Daniel Payer, BSc

# Automatic test-bed for a multifunctional electrical stimulation system

**MASTER'S THESIS**

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Biomedical Engineering

submitted to

**Graz University of Technology**

Supervisor

Ao.Univ.-Prof. Dipl.-Ing. Dr.techn. Hermann Scharfetter

Institute of Medical Engineering

Graz, February 2018

# AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis dissertation.

Date

Signature

# Acknowledgments

I would like to thank the company g.tec medical engineering GmbH for giving me the opportunity to write this thesis and Professor Hermann Scharfetter for the supervision.

Special thanks go to my family for supporting me over the whole duration of my studies.

I also owe my friends and fellow students many thanks for having a good time together.

# Abstract

Medical products of a manufacturer need to be conform with a lot of norms and policies in order to be allowed at the market. Therefore, every product has to be checked according to its functionality and safety before being sold. Because of the high numbers of produced devices and the partial lack of resources, it is recommended to service and test these products automatically.

The goal of this thesis is the development of an automatic testing system for a multifunctional electrical stimulator, which checks all included individual components against the defined specifications and thus allows an objective assessment of the functionality of the entire system. After the development and determination of appropriate test steps and test procedures for the present components, such as functional electrical stimulation (FES), transcranial electrical stimulation (tDCs), tactile stimulation and auditory stimulation, as well as a wireless EEG, component tests have to be implemented in a test setup by using the existing APIs. The results of the entire stimulation system as well as the individual single tests shall be documented in a final concluding test report. For this purpose, an own hardware and software, programmed in Python, are developed, which allow the measurement of the most important parameters and evaluate them automatically.



**Keywords: FES, tDCs, EP-unit, Wireless EEG, Test report**

# Kurzfassung

Medizinische Produkte eines Herstellers müssen mit einer ganzen Reihe an Normen und Richtlinien übereinstimmen, um sie überhaupt vermarkten zu können. Alle verkauften Produkte müssen zuvor auf ihre Funktionalität und Sicherheit überprüft werden. Aufgrund der hohen produzierten Stückzahlen und der teils geringen vorhandenen Ressourcen, empfiehlt es sich diese automatisiert zu warten und zu überprüfen.

Das Ziel dieser Arbeit ist es, genau so ein automatisiertes Testsystem für einen Multifunktionsstimulator zu entwickeln, welches die Einzelkomponenten gegenüber die festgelegten Spezifikationen überprüft und somit auch eine objektive Beurteilung der Funktionalität des Gesamtsystems zulässt. Nach der Erarbeitung und Festlegung geeigneter Prüfschritte und Testverfahren für die vorliegenden Komponenten wie funktionelle elektrische Stimulation (FES), transkranielle elektrische Stimulation (tDCs), taktile Stimulation und auditive Stimulation, sowie ein Funk-EEG, sollen Komponententests über vorliegende APIs in einem Prüfaufbau implementiert werden. Die Resultate des Gesamtsystems sowie der Einzeltests sollen in einem abschließenden Verifikationstestprotokoll dokumentiert werden. Hierzu wird eine eigene Hardware und Software, welche in Python programmiert werden soll, entwickelt. Diese erlaubt die Messung der wichtigsten Kenngrößen und wertet diese automatisiert aus.

**Schlüsselwörter: FES, tDCs, EP-Einheit, Funk-EEG, Testprotokoll**

# Contents

# List of Figures

# List of Tables

# Abbreviations

| **Abbreviation** | **Meaning** |
|---|---|
| AI | Analog Input |
| API | Application Programming Interface |
| BCI | Brain Computer Interface |
| CAD | Computer-aided Design |
| DAQ | Data Acquisition |
| DC | Direct Current |
| DLL | Dynamic Link Library |
| DO | Digital Output |
| DUT | Device Under Test |
| EEG | Electroencephalography |
| E.g. | For Example |
| EP | Evoked Potential |
| ESD | Electrostatic Discharge |
| FES | Functional Electrical Stimulaton |
| GND | Ground |
| GUI | Graphical User Interface |
| IC | Integrated Circuit |
| IDC | Insulation Displacement Connector |
| I/O | Input/Output |
| LED | Light Emitting Diode |
| MEP | Motor-evoked Potential |
| MOSFET | Metal-Oxide-Semiconductor Field-Effect Transistor |
| N.C. | Not Connected |
| NI | National Instruments |
| OEM | Original Equipment Manufacturer |
| PCB | Printed Circuit Board |
| PID | Product Identification |
| PFI | Programmable Function Input |
| PSU | Power Supply Unit |
| QMS | Quality Management System |
| RSE | Referenced Single Ended |
| SVN | Subversion |
| tDCs | Transcranial Direct Current Stimulation |
| TTL | Transistor-Transistor-Logic |
| USB | Universal Serial Bus |
| VIB | (tactile) Vibrator |
| VID | Vendor Identification |
| XML | Extensible Markup Language |

# Symbols

| **Symbols** | **Meaning** |
|---|---|
| A | Ampere |
| m | milli |
| G | Giga |
| Hz | Hertz |
| k | kilo |
| p | pico |
| s | seconds |
| S/s | Samples per second |
| V | Volt |
| W | Watt |
| $\mu$ | mikro |
| $\Omega$ | Ohm |

# 1. Introduction and problem

## 1.1. Motivation

Medical products need to be conform with a lot of specifications. Therefore, every product, which is going to be sold, has to be checked according to its functionality and norms. Because of the large quantity of produced products it would be recommended to do the testing automatically according to the device's parameters. Furthermore, the employees do not need to check every device manually, which would also consume a lot of time. An automatic testing environment has the additional advantage that the employee does not need to be present during the testing and so can perform other tasks.

## 1.2. Goal

The purpose of this thesis is to develop a computerized measurement station, which checks the functionality and properties of a multi-functional stimulation system, developed by the company *g.tec (g.tec medical engineering GmbH, Schiedlberg, Austria)*. After the determination of possible test procedures for the existing components, such as wireless electroencephalography (EEG), functional electrical stimulation (FES), transcranial electrical stimulation (tDCs), tactile stimulation and auditory stimulation, the testing station must have test cases for each of these components, which have to be implemented via the existing application programming interfaces (API). The results of each test are finally visualized in a concluding, automatically created test report, which shows whether the specification of the stimulation system corresponds with the test results. For this purpose hardware and software items need to be developed.

In the hardware design a test adapter has to be developed, which makes it possible to measure all kinds of voltages of the stimulation system and its components as well as to save the measured data on the computer for further signal processing. Therefore, a concept for the realization of this device, the electronic circuit and resultant printed circuit board (PCB), created with the software *Eagle (CadSoft Computer GmbH, Pleiskirchen, Germany)*, has to be made.

The software, programmed in *Python (Python Software Foundation, Delaware, USA)*, represents the interface between the hardware items. It controls the devices with the existing API's, which need to be adapted for Python, and checks all performance parameters of the components. Therefore, test routines need to be developed, which verify the functionality of the stimulation system and the adherence to the specification. The test results are summarized in an automatically created test report. To make the software user-friendly as well as easy-to-use, a graphical user interface (GUI) is implemented.

## 1.3. Quality management systems (QMS) according to EN ISO 13485 [19]

If a company manufactures medical devices and wants to sell them, a quality management system (QMS) is required. The EN ISO 13485 describes what such a management system should be like. It is similar to the EN ISO 9001, except for the verification of the increasing customer satisfaction and improvement of the QMS, which are not explicitly requested. A company, which is certified according to EN ISO 13485 complies with the legal requirements of a QMS in Europe. Additionally, if a company complies also with the EN ISO 14971 (Medical devices - Application of risk management to medical devices) the international requirements (USA, Canada, Japan, etc.) of a QMS are mostly satisfied. To completely fulfill the requirements of, for instance, the USA the company should also comply with the FDA 21 CFR part 820 (Quality System Regulatory) norm.
The focus of the EN ISO 13485 is the safety of the products and the implementation of the legal requirements. The essential chapters in this norm are:

- Quality management system
  Describes general requirements and requirements according to the documentation.

- Management responsibility
  Quality management is the responsibility of the company leader according to the establishment, maintenance and efficacy. The norm includes, which contents need to be documented.

- Resource management
  The employees need to be qualified for their activities, which affect the quality of the products. Furthermore, the working area needs to have equipment to manufacture the corresponding product with its properties (machines, tools, etc.).

- Product realization
  Every process for the product realization has to be documented. The norm schedules on which process an examination is done. Additionally the risk management process has also to be integrated into this part.

- Measurement, analysis and improvement
  This part is for the arrangement of characteristic factors and their evaluation. Therefore, the PDCA (Plan-Do-Check-Act) model is used (see figure 1). This model shows a principle for a permanent improvement of the product.

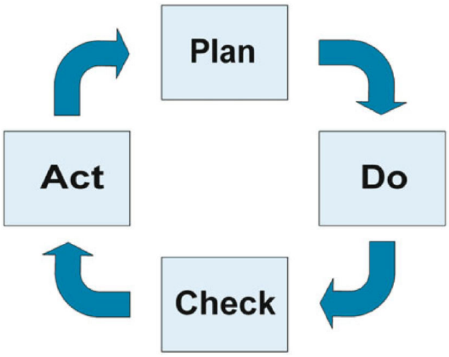


Figure 1: Plan-Do-Check-Act model [2]

- Plan
  Analysis of the current situation and determination of new goals.
- Do
  Execution of the new processes.
- Check
  Check if the processes work and if the goals have been achieved.
- Act
  Add the new processes to the routine and do counteractions if deviations occur. [2]

## 1.4. Electrical stimulation

Electrical stimulation is used to stimulate muscles or nerves of the human body by using electrical current over a specific time. Therefore, electrodes are applied to a specific body part. There are various types of electrical stimulations, depending on the area, where the electrodes are applied, the strength of the current and the duration of the stimulation. It is used in medicine for therapy, such as pain relief and re-education of the muscles, as well as in sports for muscle training and of course in the research area.

### 1.4.1. Functional electrical stimulation (FES)

The functional electrical stimulation is used to stimulate a paralyzed muscle or nerve of the human body in order to make or assist movement or any other function. Various parameters are important in order to excite a muscle.

- The used waveform

- The amplitude of the stimulation current
- The duration of the stimulation
- The frequency
- The pulse width

Depending on the stimulation duration the amplitude of the current is adjusted. The shorter the duration the bigger the needed current to create an action potential and therefore a contraction of the muscle. If the patient feels an uncomfortable pulsing during the stimulation the frequency can be increased. All in all the therapist defines all parameters, depending on the type of therapy and area of stimulation. The medical benefits of FES are cardiovascular conditioning, metabolic benefits, the increase of muscles in size and strength and the recovery of lost bone mass [22].

### 1.4.2. Transcranial electrical stimulation (tDCs)

The transcranial electrical stimulation is a non-invasive neurostimulation of the brain with a constant, low current. In 2000 experiments conducted by Nitsche and Paulus found after-effects in form of an increase of the motor-evoked potential (MEP). Therefore, they delivered a current for 5 minutes with an intensity of 1mA on a patients brain and found that it is possible to increase the MEP amplitude with anodal stimulation. The cathodal stimulation resulted in a decrease of the MEP. In summary, it is possible to enhance/reduce the cerebral excitability for some minutes after the stimulation [24].

## 1.5. Evoked Potentials (EP)

An evoked potential is an electrical potential, which occurs after a presentation of a sensory stimulus. Depending on the type of sensory stimulus an auditory (beep tone), visually (blinking picture), somatosensory (tactile), or olfactory (smell) evoked potential occurs. Every occurrence and disappearance of the stimuli triggers an evoked potential. To be able to isolate the evoked potential from the normal brain activity the mean value of all measured cycles is calculated. Thereby, all random components with no phase relation to each other are minimized while all components with a phase relation are maximized and therefore the evoked potential can be displayed in a diagram. These potentials are measured with the EEG electrodes, for example, with a P300 response. A P300 response is a positive increase of the measured amplitude, when the sensory input has already happened, after 300ms [1].

## 1.6. The stimulation system

Figure 2 shows the side view of the hardware of the stimulation system's prototype. It consists of an all-in-one computer (named medical computer further on) and the stimulation system itself. The medical computer is powered by the stimulation system, which is powered by the main power supply. The sockets of the system are used for audio output, tactile stimulation, FES and tDCs. The used tactile stimulators are shown in figure 2b.


(a) Prototype of the multifunctional electrical stimulation system

(b) Tactile stimulators

Figure 2: Multifunctional electrical stimulation system

1. Medical computer
2. Power switch and LED
3. Fuse
4. Wireless EEG status LED
5. Audio output socket
6. Power supply socket
7. EP-unit output sockets and LED's
8. FES output socket and LED's

9. tDCs output socket and LED's

### 1.6.1. Function of the stimulation system

The system is going to be used in practice for two different situations:

1. To be able to communicate with patients who are in a vegetative state[1]:
   In this situation the device is going to be used by patients who are in a vegetative state, but at least minimally conscious. It is based on brain computer interface (BCI) technology and assesses the level of consciousness of patients suffering from a disorder of consciousness. It also provides the opportunity of a basic communication tool for such patients. Therefore, electrical signals of the brain are measured with the EEG. These data are analyzed and classified by the BCI, which detects the changes in the brain activity induced by the patient's mental activity. If the patient has enough cognitive functions to understand the spoken language, it is possible that simple Yes/No answers to asked questions can be trained. These answers can be obtained by measuring the P300 response of a certain evoked potential, which is produced, for example, by counting the number of heard beep tones. The provided evoked potentials from this device are the auditory and the tactile evoked potential. Depending on the chosen paradigm and the type of used evoked potential a Yes/No answer of the patient is possible. E.g. in case of a tactile evoked potential the patient counts the number of vibrations in the right hand for a Yes, or the number of vibrations in the left hand for a No.

2. As a rehabilitation device for stroke patients[2]:
   A lot of patients who are suffering from the consequences of a stroke are not able to move their extremities anymore. This device should support those people to regain their ability to move again. The principle behind this rehabilitation is, that the patient should try to move, for example, the right or the left hand, or if not possible, to imagine this movement. This task produces a specific pattern in the brain, which can be measured with the EEG via electrodes. A brain computer interface uses this data and detects which hand the patient wants to move and which movement the patient wants to make. Further on, a functional electrical stimulator stimulates the corresponding muscles to perform the thought movement, and so fulfills or supports the patient with this movement. After some rehabilitation sessions the patient's movements will have been improved.

[1] `www.mindbeagle.at`
[2] `www.recoverix.at`

### 1.6.2. Hardware components of the stimulation system

The stimulation system consists of different hardware components. Each of those components has its own circuit board with its own firmware installed. All components are connected via a universal serial bus (USB) hub, which is also built into the housing of the multifunctional stimulation device. The power supply for most of the circuit boards and the medical computer is provided by a main power supply. The only component which is powered by the USB hub is the wireless EEG.

#### 1.6.2.1. Wireless Electroencephalography

The wireless biosignal acquisition system contains a wireless medical amplifier, which is mounted to an EEG cap and sends the measured brain signals to the base station. The base station is the receiver for these signals and represents one circuit board. It also receives a digital transistor-transistor-logic (TTL) level trigger signal, which is used as timing information for an auditory or tactile evoked potential. This hardware item is already a pre-tested device, which means that the testing system does not need to test the whole device again. Nevertheless, there will be a connection test and a timing test for the trigger signal.

#### 1.6.2.2. Functional electrical stimulator

The FES circuit board is a constant current mono/biphasic stimulator used to stimulate the muscles of the human body via electrodes to ensure a movement of a body part. The stimulation system has two functional electrical stimulators implemented (called FES 1 and FES 2). To test if this board is working correctly, a compliance voltage measurement and also current measurements with different amplitudes, frequencies and stimulation durations will be realized.

#### 1.6.2.3. Transcranial electrical stimulator

The tDCs circuit board is a constant low current stimulator. It is used as a non-invasive neurostimulator, which delivers the current to the area of interest in the brain via electrodes positioned on the scalp of the patient. The stimulation system has two transcranial electrical stimulators implemented (called tDCs 1 and tDCs 2). To test this board a compliance voltage measurement, the measurement of the ramp up and down time and various current measurements will be implemented in the test case scenario.

#### 1.6.2.4. Tactile stimulator and the evoked potential unit

The tactile stimulation is done with a small engine which starts to vibrate when it is powered. The direct current (DC) power for this unit comes from the EP-unit circuit board, which has three output sockets and 16 channels in total. If one of in total three connected vibration stimulators is powered, then the same signal, used as a trigger signal, will also be sent to the digital input/output (I/O) pin of the wireless EEG base station. The functionality of the EP-unit circuit board is checked with a measurement of the output voltage and current on each channel. Furthermore, the timing between the trigger and power signal will be measured. To test the functionality of the tactile stimulators a piezo sensor will be used. It generates a voltage corresponding to the strength of the vibration, which will also be measured.

#### 1.6.2.5. USB hub

All circuit boards, which are built into the housing of the stimulation system, are connected with the USB-hub. Therefore, every PCB has its own USB connection with the hub. It also powers the wireless EEG base station and ensures the connectivity with the computer via USB. Now it is possible to communicate with every component of the stimulation system using its corresponding API. Furthermore, each built- in component has its own device name, serial number, product identification (PID) and vendor identification (VID). This information can be used to check if the correct hardware items are built into the stimulation system and if they are supplied with power.

Figure 3 shows all PCB's, cables and sockets built into the multifunctional stimulator.

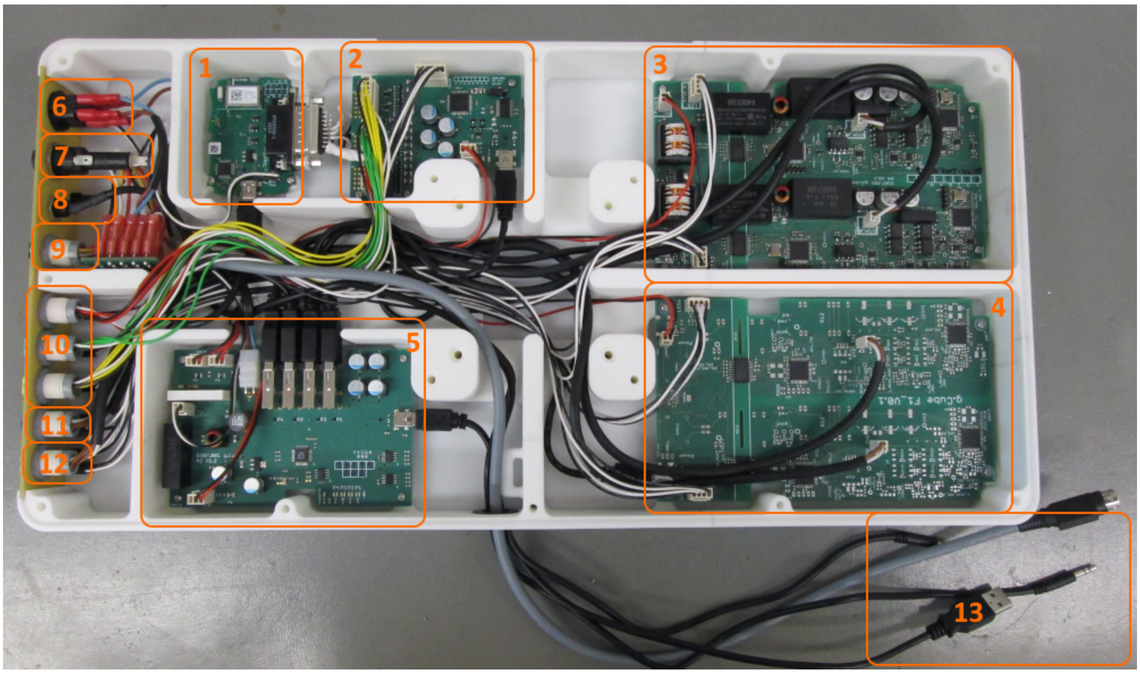



Figure 3: Components of the stimulation system

1. Wireless EEG base station
2. EP-unit
3. FES
4. tDCs
5. USB hub
6. Power switch
7. Fuse
8. Audio output socket and EEG status LED
9. Power supply socket
10. EP-unit output sockets
11. FES output socket
12. tDCs output socket
13. USB, audio input and medical PC power supply (PSU) plug

## 1.7. The testing system

### 1.7.1. Basic concept

After the assembly of every stimulation system a functionality test of each device has to be made, which guarantees that the system works correctly. In detail this means that the composition of the circuit boards, the built- in cables and the connectivity to all sockets is given and correct. Furthermore, the firmwares installed on the microcontrollers need to be fully functional, or else the computer will not be able to communicate with each component. Because of the variety of stimulation system devices and in order to safe a lot of time, a manual examination of the hardware and software is impossible. Therefore, a testing system is built, which checks the whole functionality automatically. At the end of the test of each stimulation system, all results are shown in a final test report.

### 1.7.2. Description of the test cases

#### 1.7.2.1. USB-Hub

**Test A - Functionality of the USB hub:** This test will always be the first executed test and verifies if the composition of the stimulation system's hardware is correct and also if the USB hub is working. Therefore, specific information, such as device name, hardware version, firmware version, PID and VID of each built- in component is requested and compared to a reference list, which contains the never changing parameters for a specific component, such as the device name and PID. If the USB hub or a connected component is not responding, it can be said that e.g. the hub or the component is not supplied with power or that the component is not plugged into the hub. If two components would have the same device name or PID, the composition of the stimulation system or the installation of the firmware might be wrong. If this test does not succeed, no further test will be done.

#### 1.7.2.2. Light emitting diode (LED)

**Test B - Functionality of the LEDs:** This test checks if all LEDs at the housing of the stimulation system's hardware are able to light up and have the correct color. This is the only manual test in the whole procedure, because the LED's and their color are going to be checked visually. To be able to leave the measuring station during all other automatically executed tests, this one will always be done in the beginning, but after the hub test. It has its own GUI in which the user has to turn on a specific LED by pressing a button and check/uncheck the corresponding box and verify whether the LED and its color are okay or not.

#### 1.7.2.3. FES

**Test C - Compliance voltage:** The compliant voltage is ±60V and is measured on a resistor. The resistor needs to withstand the high power of this electrical circuit.

**Test D - Stimulation current:** The stimulation current of the functional electrical stimulator is up to ±60mA. This test should measure all currents from 0 to ±60mA with the help of a shunt resistor. Due to the high power of this electrical circuit, care must be taken that the resistor can withstand this power.

**Test E - Frequency:** The frequency will indirectly be measured by the measurement of the voltage on a resistor with a data acquisition (DAQ) card, which will also be recorded as a dataset. The frequency is then calculated from this data. During the measurement the frequency will be increased step by step from 1 to 100Hz.

**Test F - Durations:** This test includes the measurement of the phase duration, the inter-phase duration and the train duration. For the definition of these durations, take a look at figure 4. These durations will also be indirectly measured and then calculated from the recorded dataset.

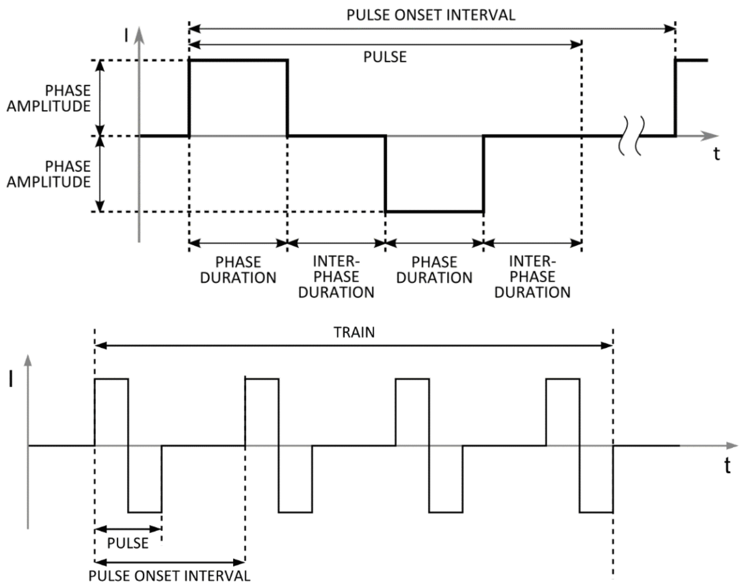



Figure 4: Different types of durations in one stimulus

#### 1.7.2.4. tDCs

**Test G - Compliance voltage:** The compliant voltage is ±20V and is measured on a resistor with a DAQ-card.

**Test H - Stimulation current:** The stimulation current of the transcranial electrical stimulator is up to ±4.5mA. This test should measure the current from 0 to ±4.5mA with the help of a shunt resistor.

**Test I - Ramp up and down time:** The purpose of this test is to ensure that the ramp up/down times of the tDCs circuit board comply with the given specification. The ramp up time is the time in which the amplitude of the current starts to increase until its maximum is reached. The ramp down time is the opposite, which means the time the current needs to decrease from its maximum to zero.

#### 1.7.2.5. EP-unit

**Test J - Output voltage:** The EP-unit's output voltage is used to power the tactile stimulators. This voltage will be measured on a resistor with a DAQ-card and should be 5V on each channel. In total there are 16 channels separated on three sockets. The first and the second socket contain five channels each and the third has six channels. For the tactile stimulators channel one of each socket is used.

**Test K - Output current:** The output current is indirectly measured on a shunt resistor and should be approximately 150mA.

**Test L - Timing** Every time a tactile stimulator is powered, the same signal is sent to the digital I/O pins of the wireless EEG base station. In practice, this signal is the trigger signal for the tactile evoked potential. In the testing scenario it is recorded by the hardware of the test-bed and will additionally be measured with the wireless EEG. The recorded signals are then compared to each other and the time difference between them is calculated.

#### 1.7.2.6. Wireless EEG

**Test M - Connection test:** This test ensures the connectivity between the headset and the base station of the wireless EEG. The easiest and fastest way to check the connectivity is to request the serial number of the headset.

#### 1.7.2.7. Audio

**Test N - Signal check:** The complete audio signal contains four channels, which are the stereo audio signal (left and right channel) and also a trigger signal for each of those channels. The trigger signal is in practice used for the auditory evoked potential. In the test scenario the stereo audio signal is measured with the test-bed's hardware and compared with the original audio signal. Furthermore, the measured data is compared to the trigger signals, which are measured with the digital I/O pins of the wireless EEG base station and allow a evaluation of the timing of these signals.

#### 1.7.2.8. Power supply for the medical computer

**Test O - Output voltage:** This voltage will be measured on a resistor of the test-bed's circuit board and should be 19V.

#### 1.7.2.9. Tactile stimulator

**Test P - Vibration:** As mentioned before, the tactile stimulators are engines which start to vibrate when they are powered. In practice they are used for the evocation of a tactile evoked potential. To test one vibration stimulator it is mounted on an additional hardware equipment, which measures the strength of the vibration with piezo sensors. The piezo sensor transforms this mechanical energy into voltage which is measured with the hardware of the test-bed. To be able to test the tactile stimulators without the multifunctional stimulation system, the power supply is provided by the test-bed's power supply.

#### 1.7.2.10. Temperature measurement

To ensure that the device is tested within a specific temperature range and also that it does not get overheated, the temperature is determined at all times. For this task two sensors will be used. One for the temperature inside the stimulation system and another one for the room temperature. The trend of these temperatures will be shown in a diagram and will be part of every test report.

#### 1.7.2.11. Self-test

Each of the integrated components of the stimulation system has a quick self-test function to ensure the basic functionality of each PCB. This self-test is provided in the API of each device.

# 2. Methods

## 2.1. Hardware

### 2.1.1. Concept of the hardware

Due to the information given in chapters 1.6 and 1.7 the following requirements for the test-bed can be obtained:

1. To realize an automatic testing system, the test-bed must have a USB connection with the computer.
2. The power supply for the testing system is USB. This has the advantage that no secondary power supply adapter is needed.
3. The sockets of the output signals of the stimulation system, which are the input signals for the test-bed, are combined into one D-SUB plug.
4. Due to the variety of components built into the stimulation system different measurement circuits have to be created for each of them.
5. The measurement and recording of the data is done with the help of a data acquisition (DAQ) card and its API created by National Instruments (NI) *(National Instruments, Austin, Texas, USA)*.
6. Most of the measured data are voltages/currents with different amplitudes. To be able to measure these and to not damage the DAQ-card and its analog inputs (AI), voltage dividers are needed. The measurement is done against the ground (GND) plane.
7. Because of the 16 output channels of the EP-unit circuit board and in order to test them simultaneously, the DAQ-card must have at least 16 analog inputs.
8. In order to achieve a good measurement accuracy, the resistors, on which the measurement is done, need to have a low tolerance (maximal $\pm 1\%$) to minimize the measurement deviation.
9. Some of the output signals of the stimulation system have a high power. Therefore, the built- in resistors of the testing system need to withstand this.

10. The tactile stimulators should have a separated output connection socket, to be able to test them without the stimulation system. The power supply for these stimulators is the USB connection with the computer. The measurement is done on a piezo sensor, which generates a voltage corresponding to the strength of the vibration.

11. To be able to switch between the components which are currently tested, solid-state-relays and transistors for the input and output of the testing circuit are used.

12. The switches are controlled via the digital outputs of the DAQ-card. Therefore, there need to be at least 23 of them.

13. The possibility to measure the temperature of the room and inside the stimulation system at all times.

Due to the listed requirements above a block diagram of the measurement circuit can be created.

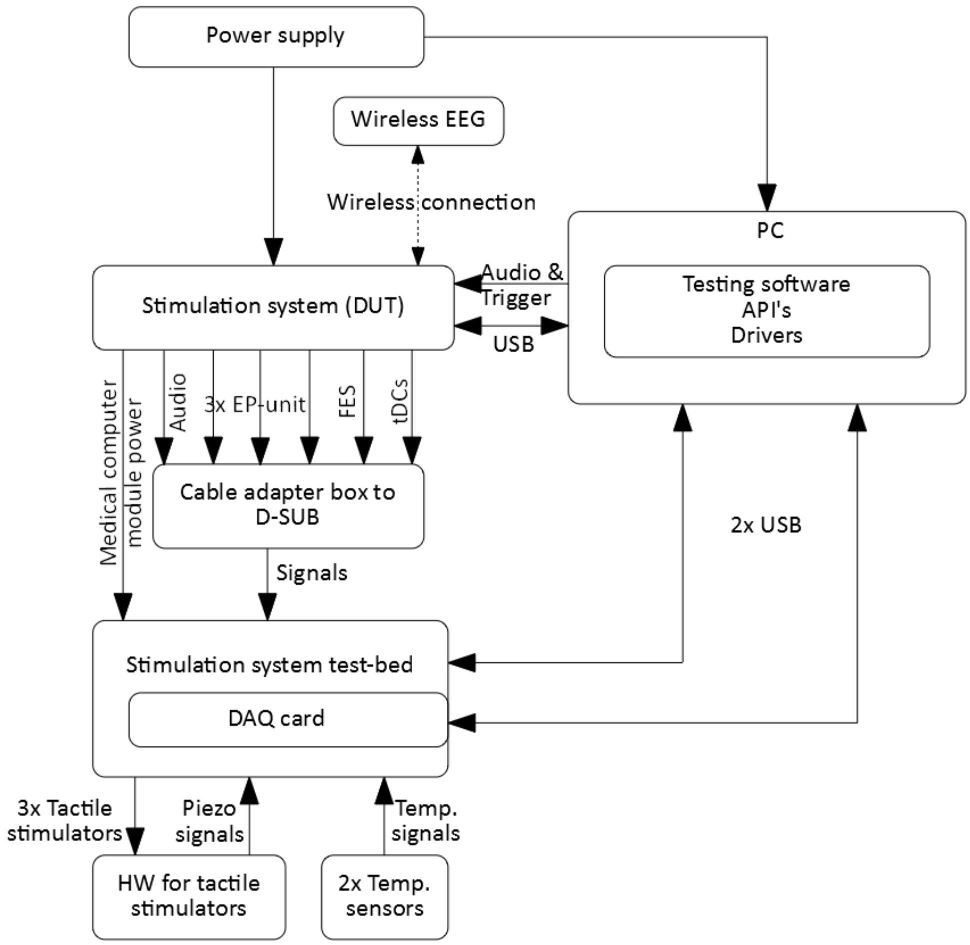



Figure 5: Block diagram of the measurement circuit

Figure 5 shows this diagram and describes the basic functionality of the measurement circuit. To actually test the stimulation system, it is connected with the computer via USB, and with the test box via D-SUB with a cable adapter. The power supply for the device under test (DUT) is the main power supply. The stimulation system has also a wireless connection with the EEG headset to measure the EEG data. In practical use this would be the patient's EEG, but in the test procedure it will just be an empty dataset. The computer also sends an audio signal to the DUT, which includes four channels. Two channels are used for the headphones (stereo sound) and the other two are used as trigger signals to be able to measure an auditory evoked potential with the EEG. In case of a tactile evoked potential the trigger signal is sent from the EP-unit instead and is received by the wireless EEG base stations digital I/O. The test-bed itself measures all output signals from the stimulation system, which are the medical computer module power, the stereo audio signal, the Ep-unit signal, the FES signal and the tDCs signal, with the embedded DAQ-card and sends them via USB to the computer for further signal processing and analysis. The signals coming from the wireless EEG such as the audio or the tactile trigger signal, can be directly obtained by using the EEG base stations API. Last but not least a measurement of the tactile stimulators needs to be done. Therefore, a separated hardware item, in which each stimulator is mounted above a piezo sensor, has been built. These piezo sensors generate a specific amount of voltage depending on the strength of the vibration of the stimulators. Last the temperature is measured by two temperature sensors. All these signals are then received by the DAQ card of the test-bed and sent to the computer via the USB connection.

