

operating manual



rhea
spectrometer





Contents

| | | | | | |
|-----|------------------------------|---|-----|---|----|
| 1 | Introduction..... | 3 | 5 | Command set description | 9 |
| 1.1 | Rhea..... | 3 | 5.1 | Commands..... | 9 |
| 1.2 | Rhea highlights | 3 | 5.2 | Command structure..... | 9 |
| 1.3 | Standards..... | 4 | 5.3 | System commands..... | 10 |
| 2 | Interfaces..... | 5 | 5.4 | Configuration commands | 10 |
| 2.1 | USB interface | 5 | 5.5 | Measurement commands..... | 12 |
| 2.2 | Ethernet interface..... | 5 | 5.6 | User EEPROM commands | 13 |
| 2.3 | RS232 interface | 5 | 5.7 | Returned results | 16 |
| 2.4 | Trigger in/out..... | 6 | 6 | Measurement example..... | 17 |
| 2.5 | Power connections..... | 6 | 7 | Auto-ranging..... | 18 |
| 3 | Communications protocol..... | 7 | 7.1 | Introduction..... | 18 |
| 3.1 | USB..... | 7 | 7.2 | How auto-ranging works | 18 |
| 3.2 | RS232 | 7 | 7.3 | Auto-range parameters | 19 |
| 3.3 | Ethernet..... | 7 | 7.4 | Auto-range in practice | 19 |
| 4 | Device drivers..... | 8 | 7.5 | Programming Rhea for auto-ranging..... | 20 |
| 4.1 | USB..... | 8 | 7.6 | Auto-ranging and the filter wheel | 20 |
| 4.2 | RS232 | 8 | 7.7 | Auto-ranging recommendations | 20 |
| 4.3 | Ethernet..... | 8 | 8 | Rhea measurement results | 21 |
| | | | 9 | Operating modes & triggering | 21 |

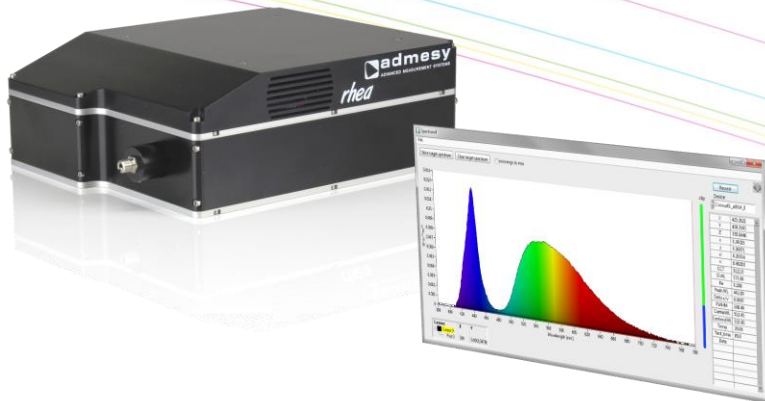


1 Introduction

1.1 Rhea

The Rhea series spectrometer offers a unique combination of ease of use and accurate measurement capabilities combined in a robust package. The Rhea utilizes a high-end cooled CCD detector for low noise and high dynamic range. The Neutral Density filter wheel extends this dynamic range and also adds a shutter function. An ideal device for measurements where ease of use, stability, performance and price are of the essence. All in all the perfect solution for hassle free integration in your product or process. The Rhea spectrometer can virtually cover any wavelength range in the 200-1100nm range. Standard versions (like UV, VIS and NIR) are available. Additionally, we also support a broad range of gratings for specific applications.

The Rhea series is available in a variety of optical fibre connected 5, 10 or 20mm collimating lenses or a 1cm² cosine corrector. For measurements of light sources Admesy offers a range of integrating spheres and other accessories. The M8 fixed position fibre connector has been developed to ensure that the optical fibre has a fixed and uniform optical connection for both calibration and customer application. Also, due to this fixed position measurement results are more stable. Additionally, Admesy also offers SMA connection.



1.2 Rhea highlights

- Various spectral ranges including UV, VIS and NIR within the 200-1100nm range
- Cooled high-end CCD detector, cooled to -10 degrees Celsius
- High optical throughput design
- Neutral density filter wheel for huge dynamic range
- Shutter function
- Low noise
- Auto-range function
- Wavelength calibrated
- Very low stray light
- Excellent linearity, internally compensated within 1%
- Dark current compensated, virtually zero over entire integration range
- USBTMC compliant, SCPI command set, high speed device
- USB, RS232, Ethernet connections and trigger in & out for ideal system integration
- Internal calculations for most common parameters, saving processing power in production environments
- Robust housing, optimized for mounting and protection in harsh production environments
- M8 or SMA fibre connection



1.3 Standards

The Rhea is compliant to the USBTMC standard and can be used in combination with external provided USBTMC compliant drivers. Currently it has been tested on Windows, Linux and Apple OSX using NI VISA (www.ni.com/visa) and using the open source drivers on Linux (i686, x86_64 and ARM).



