

MuSR Electronics: TDC
Version 1.2

Generated by Doxygen 1.7.1

Mon Jun 4 2012 10:43:02

Contents

1	TDC Manual	1
2	General Description	2
2.1	Introduction	2
2.2	Layout	2
2.3	More Information	3
3	Information for the Users	4
3.1	Introduction	4
3.2	Modify/See the TDC Settings with the deltat GUI	4
3.2.1	Modify Settings:	4
3.2.2	Look at the Settings:	8
3.2.3	Save the Settings:	8
4	Information for the Instrument Scientists	8
4.1	Setting up the CFDs	8
4.1.1	Principles of the CFD circuit	9
4.1.2	Modifying CFD Parameters	9
4.1.3	Blocking Mode	11
4.1.4	Threshold	11
4.1.5	Delay	12
4.1.6	Automatic Walk-Adjust	13
4.1.7	Width	13
4.2	Setting up the required logic	14
4.2.1	Introduction to the Logic "Philosophy"	14
4.2.2	The Setup Logic File (Syntax and Examples)	14
4.2.3	Setting Up a Template Logic File	22

1 TDC Manual

This short manual describes the basics of the so-called TDC-Electronics for the time-differential bulk MuSR experiments at PSI.

It contains the following Chapters:

- [General Description](#)
- [Information for the Users](#)
- [Information for the Instrument Scientists](#)

(new sections will be added at a later point)

The manual is available in HTML format at the URL:

<http://lmu.web.psi.ch/facilities/electronics/TDC>

Author

Alex Amato, Andrea-Raeto Raselli and Robert Scheuermann

2 General Description

2.1 Introduction

This section gives a short introduction of the new TDC Electronics concept for the bulk MuSR experiments.

The new electronics consists of a VME crate with programmable constant fraction discriminators ([PSI CFD950](#)), a multihit TDC ([CAEN V1190](#)) for digitizing of time information, and a scaler module ([SIS3820](#)) for rate measurements.

The VME frontend process for readout of the VME TDC data is running on a front-end Linux PC and connected to the VME crate by a [SIS 1100/3100](#) VME-PCI interface. (in the future, the VME readout could be performed by a SHARC processor that is an option of the SIS3100 VME card).

The complete event evaluation is done in software: the TDC is operated in Continuous Storage Mode, and the frontend process reads all the TDC data. It searches for events that fulfill logics conditions. The data are sent to the Analyzer process running on the Backend, which builds the histograms taking into account post- and pre-pilup conditions, as well as logics conditions.

The system is also characterized by its flexibility, as the whole logic diagram is simply stored in special setup files, which can be easily modified and loaded for a desired configuration into the online database (ODB) of the MIDAS DAQ system.

2.2 Layout

The following figure represents schematically the different modules used in the VME crate.

The analog signals are split by an active signal divider (SP950) into a timing branch sent to the CFD (CFD950 with 8 input channels) and a monitoring branch for CFD threshold adjustment (see section [Threshold](#)). Note that as long as the old pTA electronics is still active, one of the splitted analog signals is directed to the old ORTEC CFD935 modules instead of being used for monitoring.

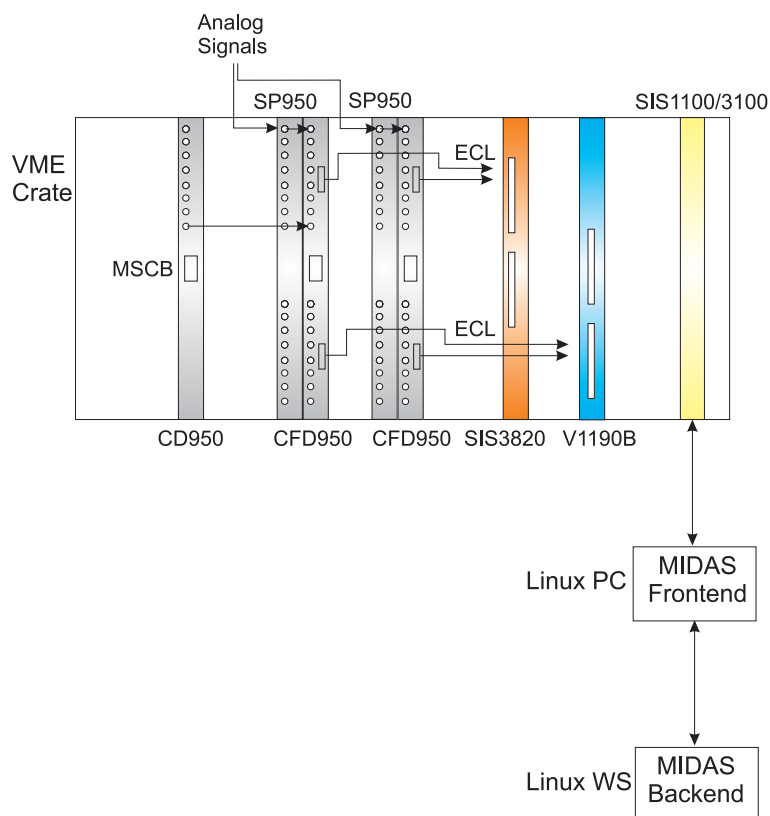


Figure 1: Schematics of a typical VME electronic crate for a bulk muSR experiment. In addition to the CFDs, TDC, Scaler and VME/PCI interface, the crate is usually also equipped with a PSI Clock (CD950). Also NIM/ECL PSI converters (LC950) and Coincidence Units can be installed.

