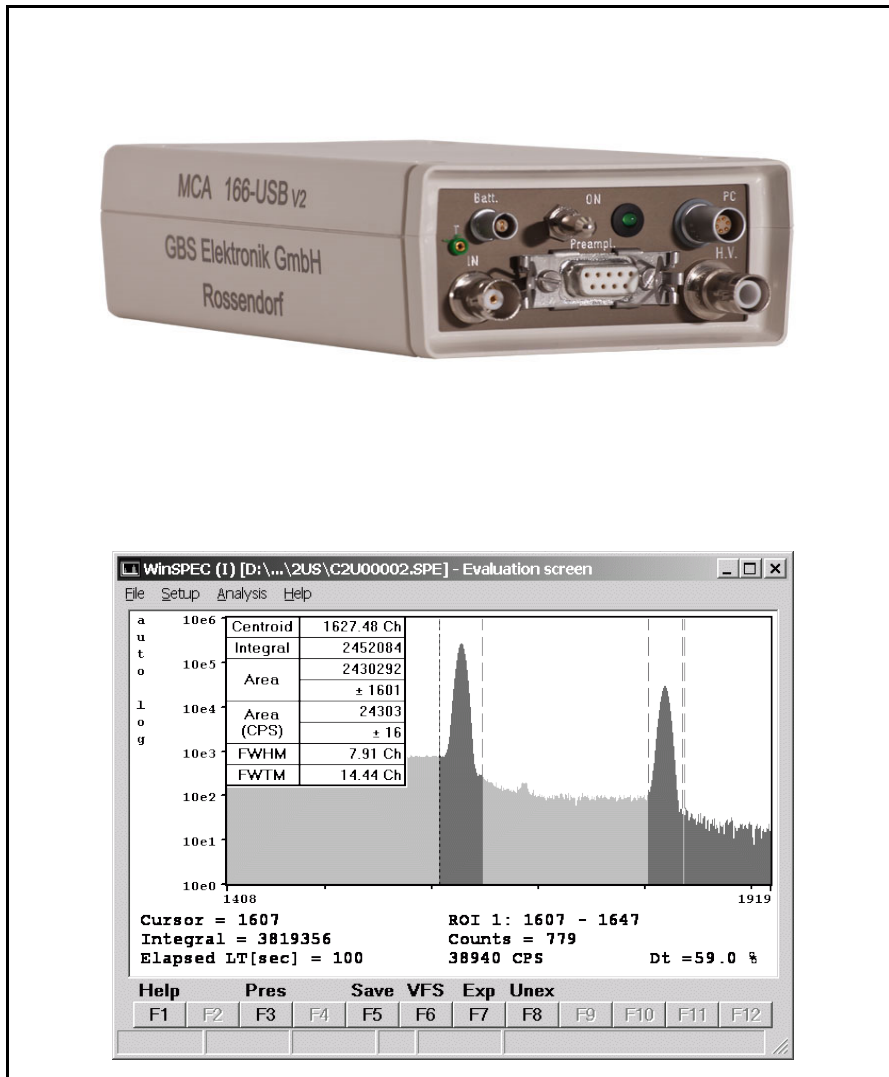# MCA166-USB

Miniature Multi-Channel Analyzer

User's Manual
Version 3.0

Exclusion of liability

The GBS Elektronik GmbH is not liable for errors and does not guarantee the specific utility of the MCA166-USB software or firmware. In particular, the GBS Elektronik GmbH is not liable for indirect or subsequent damages due to errors of the MCA166-USB software or firmware.

The information in this manual has been carefully reviewed and is believed to be accurate and reliable. However, the GBS Elektronik GmbH assumes no liabilities for inaccuracies in this manual. This manual is subject to change without notice.

Last update: 20.12.2007

GBS Elektronik GmbH

Bautzner Landstraße 22

01454 Großerkmannsdorf

Germany

Tel.: 0049 (0)351 21 70 07 – 0

Fax.: 0049 (0)351 21 70 07 – 21

For software updates or problems exceeding the frame of this manual refer to:
Internet: http://www.gbs-elektronik.de
or send email to: kontakt@gbs-elektronik.de

Table of Contents

1 Introduction

The MiniMCA MCA166-USB is a battery powered high performance 4K Multi-Channel Analyzer/Multi-Channel Scaler module comparable in its performance with laboratory grade MCA. High voltage supply for detector and preamplifier power supply are integrated as well as an internal amplifier and pulse shaping network. Together with a small detector it forms a pocket-size gamma spectroscopy system and timer / counter, which is well suited to the demands of field measurements for international safeguards, environmental monitoring, nuclear waste treatment facilities, radioactive transport control and similar applications.

Furthermore, the MCA166-USB supports a vast number of different detectors and its 4k resolution is adequate to support high resolution gamma spectrometry with HPGe detectors.

The MiniMCA software allows to operate the device as a general purpose multi channel analyzer (SPEC, WinSPEC) and multi scaler analyzer (MCS, WinMCS). Additional user programs which support safeguards specific applications as U-235 enrichment verification, spectral radiation survey meter mode supporting active length determination are available. See chapter 3 for more information.

The device can also be operated with the ORTEC software (MAESTRO, GammaVision, ScintiVision etc.); contact ORTEC for details.

The firmware of the MCA166-USB (software of MCAs internal processor) is described in Appendix 3 and 4. Instructions how to change the firmware can be found in Appendix 5.

Basic software functions

- File menu: write/read functions with drive/path and file pick list functions
- Setup menu: ADC, Amplifier, Presets, Memory splitting, MCA mode, MCS mode, Multi spectral recording mode, automated instrument configuration using setup files
- Acquire control: Start, Stop, Clear, Presets
- Automated repeated measurement functions
- Display functions: Automated linear and logarithmic, manual linear Y scale, X axis, expansion and scrolling, Cursor function, ROI setting and processing
- Peak stabilization
- Analysis functions: ROI peak area and integral, FWHM calculation
- Analysis report function
- Energy calibration function: calibration curve using up to 3 peaks
- Incorporated help texts

2 Hardware

2.1 Users safety information

Read all these instructions first!

Save these instructions for later use.

Do Not Remove Connectors

To avoid personal injury or damage of equipment, do not remove the connectors for the High Voltage supply, preamplifier supply, and the input connector until the High Voltage is shut down and the device is switched off at least for 1 minute.

Do Not Remove Covers

Do not open the device before the power is switched off.

2.1.1 Power Source

This device is intended to operate from an internal accumulator set (high performance Li-Ion battery) or together with an external wall adapter. Furthermore the device can be powered directly from a high power USB port (up to the maximum nominal input power of 2,5W).

Do not apply more than the specified voltage to the wall adapter. Usually it is 230VAC nominal. The adapter must be in a good condition. Never use a damaged wall adapter!

2.1.2 High Voltage Supply

Make sure that the High Voltage connector and the High Voltage supply cable of the detector are in a good condition before connecting them to the MCA166-USB or before switching the High Voltage on. Do not allow anything to rest on the HV cable.

Never insert objects of any kind into the High Voltage connector as they may touch the dangerous voltage point. This might cause an electric shock or a damage of the device.

2.2 General Description

The MCA166-USB is an autonomous module. The device has its own battery and provides power supply to radiation detectors. Together with a computer the MCA166-USB forms a multi channel analyzer. The spectra are collected in the memory of the MCA166-USB and are periodically transferred via a serial interface to the computer. The computer is used to setup the MCA166-USB, to display and process the measurement results, and to store the data. The design concept of the MCA166-USB also supports a Multi-Channel Scaler mode.

For operation the MCA166-USB itself has only a power switch and a green LED indicator.

2.2.1 Power management

The MCA166-USB works with a built in rechargeable Li-Ion battery, which has no memory effect and is deep discharge and short circuit protected. If the device is working, the remaining battery life time is permanently checked. When the battery voltage has dropped down to 6.8 Volts, while a data collection is in progress, the MCA166-USB gives out a warning to the user (the lamp will flash irregularly and the software will show an error message). The running measurement is automatically stopped and the detector high voltage together with the preamplifier power supply will be switched off.

Now the user can transfer the measured spectrum to the computer, but should switch the device off afterwards or connect the charger. If the user does not respond, and the battery life time is nearly used up (battery voltage 6.5 Volts) , the device switches off automatically. It is made sure, that the set up and the spectrum gathered previously is saved in the memory of the MCA166-USB for at least several month.

The remaining battery life time is sufficient to transfer the measured spectrum to a computer. It is recommended, before reading out the MCA166-USB, to connect it to the charger.

In the case that the battery voltage drops below its lowest value, the battery itself switches off. This can be, for example, the result of a lasting short circuit at the preamplifier power supply. As a consequence the internal setup and the measured values are lost. The MCA166-USB has to be connected to the charger to recharge the battery.

2.2.2 Switching on

The MCA166-USB is set into operation by turning the power switch on (see Figure 1). After that, the green power ON LED flashes with a frequency of 2.5 Hz if no MMCA software runs on the connected computer. If the power switch is already in position ON (for example after an automatic switch off due to a complete loss of battery power) switch OFF and to ON again. When the power ON LED does not flash, connect the MCA166-USB with the charger and repeat once more. The MCA166-USB is on line with the computer if the LED flashes with 1.2 Hz. Irregular flashing of the lamp indicates an error.

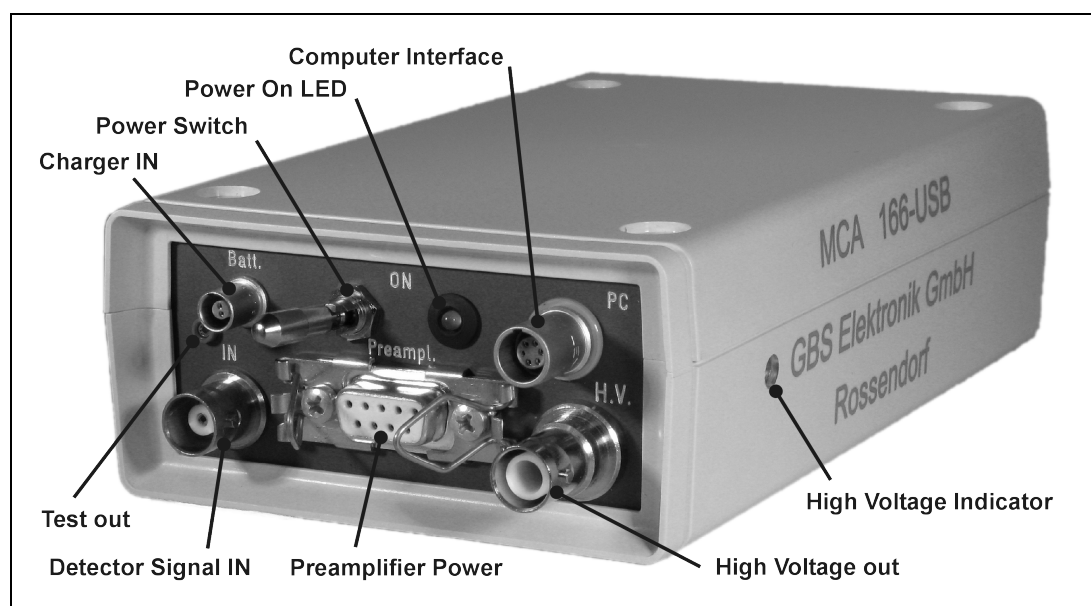Computer Interface
Power On LED
Power Switch
Charger IN
Batt.
ON
PC
IN
Preampl.
H.V.
MCA 166-USB
GBS Elektronik GmbH
Rossendorf
Test out
Detector Signal IN
Preamplifier Power
High Voltage out
High Voltage Indicator

Figure 1:

Front Panel of MCA166-USB

2.2.3   Charging the MCA166-USB

Charging the MCAs battery is usually done by connecting the wall adapter to the "Charger In" connector and power it from the mains supply. The wall adapter is able to deliver enough power to operate the MCA166-USB stationary under a maximum load condition (HV on, detector draws up to 2W preamplifier power). If the MCA166-USB is operated on a high power USB hub and the wall adapter isn't connected, the MCA166-USB can be charged from the USB port. But there is a limitation of 2,5W nominal input power which is the maximum power that can be drawn from a high power USB port (real usable input power depends on hub voltage, voltage loss on USB cable and connectors and efficiency of internal charger; it is usually about 2W). This is enough power to operate the MCA166-USB without detector or with a low power detector (e.g. CdZnTe or NaI). If a detector draws to much power from the MCA166-USB (e.g. some HPGe), charging via USB is still working but the battery runs out slowly.

If the MCA166-USB is connected to a working high power USB hub, charging will start after 6 seconds automatically. It is possible to disable USB charging by some newer application programs such as WinSpec. This is very useful if the MCA is operated at a laptop computer which runs on battery. Connecting the wall adapter to the MCA166-USB and supply it with power will always disable USB charging.

The old battery chargers supplied with the MCA166 (3 pin LEMO connector) are not compatible with the MCA166-USB! Don't try to connect them to the MCA166-USB. The connectors are mechanically different.

2.2.4   Connecting the MCA166-USB to a Computer

Communication between MCA166-USB and host computer is possible via a standard serial connection (RS232) or via USB interface. For RS232 communication the RS232 cable with SUB-D9 connector must be connected between the MCAs PC-socket and the computer, for USB communication the special USB cable with LEMO connector must be used respectively. Two baud rates can be used for communication on both interfaces, 38.400 Baud and 307.200 Baud. Both baud rates are usable on both interfaces but most computers doesn't support 307.200 Baud on the RS232 interface. The firmware of the MCA166-USB is able to detect the baud rate and changes it accordingly. Newer application programs tries to connect the MCA with the higher baud rate first, if it doesn't work they try it with 38.400 Baud. This method guarantees that communication always works on the highest possible baud rate without user interaction and maintains compatibility to older MCAs (supporting only 38.400 Baud). Because the firmware of the MCA166-USB is able to detect the baud rate, it is also possible to control this MCA with older application programs which support only 38.400 Baud.

