MEphisto Scope 1

Command-Interpreter

Firmware-Version 3.10
21.01.2010

MEILHAUS ELECTRONIC

© Meilhaus Electronic GmbH

CONTENTS

Introduction

All communication is done using 32 bit data words. Shorter data types must be padded with "null" bytes. All return values must be read back from the device immediately. If this is not done, data will be dropped word by word or block by block, depending on the mode of operation. All settings are checked and rounded to the nearest value that is realisable by the hardware of the device. This value is returned as response.

The time needed for execution can vary slightly depending on the firmware version, operation mode, and parameters being used.

The model UM202 is designed to run alongside a PC. All data will be transmitted through the USB at the moment they become available.

The variant UM203 offers the same functionality as the UM202. Besides this it can do acquisitions without a PC. In this offline mode the configuration is loaded that was last done online. After this the device awaits insertion of a SD card. After initialisation of the card is completed the device runs the measurement and saves the data on the card.

Commands

The command interpreter has been changed completely in firmware version 2.00. The commands used in firmware version 1.xx are no longer available.

Overview

Function	Command	Arguments direction	Data type	meaning
SetupWrite	*SWr	Host → Dev.	Float	amplitude channel 0
			Float	amplitude channel 1
			Float	offset channel 0
			Float	offset channel 1
			Float	timebase
			Float	memory depth
			Float	trigger point (1-99%)
			ULong	trigger channel
			ULong	trigger type
			Float	upper trigger level
			Float	lower trigger level
			ULong	GPIO data
			ULong	GPIO direction
		Dev. → Host	Float	amplitude channel 0
			Float	amplitude channel 1
			Float	offset channel 0
			Float	offset channel 1
			Float	zero point correction channel 0
			Float	zero point correction channel 1
			Float	timebase
			Float	memory depth
			Float	trigger point (1-99%)
			ULong	trigger channel
			ULong	trigger type
			Float	upper trigger level
			Float	lower trigger level
			ULong	GPIO data
			ULong	GPIO direction
SetupRead	*SRd	Host → Dev.	-	-
		Dev. → Host	Float	amplitude channel 0
			Float	amplitude channel 1
			Float	offset channel 0
			Float	offset channel 1
			Float	zero point correction channel 0
			Float	zero point correction channel 1
			Float	timebase
			Float	memory depth
			Float	trigger point (1-99%)
			ULong	trigger channel
			ULong	trigger type
			Float	upper trigger level
			Float	lower trigger level
			ULong	GPIO data

Function	Command	Arguments direction	Data type	meaning
			ULong	GPIO direction
SetAmplitude	*SAm	Host → Dev.	ULong Float	selected channel amplitude
		Dev. → Host	Float Float Float	amplitude offset zero point correction
SetOffset	*SOf	Host → Dev.	ULong Float	selected channel offset
		Dev. → Host	Float Float Float	amplitude offset zero point correction
SetTimebase	*STm	Host → Dev.	Float	sampling time
		Dev. → Host	Float	set time
SetMemory	*SMe	Host → Dev.	Float Float	memory depth trigger point (0-100%)
		Dev. → Host	s.o.	s.o.
SetTrigger	*STr	Host → Dev.	ULong ULong Float Float	trigger channel trigger type upper trigger level lower trigger level
		Dev. → Host	Float Float	upper trigger level lower trigger level
SetMode	*SMd	Host → Dev. Dev. → Host	ULong ULong	desired mode mode realised
SetRealtimeClock	*SRt	Host → Dev. Host → Dev.	ULong ULong	packed time of day packed date
		Dev. → Host	-	
Run (start acquisition)	*RUN	Host → Dev. Dev. → Host	- varying	 The data type and the number of data items depend on the mode selected by the "*SMm" command. See individual modes for details.
Break	ZZZZ	Host → Dev.	-	sub command, used only to cancel a running measurement process (e.g. if the trigger point is not reached). This command is only valid during a measurement process.
		Dev. → Host	-	
GPIOWrite	*IOW	Host → Dev.	ULong ULong	24 bit data 24 bit direction information
		Dev. → Host	ULong ULong	24 bit data 24 bit direction information
GPIORead	*IOR	Host → Dev.	-	
		Dev. → Host	ULong ULong	24 bit data 24 bit direction information
IEEE 488.2 - Inquiry (Inquiry)	*IDN?	Host → Dev.	Byte	unlimited number of characters (e.g. CR / LF)
		Dev. → Host	String	product ID string+CR+LF, 32 characters
IEEE 488.2 – Reset (Restart)	*RST	Host → Dev.	Byte	unlimited number of characters (e.g. CR / LF)
		Dev. → Host	String	"ok"+CR+LF, 4 characters
Firmware Update	*UPD	Host → Dev.	diverse	update function, for Meilhaus Electronic internal use only
enter calibration mode	*CAL	Host → Dev.	diverse	calibration mode, for Meilhaus Electronic internal use only all existing calibration parameters will be lost!

SetupWrite and SetupRead

The "SetupRead" function returns all settings for the current mode of operation. All device return values should be read back immediately after a mode change to check the values the device will use.

The "SetupWrite" function takes all the values described above as input parameters except the two correction factors OffstError*X*. The values are checked by the device and the values actually being set are returned. The output is the same as the "SetupRead" function.

