



Referenz-Handbuch Transientenrecorder

LTT Labortechnik Tasler GmbH
Friedrich-Bergius-Ring 15
D-97076 Würzburg
www.tasler.de

Tel. +49-931-359 61-0
Fax +49-931-359 61-50

Dieses Handbuch ist nach bestem Wissen erstellt worden. Die in diesem Handbuch enthaltenen Angaben und Daten sind ohne Gewähr und können ohne weitere Benachrichtigung geändert werden. Die in diesem Handbuch beschriebene Software wird auf Basis eines Software-Lizenzvertrages geliefert.

Ohne ausdrückliche schriftliche Erlaubnis darf kein Teil dieser Unterlagen für irgendwelche Zwecke vervielfältigt oder übertragen werden, unabhängig davon, auf welche Art und Weise oder mit welchen Mitteln, elektronisch oder mechanisch, dies geschieht.

© Copyright 2007 by Labortechnik Tasler GmbH
Friedrich-Bergius-Ring 15
D-97076 Würzburg
Germany
E-Mail: info@tasler.de
Internet: www.tasler.de

Inhaltsverzeichnis

Einleitung.....	5
TEIL 1: Beschreibung der Hardware	6
1. Überblick.....	6
2. LTT-184 und LTT-186.....	7
I. Frontansicht	7
II. Rückansicht	8
3. 19" - Einschub	10
4. Technische Daten.....	12
TEIL 2: Kaskadierung von LTT-184	13
TEIL 3: LTTview (V. 4.2)	15
1. Installation.....	15
2. Die Bildschirmoberfläche	16
3. Online und Offline arbeiten	16
4. Konfiguration.....	17
4.1 ADC.....	17
4.2 Channels	19
4.3 Display	26
4.4 Trigger	29
4.5 Cursor	31
4.6 Files.....	33
5. Grafische Darstellung	34
5.1 Funktionen im Display.....	34
5.2 Die Symbolleiste	36
a) Allgemeine Funktionen.....	36
b) Displayfunktionen:.....	37
5.3 Ergänzende Angaben	38
TEIL 4: Beschreibung der DLL LTT2API V1.3.0	39
Fundamental concepts	40
LTTInit.....	44
LTTDeInit	45
LTTTrigger	46
LTTTrigger_SetPreTrigger.....	48

LTTTrigger_SetReactivationTime	49
LTTAnalogChannel	50
LTTDigitalChannel	52
LTTTRPMChannel	53
LTTOutputImpedance	54
LTTSampleTime	55
LTTGetRAMBufferSettings	56
LTTRAMBufferSettings	57
LTTFileBufferSettings	58
LTTSTREAMSettings	59
LTTBufferReset	60
LTTUploadParameters	61
LTTResetParameter	62
LTTStart	63
LTTStop	64
LTTCheckTrigger	65
LTTTransfer2Buffers	66
LTTDataTransferStatus	67
LTTDataTransferBreak	68
LTTData_GetCounter	69
LTTData_GetBlock	70
LTTData_BlockDone	71
LTTGetCalibTemperature	72
LTTGetChannelIDSequence	73
LTTGetChannelSequenceLength	74
LTTGetResolution	75
LTTGetChannelRange	76
LTTGetTemperature	77
LTTGetDLLInfo	78
LTTGetDeviceInfo	79
LTTGetDevices	80
LTTSetInternReference	81
LTTGetChannelConverter	82
Index	83

Einleitung

Herzlichen Glückwunsch zu Ihrer Entscheidung für ein Meßsystem von Labortechnik Tasler. Von allen derzeit am Markt erhältlichen Systemen haben Sie sich für eines der genauesten und flexibelsten entschieden. Sie mögen gerade erst anfangen, sich mit dem System auseinander zusetzen. Oder aber Sie kennen die LTT-Systeme schon in- und auswendig und nutzen dieses Handbuch als Nachschlagewerk und Referenzhandbuch.

In beiden Fällen werden Sie staunen, welche Möglichkeiten sich in den schwarz-roten Geräten mit den kompakten Außenmaßen verbergen. Apropos „verbergen“. Damit Ihnen die vielen Features nicht verborgen bleiben, wird dieses Handbuch ständig weiter an die aktuellsten Software- und Hardwareentwicklungen angepasst. Neue Teile des Handbuchs können Sie einfach im Austausch für die alten in die entsprechenden Rubriken Ihres persönlichen Handbuchs heften.

In den einzelnen Kapiteln des Handbuchs werden Hardware- und Softwarelösungen beschrieben, die im Paket oder auch einzeln von LTT angeboten werden.

In **Teil 1** finden Sie eine detaillierte Beschreibung und Darstellung der Hardware mit sämtlichen Gerätekompontenten, soweit sie von LTT entwickelt und hergestellt wurden.

Die Systeme LTT-184 und LTT-186 sind kaskadierbar. Sie haben also die Möglichkeit, auch über mehr als 16 Kanäle zu messen und die so gewonnenen Meßdaten zu verarbeiten. **Teil 2** des Handbuchs beschreibt die Vorgehensweise, wie man die Systeme kaskadiert.

Teil 3 bietet Ihnen eine ausführliche Darstellung der Software LTTview. LTTview ist zum Betrieb der Hardware nicht unbedingt notwendig. LTTview ist jedoch ein speziell für die Hardware entwickeltes Management-Tool, das die Arbeit mit der LTT-Hardware sehr erleichtert. Alternativ zur Verwendung von LTTview ist auch die Nutzung von eigener Software oder Software von Drittherstellern möglich, soweit sie auf die entsprechenden, von LTT bereitgestellten Schnittstellen aufsetzen.

Die Beschreibung der Schnittstelle LTT2API ist Thema von **Teil 4**. Hier werden alle Informationen vermittelt, die Sie benötigen, um unsere DLL zur Anbindung an Software von Drittherstellern zu nutzen. Durch die variable Schnittstelle ergeben sich unendlich viele Möglichkeiten, die Systeme von LTT in vorhandene oder neu geplante Umgebungen einzubinden. Die Unterstützung von Standards wird auch in Zukunft hohe Priorität bei der Weiterentwicklung der LTT Produkte haben.

Kein Handbuch der Welt kann die Arbeit mit einer neuen Hardware und Software ersetzen. Sie kann sie nur wesentlich erleichtern. Sollten Sie bei der Arbeit mit unseren Produkten auf neue Fragen stoßen, rufen Sie uns bitte an. Wie in allen Lebenslagen ist es auch bei der Meßtechnik sehr wahrscheinlich, dass andere Anwender ähnlich Anforderungen haben und auf ähnliche Hindernisse stoßen wie Sie. Mit Ihrem Feedback geben uns also auch wertvolle Hilfestellung für die Weiterentwicklung dieser Dokumentation

Wir wünschen Ihnen viel Erfolg beim Einsatz der LTT Produkte!

LTT GmbH

Würzburg, 07. Mai 2007

Teil 1:

Beschreibung der Hardware

1. Überblick

LTT stellt zur Zeit drei verschiedene Systeme her, die auf derselben Technologie beruhen: LTT-184, LTT-186 und LTT-19X.

Die Systeme der Serie **LTT-184** sind mobile Transientenrekorder zur Datenerfassung und -kontrolle. LTT-184 wurde speziell für sehr schnelle Lang- und Kurzzeitmessungen entwickelt. Enthalten sind 8 bzw. 16 differenzielle Analog- Eingangskanäle mit individuell einstellbaren Eingangsströmen und einer Abtastrate von 1 kHz bis 20 MHz. Ein separater Takteingang bietet die Möglichkeit, einen externen RPM-Takt zu vermessen oder aber nach einem externen Taktsignal zu messen. So kann LTT-184 beispielsweise auch für umdrehungsbezogene Messungen verwendet werden.

Der **LTT-186** entspricht dem LTT-184, enthält aber zusätzlich eine interne Festplatte. Entweder als ein Stand-Alone Gerät oder zusammen mit einem Notebook beweist sich das LTT-186 als ein multifunktionales Meßsystem der Spitzenklasse. Ausgestattet mit einer 9 bzw. 18 GB Festplatte und einer Stoßfestigkeit von bis zu 65G (bei der 9 GB Festplatte), kann das System auch problemlos unter schwierigen und rauen Bedingungen mobil eingesetzt werden.

Eine neu entwickelte Bauform ist die LTT-Serie im 19-Zoll Format: **LTT-19X**. Zwei Geräte der Serien LTT-184 sind eingebaut in einen Einschub, der zum Einbau in einen 19-Zoll Schrank geeignet ist. Alle Vorteile der LTT-Produkte sind hier also gebündelt mit den Vorteilen durch die standardisierte Bauform. Mit insgesamt 32 analogen, differenziellen Eingangs-Kanälen ist das System prädestiniert für den Einsatz in Prüftürmen.

2. LTT-184 und LTT-186

I. Frontansicht



2. LEDs "Power",
"Trigger" und "HD"

4. Analog Out
(nicht belegt)

1. Schalter zum Start der
Messung (nur für LTT-186)

3. BNC Eingangs-
buchsen

1. Schalter zum Start der Messung (nur für LTT-186)

Das System LTT-186 ist für Messungen ohne Anschluß an einen PC geeignet. Eine interne Festplatte erlaubt die Speicherung der Messdaten. Mit dem roten Start-Knopf wird die Messung ausgelöst. Nach der Messung (z.B. in Fahrzeugen) wird das System an einen PC angeschlossen und die Daten können von der Festplatte ausgelesen werden.

2. LED Anzeigen

Power: Die LED leuchtet, wenn das Gerät betriebsbereit ist

Trigger: Die LED leuchtet, wenn ein Trigger definiert wurde und das System auf das Triggerereignis wartet. Sobald die Triggerbedingung erfüllt ist, erlischt die LED.

HD: Die LED blinkt, solange Daten vom Messgerät zum PC oder zur internen Festplatte (nur LTT-186) übertragen werden.

3. 16 bzw. 32 BNC-Eingangsbuchsen

Das 8-Kanal-System verfügt über 16 BNC-Buchsen, das 16 Kanal-System über 32. Jedem Kanal sind 2 Eingangsbuchsen (BNC+ und BNC-) zugeordnet. Die Kanäle sind aufsteigend durchnummeriert und werden so auch über die DLL angesprochen.

Zwischen den Buchsenreihen zeigen Leuchtdioden den Schaltzustand der Kanäle an.

Untere Diode: Diode A

Obere Diode: Diode B

Schaltzustand:

Single Ended + (SE+): LED A leuchtet, B nicht

Single Ended - (SE-): LED A leuchtet nicht, B leuchtet

Differential Ended (DE): Beide leuchten

Off (GND): Beide LEDs sind aus

Bei 8 Kanal-Geräten befinden sich die Eingangsbuchsen an der Vorderseite des Gehäuses. Bei 16-Kanal-Geräten sind die ersten 8 Kanäle auf der Vorderseite und weitere 8 Kanäle auf der Rückseite des Gehäuses installiert. Jeder BNC-Eingang führt auf der Schirmung die Kanalmasse und auf dem Innenleiter das Signal.

4. Analog Out

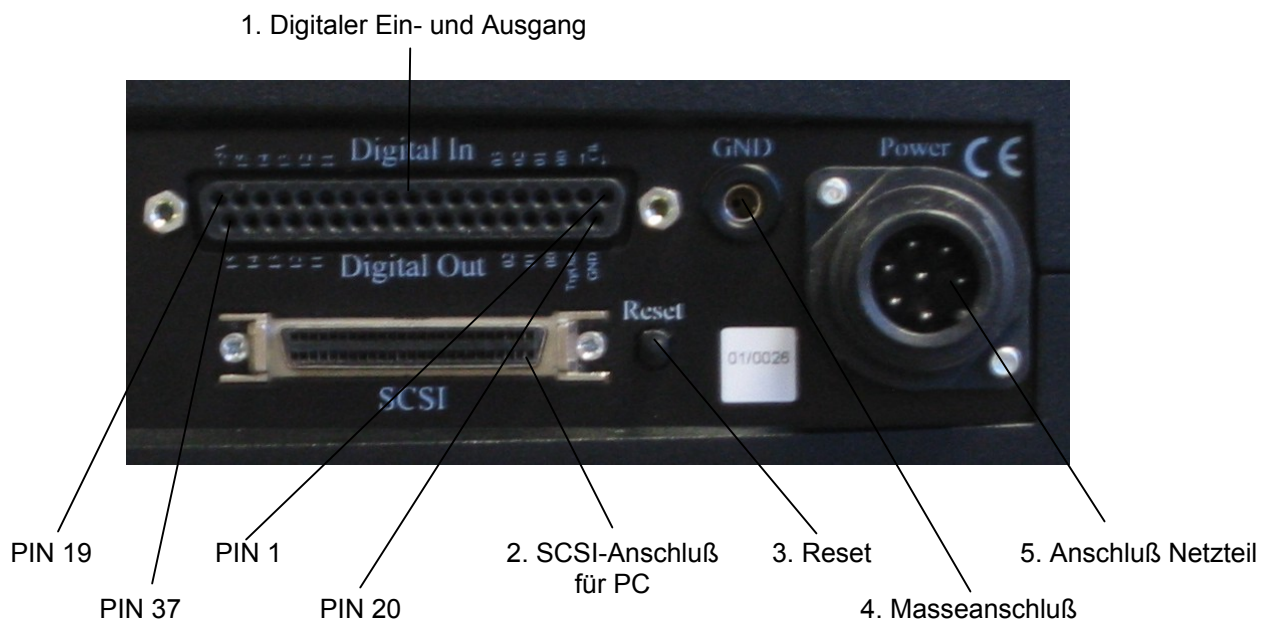
Dieser Anschluß ist nicht belegt.

II. Rückansicht

Die Abbildung zeigt ein 16 Kanal-Gerät:



Detailansicht:



1. SUB-37 Digital-Input:

Über den 37-poligen Stecker können digitale Eingänge angeschlossen werden. Eine Triggerung ist auf den Bits 9 bis 15 (FW6xxx), Bits 10 bis 15 (FW>700), und Bits 9 bis 15 (FW>719) möglich. Es gibt eine Zeitverzögerung zwischen den Analogen Kanälen und der Digital Signal, die abhängig von Abtastrate und Anzahl der aktiven Kanälen. Geräte mit dem gesetzte "Digital Delay" Feature Bit haben diese Zeitverzögerung nicht. Die Verzögerung wird automatisch korrigiert mit dem DLL LTT2API ab Version 1.3.0.3 RC11.

Steckerbelegung:

Achtung: Alle PINs sind TTL-kompatibel und ungeschützt, d.h. alle Ein- und Ausgänge sind mit maximal 5 V zu betreiben. Alle Ausgänge dürfen maximal mit 10 mA belastet werden.

PIN	Bedeutung	
1	Clock In	(TTL, ungeschützt)
2	Clock Out	(TTL, ungeschützt)
3	Digital In Bit 00	(TTL, ungeschützt)
4	Digital In Bit 01	(TTL, ungeschützt)
5	Digital In Bit 02	(TTL, ungeschützt)
6	Digital In Bit 03	(TTL, ungeschützt)

7	Digital In Bit 04	(TTL, ungeschützt)
8	Digital In Bit 05	(TTL, ungeschützt)
9	Digital In Bit 06	(TTL, ungeschützt)
10	Digital In Bit 07	(TTL, ungeschützt)
11	Digital In Bit 08	(TTL, ungeschützt)
12	Digital In Bit 09	(TTL, ungeschützt)
13	Digital In Bit 10	(TTL, ungeschützt)
14	Digital In Bit 11	(TTL, ungeschützt)
15	Digital In Bit 12	(TTL, ungeschützt)
16	Digital In Bit 13	(TTL, ungeschützt)
17	Digital In Bit 14	(TTL, ungeschützt)
18	Digital In Bit 15	(TTL, ungeschützt)
19	+5V (max. 40 mA)	(TTL, ungeschützt)
20	Ground (max. 40 mA)	(TTL, ungeschützt)
21	Trigger Out	(TTL, ungeschützt)
22	Digital Out Bit 00	(TTL, ungeschützt)
23	Digital Out Bit 01	(TTL, ungeschützt)
24	Digital Out Bit 02	(TTL, ungeschützt)
25	Digital Out Bit 03	(TTL, ungeschützt)
26	Digital Out Bit 04	(TTL, ungeschützt)
27	Digital Out Bit 05	(TTL, ungeschützt)
28	Digital Out Bit 06	(TTL, ungeschützt)
29	Digital Out Bit 07	(TTL, ungeschützt)
30	Digital Out Bit 08	(TTL, ungeschützt)
31	Digital Out Bit 09	(TTL, ungeschützt)
32	Digital Out Bit 10	(TTL, ungeschützt)
33	Digital Out Bit 11	(TTL, ungeschützt)
34	Digital Out Bit 12	(TTL, ungeschützt)
35	Digital Out Bit 13	(TTL, ungeschützt)
36	Digital Out Bit 14	(TTL, ungeschützt)
37	Digital Out Bit 15	(TTL, ungeschützt)

2. SCSI-Anschluß für PC (50 polig)

Zur Anbindung des Systems an PCs- und Notebooks. Möglich ist auch die Verwendung von FireWire-SCSI bzw. USB2-SCSI Adaptionen, die auf die SCSI-Buchse aufgesetzt werden. Über die SCSI-Schnittstelle kann das Meßsystem konfiguriert und können die Messdaten vom System zum PC übertragen werden.

3. Reset

Knopf zum Neustart des Systems. Bitte darauf achten, dass stets zuerst das System, erst dann der angeschlossene PC neu gestartet wird.

4. Masseanschluß (Ground, GND)

Buchse für Bananenstecker zur Verbindung des Meßsystems mit der Erde (Masse). **Achtung:** Die Verbindung zur Erde ist identisch zur PC-Erdung. Alle Meßergebnisse sind nur gültig, wenn Masse definiert ist, d.h. eine ausreichende Steckverbindung zwischen Erde und Messgeräte-Masse (GND) existiert.