The signals from the CFD's are sent as ECL signals to the TDC V1190 and also to the scaler. If hardware coincidence is necessary, the ECL or NIM signal outputs of the CFD can be used with a PSI Coincidence Unit FC950 (available from Studio E). The signal can be fed back to the TDC and Scaler by the ECL output of the Coincidence Unit.

The signals from the TDC and Scaler are sent to a MIDAS frontend (Linux PC) through the optical link VME/PCI interface. In the front-end Linux PC, the events are checked for coincidence conditions and are sent to the Analyzer process running on the Backend computer, where the histograms are built according to the trigger and logics conditions defined for a particular setup.

Note that for all the modules developed at PSI, a RS485 connection is available. It allows to setup the module with the MSCB (MIDAS Slow Control Bus) protocol through a MSCB submaster (connecting the RS485 bus to the ethernet). Those settings are saved on flash memory on the boards.

2.3 More Information

The following Chapters provide information on how to setup the electronics:

- [Information for the Users](#)
- [Information for the Instrument Scientists](#)

3 Information for the Users

3.1 Introduction

The TDC Electronics is characterized by the fact that, in addition to the pile-up gate conditions, all the necessary coincidence or veto conditions, as well as detectors grouping are fully performed by software. For the users, this permits a large flexibility and a quick change between different logic configurations.

The different logic configurations ("setups") are stored in special files (so-called "setup files") which can be loaded as desired through the graphic user interface **deltat**.

3.2 Modify/See the TDC Settings with the deltat GUI

The TDC settings can be modified/seen/saved by selecting the tab **TDC Settings** to be active:

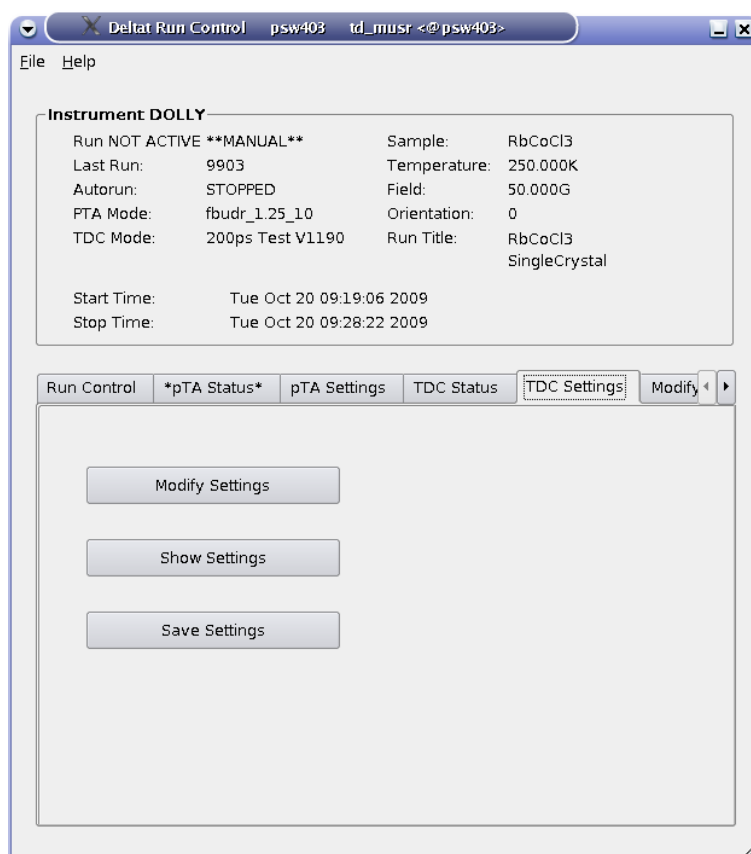


Figure 2: View of the deltat GUI with the tab TDC Settings active.

3.2.1 Modify Settings:

By pressing the button **Modify Settings**, you have the possibility to modify the set-up of the TDC used for the experiment (and saved in the ODB -- online database). Each setup configuration is saved on a so-called setup file located on the back-end computer. These setup files are maintained by the Instrument Scientist.

A setup file contains all the necessary logic as well as the histogram grouping and information on the t_0 and First Good Bins.

Note that the setup files **do NOT** contain information on the resolution and time-window, which should be selected on the fly when changing the settings.

A Wizard-Tab will lead the user through the setup configuration.

On the first wizard-tab, a drop-down menu will present the available setup files for the instrument.

Hopefully the names should be self-explanatory.

If you have the impression that a setup file is "screwed-up" you can always download its "master" copy which has the extension **.v1190_template** and which are listed at the end of the drop-down menu.

Such "master"copies cannot be overwritten by usual users.