The composition of the test-bed is shown in figure 6.

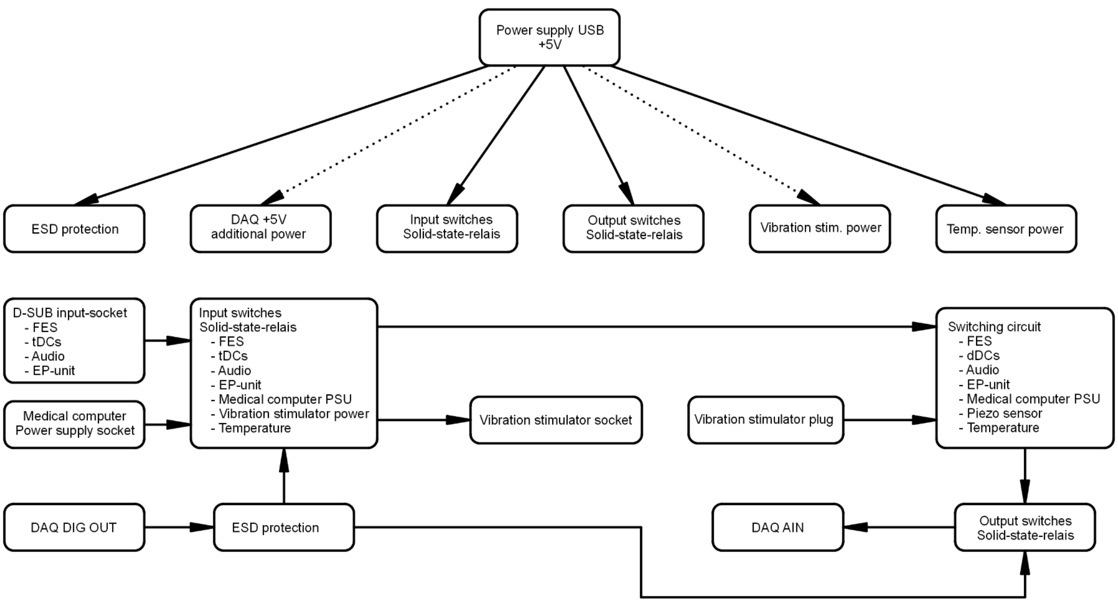



Figure 6: Block diagram of the composition of the test-bed

The basic idea is to test each component of the stimulation system separately. To be able to do this, switches for the input and output signals are implemented. These switches are controlled by the digital outputs of the DAQ-card. To actually measure a signal, the corresponding device is activated with its API. The signal is then sent via the D-SUB into the testing system. Then the DAQ-card sets the corresponding input and output switches and records the signal. The input switches are used to apply a signal to the correct measurement circuit (switching circuit). On the other hand the output switches are used to measure this signal from the measurement circuit with the correct analog input pin of the DAQ-card. The measured result is an analog voltage, which is stored into a variable of the test-bed's software and is used for further signal processing and calculation. The measurement result is always a potential difference over a specific time measured on a resistor. All other measurement parameters, such as, the frequency, the current and all kind of durations will be indirectly measured, which means calculated by using the potential difference.

#### 2.1.2. Design and development of the hardware

The circuit board of the prototype's test-bed should not be too small, because all electrical components are manually soldered. Furthermore, the manual measurement, which can be conducted with a multimeter on a specific test point, verifies the hardware. In case of a wrong connection or an incorrectly built- in component, it is also possible to make changes to the PCB. If the PCB is minimized, an automatic assembly will be needed, which is not advisable, because of the additional fees and the missing opportunity to make changes in case of an error.

##### 2.1.2.1. The DAQ-card [17]

The DAQ-card is the main component of the stimulation system's test-bed. It receives commands from the computer and sends signals to it via the USB connection. To be able to communicate with the DAQ-card the driver[3] has to be installed. The corresponding API is also provided in the driver's installation. Figure 7 shows the used DAQ-card.

[3] `http://www.ni.com/download/ni-daqmx-17.1.1/7045/en/`

Figure 7: DAQ-card NI USB-6216 OEM [17]

This version of the card has been chosen because:

- It has 16 analog inputs with a resolution of 16bit and a maximum sampling rate of 400kS/s. The 16 analog inputs are needed to be able to measure all 16 outputs of the EP-unit simultaneously. Furthermore, the resolution and the maximal possible sampling rate are high enough to achieve accurate measurement results.
- It has up to 32 digital I/O's. At the moment there are 23 of them needed as digital outputs for the control of the input and output switches. Furthermore, there is the possibility of extensibility for further versions of the test-bed.
- The USB connection is another advantage, because it is possible to use this card with every computer, which has the driver installed.
- This version of the card is already used for other projects in the company.
- The original equipment manufacturer's (OEM) version has been chosen, because of the possibility to mount the card on the PCB of the test-bed, and because it is cheaper than the normal version. This has the advantage that no other device is placed on the measurement station and no additional cables are needed. The DAQ-card is connected with the test-bed with two 50 pin insulation displacement connectors (IDC).
- The device is protected against overvoltage, undervoltage and overcurrent conditions as well as electrostatic discharge (ESD) events.

For further and more specific information take a look at the specification document [16] of this device.

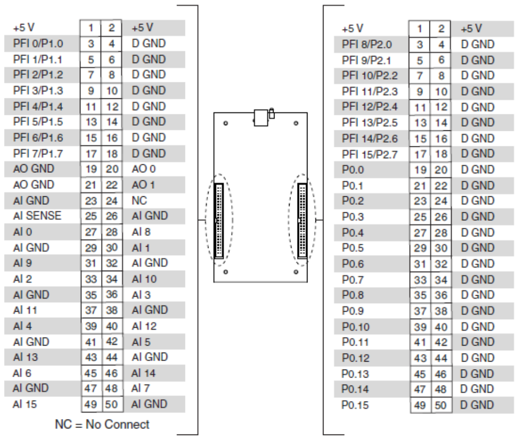



Figure 8: Connector pinout of NI DAQ-6216 OEM [18]

Figure 8 shows the IDC connector pinout of the DAQ-card. This pinout is used to create the female IDC connector in *Eagle* for the test-bed PCB to be able to mount the card on it. Any error would result in a non functioning testing system. All pins starting with a P are digital pins. They can be either configured as digital input as well as digital output pins. The programmable function input (PFI) pins can additionally be used as clocks for synchronization, counters and triggers. The total internal current limit for all digital outputs is 50mA. If more current is needed, there is the opportunity to connect an external +5V power supply with the +5V pin. The analog inputs of the DAQ-card are used to measure the potential difference on a resistor over a specific time and to record and save this signal on the computer. The input impedance, if the DAQ-card is turned on, is at least 10GΩ in parallel with 100pF. So a high value of the measurement resistor should also be no problem according to the accuracy of the measurement. Furthermore, to be as accurate as possible the DAQ-card has the opportunity to adjust the input ranges from ±0.2V up to ±10V and as a result its resolution. The measurement itself is done in the referenced single ended (RSE) mode, which means the device measures the voltage of an AI signal relative to the AI GND. [15]

#### 2.1.2.2. The power supply

The USB connection with the computer powers the test-bed with +5V and a maximum current of 500mA. This is enough power to supply the complete test-bed. Because of that, there is no additional need for an external power supply adapter. The device is saved against a short circuit with a built- in 500mA fuse. Furthermore, every electrical component is powered with these 5V, also each tactile stimulator, which requires a current of 150mA to function correctly. The only component, which is not supplied by the test-bed's power is the DAQ-card, because it has its own USB connection with the computer and therefore its own power supply.

#### 2.1.2.3. The switches

**N-channel metal-oxide-semiconductor field-effect-transistor (MOSFET) NTR3C: [26]**

This MOSFET is an N-channel MOSFET with a very low resistor value when turned on ($\sim 24m\Omega$). In the test-bed, they are used as switches to control the solid state relays. Since the transistors are controlled with the digital outputs of the DAQ-card their gates are protected against electrostatic discharges with low capacitance diode arrays (datasheet [27]), which are assembled in between.

Figure 9 shows an extraction of the PCB circuit for one output switch. As mentioned before, the transistor is used as a switch. To be able to close the circuit between the drain-source pole of the transistor, the gate-source voltage needs to be greater than the gate threshold voltage of the transistor, which is specified in the datasheet and is 1V maximum. To achieve this functionality, the DO15 pin of the DAQ-card is set to high, which means +5V are applied to the transistor. The exact potential difference on the gate-source pole is a little bit lower ($U_{GS} = U_{R_{111}} = \frac{10}{11} \cdot U_{in}$ (see formula 1)) than the applied voltage, because of the voltage divider $R_{108}$ and $R_{111}$ in the circuit. The potential difference on the gate-source pole is now greater than the gate threshold voltage of the MOSFET, which means the transistor connects its drain-source pole trough and the relay LCC110 is supplied with power.

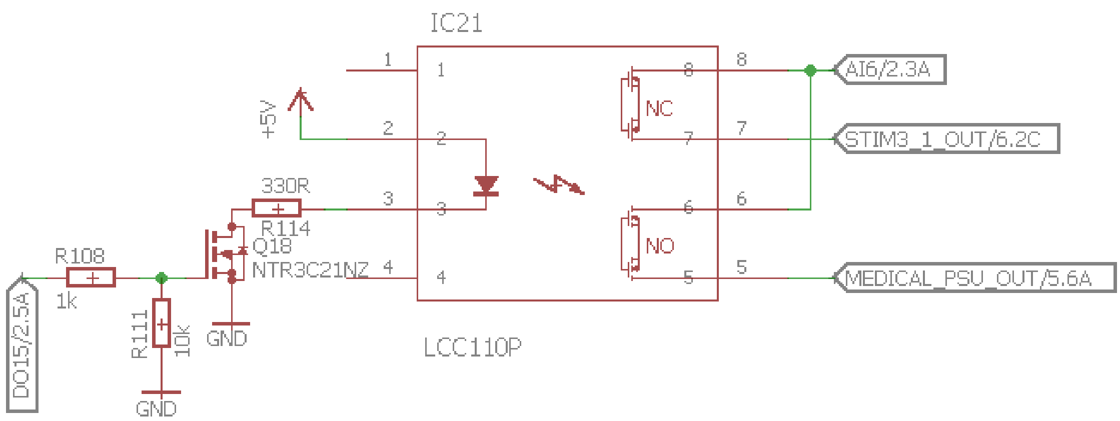



Figure 9: Example of the switching behaviour with the LCC110

$$U_2 = U_1 \cdot \frac{R_2}{R_1 + R_2} \tag{1}$$

**Solid state relay LCC110:**

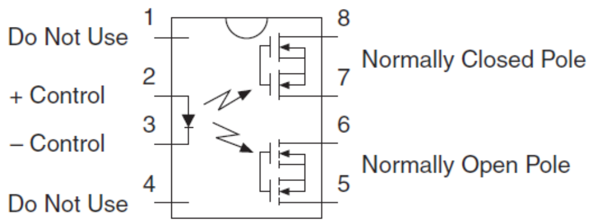



Figure 10: Solid state relay LCC110 [9]

Figure 10 shows the pin configuration and the resultant switching behavior of the LCC110P. It is a form C solid state relay, which means one pole is always closed and one is always open. One AI pin of the DAQ-card is connected with the pins 6 and 8 and one output of the measurement circuit is connected with pin 5 and another measurement circuit output with pin 7. If there is no voltage applied on pin 2 and 3, the relay behaves as it is shown in figure 10, which means the signal measurement with the DAQ-card will be done with the connected measurement circuit on pin 7. If there is a voltage applied, the behavior of the relay is vice versa and the measured signal, is the signal from the measurement circuit which is connected with pin 5. The switching speed of this relay depends on the amount of current used (in case of the test-bed it is ∼10mA) at the input and can be seen in the diagrams of the data sheet [9].The input of the LCC110 is accessed through a N-channel MOSFET which is used as a switch. This MOSFET is controlled by one DO pin of the DAQ-card as it is shown in figure 9.

*Example calculation for the adjustment of the input current for the relay by dimensioning the resistor* $R_{114}$*:*
The current I should be ∼10mA. The input voltage $U_{in}$ is 5V and the voltage drop of the diode $U_{diode}$ is ∼1.3V.

$$R_{114} = \frac{U_{in} - U_{diode}}{I} \tag{2}$$

$$U = R \cdot I \tag{3}$$

The result of equation 2 is 364Ω. Because this resistor is not available in the E12 resistor value table, the next smaller one has been taken, which is 330Ω. A smaller resistor results in a higher current. Now the exact value of the current can be calculated with Ohm's law 3, where U is the electrical voltage, I is the electrical current and R is the resistor, which is 11,2mA.

**Solid state relay CPC2030N:**

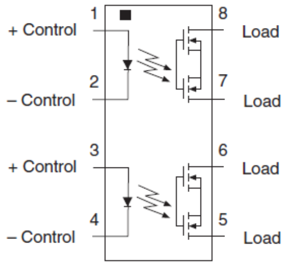



Figure 11: Solid state relay CPC2030N [8]

Figure 11 shows the pin configuration and also the switching behavior of the CPC2030N. It is a form A relay with two independent normally open solid state relays. If, for example, the input at the pins 1,2 is powered the switch between the pins 7,8 closes. This relay is used as an input switch for the signals coming from the FES and tDCs circuit board of the stimulation system. It is needed to switch potential-free because the FES and tDCs signals are differential signals, which have no reference to ground. Each relay input is controlled with the same MOSFET as the LCC110, but the input current for it is adjusted to ∼5mA. For the switching speed take a look at the diagrams in the data sheet [8].

#### 2.1.2.4. The temperature sensor MCP9701

During all tests, performed by the test-bed, the room temperature and the temperature of the stimulation system is measured. For this purpose, the temperature sensor MCP9701 has been chosen, because it needs a +5V power supply, which is already available due to the USB connection, without any further circuits and it has an accuracy of ±2°C. Figure 12 shows the used temperature sensor.

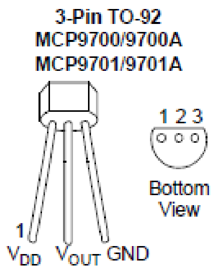



Figure 12: Temperature sensor [23]

For further details take a look at the datasheet [23].

#### 2.1.2.5. The measurement (switching) circuits

This part of the test-bed's circuit board is used to divide the input voltage, coming from the stimulation system via the D-SUB input, down to a measurable input voltage for the DAQ-card's analog inputs. Because of the maximum input voltage of ±10V of the DAQ-card and to fit into this measurement range, it contains a lot of resistors with different values, for voltage dividing, depending on the component, which is measured. Furthermore, to be able to calculate, for example, the current a reference value, given by the resistor value, is needed. The used measurement circuit is selected by the setting of the corresponding input and output switches with the digital outputs of the DAQ-card. The pinout for these switches is shown in chapter 2.1.2.8. Figure 13 shows an example of one voltage divider, which is used for one pin of the EP-unit.

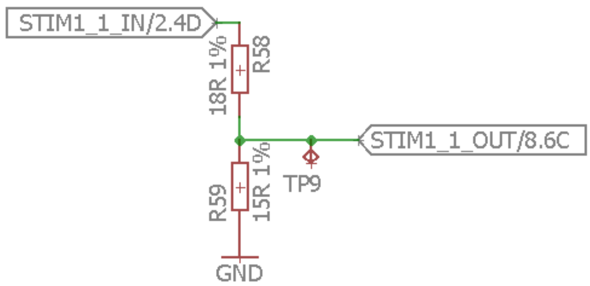



Figure 13: Used voltage divider of the EP-unit for one pin

The calculation of the resistor values is done with the formula of Ohm's law 3 and with the voltage dividing rule 1. In order to save the resistors from damage, when the specific measurement circuit is powered, care must be taken with the electrical power, which is calculated with formula 4, where P is the electrical Power, U the electrical voltage and I the electrical current.

$$P = U \cdot I \tag{4}$$

In case of a voltage or a current, which is not applied at all times, the root mean square value of this unit is calculated and further on taken for the power calculation. In case of a current, formula 5 is used, where $I_{rms}$ is the root mean square value of the electrical current, T is the periodic time, $i_{(t)}$ is the current in a specific duration of time and t is the time.

$$I_{rms} = \sqrt{\frac{1}{T} \cdot \int_0^T i_{(t)}^2 \cdot dt} \tag{5}$$

If the input signal is provided by a voltage source (EP-unit, medical computer power supply), the calculated total resistor needs to be able to draw the maximum output current of this source. In case of a current source (FES, tDCs) it needs to get the maximum voltage.

Furthermore, the resistor must be able to withstand the maximum power corresponding to the maximum voltage of this source.

Most of the built- in resistors have a power limit of 100mW, which is a standard value. This is enough for most of the taken measurements. In case of the EP-unit measurement the total resistor needs to withstand nearly 1W. This power is split onto two resistors, which can withstand a maximum power of 500mW each (see figure 13).
The resistors of the FES measurement circuit need to withstand even more. If the calculation had been done with a constant current and with formula 4 the result would have been 14.4W, but in this specific case the current is just applied for a short duration of time and so the calculation of this current is done with formula 5 before using this result for equation 4. The result of the maximum power decreases now to ~3,5W.
Some other units, such as the audio signal and the voltage of the temperature and the piezo sensors, are directly measured by the analog input pins of the DAQ-card. The reason is, that these units have a voltage amplitude, which is within the ±10V input range of the DAQ-card and additionally there is no need for a reference value for further calculation of another parameter.

At the moment this version of the prototype based on the FES component is just capable of delivering a maximum output current of ±60mA and therefore an output voltage of ±60V.But further versions of the stimulation system will have a maximum output of ±120mA and ±120V. The hardware is already designed to withstand this higher power and therefore the calculation values above are for the higher values of the current and the voltage.

#### 2.1.2.6. Hardware for the measurement of the tactile stimulator

The idea behind this hardware item is, to mount each of all three tactile stimulators above one piezo element. The pressure on the stimulators should be the same, when mounted. They are powered one after another to have no influence on each other during the measurement. The power supply for the tactile stimulators is the power supply of the test-bed, which is provided after switching the corresponding input switch. When one is powered, the piezo element generates an output voltage which can be measured on a parallel resistor. The amplitude and also the frequency of the measured output voltage depend on the strength of the vibration of this stimulator. The voltage can be measured with the AI pin of the DAQ-card and the frequency can be calculated from the recorded signal. If the amplitude of the first piezo sensor varies too much when compared to the second or the third, a calibration for the piezo sensors has to be executed.

#### 2.1.2.7. The D-SUB adapter cable

The D-SUB adapter cable connects all stimulation system output sockets with the test-bed. All signals, such as, FES, tDCs, EP-unit and the audio signal are sent via this cable to the D-SUB socket of the test-bed and are used as input signals for the measurement circuits. Table 1 shows the pinout of this cable.

Table 1: D-SUB adapter cable pinout

| **D-SUB 37 input pin** | **Stimulation system output pin** |
|---|---|
| 1 | Audio left channel |
| 2 | Audio GND |
| 3 | Not connected (N.C.) |
| 4 | FES 2 + |
| 5 | FES 2 - |
| 6 | N.C. |
| 7 | tDCs 2 + |
| 8 | tDCs 2 - |
| 9 | N.C. |
| 10 | EP-unit 3 GND pin 7 |
| 11 | EP-unit 2 GND pin 6 |
| 12 | EP-unit 3 pin 5 |
| 13 | EP-unit 3 pin 4 |
| 14 | EP-unit 3 pin 3 |
| 15 | EP-unit 3 pin 2 |
| 16 | EP-unit 3 pin 1 |
| 17 | EP-unit 2 pin 1 |
| 18 | EP-unit 2 pin 2 |
| 19 | EP-unit 3 pin 6 |
| 20 | Audio right channel |
| 21 | N.C. |
| 22 | FES 1 + |
| 23 | FES 1 - |
| 24 | N.C. |
| 25 | tDCs 1 + |
| 26 | tDCs 1 - |
| 27 | N.C. |
| 28 | N.C. |
| 29 | EP-unit 1 GND pin 6 |
| 30 | EP-unit 1 pin 5 |
| 31 | EP-unit 1 pin 4 |
| 32 | EP-unit 1 pin 3 |
| 33 | EP-unit 1 pin 2 |
| 34 | EP-unit 1 pin 1 |
| 35 | EP-unit 2 pin 5 |
| 36 | EP-unit 2 pin 4 |
| 37 | EP-unit 2 pin 3 |

#### 2.1.2.8. DAQ-card pinouts of the test-bed

Table 2 shows which DO pin of the DAQ-card controls which integrated circuit (IC) or device and which AI pin measures which signal. The ICs in the table are the solid state relays (LCC110) described in chapter 2.1.2.3, which control which device of the test-bed's output is measured. The abbreviation VIB is the power supply for the tactile stimulators. They are each connected with a MOSFET, which is used as a switch and has no connection to any relay. On the other hand, the FES and tDCs circuits are additionally connected with a solid state relay (CPC2030N).

Table 2: DAQ pinout of the analog inputs and digital outputs

| **Input Circuit** | | **Output Circuit** | | |
|---|---|---|---|---|
| **DAQ DO** | **Switching Circuit** | **Measurement 1** | **Measurement 2** | **DAQ AI** |
| 0 | IC 25 | EP-unit 2 Pin 2 | Audio Left Channel | 9 |
| 1 | IC 23 | EP-unit 3 Pin 4 | tDCs 1 | 10 |
| 2 | IC 16 | EP-unit 3 Pin 2 | Piezo 1 | 2 |
| 3 | IC 19 | EP-unit 1 Pin 3 | tDCs 2 | 3 |
| 4 | IC 26 | EP-unit 1 Pin 1 | Piezo 2 | 11 |
| 6 | IC 24 | EP-unit 3 Pin 3 | Temperature 2 | 7 |
| 8 | IC 20 | EP-unit 2 Pin 3 | Temperature 1 | 5 |
| 10 | IC 17 | EP-unit 2 Pin 1 | Piezo 3 | 4 |
| - | - | EP-unit 1 Pin 5 | - | 14 |
| - | - | EP-unit 2 Pin 5 | - | 13 |
| - | - | EP-unit 3 Pin 5 | - | 12 |
| - | - | EP-unit 3 Pin 6 | - | 15 |
| 12 | VIB 1 | - | - | - |
| 13 | VIB 3 | - | - | - |
| 14 | VIB 2 | - | - | - |
| 15 | IC 21 | EP-unit 3 Pin 1 | Medical PC PSU | 6 |
| 16 | FES 2 Load Switch | - | - | - |
| 17 | FES 1 | - | - | - |
| 18 | FES 2 | - | - | - |
| 19 | FES 1 Load Switch | - | - | - |
| 20 | tDCs 1 Load Switch | - | - | - |
| 21 | tDCs 1 | - | - | - |
| 22 | tDCs 2 | - | - | - |
| 23 | tDCs 2 Load Switch | - | - | - |
| 24 | IC 22 | EP-unit 1 Pin 4 | FES 2 | 8 |
| 25 | IC 15 | EP-unit 1 Pin 2 | Audio Right Channel | 0 |
| 26 | IC 18 | EP-unit 2 Pin 4 | FES 1 | 1 |

*To explain the table in more detail some examples are provided:*

- If the DO pin 12 is set high then the tactile stimulator 1 is supplied with power and starts to vibrate. By the means of this table no signal measurement is done. If an additional pin, such as DO 2 is set high then the measurement on piezo sensor 1 is done and the DAQ measures this voltage with the AI pin 2.
- The only measurement circuit which does not need any DO signal is the circuit for the EP-unit. All EP-unit pins are connected with the appropriate AI pin of the DAQ-card, if all switches are set to low. E.g. EP-unit 2 pin 2 is connected with AI9. Furthermore, the EP-unit has no input switch, which means, if a signal is applied the DAQ-card has the opportunity to directly measure these signals.
- In case of a FES measurement the input and the output relay needs to be set to high. For the FES 1 device this would be DO17 for the input and DO26 for the output switch. Because of the output switch the signal can be measured with AI1 of the DAQ-card.

The FES and tDCs load switch measurement circuits are going to be used in future versions of the test-bed. With these circuits a calibration of the FES and tDCs PCBs will be possible. Therefore, a voltage divider with external plugged- in resistors can be built and the output currents of each device will be measured. All measurement results, which are the current amplitudes, should be within a tolerance limit on a linear curve starting at 1mA and ending at 60mA.

#### 2.1.2.9. Housing and sockets

To assure a simple and easy to handle testing station the test-bed circuit board is built into a housing with the same sockets that the prototype of the stimulation system has. The following sockets are mounted on the test-bed:
**Front panel:**

- Two 8 pin medical sockets and one 10 pin medical socket used for the connection with the tactile stimulators
- One 4 pin socket for the connection with the medical computer power supply plug
- Six sockets for the connection with the piezo sensors
- Six sockets for the connection with the temperature sensors
- Sixteen sockets for the external resistors, which will be used for the calibration of the FES and tDCs circuit boards in future

**Back panel:**

- One D-SUB 37 pin socket used to connect the stimulation system with the test-bed via an adapter cable 2.1.2.7
- One mini-USB socket for the power supply of the testing system

Figure 14 shows the raw housing for the test-bed and figure 15 shows the dimensions of it. This box has been chosen, because of its ideal size. The front and back panel for this housing are not included and for this reason home-made. The used material for both panels is acrylic glass. For the positioning of all sockets on the front and back panel the cutouts have been defined and a computer-aided design (CAD) has been made.

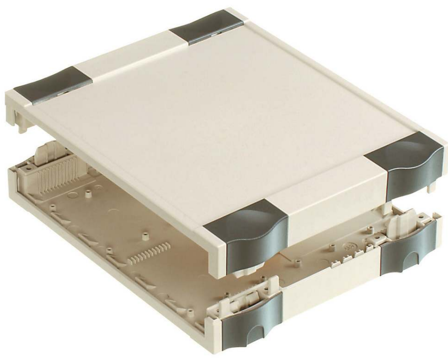


Figure 14: Raw housing of the test-bed: Bopla 52012 [13]

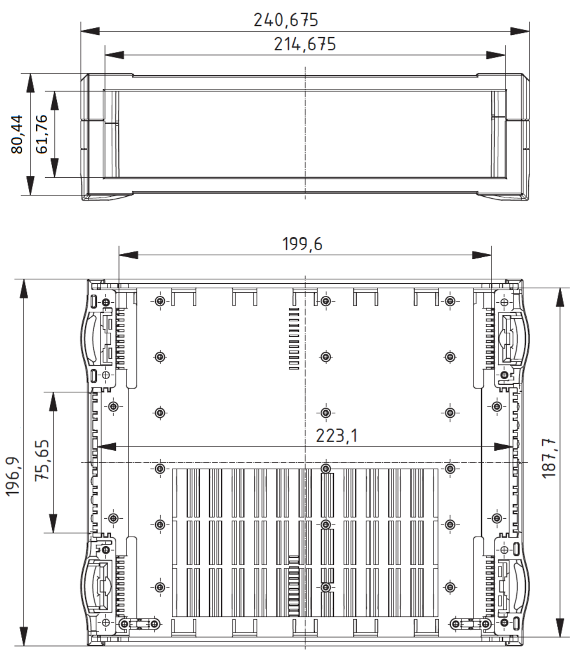



Figure 15: Dimensions of the housing: Bopla 52012 [13]

**Circuit board of the test-bed**
The PCB is created with the software *Eagle (CadSoft Computer GmbH, Pleiskirchen, Germany)* [6], where all components and circuit paths are aligned. The dimensions of the housing are imported into the CAD software and the shape of the circuit board is adjusted to it. The finished version of the PCB has been ordered by the manufacturer *Eurocircuits (Eurocircuits GmbH, Kettenhausen, Germany)* [11].

## 2.2. Software

### 2.2.1. Concept of the software

The whole software is programmed in *Python (Python Software Foundation, Delaware, USA)*, an open source development environment with modules and packages to solve a lot of problems with its big functionality. The syntax is easy and fast to learn, if the user already has programming experience in another language. Additionally a graphical user-interface (GUI) can be created, with which the user can interact more easily with the software. Python is also a script based language with the possibility of an object-oriented programming style, which is another advantage when programming a testing system. The most used packages and their roles are explained later in chapter 2.2.2.

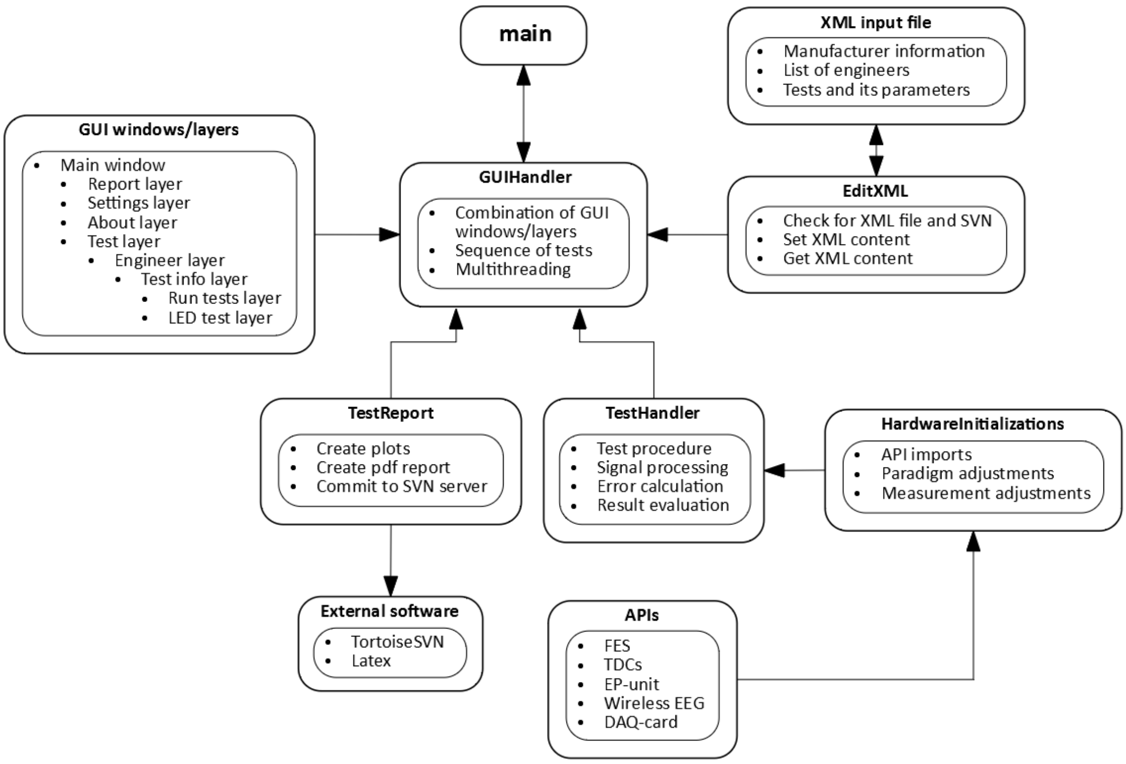



Figure 16: Block diagram of the software concept

Figure 16 shows the architecture of the software. Every block in this figure is a separate *.py file (module), which has its own purpose and is imported into another block for further use. Explanation of each block:

- **main**
  The program always starts and ends in the main function. It is part of the GUI-Handler file and creates an object of the main window and opens it. The class for this object is programmed in the GUIHandler module.