2 Interfaces

2.1 USB interface

The USB B connector is used to connect the Rhea to a PC/Laptop. The Rhea uses the USBTMC class protocol and can therefore be used directly with third party provided VISA compliant libraries like NI-VISA. Rhea is externally powered

2.2 Ethernet interface

Ethernet of Rhea is 100Mbit and is 10Mbit/GigE compatible. Ethernet can be used in the same way as USB. All commands have the same format. Ethernet is preferred over USB in situations where the distance between device and a PC is more than 5 meters.

2.3 RS232 interface

RS232 is provided to connect the Rhea to any host that doesn't provide USB or Ethernet or for which no USBTMC drivers exist. Using RS232, the functions that generate a lot of data (spectrum) are still available, but the use of it is highly discouraged due to the low speed of RS232.

| Baud rate | Data bits | Parity | Stop bits | Flow control | Termination character |
|---------------------|-----------|--------|-----------|--------------|-----------------------|
| 115200 ¹ | 8 | None | 1 | None | LF='\\n' |

Table 1 RS232 port configuration.

¹ Baud rate can be changed.

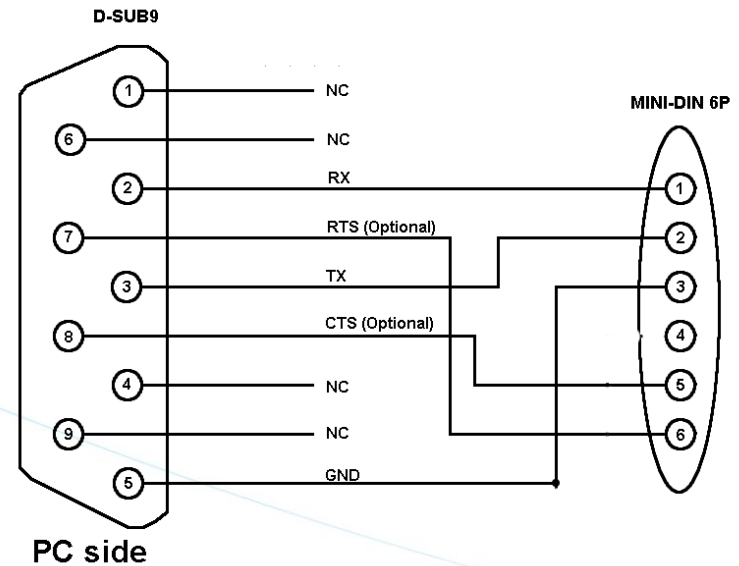


Fig 1 Rhea RS232 connection.

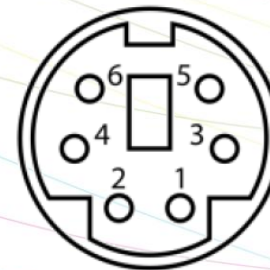


Fig 2 Rear side mini-DIN-6 pin connector.



2.4 Trigger in/out

The Rhea has two trigger connections. One trigger output and one trigger input. The connectors are SMA connectors. When triggering is enabled, the trigger output line will be set to a high level once the measurement has finished and the measurement result is available. It will stay at a high level until the next command is carried out, but has a minimum high level of $5\mu\text{s}$. A trigger will carry out the last send command and send the result to the host via the selected interface. The colorimeter main application allows external triggering in the data-logging tab. Supplied code examples show how to use this feature in an application. The trigger output line is used to indicate that the measurement is ready. Trigger signals should comply with the following timing (Fig 3).

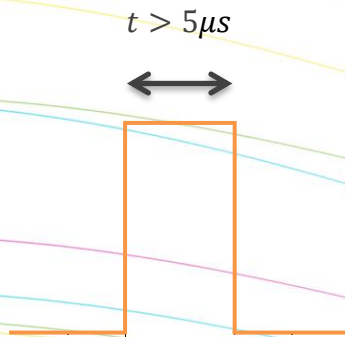


Fig 3 Trigger-in timing.

Trigger pulses arriving faster than the Rhea can measure will be ignored, but it may slowdown overall performance. Trigger pulses should not arrive faster than the measurement takes to complete. The best way is to use the trigger output to make sure measurement was finished.

The trigger out port provides a continuous high signal when the device is in use. Its signal turns low when the device is not executing any commands.

2.5 Power connections

The Rhea must be externally powered, by default we recommend to use the supplied power supply however using your own supply is also possible.

The unit shall be powered by a 15V DC voltage, reinforced separated from Mains, with a limited energy of $< 150\text{VA}$ and $< 8\text{A}$.

| Connection | Min. Voltage | Typ. voltage | Max. voltage | Max current |
|------------|--------------|--------------|--------------|-------------|
| DC powered | 14.50 V | 15.00 V | 15.50 V | 2000mA |

Table 2 Power supply levels.



3 Communications protocol

3.1 USB

The Rhea can be connected to any USB host and is a USBTMC compliant device which is a standard USB class. Device drivers for this class are available for most popular operating systems (also embedded). This makes the Rhea directly usable in popular programming languages like NI's Labview and Labwindows but also C++, Visual basic, C#, Java etc. The Rhea has two interfaces build in, which require a different device driver to be used.