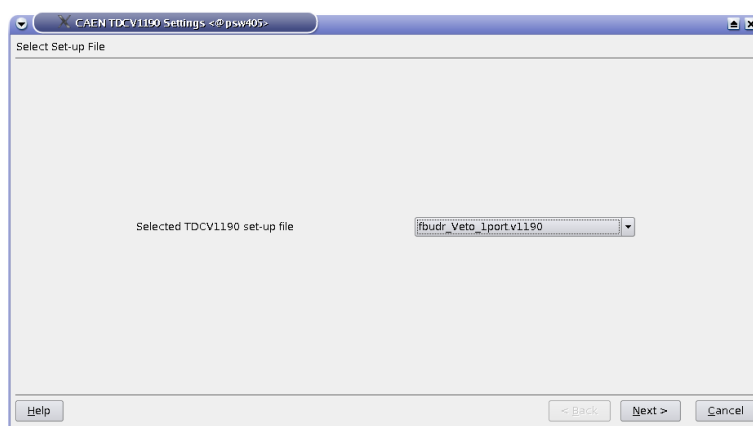


Figure 3: The first wizard tab will provide a choice for the available setup files.

On the second wizard-tab, you will have a list of the selected histograms.

The first column indicates the channel number of the TDC, the second shows the internal histogram names of these histograms, the third column signals whether one of these histograms is added to another one. Finally the last column reports the names of the saved histograms.

Do NOT change these setting unless you know what you are doing...

For example, on the figure, one sees that the channel #9 is stored internally on the histogram "R_ext", and that it is added to the internal histogram "R_int" to finally be saved under the name "Rite" of the data-file.

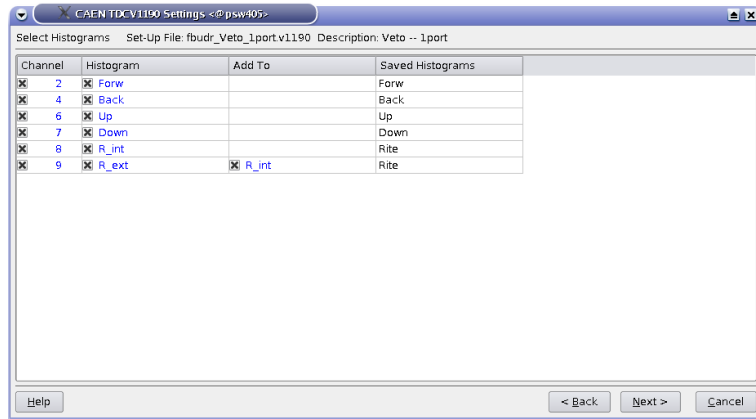


Figure 4: The 2nd wizard-tab will ask you for a confirmation of the histograms to be saved and added.

The 3rd wizard-tab is normally the only wizard-tab which will require some inputs from your side:

- you can choose the format of the output file (**so far only PSIBIN format available...**).
- you can choose the bin resolution of the saved histograms (multiple of 0.1953125 ns/channel which is the base resolution of the TDC). Remember that for the usual MuSR instrument a resolution lower than 3 (i.e. $3 \times 0.1953125 = 0.5859375$ ns) does not make any sense as the detectors have intrinsically a worse resolution.
- the third input is the length of the data-window. It can be entered in bin units (remember that the bin unit is chosen above), in TDC units (i.e. in multiple of 0.1953125 ns) or more easily in microseconds. When an input with given units is performed, the corresponding values on the other units are calculated.

Usually, only data-window of 10 or 5 microseconds are used.

- The next input defines the pileup-window with respect to the data-window.
Do NOT change this setting unless you know what you are doing...
- The final setting defines the Last Good Bin with respect to the data-window.
It is strongly advised to leave this setting ON as the Last Good Bin will be automatically calculated.
- A dialog box containing information about the total number of bins may appear when going toward the 4th wizard-tab.

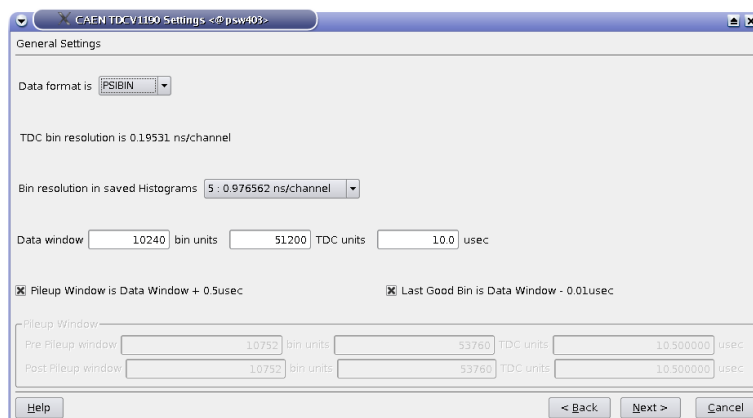


Figure 5: On the 3rd wizard-tab, the time window and the time resolution can be changed.

On the 4th wizard-tab, the time-zero, first good bin and last good bin of the histograms can be set.