- **GUI windows/layers**
  This block includes the code for the whole graphical user interface. Every window/layer has its own file, which is designed in the Qt Designer software. This software is automatically installed with the PyQt5 package. After the creation with the designer software the file is saved with the ending *.ui. To create a useable *.py file, which then can be used by the program the script pyuic5.bat (also installed with pyQt5) has to be executed with the following command in the command shell:

```
C:/WinPython-64bit-3.4.4.6Qt5/python-3.4.4.amd64/Scripts/pyuic5.bat -x C:/path.../filename.ui -o
    C:/path.../filename.py
```

  This command converts the *.ui into a *.py file, which is then written in the Python syntax. Each created file is imported into the GUIHandler.py where its functionality is implemented.

- **XML input file**
  An extensible markup language (XML) file is used as an input file on each start of the software. It contains information about the manufacturer, the list of engineers who are performing the tests and all parameters, such as name, limit and tolerance, of every test. The advantage of such an input file is that all information including the test parameters can easily be changed without modifying anything in the Python code.

- **EditXML**
  This *.py file is imported into the GUIHandler and is used to get and set all contents from the XML input file. The data is received in form of a dictionary, which is a special data type in Python. It also contains a function, which checks the availability of the XML file, and additionally the path to the folder for the subversion (SVN) repository at each startup of the software.

- **APIs**
  The already wrapped APIs are imported into the HardwareInitializations file. They contain classes and functions to be able to communicate with the corresponding hardware device via its driver. The explanation of how to make a Python wrapper for an existing C-API is shown in chapter 2.2.2.1.

- **HardwareInitializations**
  This block is used to initialize all hardware components. It has a lot of functions, which contain measurement paradigms for each test and therefore for each hardware device, using its API. Furthermore it adjusts the input range, sample rate, recording time, used input and output pins etc. of the DAQ-card for the corresponding measurement. To be able to do this, all wrapped API files are imported into this one.

- **TestHandler**
  The TestHandler is used to adjust every test procedure and get all measured data by using the functions of the HardwareInitializations file. Therefore, the initialization file is imported into the TestHandler. Further on, the measured data is analyzed by using signal processing and error calculation. Last the test result of each single test is evaluated and returned to the GUIHandler, including the measured and processed data.

- **TestReport**
  This class is used to create all plots and the test report itself. Additionally it is used to calculate all important values, such as, for example, the mean, maximum, minimum and the standard deviation, which are needed for the report. It is imported into the GUIHandler, where it receives the whole data set. The finalized report is a latex *.tex document, which is converted with the latex command *pdflatex* to a *.pdf file. It contains a summarization page and additionally an appendix with information, such as, detailed results and plots of all executed tests. At the end, the report file is automatically uploaded to a SVN directory.

  Python code to create a *.pdf document and open it:

```
import os
#os->command shell
os.system("pdflatex " +self.texfileName+".tex")
#start the file
os.startfile(self.texfileName+".pdf")
```

Python function for the SVN upload:

```
import os
def commitToSVN(self,svnPath):
    #Copy the file to a destination. (source, destination)
    copyfile(self.texfileName+".pdf",svnPath+"\\"+self.texfileName+".pdf")
    #Open the tortoise commit window
    os.system("tortoiseproc /command:commit /path:"+svnPath)
```

- **External software**
  MiKTeX[4]: This additional software package is used to convert a *.tex document to a *.pdf document. It contains the whole Latex syntax, including the used Latex packages.
  TortoiseSVN[5]: This SVN client is used to be able to upload all test reports to a SVN repository.

- **GUIHandler**
  The GUIHandler block combines all GUI elements to one. The whole functionality of the GUI is programmed in it and also the sequence of the tests is defined by this class. The main window has five buttons, which each have a different functionality. If the user presses one of these buttons another window appears, or in case of pressing the exit button, the program closes. The report layer shows all test reports, which can also be displayed by pressing another button. In the setting's layer the whole content of the XML file is shown and can be changed. These changes are also automatically made in the XML file. Therefore, the functions in the EditXML are used. The about layer gives information about the software itself and the test layer starts the test after choosing the number of tests and typing in some additional information, such as the name of the testing engineer, the serial number and an additional comment, into the engineer and test info layer. The test info layer finally summarizes the typed- in information and the user can press the start test button. When the test is started the run test layer and if chosen the LED test layer appear and the testing scenario starts. Therefore, an object of TestHandler is created with which the functions of this file are accessed and the test paradigms and measurements for each test are defined and processed one after another. Additionally most of the tests need multithreading, or else the GUI will freeze during the testing scenario. Furthermore, some tests will not be able to perform without multithreading.

[4] https://miktex.org/download
[5] https://tortoisesvn.net/downloads.de.html

Example for multithreading:

1. Thread (main-thread): Rendering the GUI
2. Thread: Specifies the sequence of the tests and is used for the measurement with the DAQ-card
3. Thread: Needed to initialize a paradigm of the hardware
4. Thread: Needed for another paradigm of the hardware
5. Thread: ....

Multithreading was implemented using the QThread class, which is part of the PyQt5 package. The class for it is programmed in the GUIHandler file. At the end of the testing scenario an object of the TestReport class is created. The complete test data is copied into this object and is used to create the final test report.

### 2.2.2. Packages

This chapter explains the most important packages of the software. To install a package using pip[6], e.g. PyDAQmx, the following command in the command shell is used:

```
pip install PyDAQmx
```

To be able to do this installation this way, the path to the corresponding Python version has to be saved into the environmental variables of the operating system. This path saving is done in the installation of Python, when checking the corresponding box, automatically.

#### 2.2.2.1. Python wrappers for C-API

The python wrappers for the individual C-APIs are automatically created with ctypesgen [20], which is a script and runs in the Python 2 interpreter. It uses the cTypes [12] package, which allows calling functions in dynamic link library (DLL) files and therefore provides C compatible data types. Additionally a C-compiler, such as GCC[7] is needed. Since the software for this thesis is written in Python 3, the Python 2 interpreter[8] needs to be installed to use this script. The wrapped API is then created with the following steps:

[6] https://pypi.python.org/pypi/pip
[7] http://mingw-w64.org/doku.php/download
[8] https://www.python.org/download/releases/2.7.8/

1. Copy the whole ctypesgen script and also the C-API including the header and DLL files into the folder where Python 2 has been installed.

2. Open the windows command shell and go to the location where Python 2 is installed by using the *cd* command.

3. Execute the following line in the command shell

```
python.exe ctypesgen.py --cpp="C:/mingw-w64/x86_64-6.3.0-win32-seh-rt_v5-rev1/mingw64/bin/gcc.exe
    -E" -l DLL_filename -o output_filename.py header_filename.h
```

   where *python.exe* is the executable Python 2 compiler, *ctypesgen.py* the used script, $--cpp=$ is the flag to call a compiler with the path to the used C-compiler, $DLL\_filename$ is the filename of the APIs DLL, $output\_filename.py$ is the wrapped Python API file, which is used for the software in this thesis, and $header\_filename.h$ is the name of the header file from the C-API.

4. After that the wrapped Python API file is created, which is programmed in Python 2. To be able to use this file in Python 3 the script $2to3.py$, which is installed automatically when installing Python 3, is needed. It converts the Python 2 into a Python 3 syntax. Therefore, use the command shell and go to the location where Python 3 is installed by using *cd* again and then use the following command:

```
python.exe C:/Python36/Tools/scripts/2to3.py -w "C:/path.../output_filename.py"
```

   where *python.exe* is the Python 3 compiler followed by the path to the script $2to3.py$ and the path to the wrapped Python 2 API file.

Now the conversion of the C-API is complete and the wrapper can be used in the Python software.

#### 2.2.2.2. PyDAQmx

This package is installed, in order to communicate with the NI-DAQ-card and use its API functions. It provides a full interface, using ctypes, between Python and the NIDAQmx driver. Therefore, the driver, provided by *National Instruments*, needs to be installed additionally and this package works on Windows and Linux. More details and examples of how to use this package are described in the documentation. [3]

#### 2.2.2.3. PyQt5

PyQt5 [7] is the package with which the whole GUI is created. Basically it is a package which wraps the C++ Qt library and therefore can be used in Python. Additionally the whole multithreading is programmed with the QThread class of this package. PyQt5 has a very large functionality, but just the basics of it are used in this thesis software. To get more familiar with Qt take a look at the C++ documentation of it [5].

#### 2.2.2.4. lxml

This toolkit is used to be able to receive and edit all information which is saved in the XML input file.
For instance to receive all engineer names the following code is used:
XML code:

```xml
</Tables>
    <Engineer>
        <TestEngineer Name="Engineer 1" />
        <TestEngineer Name="Engineer 2" />
        <TestEngineer Name="Engineer 3" />
    </Engineer>
<Tables>
```

Python code:

```python
from lxml import etree as et
#raw path of xml file
path = r'C:\ path\ ...\ config.xml'
#empty list of engineers
xml_engineer = []
#read file of a path
file = et.parse(path)
#and use the data
root = file.getroot()
#for loop used to iterate through the file
for i in root.iter('TestEngineer'):
    #Get Names of TestEngineer tag; save them into the list one after another
    xml_engineer.append(i.get('Name'))
```

The result of this code is that all engineer names, such as Engineer 1, Engineer 2 and Engineer 3 are saved into the list variable xml_engineer. More examples and details about this package are described in the documentation [25].

#### 2.2.2.5. NumPy

The NumPy package is used in this thesis for the signal processing and all kinds of calculations. With it, the handling of multi-dimensional matrices and arrays is a lot easier. In addition it has a large collection of functions for high-level-mathematical calculations, such as the discrete Fourier transformation, functions for probability, arithmetic, trigonometric calculations and many more. Furthermore, it is used to save all the measured data of each test local on the computer. In case of a software error this log can be used to specify and solve the error much quicker. For more information take a look at the documentation [4].

#### 2.2.2.6. WMI

This package is used to get all device information of the connected USB devices with the computer. Furthermore, it would be possible to retrieve any kind of information about a computer system.
Example code of printing all devices which are connected via USB:

```python
import wmi
#Create object
c = wmi.WMI()
#Just USB devices
wql = "Select * From Win32_USBControllerDevice"
#for loop to print all connected USB device information
for item in c.query(wql):
    #Name of the device
    print(item.Dependent.Caption)
    #Information, such as PID and VID
    print(item.Dependent.DeviceID)
    #Connection status
    print(item.Dependent.Status)
    print("---------------------------------")
```

WMI is described in detail in the documentation [14].

#### 2.2.2.7. matplotlib

Matplotlib is a package, which makes it possible to make two-dimensional plots of arrays. It emulates the appearance and also the syntax of the plots from MATLAB, but is independent of it.
Example Python code for a plot:

```python
import matplotlib.pyplot as plt
#Create figure
plt.figure()
#Create plot, where x is e.g. the timevector and y the data
plt.plot(x,y)
#Create title, and labels for the x and y axis
plt.title("Title of the plot")
plt.xlabel("Time [sec]")
plt.ylabel("Amplitude [\%]")
#Save and show the figure
plt.savefig("Figure1.png")
plt.show()
```

For further information and to make different plots, such as histograms, error plots, etc., take a look at the documentation [21].

#### 2.2.2.8. PyInstaller

This package is used to create an executable file *.exe from the existing programmed *.py files. It bundles the whole Python application, all used dependencies and libraries into a single file. This has the advantage that no software items which are used to program this testing software have to be installed on the testing computer to actually use the finished program. The *.exe file is created with the following command in the command prompt:

```
pyinstaller --onefile --distpath C:/path.../location/ C:/path.../main.py
```

where $--onefile$ creates a single executable file, $--$ *distpath location* describes where to save the generated output file and *main.py* is the main program file of the application. More details about PyInstaller are provided in its documentation [10].

### 2.2.3. Implementation of the testing software

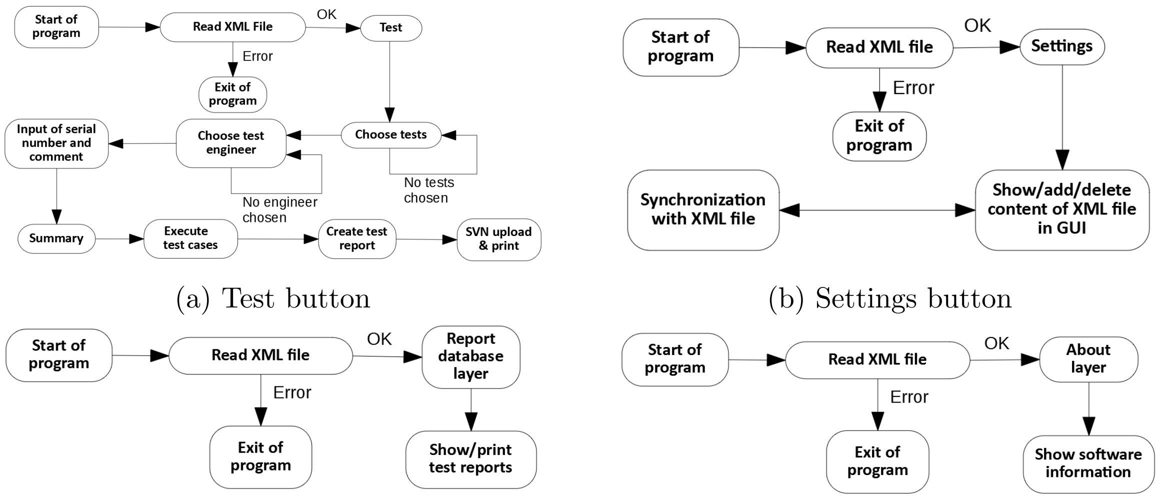
#### 2.2.3.1. The test program

By execution of the generated *.exe file the testing software starts with the main function of the GUIHandler and the following tasks are performed:

- Check if the XML file is available and not defective.
- Check if the path to the SVN repository is available.
- Open the main window of the GUI. The user has now the option to click one of five buttons.
- By clicking the *test* button, the sub-window for the testing scenario is shown.
  - Depending on the information, given by the XML file, a list of single tests is created. The user can now choose which test(s) should be executed.
  - Selection of the test engineer and input of additional information, such as the serial number of the stimulation system. The names of the engineers are also read from the XML file.
  - Execution of the test(s).
  - Signal processing, error calculation and analysis of the measured data.
  - Creation of the test report.
  - This test report is then saved into the SVN folder, followed by an automatic SVN upload, if the user agrees to it.
  - Exit from the program.
- By clicking on the *settings* button, the sub-window for the settings is shown.
  - Depending on the information of the XML file the window builds up.
  - The user can now read and change all the information, such as the test engineers and the tests and their limits.
  - If changes are made, they are synchronized with the XML file.
- By clicking on the *report database* button, the sub-window for the reports is shown.
  - Then a list of all test reports, which are saved into the SVN folder, is shown.
  - The user can now choose a test report in the list, inspect and also print it.

- By clicking on the *about* button, the sub-window, shows information about the software and the company
- By clicking on the *exit* button, the program closes.

Figure17 explains the sequence of each GUI scenario, when clicking the corresponding button, graphically.




(a) Test button

(b) Settings button

(c) Report database button

(d) About button

Figure 17: Scenario of each button click

#### 2.2.3.2. The tests

This chapter describes each test in more detail, including its limit values and the chosen parameters for the measurement. A test succeeds, if all appropriate values are in the given limits.
Every test has its own measurement function, which specifies its testing behavior. Therefore, most of the functions can be structured as follows:

- **Initialization and definition of the output paradigm of the hardware component**
  The software initializes the hardware and defines the output paradigm of this device using its API. The paradigm is chosen in dependency of the test specification.
- **Adjustment of the DAQ-card and measurement**
  The software sets the needed digital output and analog input pins of the DAQ-card to high. This setting switches the solid-state-relays of the test-bed PCB to the correct measurement circuit for the corresponding test. At the end of each measurement the first five per cent of the measured data is deleted, because of possible outliers at

the start of the measurement. Some Python example code of how to configure and how to measure with the DAQ-card can be found in the appendix A.1.

- **Storage of the raw data on the computer**
  In case of an error and for further analysis the data is additionally stored locally on the computer using the NumPy package.

- **Signal processing, calculation and analysis of the data with the corresponding limits for this test**
  The measured data is the recorded voltage on a resistor of a voltage divider over a specific time, the data has to be calculated into the desired test parameter. The resistors and switches of the corresponding measurement circuit and also the DAQ-card have a measurement tolerance in which the measured voltage can vary and consequently also the calculated test parameter. This tolerance change in the voltage has to be considered in the calculations. The calculated data is then compared with the limit for the test. This limit is read from the XML file and is defined by the specification document of the company. If the calculated values are within the given limit of the test it succeeds, else it fails. The used equations for the whole analysis are described in the description of the corresponding test.

- **Storage of the analyzed data on the computer**
  The analyzed data is then saved again on the computer. This has the advantage that discovering of a possible error will be easier in future.

- **Transmission of the data to the TestReport module**
  Every data set (measured and calculated data) of each single test is at the end of the test procedure saved into one variable. This variable is then sent to the TestReport module, which calculates all significant values (e.g. mean, maximum, minimum and standard deviation) and plots the data for each test. Finally the *.pdf test report is created and can be printed out.

**General error calculation of the DAQ-card:**
The absolute accuracy of the DAQ-card depends on the adjusted input range for the measurement. Table 3 shows the input range and the corresponding absolute accuracy at full scale range. This table can also be found in the specification document [16] on page 4.

The error corrected data is then calculated as in equation 6.

$$data = measuredData \pm (measuredData \cdot \frac{absoluteAccuracy}{measurementRange}) \tag{6}$$

Table 3: Absolute accuracy of the DAQ-card

| **Measurement range (V)** | **Absolute accuracy at full scale ($\mu$V)** |
|---|---|
| ±10 | 2710 |
| ±5 | 1420 |
| ±1 | 310 |
| ±0.2 | 89 |

When measuring a voltage the corresponding AI pin of the DAQ-card is switched using a solid-state relay (LCC110). This solid state relay has an on-resistance of $R_{ON}$ ~35Ω which causes a voltage drop on it. Normally this voltage drop needs to be considered in the error calculations. But in this case it can be unattended, because the analog input pin of the DAQ-card is in series to this relay and has a very high input impedance (>10GΩ in parallel with 100pF), which is in parallel with the measurement resistor. So the voltage drop on the switch is too low and not relevant for the calculations.

#### Test A - Functionality of the USB hub

In this test the software receives the serial numbers of the USB hub and its connected components and checks if they are working correctly. Therefore, all devices with a specific VID and PID number on the USB port are retrieved and the serial numbers and their connectivity status are called, by using the WMI package 2.2.2.6. If they are received it can be said that the USB hub and all connected components are working and additionally that the assembly of the hardware composition of the stimulation system is alright.

#### Test B - Functionality of the LEDs

This test is a visual test with its own GUI window. Therefore, an additional window opens and the testing engineer clicks on a button and the corresponding LED of the component will light up. If it lights up and if the LED has the correct color the user checks a box and verifies that the LED is working. This step has to be repeated for every LED of the device. The function of the software behind this button click is that the selected device is activated by using its API. The corresponding output pin of this device is set to high and so the LED begins to light up. In the end the value (true=checked, false=unchecked) of each box is saved into a variable. The test succeeds if all boxes are checked.

#### Test C - Compliance voltage of the FES

The compliance voltage of the FES PCB, which has two channels, is measured on a resistor, which is a part of a voltage divider. Because the FES device is a current source, the maximum voltage of ±60V can only be measured when the output current has an amplitude of ±60mA. Therefore, care must be taken in adjusting the values of the resistors due to the high power of the system. To measure this voltage an output paradigm is adjusted by using the API of the FES component. The measurement itself is done by adjusting the DAQ-card with its API. Because the measured voltage is just a part of the input voltage it is calculated using equation 7, which is the transformed voltage divider rule. The input voltage $U_{in}$ is applied on three resistors. $R_{sw} = 54\Omega \pm 20\%$ is the mean value of the ON resistors of two switches (CPC2030N and LCC110), which are connected to each other in series, $R_1 = 840\Omega \pm 1\%$ is the series connection of three resistors and $R_2 = 75\Omega \pm 1\%$ is the resistor on which the DAQ-card measures the voltage $U_2$.

$$U_{in} = U_2 \cdot \frac{R_{sw} + R_1 + R_2}{R_2} \tag{7}$$

**Configuration of the output paradigm:**

- Amplitude: ±60mA
- Phase duration: 100$\mu$s
- Inter-phase duration: 100$\mu$s
- Number of pulses: 10 pulses
- Frequency: 10Hz

**Configuration of the DAQ-card:**

- Sample rate: 200.000 samples/s
- Measurement duration: 3 seconds

**Limit:**

- Permitted range: 60V±10%

**Calculation of the measurement error:**
For the error calculation the general error of the DAQ card and additionally the minimal and maximal possible value of the voltage, caused by the tolerances of the resistors, are calculated. Equation 8 is the calculation for the minimal and equation 9 for the maximal value, where $R_{x_{err}}$ is the tolerance of the resistor x and $U_{2_{min}}$ and $U_{2_{max}}$ are the minimal

and maximal values of the measured output voltage after the error calculation of the DAQ-card.

$$U_{in_{min}} = U_{2_{min}} \cdot \frac{(R_{sw} - \frac{R_{sw} * R_{sw_{err}}}{100}) + (R_1 - \frac{R_1 * R_{1_{err}}}{100}) + (R_2 + \frac{R_2 * R_{2_{err}}}{100})}{R_2 + \frac{R_2 * R_{2_{err}}}{100}} \tag{8}$$

$$U_{in_{max}} = U_{2_{max}} \cdot \frac{(R_{sw} + \frac{R_{sw} * R_{sw_{err}}}{100}) + (R_1 + \frac{R_1 * R_{1_{err}}}{100}) + (R_2 - \frac{R_2 * R_{2_{err}}}{100})}{R_2 - \frac{R_2 * R_{2_{err}}}{100}} \tag{9}$$

**Example Python code for the error calculation function:**

```python
def errorCalcVoltageFEStDCs(self,data,daq_range,R1,R1_err,R2,R2_err):
    Rswitch = 54 #Ohm
    Rswitch_err = 20 #%

    #Calculate the +- error of the DAQ card in the specific range
    if daq_range == 10:
        daq_abs_err = 0.002710 / daq_range
    if daq_range == 5:
        daq_abs_err = 0.00142 / daq_range
    if daq_range == 1:
        daq_abs_err = 0.00031 / daq_range
    if daq_range == 0.2:
        daq_abs_err = 0.000089 / daq_range

    data_min=data-(data*daq_abs_err)
    data_max=data+(data*daq_abs_err)

    input_val=data*(Rswitch+R1+R2)/R2 #Calculation of the input voltage
    min_val=data_min*((Rswitch-(Rswitch*Rswitch_err/100))+(R1-(R1*R1_err/100))+
   (R2+(R2*R2_err/100)))/(R2+(R2*R2_err/100))
    max_val=data_max*((Rswitch+(Rswitch*Rswitch_err/100))+(R1+(R1*R1_err/100))+
   (R2-(R2*R2_err/100)))/(R2-(R2*R2_err/100))

    return input_val,min_val, max_val
```

### Test D - Stimulation current of the FES

The stimulation current of FES channel 1 and FES channel 2 is indirectly measured on the same shunt resistor as the voltage. Therefore, the current of the output paradigm is adjusted from ±1mA to ±60mA and the voltage is again measured with the DAQ-card. After the measurement the current is calculated with the transformed Ohms law 10.

$$I = \frac{U}{R} \tag{10}$$

**Configuration of the output paradigm:**

- Adjusted amplitude: ±1mA, ±10mA, ±20mA, ±30mA, ±40mA, ±50mA, ±60mA
- Phase duration: 100$\mu$s
- Inter-phase duration: 100$\mu$s
- Number of pulses: 10 pulses per adjustment
- Frequency: 10Hz

**Configuration of the DAQ-card:**

- Sample rate: 200.000 samples/s
- Measurement duration: 15 seconds

**Limit:**

- Permitted range: $X$mA±10%, where $X$ is the adjusted amplitude

**Calculation of the measurement error:**
For the error calculation the general error of the DAQ card and additionally the minimal and maximal possible value of the current, caused by the tolerance of the measurement resistor, are calculated with equations 11 and 12, where again $U_{2_{min}}$ and $U_{2_{max}}$ are the minimal and maximal values of the measured output voltage after the error calculation of the DAQ-card.

$$I_{min} = \frac{U_{2_{min}}}{R_2 + \frac{R_2 * R_{2_{err}}}{100}} \tag{11}$$

$$I_{max} = \frac{U_{2_{max}}}{R_2 - \frac{R_2 * R_{2_{err}}}{100}} \tag{12}$$

### Test E - Frequency of the FES

This test checks the frequency of the recorded voltage of both FES channels from 1Hz to 100Hz. The frequencies are adjusted after every pulse period. Therefore, the voltage is measured and recorded on the same resistor compositions as the previous tests and each frequency is calculated from the period of one pulse. Equation 13 shows how to calculate the frequency, where sampleRate is the adjusted sample rate of the DAQ-card, $\Delta samples$ is the number of samples of one period and f is the frequency.

$$f = \frac{sampleRate}{\Delta samples} \tag{13}$$

**Configuration of the output paradigm:**

- Amplitude: ±20mA
- Phase duration: 100$\mu$s
- Inter-phase duration: 100$\mu$s
- Number of pulses: 10 pulses per adjustment except 1Hz (3pulses)
- Adjusted frequency: 1Hz, 10Hz, 20Hz, 30Hz, 40Hz, 50Hz, 60Hz, 70Hz, 80Hz, 90Hz, 100Hz

**Configuration of the DAQ-card:**

- Sample rate: 200.000 samples/s
- Measurement duration: 17 seconds

**Limit:**

- Permitted range: $X$Hz±10%, where $X$ is the adjusted frequency

The expected measurement error of this test depends on the length of the period window and the adjusted sampling frequency. The error of the DAQ-card is ±1 sample, which are ±5$\mu$s at an sampling rate of 200.000 samples/s. The measurement error in the smallest window (10ms), which is the 100Hz adjustment, is ±0.05%. Because the measurement error is so low and the accuracy of the DAQ-card so high, it has been disregarded in the calculations.

### Test F - Durations of the FES

The duration test checks the duration of the phase and inter-phase of both FES channels for 10 pulses per adjustment. The durations are adjusted after every tenth pulse. Moreover, the voltage is measured on the same resistor as in the previous tests. This recorded voltage is then analyzed and the durations are calculated from the samples. Equation 14 shows how to calculate the duration, where $\Delta t$ is the phase duration, or the inter-phase duration and $\Delta samples$ is the number of samples from the beginning to the end of one high pulse (phase-duration), or between two high pulses (inter-phase duration).

$$\Delta t = \frac{\Delta samples}{sampleRate} \tag{14}$$

**Configuration of the output paradigm for the phase duration:**

- Amplitude: ±20mA
- Adjusted phase duration: 50$\mu$s, 100$\mu$s, 150$\mu$s, 200$\mu$s, 250$\mu$s, 300$\mu$s
- Inter-phase duration: 100$\mu$s
- Number of pulses: 10 pulses per adjustment
- Frequency: 10Hz

**Configuration of the output paradigm for the inter-phase duration:**

- Amplitude: ±20mA
- Phase duration: 100$\mu$s
- Adjusted inter-phase duration: 50$\mu$s, 100$\mu$s, 200$\mu$s, 300$\mu$s, 400$\mu$s, 500$\mu$s, 600$\mu$s, 700$\mu$s, 800$\mu$s, 900$\mu$s, 1000$\mu$s
- Number of pulses: 10 pulses per adjustment
- Frequency: 10Hz

**Configuration of the DAQ-card:**

- Sample rate: 200.000 samples/s
- Measurement duration: 40 seconds

**Limit:**

- Permitted range: $X\mu s \pm 5\%$ or $\pm 5\mu s$, whichever greater, where $X$ is the adjusted duration

The explanation for the expected measurement error is the same as the explanation in Test E - Frequency of the FES.

#### Test G - Compliance voltage of the tDCs

This test has been skipped, because of the unfinished hardware of the prototype. The PCB and also the firmware of this component will be similar to the FES PCB and firmware and so the software test and the error calculation will also be related to the FES test C - Compliance voltage of the FES. The composition of the voltage divider of the test-bed for the tDCs tests is: $R_{sw} = 54\Omega \pm 20\%$, $R_1 = 4.2k\Omega \pm 0.1\%$ and $R_2 = 150\Omega \pm 0.1\%$, where $R_2$ is the resistor on which the voltage is measured.

#### Test H - Stimulation current of the tDCs

This test has been skipped, because of the unfinished hardware of the prototype. The PCB and also the firmware of this component will be similar to the FES PCB and firmware and so the software test and the error calculation will also be related to the FES test D - Stimulation current of the FES.

#### Test I - Ramp up and down time of the tDCs

This test has been skipped, because of the unfinished hardware of the prototype. The PCB and also the firmware of this component will be similar to the FES PCB and firmware and so the software test and the error calculation will also be related to the FES test F - Durations of the FES.

#### Test J - Output voltage of the EP-unit

This test checks the output voltage of all 16 channels of the EP-unit. Therefore one channel after another is set to high by using the EP-units API. The voltage is then measured on a resistor of a voltage divider with the DAQ-card. The composition of each voltage divider of the test-bed for each channel and for all EP-unit tests is: $R_1 = 18\Omega \pm 1\%$ and $R_2 = 15\Omega \pm 1\%$, where $R_2$ is the resistor on which the voltage is measured with the

DAQ-card.

The calculation and also the error calculation is the same as for test C - Compliance voltage of the FES, except that the voltage divider just has two resistors $R_1$ and $R_2$.

**Configuration of the output paradigm:**

- Set one channel after another to high (pin 1-16).
- Set each first channel of each socket (pin 1, pin 6 and pin 11) to high.

**Configuration of the DAQ-card:**

- Sample rate: 1.000 samples/s
- Measurement duration: 50 seconds

**Limit:**

- Permitted range: 5V±5% per channel

#### Test K - Output current of the EP-unit

This test analyzes the same measured data as test J - Output voltage of the EP-unit. If test J has been performed, the measurement will be skipped, followed by just an analysis of the data, else the measurement will be conducted before the analysis. Therefore, the voltage is again measured on a shunt resistor and the current is calculated from the measured results.

The calculation and also the error calculation is the same as for test D - Stimulation current of the FES, except that the voltage divider just has two resistors $R_1$ and $R_2$.

**Configuration of the output paradigm:**

- Set one channel after another to high (pin 1-16).
- Set each first channel of each socket (pin 1, pin 6 and pin 11) to high.

**Configuration of the DAQ-card:**

- Sample rate: 1.000 samples/s
- Measurement duration: 50 seconds

**Limit:**

- Permitted range: 150mA±5% per channel

### Test L - Timing of the EP-unit

This test checks the time difference between the EP-unit output on every first pin of each socket and the three digital input pins of the wireless EEG. Therefore, the output voltage of the EP-unit on a resistor is measured/recorded with the DAQ-card and parallel to this measurement the wireless EEG measures the same signal with its digital input pins. Then both measurements are put over each other with a starting point. The starting point is the first high sample (when all three pins are set to high) of each measurement. Last the time differences of all following high peaks are calculated.

The calculation for this test is similar to the calculation of test F - Durations of the FES.

**Configuration of the output paradigm of the EP-unit:**

- Set all first channels of each output socket to high (pin 1, 6 and 11).
- Set the first channels of each output socket to high one after another (pin 1, followed by pin 6, followed by pin 11)
- Set all first channels of each output socket to high (pin 1, 6 and 11).

