

MEPHISTO SCOPE 1

KOMMANDO-

INTERPRETER

***Firmware-Version 3.10
21.01.2010***



© Meilhaus Electronic GmbH

INHALT

EINLEITUNG	3
KOMMANDOS	3
Übersicht	3
SetupWrite und SetupRead	4
SetAmplitude und SetOffset	7
SetTimebase	8
SetMemory	8
SetTrigger	8
SetMode	8
SetRealtimeClock	8
Run	9
Break	9
GPIOWrite	9
GPIORead	9
Inquiry	9
Restart	9
MESSMODI	11
Übersicht	11
VMD0	11
VMD1	11
VMA0	11
VMA1	12
OSA0	12
DLA0	12
LAIO	12
DLDI	13
TRIGGER-TYPEN, ANALOG	14
Übersicht	14
Manuell	14
Schwellwert Über-/Unterschreitung	14
Fenster-Eintritt/-Austritt	14
Flanke steigend/fallend	15
Steigungstrigger positiv/negativ	16
Extern, steigende/fallende Flanke	17
TRIGGER-TYPEN, DIGITAL	18
Übersicht	18
Bit-Muster	18
Manuell	18
Muster	18
Extern, steigende/fallende Flanke	19
GPIO-RICHTUNGSANGABE	21
USB-TREIBER	22
Kommunikation unter Windows	22
Kommunikation unter Linux	22
Anmerkungen	22

Einleitung

Jede Kommunikation läuft mit 32 Bit Datenworten ab. Kürzere Datentypen sind mit Null-Bytes aufzufüllen. Es müssen immer alle Rückgabewerte des Gerätes sofort gelesen werden. Geschieht dies nicht, werden sie je nach Betriebs-Modus wortweise oder als Block verworfen. Alle Einstellungen werden vom Gerät geprüft und auf den nächsten technisch realisierbaren Wert gerundet. Dieser Wert wird in der Antwort übergeben.

Die Laufzeiten können je nach Firmware-Revision, Betriebsmodus und übergebenen Werten leicht variieren.

Das Modell UM202 ist nur für den Betrieb am PC geeignet. Alle Meßdaten werden über die USB-Schnittstelle übertragen, sobald sie zur Verfügung stehen.

Das Modell UM203 hat die gleiche Funktionalität wie UM202. Darüber hinaus kann es Messungen auch ohne PC vornehmen. In diesem Offline-Modus wird die letzte online vorgenommene Konfiguration geladen. Das Gerät wartet dann auf das Einstecken einer SD-Karte. Nach deren Initialisierung nimmt es autonom Messungen vor und speichert die Ergebnisse auf der Karte ab.

Kommandos

In der Firmware-Version 2.00 wurde der Kommando-Interpreter grundlegend neu gestaltet. Kommandos der Versionen 1.xx werden nicht mehr unterstützt.

Übersicht

Funktion	Befehl	Argumente Richtung	Daten- Typ	Inhalt
vollständiges Setup (SetupWrite)	*SWr	Host → Dev.	Float Float Float Float Float Float Float ULong ULong Float Float ULong ULong	Amplitude Kanal 0 Amplitude Kanal 1 Offset Kanal 0 Offset Kanal 1 Zeitbasis Speichertiefe Trigger-Punkt (1-99%) Trigger-Kanal Trigger-Typ oberer Trigger-Level unterer Trigger-Level Daten GPIO Richtungsinformationen GPIO
		Dev. → Host	Float Float Float Float Float Float Float Float ULong ULong Float Float ULong ULong	Amplitude Kanal 0 Amplitude Kanal 1 Offset Kanal 0 Offset Kanal 1 Nullpunkt-Korrektur Kanal 0 Nullpunkt-Korrektur Kanal 1 Zeitbasis Speichertiefe Trigger-Punkt (1-99%) Trigger-Kanal Trigger-Typ oberer Trigger-Level unterer Trigger-Level Daten GPIO Richtungsinformationen GPIO
Parameter holen (SetupRead)	*SRd	Host → Dev.	-	-
		Dev. → Host	Float Float Float Float Float Float Float Float ULong ULong ULong ULong ULong ULong ULong ULong	Amplitude Kanal 0 Amplitude Kanal 1 Offset Kanal 0 Offset Kanal 1 Nullpunkt-Korrektur Kanal 0 Nullpunkt-Korrektur Kanal 1 Zeitbasis Speichertiefe Trigger-Punkt (1-99%) Trigger-Kanal Trigger-Typ oberer Trigger-Level

Funktion	Befehl	Argumente Richtung	Daten- Typ	Inhalt
			Float ULong ULong	unterer Trigger-Level Daten GPIO Richtungsinformationen GPIO
Amplitude setzen (SetAmplitude)	*SAm	Host → Dev.	ULong Float	einzustellender Kanal Soll-Amplitude
		Dev. → Host	Float Float Float	eingestellte Amplitude eingestellter Offset-Wert Nullpunkt-Korrektur
Offset setzen (SetOffset)	*SOf	Host → Dev.	ULong Float	einzustellender Kanal Soll-Offset
		Dev. → Host	Float Float Float	eingestellte Amplitude eingestellter Offset-Wert Nullpunkt-Korrektur
Zeitbasis setzen (SetTimebase)	*STm	Host → Dev.	Float	Zeitvorgabe
		Dev. → Host	Float	eingestellte Zeit
Speicher-Parameter einstellen (SetMemory)	*SMe	Host → Dev.	Float Float	Soll-Speichertiefe Trigger-Punkt (0-100%)
		Dev. → Host	s.o.	s.o.
Trigger-Parameter einstellen (SetTrigger)	*STr	Host → Dev.	ULong ULong Float Float	Trigger-Kanal Trigger-Typ oberer Trigger-Level unterer Trigger-Level
		Dev. → Host	Float Float	oberer Trigger-Level unterer Trigger-Level
Meßmodus setzen (SetMode)	*SMd	Host → Dev. Dev. → Host	ULong ULong	Meßmodus Vorgabe gesetzter Meßmodus
Echtzeituhr setzen (SetRealtimeClock)	*SRt	Host → Dev.	ULong ULong	Tageszeit Datum
		Dev. → Host	-	
Messung starten (Run)	*RUN	Host → Dev.	-	
		Dev. → Host	variabel	Der Datentyp und die Anzahl der Daten richtet sich nach dem mit dem Kommando *SMm gesetzten Meßmodus. Es gilt das für die einzelnen Modi gesagte.
Abbruch (Break)	ZZZZ	Host → Dev.	-	Sub-Kommando, nur zum vorzeitigen Beenden einer Messung, z.B. wenn das Trigger-Ereignis nicht erreicht wird. Es ist ausschließlich innerhalb einer Messung gültig.
		Dev. → Host	-	
GPIO setzen (GPIOWrite)	*IOW	Host → Dev.	ULong ULong	24 Bit Daten 24 Bit Richtungsinformationen
		Dev. → Host	ULong ULong	24 Bit Daten 24 Bit Richtungsinformationen
GPIO lesen (GPIORead)	*IOR	Host → Dev.	-	
		Dev. → Host	ULong ULong	24 Bit Daten 24 Bit Richtungsinformationen
IEEE 488.2 - Inquiry (Inquiry)	*IDN?	Host → Dev.	Byte	unbeschränkte Zahl weiterer Zeichen, z.B. CR+LF
		Dev. → Host	String	Produkt-ID-String+CR+LF, 32 Zeichen
IEEE 488.2 – Reset (Restart)	*RST	Host → Dev.	Byte	unbeschränkte Zahl weiterer Zeichen, z.B. CR+LF
		Dev. → Host	String	„ok“+CR+LF, 4 Zeichen
Firmware-Update	*UPD	Host → Dev.	diverse	Update-Funktion, nur zum internen Gebrauch der Firma Meilhaus Electronic.
Kalibriermodus aktivieren	*CAL	Host → Dev.	diverse	Kalibrier-Modus, nur zum internen Gebrauch der Firma Meilhaus Electronic. Alle bisherigen Kalibrierparameter werden beim Aufruf dieses Befehls gelöscht!