5. Anschluß für das mitgelieferte Netzteil

Der Schraub-Anschluß erlaubt eine äußerst sichere und stabile Verbindung des Meßsystems mit der Stromversorgung.

3. 19-Zoll-Einschub

Im folgenden werden die Anschlüsse und deren Belegung, die der 19"-Einschub zur Verfügung stellt beschrieben. Alle Anschlüsse liegen auf der Rückseite des Einschubs (s. Abb. 1).

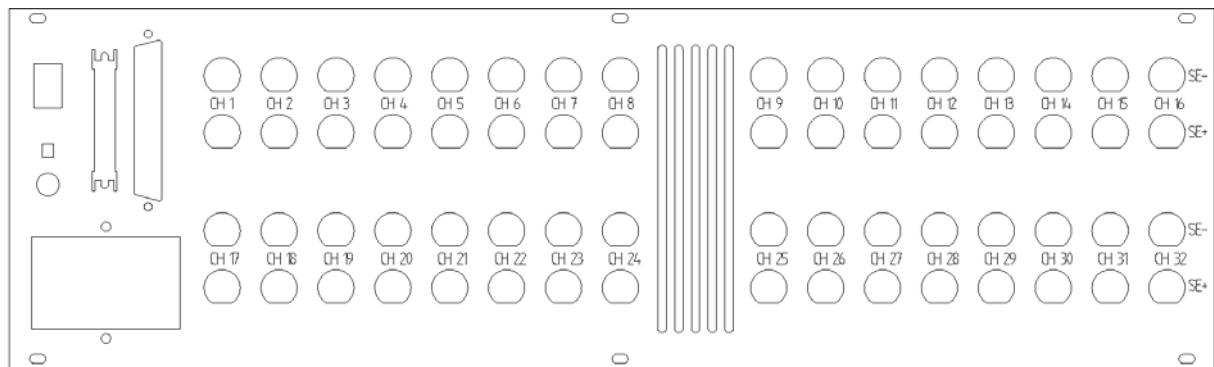


Abbildung 1: Rückansicht des 19"-Einschubs

Folgende Anschlüsse stehen zur Verfügung:

1. Netzstecker-Buchse für die 230V Versorgung:

An der Buchse gibt es auch einen Schalter zur Unterbrechung der Stromversorgung. Dieser muss geschaltet sein, bevor man an der Gehäusevorderseite den Schalter benutzen kann. Weiterhin sind in diese Buchse auch eine 230V-Sicherung eingebaut, die von extern zugänglich ist.

2. GND-Buchse:

An dieser Buchse lässt sich die Messgerätemasse festlegen.

3. IEEE1394 Anschluss; Kabeldurchführung für Fire Wire Kabel:

4. SCSI-Anschluss:

Der SCSI-Stecker wird derzeit nicht benutzt, ist aber dennoch bestückt. Er ist intern nicht mit den LTT-Geräten verbunden.

5. SUB-37 Digital-Input:

Über den 37-poligen Stecker können digitale Eingänge angeschlossen werden. Am 19"-Einschub steht 1 digitaler Kanal mit 15 Eingangsbits (Bit 1 bis 15) zur Verfügung. Das Bit 0 des Kanals ist für interne Zwecke reserviert. Eine Triggerung ist nur auf den Bits 9 bis 15 möglich.

Steckerbelegung:

PIN	Bedeutung
1	Nicht benutzt
2	Nicht benutzt
3	Nicht benutzt
4	Digitaler Eingang, Bit 1
5	Digitaler Eingang, Bit 2
6	Digitaler Eingang, Bit 3
7	Digitaler Eingang, Bit 4
8	Digitaler Eingang, Bit 5
9	Digitaler Eingang, Bit 6
10	Digitaler Eingang, Bit 7
11	Digitaler Eingang, Bit 8

12	Digitaler Eingang, Bit 9
13	Digitaler Eingang, Bit 10
14	Digitaler Eingang, Bit 11
15	Digitaler Eingang, Bit 12
16	Digitaler Eingang, Bit 13
17	Digitaler Eingang, Bit 14
18	Digitaler Eingang, Bit 15
19	+5V
20	Nicht benutzt
21	Nicht benutzt
22	Nicht benutzt
23	Nicht benutzt
24	Nicht benutzt
25	Nicht benutzt
26	Nicht benutzt
27	Nicht benutzt
28	Nicht benutzt
29	Nicht benutzt
30	Nicht benutzt
31	Nicht benutzt
32	Nicht benutzt
33	Nicht benutzt
34	Nicht benutzt
35	Nicht benutzt
36	Nicht benutzt
37	Nicht benutzt

6. 64 BNC-Buchsen:

Jedem Kanal sind 2 Eingangsbuchsen (BNC+ und BNC-) zugeordnet. Die Kanäle sind aufsteigend durchnummeriert und werden so auch über die DLL angesprochen. Eine externe Anzeige über den Schaltzustand der Kanäle gibt es nicht.

Jeder BNC-Eingang führt auf der Schirmung die Kanalmasse und auf dem Innenleiter das Signal.

4. Technische Daten

Modell	LTT-184-8ch	LTT-184-16ch	LTT-186-8ch	LTT-186-16ch	LTT-19x
Kanalzahl:	8 Differenz	16 Differenz	8 Differenz	16 Differenz	32 Differenz
AD-Wandler:	8 Stück, simultan	16 Stück, simultan	8 Stück, simultan	16 Stück, simultan	32 Stück, simultan
Stand-alone:	Nein	Nein	9 bzw. 18 GB HD	9 bzw. 18 GB HD	Nein
Speicher:	128 MB	128 MB	128 MB	128 MB	2 mal 128 MB
Externer Takt / RPM:	Ja / Nein	Ja / Ja	Ja / Nein	Ja / Ja	Nein / Nein

Vorverstärker				
Eingangskopplung	Option A: differenziell oder single-ended massebezogen, einzeln schaltbar.		Option B: differenziell oder single-ended massebezogen, einzeln schaltbar	
	AC	DC	DC (1:1)	DC (1:10)
Eingangsimpedanz	10 nF in 0,1 M Ω , 150 pF	0,1 M Ω , 150 pF	0,1 M Ω , 150 pF	1 M Ω , 15 pF
Schutz	± 200 V differenziell, ± 100 V auf jedem Eingang			
Meßbereiche	$\pm [50V, 20V, 10V, 5V, 2V, 1V]$ pro Kanal wählbar			
Gleichtaktspannung	$\leq (\text{Meßbereich} \times 7.7) \leq 100V$			
Verstärkungsdrift	≤ 25 ppm/ $^{\circ}C$			
Offsetdrift	≤ 0.2 LSB/ $^{\circ}C$			
Digitaler Eingang	16 Bit, TTL			
Signalbandbreite	6.5 MHz (-3dB)			
Crosstalk	< Meßgenauigkeit			
Abtastrate	1 kHz bis 2,5 MHz (16 Bit); bis 20 MHz (12 Bit)			
Taktgeber	Intern, extern			

Trigger	
Trigger	Pre- / Posttrigger (Flanke, fehlende Ereignisse uvm.)
Quelle	Analoge und digitale Eingänge

Gesamtgenauigkeit		
Zeitachsenabweichung	≤ 20 ppm bei interner Zeitgebung	
Eigenrauschen bei 20 MHz Abtastrate	≤ 5 LSB	
Eigenrauschen bei 2,5 MHz Abtastrate	≤ 4 LSB (0,006 %)	
Meßbereich	Nullpunktsfehler	Meßabweichung
± 1 V, ± 2 V	≤ 5 mV	$\leq 0,08\%$ Bereichsendwert
± 5 V, ± 10 V	≤ 10 mV	$\leq 0,05\%$ Bereichsendwert
± 20 V	≤ 20 mV	$\leq 0,1\%$ Bereichsendwert
$\pm 50V$	≤ 40 mV	$\leq 0,2\%$ Bereichsendwert
AC-Kopplung (nur Option A)		$\leq 3\%$ Bereichsendwert

Transfer - Geschwindigkeiten	
Internes RAM	200 MB/s (100 MS/s)
Interne Festplatte (nur LTT-186)	max. 19,6 MB/s (9,8 MS/s)
PC (mit SCSI)	typ. 17 MB/s (8,5 MS/s)
PC (mit IEEE1394)	typ. 9 MB/s (4,5 MS/s)
PC-Festplatte	Die Speichergeschwindigkeit hängt vom verwendeten PC ab. Typ. (Stand April 2003): 10 MB/s (5 MS/s)

Speicher	
Speichertiefe (RAM)	128 MB (64MS)
Zugriffszeit	4 ns
Interne HD (nur LTT-186)	8 GB bzw. 16GB Datenpuffer (Optional: 1 GB Flash Disk)

Betriebsbedingungen	
Versorgung	9-36VDC / 100-240VAC, 60 bis 120 W Leistungsaufnahme
Temperatur	+10 bis +40 $^{\circ}C$
PC-Anbindung	SCSI-II-Interface, 8 Bit, 20 MHz ULTRA oder IEEE1394 400 MBit/s
Versorgung	Win95B, Win98, Win98SE, Win2000, WinXP SP1, WinNT4 (nur Desktop)

Abmessungen	LTT-18x	9-36VDC / 100-240 VAC Netzteil	LTT-19x (19"-Einschub)
Höhe:	63 mm	44 mm	3 HE
Breite:	325 mm	75 mm	437 mm
Tiefe:	280 mm	166 mm	565 mm
Gewicht:	ca. 4.5 kg	ca. 660 g	ca. 20 Kg

Stand: 15.5.2003

Teil 2:

Kaskadierung von LTT-184

Die Transientenrekorder LTT-184 können auf einfache Weise kaskadiert werden. Damit sind Messungen auf nahezu unbeschränkt vielen Kanälen durchführbar. Alle Messergebnisse werden auf einem zentralen Rechner in einer einzigen Datei gespeichert. Die kaskadierten LTT-Systeme arbeiten dabei zeitsynchron.

Der Aufbau einer solchen Kaskade kann folgendermaßen bewerkstelligt werden:

Beispiel: 2 Geräte

1. Stellen Sie das Mastergerät auf das Slave-Gerät. Master muß immer das Gerät mit der kleineren bzw. der kleinsten Seriennummer sein. Schließen Sie nun das obere Gerät (Master) mit Hilfe des von LTT gelieferten SCSI Kabels an Ihren PC an. Benutzen Sie den Stecker am Ende des SCSI-Flachband-Kabels.
2. Verbinden Sie die beiden Geräte, indem Sie den zweiten Stecker am SCSI-Flachband-Kabel an die SCSI-Buchse des unteren Gerätes (Slave) anschließen.
3. Verbinden Sie nun die beiden LTT-Systeme mit Hilfe des von LTT gelieferten Kabels an den digitalen Eingangsbuchsen (für die Synchronisierung)
4. Starten Sie erst die beiden Systeme und dann Ihren PC.
5. Starten Sie LTTview. Beide Geräte werden automatisch von LTTview erkannt (siehe auch Teil 3). Unterhalb der Karteikarten „ADC“, „Channels“ usw. sehen Sie in einer Anzeige, welches der beiden Geräte im Moment zur Konfiguration bereit ist. Durch einen Klick auf die Anzeige (#1, #2) können Sie zwischen den Einstellungen für die Geräte hin und herschalten.

Beispiel: 3 und mehr Geräte

Die Kaskadierung von 3 und mehr Geräten folgt analog. LTT liefert Ihnen je nach Anzahl der zu kaskadierenden Geräte die entsprechenden SCSI- und Synchronisations-Kabel. Sollten diese Teile in Ihrer Lieferung nicht enthalten sein, kontaktieren Sie bitte LTT.

LTTview sortiert die Geräte entsprechend ihrer Seriennummer. Gerät #1 (Master) ist das Gerät mit der kleinsten Seriennummer, Gerät #2 das mit der nächsthöheren usw. Daher ist es sinnvoll, die Geräte im Turm entsprechend der Seriennummer anzuordnen.

Achtung bei Geräten, die einzeln gekauft wurden und nachträglich kaskadiert werden sollen:

- Geräte in einer Kaskade dürfen nicht über identische SCSI-ID's verfügen. Einzeln von LTT gelieferte Transientenrekorder haben die SCSI-ID 2. Die LTT-Hotline kann Ihnen Hilfestellung geben, wenn die SCSI-ID geändert werden muß.
- Bei kaskadierten Systemen ist der Master terminiert. Slaves müssen unterterminiert sein.

Diese Angaben gelten für Kaskadierungen über den SCSI-Bus. Bei Verwendung von FireWire müssen alle Systeme terminiert sein.

Teil 3:

LTTview (V. 3.0)

LTTview ist die Management-Software zur Konfiguration von Messungen mit den Transientenrekordern von LTT. Die grafische Benutzeroberfläche ermöglicht eine besonders leichte Handhabung beim Beobachten und Analysieren von Messungen. Mit LTTview können Sie Messungen sehr schnell und einfach konfigurieren und sich die Daten in Echtzeit oder offline ansehen, oder die Daten in Ihre gewohnte Analyse-Software exportieren.

1. Installation

Die folgende Erklärung setzt voraus, dass SCSI- oder FireWire-Adapter bereits in Ihrem PC oder Notebook eingebaut und installiert sind.

1. Stellen Sie sicher, daß beide Systeme (PC und LTT-Gerät) ausgeschaltet sind, bevor Sie sie verbinden.
2. Verbinden Sie das LTT-System mit Ihrem PC über die SCSI-Schnittstelle des LTT-Gerätes und schließen Sie das Netzteil an.
3. Starten Sie das LTT-Gerät und anschließend Ihren PC (Achtung: Dies ist stets die notwendige Reihenfolge). Hinweis für Nutzer von Win2000, WinNT oder WinXP: Bei der Erstinstallation müssen Sie mit Administratorrechten angemeldet sein.

Die Software LTTview ist bereits im LTT-System gespeichert und wird nun automatisch auf Ihren PC installiert. Auf dem PC oder Notebook sind übrigens keine Treiber für das LTT-Gerät notwendig. LTTview wird automatisch eine Reihe von Einstellungen vornehmen und den Adaptec ASPI-Layer installieren.

4. Sollte LTTview wider Erwarten nicht automatisch starten, benutzen Sie bitte Ihren Windows Explorer, um die Datei lttview.exe direkt vom Laufwerk „LTT18x“ zu starten.
5. Bitte booten Sie Ihren PC neu, sobald Sie dazu aufgefordert werden.



Nach dem Neustart Ihres Rechners erscheint das Symbol für LTTview auf Ihrem Desktop. Ein Doppelklick auf das Symbol startet LTTview. Der Transientenrekorder ist sofort nach dem Start der Software betriebsbereit.

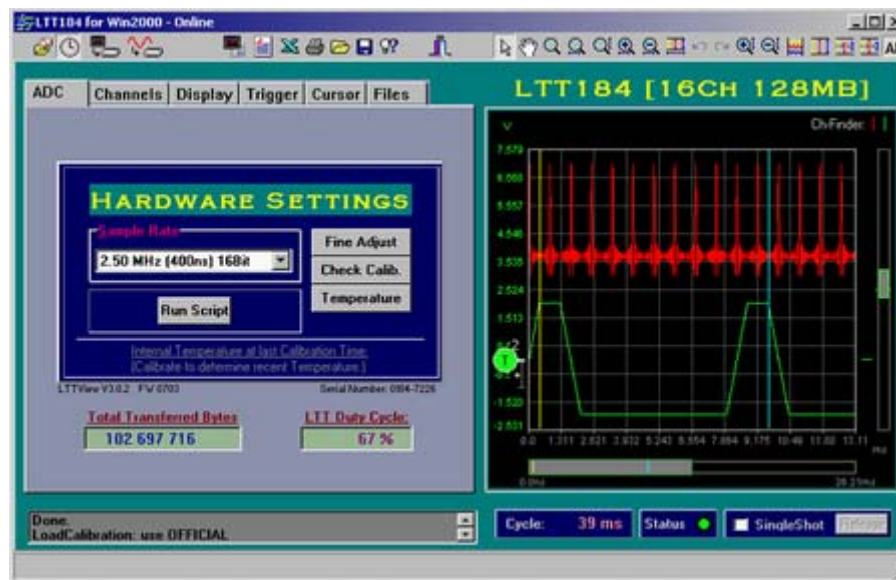
LTT-18x Systeme werden von Windows als reguläre PC-Festplatten erkannt. Mit Hilfe einer Software wie Windows Explorer können Sie direkt auf die Daten zugreifen. LTT-186 Geräte verfügen über eine interne Festplatte, die unter dem Namen LTTDATA angezeigt wird. Bitte beachten Sie: Da LTT-18x als Festplatte erkannt wird, könnte Ihre Virenschutz-Software fälschlicherweise einen „New Boot Virus“ anzeigen.

2. Die Bildschirmoberfläche

Die grafische Benutzeroberfläche umfaßt auf der linken Seite mehrere Karteikarten, über die zahlreiche Einstellungen für Ihre speziellen Messungen vorgenommen werden können. Ein

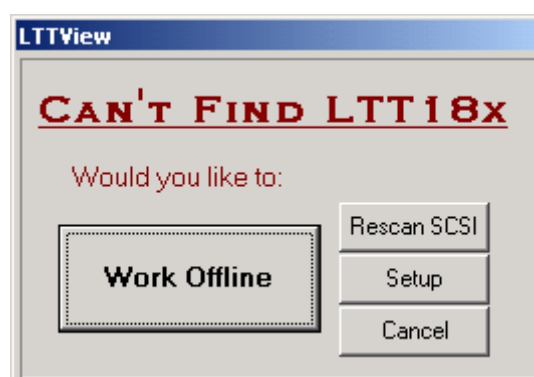
einfacher Klick auf die Reiter der Karteikarten öffnet diese und zeigt die entsprechenden Optionen an.