Running the MCA166-USB on a Palmtop computer or Pocket PC is principal possible but needs a special communication cable. Please contact GBS-Elektronik for availability. For proper wiring see Table 5: Wiring Diagram of the PC Interface Connector (female Lemo 0S-306) on Page 21.

2.2.5 Connection of detectors

To connect a detector **switch the device off**. Before connecting the high voltage connector of the detector make sure that the built in High Voltage supply has the **correct polarity** and that the power consumption of the detector does not **exceed the maximum value (0.5mA or 0.25 Watt).**

On the right side of the MCA166-USB case (front view) the high voltage indicator is located (Figure 1). If it shines red a positive HV module is inserted, if it shines blue a negative HV module is present. If the indicator is black / dark gray no HV module is present. Furthermore, it is possible to check the HV polarity by software; go to the menu setup / high voltage and look at indicated polarity. The correct values for HV, polarity, and power consumption should be found in the detectors manual.

- Check by the detector manual that the pin assignment of the preamplifier power supply connector is compatible with the MCA166-USB, and that the power consumption does not exceed the maximum ratings (see Table 6, Chapter 6.2).

- Connect the cable for the preamplifier power supply of the detector to the concerning female D9 connector on the MCA166-USB and attach it by the clamps.

- Plug the BNC Signal connector of the detector to the female BNC input connector of MCA166-USB.

- Plug the High Voltage connector of detector into the female SHV connector of MCA166-USB.

If problems occur, it is possible to check the detector signal and the main amplifier signal (Amplifier Test out, see figure 1) with a scope. A typical signal which should be seen is shown in figure 2.
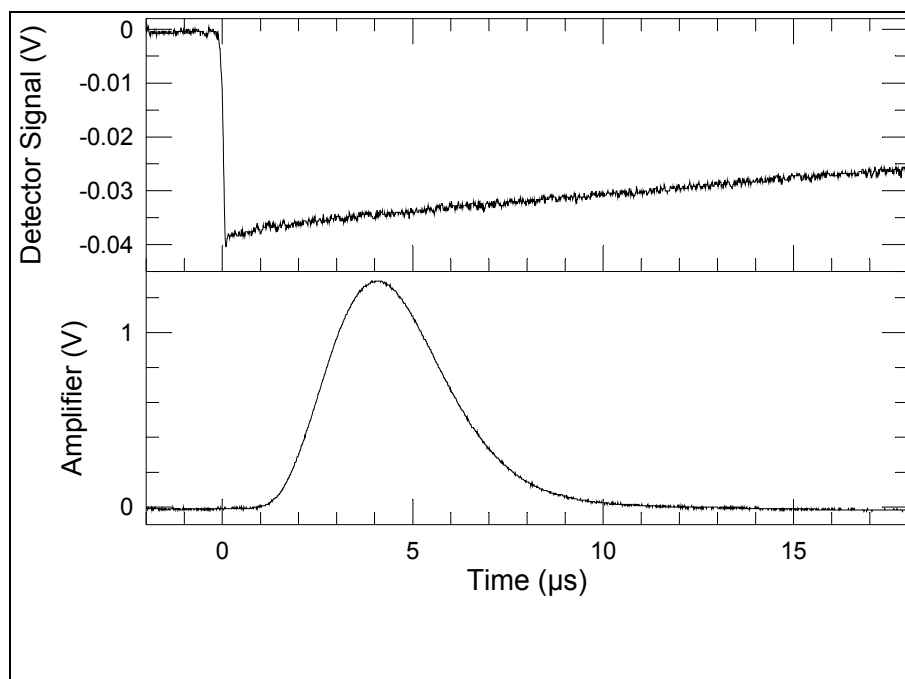
Figure 2:

*Typical signals which can be seen at the detector preamplifier out and the amplifier test out. This example: CdZnTe Detector SDP310/Z/60, measuring a Cs137 sample, amplifier setting 200*0.91, negative pulses, 1µs shaping time, pulse height equals to about channel 662 of 1024 channels.*

Typical errors which may be observed with the detector preamp signal are:

- Rise time is too slow (>0.5 µs). This may cause that the pile up rejector does not work correctly. There may be even a large amount of regular pulses rejected and especially the high energetic part of the spectrum seriously affected. When using such detectors switch pile up rejection off.

- The fall time is too fast (time constant resp. 1/e fall time < 40µs). This causes that the pole/zero setting cannot be correctly adjusted. Consequences may be peak shift and peak broadening with higher count rates and increased low energy spectrum cutoff. Try to use another preamplifier.

2.2.6 Additional external battery pack and Power Pack

Because of the changed charging mechanism of the MCA166-USB the older battery packs and power packs cannot be used! The MCA166-USB offers a higher battery capacity, improved efficiency and is able to supply more power on ±24V than the older MCA166. So two main features offered by the battery pack / power pack are now partially integrated in the MCA itself and makes this devices obsolete. If more battery capacity is needed, please contact GBS-Elektronik GmbH.

3 Software

3.1 Overview

There are several programs to operate the MCA166-USB. All of them serve a special purpose. There are programs for MS-DOS and Windows (32 Bit). The MS-DOS software was originally developed for the HP 200LX, whose footprint coincident with the footprint of the MCA166. The Windows programs were developed to operate the MCA166 more comfortably. Besides, they are (together with at least Windows 98 or Windows 2000) essential to connect the MCA166-USB via USB cable, for the rest, only RS232 communication is possible.

Furthermore, there are programs and software components to view and evaluate the measured data.

The following sections give brief descriptions of the programs. You can find more detailed information on the CD delivered together with the device. You should also use the online help of each program.

The diagnostics menu of each program shows the primary currents of the preamplifier power supply (±12V and ±24V). This are not the currents that draws the detector. However, this currents can be easily calculated using the following formula:

$$I_{out} \approx 0.8 \cdot I_{Batt} \cdot \frac{U_{Batt}}{U_{out}}$$

3.2 SPEC (MS-DOS), WinSPEC (Windows)

These programs are the default programs for measuring spectra. It supports semi-automated measurements of gamma ray spectra with the MCA166-USB. The spectra are stored on mass storage media on a PC. There are two variants of WinSPEC. WinSPEC-I (for Inspectors) was designed for standard use. WinSPEC-A (for Automation) was designed for unattended measurements. Special features of WinSPEC-A are:

- Automatic restart of the program after lost of mains power, program or operating system crash
- Sending commands to a special hardware module on LPT1 for monitoring the "State of health" (already integrated) and other states (not integrated yet)
- Writing zip files
- Data file retrieval to a flashcard
- Archiving of data files
- Writing a log file

3.3 MCS (MS-DOS), WinMCS (Windows)

These programs support semiautomated measurements of time distributions with any radiation detector (HPGe, NaI, CdTe, CdZnTe and neutron counters. There are two variants of WinMCS. WinMCS-I (for Inspectors) was designed for standard use. WinMCS-A (for Automation) was designed for unattended measurements. It contains the same special features like WinSPEC-A (see above).

3.4 U235 (MS-DOS), WinU235 (Windows)

These programs support stabilized U-235 enrichment verification measurements with a NaI or CdZnTe detector (PMCN, PMCC). The algorithm bases on absolute intensity measurement of the 186 keV photon energy. It also needs a two point intensity calibration with two standards.

Reevaluation/recalibration using previously recorded spectra is possible. Verification results are documented in a report file, which is automatically saved with the extension *.rep.

3.5 UF6 (MS-DOS), WinUF6

Similar to U235 or WinU235, but optimized for HPGe detectors. Only one intensity calibration measurement is necessary.

3.6 LENG

This program supports semiautomated active length measurement (HM-4) using a miniature NaI or CdZnTe detector.

3.7 RATE

This program supports measurements of count rates with any radiation detector (HPGe, NaI, CdTe, CdZnTe and neutron counters). In the most convenient mode of operation only the menu button "measurement" needs to be "pressed" to start a measurement and showing the count rates in form of a LCD display and a bar graph.

3.8 WinSCAN

WinSCAN is used for candu bundle verifications. WinSCAN is in its features somewhere between MCA and MCS. It can measure up to 500 Spectra and save them in one file. Evaluation is similar to MCS, just that there is not only an integral spectrum but one spectrum for every point so it is possible to evaluate courses off arbitrary net or integral ROI areas.

3.9 Identify

Identify is an intelligent, interactive software tool to evaluate spectra measured with the MCA166-USB or other MCAs. It does peak search and nuclide identification. HPGe, CdZnTe and NaI detector gamma spectra are supported. For correct peak search a starting point for the detector resolution and efficiency is assumed based on detector type and size. Features:

- includes full master library of gamma lines (derived from table of radioactive isotopes)
- editor for creating application specific evaluation libraries
- detector function is calculated from detector data sheet, no efficiency calibration needed
- automatic determination of FWHM of the peaks found in a spectrum as function of energy
- supports 1 to 3 point energy calibration
- linear, logarithmic, square root, and double log representation possible
- error estimations/confidence assessment for identified isotopes and visualization (by overlay of calculated spectrum for a certain isotope)
- switching between English and German languages
- interactive nuclide pattern identification
- nuclide assignment suggestions also for each single peak
- instant graphical comparison to simulated spectra by clicking on a nuclide
- Win 3.1 version also available
- IDENTIFY is not included in the MCA price
- IDENTIFY routines are also available as library for MS Windows and Linux

3.10 MCAPlot and MCAPrint

MCAPlot and MCAPrint are programs to view and print spectra recorded with the MCA166-USB. Up to 32 spectra can be viewed at the same time and be printed on one page. Also some evaluation functions are available. MCAPlot is not included in the MCA price. MCAPrint is a free downgraded version of MCAPlot.

3.11 MMCAEVAL

MMCAEVAL is a program for evaluating groups of spectra and extracting information to a MS Access style data base.

3.12 MCAWAND

MCAWand is a program for converting different file formats. Please note that there may be always a loss of information as not all kind of information about the spectrum is supported by every file format. The following information is converted:

- Channel Data
- Energy Calibration
- Energy Calibration points
- Live time, Real time
- Date and Time of measurement
- Spectrum Remark

The following file formats are supported:

Table 1: Supported file formats by MCAWAND

Source	Target
*.spe (MCA166 native)	*.spe (MCA166 native)
*.chn (Ortec)	*.chn (Ortec)
*.spe (Interwinner)	*.spe (Interwinner)
*.spc (Target)	
*.dat	*.dat (two column: energy, Channel content)
*.de1 (Canberra S100)	
*.dat (Silena Gamma 2000)	
*.spk (Röntgenanalytik)	
*.spa (Sarad)	
*.usf (URSA)	

3.13 Miscellaneous

For communication and evaluation some libraries for MS Windows and Linux are available. This may help the make own application programs or to integrate the MCA166-USB in an existing system.

4 MCA166-USB parameters for use with different detectors

This table shall just give an overview of possible settings useful with the MCA166-USB. It is far from being complete. For actual settings with your detector, see the detectors manual.

Table 2: Parameters for different detectors

Type	Model	Manufact.	HV (V)	+ 12 V	-12V	+24V	- 24 V	Input signal polarity	Channels	Gain for 2 MeV full range	Shaping time (µs)
CdZnTe	SDP/Z/60	Ritec	+300	x	x	-	-	neg	1024	90	1
CdZnTe	SDP/Z/20	Ritec	+500	x	x	-	-	neg	1024	80	1
CdZnTe	CZT500	Ritec	+600	x	x	-	-	pos	1024	128	1
NaI	Scintiblock	Crismatec	+500					pos	512	7.2	1
NaI	25825	Scionix	+550	x	x	-	-	pos	512	90	1
NaI	40*40	Amcrys-h	-800	x	x			pos	512	14	1
CsI	Sc1010	Eurorad	-	x	x	-	-	neg	256	160	1
HPGe	GMX-20190-S	Ortec	-3000	x	x	x	x	neg	4096	18	2
HPGe	GEM 15-190	Ortec	+3000	x	x	x	x	pos	4096	24	2
HPGe	GR 2018	Canberra	-3000					neg	4096	11	2
HPGe	GL 0310	Canberra	-2000	x	x	x	-	neg	4096	2.3	2
Si planar	7865S	Ortec	-1000					neg	4096	1.9	2
HPGe	EGC 30-190R	Eurisys	-3000					pos	4096	52	2
HPGe	EGPC 25-185	Eurisys	+2000					neg	4096	35	2

5 Some of the most important photon energies

Table 3: Photon energies for different isotopes

Isotope	Half-life	Energy (keV)	Branching ratio (%)
Am241	432.2y	26.34/59.54	2.4/36
Cd109	462.6d	88.03	3.63
Bi214	19.9min, daughter of Ra226	609.31	46.09
Ce139	137.64d	165.86	79.87
Co60	5.271y	1173.23/1332.49	99.85/99.98
Co57	271.8d	14.4/122.06/136.47	9.16/85.6/10.68
Cs137	30.07y	661.66	85.1
Eu152	13.3.y	121.77/344.28/778.91/ 964.11/1112.07/1408. 00	28.38/26.59/12.98/14.46/ 13.57/20.85
Hg203	46.61d	279.19	81.84
Mn54	312.2d	834.82	99.98
Pb214	26.8min, daughter of Ra226	241.91/295.09/351.86	7.46/19.17/37.06
Ra226	1600y	186.11	3.28
Sn113	115.09d	391.7	64.89
Sr85	64.84d	514.0	98.4
U235	7.04E8y	143.78/163.37/185.73/ 205.33	10.53/4.7/53.15/4.7
Y88	106.65	898.04	94
Pb fluorescence x-rays	-	74.96/72.79/84.99/ 87.34	

6 Technical Data

6.1 MCA166-USB Hardware Specifications

6.1.1 Amplifier

1. Type

   - Gaussian programmable shaping amplifier with pile-up rejector and gated base line restorer

2. Gain

   - From 1 ... 1000

   - Minimum gain step 0.5 channel for 2K scale

   - Gain tuning under visual control of spectrum

   - Gain adjustment by peak stabilizer

   - Integral linearity of amplifier <0.1% for full output range

3. Input

   - accepts positive and negative preamplifier signals

   - input resistance 1kΩ

   - DC coupled, accepting +/-200mV DC levels

   - maximum input signal corresponding to full ADC scale at minimum gain: +/-3V (+/-10V with external attenuator) and input set to direct input. With input set to amplifier the maximum voltage step from preamplifier corresponding to ADC full scale at gain 1 is 9.5V, corresponding to a peak voltage at the test point of 2.2V.

