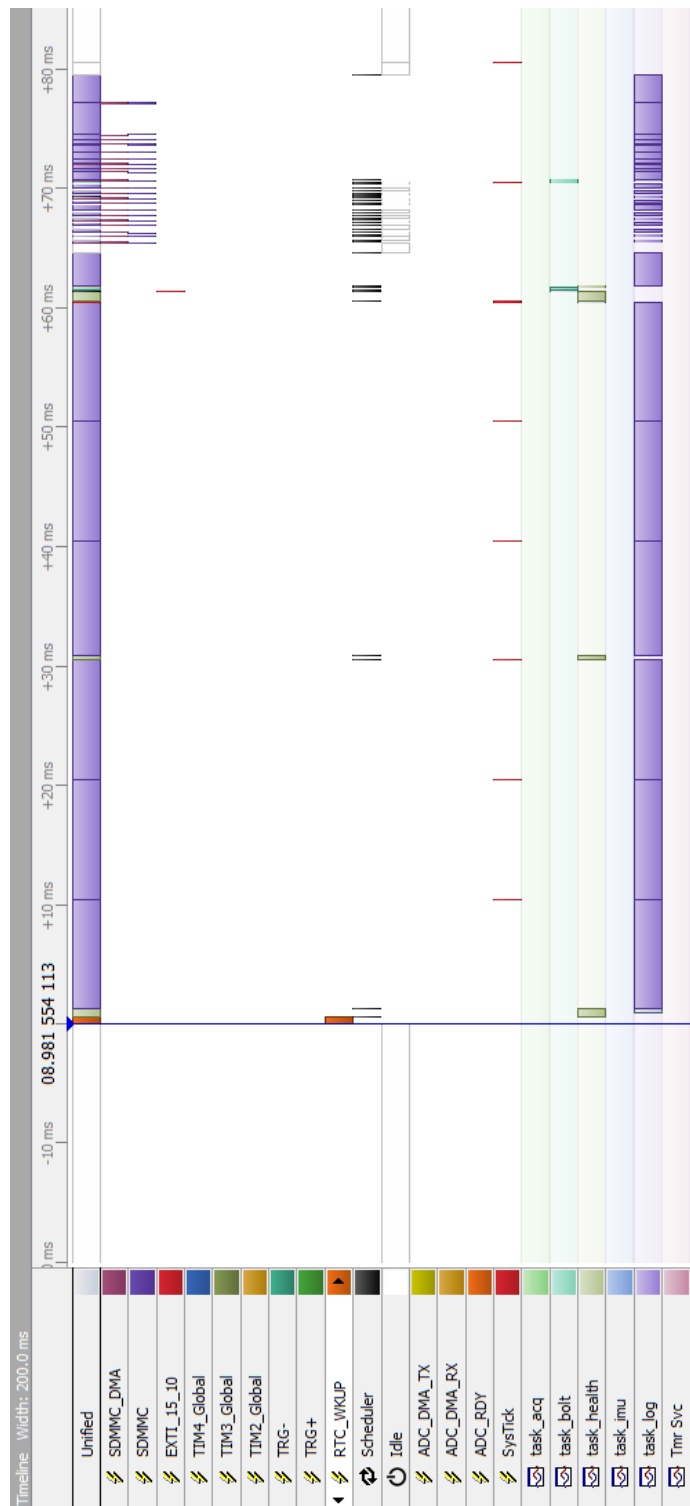


Figure E.4: Trace of a periodic health wakeup

*This page intentionally left blank.*



APPENDIX F

# Field Test Data

---

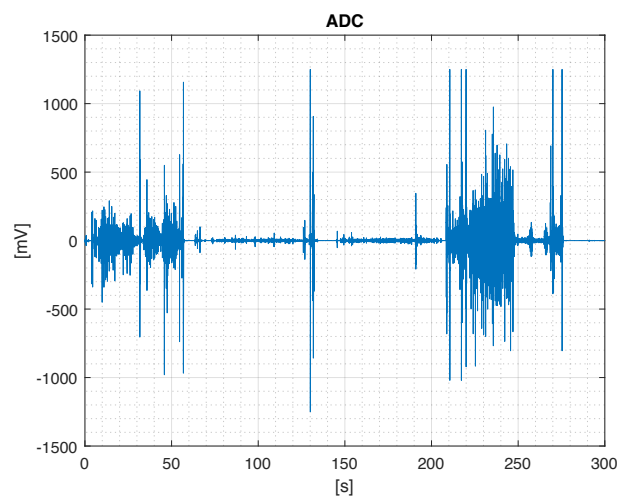


Figure F.1: Data visualization of initialization and enclosure screwing

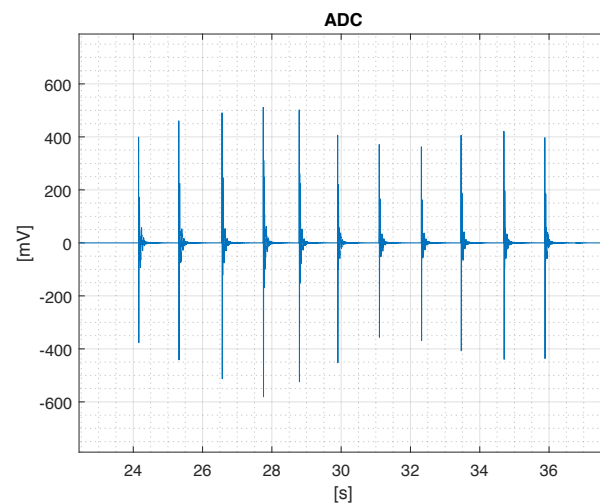


Figure F.2: Data visualization of manual synchronization with hammer