Auf der rechten Seite sehen Sie einen Bildschirm ähnlich dem eines Oszilloskops. Mit Hilfe der Symbole am oberen Rand des Bildschirms lassen sich zahlreiche Funktionen ansteuern wie z.B. Zoom entlang der X oder Y-Achse, Fokus auf bestimmte Kanäle, Auflösung der Bildschirmdarstellung und vieles mehr (siehe Abschnitt 5).



3. Online und Offline arbeiten

Sobald LTTview gestartet wird, sucht die Software an den Schnittstellen Ihres PC's nach dem LTT-System. Sobald ein LTT-Gerät erkannt wird, zeigt LTTview automatisch die Seriennummer des Systems auf der Karteikarte „ADC“ direkt unterhalb der „Hardware Settings“ an. Hardware und Software sind ab sofort bereit für die Durchführung von Messungen.



Sie haben auch die Möglichkeit, mit LTTview „offline“ zu arbeiten (also ohne ein angeschlossenes LTT-System). Starten Sie dazu LTTview. Auf dem Bildschirm wird ein Fenster erscheinen mit der Frage, ob Sie offline arbeiten möchten, da kein System erkannt wurde. Klicken Sie auf „Work offline“. Über das Symbol „open file“ (alternativ über die Karteikarte „Files“ und den Knopf „Load Data“) können Sie nun gespeicherte Messdaten wieder auf den Bildschirm holen.

Bitte beachten Sie, daß im Offline-Modus keine Änderungen an den Einstellungen von bereits abgespeicherten Messungen vorgenommen werden können. Sie sehen auf dem Bildschirm die Messung mit den ursprünglichen Einstellungen zum Zeitpunkt der Messung.

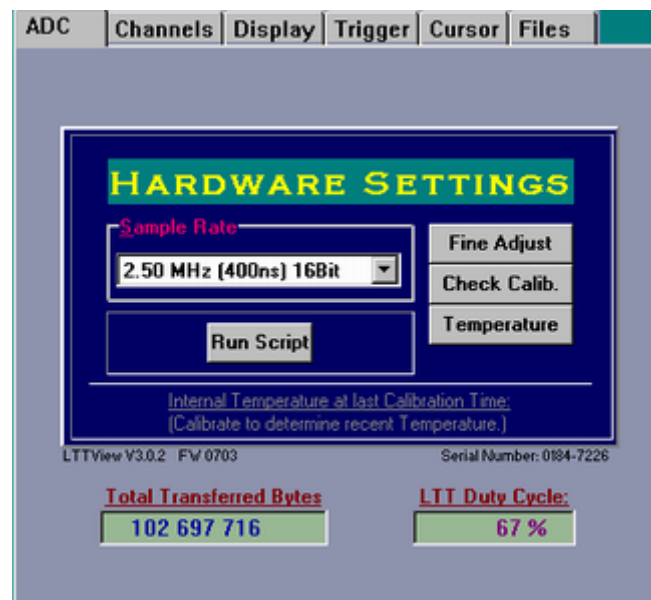
Falls LTTview kein LTT-System erkennt, obwohl Sie eines angeschlossen haben, können Sie auch manuell nach dem System suchen (Rescan SCSI) oder Setup für LTTview starten.

4. Konfiguration

LTTview bietet Ihnen zahlreiche Möglichkeiten, Messungen entsprechend Ihrer Anforderungen durchzuführen. Folgende Einstellungsmöglichkeiten stehen Ihnen zur Verfügung:

4.1 ADC

ADC (Analog to Digital Converter)



Sample Rate: Wählen Sie die Abtastrate, mit der alle Kanäle synchron abgetastet werden sollen. Neben der Abtastrate (z.B. 10 MHz) wird die pro Abtastung notwendige Zeitdauer angezeigt sowie die Genauigkeit, mit der das LTT Gerät die Messung durchführen kann. Die Spanne reicht von 751 Hz (bei 16 bit) bis 20 MHz (bei 12 bit). Es kann auch ein externer Takt eingestellt werden mit der Auflösung 12, 14, 15 oder 16 Bit. Die maximale Summenabtastrate beträgt 100 MHz.

Fine Adjust: Nach dem Klick auf Fine Adjust öffnet sich ein Fenster, das erweiterte Einstellungen für die Eingangswiderstände der Kanäle bietet:

LTT Fine Adjust

ADVANCED SETUP

☐ Delay Data Transfer

☒ Use exact Signal Source Impedance

DC SOURCE RESISTANCES

	BNC+ to GND	BNC+ to BNC-	BNC- to GND
Ch01:	50	infini	50
Ch02:	50	infini	50
Ch03:	50	infini	50
Ch04:	50	infini	50
Ch05:	50	infini	50
Ch06:	50	infini	50
Ch07:	50	infini	50
Ch08:	50	infini	50

	BNC+ to GND	BNC+ to BNC-	BNC- to GND
Ch09:	50	infini	50
Ch10:	50	infini	50
Ch11:	50	infini	50
Ch12:	50	infini	50
Ch13:	50	infini	50
Ch14:	50	infini	50
Ch15:	50	infini	50
Ch16:	50	infini	50

Cancel Default OK

Standardmäßig ist für alle Kanäle der Signalquellen-Widerstand eingestellt (**Use exact Signal Source Impedance**). Sie können nun den Widerstand pro BNC+ Eingangsbuchse zur Masse, BNC+ zu BNC- (bei differenziellen Messungen) und BNC- zur Masse einstellen.

Delay Data Transfer: Hochgenauer Meß-Modus. Es findet kein Datentransfer während der Messung statt. Die Datenmenge ist in diesem Fall auf die Größe des internen Speichers begrenzt (z.B. 128 MB). Hinweis: Dieser Modus wird nicht von allen PC's unterstützt.

Default: Setzt alle in der Tabelle eingetragenen Werte auf Standardwerte zurück.

Check Calib.: Führt eine „User-Kalibrierung“ durch.

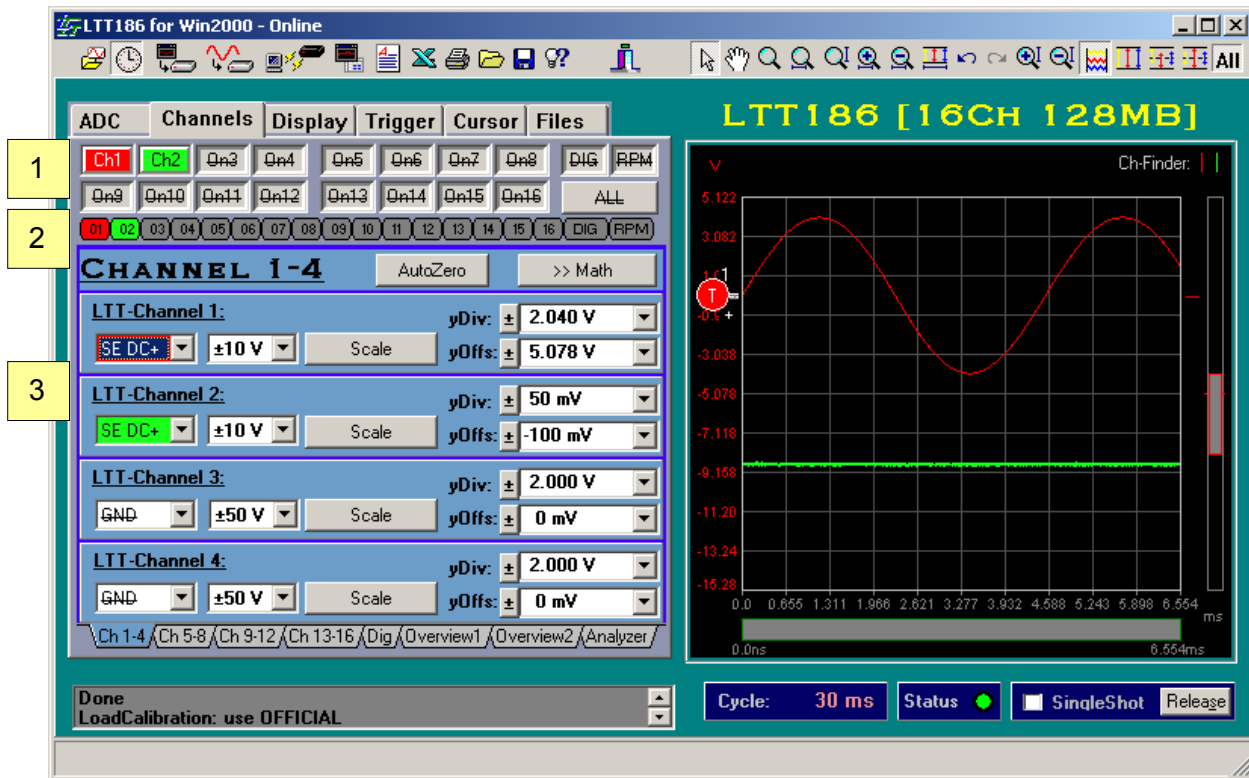
Temperature: In den LTT-Geräten sind zur Kontrolle Temperaturfühler eingebaut. Über die Taste Temperature wird die Innen-Temperatur für das bzw. die angeschlossenen Geräte angezeigt.

Total Transferred Bytes: Anzeige der insgesamt über den SCSI-Bus geflossenen Datenmenge, seit das Gerät online ist (zur Information).

LTT Duty Cycle: Gibt die Auslastung des LTT-Systems an.

Im kleinen grauen Fenster am unteren Rand des Fensters werden Statusmeldungen des Systems angezeigt wie z.B. gefundenener Gerätetyp, ob eine Messung vollständig abgeschlossen wurde etc. Ein Doppelklick auf das Fenster öffnet einen Editor mit den gesamten bislang geloggten Statusmeldungen.

4.2 Channels



1

Ch1 – Ch16: Ein Klick auf den jeweiligen Button aktiviert bzw. deaktiviert den jeweiligen Eingangskanal. Wenn der Mauszeiger über einen aktiven Kanal gezogen wird, wird dieser Kanal im Display im Vordergrund angezeigt mit der entsprechenden Skala an der Y-Achse.

Ein Klick mit der rechten Maustaste öffnet ein Fenster zur Einstellung der Farbe für den jeweiligen Kanal.

DIG: Aktiviert die digitalen Eingangskanäle. An der Rückseite des LTT-Gerätes stehen 16 einzelne TTL-Eingänge über eine SUB-37-Buchse zur Verfügung.

RPM: Rotation per Minute: Der RPM-Kanal ermöglicht die Darstellung der zeitlichen Differenz zwischen zwei aufeinander folgenden externen Takten. Dieses Feature der LTT-Messsysteme ist äußerst nützlich beispielsweise bei der Messung von rotierenden Bewegungen, um z.B. den Gleichlauf von Motoren zu vermessen oder Torsionsschwingungen festzustellen.

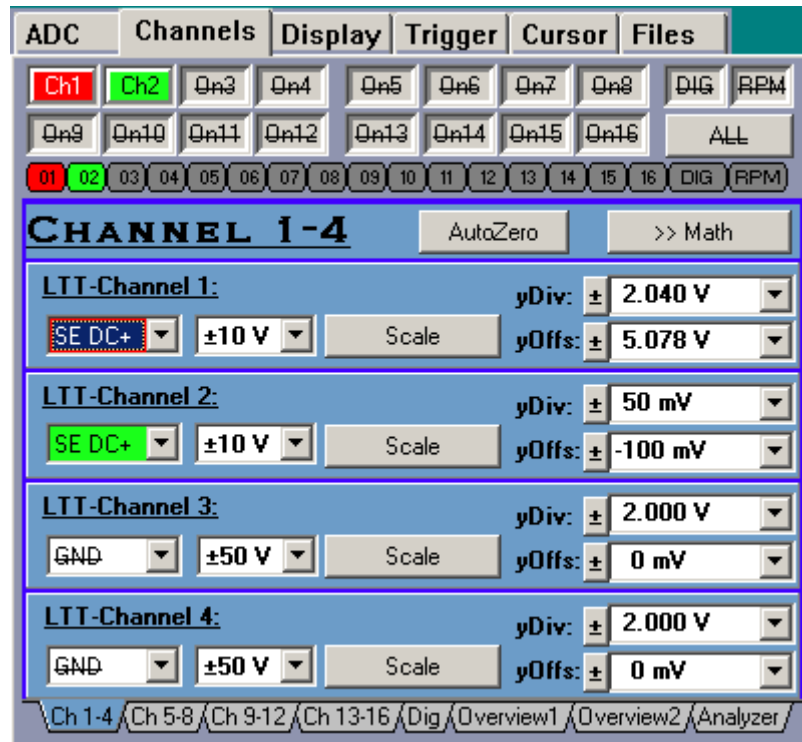
Besonders interessant ist RPM, wenn die Messwerte synchron zu einem externen Takt gemessen werden. Bei dieser Konfiguration liefern die LTT-Systeme die Messwerte der einzelnen Sensoren abhängig vom externen Takt. Über den RPM-Kanal, der diesen externen Takt zeitlich vermisst, wird zusätzlich der zeitliche Bezug der Messwerte zueinander hergestellt.

2

01-16, DIG, RPM: Ein Klick auf den jeweiligen Button aktiviert bzw. deaktiviert die Darstellung des entsprechenden Signals im Darstellungsfenster. Der Kanal bleibt aktiv.

3

Im Fenster dieses Bereiches können nun für alle aktivierten Kanäle verschiedene Einstellungen vorgenommen werden.



Mit dem Auswahlménú auf der linken Seite kann die Eingangs-Kopplung für den entsprechenden Kanal festgelegt werden:

Kanal mit AC/DC Umschaltung	
GND	Masse (Ground)
SE DC+	Single Ended DC Plus (0,1 M Ω)
SE DC-	Single Ended DC Minus (0,1 M Ω)
DE DC	Differential Ended DC (0,1 M Ω)
SE AC+	Single Ended AC Plus
SE AC-	Single Ended AC Minus
DE AC	Differential Ended AC

Kanal mit Impedanzumschaltung	
GND	Masse (Ground)
SE DC+	Single Ended DC Plus (0,1 M Ω)
SE DC-	Single Ended DC Minus (0,1 M Ω)
DE DC	Differential Ended DC (0,1 M Ω)
SE 1M+	Single Ended DC 1:10 Plus (1 M Ω)
SE 1M-	Single Ended DC 1:10 Minus (1 M Ω)
DE 1M	Differential Ended DC 1:10 (1 M Ω)

SE steht für „Single Ended“, es wird also nur eine der beiden differentiellen Buchsen aktiviert. Plus steht für die untere Buchsenreihe und Minus für die obere. DE steht für „Differential Ended“, zur Messung von differentiellen Signalen. Der ausgewählte Zustand wird durch Leuchtdioden am Gerät signalisiert.

Im Auswahlménú daneben wird der Eingangsbereich festgelegt. Dieser ist abhängig von dem Eingangswiderstand des jeweiligen Kanals:

0,1 MΩ- Eingangswiderstand	1 MΩ- Eingangswiderstand
± 50 V	± 500 V
± 20 V	± 200 V
± 10 V	± 100 V
± 5 V	± 50 V
± 2 V	± 20 V
± 1 V	± 10 V
± 25V	± 250 V
± 2,5V	± 25 V

Scale: Für jeden Kanal kann der Benutzer manuell (per Eingabe der Skala) oder automatisch (mit Hilfe der Cursor im Display) eine eigene Skalierung festlegen.

- **Auto Scaling:**

Mit Hilfe des Displays kann die Skalierung automatisch definiert werden. Setzen Sie dazu den ersten Cursor auf die gewünschte Position des Signals und geben Sie den Wert ein (**Signal at Crs1**), den das Signal an der Position annehmen soll. Bestätigen Sie den Wert mit **OK 1**. (Hinweise zur Definition eines Cursors finden Sie in Kapitel 4.5).

Plazieren Sie den zweiten Cursor auf der gewünschten zweiten Position des Signals und geben Sie den entsprechenden Sollwert ein. Bestätigen Sie den Wert mit **OK 1+2**. Das Programm ermittelt nun automatisch die notwendige Skalierung.

- **Manual Scaling:**

Die Skalierung umfasst:

$$\text{Val}_{\text{User}} = \frac{\text{Val}_{\text{Volt}}}{\text{Volt}} * \text{Faktor}_{\text{User}} + \text{Offset}_{\text{User}}$$

Auch die Einheit kann manuell angepasst werden.