- The values can be given either in 'bin units' (resolution depending of the chosen binning) or in 'TDC units' (representing 0.1953125 ns/channel -- light grey columns).
- Usually these values do not need to be changed as the time-zero, first good bin values are rather fixed in TDC units and the last good bin will usually automatically depend on the time-window selected on the previous wizard-tab.
- Note that when attempting to change the TDC settings when a run is active, these options are the only one which can be modified.

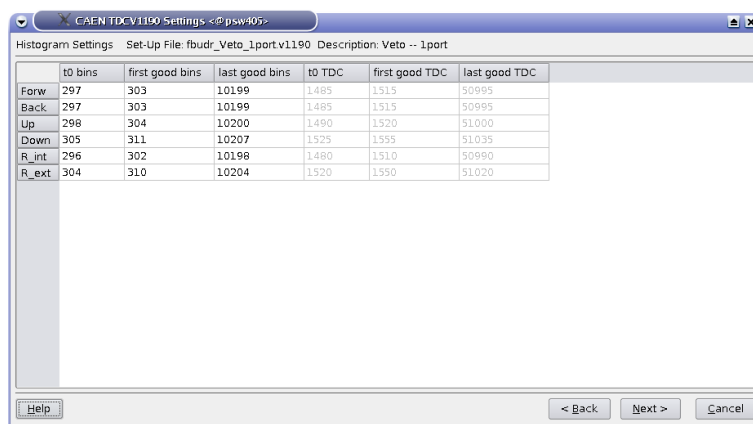


Figure 6: On the 4th wizard-tab, the time-zero, first good bin and last good bin of the histograms can be set.

By going on the next wizard-tab and pressing the button `Apply`, the modified settings are written to the online database ODB.

If necessary, the changes can finally be saved on a TDC setup file.

Remember that the time resolution and length of the data-window are NOT written on the setup file. Therefore it is not necessary to save the file if the only changes that you performed were on these parameters.

3.2.2 Look at the Settings:

By pressing the button **Show Settings** one can show the present settings used by the TDC. A short version is displayed, but an extensive list of the setting parameters can be reached by hitting the button **More . . .**

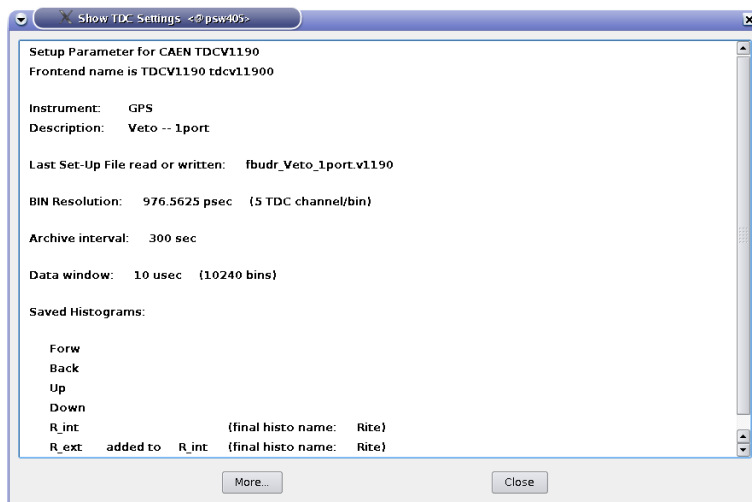


Figure 7: The main TDC settings are display when choosing the *Show Settings* option. More settings are available with the *More...* button.

3.2.3 Save the Settings:

The button **Save Settings** does what is says....

4 Information for the Instrument Scientists

The following sections provide information how to setup the electronics and the logic used by the TDC electronics

- [Setting up the CFDs](#)
- [Setting up the required logic](#)

4.1 Setting up the CFDs

This section describes how to setup the different parameters of the CFDs used with the TDC Electronics:

- [Principles of the CFD circuit](#)
- [Modifying CFD Parameters](#)
- [Blocking Mode](#)
- [Threshold](#)
- [Delay](#)

- [Automatic Walk-Adjust](#)
- [Width](#)

4.1.1 Principles of the CFD circuit

In a *Leading Edge Triggering* discriminator, when the input signal has reached the given threshold, an output signal is produced. If the input signal can significantly vary in amplitude (like in our large detectors) and as the rise time from scintillator signals is independent of the signal amplitude, a time shift (so-called *Walk*) can appear even for simultaneous input signals. This problem is overcome by using a so called *Constant Fraction Discriminator (CFD)*.

In a CFD, the unipolar input signal is turned into a bipolar signal by an appropriate electronic chain. This is done by splitting the input signal with different ratio. One of the signal experiences a certain delay and gets inverted, while the other gets attenuated. Finally both signals are again added to form the CFD bipolar signal. If the delay is chosen so that the zero-crossing occurs when the original signal has its maximum (or better say its minimum), the output signal of the CFD will be independent of the amplitude of the input signal and therefore is not influenced by the input pulse height, i.e. the walk will be strongly reduced .