The return values are:

- Amplitude0, Amplitude1

  maximum range of the input signal, centred around the offset voltage
  The voltage can be set in a 1-2-5 range between 200mV and 20V. Other values are matched as close as possible. A change in the amplitude can cause a change in the offset, if the maximum voltage of the selected range is exceeded. Example: If the offset is 5V but the new voltage range is only 2V, the offset value is limited to 2V.

For all digital modes, the value depends on the voltage of a logical “1”. Without an external converter, the value is 5V.
After reset, the value is 20V for all analogue modes. The value for digital modes depends on the level of a logical “1”. If no external converter is used, the value is 5V. If no mode has been selected, the value is always 0V.

- Offset0, Offset1
  centre of the range
  This value can be set within the limits of ±*Amplitude*/2 in 4096 steps. For the maximum range of ±10V the offset is always 0V.
  After reset, the value is 0V for all analogue modes. The value for digital modes depends on the level of a logical “1”. If no external converter is used, it is 2.5V. If no mode has been selected, the value is always 0V.
- OffsetError0, OffsetError1
  This is the correction value for analogue modes.
  This value is a calibration parameter determined by factory individually for each device at 25°C. It must be subtracted after the voltage value has been calculated from the raw data values. This parameter can only be read. The “SetupWrite” function does not provide access to this parameter.
- SamplingTime
  This is the time between two measurements
  In the oscilloscope, logic analyser and data logger modes, the value is the delay between two samples. The value can be set between 1µs/S and 2.5s/S in oscilloscope mode and between 10µs and 2.5s for the logic analyser mode. In the data logger modes the value is limited to 10µs as well. For values below 10ms/S only one acquisition is done. The values can be set with a resolution of 1µs. For values above 10ms/S an averaging algorithm is used. Above this time, the values are set in steps of 10ms. The signal is oversampled with a rate of 10ms.
  If no mode is selected, the value is always 0. After a reset, the value is set to the minimum value for all other modes of operation.
- MemoryDepth
  This is the amount of memory used in the oscilloscope and logic analyser modes.
  The minimum depth is 100 samples per channel. It can be set in the 1-2-5 range. The maximum memory does not fit this grid. In oscilloscope mode it is 131000 samples, and in logic analyser mode it is 262000 samples.
  The measurement time is calculated as
  AcquisitionTime = SamplingTime • MemoryDepth
  After reset, the value for the oscilloscope and logic analyser mode is 1000. All other modes have a fixed value of 1. If no mode is selected, the value is 0.
- TriggerPoint
  This is the position of the trigger point within the acquisition time.
  The trigger point is set as percentage of the sample memory. The value range depends on the firmware version. The values 0% and 100% are not possible in oscilloscope and logic analyser modes. A small setup and trailing time has to be tolerated.

In data logger mode, all values will be rounded towards 0% or 100%. Because the sample memory is not used in this mode, the values represent either pre or post trigger.
In volt meter mode or with no mode selected, the value is always 0%.
After a reset, the value is 50% for the oscilloscope and logic analyser modes, and 0% for the data logger mode.

- TriggerChannel
  This value can be interpreted in 2 ways, depending on the mode of operation. In oscilloscope mode, the value can only be a "1" or "0". Only the lowest-order byte (0) is used.
  After a reset, the trigger channel value is 0 for the oscilloscope and data logger modes.
  For the logic analyser mode, the parameter is determined using the comparison bit pattern. For more information, please refer to the table in the chapter of digital trigger types. Bytes 1 to 3 are used in this mode.
  After a reset, the trigger bit pattern for the logic analyser and digital data logger mode are set to 0, so the trigger is not active. If no mode is selected, the value is fixed to 0.
- TriggerType
  The trigger type is represented by an ASCII character that must be filled up to build a 32 bit word.
  Different numbers of modes are available depending on the firmware version. The modes currently available are listed in the tables in their respective chapters. After a reset, the value is set to "M" (manual) for all modes which offer a trigger. All other modes always have a value NUL.
- TriggerLevelUp
  This is the voltage value for trigger types T, t, W, w, E, e, D and d.
  For a window trigger, this is the upper boundary. For all other modes, this is the only value. Types X, x, and M ignore this value. In logic analyser mode, this value is ignored as well.
  After reset, the value is set to 0. In modes where no trigger is available, in digital modes, or if no mode is set, the value is always 0.
- TriggerLevelDown
  This is the voltage value for trigger types W and w. This is the value for the lower limit. All other types ignore this value.
  After a reset, the value is always 0. In modes where no trigger is available, in digital modes, or without a selected mode, the value is constantly 0.
- GPIOData
  This INT32 value represents all 24 bit values of the GPIO port, in the order of their bit weight. In the logic analyser mode the data values in byte 3 are used by the device to control level converters. In digital operation, only bytes 0 and 1 (bits 0 - 15) should be connected directly.
- GPIODir
  This INT32 value represents all 24 bit directions of the GPIO port, in the order of their bit weight. The bit value "1" represents an output pin, "0" represents an input. If digital operating modes are active, the bits 16 thru 22 are automatically set by the device to control, if present, a level converter. Using external trigger, bit 23 is always set as an input.