**Configuration of the DAQ-card:**

- Sample rate: 250 samples/s
- Measurement duration: 20 seconds

**Limit:**

- Permitted range: $\pm$10ms per channel

### Test M - Connection test of the Wireless EEG

The wireless EEG connection test is a basic test which verifies the functionality of this device. This test does not need to be comprehensive, because this component is already a pre-tested device. In this test the base station of the EEG connects with its headset and requests the headsets serial number by using the corresponding API. If it is received it can be said that the wireless connection works and the test succeeds.

### Test N - Audio signal check

This test has been skipped, because of the unfinished hardware of the prototype. In the future version of the software this signal will be measured and recorded directly with the DAQ-card. The measured data will be checked regarding its amplitude and if the timing of the signal equals the timing of the wireless EEG digital inputs as in test L - Timing of the EP-unit.

### Test O - Output voltage of the medical computer

The medical computer is supplied with the power supply from the stimulation system. Therefore, this test checks if this output voltage is in the corresponding limit. The measurement is again made with the DAQ-card. It measures the voltage on a resistor of a voltage divider. The composition of the voltage divider is the following: $R_1 = 10k\Omega \pm 0.1\%$ and $R_2 = 10k\Omega \pm 0.1\%$, where $R_2$ is the resistor on which the voltage is measured with the DAQ-card.

The calculation and also the error calculation is the same as for test C - Compliance voltage of the FES, except that the voltage divider just has two resistors $R_1$ and $R_2$.

**Configuration of the DAQ-card:**

- Sample rate: 1000 samples/s
- Measurement duration: 1 second

**Limit:**

- Permitted range: 19V±3%

### Test P - Vibration of the tactile stimulators

This test is used to check if the tactile stimulators are working correctly. Therefore, they are mounted above a piezo sensor and powered one after another. The piezo sensor generates an output voltage, depending on the strength of the vibrations of one powered tactile stimulator, which is then directly measured with the analog input pin of the DAQ-card. The measured voltage and also the calculated frequency is then analyzed and should be within a given limit. This given limit refers to the amplitude of a calibrated tactile stimulator.

The calculation of the frequency is similar to the one in test E - Frequency of the FES.

**Configuration of the DAQ-card:**

- Sample rate: 100.000 samples/s
- Measurement duration: 5 second (4 seconds for the calculation)

**Limit:**

- Permitted range: 70% of the amplitude in comparison to one calibrated tactile stimulator. Additionally the measured frequency has to be between 150 and 250Hz.

#### Temperature measurement

The output voltage of the temperature sensor is measured before and after each test with the DAQ-card. Equation 15 shows how the voltage is associated with the temperature as described in the data sheet [23], where T is the temperature and $U_2$ is the measured voltage. Additionally the current time when the temperature is measured is saved into a variable. Because of the saving of the time the user knows how long each test and how long the whole test procedure has taken.

$$T = \frac{U_2 - 0.4V}{0.0195V/°C} \tag{15}$$

**Configuration of the DAQ-card:**

- Sample rate: 100 samples/s
- Measurement duration: 0.5 seconds each

#### 2.2.3.3. The test report

The test report is automatically created at the end of each test scenario and saved as a *.pdf document. It contains a summary page, which tells the user the basic information about the stimulation system, the company, the testing software, the tests including its tolerances and results ("SUCCESS", "FAILED" or "SKIPPED"), the test engineer and the final test result and an additional appendix with more detailed information about the executed tests.
Content of the appendix:

- Name of the executed test

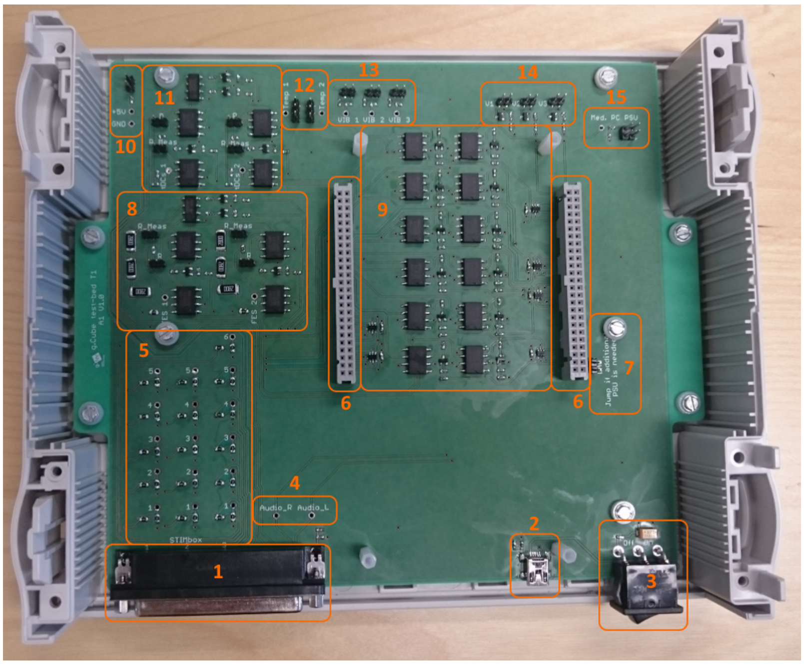
- Parameters for the measurement, such as the sampling rate and the measurement duration
- Result of the test (”SUCCESS” or ”FAILED”)
- Description of the test
- Test criteria and their test limits
- Table(s) with calculated results
- Diagram(s)

# 3. Results

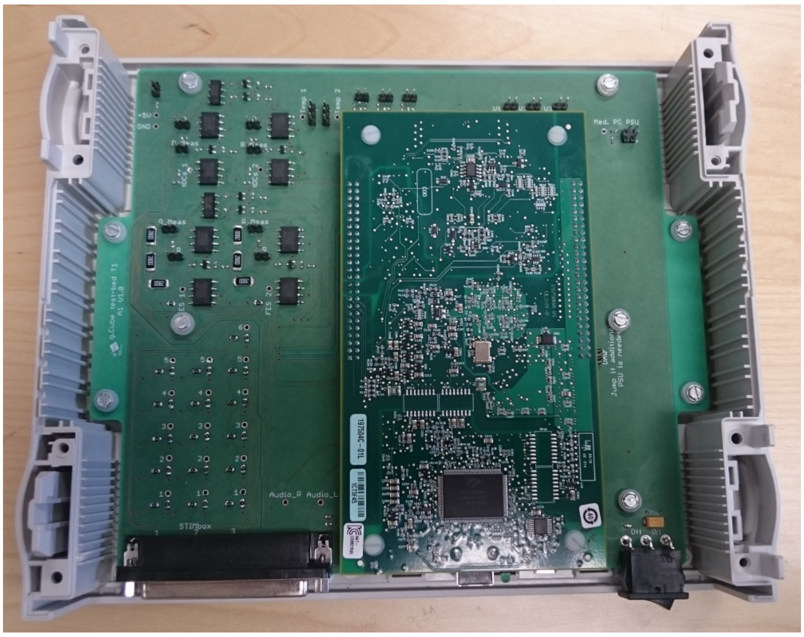
## 3.1. Hardware

### 3.1.1. The circuit board

Figure 18 shows the assembled circuit board with and without mounted DAQ-card.




(a) PCB without DAQ-card



(b) PCB with mounted DAQ-card

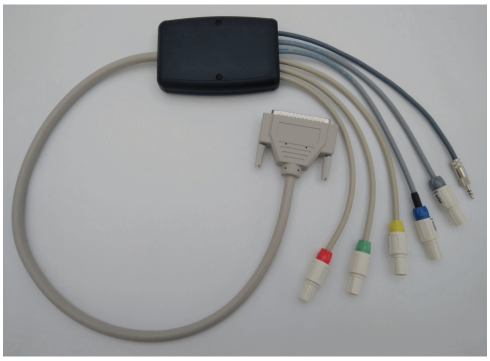
Figure 18: Circuit Board of the test-bed

The circuit board consists of four layers. The bottom level has been used for some circuit paths, top level for the components and circuit paths and the other two layers in between are used as GND plane and +5V plane.
The most important measurement circuits and components can be seen in figure 18a:

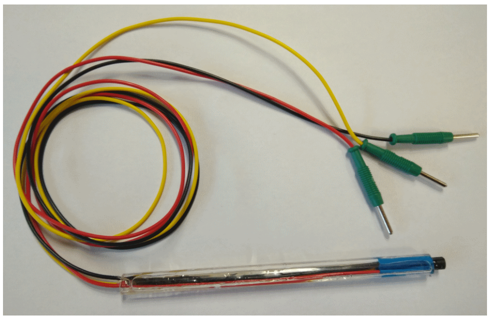
1. D-SUB 37 pin socket for the signals from the stimulation system
2. Mini-USB socket for the power supply
3. On/Off switch for the power supply
4. Test points for the audio measurement circuit
5. EP-unit measurement circuit
6. Two times IDC connectors, used to mount the DAQ-card
7. Jumper to have the possibility for an additional power supply for the DAQ-card
8. FES 1 and FES 2 measurement circuit
9. ESD protection and output relays, used to switch between the AI pins of the DAQ-card
10. Connector for the power supply LED
11. tDCs 1 and tDCs 2 measurement circuit
12. Connectors for two temperature sensors
13. Connectors for three piezo sensors
14. Connectors to supply the tactile stimulators with power
15. Medical computer power supply measurement circuit

### 3.1.2. Additional hardware equipment

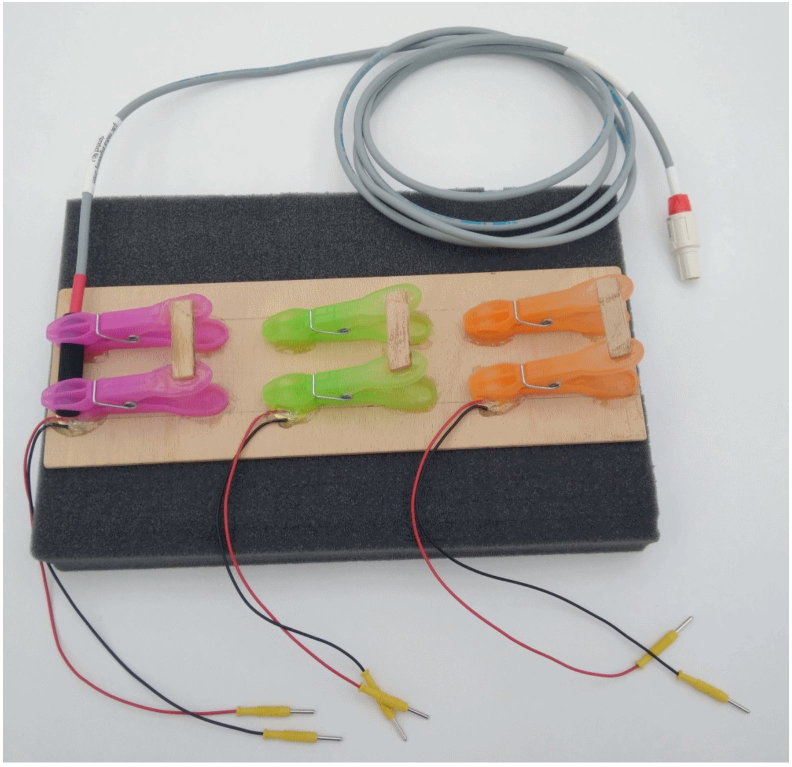
Figure 19 shows every additional hardware equipment.
The adapter cable, shown in figure 19a is used to connect the stimulation system with the testing system. It consists of a stereo audio plug and five medical plugs (gray -> tDCs 1 and tDCs 2, blue -> FES 1 and FES 2, yellow -> EP-unit 3, green -> EP-unit 2, red -> EP-unit 1) on one side and one D-SUB 37 pin plug on the other. The pinout of this cable is described in chapter 2.1.2.7. Figure 19b shows the temperature sensor and figure 19c shows the device to mount the tactile stimulators, where one is shown in the image, and measures its vibrations with the piezo sensor underneath the clip.



(a) Adapter cable to connect the stimulation system with the test-bed



(b) Temperature sensor



(c) Hardware to test the tactile stimulators

Figure 19: Additional hardware equipment

### 3.1.3. Housing & label

With the CAD drawings of the front and back panel the labels are created and shown in figure 20.

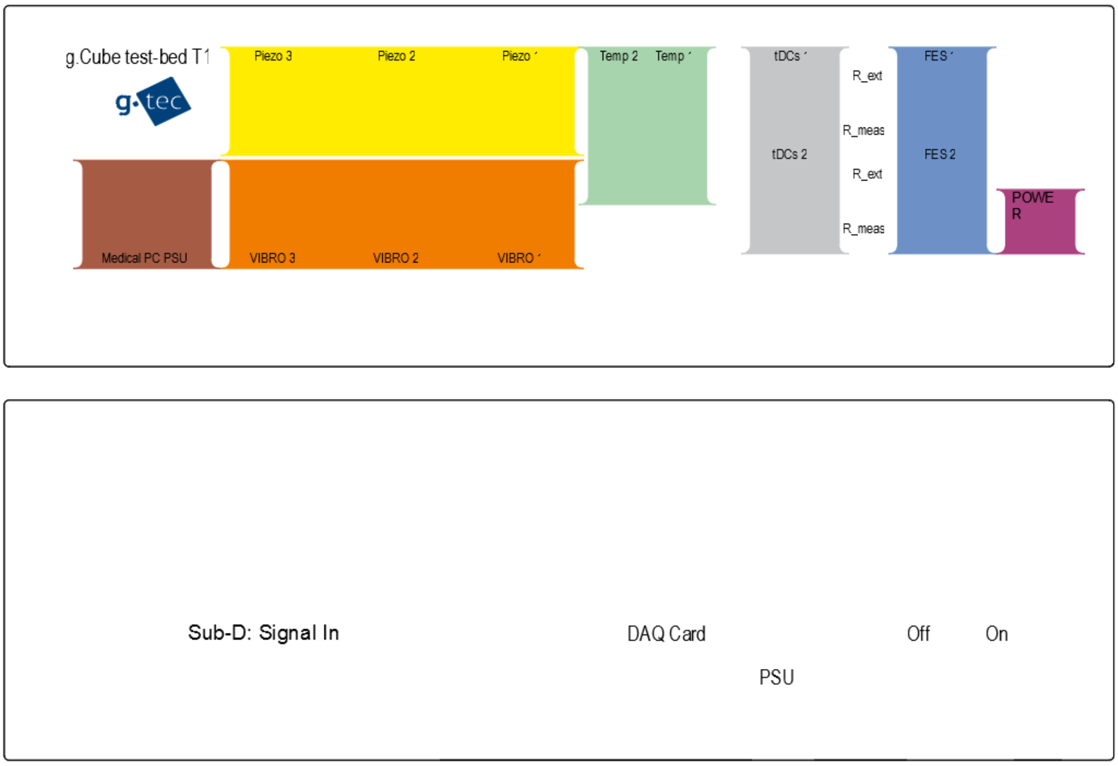
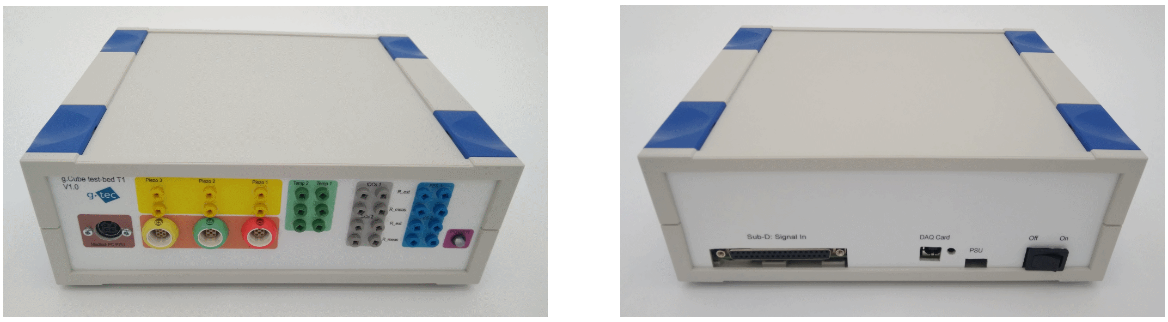



Figure 20: Labels for the front and back panel of the housing

Figure 21 shows the front and the back side of the finished testing system after the creation of the front and back panel, gluing of the label, insertion of the circuit board into the housing and the connection of all sockets.



(a) Front side of the test-bed (b) Back side of the test-bed

Figure 21: Test-bed with labels and build in sockets

Figure 22 shows the setup of all hardware items during a test scenario. On the left side of the image there is the multifunctional stimulation system with the EEG cap and the equipment for the tactile stimulators on top of it and on the right side the test-bed.

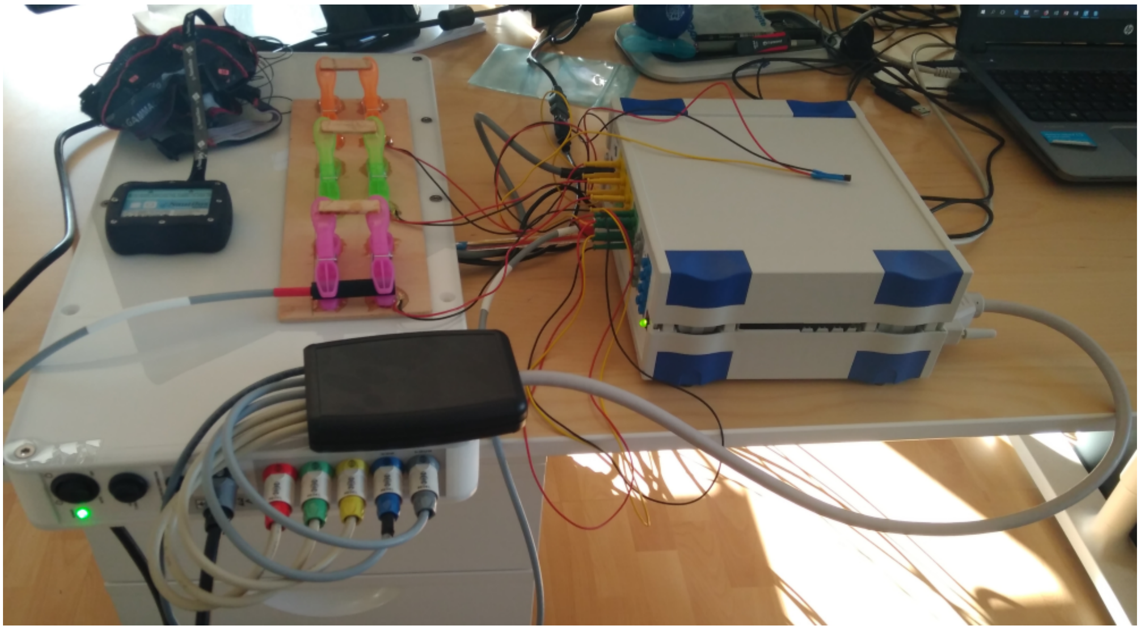
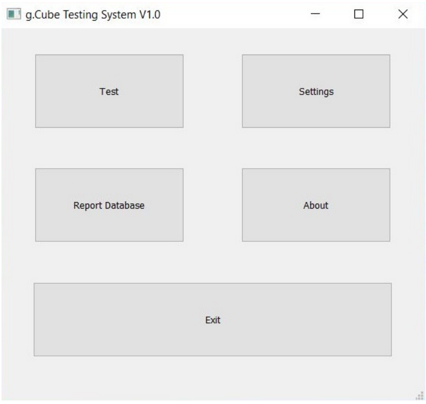


Figure 22: Stimulation system and test-bed during a test

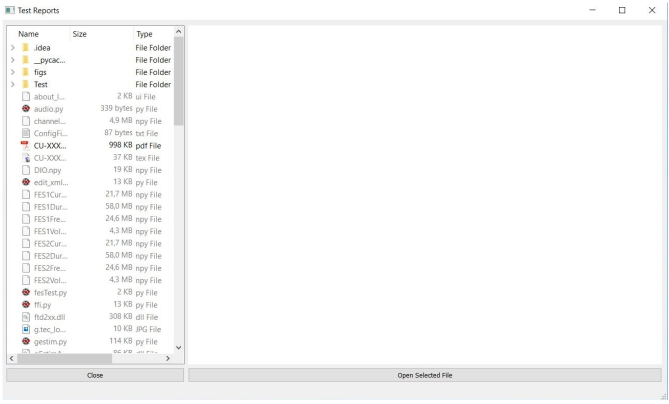
## 3.2. Software

### 3.2.1. The GUI (graphical user interface)

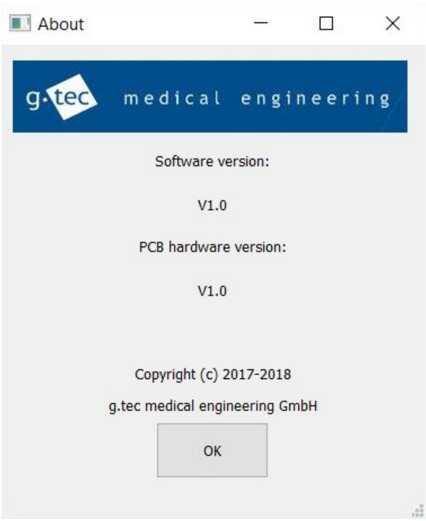
The program starts with the main function, which is connected to the GUIHandler. On startup the GUIHandler checks if the path to the input file, written in extensible markup file (XML), and also if the path to the subversion (SVN) repository is available. If not, the user is asked to type them in. These locations are saved into a text file and its availability is checked on each start of the program. If the locations are available and correct the main window of the GUI opens as shown in figure 23a. The user can now choose between the buttons and the corresponding sub-window appears (figures 23b, 23c, 23d or 24a).



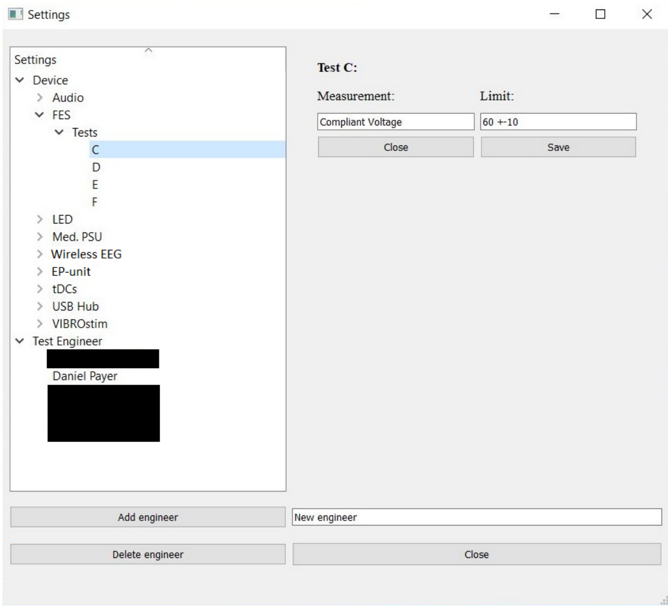

(a) Main window of the GUI




(b) Report database sub-window of the GUI




(c) About sub-window of the GUI

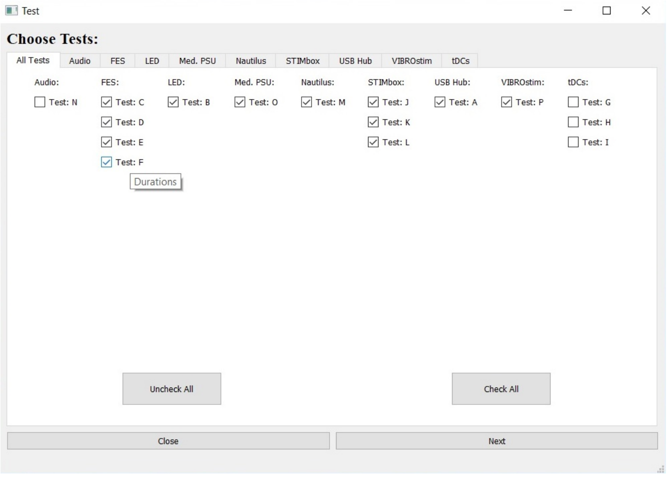



(d) Settings sub-window of the GUI

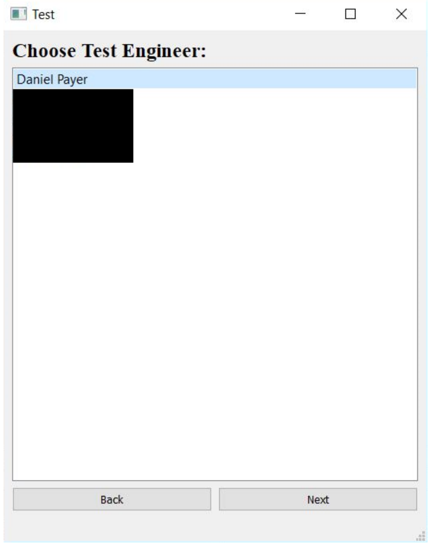
Figure 23: Main window and sub-windows of the GUI

The report database sub-window shows all files in a specific directory. Only *.pdf files can be selected and shown if the button is clicked. The about sub-window displays information about the software and hardware. In the settings sub-window it is possible to add and delete test engineers and additionally change the parameters of a test. These changes are also automatically done in the XML input file.

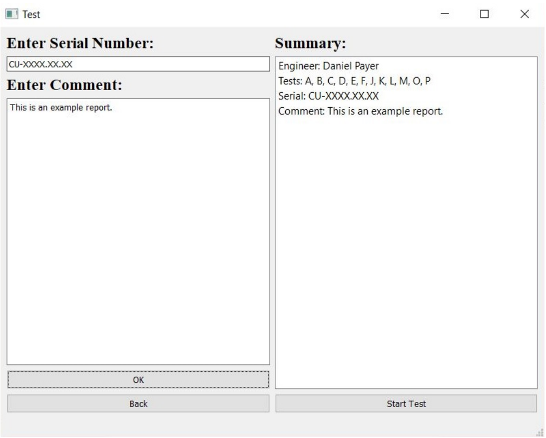
Figure 24 shows the sub-windows in ordered sequence when starting a testing scenario by clicking the test button of the main window.



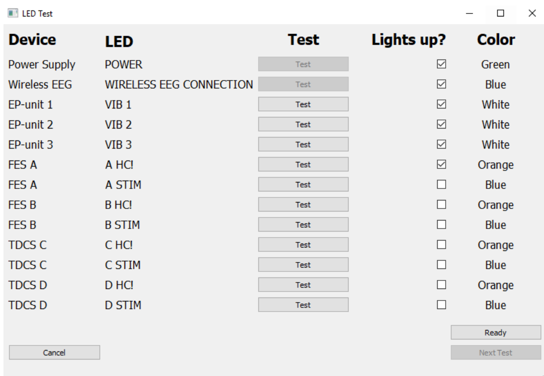

(a) Test sub-window of the GUI



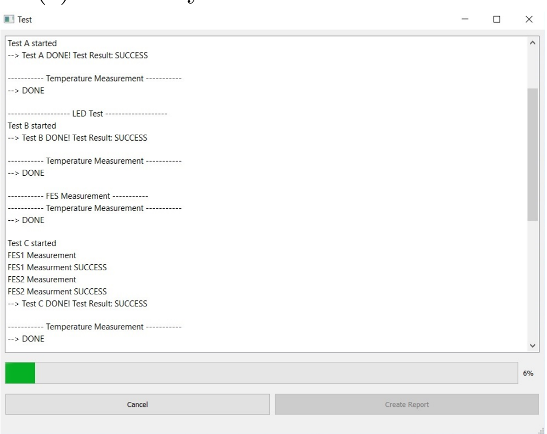

(b) Engineer sub-window of the GUI



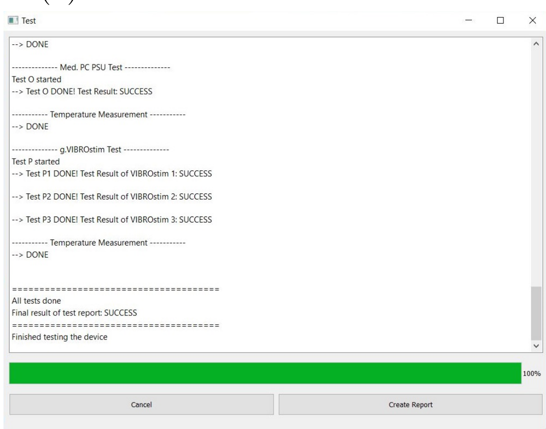

(c) Summary sub-window of the GUI




(d) LED test sub-window of the GUI




(e) Exectue test sub-window of the GUI




(f) Exectue test sub-window of the GUI

Figure 24: Sub-window test and its sub-windows

The user can now choose which tests should be executed (figure 24a). When moving over one test with the mouse a hint appears, which tells you what kind of test it is. Afterwards the test engineer is chosen (figure 24b), followed by typing in the serial number of the stimulation system and an additional comment (figure 24c). By clicking the OK button the summary appears. If all information is correct the user can now start the test procedure. After test A (USB hub test) the LED test sub-window appears (figure 24d) in which the engineer verifies that all LEDs of the stimulation system are working. All other tests of the scenario are fully automatic. Figure 24e shows which test is executed at the moment and also tells you the results of each test. At the end of all tests the create report button is enabled as in figure 24f and the test report is created automatically when clicking on this button. Finally the test report is shown and can be uploaded to the SVN repository.

### 3.2.2. Test results

This chapter shows some results of the complete test report. The summary page and the complete report are added to the appendix A.2. The tables and diagrams shown here are minor extractions of the complete ones, but are sufficient enough to explain the results.

#### 3.2.2.1. Test A - Functionality of the USB hub

Table 4 shows the received serial numbers of all components, which are connected to the USB hub, and if they are working properly.

Table 4: Result of test A

| Hardware Item | Serial Number | Recognized |
|---|---|---|
| USB Hub | CU-2017.06.02 | OK |
| FES | FR-2017.05.03 | OK |
| tDCs | - | - |
| EP-unit | CS-2017.01.01 | OK |
| Wireless EEG | NB-2017.09.50 | OK |

#### 3.2.2.2. Test B - Functionality of the LEDs

Table 5 displays an extraction of the complete table, which shows if a LED is lighting and has the correct color.

Table 5: Result of test B

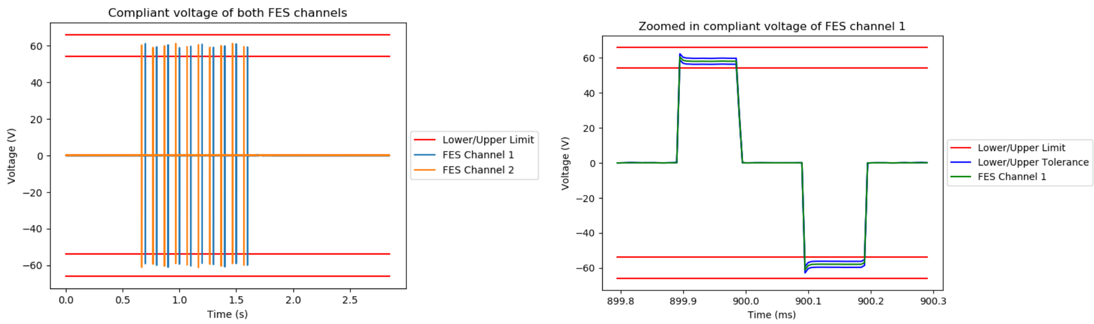
| LED | Color | Lighting |
|---|---|---|
| FES A high current | Orange | YES |
| FES A stimulate | Blue | YES |

#### 3.2.2.3. Test C - Compliance voltage of the FES

Table 6 and figure 25 show a part of the measured results of the voltage test. The red horizontal lines in the figure display the minimal and maximal limit. The peaks of the measured data have to be within those red lines. Figure 25b shows the zoomed- in measurement of one pulse. The green curve is the actually measured data and the blue curves are the lower and upper measurement tolerances, which are calculated from the resistor and DAQ-card tolerances.