4. Pulse shaping

   - active integrator and single, pole zero compensated differentiator with two shaping time constants (1μs and 2μs)

5. Base line restorer (BLR)

   - Active, gated Base Line Restorer

6. Pole Zero Cancellation (PZC)

   - settings: automated by MCA software and manual setting option

   - designed for decay time of the preamplifier pulses >40μs; screen indicator allows PZC without scope

7. Pile UP Rejector (PUR)

   - settings: OFF and ON

6.1.2 ADC

1. Successive approximation ADC with sliding scale linearisation
   - Channel splitting 128, 256, 512, 1K, 2K, or 4K
   - Full input voltage range 3V, positive and negative (10V with external attenuator)
2. ADC Dead time (including time needed to write into memory)
   - < 8 µsec
3. Nonlinearities
   - Differential nonlinearity (over 95% channel range) <2%
   - Integral nonlinearity (over 95% channel range) <0.1%
4. Peak capturing
   - Works with pulse of gaussian shaping amplifier (range of shaping time constant between 0.5µs and 5µs)
5. Temperature Stability
   - TK 100 (ADC), TK 500 (Amplifier)
6. Discriminators and threshold
   - Digital LLD and ULD (allows channelwise cutting)
   - Analogue threshold (2...60% of ADC range)
   - Multi Channel Scaling (MCS) mode allowing to count the following pulses
   - TTL input pulses (minimum width of flat top 0.2µs) not routed through the ADC (maximum rate 1 MHz)
   - LLD/ULD counting mode, counting pulses within a preset digital window
   - Count rate derived from internal amplifier ICR signal (maximum count rate determined by count rate limitation of amplifier)
   - Dwell time 0.01s to 500s
7. Spectrometric performance:
   - Resolution (FWHM) for typical 500mm² planar HPGe detector for count rates < 10000cps and a Co-57 source at 122keV:
   - 610±20eV at 1µs shaping time
   - 580±20eV at 2µs shaping time
   - Peak shift:
     - < 0.5% for 1µs shaping time and count rates from 1000 to 70000cps
     - < 0.2% for 2µs shaping time and count rates from 1000 to 40000cps
   - deterioration of FWHM < 5% for both conditions
   - Throughput in memory:
     - at least 25000cps in memory at 50000cps input rate and 1µs shaping time
     - at least 13000cps in memory at 30000cps input rate and 2µs shaping time

6.1.3   Power supply

- Power supply adequate for commonly used NaI, HPGe, CdTe and proportional counters
- Current limited mode with indication of overload condition through software; short circuit protected and automated recovery after overload or shortening

6.1.4   Preamplifier Voltages

- ± 12V; maximum current 60mA for each of the voltages
- ± 24V; maximum current 60mA for each of the voltages
- Maximum total power at least 2W

6.1.5   High Voltage

- Single positive or negative modular HV supply reaching from 50V to 3000V
- Indication of high voltage sign and value by software
- Calibration within 1% at maximum of bias voltage
- Maximum current 0.5mA or 0.25W (e.g. 1000V and 0.25mA)

6.1.6   Battery

- Rechargeable Li-ion battery with no memory effect
- Deep discharge protected
- Battery capacity 32Wh at 20°C
- Charger input: 9V ... 14V / 0.8A
- Measured values of the battery life time
  - If no detector connected: 30 hours
  - Expected battery life time with detection systems connected:
    - Standard IAEA NaI detector (ORTEC/SCIONIX): 27 hours
      Bias: 500V, 100µA
      Preamplifier: 100mW
    - Miniature CdTe detection probe (SDP310, RITEC, Latvia): 25 hours
      Bias: 400V, 100nA
      Preamplifier: +12V/20mA, -12V/9.5mA (@8.4V)
    - Standard planar HPGe detector (e.g. Canberra, Ortec): 14 hours
      Bias: 2000V, 1nA
      Preamplifier: ±12V: 35/-25mA; ±24V: 55/-27mA (@8.4V)

6.1.7 Computer Interface

- RS-232C serial port
- USB 1.1; virtual COM port is used
- 38.400Baud and 307.200Baud; both baud rates available on both interfaces
- proprietary software protocol

6.1.8 Mechanical specification

- Size: 155mm x 95mm x 45mm (footprint of palmtop HP200LX)
- Weight: app. 700g (incl. battery)

6.1.9 Environmental ratings

- Range of storage temperature: -20°C ... +60°C
- Range of operational temperatures: 0°C ... 50°C
- Range of humidity: up to 90%, noncondensing

6.2 Diagrams and pin connections
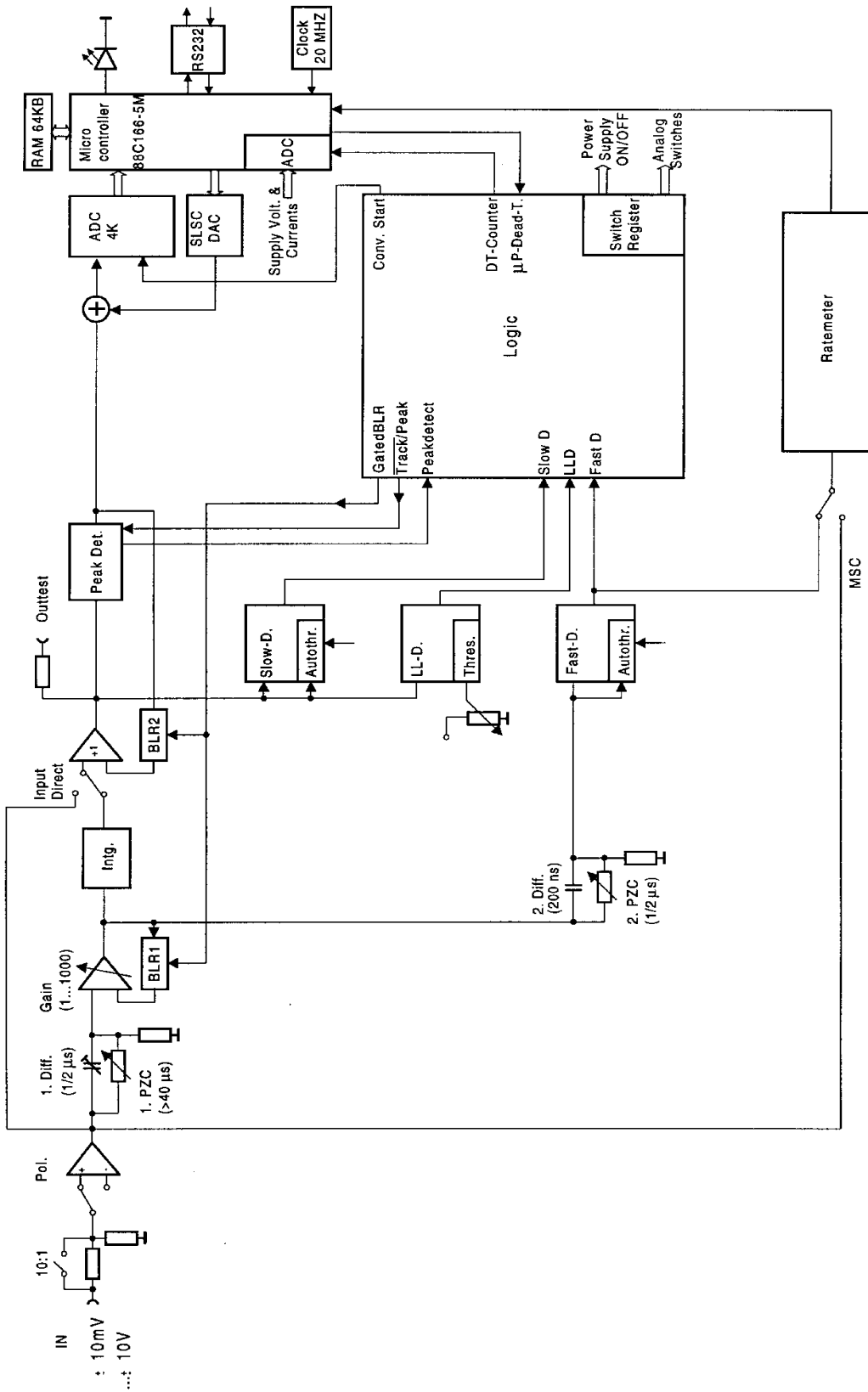
Figure 3: Block Diagram of the MCA166-USB

Table 4: Wiring Diagram of the Charger Input Connector (female Lemo 00-302)

Contact	Signal
1	+ charge voltage
2	- charge voltage

Table 5: Wiring Diagram of the PC Interface Connector (female Lemo 0S-306)

Contact	Signal
1	5V USB
2	D- USB
3	D+ USB
4	GND
5	RxD RS232 (Input)
6	TxD RS232 (Output)

Table 6: Preamplifier Power Supply Connector (D9 female)

Contact	Voltage	max. Current
1	GND	
2	GND	
3	-	
4	+12 V	60mA
5	HV-Inhibit Signal [1]/ ext. Trigger	
6	-24 V	60mA
7	+24 V	60mA
8	-	
9	-12 V	60mA

Table 7: HV-Inhibit Signal

HV-Inhibit mode[2]	Voltage at Pin 5
off	-
Canberra, DFG	< 0.5V
Ortec	+5V

---

1   see Table 4
2   see Windows DLL description

7 Troubleshooting

General Remark:

Before changing anything concerning the hardware, plugging or pulling cables etc. shut down the high voltage and switch off the MCA.

Soldering or manipulating with the SMD boards is strongly not recommended for users. We have specialists for that. If you have a problem which cannot be solved by the table below, contact us. If it is really a hardware fault, we can repair this for a fair price.

1. Software tells "MCA not connected!"
   - Switch on the MCA
   - Check cable between MCA and Computer. Serial cable may be loose or not connected. Connect cable correctly
   - Check MCA battery. Charge MCA battery
   - The MCA is not recognized ("MCA not connected") running MCA DOS Software in a DOS Box on Notebooks
   - Disable Power Save Mode for COM Port

2. MCA lamp does not flash regardless if the switch is on or off
   - MCA battery empty. Charge MCA battery.
   - The MCA firmware programming switches are in a wrong position. The firmware programming switches must be all in "off" position. Do not play around with them!

3. MCA lamp flashes irregularly, Computer tells "MCA power failure" or beeps
   - MCA battery very low. Charge MCA battery.
   - MCA internal or communication error. Switch off and on MCA, restart software.

4. Power Supply Error: Goto Diagnostics, Measurement is not possible
   - MCA battery very low. Charge MCA battery and switch on HV and preamplifier supplies again.
   - Go to diagnostics, blinking power supply currents indicate overload.
   - Short circuit in detector? Check if detector is defective. Disconnect detector, try again. (Do not pull or plug connectors when MCA is on!) Change detector.
   - Too power-hungry preamplifier. A fully charged MCA battery may help sometimes. Otherwise you have to exchange the preamplifier.
   - If the diagnostics menu shows irregular high currents or even overload although there is no detector connected to the MCA and the battery voltage is ok, there may be a fault in the internal power supply. Contact GBS for repair.

5. The threshold of the spectrum seems to be much higher than expected.

- The input polarity may be wrong. Does the threshold depend on radioactive source and count rate? Try some other input polarity.

- P/Z may not be set correctly. Please check if P/Z is correctly adjusted.

- If the lower cutoff rises by itself, this is most probable caused by the auto threshold circuit which is responsible for detecting the noise level and adjusting the threshold to it. This auto threshold circuit uses the negative part of the amplifier output pulses to estimate noise. If now pole zero is severely maladjusted with strong undershoot, the undershoot is misinterpreted as noise and the threshold is increased, especially at medium and high count rates.

- Please check the detector preamplifier waveform. The decay half time of the signal should be optimum 40 us, but never shorter than 25 us. If it is shorter (as experienced with some NaI's) then it is not possible to adjust P/Z correctly, the time constant of the preamp has to be adjusted instead.

6. The high energy part of the spectrum is reduced or even missing

- Check if the behavior changes if the pile up rejector is switched off. Check the signal from the preamplifier. If the preamplifier signal rise time is slower than 500ns, then there is the danger that regular pulses (especially the large ones) are misinterpreted as pile-up. Try to use another preamplifier or just switch the pile-up rejector off.

- Similar effects can be seen if detector cables longer than 50m are used.

7. Dead time shown is very high although the count rate is low.

- There may be some detectors with excessive electronic noise. Try higher fast and slow threshold (You have to edit a setup file with a text editor for this).

8. Too high background rate with BICRON detectors

- Set Fast Threshold to 600 (You have to edit a setup file with a text editor for this) for gain > 50 (Edit setup file)

9. Everything should be okay, but the MCA gets no signals or just measures strange spectra

- wrong input polarity. Check input polarity, just toggle it for a test.

- The amplification is severely wrong set.

- wrong input mode. Check also ADC input mode

10. Strange error messages

- MCA opened recently, flat cables loose? Connect flat cable again .

11. Very bad resolution in a HPGe spectrum

- Detector worn out or defective. Change and regenerate detector.

- Problems with the high voltage module. Try another high voltage module.

- Check cables. Keep them away from switch mode power supplies and other possible sources of electromagnetic noise.

12. Bad resolution in a HPGe spectrum or from a test generator, specially at high amplification factors

- Check electrical noise and EMC compatibility of surroundings. Some laptops and also some switch mode power supply battery chargers may disturb. Remove all connections from the MCA to ground or to mains voltages. Try a different computer / laptop / palmtop.