SetAmplitude and SetOffset

The first argument for both functions is the number of the affected channel. The second is the desired value for the amplitude or offset. The device realises the given values upon it's capabilities. Only settings with an offset of 0V are calibrated.
The function also checks the trigger level and scales it to the new measurement range. If, for example, the previous value was 6V and the new measurement range is ±5V, the value is limited down to 4.96V. If the new measurement range is 0-10V, the trigger level is modified within the resolution tolerances. If the measurement range is repeatedly changed, it is recommended to monitor and correct the values as necessary.
The value range and other considerations are described in the “SetupRead” and “SetupWrite” descriptions.
Return values:

- Amplitude
  Setting of the maximum amplitude of the input signal.
- Offset
  Setting of the DC component of the input signal.
- OffsetError
  Correction value for the oscilloscope mode. The value must be subtracted after the calculation.

SetTimebase

This function is used to set the sampling rate for oscilloscope, logic analyser and data logger modes. The value range and other considerations are described in the “SetupRead” and “SetupWrite” descriptions.
The return value is the actually realised time setting.

SetMemory

This function requires the memory depth and the trigger point as arguments. All settings are checked and rounded to the nearest value that is technically possible for the device. The value range and other considerations are described in the “SetupRead” and “SetupWrite” descriptions.
Return values:

- Memory
  realised memory depth
- TriggerPoint
  location of the trigger point within the sample memory

SetTrigger

This setting determines how the trigger operates in the oscilloscope, logic analyser, and data logger modes. The function requires parameters for the channel, the trigger type, the upper limit, and the lower limit. A detailed description of the parameters can be found in the “SetupRead” description. The values for maximum and minimum trigger voltage cannot be chosen close to the measurement range to guarantee proper trigger operation.
The values for the upper and lower limit are ignored in the logic analyser mode.
After a reset, the trigger is set to “Manual” for all modes able to use a trigger, otherwise it is set to “NUL”.

SetMode

By this function the measurement mode of the device is set. The measurement mode determines the restrictions for the rest of the parameters. For instance the sample time in the oscilloscope mode can be set to a much smaller value than in all other modes. If another mode is selected the device uses previous set values or, if never given before, the default values of the new mode. The return value indicates the actual mode. The identifiers given in the “Measurement Modes” section are just mnemonics. The device interprets the values as unsigned 32 bit numbers. If they are sent as ASCII characters by a standard PC using Intel byte order, the order must be reversed. For example the mode "VMA1" should be sent as "1AMV".

SetRealtimeClock

This command adjusts the real time clock that is built in the model UM203. The model UM202 ignores it. Two arguments build of long words are to follow. The first includes the time of day, the second transports the date. The data has to be stuffed into following this order:

Argument	MSB	Byte 2	Byte 1	LSB
1	0	hour (0 - 23)	minute (0 - 59)	second (0 - 59)
2	year 20...(00 - 99)	month (1 - 12)	day (1 - 28/29/30/31)	day of week (1 - 7)

The clock keeps running after switching off the power supply for some hours up to several days. The exact backup time depends on the energy in the buffer capacitor. For maximum duration charging has to last about 24 hours. Damage by longer charging can not occur.

Run

A measurement process is started in the mode, previously selected by “SetMode”. The behaviour during the measurement is the same as it was with the old specific commands.
Return values and delays depend on the mode, memory and time setting and even more on reaching the trigger condition.

Break

This command is used to cancel a running measurement process. The reaction to this command will differ depending on the mode.
For oscilloscope and logic analyser mode the process can be cancelled up until the last measured value is retrieved. In this case, a correct number of fully synthesised values are returned immediately. There is no fixed time relationship between the sending of the “Break” command by the application and the moment when it is received by the device. Because of this, it is not possible for the PC to determine whether the “Break” command occurred during a measurement or during transmission of the real measurement values. In doubt, the received values should be suggested to be invalid.
In data logger mode, the measurement process is cancelled immediately, no further values are sent. Due to uncertain delays, it is possible that data values will still be sent from the device after the “Break” command has been sent. These values can still be read, or be discarded by using the “FT_StopInTask”, “Wait”, “FT_Purge”, or “FT_RestartInTask” commands.
The “Break” function is not a function at all. Any character that is sent will cause a cancellation. It is recommended to use a string that is invalid for the command interpreter. However, the string “ZZZZ” may be obligatory in future firmware versions.

GPIOWrite

This function is used to set the direction and output data of the GPIO ports. It reads back the actual status of the digital ports after writing is performed. However, the hardware does not allow collision detection because the outputs are read from an internal data register, instead of the port pins.

GPIORead

This function is used to read the status of the digital ports. For output ports, the value is read from an internal data register, not from the output port pins.

Inquiry

This function reads the ID string of the device. This string also includes the firmware version, but not the hardware version. The hardware version can read from the device serial number.

Restart

Executing this command will return the device to the manufacturers default settings. The values will always result in a non-critical configuration. Amplitudes are set to their maximum and the sampling rate is set to the minimum value. Digital ports are configured as input. All other values, the trigger parameter for example, are not specified. It is possible that they will be set in future firmware versions. After successful completion, this device will answer with the string “ok”+CR+LF.