Table 6: Result of test C

| Parameter | FES channel 1 | FES channel 2 |
|---|---|---|
| Maximal positive voltage (V) | 58.21 | 58.36 |
| Maximal negative voltage (V) | -58.21 | -58.34 |
| Minimal positive voltage (V) | 57.92 | 58.08 |
| Minimal negative voltage (V) | -57.93 | -58.08 |
| Mean of positive voltage (V) | 58.01 | 58.15 |
| Mean of negative voltage (V) | -57.99 | -58.14 |
| Standard deviation of positive voltage (V) | 0.05 | 0.05 |
| Standard deviation of negative voltage (V) | 0.04 | 0.04 |
| Mean of amplitude (V) | ±58.0 | ±58.14 |
| **Test result** | SUCCESS | SUCCESS |




(a) Measurement of the voltage of both FES channels

(b) Zoomed in measurement of the voltage of FES channel 1

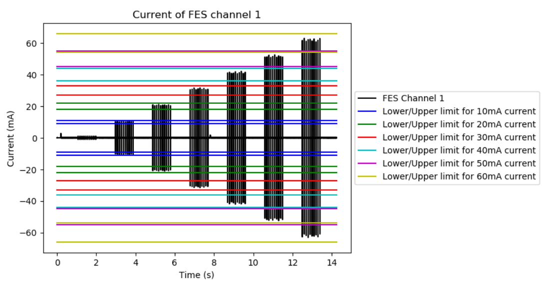
Figure 25: Measured voltage of FES

#### 3.2.2.4. Test D - Stimulation current of the FES

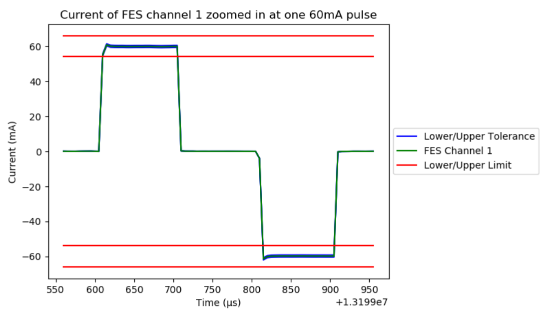
Table 7 and figure 26 show a part of the measured results of the current test. Figure 26a shows the whole measurement of every adjusted current (±1mA, ±10mA, ±20mA, ±30mA, ±40mA, ±50mA, ±60mA, with ten pulses each) of FES channel 1. The horizontal lines in the figure display the minimal and maximal limits. The peaks of the measured data have to be within those horizontal lines. Figure 26b shows the zoomed- in measurement of one pulse with an amplitude of ±60mA. The green curve is the actually measured data and the blue curves are the lower and upper measurement tolerances, which are calculated from the resistor and DAQ-card tolerances.

Table 7: Result of test D

| **Adjusted Current (mA)** | **Parameter** | **FES channel 1** | **FES channel 2** |
|---|---|---|---|
| 60mA | Maximal positive current (mA) | 60.18 | 60.34 |
| | Maximal negative current (mA) | -60.19 | -60.35 |
| | Minimal positive current (mA) | 59.73 | 59.89 |
| | Minimal negative current (mA) | -59.78 | -59.93 |
| | Mean of positive current (mA) | 59.86 | 60.02 |
| | Mean of negative current (mA) | -59.86 | -60.0 |
| | Standard deviation of positive current (mA) | 0.07 | 0.07 |
| | Standard deviation of negative current (mA) | 0.06 | 0.06 |
| | Mean of amplitude (mA) | ±59.86 | ±60.01 |
| | **Test result** | SUCCESS | SUCCESS |




(a) Measurement of the current of FES channel 1




(b) Zoomed in measurement of the current (60mA) of FES channel 1

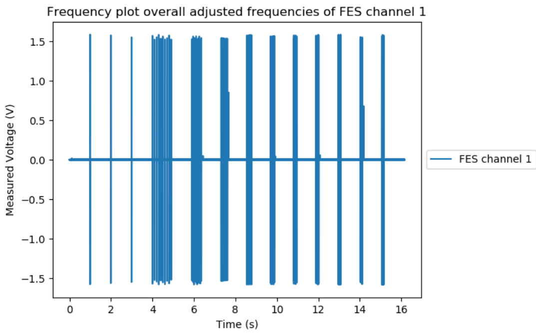
Figure 26: Measured current of FES channel 1

#### 3.2.2.5. Test E - Frequency of the FES

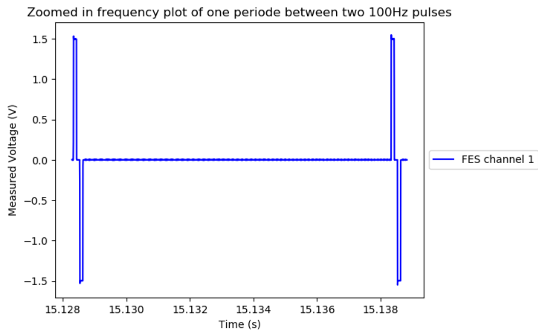
Table 8 and figure 27 show a part of the measured results of the frequency test. Figure 27a shows the whole recorded data of FES channel 1. The first three pulses on the left side have a frequency of 1Hz, followed by the other frequency (10Hz, 20Hz, 30Hz, 40Hz, 50Hz, 60Hz, 70Hz, 80Hz, 90Hz, and 100Hz) pulses (10 pulses each). Figure 27b displays the zoomed -in measurement of two 100Hz pulses.

Table 8: Result of test E

| Adjusted frequency | Channel | Maximum (Hz) | Minimum (Hz) | Mean (Hz) | Standard deviation (Hz) | Test result |
|---|---|---|---|---|---|---|
| 100Hz | FES 1 | 100.0 | 99.950 | 99.983 | 0.023 | SUCCESS |
| | FES 2 | 100.0 | 99.950 | 99.983 | 0.023 | SUCCESS |




(a) Measurement of the frequencies (left side 1 Hz - right side 100Hz) of FES channel 1




(b) Zoomed in measurement of two 100Hz pulses of FES channel 1

Figure 27: Measured frequency of FES channel 1

#### 3.2.2.6. Test F - Durations of the FES

Tables 9, 10 and figure 28b show the extracted results for the adjusted phase duration of 100$\mu$s. Table 11 and figure 28c show the results for the adjusted inter-phase duration of 500$\mu$s. Figure 28a displays the whole recording of the measurement for FES channel 1. The recording for every adjusted phase duration happens before the pause (~11-14s in the figure) and the inter-phase duration recordings afterwards.

Table 9: Result of the phase duration of test F (positive pulse)

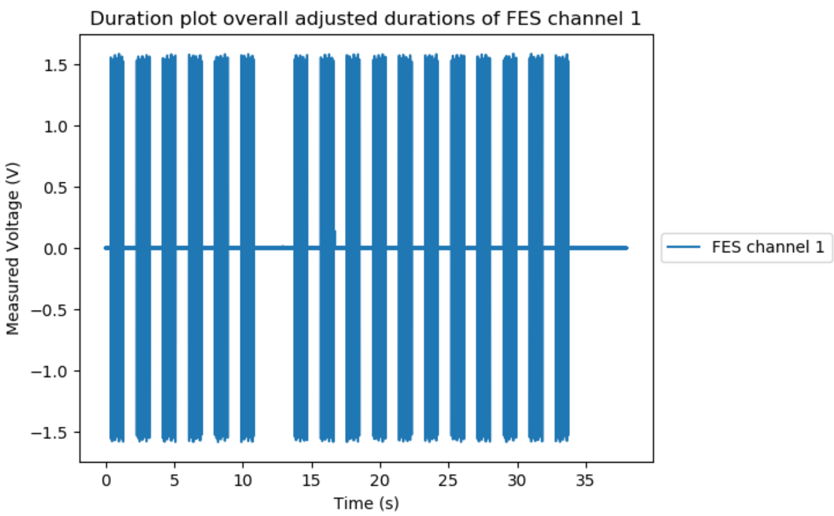
| Positive phase duration | | | | | | |
|---|---|---|---|---|---|---|
| **Adjusted phase duration** | **Channel** | **Maximum ($\mu$s)** | **Minimum ($\mu$s)** | **Mean ($\mu$s)** | **Standard deviation ($\mu$s)** | **Test result** |
| 100$\mu$s | FES 1 | 100.0 | 95.0 | 96.5 | 2.291 | SUCCESS |
| | FES 2 | 100.0 | 95.0 | 97.0 | 2.44949 | SUCCESS |

Table 10: Result of the phase duration of test F (negative pulse)

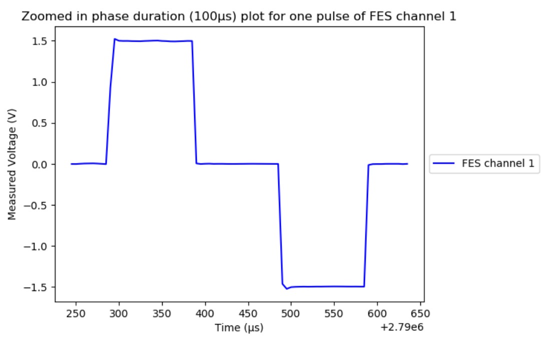
| **Positive phase duration** | | | | | | |
|---|---|---|---|---|---|---|
| **Adjusted phase duration** | **Channel** | **Maximum ($\mu$s)** | **Minimum ($\mu$s)** | **Mean ($\mu$s)** | **Standard deviation ($\mu$s)** | **Test result** |
| 100$\mu$s | FES 1 | 100.0 | 95.0 | 98.5 | 2.291 | SUCCESS |
| | FES 2 | 100.0 | 95.0 | 99.0 | 2.0 | SUCCESS |

Table 11: Result of the inter-phase duration of test F

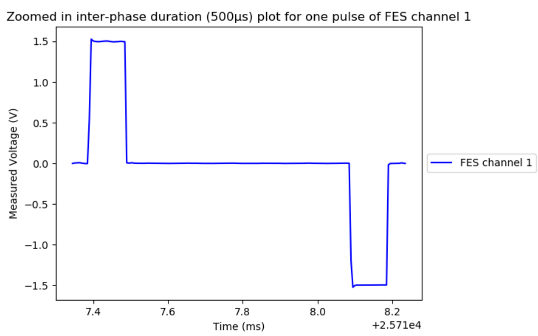
| **Inter-phase duration** | | | | | | |
|---|---|---|---|---|---|---|
| **Adjusted inter-phase duration** | **Channel** | **Maximum ($\mu$s)** | **Minimum ($\mu$s)** | **Mean ($\mu$s)** | **Standard deviation ($\mu$s)** | **Test result** |
| 500$\mu$s | FES 1 | 505.0 | 500.0 | 502.5 | 2.5 | SUCCESS |
| | FES 2 | 505.0 | 500.0 | 503.0 | 2.449 | SUCCESS |




(a) Measurement of the durations (phase duration: before pause; inter-phase duration: after pause) of FES channel 1




(b) Zoomed in measurement of the phase duration (100$\mu$s) of FES channel 1




(c) Zoomed in measurement of the inter-phase duration (500$\mu$s) of FES channel 1

Figure 28: Measured durations of FES channel 1

#### 3.2.2.7. Test J - Output voltage of the EP-unit

Table 12 and figure 29 show a part of the measured results of the voltage test for the EP-unit. The horizontal red lines in the figure display the minimal and maximal limits. The peak of each measured channel has to be within those horizontal lines. The last peak in the figure is the measurement of three powered channels (1, 6 and 11) at the same time. This peak will not be evaluated in this version of the software.

Table 12: Result of test J

| Channel | Maximum of voltage (V) | Minimum of voltage (V) | Mean of voltage (V) | Standard deviation of voltage (V) | Result |
|---|---|---|---|---|---|
| EP-unit 1 Pin 1 | 4.778 | 4.767 | 4.777 | 0.001 | SUCCESS |
| EP-unit 2 Pin 1 | 4.811 | 4.76 | 4.776 | 0.005 | SUCCESS |
| EP-unit 3 Pin 1 | 4.811 | 4.754 | 4.774 | 0.007 | SUCCESS |

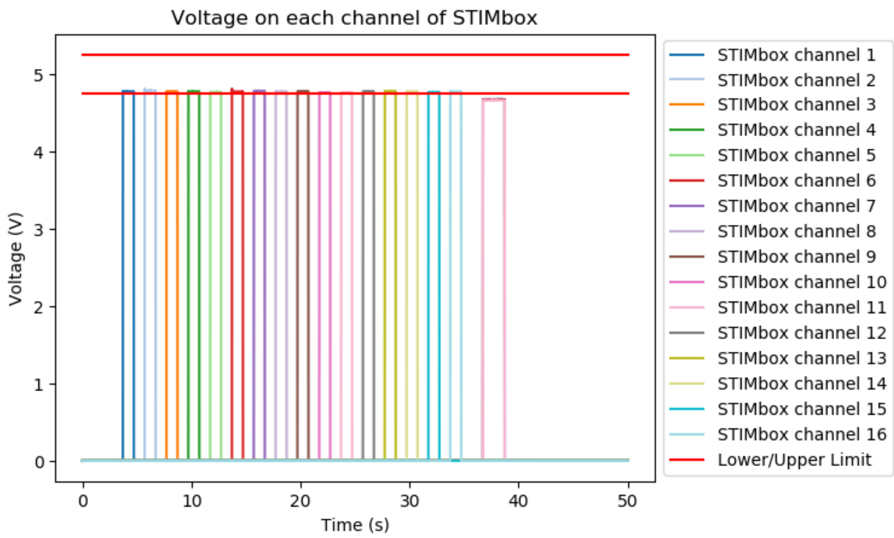



Figure 29: Measurement of the voltage of EP-unit

#### 3.2.2.8. Test K - Output current of the EP-unit

Table 13 and figure 30 show a part of the measured results of the current test for the EP-unit. The horizontal red lines in the figure display the minimal and maximal limits. The peak of each measured channel has to be within those horizontal lines. The last peak in the figure is the measurement of three powered channels (1, 6 and 11) at the same time. This peak will not be evaluated in this version of the software.

Table 13: Result of test K

| Channel | Maximum of current (mA) | Minimum of current (mA) | Mean of current (mA) | Standard deviation of current (mA) | Result |
|---|---|---|---|---|---|
| EP-unit 1 Pin 1 | 144.791 | 144.451 | 144.748 | 0.021 | SUCCESS |
| EP-unit 2 Pin 1 | 145.778 | 144.232 | 144.718 | 0.158 | SUCCESS |
| EP-unit 3 Pin 1 | 145.778 | 144.057 | 144.675 | 0.204 | SUCCESS |

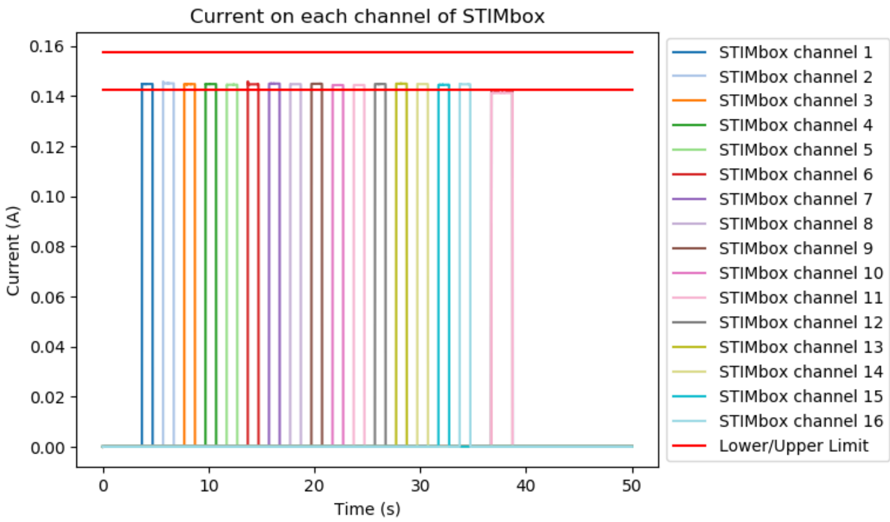



Figure 30: Measurement of the current of EP-unit

#### 3.2.2.9. Test L - Timing of the EP-unit

Table 14 and figure 31 show the measured results of the timing test for the EP-unit. The first and the last green peak in this figure display when all three channels of the EP-unit are high at the same time. The red peaks show the measured digital input signals of the wireless EEG. The first green peak has been taken to adjust the measured signal (red) from the Wireless EEG to the whole measurement. The peaks afterwards are taken for the time difference calculations of the trigger signal between both components.

Table 14: Result of test L

| Time difference (ms) | Wireless EEG DI 1 | Wireless EEG DI 2 | Wireless EEG DI 3 | Wireless EEG DI 1-3 |
|---|---|---|---|---|
| EP-unit 1 Pin 1 | 4.0 | - | - | 4.0 |
| EP-unit 2 Pin 1 | - | 4.0 | - | 4.0 |
| EP-unit 3 Pin 1 | - | - | 8.0 | 4.0 |

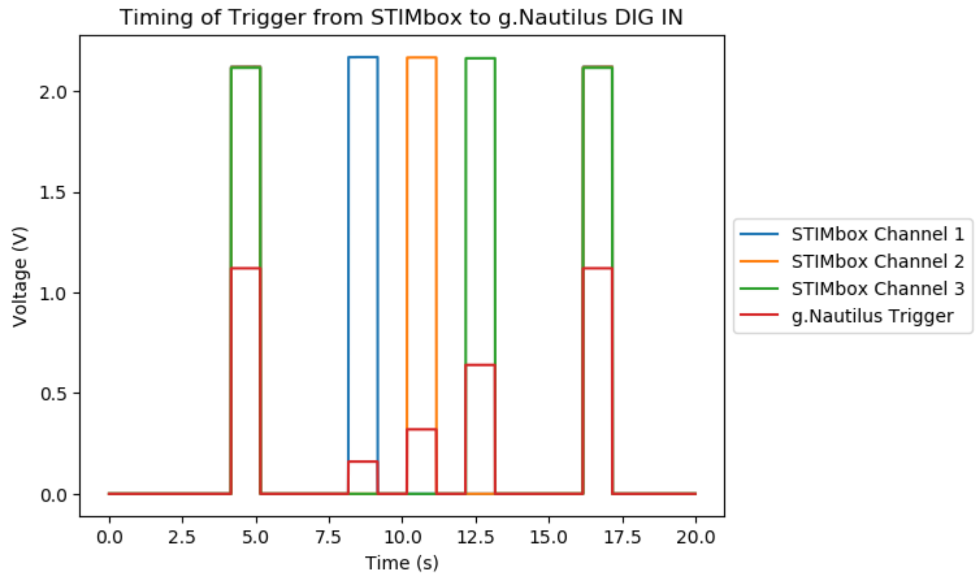



Figure 31: Measurement of the time difference between the EP-unit and the wireless EEG

#### 3.2.2.10. Test M - Connection test of the Wireless EEG

Table 15 shows the serial numbers of the wireless EEG. Therefore, the base station receives the serial number of the headset and by doing so verifies that the wireless connection between both components is working.

Table 15: Result of test M

| | **Serial Number** |
|---|---|
| Wireless EEG base station | NB-2017.09.50 |
| Wireless EEG headset | NA-2017.09.50 |

#### 3.2.2.11. Test O - Output voltage of the medical computer

Table 16 and figure 32 show the measured results of this voltage test. The red horizontal lines in the figure display the minimal and maximal limit. The measured data has to be within those red lines. The green curve is the actual measured data and the blue curves are the lower and upper measurement tolerances, which are calculated by the resistor and DAQ-card tolerances.

Table 16: Result of test O

| **Voltage [V]** | | | |
|---|---|---|---|
| Minimum | Maximum | Mean | Standard Deviation |
| 19.0786 | 19.1148 | 19.0971 | 0.006721 |

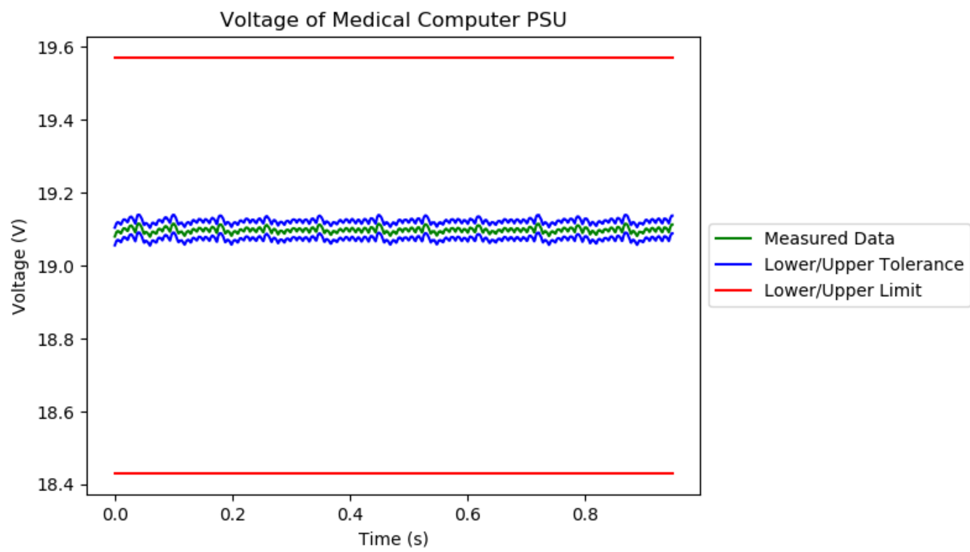



Figure 32: Power supply for the medical computer

#### 3.2.2.12. Test P - Vibration of the tactile stimulators

Table 17 and figure 33 show the measured results of this test. Therefore, the measured result is multiplied by a calibration value. This calibration value is the result of the measurement of one calibrated tactile stimulator, which has a mean amplitude of 100%. The table shows how much the three stimulators are vibrating compared to the calibrated one. The figure shows this vibrations graphically. The red horizontal lines in the figure display the minimal per centage limit. The measured data has to be above (positive half-wave), or below (negative half-wave) those red lines.

Table 17: Result of test P

| **Values** | **Stimulator 1** | **Stimulator 2** | **Stimulator 3** |
|---|---|---|---|
| Minimal maximal value | 91.18% | 89.5% | 103.79% |
| Maximum minimal value | -91.35% | -89.01% | -100.52% |
| Mean of maxima | 94.62% | 93.22% | 105.74% |
| Mean of minima | -95.41% | -92.3% | -102.14% |
| Mean amplitude | ±95.02% | ±92.76% | ±103.94% |
| Frequency | 197.06Hz | 183.16Hz | 188.67Hz |
| Test result | SUCCESS | SUCCESS | SUCCESS |

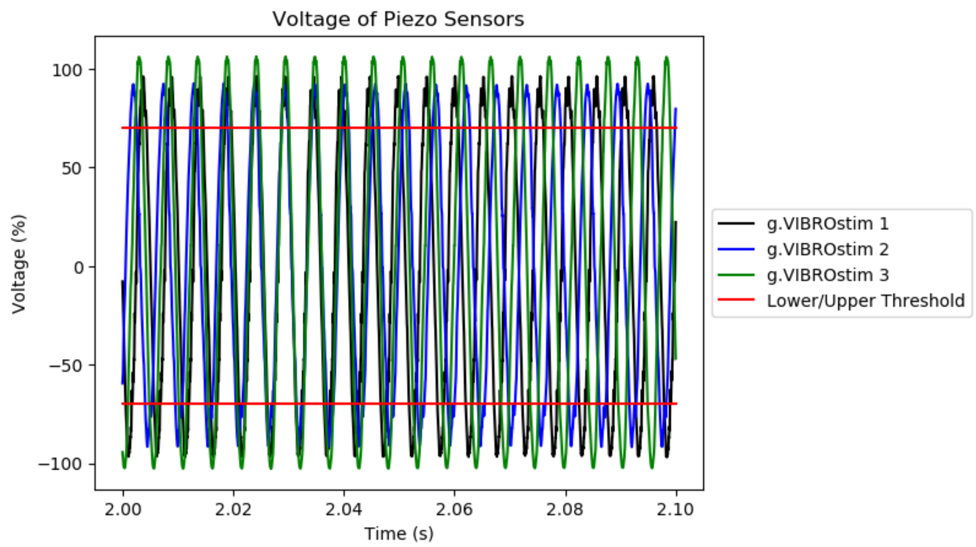



Figure 33: Measurement of the tactile stimulators

#### 3.2.2.13. Temperature measurement

Figure 34 shows the measured temperature during the whole testing scenario. The red curve is the temperature inside the stimulation system and the blue curve is the room temperature.

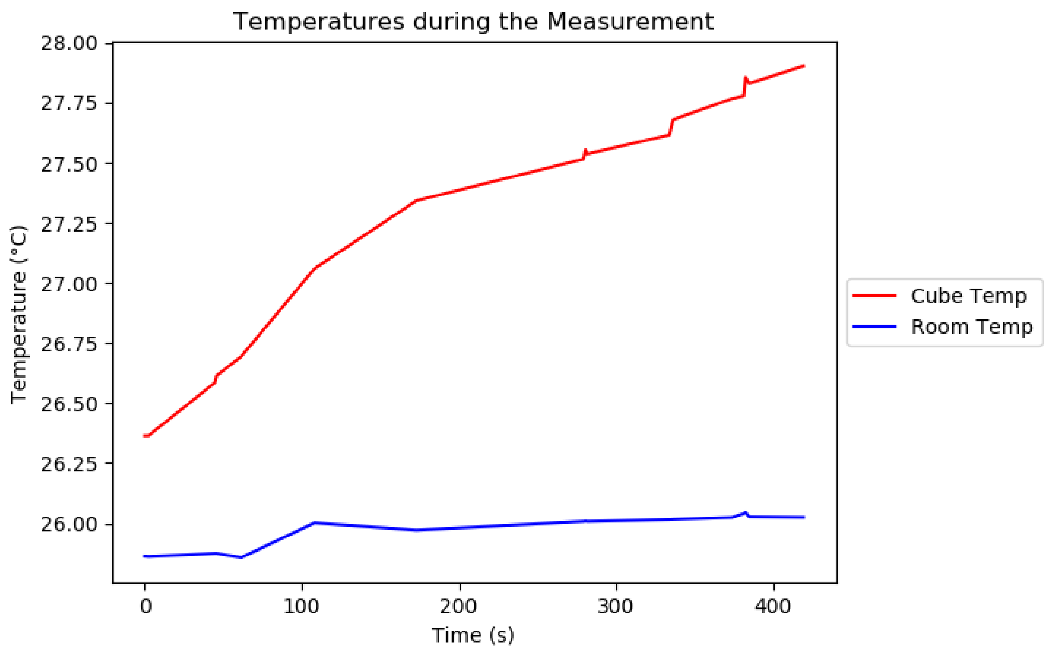



Figure 34: Temperatures

# 4. Discussion

## 4.1. Hardware

The accomplished hardware of the test-bed satisfies the required specifications. All requested test measurements, which are possible with this version of the stimulation system prototype, have been implemented. Each measurement circuit of the hardware of the test-bed has been controlled according to its functionality in form of a verification, which is also documented at the company. Therefore, the included software for the DAQ-card has been used. By the means of it, it is possible to set the input and output pins of the DAQ-card without the *Python* software used in this masters thesis. All switches (solid-state-relays) and the output signals on all output test points have been controlled according to the specification and are working as expected.

The additional hardware equipment (seen in figure 19) also works as it should. The adapter cable has been checked with the continuity test and the temperature sensors have been controlled with a comparison temperature measurement with a calibrated multimeter. The hardware for the tactile stimulators works, but in future there may be a better solution for it. The problem is, that every tactile stimulator is pressed down on the piezo sensor with a slightly different pressure, which results in different output voltages delivered by each piezo sensor. For now this problem has been solved by multiplying the measured output voltage with a calibration value. So every tactile stimulator gets compared with the calibrated one, which is adjusted to deliver 100%. The measurement limits of this test may also differ from the one taken in this thesis in future.

Since nearly every measurement is done on a voltage divider, it would be possible to change the maximum possible output values in the specification, which are measured by the DAQ-card, by just changing the resistors on the printed circuit board. The only thing, which needs attention and perhaps a change is that the measured voltage on the analog input pins of the DAQ-card does not exceed its $\pm 10V$ maximum input range.The resistors for the FES measurement circuit, for instance, are adjusted for a possible measurement of $\pm 120V$ and $\pm 120mA$. These resistor values have been chosen during an older version of the specification document. At the moment the maximum output of the stimulation system's FES component is $\pm 60V$ and $\pm 60mA$, but it may be higher again in the future.

Because of the DAQ-card mounted on the PCB the housing and also the PCB have to be made bigger than initially expected. But this has the advantages that the routing in *Ea-*

*gle* and also the manual soldering of the components was easier. Moreover, no additional hardware item (in this case the DAQ-card) is standing on the validated measurement place. Additionally the housing cannot be opened without tools and the power supply for the test-bed is provided with two USB plugs (one for the test-bed PCB and one for the DAQ-card). So no additional power supply adapter is needed, which is another advantage.

## 4.2. Software

The described test cases and their limits in chapter 2.2.3.2 are defined by the specification document of the company for the prototype of the stimulation system. It may be that further tests will be implemented and minor changes will be made, according to the limits of each test, in the future. It could be that some limits are too slightly or strictly chosen. The first test results of the first prototypes will show whether the limits are in an acceptable range or need to be adapted. The limits have to be chosen in a way that the test cases, on the one hand, will succeed and on the other hand, that possible errors will be noticed.

The *Python* software is programmed user- friendly, easy to use and to expand. It is, for example, easy to add a new test case to the already existing ones by adding it to the XML configuration input file. The software recognizes this on its own and builds the GUI depending on the written lines in the XML input file. The only things, which need to be programmed are the new test case's tasks. The multithreading, which is needed if a test needs for example an own output paradigm and measurement, is programmed in a way that the main thread waits until all sub-threads are finished. The main thread then calculates all needed results and starts the next test when finished.

Concerning the measurement care must be taken in choosing the sampling rate of the DAQ-card. The sampling rate needs to be in compliance with the Nyquist-Shannon sampling theorem, which is shown in equation 16, where $f_a$ is the sampling frequency and $f_s$ is the maximum frequency of the measured signal.

$$f_a > 2 \cdot f_s \tag{16}$$

To be able to sample the signal, the sampling frequency of the Nyquist-Shannon sampling theorem would be sufficient, but because of the capabilities of the DAQ-card, a value of at least ten times the maximum frequency of the measured signal has been chosen. Because of the low frequencies of the signal in each test and the maximum possible sampling rate of 400.000S/s of the DAQ-card this was achieved easily. The adjusted sampling rates for each test can also be found in chapter 2.2.3.2.

### 4.2.1. Test cases

Most of the referenced figures and tables of each test found in chapter 3.2.2 are just a minor extraction of the complete results of the test report, which can be found in the appendix A.2, but they are sufficient for the discussion of the test cases.

**Test A - Functionality of the USB hub**

The functionality of the USB hub is always the first test. It checks whether the composition of the stimulation system is correct and whether they are communicating with the USB hub. In case they are not, all further tests would not make any sense, because the software calls the API of each component, which would not be possible if this component was not built in. Therefore they are aborted at this point. Table 4 shows the succeeding result of this test.

**Test B - Functionality of the LEDs**

This test is the only manually conducted test in the whole testing procedure and verifies if each LED is working. One suggestion for making this test automatic, would be the use of phototransistors, which recognize light. But to actually do this without errors, the phototransistor needs to completely cap the LED so that no light from outside is recognized. So if the phototransistor is not mounted correctly over the LED the result would be misdiagnosed. Parts of the subsequent results can be seen in table 5.

**Test C - Compliance voltage of the FES**

The compliance voltage of the FES component at this prototype's version should be $\pm 60V \pm 10\%$. Table 6 shows all important parameters of this test which were calculated, including the final result. Figure 25 shows the whole measurement and the zoomed in plot of FES channel 1. It can be seen that all calculated and measured values of both FES channels are inside the given limit and further on that the standard deviation is not too high. The plots have been made for a fast visual inspection of the results.