- Winding the cable to the detector / computer several times around a ferrite ring also may help

13. While opening a spectrum, the MCA program tells "Wrong data format" or "data format error"

- spectrum was created by another program or another program version. Check results, just ignore.

14. too high count rate

- Keep a bigger distance to radiation source. Try a less sensitive detector. Watch for your personal safety!

- Check detector

APPENDIX 1 - MCA166-USB spectral data format

The MiniMCA spectral data format is in compliance with the IAEA SPE Spectral Data Format (see SPEDAC PRO User's Manual Rel. 1.0, IAEA, Feb 1994). All information is stored in a block structured ASCII BSA file so that its content can be viewed with any text editor and the file content can be printed directly.

Each block of the file is identified by a string that starts with the dollar sign ($) and ends with a colon (:). The block name is case sensitive. The structure of the data and the order in which blocks appear in the files is uniquely defined.

There are four general types of blocks used by the MiniMCA software (Release 1998):
1. Data Block
2. ROI information
3. Energy or Enrichment Calibration Data
4. Setup information

Data Block

FILE CONTENT	EXPLANATION
$MCA_166_ID:	**Mini MCA module identification**
SN# 5	Serial number
HW# 9503	Hardware version
FW# 9619	Firmware version
$SPEC_REM:	**Notes about the spectrum**
Mini MCA (MCA166-USB)	Remarks from the operator prompted before file save.
Spectrum measured in facility	
C:\DATA\format.spe	Original storage location of the file.
$DATE_MEA:	**Start date and time of the measurement.**
12/31/1996 16:00:00	mm/dd/yyyy hh:mm:ss
$MEAS_TIM:	**Spectrum measurement time in seconds**
120 203	(lifetime, real-time)
$DATA:	**Spectral data**
0 4095	First channel (i.e. 0) last channel(i.e. 4095)
0	
. . . .	Channel content
0	

ROI Information

FILE CONTENT	EXPLANATION
$ROI:	**ROI (Region of Interest)**
1	Number of ROI's
266 332	ROI start (i.e. 266) ROI begin (i.e. 332)

Energy Calibration Data

FILE CONTENT	EXPLANATION
$ENER_FIT:	**Energy calibration coefficients**
0.000000 0.393559	Offset and slope (KeV/channel)
$ENER_DATA:	**E-calibration data**
2	n point calibration (in this case n=2)
0.000000 0.000000	Channel energy pair (lower)
2981.000000 1173.199951	Channel energy pair (higher)

MCA166-USB

Setup Information

FILE CONTENT	EXPLANATION
$ADC:	**ADC resolution**
4096	Channels
0	LLD (lower level discriminator channel)
3967	ULD (upper level discriminator channel)
$PRESETS:	**MCA Presets**
Integral	Live Time, Real Time, Integral (sum of all counts within a ROI) Area (sum of all net counts within a ROI)
10000	Value
1	ROI number
266 332	ROI begin and end (optional line, depend on the application)
$PZC_VALUE:	**PZC Settings**
1232	DAC adjustment value for PZC (0..2499 mV)
15	Pre-defined time parameter for PZC adjustment
15	Pre-defined time parameter for PZC adjustment
$FAST_DISCR:	**Fast discriminator level**
400	Factory setting for auto threshold
$SLOW_DISCR:	**Slow discriminator level**
400	Factory setting for auto threshold
$THR:	**Threshold value (%)**
2	% of ADC channels; 2-60 % possible
$GAIN_VALUE:	**Amplifier gain**
200	Coarse gain (2...1000)
1.0261	Fine gain (0.5000...1.5000)
$DTC:	**Shaping Time**
1	1, 2 µs[3]
4.0	Actual shaping time[4]
$INPUT:	**ADC input source and polarity**
Amplifier	Amplifier (internal main amplifier), Direct (+3V, -3V), direct input with full range amplitude of 3V
Neg	pos or neg (amplifier input polarity)
$PUR:	**State of the Pile Up Rejector (PUR)**
On	"on" or "off"
$STAB:	**Stabilization**
On	"on" or "off"
230	ROI limits of the stabilization peak
370	
300	Stabilization target channel
$STAB_PARAM	**Stabilization parameter**
10	Stabilization time
25000	Stabilization error
$POWER:	**Preamplifier power supply**
+12= on	"on" or "off"
-12= on	"on" or "off"
+24=off	"on" or "off"
-24=off	"on" or "off"
$HV:	**High voltage polarity and value**
+500V	Value
Unused	HV Inhibit Mode

---

3 1 and 2 µs are the shaping time of a standard MMCA. By use a modified MMCA: 1 = low shaping time, 2 = high shaping time

4 The actual shaping time (only written by newer applications)

Setup information - continued

$MCS_CHANNELS:	Multi Channel Scaler (MCS) Setup
4096	Channels
$MCS_INPUT:	**MCS input**
Extern TTL	Extern TTL (external TTL signal), Input Rate (count rate from the internal amplifier / fast discriminator)
$MCS_TIME:	**MCS time per channel**
10	Time in msec , 0.01 – 655.35 sec
$MCS_SWEEPS:	**Number of sweeps**
0	0 65535
0	MCS Repeat Mode Type
$MODE:	**Operation Mode**
MCA	MCA or MCS
$MCA_REPEAT:	**Number of sweeps**
1	0 65535
0	MCA Repeat Mode Type
$TDF	**Dead Time Correction Factor**
800	100 ... 3000, default 800 nsec
$POWER_STATE:	**Power state at the end of the measurement**
I+12= 8mA	Input current of the DC-DC converters for +12V preamplifier power supply from the battery
I-12= 3mA	Input current of the DC-DC converters for -12V preamplifier power supply from the battery
I+24= 0mA	Input current of the DC-DC converters for +24V preamplifier power supply from the battery
I-24= 1mA	Input current of the DC-DC converters for -24V preamplifier power supply from the battery
IBAT= 135mA	Total current drawn from battery
IHV = 14mA	Input current of the HV DC-DC converter drawn from the battery
ICHR= 0mA	External charger current
UBAT=7900mV	Battery Voltage
UHVs= 418mV	Control voltage of the HV module
$COUNTS:	**Integral counts**
8370252	Sum of all input counts (from the internal amplifier / fast discriminator) of the whole measurement
$PD_COUNTS:	**Integral peak detector counts**
5113594	Sum of all input counts from the peak detector
$RT:	**Real time [s]**
203	
$DT:	**Dead time [ms]**
883027	
$BT:	**Busy time [ms] of the ADC**
64042	
$STAB_OFFSET:	**Current offset**
315	
$STAB_OFFSET_MIN:	**Minimal offset**
310	
$STAB_OFFSET_MAX:	**Maximal offset**
408	
$STAB_COUNTER:	**Stabilization cycles**
19	

Setup information's – continued

FILE CONTENT	EXPLANATION
$REC_COUNTER:	**Counter of received commands**
2823	
$REC_ERROR_COUNTER:	**Counter of received commands with errors**
2	
$SPEC_INTEGRAL:	**Counts in the spectrum**
4098917	
$ROI_INFO:	**ROI information**
1 266 332 299.74 24.19 1233477 1142868 2066	ROI# Begin End Centroid FWHM Integral Area Area_Error

According to the specifications other blocks can be added depending on the type of the application. Almost each application add inspection information to the spectrum file. Following tables show the inspection information of applications written by the Research Center Rossendorf or the GBS Elektronik GmbH.

WinSPEC (Inspection information if no extern analysis application or MGAU is attached)

FILE CONTENT	EXPLANATION
$WINSPEC_INFO:	**WinSPEC inspection information**
ANALYSIS:MGAU	Extern analysis application attached with WinSPEC
1999/001	Inspection number
1999/10/31	Inspection date
Inspector1/Inspector2	Inspector names
DIV1/DIV2	Division or section of the inspectors
Facility name	Facility name
FACC	Facility code
MBA	MBA
Stratum	Stratum
ItemID	Item ID
10.000 ± 1.000 wt%	Declared enrichment
Item description	Item description
767	Instrument code
0156	MMCA ID
8289/025	Computer ID
NAJ	Detector type
9475/020	Detector ID
Collimator	Collimator
Filter	Filter

WinSPEC (inspection information if CsRation is attached)

FILE CONTENT	EXPLANATION
$WINSPEC_INFO:	**WinSPEC inspection information**
ANALYSIS:CSRATIO	Extern analysis application attached with WinSPEC
1999/001	Inspection number
1969/12/31	Inspection date
Inspector1/Inspector2	Inspector names
DIV1/DIV2	Division or section of the inspectors
FACC	Facility code
ItemID	Item ID
1997/01/01	Date of discharge
0.000 Mwd/tU	Burnup
0.000	**Neutron rate**
0.00E00	Calculated Cs ratio at discharge
0.000 %	**Initial enrichment**
ICOD	Instrument code
0156	MMCA ID
8289/025	Computer ID
NAJ	Detector type
9475/020	Detector ID
Collimator	Collimator
Filter	Filter

WinSPEC (inspection information if MGA is attached)

FILE CONTENT	EXPLANATION
$WINSPEC_INFO:	**WinSPEC inspection information**
ANALYSIS:MGA	Extern analysis application attached with WinSPEC
1999/001	Inspection number
1999/01/01	Inspection date
Inspector1/Inspector2	Inspector names
DIV1/DIV2	Division or section of the inspectors
Facility name	Facility name
FACC	Facility code
MBA	MBA
ItemID	Item ID
Batch ID	**Batch ID**
1997/01/01	Date of declaration
10.000 wt% ± 1.000 %	**Decl. isotopic abundance and relative uncertainty for 238 Pu**
20.000 wt% ± 2.000 %	... for 239 Pu
20.000 wt% ± 2.000 %	... for 240 Pu
25.000 wt% ± 2.500 %	... for 241 Pu
10.000 wt% ± 1.000 %	... for 242 Pu
5.000 wt% ± 0.500 %	... for 241 Am
1.000 ± 0.100 %	U / Pu ratio
ICOD	Instrument code
0156	MMCA ID
8289/025	Computer ID
NAJ	Detector type
9475/020	Detector ID
Collimator	Collimator
Filter	Filter

WinSCAN (inspection information)

FILE CONTENT	EXPLANATION
$WINSCAN_INFO:	**WinSCAN inspection information**
1999/001	Inspection number
1999/11/01	Inspection date
Inspector1/Inspector2	Inspector names
DIV1/DIV2	Division or section of the inspectors
Facility name	Facility name
FACC	Facility code
MBA	MBA
ItemID	Item ID
Stratum	**Stratum**
Item description	Item description
ICOD	**Instrument code**
0156	MMCA ID
8289/025	Computer ID
NAJ	Detector type
9475/029	Detector ID
1234/345	Collimator ID
2356/432	Motor drive ID
3478/384	Motor controller ID
3456/419	System ID
2578/456	External amplifier ID
1000/150	Gain (coarse/fine)
0.000	Time constant
OUT	Output
Good measurement.	Remarks

UF6 / WinUF6 (inspection information)

FILE CONTENT	EXPLANATION
$UF6_INSP_INFO:	**UF6 / WinUF6 inspection information**
123	Inspection number
19991006	Inspection date
Inspector ID	Inspector ID
Facility code	Facility code
LOT	LOT ID
MBA	MBA
4.6%+10mmAl	Sample ID
4.462	Declared enrichment
0.003	Declared enrichment uncertainty
wt%	Scale unit (wt%=weight percent or At%= atom percent)
1.200	Wall thickness
0.012	Wall thickness uncertainty
0.340	Attenuation
0.000	Attenuation uncertainty
1.014	Material correction factor
0.000	Material correction factor uncertainty

UF6 / WinUF6 (inspection information) - continued

FILE CONTENT	EXPLANATION
ICOD	Instrument code[5]
0156	MMCA ID
8289/025	Computer ID
NAJ	Detector type
9475/029	Detector ID
Collimator	Collimator
Filter	Filter

U235 / WinU235 (inspection information)

FILE CONTENT	EXPLANATION
$INSP_INFO:	**U235 / WinU235 inspection information**
123	Inspection number
19991006	Inspection date
Inspector ID	Inspector ID
Facility code	Facility code
MBA	MBA
LOT	LOT ID
4.6%+10mmAl	Sample ID
4.462	Declared enrichment
0.003	Declared enrichment uncertainty
wt%	Scale unit (wt%=weight percent or At%= atom percent)
1.200	Wall thickness
0.012	Wall thickness uncertainty
0.340	Attenuation
0.000	Attenuation uncertainty
1.014	Material correction factor
0.000	Material correction factor uncertainty
ICOD	Instrument code[6]
0156	MMCA ID
8289/025	Computer ID
NAJ	Detector type
9475/029	Detector ID
Collimator	Collimator
Filter	Filter

---

5 From here the items are only written and read by WinUF6
6 From here the items are only written and read by WinU235

APPENDIX 2 - MCA166-USB algorithm, formulas

This appendix lists the algorithms used by the MiniMCA for the following functions:

- Area, Area uncertainty
- Centroid
- FWHM
- Smooth
- Strip
- Energy Calibration

The Area and the Area Uncertainty Function:

The area algorithm calculates the number of counts above the background in a ROI. The background area is determined by averaging 4 points on both sides of the peak (the ROI limit points and 3 outer points) and linear fit:

Area = Integral - Background

$$\text{Background} = \frac{\left(ROI_{end} - ROI_{begin} + 1\right)}{8} * \left(\sum_{i=ROI_{begin}-3}^{ROI_{begin}} Spectrum_i + \sum_{i=ROI_{end}}^{ROI_{end}+3} Spectrum_i\right)$$

$$\text{Integral} = \sum_{i=ROI_{begin}}^{ROI_{end}} Spectrum_i$$

Area Uncertainty =

$$\sqrt{Integral + \left(\left(\sum_{i=ROI_{begin}-3}^{ROI_{begin}} Spectrum_i + \sum_{ROI_{end}}^{ROI_{end}+3} Spectrum_i\right) * \left(\frac{\left(ROI_{end} - ROI_{begin} + 1\right)}{8}\right)^2\right)}$$

where,

$Spectrum_i$ = the absolute number of counts in channels i

ROI_{begin} = the ROI's start channel

ROI_{end} = the ROI's stop channel

Centroid:

The peak centroid is the sum of the channel contents[7] times the channel number divided by the sum of the channel contents in the range of the channels above the half maximum:

$$Centroid = \frac{\sum_{i=l}^{h} i * Spectrum_i}{\sum_{i=l}^{h} Spectrum_i}$$

where,
i = channel number
Spectrum$_i$ = net contents of channel i

FWHM:

The Full Width at Half Maximum (FWHM) is the background corrected peak's width at one-half of its maximum amplitude. The algorithm first proceeds down one side of the peak until it finds a channel (e.g. 1203) whose counts fall below the half maximum channel. It is now made sure that the following channel (e.g. 1204) also falls below the half maximum channel. The same algorithm is applied for the other side of the peak. The left and the right FWHM points are the interpolated channels between the counts of the channel below and above the half maximum value.