Beispiel: Messung von Strömen. Mit der Ausgangscharakteristik des Sensors (z.B. $2\text{V} \hat{=} 1\text{A}$) ergibt sich:

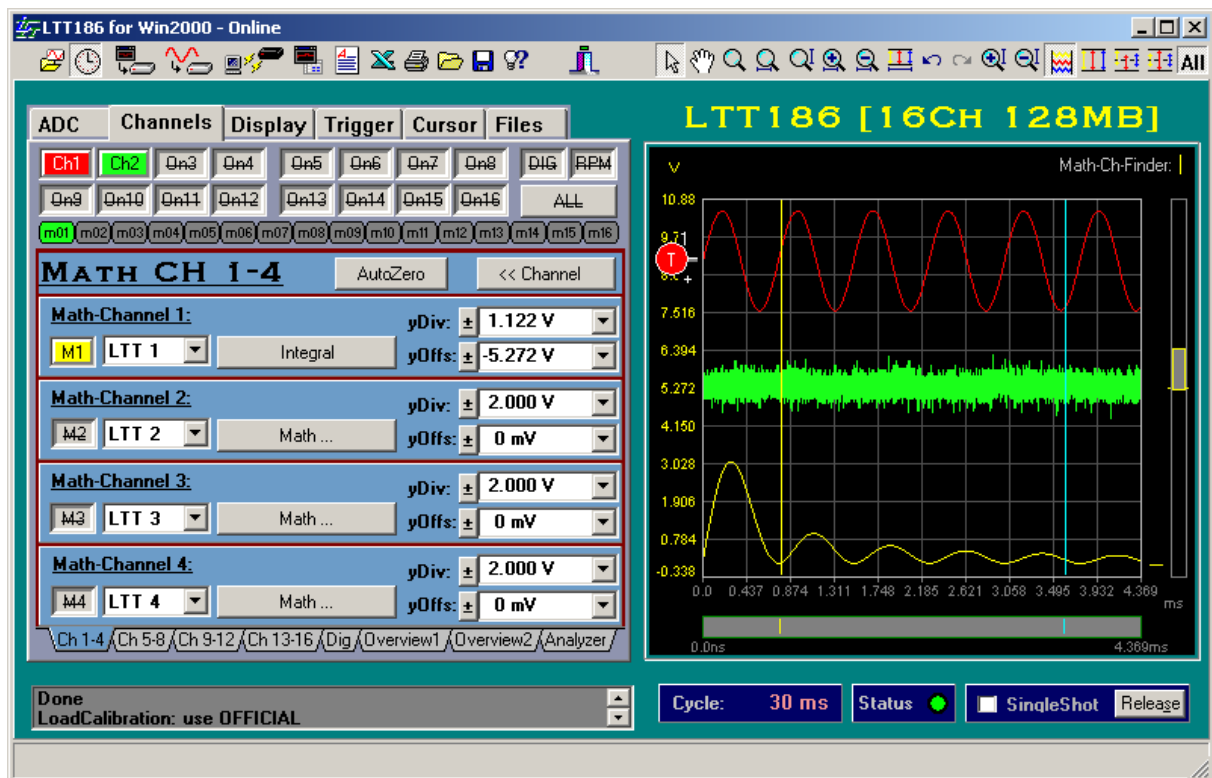
$$\text{Val} [\text{A}] = 0,5 \text{ A} * \frac{\text{Val} [\text{V}]}{\text{Volt}} + 0 \text{ A}$$

Nach dem Schließen des Dialogs ist die y-Achse des jeweiligen Kanals neu skaliert.

yDiv: Gibt ähnlich wie bei Oszilloskopen die Skalierung der Signaldarstellung an (y-Achse; vertikale Ablenkung). Im Unterschied zu Oszilloskopen wird hier jedoch nur der dargestellte Signalbereich gewählt, ohne den tatsächlichen Messbereich (Range) des Messgerätes zu verändern.

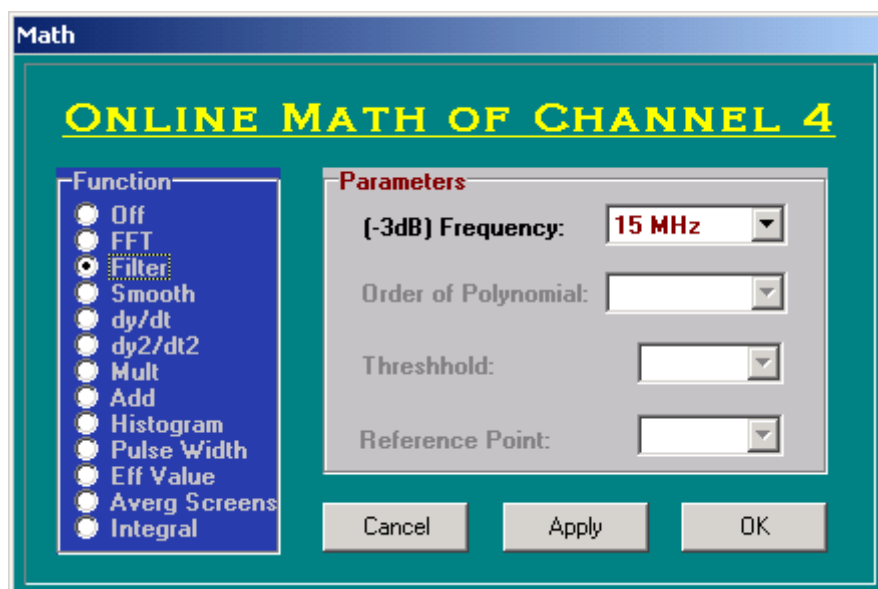
yOffs: Einstellung des Y-Offsets für die Darstellung. Erlaubt die Verschiebung des dargestellten Signalbereichs.

Math: Mit diesem Button kann vom „Channel-Konfigurationsfenster“ zum „Math-Channel-Konfigurationsfenster“ gewechselt werden und zurück.



Mit dem Button ganz links (M1, M2, ...) können die Mathematik-Kanäle eingeschaltet werden. Im Feld daneben wird der Quell-Kanal ausgewählt. Als Quellkanal stehen wiederum sämtliche echten Eingänge des LTT-Geräts zur Verfügung, ebenso wie die Mathe-Kanäle.

Math...: Ein Klick auf den Knopf öffnet ein Fenster zur Einstellung der verschiedenen Algorithmen, mit denen die Quellsignale online weiterverarbeitet werden können. Nach Auswahl einer mathematischen Funktion ändert sich die Beschriftung des Knopfes (z.B. „Filter (-3dB):26kHz“).



Folgende Funktionen stehen Ihnen zur Verfügung:

Off: Deaktiviert den Kanal und stellt den Default Wert ein



FFT: Wählt die „Fast Fourier Transformation“ mit einer Auflösung (Resolution) abhängig von der Abtastrate und Speichertiefe

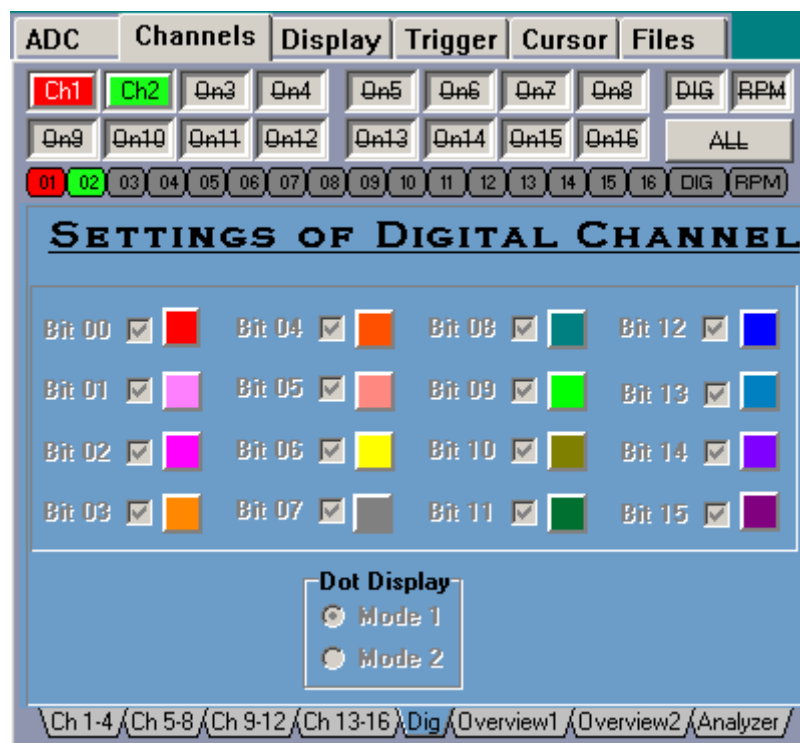
Filter:	Low Pass Filter, dessen -3dB Punkt von der Abtastrate abhängt.
Smooth:	Sovitsky-Golay Glättungsfilter, der einen lokalen ungewichteten Mittelwert berechnet
dy/dt:	1. zeitliche Ableitung
dy2/dt2:	2. zeitliche Ableitung
Mult:	Multiplikation zweier Kanäle
Add:	Addition zweier Kanäle
Histogram:	Darstellung des Signals in einem Histogramm (Häufigkeitsbetrachtung für die Amplitudenwerte)
Pulse Width:	Pulsbreiten- und Pulsfrequenz-Analyse
Eff Value:	Effektivwertanalyse einer Abfolge von Pulssignalen
Averg Screens:	Aufnahme eines bestimmten Signalbereichs mehrmals hintereinander und Mittelwertbildung
Integral:	Integralbildung der Signalwerte des Quellsignals

Die Änderungen können Sie mit **Apply** und **OK** festlegen.

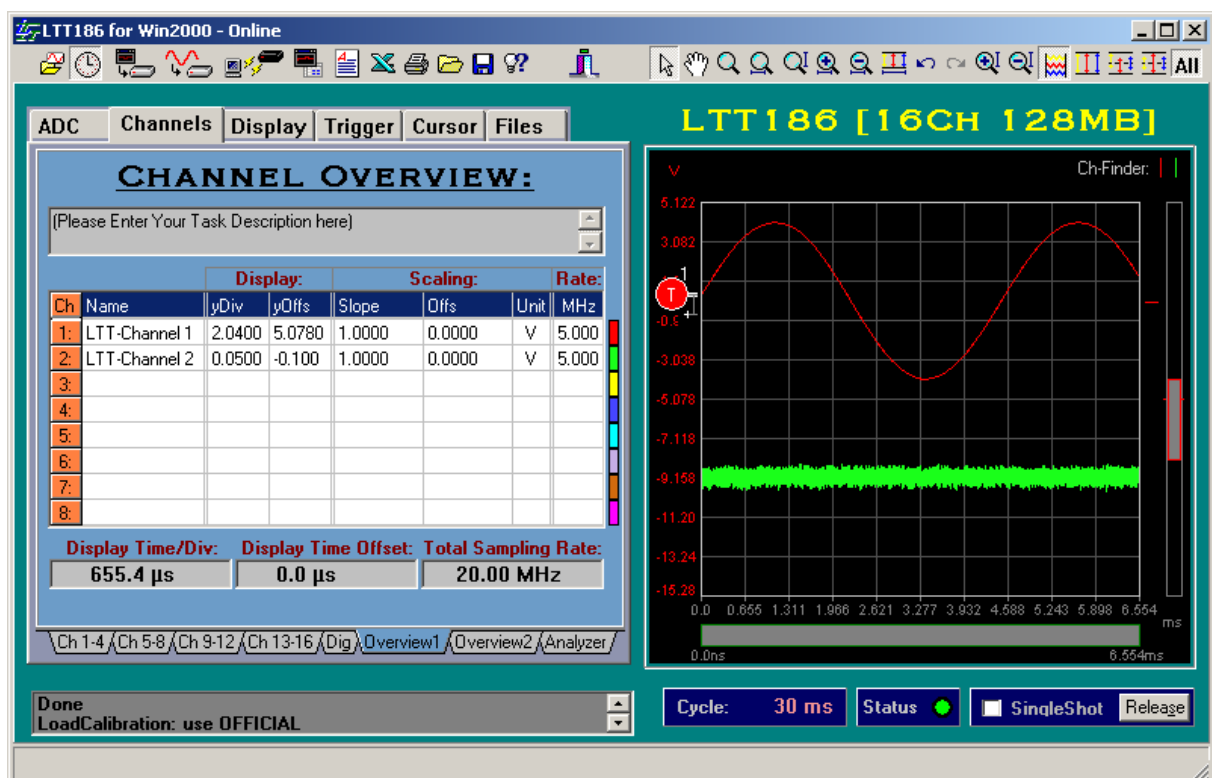
Auch bei den Mathe-Kanälen gibt es analog zu den echten Eingangskanälen des LTT-Systems die Möglichkeit, die Darstellung am Bildschirm anzupassen:

yDiv:	Einstellung der Skalierung für die Darstellung in vertikaler Richtung
yOffs:	Einstellung des Offsets für die Darstellung (Verschiebung entlang der y-Achse)

Digital Input: Öffnet ein Fenster, um die einzelnen Bits des digitalen Kanals an und auszuschalten und die Farben einzustellen. Über **Dot Display** kann gewählt werden, ob die Signale über Kurven  oder Farben  („Color Stream Display“) dargestellt werden sollen.

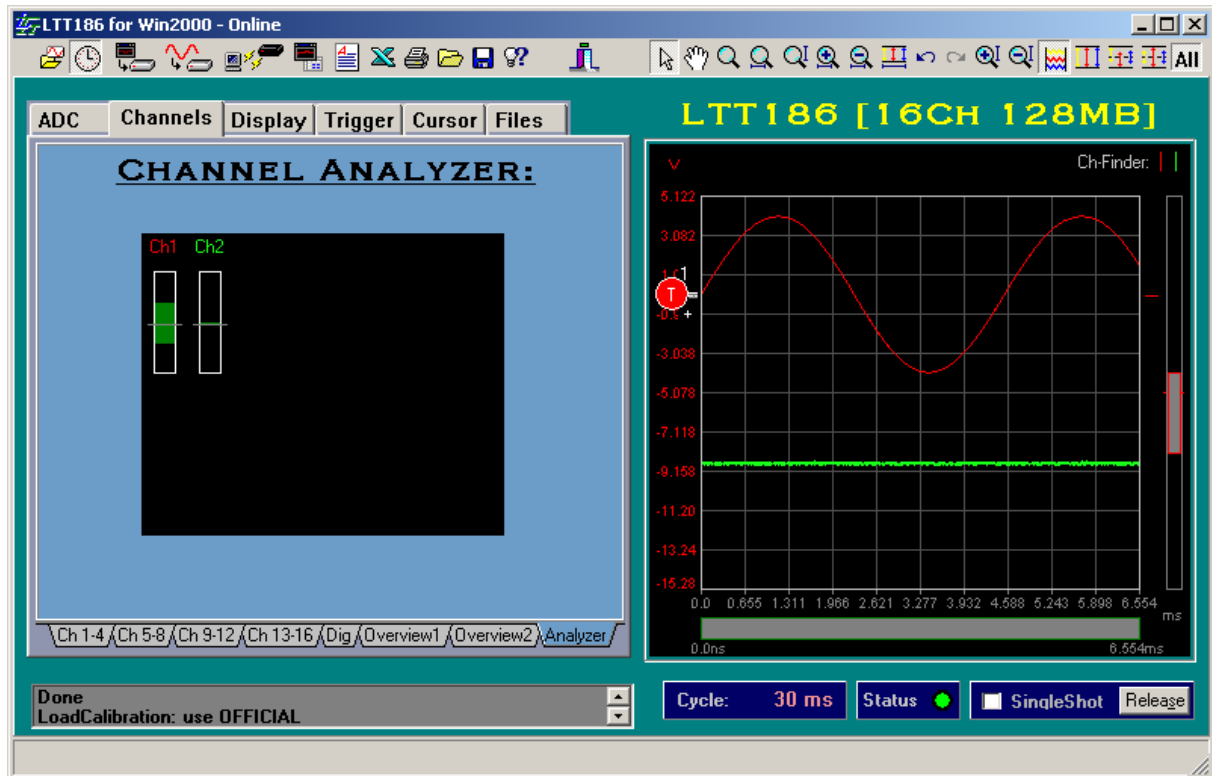


Overview1/Overview2: Übersicht über die Einstellungen der Kanäle 1-8 bzw. 9-16.

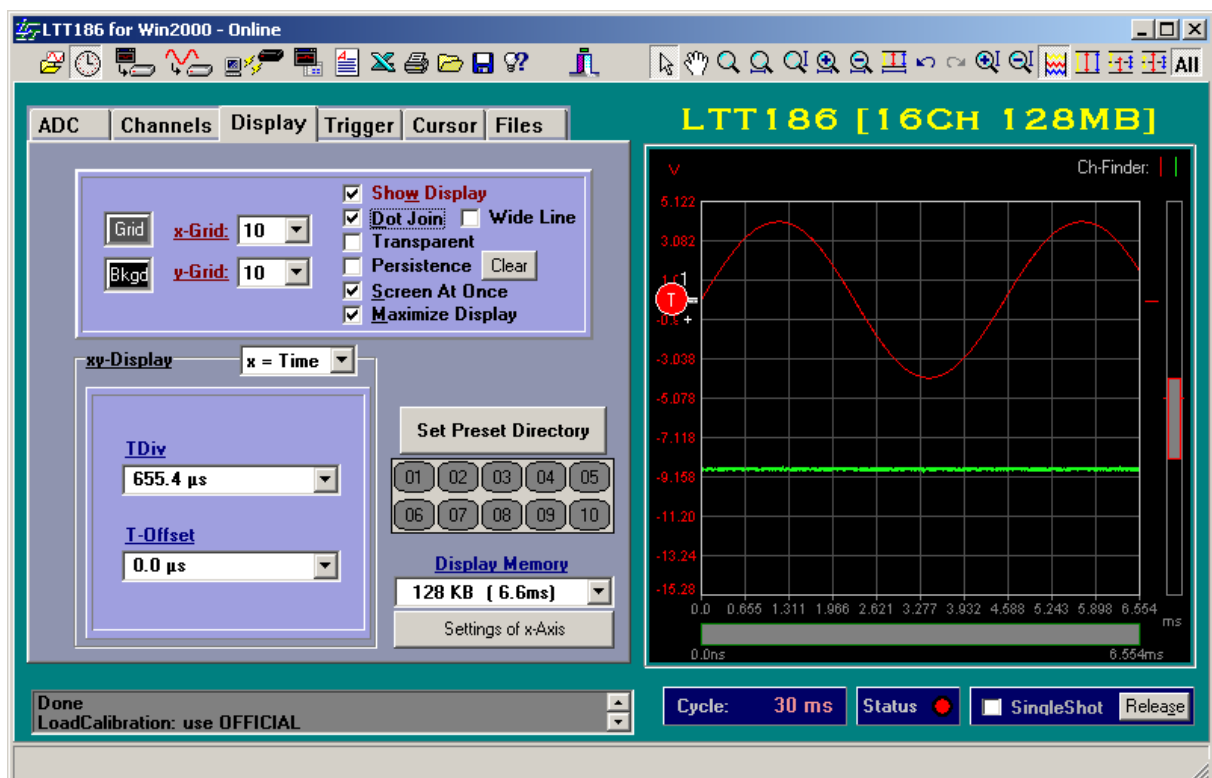


Der Text der „Task Description“ wird im Druckformular zusammen mit Messungs-Details und den Signalkurven angezeigt.

