



Fachgebiet
Regelungssysteme



FES Testbed for Multi-Channel Transcutaneous Stimulation Systems

Markus Valtin and **Thomas Schauer**

Technische Universität Berlin, Germany

Disclosure

- Markus Valtin works for the HASOMED GmbH, which cooperates in the iTrust project with the SensorStim Neurotechnology GmbH.
- Dr. Thomas Schauer is a co-founder of the startup SensorStim Neurotechnology GmbH.

What is the FES Testbed?

In General

“A testbed is a platform for conducting rigorous, transparent, and replicable tests.”

What is the FES Testbed?

In General

"A testbed is a platform for conducting rigorous, transparent, and replicable tests."

FES Stimulation Systems

- realistic model of the electrical behavior of the human skin
- measurement of the stimulation voltage and current
- high measurement resolution
- high temporal resolution
- no interference with the stimulation

What is the FES Testbed?

In General

"A testbed is a platform for conducting rigorous, transparent, and replicable tests."

FES Stimulation Systems

- realistic model of the electrical behavior of the human skin
 ⇒ FES Skin Model and Audit Model
- measurement of the stimulation voltage and current ⇒ both
- high measurement resolution ⇒ $5\mu A / 5mV$
- high temporal resolution ⇒ 1 MSamples → $1\mu S$ resolution
- no interference with the stimulation ⇒ galvanic isolated

Use Case

Main Question to Answer

How to verify the correct timing of the stimulation pulse and the correct pulse form for every stimulation pulse?

Use Case

Main Question to Answer

How to verify the correct timing of the stimulation pulse and the correct pulse form for every stimulation pulse?

- (I) clean software design (correct handling of race conditions, abstraction, separation of concerns, use of hardware features, ...)

Use Case

Main Question to Answer

How to verify the correct timing of the stimulation pulse and the correct pulse form for every stimulation pulse?

- (I) clean software design (correct handling of race conditions, abstraction, separation of concerns, use of hardware features, ...)
- (II) specific tests, e.g. in dept measurements on one pulse via oscilloscope and extrapolation the result to every pulse

Use Case

Main Question to Answer

How to verify the correct timing of the stimulation pulse and the correct pulse form for every stimulation pulse?

- (I) clean software design (correct handling of race conditions, abstraction, separation of concerns, use of hardware features, ...)
- (II) specific tests, e.g. in dept measurements on one pulse via oscilloscope and extrapolation the result to every pulse
- (III) long term tests (several hours) in real FES use cases

Use Case

Main Question to Answer

How to verify the correct timing of the stimulation pulse and the correct pulse form for every stimulation pulse?

- (I) clean software design (correct handling of race conditions, abstraction, separation of concerns, use of hardware features, ...)
- (II) specific tests, e.g. in dept measurements on one pulse via oscilloscope and extrapolation the result to every pulse
- (III) long term tests (several hours) in real FES use cases

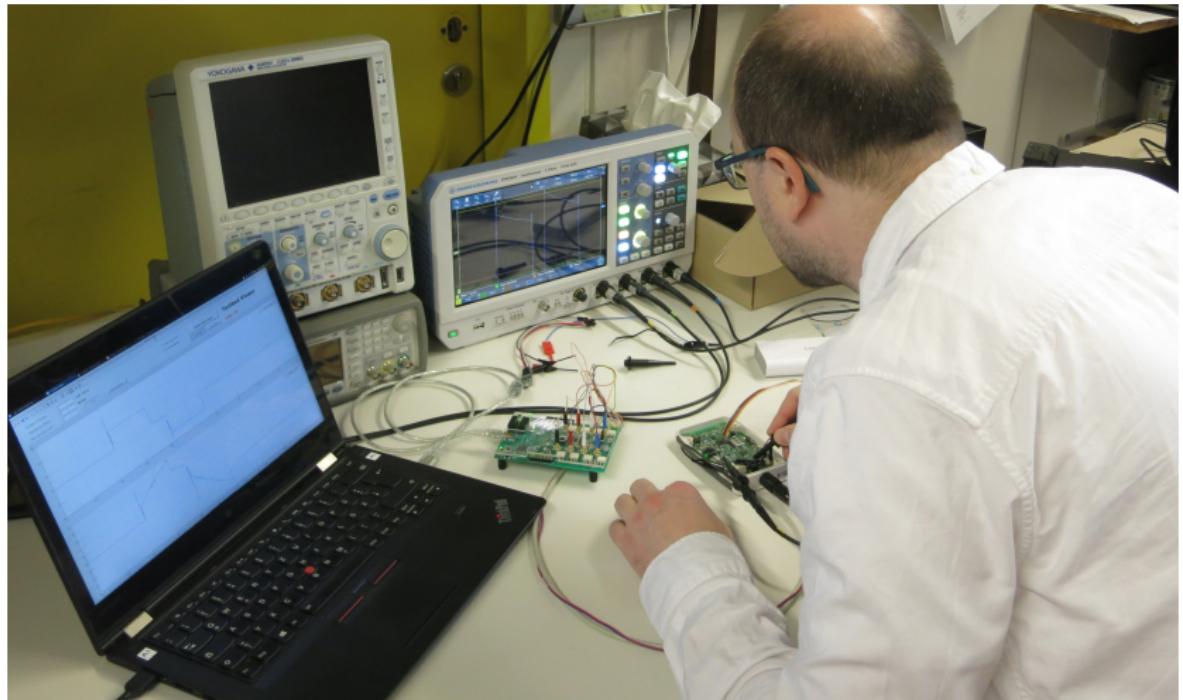
⇒ automated testbed needed

Secondary Use Cases

Other Use Cases

- general development, e.g. during design and optimization of a stimulator

Secondary Use Cases



Secondary Use Cases

Other Use Cases

- general development, e.g. during design and optimization of a stimulator
- set up of new stimulation wave forms

Secondary Use Cases

Other Use Cases

- general development, e.g. during design and optimization of a stimulator
- set up of new stimulation wave forms
- as a dummy load, e.g. to check a feedback control loop