- Rhea bootloader
(USB RAW device driver, Vendor ID: 0x23CF, Product ID 0x0105)
- Rhea01
(USBTMC device driver, Vendor ID: 0x23CF, Product ID 0x1040)

When the Rhea is connected to the host, it will start the Rhea firmware. As soon as the firmware is idle to receive commands, the Power LED goes to the ON state. The Admesy bootloader is a RAW USB device and in order to use this device in Windows, a driver must be installed which is supplied by Admesy. Besides upgrading to new firmware, it is also allowed to downgrade firmware in case this is required. Note that older firmware also may require the use of older software libraries and/or executable versions of software. The Rhea is USBTMC compliant and can be used with libraries that contain a USBTMC compliant driver like NI-VISA. The Rhea is a USB 2.0 high speed device. In case a USB host is detected, it is assumed that the Rhea operates only via USB.

3.2 RS232

Rhea commands are equal for all interfaces. Note that for high speed transfers it is best to use USB. When RS232 will be used, the device mode should be set to RS232. This is done via software and stored in the device EEPROM memory.

3.3 Ethernet

Rhea commands are equal for all interfaces. When the Ethernet connection will be used, the device mode should be set to ETH (Ethernet). This is done via the Iliad application software: Device→Start-up settings→Rhea and stored in the device EEPROM memory. In the start-up menu the IP address, gateway and network mask can also be changed.



4 Device drivers

4.1 USB

The following table shows an overview of USB support on various operating systems.

| OS | NI-VISA | Libusb | Native kernel | Agilent USBTMC |
|-------------------------|---------------|-------------------------|---------------|-------------------------|
| Windows XP ² | ▪ | ▪ | Not available | Not tested ¹ |
| Windows VISTA | ▪ | ▪ | Not available | Not tested ¹ |
| Windows 7 | ▪ | ▪ | Not available | Not tested ¹ |
| Windows 8(.1) | ▪ | Not tested ¹ | Not available | Not tested ¹ |
| Windows 10 | ▪ | Not tested ¹ | Not available | Not tested ¹ |
| Windows CE | ▪ | Not tested ¹ | Not available | Not tested ¹ |
| Apple PPC OSX | ▪ | Not tested ¹ | Not available | Unknown |
| Apple Intel OSX | ▪ | Not tested ¹ | Not available | Unknown |
| Linux (32bit) i386 | ▪ | ▪ | ▪ | ▪ |
| Linux (64bit) i386 | ▪ | ▪ | ▪ | ▪ |
| Linux ARM | Not available | ▪ | ▪ | ▪ |
| Linux other | Not available | ▪ | ▪ | ▪ |

Table 3 Supported operating systems.

¹ Not tested: Available, but not tested by Admesy, ² Native Kernel: Driver included with OS.

² Windows XP SP3 is supported: Windows official support has ended as of April 8 2014

Admesy supports all tested platforms but does not provide standard applications on all platforms. The matrix is provided to show the possible platforms for software development. Admesy does however provide software examples for most of the tested platforms. Most of these examples can be found on our support web page.

4.2 RS232

When no USB driver is available or the host system does not provide USB, RS232 can be used as it does not require additional drivers for the Rhea.

4.3 Ethernet

No special drivers are needed for Ethernet operation. The Rhea can be directly addressed through a TCP/IP socket on port 10000. In case firewalls are used, the TCP/IP port needs to be opened.



5 Command set description

5.1 Commands

The functions of the Rhea can be best described via the following categories.

- System commands
- Configuration commands
- Measurement commands
- User EEPROM commands

The Rhea uses SCPI like commands for control and measurement. These are ASCII based commands and follow specific rules regarding syntax. Although the Rhea uses SCPI like commands, they deviate from the SCPI standard.

5.2 Command structure

Every command starts with a colon “:”, which identifies the root of the command tree. Each further keyword is also separated by a colon. In case parameters need to be specified, the last keyword and parameters are separated by a single space character. In case more than one parameter needs to be specified, the parameters need to be separated by a comma.

The command tables show commands in long and short format. The short format is specified by upper case characters. It is allowed to use long and short format or a mixed format. Optional keywords are shown between brackets [...]. Commands are not case sensitive, so it is allowed to use both or a mix of upper and lower case. The command structure is valid for all communication interfaces of the Rhea. It is recommended to terminate a command by a newline character “\n”.

| Command table | Valid command syntax examples | Notes |
|--------------------------|--|--|
| :SENSe: INT 50000 | :sens:int 500000 :sense:int 500000 :SENS:INT 500000 :SENSE:INT 500000 | Sets the integration time of the Rhea |
| :MEASure:XYZ | :measure:XYZ :measure:xyz :meas:XYZ :MEASure:XYZ | The measure commands use the averaging and integration time parameters that have to be set first |

Table 4 Example commands.