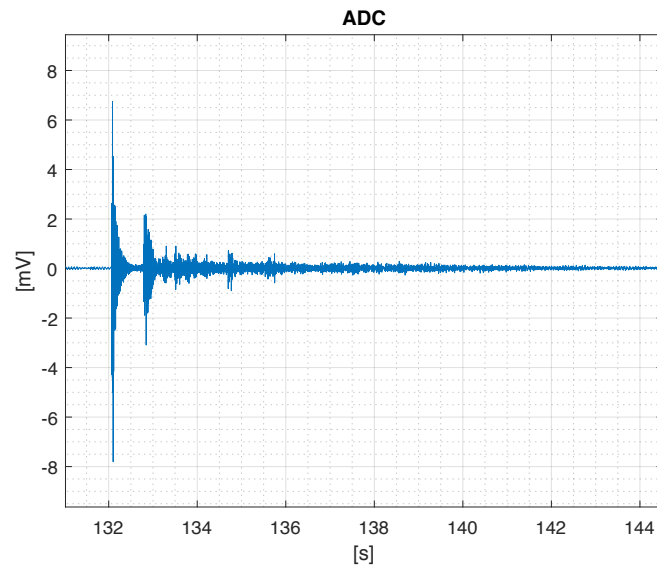


Figure F.3: Visualization of a heavy rock thrown from 5m distance into the pit

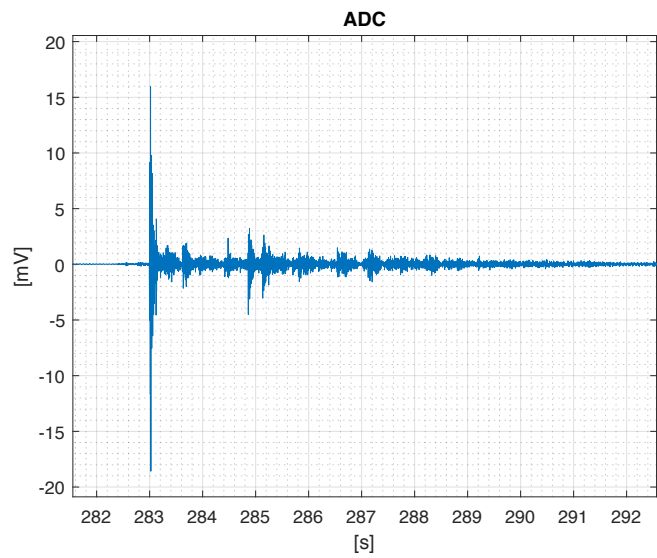


Figure F.4: Visualization of a heavy rock thrown next to node into the pit



Figure F.5: Rock rolling down the pit after hitting the slope

*This page intentionally left blank.*