Secondary Use Cases

Other Use Cases

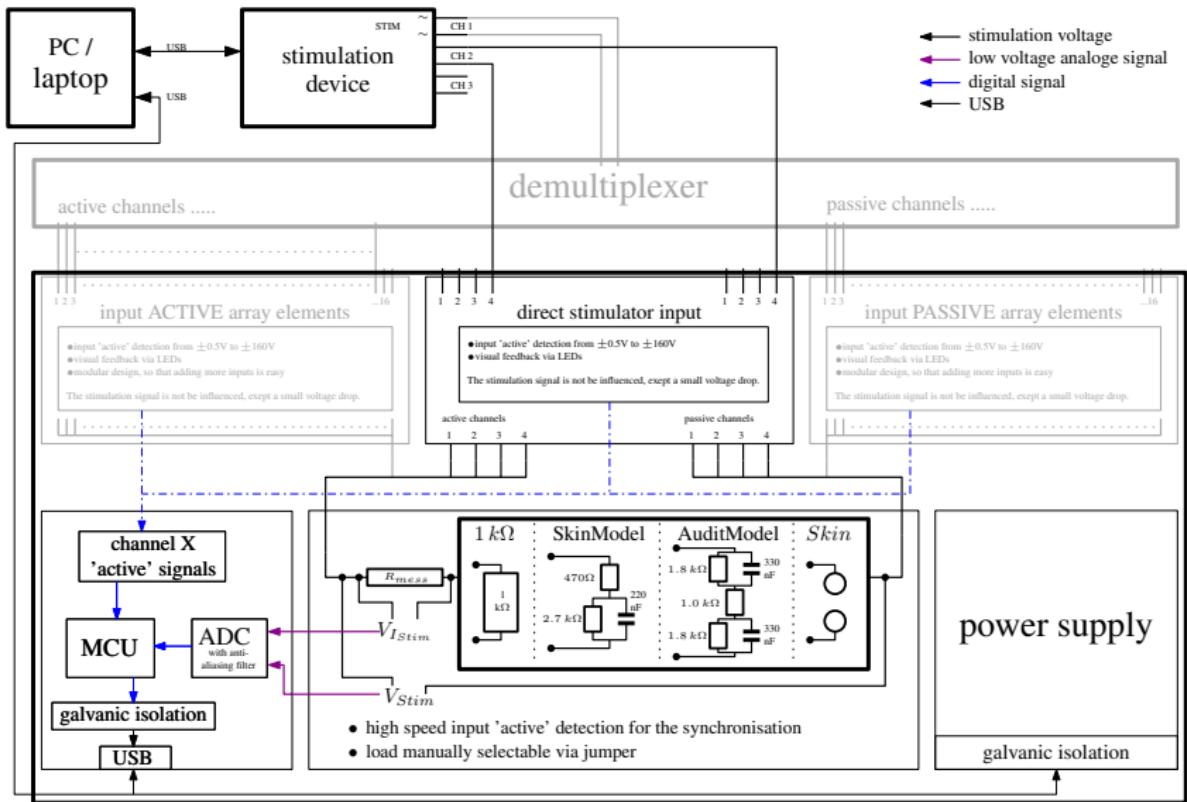
- general development, e.g. during design and optimization of a stimulator
- set up of new stimulation wave forms
- as a dummy load, e.g. to check a feedback control loop
- monitoring real FES sessions, e.g. for the saturation of the stimulation voltage, timing, and pulse form, or hardware compliance

Secondary Use Cases

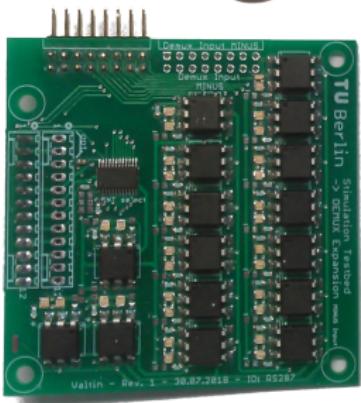
Other Use Cases

- general development, e.g. during design and optimization of a stimulator
- set up of new stimulation wave forms
- as a dummy load, e.g. to check a feedback control loop
- monitoring real FES sessions, e.g. for the saturation of the stimulation voltage, timing, and pulse form, or hardware compliance
- research involving the current or voltage response, like e.g. evaluating skin models

