Version: 36

Document-ID: 61442442

EIT - 16, 32, 64, 128 and 256 Ch

certified in accordance with
ISO 9001:2015
ISO 13485:2016

Exported on 2023-10-24

Released on 2023-10-24

Sciospec Scientific Instruments GmbH
Leipziger Straße 43b,
D-04828 Bennewitz, GERMANY

Tel: +49 3425 88399 00
Email: info@sciospec.de

www.sciospec.com

Table of Contents

1 Which products are covered by this manual

This manual is valid for:

- EIT (electrical impedance tomography) system with 16, 32, 64, 128 or 256 channels

2 General

The content and works published in this manual are governed by the copyright laws of Germany. Any duplication, processing, distribution or any form of utilization beyond the scope of copyright law shall require the prior written consent of the author or authors in question.


Please pay attention to the comment.


Please pay attention to the note/warning.


Please pay attention to the warning.

2.1 Warranty

This product is warranted against defects in material and workmanship for a period of one year. During the warranty period any parts or services that show defects with a cause that was present before the transfer of risk shall – at the option of Sciospec – be replaced, reworked or re-performed free of charge (rectification).

For warranty service or repair, this product must be returned to a service facility designated by Sciospec. The buyer shall pay all shipping charges, duties, and taxes for products returned to Sciospec from another country.

Sciospec warrants that its software and firmware designated by Sciospec for use with an instrument will execute its programming instruction when properly installed on that instrument. Sciospec does not warrant that the operation of the instrument, or software, or firmware will be uninterrupted or error free.

Further details on the warranty and transfer of risk are described in the general terms and conditions of Sciospec. No other warranty is expressed or implied. Sciospec specifically disclaims the implied warranties of merchantability and fitness for a particular purpose. Further and/or divergent regulations require contractual agreements in written form.

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by the buyer, operation or service through unauthorized personnel, the buyer-supplied software or interfacing, unauthorized modification or misuse, operation outside the environmental specifications for the product, or improper site preparation or maintenance.

2.2 Intended (normal/contractual) use

The product is intended for indoor use in qualified technical environments and laboratories with all necessary safety measures for electric installations and cabling in place. In order to fulfill the requirements towards electrical safety and electromagnetic compatibility the system is to be installed in a safe environment with

electrical connections (especially power supply) according to general safety regulations. In addition to general safety regulations all safety measures and precautions noted in this document are to be followed.

For outdoor use separate precautions and safety measures are required. Specifically it might be required to mount the system into a protective housing according to the ambient/environmental conditions.

2.3 Non-intended use

Any form of use deviating from the intended use qualifies as non-intended use/misuse/maloperation. Other applications, modes of operations, modifications or types of installations can impair safety and functionality and are thus for safety reasons not permitted for the user or third parties.

2.4 Authorized personnel

In general different levels of qualification of operating and maintenance personnel have to be distinguished.

> Danger through insufficient qualification of operating/service personnel.

The instrument may only be operated, functionally tested and serviced by sufficiently qualified personnel. Work reserved for professional technical personnel may only be done by staff authorized through Sciospec.

2.4.1 Users

Users are all personnel older than 18 years that have been instructed in the operation of the instrument by Sciospec or an authorized representative. They have to have read and understood this manual completely.

User may be tasked with the following:

- General operation of the system through externally accessible interfaces
- Operation of external controls on the instrument
- functional tests through externally accessible interfaces
- to some extent elimination of malfunctions or initiation of measures to eliminate malfunctions through externally accessible interfaces

2.4.2 Professional technical personnel

Professional technical personnel are all persons who in accordance to their professional education, professional experience and contemporary professional occupation

- posses respective knowledge of electrical impedance analyzers
- have been trained, authorized and instructed by Sciospec for maintenance, installation and service tasks on/for the instrument

Further the professional technical personnel have to be qualified for those tasks through their professional education, experience and technical knowledge on electrical impedance analyzers.

In addition to the authorization of users professional technical personnel may be tasked with

- Elimination of malfunctions exceeding the measures permissible for a user
- installations

- maintenance and service work as instructed by Sciospec

2.4.3 Unauthorized personnel

Through operation and service procedures executed by unauthorized personnel substantial danger up to danger to life may arise.

Defects and damage caused by measures performed by unauthorized personnel are not covered by warranty and Sciospec will not be made liable for these.

2.5 Danger and risks

Even with proper professional installation and proper condition of the instrument not all dangers can be covered. Following some of the remaining risks are described.

2.5.1 Electrical currents

Danger to life through electrical currents

- operation and maintenance only to be done by trained personnel
- no access to the instrument for other than authorized personnel
- never bypass fuses or protective parts
- maintenance and installation only in volt-free state
- keep moisture and excessive heat away from the instrument

2.5.2 Risk for danger to personnel

Increased risk for danger to the user exists

- when minors, elderly or disabled persons are allowed to operate the instrument
- when proper instruction, training and monitoring of the user is not possible
- when no authorized personnel exist
- when the number of users to the instrument are very high
- when modifications to the instrument or its installation are done

Modifications, extensions/additions to the instrument are not permissible without prior authorization through Sciospec. Any unauthorized technical or procedural modification voids the permission for operation and the warranty.

2.6 Safety information and precautions

When you notice any of the unusual conditions listed below, immediately terminate operation and disconnect the power cable. Contact Sciospec Scientific Instruments for repair of the instrument. If you continue to operate without repairing the instrument, there is a potential fire or shock hazard for the operator.

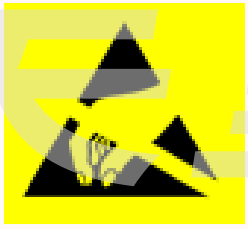
- Instrument operates abnormally.
- Instrument emits abnormal noise, smell, smoke or a spark-like light during the operation.
- Instrument generates high temperature or electrical shock during operation.
- Power cable, plug, or receptacle on instrument is damaged.
- Foreign substance or liquid has fallen into the instrument.

The following measures help assuring safe instrument operation and are to be followed in order to comply with intended use specifications.

⚠ Technical expertise is required. Risk of damage to the instrument exists through unintended use and wrong cabling/installation.

- Do not operate the instrument in the presence of inflammable gasses or fumes. Operation of any electrical instrument in such an environment clearly constitutes a safety hazard.

Electro static discharge warning



This product, like all electronic products, uses semiconductors that can be damaged by electrostatic discharge (ESD). Use care when handling the devices to prevent damage. Damage due to inappropriate handling is not covered by the warranty.
ESD damage is most likely to occur as the test fixtures are being connected or disconnected. Protect them from ESD damage by wearing a grounding strap that provides a high resistance path to ground. Alternatively, ground yourself to discharge any static charge built-up by touching the outer shell of any grounded instrument chassis before touching the test port connectors.
Do not exceed the operating input power, voltage, and current level and signal type appropriate for the instrument being used, refer to technical specifications sections in this manual.

2.7 Replacement parts and equipment

Obtain replacement parts only through Sciospec, authorized customer support services or authorized distributors.

Danger of injury through use of wrong or faulty replacement parts!

The use of wrong, faulty or unauthorized/unintended replacement parts can lead to danger of injury to the operating/service personnel, malfunction or complete failure of the instrument.

Use of unauthorized replacement parts voids the instruments warranty.

3 Safety Concept for EIT System (including optional add-ons) in medical research use

3.1 General Safety Instructions

- Safe operation depends on the use case setting including instrument settings, cabling, environment, other equipment etc. Carefully plan the use of the instrument for safe operation!
- Refer to the functional description and technical specification for details on the operational parameters.
- Operation outside the specified ranges may cause damage and/or unsafe operating conditions
- Choosing adequate measurement ranges with sufficient headroom avoids saturation of the measurement port input amplifiers and helps maintaining a safe operating point.
- While not always required, in general the use of AC coupling of the measurement ports is recommended for safe operation. If used without the AC coupling frontend, DC leakage currents might be a concern.
- Assure that the settings of the instrument - especially the excitation parameters - lead to acceptable currents through the measurement object. Ideally this is done on a previously established model (e.g. equivalent impedance network), but if this not feasible the resulting currents should at least be checked for safe operating range during inital setup of the measurements.
- If used in combination with other electrical equipment, this needs to be considered during the assessment of safe operation.
- if battery supply is available in a given use case scenario, disconnecting the system from mains and sourcing from a floating battery supply can increase electrical safety

> The Sciospec EIT system is not a medical product and as such its intended purpose does not include diagnosis and/or treatment of patients.

3.2 Functional and Structural Overview

The functional concept of the EIT system is to inject an electrical excitation stimulus (inject current) and measure the response of the measurement object (potential differences on multiple ports). The stimulus is applied as sinusoidal signal with varying frequency. Optionally the amplitude of the excitation signal can also be varied.

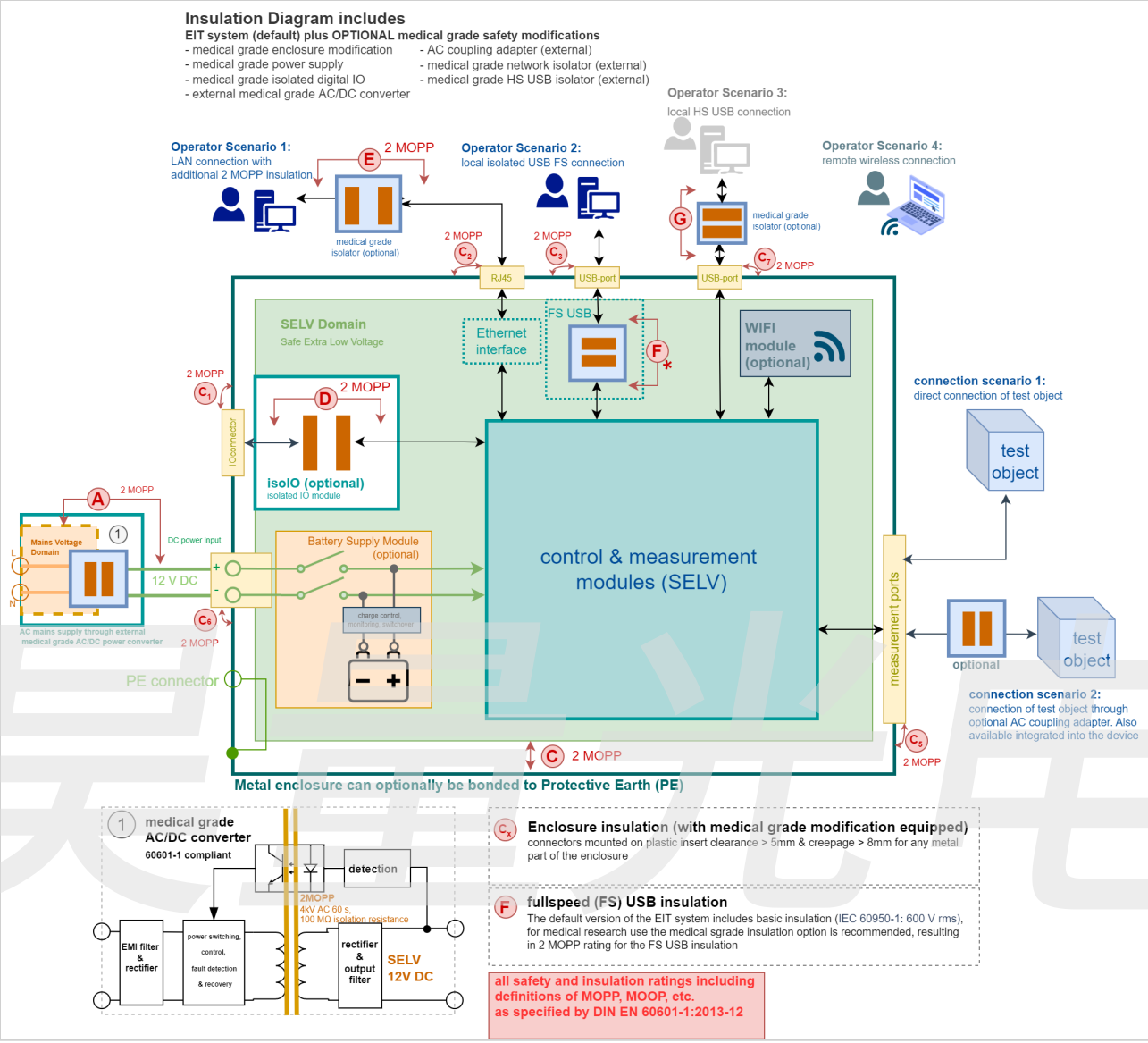
The system can be equipped with a number of **optional** add-ons and safety measures to facilitate the use of electrical impedance tomography in prototypic product development and research scenarios. This includes measures for enhanced electrical safety including medical grade isolation enabling a safe operation of the instrument in medical research settings in accordance with DIN EN 60601-1:2013-12 "Medical electrical equipment – Part 1: General requirements for basic safety and essential performance". The most of important principles thereby are

- medical grade isolation towards mains
- optional battery supply
- medical grade isolation for operator side interfaces
- optional DC blocking through AC coupling on the measurement ports (equivalent to "patient" interface connections)

The system consists of the following main building blocks:

Component	Function	Safety Notes
metal enclosure	prevention of contact between electrical circuitry and external equipment or humans, shielding of measurement frontend (noise reduction) and mechanical safety	for medical research use it is recommended to use the optional "medical safety enclosure option" where all feedthroughs and connectors are electrically isolated from enclosure, by 2 means of patient protection (2 MOPP), there is also an optional connection to protective earth for scenarios where this is required
power supply (external)	System is powered in safe extra low voltage domain (SELV) - nominal 12V DC.	for medical research use it is recommended to use medical grade power supply units with 2 MOPP isolation ratings for safe operation
battery supply module (optional)	automatic switchover to battery on loss on disconnection of external supply source. Optional: active switchover to battery including hard disconnect of external source through internal relays (both + and - source get disconnected).	if battery supply is available in a given use case scenario, disconnecting the system from mains and sourcing from a floating battery supply can increase electrical safety
measurement and control modules	core functionality of the impedance analyzer including measurement control, signal processing, excitation generation and analog measurement functions	SELV operation
USB (FS) interface	full speed USB port for instrument control and data readout from PC	the default version of the system includes basic insulation (IEC 60950-1: 600 V rms), for medical research use the medical grade insulation option is recommended, resulting in 2 MOPP rating for the FS USB insulation
USB (HS) interface	high speed USB port for instrument control and data readout from PC	The interface is not internally isolated. An external medical grade isolator (optional accessory) can be inserted in the communication path to ensure safe isolation ratings.
WIFI interface (optional)	standard 802.11 n (2.4 GHz) wireless interface	
Ethernet interface	standard RJ45 Ethernet interface for remote control of the instrument	The interface is not internally isolated. An external medical grade isolator (optional accessory) can be inserted in the communication path to ensure safe isolation ratings.
IsoIO module (optional)	low level communication interfaces including hardware synchronization and flexible programmable IO standards including serial (UART, SPI, I2C) for use of the instrument with other external equipment	2 MOPP rated isolation on all ports
additional AC coupling adapter (optional)	Optional AC coupling adapters can be inserted into the electrical path between measurement object and measurement module. These can be integrated into the device (ask us for factory side configuration options) or attached externally. It includes structures for optional AC coupling components that block DC currents and add a high pass filter characteristic limiting low frequency currents. With this an additional galvanic isolation barrier can be introduced between the instrument and the measurement object or patient interface.	DC blocking always implies a low frequency cutoff - care must be taken not to deteriorate the measurement performance when using this safety feature.

3.3 Isolation Diagram



4 Technical Specification

All specifications are stated for a operating at temperatures of 0°C to 40°C unless specified otherwise.
Warm-up time must be greater than or equal to 30 minutes after power on to comply with all specifications.

Maximum Ratings

Stresses above the listed absolute maximum or maximum ratings may cause permanent damage to the device. This is a stress rating only; functional operation of the device at these or any other conditions above those indicated in the operational section of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

4.1 Measurement Specifications

Measurement Parameters

Complex electrical voltage	Real and imaginary

Backpanel Ground Terminal

configuration	Optional terminal for accessing internal "ground" potential is internally connected to the electronics 0 V potential ground structure.
connector type	2 mm banana connector (female)
mating plug	2 mm banana plug

Input buffers and current source are optimized for an impedance range of 10 Ω to 10 kΩ. The actual limits of impedance resolution in EIT measurements depends on the specific conductivity distributions and cannot be specified with a generalized range. Make sure to validate settings for a specific setup don´t violate any specifications to avoid erroneous electrical behavior and corrupted measurement data.

Excitation and Measurement Frequency

range	100 Hz to 1 MHz

resolution	40µHz @ f<10kHz 2mHz @ 10kHz < f <500kHz 220mHz @ f>500kHz
precision absolute	±100ppm (at 25°C)
temperature drift	±10ppm over operating temperature range
long time stability	±5ppm first year

Excitation Signal

Range	1 µA to 10 mA peak-amplitude
Resolution	2.81 µA
Maximum continuous output current	50 mA
Compliance Voltage	± 11 V (battery powered ±5V)

Measurement Timing

Frame rate	0.1 ... 100 fps (refer to **Functional Description** for information on constraints towards the maximum allowable frame rate)
Absolute time base precision	± 100 ppm
frame to frame jitter	± 200 ns

Frequency Sweep Settings

sweep type	linear, logarithmic, list
points	1 to 128

Injection/Excitation Pattern Sweep Settings

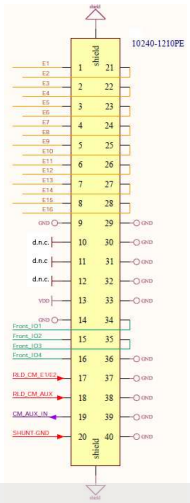
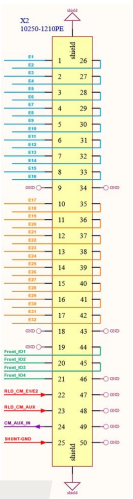
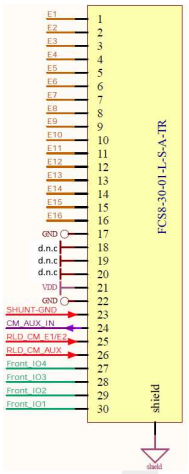
pattern count	1 ... 256
Inject switch delay	600 µs (default)

4.2 Measurement Interfaces

4.2.1 General **Measurement Terminal** Specifications

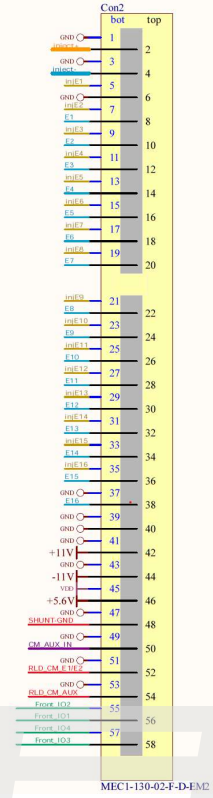
configuration	16/32/64/128 dual role ports with high input impedance voltage measurement and optional current injection function
connector type	MEC1-130 connector, D-Sub-Micro-D connector 40P, FCS8 connector
Input voltage range	±5 V
Measurement voltage range	±5 V
Absolute maximum Input voltage	±11 V
Input Impedance (in non injecting state)	≈ 1 TΩ, 2.3 pF
Output voltage range	±11 V (battery powered ±5V)
Absolute max. short circuit current (source/sink) for any port in injection state	120 mA
Measurement port shielding	a grounded passive (capacitive) shield connection is provided

4.2.2 Standard Connectors D-Sub-Micro-D connector and **FCS8 connector**

D-Sub-Micro-D connector (MDR40)	D-Sub-Micro-D connector (MDR50)	**FCS8**
		

Pin	Function	Comment
E1...E16	EIT Electrode 1 to 16	Injection and Measurement Connection
GND	EIT Device Ground Potential	only for Digital GPIO
VDD	EIT Device Power Supply 3.3V	only for Digital GPIO
d.n.c	Do Not Connect	reserved for later applications The EIT device may apply potentials in the range of +-11V to those terminals.
Front_IO1...4	General Purpose IOs	Can be accessed by using the Sciospec COMInterface
RLD_CM_E1/E2	Right Leg Drive Potential derived from Common Mode E1 and E2	Special purpose connections for biasing the device under test.
RLD_CM_Aux	Buffered Right Leg Drive Potential derived from Auxiliary Input	
CM_AUX_IN	Auxiliary Input	
SHUNT-GND	1MΩ to GND	

4.2.3 Custom Connector MEC1-130 connector



Pin	Function	Comment
E1...E16	EIT Measurement Electrode 1 to 16	Only Measurement Connection
injE1...injE16	EIT InjectionElectrode 1 to 16	Only Injection Connection
inject+	additional inject source connection	
inject-	additional inject sink connection	
GND	EIT Device Ground Potential	only for Digital GPIO
VDD	EIT Device Power Supply 3.3V	only for Digital GPIO
d.n.c	Do Not Connect	reserved for later applications The EIT device may apply potentials in the range of +-11V to those terminals.
Front_IO1...4	General Purpose IOs	Can be accessed by using the Sciospec COMInterface

Pin	Function	Comment
RLD_CM_E1/E2	Right Leg Drive Potential derived from Common Mode E1 and E2	Special purpose connections for biasing the device under test.
RLD_CM_Aux	Buffered Right Leg Drive Potential derived from Auxiliary Input	
CM_AUX_IN	Auxiliary Input	
SHUNT-GND	1MΩ to GND	

ⓘ For standard use cases injEX has to be shorted to EX on all 16 Electrode connections. For assistance on more advances applications please contact Sciospec for details.

4.3 Master Interface

4.3.1 Ethernet Interface

standard conformity	10/100 Base-T, RJ45
protocol	TCP/IP

4.3.2 Isolated Full Speed USB Interface

standard conformity	USB 2.0, FS (12 Mbits/s)
connector	Micro USB Type B
protocol	Full Speed USB
ESD Protection	Class 3A contact ESD performance per ANSI/ESD STM5.1-2007
Isolation (standard configuration)	2500 V rms for 1 minute per UL 1577 IEC 60950-1: 600 V rms (basic) DIN V VDE V 0884-10 (VDE V 0884-10):2006-12 VIORM = 560 V peak
Isolation (medical research configuration)	UL recognition: 5000 V rms for 1 minute per UL 1577 CSA Component Acceptance Notice #5A IEC 60601-1: 250 V rms (reinforced) IEC 60950-1: 400 V rms (reinforced) VDE Certificate of Conformity DIN V VDE V 0884-10 (VDE V 0884-10):2006-12 V_{IORM} = 846 V peak

4.3.3 High Speed USB Interface

standard conformity	USB 2.0, HS (480 Mbit/s)
connector	Micro USB Type B
protocol	High Speed USB
ESD Protection	±12kV IEC 61000-4-2 contact ESD clamb voltage 13V (min) break-down voltage 5.5V (min)

4.3.4 WiFi Interface (optional)

standard conformity	2.4 GHz, IEEE Std. 802.11b
protocol	TCP/IP

4.4 General Specifications

4.4.1 General Specifications standard device configuration

power requirements	12VDC (typ.), 18W (max), Barrel connector panel jack (outside diameter 10.5mm, inside diameter 2.1mm) (PN: HEBL 21)
dimensions	184.0mm x 54.8mmx 113.1mm (width x height x depth) see Figure 1 and Figure 2
weight	1kg (typical)
operating conditions	0°C to 40°C, <80% relative humidity non condensing, 0...3000m altitude
non-operating conditions[1]	-25°C to 80°C, <80% relative humidity non condensing The temperature gradient should not exceed 1K/min to reach operating conditions.

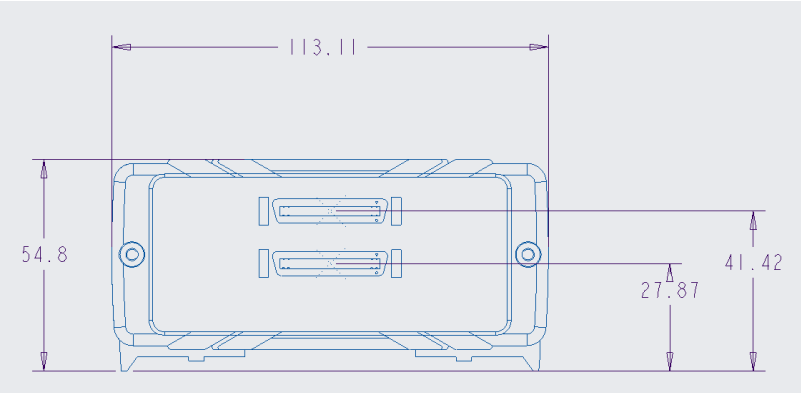


Figure 1: enclosure front view (measures in mm)

Figure 2: enclosure side view (measures in mm)

4.4.2 General Specifications device configuration with specific options

power requirements	12VDC (typ.), 18W (max), Barrel connector panel jack (outside diameter 10.5mm, inside diameter 2.1mm) (PN: HEBL 21)
dimensions	248.67mm x 97.1mmx 193.2mm (width x height x depth) see Figure 3
weight	2.5kg (typical)
operating conditions	0°C to 40°C, <80% relative humidity non condensing, 0...3000m altitude
non-operating conditions[1]	-25°C to 80°C, <80% relative humidity non condensing The temperature gradient should not exceed 1K/min to reach operating conditions.
enclosure isolation (medical grade configuration)	connectors mounted on plastic insert clearance > 5mm & creepage > 8mm for any metal part of the enclosure

ⓘ The enclosure front view depends on the device configuration.

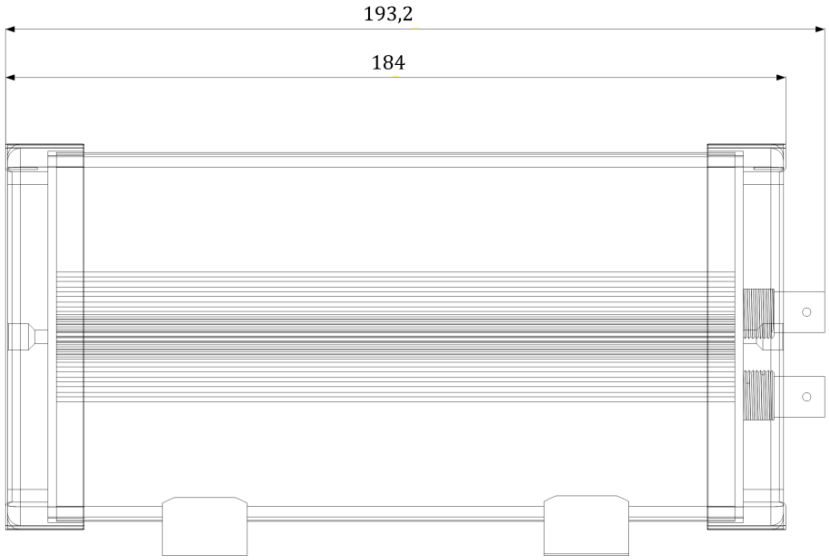


Figure 3: enclosure side view (measures in mm)

[1] Conditions the device cannot be operated, but held, transported and stored under

5 Functional Description

5.1 General

The system is available with different channel numbers: 16-channel, 32-channel, 64-channel, 128-channel. This description is valid for all channel numbers.