Analyzer: Der Channel Analyzer gibt eine nützliche Übersicht darüber, ob die Signale auf allen aktiven Kanälen zwischen $\pm 90\%$ des Eingangsbereichs liegen. Der senkrechte Balken ist grün, solange die Extrema der Signale innerhalb der Eingangsbereiche liegen. Sobald die Signale außerhalb des Bereichs sind, erscheint der Balken rot.



4.3 Display



Grid: Stellt die Gitternetzlinien ein oder aus (Rechte Maustaste: Gitterfarbe ändern)
x-Grid: Anzahl der Kästchen entlang der X-Achse
y-Grid: Anzahl der Kästchen entlang der Y-Achse

Bkgd: Einstellung der Hintergrundfarbe. Mit einem Klick der rechten Maustaste auf den Button erscheint die Standard-Farbpalette, aus der Sie eine beliebige Farbe auswählen können.

Show Display: Ausblenden/Einblenden der grafischen Darstellung (Display)

Dot Join: Verbindung der Messpunkte durch Linien

Wide Line: Linien werden fett dargestellt

Transparent: Wenn sich Linien überlagern, „scheint“ die hintere Linie durch.

Persistence: Simuliertes Nachleuchten aller Signalkurven. Wie in einem Histogramm werden Signalereignisse gezählt, es werden aber Farben zur Darstellung der Häufigkeit dieser Ereignisse genutzt. Je öfter ein Signalereignis eintritt, desto heller wird es angezeigt. Persistence ist ein sehr hilfreiches Werkzeug, wann immer wiederkehrende Signale nach Abweichungen überprüft werden sollen. Persistence kann im Online- und Offline-Betrieb genutzt werden. Offline gestattet der Persistence-Modus, durch mehrere nacheinander aufgenommene Meß-Dateien zu gehen z.B. mit Fokus auf eine ganz bestimmte Messkurve. Auf diese Weise kann man auch bei langen Messungen schnell mögliche Veränderungen der Messwerte feststellen.

Screen at Once: Darstellung des gesamten Verlaufs einer Kurve, so daß auf der rechten Seite des Displays immer die neuesten Daten dargestellt werden. Man vermeidet dadurch Fehler, die beim Ablesen im Standardmodus („Scheibenwischermodus“) passieren können, wenn sich ein Bild immer wieder von links neu aufbaut, während rechts noch die „älteren“ Daten angezeigt werden.

Maximize Display: Maximiert das Display-Fenster

xy-Display: Konfiguration der xy-Darstellung durch die Wahl des Typs für die x-Achse. Je nach Wahl der Einheit für die x-Achse ergeben sich die Einstellungsmöglichkeiten für die jeweiligen Parameter.

Beispiel:

x=Time: Die x-Achse zeigt Zeiteinheiten an und die beiden folgenden Parameter können eingestellt werden:

TDiv: Einstellung der Zeit-Einheit eines Kästchens

T-Offset: Beginn der x-Achse

Set Preset Directory: Bietet die Möglichkeit, Darstellungen abzuspeichern, um sie später wieder aufrufen zu können. Direkt unterhalb des Buttons befinden sich 10 Speicherknöpfe. Sie können also 10 verschiedene Ansichten (Zoomfaktoren usw.) abspeichern und wieder anzeigen. Diese Voreinstellungen betreffen nur die Anzeige im Display. Zum Speichern drücken Sie einen „Preset-Button“ bis er blinkt. Die Voreinstellung ist nun gespeichert. Mit Hilfe der Preset-Tasten können viele Kanäle schnell und einfach betrachtet und verglichen werden.

Display Memory: Gibt die Speichertiefe des Displays an (die momentan auf dem Display angezeigten Daten). Das Maximum liegt bei 16 MB.

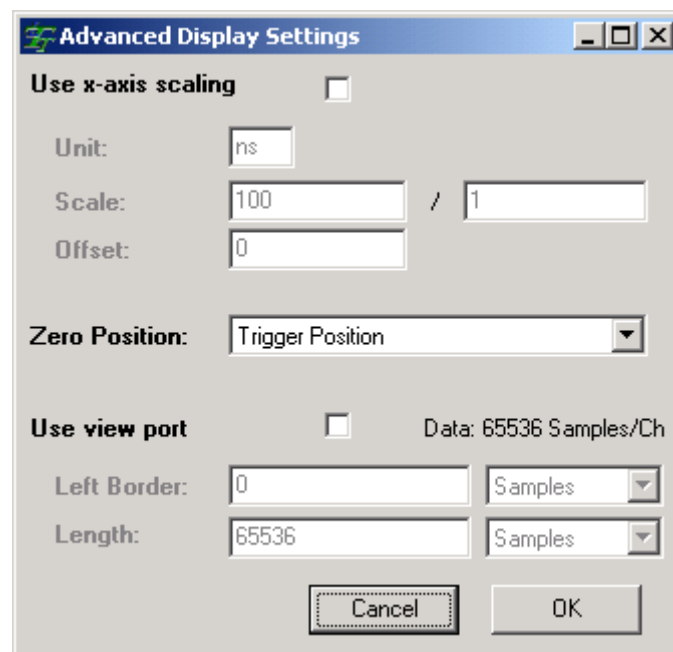
Settings of x-Axis: Öffnet das Fenster **Advanced Display Settings** für erweiterte Einstellungen der Beschriftungen für die x-Achse und für die Wahl des „View Ports“. Als „View Port“ wird der Ausschnitt der Meßdaten bezeichnet, der im Display von LTTview verwaltet werden soll. Bei deaktiviertem View Port können die gesamten Meßdaten von LTTview angezeigt werden.

Anwendungsbeispiel:

An einer rotierenden Kurbelwelle ist eine Lochscheibe mit 3600 Schlitzen angebracht, die den Messtakt vorgibt. Am Bildschirm soll jeweils exakt eine Umdrehung (360°) dargestellt werden.

LTTview erlaubt nur spezielle diskrete Werte für die Bildschirmspeichertiefe (z.B. 4096 Samples = 8 kB). Mit Hilfe des View Ports können aber die darzustellenden 3600 Samples als Ausschnitt der Gesamtmessung definiert werden.

Für die Beschriftung der x-Achse kann dann als Einheit 1° pro 10 Samples eingestellt werden.



Use x-axis scaling: Anpassung der Beschriftung für die x-Achse

Unit: gewünschte Einheit der x-Achse (max. 5 Zeichen, z.B. mm/s²)

Scale: Umrechnungsfaktor für die Einheit pro Sample

Offset: Offset der x-Skalierung in Samples pro Kanal

Zero Position: Bezugspunkt der x-Achsen-Beschriftung. Zur Auswahl stehen „Triggerposition“ und „First Sample“:

Trigger Position: Der Nullpunkt der x-Achse wird auf den Trigger-Zeitpunkt gelegt. Wenn ein pretrigger definiert wird, ist dieser Punkt nicht gleich dem Datenanfang.

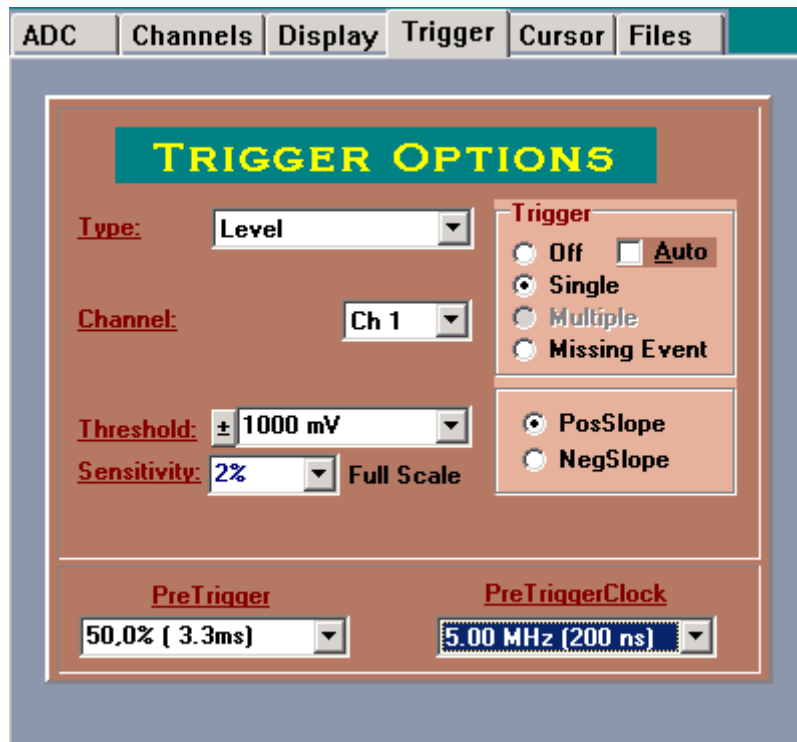
First Sample: Der Nullpunkt der x-Achse liegt am Datenanfang

Use view port: Auswahl des Datenausschnittes, der unter LTTview bearbeitet werden soll.

- Left Border:** Anzahl der Samples pro Kanal, die am Meßdatenanfang ignoriert werden sollen.
- Length:** Anzahl der Samples pro Kanal, die von Left Border angefangen an LTTview weitergegeben werden.

Hinweis: Wenn als Bezugspunkt der x-Achse der Triggerpunkt gewählt wird und der View Port so definiert wird, dass der Triggerzeitpunkt außerhalb des View Ports liegt, liegt der Nullpunkt der x-Achse auch außerhalb!

4.4 Trigger



Off: Triggeroption ausgeschaltet

Single: Bei Wahl dieser Trigger-Option wird im Gegensatz zu „Multiple Trigger“ der Einfachtriggermodus aktiviert. Es ergeben sich dann die folgenden Konfigurationsmöglichkeiten:

Type: Erlaubt die Wahl des Triggertyps

- **Level:** Schwellwert, der über oder unterschritten werden soll, um die Aufzeichnung einer Messung zu starten. Der Trigger ist abhängig vom Schwellwert und von der Empfindlichkeit (= Hysterese).

Die Empfindlichkeit definiert bei einer positiven Steigung (pos. Slope) einen unteren Schwellwert bzw. bei einer negativen Steigung (neg. Slope) einen oberen Schwellwert, der vom Signal erst unterschritten bzw. überschritten werden muß, bevor das Überschreiten bzw. Unterschreiten des Trigger-Levels ein Triggerereignis produziert. Hilfreich ist dies bei sehr verrauschten Signalen. Die Empfindlichkeit wird in Prozent des „Full Scale Ranges“ angegeben. Bei einem Range von 50V entspricht eine Empfindlichkeit von 2% also 1V.

- **Comparison:** Die Aufzeichnung der Messung wird begonnen, wenn der Messwert größer oder kleiner eines bestimmten Schwellwertes ist. Der Comparison Trigger ist also ein „Level-Trigger“ ohne Einstellung einer Empfindlichkeit oder Flanke. Die

Historie des Signalverlaufs wird also nicht berücksichtigt. Die Messung startet also nicht erst beim Übergang in den definierten Bereich, sondern sobald sich das Signal im definierten Bereich befindet.

- **Region:** Die Aufzeichnung der Messung wird begonnen beim Eintritt (Austritt) der Meßdaten in (aus) einen (einem) Wertebereich. Der Bereich wird definiert durch die Grenzwerte Upper und Lower Limit sowie durch die Enter- und Exit-Optionen.
- **Extrema in Region:** Bei dieser Variante wird auf Maximalwerte innerhalb eines definierten Wertebereichs getriggert. Dazu wird eine bestimmte Zeitverzögerung bestimmt (Noise Suppressor). Ziel ist das Triggern auf Ausreißersignale, die kleiner oder größer als reguläre Signale sind.

Dazu kann ein Fenster (= Region, Wertebereich) definiert werden, in dem das Maxima (Minima) eines Signalausreißers liegen muß, um die Messung zu triggern. Um lokale Extrema wie z.B. Rauschen zu unterdrücken, kann eine Zeitdauer (Noise Suppressor) angegeben werden, die bestimmt, welche zeitliche Breite ein Signalausreißer mindestens haben muß, um einen Trigger auslösen zu können.

- **Level+Delta:** Entspricht einem Level-Trigger mit zusätzlicher Bedingung. Damit ein Signal ein Triggerereignis auslösen kann, muß es zusätzlich zu den Bedingungen des Level-Triggers eine minimale Steigung aufweisen. Zwischen zwei aufeinanderfolgenden Messwerten muß also eine Minimaldifferenz vorhanden sein. Diese Minimaldifferenz wird in Einheiten des Kanals angegeben.

Channel: Wahl des Kanals bzw. der Kanäle, auf dem/denen getriggert werden soll. Wird der digitale Kanal ausgewählt, kann noch das Bit (15 bis 07) des externen Digital Input-Steckers gewählt werden, auf das getriggert werden soll. Alternativ kann eingestellt werden, dass das Drücken des Start Knopfes am LTT-System die Messung auslöst (**Start Button**).

Die Option **Scan** ermöglicht die Auswahl von mehreren Triggerkanälen, die mit einer ODER-Verknüpfung die Messung starten können.

Threshold: Der Schwellwert kann jeden beliebigen Wert im gültigen Messbereich des ausgewählten Kanals annehmen.

Sensitivity: Empfindlichkeit (Hysterese), sie kann zwischen 0,02% und 50% eingestellt werden (bezogen auf Full Scale Range).

PosSlope / NegSlope: Stellt die Flanke für das Trigger-Ereignis ein

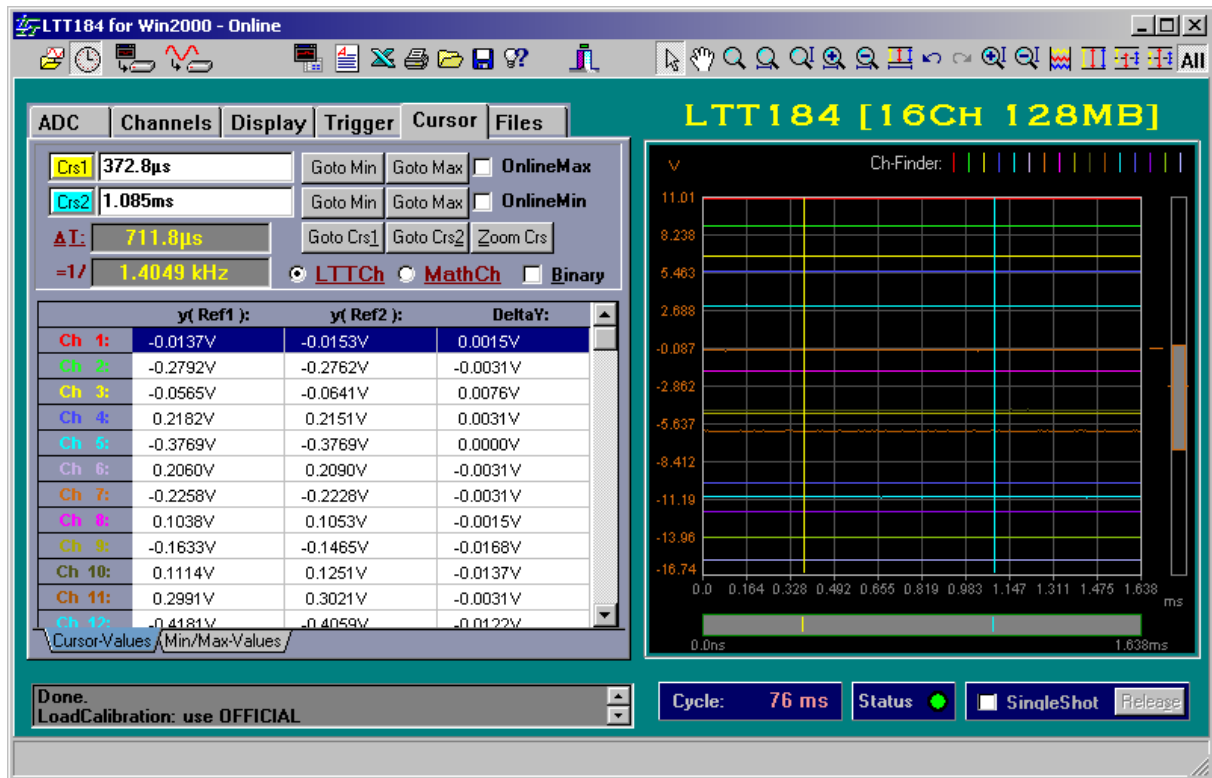
Bei multiplen Triggern (**Multiple**) kann außerdem ein Wert eingegeben werden für den Zeitraum, nach dem der Trigger wieder zurückgesetzt werden soll (**Reset Trigger after**). Die Totzeit zwischen zwei Triggern beträgt wenige µs.

Missing Event: Die Messung startet dann, wenn nach einem erfolgreichen Triggerereignis kein zweites Triggerereignis innerhalb einer definierten Zeit (**Delay Time**) auftritt. *Wichtig:* Der Trigger startet erst durch das Auftreten eines Triggerereignisses. Kommt vor Ablauf des Timers ein erneutes Ereignis, startet der Timer wieder von vorne. Dadurch erhält man die Möglichkeit, auf ein ausbleibendes Signal zu triggern → „Missing Event“. Der Missing Event Trigger kann auch zusammen mit einem prretrigger verwendet werden.

Pretrigger: Wenn ein pretrigger definiert wird, werden auch Daten vor Eintritt des Trigger-Events angezeigt. Die pretrigger-Speichertiefe ist maximal 120 MB. Das Trigger-Ereignis ist skaliert als Prozentsatz des Display-Memory. (Es ist daher auch möglich, einen pretrigger größer als 100% festzulegen.)

pretriggerClock: A-B Modus. Mit Hilfe des pretriggers kann vor dem Triggerereignis eine andere Abtastrate gewählt werden als die Abtastrate, die über die ADC-Karteikarte eingestellt wurde (PostTriggerClock). Die pretriggerClock kann maximal so definiert werden, dass die Summenabtastrate des Meßsystems nicht überschritten wird (Anzahl der aktiven Kanäle mal Abtastrate < 100 MS).