Smooth:

Smooth averages the current spectrum (the original data will be overwritten) using a binomial formula:

3 point smoothing:

$$Spectrum_i' = \frac{Spectrum_{i-1} + 2 * Spectrum_i + Spectrum_{i+1}}{4}$$

5 point smoothing:

$$Spectrum_i' = \frac{Spectrum_{i-2} + 4 * Spectrum_{i-1} + 6 * Spectrum_i + 4 * Spectrum_{i+1} + Spectrum_{i+2}}{16}$$

where,
$Spectrum_i$ - the original contents of channel i
$Spectrum_i'$ - the smoothed contents of channel i

---

7   background corrected

MCA166-USB

Strip:

Strip subtracts a specified fraction of the spectrum. The Strip factor can be "positive" or "negative" :

$$Spectrum_y^{'} = Spectrum_y - F*Spectrum_x$$

where

$Spectrum_y^{'}$ - Count content of channel i in the resulting spectrum

$Spectrum_y$ - Count content of channel i in the original spectrum

F - Strip factor

$Spectrum_x$ - Count content of channel i in the spectrum to be subtracted

E-calibration:

The Energy calibration allows to convert the x-axis channel numbers into energy values in KeV. The following formula is used:

$$E = S * Channel + O$$

where,
S - Slope
O - Offset

The coefficients can be set as numeric values or by linear fit of two energy channel/peak centroid pairs.

APPENDIX 3 - Description of the communication DLLs[8]

A3.1   MS-Windows DLL for communication with one MCA166-USB

The Dynamic Link Library exists as a 16-bit variant (mca_comm.dll) and as a 32-bit variant (mca32com.dll). The DLL must reside in the system directory of Windows or in the same directory as the calling application. In order to include the DLL in your program, you have to use in Visual Basic the file MCA32COM.BAS (valid only for Win32 applications) or in C/C++ the files MCA_COMM.H and MCA_COMM.LIB or MCA32COM.LIB. All functions and structures are declared in these files.

A3.1.1   Initializing and closing the serial communication port

Before you can use any function, the serial communication port must be initialized. Pass to the function COMM_INIT a string with the name of the serial communication port (e.g. "COM1"). If the return value is unequal zero, the initialization was successful. The serial communication port is unable to initialize if another device (or program) is already using this one. Before you exit your program, you have to close the serial communication port so that it can be used from other programs now. You can use only one communication port at the same time. If you want to change the communication port, you have to close the used communication port before you can initialize another one.

A3.1.2   Sending commands to the MCA

With the function MCA_COMM you can send single firmware commands[9] to the MCA. This function is only intended for testing. For the real work with the MCA you should use the other functions (MMCA_ ...) described in chapter A3.1.3. The function MCA_COMM is declared as follows:

in C/C++

```
ERROR_FLAG FAR PASCAL MCA_COMM(      LPSTR command,
                                     LPSTR rec_data,
                                     unsigned long param1=0,
                                     unsigned long param2=0,
                                     unsigned long param3=0,
                                     unsigned long param4=0);
```

in Visual Basic

```
Declare Function MCA_COMM Lib "Mca32com" (      ByVal Command$,
                                     Received As RECDATA,
                                     ByVal Param1&,
                                     ByVal Param2&,
                                     ByVal Param3&,
                                     ByVal Param4&) As Integer
```

---

8   MCA_COMM.DLL (16 Bit) and MCA32COM.DLL (32 Bit) Version 1.04.0006 for MCA166 Firmware #9901
    MCA_COMX.DLL (16 Bit) and MCA32CMX.DLL (32 Bit) Version 1.00.0006 for MCA166 Firmware #9901
9   The firmware commands are described in Appendix 4.

MCA166-USB

The first argument is a string with the name of the command. In the following table all available commands (use of capital and small letters is possible), the corresponding firmware specific names and the number of the required parameters are listed.

Name of the command	Firmware specific name	Number of the parameters
INIT	CMD_INIT	1
START	CMD_START	2
STOP	CMD_STOP	0
CLEAR	CMD_CLEAR	1
MODE	CMD_SET_MODE	1
ADC_RES_DISCR	CMD_SET_ADC_RES_DISCR	3
THRESHOLD	CMD_SET_THRESHOLD	1
PRESETS	CMD_SET_PRESETS	2
ROI	CMD_SET_ROI	2
REPEAT	CMD_SET_REPEAT	1
TIME_PER_CHANNEL	CMD_SET_TIME_PER_CHANNEL	1
GAIN	CMD_SET_GAIN	2
STABILISATION	CMD_SET_STABILISATION	3
PREAMPLIFIER_POWER	CMD_SET_PREAMPLIFIER_POWER	1
BIAS	CMD_SET_BIAS	2
FAST	CMD_SET_FAST	1
SLOW	CMD_SET_SLOW	1
SHAPING_TIME	CMD_SET_SHAPING_TIME	1
PUR	CMD_SET_PUR	1
MCA_INPUT	CMD_SET_MCA_INPUT	1
MCS_INPUT	CMD_SET_MCS_INPUT	1
INPUT_POLARITY	CMD_SET_INPUT_POLARITY	1
USER_DATA	CMD_SET_USER_DATA	2
SET_MEASURE_PZC	CMD_SET_MEASURE_PZC	2
QUERY_POWER	CMD_QUERY_POWER	0
QUERY_STATE	CMD_QUERY_STATE	0
QUERY_SPECTRA	CMD_QUERY_SPECTRA	2
QUERY_IMAGE	CMD_QUERY_IMAGE	3
QUERY_ENRICHMENT	CMD_QUERY_ENRICHMENT	4
QUERY_USER_DATA	CMD_QUERY_USER_DATA	1
QUERY_CENTROID	CMD_QUERY_CENTROID	2
PZC_TIME_OFFSET	CMD_SET_PZC_TIME_OFFSET	2
TDF	CMD_SET_TDF	1
QUERY_SYSTEM_DATA	CMD_QUERY_SYSTEM_DATA	0
MCS_CHANNEL	CMD_SET_MCS_CHANNEL	1
UF6_ROIS	CMD_SET_UF6_ROIS	3
QUERY_UF6_ROIS	CMD_QUERY_UF6_ROIS	0
QUERY_UF6_INFO	CMD_QUERY_UF6_INFO	0
STAB_PARAM	CMD_SET_STAB_PARAM	2
QUERY_VOLTAGE_CURRENT	CMD_QUERY_VOLTAGE_CURRENT	0

The second argument of MCA_COMM refers to the received data of the command (in Visual Basic always a RECDATA data type, in C/C++ a RECDATA or a command specific structure). Information about the data structures you can find in chapter A3.2 and in the files mca32com.bas (Visual Basic) or mca_comm.h (C/C++). The received data are significant if the command is a QUERY command.

The other arguments are the parameters of the firmware command. These arguments are optional since not all commands use the same number of parameters. Unnecessary arguments you can leave out. Information about the parameters you can find in appendix 4. You can ignore the parameters "preamble", "command" und "end flag" listed in this document. These parameters are assigned the right value automatically. The parameters that you pass to the function MCA_COMM are always long integer. Within the function these parameters are partially passed on to variables with another data type, therefore it is imperative that the maximum values specified in the document are not exceeded. The return value indicates the result of the data transfer. It can be one of the following values:

```
ERROR_OK                  = 0   successful data transfer
ERROR_INTERFACE           = 1   communication port is not initialized
ERROR_INVALID_COMMAND     = 2   unknown command
ERROR_COMMUNICATION       = 3   faulty data transfer
ERROR_INVALID_PARAM       = 4   invalid parameter
```

A3.1.3   Functions to work with the MCA

For the work with the MCA a lot of functions are available. You can subdivide all functions in five groups:

- Function for resetting the MCA (MMCA_RESET)
- Functions for starting and stopping the data acquisition (MMCA_..._AQUIRE)
- Functions for clearing (MMCA_CLEAR_...)
- Functions for setting (MMCA_SET_...)
- Functions for query (MMCA_QUERY_...)

The query functions are handed over a reference to a specific data structure. You can derive the name of the structure from the name of the function, by leaving out "MMCA_" from the function name. (e. g. function MMCA_QUERY_STATE requires the data structur QUERY_STATE). The functions and data structures are listed and described in chapter A3.2. You can also find important information in the files mca32com.bas (Visual Basic) and mca_comm.h (C/C++).

The return value indicates the result of the data transfer. It can be one of the following values:

```
ERROR_OK                  = 0   successful data transfer
ERROR_INTERFACE           = 1   communication port is not initialized
ERROR_COMMUNICATION       = 3   faulty data transfer
ERROR_INVALID_PARAM       = 4   invalid parameter
```

A3.2   Command set

MCA Reset Command:

Name	**MMCA_RESET**
Description	All MCA Parameters will be reset to their initial state and the spectra are cleared. The measurement is aborted, but the states of the preamplifier power and the value of the high voltages are unchanged!
Parameter	-
Results	-
Return Code	ERROR_OK = 0 (command accepted successfully) ERROR_INTERFACE = 1 (communication port not initialised) ERROR_INVALID_COMMAND = 2 (command not yet implemented) ERROR_COMMUNICATION = 3 (serial communication error, off line) ERROR_INVALID_PARAMETER = 4 (parameter out of range)

MCA Acquire Commands:

Name	**MMCA_START_ACQUIRE**		
Description	The acquisition is started or continued with the actual parameter (mode, resolution, …)		
1. Parameter	Clear flag Repeat mode flag Trigger flags	integer value	0 leaves spectrum and times unchanged the start time is ignored 1 spectrum and time are cleared 2 Repeat Mode 1[10] (MCA/MCS) 3 Repeat Mode 2[11] (MCA) 4 Repeat Mode 3[12] (MCA) 5 Repeat Mode 4[13] (MCS) Bit 14 Trigger: 1=on / 0=off Bit 13 Trigger signal edge: 1=pos / 0=neg
[2. Parameter]	Start time	long value	if the 1. Parameter $\neq$ 0, then the value is stored in status field (start_time)
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_STOP_ACQUIRE**
Description	The acquisition is stopped
Parameter	-
Results	The mca_state is set to STATE_STOP
Return Code	see MMCA_RESET command

---

10 Measurement will be stopped if buffer overrun. In MCS mode a differential amplitude spectrum per sweep is collected.

11 Measurement will be resumed, after the buffer is read out.

12 Buffers will be overwritten, even if not read out.

13 MCS Repeat Mode with one integral amplitude spectrum for all MCS sweeps.

MCA Mode Commands:

Name	**MMCA_SET_MODE_MCA**
Description	The command sets the acquire mode to Multi-Channel Analyser
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_MODE_MCS**
Description	The command sets the acquire mode to Multi-Channel Scaler
Parameter	-
Results	-
Return Code	see MMCA_RESET command

MCA Clear Commands:

Name	**MMCA_CLEAR_MEMORY**
Description	The acquisition is stopped and the spectrum is cleared
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_CLEAR_TIME**
Description	The acquisition is stopped and the dead time together with the real time are set to 0
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_CLEAR_ROI**
Description	The acquisition is stopped and the ROI limits are set to LLD and ULD
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_CLEAR_ALL**
Description	The command is a combination of MMCA_CLEAR_ROI, MMCA_CLEAR_MEMORY and MMCA_CLEAR_TIME
Parameter	-
Results	-
Return Code	see MMCA_RESET command

MCA166-USB

MCA Measurement Setup Commands:

Name	**MMCA_SET_PRESET_NONE**		
Description	None preset		
Parameter	-		
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_PRESET_LIVE_TIME**		
Description	The command sets the time for measurement (dead time corrected).		
1. Parameter	Live time	integer value	1 ... 65535 sec
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_PRESET_REAL_TIME**		
Description	The command sets the time for measurement		
1. Parameter	Real time	integer value	1 ... 65535
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_PRESET_INTEGRAL**		
Description	The command sets the value for the ROI integral		
1. Parameter	Integral	long value	1 ... 4294967295
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_PRESET_AREA**		
Description	The command sets the value for the ROI area		
1. Parameter	Area	long value	1 ... 4294967295
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_ADC_RES_DISCR**		
Description	The command sets the ADC resolution and the software discriminator range.		
1. Parameter	Channels	integer value	128, 256, 512, 1024, 2048 or 4096
2. Parameter	LLD	integer value	0 <= LLD < ULD
3. Parameter	ULD	integer value	LLD < ULD <(Channels-(Channels>>5))
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_ROI**		
Description	The command sets the begin and end of the ROI for the preset integral and area.		
1. Parameter	ROI begin channel	Integer value	LLD <= begin < end
2. Parameter	ROI end channel	Integer value	LLD < end <= ULD
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_REPEAT**		
Description	The command sets the number of sweeps for repetitive measurement.		
1. Parameter	Sweeps	Integer value	0 ... 65535
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_MCS_CHANNEL**		
Description	The command sets the number of channels for the MCS and Rate mode.		
1. Parameter	MCS channels	Integer value	0 (only for Rate mode), 128, ... 4096
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_TIME_PER_CHANNEL**		
Description	The command sets the dwell time per channel for the MCS mode.		
1. Parameter	Time[14]	Integer value	1 ... 65535
Results	-		
Return Code	see MMCA_RESET command		