This system is designed for measurements for electrical impedance tomography or general multi-port impedance measurement applications. For that purpose it contains:

- multiple dual role electrode connections, serving both as voltage measurement and current injection ports
- the number of available ports equals the number of channels of the system
- a pseudo differential current source for single sine excitation signals up to 10 mA
- tightly synchronized data acquisition block for truly simultaneous sampling of all channels
- an ultra low leakage reed relay cross point switch matrix for assigning a positive and a negative injection port to a measurement channel (Any combination of ports is possible for every measurement channel.)
- the Sciospec ***SineFit*** algorithm for signal extraction

The measurement mode allows automatic sequential measurement of user configurable excitation settings. Each excitation setting contains a pair of electrode numbers. One to be used as positive current injection port and one as negative current injection port (current return). While one excitation setting is active:

- a single frequency sinusoidal test current will flow between the two selected ports (if a multi-frequency sweep is configured the configured frequency sweep is carried out on each excitation setting)
- voltage will be measured on all measurement channels simultaneously

After the measurement phase is over the next excitation setting which has been set by the user is configured and according to the timing specification a new measurement period starts, again acquiring voltage signals for all measurement channels, but now with current flowing according to the new excitation setting. For multi-frequency measurements a measurement phase contains a whole sweep. Hence a frequency sweep is always completed before switching to the next excitation setting.

All measurement data is processed in real time by the Sciospec SineFit algorithm extracting real and imaginary parts of the acquired voltage signal.

Output data is organized in units called "EIT-frame". An EIT-frame contains the measurement results of one 16-channel-group. To distinguish between the groups it contains the group address. (see COMinterface for more information)

5.2 Master Interfaces

5.2.1 Full Speed USB

The device creates a virtual COM port named "serial USB device", which can be used like a serial port.

Any COM port obtains a unique number by the operating system. (i.e. "COM6") Which number is assigned to a certain port is decided by the operating system and can not be altered.

The port can be used either to connect the Sciospec Software or any terminal program to the device. There is no need to configure the general settings for the serial port (e.g. baud rate, stop bit count, etc.). Any setting of your terminal program will work.

To connect the Sciospec Software to the virtual COM port, choose "serial" in the connection dialog and then choose the desired port.

Driver installation

For Microsoft Windows 8 or more recent versions, there is no need to install any driver to connect to the device. After establishing a USB connection between the device and a PC, the new serial COM port will be available.

For Microsoft Windows 7 or XP, the driver named "Sciospec FS USB driver" provided in the Sciospec Software package, has to be installed before connecting the device to the PC. If you encounter a compatibility or verification warning during installation, choose to continue and install the driver anyway. When the installation has finished, you can connect the device to your PC. The new serial COM port will then be available.

Finding the correct COM port

If more than one COM is active on the PC, you have to determine which one belongs to the Sciospec device. An overview of known COM ports is given by the Windows device manager (Shortcut for Windows 8 or more recent: 'WIN + X') under "Ports (COM & LPT1)". Plugging in the USB cable while observing the port list shows which number is assigned to it.

Example: C-Code for establishing a connection

```
Handle hSerial = CreateFile(  "\\\\.\\COM6",
                              GENERIC_READ|GENERIC_WRITE,
                              0,
                              NULL,
                              OPEN_EXISTING,
                              FILE_ATTRIBUTE_NORMAL,
                              NULL );
        if (hSerial == INVALID_HANDLE_VALUE)
        {
                fprintf(stderr, "Error\n");
                return 0;
        }
          return 1;
```

5.2.2 High Speed USB

The device identifies itself to the USB-host (e.g. PC) by a unique combination of VID (Vendor-ID) and PID (Product-ID).

Driver installation

Before connecting the device to the PC, the driver provided by Sciospec, has to be installed. The communication interface to the driver is provided by the FT2xx.dll, which is part of the driver package. The complete specification of the functionality can be found in the “D2XX Programmers Guide“ (http://www.ftdichip.com).

To install the driver on your PC

- Unzip the "Sciospec Device Driver_signed.zip" archive
- Open the extracted "Driver" folder
- On a 32bit operating system start the "DriverInstall.exe"
- On a 64bit operating system start the "DriverInstall64.exe"
- Follow the installation instructions

After finishing the installation you can connect the device to an USB port of your PC.

Configuration

When a connection is established and the device has been detected by the operating system, the Sciospec software can connect to the device by choosing "USB" in the connection dialog and selecting the device from the appearing list. If you want to communicate with the device using the Sciospec COMInterface, the mode of communication must be set to “245-FIFO-Mode” before data can be transferred to the device. This can be done with the “FT_SetBitMode”-function. This functionality is documented in the “D2XX Programmers Guide“. A C-Code example is provided below.

Example: C-Code for establishing a connection

```
FT_STATUS ftStatus;
FT_DEVICE_LIST_INFO_NODE* devInfo;
DWORD numDevs; // create the device information list
ftStatus = FT_CreateDeviceInfoList(&numDevs); // allocate storage for list based on
numDevs
devInfo = (FT_DEVICE_LIST_INFO_NODE*)malloc(sizeof(FT_DEVICE_LIST_INFO_NODE)*numDevs)
; // get the device information list
ftStatus = FT_GetDeviceInfoList(devInfo,&numDevs);
FT_HANDLE handle;
int i;
for (i = 0; i < numDevs; i++) {
        if((devInfo[i].ID & 0xFFFF) == 0x89D4){
                ftStatus = FT_Open(i, &handle);
                ftStatus = FT_SetBitMode(handle, 0x00, 0x40);
                break;
        }
}
```

VID	PID
0x0403	0x6014

Ethernet

The Sciospec device uses a bidirectional socket communication in which it implements the socket server and the connected PC the socket client. It is delivered with a ready to use ethernet configuration. It supports DHCP to receive an IP address from a DHCP server. Since DHCP is enabled by default, the device will obtain an IP address if you connect it to your ethernet router and can instantly be operated via this interface. This is the default setting but can be deactivated using the software or the “set Ethernet configuration” command. When DHCP is deactivated a static IP address must be set by the user. All Sciospec devices use the following port.

port	5000

The Sciospec device also can identify itself in a local area network by answering a UDP broadcast call. It answers with its device id. From this answer you may obtain the IP address by checking the UDP package. The broadcast message to be send is the "Get Device ID" command and needs to be send on the following port

port	8888

Example: Establishing a connection using the COMinterface

1. Connect the device Ethernet network and via USB to your computer.
2. Open terminal program and connect to the USB port of the device. (see "Setting up a connection to the device via USB")
3. Send the "get IP address" command **[BE] [01] [01] [BE]**.
4. The device will return the IP address, which it received from the DHCP server. (See command description for return syntax or disabling DHCP.)
5. Disconnect the terminal programm.
6. Use the obtained IP address and the port to establish a socket connection from your software to the device.
7. From now on the device can be controlled by regular COMinterface commands.

Example: C-Code for establishing a connection

```
    int sock;

/**This block is needed only if using WinSocket (Windows) to initialize the socket*/
    printf("\nInitialising Winsock...\n\n");
    WSADATA wsa;
    if (WSAStartup(MAKEWORD(2,2),&wsa) != 0){
        printf("Failed. Error Code : %d",WSAGetLastError());
        return 1;
    }
/** End of WinSocket initializing*/


    if( (sock = socket(AF_INET, SOCK_STREAM , 0)) == INVALID_SOCKET ){
```

```
            printf("Could not create socket");
            return -1;
        }

        /*Initialize address of device
         * IP address needs to be adapted, depending on local network and Ethernet
settings of the device*/
        struct sockaddr_in server;
        server.sin_addr.s_addr = inet_addr("192.168.100.115");
        server.sin_family = AF_INET;
        server.sin_port = htons( 5000 );

        //Connect to remote server
        if (connect(sock , (struct sockaddr *)&server , sizeof(server)) < 0){
            printf("connect error");
            return -1;
        }
```

Example: Java-Code for UDP broadcast call

```
String host = "255.255.255.255";    //Broadcast address
int port = 8888;                    //the destination port of the broadcast
char data[] = {0xD1, 0, 0xD1};      //bc char is unsigned
String message = new String(data);

try{
    InetAddress adds = InetAddress.getByName(host);
    DatagramSocket ds = new DatagramSocket();
    DatagramPacket dp = new DatagramPacket(message.getBytes(), message.length(),
adds, port);
    ds.send(dp);
    ds.close();
}
catch (UnknownHostException e){
    e.printStackTrace();
} catch (SocketException e) {
    e.printStackTrace();
} catch (IOException e){
    e.printStackTrace();
}

byte[] buf = new byte[1024];    //store the message sent

try {
    //Bind port
    DatagramSocket ds = null;
    DatagramPacket dp = null;
    ds = new DatagramSocket(port);
    dp = new DatagramPacket(buf, buf.length);
```

```
    System.out.println("The listening broadcast port is open:"+port);

    ds.receive(dp);
    ds.close();

    System.out.print("Recieved Msg:");

    for(int j=0;j<dp.getLength();j++) {
         System.out.print(" "+String.format("%02X", buf[j]));
    }

    System.out.println("");
    System.out.print("Received from:"+dp.getAddress());
}
catch (SocketException e) {
    e.printStackTrace();
} catch (IOException e){
    e.printStackTrace();
}
```

5.2.3 WiFi

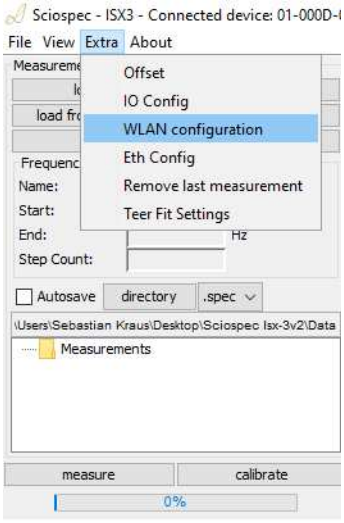
protocol	TCP/IP
supported standard	802.11 n (2.4 GHz)
supported security protocols	WEP, WPA PSK, WPA2 PSK, WPA2 Enterprise

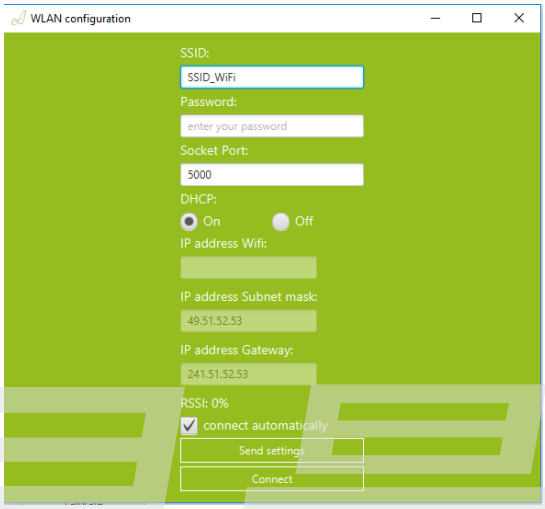
To establish a wireless connection, an existing wireless network of the above mentioned standard is needed. Before connecting to the network, a correct setup has to be written to the device. It contains settings such as "SSID" (the name of the wireless network), password, IP-address and others. For simplicity the IP address can be obtained by DHCP or set manually as static address. A configuration of the WiFi interface can be performed via any of the master interfaces. To do so the ComInterface command "WLAN configuration" can be used. Alternatively, if you are using the Sciospec software, you can quickly enter the setup by opening the "WLAN configuration" dialog.

After connecting to the wireless network bidirectional socket communication is used, in which the Sciospec WiFi endpoint implements the socket server and the connecting endpoint (e.g. the PC running the Sciospec software) acts as the socket client.

Example: Establish a connection using the Sciospec Software

1. Connect the device to a USB port of your PC.
2. Open the Sciospec Software and connect it to your USB device.
3. Open the "WiFi configuration" dialog, which can be found in the "Extra" menu.
4. Fill in all information necesarry to connect to your desired WiFi network and press "Send settings" to permanently save it to the device. The next time you want to connect to the network, you don't have to enter all these information again.
5. Click "Connect" to establish the connection. When the connection has been succsessfully established, the label of the "Connect" button changes to "Disconnect".





ⓘ To find out which IP address has been obtained when DHCP is enabled, just open the "WLAN configuration" dialog. The current IP address will be displayed in the "IP address Wifi" field. Furthermore the signal strength (RSSI) of the active connection will be displayed.

ⓘ If "connect automatically" is choosen in the dialog, the device tries to connect to the configurad network automatically, every time it is turned on.

Example: C-Code for establishing a connection

```
int sock;
char *cmd, *readBuffer;

/**This block is needed only if using WinSocket (Windows) to initialize the socket*/
printf("\nInitialising Winsock...\n\n");
WSADATA wsa;
```

```
if (WSAStartup(MAKEWORD(2,2),&wsa) != 0){
printf("Failed. Error Code : %d",WSAGetLastError());
return 1;
}
/** End of WinSocket initializing*/

if( (sock = socket(AF_INET, SOCK_STREAM , 0)) == INVALID_SOCKET ){
printf("Could not create socket");
return -1;
}

/*Initialize address of Device
* IP address needs to be adapted, depending on local network and Ethernet settings in
Device*/
struct sockaddr_in server;
server.sin_addr.s_addr = inet_addr("192.168.100.115");
server.sin_family = AF_INET;
server.sin_port = htons( 5000 );

//Connect to remote server
if (connect(sock , (struct sockaddr *)&server , sizeof(server)) < 0){
printf("connect error");
return -1;
}

/* ************************** *
* Reading DeviceID *
* ************************** */
int numberOfBytes = 3;
printf("Reading DeviceId\n");
cmd = (char*)malloc(sizeof(byte)*numberOfBytes);
cmd[0] = 0xD1; cmd[1] = 0x00; cmd[2] = 0xD1;
if( send(sock , cmd , numberOfBytes , 0) < 0) {
printf("Could not send cmd.");
return -1;
}
free(cmd);

readBuffer = (char*)malloc(sizeof(byte)*19);
recv(sock , readBuffer , 19 , 0);

printf("DeviceID: ");
int i;
for(i=0; i<19;i++){
printf("%.2X ", (byte)readBuffer[i]);
}
```

Troubleshooting

Connection can not be established

- Check if the configuration has been entered correctly (such as SSID, password, etc.).
- Check if you have coordinated permissions to use the network.
- Does the Wifi router you are trying to connect to support the standards and protocolls expected by the Sciospec WiFi device? (i.e. 2.4GHz)
- Is the Sciospec device in range of the network you are trying to connect to? You can check the signal strength the device is receiving.

Slow or unreliable connection

- Check the signal strength the device is receiving. If the router is to far away and the signal strenght to weak, this will lead to less bandwidth or even temporary loss of connection.
- Are other devices with data intensive transmission processes connected to the same network? This situation may lead to a remaing bandwidth which is less for your application.
- If many devices using the same channel of your WiFi network, this can lead to interferences. Many newer routers are capable of automatically selecting the least crowded frequencies upon rebooting. If you have the permission to do so, perform a power cycle on your router or, log in to the administrator panel of your router and manually select a different channel.

5.3 Measurement Interfaces

The device can be equipped with different connector types to satisfy different requirements.

5.3.1 D-Sub-Micro-D connector (40 pin MDR connector)

Type: 10240-1210PE

Benefits

- Cables and connectors can be self-assembled.

Disbenefits:

- Large form factor.

5.3.2 30 pin FCS 8 connector

Type: FCS8-30-01-L-S-A-TR

Benefits:

- Small and compact form factor.
- Shielded

Disbenefits:

- Not suitable for self-assembling. Must be ordered pre-assembled.

5.3.3 Custom edge-card-connector (MEC1-130 connector)

Holding fixture for PCBs. For use either with one of Sciospecs PCBs or a custom PCB.

Benefits:

- High flexibility adaptability to special applications.
- The separate measurement electrodes and current sources allow for active probes.
- Other application could be additional AC-coupling, other analog filters or additional biasing.
- The provided power supply and IO-pins allow for run and control of external devices.

Disbenefits:

- High complexity. Requires good understanding of the application.

5.4 Measurement process and timing

5.4.1 Measurement cycle execution

A measurement cycle consists of all processes needed to generate a set or a continuous stream of measurement data. The setup of a measurement is done by using the Sciospec COMInterface or the Sciospec software. When the device receives the start command it generates a proper measurement setup according to the users settings and enters the measurement cycle (see Fig 3: measurement cycle).

The measurement cycle is repeated continuously. In every cycle a dataset is generated that allows to reconstruct one single image. Below these datasets will be referred to as ***"EIT-frame"***. These frames are generated with the selected *Frame Rate*. The measurement cycle is left either after a certain number[1] of EIT-frames or when the stop command is sent.

During data acquisition for an *EIT-frame* the source and sink of the test current injection is automatically changed by the device according to the user settings. The order in which excitation is switched from channel to channel is called ***"Excitation Sequence"***. Exactly one *Excitation Sequence* is needed to complete an *EIT-frame*. For every frame the sequence is repeated in the same order.

An *Excitation Sequence* consists of multiple ***"Excitation Settings"***. An *Excitation Setting* specifies the source and sink electrode for a single current injection setting. The excitation is always disabled before switching to the next setting. If a multi-frequency-measurement is processed, the whole frequency-sweep is finished without interruption before switching to the next excitation setting. To ensure best results, the excitation for the next measurement is started after a sufficient amount of time[2] to let the switches contact certainly.

All measurement data that belongs to a specific Excitation Setting in one Excitation Sequence is called ***"Excitation Dataset"***. Thus an *EIT-frame* consists of as many *Excitation Datasets* as its corresponding *Excitation Sequence* contains *Excitation Settings*.

An Excitation Dataset consists of one or more data frames. A data frame contains all ***"Measurement Values"*** of a 16-channel-group for a single frequency measurement. Hence the number of data frames in an Excitation Dataset equals the number of frequencies in the frequency-sweep.

A *Measurement Value* is a single complex voltage or current value.

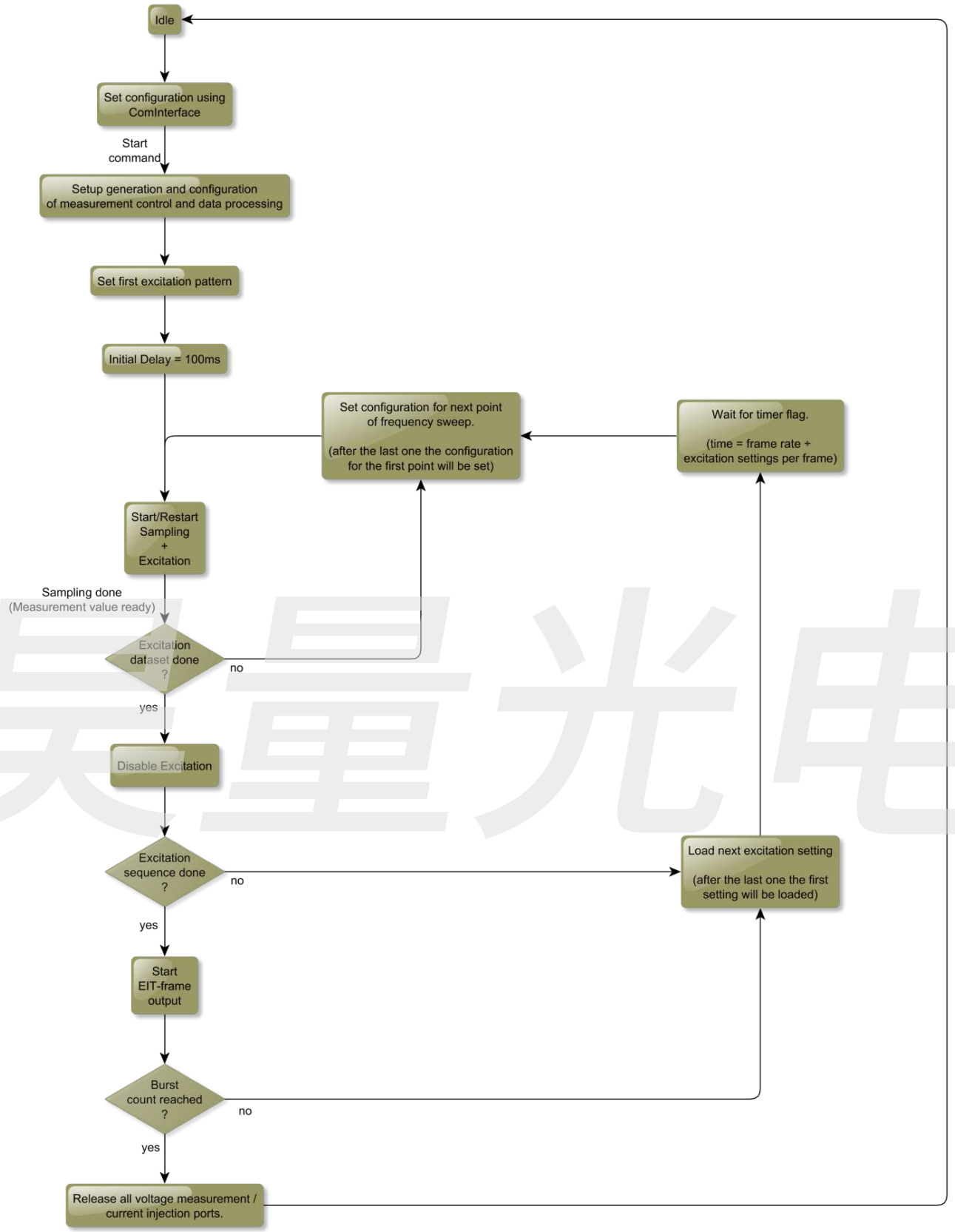


Fig: measurement cycle

For syntax of data frames see Sciospec COMinterface.

6 Measurement Software Description

6.1 Installation

6.1.1 System requirements

- Windows® 7 or more recent

> ⓘ Attach all required extension modules to your device before turn on the device.

6.1.2 Software Installation

- start the downloaded installation file (Sciospec_[device name]_[date].exe) by double click
- follow the instructions

6.1.3 Installation for full speed USB devices

- driver for full speed USB devices are standard drivers for standard operating system (Windows, Linux, macOS)
- no additional installation needed

6.1.4 Installation for high speed USB devices

> ⓘ **Special case**
>
> For the high speed USB connection of the EIT system no installation of drivers is needed.
> Run the Installation before connecting your device to the PC!

- Allow the driver installation in the software installation file

> ⓘ Logfiles are stored at the user directory "%APPDATA%\SciospecSoftware"

6.2 Establishing connection

After the software has started, one can choose the type of connection USB, Ethernet, or Serial. By clicking on [Serial] all devices connected to the USB serial ports of the PC are listed in the connection window. Select the desired device and press [connect]. The main window of the software appears. In case no device is detected, close the software, replug the USB cable and repeat the process. For the devices supporting high-speed USB such as EIT Prototype 2016, click [USB] for device connection.

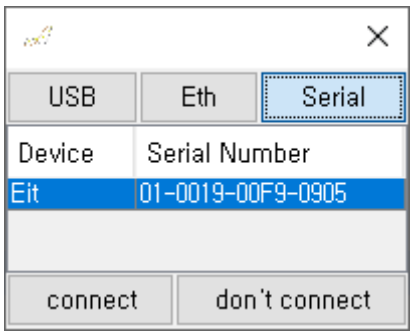


Fig 4: EIT software connection dialog

The option [not connect] gives you the opportunity to load and visualize the data without the need of a connected Sciospec EIT system.

6.3 Measurement Software

The main window opens as shown in the figure below.

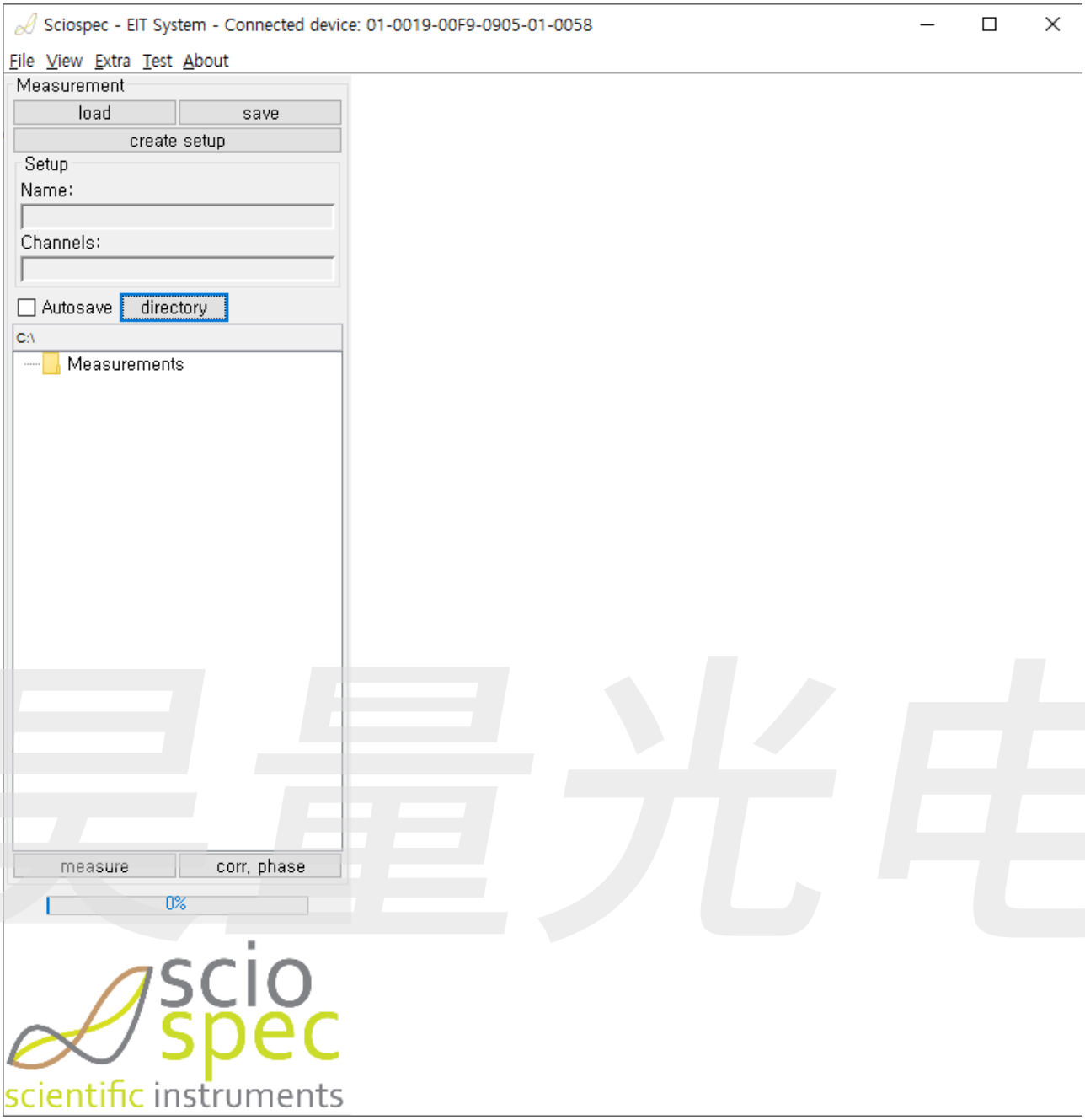


Fig 5: Main window

6.3.1 Measurement setup

By choosing [create setup] a new configuration of the system can be created. The selection of the frequency list can be configured block wise, which enables a very freely distribution of the selected frequency points. First organize the complete setup by selection of the frequency range, measurement range, amplitude of the excitation signal and number of repeats.