SetupWrite und SetupRead

Die Funktion „SetupRead“ gibt alle für den aktuellen Meßmodus im Gerät gespeicherten Einstellungen zurück. Sie sollte unmittelbar nach dem Setzen eines neuen Modus aufgerufen werden, um alle Parameter zu erhalten, nach denen das Gerät arbeiten wird.

Die Funktion „SetupWrite“ nimmt bis auf die beiden Korrektur-Faktoren OffsetErrorX alle oben beschriebenen Werte als Eingangsparameter an. Die Werte werden vom Gerät geprüft und die tatsächlich eingestellten Werte wieder ausgegeben. Die Ausgabe entspricht exakt der der Funktion „SetupRead“.

Die Rückgabewerte sind:

- Amplitude0, Amplitude1
Maximaler Hub des Eingangssignals, zentriert um die Offset-Spannung

Die Spannung kann im 1-2-5-Raster zwischen 200mV und 20V eingestellt werden. Abweichende Werte werden bestmöglich angeglichen. Eine Änderung der Amplitude kann eine Änderung des Offset nach sich ziehen, wenn dadurch die Maximalspannung des gewählten Bereiches überschritten würde. Ist der Offset 5V, der neue Spannungsbereich jedoch nur 2V, so wird der Offset-Wert auf 2V reduziert.

Für alle digitalen Modi richtet sich der Wert nach der Spannung einer logischen „1“. Ohne externen Pegelwandler beträgt der Wert 5V.

Der Wert nach Reset beträgt für alle analogen Modi 20V. Für alle digitalen Modi richtet sich der Wert nach der Spannung einer logischen „1“. Ohne externen Pegelwandler beträgt der Wert 5V. Ohne gesetzten Modus ist der Wert unveränderlich 0.

- Offset0, Offset1

Mittelpunkt des Meßbereiches

Diese Angabe kann innerhalb der Grenzen von $\pm Amplitude/2$ in 4096 Stufen frei gewählt werden. Bei der maximalen Spannung von $\pm 10V$ ist der Offset jedoch immer 0V.

Der Wert nach Reset beträgt für alle analogen Modi 0V. Für alle digitalen Modi richtet sich der Wert nach der halben Spannung einer logischen „1“. Ohne externen Pegelwandler beträgt der Wert 2,5V. Ohne gesetzten Modus ist der Wert unveränderlich 0.

- OffsetError0, OffsetError1

Korrektur-Wert für analoge Modi

Bei diesem Wert handelt es sich um einen Kalibrier-Parameter, der werksseitig für jedes Gerät individuell bei 25°C ermittelt wird. Dieser Wert muß nach der Berechnung der Spannung aus Rohdaten abgezogen werden. Der Parameter kann nur gelesen werden. Die Funktion SetupWrite bietet ihn nicht zur Korrektur an.

- SamplingTime

Zeit zwischen zwei Messungen

Im Oszilloskop-, Logic-Analyser- und Daten-Logger-Modus gibt diese Zeit die Spanne zwischen zwei Meßwerten an. Der Wertebereich erstreckt sich von 1 μ s/S bis 2.5s/S im Oszilloskop-Modus und von 10 μ s bis 2.5s im Logic-Analyser-Modus. Im analogen Daten-Logger-Modus ist die Abtastrate ebenfalls auf 10 μ s begrenzt. Unterhalb von 10ms/S wird pro Messung einmal abgetastet. Die Werte können mit einer Auflösung von 1 μ s eingestellt werden. Oberhalb dieser Schwelle kommt ein mittelwert-bildender Meßalgorithmus zum Einsatz. Die Zeit läßt sich oberhalb dieses Wertes nur in Schritten von 10ms festlegen. Das Signal wird fest mit einer Abtastrate von 10ms gemessen.

In den Voltmeter-Modi hat diese Angabe immer den Wert 0.9s und repräsentiert die ungefähre Dauer einer Messung.

Ohne gesetzten Modus ist die Angabe unveränderlich 0s. Für alle anderen Modi nimmt der Wert nach dem Reset sein jeweiliges Minimum an.

- MemoryDepth

Tiefe des Meßwert-Speichers im Oszilloskop- und Logic-Analyser-Modus

Die kleinste einstellbare Speichertiefe ist 100 Samples pro Kanal. Sie ist im 1-2-5-Raster einstellbar. Der maximale Speicher weicht vom

Raster ab. Er beträgt im Oszilloskop-Modus 131000 Samples, im Logic-Analyser-Modus dagegen 262000 Samples.

Der Meßzeitraum ergibt sich aus dem Produkt

$$AcquisitionTime = SamplingTime \cdot MemoryDepth$$

Der Wert nach Reset beträgt für Oszilloskop- und Logic-Analyser-Modus 1000, alle anderen Modi melden unveränderlich den Wert 1. Ohne gesetzten Modus ist der Wert unveränderlich 0.

- **TriggerPoint**

Lage des Triggerpunktes innerhalb des Meßzeitraumes

Die Angabe des Trigger-Punktes erfolgt in Prozent vom Meßwert-Speicher. Je nach Firmware reicht der Wertebereich unterschiedlich weit an die Grenzen heran. Die Werte 0% und 100% sind im Oszilloskop und Logic-Analyser-Modus nicht einstellbar. Ein geringer Vor- bzw. Nachlauf muß toleriert werden.

Bei den Daten-Logger-Modi hingegen werden alle Werte nach 0% oder 100% hin ab – oder aufgerundet. Da der Meßwert-Speicher für diese Modi nicht genutzt wird, repräsentieren die Werte Pre- bzw. Post-Trigger.

In den Voltmeter-Modi und ohne gesetzten Modus ist der Wert unveränderlich 0.

Nach Reset liegt der Triggerpunkt in den Modi Oszilloskop und Logic Analyser bei 50%, bei den Daten-Logger-Modi beträgt er 0%.

- **TriggerChannel**

Diese Angabe hat zwei Interpretationsarten, abhängig vom Meßmodus.

Soll das Gerät als Oszilloskop messen, kann der Trigger-Kanal den Wert 0 oder 1 annehmen. Ausgewertet wird hierbei nur das niederwertigste Byte 0.

Nach dem Reset ist der Trigger-Kanal für Oszilloskop- und Analog-Datenlogger-Modus auf 0 eingestellt.

Beim Logic-Analyser enthält dieser Parameter Angaben über das zu vergleichende Bit-Muster. Genauere Angaben hierzu finden sich in der Tabelle im Kapitel über digitale Trigger-Typen. In diesem Modus werden die Bytes 1 bis 3 genutzt.

Nach dem Reset ist das Trigger-Muster für Logic-Analyser- und Digital-Datenlogger-Modus auf 0 eingestellt, der Trigger somit deaktiviert.

Ohne gesetzten Modus ist der Wert unveränderlich 0.