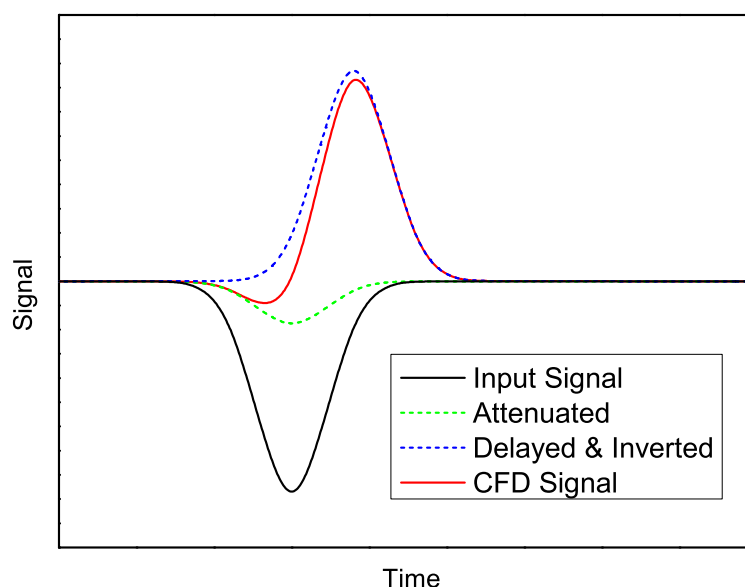


Figure 8: Principle of a CFD discriminator. The input signal is split with one branch being inverted and the other being inverted and delayed. The CFD signal is the sum of the two branches and the CFD output signal will appear at the zero crossing of the CFD signal. For simplification, a Gaussian pulse are assumed.

Hence the main parameters to set for the CFD are: the *Threshold*, the *Delay*, the *"Walk-Adjust"* (i.e. the DC level of the internally attenuated signal to be sure that input signals with different amplitudes are transformed in CFD signals crossing at zero). Note that the CFD950 is equipped with an automatic adjustment of the walk (*Automatic Walk-Adjust* -- See Section [Automatic Walk-Adjust](#)).

4.1.2 Modifying CFD Parameters

4.1.2.1 Modifying parameters through the command line

- login into the backend computer as root

- start the "Midas Slow Control Bus" application invoking: `/usr/local/midas/msc -d mscbNNN`, where **NNN** represents the number of your mscb submaster
- enter the corresponding password
- invoke the command **scan** to find the different mscb nodes
- identify the node(s) corresponding to your CFD(s)
- connect to your CFD(s) using the command **add XX**, where **XX** correspond to the node address of the corresponding CFD
- list all the variables using the command **read**
- modify the variable that you need.
For example to put the CFD in Blocking Mode (see Section [Blocking Mode](#)) the variable 44 ("WIDTH_BU") should be set to 65535:
write 44 65535
- secure your changes in the flash memory: **flash**
- reboot the addressed node: **reboot**
- the setup can be dumped into an XML file using the command **save FILENAME.xml**. Keep in mind that this also includes the address of the node when loading the file onto a new CFD module (**load FILENAME.xml**)

4.1.2.2 Modifying parameters with the GUI **deltat**

The CFD parameters can also be set through the GUI **deltat**.

Be sure to start **deltat** with the special flag.

- on the tab **Modify Devices** choose the entry which corresponds to the CFD(s) (for example **VME_CFD**)
- choose the option **Modify**
- modify the variable that you need.
For example to put the CFD in Blocking Mode (see Section [Blocking Mode](#)), click on the entry **cfdn_width_B_U** and write the value 65535 in the input field.
- at this point, you will be asked whether you want to perform a flash & reboot. You can either perform this operation at this point or when you are done changing all the parameters for a given CFD. If you have more than one CFD and are changing parameters on them, be sure to perform a flash & reboot for each CFD.
- Be sure to scroll down to change these values for all the CFDs in use.

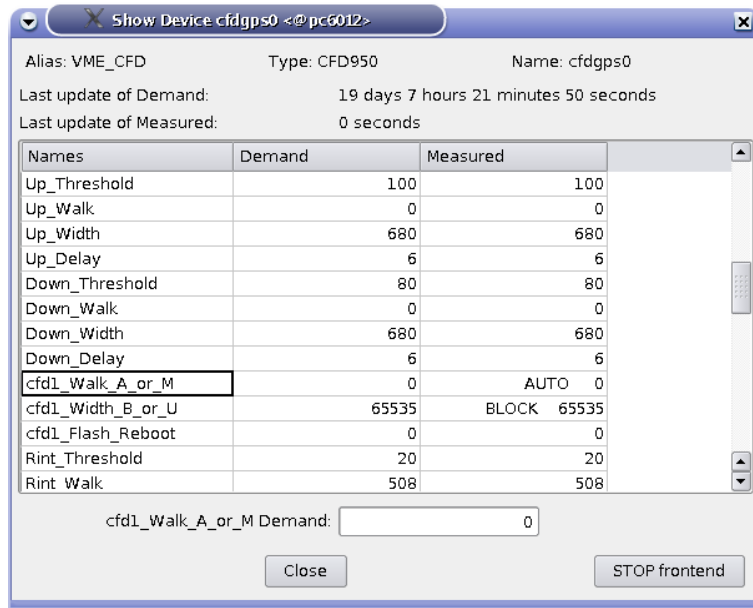


Figure 9: Dialog to change parameters of the CFD. In the present case, the `cfdN_Walk_A_or_M` has been clicked and can be edited in the input field.