**Test D - Stimulation current of the FES**

The stimulation current is calculated from the measured voltage of the DAQ-card. Table 7 and figure 26 show the measured/calculated results for $\pm 60mA \pm 10\%$. All other tables and figures with the other current adjustments and its limits can be found in the appendix. The table shows all important results, which are in its given limits. In the beginning of figure 26a an outlier can be seen. All outliers are detected and deleted from the software automatically before the result calculations.

**Test E - Frequency of the FES**
Table 8 and figure 27 show the result of the frequency test of FES channel 1. The table just shows the result for the adjusted $100Hz \pm 10\%$. Again it can be seen that all calculated values are within the limits. The final frequency is calculated from the mean value of all frequencies, which are determined with the reciprocal value of the period duration of one pulse. First, there was the idea of calculating the frequency using the fast Fourier transformation, but because of the high sampling rate and the long measurement duration it took too long to get a result. To avoid these the signal could be sampled with a lower sampling rate.

**Test F - Durations of the FES**
The duration test for the FES component checks the phase duration and inter-phase durations of each pulse. Tables 9, 10 and 11 are again an extraction of the complete test report and show the calculated results for the phase duration ($100\mu s \pm 5\%$) and inter-phase duration ($500\mu s \pm 5\%$). All results were again in the given limits and the test was successful. Figure 28 shows the complete recorded measurement and two zoomed in pulses for the phase and inter-phase duration. On the abscissa of the zoomed in plots both durations can be read out.

**Test J - Output voltage of the EP-unit**
The output voltage of the EP-unit should be $5V \pm 5\%$ on each channel. In total there are 16 channels. The first and the second socket have 5 channels each and the third socket has 6. Table 12 is an extraction of the full table of the complete test report in the appendix and figure 29 shows the plot of all channels. Each first channel of each socket is used as power supply for the tactile stimulators. The calculated results of these three channels are also shown in the table and are within the given limit. As mentioned before the figure shows all 16 channels. The $17^{th}$ (last) peak shows the amplitude when each first channel is set to high at the same time (at this version of the software not included in the results). It is observable that the last peak would fail the test and that all other channels succeed the test barely. All amplitudes are very close to the lower limit. This is because of the circuitry of the stimulation systems output. To fix this the resistor $R_1$ will be lowered from $18\Omega$ to $15\Omega$ in version 1.1. After that the voltage on the measurement resistor should be higher and about in the middle of the two red tolerance lines of figure 29. Further on the last peak will also be evaluated in version 1.1 of the test-bed.

**Test K - Output current of the EP-unit**
The output current of the EP-unit should be $150mA \pm 5\%$. The current and their results (seen in table 13 and figure 30) are calculated from the measured data of test J. So this test has the same problem as test J. After changing the resistor value the measured current will be higher and as mentioned before this will be part of version 1.1. Nevertheless, the

test succeeds barely.

**Test L - Timing of the EP-unit**

Table 14 and figure 31 show the results for this test. The results are the time differences between the high peaks of the EP-unit and the wireless EEG. It is observable that the measured trigger signal of the wireless EEG (in red) is rising in its amplitude. This is because the wireless EEG sets the appropriate bit of the corresponding channel to high. Digital input one is 001, digital input two is 010 and digital input three is 100. It is visible that digital input two has an amplitude which is twice ($2^1$) as high as the amplitude of digital input one and digital input three is four times higher ($2^2$). The first and the last peak correspond to the setting 111 (all first pins of the EP-unit are set to high), which means that the amplitude is seven times higher ($2^0 + 2^1 + 2^2$) than the amplitude of digital input one. All in all the test succeeds.

**Test M - Connection test of the Wireless EEG**

This test is used to show that the wireless connection of the wireless EEG works. The base station connects with its headset and retrieves its serial number. If the connection did not work the serial number field for the headset would be empty in table 15. This test also succeeds, because both serial numbers are listed in the table.

**Test O - Output voltage of the medical computer**

The results of the output voltage of the medical computer's power supply is shown in table 16 and figure 32. The result is within the limit of $19V \pm 3\%$ and the test is successful. As it can be seen in the figure, the voltage ripples during the whole measurement. It could be that the power supply is rippling by itself, or that these changes occur during the measurement with the DAQ-card when the analog-digital-converter switches its states.

**Test P - Vibration of the tactile stimulators**

This test will always be done at last, because the vibrations of the tactile stimulators are very loud and so the test engineer hears when the test is nearly finished. As mentioned before, the measurement is conducted with piezo sensors and the measured results are multiplied with a calibration value. The results are shown in table 17, where every tactile stimulator succeeds the test. To succeed the test the limit for the amplitude is chosen to be at least $\pm 70\%$ of the calibrated one. Furthermore, the frequency of each stimulator needs to be within 150 and 250Hz. Figure 33 shows the amplitude of the vibrations of each stimulator, which are all above the defined threshold. Nevertheless, this test could be completely changed in further versions of the test-bed, because of the possible modification of the additional hardware equipment.

**Temperature measurement**
The temperature is determined before and after each test with the temperature sensors. Figure 34 shows the temperatures during the whole measurement. For now there is no specification for a maximum allowed temperature, but this may vary in future versions.

## 4.3. Expansions in further versions of the test-bed

Because of some missing parts of the stimulation system's prototype it is not yet possible to do the test cases for the tDCs (tests G,H and I) and the audio signal (test N). The hardware of the test-bed is already finished, but the measurement cannot be done now and so they are not included in the software yet. Nevertheless, the description of those test cases can be found in chapter 1.7.2 and 2.2.3.2 and they will be added in future versions of the test-bed. Additionally the calibration of the FES and tDCs component, which is at the moment done separately, may also be included into this software in the future. The circuit board of the hardware already offers the opportunity of plugging in external resistors for the FES and tDCs calibrations. Further on, there may be some bugs in the software, which also require fixing.

## 4.4. Conclusion

The finished test-bed is used for the quality assurance. It identifies possible errors in the composition and also in the firmware of the stimulation system. The complete test report, being found in the appendix, shows the results of one stimulation system's prototype. With the help of the summary page and the tables and figures of the report it is possible to identify errors very fast. Moreover, it is possible to do just a single test, so that after the fixation of one error the appropriate test can be repeated. This allows a fast reparation of a defective device.

The structure of the software allows an easy extension and editing of the tests. Further on it is protected against faulty insertions in case of limits, etc., because of the external XML input file, which is loaded at each program startup.

All in all the test-bed is working as expected and can be used to identify possible errors of the stimulation system of this version and also of future versions with slight modifications in the test-bed's software.

# References

[1] Alois Schlögl. Script to Fundamentals of Biomedical Engineering, Laboratory. The Electroencephalogram. 2009.

[2] Christian Baumgartner. Script to Medical Devices Legislation, Pages 241-264. 2015.

[3] Pierre CLADÉ. PyDAQmx: A Python interface to the National Instruments DAQmx driver, August 2017. https://pythonhosted.org/PyDAQmx/.

[4] The Scipy community. Numpy and Scipy Documentation, August 2017. https://docs.scipy.org/doc/.

[5] The Qt Company. Qt documentation, August 2017. http://doc.qt.io/qt-5/.

[6] CadSoft Computer GmbH. Eagle, June 2017. https://www.autodesk.com/products/eagle/overview.

[7] Riverbank Computing. PyQt5 Reference Guide, August 2017. http://pyqt.sourceforge.net/Docs/PyQt5/.

[8] IXYS Corporation. CPC2030N, June 2017. http://www.ixysic.com/home/pdfs.nsf/www/CPC2030N.pdf/$file/CPC2030N.pdf.

[9] IXYS Corporation. LCC110, June 2017. http://www.ixysic.com/home/pdfs.nsf/www/LCC110.pdf/$file/LCC110.pdf.

[10] David Cortesi. Pyinstaller, August 2017. https://pythonhosted.org/PyInstaller/.

[11] Eurociruits GmbH, July 2017. https://www.eurocircuits.de.

[12] Python Software Foundation. ctypes - A foreign function library for Python, August 2017. https://docs.python.org/3/library/ctypes.html.

[13] Bopla Gehäuse Systeme GmbH. Bopla Housing BO 52012, July 2017. https://www.bopla.de/de/gehaeusetechnik/product/botego/botego-bo-520-l-gehaeuse/bo-52012.html.

[14] Tim Golden. WMI - Windows Management Instrumentation, August 2017. http://timgolden.me.uk/python/wmi/.

[15] National Instruments. DAQ M Series NI USB-621x User Manual Data Sheet, May 2017. http://www.ni.com/pdf/manuals/371931f.pdf.

[16] National Instruments. Device Specifications NI USB-6216 Data Sheet, May 2017. http://www.ni.com/pdf/manuals/375198c.pdf.

[17] National Instruments. NI USB-6216 OEM, May 2017. http://sine.ni.com/nips/cds/view/p/lang/de/nid/204358.

[18] National Instruments. User Guide NI USB-621x OEM Data Sheet, May 2017. http://www.ni.com/pdf/products/us/372083b.pdf.

[19] International Organisation for Standardization. EN ISO 13485:2012 Medical devices - Quality management systems - Requirements for regulatory purposes. 2012.

[20] David James. Python wrapper generator for ctypes, August 2017. https://pypi.python.org/pypi/ctypesgen/0.r125.

[21] John Hunter, Darren Dale, Eric Firing, Michael Droettboom. WMI - Windows Management Instrumentation, August 2017. https://matplotlib.org/contents.html.

[22] Martin R, Sadowsky C, Obst K, Meyer B, McDonald J. Functional electrical stimulation in spinal cord injury: From theory to practice. *Top Spinal Cord Inj Rehabil.*, 18(1):28–33, 2012.

[23] Microship. Low power linear active thermistor ICs, June 2017. http://ww1.microchip.com/downloads/en/DeviceDoc/20001942G.pdf.

[24] Nitsche MA, Paulus W. Excitability changes induced in the human motor cortex by weak transcranial direct current stimulation. *The journal of physiology*, 527(Pt 3):633–639, 2000 Sep 15.

[25] Stephan Richter. lxml - XML and HTML with Python, August 2017. http://lxml.de/index.html.

[26] ON Semiconductor. NTR3C21NZ Power MOSFET, June 2017. http://www.onsemi.com/pub/Collateral/NTR3C21NZ-D.PDF.

[27] STMicroelectronics. DALC208 low capacitance diode array, June 2017. http://www.st.com/content/ccc/resource/technical/document/datasheet/4c/e1/0f/3d/c5/a4/46/eb/CD00001734.pdf/files/CD00001734.pdf/jcr:content/translations/en.CD00001734.pdf.

# A. Appendix

## A.1. Class for the DAQ-card measurements

```python
import PyDAQmx as daq
# DAQ-card measurement initializations (AI, DO) for a device
# -----------------------------------------------------------------------------
class MeasurementDevice():
    def __init__(self,sample_rate, n_samples, min_range, max_range):
        self.sample_rate = sample_rate # max ist 400kS
        self.n_samples = n_samples
        self.high = np.array([1], dtype=np.uint8)
        self.low = np.array([0], dtype=np.uint8)
        self.min_range = min_range
        self.max_range = max_range
        self.data = np.zeros((n_samples,), dtype=np.float64)
        self.read = daq.int32()
        self.ai = "low"
        self.do = "low"
        #Create DAQ tasks
        self.do_task = daq.Task()
        self.ai_task = daq.Task()
        self.channels = 1

    #Measure device with the already set AI and DO channel
    #-----------------------------------------------------------------------------
    def measureDevice(self):
        try:
            # Create DO channel on DO
            if self.do != "low":
                self.do_task.CreateDOChan(self.do, "",
                    daq.DAQmx_Val_ChanPerLine)

            #Select voltage range on AI
            if self.ai != "low":
                data = np.zeros((self.n_samples,), dtype=np.float64)
                self.ai_task.CreateAIVoltageChan(self.ai, "",
                    daq.DAQmx_Val_RSE, self.min_range, self.max_range,
                    daq.DAQmx_Val_Volts, None)
            # Select how long the measurement will be in samples (sample_rate
                and N_samples)
```

```
            self.ai_task.CfgSampClkTiming("", self.sample_rate,
                daq.DAQmx_Val_Rising, daq.DAQmx_Val_FiniteSamps,
                self.n_samples)

        #Start tasks
        if self.do != "low":
            self.do_task.StartTask()
        if self.ai != "low":
            self.ai_task.StartTask()

        # DO to high
        if self.do != "low":
            self.do_task.WriteDigitalLines(1, 1, 10.0,
                daq.DAQmx_Val_GroupByChannel, self.high, None, None)

        # Read AI pin for max. (-1) seconds (-1 = infinity)
        if self.ai != "low":
            self.ai_task.ReadAnalogF64(self.n_samples, -1,
                daq.DAQmx_Val_GroupByScanNumber, self.data,
                self.n_samples*self.channels, daq.byref(self.read), None)

        # DO to low
        if self.do != "low":
            self.do_task.WriteDigitalLines(1, 1, 10.0,
                daq.DAQmx_Val_GroupByChannel, self.low, None, None)

        #Stop and clear tasks
        if self.do != "low":
            self.do_task.StopTask()
            self.do_task.ClearTask()
        if self.ai != "low":
            self.ai_task.StopTask()
            self.ai_task.ClearTask()

        #Delete the first 5% of data if just one channel has been measured
            and return the measured data
        if self.ai != "low":
            if self.channels == 1:
                if self.n_samples >= 20:
                    self.data = self.data[round(self.n_samples*0.05):]
                        #Delete the first 5% of data and make it always an
```

```
                        integer with round
            return self.data
        except:
            print("Could not start measurement. \nNo DAQ card connected")
            return -1

    #Set AI and DO channels for the desired measurement device
    #----------------------------------------------------------------------------
    #Tactile stimulator 1
    def setVIB1(self,n_samples):
        self.do = "/Dev1/port0/line12, /Dev1/port0/line2" #DO12 VIB1 PSU and
            DO2 Piezo1
        self.ai = "Dev1/ai2"         #AI2
        self.high = np.array([1,1], dtype=np.uint8)
        self.low = np.array([0,0], dtype=np.uint8)
        self.channels = 1
        self.data = np.zeros((n_samples,), dtype=np.float64)
    #Temperature sensor 1
    def setTemp1(self,n_samples):
        self.do = "/Dev1/port0/line8" #DO8
        self.ai = "Dev1/ai5"         #AI5
        self.high = np.array([1], dtype=np.uint8)
        self.low = np.array([0], dtype=np.uint8)
        self.channels = 1
        self.data = np.zeros((n_samples,), dtype=np.float64)
    #FES channel 2
    def setFES2(self,n_samples):
        self.do = "/Dev1/port2/line0, /Dev1/port1/line2" #DO24, DO18
        self.ai = "Dev1/ai8"         #AI8
        self.high = np.array([1,1], dtype=np.uint8)
        self.low = np.array([0,0], dtype=np.uint8)
        self.channels = 1
        self.data = np.zeros((n_samples,), dtype=np.float64)
    #EP-unit all channels
    def setEPunit(self,n_samples):
        self.ai = "Dev1/ai0:15"      #AI0-15
        self.do = "low"
        self.data = np.zeros((16*n_samples,), dtype=np.float64)
        self.channels = 16
```

## A.2. Automatically created test report

Due to the formatting of the test report it is shown from the next page.

# CU-XXXX.XX.XX Test Report

## Device Information:

| | | | | |
|---|---|---|---|---|
| **Serial Number:** | CU-XXXX.XX.XX | | **Manufacturer:** | g.tec medical engineering GmbH |
| **Components SNR:** | USB Hub: | CU-2017.06.02 | **Street:** | Herbersteinstraße 60 |
| | FES: | FR-2017.05.03 | **City:** | Graz |
| | tDCs: | - | **Country:** | Austria |
| | STIMbox: | CS-2017.01.01 | **Building:** | H60 |
| | g.Nautilus: | NB-2017.09.50 | **Room:** | 1 |

**Test Adapter SNR:** g.Cube test-bed T1
**DAQ Card:** USB - 6216 OEM **DAQ SNR:** 1c73f45

**Comment:** This is an example report.

## Test Specification:

**Tests:** A, B, C, D, E, F, J, K, L, M, O, P
**Maximum Sampling Rate:** 400.000 Samples/s
**Date:** 18.12.2017
**Duration of the whole test:** 6.99 minutes
**Test Software:** V1.0

### Summary

| Hardware | Test | Limit | Unit | Result |
|---|---|---|---|---|
| USB Hub | A: Functionality | On/Off | - | SUCCESS |
| LEDs | B: Functionality | On/Off | - | SUCCESS |
| FES | C: Compliant Voltage | 60±10% | V | SUCCESS |
| | D: Stimulation Current | ±10% | mA | SUCCESS |
| | E: Frequency | ±10% | Hz | SUCCESS |
| | F: Duration | ±5% | $\mu$s | SUCCESS |
| tDCs | H: Compliance Voltage | - | V | SKIPPED |
| | G: Stimulation Current | - | mA | SKIPPED |
| | I: Ramp up and Down time | - | $\mu$s | SKIPPED |
| STIMbox | J: Voltage | 5±5% | V | SUCCESS |
| | K: Current | 150±5% | mA | SUCCESS |
| | L: Timing | ±10 | ms | SUCCESS |
| g.Nautilus | M: Connection Test | On/Off | - | SUCCESS |
| Audio | N: Signal check | - | V | SKIPPED |
| Medical PC PSU | O: Voltage | 19±3% | V | SUCCESS |
| g.VIBROstim | P1: g.VIBROstim 1 | 70 | % | SUCCESS |
| | P2: g.VIBROstim 2 | 70 | % | SUCCESS |
| | P3: g.VIBROstim 3 | 70 | % | SUCCESS |

**Min Temp:** 26.4°C **Max Temp:** 27.9°C **Mean Cube Temp:** 27.3°C **Room Temp:** 26.0°C

### Test Result

## SUCCESS

| **Test Engineer:** Daniel Payer |
|---|
| **Signature:** |

# Appendix

## Test A: Functionality of USB Hub

Serial: CU-XXXX.XX.XX
Final result: SUCCESS

### Description:

Checks if all built in components have a connection with the USB hub and receives its serial number.

### Test Criteria:

The components have been recognized by its serial numbers and are okay.

### Detailed Results:

| Hardware Item | Serial Number | Recognized |
|---|---|---|
| USB Hub | CU-2017.06.02 | OK |
| FES | FR-2017.05.03 | OK |
| tDCs | - | - |
| STIMbox | CS-2017.01.01 | OK |
| g.Nautilus Base Station | NB-2017.09.50 | OK |

## Test B: Functionality of LEDs

Serial: CU-XXXX.XX.XX
Final result: SUCCESS

### Description:

Visual test to check if every LED is working and has the correct color.

### Test Criteria:

LED lights are ok.

### Detailed Results:

| LED | Color | Lighting |
|---|---|---|
| g.Cube Power Supply | Green | YES |
| g.Nautilus | Blue | YES |
| STIMBOX 1 | White | YES |
| STIMBOX 2 | White | YES |
| STIMBOX 3 | White | YES |
| FES A high current | Orange | YES |
| FES A stimulate | Blue | YES |
| FES B high current | Orange | YES |
| FES B stimulate | Blue | YES |
| tDCs C high current | Orange | YES |
| tDCs C stimulate | Blue | YES |
| tDCs D high current | Orange | YES |
| tDCs D stimulate | Blue | YES |

## Test C: FES Voltage Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement duration: 3 seconds
Sample rate: 200.000 Samples/s

### Description:

Checks the voltage of the positive and negative pulse of both FES channels for 10 pulses. Therefore the voltage is measured/recorded and calculated.

### Test Criteria:

The maximum voltage of both FES channels is in between the range of 60V±10% [54.0V; 66.0V].

### Detailed Results:

| Parameter | FES channel 1 | FES channel 2 |
|---|---|---|
| Maximal positive voltage (V) | 58.21 | 58.36 |
| Maximal negative voltage (V) | -58.21 | -58.34 |
| Minimal positive voltage (V) | 57.92 | 58.08 |
| Minimal negative voltage (V) | -57.93 | -58.08 |
| Mean of positive voltage (V) | 58.01 | 58.15 |
| Mean of negative voltage (V) | -57.99 | -58.14 |
| Standard deviation of positive voltage (V) | 0.05 | 0.05 |
| Standard deviation of negative voltage (V) | 0.04 | 0.04 |
| Mean of amplitude (V) | ±58.0 | ±58.14 |
| **Test result** | SUCCESS | SUCCESS |

### Plots:

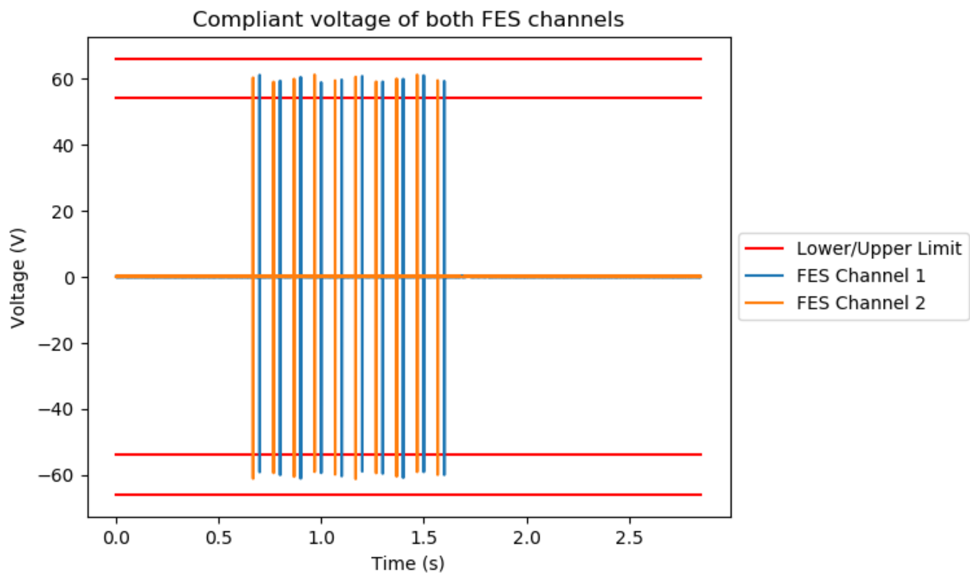



Figure 1: Measurement of the voltage of both FES channels

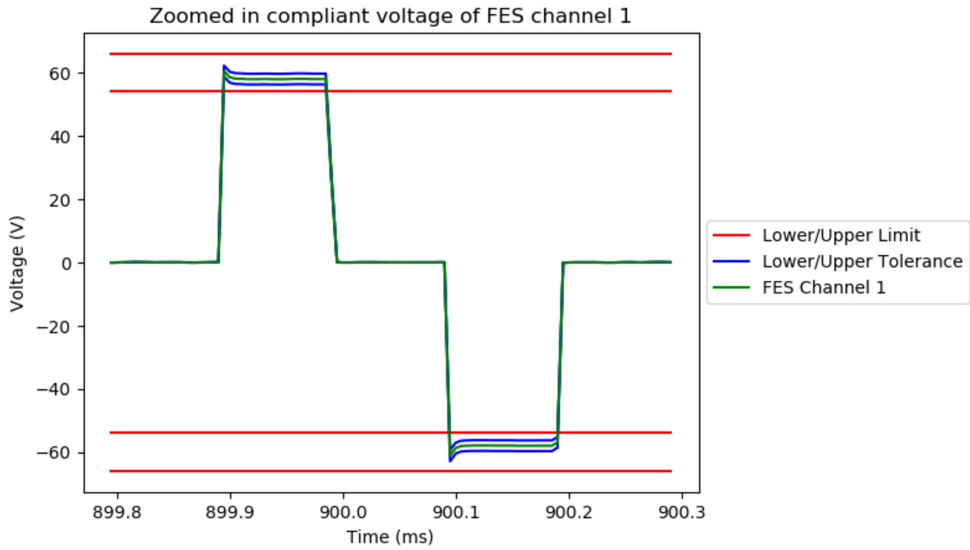



Figure 2: Zoomed in measurement of the voltage of FES channel 1

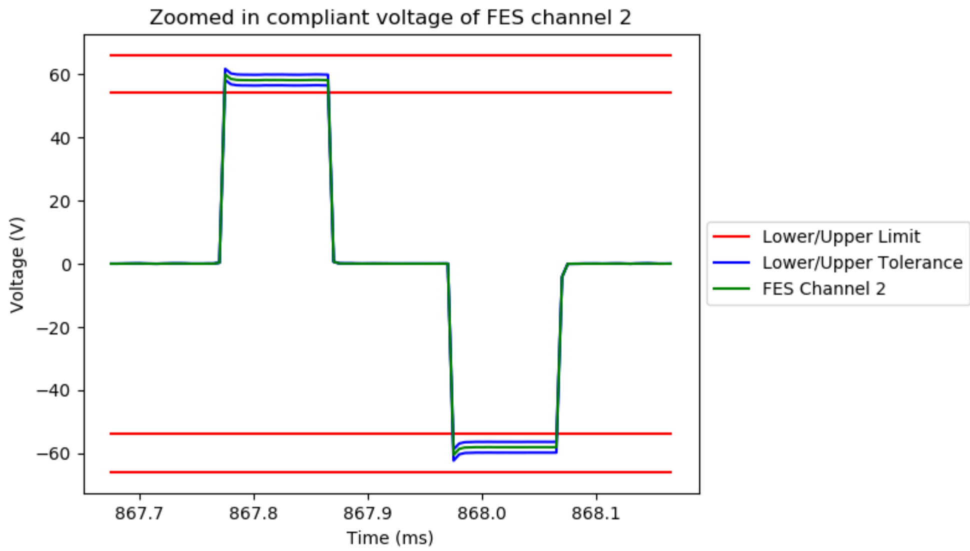



Figure 3: Zoomed in measurement of the voltage of FES channel 2

### Test D: FES Current Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement duration: 15 seconds
Sample rate: 200.000 Samples/s

#### Description:

Checks the current of the positive and negative stimuli of both FES channels for 10 pulses of each adjusted current. Therefore, the voltage is measured on a shunt resistance and then the current is calculated.
Adjusted currents: 1mA, 10mA, 20mA, 30mA, 40mA, 50mA, 60mA.

#### Test Criteria:

The measured current of both FES channels equals the adjusted current ±10%.
Limits: [0.9mA; 1.1mA], [9.0mA; 11.0mA], [18.0mA; 22.0mA], [27.0mA; 33.0mA], [36.0mA; 44.0mA], [45.0mA; 55.0mA], [54.0mA; 66.0mA].

#### Detailed Results:

| **Adjusted Current (mA)** | **Parameter** | **FES channel 1** | **FES channel 2** |
|---|---|---|---|
| 1mA | Maximal positive current (mA) | 1.09 | 1.06 |
| | Maximal negative current (mA) | -1.02 | -1.01 |
| | Minimal positive current (mA) | 0.92 | 0.95 |
| | Minimal negative current (mA) | -0.97 | -0.98 |
| | Mean of positive current (mA) | 1.01 | 1.01 |
| | Mean of negative current (mA) | -1.0 | -1.0 |
| | Standard deviation of positive current (mA) | 0.04 | 0.03 |
| | Standard deviation of negative current (mA) | 0.01 | 0.01 |
| | Mean of amplitude (mA) | ±1.0 | ±1.0 |
| | **Test result** | SUCCESS | SUCCESS |

| **Adjusted Current (mA)** | **Parameter** | **FES channel 1** | **FES channel 2** |
|---|---|---|---|
| 10mA | Maximal positive current (mA) | 10.09 | 10.08 |
| | Maximal negative current (mA) | -10.03 | -10.05 |
| | Minimal positive current (mA) | 9.9 | 9.95 |
| | Minimal negative current (mA) | -9.94 | -9.98 |
| | Mean of positive current (mA) | 9.99 | 10.01 |
| | Mean of negative current (mA) | -9.97 | -10.0 |
| | Standard deviation of positive current (mA) | 0.04 | 0.03 |
| | Standard deviation of negative current (mA) | 0.02 | 0.01 |
| | Mean of amplitude (mA) | ±9.98 | ±10.0 |
| | **Test result** | SUCCESS | SUCCESS |

| **Adjusted Current (mA)** | **Parameter** | **FES channel 1** | **FES channel 2** |
|---|---|---|---|
| 20mA | Maximal positive current (mA) | 20.06 | 20.08 |
| | Maximal negative current (mA) | -20.01 | -20.13 |
| | Minimal positive current (mA) | 19.88 | 19.94 |
| | Minimal negative current (mA) | -19.92 | -19.98 |
| | Mean of positive current (mA) | 19.96 | 20.01 |
| | Mean of negative current (mA) | -19.95 | -20.0 |
| | Standard deviation of positive current (mA) | 0.04 | 0.03 |
| | Standard deviation of negative current (mA) | 0.02 | 0.02 |
| | Mean of amplitude (mA) | ±19.95 | ±20.01 |
| | **Test result** | SUCCESS | SUCCESS |

| Adjusted Current (mA) | Parameter | FES channel 1 | FES channel 2 |
|---|---|---|---|
| 30mA | Maximal positive current (mA) | 30.07 | 30.13 |
| | Maximal negative current (mA) | -30.08 | -30.15 |
| | Minimal positive current (mA) | 29.84 | 29.94 |
| | Minimal negative current (mA) | -29.87 | -29.97 |
| | Mean of positive current (mA) | 29.94 | 30.01 |
| | Mean of negative current (mA) | -29.92 | -30.0 |
| | Standard deviation of positive current (mA) | 0.05 | 0.03 |
| | Standard deviation of negative current (mA) | 0.03 | 0.03 |
| | Mean of amplitude (mA) | ±29.93 | ±30.01 |
| | **Test result** | SUCCESS | SUCCESS |

| Adjusted Current (mA) | Parameter | FES channel 1 | FES channel 2 |
|---|---|---|---|
| 40mA | Maximal positive current (mA) | 40.08 | 40.22 |
| | Maximal negative current (mA) | -40.12 | -40.22 |
| | Minimal positive current (mA) | 39.8 | 39.95 |
| | Minimal negative current (mA) | -39.86 | -39.96 |
| | Mean of positive current (mA) | 39.91 | 40.01 |
| | Mean of negative current (mA) | -39.9 | -40.0 |
| | Standard deviation of positive current (mA) | 0.05 | 0.04 |
| | Standard deviation of negative current (mA) | 0.03 | 0.03 |
| | Mean of amplitude (mA) | ±39.9 | ±40.0 |
| | **Test result** | SUCCESS | SUCCESS |

| Adjusted Current (mA) | Parameter | FES channel 1 | FES channel 2 |
|---|---|---|---|
| 50mA | Maximal positive current (mA) | 50.15 | 50.34 |
| | Maximal negative current (mA) | -50.12 | -50.26 |
| | Minimal positive current (mA) | 49.8 | 49.94 |
| | Minimal negative current (mA) | -49.8 | -49.96 |
| | Mean of positive current (mA) | 49.89 | 50.02 |
| | Mean of negative current (mA) | -49.88 | -50.01 |
| | Standard deviation of positive current (mA) | 0.06 | 0.06 |
| | Standard deviation of negative current (mA) | 0.05 | 0.05 |
| | Mean of amplitude (mA) | ±49.89 | ±50.02 |
| | **Test result** | SUCCESS | SUCCESS |

| Adjusted Current (mA) | Parameter | FES channel 1 | FES channel 2 |
|---|---|---|---|
| 60mA | Maximal positive current (mA) | 60.18 | 60.34 |
| | Maximal negative current (mA) | -60.19 | -60.35 |
| | Minimal positive current (mA) | 59.73 | 59.89 |
| | Minimal negative current (mA) | -59.78 | -59.93 |
| | Mean of positive current (mA) | 59.86 | 60.02 |
| | Mean of negative current (mA) | -59.86 | -60.0 |
| | Standard deviation of positive current (mA) | 0.07 | 0.07 |
| | Standard deviation of negative current (mA) | 0.06 | 0.06 |
| | Mean of amplitude (mA) | ±59.86 | ±60.01 |
| | **Test result** | SUCCESS | SUCCESS |