- **TriggerType**

Der Typ wird durch ein ASCII-Zeichen repräsentiert, das zu einem 32-Bit-Wort aufgefüllt werden muß.

Je nach Firmware stehen unterschiedlich viele Modi zu Verfügung. Die derzeit unterstützten Modi sind in den Tabellen ihrer jeweiligen Kapitel aufgelistet. Der Wert nach Reset ist für alle trigger-fähigen Modi „M“, manuell. Alle anderen Modi melden unveränderlich den Wert NUL.

- **TriggerLevelUp**

Spannungsangabe für die Trigger-Typen T, t, W, w, E, e, D und d

Beim Fenster-Trigger ist dies die Angabe der oberen Grenze, bei allen anderen Arten ist dies die einzige Angabe. Die Typen X, x und M ignorieren diesen Wert. Im Logic-Analyser-Modus wird dieser Wert ebenfalls ignoriert.

Nach einem Reset ist der Wert 0 voreingestellt. In nicht triggerbaren und den digitalen Modi sowie ohne gesetzten Modus ist der Wert 0 nicht veränderbar.

- **TriggerLevelDown**
Spannungsangabe für die Trigger-Typen W und w
Dies ist die Angabe der unteren Grenze. Alle anderen Typen ignorieren diesen Wert.
Nach einem Reset ist der Wert 0 voreingestellt. In nicht triggerbaren und den digitalen Modi sowie ohne gesetzten Modus ist der Wert 0 nicht veränderbar.
- **GPIOData**
In diesem INT32-Wert sind alle 24 Bit-Werte des GPIO-Ports entsprechend ihrer Wertigkeit abgelegt. Die Daten des Byte 3 werden im Logic-Analyser-Modus nur vom Gerät verändert, um digitale Pegelwandler ansteuern zu können. Im Digital-Betrieb sollten daher nur die Bytes 0 und 1 (Bits 0 bis 15) direkt beschaltet sein.
- **GPIODir**
In diesem INT32-Wert sind die Richtungsangaben aller 24 Bit-Werte des GPIO-Ports entsprechend ihrer Wertigkeit abgelegt. Der Bit-Wert 1 steht dabei für einen Ausgang, 0 für einen Eingang. Im Digital-Betrieb werden die Bits 16 bis 22 automatisch vom Gerät angepaßt, um einen eventuell vorhandenen Pegelwandler betreiben zu können. Das Bit 23 ist in allen Betriebsmodi bei Verwendung des externen Trigger als Eingang geschaltet.

SetAmplitude und SetOffset

Diese Funktionen übernehmen als erstes Argument den einzustellenden Kanal. Als zweites Argument ist ihnen der Soll-Wert für die Amplitude bzw. den Offset mitzugeben. Das Gerät stellt sie gemäß seiner technischen Möglichkeiten ein. Kalibriert sind nur Einstellungen mit einem Offset von 0V.

Die Funktionen prüfen ebenfalls den Trigger-Level und passen ihn ggf. dem neuen Meßbereich an. War die Vorgabe 6V und der neue Bereich $\pm 5V$, so wird er auf das nun verfügbare Maximum von 4,96V herabgesetzt. Ist der neue Meßbereich 0-10V, so ändert sich der Trigger-Level nur im Rahmen der Auflösungstoleranzen. Bei wiederholten Bereichswechseln ist es jedoch ratsam, die implizite Einstellung der Trigger-Schwellen zu beobachten und ggf. zu korrigieren.

Wertebereiche und Randbedingungen sind bei der Beschreibung der Parameter der Funktionen SetupRead und SetupWrite nachzulesen.

Die Rückgabewerte sind:

- **Amplitude**
eingestellte maximale Amplitude des Eingangssignals
- **Offset**
eingestellter Gleichspannungsanteil des Eingangssignals
- **OffsetError**
Korrektur-Wert für den Oszilloskop-Modus
Dieser Wert muß nach der Berechnung abgezogen werden.

SetTimebase

Mit dieser Funktion wird die Abtastrate des Oszilloskop-, Logic-Analyser und Daten-Logger-Modus gesetzt. Weitere Randbedingungen sind bei der Beschreibung der Parameter der Funktionen SetupRead und SetupWrite nachzulesen.

Der Rückgabewerte ist die tatsächlich realisierte Zeiteinstellung.

SetMemory

Diese Funktion übernimmt Speichertiefe und Triggerpunkt als Argumente. Das Gerät stellt sie gemäß seiner technischen Möglichkeiten ein.

Wertebereiche und Randbedingungen sind bei der Beschreibung der Parameter der Funktionen SetupRead und SetupWrite nachzulesen.

Die Rückgabewerte sind:

- Memory
eingestellte Speichertiefe
- TriggerPoint
Lage des Triggerpunktes innerhalb des Meßwertspeichers

SetTrigger

Diese Funktion bestimmt das Trigger-Verhalten im Oszilloskop- und Logic Analyser-Modus sowie bei den Daten-Logger-Modi. Als Parameter ist der Kanal, der Trigger-Typ, die obere sowie die untere Grenze anzugeben. Eine ausführliche Beschreibung der Argumente ist bei der Funktion „SetupRead“ nachzulesen. Die Werte für Ober- und Untergrenze reichen nicht ganz bis an die Grenze des Meßbereichs heran, um sicheres Auslösen auch bei Flanken- und Steigungs-Trigger zu gewährleisten.

Im Logic-Analyser-Modus werden die Werte für Ober- und Untergrenze ignoriert.

Die Standard-Vorgabe für den Triggertyp nach Reset ist „manuell“ für alle trigger-fähigen Modi und unveränderlich NUL sonst.

SetMode

Über diese Funktion wird der zukünftige Meßmodus gesetzt. Er bestimmt die Restriktionen für die restlichen Parameter. So kann die Zeitvorgabe im Oszilloskop-Modus wesentlich kleiner gewählt werden, als in allen anderen Modi. Beim Umschalten auf einen anderen Modus lädt das Gerät, wenn vorhanden, zuvor gemachte Vorgaben, sonst die Standard-Werte des neuen Modus. Der Rückgabewert informiert über den tatsächlich eingestellten Modus. Die im Kapitel „Meßmodi“ angegebenen Kennungen verstehen sich als Mnemonic. Interpretiert werden sie vom Gerät als vorzeichenlose 32-Bit-Zahl. Sollen sie als ASCII-Zeichen von einem Standard-PC mit Intel-Byte-Order gesendet werden, so ist die Zeichen-Folge umzukehren. Aus dem Modus „VMA1“ wird so „1AMV“.

SetRealtimeClock

Dieses Kommando setzt die interne Uhr des UM203. Das Modell UM202 ignoriert den Befehl. Als Argument werden zwei Langworte erwartet. Mit dem ersten wird die Tageszeit übergeben, das zweite transportiert das Datum. Die Angaben sind in folgender Weise zu machen:

Argument	MSB	Byte 2	Byte 1	LSB
1	0	Stunde (0 - 23)	Minute (0- 59)	Sekunde (0-59)
2	Jahr 20... (00 - 99)	Monat (1 - 12)	Tag (1 - 28/29/30/31)	Wochentag (1 - 7)

Die Uhr läuft nach dem Abschalten der Stromversorgung noch einige Stunden bis Tage weiter. Die Dauer hängt vom Ladezustand des Puffer-Kondensators ab. Zur maximalen Dauer muß er mindestens 24 Stunden geladen werden. Ein Überladen ist nicht möglich.