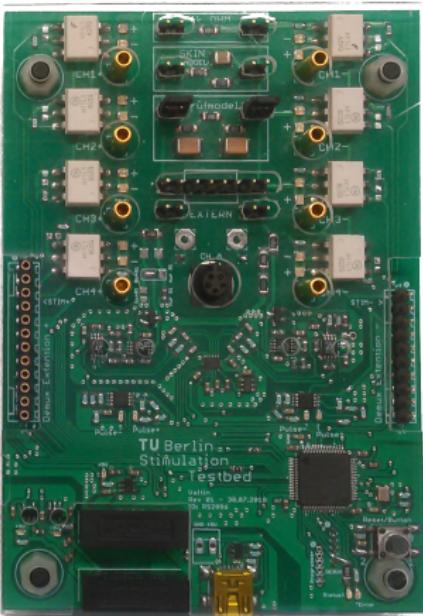
System Overview



Hardware

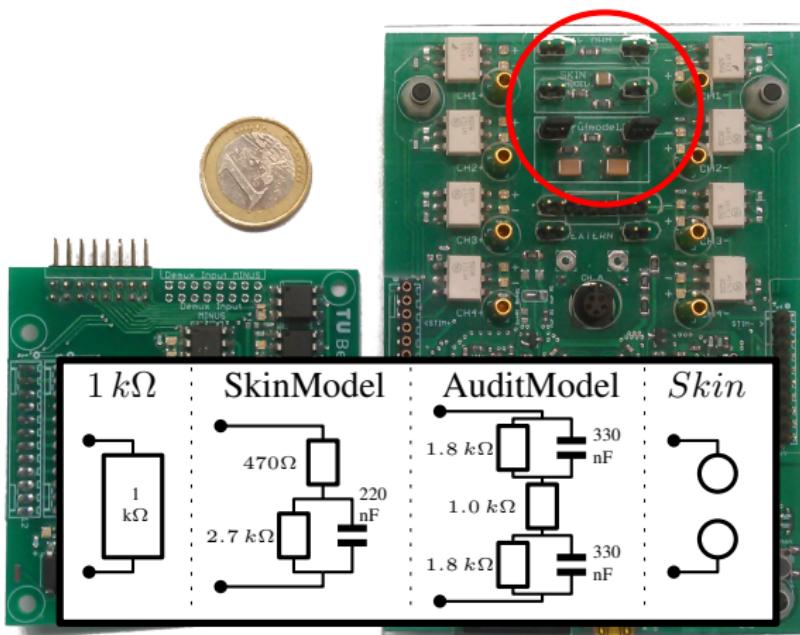


demultiplexer extension,
adds 16 additional
channels, stackable



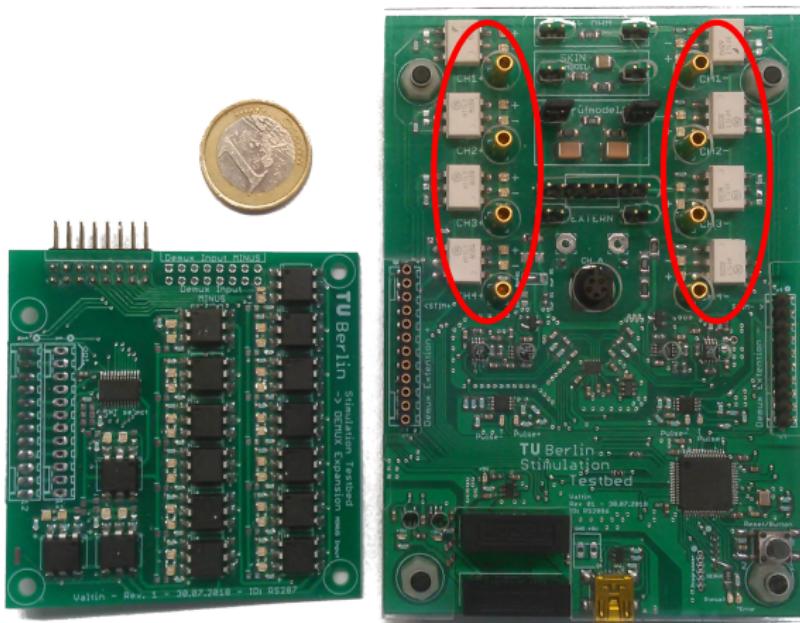
main board with 4 input
channels

Hardware



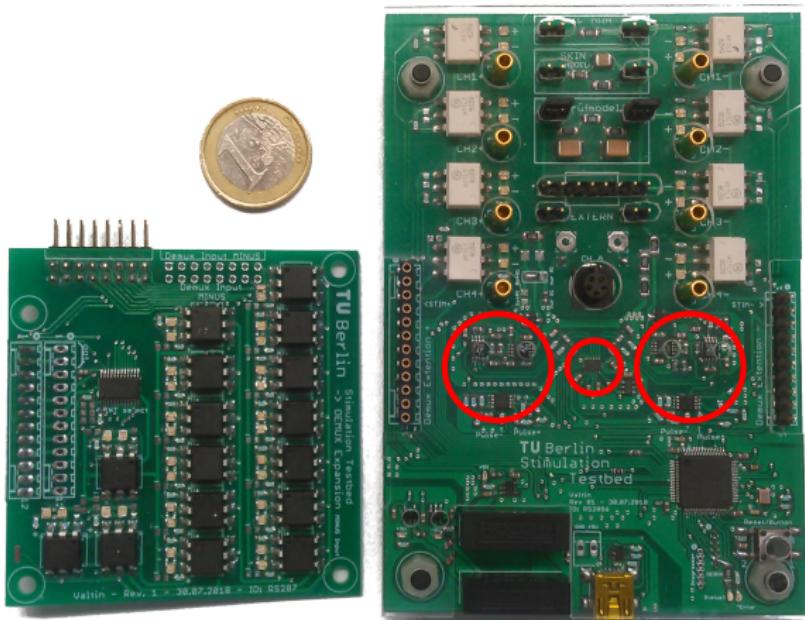
- skin replacement model selection

Hardware



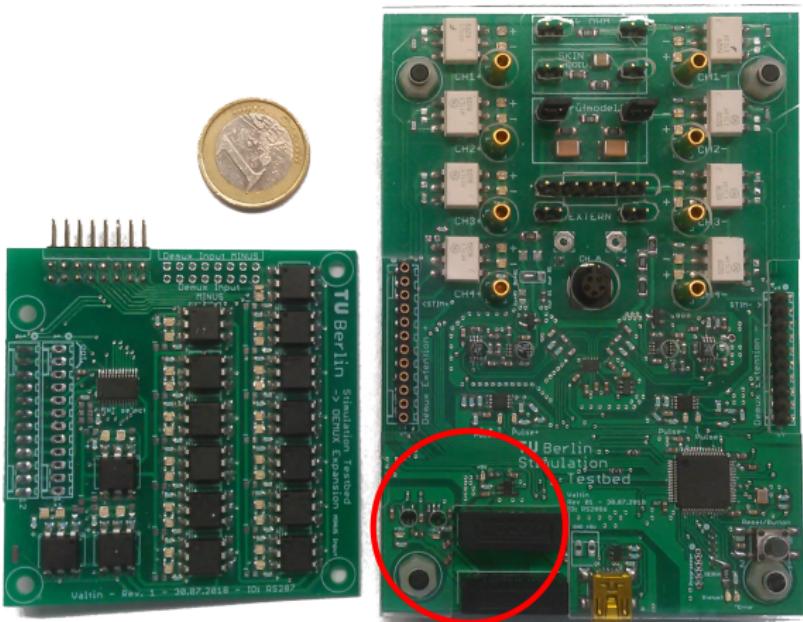
- 4 input channels with channel detection via optocoupler and visual feedback via LEDs

Hardware



- anti-aliasing filter, signal preconditioning, and 16 bit ADC

Hardware



- galvanic isolated power supply

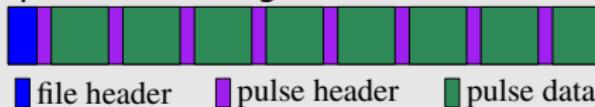
Principle of Operation

- A small data logger program initiates the system with a few configuration parameter, like the voltage or current thresholds, and starts the measurement.
 - The current shunt voltage and stimulation voltage is continuous measured with 1 MSamples/sec.
 - If the voltage or current exceeds configurable thresholds, a data block with 1000 samples for both channels is send to the PC.
 - The data logger program receives these samples and saves the 16 bit raw data, as well as a global block counter into a log file. Two or more consecutive data blocks are treated as one pulse.
-
- The binary log file is parsed via MatlabTM into a Matlab struct with each pulse as independent dataset.
 - These pulses can now processed with other Matlab scripts and/or visually inspected.