5.3 System commands

The following commands can be used to control the Rhea or read back information.

| Command syntax | Parameters | Purpose |
|----------------------------|------------|----------------------------------|
| :*CLS | None | Clear status |
| :*IDN? | None | Identification query |
| :*RST | None | Reset Command |
| :*STB? | None | Read Status Byte query |
| :*TST | None | Self-test query |
| :*FWD? | None | Firmware date query |
| :*FWT? | None | Firmware time query |
| :SYSTem:VERSion? | None | Get system version information |
| :SYSTem:ERRor? | None | Retrieve the last occurred error |
| :SYSTem:ERRor:NEXT? | None | Retrieve previous errors |

Table 5 System commands.

The Status byte can be used to retrieve information about the status of a command or the system. Return values of the status command can be seen in the table below.

| Code | Description |
|----------|---|
| 0 | System is idle |
| 1 | Data is available |
| 2 | Command processed |
| 4 | Data in buffer (should not occur) |
| 8 | An error occurred Use “:SYSTem:ERRor?” to get the exact error that occurred |

Table 6 Status commands.

*Note: The status and error commands should not be used in standard programs. They are not necessary when programming has been done correctly. They are available in case a problem occurs during development. The :*RST command is also not necessary in normal programs. It is advised to use the :SENSe and :CONF commands in case the behaviour of the Rhea needs to be changed to its default settings.*

5.4 Configuration commands

Configuration commands are used to set parameters of the Rhea that are used by the measurement functions. The settings are used globally by other measurement functions. The selected white standard is used for dominant wavelength calculation. The integration time setting is specified in μ s. Results from the Rhea include a clip and noise indication which indicate, whether the measured light is too bright (clip) or too low (noise). When clipping is detected, the resulting colour will not be correct and a lower integration time should be chosen. When noise is detected, a larger integration time should be chosen, see table 7 on next page.



| Command syntax | Parameters | Range | Purpose |
|---------------------------|----------------------|---|---|
| :SENSe:INT | Int time | 4800 – 3600000000 | Set integration time (µs) of the spectrometer |
| :SENSe:INT? | None | | Returns the current integration time setting of the spectrometer |
| :SENSe:INTERPOL | Interpolation method | 0 = linear 1 = cosine 2 = cubic 3 = Catmull-Rom 4 = Hermite | Set the spectrum interpolation method |
| :SENSe:INTERPOL? | None | | Returns the current interpolation method |
| :SENSe:SP:AVERAge | Averaging | 1 – 200 | Set the averaging for the spectrometer |
| :SENSe:SP:AVERAGE? | None | | Returns the averaging setting for the spectrometer |
| :SENSe:RESolution | Resolution | 0 = 0.5nm 1 = 1nm 2 = 2.5nm 3 = 5nm 4 = 10nm | Set the spectrum output resolution. Interpolation is done using the earlier selected method |

| Command syntax | Parameters | Range | Purpose |
|--------------------------|--------------------|--------------------------------------|---|
| :SENSe:SP:SBW | Calibration matrix | “off” or “user” | Set calibration matrix for the spectrometer |
| :SENSe:CAL | User cal | 0/1 | Activate or deactivate user spectrum calibration |
| :SENSe:MODE | Mode | 0 = USB 1 = Ethernet 2 = RS232 | Set operating mode (note, this one is not stored in EEPROM). This is useful for triggering only |
| :SENSe:MODE? | None | | Returns the current setting |
| :SENSe:TRIG | Trigger | 0 – 1 | Set trigger mode (note this is note stored in EEPROM) |
| :SENSe:TRIG? | None | | Returns the current setting |
| :SENSe:TRIGDELAY | Delay | 0 - 30000000 | Configures a delay before triggering in microseconds |
| :SENSe:TRIGDELAY? | None | | Reads the configured delay before triggering |
| :SENSe:AUTORANGE | Auto-range | 0 – 1 | Set auto-range mode |
| :SENSe:AUTORANGE? | None | | Returns current setting |
| :SENSe:FILTER | Filter position | 0 – 5 | Set filter wheel position |
| :SENSe:FILTER? | None | | Returns filter wheel position |
| :SENSe:SHUT | Filter position | 0 – 1 | Filter wheel closed or open (last position) |

Table 7 Sense and configuration commands.



5.5 Measurement commands

Table 8 and 9 show the measurement commands of the Rhea. The MEASure commands measure the requested values using the set averaging and integration time and returns the result in ASCII format. The SAMPlE commands measure the requested parameters using a sample count and delay time and return an array of data. The array contains single float array of data.

| Command syntax | Parameters | Range | Purpose |
|-----------------------|------------|---------|--|
| :MEASure:XYZ | None | | Measure XYZ |
| :MEASure:Yxy | None | | Measure Y and x,y colour point |
| :MEASure:Yuv | None | | Measure Y and u',v' colour point |
| :MEASure:SPECTrum | None | | Measure the spectrum The output is in the selected resolution and either uses factory calibration or user calibration |
| :MEASure:RAWSPECTrum | None | | Measures the raw spectrum, suitable for user calibration |
| :MEASure:TEMP | None | | Measures the sensor temperature |
| :GET:WAVElengths | Pixels | 0 | Get the wavelength axis according to the set resolution |
| :SAMPlE:SPECTrum | Samples | 1 – 500 | Performs High speed sampling of a spectrum |
| :SAMPlE:SPECTrum:READ | Index | 1 – 500 | Reads the sampled spectrum at index. |