Run

Es wird eine Messung im mit „SetMode“ gesetzten Meßmodus gestartet. Das Verhalten bei der Messung ist identisch mit dem, welches durch die alten spezifischen Kommandos ausgelöst wurde.

Rückgabewerte und Verzögerung richten sich nach Modus, Speicher- und Zeitvorgabe sowie dem Erreichen der Trigger-Bedingung.

Break

Dieses Kommando bricht eine laufende Messung ab. Die Reaktion auf das Kommando ist unterschiedlich.

Bei den Meßmodi Oszilloskop und Logic-Analyser kann die laufende Messung bis zur Aufnahme des letzten Meßwerts abgebrochen werden. In diesem Fall sendet das Gerät sofort die geforderte Anzahl an Werten, die jedoch synthetisch erzeugt sind. Da es keine feste zeitliche Beziehung zwischen dem Auslösen des Break-Kommandos durch die Applikation und dem Empfang durch das Gerät gibt, kann der PC nicht erkennen, ob das Kommando noch während der Messung beim Gerät eintreffen wird oder erst während des Sendens der realen Meßwerte. Im Zweifel sind alle empfangenen Daten als ungültig anzusehen.

In den Daten-Logger-Modi wird die Messung sofort abgebrochen und kein weiterer Wert gesendet. Aufgrund der unkalkulierbaren Verzögerung können nach dem Absetzen des Break-Kommandos noch eine Reihe von Werten vom Gerät gesendet werden. Diese können entweder ausgelesen werden oder durch eine Kombination aus FT_StopInTask, Warten, FT_Purge und FT_RestartInTask verworfen werden.

Die Funktion „Break“ ist keine Funktion im eigentlichen Sinn. Jedes während einer Messung gesendete Byte führt zu sofortigem Abbruch. Es empfiehlt sich, eine für den Kommando-Interpreter ungültige Zeichenkette zu senden. Die Zeichenfolge „ZZZZ“ kann jedoch in späteren Firmware-Versionen verbindlich werden.

GPIOWrite

Diese Funktion setzt Datenrichtung und Ausgabe-Daten des GPIO-Ports. Sie liest anschließend den tatsächlichen Status der Digital-Ports wieder ein. Die Hardware ermöglicht dabei keine Kollisionserkennung, da bei Ausgängen nur der interne Zustand zurückgelesen wird, nicht der des Port-Pins.

GPIORead

Diese Funktion liest den Status der Digital-Ports. Bei Ausgängen wird der Wert des Registers gelesen, nicht der des tatsächlich am Pin anliegenden Pegels.

Inquiry

Dieses Kommando fragt den ID-String des Gerätes ab. Darin enthalten ist auch die Firmware-Revision des Gerätes, nicht jedoch die Hardware-Version. Diese muß der USB-Serien-Nummer entnommen werden.

Restart

Mittels dieses Kommandos wird das Gerät auf die Werkseinstellungen zurückgesetzt. Die Werte werden immer eine unkritische Konfiguration erzielen. Amplituden werden auf das Maximum gesetzt, die Sampling-Zeit auf das Minimum. Digital-Ports werden als Eingang konfiguriert. Alle anderen Werte, z.B. für Trigger-Parameter, sind nicht endgültig spezifiziert.

Sie können eventuell bei neuen Firmware-Revisionen dem Bedarf angepaßt werden. Nach erfolgter Grundeinstellung antwortet das Gerät mit dem String „ok“+CR+LF.

Meßmodi

Übersicht

Das Gerät stellt acht verschiedene Meßmodi zur Verfügung. Alle Modi greifen auf separate Konfigurationen zurück. Davon ausgenommen sind die vier Voltmeter-Modi, die auf eine gemeinsame Konfiguration zurückgreifen. Alle Parameter werden abhängig vom Modus kontrolliert. Die Messung wird durch das Kommando „*RUN“ ausgelöst. So ist jederzeit gewährleistet, daß alle Einstellungen für den aktuellen Modus umsetzbar sind. Die angegebenen Kennungen verstehen sich als Mnemonic. Interpretiert werden sie vom Gerät als vorzeichenlose 32-Bit-Zahl. Sollen sie als ASCII-Zeichen gesendet werden, so ist die Zeichen-Folge umzukehren. Aus dem Modus „VMA1“ wird so „1AMV“.

Beschreibung	Kennung	online	offline
Voltmeter, DC	VMD0	✓	
Voltmeter, DC, Rohdaten	VMD1	✓	
Voltmeter, true RMS	VMA0	✓	
Voltmeter, true RMS, Rohdaten	VMA1	✓	
Oszilloskop, analog	OSA0	✓	✓
Datenlogger, analog	DLA0	✓	✓
Logic-Analyser auf IO-Ports	LAIO	✓	✓
Datenlogger auf IO-Ports	DLDI	✓	(✓) eingeschränkte Abtastrate

VMD0

Dieser Modus ermittelt den Gleichspannungsanteil beider Eingangskanäle. Alle relevanten Einstellungen müssen zuvor gemacht sein. Passive Tastköpfe mit einem Teilerverhältnis verschieden von 1 müssen eventuell in der Berechnung angepaßt werden. Die Eingangsimpedanz des MEphisto Scope 1, Hardware-Version 1.0 beträgt 850 kΩ. Sie ist zu erkennen am Inquiry-String „MEphisto Scope 1.0“ oder „MEphisto Scope 1“ bei älterer Firmware. Tastköpfe für Oszilloskope sind für eine Impedanz von 1 MΩ vorgesehen. Die reale Spannung muß daher wie folgt berechnet werden:

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

Die Hardware-Version 1.1 meldet sich ausschließlich mit dem Inquiry-String „MEphisto Scope 1.1“. Die Eingangsimpedanz beträgt genau 1 MΩ. Es ist daher keine gesonderte Korrektur erforderlich.

Es ist dennoch zu beachten, daß Tastköpfe mit einem Teilungsfaktor ungleich 1 oft ungenau sind.

Die zwei Rückgabewerte sind die gemessenen Spannungen in Volt mit dem Datentyp Float. Kanal 0 wird zuerst gesendet.

VMD1

Dieser Modus ist identisch mit VMD0, jedoch werden die beiden Meßwerte als Rohdaten zu 2•16 Bit übermittelt. Das Transport-Format ist Unsigned Long (32Bit). Kanal 0 wird zuerst gesendet. Die Spannung errechnet sich durch

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

VMA0

Dieser Modus ist identisch mit VMD0, jedoch werden die Meßdaten als RMS-Wert erfaßt.

VMA1

Dieser Modus ist identisch mit VMA0, jedoch werden die Meßdaten als Rohdaten erfaßt. Die Übertragung erfolgt im Format Unsigned Long.

OSA0

In diesem Modus arbeitet das Gerät als analoges Oszilloskop. Die Anzahl der aufzunehmenden Meßwerte wird durch die Speichertiefe vorgegeben. Da das Gerät die Meßdaten im Rohformat liefert, müssen sie nachträglich in Spannungen umgerechnet werden. Pro Wertepaar überträgt das Gerät eine Zahl im Format Unsigned Long. Diese 32-Bit-Integer-Zahl wird zunächst in zwei 16-Bit-Werte geteilt. Die höherwertigen 16 Bit sind die Meßwerte des Kanals 0, die niederwertigen 16 Bit stammen vom Kanal 1. Die Spannung errechnet sich wie im Modus „VMD1“ angegeben. Passive Tastköpfe mit einem Teiler-Verhältnis verschieden von 1 müssen eventuell in der Berechnung angepaßt werden. Der Korrektur-Faktor ist beim Modus „VMD0“ angegeben.