Software Design

1. Data Logger / Data Recorder

- simple logging program, without any post-processing was chosen for simplicity
- program initializes the testbed, receives the data and saves the data for each pulse into a log file



- each pulse is also send via TCP connection to other connected programs, e.g. a live monitor
- can run on low-powered hardware like the Raspberry Pi

Software Design

2. Post-Processing in Matlab

- chosen because of it's flexibility and ease of use
- modular, function based design with functions for:
 - read the log file and convert the raw data into a Matlab structure
 - pulse detection and feature extraction, like e.g. start index, pulse parts indexes, current, ...
 - statistic data generators
 - data management
 - data visualization
 - Matlab GUI for efficient manual inspection

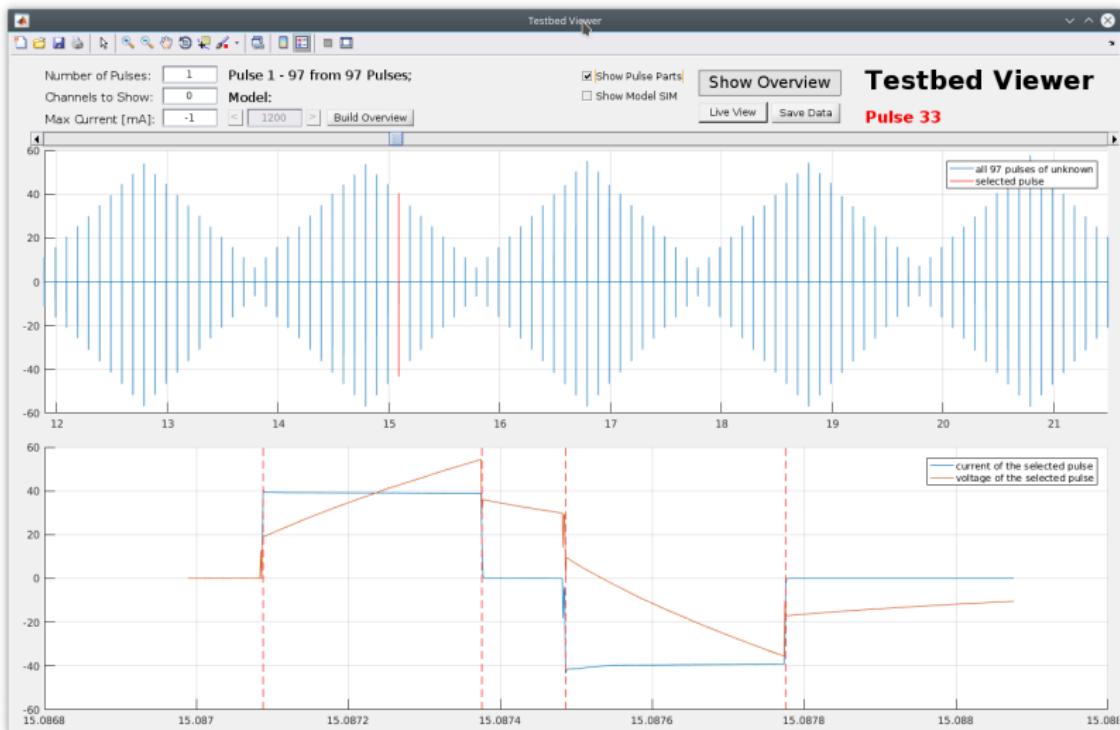
dataset struct:

nPulses: 97
pulseRaw: [1x97 struct]
setup: [1x1 struct]
pulse: [1x97 struct] ->
model: [1x1 struct]

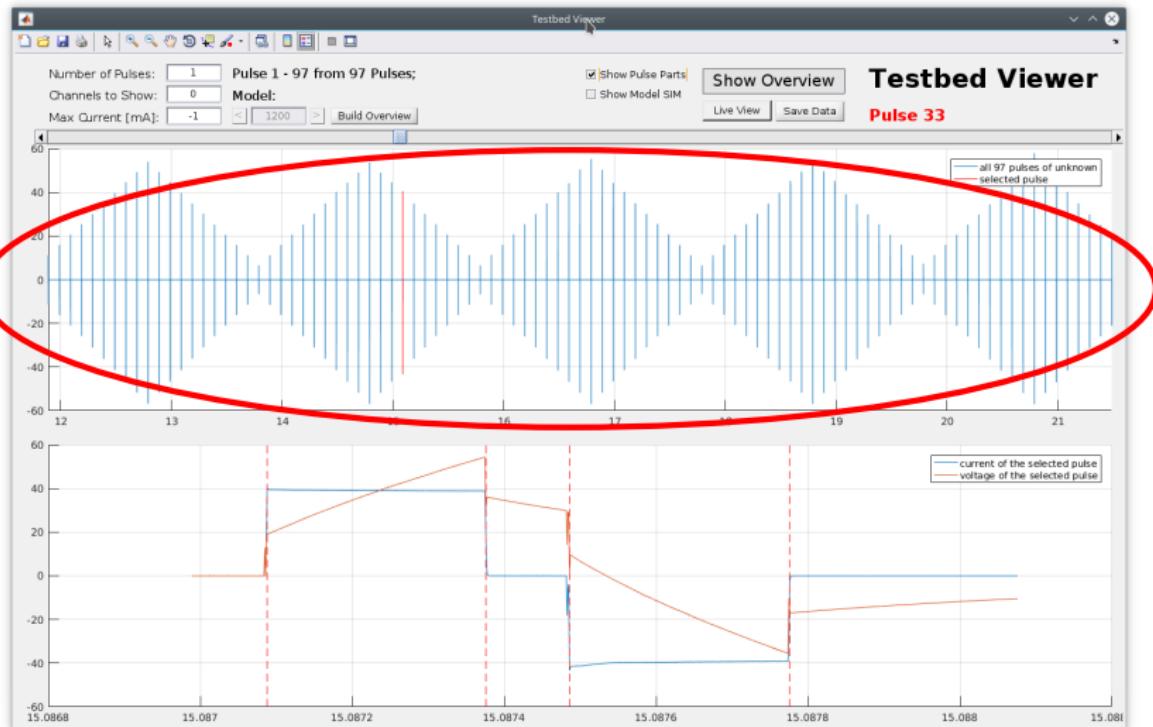
one pulse struct:

info: [1x1 struct]
nSamples: 1088
time: [1088x1 double]
stimCurrent: [1088x1 double]
stimVoltage: [1088x1 double]
nSample: [1088x1 double]
stats: [1x1 struct]
model: [1x1 struct]