Table 8 Measurement commands.

| Special measurement commands | Parameters | Range | Purpose |
|------------------------------|------------------------|---------|--|
| :MEASure:CRI | Number of test samples | 8 or 15 | Measure the CRI of a light source, see comment |
| :MEASure:PEAKWL | None | | Measures the peak wavelength |
| :MEASure:DWL | None | | Measures the Dominant wavelength and purity |
| :MEASure:ALL | CRI | 0/1 | Measures all parameters with or without CRI |
| :MEASure:LEDCHAR | CRI | 0/1 | Measures all parameters with or without CRI |
| :MEASure:CCT | None | | Measures the correlated colour temperature |

Table 9 Measurement commands.

Note: CRI calculation takes about 11ms additional time to measure a spectrum. Using the Admesy colour DLL, this can be achieved faster.



5.6 User EEPROM commands

Table 10 shows the commands which can be used to store values in the user EEPROM space. It is advised to reboot the Rhea after writing new values to the EEPROM.

| Command syntax | Parameters | Range | Purpose |
|--|------------|---|---|
| :EEPROM: STARTUP:READ | None | | Copies start up conditions from EEPROM to internal variables. Values can then be read using :SENSe:AVERage? for example |
| :EEPROM: STARTUP:WRITE | None | | Copies internal variables to EEPROM |
| :EEPROM: CONFigure:MODE | Mode | 0 = USB 1 = Ethernet 2 = RS232 | Configures the mode |
| :EEPROM: CONFigure: MODE? | None | | Reads the mode at set in the EEPROM |
| :EEPROM: CONFigure: BAUDRATE | Baudrate | 0 = 9600 1 = 19200 2 = 38400 3 = 57600 4 = 115200 5 = 230400 | Configures the RS232 baudrate in the EEPROM |
| :EEPROM: CONFigure: BAUDRATE? | None | | Read the current RS232 baudrate from EEPROM |
| :EEPROM: CONFigure:TRIG | Trigger | 0 = off 1 = on | Configures triggering |
| :EEPROM: CONFigure:TRIG? | None | | Reads from EEPROM if triggering is used. |
| :EEPROM: CONFigure:TRIG DELAY | Delay | 0 - 30000000 | Configures a delay before triggering in microseconds |

| Command syntax | Parameters | Range | Purpose |
|---|---------------|--|---|
| :EEPROM: CONFigure:TRIG DELAY? | None | | Reads the configured delay before triggering |
| :EEPROM: CONFigure:IP | IP address | Example: 192.168.0.100 | Configures the Ethernet IP address |
| :EEPROM: CONFigure:IP? | None | | Reads the current IP address from EEPROM |
| :EEPROM: CONFigure:GW | Gateway | Example: 192.168.0.1 | Configures the Ethernet gateway |
| :EEPROM: CONFigure:GW? | None | | Reads the current gateway address from EEPROM |
| :EEPROM: CONFigure:MASK | Ethernet mask | Example: 255.255.255. 0 | Configures the Ethernet mask |
| :EEPROM: CONFigure:MASK ? | None | | Reads the current ethernet mask from EEPROM |
| :EEPROM: CONFigure:RES | Resolution | 0 = 0.5nm 1 = 1nm 2 = 2.5nm 3 = 5nm 4 = 10nm | Configures the resolution in EEPROM |
| :EEPROM: CONFigure:RES? | None | | Reads the resolution from EEPROM |
| :EEPROM: CONFigure:SPINT | Int time | 1000 – 500000 | Configures default spectrometer integration time |
| :EEPROM: CONFigure: SPINT? | None | | Reads the default spectrometer integration time from EEPROM |
| :EEPROM: CONFigure: SPAVG | Averaging | 1 – 4000 | Configures default spectrum averaging |



| Command syntax | Parameters | Range | Purpose |
|---|-------------------------|---|---|
| :EEPROM: CONFigure: SPA VG? | None | | Reads the default spectrum averaging from EEPROM |
| :EEPROM: CONFigure: AUTORANGE | Auto-range | 0 = off 1 = on | Configures auto-ranging |
| :EEPROM: CONFigure: AUTORANGE? | None | | Reads the current auto-range setting from EEPROM |
| :EEPROM: CONFigure: INTERPOL | Interpolation | 0 = linear 1 = cosine 2 = cubic 3 = Catmull-Rom 4 = Hermite | Configures the interpolation method to be used |
| :EEPROM: CONFigure: INTERPOL? | None | | Reads the current interpolation method from EEPROM |
| :EEPROM: CONFigure: SPSBW | Calibration matrix | “off” or “user” | Configures the default spectrometer calibration matrix to be used |
| :EEPROM: CONFigure: SPSBW? | None | | Reads the default spectrometer calibration matrix from EEPROM |
| :EEPROM: SPSBW:WRITE | Row, Column Value | 0 – 2 0 – 2 single float | Writes the values of the spectrum matrix to Rhea |
| :EEPROM: SPSBW:READ | Row Column | 0 – 2 0 – 2 | Reads the values of the spectrum matrix from Rhea |
| :EEPROM:READ: USERCAL | None | | Reads the user calibration values from EEPROM to memory |