Nach dem Absetzen des Kommandos muß kontinuierlich auf Antwort gewartet werden. Sind Werte verfügbar, muß sofort mit dem Einlesen begonnen werden. Längeres Warten führt dazu, daß das Gerät die Übertragung abbricht. Es muß daher immer der gesamte Meßwert-Speicher ausgelesen werden. Nicht gelesene Werte werden nach kurzer Zeit verworfen. Die Dauer der eigentlichen Messung ist bestimmt durch die Abtastrate, die Speichertiefe und die Dauer bis zum Erkennen einer Trigger-Situation. Auch wenn eine Messung mittels Break-Kommando abgebrochen wird, sendet das Gerät Daten. Diese sind synthetisch, entsprechen jedoch in Umfang und Format dem erwarteten Datensatz.

DLA0

Der analoge Daten-Logger-Modus arbeitet ähnlich wie der Oszilloskop-Modus. Auch das Transportformat entspricht dem des Oszilloskop-Modus. Der Trigger kann zum Starten oder zum Stoppen des Datenstroms genutzt werden. Das Break-Kommando kann die gestartete Messung jederzeit beenden. Eine Meßreihe schließt immer mit den Werten 0xFFFF0000, 0x0000FFFF, 0xFFFF0000 und nochmals 0x0000FFFF ab. Dieses Muster kann bauartbedingt nicht als Meßwert auftreten.

LAIO

Der Modus Logic-Analyser auf dem GPIO-Port erfaßt rein digitale Daten. Der Analogteil des Gerätes ist inaktiv. Erfasst werden die Digital-Kanäle 0 bis 15. Trigger-Muster berücksichtigen die Kanäle 0 bis 7. Der Digitalport wird bei Aufruf dieses Modus, sofern noch nicht geschehen, in die passenden Datenrichtungen geschaltet. Die Bits 0-15 sind Eingang und werden aufgezeichnet. Die Bits 16 bis 22 sind Ausgang. Dies ermöglicht Ansteuerung und Spannungsversorgung für einen externen Pegelwandler. Das Bit 23 ist Eingang und kann als externes Trigger-Signal herangezogen werden. Eine laufende Messung läßt sich mit dem Kommando „Break“ abbrechen. Es gilt das in der Beschreibung zu dem Kommando gesagte.

Die Dauer der Messung ist abhängig von Speicherumfang, Zeitvorgabe und Dauer bis zum Triggerereignis. Das Signal des externen Triggers wird nicht mit aufgezeichnet. Es muß bei Bedarf zusätzlich auf einen der Meßkanäle gelegt werden.

Das Auslesen erfolgt in zwei Gruppen zu je 8 Bit, die untereinander synchron sind. Zwischen den Bits 0 bis 7 und den Bits 8 bis 15 besteht eine feste zeitliche Verzögerung von 2µs. Das externe Trigger-Signal hat eine maximale Latenz von 1 Sample, unabhängig von der Abtastzeit.

Als Transportformat dient Unsigned Long. Darin sendet das Gerät gepackt je zwei 16-Bit Meßwerte. Der ältere Wert ist in den höherwertigen zwei Bytes abgelegt, der neuere in den niederwertigen. Eine abgebrochene Messung erzeugt einen synthetischen Datensatz.

DLDI

In diesem Modus arbeitet das Gerät als digitaler Daten-Logger. Das Verhalten ist identisch mit dem des analogen Daten-Loggers. Das Datenformat unterscheidet sich von dem des Logic-Analysers. Das Transport-Format ist Unsigned Long und umfaßt einen einzelnen 16-Bit Meßwert. Dies stellt sicher, daß das Ende-Kennzeichen eindeutig erkennbar ist. Eine Meßreihe schließt auch in diesem Modus mit den Werten 0xFFFF0000, 0x0000FFFF, 0xFFFF0000 und 0x0000FFFF ab.

Im Offline-Modus ist die Abtastrate beschränkt auf 2,5kHz bzw. 400µs. Die Korrektur für die Zeiteinstellung erfolgt erst unmittelbar vor der Offline-Messung. Online akzeptiert das Gerät die maximal zulässige Geschwindigkeit.

Trigger-Typen, analog

Übersicht

Trigger-fähig sind alle analogen und Digitalen Modi mit Ausnahme der vier Voltmeter-Modi.

Beschreibung	Trigger-Typ	Genutzte Einstellung
Schwelle, Überschreitung	T	oberer Trigger-Level
Schwelle, Unterschreitung	t	oberer Trigger-Level
Fenster, Eintritt	W	oberer Trigger-Level, unterer Trigger-Level
Fenster, Austritt	w	oberer Trigger-Level, unterer Trigger-Level
Flanke, positiv	E	oberer Trigger-Level
Flanke, negativ	e	oberer Trigger-Level
Ableitung, positive Steigung	D	oberer Trigger-Level
Ableitung, negative Steigung	d	oberer Trigger-Level
Extern, positiv	X	-
Extern, negativ	x	-
Manuell	M	-

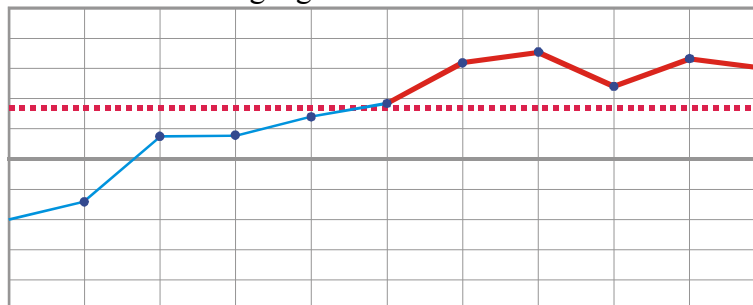
Manuell

Dieser Trigger-Modus steht für die frei laufende Messung. Sie beginnt sofort nach dem Absetzen des Kommandos. Dies ist die Standard-Vorgabe nach dem Einschalten für alle analogen und digitalen Modi.

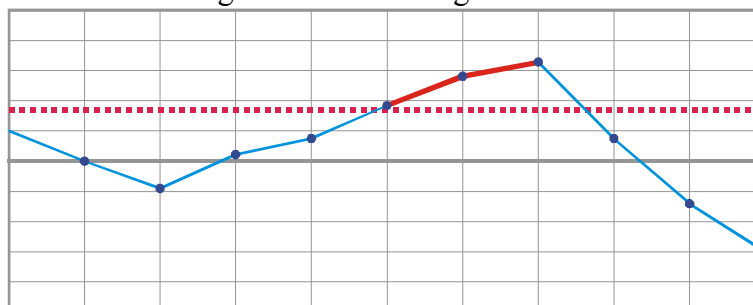
Schwellwert Über-/Unterschreitung

In diesem Modus reagiert das Oszilloskop auf das Über- bzw. Unterschreiten der vorgegebenen Schwelle. Es ist dabei unerheblich, wie die Überschreitung zustande kam. Diese Art des Triggers eignet sich besonders für zwei Fälle:

1. Verfolgen eines Einschaltvorganges



2. Detektieren einer kurzzeitigen Überschreitung

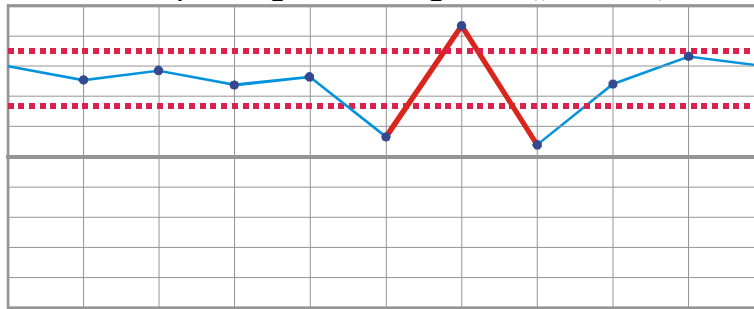


Fenster-Eintritt/-Austritt

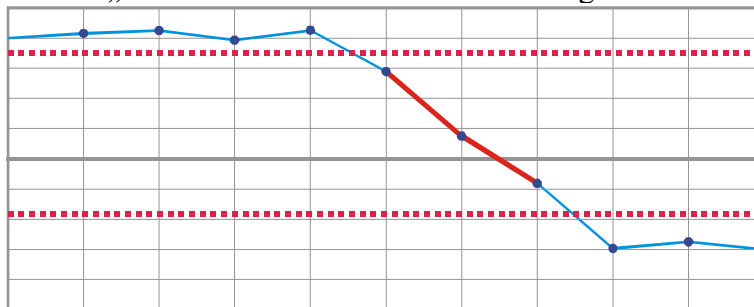
In diesem Modus wird ein Fenster mittels einer oberen und einer unteren Schwelle festgelegt. Das Trigger-Ereignis tritt ein, wenn das Signal in das Fenster eintritt oder das Fenster verläßt.

Es ist dabei egal, ob es dies an der oberen oder der unteren Schwelle tut. Die Anwendung liegt hauptsächlich in der Überwachung von Gleichspannungen.

1. Kontrolle einer Gleichspannung auf Störungen mit „Window, inside“



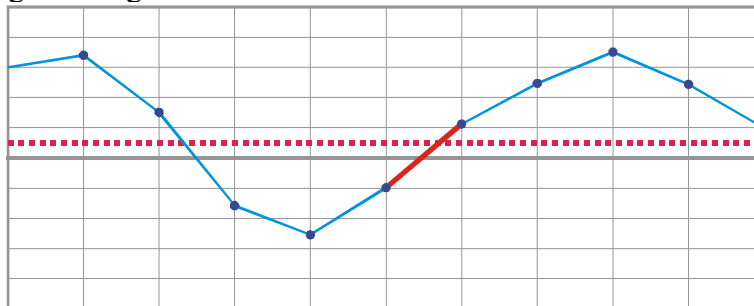
2. Detektieren eines „verbotenen Bereichs“ z.B. in der Digitaltechnik



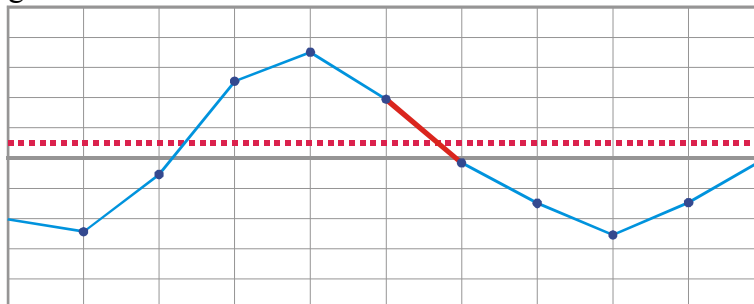
Flanke steigend/fallend

Ist dieser Modus aktiv, wird, ähnlich wie beim Schwellwert-Trigger, auf das Über- bzw. Unterschreiten der vorgegebenen Schwelle gewartet. Beim Flanken-Trigger muß jedoch die Schwelle überquert werden. Ein statisch zu hohes oder zu niedriges Signal löst den Trigger nicht aus. Der Flanken-Modus ist für sich wiederholende Wechselspannungssignale einsetzbar, um stehende Bilder zu erzeugen.

1. Darstellung der steigenden Flanke



2. Darstellung der fallenden Flanke



Steigungstrigger positiv/negativ

Dieser Trigger-Modus ist für ganz spezielle Fälle dynamischer Signale vorgesehen. Er ist nahezu unabhängig von der tatsächlichen Spannung. Es wird nur die Veränderung des Signals kontrolliert. Steigt oder fällt das Signal schneller als vorgegeben, so wird der Trigger ausgelöst.

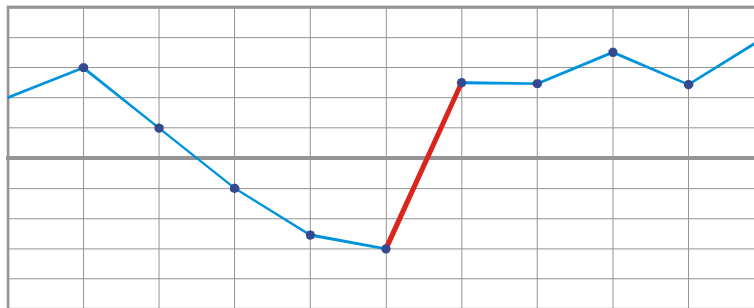
Die Steigung wird angegeben als Spannung/Sample-Zeit. Die Spannungsangabe erfolgt in Volt. Die Sample-Zeit muß errechnet werden aus dem Quotienten Time/Memory Depth.

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

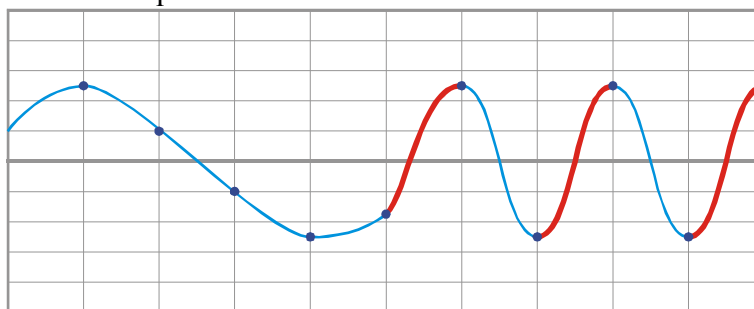
Die Kontrolle der Steigung ist hauptsächlich in zwei Anwendungsbereichen wichtig. Zum einen ist dies die Prozeßkontrolle in der Chemie, der Biologie etc. Hier sind selbst große Schwankungen oft normal, plötzliches starkes Ansteigen jedoch Zeichen für einen Fehler. Zum anderen kann mit diesem Modus eine Frequenzerkennung realisiert werden. Hier sei besonders das Detektieren von transienten Störungen innerhalb stationärer Signale erwähnt.

Das MEphisto Scope 1 hat einen begrenzten Meßbereich. Um nahe des oberen oder unteren Randes keine Schwankungen unkontrolliert aus dem Meßbereich heraus treten zu lassen, löst der Trigger auch aus, wenn die aktuelle Spannung plus des Anstiegs den Meßbereich über- bzw. unterschreiten würde.