Matlab GUI - Overview

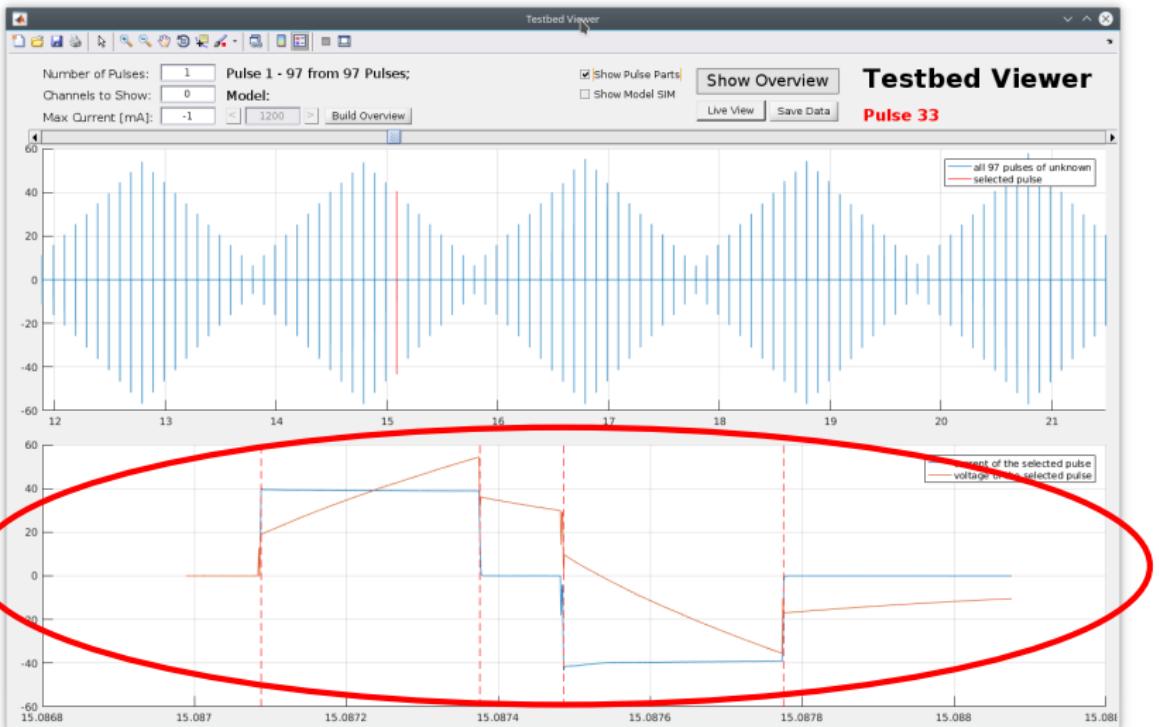


Matlab GUI - Overview



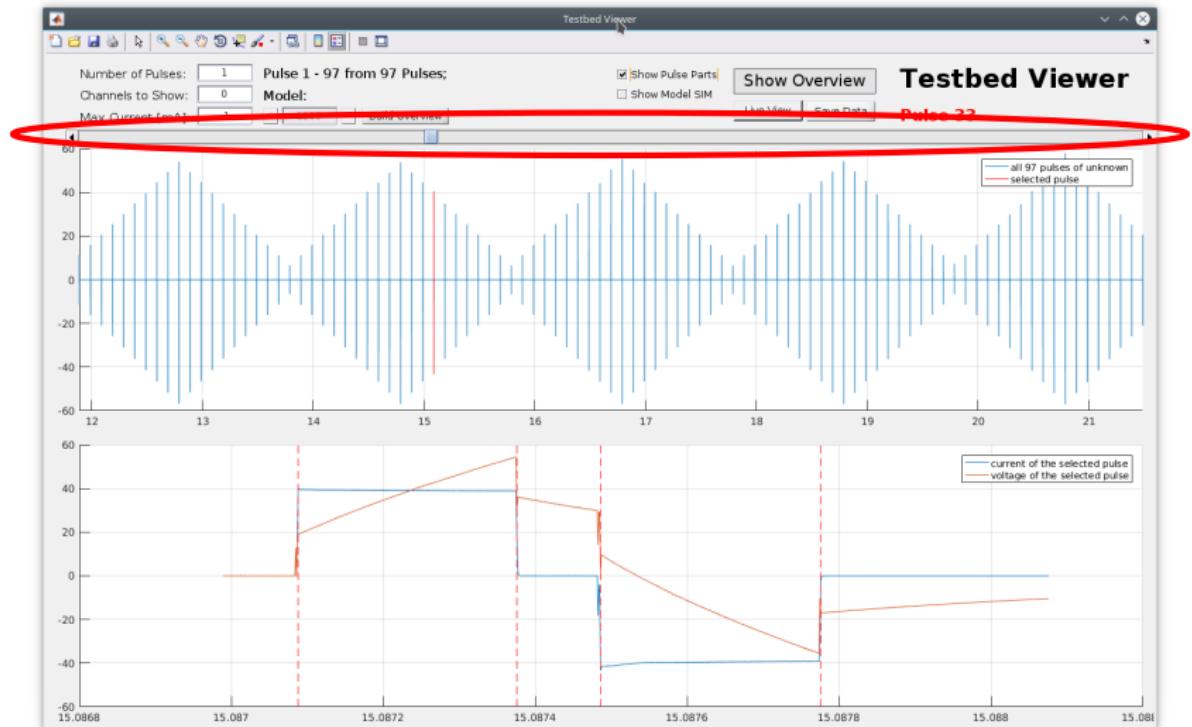
- time based overview with the selected pulse in red

Matlab GUI - Overview



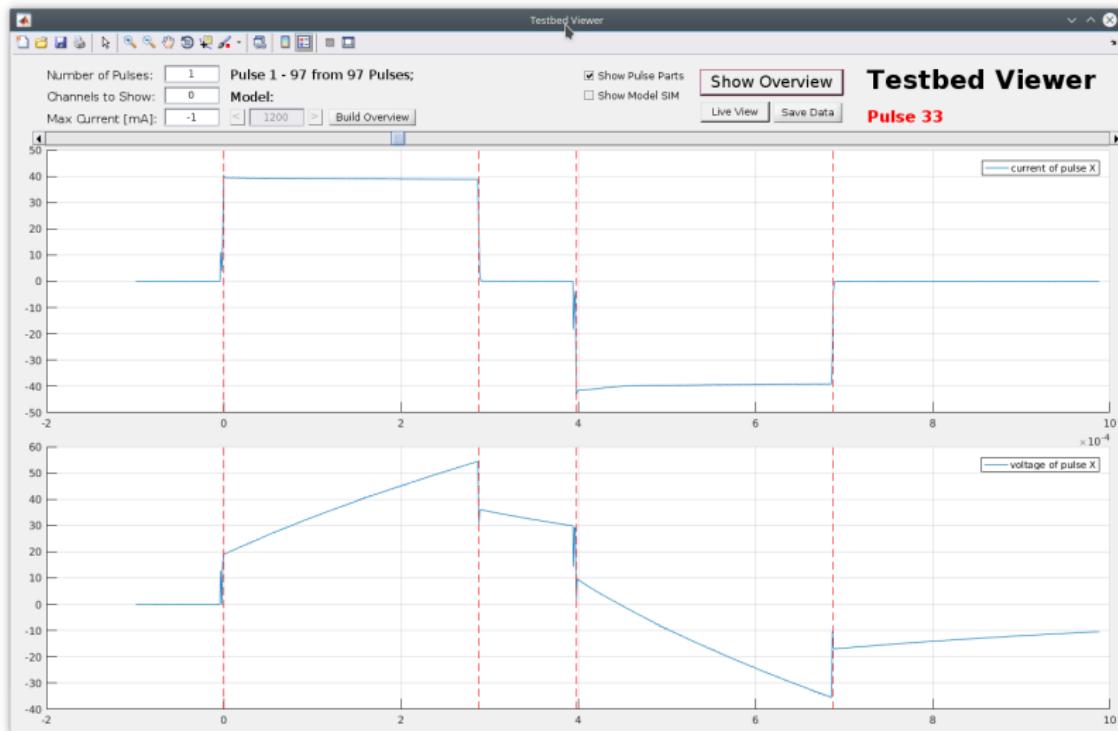
- selected stimulation pulse (current and voltage + pulse parts)