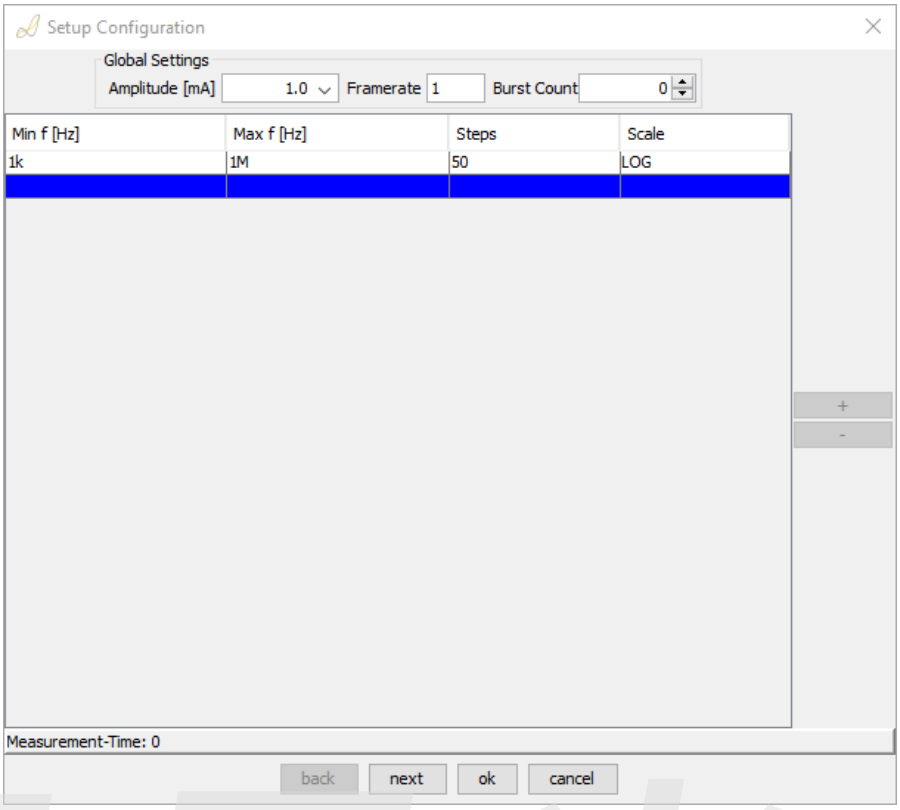


Fig 6: Setup Configuration Dialog

- Amplitude:
  - Amplitude of the current excitation in milliampere
- Burst Count (<256):
  - The number of measurements to be performed.
  - The measurement will stop automatically.
- Setting of the frame rate [Framerate] (>0):
  - The number of measurements to be obtained in a second
  - The configured frame rate may not be achieved depending on measurement settings

The following settings have to be made for each frequency block. It is possible to combine as many frequency blocks as needed. The total number of frequency points is limited to 128. Duplicated frequencies are eliminated automatically.

- Choosing the minimal [Min f], maximal [Max f] frequency and the number of points in between [Steps]
- Selection of the type of scale [Scale] (logarithmic or linear distribution)

Switch to the current injection pattern panel by clicking on [next]

- Select one of pre-configured current injection patterns
  - Adjacent - Default setting

- 1-skip
- 2-skip
- 3-skip
- 4-skip
- 5-skip
- User-defined - With this mode, configured injection pattern can be modified freely. By pressing [clear], all inject rows can be removed. By pressing [clear], an injection row can be added up to 128. To remove a selected injection row, simply press delete key. The injection pattern can be saved or loaded by pressing [save], or [load]

- The pre-configured pattern is decided depending on the channel order. By default it is increasing order. It can be modified freely by selecting channel numbers with the combo boxes. Press [Apply] after modification.

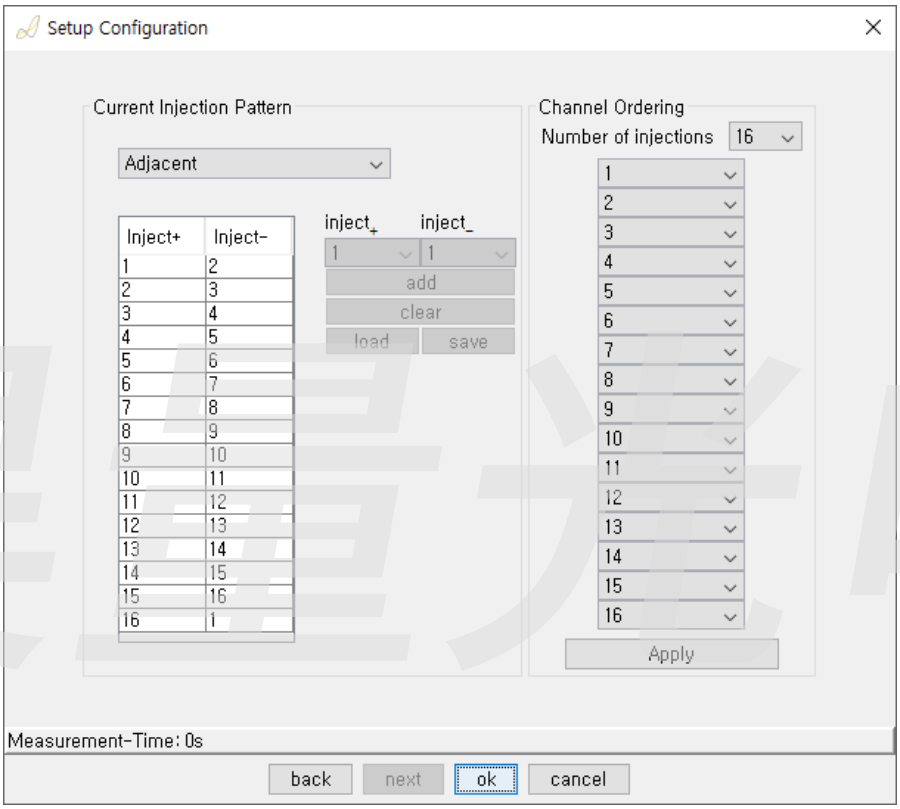


Fig 7: Selection of current injection pattern

Confirm the parameters by pressing [Ok].

6.3.2 Running a measurement

- Choose [measure] to start the measurement
- Options of visualization of the data:
  - In a data table by double clicking on the measured dataset in the data tree. For details, see **EIT Frame**
  - In a static plotter (with absolute value and phase) by right-clicking on a dataset and select [show voltage] and an injection pair. For details see **Voltage Spectrum Plot**

- To stop the measurement press [measure] again
- It is possible to start multiple setups at once. Select them by holding the control key while clicking on each one. Once the measurement has been started each setup will run one after another.

Every time a complete EIT frame is captured it will appear in the data tree

6.3.3 "Autosave" feature

The [Autosave] Option allows to automatically saving the measured data to a previously selected folder.

- Select the path of the folder by clicking on [directory]. (By default is the folder “Data” in the Sciospec EIT folder selected)
- Once a measurement is started a subfolder with the name of the setup configuration will be created and all data will be saved there. The data will not be displayed in the data tree to reduce the amount of memory needed by the software.
- The saved data can be displayed by loading them into the software.
- The major advantage of this feature is that the required program memory will not increase significantly over the runtime of the experiment and all measured data is stored to the hard drive. A fault condition will not compromise the already measured spectra. In case the “Autosave” feature is not active the measured data will only be held in the program memory.

ⓘ It is highly recommended to use the “Autosave” feature when performing long experiments.

6.4 Handling setups and data

Setup and spectrum datasets can be renamed by pressing F2.

6.4.1 Spectra

- Saving data
  - Select the datasets you want to save
  - Press [save]
  - The files will be stored in the selected directory including a number and with the file extension “data” (see **File description „.eit“**)
- Loading data
  - Press [load]
  - Select the datasets you want to load and confirm by pressing [Open]
  - You can also “Drag and Drop” the data files into the software
- The [Autosave] option allows for automatic saving of files to the hard drive.

6.4.2 File description „.eit“

Basically the files can be interpreted as a plain text tab separated (tsv) file with a header and a main part.

- Header part:

Row	Content
1	Number of header rows N (including this one)
2	File Version number
3	Name of dataset
4	Timestamp (yyyy.mm.dd. hh:mm:ss.SSS)
5	Minimum frequency [Hz]
6	Maximum frequency [Hz]
7	Frequency scale (linear = 0, logarithmic = 1)
8	Frequency count
9	Amplitude [A]
10	Frame rate [Frames/s]
11	Phase correction parameter
12	Gain-Setting (0...1; 1...10; 2...100; 3...1000)
13	ADCRange (1... +-1V; 2...+-5V; 3...+-10V)
14	Measure Mode (1...Single Ended; 2...Diff Skip-0; 3...Diff Skip-2; 4...Diff Skip-4)
15	Boundary (1...Internal, 2...External)
16	Switch Type (1... Reed Relais, 2... Semicondutor Switches)
17	MeasurementChannels
18	MeasurementChannelsIndependentfromInjectionPattern

- Main part

Row	Content
1	Injection setting: [Injection $_{+}$] [Injection $_{-}$]
2	Measurement values for 1st frequency Every value for every electrode is represented by its real part and its imaginary part. [re_{E1}] [im_{E1}] [re_{E2}] [im_{E2}] ... [re_{E16}] [im_{E16}]

3	Measurement values for 1nd frequency
...	...
N+1	Measurement values for Nth frequency

This block is repeated for every injection setting.

6.4.3 Setups

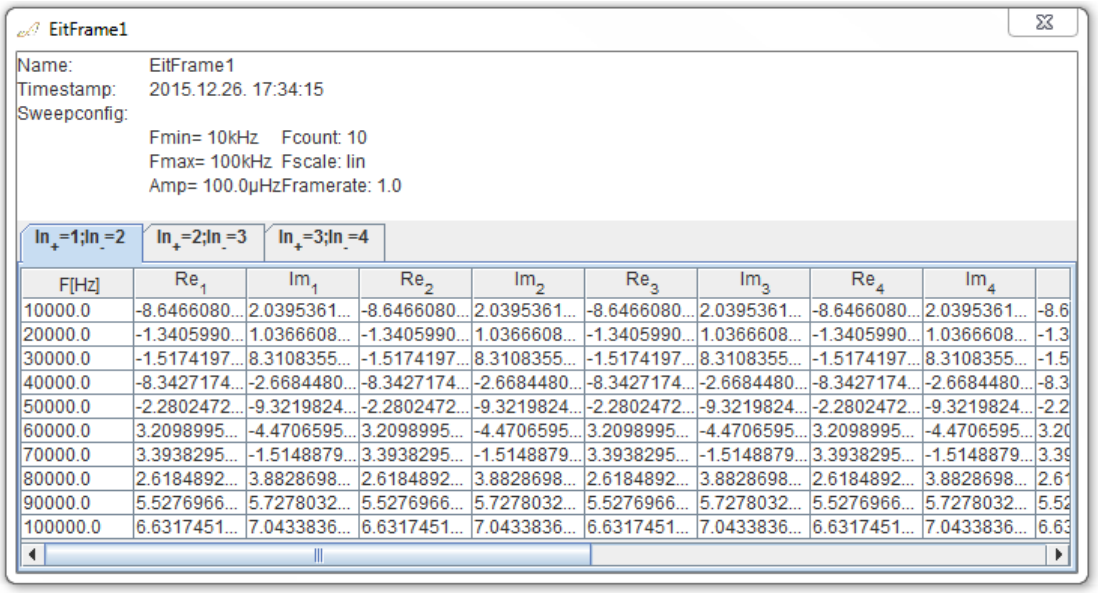
- Saving setup configurations
  - Right click a setup or press on [save] after a setup is selected.
  - You can automatically save all data obtained with this setup by selecting “include sub data”
  - The file extension is „setUp“.
- Loading setup configurations
  - Press [load]
  - Select the file extension „setUp“ from the file filter drop down menu
  - Select the setup file you want to load and confirm by pressing [Open]
  - Or „Drag and Drop“ a setup file into the software

> The EIT data generated using a Sciospec EIT system and corresponding Sciospec software can be imported, processed and visualized in the Matlab based free software EIDORS. Therefore refer to the application note *"Import, process and visualize Sciospec EIT data in the open-source software EIDORS by taking the example of a standard phantom experiment"*.

6.5 Data Visualization

6.5.1 EIT Frame

Measured data which is in the *data tree* can be opened by double clicking on row in the list.



F[Hz]	Re_1	Im_1	Re_2	Im_2	Re_3	Im_3	Re_4	Im_4	
10000.0	-8.6466080...	2.0395361...	-8.6466080...	2.0395361...	-8.6466080...	2.0395361...	-8.6466080...	2.0395361...	-8.6
20000.0	-1.3405990...	1.0366608...	-1.3405990...	1.0366608...	-1.3405990...	1.0366608...	-1.3405990...	1.0366608...	-1.3
30000.0	-1.5174197...	8.3108355...	-1.5174197...	8.3108355...	-1.5174197...	8.3108355...	-1.5174197...	8.3108355...	-1.5
40000.0	-8.3427174...	-2.6684480...	-8.3427174...	-2.6684480...	-8.3427174...	-2.6684480...	-8.3427174...	-2.6684480...	-8.3
50000.0	-2.2802472...	-9.3219824...	-2.2802472...	-9.3219824...	-2.2802472...	-9.3219824...	-2.2802472...	-9.3219824...	-2.2
60000.0	3.2098995...	-4.4706595...	3.2098995...	-4.4706595...	3.2098995...	-4.4706595...	3.2098995...	-4.4706595...	3.20
70000.0	3.3938295...	-1.5148879...	3.3938295...	-1.5148879...	3.3938295...	-1.5148879...	3.3938295...	-1.5148879...	3.39
80000.0	2.6184892...	3.8828698...	2.6184892...	3.8828698...	2.6184892...	3.8828698...	2.6184892...	3.8828698...	2.61
90000.0	5.5276966...	5.7278032...	5.5276966...	5.7278032...	5.5276966...	5.7278032...	5.5276966...	5.7278032...	5.52
100000.0	6.6317451...	7.0433836...	6.6317451...	7.0433836...	6.6317451...	7.0433836...	6.6317451...	7.0433836...	6.63

Fig 8: EIT frame window

This window shows all information saved for this measurement. The upper part of this window shows the name, the timestamp where the measurement took place the sweep settings configured for this measurement.

The lower part shows the measured data itself. Each injection setting is shown in a separate tap. The data in the table can be easily copied to the clipboard by selecting the data which should be copied and pressing CTRL + c. This data is compatible to Microsoft Excel and can also be pasted into an Excel worksheet.

6.5.2 Voltage Spectrum Plot

This plot shows a frequency spectrum of voltage of selected measured data set. To open this plot, select a data set in the data tree and do right-click. Then, a pop-up window will appear with a selection menu. Select [show voltage] and an injection pair. By default, the absolute value and phase are displayed. To display real and imaginary values, hold the control key while opening a plot. To display Nyquist plot, hold the shift key.

Voltage spectra of multiple data sets can be displayed. In this case, another selection menu for channel will be appear after the selection for inject pair.

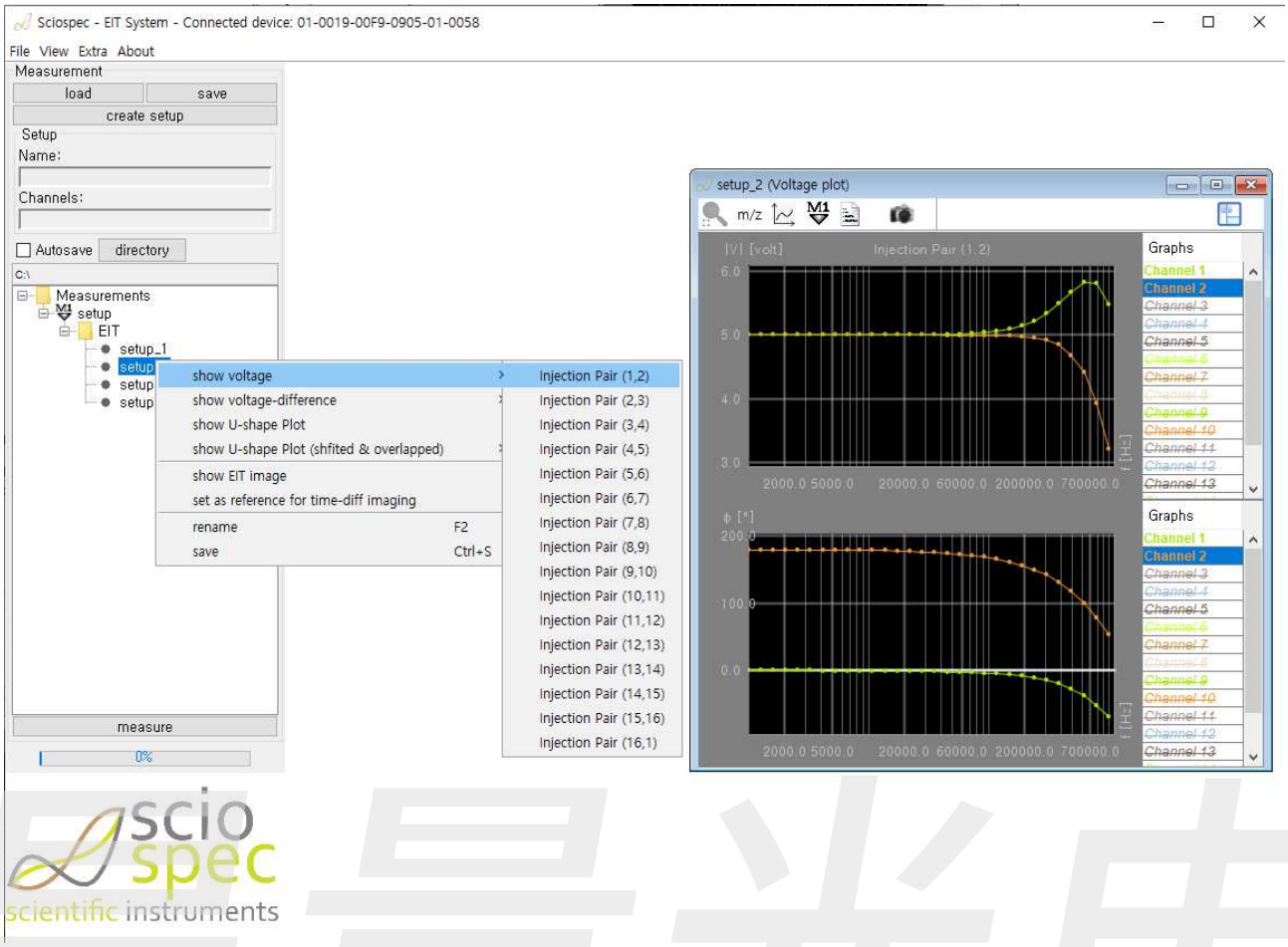


Fig 9: Display of measured data set on plotter

6.5.3 Voltage-Difference Spectrum Plot

This plot shows a frequency spectrum of a voltage-difference between two channels. Do right-click on a data set and select [show voltage-difference] in the selection menu and an injection pair. The channels for the difference are determined by the previously configured current injection pattern in **Measurement setup**. In the case that the current injection pattern is configured as User-defined, the plot shows voltage-difference between neighboring channels. By default, the absolute value and phase are displayed. To display real and imaginary values, hold the control key while opening a plot. To display Nyquist plot, hold the shift key.

6.5.4 U-Shape Plot

This plot shows all the voltage-differences in the same axis. to open this plot, select a data set in the data tree and do right-click. Then, a pop-up window will appear with a selection menu . Select [show U-shape Plot] or [show U-shape Plot (shifted & overlapped)]. U-shape plot is a good indicator for checking measurements when channels are configured in a certain order to a symmetrical object such as a circular phantom. The figure below shows the U-shape plot of a circular phantom with the adjacent current injection pattern. Due to the symmetry of the shape of the phantom, the voltage-differences have a certain pattern and its graph shows U shape for each injection pair as shown in the left plot in the figure. After shifting voltage-differences within in each injection pair and

overlapping the voltage-differences for all injection pairs, its graphs become single U-shape. By default, the absolute value is displayed. To display real values, hold the control key while opening a plot. For displaying phase values, or imaginary value, use **EIT Image Plotter**.

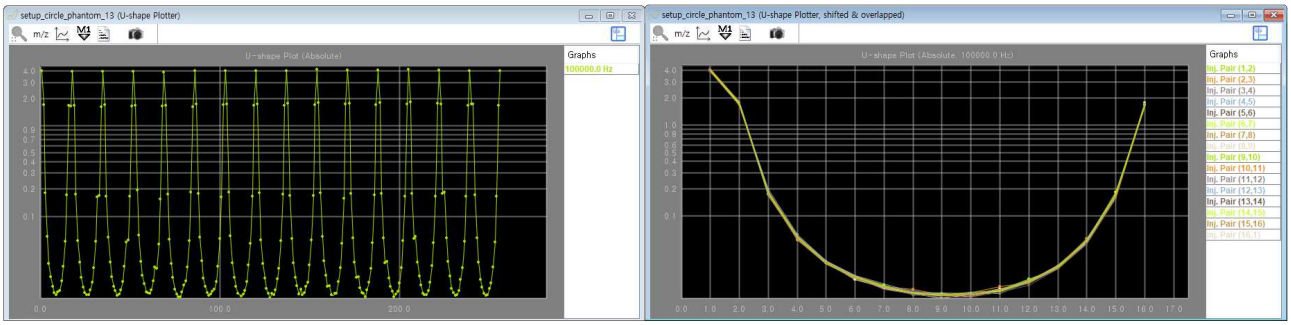


Fig 10: Display of U-shape Plots

6.6 EIT Image Plotter

This plotter performs EIT image reconstructions and plots EIT images. To open this plotter, go to [Extra] in the top menu bar and select [EIT Image Plotter]. Then, EIT image plotter will be appear as shown the figure below. To generate a EIT image, press [Show image] while measurement is running or a data set is selected in the data tree. While measuring, EIT image will be updated continuously for a newly measured data set. EIT image reconstruction is supported only for the adjacent current injection pattern with the circle as the imaging domain. U-shape plot can be monitored while measurement is running by pressing [Monitor data].

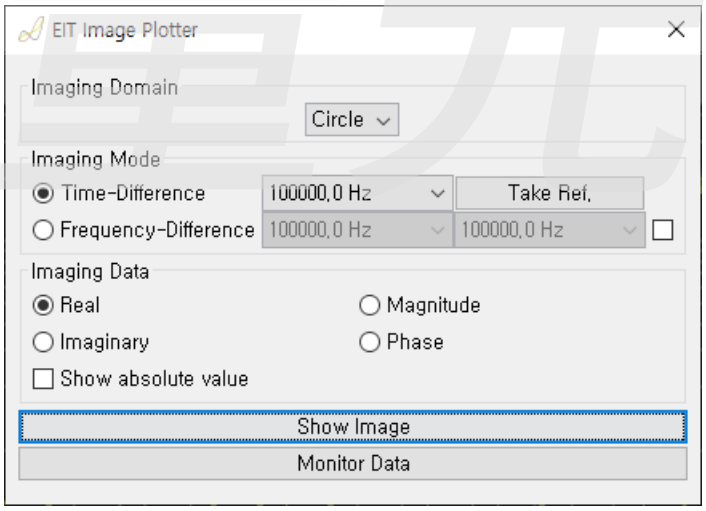


Fig 11: EIT Image Plotter

Imaging Mode

EIT Image Plotter supports the both time-difference and frequency-difference imaging. The time-difference imaging is set by default. The reference data is set as the first EIT frame measured after a measurement running. The reference data can be updated by pressing [Update Ref.], or by selecting [set as a reference for time-diff

imaging] that appears when right-clicking on a data set in the data tree. For the frequency-difference imaging, the data set has to contain more than one frequency point.

Imaging Data

EIT image is generated using real part of voltage data by default. The imaginary, magnitude, or phase values can be used for generating EIT images. This setting also influences the display value type for U-shape plot. In this case, the absolute value of a selected display data type of voltage data can be shown by checking [Show absolute value]. This setting does not influence EIT image.

EIT Image

Whenever press [Show image] in EIT Image Plotter, new EIT Image window appears. Different EIT images for different settings can be displayed. The red and blue colors indicate positive and negative values, respectively. And the black color indicates zero.