Operating Modes

Overview

The device has eight possible modes of operation. Each mode has a separate configuration. As an exception the four voltmeter modes use a common parameter set. All given parameters are checked, depending on the current mode. The measurement process is started with the "*RUN" command. This ensures that the values for the current configuration are valid. The IDs used are just mnemonic. The device interprets them as unsigned 32bit values. If they are sent as ASCII characters, the order must be reversed (example: "VMA1" must be sent as "1AMV").

Description	Mnemonic	online	offline
voltmeter, DC	VMD0	✓	
voltmeter, DC, raw data	VMD1	✓	
voltmeter, true RMS	VMA0	✓	
voltmeter, true RMS, raw data	VMA1	✓	
oscilloscope, analogue	OSA0	✓	✓
data logger, analogue	DLA0	✓	✓
logic analyser on IO ports	LAIO	✓	✓
data logger on IO ports	DLDI	✓	(✓) reduced sample rate

VMD0

This mode measures the DC component of the input channels. All settings must be done before calling this function. Passive probes with attenuation factors different from 1 might have to be taken into account for the calculation. The input impedance of the MEphisto scope 1, hardware version 1.0 is 850 kΩ. You can see the hardware version by inspecting the inquiry string. For version 1.0 it starts witch "Mephisto Scope 1.0" or "Mephisto Scope 1" in older firmware versions. The test probes for the oscilloscope are set for an impedance of 1 MΩ. The actual voltage must be calculated as follows:

$$U[V] = \left(\frac{n_{Probe} - 1}{0{,}85} + 1 \right) \cdot U_{Scope}$$

The hardware version 1.1 always returns "Mephisto Scope 1.1". The impedance is exactly 1 MΩ. Therefore no correction factor has to be used.

Nevertheless, please note that test probes with an attenuation factor different from 1 are often inaccurate.

The two values returned are the measured voltages with data type "float". Channel 0 is sent first.

VMD1

This mode is the same as VMD0 with the exception that the measured values are returned as raw data 2•16bit. The transmission format is "unsigned long". Channel 0 is sent first. The voltage is calculated as follows:

$$U[V] = \left(\frac{n_{Sample}}{32768} - 1 \right) \cdot \frac{Amplitude}{2} + Offset - ZeroPointCorrection$$

VMA0

This mode is the same as VMD0, with the exception that the measured values are returned as RMS values.

VMA1

This mode is the same as VMA0, with the exception that the measured values are returned as raw data. The transmission format is “unsigned long”.

OSA0

In this mode, the device operates as an analogue oscilloscope. The number of data values to be measured depends on the memory depth. The device returns the values as raw data, therefore it is necessary to convert them to a voltage value. The device sends a data pair in the format “unsigned long”. The 32 bit value must be separated into two 16 bit values. The upper 16 bits are the sample values of channel 0, the lower half of channel 1. The voltage is calculated as described for “VMD1”. Passive test probes with an attenuation factor different from 1 must be taken into account for the calculation. The correction factor is as described for “VMD0”.
When the command is sent, it is necessary to continually wait for response. If samples are available, they must be read immediately. Longer delays cause the device to cancel the transmission. The entire sample memory content must always be read completely. Values not read will be dropped after a short time. The measurement time depends on the sampling rate, the memory depth, and the time required to detect the trigger. Even if a process is cancelled with the “Break” command, the device will send data. This data set consists of fully synthetic values, but has the expected length and format.

DLA0

The analog data logger mode operates similar to the oscilloscope mode. The format of the data transmission is also the same as in the oscilloscope mode. The trigger can be used to either start or stop the stream. The “Break” command can be used to cancel a running process. A stream will always terminate with the values 0xFFFF0000, 0x0000FFFF, 0xFFFF0000 and once more 0x0000FFFF. The design of the device does not allow this pattern to appear as real sample values.

LAIO

The logic analyser mode on the GPIO port captures only digital data. The analogue part of the device is inactive. Sampling includes the digital channels from 0 to 15. The trigger pattern can be applied to the channels 0 through 7. When calling this mode, the digital port, if not already done, is set to the appropriate data direction. Bits 0 to 15 are input and will be sampled. Bits 16 to 22 are output. This provides communication and voltage source for an external level converter. Bit 23 is an input channel that can be used as an external trigger. The “Break” command can be used to cancel a running process. For further information, please refer to the description of this command.
The measurement time depends on the amount of memory, the sampling time, and the time required to meet the trigger condition. The external trigger signal is not stored. If this is required, the signal must additionally be connected to a measurement channel.
Reading the data is done in two synchronous groups of each 8 bits. A fixed time delay of 2µS will occur between bits 0 to 7 and bits 8 to 15. The external trigger has a maximum latency of 1 sample, regardless of the sampling rate.
Sampling data is transferred as “unsigned long” values. The device sends 2 x 16bit packed values as measured data. The older value is in the upper 16 bits, whereas the newer value is in the lower half. A cancelled process will result in synthesised values being sent.

DLDI

This mode is used as a digital data logger. The device acts the same as for the analogue data logger. The data format is different from that of the logic analyser. The data is transferred as “unsigned long” values, but only a single 16 bit value is sent. This guarantees that the “end” pattern is unique and can be easily recognized. A measurement block will always end with the values 0xFFFF0000, 0x0000FFFF, 0xFFFF0000 and again 0x0000FFFF.