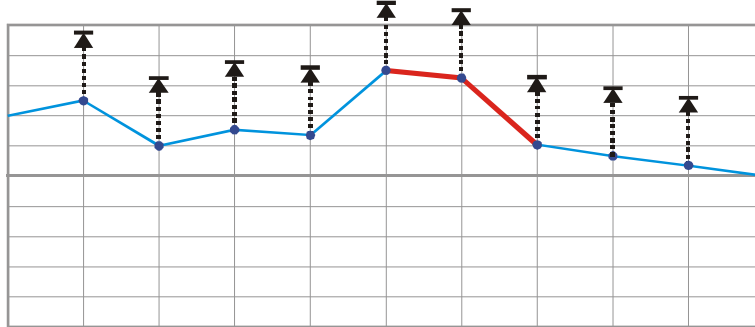
1. Erkennen eines kritischen Zustandes



2. Detektieren einer Frequenz

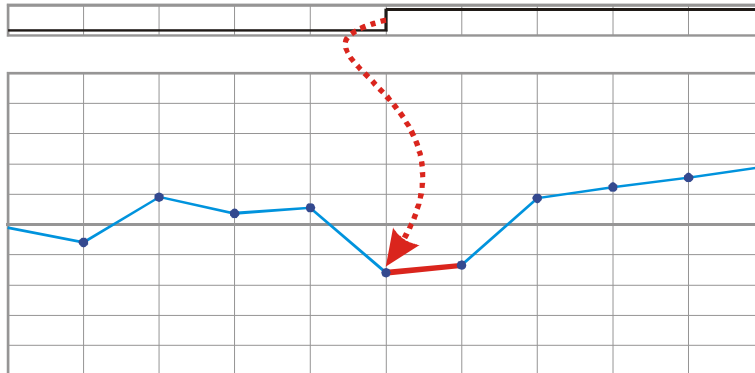


3. Potentielles Überschreiten des Meßbereichs beim nächsten Meßwert



Extern, steigende/fallende Flanke

Ist dieser Modus aktiv, wird, ähnlich wie beim Flanken-Trigger, auf eine steigende bzw. fallende Flanke gewartet. Ausschlaggebend ist in diesem Fall jedoch nicht eines der analogen Signale sondern das digitale Signal an I/O Port D23 (Pin 26). Wie beim analogen Flanken-Trigger löst nur eine logische Flanke den Trigger aus. Ein statisches Signal wird ignoriert. Der externe Trigger ist zur Synchronisierung mit anderen Geräten vorgesehen. Bitte prüfen Sie die Kompatibilität von Trigger-Quelle und dem MEphisto Scope 1. Moderne Signalgeneratoren erzeugen meist ein 5V-TTL-Signal, das Sie als Synchron-Impuls nutzen können.



Trigger-Typen, digital

Übersicht

Beschreibung	Trigger-Typ	Genutzte Einstellung
Muster	P	Bit-Muster
Extern, positiv	X	-
Extern, negativ	x	-
Manuell	M	-

Bit-Muster

Zur Erkennung eines Trigger-Ereignisses werden die acht niederwertigsten Bit des 16-Bit breiten Meßsignals ausgewertet. Das Bit-Muster des digitalen Triggers wird in der Angabe des Kanals übergeben. Drei der vier Bytes werden für die Kodierung der fünf möglichen Zustände genutzt. Das Byte 0 ist im Digital-Modus ohne Bedeutung.

Byte	Beschreibung	Bit = 0	Bit = 1
0	-	-	-
1	Relevanz	Bit wird ignoriert	Bit wird ausgewertet
2	Dynamik	Bit wird als Pegel ausgewertet	Bit wird als Flanke ausgewertet
3	Pegel	Pegel=Low oder Flanke=fallend	Pegel=High oder Flanke=steigend

Manuell

Dieser Trigger-Modus steht für die frei laufende Messung. Sie beginnt sofort nach dem Absetzen des Kommandos. Dies ist die Standard-Vorgabe nach dem Einschalten für alle triggerbaren Modi.

Muster

In diesem Modus triggert der Logic-Analyser auf ein vorgegebenes Bit-Muster. Das Muster wird nur für die unteren acht Bit ausgewertet. Jede Bit-Vorgabe kann einen der folgenden fünf Zustände annehmen:

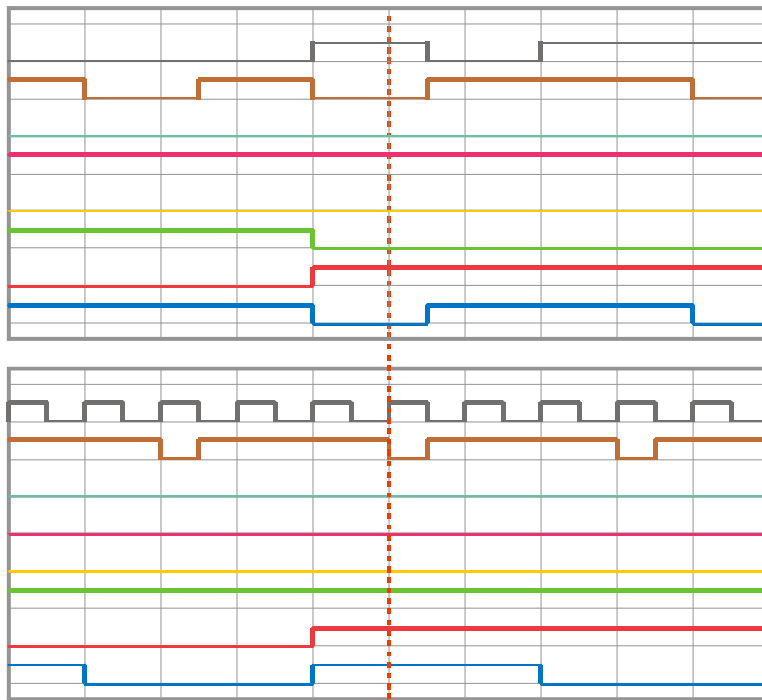
- High
- Low
- Edge, rising
- Edge, falling
- don't care

Die Zustände werden wie in der obigen Tabelle beschrieben in 3 Bytes übergeben.

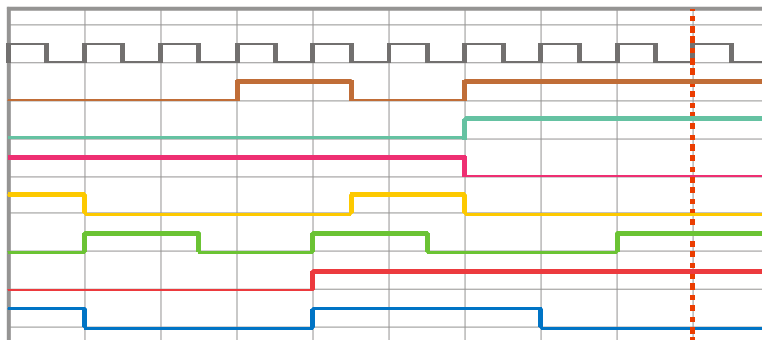
Ein sicherer Triggerzustand wird meist erst durch die Kombination mehrerer Kriterien erreicht. Beispielhaft betrachten wir einen Speicherzugriff eines Prozessors. So kann z.B. auf Kanal 7 ein Takt anliegen, dessen steigende Flanke relevant ist. Kanal 6 zeigt einen Schreibimpuls, der mit der fallenden Flanke aktiv wird. Die Kanäle 0 bis 5 können in diesem Szenario Adressen sein. Die verbleibenden Kanäle 8 bis 15 repräsentieren die Daten und tragen nicht zum Triggerereignis bei.

Soll herausgefunden werden, welcher Wert auf die Speicherstelle 7_{hex} (000111_{bin}) geschrieben wird, so muß das Trigger-Muster RF000111 angegeben werden. Die Kanäle 8 bis 15 zeigen die Daten an. Im Beispiel ergibt sich ein Wert von 10010010_{bin} (92_{hex}). Ebenfalls ersichtlich ist, daß Setup- und Hold-Zeiten vermutlich eingehalten werden.