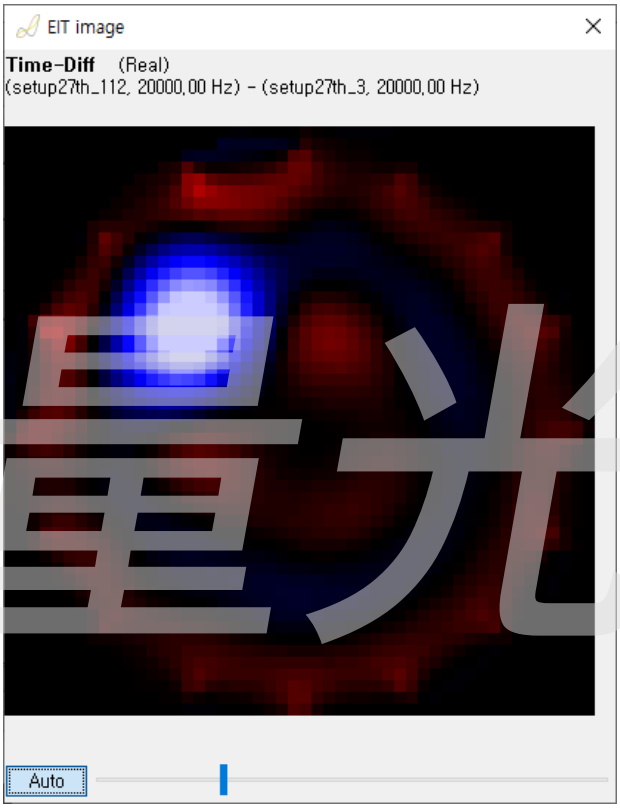


Fig 12: EIT Image

6.7 Voltage-Difference Plotter

The voltage-difference plotter is a tool to visualize voltage difference on an user-defined electrode pair. To open this plotter, go to [Extra] in the top menu bar and select [Voltage Difference Plotter]. After selecting a loaded raw data in the data tree, click 'Voltage Difference Plotter' in 'Extra' menu. Then, you will see a pop-up window, where you can configure voltage measure patterns. As default, the adjacent voltage measure pattern is configured. By selecting 'User-defined' in the combo box, you can configure own measure patterns. You can save or load your patterns into or from a text file.

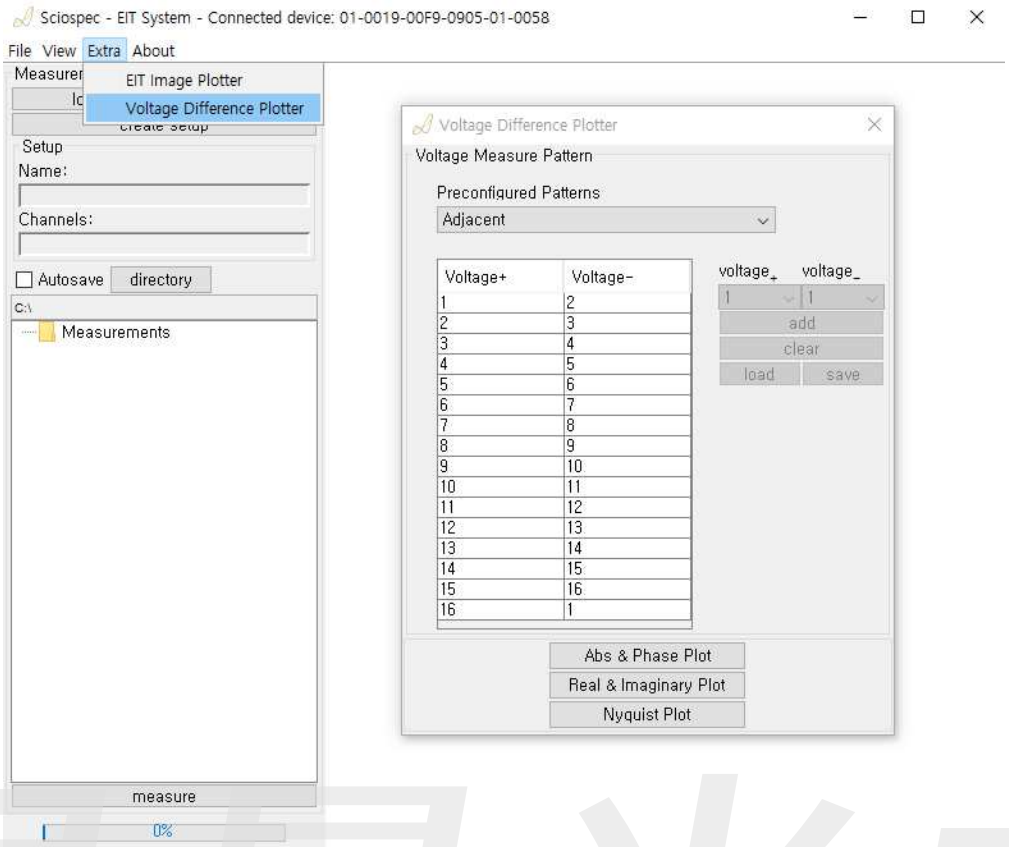


Fig 13: Voltage-Difference Plotter

After configuring the voltage measure pattern, you can create the absolute, phase, real, imaginary, and Nyquist plots by clicking buttons, 'Abs & Phase Plot', ' Real & Imaginary Plot' and 'Nyquist Plot'.

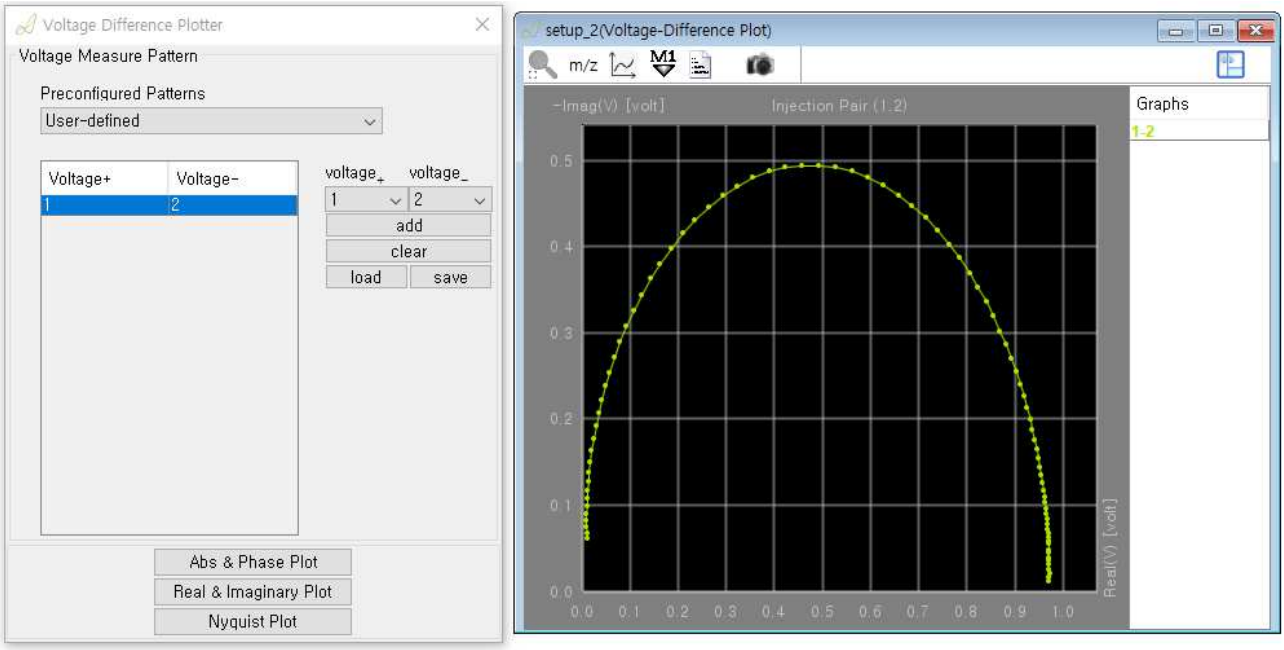


Fig 14: Nyquist Plot

6.7.1 Plotter functions

Action to perform	Function / Result
Middle mouse button, Right click pop-up menu and "Auto zoom" or	Auto zoom
Mouse Wheel or /	Zoom
Left mouse button or and drag	Zooms into selected rectangle
	open diagram configuration panel
Right mouse button and drag	Moving the displayed area
Right click pop-up menu and "Copy zoom" or	Copy the current zoom setting
Right click pop-up menu and "Paste zoom" or	Paste a copied zoom setting
	Saving the displayed data to a ".csv" file
M1	Creating a new marker

Action to perform	Function / Result
	Taking a snapshot as a „png“ file
Double click on a dataset	The absolute value and the phase will be displayed
Holding [shift] while opening data	Only the absolute value will be displayed
Holding [ctrl] while opening data	Only the phase value will be displayed
Holding [ctrl] while operating mouse wheel	Zoom with a lower magnification
Dragging a dataset into an active Plotter	The absolute value and the phase value will be added to the plotter
Double click on the plotter	Full screen mode (Escape to exit full screen)

Graphs panel

Right click on the data	Disabling or enabling the visualization and computing methods of the data
	Hiding the graphs panel

6.8 Error Messages

When the software encounteres a problem, a message window appears, which shows the respective error code and the failed command. The complete list of error codes can be found in subsequent table. For detailed description of the relevant command see Sciospec ComInterface section of the device.

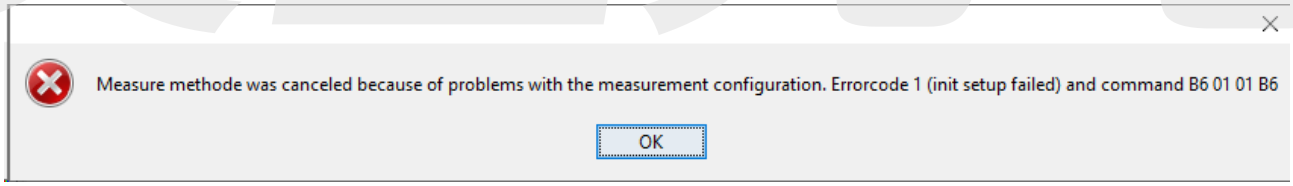


Fig: Error Message window

error code	description
1	init setup failed
2	add frequency block failed
3	set parasitic parameters failed

error code	description
4	set acceleration settings failed
5	set sync time failed
6	set channel settings failed
7	set calibration data failed
8	set timestamp failed
9	start measurement failed
23	set current source failed
24	reset setup failed
25	set burst count failed
26	set excitation amplitude failed
27	set frame rate failed
29	add excitation frequencies failed
30	set extension port channel failed
31	setup add excitation settings failed
32	add frequency block debug failed
33	set output config failed
39	set measure mode failed
40	set excitation switch type failed
41	set time data rate failed
42	set stimulator settings failed
43	set AUX module settings failed
44	set ADC Range settings failed

7 Sciospec Communication Interface

The Sciospec Communication Interface (COMinterface) enables the user to access all functions of the device by using any of the available master interfaces. The actual command structure is identical in all connection types.

7.1 Syntax

The general structure of each communication with a Sciospec device:

- The communication is done by frames
- Each communication frame is constructed as follows

- 1 byte command-Tag (Frame-Start)
- 1 byte number of data-bytes (0...255)
- 0...255 data-bytes
- 1 byte Command-Tag (Frame-End)

- The command-tag identifies the command (see Command list)
- Frame-Start and –End must be identical

Example: "System-Ready-Message"

CMD-Tag	Number of bytes	Data	CMD-Tag
0x18	0x01	0x84	0x18

7.2 Acknowledge messages

- Communication-frames with incorrect syntax will cause a “Frame-Not-Acknowledge” message
- If the transmission of a communication-frame is interrupted for more than 10 ms a “Timeout” message is send
- Every invalid command-tag will cause a “Not-Acknowledge” message
- Every valid command is acknowledged with an acknowledge command [ACK]
- For commands with a return value the returning frame comes before the acknowledge message
- When commands are sent during the current measurements, measurement data can be transmitted between the command and the following returning frame and the acknowledge-message (commands are handled asynchronously)
- Before sending a new command, the resulting acknowledge or not acknowledge of the previous command has to be awaited.

The ACK-Frame:

0x18	0x01	[ACK]	0x18

General System Messages	
0x01	Frame-Not-Acknowledge: Incorrect syntax

General System Messages	
0x02	Timeout: Communication-timeout (less data than expected)
0x04	Wake-Up Message: System boot ready
0x11	TCP-Socket: Valid TCP client-socket connection
0x81	Not-Acknowledge: Command has not been executed
0x82	Not-Acknowledge: Command could not be recognized
0x83	Command-Acknowledge: Command has been executed successfully
0x84	System-Ready Message: System is operational and ready to receive data
0x92	Data holdup: Measurement data could not be sent via the master interface

7.3 Abbreviations

abbreviation	full name
[CT]	command tag
[LE]	length
[OB]	option byte
[CD]	command data

7.4 Command list

The leading hex code of each command heading represents the [command code] of the respective function.

7.5 Command description

7.5.1 0x90 - Save settings

General Syntax

[CT] 00 [CT]

Return

ACK

Description

Saves the following parameters permanently into the flash memory of the EIT system:

- Ethernet parameters (DHCP, static IP)
- WLAN parameters (SSID, DHCP, autoConnect, StaticIpSettings)
- battery mode parameter (MeasureOnBat, minCapacity)
- LED automode

This command can only be used if no measurement is currently running.

7.5.2 0xA1 - Software Reset

General Syntax

[CT] 00 [CT]

Return

ACK

Wake-Up Message

System-Ready-Message

Description

This command causes a reboot of the Sciospec EIT-Device.

7.5.3 0xB0 - Set Measurement Setup

General Syntax

[CT] [LE] **[OB] [CD]** [CT]

Return

ACK

[OB]

Function	code
Reset Setup	0x01
Burst Count	0x02
Frame Rate	0x03
Excitation Frequencies	0x04
Excitation Amplitude (Double Precision)	0x05
Excitation Amplitude (Single Precision)	0x05
Add Excitation Setting to excitation sequence	0x06
Set Single-Ended or Differential Measure Mode	0x08
Gain Settings	0x09
Excitation Switch Type	0x0C
ADC Range	0x0D

Reset Setup (0x01)

Reset Setup to Default Values. Set all settings to their default value.

Syntax

- Syntax set: **[CT] 01 01 [CT]**

Burst Count (0x02)

Number of frames generated before measurement stops automatically.

Syntax

- Syntax set: **[CT] 03 02 [burst count] [CT]**

[burst count]

- 2 Byte unsigned integer
- value range: 0 - 65535
- For continuous streaming set to 0.
- default: [CD] = 0

Frame Rate (0x03)

Number of EIT-frames per second. Refer to section "Maximum allowable frame rate" for constraints and dependencies.

Syntax

- Syntax set: **[CT] 05 03 [frame rate] [CT]**

[frame rate]

- 4 Byte float
- value range: 0.1 - 100
- default: frame rate = 1 frame/s

Excitation Frequencies (0x04)

Add excitation frequency block in Hz.
It is possible to add multiple blocks to the stack (limited by a max total number ot 128 frequency settings , sum over f_{count} of all added blocks.

Syntax

- Syntax set: **[CT] 0C 04 [f_{min}] [f_{max}] [f_{count}] [f_{type}] [CT]**

[f_{min}]

- minimum frequency f_{min}
- 4 Byte floating point single precision value
- range = 100 Hz - 10 MHz
- Default: f_{min} = 100 kHz

[f_{max}]

- maximum frequency f_{max}
- 4 Byte floating point single precision value
- range = 100 Hz - 10 MHz
- Default: f_{max} = 100 kHz

[f_{count}]

- frequency count f_{count}
- 2 Byte unsigned integer value
- range = 1 - 128
- Default: f_{count} = 1

[f_{type}]

- frequency type f_{type}
- 1 Byte unsigned interger value
- f_{type} = 0: linear frequency distribution | 1: logarithmic frequency distribution
- Default: f_{type} = 0

ⓘ Default: f_{min} = 100 kHz, f_{max} = 100 kHz, f_{count} = 1, f_{type} = 0
This default is only valid after system power up or after a software reset. After sending the **Reset Setup (OB = 0x01)** no frequency is set to the system.

Excitation Amplitude (Double Precision) (0x05)

Set excitation amplitude in ampere.

Syntax

- Syntax set: **[CT] 09 05 [excitation amplitude] [CT]**

[excitation amplitude]

- 8 Byte Floating Point double precision value.
- A_{min} = 100 nA
- A_{max} = 10 mA
- Step size see Chapter “Technical Specification”
- Default: A = 0.01 A

Excitation Amplitude (Single Precision) (0x05)

Set excitation amplitude in ampere.

Syntax

- Syntax set: **[CT] 05 05 [excitation amplitude] [CT]**

[excitation amplitude]

- 4 Byte Floating Point single precision value.
- A_{min} = 100 nA
- A_{max} = 10 mA
- Step size see Chapter “Technical Specification”
- Default: A = 0.01 A

Add Excitation Setting to excitation sequence (0x06)

Each excitation setting must be added with a separate command. The excitation settings are placed in the excitation sequence in the same order as transmitted to the system.

Syntax

- Syntax set: **[CT] 03 06 [CD_{out}] [CD_{in}] [CT]**

[CD_{out}] / [CD_{in}]

- each 1 Byte unsigned integer value or 2 Byte unsigned integer value (in case of 2 Byte, data length of the communication frame is 0x05)
- Every set of [CD_{out}] / [CD_{in}] represents one excitation setting. To clear the excitation sequence stack an the **Reset Setup (OB = 0x01)** must be send to the system.
- Maximum count of configurable excitation setting: 256
- Setting **CD_{out}** to 0 "inject out" is disabled.
- Setting **CD_{in}** to 0 "inject in" is disabled.
- Default: 1→2; 2→3; 3→4; 4→5; 5→6; 6→7; 7→8; 8→9; 9→10; 10→11; 11→12; 12→13; 13→14; 14→15; 15→16; 16→1

> ⓘ This default is only valid after system power up or after a software reset. After sending the **Reset Setup (OB = 0x01)** no excitation setting is set to the system.

Set Single-Ended or Differential Measure Mode (0x08)

Configure hardware supported differenential measurements.

Syntax

- Syntax set: **[CT] 03 08 [Mode] [Boundary] [CT]**

[Mode]

- Length: 1 byte
- possible values are
  - Mode = 0x01: Single ended (default after system powerup or reset setup)
  - Mode = 0x02: Differential Skip 0 (Neighbour)
  - Mode = 0x03: Differential Skip 2
  - Mode = 0x04: Differential Skip 4

[Boundary]

This is only valid for Differential Measurement Modes Skip 0,2 or 4.

It allows to take the differences inside one channel group of 16 channels (internal) or take the differences across all channel groups mounted in a system (external).

- Length: 1 byte
- possible values are
  - option = 0x01: Internal
  - option = 0x02: External (this is only valid if more than one channel group is available)

Example 1: 64 channel system with selected Differential Skip 0 Setting and internal boundary conditions. The outputted data is as follows.

1-2,2-3,...,15-16,16-1, 17-18,18-19,...,31-32, 32-17, 33-34,34-35,...,47-48,48-33, 49-50,50-51,...,63-64,64-49

Example 2: 64 channel system with selected Differential Skip 0 Setting and external boundary conditions. The output data is as follows.

1-2,2-3,...,15-16,16-17, 17-18,18-19,...,31-32, 32-33, 33-34,34-35,...,47-48,48-49, 49-50,50-51,...,63-64,64-1

In the case of a 16 channel system it is only possible to differentially measure 16 channels in a circular arrangement (internal setting only), use single ended measurements for all other settings.
In the case of a 32 channel system it is only possible to differentially measure 16 or 32 channels in a circular arrangement, use single ended measurements for all other settings.
In the case of a 64 channel system it is only possible to differentially measure 16 or 64 channels in a circular arrangement, use single ended measurements for all other settings.
In the case of a 128 channel system it is only possible to differentially measure 16 or 128 channels in a circular arrangement, use single ended measurements for all other settings.

Gain Settings (0x09)

Allows to configure the gains for the measurement in single ended and differential mode. In the case of single ended mode the voltage at each electrode is measured including the gain. In differential mode the difference is measured and then multiplied by the gain.

The Gain mode setting is resetted by the reset Setup-Command (Optionbyte = 0x01). In this case Mode = 0x01 (Gloabal Gain) with Gain = 1 would be default. The individual gain stack is resetted too. Every gain setting for all injections and all channels will be set to gain = 1.

The Gain-Mode that is used for measurement depends on the last send command.

Syntax

- Syntax set: **[CT] [LE] 09 [Mode] [Data] [CT]**

[Mode]

- Length **[LE]**: 2 byte + Length of data for specific mode setting
- possible values are
    - Mode = 0x01: Global Gain (Default)
        - Data Length: 1byte
        - [Data] Gain:
            - 0x00 = 1; (Default)
            - 0x01 = 10;
            - 0x02 = 100;
            - 0x03 = 1000;

Excitation Switch Type (0x0C)

For the excitation two different types of switches can be used. By default reed relays switches with a low on resistance (around 100mOhm) and a low parasitic capacitance (around 0.3pF), but a higher switching delay (around 600µs) are used. Alternatively semiconductor switches can be used. They provide a higher switching speed (around 200ns), but a high on resistance (around 300Ohm) and a slightly higher parasitic capacitance (around 3pF).

Syntax

- Syntax set: **[CT] 02 0C [Type] [CT]**

[Type]

- ReedRelais = 0x01 (Default)
- Semiconductor switch = 0x02

> The use of semiconductor switches allows for higher frame rates compared to reed relays for certain configurations. Please be aware, the use semiconductor switches adds a relativly higher output impedance on both positive and negative injection electrodes.

ADC Range (0x0D)

For the measurement multiple different settings for the voltage input range of the analog to digital converters are available. Depending if the device is battery powered or not voltages up to +-10V can be measured.

Syntax

- Syntax set: **[CT] 02 0D [Range] [CT]**

[Range]

- +-1V = 0x01
- +-5V = 0x02 (default if battery powered)
- +-10V = 0x03 (default for not battery powered systems, not available for battery powered systems

ⓘ This setting applies to all channels of the EIT System. Values above the limit are clipped to the highest possibile value. This does not affact the maximum allowable voltage level which can be applied to this electrode, so no harm is done on the electrode if the setting is f.e. set to +-1V and 5V are applied.

7.5.4 0xB1 - Get Measurement Setup

General Syntax

[CT] **[LE] [OB]** [CT]

Return

[CT] **[LE] [OB] [CD]** [CT]

ACK

[OB]

Function	code
Burst Count	0x02
Frame Rate	0x03
Excitation Frequencies	0x04
Excitation Amplitude	0x05
Excitation Sequence	0x06
Single-Ended or Differential Measure Mode	0x08
Gain Settings	0x09
Excitation Switch Type	0x0C
ADC Range	0x0D

Burst Count (0x02)

Returns the number of frames generated before measurement stops automatically.

Syntax

- Syntax get: **[CT] 01 02 [CT]**
  - Return:**[CT] 03 02 [burst count] [CT]**

[burst count]

- 2 Byte unsigned integer
- If Value equals 0 the streaming won't stop automatically.

Frame Rate (0x03)

Returns the number of full frames per second.

Syntax

- Syntax get: **[CT] 01 03 [CT]**
  - Return:**[CT] 05 03 [frame rate] [CT]**

[frame rate]

- 4 Byte float
- value range = 0.1 - 100
- Depending on the number of **Active Channels** setting, the number of **Excitation Frequencies** and the **Excitation Sequence** the returned value could deviate from the set value.

Excitation Frequencies (0x04)

Returns the excitation frequency settings.

Syntax

- Syntax get: **[CT] 01 04 [CT]**
  - Return:**[CT] [LE] 04 [f_{min}** 1st block**] [f_{max}** 1st block**] [f_{count}** 1st block**] [f_{type}** 1st block**] [f_{min}** 2nd block**] [f_{max}** 2nd block**] [f_{count}** 2nd block**] [f_{type}** 2nd block**] ... [CT]**

[LE]

- Length depending on the added frequency blocks

[f_{min}]

- minimum frequency f_{min} of corresponding block
- 4 Byte floating point single precision value

[f_{max}]

- maximum frequency f_{max} of corresponding block
- 4 Byte floating point single precision value

[f_{count}]

- frequency count f_{count} of corresponding block
- 2 Byte unsigned integer value

[f_{type}]

- frequency type f_{type} of corresponding block
- 1 Byte unsigned integer value
- f_{type} = 0: linear frequency distribution | 1: logarithmic frequency distribution

Excitation Amplitude (0x05)

Returns the excitation amplitude in ampere.

Syntax

- Syntax get: **[CT] 01 05 [CT]**
  - Return:**[CT] 05 05 [excitation amplitude] [CT]**

[excitation amplitude]

- 4 Byte floating point single precision value.

Excitation Sequence (0x06)

Returns the complete excitation sequence.

Syntax

- Syntax get: **[CT] 01 06 [CT]**
  - Return:**[CT] [LE] 06 [excitation sequence] [CT]**

[LE]

- Depending on stack length.
- [LE] = (stack length) * 2 Byte/Value * 2Values + 1

[excitation sequence]

- [excitation sequence] = [excitation output port 1] [excitation input port 1] [excitation output port 2] [excitation input port 2] ... [excitation output port x] [excitation input port x]

Single-Ended or Differential Measure Mode (0x08)

Returns the current configured single-ended or differential measure mode and boundary settings.

Syntax

- Syntax get: **[CT] 01 08 [CT]**
  - Return: **[CT] 03 08 [Mode] [Boundary] [CT]**

[Mode]

- Length: 1 byte
- possible values are
  - Mode = 0x01: Single ended (default after system powerup or reset setup)
  - Mode = 0x02: Differential Skip 0 (Neighbor)
  - Mode = 0x03: Differential Skip 2
  - Mode = 0x04: Differential Skip 4

[Boundary]

This is only valid for Differential Measurement Modes Skip 0,2 or 4.

It allows to take the differences inside one channel group of 16 channels (internal) or take the differences across all channel groups mounted in a system (external).

- Length: 1 byte
- possible values are
  - option = 0x01: Internal
  - option = 0x02: External (this is only valid if more than one channel group is available)

Gain Settings (0x09)

Returns the configured gain settings.

Syntax

- Syntax get: **[CT] 01 09 [CT]**
- Return: **[CT] [LE] 09 [Mode] [Data] [CT]**

[Mode]

- Length **[LE]**: 2 byte + Length of data for specific mode setting
- possible values are
  - Mode = 0x01: Global Gain
    - Data Length: 1byte
    - [Data] Gain:
      - 0x00 = 1;
      - 0x01 = 10;
      - 0x02 = 100;
      - 0x03 = 1000;

Excitation switch type (0x0C)

Returns the current configured excitation switch type.

Syntax

- Syntax get: **[CT] 01 0C [CT]**
  - Return: **[CT] 02 0C [Type] [CT]**

[Type]

- Length: 1 byte
- possible values are
  - Type = 0x01: Reed Relay Switches (default after system powerup or reset setup)
  - Type = 0x02: Semiconductor Switches

ADC Range (0x0D)

Returns the current configured ADC range setting.