Matlab GUI - Overview

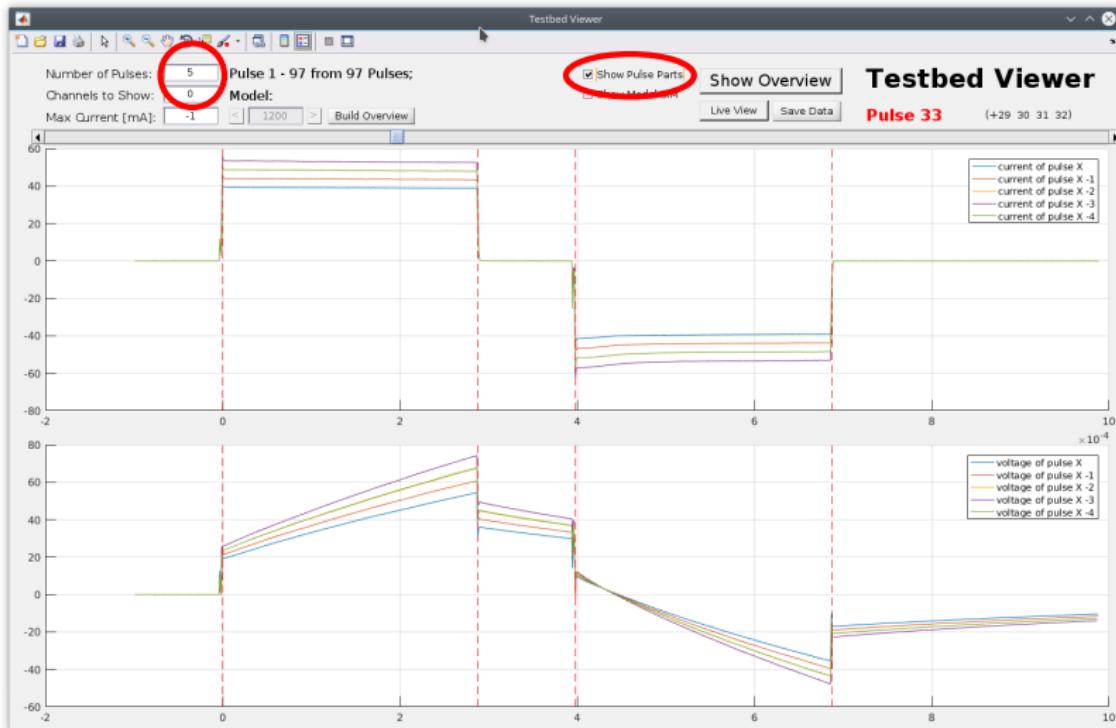


- pulse selection slider

Matlab GUI - Pulse Details

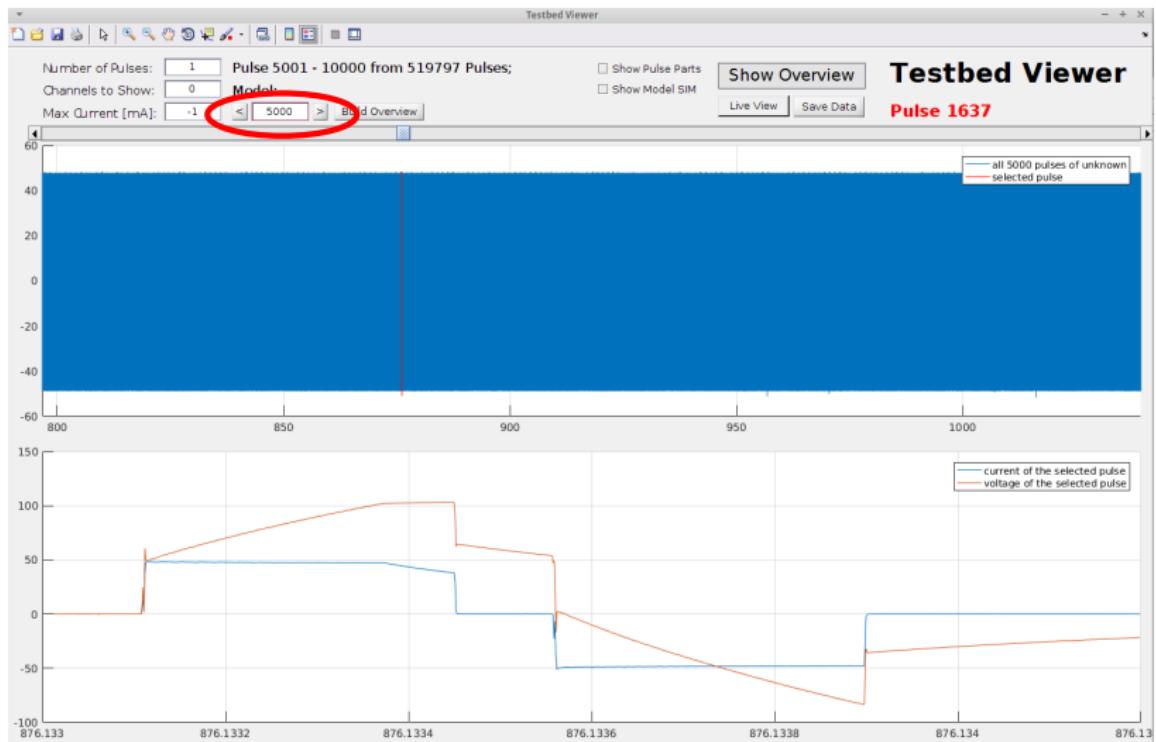


Matlab GUI - Pulse Details



- multiple pulses for comparison, aligned by the found pulse parts

Matlab GUI - Large Datasets



- large datasets are split into manageable sections

Matlab GUI - Other Features

Data Overview

- graphical overview of max. current, voltage, pulse duration, and stimulation period
- for the current selection or the entire dataset