In offline mode, the sampling rate is limited to 2.5kHz resp. 400µs. Correction of the timebase will happen just before the offline measurement. Being online, the device accepts all values up to the system limit.

Trigger Types, analogue

Overview

All analogue and digital modes can use a trigger, except the volt meter mode.

Description	Trigger Type	Used Parameter
level, above	T	upper trigger level
level, below	t	upper trigger level
window, inside	W	upper trigger level, lower trigger level
window, outside	w	upper trigger level, lower trigger level
edge, positive	E	upper trigger level
edge, negative	e	upper trigger level
derivative, positive slope	D	upper trigger level
derivative, negative slope	d	upper trigger level
external, positive edge	X	-
external, negative edge	x	-
manual	M	-

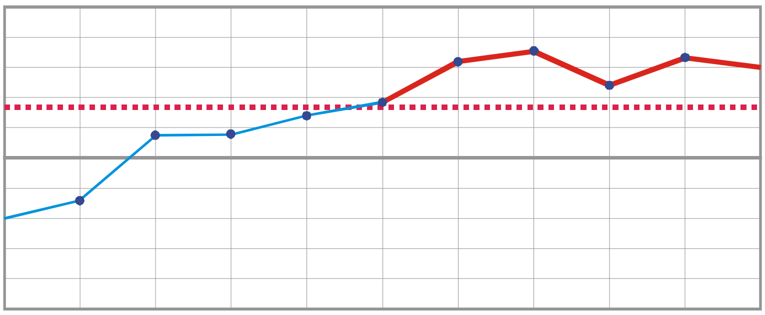
Manual

This trigger mode indicates a free running measurement. It begins immediately after sending the command. This is the default setting after a power up for all triggerable modes.

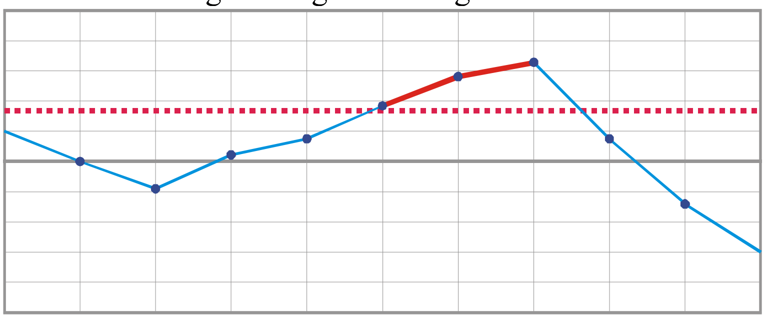
Level, above/below

In this mode, the oscilloscope reacts to the input exceeding or dropping below the given threshold. How this occurs does not matter. This trigger mode is especially useful for two cases:

1. tracking a “switch on” event



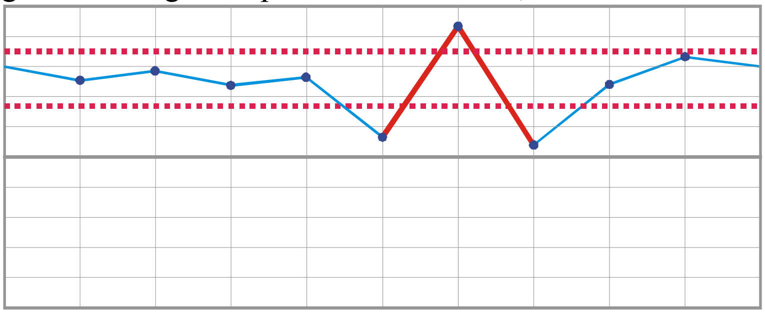
2. detecting a short exceeding of the given voltage



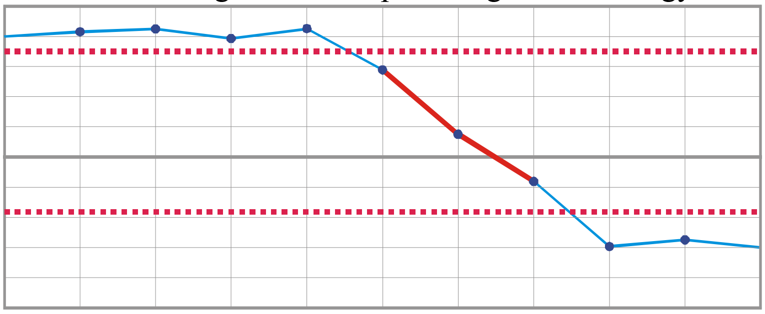
Window, inside/outside

These modes set a range with upper and lower limit values. The trigger event occurs when the input signal enters or leaves the defined window range. It does not matter if this occurs on the upper or lower limit. This mode is used for monitoring DC voltages.