Syntax

- Syntax get: **[CT] 01 0D [CT]**
  - Return: **[CT] 02 0D [Range] [CT]**

[Range]

- Length: 1 byte
- possible values are
  - Range = 0x01: +-1V
  - Range = 0x02: +-5V
  - Range = 0x03: +-10V

7.5.5 0xB2 - Set Output Configuration

This command is used to enable or disable additional information in output data stream of measured data (see section "Measured Data"). This command is only valid while no measurement is ongoing. In this case a not acknowledge (NACK) is returned.

General Syntax

[CT] **[LE] [OB] [CD]** [CT]

Return

ACK

[OB]

Function to enable / disable	code
Excitation setting	0x01
Current row in the frequency stack	0x02
Timestamp	0x03

Excitation setting

Enable or disable Excitation setting (additional 2 Byte in output stream).

Syntax

- Syntax set: **[CT] 02 01** [enable/disable] [CT]

[enable/disable]

- 1 Byte unsigned integer value
- 0 - disable, 1 - enable

Current row in the frequency stack

Enable or disable current row in the frequency stack (additional 2 Byte in output stream)

Syntax

- Syntax set: **[CT] 02 02 [enable/disable] [CT]**

[enable/disable]

- 1 Byte unsigned integer value
- 0 - disable, 1 - enable

Timestamp

Enable or disable timestamp (additional 4 Byte in output stream).

Syntax

- Syntax set: **[CT] 02 03 [enable/disable] [CT]**

[enable/disable]

- 1 Byte unsigned integer value
- 0 - disable, 1 - enable

ⓘ By default all settings are disabled.

7.5.6 0xB3 - Get Output Configuration

Returns the display option in the data stream of measured date.

General Syntax

[CT] **[LE] [OB]** [CT]

Return

[CT] **[LE] [OB] [enable/disable]** [CT]

ACK

[OB]

Function to enable / disable	code
Excitation setting	0x01
Current row in the frequency stack	0x02
Timestamp	0x03

Excitation setting

Returns if Excitation setting is disabled or enabled (additional 2 Byte in output stream).

Syntax

- Syntax get: **[CT] 01 01 [CT]**
  - Return: **[CT] 02 01 [enable/disable] [CT]**

[enable/disable]

- 1 Byte unsigned integer value
- 0 - disable, 1 - enable

Current row in the frequency stack

Returns if current row in the frequency stack is disabled or enabled (additional 2 Byte in output stream).

Syntax

- Syntax get: **[CT] 01 02 [CT]**
  - Return: **[CT] 02 02 [enable/disable] [CT]**

[enable/disable]

- 1 Byte unsigned integer value
- 0 - disable, 1 - enable

Timestamp

Returns if timestamp is disabled or enabled (additional 4 Byte in output stream).

Syntax

- Syntax get: **[CT] 01 03 [CT]**
  - Return: **[CT] 02 03 [enable/disable] [CT]**

[enable/disable]

- 1 Byte unsigned integer value
- 0 - disable, 1 - enable

ⓘ By default all settings are disabled.

7.5.7 0xB4 - Start/Stop Measurement

A new measurement can only be started when no other measurement is ongoing. A description is can be found in section "Measured Data".

General Syntax

[CT] **[LE] [OB]** [CT]

Return

ACK

[OB]

Function	code
Stop Measurement	0x00
Start EIT Measurement	0x01

Stop Measurement

Syntax

- Syntax set: **[CT] 01 00 [CT]**

Start Measurement

Syntax

- Syntax set: **[CT] 01 01 [CT]**

7.5.8 0xB5 - Get temperature

General Syntax

[CT] 01 **[TempSensor]** [CT]

Return

[CT] 05 **[TempSensor]** **[Temperature]** [CT]

ACK

Description

This commands gets the temperature of the internal controller- or frontend- temperature sensors.

[TempSensor]

- Length: 1 byte

Funktion	code
gets internal controller temperature	0x00
gets frontend temperature	0x01

[Temperature]

- Length: 4 byte
- Data format: Float value
- Unit: °C

7.5.9 0xBD - Set Ethernet Configuration

Configure DHCP setting and IP address.

General Syntax

[CT] **[LE]** **[OB]** **[CD]** [CT]

Return

ACK

[LE]

- Represents the byte count of the command frame and varies with the amount [data] bytes. [length] is always the amount of [data]-bytes + 1.

[CD]

- Command data whose syntax and content depends on the used [option byte].

[OB]

Function	code
IP address	0x01
DHCP on/of	0x03

IP address

- Set static IPv4 address of the device.
- This address will only be used, when DHCP is disabled.
- The default setting is 0.0.0.0
- Syntax: **[CT] [05] [01] [address] [CT]**
- [address]
  - Desired address.
  - Length: 4 byte
  - Data format: byte-wise unsigned integer
  - Value range of each byte: 0 ... 255

DHCP on/off

- Activate/deactivate DHCP usage.
- To apply this setting, a system reboot is required.
- Syntax: **[CT] [02] [03] [switch] [CT]**
- [switch]
  - Length: 1 byte
  - Values:
    - 0x00 - off
    - 0x01 - on (default)

To save these settings persistently, the "save settings" command must be sent before powering off the device.

7.5.10 0xBE - Get Ethernet Configuration

Read out parameters like DHCP setting, MAC and IP address.

General Syntax

[CT] **[LE] [OB]** [CT]

Return

[CT] **[LE] [OB] [CD]** [CT]
ACK

[LE]

- Represents the byte count of the command frame and varies with the amount [data] bytes. [length] is always the amount of [data]-bytes + 1.

[CD]

- Command data whose syntax and content depends on the used [option byte].

[OB]

Function	get
IP address	0x01
MAC address	0x02
DHCP on/off	0x03

IP address

- Read out currently allocated IPv4 address of the device. This is the actual address the device
- Get syntax: **[CT] [01] [01] [CT]**
- Return syntax: **[CT] [05] [01] [address] [CT]**
- [address]
  - Current address.
  - Length: 4 byte
  - Data format: byte-wise unsigned integer

MAC address

- Read out currently allocated MAC address of the device. This is the actual address the device
- Get syntax: **[CT] [01] [02] [CT]**
- Return syntax: **[CT] [07] [02] [address] [CT]**
- [address]
  - MAC address.
  - Length: 6 byte
  - Data format: byte-wise unsigned integer

DHCP on/off

- Read out the current DHCP setting.
- Get syntax: **[CT] [01] [03] [CT]**
- Return Syntax: **[CT] [02] [03] [switch] [CT]**
- [switch]
  - Length: 1 byte
  - Values:
    - 0x00 - off
    - 0x01 - on

7.5.11 0xC6 - Set Battery Control

General Syntax

[CT] [LE] **[OB] [CD]** [CT]

Return

ACK

[OB]

Function	set
Measure on Battery	0x02
Measure on Battery min capacity	0x03

Measure on Battery

Set the Battery Mode to measure on battery.

Syntax

- Syntax set: **[CT] 02 02 [CD] [CT]**

[CD]

- 0x00: Disable
- 0x01: Enable

Measure on Battery min capacity

Set the Battery Mode to change to external power supply when the battery min capacity is reached. Measurements will be stopped.

Syntax

- Syntax set: **[CT] 02 03 [CD] [CT]**

[CD]

- minimum state of charge in percent (≤ 100)

7.5.12 0xC7 - Get Battery Control

General Syntax

[CT] [LE] **[OB] [CD]** [CT]

Return

[CT] [LE] **[OB] [status] [remaining capacity]** [CT]

ACK

[OB]

Function	get
Status / remaining capacity	0x01
Measure on Battery	0x02
Measure on Battery min capacity	0x03

Status / remaining capacity

Gets Status of Battery Mode and remaining capacity of battery.

Syntax

- Syntax get: **[CT] 01 01 [CT]**
  - Return: **[CT] [LE] 01 [status] [remaining capacity] [CT]**

[status]

1 Byte bit coded

MsBit - Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	LsBit - Bit 0
d. c.	d. c.	d. c.	d. c.	d. c.	d. c.	powers source	charging

power source:

- 1 = battery
- 0 = external

[remaining capacity]

- 1 byte hexadecimal remaining capacity in percent (≤ 100 percent)

Measure on Battery

Set the Battery Mode to measure on battery.

Syntax

- Syntax get: **[CT] 01 02 [CT]**
  - Return: **[CT] [LE] 02 [CD] [CT]**

[CD]

- 0x00: Disable
- 0x01: Enable

Measure on Battery min capacity

Set the Battery Mode to change to external power supply when the battery min capacity is reached. Measurements will be stopped.

Syntax

- Syntax get: **[CT] 01 03 [CT]**
  - Return: **[CT] [LE] 03 [CD] [CT]**

[CD]

- minimum state of charge in percent (≤ 100)

7.5.13 0xC8 - Set LED Control

Set the LED control to automatic or manual mode. In manual mode LEDs can be enabled or disabled by command.

General Syntax

[CT] **[LE] [OB] [CD]** [CT]

Return

ACK

[OB]

Function	code
Activate / deactivate automode	0x01
Set Manual LED status	0x02

Activate / deactivate automode

Syntax

- Syntax set: **[CT] 03 01 [LED#] [disable/enable] [CT]**

[LED#]

- Length: 1 byte integer
- LED number
- 0x01 ≤ LED# ≤ 0x04

[disable/enable]

- Length: 1 byte integer
- 0x00 - disable
- 0x01 - enable

Set Manual LED status

Syntax

- Syntax set: **[CT] [03] 02 [LED#] [disable/enable/blink] [CT]**

[LED#]

- Length: 1 byte integer
- LED number
- 0x01 ≤ LED# ≤ 0x04

[disable/enable]

- Length: 1 byte integer
- 0x00 - disable
- 0x01 - enable
- 0x02 - blink

7.5.14 0xC9 - Get LED Control

Gets the mode of the LED control: automatic or manual mode. Gets the status of the LEDs.

General Syntax

[CT] **[LE] [OB] [LED#]** [CT]

Return

[CT] **[LE] [OB] [CD]** [CT]

ACK

[OB]

Function	code
Get LED control mode	0x01
Get Manual LED status	0x02

Get LED control mode

Syntax

- Syntax Get: **[CT] 02 01 [LED#] [CT]**
  - Return: **[CT] 03 01 [LED#] [disable/enable] [CT]**

[LED#]

- Length: 1 byte integer
- LED number
- 0x01 ≤ LED# ≤ 0x04

[disable/enable]

- Length: 1 byte integer
- 0x00 - disable
- 0x01 - enable

Get Manual LED status

Syntax

- Syntax get: **[CT] 02 02 [LED#] [CT]**

- Return: **[CT] [03] 02 [LED#] [disable/enable/blink] [CT]**

[LED#]

- Length: 1 byte integer
- LED number
- 0x01 ≤ LED# ≤ 0x04

[disable/enable/blink]

- Length: 1 byte integer
- 0x00 - disable
- 0x01 - enable
- 0x02 - blink

7.5.15 0xCB - FrontIOs (optional, contact Sciospec if you device is equipped with this feature)

Configure and read the state of the FrontIOs.

General Syntax

[CT] **[LE] [OB] [CG] [IO] [ST]** [CT]

Return

[CT] **[LE] [OB] [CG] [IO] [ST]** [CT]

ACK

[OB]

	set	get
IO direction	0x01	0x81
IO state	0x02	0x82

[CG] - Channel group

- Length: 1 byte
- Data format: unsigned integer
- CG ≥ 1
- 16 channels are grouped into one group

[IO] - IO number

There are 4 IOs per channel group available: 1 ≤ IO ≤ 4
In case of OP=0x82 (get input state) the IO number is don't care and can be omitted. In this case all IO states are read at the same time.

- Length: 1 byte
- Data format: unsigned integer

[ST] - IO state

This is only relevant for OP = 0x01 (set IO direction) or OP = 0x02 (set output state). For all other options this is don't care and can be omitted.
set IO direction: ST must be either 0x01 (output) or 0x00 (input, default).
set output state: ST must be either 0x01 (high state) or 0x00 (low state).

- Length: 1 byte
- Data format: unsigned integer

Example:

- set IO direction for IO#1 to output
  CB 04 01 01 01 01 CB
- set IO state of IO#1 to low
  CB 04 02 01 01 00 CB
- get Input states:
  CB 02 82 01 CB
  Return: CB 03 82 01 0C CB

7.5.16 0xCC - Power Plug Detect

Gives information if the DC power plug is inserted.

General Syntax

[CT] **[LE] [OB]** [CT]

[LE]

- depending on option byte

OB 0x81 get IO state

Return: [CT] 02 81 **[state]** [CT]

[state]:

- 0x01 DC power plug is inserted
- 0x00 DC power plug is not inserted

7.5.17 0xCE - Set LTC

General Syntax

[CT] [LE] **[OB] [CD]** [CT]

Return

ACK

[OB]

Enable LTC Signal transmission	0x01

Enable LTC Signal transmission

Enable the transmission of the LTC-Data while the measurement is running.

Syntax

- Syntax set: **[CT] 03 01 [interval] [CT]**

[interval]

- 2 Byte unsigned integer
- value range: 0 - 65535
- interval = 0: transmission of time stamp is disabled
- interval != 0: timestamp will be transmitted every x EIT-Frame. First transmission will be right before the first data frame of the measurement.

Response

CE 0F 01 [LTC] [LTC] [LTC] [LTC] [LTC] [LTC] [LTC] [LTC] [LTC] [LTC] [TS] [TS] [TS] [TS] CE

- [LTC] 80 bit timestamp ending with the LTC-Sync-Word (0x3FFD)
- [TS] 32 bit internal timestamp (only valid while measurement is runing)

Example Response

```
CE 0F 01 C0 00 60 00 90 40 80 00 3F FD 00 00 00 00 CE
B4 89 01 01 02 00 00 00 00 00 00 3E 5F EE FC BC 6D 47 3B BE 5F E9 3B 3C 72 25 27 BD
59 B1 3C 3B 72 F7 08 BC FC BE 13 3B 11 3A E1 BC 91 96 70 3A AF 84 17 BC 28 A0 19 3A
5B 97 FA BB BC EB C6 3A 0B BC CF BB 3E 8A 67 39 B2 9E 10 BA 65 47 F3 39 58 7A AC 3A
70 42 DA 38 B0 78 15 3B 41 AD F7 B8 60 9B 39 3B BE DE B9 B9 81 62 E4 3C 28 DB 2C BA
0E FF 0C 3C 91 F0 E9 BA 89 B3 39 3C FB D8 88 BA FB 27 EA 3D 58 9A 8E BB 5E CD 71 B4
[following data of the EITFrames]
CE 0F 01 20 00 90 00 90 40 80 00 3F FD 00 00 0B 96 CE
B4 89 01 01 02 00 00 00 00 0B B8 3E 5F AF 05 BC 6D 62 A5 BE 5F A8 93 3C 73 05 DF BD
59 73 C2 3B 75 2C 25 BC FC 75 15 3B 13 31 A6 BC 91 67 B7 3A B2 FD 26 BC 28 68 36 3A
62 08 DD BB BC A7 B5 3A 12 6D CF BB 3E 41 56 39 BF 2E 73 BA 64 C8 4B 39 70 4E 0B 3A
70 38 C0 38 E1 F2 F4 3B 41 74 A8 B7 FD 29 21 3B BE A9 7C B9 6A 0C 6A 3C 28 B3 B2 BA
09 5C 01 3C 91 CA 3B BA 87 10 92 3C FB 94 7E BA F8 7B 69 3D 58 60 DB BB 5D AA 35 B4
[following data of the EITFrames]
CE 0F 01 C0 00 40 80 90 40 80 00 3F FD 00 00 17 1B CE
B4 89 01 01 02 00 00 00 00 17 6F 3E 5F D8 50 BC 6D 98 E8 BE 5F D1 58 3C 73 39 2B BD
59 9A 19 3B 75 6B D4 BC FC A4 08 3B 13 5B 0E BC 91 84 AB 3A B3 5B 33 BC 28 8A 17 3A
62 4F 2C BB BC D6 75 3A 12 69 91 BB 3E 70 DE 39 BF D1 87 BA 65 1D 05 39 70 3E 35 3A
70 27 03 38 DF 79 3D 3B 41 8D 29 B7 FD 2E 71 3B BE C3 F0 B9 6B AF 66 3C 28 CD 5E BA
09 77 21 3C 91 E3 03 BA 87 23 47 3C FB BF F8 BA F8 C1 21 3D 58 87 CD BB 5E 07 07 B4
```

7.5.18 0xCF - TCP connection watchdog

General Syntax

[CT] **05 00 [interval]** [CT]

Return

ACK

Description

The device checks periodically if the connection from the TCP socket to the client is still active. This is done by the device by periodically sending the TCP socket acknowledge command (see chapter "Acknowledge messages"). If the command cannot be send, i.e. because of a lost client connection, the socket server state will be reset to "listening". Thus, a new socket-client connection can be established.

This command provides access to setting the interval for sending the TCP socket acknowledge command.

[interval]

- Set the interval for TCP communication watchdog
- Length: 4 Byte
- Data format: unsigned integer
- Values:
  - Minimum: 1 s
  - Maximum: 600 s
  - Increment: 1 s
  - Default: 60 s

⚠ To save these settings persistently, the "save settings" command must be sent before powering off the device.

7.5.19 0xD1 - Get Device Info

Reads the device specific data.

General Syntax

[CT] **00** [CT]

Return

[CT] **[LE] [serial number]** [CT]

ACK

[LE]

- number of bytes of the serial number (depends on the device configuration)

[serial number]

- Serial number of the device (first 7 bytes are printed on the case of the device)
- The **[serial number]** contains of the following information
  - 1Byte Serial Number version: 0x01
  - 2Byte device identifier: 0x0019 → EIT
  - 2Byte actual unique serial number
  - 2Byte Manufacturing Date (1st Byte years since 2010, 2nd Byte Month of Year)
  - 1Byte Version of the Device Specific Data: 0x03 (current version)
  - 2Bytes Options (hot bit coded):

        - LSBit: Battery Module
        - LSBit+1: WiFi Module
        - LSBit+2: IsoIO Module
    - 2Byte Controller board ID (master)
    - 2Byte Frontend1 board ID
    - 2Byter Connector1 board ID
    - 2Byte Frontend2 board ID
    - 2Byter Connector2 board ID
    - ... additional slave modules starting with Controller board and following the order from above.

7.5.20 0xD2 - Get firmware IDs

Syntax

D2 00 D2

Return

for EIT16 and EIT32:

D2 [length] [2 Byte developer information ARM] **[2 Byte revision number ARM] [2 Byte build number ARM] [5 Byte** developer information FPGA] **[2 Byte revision number FPGA] [2 Byte build number FPGA]** D2

for EIT64:

D2 [length] [2 Byte developer information ARM] **[2 Byte revision number ARM] [2 Byte build number ARM] [5 Byte** developer information FPGA] **[2 Byte revision number FPGA] [2 Byte build number FPGA]** [2 Byte developer information ARM] **[2 Byte revision number ARM] [2 Byte build number ARM]** [5 **Byte** developer information FPGA] **[2 Byte revision number FPGA] [2 Byte build number FPGA]** D2

for EIT128:

D2 [length] [2 Byte developer information ARM] **[2 Byte revision number ARM] [2 Byte build number ARM] [5 Byte** developer information FPGA] **[2 Byte revision number FPGA] [2 Byte build number FPGA]** [2 Byte developer information ARM] **[2 Byte revision number ARM] [2 Byte build number ARM]** [5 **Byte** developer information FPGA] **[2 Byte revision number FPGA] [2 Byte build number FPGA] [2 Byte developer information ARM] [2 Byte revision number ARM] [2 Byte build number ARM] [5 Byte developer information FPGA] [2 Byte revision number FPGA] [2 Byte build number FPGA] [2 Byte developer information ARM] [2 Byte revision number ARM] [2 Byte build number ARM] [5 Byte developer information FPGA] [2 Byte revision number FPGA] [2 Byte build number FPGA]** D2

for all:

ACK

Description

- Reads out version numbers of the firmware.

[length]

- depends on the number of hardware modules

[developer information XX]

- This information is for internal development purposes only.

[revision number XX]

- Length: 2 Byte
- Data format: unsigned integer

[build number XX]

- Length: 2 Byte
- Data format: unsigned integer

7.5.21 0xF1 - WLAN configuration

Configure or read out parameters like DHCP, router SSID, password, server socket port and IP address.

General Syntax

[CT] **[LE] [OB] [CD]** [CT]

Return

[CT] **[LE] [OB] [CD]** [CT] *

ACK

** This frame will only be returned on get commands.*

[LE]

- Represents the byte count of the command frame and varies with the amount [data] bytes. [LE] is always the amount of [data]-bytes + 1.

[OB]

Function	set	get
connect	0x00	0x80
SSID	0x01	0x81
password	0x02	/
DHCP on/off	0x03	0x83
IP	0x04	0x84
socket port	0x05	0x85
auto connect	0x06	0x86
WiFi signal strength	/	0x87

Function	set	get
current signed IP	/	0x88

To save settings persistently, the "save settings" command must be sent before powering off the device!

connect

Connect / disconnect to the WLAN network chosen by the SSID setting or read out the current connection status.

When the set command is received, the device tries to connect to this network using the settings of the "WLAN configuration" command (password, IP, etc.).

Syntax

- Syntax set: **[CT] 02 00 [switch] [CT]**
  - Return: **[CT] 02 80 [switch] [CT]** (can be delayed up to 10 seconds)
- Syntax get: **[CT] 01 80 [CT]**
  - Return: **[CT] 02 80 [switch] [CT]**

[switch]

- Length: 1 Byte
- Values:
  - 0x00 - disconnect
  - 0x01 - connect

SSID

Set or read the SSID of the desired WLAN network.

Syntax

- Syntax set: **[CT] [LE] 01 [SSID] [CT]**
- Syntax get: **[CT] 01 81 [CT]**
  - Return: **[CT] [LE] 81 [SSID] [CT]**

[SSID]

- ASCII coded SSID
- Length: 1 ... 32 byte

password

Set the password for the desired WLAN network.

Syntax

- Syntax set: **[CT] [LE] 02 [password] [CT]**

[password]

- ASCII coded password

- Length: 1 ... 63 byte

DHCP on/off

Activate/deactivate DHCP usage. / Read out the current DHCP setting.

Syntax

- Syntax set: **[CT] 02 03 [switch] [CT]**
- Syntax get: **[CT] 01 83 [CT]**
  - Return: **[CT] 02 83 [switch] [CT]**

[switch]

- Length: 1 byte
- Values:
  - 0x00 - deactivate DHCP
  - 0x01 - activate DHCP

IP

Set or read our IPv4 address of the device, subnet mask or gateway of the network.

Syntax

- Syntax set: **[CT] 06 04 [item] [address] [CT]**
- Syntax get: **[CT] 02 84 [item] [CT]**
  - Return: **[CT] 06 84 [item] [address] [CT]**

[item]

- Select the target item.
- Length: 1 byte
- Values:
  - 0x01 - IP address
  - 0x02 - subnet mask
  - 0x03 - gateway

[address]

- Length: 4 byte
- Data format: byte-wise unsigned integer
- Value range of each byte: 0 ... 255
- All default settings are 255.255.255.255

socket port

Set or read out the port for the socket server.

Syntax

- Syntax set: **[CT] 03 05 [port] [CT]**
- Syntax get: **[CT] 01 85 [CT]**
  - Return: **[CT] 03 85 [port] [CT]**

[port]

- Port number
- Length: 2 byte
- Data format: unsigned integer
- Value range: 0 ... (2^16)-1
- Default: 65535

auto connect

Enable/disable automatic connection after turning on the device. Or read out current setting for automatic connection after turning on the device.

Syntax

- Syntax set: **[CT] 02 06 [switch] [CT]**
- Syntax get: **[CT] 01 86 [CT]**
    - Return: **[CT] 02 86 [switch] [CT]**

[switch]

- Length: 1 byte
- Values:
    - 0x00 - disabled (default)
    - 0x01 - enabled

WiFi signal strength

Read out current signal strength of the WLAN network chosen by "SSID"-setting.

Syntax

- Syntax get: **[CT] 01 87** [CT]
    - Return: **[CT] 02 87 [strength] [CT]**

[strength]

- Current signal strength in percent.
- Length: 1 byte
- Data format: unsigned integer
- Unit: percent
- Value range: 0 ... 100

signed IP

Read out currently assigned IP address.

This address is the actual address of the device in the network. When DHCP is used, it may differ from the static IP setting.

Syntax

- Syntax get: **[CT] 01 88 [CT]**
    - Return: **[CT] 05 88 [address] [CT]**

[address]

- Length: 4 byte
- Data format: byte-wise unsigned integer

7.6 Measured Data

After the start '*Start/Stop Measurement*' command (0xB4) is received, the instrument starts the measurement and transmits data via the COMInterface. Depending on the configured settings the data load can vary. If the resulting data could not be transferred via the selected interface the measurement is paused and a data holdup message is sent. Each EIT-frame consists of a number of excitation settings (which can be configured with the '*Set Measurement Setup*' command (0xB0) and option byte 0x06). Every frequency of the configured sweep is sent in a separate communication frame. For example an excitation sequence with 8 excitation settings and a frequency sweep with 10 frequencies are configured. In this case an EIT frame consists of 80 communication frames.

For better synchronization of communication several optional flags can be enabled. This option can be configured with the "*Set Output Configuration*" command.