Figure 10: Dialog to perform a flash & reboot of the CFD.

4.1.3 Blocking Mode

The first thing to perform is to set the CFD on blocking mode for the output pulse width (basically to prevent ringing).

To set the CFD on blocking mode, one has to set the MSCB variable 44 ("`cfdN_Width_B_U`") to the value 65535.

If, for special measurements, the CFD needs to be put on Updating Mode, the entry `cfdN_Width_B_U` has to be set to 0.

This can be done either from the command line or through the Deltat GUI (see Sections [Modifying parameters through the command line](#) and [Modifying parameters with the GUI deltat](#)).

4.1.4 Threshold

For each channel, the threshold should be set accordingly. As the threshold scales can slightly vary from channel to channel, it is recommended to perform an active check of the set values. This is performed by splitting the analog signal with the splitter SP950:

- one output channel of the splitter is connected to the input of the CFD. The NIM output of the CFD is sent to the oscilloscope.
- the second output of the splitter is sent to the oscilloscope.

The trigger of the oscilloscope is performed by the NIM output of the CFD. This allows one to observe only the analog signals which are above the effective threshold.

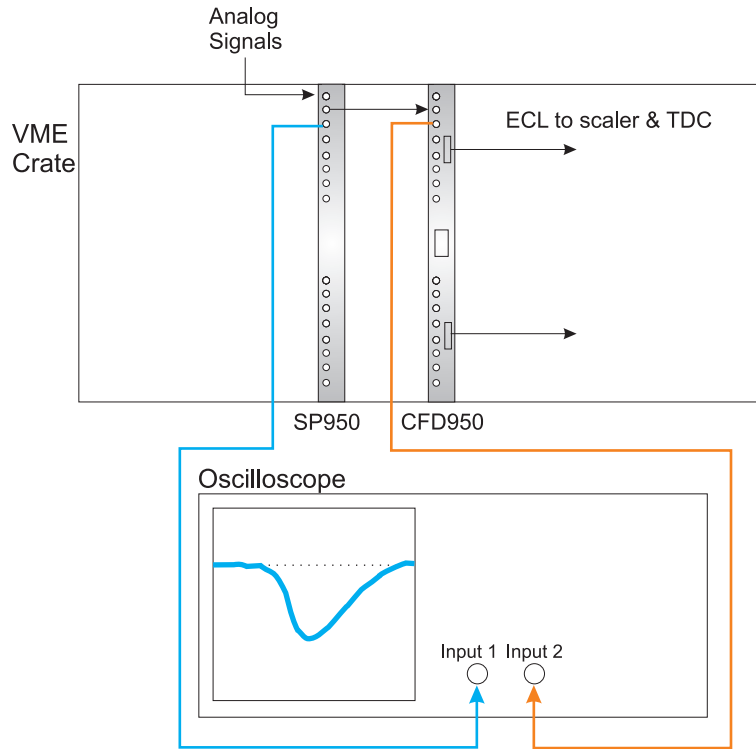


Figure 11: Schematics of the connections to check for the thresholds values. The input 2 is used as trigger.

The threshold values for each channel of the TDC can be modified by setting the MSCB variables 11 to 18 ("THR_CHN", where N is the number of the channel) of the CFD. The value to be set is in millivolts. This can be performed either through the command line (in a similar way as described in the Section [Modifying parameters through the command line](#)) or through the GUI deltat (see Sections [Modifying parameters through the command line](#) and [Modifying parameters with the GUI deltat](#)).

4.1.5 Delay

When equipped with standard delay boards the delay time for the internal non-attenuated inverted signal can be set in 0.5 ns steps. As the CFD board has an internal delay of 1.0 ns, the overall delay can be adjusted by steps from 1.0 to 8.5 ns (i.e. 15 steps).

An approximation of the correct delay is obtained by determining the rise time (t_{rise}) of the input signal. Consequently, the delay time is set to:

$$t_{delay} = t_{rise}(1 - f),$$

where f is the attenuation factor (in our case 0.2).

The following figure represents different situations with good delay, too long delay and too short delay. The CFD signal can be looked with an oscilloscope connected to the *Monitor* output of the CFD950 (see the [CFD950 Manual](#) for details).