**Plots:**

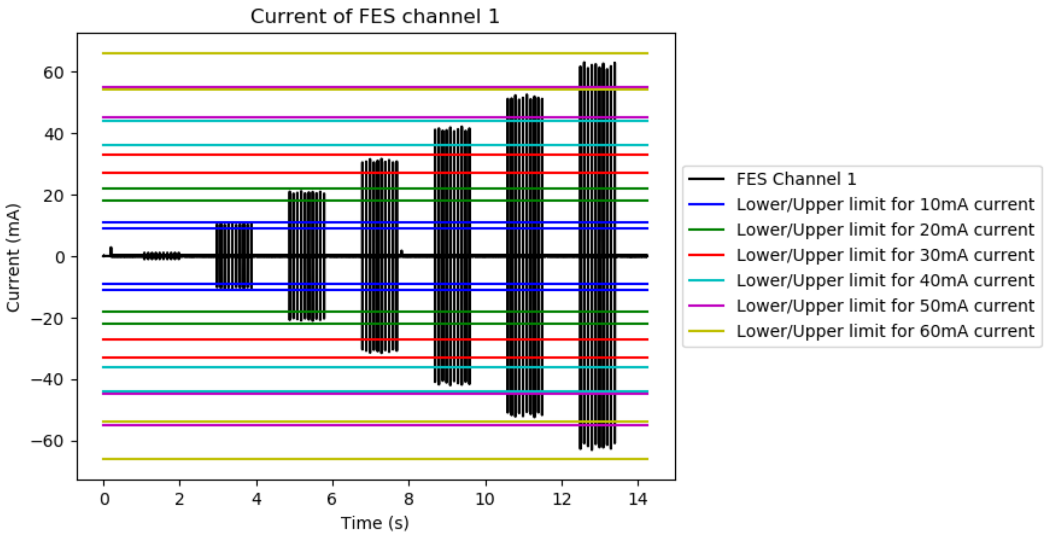



Figure 4: Measurement of the current of FES channel 1

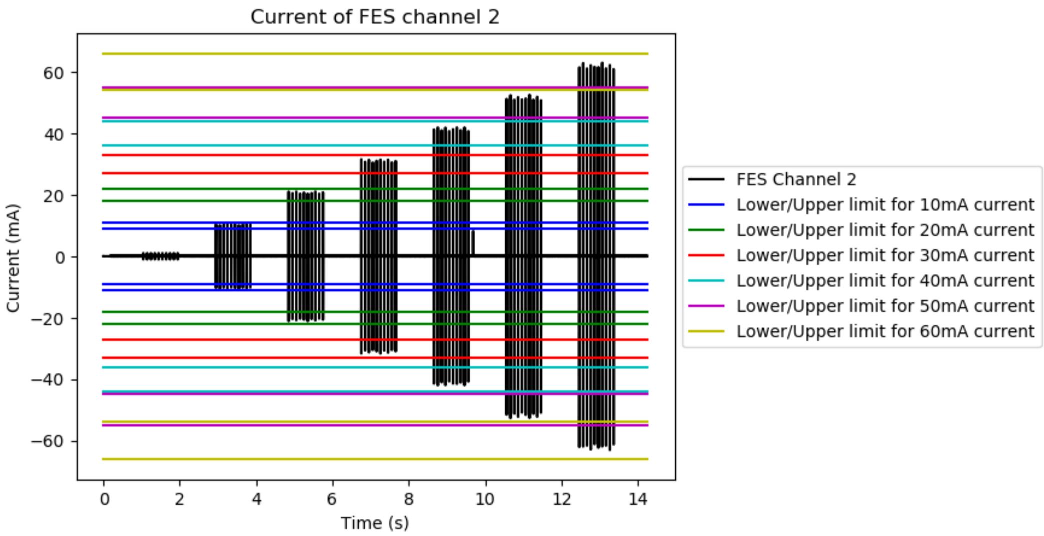



Figure 5: Measurement of the current of FES channel 2

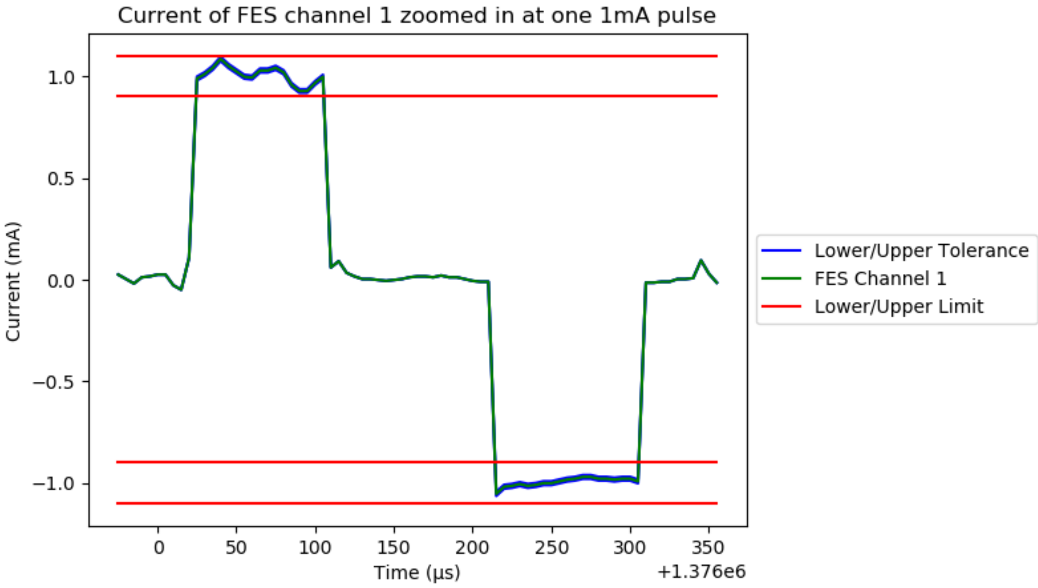



Figure 6: Zoomed in measurement of the current (1mA) of FES channel 1

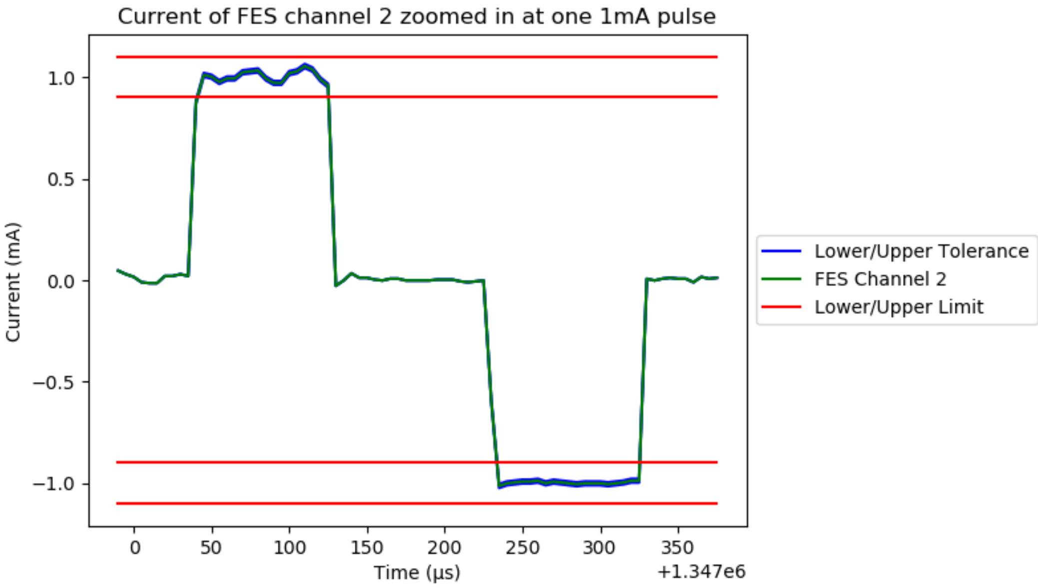



Figure 7: Zoomed in measurement of the current (1mA) of FES channel 2

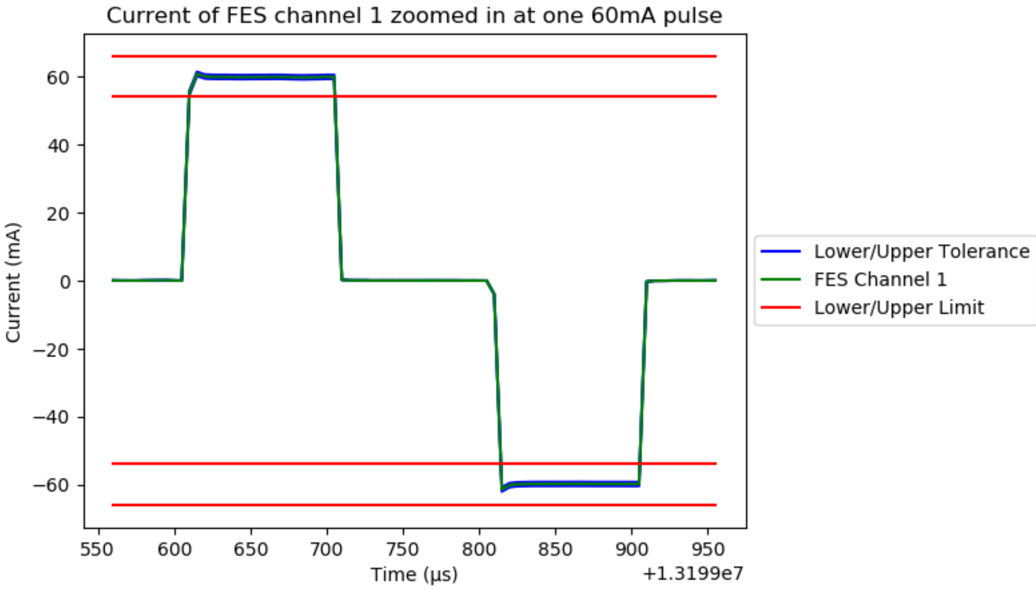



Figure 8: Zoomed in measurement of the current (60mA) of FES channel 1

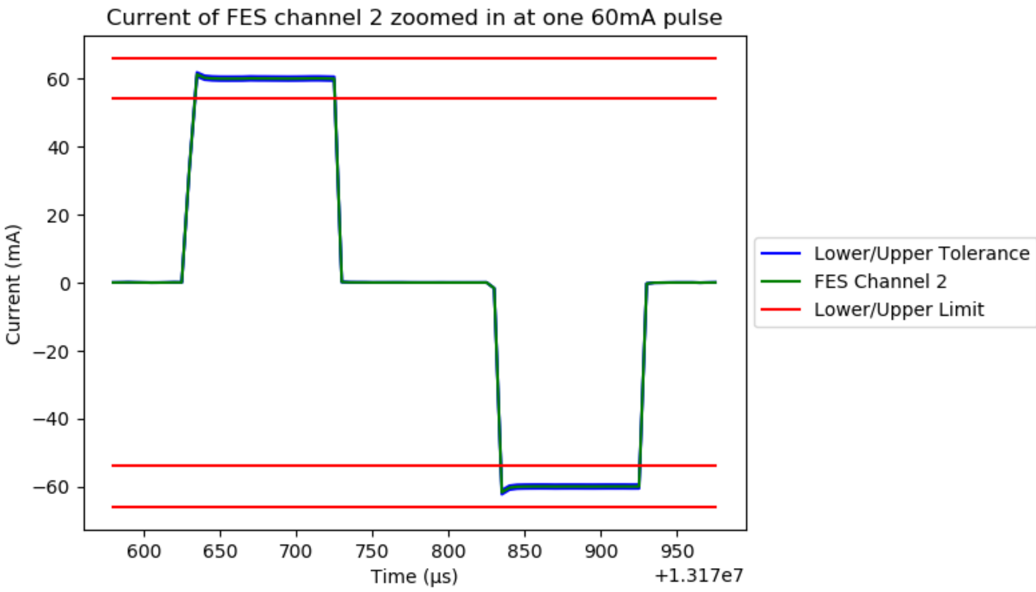



Figure 9: Zoomed in measurement of the current (60mA) of FES channel 2

## Test E: FES Frequency Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement duration: 17 seconds
Sample rate: 200.000 Samples/s

### Description:

Checks the frequency of the recorded voltage of both FES channels for 3 (1Hz) or 10 (all other frequencies) pulses each. The frequencies are adjusted after every pulse periode. Therefore the voltage is measured and recorded on a resistor and the frequencies are calculated from the samples.
Adjusted frequency: 1Hz, 10Hz, 20Hz, 30Hz, 40Hz, 50Hz, 60Hz, 70Hz, 80Hz, 90Hz, 100Hz

### Test Criteria:

The measured frequencies of both FES channels equals the adjusted frequency ±10%.
Limits: [0.9Hz;1.1Hz], [9.0Hz;11.0Hz], [18.0Hz;22.0Hz], [27.0Hz;33.0Hz], [36.0Hz;44.0Hz], [45.0Hz;55.0Hz], [54.0Hz;66.0Hz], [63.0Hz;77.0Hz], [72.0Hz;88.0Hz], [81.0Hz;99.0Hz], [90.0Hz;110.0Hz].

### Detailed Results:

| Adjusted frequency | Channel | Maximum (Hz) | Minimum (Hz) | Mean (Hz) | Standard deviation (Hz) | Test result |
|---|---|---|---|---|---|---|
| 1Hz | FES 1 | 0.999 | 0.999 | 0.999 | 0.0 | SUCCESS |
| | FES 2 | 0.999 | 0.999 | 0.999 | 0.0 | SUCCESS |
| 10Hz | FES 1 | 10.0 | 9.999 | 9.999 | 0.0002 | SUCCESS |
| | FES 2 | 10.0 | 9.999 | 9.999 | 0.0002 | SUCCESS |
| 20Hz | FES 1 | 20.0 | 19.998 | 19.999 | 0.0009 | SUCCESS |
| | FES 2 | 20.0 | 19.998 | 19.999 | 0.0009 | SUCCESS |
| 30Hz | FES 1 | 29.998 | 29.998 | 29.998 | 0.0 | SUCCESS |
| | FES 2 | 29.998 | 29.998 | 29.998 | 0.0 | SUCCESS |
| 40Hz | FES 1 | 40.0 | 39.992 | 39.997 | 0.004 | SUCCESS |
| | FES 2 | 40.0 | 39.992 | 39.997 | 0.004 | SUCCESS |
| 50Hz | FES 1 | 50.0 | 49.987 | 49.996 | 0.006 | SUCCESS |
| | FES 2 | 50.0 | 49.987 | 49.995 | 0.006 | SUCCESS |
| 60Hz | FES 1 | 60.006 | 59.988 | 59.996 | 0.009 | SUCCESS |
| | FES 2 | 60.006 | 59.988 | 59.996 | 0.009 | SUCCESS |
| 70Hz | FES 1 | 70.003 | 69.979 | 69.992 | 0.012 | SUCCESS |
| | FES 2 | 70.003 | 69.979 | 69.992 | 0.012 | SUCCESS |
| 80Hz | FES 1 | 80.0 | 79.968 | 79.989 | 0.015 | SUCCESS |
| | FES 2 | 80.0 | 79.968 | 79.989 | 0.015 | SUCCESS |
| 90Hz | FES 1 | 90.009 | 89.968 | 89.991 | 0.02 | SUCCESS |
| | FES 2 | 90.009 | 89.968 | 89.991 | 0.02 | SUCCESS |
| 100Hz | FES 1 | 100.0 | 99.95 | 99.983 | 0.023 | SUCCESS |
| | FES 2 | 100.0 | 99.95 | 99.983 | 0.023 | SUCCESS |

**Plots:**

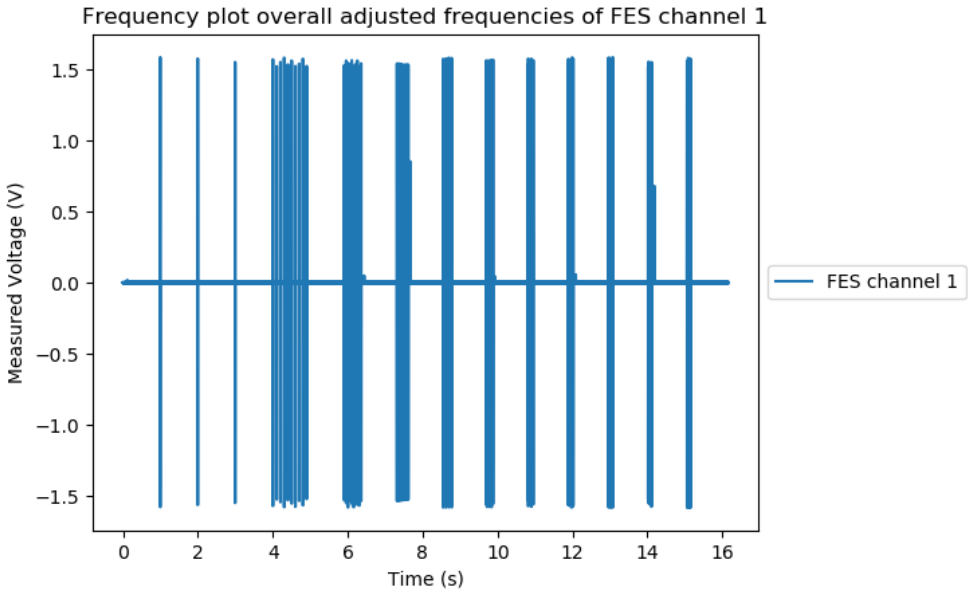



Figure 10: Measurement of the frequencies (left side 1 Hz - right side 100Hz) of FES channel 1

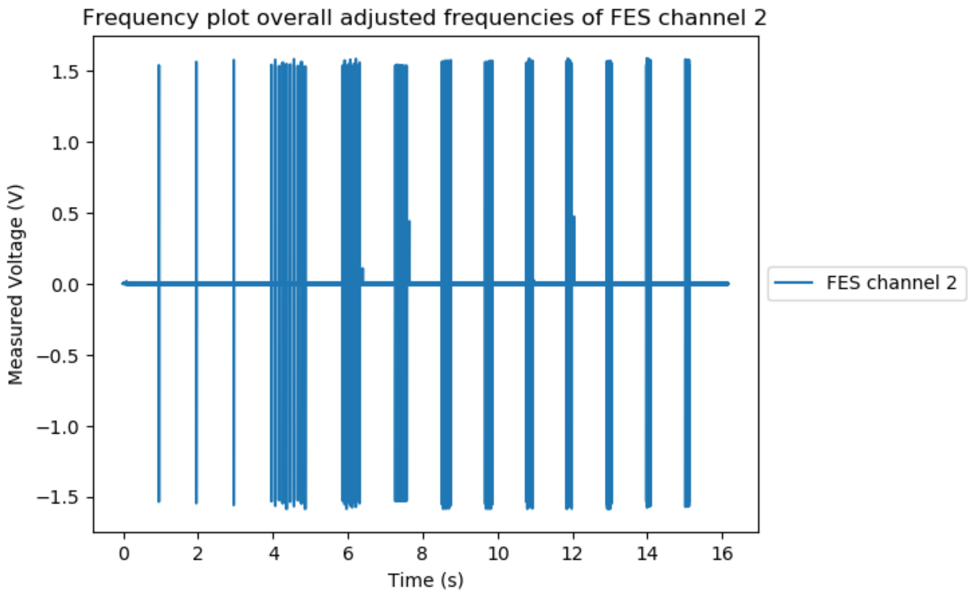



Figure 11: Measurement of the frequencies (left side 1 Hz - right side 100Hz) of FES channel 2

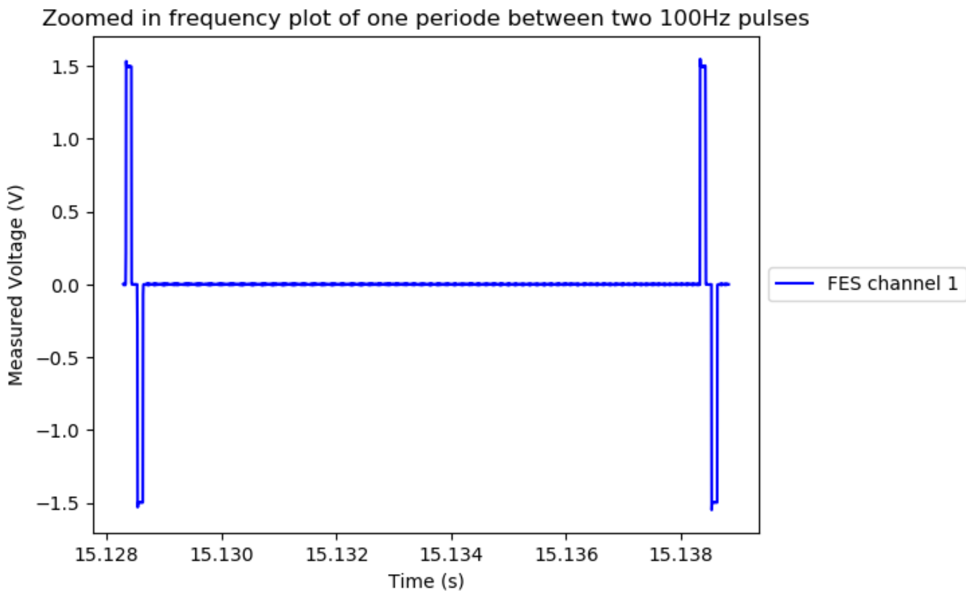



Figure 12: Zoomed in measurement of two 100Hz pulses of FES channel 1

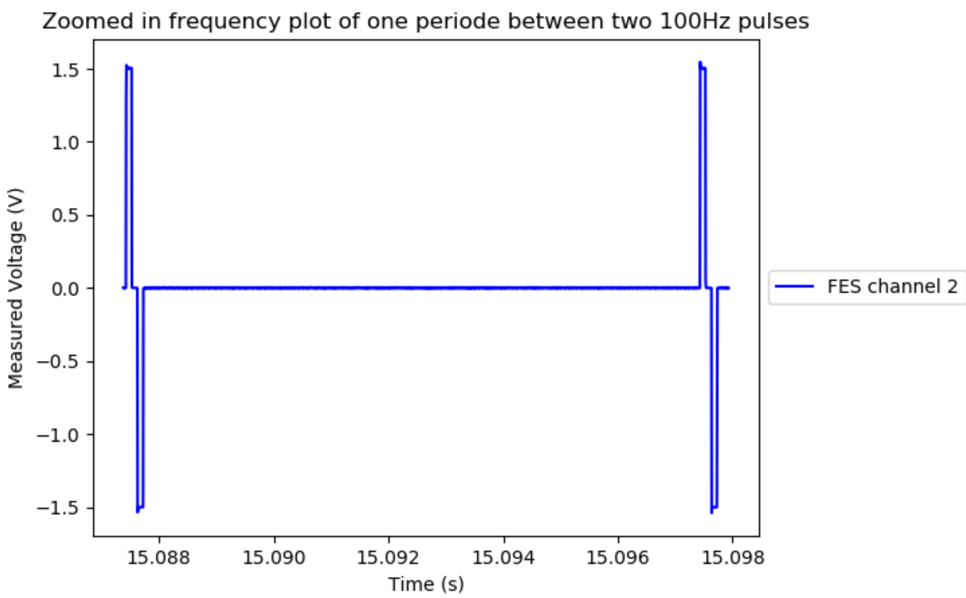



Figure 13: Zoomed in measurement of two 100Hz pulses of FES channel 2

## Test F: FES Durations Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement duration: 40 seconds
Sample rate: 200.000 Samples/s

### Description:

Checks the duration of the phase and inter-phase of both FES channels for 10 pulses each. The durations are adjusted after every tenth pulse. Therefore the voltage is measured on a resistor. This voltage is recorded and the durations are calculated from the samples.
Adjusted phase duration: 50$\mu$s, 100$\mu$s, 150$\mu$s, 200$\mu$s, 250$\mu$s, 300$\mu$s
Adjusted inter-phase duration: 50$\mu$s, 100$\mu$s, 200$\mu$s, 300$\mu$s, 400$\mu$s, 500$\mu$s, 600$\mu$s, 700$\mu$s, 800$\mu$s, 900$\mu$s, 1000$\mu$s

### Test Criteria:

The measured durations of both FES channels equals the adjusted duration ±5%, or ±5$\mu$s, whichever greater.
Phase duration limits: [45$\mu$s; 55$\mu$s], [95$\mu$s; 105$\mu$s], [142.5$\mu$s; 157.5$\mu$s], [190.0$\mu$s; 210.0$\mu$s], [237.5$\mu$s; 262.5$\mu$s], [285.0$\mu$s; 315.0$\mu$s].
Inter-phase duration limits: [45$\mu$s; 55$\mu$s], [95$\mu$s; 105$\mu$s], [190.0$\mu$s; 210.0$\mu$s], [285.0$\mu$s; 315.0$\mu$s], [380.0$\mu$s; 420.0$\mu$s], [475.0$\mu$s; 525.0$\mu$s], [570.0$\mu$s; 630.0$\mu$s], [665.0$\mu$s; 735.0$\mu$s], [760.0$\mu$s; 840.0$\mu$s], [855.0$\mu$s; 945.0$\mu$s], [950.0$\mu$s; 1050.0$\mu$s].

### Detailed Results:

| **Positive phase duration** | | | | | | |
|---|---|---|---|---|---|---|
| **Adjusted phase duration** | **Channel** | **Maximum ($\mu$s)** | **Minimum ($\mu$s)** | **Mean ($\mu$s)** | **Standard deviation ($\mu$s)** | **Test result** |
| 50$\mu$s | FES 1 | 50.0 | 45.0 | 47.5 | 2.5 | SUCCESS |
| | FES 2 | 50.0 | 45.0 | 47.0 | 2.449 | SUCCESS |
| 100$\mu$s | FES 1 | 100.0 | 95.0 | 96.5 | 2.291 | SUCCESS |
| | FES 2 | 100.0 | 95.0 | 97.0 | 2.449 | SUCCESS |
| 150$\mu$s | FES 1 | 150.0 | 145.0 | 147.0 | 2.449 | SUCCESS |
| | FES 2 | 150.0 | 145.0 | 146.5 | 2.291 | SUCCESS |
| 200$\mu$s | FES 1 | 200.0 | 195.0 | 197.5 | 2.5 | SUCCESS |
| | FES 2 | 200.0 | 195.0 | 197.0 | 2.449 | SUCCESS |
| 250$\mu$s | FES 1 | 250.0 | 245.0 | 247.0 | 2.449 | SUCCESS |
| | FES 2 | 250.0 | 245.0 | 247.0 | 2.449 | SUCCESS |
| 300$\mu$s | FES 1 | 300.0 | 295.0 | 297.0 | 2.449 | SUCCESS |
| | FES 2 | 300.0 | 295.0 | 297.0 | 2.449 | SUCCESS |

| **Negative phase duration** | | | | | | |
|---|---|---|---|---|---|---|
| **Adjusted phase duration** | **Channel** | **Maximum ($\mu$s)** | **Minimum ($\mu$s)** | **Mean ($\mu$s)** | **Standard deviation ($\mu$s)** | **Test result** |
| 50$\mu$s | FES 1 | 50.0 | 45.0 | 49.0 | 2.0 | SUCCESS |
| | FES 2 | 50.0 | 45.0 | 49.0 | 2.0 | SUCCESS |
| 100$\mu$s | FES 1 | 100.0 | 95.0 | 98.5 | 2.291 | SUCCESS |
| | FES 2 | 100.0 | 95.0 | 99.0 | 2.0 | SUCCESS |
| 150$\mu$s | FES 1 | 150.0 | 145.0 | 148.5 | 2.291 | SUCCESS |
| | FES 2 | 150.0 | 145.0 | 148.0 | 2.449 | SUCCESS |
| 200$\mu$s | FES 1 | 200.0 | 195.0 | 198.0 | 2.449 | SUCCESS |
| | FES 2 | 200.0 | 195.0 | 198.0 | 2.449 | SUCCESS |
| 250$\mu$s | FES 1 | 250.0 | 245.0 | 248.0 | 2.449 | SUCCESS |
| | FES 2 | 250.0 | 245.0 | 249.0 | 2.0 | SUCCESS |
| 300$\mu$s | FES 1 | 300.0 | 295.0 | 298.0 | 2.449 | SUCCESS |
| | FES 2 | 300.0 | 295.0 | 298.0 | 2.449 | SUCCESS |

| Inter-phase duration | | | | | | |
|---|---|---|---|---|---|---|
| **Adjusted inter-phase duration** | **Channel** | **Maximum ($\mu$s)** | **Minimum ($\mu$s)** | **Mean ($\mu$s)** | **Standard deviation ($\mu$s)** | **Test result** |
| 50$\mu$s | FES 1 | 55.0 | 50.0 | 52.5 | 2.5 | SUCCESS |
| | FES 2 | 55.0 | 50.0 | 52.0 | 2.449 | SUCCESS |
| 100$\mu$s | FES 1 | 105.0 | 100.0 | 102.5 | 2.5 | SUCCESS |
| | FES 2 | 105.0 | 100.0 | 103.0 | 2.449 | SUCCESS |
| 200$\mu$s | FES 1 | 205.0 | 200.0 | 202.5 | 2.5 | SUCCESS |
| | FES 2 | 205.0 | 200.0 | 202.5 | 2.5 | SUCCESS |
| 300$\mu$s | FES 1 | 305.0 | 300.0 | 302.5 | 2.5 | SUCCESS |
| | FES 2 | 305.0 | 300.0 | 303.0 | 2.449 | SUCCESS |
| 400$\mu$s | FES 1 | 405.0 | 400.0 | 402.5 | 2.5 | SUCCESS |
| | FES 2 | 405.0 | 400.0 | 402.0 | 2.449 | SUCCESS |
| 500$\mu$s | FES 1 | 505.0 | 500.0 | 502.5 | 2.5 | SUCCESS |
| | FES 2 | 505.0 | 500.0 | 503.0 | 2.449 | SUCCESS |
| 600$\mu$s | FES 1 | 605.0 | 600.0 | 602.5 | 2.5 | SUCCESS |
| | FES 2 | 605.0 | 600.0 | 603.0 | 2.449 | SUCCESS |
| 700$\mu$s | FES 1 | 705.0 | 700.0 | 702.5 | 2.5 | SUCCESS |
| | FES 2 | 705.0 | 700.0 | 703.0 | 2.449 | SUCCESS |
| 800$\mu$s | FES 1 | 805.0 | 800.0 | 803.0 | 2.449 | SUCCESS |
| | FES 2 | 805.0 | 800.0 | 802.0 | 2.449 | SUCCESS |
| 900$\mu$s | FES 1 | 905.0 | 900.0 | 902.5 | 2.5 | SUCCESS |
| | FES 2 | 905.0 | 900.0 | 903.0 | 2.449 | SUCCESS |
| 1000$\mu$s | FES 1 | 1005.0 | 1000.0 | 1002.5 | 2.5 | SUCCESS |
| | FES 2 | 1005.0 | 1000.0 | 1002.5 | 2.5 | SUCCESS |

**Plots:**

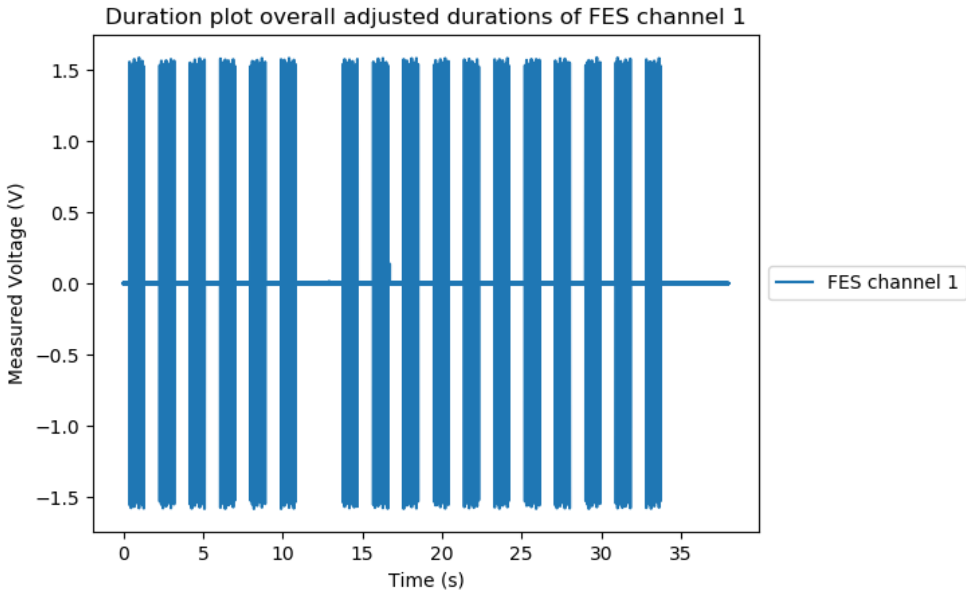



Figure 14: Measurement of the durations (phase duration: before pause; inter-phase duration: after pause) of FES channel 1