CT	**LE**	**CG**	**ES**		**FR**		**TS**				**1st channel**				**…**	**16th channel**				**CT**
B4	XX	XX	ES_{out}	ES_{in}	MSB	LSB	MSB			LSB	MSB			LSB		MSB			LSB	B4

CT	Command Tag
LE	Length Number of data bytes in this frame = 128 (+ additional options)
CG	Channel Group CG ≥ 1 There is only a maximum of 16 channels in a measurement frame. To support more channels CG can be > 1.
ES	Excitation Setting (optional) ES_{out}: Excitation output ES_{in}: Excitation input In case of EIT256 ES_{out} and ES_{in} are two Byte each.
FR	Number of Frequency (optional) Corresponding frequency number in measurement sequence
TS	Timestamp (optional) time since start of the measurement in ms
n^{th} channel	complex voltage value in volt. Each value consists of real and imaginary part of the complex value

Example:

This example shows two frames generated by the instrument. In this case the output configuration is setup to send the current excitation setting, the current frequency row and the timestamp for each dataset is activated:

B4 89 01 01 02 00 01 00 00 01 23 3F 80 00 00 40 00 00 00 [values 2 .. 15] 40 40 00 00 40 80 00 00 B4

Channel Group: CG = 1 → Channels 1 to 16
Excitation Setting: ESout = electrode 1; ESin = electrode 2
Frequency Row: FR = 1
Timestamp: TS = 291 ms
Ch1: real = 1; imag = 2
Ch16: real = 3; imag = 4

B4 89 02 01 02 00 01 00 00 01 23 40 a0 00 00 40 c0 00 00 [values 18 .. 31] 40 e0 00 00 41 00 00 00 B4

Channel Group: CG = 2 → Channels 17 to 32
Excitation Setting: ESout = electrode 1; ESin = electrode 2
Frequency Row: FR = 1
Timestamp: TS = 291 ms
Ch17: real = 5; imag = 6
Ch32: real = 7; imag = 8

8 IsoIOport (optional)

8.1 Functional Description

The *IsoIOport* is Sciospecs interface standard for connecting peripheral equipment while keeping it isolated from the power line with electrical ratings adequate for medical safety requirements. The standard offers

- Pins:
  - 8 isolated output pins
  - 8 isolated input pins
  - 2x isolated I2C on 4 additional pins (1x independent; 1x shared functionality with two of the outputs)
  - isolated 3.3V output rail
- Functionality:
  - Input and output ports.
  - 1x SPI Master on Bank A and B, assignable to any input/output
  - 1x asynchronous serial interface (UART) on Bank A and B, assignable to any input/output
  - 1x I^2C interface on Bank B and C, assignable to dedicated I^2C-ports
  - synchronization input port(s) and synchronization output port(s)

⚠ Due to the directional nature of the isolators some additional restrictions have to be respected when assigning IO functionality and when assigning serial interface line to a particular input or output.

- Input functions such as 'SPI SCLK' or 'UART Rx' may only be asserted to input ports.
- Output functions such as 'SPI MOSI' or 'UART Tx' may only be asserted to output ports.
- The I^2C function may only be assigned to the dedicated I^2C-ports.

The *Sciospec IsoIOport* standard provides two IO banks with each 8 pins, which can be nearly freely configured to serve as a number of different interface types. These particular interface functionalities are described below.

8.1.1 Interfaces

Digital Inputs

- This is the default configuration for all input ports.
- A weak pull up resistor (10kΩ) to VDD is connected to each I/O.
- The logic stack of all I/Os can be read out by command ("get IO state register"). The command provides a bit vector, where every bit represents the actual logic state of one particular I/O, even if it is configured as "digital output".

Digital Outputs

- This is the default configuration for all output ports.
- The logic state of each pin can be set independently, using the "set output register" command.

- Default state of this register ist 0 for every I/O.

SPI Master

- Type: 4-wire (MOSI as well as MISO can be left unused).
- The polarity of the SPI (CPOL) is configurable by command ("set SPI polarity").
- CPHA = 0
- The SCLK-frequency is configurable within the range
  (mSPI_f_SCLK_source ÷ 2) ≥ f_SCLK ≥ mSPI_f_SCLK_min
- Word length = 8 Bit
- Send and receive a maximum of "mSPI_max_byte" Byte in a single communication process framed by #CS = '0'.
- Data, read from MOSI, always will be transmitted to the master interface where the command came from, which initialized the communication process.
- Application dependend parameters:
  - mSPI_f_SCLK_source
  - mSPI_f_SCLK_min
  - mSPI_max_byte

UART

- Parameters:
  - Baud is rate configurable from BaudrateMin to BaudrateMax (Baudrate = BaudrateMax/ BaudrateDivider).
  - Parity configurable (odd, even, no parity)
  - Startbit = 1
  - Stopbit = 1
  - Idle state = high
  - Receive timeout always is 10 word durations.
  - Send and receive a maximum of 128 Bytes in one continuous transmission.
- Fully asynchronous reception
- Configurable output interface for received data (any of the master interfaces)

I^2C

- SCK and SCL line open drain
- Parameters:
  - fSCK = 100kBaud
  - Send and Receive a maximum 15 Bytes

Synchronization Signals

For synchronization of the measurement process with other customer hardware the Sciospec device can be equipped with the Synchronization Signals on the *IsoIOport*. It offers one “Sync Out” and one “Sync In” signals located at the *IsoIOport* of the instrument.

The “Sync Out” port gives information about the current state of the measurement.

- Logic high → Measurement is running

- Logic low → Measurement is paused between two measurement sweeps.

The “Sync In” port allows to control the measurement process

- By default the signal is logic high, which allows for a continuous measurement.
- By connecting this port to ground (logic low) the next measurement will be paused until the signal is released again.
- An already running sweep will not be interrupted by the “Sync In” signal.

⚠ Do not apply any voltage source directly to the synchronization ports!

Technical Specifications

isolated inputs		
high level input voltage	0.7 × Vsupply	
	2.31 V (min)	@ 3.3 V supply
low level input voltage	0.3 × Vsupply	
	0.66 V (max)	@ 3.3 V supply
input current (max)	±10 µA (max)	
absolute max input voltage	Vsupply + 0.5 V	
data rate	0...100 Mbps	
pull-up resistors	10 kΩ (default)	to Vsupply
Isolation ratings	5kVrms, 12.8 kV surge, ±100 kV/µs Typical CMTI reinforced insulation according to 60950-1 (800 Vrms) and 60601-1 (2 MOPP @ 250Vrms)	
ESD protection	±8 kV contact discharge protection per IEC 61000-4-2 ±6 kV human body model (HBM), per ANSI/ESDA/JEDEC JS-001 ±1.5 kV charged device model (CDM) per JESD22-C101	
isolated outputs		
high level output voltage	3V (min)	@ 3.3V supply, Iout ≤ 2 mA
	4.6V (min)	@ 5V supply, Iout ≤ 4 mA
low level output voltage	0.3 V (max)	@ 3.3V supply, Iout ≤ 2 mA
	0.4V (max)	@ 5V supply, Iout ≤ 4 mA
output current capability	±2 mA (min)	@ 3.3V supply

	±4 mA (min)	@ 5V supply
isolation ratings	5kVrms, 12.8 kV surge, ±100 kV/µs Typical CMTI reinforced insulation according to 60950-1 (800 Vrms) and 60601-1 (2 MOPP @ 250Vrms)	
ESD protection	±8 kV contact discharge protection per IEC 61000-4-2 ±6 kV human body model (HBM), per ANSI/ESDA/JEDEC JS-001 ±1.5 kV charged device model (CDM) per JESD22-C101	
isolated bidirectional open drain I^2C ports		
high level input voltage	0.7 x Vsupply (min)	
low level input voltage	0.3 x Vsupply (max)	
low level output voltage	900 mV (max)	
current sink capability	30 mA (max)	
pull-up resistors	10 kΩ (default)	to Vsupply
operating frequency	400 kbps (max)	
Isolation ratings	5kVrms reinforced insulation according to 60950-1 (380 Vrms) and 60601-1 (2 MOPP @ 250Vrms)	
ESD protection	±12 kV contact discharge protection per IEC 61000-4-2 ±15 kV air gap discharge protection per IEC 61000-4-2	
isolated supply		
voltage	3.3 V (typ)	± 180 mV @ 0...10 mA
output current capability	60 mA (max)	
line regulation	2 mV/V (typ)	
load regulation	5% (max)	@ 10...54 mA
output ripple	50 mVpp	20 MHz BW, Cload = 0.1 µF||10 µF, Iout = 54 mA
output noise	130 mVpp	Cload = 0.1 µF||10 µF, Iout = 54 mA
Isolation ratings	5kVrms reinforced insulation according to 60950-1 (415 Vrms), 60601-1 (2 MOPP @ 250Vrms) and 61010-1 (300Vrms mains, 415 V secondary)	
ESD protection	±15 kV air gap discharge protection per IEC 61000-4-2	

8.1.2 Pinout and *InterfacePort* mapping

Used connector type

10240-1210PE, 40 pin .050" Mini D Ribbon (MDR) receptible, SMT right angle, shielded

Pin	Name	*InterfacePort* mapping	Default Function	*Important information*
1	GNDiso			
2	GNDiso			
3	reserved			
4	reserved			
5	reserved			
6	reserved			
7	PowerPlugDetect			High if DC Power Plug is connected Low if DC Power Plug is disconnected Status can also be read by using the Sciospec COMInterface 0xCC
8	I^2C SCL, bank C	Bank C, IO 1	I^2C SCL	Preconfigured *InterfacePort* settings. No configuration necessary and possible!
9	I^2C SDA, bank C	Bank C, IO 2	I^2C SDA	
10	in 8	Bank B, IO 8	input	
11	in 7	Bank B, IO 7	input	
12	in 6	Bank B, IO 6	input	
13	in 5	Bank B, IO 5	input	
14	in 4	Bank A, IO 8	input	
15	in 3	Bank A, IO 7	input	
16	in 2	Bank A, IO 6	input	
17	reserved			
18	reserved			

Pin	Name	*InterfacePort* mapping	Default Function	*Important information*
19	GNDiso			
20	GNDiso			
21	GNDiso			
22	GNDiso			
23	VDDiso			
24	GNDiso			
25	VDDiso			
26	I^2C SCL	Bank B, IO 4	output	Shares functionality with "out 8". This pin may be used as I^2C interface.
27	I^2C SDA	Bank B, IO 3	output	Shares functionality with "out 7". This pin may be used as I^2C interface.
28	n.c.			
29	in 1	Bank A, IO 5	input	
30	out 8	Bank B, IO 4	output	Shares functionality with "I^2C SCL". This pin may **not** be used as I^2C interface.
31	out 7	Bank B, IO 3	output	Shares functionality with "I^2C SDA". This pin may **not** be used as I^2C interface.
32	out 6	Bank B, IO 2	output	
33	out 5	Bank B, IO 1	output	
34	out 4	Bank A, IO 4	output	
35	out 3	Bank A, IO 3	output	
36	out 2	Bank A, IO 2	output	
37	out 1	Bank A, IO 1	output	
38	reserved			
39	GNDiso			
40	GNDiso			

8.2 Sciospec COMInterface

8.2.1 0xCA - Configure IsoIOport

General Syntax

[CT] [LE] **[OB] [bank] [command]** [CT]

Return

[CT] [LE] **[OB] [bank] [command]** [CT]*

ACK

** This frame will only be returned (1) on get commands, (2) on a read operation on synchronous interfaces or (3) if data has been received on asynchronous interfaces.*

Due to the directional nature of the isolators some additional restrictions have to be respected when assigning IO functionality and when assigning serial interface line to a particular input or output.

- Input functions such as 'SPI SCLK' or 'UART Rx' may only be asserted to input ports.
- Output functions such as 'SPI MOSI' or 'UART Tx' may only be asserted to output ports.
- The I^2C function may only be assigned to the dedicated I^2C-ports.

[LE]

- Represents the byte count of the command frame and varies with the amount [data] bytes.

[OB]

Function	set	get
IO functionality	0x00	0x80
UART Rx data sink	0x01	0x81
interface data	0x02	0x82
interface settings	0x03	0x83

[bank]

- Select the target bank of a command. In a returned frame this element indicates the source bank.
- Length: 1 byte
- Data format: unsigned integer
- Values:
  - 0x01 ... Bank A
  - 0x02 ... Bank B
  - 0x03 ... Bank C

IO functionality

- Defines the functionality of a certain Input or Output.
- Syntax set: **[CT] 04 00 [bank] [IO#] [function] [CT]**
- Syntax get: **[CT] 03 80 [bank] [IO#] [CT]**
  - Return get: **[CT] 04 80 [bank] [IO#] [function] [CT]**
- [IO#]:
  - Defines the target Input or Output of the command.
  - Values:
    - 0x01 ... 0x08 for IO 1 to 8
- [function]:
  - Defines the functionality that will be assigned to the Input or Output.
  - Values:
    - 0x00 - Input
    - 0x01 - Output
    - 0x02 - I^2C SCL (only for Bank B, IO 4 and IO 3)
    - 0x03 - I^2C SDA (only for Bank B, IO 4 and IO 3)
    - 0x04 - SPI Master SCLK
    - 0x05 - SPI Master CS#
    - 0x06 - SPI Master MOSI
    - 0x07 - SPI Master MISO
    - 0x08 - UART Tx
    - 0x09 - UART Rx
    - 0x0A - Sync Out
    - 0x0B - Sync In

UART Rx data sink

- Defines output interface for received data of the asynchronous peripheral interfaces.
- Syntax set: **[CT] 04 01 [bank] [interface] [sink] [CT]**
- Syntax get: **[CT] 03 81 [bank] [interface] [CT]**
  - Return get: **[CT] 04 81 [bank] [interface] [sink] [CT]**
- [interface]:
  - Names the source interface of data which a certain sink should be assigned to.
  - Values:
    - 0x04 - UART Rx
- [sink]:
  - Values:
    - 0x00 - High speed USB
    - 0x01 - Full speed USB (default)

interface data

- Commit/Receive data to a certain peripheral interface.
- Syntax set: **[CT] [LE] 02 [bank] [interface] [data] [CT]**
- Syntax get: **[CT] 03 82 [bank] [interface] [CT]** (only defined for [interface] = "Input register" and "Output register")
- Return: **[CT] [LE] 82 [bank] [interface] [data] [CT]**
- [interface]:

- Defines the target interface of the command. The content of [data] will be committed to this interface.
- Values:
  - 0x00 - IO state register
  - 0x01 - Output register
  - 0x02 - I^2C
  - 0x03 - SPI Master
  - 0x04 - UART

- [data]:
  - Carries the data for/from the target interface. Length and structure of this element depends on the interface and the amount of transmitted data.
  - IO state register
    - Used only within "get" command. Returns NACK when used with "set" command.
    - [data] contains one byte where every bit represents the actual logic state of one certain Input or Output. 1st bit = IO1, 2nd bit = IO2 and so on.
    - Bit order: most significant bit first
    - [data] is part of the return frame which will be sent on the "get IO state register" command.
  - Output register
    - [data] contains one byte where every bit represents the logic state which will be driven to the output pin. 1st bit = IO1, 2nd bit = IO2 and so on.
    - Bit order: most significant bit first
    - [data] is both part of the "set output register" command and the return on the "get output register" command.
  - I^2C
    - Used only within "set" command. Returns NACK when used with "get" command.
    - Write process: [data] = **[I^2C address + R/W bit] [txDataCount] [txData]**
      - [I^2C address + R/W bit]
        - I^2C address:
          - Address of the I^2C device
          - length: 7 Bit
          - data format: unsigned integer
        - R/W bit
          - Indicates the communication direction.
          - must be '0' (= Write)
      - [txDataCount]
        - Number of bytes to be sent.
        - length: 1 Byte
        - data format: unsigned integer
        - value margin: 0 ... 15
      - [txData]
        - Actual data to be sent.
        - length: Must match [txDataCount].
    - Read process: [data] = **[I^2C address + R/W bit] [txDataCount] [txData] [rxDataCount]**
      - [I^2C address + R/W bit]
        - I^2C address:
          - Address of the I^2C device

- length: 7 Bit
- data format: unsigned integer

- R/W bit
  - Indicates the communication direction.
  - '0' = Write
  - '1' = Read

- [txDataCount]
  - Number of bytes to be sent before read operation is executed.
  - length: 1 Byte
  - data format: unsigned integer
  - value margin: 0 ... 15
  - ℹ️ If [txDataCount] is ≠ 0, then a sequential read process is executed. First [txData] is sent to the device. The stop condition of this write process is omitted. Then [rxDataCount] is read from the device.
- [txData]
  - Actual data to be sent.
  - length: Must match [txDataCount].
- [rxDataCount]
  - Number of bytes to be received.
  - length: 1 Byte
  - data format: unsigned integer
  - value margin: 0 ... 15

▪ Return: [data] contains only the data read from the I^2C slave. A return frame will only be sent after a successful read process.

◦ SPI Master

▪ Used only within "set" command. Returns NACK when used with "get" command.
▪ ⚠️ Read process: Besides the data to be written to the MOSI line the [data] part of the set command contains an additional byte before this write data, which carries the information how many bytes shall be read after finishing the transmission of the data to be written. Data format of this byte is unsigned integer. (See example below)
▪ In a return, which will be sent on a set command where the number of bytes to read ≠ 0, [data] only contains the read data bytes.
▪ Constraint: [data] can contain a maximum of 200 bytes.

◦ UART

▪ Used only within "set" command. Returns NACK when used with "get" command.
▪ Only contains transmit or received data without any additional information. A frame with received data will be returned everytime the device received data on the UART Rx line. As the interface is of asynchronous nature, this can occur anytime and without being initialized by user command.
▪ Constraint: [data] can contain a maximum of 200 bytes

interface settings

- Set miscellaneous settings such as SPI SCLK-rate.
- Syntax set: **[CT] [LE] 03 [bank] [interface] [setting] [value] [CT]**
- Syntax get: **[CT] [LE] 83 [bank] [interface] [setting] [CT]**
- Syntax return: **[CT] [LE] 83 [bank] [interface] [setting] [value] [CT]**

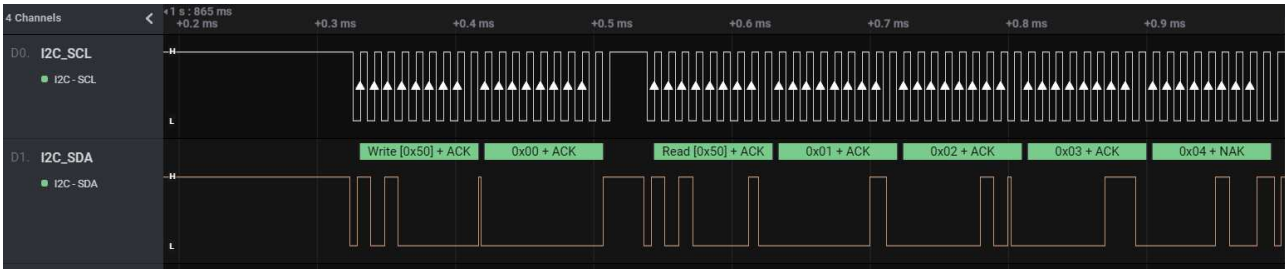
- [interface]:
  - Defines the target interface of the command. The setting will be applied to this interface.
  - Values:
    - 0x03 - SPI Master
    - 0x04 - UART
- [setting]:
  - Defines the setting to be applied.
  - Possible values depend on the [interface]:
    - "SPI Master"
      - 0x00 - SPI SCLK-rate
      - 0x01 - SPI polarity
    - "UART"
      - 0x00 - UART baud rate
      - 0x01 - UART parity
- [value]:
  - Actual set value.
  - Length and possible values of this element depend on the chosen [setting]:
    - "SPI SCLK-rate"
      - Defines the SCLK frequency (f_SCLK).
      - 32 bit, unsigned integer
      - f_SCLK = mSPI_f_SCLK_source ÷ [value]
      - mSPI_f_SCLK_source = 115MHz
      - f_SCLK-range: (mSPI_f_SCLK_source ÷ 2) ≥ f_SCLK ≥ 1kHz
      - Default: 1MHz
    - "SPI polarity"
      - Defines the SPI polarity, thus the idle state of the SCLK signal.
      - values:
        - 0x00 - CPOL = 0 (idle low) (default)
        - 0x01 - CPOL = 1 (idle high)
    - "UART baud rate"
      - Defines the baud rate of the UART interface.
      - 32 bit, unsigned integer
      - UartBaudRate = UartBaudrate_source ÷ [value]
      - UartBaudRate_source = 115MHz
      - UartBaudRate-range: 3 MBaud ≥ UartBaudRate ≥ 9600 Baud
      - Default: 9600Baud
    - "UART parity"
      - Defines the use and polarity of the parity bit.
      - values:
        - 0x00 - no parity (default)
        - 0x01 - even parity
        - 0x02 - odd parity

Examples

I^2C - Read / Write process

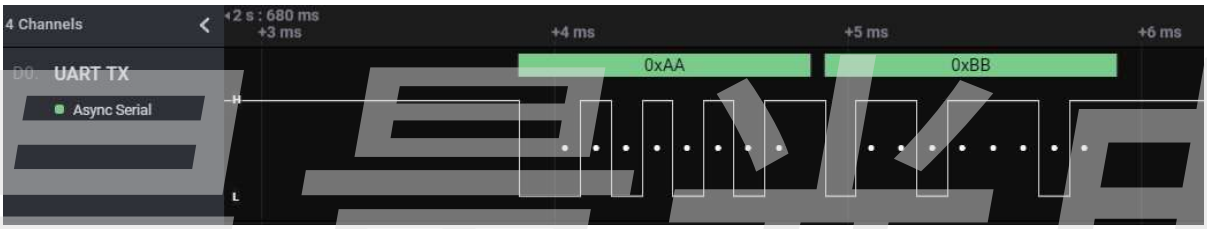
- Set command interface data: **CA 07 02 03 02 A1 01 00 04 CA** → write 00 and read 4 byte of I^2C Bank 3

- Set command IO functionality (set Bank B, IO 3 to I^2C SDA): **CA 04 00 02 03 03 CA**
- Set command IO functionality (set Bank B, IO 4 to I^2C SCL): **CA 04 00 02 04 02 CA**
- Set command interface data: **CA 07 02 02 02 A1 01 00 04 CA** → write 00 and read 4 byte of I^2C Bank 2
  - Return: **CA 07 82 02 02 01 02 03 04 CA**
- Set command interface data: **CA 0A 02 02 02 A0 05 00 01 02 03 04 CA** → write 5 byte to I^2C Bank



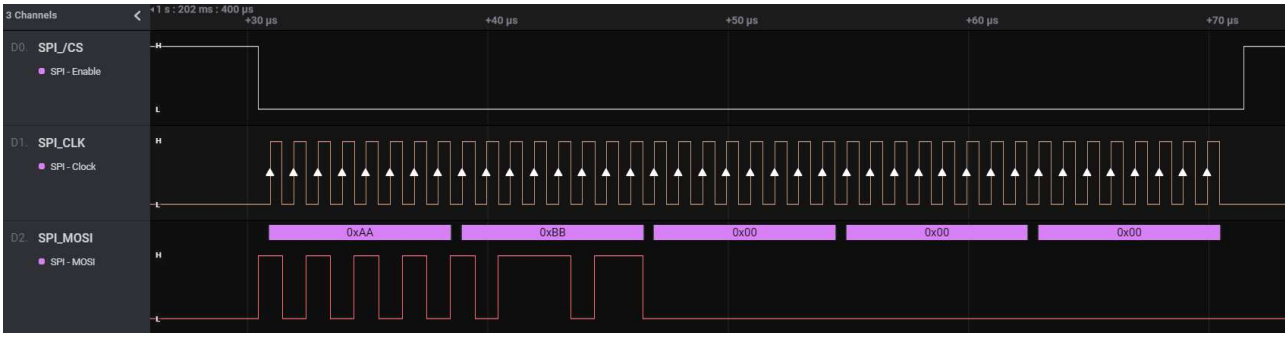
UART

- Set command: **CA 04 01 01 04 00 CA** → set UART Rx data sink to HS USB
- Set command: **CA 04 00 01 01 08 CA** → set IO 1 Bank A to UART Tx
- Set command: **CA 04 00 01 06 09 CA** → set IO 6 Bank A to UART Rx
- Set command: **CA 05 02 01 04 AA BB CA** → send AA BB on UART Bank1
  - Return: **CA 05 82 01 04 AA BB CA**



SPI

- Get command: **CA 04 83 01 03 00 CA** Read SPI SCLK-rate.
  - Return: **CA 08 83 01 03 00 00 00 00 73 CA** → 1MHz (because f_SCLK = mSPI_f_SCLK_source ÷ [value], mSPI_f_SCLK_source = 115MHz)
- Get command: **CA 04 83 01 03 01 CA** Read SPI polarity.
  - Return: **CA 05 83 01 03 01 00 CA** → CPOL = 0 (idle low)
- Set command: **CA 04 00 01 01 05 CA** → set IO 1 Bank A to SPI Master CS#
- Set command: **CA 04 00 01 02 04 CA** → set IO 2 Bank A to SPI Master SCLK
- Set command: **CA 04 00 01 03 06 CA** → set IO 3 Bank A to SPI Master MOSI
- Set command: **CA 04 00 01 05 07 CA** → set IO 5 Bank A to SPI Master MISO
- Set command: **CA 07 02 01 03 02 03 AA BB CA** → send AA BB to SPI Master Bank A and reads 3 Bytes



9 Appendix

9.1 Data formats

All data sent to the device or sent from the device is either “Integer-Straight-Binary” or 32 Bit Float coded (according to IEEE 745). Either way the data is in Big-Endian format.