---

14 It is the real time per channel multiplied with 10 msec.

Name	**MMCA_SET_MCS_INPUT_ICR**
Description	The command selects the count rate signal as MCS input.
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_MCS_INPUT_TTL**
Description	The command selects the external TTL signals as MCS input.
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_MCS_INPUT_DISCR**		
Description	The command selects the software discrimination of ADC-input pulses as MCS input.		
1. Parameter	Channels	integer value	128, 256, 512, 1024, 2048 or 4096
2. Parameter	LLD	integer value	0 <= LLD < ULD
3. Parameter	ULD	integer value	LLD < ULD< (Channels-(Channels>>5))
Results	-		
Return Code	see MMCA_RESET command		

MCA Hardware Setup Commands:

Name	**MMCA_SET_GAIN**		
Description	The command sets the amplifier coarse and fine gain.		
1. Parameter	Coarse gain	integer value	2, 5, 10, 20, 50, 100, 200, 500 or 1000
2. Parameter	Fine gain[15]	integer value	5000 ... 15000 (10000 if Coarse Gain=1000)
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_MCA_INPUT_AMPLIFIER_POS**
Description	The command sets the amplifier input polarity to positive.
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_MCA_INPUT_AMPLIFIER_NEG**
Description	The command sets the amplifier input polarity to negative.
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_MCA_INPUT_DIRECT_POS**
Description	The command sets the ADC input to direct (0 to +3V input range) and PUR to off.
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_MCA_INPUT_DIRECT_NEG**
Description	The command sets the ADC input to direct (0 to -3V input range) and PUR to off.
Parameter	-
Results	-
Return Code	see MMCA_RESET command

---

15 It is the real fine gain value multiplied with 10000.

Name	**MMCA_SET_THRESHOLD**		
Description	The command sets the analog threshold		
1. Parameter	Threshold	integer value	2 ... 60
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_SHAPING_TIME_LOW**
Description	The command sets the amplifier shaping time (1µs). [16]
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_SHAPING_TIME_HIGH**
Description	The command sets the amplifier shaping time (2µs). [17]
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_SET_PILE_UP_REJECTION**		
Description	The command sets the pile up rejection.		
1. Parameter	Pile up switch	integer value	0 turns PUR off and ≠ 0 turns PUR on
Results	-		
Return Code	see MMCA_RESET command		

---

16 The values may be different for some MMCA configurations.
17 The values may be different for some MMCA configurations.

Name	**MMCA_SET_FAST**		
Description	The command sets the fast discriminator threshold.		
1. Parameter	Threshold	integer value	0 … 2499, default 400
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_SLOW**		
Description	The command sets the slow discriminator threshold.		
1. Parameter	Threshold	integer value	0 … 2499, default 400
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_PZC_MANUAL**[18]		
Description	The command sets the PZC value and returns the PZC offset.		
1. Parameter	PZC value	integer value	0 … 2499
Result array Byte offset 128	Number of measured pulses	integer value	
Result array Byte offset 130	averaged negative offset of measured input pulses	integer value	
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_PZC_VALUE**[19]		
Description	The command sets the PZC value.		
1. Parameter	PZC value	integer value	0 … 2499
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_PZC_TIME_OFFSET**		
Description	The command sets the time offset for the PZC measurement.		
1. Parameter	Offset for low DTC	integer value	0 … 31, default 15
2. Parameter	Offset for high DTC	integer value	0 … 31, default 15
Results	-		
Return Code	see MMCA_RESET command		

---

18 Please note that this function may cause a firmware hang up if the following requirements are not meets:
- statistical distributed input pulses with a count rate between 500 and 30000 Cps
- the significant peak is located above the first quarter of the spectrum
19 The function requires a reference to a pzc result array, but the function has no effect on this array at the moment.

MCA Stabilisation Commands:

Name	**MMCA_SET_STABILISATION**		
Description	The command sets the peak stabilisation.		
1. Parameter	Stabilisation flag	Integer value	0 turns stabilisation off 1 turns stabilisation on (keep peak at current position) >1 stabilisation channel (move peak to the stabilisation channel and hold peak at the stabilisation channel)
2. Parameter	Peak ROI begin	Integer value	LLD<= begin < end
3. Parameter	Peak ROI end	Integer value	begin < end <= ULD, (end-begin) < 250
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_STAB_PARAM**		
Description	The command sets the peak stabilisation parameters.		
1. Parameter	Stabilisation time	Integer value	default 10 sec interval
2. Parameter	Stabilisation area	long value	default 25000
Results	-		
Return Code	see MMCA_RESET command		

MCA Detector Power Commands:

Name	**MMCA_SET_PREAMPLIFIER_POWER**		
Description	The command sets the preamplifier power.		
1. Parameter	Preamplifier switch	integer value	0X80 -24V 0X40 +24V 0X20 -12V 0X10 +12V
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_HIGH_VOLTAGES**		
Description	The command sets the detector high voltage and controls the HV-inhibit-signal.[20]		
1. Parameter	High voltage	integer value	0 ... 3000

2. Parameter	High voltage inhibit	long value	i = 0 Inhibit off
			i = 1 "Canberra-/ DGF-mode", HV shut down
			      if inhibit signal high
			i =-1 "Ortec-mode", HV shut down
			      if inhibit signal low

Results	-		

---

20 The high voltages will ramp up or down by the MMCA.

Return Code	see MMCA_RESET command

MCA Data Setup Commands:

Name	**MMCA_SET_TDF**		
Description	The command sets the dead time correction parameter.		
1. Parameter	Tdf	integer value	100 ... 3000, default 800
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_USER_DATA**		
Description	The command stores a 32 bit value in the MCA parameter memory.		
1. Parameter	Number	integer value	0 ... 63
2. Parameter	Value	long or float value	0 ...
Results	-		
Return Code	see MMCA_RESET command		

Name	**MMCA_SET_UF6_ROIS**		
Description	The command sets the begin and the end of a ROI used by the other UF6 commands.		
1. Parameter	ROI number	integer value	1 ... 3
2. Parameter	ROI begin	integer value	LLD<= begin < end
3. Parameter	ROI end	integer value	begin < end <= ULD, (end-begin) < 250
Results	-		
Return Code	see MMCA_RESET command		

MCA Query Commands:

Name	**MMCA_QUERY_USER_DATA**		
Description	The command reads 32 user data (32 bit values) from the MCA parameter memory.		
1. Parameter	Number n	integer value	0 ... 63
Result array	User_data[n], ...	long or float value	0 ...
Return Code	see MMCA_RESET command		

Contents of the MMCA User Date memory:[21]

Byte Offset 0[22]	U235: Enrichment Constant 1 WinSPEC: Index for the analysis program WinSCAN: Scan method and scan parameter
Byte Offset 4	U235: Enrichment Constant 2 WinSPEC-A: Repeat mode options WinSCAN: Gridlines offset and period
Byte Offset 8	U235: Enrichment Constant 3 WinSPEC-A: Repeat mode options
Byte Offset 12	U235: Enrichment Constant 4 WinSPEC-A: Repeat mode options
Byte Offset 16	MCA Application Mode: U235, LENG, SPEC, MCS, RATE, UF6, WMCA , WMCS , WSPC, WSCN, WUF6, W235, MCAE
Byte Offset 20	HV Inhibit Mode: 0, 1, 2, -1[23]
Byte Offset 22	HV Preset value: 0 ... 3000
Byte Offset 24	Power Switches
Byte Offset 28	Stab. Preset: auto/Channel
Byte Offset 30	Stab. on/off
Byte Offset 32	Repeat Mode : 0, 1, 2, 3, 4 ,100 (100 means software repeat mode), (Bit 15: 0=inactive, 1=active)
Byte Offset 34	El. Repeats (Software mode)
Byte Offset 36	HV Inhibit Mode: 0, 1, 2, -1[24]
Byte Offset 38	HV Preset value: 0 ... 3000
...	-
Byte Offset 48	Energy Calibration: Channel 1 or Null
Byte Offset 52	Energy Calibration: Channel 2 or Null
Byte Offset 56	Energy Calibration: Energy 1 or actual Slope
Byte Offset 60	Energy Calibration: Energy 2 or actual Offset

---

21 Each program should set and test at least the MCA application mode (Byte offset 16). Developer of new applications for the MCA166-USB should publish the shorthand expression of their application.
22 The bytes 0 ... 15 are used differently by the single applications
23 Actual values (are used for automatic restart in applications for unattended measurements)
24 Values can be used as preset values after the MCA powered on

Byte Offset 64	1. ROI pair (ROI End)[25]
Byte Offset 66	1. ROI pair (ROI Begin)
Byte Offset 68	2.
...	...
Byte Offset 120	15.
Byte Offset 124	16. ROI pair
Byte Offset 128[26]	U235/UF6: Inspection description WinMCS: 16 ROI pairs of the MCA spectrum (similar the other ROIs)

---

25 In applications with MCS spectrum the ROIs of the MCS spectrum are saved here. In WinMCS the ROIs of the MCA spectrum are saved behind it.

26 The bytes 128 ... 255 are used differently by the single applications

Name	**MMCA_QUERY_POWER**		
Description	The command reads the MMCA power state.		
Parameter	-		
Result array	Battery current [mA]	long value	
Byte Offset 4	HV primary current [mA]	long value	
Byte Offset 8	+12 primary current [mA]	long value	
Byte Offset 12	-12 primary current [mA]	long value	
Byte Offset 16	+24 primary current [mA]	long value	
Byte Offset 20	-24 primary current [mA]	long value	
Byte Offset 24	Battery voltage [mV]	long value	
Byte Offset 28	HV [V][27]	long value	
Byte Offset 32	HV state	long value	
Byte Offset 36			
Byte Offset 40	Current high voltage [V]		
Byte Offset 44			
Byte Offset 48	Power Switches	long value	0X80 -24V ON 0X40 +24V ON 0X20 -12V ON 0X10 +12V ON 0X02 HV ON
Byte Offset 52	Charger current [mA]	long value	
...			
Byte Offset 64	Battery current [mA] at stop	long value	
Byte Offset 68	HV primary current [mA] at stop	long value	
Byte Offset 72	+12 primary current [mA] at stop	long value	
Byte Offset 76	-12 primary current [mA] at stop	long value	
Byte Offset 80	+24 primary current [mA] at stop	long value	
Byte Offset 84	-24 primary current [mA] at stop	long value	
Byte Offset 88	Battery voltage [mV] at stop	long value	
Byte Offset 92	HV [V][28] at stop	long value	
Byte Offset 116	Charger current [mA] at stop	long value	
...			
Byte Offset 128	mca_state[29]	integer value	
Return Code	see MMCA_RESET command		

---

27 multiply value with 1.2 V
28 multiply value with 1.2 V
29 see MMCA_QUERY_SYSTEM_DATA

MCA166-USB

Name	**MMCA_QUERY_STATE**		
Description	The command reads the MMCA state.		
Parameter	-		
Results array	MCA acquire mode	MCA_MODE	MODE_MCA, MODE_MCS
Byte Offset 2	MCA preset	MCA_PRESETS	PRESET_NONE PRESET_REAL PRESET_LIVE PRESET_INT PRESET_AREA
Byte Offset 4	Preset value	long	
Byte Offset 8	Elapsed preset	long	
Byte Offset 12	Repeat value	integer value	
Byte Offset 14	Elapsed sweeps	integer value	
Byte Offset 16	MCS time per channel [∗0.01 sec]	integer value	
Byte Offset 18	Elapsed time per channel [∗0.01 sec]	integer value	
Byte Offset 20	Real time [sec]	long	
Byte Offset 24	Counts per seconds	long	
Byte Offset 28	Dead time [msec]	long	
Byte Offset 32	Busy time [msec]	long	
Byte Offset 36	MCA channels	integer value	128, ... , 4096
Byte Offset 38	Threshold	integer value	2 .. 60
Byte Offset 40	LLD	integer value	0 ...
Byte Offset 42	ULD	integer value	
Byte Offset 44	ROI begin (preset integral/area)	integer value	
Byte Offset 46	ROI end (preset integral/area)	integer value	
Byte Offset 48	Amplifier coarse gain	integer value	2, ... , 1000
Byte Offset 50	Amplifier fine gain	integer value	5000 ... 15000
Byte Offset 52	Slow discriminator value	integer value	0 ... 2499 mV
Byte Offset 54	Fast discriminator value	integer value	0 ... 2499 mV
Byte Offset 56	High voltage	integer value	0 ... 3000 V
Byte Offset 58	High voltage polarity	integer value	0 positive / 1 negative
Byte Offset 60	Power switches	integer value	0X80 -24V ON 0X40 +24V ON 0X20 -12V ON 0X10 +12V ON 0X02 HV ON
Byte Offset 62	PZC value	integer value	0 ... 2499 mV
Byte Offset 64	Time offset for PZC and DTC low[30]	integer value	0 ... 15
Byte Offset 66	Time offset for PZC and DTC high	integer value	0 ... 15
Byte Offset 68	Stabilisation state or	integer value	