Matlab GUI - Other Features

Data Overview

- graphical overview of max. current, voltage, pulse duration, and stimulation period
- for the current selection or the entire dataset

Live Data View

- TCP connection to the data recorder
- live display of the recorded raw data
- optional sound feedback

Matlab GUI - Other Features

Data Overview

- graphical overview of max. current, voltage, pulse duration, and stimulation period
- for the current selection or the entire dataset

Live Data View

- TCP connection to the data recorder
- live display of the recorded raw data
- optional sound feedback

Data Management

- save function with selectable data reduction

Conclusions

Summary

- modular data acquisition hardware specialized for FES
- simple PC program to control the hardware and to save the data
- Matlab based data post-processing, including the Matlab GUI

Conclusions

Summary

- modular data acquisition hardware specialized for FES
- simple PC program to control the hardware and to save the data
- Matlab based data post-processing, including the Matlab GUI

Future Work

- possible future functions: pulse notes, pulse flagging for later inspection, automatic generation of a statistic report
- easily extendable via own Matlab functions using the pulse based Matlab data structures

Conclusions

Summary

- modular data acquisition hardware specialized for FES
- simple PC program to control the hardware and to save the data
- Matlab based data post-processing, including the Matlab GUI

Future Work

- possible future functions: pulse notes, pulse flagging for later inspection, automatic generation of a statistic report
- easily extendable via own Matlab functions using the pulse based Matlab data structures

Source Code and Hardware Files

⇒ https://github.com/worldwidemv/fes_testbed

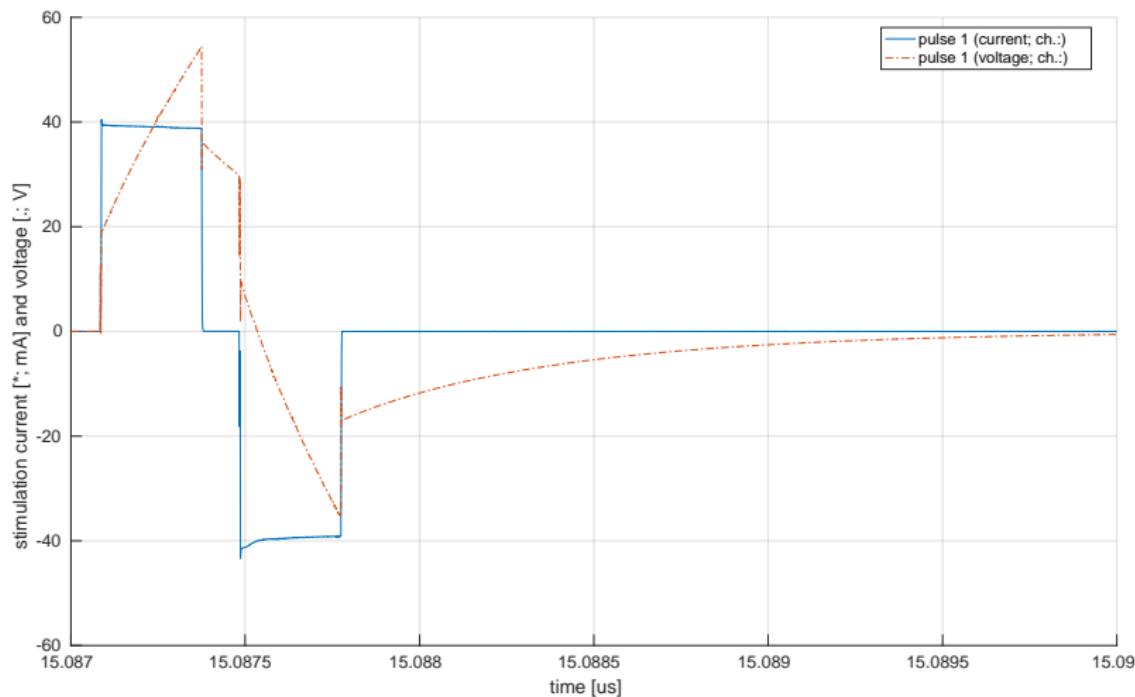
Thank you for your attention.