1. Monitoring a DC voltage for spikes with „window, inside“



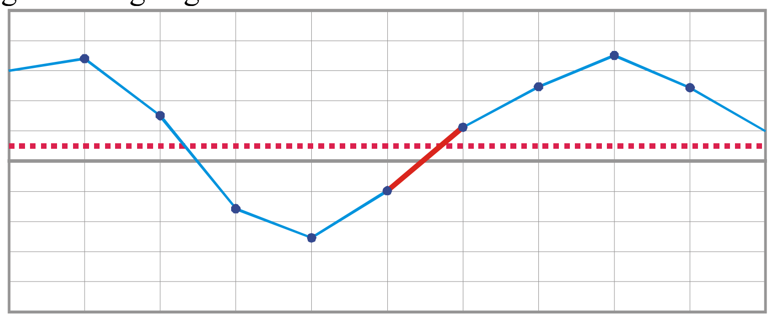
2. Detecting a „forbidden range“ for example in digital technology



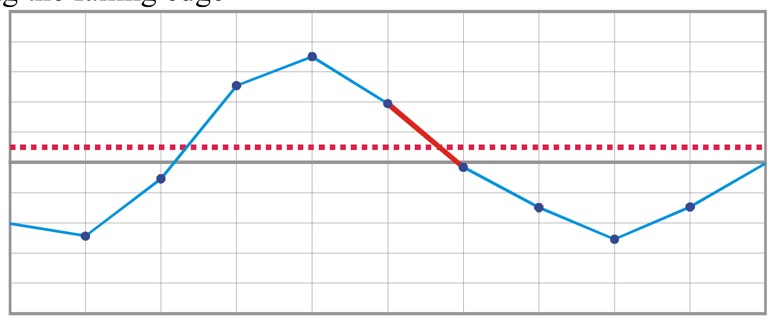
Edge, rising/falling

This mode is similar to the level trigger. The trigger event occurs when the threshold is crossed rising or falling. The edge trigger does not occur when the signal's voltage is continuously above or below the set values. This mode is used to create a steady graph for periodic AC signals.

1. Displaying the rising edge



2. Displaying the falling edge



Derivative Trigger, positive/negative

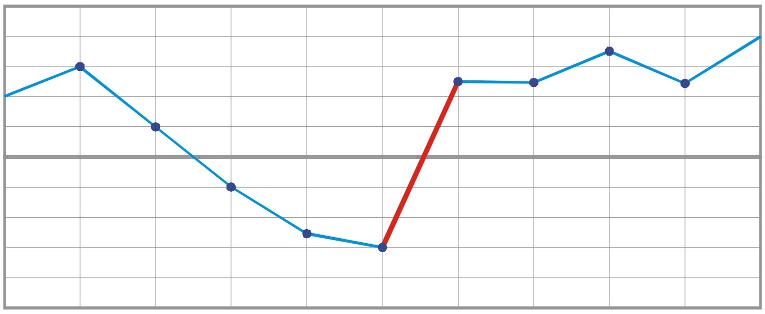
This mode is used for special cases of dynamic signals. It is almost independent of the actual voltage. Only the change of the signal is monitored. If the signal rises or falls faster than the set rate, the trigger event occurs.

The rate is given as a voltage/sample time. The sample time must be calculated from the quotient time/memory depth.

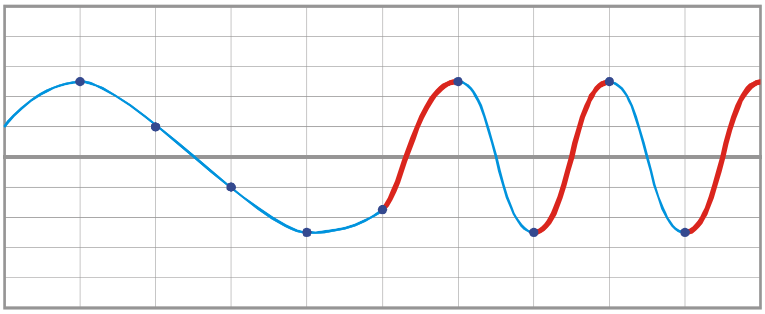
$$\frac{dU}{dt} = \frac{U[\mathrm{V}] \cdot Memory[\mathrm{Samples}]}{t_{\mathrm{TimeBase}}[\mathrm{s}]} \quad \text{bzw.} \quad U[\mathrm{V}] = \frac{dU}{dt} \cdot \frac{t_{\mathrm{TimeBase}}[\mathrm{s}]}{Memory[\mathrm{Samples}]}$$

Checking the slope is important for two applications. The first is for process control in chemical, biological, or other science fields. Even large swings are normal in these areas, but sudden changes often indicate an error. The second area is for doing frequency identification. Detection of transient spikes in steady signals has to be mentioned here as highlight.
The MEphisto scope 1 has a limited measurement range. To avoid uncontrolled swings out of bounds, the trigger also occurs when the current voltage plus the change in voltage would exceed the range.