---

30 value multiplied with 0.4µsec

Name	**MMCA_QUERY_STATE**		
	channel		
Byte Offset 70	Stabilisation result	MCA_STAB_RESULT	STAB_STARTED, STAB_ACTIVE, STAB_IDLE, STAB_ERROR
Byte Offset 72	Stab. ROI begin	integer value	
Byte Offset 74	Stab. ROI end	integer value	
Byte Offset 76	ADC input	MCA_INPUT_ADC	ADC_AMPLIFIER, ADC_ DIRECT_POS, ADC_DIRECT_NEG
Byte Offset 78	ADC input polarity	MCA_INPUT_POL	INPUT_POLARITY_POSITIVE, INPUT_POLARITY_NEGATIVE
Byte Offset 80	DTC	integer value	
Byte Offset 82	PUR state	MCA_PUR	OFF, ON
Byte Offset 84	MCS input	MCA_INPUT_MCS	MCS_INPUT_TTL, MCS_INPUT_RATE, MCS_INPUT_DISCR
Byte Offset 86	MMCA number	integer value	
Byte Offset 88	MMCA hardware version	integer value	
Byte Offset 90	MMCA firmware version	integer value	
Byte Offset 92	MCS Channels	integer value	
Byte Offset 94	Last power state	integer value	
Byte Offset 96	Battery capacity [%]	integer value	
Byte Offset 98	0	integer value	
Byte Offset 100	Start time	long	
Byte Offset 104	TDF	integer value	
Byte Offset 106	Last command flag	integer value	
Byte Offset 108	Last command parameter 1	integer value	
Byte Offset 110	Last command parameter 2	integer value	
Byte Offset 112	Last command parameter 3	integer value	
Byte Offset 114	Buffer state[31]	Integer value	
Byte Offset 116	Internal DAC value	long	
Byte Offset 120	Differential dead time [0/00]	integer value	
Byte Offset 122	HV inhibit mode	integer value	
Byte Offset 124	HV inhibit state	integer value	
Byte Offset 128	MMCA state[32]		
Return Code	see MMCA_RESET command		

---

31 see MMCA_QUERY_SPECTRA
32 see MMCA_QUERY_SYSTEM_DATA

MCA166-USB

Name	**MMCA_QUERY_SYSTEM_DATA**		
Description	The command reads the MMCA system data.		
Parameter	-		
Results array	0	integer	
Byte Offset 2	Peak detect counter	48 Bit integer	
Byte Offset 8	0	integer	
Byte Offset 10	Fast detect counter	48 Bit integer	
Byte Offset 16	0	integer	
Byte Offset 18	Peak detect counter at time-1	48 Bit integer	
Byte Offset 24	0	integer	
Byte Offset 26	Fast detect counter at time-1	48 Bit integer	
Byte Offset 32	Busy Time [ms] at time-1	long	
Byte Offset 36	MMCA on Time [s]	long	
Byte Offset 40	Real Time [s] of previous sweep	long	
Byte Offset 44	Dead Time [ms] of previous sweep	long	
Byte Offset 48	Start time of previous sweep	long	
Byte Offset 52	0	long	
Byte Offset 56	Elapsed sweeps	long	
Byte Offset 60	Busy time [ms] of previous sweep	long	
Byte Offset 64	0	integer	
Byte Offset 66	Peak detect counter of previous sweep	48	
Byte Offset 72	0	integer	
Byte Offset 74	Fast detect counter of previous sweep	48	
Byte Offset 80	Counter of stabilisation steps	long	
Byte Offset 84	Current stabilisation offset	integer	
Byte Offset 88	Maximal negative stabilisation offset	integer	
Byte Offset 92	Maximal positive stabilisation offset	integer	
Byte Offset 96	Counter of received commands	long	
Byte Offset 100	Counter receive errors	long	
Byte Offset 104	0	integer	
Byte Offset 106	Last command flag	integer value	
Byte Offset 108	Last command parameter 1	integer value	
Byte Offset 110	Last command parameter 2	integer value	
Byte Offset 112	Last command parameter 3	integer value	
Byte Offset 114	Buffer state[33]	Integer value	

33 see MMCA_QUERY_SPECTRA

54

Byte Offset 116	Stabilisation area preset	long	
Byte Offset 120	Stabilisation time preset	integer value	
Byte Offset 122	Low shaping time[34]	byte value	
Byte Offset 123	High shaping time[35]	byte value	
Byte Offset 124	0	integer value	
Byte Offset 128	MMCA state		STATE_POWER_ON, STATE_READY, STATE_RUN, STATE_SUSPEND, STATE_FINISH, STATE_ERROR, STATE_FAIL
Return Code	see MMCA_RESET command		

Name	**MMCA_QUERY_IMAGE[36]**		
Description	The command read the MMCA spectrum image (120x256).		
1. Parameter	No. of 1. Channel	0 ...	
2. Parameter	Compress factor	1 ... 32	
3. Parameter	Vertical full scale		
Result array	Channel image N	byte value	
Byte Offset 1	Channel image N+1	byte value	
...			
Byte Offset 127	Channel image N+127	byte value	
Return Code	see MMCA_RESET command		

---

34 The value must be multiplied with 0.1 µsec
35 The value must be multiplied with 0.1 µsec
36 This command should only be used to display the spectrum in a 120 x 256 pixel window.

Name	**MMCA_QUERY_SPECTRA**		
Description	The command read the MMCA spectrum data.		
1. Parameter	No. of 1. Channel and read out control[37]	integer value	
2. Parameter	Compress factor (1 ... 32)	long value	
Result array	Channel contents N	long value	
Byte Offset 4	Channel contents N+1	long value	
...			
Byte Offset 124	Channel contents N+31	long value	
Byte Offset 128	Buffer state[38]	integer value	
Return Code	see MMCA_RESET command		

Name	**MMCA_QUERY_ENRICHMENT**		
Description	The command reads the begin and end of the 2 ROIs		
1. Parameter	Peak begin	integer value	LLD<= begin < end
2. Parameter	Peak end	integer value	begin < end <= ULD, (end-begin) < 250
3. Parameter	Background begin	integer value	LLD<= begin < end
4. Parameter	Background end	integer value	begin < end <= ULD, (end-begin) < 250
Result array	Begin peak ROI	long value	
Byte Offset 4	End peak ROI	long value	
Byte Offset 8	Begin background ROI	long value	
Byte Offset 12	End background ROI	long value	
Return Code	see MMCA_RESET command		

---

37 Read out control Bit[15:12], Channel number: Bit[11:0]

Bit[15:12] = 0x00 Read spectrum
Bit[15:12] = 0x01 Read MCS amplitude spectrum
Bit[15:12] = 0x02 Read spectrum buffer and unlock buffer
Bit[15:12] = 0x03 Read spectrum buffer and lock buffer
Bit[15:12] = 0x07 Read MCS amplitude spectrum buffer and lock buffer
38 Read out buffer state

Bit[15] = 1   Buffer filled
Bit[14] = 1   Buffer overrun

Name	**MMCA_QUERY_UF6_INFO**	
Description	The command reads current dead time, real time the integral, begin and end of the UF6 ROIs	
Parameter	-	
Result array	Real time	long value
Byte Offset 4	Dead time [msec]	long value
Byte Offset 8	Integral ROI 1	long value
Byte Offset 12	Integral ROI 2	long value
Byte Offset 16	Integral ROI 3	long value
Byte Offset 20	Begin ROI 1	long value
...	...	long value
Byte Offset 40	End ROI 3	long value
Return Code	see MMCA_RESET command	

Name	**MMCA_QUERY_UF6_ROIS**	
Description	The command reads the begin and end of the 3 ROIs	
Parameter	-	
Result array	Begin ROI 1	long value
Byte Offset 4	End ROI 1	long value
Byte Offset 8	Begin ROI 2	long value
Byte Offset 12	End ROI 2	long value
Byte Offset 16	Begin ROI 3	long value
Byte Offset 20	End ROI 3	long value
Return Code	see MMCA_RESET command	

Name	**MMCA_QUERY_CENTROID**		
Description	The command reads peak centroid of the specified ROI		
1. Parameter	Peak ROI begin	integer value	LLD<= begin < end
2. Parameter	Peak ROI end	integer value	begin < end <= ULD, (end-begin) < 250
Result	Peak centroid	float value	
Return Code	see MMCA_RESET command		

Name	**MMCA_QUERY_ON_LINE**
Description	The command checks the communication between the MCA and the PC.
Parameter	-
Results	-
Return Code	see MMCA_RESET command

Name	**MMCA_QUERY_VOLTAGE_CURRENT**		
Description	The command reads the voltages and currents from the MMCA.		
Parameter	-		
Results array	Charger current [mA]	long	
Byte Offset 4	HV primary current [mA]	long	
Byte Offset 8	Battery current [mA]	long	
Byte Offset 12	Battery voltage [mV]	long	
Byte Offset 16	HV reference voltage [V]	long	
Byte Offset 20	HV control voltage [V]	long	
Byte Offset 24	+12V primary current [mA]	long	
Byte Offset 28	+24V primary current [mA]	long	
Byte Offset 32	-24V primary current [mA]	long	
Byte Offset 36	-12V primary current [mA]	long	
Return Code	see MMCA_RESET command		

A3.3 MS-Windows DLL for the communication with several MCA166-USB

This Dynamic Link Library allows the communication with several MCA166-USB. It can manage the serial communication ports COM1 to COM20.

Die Dynamic Link Library exists as a 16-bit variant (mca_comx.dll) and as a 32-bit variant (mca32cmx.dll). The DLL must reside in the system directory of Windows or in the same directory as the calling application. In order to include the DLL in your program, you have to use the in Visual Basic the file MCA32CMX.BAS (valid only for Win32 applications) or in C/C++ the files MCA_COMX.H and MCA_COMX.LIB or MCA32CXM.LIB. All functions and structures are declared in these files.

The use of the function is analogous to the functions of the DLLs for the comunication with one MCA166-USB (see chapter A3.1). However, following differences exist: The return value of the function COMM_INIT is an index of the communication port. You have to pass this index to each function as first parameter so that you can distinguish the communication ports.

The return value indicates the result of the data transfer. It can be one of the following values:

```
ERROR_OK                    = 0    successful data transfer
ERROR_INTERFACE             = 1    communication port is not initialized
ERROR_COMMUNICATION         = 3    faulty data transfer
ERROR_INVALID_PARAM         = 4    invalid parameter
ERROR_FORBIDDEN_INDEX       = 5    forbidden index
```

APPENDIX 4 - Description of the MCA166-USB firmware functions

MCA Reset Command:

Command name	CMD_INIT										
Description	All MCA Parameters will be reset to their initial state and the spectra are cleared. The measurement is aborted, but the states of the preamplifier power and the value of the high voltage are unchanged!										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		0		0				end flag
Byte String (HEX)	A5	5A	41	00	00	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	All										
MCA state	STATE_READY										

MCA Acquire Commands:

Command name	CMD_START										
Description	Depending on the value of the first parameter(cf) the command clears the spectrum and the time or not and then the measurement will be started. If the first parameter is not equal zero, the spectrum and the time where cleared and the second parameter(st) is stored as start time.[39]										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		cf		st				end flag
Byte String (HEX)	A5	5A	42	00	cf	00	st_l	..	..	st_h	B9	9B
Used setup value	All										
Affected setup value	start time										
MCA state	STATE_RUN										

Command name	CMD_STOP										
Description	The measurement is stopped immediately. There is no synchronisation with the internal clock or dwell time.										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		0		0				end flag
Byte String (HEX)	A5	5A	43	00	00	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	None										
MCA state	STATE_STOP										

---

39 See Appendix 3

MCA Mode Command:

Command name	**CMD_SET_MODE**									
Description	Depending on the value of the first parameter (mod): MODE_MCA = 0 MODE_MCS = 1 the command set MMCA to the required mode by activating the previous specified setup parameter. The command will be ignored and responded with a error messages, if a measurement is still running.									
Command syntax										
Format	integer		integer		integer	long				integer
Parameter	preamble		command		mod	0				end flag
Byte String (HEX)	A5	5A	45	00	mod	00	00	00	00	00	B9	9B
Used setup value	coarse_gain, fine_gain, mca_input_pol, mca_input_adc									
Affected setup value	mca_mode, stab_state									
MCA state	STATE_READY									

MCA Clear Command:

Command name	**CMD_CLEAR**									
Description	A running measurement is stopped immediately. Depending on the value of the first parameter (clear): CLEAR_MEMORY = 0 CLEAR_TIME = 1 CLEAR_ROI = 2 CLEAR_ALL = 3 the command clears the spectrum, the time information, the ROI limits exclusively or all together.									
Command syntax										
Format	integer		integer		integer	long				integer
Parameter	preamble		command		clear	0				end flag
Byte String (HEX)	A5	5A	44	00	clear	00	00	00	00	00	B9	9B
Used setup value	None									
Affected setup value	real time, dead time, ROI begin, ROI end									
MCA state	STATE_READY									

MCA Measurement Setup Commands:

Command name	CMD_SET_ADC_RES_DISCR										
Description	The command set the ADC resolution(res: 128, 256 ... 4096), the LLD(Low level discriminator) and the ULD(Upper level discriminator) to the values of its three parameter. The command will be ignored and responded with a error messages, if one ore more parameter are invalid or a measurement is still running.										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		res		LLD, ULD				end flag	
Byte String (HEX)	A5	5A	46	00	res_l	res_h	LLD_l	LLD_h	ULD_l	ULD_h	B9	9B
Used setup value	None											
Affected setup value	channels, lld, uld, stab_state											
MCA state	STATE_READY											

Command name	CMD_SET_PRESETS										
Description	The command sets the choice(pre) PRESET_NONE = 0 PRESET_REAL = 1 PRESET_LIVE = 2 PRESET_INT = 3 PRESET_AREA = 4 and value(val) of the preset. The parameter val must be lower then 65536 for PRESET_REAL and PRESET_LIVE! If a measurement is running, only val can be changed.										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		pre		val				end flag	
Byte String (HEX)	A5	5A	48	00	pre_l	pre_h	val_l	...	...	val_h	B9	9B
Used setup value	None											
Affected setup value	presets, preset_value											
MCA state	STATE_READY											

Command name	CMD_SET_ROI										
Description	The command sets the begin and end of the ROI for the preset integral and area.(LLD <= begin < end and LLD < end <= ULD)										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		beg		end				end flag	
Byte String (HEX)	A5	5A	49	00	beg_l	beg_h	end_l	end_h	00	00	B9	9B
Used setup value	LLD, ULD											
Affected setup value	roi_begin, roi_end											
MCA state	STATE_READY											

Command name	**CMD_SET_REPEAT**										
Description	The command sets the number(rep: 0 … 65535) of sweeps for repetitive measurement.										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		rep		0				end flag	
Byte String (HEX)	A5	5A	4A	00	rep_l	rep_h	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	Repeat										
MCA state	STATE_READY										