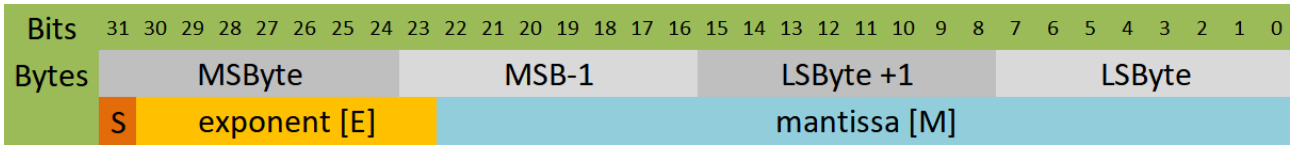
Big-Endian means that if a value consists of more than one byte the byte order is according to their value:

Example 4 byte Big-Endian Value: [MSByte] [MSByte-1] [LSByte-1] [LSByte]

The bit order is within one byte MSBit-first:

[MSBit] [MSBit-1] [MSBit-2] [MSBit-3] [LSBit+3][LSBit+2] [LSBit+1] [LSBit]

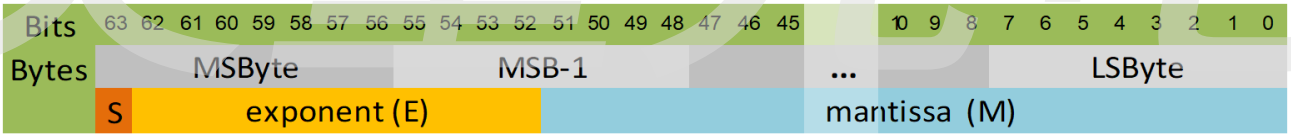
9.1.1 Float data format - single precision



The float value is calculated as follows:

$$float = (-1)^S * (1 + M * 2^{-23}) * 2^{E-127}$$

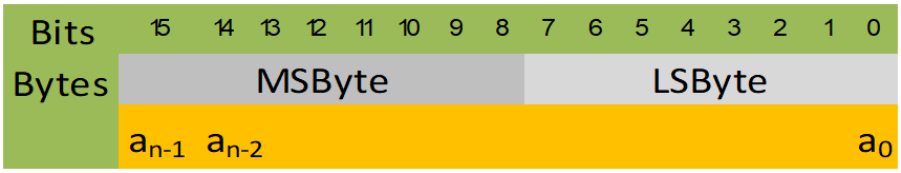
9.1.2 Float data format - double precision



The double precision value is calculated as follows:

$$float = (-1)^S * (1 + M * 2^{-52}) * 2^{E-1023}$$

9.1.3 Integer straight binary data format



The numerical value is calculated as follows:

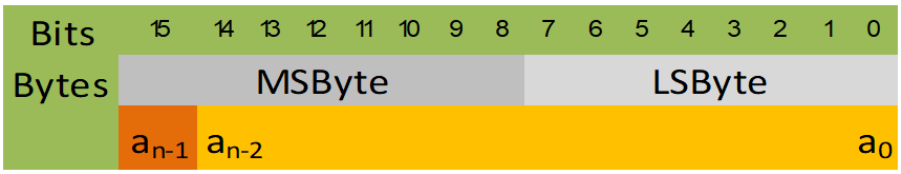
$$x_{ISB} = 2^{n-1} * a_{n-1} + 2^{n-2} * a_{n-2} + \cdots + 2^1 * a_1 + 2^0 * a_0$$

with

- n - bit width (in this example n=16)
- a_i - bit numbering i (a_{n-1}...most significant bit; a_0...least significant bit)

9.1.4 Two's complement data format

Example: 16bit two's complement



The numerical value is calculated as follows:

$$x_{twos} = -2^{n-1} * a_{n-1} + 2^{n-2} * a_{n-2} + \cdots + 2^1 * a_1 + 2^0 * a_0$$

with

- n - bit width (in this example n=16)
- a_i - bit numbering i (a_{n-1}...most significant bit; a_0...least significant bit)

10 EIT System Accessories

10.1 Standard EIT phantom tanks

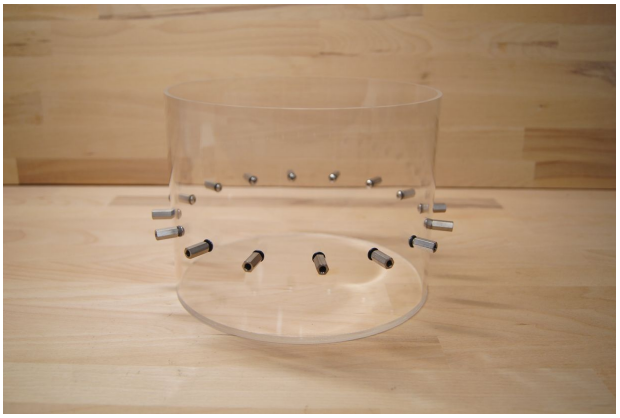


Figure 15: Standard EIT16 phantom tank

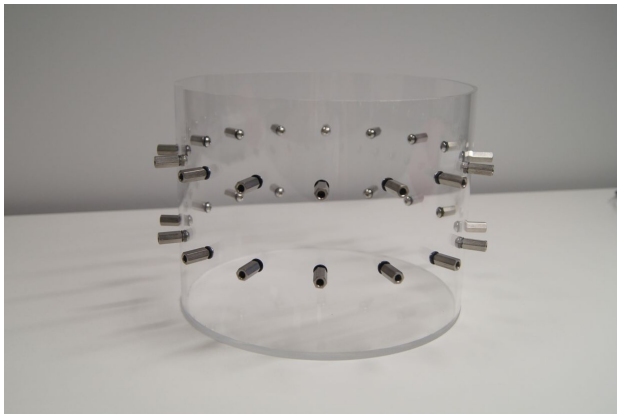


Figure 16: Standard EIT phantom tank 16x2

Figure 17: Standard EIT16 phantom tank with standard measurement cables

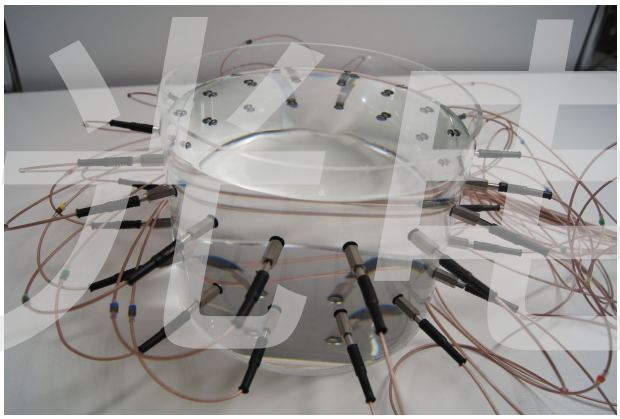


Figure 18: Standard EIT phantom tank 16x2 with measurement cables

	EIT16 phantom tank	EIT phantom tank 16x2
diameter (external)	20 cm	20 cm
height (external)	15 cm	15 cm
material	acrylic glass	acrylic glass

electrodes	16 electrodes on one level, evenly distributed stainless steel A2	two rows, each 16 electrodes on one level, evenly distributed stainless steel A2
banana jack material	stainless steel DIN EN 10088-3 EN steel name: 1.4305	stainless steel DIN EN 10088-3 EN steel name: 1.4305

ⓘ Other standard EIT tanks:

- EIT32 phantom tank (one rings with 32 electrodes)
- EIT phantom tank 32x2 (two rings with 32 electrodes each)

Custom EIT phantom tanks with alternative electrode configurations (number of electrodes, multiple rows) and diameters/heights are available on request.

10.2 EIT cable sets

10.2.1 Standard EIT measurement cable set

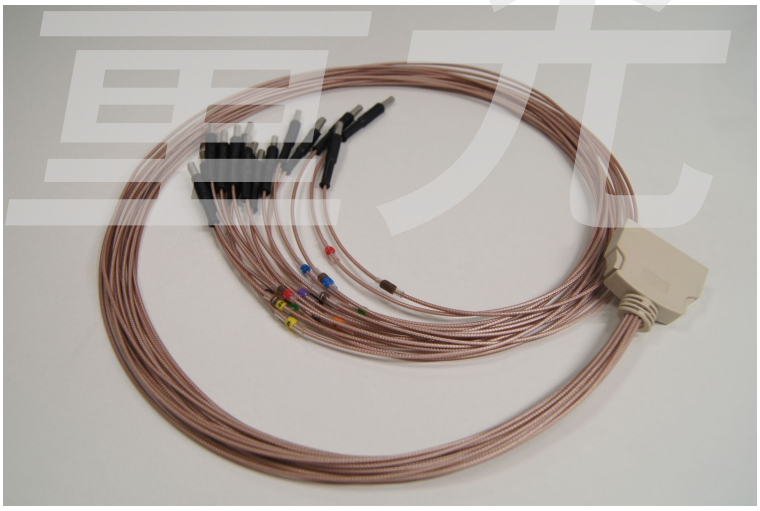
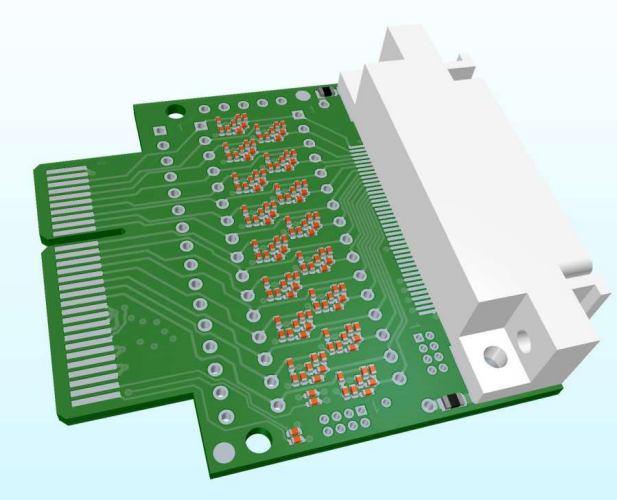


Figure 19: Standard EIT measurement cable set

connector type (device)	MDR40
connector type (DUT)	Banana plug
Standard length	1 m

ⓘ Custom cabling sets (length, connector type..) are available on request.

10.3 EITSC58conAdapter

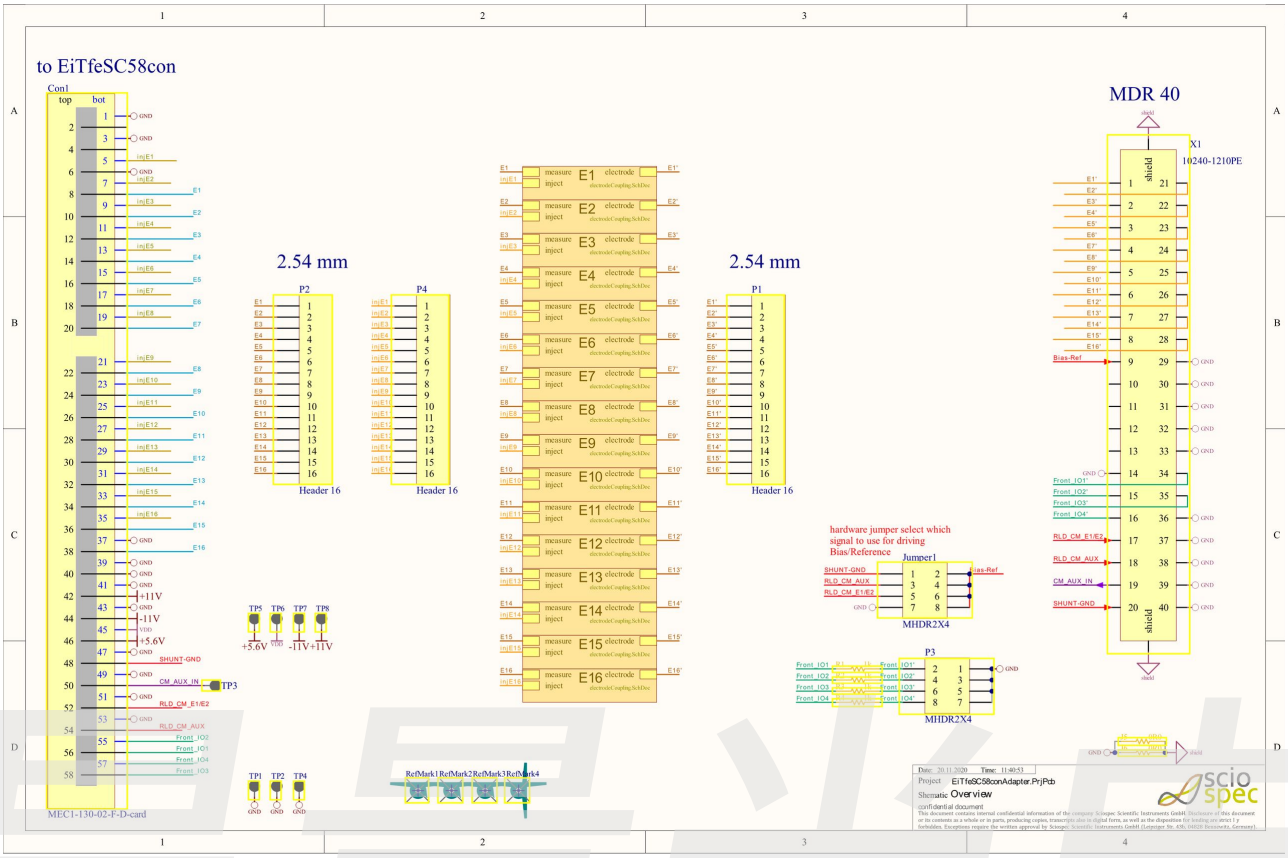


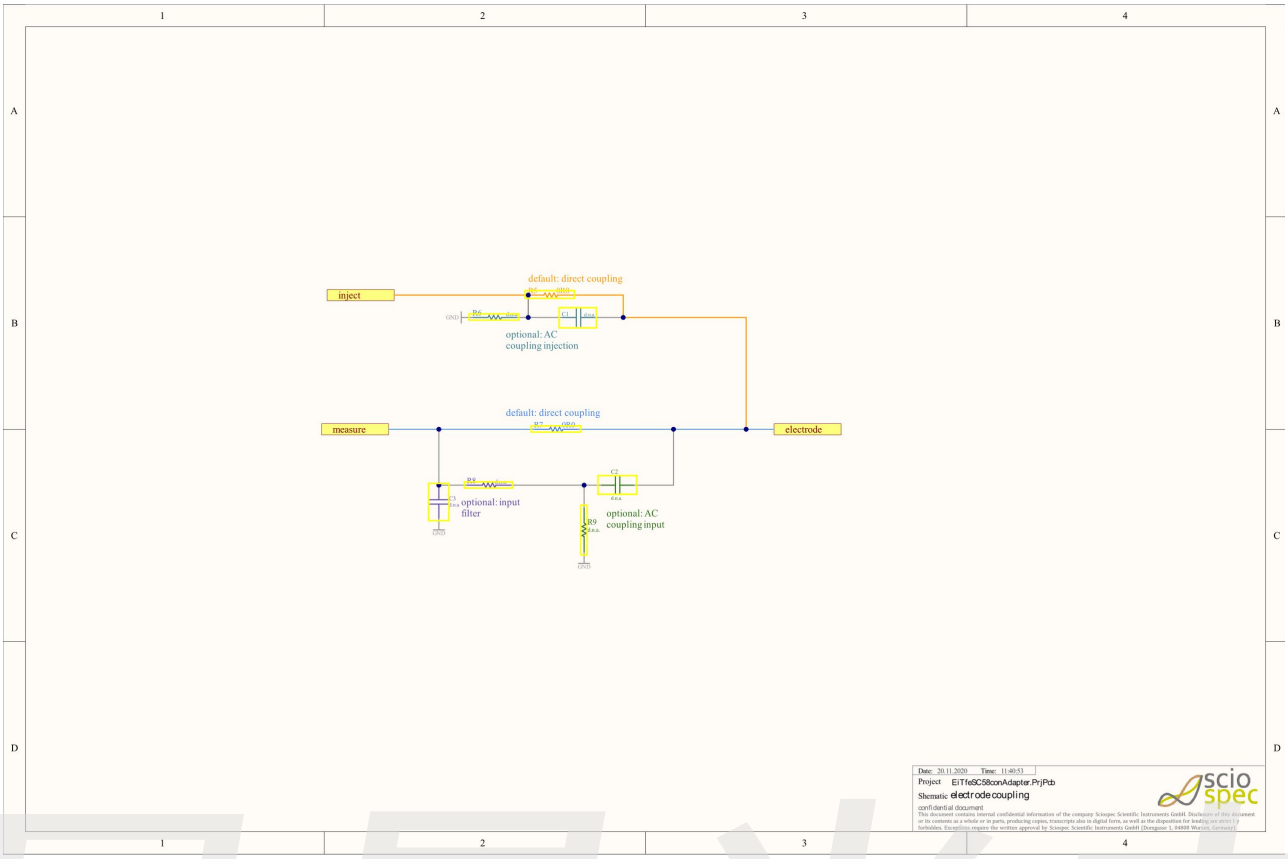
10.3.1 Description

The EITSC58conAdapter is intended to be used on a EIT System (16, 32, 64 or 128 channel) equipped with the MEC1 connector. It enables to use standard Sciospec MDR40 EIT cables for EIT measurements.

It is not recommended to be used for any other purpose or device.