Bei diesem Beispiel sei abschließend darauf hingewiesen, daß es in der Praxis nur bei Verwendung eines sehr langsam getakteten Emulators oder einem Microcontroller im Single-Step-Betrieb möglich ist. Es soll nur zur Veranschaulichung der Möglichkeiten dienen.



Ein Zustandsautomat bzw. eine Ablaufsteuerung kann daraufhin getestet werden, ob ein verbotener Zustand angenommen wird, indem dieser Zustand als Trigger-Kriterium angegeben wird. Im Beispiel ist das Trigger-Muster R1100110. Die Taktflanke sollte mit angegeben werden, um definiert den Zustand selbst, und nicht zufällig das asynchrone Umschalten zwischen zwei anderen Zuständen zu erkennen. Der Verlauf vor dem Ereignis zeigt dann, wie es zum Fehler kam.



Diese beiden völlig verschiedenen Anwendungen zeigen, daß das Triggern auf ein so frei definierbares Muster ein mächtiges Werkzeug darstellt. Dennoch ist zu sagen, daß insbesondere die Kombination von nicht exakt zeitlich korrelierten Signalen kaum zuverlässige und reproduzierbare Triggerereignisse verspricht. Die Anzahl der herangezogenen Kanäle sollte unter diesem Aspekt so gering wie möglich sein. Andererseits steigt mit abnehmender Zahl von Kriterien die Anzahl ungewollter Triggerereignisse. Es gilt daher einen Kompromiß zu finden, der eine möglichst hohe Auslösesicherheit bietet.

Extern, steigende/fallende Flanke

Ist dieser Modus aktiv, wird, ähnlich wie beim Muster-Trigger, auf eine steigende bzw. fallende Flanke gewartet. Ausschlaggebend ist in diesem Fall jedoch nicht einer der Meßkanäle sondern das Signal an I/O Port D23 (Pin 26). Wie beim Flanken-Trigger löst nur eine logische Flanke den Trigger aus. Ein statisches Signal wird ignoriert. Der externe Trigger ist zur Synchronisierung mit anderen Geräten vorgesehen. Bitte prüfen Sie die Kompatibilität

von Trigger-Quelle und dem MEphisto Scope 1. Moderne Signalgeneratoren erzeugen meist ein 5V-CMOS-Signal, das sich als Synchron-Impuls nutzen läßt.

GPIO-Richtungsangabe

Richtung	Bit-Wert
Eingang	0
Ausgang	1

Das „MEphisto Scope 1“ ist mit 24 Digital-I/Os ausgestattet. In den analogen Modi kann jeder Kanal unabhängig als Eingang oder Ausgang konfiguriert werden. Nach dem Einschalten sind alle Kanäle als Eingänge konfiguriert.

Der Digital-Kanal DIO_24 (Pin 26) kann als externer Triggereingang genutzt werden. Sobald dieser Trigger-Modus gesetzt ist, wird der Kanal exklusiv für den externen Triggerbetrieb reserviert. In diesem Fall ist der Kanal immer als Eingang geschaltet und lässt sich auch nachträglich nicht umprogrammieren. Erst nach Setzen eines anderen Trigger-Modus ist der Kanal wieder frei verfügbar.

USB-Treiber

Der USB-Baustein des MEphisto Scope 1 ist der FT245BM der Firma FTDI. Aktualisierte Treiber für Windows, Windows CE, Linux etc. werden dort zum Download bereit gestellt. Die Device ID des MEphisto Scope 1 lautet „DCD0“. Bitte beachten Sie, daß diese ID von keinem der Original-Treiber unterstützt wird. Die Firma Meilhaus Electronic wird angepaßte Treiber nach Bedarf auf der Support-Seite des MEphisto Scope 1 bereit stellen.

Kommunikation unter Windows

Der von der Firma Meilhaus Electronic bereitgestellte Treiber ist an die Geräte ID angepaßt. Der Treiber sorgt dafür, daß Daten in ausreichend kurzer Zeit vom Gerät in den Puffer gelesen werden. Bei normalen Kommandos kann bei einmal etablierter Kommunikation kein Zeichen verloren gehen. Erfahrungswerte für längere Wartezeiten beim Auslesen des Treiber-Puffers existieren nicht. Bitte beachten Sie dazu auch die Anmerkungen.

- | | |
|------------------------------------|-------------------|
| 1. Öffnen des Gerätes: | FT_OpenEX |
| 2. Kommunikation | |
| a. Löschen des Puffers (optional): | FT_Purge |
| b. Daten schreiben: | FT_Write |
| c. Warten auf Antwort: | FT_GetQueueStatus |
| d. Zeichen lesen: | FT_Read |
| 3. Schließen des Kanals: | FT_Close |

Kommunikation unter Linux

In vielen Linux-Distributionen ist ein Treiber für den USB-Baustein des MEphisto Scope 1 enthalten. Bitte beachten Sie, daß die Device ID des Gerätes „DCD0“ lautet und von diesem Standard-Treiber nicht unterstützt wird. Ein für das Scope 1 angepaßter Treiber existiert derzeit nicht.

Ein anderer Ansatz ist, das Gerät über den USB-Treiber des Kernel anzusprechen. Hierbei ist zu beachten, daß der USB-Baustein fortlaufend 64-Byte-Pakete sendet, in denen je 2 Byte Status-Informationen enthalten sind.

Anmerkungen

Bei der Entwicklung der PC-Software für Windows sind einige Treiber- oder Chip-Anomalien aufgetreten, die teilweise unter Linux ihre Entsprechung finden. In wie weit sie auf andere Programmiersprachen übertragbar sind, ist der Firma Meilhaus Electronic nicht bekannt. Sie sind nur als Hinweis zur Fehlersuche aufgeführt.

Nach dem Öffnen des Kanals empfiehlt es sich, als ersten Befehl das sehr langsame Kommando „*IDN?“ abzusetzen. Der Treiber reicht bei der ersten Kommunikation offenbar erst nach einer gewissen Verzögerung die Daten weiter. Bei schnellen Antworten des Gerätes können auf manchen Computern sofort oder später Zeichen verloren gehen. Eine Systematik ist nicht erkennbar. Dieser Fehler zeigt sich sowohl bei langsamen als auch bei schnellen Prozessoren und ist unabhängig vom Betriebssystem. Er tritt verstärkt bei mehrmaligem Anschließen Gerätes an den USB auf. Ist die Kommunikation einmal geglückt, gibt es keine Einschränkungen mehr.

Die Funktion FT_GetQueueStatus des Windows-Treibers gibt die Anzahl der im Empfangspuffer vorhandenen Zeichen nur bis zu einem Wert von 63488 Byte korrekt an. Diese Grenze wird nicht überschritten. Dieser Fehler tritt in der Praxis bei der Antwort von

Oszilloskop und Logic-Analyser mit großen Speichertiefen auf. Ebenso kann er in den Logger-Modi bei hohen Abtastraten vorkommen. Wird der Wert 63488 erreicht, muß spätestens mit dem Lesen begonnen werden, da der Treiber-Spooler insgesamt nur 65535 Zeichen vom Gerät entgegen nimmt. In den Logger-Modi sollte bereits bei einem geringeren Wert gelesen werden, da kontinuierlich weitere Daten eintreffen.