Command name	**CMD_SET_MCS_CHANNEL**										
Description	The command set the number of channels(ch: 128 … 4096) for MCS mode										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		ch		0				end flag	
Byte String (HEX)	A5	5A	63	00	ch_l	ch_h	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	mcs_channels										
MCA state	STATE_READY										

Command name	**CMD_SET_TIME_PER_CHANNEL**										
Description	The command sets the dwell time(tpc: 1 … 65535) per channel for the MCS mode[40].										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		tpc		0				end flag	
Byte String (HEX)	A5	5A	4B	00	tpc_l	tpc_h	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	time_per_channel										
MCA state	STATE_READY										

---

40 It is the real time per channel multiplied with 10

MCA Hardware Setup Commands:

Command name	**CMD_SET_GAIN**								
Description	The command sets the amplifier coarse(cg) and fine gain(fg). cg:　　2, 5, 10, 20, 50, 100, 200, 500 or 1000 fg:　　5000 ... 15000 (for cg = 1000 fg: 5000 ... 10000)								
Command syntax									
Format	integer		integer		integer		long			integer
Parameter	preamble		command		cg		fg			end flag
Byte String (HEX)	A5	5A	4C	00	cg_l	cg_h	fg_l	fg_h	00	00	B9	9B
Used setup value	None								
Affected setup value	coarse_gain, fine_gain								
MCA state	STATE_READY								

Command name	**CMD_SET_INPUT_POLARITY**								
Description	The command sets the input polarity for the internal amplifier. ip = 0　positive input signals ip = 1　negative input signals								
Command syntax									
Format	integer		integer		integer		long			integer
Parameter	preamble		command		ip		0			end flag
Byte String (HEX)	A5	5A	56	00	ip	00	00	00	00	00	B9	9B
Used setup value	None								
Affected setup value	mca_input_pol								
MCA state	STATE_READY								

Command name	**CMD_SET_MCA_INPUT**								
Description	The command sets the ADC input to amplifier or to direct. ip = 0　input amplifier ip = 3　input direct (0 ... +3V, PUR OFF) ip = 4　input direct (0 ... -3V, PUR OFF)								
Command syntax									
Format	integer		integer		integer		long			integer
Parameter	preamble		command		ip		0			end flag
Byte String (HEX)	A5	5A	54	00	ip	00	00	00	00	00	B9	9B
Used setup value	None								
Affected setup value	mca_input_adc								
MCA state	STATE_READY								

Command name	**CMD_SET_MCS_INPUT**										
Description	The command sets the input for the Multi-Channel Scaler. ip = 0 external TTL signals ip = 1 internal count rate signal ip = 2 counts only pulses with amplitude between LLD and ULD threshold										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		ip		0				end flag
Byte String (HEX)	A5	5A	55	00	ip	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	mca_input_mcs										
MCA state	STATE_READY										

Command name	**CMD_SET_THRESHOLD**										
Description	This command set the analogue threshold value (thr: 2 …60).										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		thr		0				end flag
Byte String (HEX)	A5	5A	47	00	thr	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	Threshold										
MCA state	STATE_READY										

Command name	**CMD_SET_SHAPING_TIME**										
Description	The command sets the amplifier shaping time(dtc). dtc = 1 shaping time low (1 µsec) dtc = 3 shaping time high (2 or 3 µsec)[41]										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		dtc		0				end flag
Byte String (HEX)	A5	5A	52	00	dtc	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	Dtc										
MCA state	STATE_READY										

---

41 The values may be different for some MCA166-USB configurations.

Command name	**CMD_SET_PUR**										
Description	The command sets the pile up rejection.										
Command syntax	pur = 0 turns PUR off pur ≠ 0 turns PUR on										
Format	integer		integer		integer		long				integer
Parameter	preamble		command		pur		0				end flag
Byte String (HEX)	A5	5A	53	00	pur	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	mca_pur										
MCA state	STATE_READY										

Command name	**CMD_SET_FAST**										
Description	This command set the fast discriminator threshold (0 … 2499, default 400).										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	Preamble		command		fast		0				end flag
Byte String (HEX)	A5	5A	50	00	fast	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	Fast threshold										
MCA state	STATE_READY										

Command name	**CMD_SET_SLOW**										
Description	This command set the slow discriminator threshold (0 … 2499, default 400).										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	Preamble		command		slow		0				end flag
Byte String (HEX)	A5	5A	51	00	slow	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	Slow threshold										
MCA state	STATE_READY										

Command name	**CMD_SET_PZC**									
Description	The command performs the Pole Zero Cancellation. pv: 0 … 2499 PZC control voltage pm = 0 set PCZ only pm > 0 set PZC and measure PZC offset									
Command syntax										
Format	integer		integer		integer		long			integer
Parameter	preamble		command		pm		Pv			end flag
Byte String (HEX)	A5	5A	58	00	pm$_l$	pm$_h$	pv$_l$	pv$_h$	00	00	B9	9B
Used setup value	None									
Affected setup value	pzc_value									
Results	If pm > 0 the function returns with the number of counts and the - PZC offset.									
MCA state	STATE_READY									

Command name	**CMD_SET_PZC_TIME_OFFSET**									
Description	The command sets the time offset for Pole Zero Cancellation. t1: 0 … 31 Time offset for low shaping time t2: 0 … 31 Time offset for high shaping time									
Command syntax										
Format	integer		integer		integer		long			integer
Parameter	preamble		command		t1		t2			end flag
Byte String (HEX)	A5	5A	60	00	t1	00	t2	00	00	00	B9	9B
Used setup value	None									
Affected setup value	pzc_dtc_1_offset, pzc_dtc_3_offset									
MCA state	STATE_READY									

MCA Stabilisation Commands:

Command name	CMD_SET_STABILISATION											
Description	The command sets the peak stabilisation. fl: 0 turns stabilisation off 1 turns stabilisation on rb+3<fl<re-3 stabilisation to channel fl rb: Peak ROI begin (LLD<= begin < end) re: Peak ROI end (begin < end <= ULD, (end-begin) < 250)												
Command syntax												
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		fl		rb		re		end flag	
Byte String (HEX)	A5	5A	4D	00	fl_l	fl_h	rb_l	rb_h	re_l	re_h	B9	9B
Used setup value	channels, lld, uld											
Affected setup value	stab_state, stab_roi_begin, stab_roi_end											
MCA state	STATE_READY											

Command name	CMD_SET_STAB_PARAM											
Description	The command sets the stabilisation parameters. st: time interval (default 10) sa: area (default 25000)											
Command syntax												
Format	integer		integer		integer		long				integer	
Parameter	Preamble		command		st		sa				end flag	
Byte String (HEX)	A5	5A	67	00	st_l	st_h	sa_l	sa_l	sa_l	sa_h	B9	9B
Used setup value	-											
Affected setup value	stab_time, stab_area											
MCA state	STATE_READY											

MCA Detector Power Commands:

Command name	**CMD_SET_PREAMPLIFIER_POWER**										
Description	The command sets the preamplifier power(pp). 0X80 -24V on 0X40 +24V on 0X20 -12V on 0X10 +12V on 0XF0 all on 0X00 all off										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		pp		0				end flag	
Byte String (HEX)	A5	5A	4E	00	pp	00	00	00	00	00	B9	9B
Used setup value	None										
Affected setup value	preamplifier_power										
MCA state	STATE_READY										

Command name	**CMD_SET_BIAS**										
Description	The command sets the detector high voltage[42]and controls the HV-inhibit-input. hv: 0 ... 3000 i = 0 Inhibit off i = 1 "Canberra- or DFG-mode", HV shut down if inhibit input < 0.5V i =-1 "Ortec-mode", HV shut down if inhibit input = +5V										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		hv		I				end flag	
Byte String (HEX)	A5	5A	4F	00	hv_l	hv_h	i_l	i_h	00	00	B9	9B
Used setup value	None										
Affected setup value	detector_bias, stab_state										
MCA state	STATE_READY										

---

42 The high voltages will ramp up or down by the MCA166-USB.

MCA Data Setup Commands:

Command name	CMD_SET_TDF										
Description	The command sets the TDF parameter. tdf: TDF(100 ... 3000, default 800)										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		tdf		0				end flag
Byte String (HEX)	A5	5A	61	00	tdf_l	tdf_h	0	0	0	0	B9	9B
Used setup value	None										
Affected setup value	tdf										
MCA state	STATE_READY										

Command name	CMD_SET_UF6_ROI										
Description	The command sets the begin and the end of a ROI used by the other UF6 commands. r: ROI number b: LLD<= b < e e: b < e <= ULD, (e-b) < 500										
Command syntax											
Format	integer		integer		integer		integer		integer		integer
Parameter	Preamble		command		r		b		e		end flag
Byte String (HEX)	A5	5A	64	00	r	00	b_l	b_h	e_l	e_h	B9	9B
Used setup value	LLD, ULD										
Affected setup value	roi_begin[n], roi_end[n]										
MCA state	STATE_READY										

Command name	CMD_SET_USER_DATA										
Description	The command stores a 32 bit value(f) in the MCA parameter memory(64 entries). e: 0 ... 63 parameter number p: 0 ... long or float parameter										
Command syntax											
Format	integer		integer		integer		long or float				integer
Parameter	preamble		command		e		P				end flag
Byte String (HEX)	A5	5A	57	00	e	00	p_l	p_l	p_h	p_h	B9	9B
Used setup value	None										
Affected setup value	None										
MCA state	STATE_READY										

MCA Query Commands:

Command name	**CMD_QUERY_POWER**										
Description	The command reads the power state.										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		0		0				end flag	
Byte String (HEX)	A5	5A	59	00	00	00	00	00	00	00	B9	9B
Results	see MMCA_QUERY_POWER function in DLL description.										

Command name	**CMD_QUERY_STATE**										
Description	The command reads the state.										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		0		0				end flag	
Byte String (HEX)	A5	5A	5A	00	00	00	00	00	00	00	B9	9B
Results	see MMCA_QUERY_STATE function in DLL description.										

Command name	**CMD_QUERY_SPECTRA**										
Description	The command reads the spectrum data. n: No. of 1. channel and buffer control (Bit 15 ... 12) c: Compress factor (1 ... 32)										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		n		C				end flag	
Byte String (HEX)	A5	5A	5B	00	n_l	n_h	c_l	c_h	00	00	B9	9B
Used setup value	channels, lld, uld										
Results	see MMCA_QUERY_SPECTRA function in DLL description.										

Command name	**CMD_QUERY_IMAGE**										
Description	The command read the MMCA spectrum image (120x256). n: No. of 1. Channel c: Compress factor (1 ... 32) v: Vertical full scale (see MCA_VFS in mca_comm.h)										
Command syntax											
Format	integer		integer		integer		integer		integer		integer	
Parameter	preamble		command		n		C		v		end flag	
Byte String (HEX)	A5	5A	5C	00	n_l	n_h	c_l	c_h	v_h	v_h	B9	9B
Used setup value	channels, lld, uld										
Results	see MMCA_QUERY_IMAGE function in DLL description.										

Command name	CMD_QUERY_USER_DATA										
Description	The command reads a 32 bit value from the MCA parameter memory. e: 0 … 63 parameter number										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		e		0				end flag
Byte String (HEX)	A5	5A	5E	00	e	00	00	00	00	00	B9	9B
Used setup value	None										
Result	see MMCA_QUERY_USER_DATA function in DLL description.										

Command name	CMD_SYSTEM_DATA										
Description	The command reads the special MCA data.										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		00		End				end flag
Byte String (HEX)	A5	5A	62	00	00	00	00	00	00	00	B9	9B
Used setup value											
Result	see MMCA_QUERY_SYSTEM_DATA function in DLL description.										

Command name	CMD_QUERY_CENTROID										
Description	The command reads the centroid of a ROI (beg<end, LLD<=beg, end<=ULD, end-beg<250).										
Command syntax											
Format	integer		integer		integer		long				integer
Parameter	preamble		command		beg		End				end flag
Byte String (HEX)	A5	5A	5F	00	beg$_l$	beg$_h$	end$_l$	end$_h$	00	00	B9	9B
Used setup value	channels, lld, uld										
Result	see MMCA_QUERY_CENTROID function in DLL description.										

Command name	**CMD_QUERY_ENRICHMENT**										
Description	The command read the integral of two ROIs and the real and dead time. pb: Peak$_1$ begin bb: Peak$_2$ begin pl: Peak$_1$ length (8 ... 250 channels) bl: Peak$_2$ length (8 ... 250 channels)										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	Preamble		command								end flag	
Byte String (HEX)	A5	5A	5D	00	pl	bl	pb$_l$	pb$_d$	bb$_l$	bb$_h$	B9	9B
Used setup value	-											
Result	see MMCA_QUERY_ENRICHMENT function in DLL description.											

Name	**MMCA_QUERY_UF6_INFO**										
Description	The command reads current dead time, real time the integral, begin and end of the UF6 ROIs										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	preamble		command		0		0				end flag	
Byte String (HEX)	A5	5A	66	00	00	00	00	00	00	00	B9	9B
Used setup value	-											
Result	See MMCA_QUERY_UF6_INFO function in DLL description.											

Name	**CMD_QUERY_UF6_ROIS**										
Description	The command reads begin and end of the UF6 ROIs										
Command syntax											
Format	integer		integer		integer		long				integer	
Parameter	Preamble		command		0		0				end flag	
Byte String (HEX)	A5	5A	65	00	00	00	00	00	00	00	B9	9B
Used setup value	-											
Result	see MMCA_QUERY_UF6_ROIS function in DLL description.											

APPENDIX 5 - Changing the MCA166-USB Firmware

Attention:

The order of changing the switches must be followed exactly, otherwise the µP will be destroyed!

The programming voltage (+12V) shall only be applied when the working voltage (+5 V) is switched on !!

Each MMCA must be programmed with it's corresponding HEX-file (e.g. MMCA #6 with Mca6.hex).

Programming the firmware can be done very easy by using the Firmware Wizard program. It is available from the Internet or from the supplied software CD.