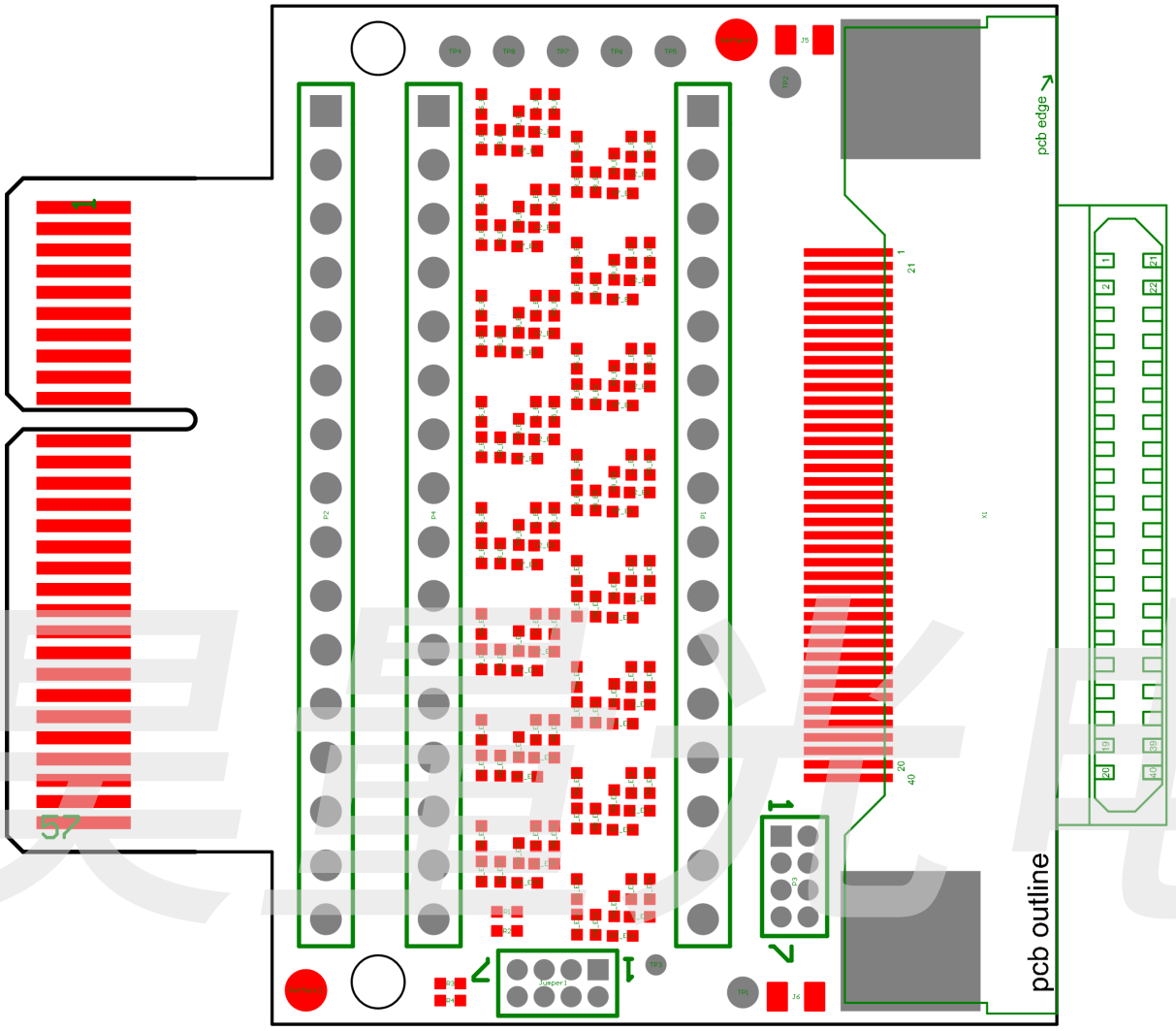
10.3.2 Schematic



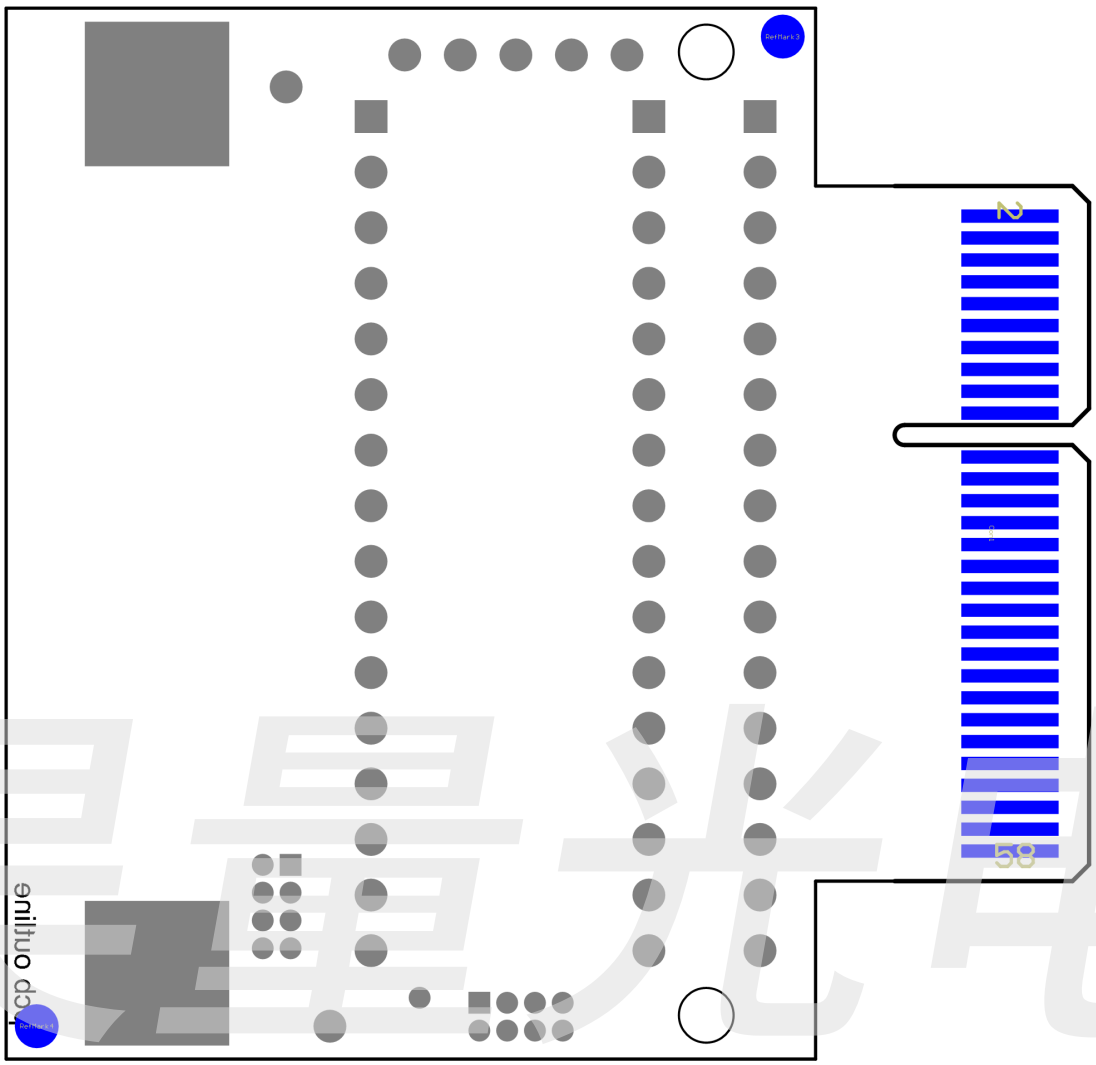


10.3.3 Assembly

Top

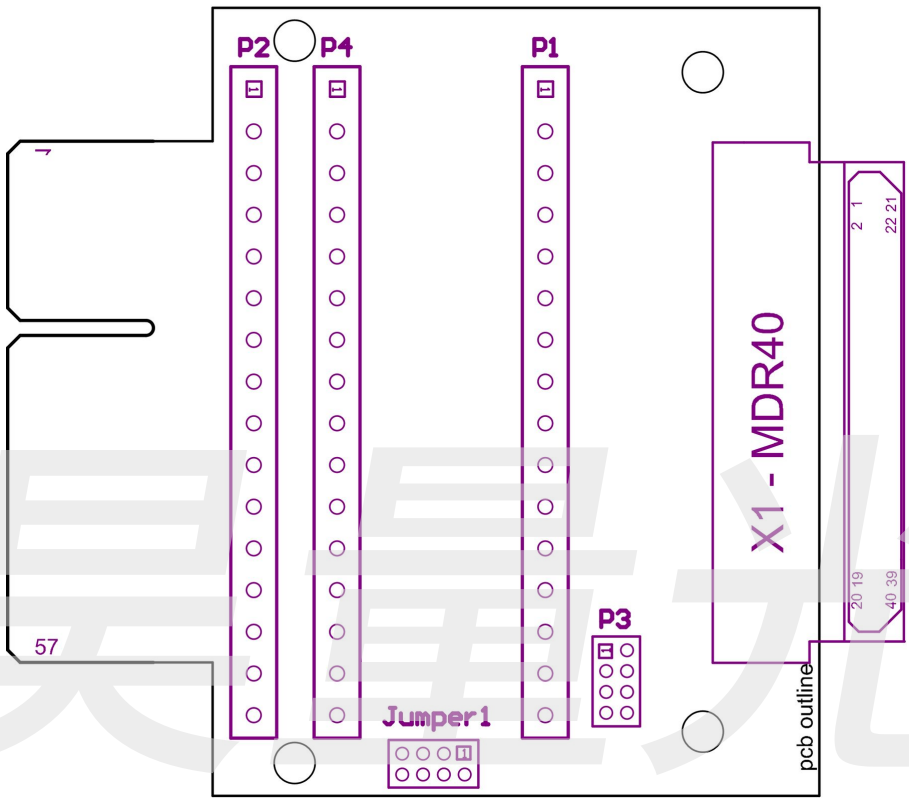


Bottom

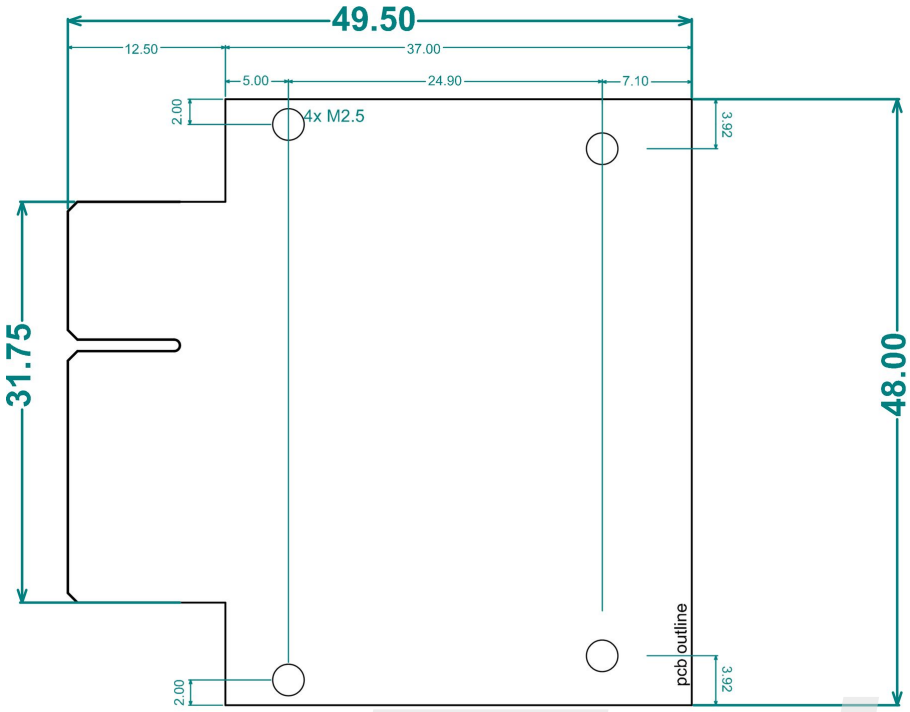


10.3.4 Connectors

Top Connectors

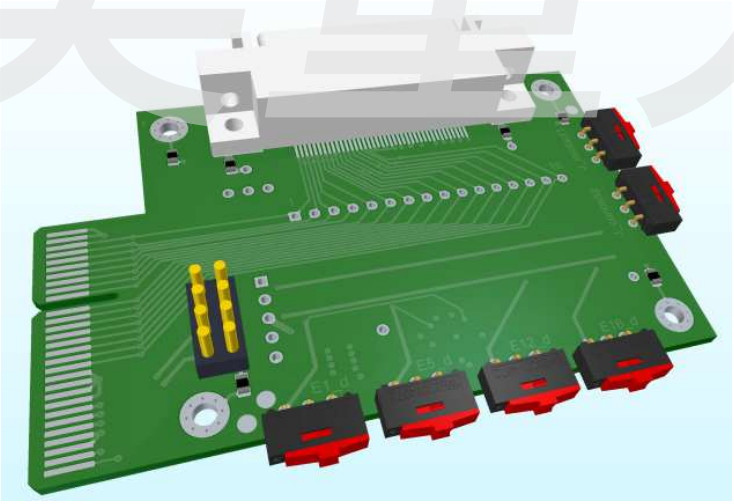


10.3.5 Mechanical



dimensions in mm

10.4 EiTphantom2011

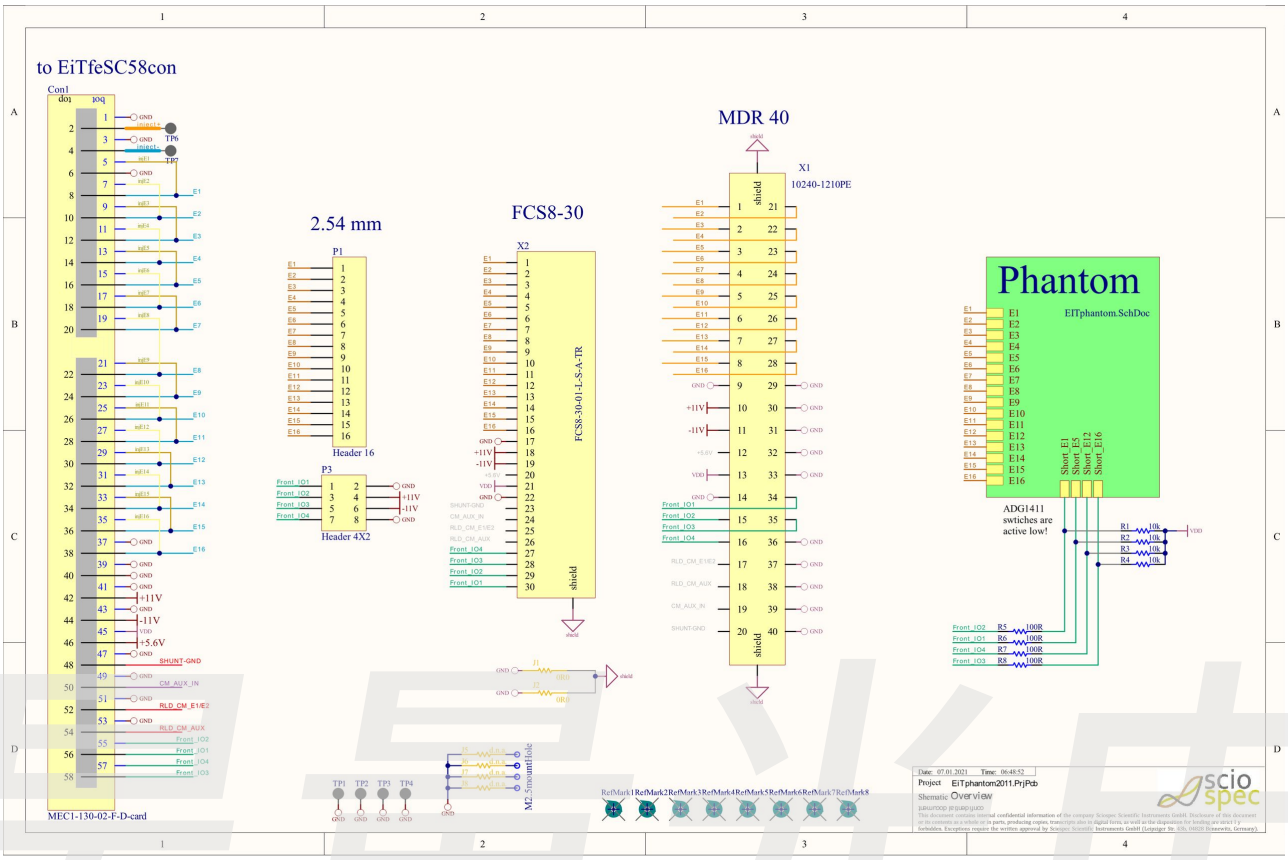


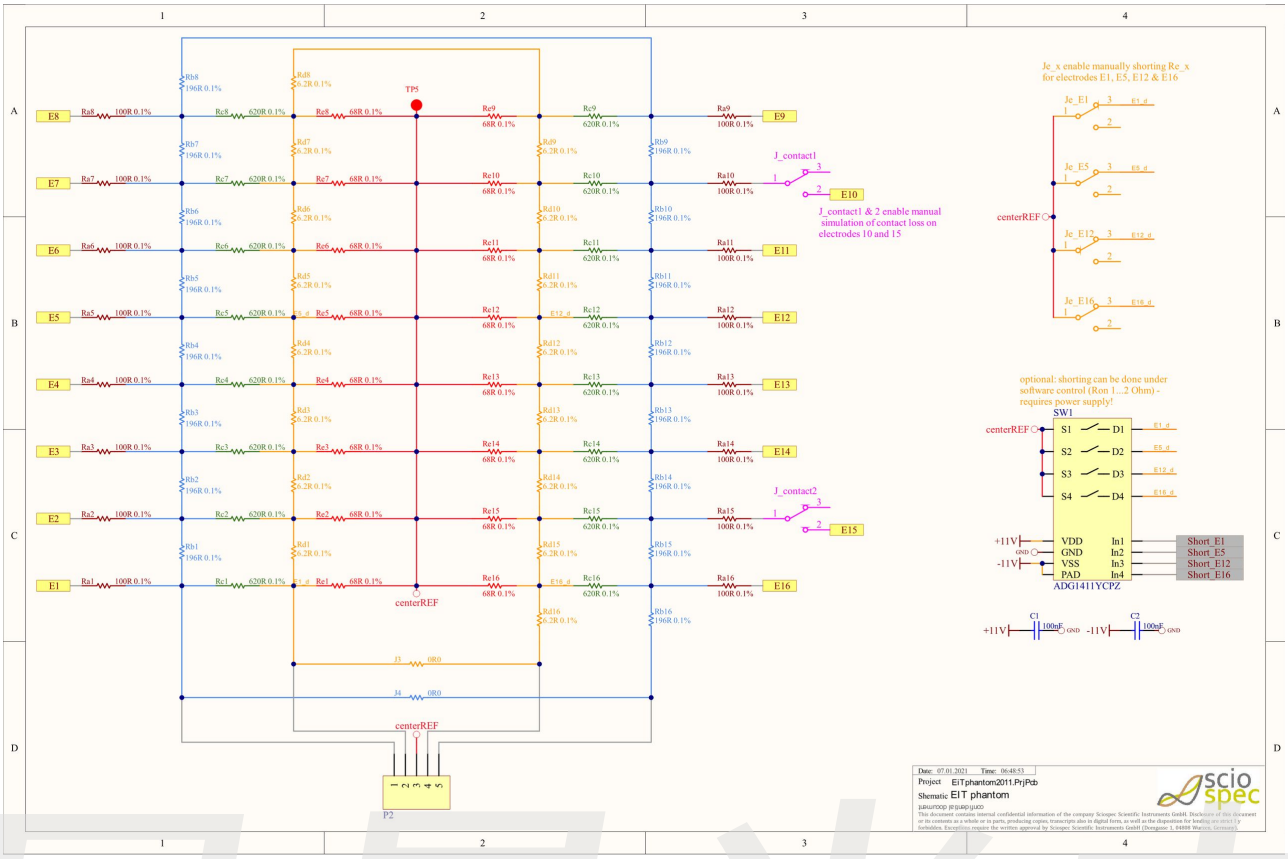
10.4.1 Description

The EiTphantom2011 is intended to be used with the EIT System (16, 32, 64 or 128 channel). It allows for an easy way to get example and reference data sets to test the device and image reconstruction algorithms.

It is not recommended to be used with any other device.

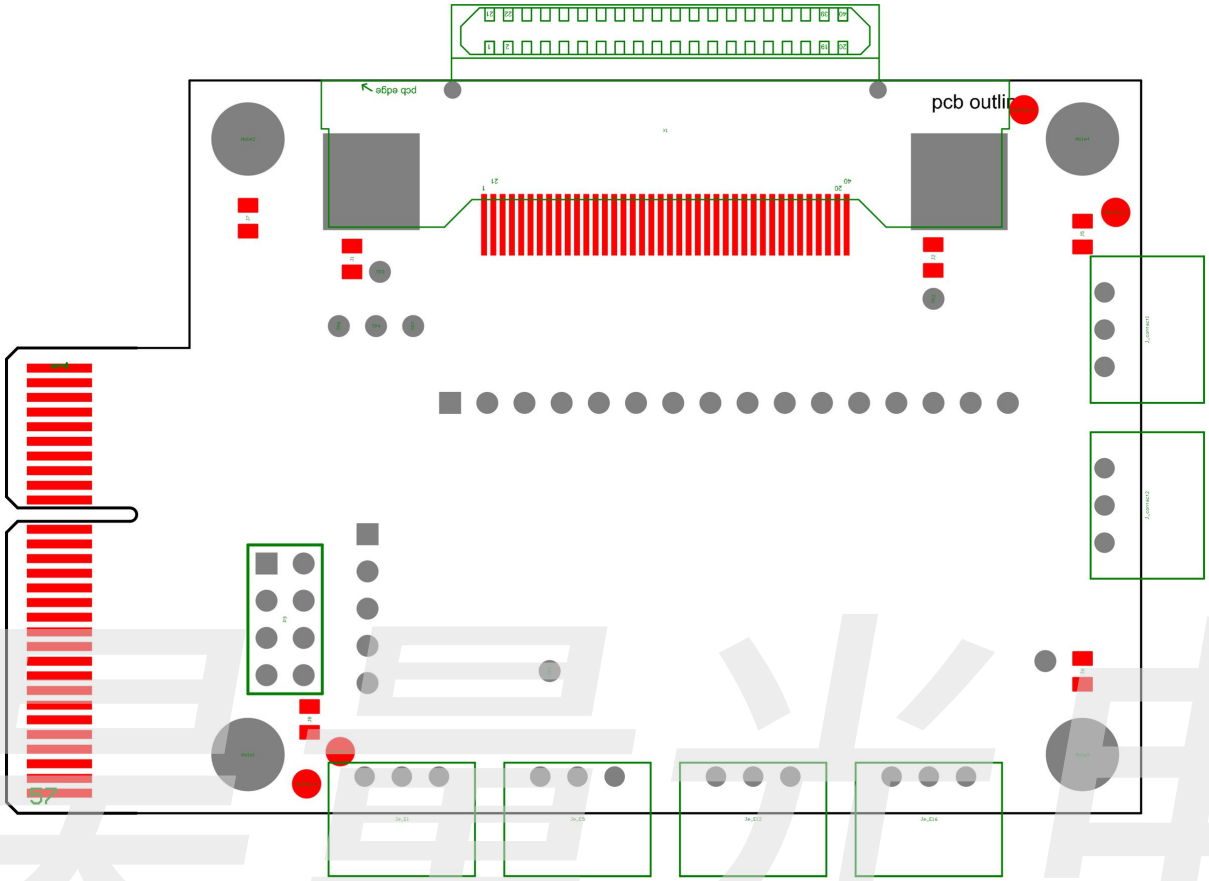
10.4.2 Schematic



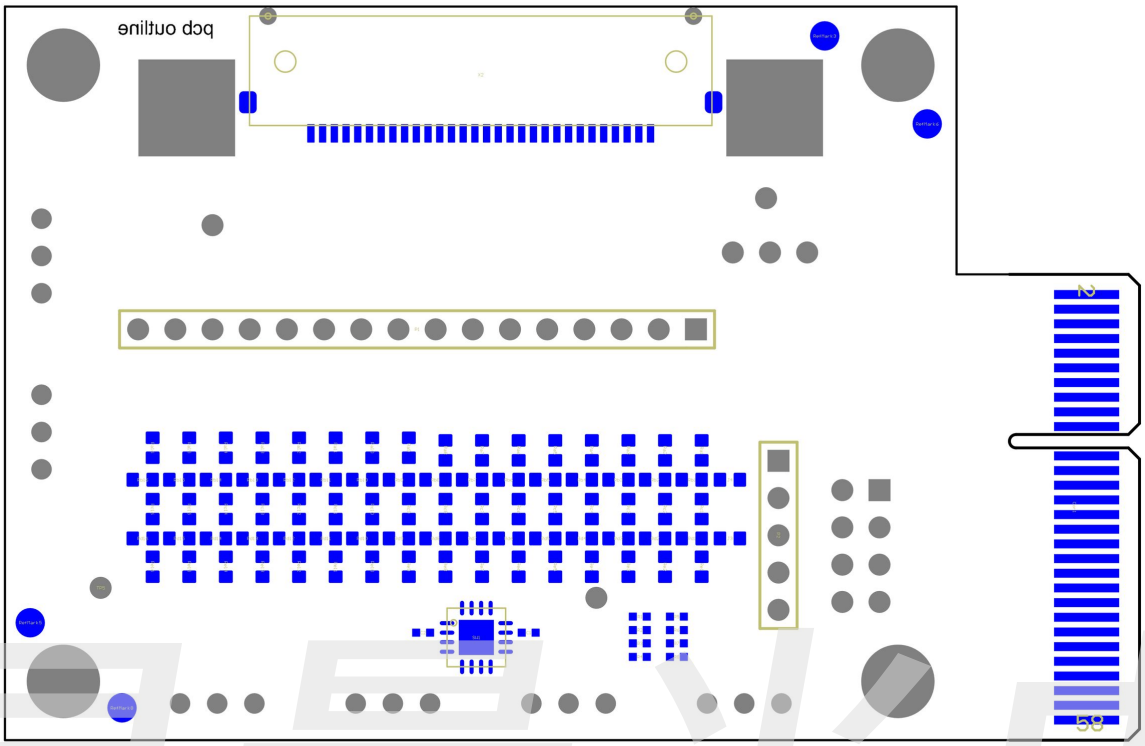


10.4.3 Assembly

Top

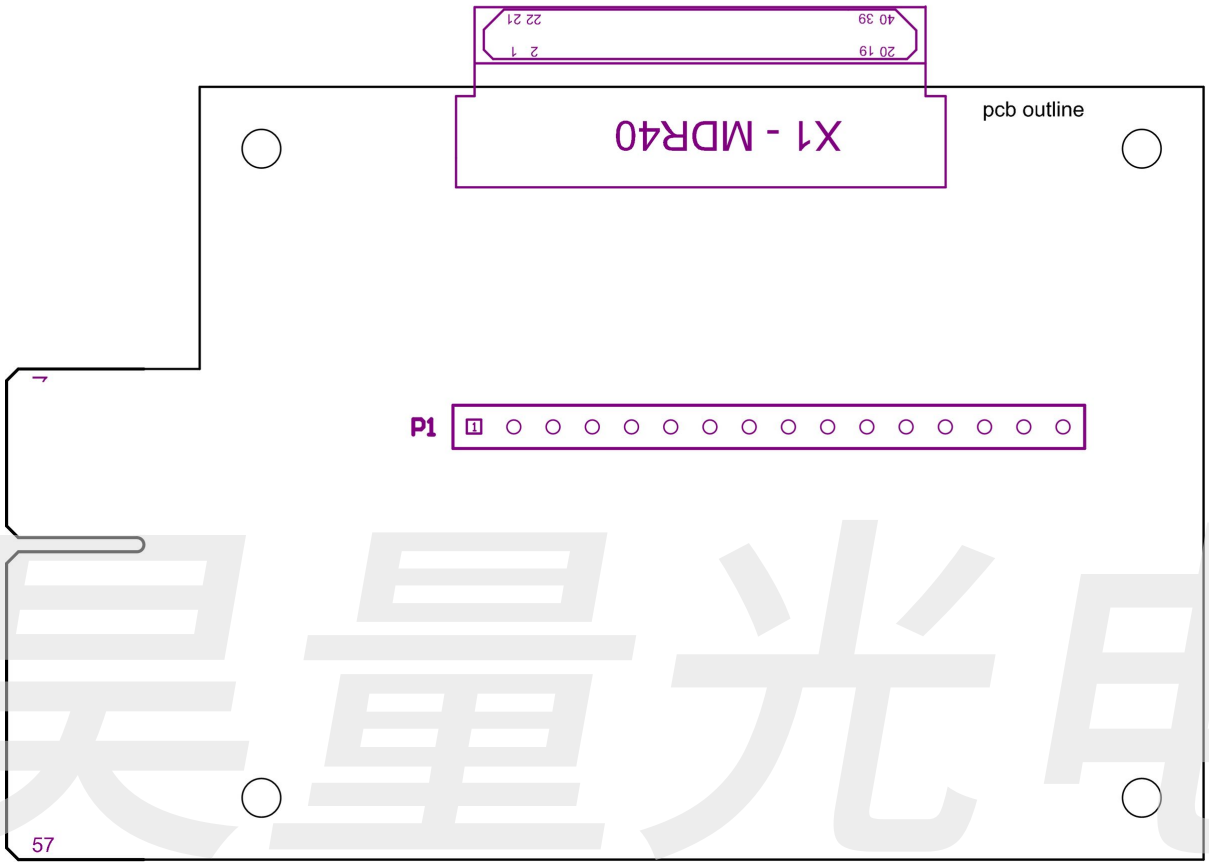


Bottom

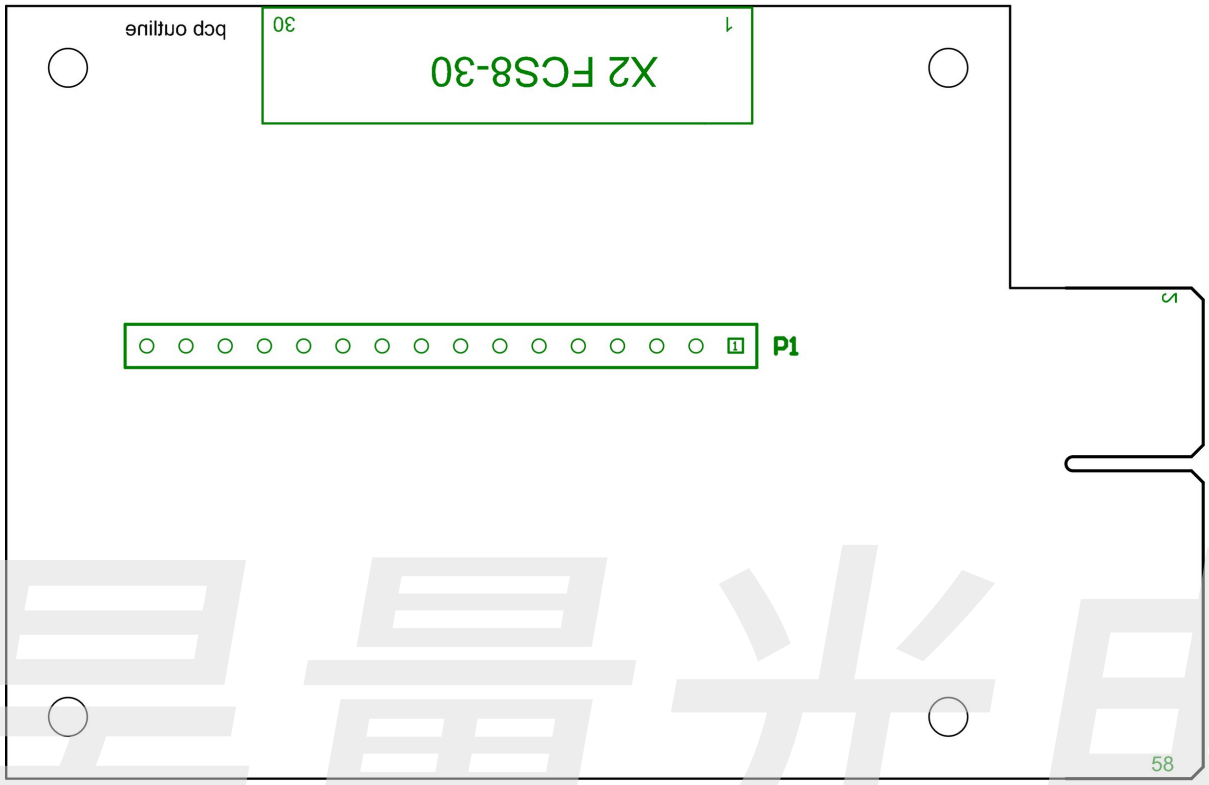


10.4.4 Connector

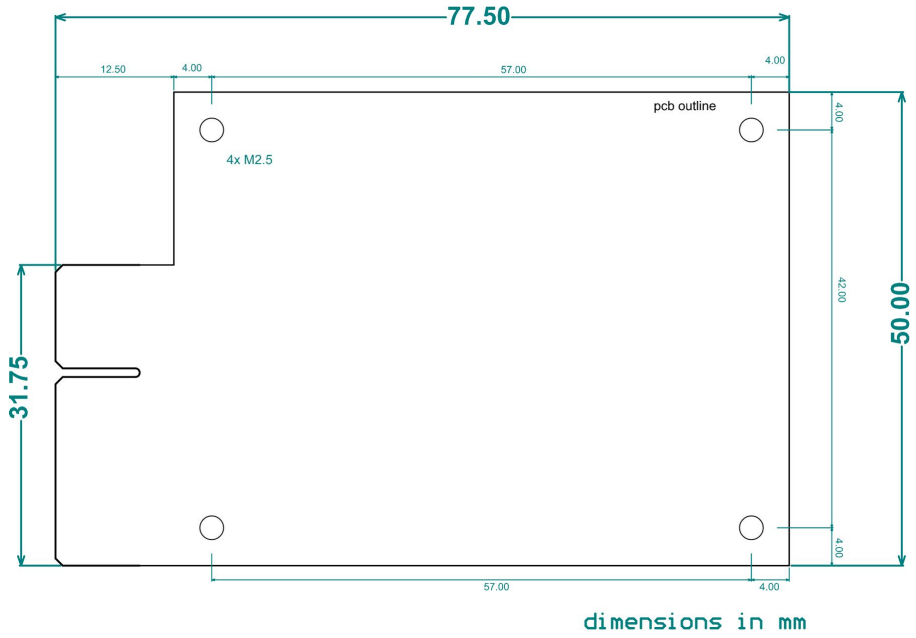
Top Connectors



Bottom Connectors



10.4.5 Mechanical



11 Changelog

Revision 15

- Initial release of this document

Revision 16

- Added D0 and D2 commands to COMInterface

Revision 19

- Updated D2 command in COMInterface
- Added additional data formats to appendix
- Added measurement software description

Revision 21

- Excluded 0xC2, 0xC3 and 0xC5 commands
- Added 0xB5 - Get temperature command
- Added IsoIOport
- Updated 0xD2 and 0xD1 command

Revision 24

- Added Measurement Terminal description
- Added Accessories EitPhantom2011 and EITSC58conAdapter

Revision 25

- Added Set Single-Ended or Differential Measure Mode Option to 0xB0 und 0xB1 commands
- Added no excitation to 0xB0 commands
- Updated Wifi command 0xF1, excluded 0xF4 command
- Added 0xCB - FrontIOs command
- Added 0xCC - Power Plug Detect command
- Minor changes in 0xB0, 0xB1, 0x90 command
- Updated Setup Config Screen in the Software to display the framerate as a global parameter

Revision 26

- Changed IsoIOport default for "in 7" and "in 8".
- Updated description of commands 0xBE (Get Ethernet Configuration), 0xC7 (Get Battery Control)
- Changed product name in documentation
- Updated Accessoires Section

Revision 27

- Fixed cmd description 0xB0 Option 0x08 "Set Single-Ended or Differential Measure Mode"
- Fixed cmd description 0xB0 Option 0x04 "Excitation Frequencies"
- Fixed cmd description 0xD2 Firmware information
- Fixed cmd description 0xF1 Option 0x87 Wifi signal strength

Revision 30

- Fixed cmd description 0xB0/0xB1 Option 0x04 "Excitation Frequencies"

- Added cmd 0xB0/0xB1 Option 0x0C "Excitation Switch Type"
- Added 0x11 ACK-Messages to COMInterface
- Changed product name in whole documentation from EIT2020 to EIT
- Data holdup message added in COMInterface
- Added command 0xCF "TCP connection watchdog" to COMInterface
- Added error codes 39 - 44 to error messages in software description

Revision 34

- Added ethernet broadcast functionality
- Changed product name in documentation
- Changes in 0xB0 - Set Measurement Setup (OptionByte 0x06: adjustments for EIT256)
- Changes in 0xB1 - Get Measurement Setup (OptionByte 0x06: adjustments for EIT256)
- Changes in Measured Data in COMInterface
- Updated CMD 0xCA "IsoIOport"

Revision 35

- Updated technical specifications of Isolation of Full Speed USB Interface
- Updated chapter software installation
- Updated 0xB1 - Get Measurement Setup (OptionByte 0x0D ADC Range)
- Moved Datahold Tag from 0x91 to 0x92
- Added LTC description
- Updated Software requirement from Windows XP to 7

Revision 36

- added MDR50 connector