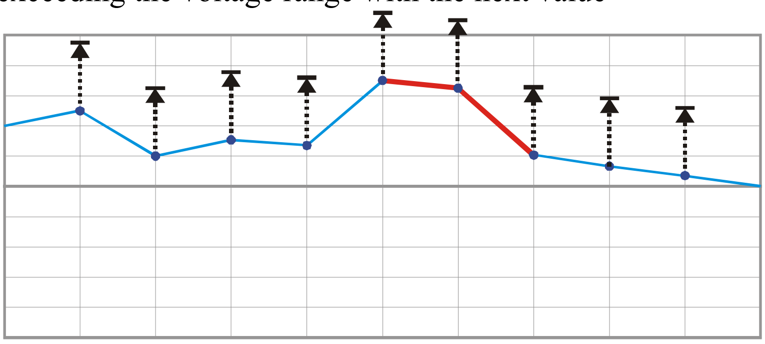
1. Recognition of a critical state



2. Frequency detection



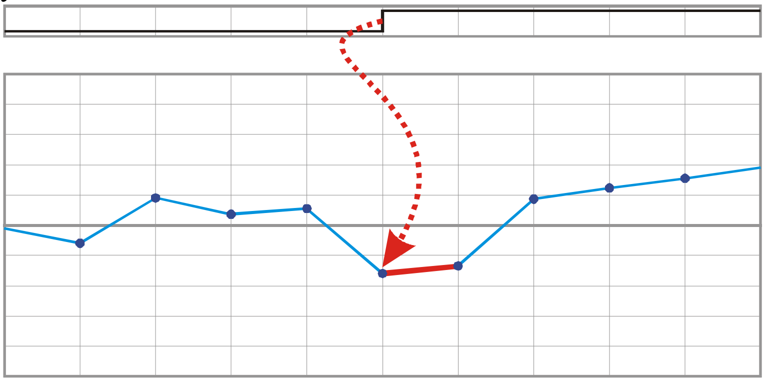
3. Probably exceeding the voltage range with the next value



External, rising/falling edge

This mode is similar to the edge trigger in that it waits for a rising or falling edge. The difference is that the edge is not on one of the analogue input signals, but it must be on the digital signal on I/O port D23 (pin 26). Like the edge trigger, only a logical edge will cause the trigger event. A static signal is ignored. This trigger is used for synchronisation with other devices. It is important to check the compatibility of the trigger source and the MEphisto

Scope 1. Most signal generators available offer a 5V TTL signal which can be used to synchronize the system.



Trigger Types, digital

Overview

Description	Trigger Type	Used Parameter
pattern	P	bit pattern
external, positive edge	X	-
external, negative edge	x	-
manual	M	-

Bit Pattern

The lower 8 bits of the 16 bit measurement signal are used to identify the trigger event. The bit pattern of the digital trigger is passed in the channel selection. Three of the four bytes are used to encode the five possible states. Byte 0 is ignored in digital mode.

Byte	Description	Bit = 0	Bit = 1
0	-	-	-
1	relevance	bit will be ignored	bit will be tested
2	dynamic	bit will be tested for it's level	bit will be proved to represent an edge
3	level	level=low or edge=falling	level=high or edge=rising

Manual

This trigger mode indicates a free running measurement. It begins immediately after sending the command. This is the default setting after a power up for all triggerable modes.

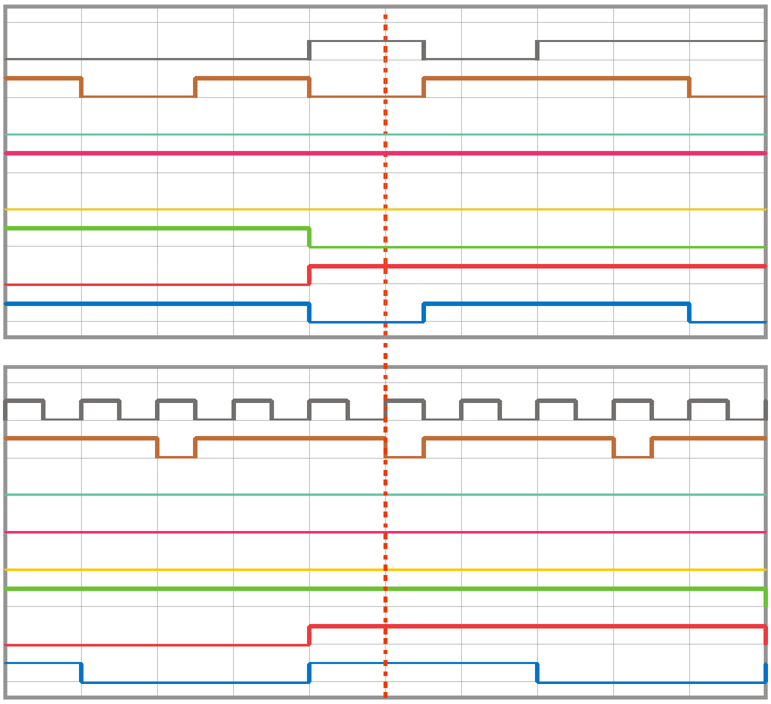
Pattern

This mode causes the logic analyser to trigger on a given bit pattern. Only the lower 8 bits are evaluated. Each of the 8 bits can have one of the following states:

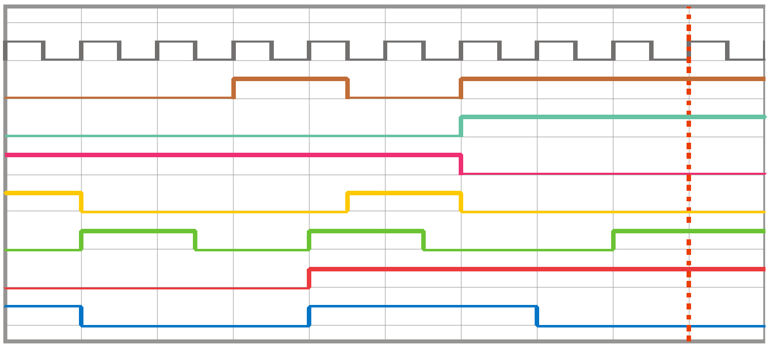
- level, high
- level, low
- edge, rising
- edge, falling
- don’t care

The states are transmitted as described in the table above.