| Command syntax | Parameters | Range | Purpose |
|---|-------------------------------|--|---|
| :EEPROM:WRITE: USERCAL | None | | Write the user calibration values from memory to EEPROM. This step fixes all values for a next restart of the instrument |
| :EEPROM: READ:ID | None | | Reads the device ID. This is useful for use with Medusa or any other system with multiple units attached. The deviceID can be used to link to a mechanical position of the instrument, which can then be selected by software for a specific task |
| :EEPROM:WRITE: ID | ID number | 1 – 255 | Write the device ID to EEPROM |
| :EEPROM:ABS: WRITE | Index, Parameter, value | | Write the user absolute spectrum parameters |
| :EEPROM:ABS: READ | Index, parameter | | Reads the user absolute spectrum parameters |
| :EEPROM:ABS: FIX | None | | Fixes the user absolute spectrum parameters in the EEPROM |
| :EEPROM: CONFigure: WHITE | White point | “D65”, “A”, etc | Used for Dominant wavelength |
| :EEPROM: CONFigure: WHITE? | None | | Gets the currently configured white point |
| :EEPROM: READ:UNIT | None | 0 Radiant power 1 Radiance 2 Irradiance 3 Radiant intensity | Reads the calibrated unit as set from factory |



| Command syntax | Parameters | Range | Purpose |
|---------------------------------------|----------------------|-------------------------|--|
| :EEPROM:CONFigure:AUTO:FREQ | Frequency (Hz) | 1 – 255 | Auto-range parameter: frame frequency of the source (display) to measured |
| :EEPROM:CONFigure:AUTO:FREQ? | None | | Reads the set frequency |
| :EEPROM:CONFigure:AUTO:ADJMIN | Adjmin (%) | 1 – 100 | Auto-range parameter: The minimum level to adjust to |
| :EEPROM:CONFigure:AUTO:ADJMIN? | None | | Reads the set Adjmin for the spectrometer |
| :EEPROM:CONFigure:AUTO:MAXINT | Max int | 1000 – 1000000 | Maximum spectrometer integration time. If time is too short, measuring dark level may be difficult |
| :EEPROM:CONFigure:AUTO:MAXINT? | None | | Query the maximum spectrometer integration time |
| :EEPROM:CONFigure:USERABS | Absolute calibration | 0 = factory 1 = user | Configures the absolute calibration mode to be used. |
| :EEPROM:CONFigure:USERABS? | None | | Reads the absolute calibration mode from EEPROM |
| :EEPROM:READ:SBWNAME | Matrix number | | Reads the name of an XYZ matrix |
| :EEPROM:READ:SN | None | | Read serial number |

Table 10 User EEPROM commands.



5.7 Returned results

:MEASure command return their result in ASCII formatted floating point as shown below.

$$(X,Y,Z,clip,noise) \rightarrow \%f,\%f,\%f,\%d,\%d \backslash n$$

X,Y,Z can be substituted for Y,u,v or other colour spaces. :SAMPlE command return all measurement data also in ASCII format., except the :Sample:Y function. The first three values indicate the delta time between samples and the clip and noise values.

| MEAS command | |
|--------------|------|
| dt | %f\n |
| Clip | %f\n |
| Noise | %f\n |
| Value1 | %f\n |
| Value2 | %f\n |
| Value3 | %f\n |

Table 11 Return result MEAS command.

Exceptions to the above is the :MEASure:TEMP command which only gives back the temperature.



6 Measurement example

The Rhea uses default settings when the device is started. These can be programmed by the end user so that the device starts with the same settings each time it is connected.

Although it's possible to program all Rhea devices in production environment to start with equal settings, it is recommended to set the averaging, integration time and SBW values in the initialization routine of the host software.

A typical measurement example of XYZ would include the following commands as shown on the right.