4.5 Cursor



Crs1: Schaltet den ersten Cursor ein / aus und stellt die Farbe für den Cursor mit Hilfe der rechten Maustaste ein.

GotoMin / GotoMax: Setzt Cursor 1 auf das Minimum/Maximum des gewählten Kanals im aktuellen Bildschirmausschnitt. Als „gewählter Kanal“ gilt der sog. „Display Channel“, also der Kanal, dessen y-Achsenbeschriftung am linken Rand des grafischen Displays steht. Der Display Channel kann auf verschiedene Weise umgeschaltet werden (siehe auch Abschnitt 5. Grafische Darstellung).

1. Ch-Finder: Rechts oben im grafischen Display gibt es für jeden Kanal eine farbige Linie. Durch Anklicken mit der Maus wechselt der Display Channel.
2. Y-Scrollbar: Rechts vom grafischen Display kann durch Klicken der rechten Maustaste eine Liste geöffnet werden, in der man den Display Channel wählen kann
3. Karteikarte „Channels“: Durch Bewegen des Mauszeigers über die Ein/Ausschaltknöpfe der Kanäle wechselt der Display Channel.
4. Karteikarte „Cursor“: Durch Bewegen des Mauszeigers über die Kanalnummern in der Cursor-Werteliste wechselt der Display Channel.

OnlineMax: Setzt Cursor 1 dynamisch zum Maximum des gewählten Kanals

Crs2: Schaltet den zweiten Cursor ein / aus und stellt die Farbe für den Cursor mit Hilfe der rechten Maustaste ein.

GotoMin / GotoMax: Setzt Cursor 2 auf das Minimum/Maximum des gewählten Kanals.

OnlineMin: Setzt Cursor 2 dynamisch zum Minimum des gewählten Kanals

Goto Crs1: Paßt die Bildschirmanzeige so an, daß der erste Cursor sichtbar wird.

Goto Crs2: Paßt die Bildschirmanzeige so an, daß der zweite Cursor sichtbar wird.

Zoom Crs: Paßt die Anzeige der Meßkurve so an, daß beide Cursor im Display sind.

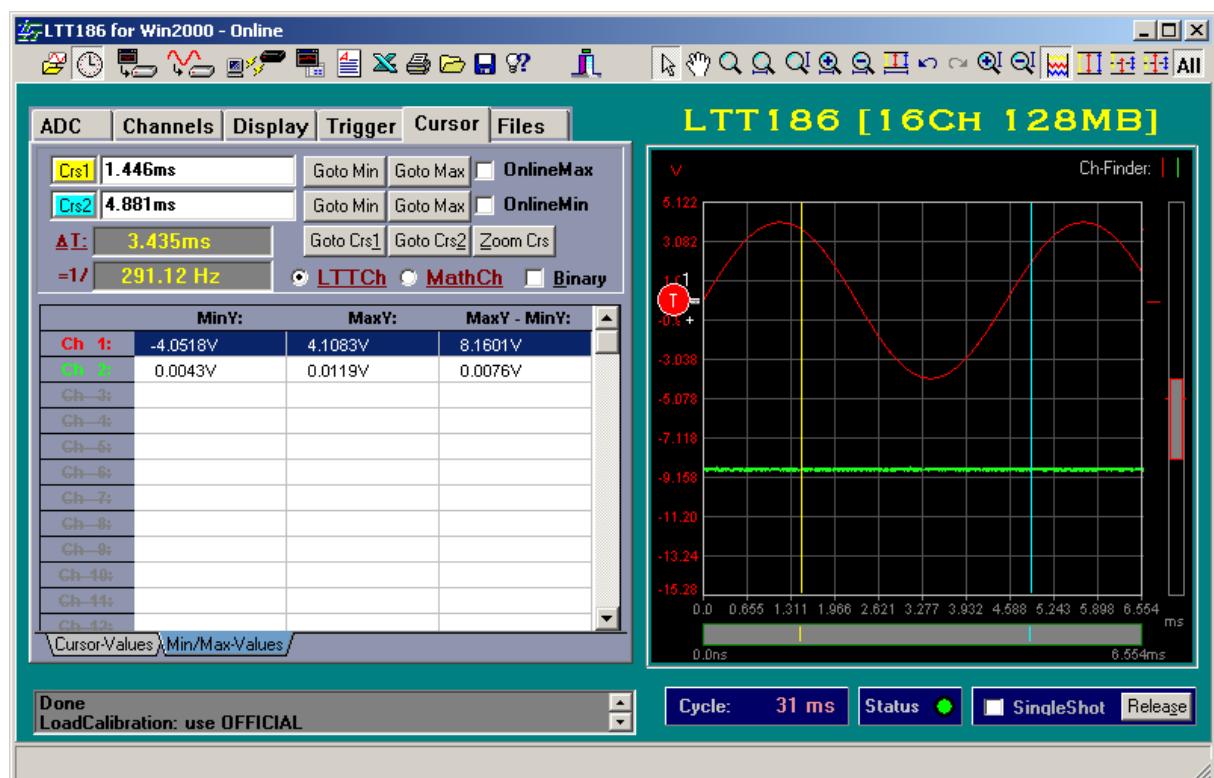
LTTCh / MathCh: Schaltet die Anzeige in der Tabelle um zwischen den Analog- und den Mathe-Kanälen.

Binary: Gibt die Daten im 16 bit Binär-Format wieder.

Cursor-Values: Die Tabelle Cursor-Values zeigt die y-Werte für alle Kanäle im Punkt der beiden Cursor an.

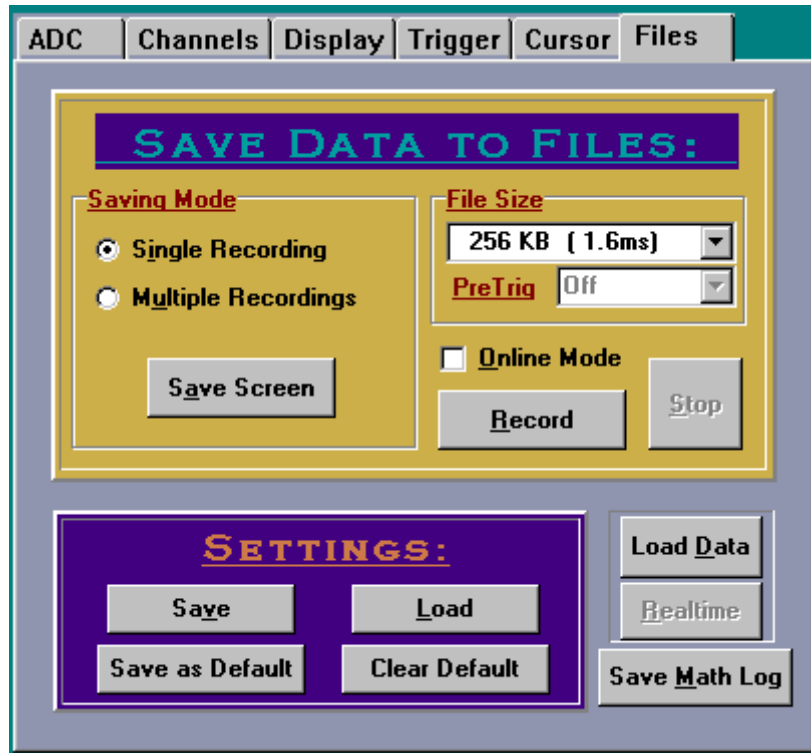
Hinweis: Die Kanäle können an- und ausgeschaltet werden, indem mit der linken Maustaste auf die erste Spalte in der Tabelle geklickt wird. Mit Hilfe der rechten Maustaste können die Farben geändert werden.

Min/Max-Values: Die Tabelle Min/Max-Values gibt für alle aktivierten Kanäle die minimalen und maximalen Messwerte sowie die Differenz zwischen beiden an. Grundlage ist die Gesamtheit der Meßdaten im Bereich des View Ports (siehe Abschnitt 4.3 Display).



4.6 Files

Diese Seite bietet alle Einstellmöglichkeiten für Konfiguration und Start sowie die Speicherung von Messungen.



Saving Mode: Drei verschiedene Messtypen können gewählt werden:

Single Recording: Die Messung wird einmal durchgeführt

Multiple Recordings: Mehrfachmessung, für die eine Triggeroption aktiviert sein muß. Für jede Messung wird eine eigene Datei erstellt.
Achtung: Hier übernimmt nicht das Messgerät die automatische Reaktivierung des Triggers (siehe „Multiple Trigger“ auf der Karteikarte **Trigger**), sondern die Software LTTview stoppt und startet die Messung in eigener Regie. Daraus ergeben sich deutlich längere Totzeiten (~1 ms).

Save Screen: Speichert den aktuell auf dem Bildschirm sichtbaren Zeitausschnitt der Messdaten in ein frei wählbares Verzeichnis.

File Size: Definition der zu generierenden Datei:

File Size: Bestimmung der Dateigröße, die die Datei maximal haben darf. Zur Information wird zusätzlich angegeben, wie lange die Messung dauern wird.

PreTrig: Einstellung des pretriggers für die **Speicherung** der Messung. Dieser kann sich von der Angabe in der Karteikarte **Trigger** unterscheiden. Dort bezieht sich die Angabe zum pretrigger auf den Display Memory. Es sind maximal 120 MB Speichertiefe möglich. Die pretrigger-Zeit wird automatisch angepasst an die gewählte Abtastrate.

Online Mode: Während der Messung kann das Messsignal im Display beobachtet werden. Per Default ist diese Option ausgeschaltet, um maximalen Datendurchsatz zu erreichen.

Fast Save: Standardmäßig wird bei der Speicherung von großen Dateien parallel eine komprimierte Datei erstellt. Wird die Option Fast Save gewählt, wird die Generierung der komprimierten Datei unterdrückt, um einen höheren Durchsatz zu erreichen.

Record: Startet die Messung

Im Offline Modus erscheint die Wahlmöglichkeit **Export Screen:** Exportiert die im Display sichtbaren Daten in ein wählbares Verzeichnis

Stop: Stoppt die Messung

Settings: Alle Änderungen in LTTview können hier gespeichert und geladen werden

Save: Speichert die Einstellungen in einem frei wählbaren Verzeichnis

Load: Lädt die zuvor abgespeicherten Settings

Save as Default: Speichert die Settings als Default. Wenn LTTview neu gestartet wird, werden diese Einstellungen übernommen.

Clear Default: Löscht die Default-Settings

Load Data: Laden von zuvor abgespeicherten Messdaten

Realtime: Wechselt vom Offline- in den Online Modus

Save Math Log: Speichert die Logdatei für angewendete mathematische Funktionen in ein frei wählbares Verzeichnis

5. Grafische Darstellung

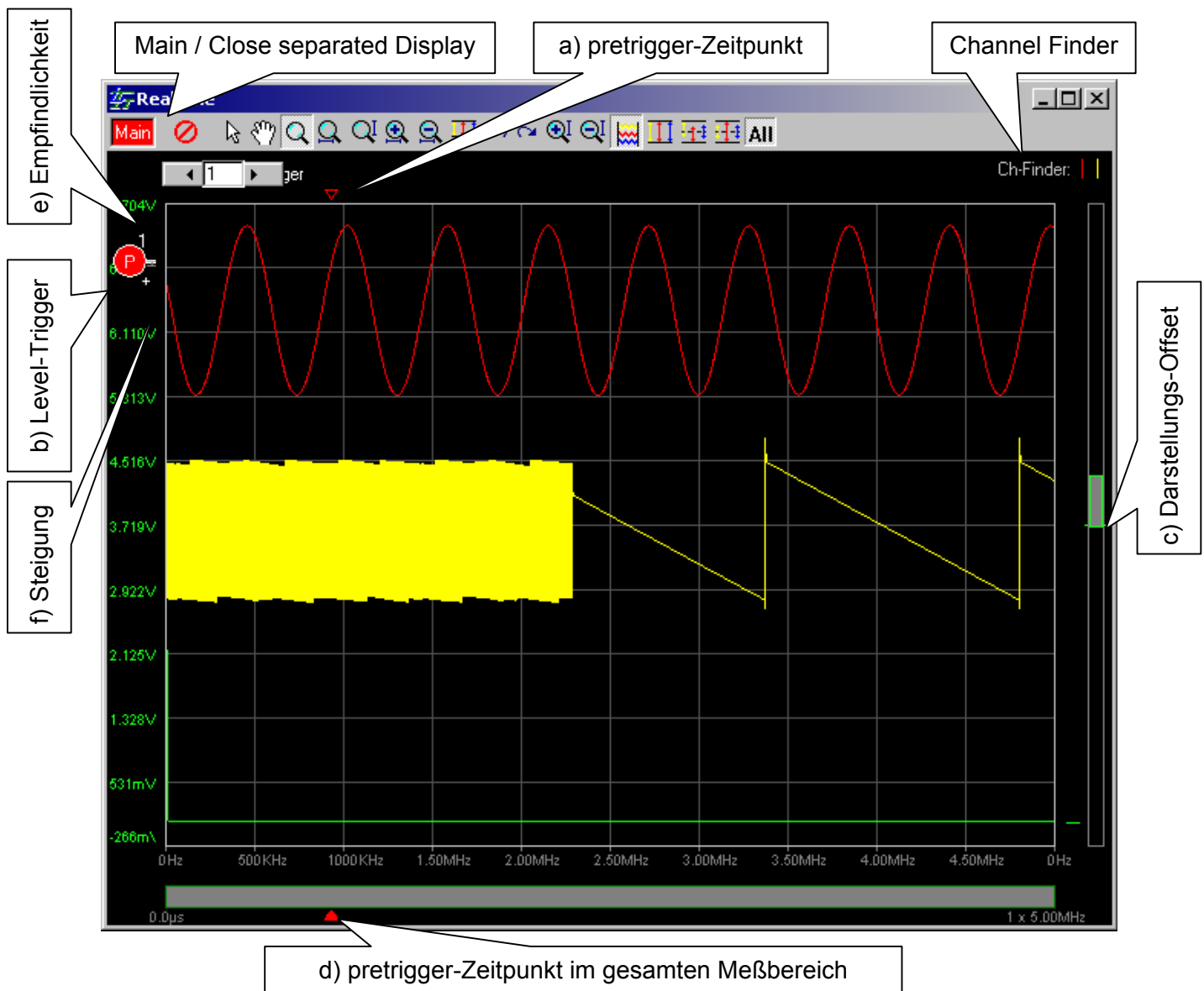
5.1 Funktionen im Display

Die grafische Darstellung von LTTview erinnert an den Bildschirm eines Oszilloskops. Bei näherem Hinsehen werden Sie erkennen, daß sich im Display sehr viele Funktionen verbergen. Alleine durch die vielfältigen Zooms ergeben sich interessante Möglichkeiten für Ihre Messdaten-Analyse.

Im folgenden Beispiel sehen Sie das Display, wenn es von den Karteikarten getrennt ist. Dargestellt wird das Ergebnis einer multiplen Messung mit einem 6,5 ms pretrigger und einem FFT-Mathe-Kanal (grün). Normalerweise nicht sichtbar sind die ersten beiden Buttons aus der Symbolleiste: **Main** und **Close Separated Display**.

Main: Macht die Karteikarten für die Einstellungen sichtbar

Close Separated Display: Schaltet zurück zur Standard-Anzeige mit dem grafischen Display neben den Karteikarten für die Einstellungen. Die weiteren Icons werden wieder unten erklärt.



Ch-Finder: Der „Channel Finder“ dient der schnellen Darstellung einzelner Signalkurven auf dem Display. Ein Klick auf eine der vertikalen Linien im „Channel Finder“ stellt den so markierten Kanal in den Vordergrund, der so der „angezeigte Kanal“ (Display Channel, DC) wird.

Sobald ein Mathekanal aktiviert ist, existieren zwei Channel-Finder: Einer für die analogen Kanäle und einer für die Mathe-Kanäle. Ein Klick auf die Schrift **Ch-Finder** bzw. **Math-Ch-Finder** schaltet zwischen den „Channel-Findern“ um.

Oberhalb, links, rechts und unterhalb des Displays wie auch im Display selbst sind verschiedene Funktionen implementiert:

- a) Ein Klick auf die Oberseite der Anzeige setzt einen pretrigger zum DC, wenn möglich
- b) Ein Klick auf die y-Achse setzt einen Level-Trigger für den DC. Es erscheint ein eingekreistes „T“ bzw. „P“ (wenn auch ein pretrigger definiert ist).
- c) Ein Klick auf die rechte y-Achse setzt einen „Darstellungs-Offset“ für einen Kanal im Verhältnis zur Vollanzeige. Der graue „Slider“ ist umrahmt in der selben Farbe wie der DC und skaliert zur Vollanzeige.

- d) Der graue Balken steht für die gesamte Meßdauer. Der Pfeil gibt den pretrigger-Zeitpunkt im Verhältnis zum gesamten Messbereich an. Ein Klick bzw. Ziehen entlang der unteren x-Achse verschiebt den Bildausschnitt („Slider“) bzw. verändert die Größe des Bildausschnitts.
 - e) Rechts neben dem Triggersymbol ist das Symbol für die Empfindlichkeit zu erkennen. Auseinanderziehen oder Zusammenschieben dieses durch einen kleinen Strich unten und einen größeren Strich oben begrenzten Bereichs verändert die Empfindlichkeit.
 - f) Ein + bzw. – zeigt die positive oder negative Steigung (slope) an
- Ein Klick mit der rechten Maustaste startet ein Pop-up Fenster um verschiedene Triggermöglichkeiten zu definieren.