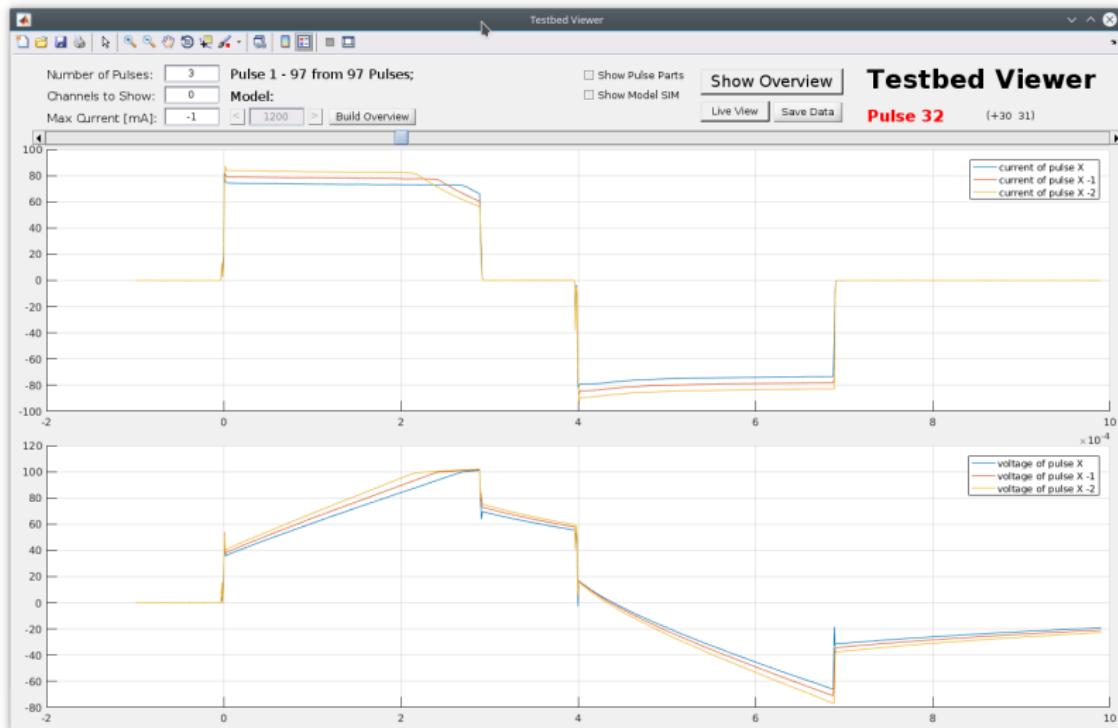
Source Code and Hardware Files

⇒ https://github.com/worldwidemv/fes_testbed

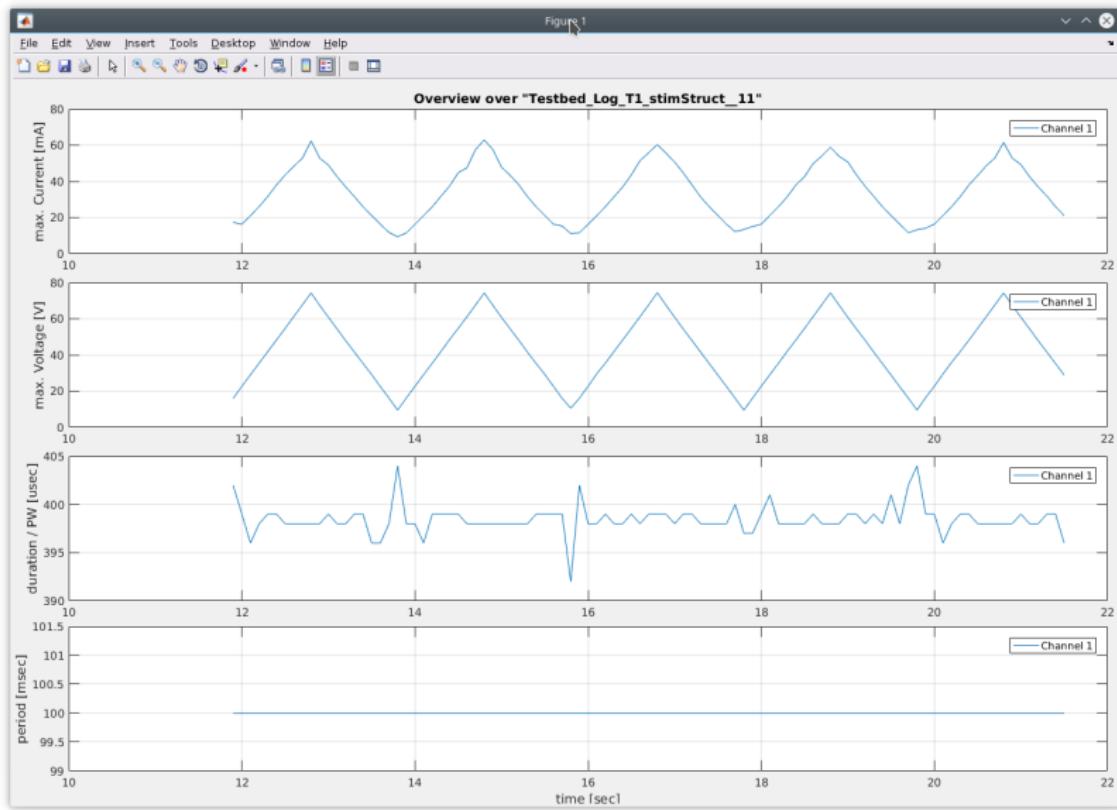
Raw Data of Pulse 33



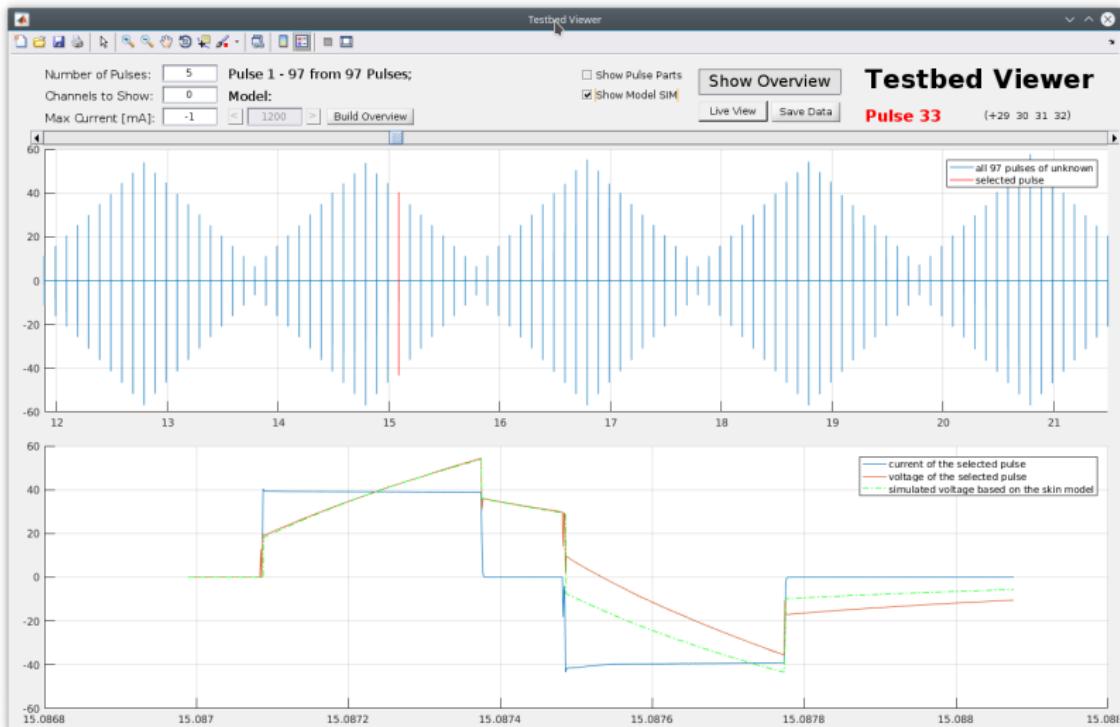
Saturation of the Stimulation Voltage



Data Set Overview



Skin Model Parameter Estimation



GUI Save Dialog