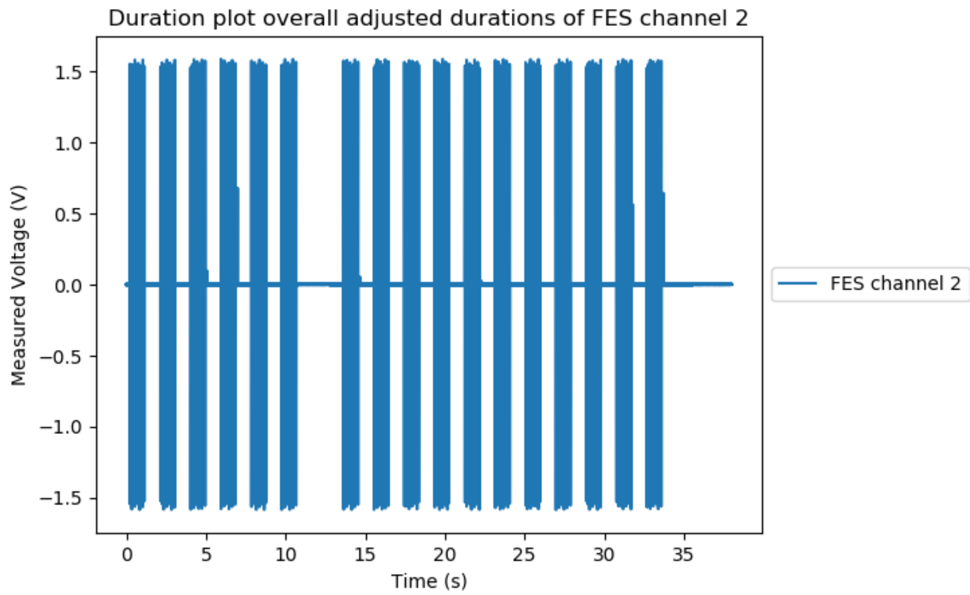



Figure 15: Measurement of the durations (phase duration: before pause; inter-phase duration: after pause) of FES channel 2

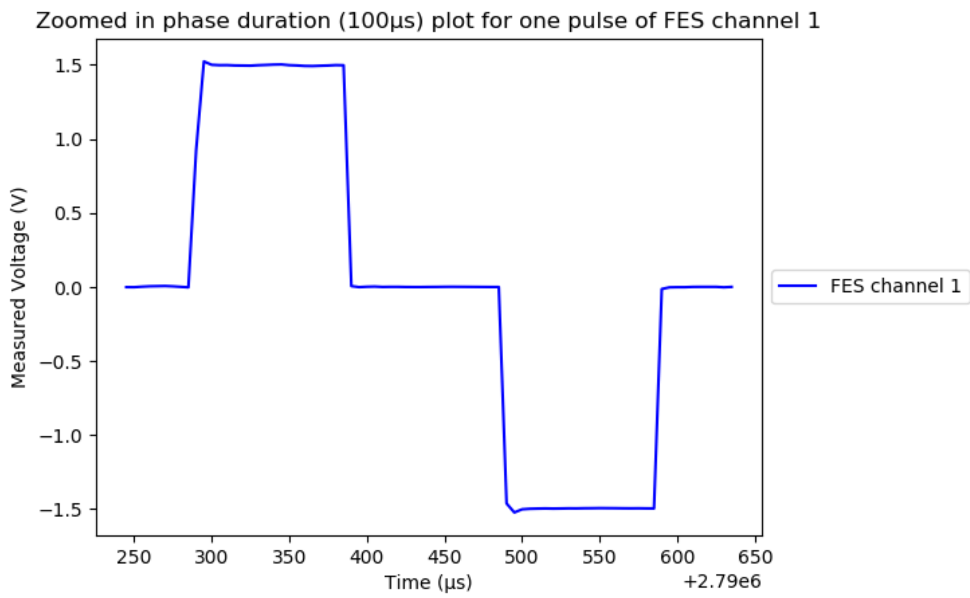



Figure 16: Zoomed in measurement of the phase duration (100$\mu$s) of FES channel 1

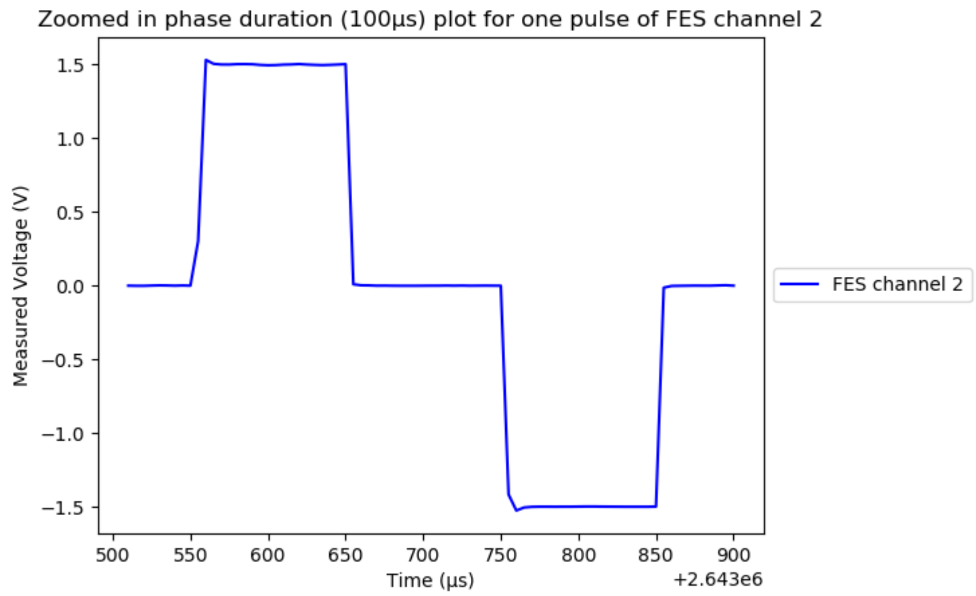



Figure 17: Zoomed in measurement of the phase duration (100$\mu$s) of FES channel 2

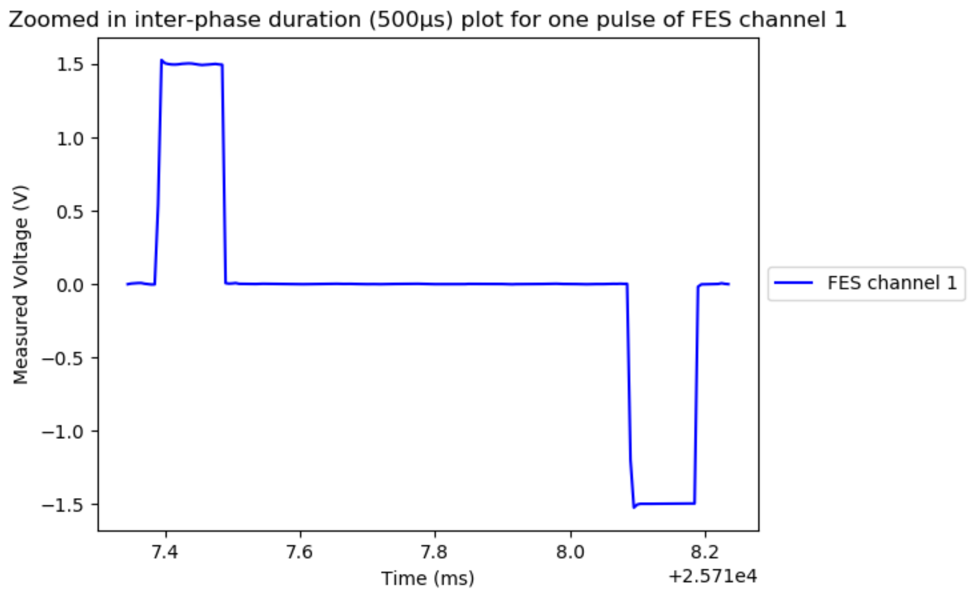



Figure 18: Zoomed in measurement of the inter-phase duration (500$\mu$s) of FES channel 1

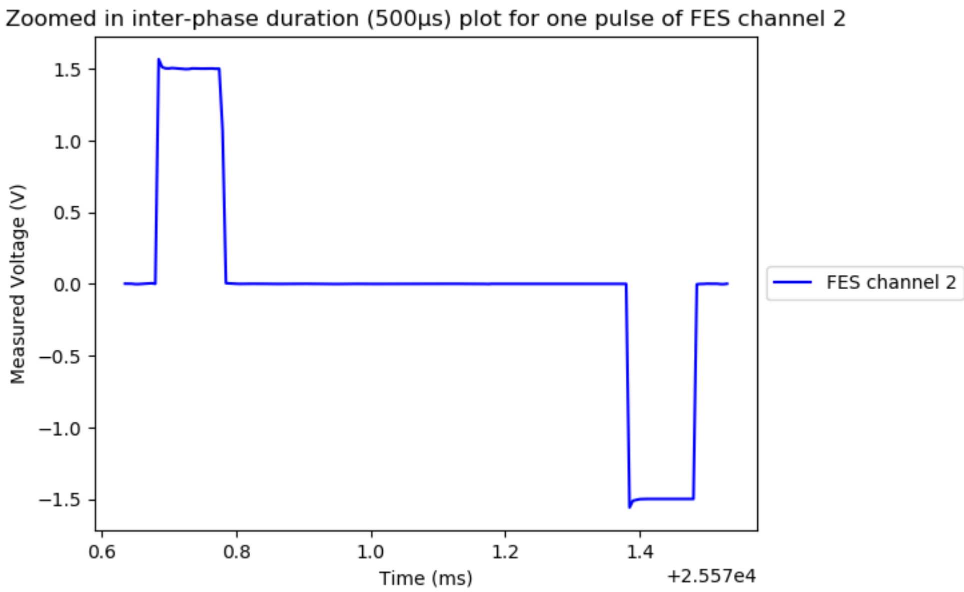



Figure 19: Zoomed in measurement of the inter-phase duration (500$\mu$s) of FES channel 2

## Test J: STIMbox Voltage Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement duration: 50 seconds
Sample rate: 1.000 Samples/s

### Description:

Checks the voltage of each channel of the STIMbox. Therefore, the voltage is measured on a resistance and then calculated.

### Test Criteria:

The measured voltage is in between the range of 5V ±5% [4.75V; 5.25V].

### Detailed Results:

| Channel | Maximum of voltage (V) | Minimum of voltage (V) | Mean of voltage (V) | Standard deviation of voltage (V) | Result |
|---|---|---|---|---|---|
| STIMbox 1 Pin 1 | 4.778 | 4.767 | 4.777 | 0.001 | SUCCESS |
| STIMbox 1 Pin 2 | 4.81 | 4.767 | 4.781 | 0.004 | SUCCESS |
| STIMbox 1 Pin 3 | 4.81 | 4.767 | 4.779 | 0.005 | SUCCESS |
| STIMbox 1 Pin 4 | 4.81 | 4.767 | 4.778 | 0.004 | SUCCESS |
| STIMbox 1 Pin 5 | 4.81 | 4.76 | 4.776 | 0.006 | SUCCESS |
| STIMbox 2 Pin 1 | 4.811 | 4.76 | 4.776 | 0.005 | SUCCESS |
| STIMbox 2 Pin 2 | 4.811 | 4.76 | 4.777 | 0.005 | SUCCESS |
| STIMbox 2 Pin 3 | 4.811 | 4.76 | 4.776 | 0.005 | SUCCESS |
| STIMbox 2 Pin 4 | 4.811 | 4.76 | 4.777 | 0.005 | SUCCESS |
| STIMbox 2 Pin 5 | 4.811 | 4.754 | 4.775 | 0.006 | SUCCESS |
| STIMbox 3 Pin 1 | 4.811 | 4.754 | 4.774 | 0.007 | SUCCESS |
| STIMbox 3 Pin 2 | 4.811 | 4.754 | 4.775 | 0.006 | SUCCESS |
| STIMbox 3 Pin 3 | 4.811 | 4.754 | 4.775 | 0.006 | SUCCESS |
| STIMbox 3 Pin 4 | 4.811 | 4.754 | 4.775 | 0.006 | SUCCESS |
| STIMbox 3 Pin 5 | 4.811 | 4.754 | 4.775 | 0.006 | SUCCESS |
| STIMbox 3 Pin 6 | 4.811 | 4.754 | 4.774 | 0.006 | SUCCESS |

**Plots:**

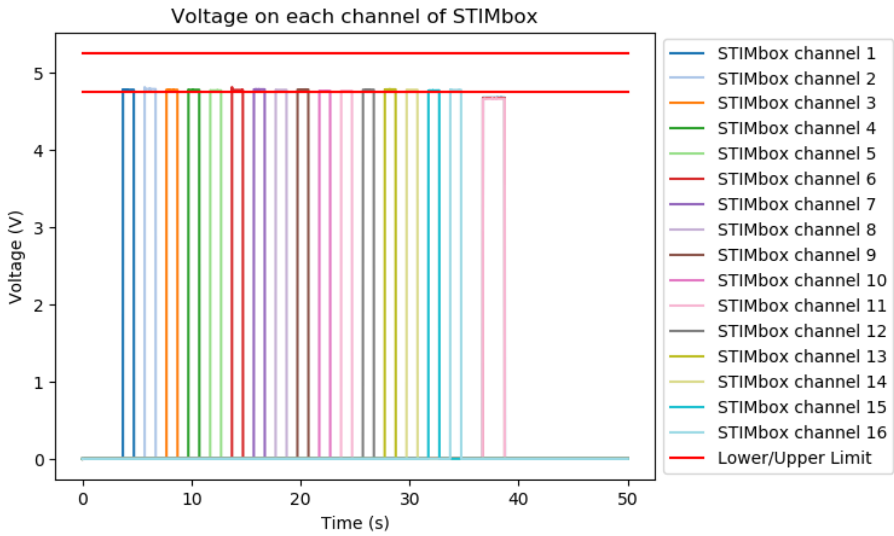



Figure 20: Measurement of the voltage of STIMbox

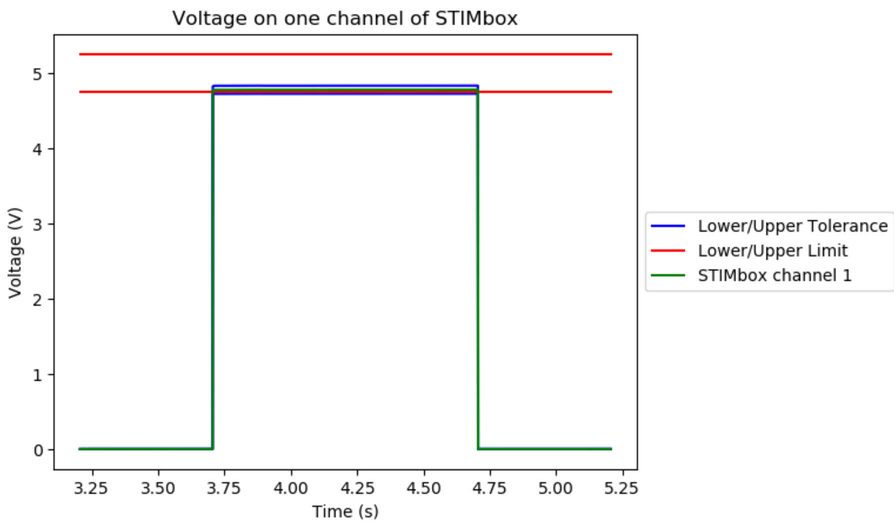



Figure 21: Measurement of the voltage on one channel of the STIMbox

## Test K: STIMbox Current Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement duration: 50 seconds
Sample rate: 1.000 Samples/s

### Description:

Checks the current of each channel of the STIMbox. Therefore, the voltage is measured on a shunt resistance and then the current is calculated.

### Test Criteria:

The measured current is in between the range of 150mA ±5% [142.5mA; 157.5mA].

### Detailed Results:

| Channel | Maximum of current (mA) | Minimum of current (mA) | Mean of current (mA) | Standard deviation of current (mA) | Result |
|---|---|---|---|---|---|
| STIMbox 1 Pin 1 | 144.791 | 144.451 | 144.748 | 0.021 | SUCCESS |
| STIMbox 1 Pin 2 | 145.756 | 144.451 | 144.878 | 0.132 | SUCCESS |
| STIMbox 1 Pin 3 | 145.756 | 144.44 | 144.818 | 0.138 | SUCCESS |
| STIMbox 1 Pin 4 | 145.756 | 144.44 | 144.794 | 0.127 | SUCCESS |
| STIMbox 1 Pin 5 | 145.756 | 144.232 | 144.732 | 0.168 | SUCCESS |
| STIMbox 2 Pin 1 | 145.778 | 144.232 | 144.718 | 0.158 | SUCCESS |
| STIMbox 2 Pin 2 | 145.778 | 144.232 | 144.745 | 0.161 | SUCCESS |
| STIMbox 2 Pin 3 | 145.778 | 144.232 | 144.738 | 0.152 | SUCCESS |
| STIMbox 2 Pin 4 | 145.778 | 144.232 | 144.748 | 0.146 | SUCCESS |
| STIMbox 2 Pin 5 | 145.778 | 144.057 | 144.706 | 0.187 | SUCCESS |
| STIMbox 3 Pin 1 | 145.778 | 144.057 | 144.675 | 0.204 | SUCCESS |
| STIMbox 3 Pin 2 | 145.778 | 144.057 | 144.682 | 0.197 | SUCCESS |
| STIMbox 3 Pin 3 | 145.778 | 144.057 | 144.698 | 0.196 | SUCCESS |
| STIMbox 3 Pin 4 | 145.778 | 144.057 | 144.7 | 0.19 | SUCCESS |
| STIMbox 3 Pin 5 | 145.778 | 144.057 | 144.682 | 0.195 | SUCCESS |
| STIMbox 3 Pin 6 | 145.778 | 144.057 | 144.68 | 0.189 | SUCCESS |

**Plots:**

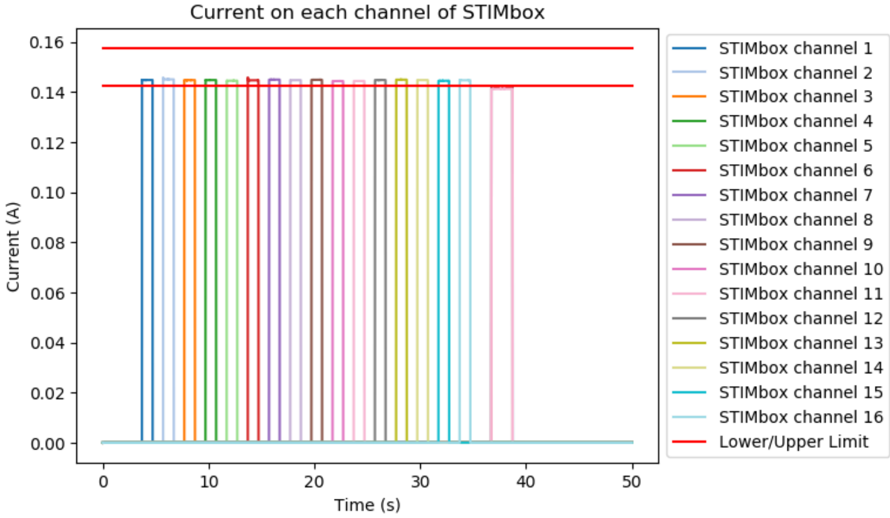



Figure 22: Measurement of the current of STIMbox

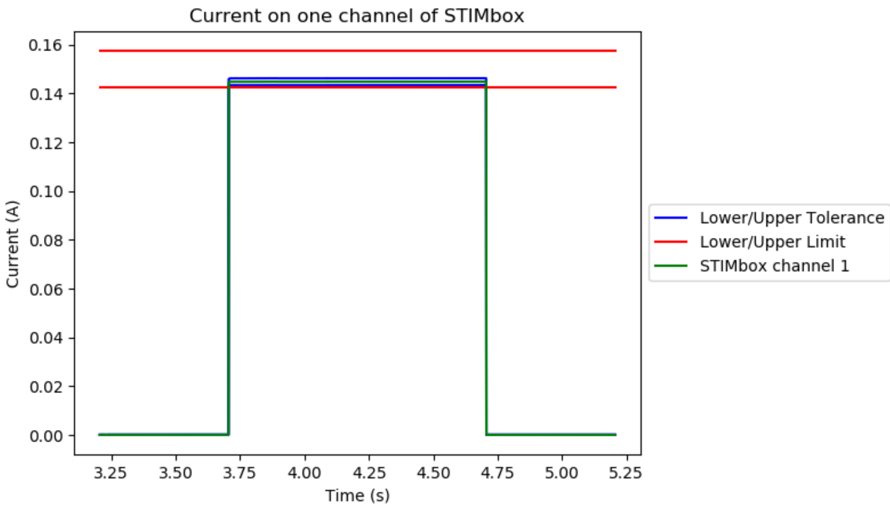



Figure 23: Measurement of the current on one channel of the STIMbox

## Test L: STIMbox Timing Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement duration: 20 seconds
Sample rate: 250 Samples/s

### Description:

Checks the time difference between each Channel 1 of the STIMbox and g.Nauitlus DIN. Therefore, the voltage is measured/recorded and then the time difference is calculated.

### Test Criteria:

The measured time difference between the STIMbox and g.Nautilus DIN is the range of ±10ms.

### Detailed Results:

| **Time difference (ms)** | g.Nautilus DI 1 | g.Nautilus DI 2 | g.Nautilus DI 3 | g.Nautilus DI 1-3 |
|---|---|---|---|---|
| STIMbox 1 Pin 1 | 4.0 | - | - | 4.0 |
| STIMbox 2 Pin 1 | - | 4.0 | - | 4.0 |
| STIMbox 3 Pin 1 | - | - | 8.0 | 4.0 |

### Plot:

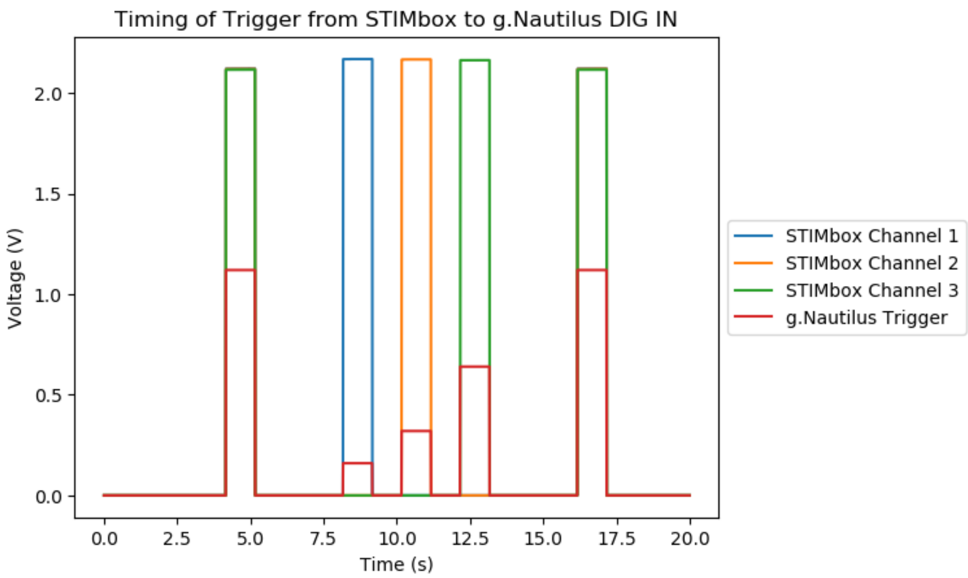



Figure 24: Measurement of the time difference between the STIMbox and g.Nautilus

## Test M: g.Nautilus Connection Test

Serial: CU-XXXX.XX.XX
Final result: SUCCESS

### Description:

Checks if the g.Nautilus base station can connect with its headset and reads out the serial numbers of both.

### Test Criteria:

If the devices are connected, read the serial numbers of both.

### Detailed Results:

| | **Serial Number** |
|---|---|
| g.Nautilus base station | NB-2017.09.50 |
| g.Nautilus headset | NA-2017.09.50 |

## Test O: Power Supply for the medical computer

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement Duration: 1 second
Sampling Rate: 1.000 Samples/s

### Description:

Voltage measurement of the power supply for the medical computer.

### Test Criteria:

The voltage of the power supply is in between the range of 19V ±3% [18.43V; 19.57V].

### Detailed Results:

| **Voltage [V]** | | | |
|---|---|---|---|
| Minimum | Maximum | Mean | Standard Deviation |
| 19.0786 | 19.1148 | 19.0971 | 0.006721 |

### Plot:

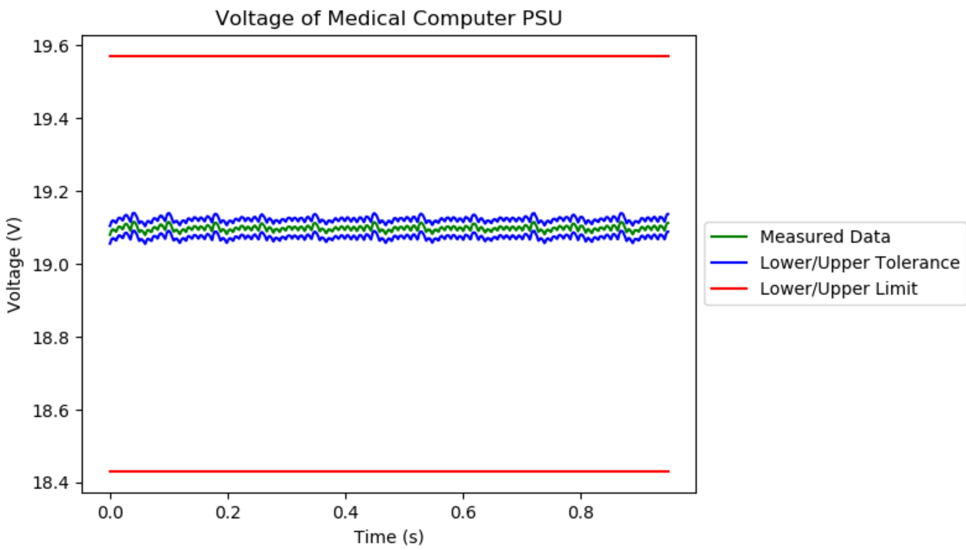



Figure 25: Power Supply for the Medical Computer

## Test P: Functionality of g.VIBROstims

Serial: CU-XXXX.XX.XX
Final result: SUCCESS
Measurement Duration: 5 seconds (4 seconds for the calculations)
Sampling Rate: 100.000 Samples/s

### Description:

Powers one after another g.VIBROstim for five seconds and checks if it is vibrating constantly. Also measures the amplitude in % against a calibrated g.VIBROstim.

### Test Criteria:

The amplitude has to be at least 70% of the amplitude of the calibrated g.VIBROstim. The measured frequency also has to be between 150Hz and 250 Hz.

### Detailed Results:

| **Values** | **g.VIBROstim 1** | **g.VIBROstim 2** | **g.VIBROstim 3** |
|---|---|---|---|
| Minimal maximal value | 91.18% | 89.5% | 103.79% |
| Maximum minimal value | -91.35% | -89.01% | -100.52% |
| Mean of maxima | 94.62% | 93.22% | 105.74% |
| Mean of minima | -95.41% | -92.3% | -102.14% |
| Mean amplitude | ±95.02% | ±92.76% | ±103.94% |
| Frequency | 197.06Hz | 183.16Hz | 188.67Hz |
| Test result | SUCCESS | SUCCESS | SUCCESS |

### Plot:

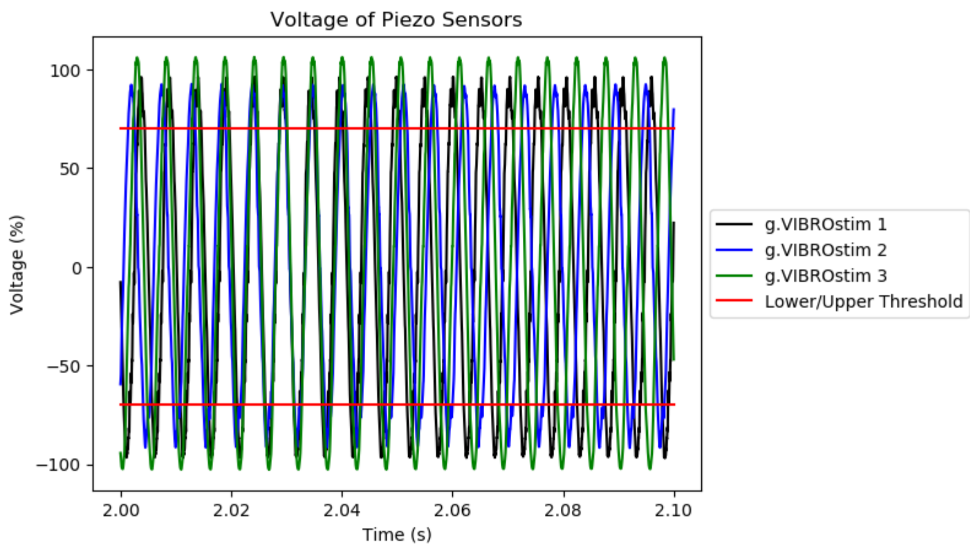



Figure 26: Measurement of g.VIBROstim's

### Temperature Plot:

Serial: CU-XXXX.XX.XX
Measurement Duration: 0.5 seconds each
Sampling Rate: 100 Samples/s

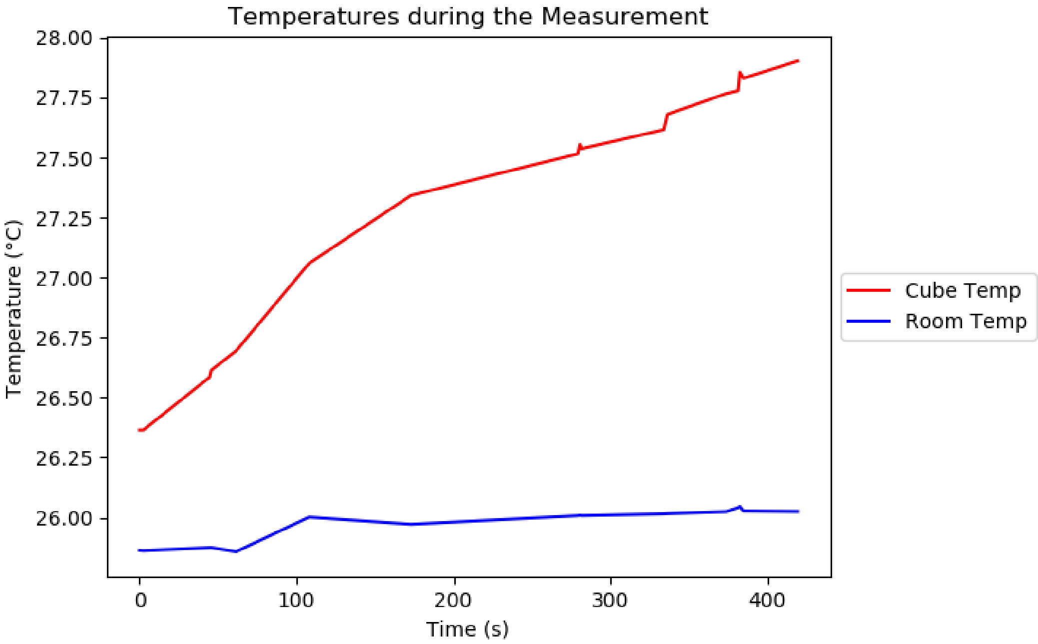



Figure 27: Temperatures