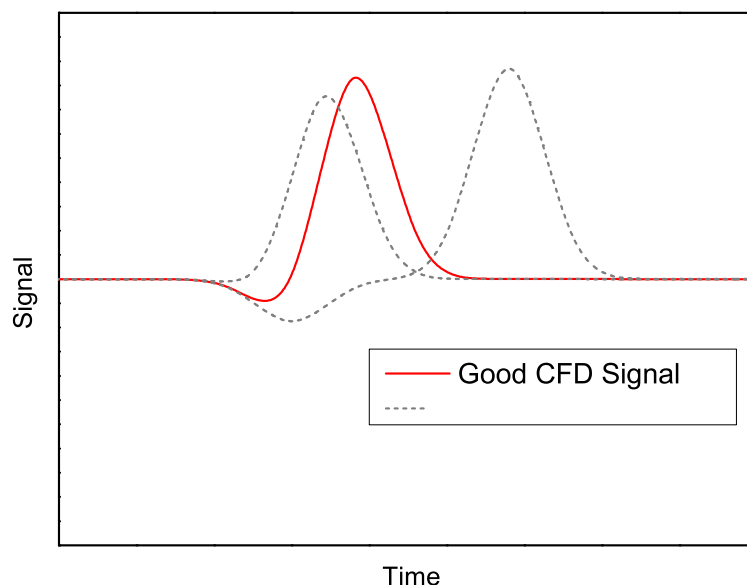


Figure 12: CFD signals with good, too short or too long delay.

The delay values for each channel of the TDC can be modified by setting the MSCB variables 35 to 42 ("DELAY_CN", where N is the number of the channel) of the CFD. The value to be set is between 0 and 15 corresponding to 0.5 ns steps. For example a value of 6 will correspond to a delay of 4 ns (6 steps of 0.5ns plus an internal delay of 1 ns).

The modification of the delay parameter can be performed either through the command line (in a similar way as described in the Section [command_line](#)) or through the GUI [deltat](#) (see Sections [Modifying parameters through the command line](#) and [Modifying parameters with the GUI deltat](#)).

4.1.6 Automatic Walk-Adjust

At this point, one can set the CFD on Automatic Walk-Adjust.

The status of the Automatic Walk-Adjust can be modified by setting the MSCB variable 43 ("WALK_A_M"). It should be set to zero for Automatic Walk-Adjust. Normal measurements are performed with Automatic Walk-Adjust.

For Manual Walk-Adjust this variable should be set to 65535.

Again, this can be done either from the command line or through the [Deltat GUI](#) (see Sections [Modifying parameters through the command line](#) and [Modifying parameters with the GUI deltat](#)).

4.1.7 Width

The width of the pulse signals can be set independently for each channel. This is done by setting the MSCB variables 27 to 34 ("WIDTH_CN", where N is the number of the channel) of the CFD. The output pulse width is adjustable on 10-bit from 5ns to 50ns with a non-linear relation for intermediate values.

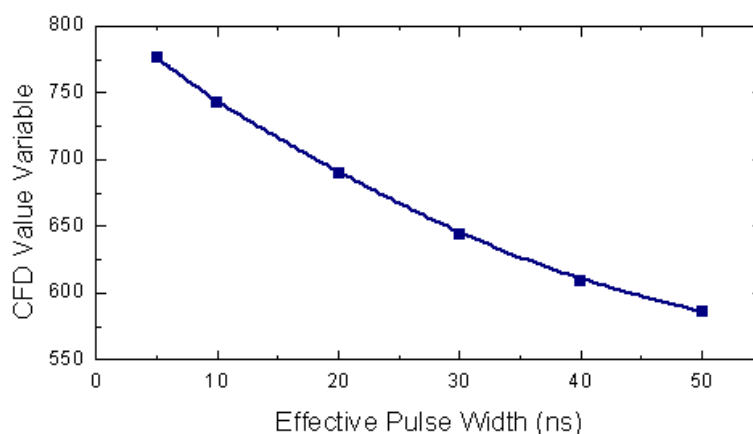


Figure 13: Rough relation between the effective pulse width and the set values of the parameters WIDTH_CN.

The parameters can be changed through the command line or through the GUI deltat (see Sections [Modifying parameters through the command line](#) and [Modifying parameters with the GUI deltat](#)).

4.2 Setting up the required logic

This section describes how to setup the logic required for a particular experiment:

- [Introduction to the Logic "Philosophy"](#)
- [The Setup Logic File \(Syntax and Examples\)](#)
- [Setting Up a Template Logic File](#)

4.2.1 Introduction to the Logic "Philosophy"

The TDC Electronics is characterized by the fact that, in addition to the pile-up gate conditions, all the necessary coincidence or veto conditions (i.e. logic conditions) are fully performed by software. If on one side such principle can appear as a "black box" without possibility to hardware checks, on the other side it permits a large flexibility and a very much reduced hardware.

Based on a first version of the front-end software, written by Thomas Prokscha, to control the CAEN TDC, Andrea Raselli has written a new version characterized by the capability to change all the logic conditions by a so-called setup logic file. Hence, between runs, it is possible to switch the logic conditions just by loading another setup logic file which contains all the necessary conditions and parameters.

Such setup logic files are usually maintained by the Instrument Scientist but can also be modified by the users. To prevent any loss of information, a so-called template file containing the "master copy" of the setup logic file can be created. It can then be loaded at any time and cannot be overwritten by the users.

4.2.2 The Setup Logic File (Syntax and Examples)

4.2.2.1 Location

The Setup Logic Files have the extension `.v1190` and must be stored in the backend computer in the directory `/userdisk0/musr/exp/td_musr/dat/tdcv1190`.

4.2.2.2 Syntax of the Setup Logic File

The following typographic convention is adapted in this manual.