A safe trigger condition is usually reached using a combination of different criteria. An example may be a memory access of a processor. Channel 7 can, for example, read in a clock signal whose rising edge is relevant. Channel 6 can show a write pulse that is activated on a falling edge. Channels 0 to 5 can be used for addressing in this scenario. The rest of the channels, 8 to 15, will represent data and have no influence on the trigger condition.
If we wanted to determine which value is written to memory address 7_{hex} (000111_{bin}), the trigger pattern RF000111 would be used. Channels 8 to 15 would contain the data. In this example, it would be a value of 92_{hex} (10010010_{bin}). Furthermore you see that the setup and hold times would most likely meet the requirements.
This example is intended for illustration purposes and would only be possible if a single stepping micro controller or emulator is being used.



A state machine or a sequencer can be tested to make sure that a critical state does not occur by making this critical state a trigger condition. In this example, the trigger pattern is R110010. The clock edge should be used right as above to make sure that the trigger condition does not occur during an asynchronous glitch of other signals and states. The process recorded before the trigger condition would indicate what happened before the failure occurred.



These two very different examples show how powerful a tool the triggering on a freely defined pattern can be. It should be noted that signals that are not exactly timed and correlated make it very difficult to get a reliable trigger. The number of channels used to define the trigger event should therefore be kept to a minimum. When the number of trigger conditions is reduced, the chance that an unwanted trigger will occur increases. It is important to find a compromise that offers the safest way of defining a trigger condition.

External, rising/falling edge

This mode of trigger operation, similar to the pattern trigger, waits for a rising or falling edge. The difference is that the edge is not on one of the input channels, but it must be on the digital signal on I/O port D23 (pin 26). Like the edge trigger, only a logical edge will cause the trigger event. A static signal will be ignored. This trigger is used for synchronisation with other devices. It is important to check the compatibility of the trigger source and the MEphisto Scope 1. Most signal generators available offer a 5V TTL signal which can be used to synchronize the system.

GPIO Direction Information

Direction	Bit Value
input	0
output	1

The “MEphisto Scope 1” offers 24 digital I/O ports. In the analogue modes, each channel can be independently configured as input or output. On power up, all channels are configured as inputs.
The digital channel DIO_24 (pin 26) can be used as an external trigger. The channel is reserved exclusively for external trigger operation as soon as the trigger mode is set. After this the channel is remains an input. It is not possible to reprogram it after the mode is set. The channel is released when a different trigger mode is chosen.

USB Driver

The USB component of the MEphisto Scope 1 is the FT245BM produced by FTDI. Driver updates for Windows, Windows CE, Linux etc. are always available there for download. The device ID of the MEhpisto Scope 1 is "DCD0". Note that this ID is not supported by any of the original drivers. Meilhaus Electronic will supply drivers on the MEphisto Scope 1 support site if necessary.

Communication using Windows

The driver supplied by Meilhaus Electronic is adapted to support the device ID. It ensures that data is read from the device in a short enough time. Once the communication has been established, it is not possible that any data will be lost. Tests results for processes that use longer wait times before reading the driver buffer do not exist. Make sure to read the notes listed below.

1. open the device `FT_OpenEX`
2. communication
   a. clear the buffer (optional): `FT_Purge`
   b. write data: `FT_Write`
   c. wait for response: `FT_GetQueueStatus`
   d. read data: `FT_Read`
3. close the channel: `FT_Close`

Communication using Linux

Many available Linux distribution packages already have a driver for the USB component of the MEphisto Scope 1. Please note, that the device ID is "DCD0" and will not be supported by the standard drivers. The communication is handled using a virtual serial port. Currently, there is no specialised driver available for the MEphisto Scope 1.
An alternative could be to access the device using the kernel USB driver. The USB component sends continuous 64 byte packets, each of which contains 2 status bytes.

Notes

During the development of the PC software for windows some driver and chip anomalies were found. Some of them also occur with Linux. Meilhaus Electronic has no knowledge about occurrences in other programming languages. They are noted for helping to trace errors.

After opening a channel, it is recommended to send the very slow command "`*IDN?`" command. The driver seems to have a significant delay for the first communication before passing the data. If the device answers too quickly, it is possible that data will be lost immediately or later on on some computers. No schemes could be found by now. This error seems to occur with both fast and slow processors and with all different operating systems. It seems to occur more often when the device is repeatedly connected to the USB port. Once the communication has been successfully established, there are no known problems or restrictions.

The function "`FT_GetQueueStatus`" of the windows driver only returns the correct number of data items in the receive buffer up to a value of 63488. This value will never be exceeded. This error occurs when the oscilloscope or logic analyser modes are used with large memory settings. It might also occur, when using the logger modes with high sample rates. The very latest moment to start the read process is when the value 63488 is reached, because the driver spooler does not to accept 65535 bytes of data from the device. Using logger modes, the reading should start at lower values, because of continuous subsequent incoming data.