5.2 Die Symbolleiste






Mit der Symbolleiste können verschiedene Funktionen direkt angesteuert werden. Die Symbole links von der Mitte stehen für allgemeine Funktionen, die Symbole rechts von der Mitte dienen der Einstellung des Displays.

a) Allgemeine Funktionen












	Datei öffnen: Erlaubt das Laden von Offline-Daten von der Festplatte
	Real Time: Umstellen vom Offline- auf Online-Betrieb
	Speichern: Speichern der Messdaten auf dem Display in ein Verzeichnis auf der Festplatte
	Messen: Messdaten werden während des Messvorgangs online auf die Festplatte gespeichert
	Konfigurationsmenü für LTT-186 (Stand Alone Configuration). Für den Stand Alone Betrieb können hierüber verschiedene Einstellungen geladen werden.
	Trennung von Display und Konfigurationsmenü. Dies erlaubt, das Display auf Bildschirmgröße zu erweitern für eine genauere Beobachtung von Messwerten
	Zwischenablage: Speichert angezeigte Daten in die Zwischenablage
	Export zu Microsoft Excel: Erlaubt die weitere Bearbeitung von Messdaten über Microsoft Excel
	Drucken: Druckt den Inhalt des Displays
	Einstellungen öffnen: Öffnen von gespeicherten Parametern für eine Messung
	Einstellungen speichern: Erlaubt das Speichern von Parametern für eine Messung
	Hilfe: Öffnet die Online-Hilfe
	Exit: Schließt LTTview

b) Displayfunktionen:

Mausfunktionen:

	Cursor umschalten auf „Cursor Modus“
	Cursor umschalten auf „Greif-Modus“. Dies erlaubt Ihnen, Messkurven direkt über das Display entlang der Zeitachse oder der Amplitude zu verschieben
	Zoom auf einen bestimmten Ausschnitt (Amplitude und Zeit)
	Zoom auf einen bestimmten Ausschnitt (nur Zeit)
	Zoom auf einen bestimmten Ausschnitt (nur Amplitude)

Direkte Funktionen:

	x-Achse: „Zoom in“ auf den Zeitabschnitt, der aktuell auf dem Display angezeigt wird (nur Zeit)
	x-Achse: „Zoom out“ auf den Zeitabschnitt, der aktuell auf dem Display angezeigt wird (nur Zeit)
	x-Achse: Show full scale: Zeigt alle Messdaten vom Nullpunkt bis zum Ende der Messung
	Rückgängig
	Wiederholen
	y-Achse: „Zoom in“ (nur Amplitude)
	y-Achse: „Zoom out“ (nur Amplitude)
	Anzeige aller aktiven Kanäle mit einem Offset, so dass sich die Kurven nicht überschneiden
	y-Achse: „Zoom all of all Channels“: Zoom der Amplitude aller Kanäle über die gesamte Display-Höhe
	y-Achse: „Zoom Range of all Channels“: Zoom auf die gesamte Amplitude aller Kanäle
	y-Achse: Center all channels: Alle Kanäle im Display zentrieren
All	Wenn „All“ gedrückt ist, beziehen sich die genannten y - Funktionen auf alle Kanäle. Wenn nicht, nur auf den aktuell markierten Kanal.

5.3 Ergänzende Angaben

Unter dem Display sehen Sie folgende Angaben:

- Cycle:** Dauer einer abgeschlossenen Messung einschließlich der Darstellung der Messung im Display
- Status:** Grün – Daten wurden erfasst
Rot – System wartet auf ein Trigger-Ereignis
Schwarz – Störung in LTTview
- Single Shot:** Gibt eine Momentaufnahme aus einer laufenden Messung wider. Mit dem Schalter **Release** wird die Single Shot Aufnahme wieder zurückgesetzt.

Teil 4:

Beschreibung der DLL LTT2API

Für die LTT Meßsysteme wurde eine DLL entwickelt, die die Einbindung in verschiedenste Software-Umgebungen erlaubt. Dieses Dokument beschreibt diese Programmierschnittstelle in der Version **LTT2API Version 1.3.5**.

Die Erklärung der einzelnen Befehle ist durchgehend in Englisch gehalten. Bitte kontaktieren Sie LTT, wenn einzelne Punkte unklar erscheinen.

Jede Funktion wird nach folgendem Schema beschrieben:

<Function Name>

[Device]

Declaration:

<Declaration of the function>

Description:

<Description of the function and explanation of the use of the function>

Input:

<specifies and describes the function parameters>

Return:

<specifies the return value>

Note: The note [Device] in the 2nd line after the <Function Name> indicates a direct communication with the LTT-System.

Fundamental concepts

The API is designed to manage one or a group of LTT devices in an efficient manner. The most important design goals are stability and performance. The calls are described in STDCALL calling convention. It's also possible to access the API-calls with CDECL calling convention. You must add only CDECL to every function call:

```
__stdcall LTTInit()    == >    __cdecl LTTInitCDECL()  
...
```

The use of the API is divided in several steps:

Initialization:

The call **LTTInit** must be the first call to the API. With this call the internal resources are initialised and the system is scanned for connected LTT devices. Note: This function should be used **only once** at the beginning.

Setup:

The devices can now be configured with a certain setup, as turn on several channels, select its input type and range or select a trigger. Also the amount of data has to be specified. These calls don't communicate directly with the devices. Instead, all setup calls build up a certain configuration which is passed to the devices with **LTTUploadParameters**. This call does an integrity check to see if the configuration is a valid one.

Transfer:

The devices will be started with **LTTStart**, which basically means that the trigger will be active or in the case of no trigger, the measurement process of the devices begins.

Now the transfer of the data from the devices to the RAM-buffer/File can be done in two ways. In both cases the transfer is invoked with **LTTTransfer2Buffers**.

BLOCKED_MODE:

The call **LTTTransfer2Buffers** returns when the transfer of the data to the RAM-buffer/File is completed. This is the easiest way, and the most used one.

NONE_BLOCKED_MODE:

The call **LTTTransfer2Buffers** returns **immediately** after invoking the transfer. Now the transfer to the RAM-buffer/File needs the cooperation of the application. The transfer must be monitored with **LTTTransferStatus** to catch when the transfer is completed or interrupted.

In STREAM-mode the internally used buffer is mostly smaller then the requested transfer amount of data, so this buffer is used like a ring-buffer. To avoid the overwriting of already in the buffer transferred data, the application must read out blocks of already transferred data. If the call **LTTTransferStatus** signals that there is an amount of data ready for read-out, then the start index and the length of the block inside the buffer must be obtained with **LTTData_GetBlock**. When the processing of the block is finished, **LTTData_BlockDone** must be called. Now go back to monitoring the transfer with **LTTTransferStatus**.

Deinitialization:

To close all internal resources, call **LTTDeInit**.

Other calls:

The API provides a number of different informative calls. They are not necessary for the data transfer, but quite a lot of information about the status and the devices can be obtained.

Source Code samples

In the following a source code sample for each of the transfer modes is provided. Both sample are written in plain C:

BLOCKED_MODE:

```
unsigned short usMaxAn, usMaxDig, usMaxRPM, usMaxMath;
short sBuffer[128*1024];
long lAmount;

...
LTTInit( &usMaxAn, &usMaxDig, &usMaxRPM, &usMaxMath );

...
// channel 0 on, +/-10V range, single ended DC+
LTTAnalogChannel( 0, 1, 10000, 1 );
// channel 3 on, +/-2V range, diff. ended DC
LTTAnalogChannel( 3, 1, 2000, 3 );

// trigger at ch 0, LEVEL, 2V-threshold,
// pos. slope, 200mV sensitivity
LTTTrigger( 1, 0, 0, 0, 0, 2000, 200 );

// sample rate: 1,04 Mhz <==> 960 ns
LTTSampleTime( 960, 0 );

// RAMBuffer: 32kS per channel ==> 2 * 32 * 1024 * 2 = 128 kB
LTTRAMBufferSetting( &sBuffer, 32 * 1024, 2 );

// submit to device(s)
if( LTTUploadParameters() == 0 ) {
    // start LTT device(s)
    LTTStart();

    // invoke transfer
    lAmount = LTTTransfer2Buffers( 0, 0 );
}

...
// clean up
LTTDeInit();
```

NON_BLOCKED_MODE:

```
unsigned short usMaxAn, usMaxDig, usMaxRPM, usMaxMath;
unsigned long ulStartIdx, ulLength;
short sBuffer[128*1024];
long lStatus;

...
LTTInit( &usMaxAn, &usMaxDig, &usMaxRPM, &usMaxMath );

...
// channel 5 on, +/-100V range, 1MOhm, SE DC+
LTTAnalogChannel( 5, 1, 100000, 7 );
// channel 10 on, +/-5V range, SE DC-
LTTAnalogChannel( 10, 1, 5000, 1 );
// digital channel
LTTDigitalChannel( 0, 1 );

// trigger at ch 5, REGION, 70V upper threshold
// -40V lower threshold, exit region
LTTTrigger( 1, 1, 5, 0, 2, 70000, -40000 );

// sample rate: 1,04 Mhz <==> 960 ns
LTTSampleTime( 960, 0 );

// STREAMuffer: 1MS per channel, buffer 64kS
LTTSTREAMSettings( &sBuffer, 64 * 1024, 1024 * 1024, 3 );

// submit to device(s)
if( LTTUploadParameters() == 0 ) {
    // start LTT device(s)
    LTTStart();

    // invoke transfer
    LTTTransfer2Buffers( 0, 0 );

    // monitor transfer
    lStatus = 0;
    while( lStatus <= 0 ) {
        lStatus = LTTDataTransferStatus();

        if( lStatus == -255 )
            // error occurred
            break;

        else if( lStatus == -1 )
            // to something
            DoSomething();

        else if( lStatus == -2 || lStatus > 0 ) {
            LTTData_GetBlock( &ulStartIdx, &ulLength );

            // do something with data or
            // simply call done!

            LTTData_BlockDone();
        }
    }
}
```

```
    }  
}  
  
...  
// clean up  
LTTDeInit();
```

LTTInit

[Device]

Declaration:

```
long __stdcall LTTInit(  
    unsigned short *pNumCh,  
    unsigned short *pNumDigCh,  
    unsigned short *pNumRPM,  
    unsigned short *pNumMathCh  
);
```

Description:

Initializes internal resources and searches for all connected LTT devices.

Input:

pNumCh
number of available analog channels

pNumDigCh
number of available digital channels

pNumRPM
number of available RPM channels

pNumMathCh
number of available math channels

Return:

Number of LTT devices found.

LTTDeInit



Declaration:

```
void __stdcall LTTDeInit( void );
```

Description:

Shuts down internal systems and frees resources.

Input:

Return:

LTTResetParameter

[Device]

Declaration:

```
long __stdcall LTTResetParameter( long device_reset );
```

Description:

Resets API-Parameter internally to default setup.

Input:

device_reset
option to reset the devices
0 – not passed; only API is reset
>0 – additionally resets devices;

Return:

LTTTrigger



Declaration:

```
long __stdcall LTT Trigger(  
    unsigned short Trigger,  
    unsigned short Slope,  
    unsigned short Channel,  
    unsigned short ChType,  
    unsigned short TrigType,  
    long x1,  
    long x2  
);
```

Description:

Enables or disables the trigger channel and sets trigger parameters.

CAUTION:

The channels which will be used as trigger channels must be turned on!

Input:

Trig

- 0 - trigger OFF
- 1 - trigger ON

Slope

Level, Comparison + Delta:

- 0 - positive slope
- 1 - negative slope

Comparison:

- 0 - bigger then threshold
- 1 - smaller then threshold

Region:

- 0 - enter region
- 1 - exit region

Channel

set trigger channel (index begins with 0)

ChType

- 0 - analog channel
 - 1 - digital channel
 - firmware 6xxx
 - x1 - digital trigger bit [8..15]
 - firmware 7xxx
 - x1 - digital trigger bit [10..15]
- CAUTION:
only the specified bits are allowed !!!

TrigType

- | | |
|----------------|---------------------|
| analog | digital |
| 0 – Level | 0 - single bit mode |
| 1 – Comparison | 1 - multi bit mode |
| 2 - Region | |

4 - Level + Delta

x1

Level, Comparison, Comparison + Delta:
Threshold [mV]

Region:
Lower Threshold [mV]

x2

Level:
Sensitivity [mV]

Comparison:
not used

Region:
Upper Threshold [mV]

Comparison + Delta:
minimal difference to previous sample [mV]

Return:

- 0 if successful.
- 1 if channel doesn't exist.
- 2 level trigger (TrigType=0) can't have negative hysteresis.
- 4 level+delta trigger (TrigType=4) invalid range for x2 ($0 < x2$).
- 5 invalid digital trigger channel (ChType=1) .

LTTTrigger_SetPreTrigger



Declaration:

```
long __stdcall LTTTrigger_SetPreTrigger (  
    unsigned long dwpretrigger  
    unsigned long dwPreSampleTimeSlow  
);
```

Description:

Sets the pretrigger length per channel. Units are in kilo samples.

For example, desired is a 500ms pretrigger using a 400ns sampling rate. The pretrigger length is calculated.

$$500\,000\,000\text{ ns} (1\text{ Sample} / 400\text{ ns}) = 125\,000\text{ samples}$$

$$125\,000\text{ samples} / (1024\text{ samples}) = 122.070$$

Must be integer number so either 122 (499.7 ms) or 123 (503.8ms) are valid

pretrigger is only allowed under certain conditions. Per devices only one or a pair number of active channels are allowed.

The pretrigger length is limited to 60 Megasamples (120*1024*1024 byte).

So with 6 active channels the maximum pretrigger per channel is 10 Megasamples.

Input:

dwpretrigger

length of pretrigger as multiples of kilo-sample blocks

dwPreSampleTimeSlow

sample time in [ns] for pretrigger phase

can be zero (pretrigger clock equals posttrigger clock)

Return:

- 0 OK
- 1 No trigger active
- 2 Wrong pretrigger length

LTTTrigger_SetReactivationTime



Declaration:

```
long __stdcall LTTTrigger_SetReactivationTime(  
    unsigned long dwTime  
);
```

Description:

Sets reactivation time for Trigger. Typical dead-time for reactivation $<5\mu\text{s}$. Units are in [ns]. The timer resolution is in steps of 100ns. The valid range depends on the used sample time:

0	- Reactivation timer OFF
$4 * \text{Sampletime} \dots 1\text{s}$	- Valid reactivation time

CAUTION:

This option can't be activated together with LTTTrigger_Setpretrigger!! Both are mutually exclusive.

Input:

dwTime
 reactivation time in [ns]

Return:

0	OK
-1	No trigger active
-2	Reactivation time too big

LTTTrigger_SetABAMode



Declaration:

```
long __stdcall LTTTrigger_SetABAMode(  
    unsigned long dwTime  
    unsigned long dwABASampleTimeSlow  
);
```

Description:

Sets trigger in ABA.Mode

CAUTION:

This function can't be used with LTTTrigger_Setpretrigger!!
They are mutually exclusive.

Input:

dwTime

Time after which the trigger
changes back to pretrigger speed
allowed values:

0 - No ABA mode
100ns..1s - allowed range

dwABASampleTimeSlow

Sample time in [ns] before
trigger has happened and to
which is switched back after
timer has expired.

CAUTION:

Not all times are allowed.

Return:

0 OK
-1 No trigger active
-2 Reactivation time too big

LTTAnalogChannel



Declaration:

```
long __stdcall LTTAnalogChannel(  
    unsigned short Channel,  
    unsigned short OnOff,  
    unsigned long chRange,  
    unsigned short AmpType  
);
```

Description:

Turns on/off analogs channels and apply the channel settings.

CAUTION:

The 16 Channel devices have some hardware constraints regarding the possible channel settings. Due to the internal design it's not possible to turn on a backside channel without at least one active front channel. The RPM channel counts as back channel and the digital channel as front channel:

0, 9, D1	valid
12, D1	valid
7, RPM	valid

but

14, 15	not valid
RPM	not valid

Input:

Channel

channel ID

OnOff

0 - channel OFF
1 - channel ON

chRange

channel range in [mV]

Allowed settings:

[1000, 2000, 2500, 5000,
10000, 20000, 25000, 50000,
100000, 200000, 250000, 500000]

AmpType

- 0 - GND
- 1 - DC, Single ended +
- 2 - DC, Single ended -
- 3 - DC, Differential ended

- 4 - AC, Single ended +
- 5 - AC, Single ended -
- 6 - AC, Differential ended

- 7 - 1MOhm, DC, Single ended +
- 8 - 1MOhm, DC, Single ended -
- 9 - 1MOhm, DC, Differential ended

Return:

- 0 if successful
- 1 if channel doesn't exist
- 2 invalid AmpType
- 3 AmpType doesn't match LTT-Hardware
- 4 Channel range doesn't match AmpType settings

LTTDigitalChannel



Declaration:

```
long __stdcall LTTDigitalChannel(  
    unsigned short Channel,  
    unsigned short OnOff  
);
```

Description:

Turns on/off digital channels.

CAUTION:

See LTTAnalogChannel for details about valid channel settings!

Input:

Channel
channel ID, which to be modified.

OnOff
0 - channel OFF
1 - channel ON

Return:

0 if successful
-1 if channel doesn't exist.

LTTTRPMChannel



Declaration:

```
long __stdcall LTTTRPMChannel(  
    unsigned short Channel,  
    unsigned short OnOff  
);
```

Description:

Turns on/off RPM channels.

CAUTION:

Only the RPM-channel at the master device per device stack is supported.

See `LTTAnalogChannel` for details about valid channel settings!

Input:

Channel

channel ID, which to be modified.

Must be 0 !!

OnOff

0 - channel OFF

1 - channel ON

Return:

0 if successful

-1 if channel doesn't exist.

LTTAnalogAutoZero

[Device]

Declaration:

```
long __stdcall LTTAnalogAutoZero(  
    unsigned long chnl_cnt,  
    unsigned long *p_chnl_list  
);
```

Description:

Automatic ReZero of GND-Level for selected active channels in current range.