Action may be performed in a loop

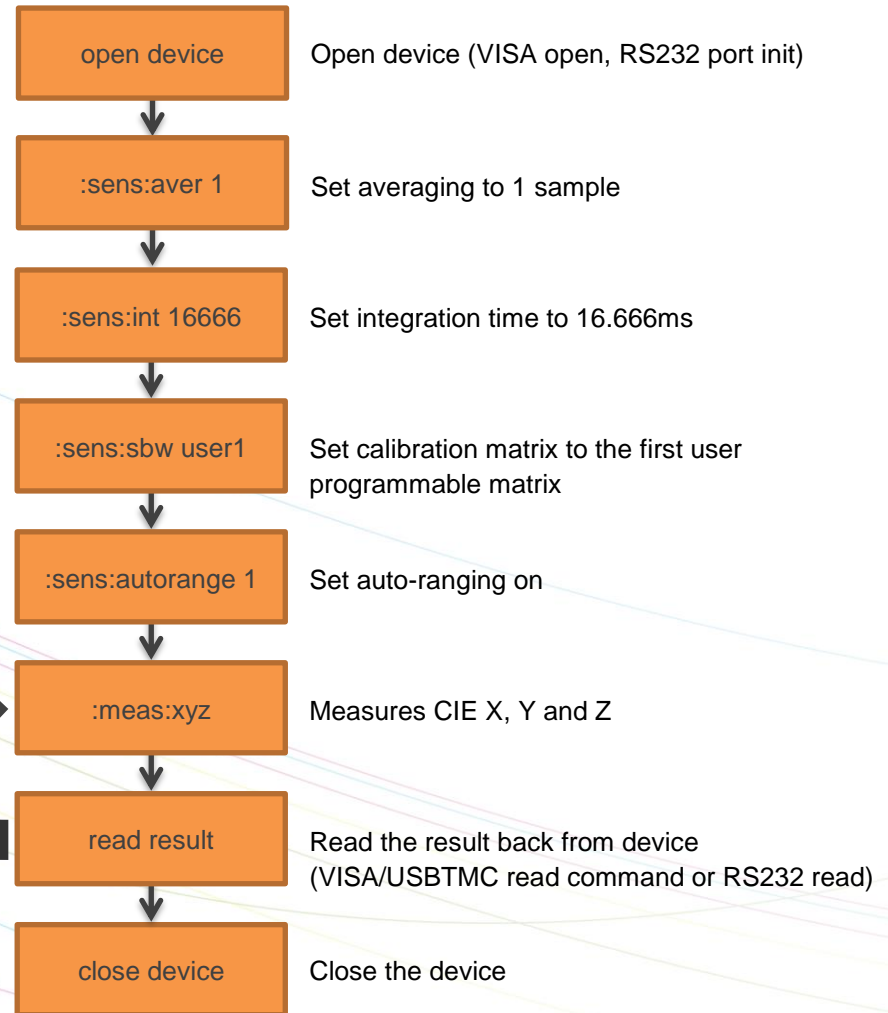


Fig 4 Measurement example.



7 Auto-ranging

7.1 Introduction

The Rhea includes an auto-range function for the spectrometer sensor. This function is useful in case the measured object shows an unknown luminance value. In this case, the Rhea will try to find the optimum setting which is a trade-off between speed and the stability of the instrument. The auto-ranging function can also be fine-tuned to reach better stability levels by setting a few parameters. Auto-ranging can be controlled by 3 parameters.

- Frequency : supposed to be frame frequency of the source (display) that is measured
- Adjmin : The minimum level to adjust to

Adjmin can vary between 1 and 100. Reasonable results will be achieved when set to 40 for the spectrometer regarding the speed and the stability. If high stability is needed, this setting must be increased (but measurements will become slower). If a faster measurement is needed and the stability level may be lowered a little, the value of adjmin can be decreased. When the Rhea measures in auto-range mode, it can happen that the found integration time is very low. For example when measuring white it may be just 7ms. When this happens, the Rhea will automatically increase the averaging so that the total measurement time equals $\geq (1/\text{frequency}) \cdot \text{averaging}$.

7.2 How auto-ranging works

The auto-ranging works, by first setting a default integration time. If this already meets the criteria for a good measurement, the measurement will be done using that integration time. It should be clear that this is the fastest because no adjustment will be done. In case the measured result is either too low or too high (clip), then the Rhea will adjust to a better level. The following graph shows how levels inside the Rhea work.

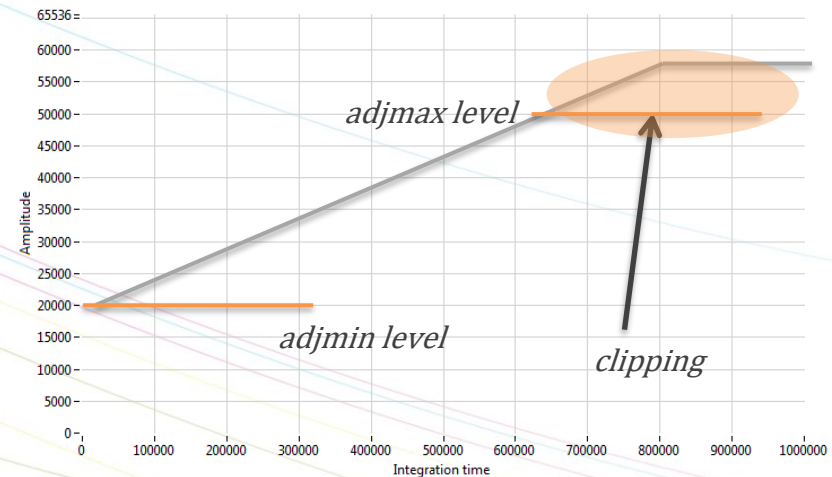


Fig 5 Auto-range levels

When auto-ranging is set, the Rhea will accept any level between “Adjmin” and “Adjmax” as a good signal. The Adjmin level can be set by the user through a software command, allowing some fine tuning of the stability and the speed of the instrument.