Input:

chnl_cnt
length of channel-list

p_chnl_list
pointer to channel-list for rezero'ing

Return:

- 0 - successful
- 1 - no LTT devices present
- 2 - invalid channel-list pointer
- 3 - invalid channel in channel-list
- 4 - trying to rezero non-active channel
- 5 - upload-error for rezero; current setup is not valid
- 6 - error in data transfer

LTTAnalogAutoZeroClear

[Device]

Declaration:

```
long __stdcall LTTAnalogAutoZeroClear(  
    unsigned long chnl_cnt,  
    unsigned long *p_chnl_list  
);
```

Description:

Clear's AutoZero offsets for selected channels in current range.

Input:

chnl_cnt
length of channel-list

p_chnl_list
pointer to channel-list for rezero'ing

Return:

- 0 - successful
- 1 - no LTT devices present
- 2 - invalid channel-list pointer
- 3 - invalid channel in channel-list

LTTOutputImpedance



Declaration:

```
long __stdcall LTTOutputImpedance(  
    unsigned short wAnalogChannel,  
    unsigned short wConnectionType  
    unsigned long dwImpedance  
);
```

Description:

Sets output impedance of signal source for specified input coupling.

Input:

wAnalogChannel
Analog channel ID

wConnectionType
type of input coupling
0 – SE+ coupling
1 – SE- coupling
2 - DE coupling

dwImpedance
impedance value

Return:

0 if successful
-1 if channel doesn't exist.
-2 invalid connection type.

LTTSampleTime



Declaration:

```
long __stdcall LTTSampleTime(  
    unsigned long SampTime,  
    short TimeFlag  
);
```

Description:

Sets the sampling time.

Input:

SampTime

Sample time in nano-seconds.

CAUTION: not all times are allowed!

TimeFlag

If sample time is between two allowed sample times, this parameter indicates, which of both is chosen for legal sample time. Chooses the sampling time below (1) or above (0)

Return:

The actual sampling time used.

LTTGetRAMBufferSettings



Declaration:

```
long __stdcall LTTGetRAMBufferSettings(  
    unsigned long dwSamplesPerChannel ,  
    unsigned long dwChannels  
);
```

Description:

Gives back needed buffer size in bytes.

This is a helper function to calculate the required buffer size for a certain channel settings and data amount.

Input:

dwSamplesPerChannel
 Number samples per channels

dwChannels
 Number of channels

Return:

The size of the buffer in bytes.

LTTRAMBufferSettings



Declaration:

```
long __stdcall LTTRAMBufferSettings(  
    short *pwBuffer,  
    unsigned long dwSamplesPerChannel,  
    unsigned long dwChannels  
);
```

Description:

Setting up the RAM buffer system for data transfer.

The buffer must be allocated by the application. The size must match the channel settings and data amount.

Input:

pwBuffer

Pointer to RAM buffer.

dwSamplesPerChannel

Number of samples per channel.

dwChannels

Number of channels.

Return:

- 0 if successful
- 1 STREAM already set up !!
RAM and STREAM are exclusive buffer settings.
- 2 length per channel is invalid

LTTFileBufferSettings



Declaration:

```
long __stdcall LTTFileBufferSettings(  
    char *pcFileName,  
    unsigned long dwMode,  
    unsigned long dwMaxCount,  
    unsigned long dwSamplesPerChannel,  
    unsigned long dwChannels  
);
```

Description:

Sets the parameters to measure to a file.

Input:

pcFileName
chooses which channel to be modified

dwMode
Reserved! Always 0.

dwMaxCount
Reserved! Always 0.

dwSamplesPerChannel
Number of samples per channels

dwChannels
Number of channels

Return:

0	success
-1	file extension not specified
-2	wrong mode
-3	file size too large
-4	unknown file extension

LTTSTREAMSettings



Declaration:

```
long __stdcall LTTSTREAMSettings(  
    short *pwBuffer,  
    unsigned long dwBufferSize,  
    unsigned long dwSamplesPerChannel,  
    unsigned long dwChannels  
);
```

Description:

Setting up STREAM buffer system.

In contrary to RAM buffer system, it's possible to transfer data much bigger than the specified buffer size. It gives more control on the transfer to the application, but also is not so easy to work with. It requires the cooperation of the application during the transfer.

Input:

pwBuffer

Pointer to RAM-Buffer.

dwBufferSize

Buffer size in samples

dwSamplesPerChannel

Number of samples per channel

dwChannels

Number of channels

Return:

- 0 everthing's OK
- 1 ERROR; sample per channel too small
- 2 ERROR; buffer size too small
- 3 ERROR; buffer size not multiple of channel sequence size
- 4 ERROR; RAM buffer already set

LTTBufferReset



Declaration:

```
void __stdcall LTTBufferReset( void );
```

Description:

Resets the transfer buffers systems.

Input:

Return:

LTTUploadParameters

[Device]

Declaration:

```
long __stdcall LTTUploadParameters( void );
```

Description:

Checks and sends the parameters to the LTT devices.

The parameters have been gathered using LTTAnalogChannel, LTTSampleTime, LTTRAMBufferSetting, etc. but the device has not yet been configured.

LTTUploadParameters makes a final check to see if the combination of all the parameters don't make an error and then if successful configures the devices. The error codes are additive.

Input:

Return:

The return value is a sum of the following error codes.

- 0 Everything OK
- 1 LTT device not found or ASPI dll not found
- 2 Wrong sample rate by current channel setting
- 4 No channels active
- 10 Wrong channel combination!! When backside channels used, at least one front channel must be active also!
- 20 Trigger channel not active
- 40 pretrigger settings wrong - no channel set
- 100 pretrigger settings wrong - invalid pretrigger length
- 200 No Buffer settings specified!!
- 400 amount of active channels does not fit channel parameter in buffer settings.
- 1000 Math error
- 2000 No active channel for master device.
- 4000 - trigger timer and PRE-trigger are mutually exclusive settings.
- 10000 - trigger timer too small;
condition: timer > 4 * sampletime

LTTResetParameter



Declaration:

```
void __stdcall LTTResetParameter( long device_reset );
```

Description:

Resets all parameter settings in LTT2API to default. When device_reset is 0 no upload to the device(s) occurs and only the dll parameters are affected. If, however, device_reset equals 1 the channels are set to internal ground, that is OFF.

Input:

device_reset : 1 or 0.

Return:

LTTStart

[Device]

Declaration:

```
long __stdcall LTTStart( void );
```

Description:

Starts the LTT devices and activates the trigger if on.

Input:

Return:

0 Always

LTTStop



Declaration:

```
void __stdcall LTTStop( void );
```

Description:

Stops the measurement at LTT devices.

Input:

Return:

LTTCheckTrigger



Declaration:

```
long __stdcall LTTCheckTrigger(  
    unsigned long MilliSecond  
);
```

Description:

Checks if the trigger has been fulfilled.

In the case the trigger is still active, the call waits the specified amount of time before the call returns.

Input:

MilliSecond
time in milliseconds to wait after a negative trigger result.

Return:

0	trigger not fulfilled
1	trigger successful

LTTTransfer2Buffers

[Device]

Declaration:

```
long __stdcall LTTTransfer2Buffers(  
    unsigned short BufferType,  
    unsigned short BlockMode  
);
```

Description:

Transfers the data from the recorder to the assigned buffers.

This call initiates the real transfer. The transfer can work in BLOCKED or NONE_BLOCKED mode.

BLOCKED:

The call returns, when the transfer is done.

This is not possible with the STREAM buffer system.

NONE_BLOCKED:

The transfer is initiated and the call returns immediately.

The Transfer must be monitored with LTTDataTransferStatus.

Input:

BufferType

Reserved; Always 0.

BlockMode

0 - (Blocked Mode)

1 - (NON-Blocked Mode)

Return:

- 1 No LTT devices.
- 2 No buffer type set.
- 3 Transfer is active, so can't invoke another transfer !!
- 4 File transfer already done. Reconfigure file buffer settings!

BLOCKED mode:

>=0 Number of transferred samples

CAUTION:

The amount of transferred samples must be checked against the requested data length. If the amount of transferred samples is less than the requested amount, an irrecoverable error has occurred during the transfer, which led to a break of the transfer. In the data buffer the correctly transferred data are stored up to the point where the error has occurred!!

NONE_BLOCKED mode:

0transfer

invoked

LTTDataTransferStatus



Declaration:

```
long __stdcall LTTDataTransferStatus( void );
```

Description:

Monitors the status of the transfer.

This function is necessary for the **NONE_BLOCKED** Mode of transfer.

Input:

Return:

- 0 nothing TODO
- 1 waiting for trigger
- 2 new block available
- 255 error happened

- >0 transfer complete, number of transmitted samples

CAUTION:

The amount of transferred samples must be checked against the requested data length. If the amount of transferred samples is less than the requested amount, an irrecoverable error has occurred during the transfer, which led to a break of the transfer. In the data buffer the correctly transferred data are stored up to the point where the error has occurred!!

LTTDataTransferBreak

[Device]

Declaration:

```
long __stdcall LTTDataTransferBreak( void );
```

Description:

Breaks a running transfer.

Input:

Return:

0	OK
-1	error occurred

LTTData_GetCounter



Declaration:

```
long __stdcall LTTData_GetCounter( void );
```

Description:

Gives back the amount of transferred bytes.

This call can be used in **NONE_BLOCKED** Mode of transfer to monitor the progress of the transfer.

Input:

Return:

amount of up to now transferred samples

LTTData_GetBlock



Declaration:

```
long __stdcall LTTData_GetBlock(  
    unsigned long *dwStartIndex,  
    unsigned long *dwAmount  
);
```

Description:

Gives back a data block, which can safely be read by the application.

CAUTION:

Only used in **NONE_BLOCKED** mode of transfer. Transfer must be monitored with LTTTransferStatus, which indicates if a data block is ready for read-out.

Input:

dwStartIndex

Gives back the start index of the block of data.

dwAmount

Gives back the total number of sample available in the block.

Return:

0	if no LTT devices found
1	if successful

LTTData_BlockDone



Declaration:

```
long __stdcall LTTData_BlockDone( void );
```

Description:

Tells LTTAPI that the block obtained with the call LTTData-GetBlock is not anymore used by the application.

Input:

Return:

0Always

LTTGetCalibTemperature



Declaration:

```
long __stdcall LTTGetCalibTemperature(  
    unsigned long dwDevice  
);
```

Description:

Gives back internal temperature of specified device at last calibration.

Input:

dwDevice

Device ID for which the calibration temperature is requested

Return:

Internal devices temperature in degrees C

LTTGetChannelIDSequence



Declaration:

```
void __stdcall LTTGetChannelIDSequence(  
    unsigned short *pSequenceList  
);
```

Description:

Gives back channel ID in channel sequence.

Input:

pSequenceList
pointer to list. Application must provide sufficient memory !!

Return:

LTTGetChannelSequenceLength



Declaration:

```
long __stdcall LTTGetChannelSequenceLength( void );
```

Description:

Gives back length of channel sequence.

Input:

Return:

Length of channel sequence.

LTTGetResolution



Declaration:

```
long __stdcall LTTGetResolution( void );
```

Description:

Gets the current bit resolution for the sampling time.

Input:

Return:

12	12Bit resolution;
14	14Bit resolution;
15	15Bit resolution;
16.	16Bit resolution;

LTTGetChannelRange



Declaration:

```
long __stdcall LTTGetChannelRange(  
    unsigned long dwAnalogChannel  
);
```

Description:

gives back range for the requested channel.

Input:

dwAnalogChannel
Analog channel index

Return:

maximum range; +/- value in [mV]

LTTGetTemperature

[Device]

Declaration:

```
long __stdcall LTTGetTemperature(  
    unsigned short wDevice  
);
```

Description:

Measure temperature in specified device.

Input:

wDevice

Device ID for which the calibration temperature is requested

Return:

current internal devices temperature

LTTGetDLLInfo



Declaration:

```
long __stdcall LTTGetDLLInfo(  
    char *pcBuffer  
);
```

Description:

Obtain identification string of the DLL.

Input:

pcBuffer
pointer to buffer where the null terminated string will be put.
Must be at least 64 bytes.

Return:

0 Always

LTTGetDLLVersion



Declaration:

```
long __stdcall LTTGetDLLVersion( void );
```

Description:

Obtain version number of the DLL.

The version number is build in the following manner:

```
( LTT2API_MAJOR << 16 ) | ( LTT2API_MINOR << 8 ) | LTT2API_BUG
```

Input:

Return:

Version number

LTTGetDeviceInfo



Declaration:

```
long __stdcall LTTGetDeviceInfo(  
    unsigned long dwDevice,  
    char *pcBuffer  
);
```

Description:

Obtain identification string of the LTT device.

Input:

dwDevice

Selects the LTT device.

pcBuffer

pointer to buffer where the null terminated string will be put.
Must be at least 64 bytes.

Example:

LTT-184/16 SN:3214 FW:6.0.607 ch:(1 - 16)

Return:

0 if successfull
-1 if LTT device not found.

LTTGetDevices



Declaration:

```
long __stdcall LTTGetDevices( void );
```

Description:

Returns the number of devices found.

Input:

Return:

number of devices found.

LTTSetInternReference



Declaration:

```
long __stdcall LTTSetInternReference(  
    unsigned long dwDevice,  
    unsigned long dwRefIdx  
);
```

Description:

Sets all the channels for a given device to the internal reference.

Input:

dwDevice

Selects the LTT device.

dwRefIdx

index of reference value table

0 <=> GND

1 <=> 857mV

2 <=> 3.00V

3 <=> V_{temp}

4 <=> $V_{temp} - 3.00V$

5 <=> $3.00V - 3.00V$

6 <=> 0 - 857mV

Return:

0 OK

-1 invalid device index

-2 invalid reference index

LTTGetChannelConverter



Declaration:

```
double __stdcall LTTGetChannelConverter(  
    unsigned long dwAnalogChannel  
);
```

Description:

Gives back conversion factor for requested analog channel.
Applying this factor to a 16-Bit data sample value will convert the value in units of [mV]:

$$[\text{mV}] = [\text{16Bit}] * \text{LTTGetChannelConverter}([\text{channel}])$$

Input:

dwAnalogChannel
Analog channel index

Return:

conversion factor

Index

19-Zoll	6, 10	gewählter Kanal.....	31
Ableitung	24	Gitternetzlinien	27
AC/DC	20	Glättungsfilter	24
Achsenbeschriftung.....	31	GotoMax.....	32
ADC	17	GotoMin.....	32
Addition	24	grafische Benutzeroberfläche.....	15
Advanced Display Settings.....	28	Grafische Darstellung	35
Analog Out	7	Grenzwerte	30
Analog to Digital Converter	17	Grid.....	27
Analyzer	26	Histogram	24
Auto Scaling	21	Hysteresis	29
Average Screens.....	24	Signalbereich.....	22
Bildschirm.....	16	Impedanzumschaltung	20
Bkgd, Background	27	Installation	15
BNC-Eingangsbuchsen	7	Integral	24
Channel	19, 30	Kaskadierung	13
Channel Analyzer.....	26	LED Anzeigen	7
Channel Finder.....	36	Leuchtdioden.....	20
Comparison	30	Level.....	29
Cursor.....	21, 31	Level+Delta	30
Cycle	39	Level-Trigger	36
Darstellung	20	Load Data.....	17, 35
Datei	34	LTTCh	32
Datei öffnen	37	LTTDATA	15
Default.....	35	LTTInit	41
Delay Data Transfer	18	LTTview.....	15
Delay Time	31	Main.....	35
Differential Ended.....	20	Manual Scaling.....	22
Digital Input	8, 10, 19, 25	Masse.....	9, 20
Display.....	26, 35, 37	Master	13
Display Channel	31, 36	Math	22, 23
DLL LTT2API.....	40	MathCh.....	32
Dot Display	25	Mathematik-Kanäle	23
Dot Join	27	mathematischen Funktion	23
Effektivwertanalyse	24	Maxima.....	30
Eingangs-Kopplung	20	Maximum.....	31, 32
Eingangswiderstand	17, 21	Meßdaten	28
Einheit	22, 28	Messdaten-Analyse.....	35
Empfindlichkeit	29, 37	Meßdauer	37
Export Screen.....	35	Minima	30
externer Takt	17, 19	Minimum.....	31
Extrema in Region.....	30	Missing Event	31
Fast Save	35	Multiple Recordings.....	34
FFT	24	Multiple Trigger.....	29, 30
File Size.....	34	Multiplikation.....	24
Files	33	Netzteil	9
Filter	24	New Boot Virus.....	15
Fine Adjust	17	Noise Suppressor.....	30
FireWire.....	14, 15	Nullpunkt	29
Full Scale Range	30	Offset.....	36
Funktion.....	40	Online Mode	35
Funktionen.....	36	Online und Offline.....	16
Genauigkeit	17	OnlineMax	32

Oszilloskop	22, 35	Source Impedance	18
Persistence	27	Speichern	37
pretrigger	31, 34, 37	Speichertiefe	31, 34
pretriggerClock	31	Speicherung	34
Pulsbreite	24	Start Button	7
Quellkanal	23	Status	39
Realtime	35	Steigung	29, 37
Record	35	Stop	35
Region	30	Summenabtastrate	17
Release	39	Symbolleiste	37
Reset	9	Synchronisation	13
Reset Trigger after	31	Task Description	25
RPM (Rotation per Minute)	19	Technische Daten	12
Sample Rate	17	Temperature	18
Save Screen	34	Threshold	30
Saving Mode	34	Total Transferred Bytes	18
Scale	21, 28	Totzeit	31
Schwellwert	29	Trigger	29
Screen at Once	27	Trigger Position	29
SCSI	15	Type	29
SCSI-ID	14	Umdrehung	19, 28
Sensitivity	30	Umrechnungsfaktor	28
Show Display	27	Upper und Lower Limit	30
Signalausreißer	30	View Port	28
Single	29	Wertebereich	30
Single Ended	20	x-Axis	28
Single Recording	34	yDiv	22, 24
Single Shot	39	Y-Offset	22, 24
Skalierung	21, 22, 28	Zeitdauer	17
Slave	13	Zero Position	28
Slope	29, 30	Zoom	16, 38
Smooth	24		