7.3 Auto-range parameters

Auto-range controls the following parameters:

- **Frequency:** This setting must be set to the frame frequency of the measured sample. The range is 1 to 255 Hz.
- **Adjmin:** The adjmin parameter is very important. The higher this level is set, the more stable the measurement becomes, but it also slows down the measurement if set too high. That means, there is a trade-off between speed and stability. It mainly affects the lower grey scale levels. If these levels appear not stable, adjmin needs to be increased. Adjmin is set from 1-100, which means 1-100% of the Adjmax level (Adjmax is fixed by Admesy).
- **Maximum integration time:** Although not really part of the auto-ranging algorithm, this parameter is used when the auto-ranging result exceeds this setting of integration time. When the grey scales are low (grey scale 0 for example), the maximum integration time setting can be used to allow only a maximum measurement time. Of course this affects stability (longer measurement is always more stable), but again this is a trade-off for production environments to save on test time.
- **Averaging:** The averaging will only be used if the auto-range algorithm does calculate a averaging that is less than the set averaging for the spectrometer sensor.

7.4 Auto-range in practice

When auto-ranging is set, the measurement settings (integration time, gain and averaging) are automatically adjusted. An initial measurement is done to determine the final settings of averaging and integration time, based on the settings of frequency and adjmin.

The set integration time (1/frequency as set in the EEprom auto-range frequency parameter) of the instrument will always be used as the first value to try. If this setting results in a value between adjmin and adjmax, no further actions are necessary and the speed is optimal.

When measuring a display we can measure random patterns, but mostly known patterns are measured. Setting an integration time that is nearly right, 1/frequency (Hz) is preferred, as it optimizes the speed for adjusting the auto-ranging settings and thus provides optimal results.

Example: a 23" TFT display (standard desktop LCD) with white at 290 cd/m² and a frame rate of 60Hz. Settings for auto-ranging are:

- Frequency = 60
- Adjmin SP = 10
- Max int SP = 300000

These settings will provide a stable measurement on white and black. Auto-range algorithm will adjust to measure black 0.3cm/m² with 3000ms integration time, resulting in ~5600ms tact time for the spectrometer sensor.



7.5 Programming Rhea for auto-ranging

The Rhea has start-up settings. This means that when the Rhea is connected to a PC or pattern generator, it only needs the “:meas:Yxy” command to get luminance and colour data. All other settings can be saved as preset and are loaded when the instrument is started. However, it requires the right settings to be set in advance. This can be done though the Admesy Iliad application and select Device→Start-up settings→Rhea. These setting may be optimized per display type. There are a few categories we can define. Those are shown in the next chapters.

- **Display without PWM:** This can be measured using almost any settings, but the settings may be optimized for high speed. This means, we can use a short integration time for bright images (white for example) and decreased the maximum integration of black to a level that gives stable results and is still fast. In this case the frequency setting is not critical.
- **Display with PWM:** Since the luminance will show PWM, a measurement must at least cover 1 full frame. Taking more frames will result in better stability. In this case the frequency setting is apparently more critical to get a good stability.

7.6 Auto-ranging and the filter wheel

When auto-range is enabled the Rhea will choose by itself the appropriate ND filter of the filter wheel.

7.7 Auto-ranging recommendations

It is recommended to apply auto-ranging in case process variation is high or in case various grey scales are measured. In cases where luminance setting is known, it may be desirable to turn off auto-ranging to save additional measurement / test time.

Regarding the grey scales, like a gamma measurement is performed, speeds up the measurement significantly, when using auto-ranging in combination with the feedback function.

There are a lot of ways to fine tune the Rhea for each application. Take this document into consideration and apply the comments correctly. By assuming that any default setting will do, it will lead to a non-optimal measurement results. Any measurement is application dependent. One should consider that auto-range settings exist for the sole purpose of optimizing each application.



8 Rhea measurement results

The Rhea is a spectrometer that provides direct calibrated output. As from many other mini spectrometers the Rhea differs in great processing power. Spectrum conversion to XYZ is done inside the Rhea and so are all colour space conversions, including CRI calculation. Apart from this internal calculation, it is also possible to receive the spectral data and do the processing on the host PC. This can be done by Admesy's colour DLL. The calculations performed inside the Rhea are based on the same formula's as the DLL. Calculations done inside Rhea are performed using single precision floats and the DLL uses double precision. For formula's please check the user manual of the Admesy colour DLL.

9 Operating modes & triggering

The operating mode only matters in trigger mode. The mode is used to send back results by USB, Ethernet or RS232. The modes of the Rhea are:

- USB mode
- Ethernet mode
- RS232 mode

A trigger activates only one command, for example “:meas:xyz”. This command needs to be set first by the host. After a trigger is received, the command will execute and the result will be presented on the selected interface as shown above. The Rhea responds to a rising edge of the trigger signal.



Admesy B.V.
Sleestraat 3
6014 CA Ittervoort
The Netherlands

T +31 (0)475 600 232
F +31 (0)475 600 316

www.admesy.com
info@admesy.com

The material in this document is subject to change. No rights can be derived from the content of this document. All rights reserved. No part of this document may be reproduced, stored in a database or retrieval system, or published in any form or way, electronically, mechanically, by print, photo print, microfilm or any other means without prior written permission from the publisher.