- `< . . . >` represents an mandatory argument.
- `[< . . . >]` represents an optional argument.
- `< . . . > | < . . . >` represents a choice between different arguments.

In the following, we briefly describe the syntax of the setup logic file.

- **Preamble Block:**

```
# [a-comment]
```

Any line beginning with "#" is considered as a comment. Note that "!" has the same effect in the preamble block (note that this sign has another meaning in the TDC Channels Description Block).

```
INSTRUMENT=<name-of-the-instrument>
```

With this command, the name of the instrument (e.g. GPS) can be defined. It is used...

```
DESCRIPTION=<description-of-the-logic>
```

The description of the setup logic file will for example be used in the GUI deltat. For a normal user, this name should unambiguously describe the logic used by the TDC software (see example below).

```
TYPE=<type-of-TDC>
RESOLUTION=200
```

These two entries define the type of CAEN TDC (usually TDCV1190, but could also be TDV1290) and the TDC resolution. For TDCV1190 RESOLUTION is usually 200 (here 200 means actually 195.3125 ps).

For TDCV1290 RESOLUTION is usually 25.

TYPE and RESOLUTION are used to verify if the setup file and the currently running TDC frontend are compatible (i.e. same TYPE and RESOLUTION). If they do not match, the new setup will not be loaded.

```
MDELAY=<muon-delay-value>
PDELAY=<positron-delay-value>
```

These two entries define delays to be added to muons (MDELAY) and positrons (PDELAY) counters. Usually MDELAY is set to zero. By having a reasonable PDELAY, the "negative" time region can be well defined allowing a proper determination of the background.

```
MCOINCIDENCEW=<time-window-for-M-coincidences>
PCOINCIDENCEW=<time-window-for-P-coincidences>
VCOINCIDENCEW=<time-window-for-VETO-anticoincidences>
```

These entries define the time width (in TDC channel units, i.e. 195.3125 ps/channel) given to the analysis software to find muons, positron coincidences, as well as veto anti-coincidences.

- **TDC Channels Description Block:**

The next block of the setup file contains the description of the TDC channels as well as the coincidence conditions:

- **Muon Channel:**

```
Channel_Number; "Name"; M; Time_offset; [Coinc_chan] [-Anti_chan];
```

- * The parameter "Channel_Number" is the channel number (starting from 0 and max. 31) of the TDC.
- * "Name" represents the name given to this channel (max. 31 characters, use double quotes when a space is included in the name).
- * The letter "M" indicates that we have to deal with a muon channel being used as started channel.
- * The "Time_offset" represents a time offset in TDC channel units (usually set to 2000 for the muon channel, see also below).
- * The optional parameters "Coinc_chan" and "Anti_chan" (preceded by a minus sign) represent the numbers of the channels which have to be in coincidence or in anticoincidence, respectively. An event will be valid only if these conditions are fulfilled. The channels in coincidence are of type "K" (see below) and the one in anticoincidence are of type "V" (see below).

- * **Example:**

```
0; "M up"; M; 2000; 1 -10 -11 -12 -13;
```

The TDC channel is here 0, its name is "M up", its is a muon channel with a time offset of 2000 TDC channels. An event is good if it is in coincidence with the channel number 1 and in anticoincidence with the channel numbers 10, 11, 12 and 13.

- **Positron Channel:**

```
Chan_Num; "Name"; P; t_offs; [C_ch] [-A_ch]; Name_his; t_0; first; last; [Add_to]; [Final_name];
```

- * The parameter "Chan_Numb" is the channel number (starting from 0 and max. 31) of the TDC.
- * "Name" represents the name given to this channel (max. 31 characters, use double quotes when a space is included in the name).
- * The letter "P" indicates that we have to deal with a positron channel with histogram information.
- * The "t_offs" represents a time offset in in TDC channel units. For a positron detector, this number differs from the value of 2000 set for the muon channel. The difference corresponds to the time difference between a same event recorded by this channel and by the muon channel.
For example, if this channel records an event 20 ns after that it has been recorded by the muon channel, a value of 1898 (=2000-int(20/.1953125)) has to be set for the "Time_offset" parameter of this channel.
- * The optional parameters "Coinc_chan" and "Anti_chan" (preceded by a minus sign) represent the numbers of the channels which have to be in coincidence or in anticoincidence, respectively. An event will be valid only if these conditions are fulfilled. The channels in coincidence are of type "K" (see below) and the one in anticoincidence are of type "V" (see below).
- * The parameter "Name_his" represent the name of the saved histogram.
- * The parameters "t_0", "first" and "last" represent the time-zero, first and last good bins in TDC units, respectively.

- * The optional parameter "Add_to" indicates that the present histogram will be added to histogram built from the positron channel number "Add_to" to form a final histogram. The optional parameter "Final_name" is the final name of the composed histogram. Note that the bins of the histograms are added relative to the define time-zeros and that histogram bins not present in both channels are set to 0.

- * **Example:**

```
14; "F center";      P; 2073; -10 -11 -12 -13;   F_cnr; 720; 750; 50230; 2; Forw
```

The TDC channel is here 14, its name is "F center", its is a positron channel with a time offset of 2073 TDC channels. An event is good if it is in anticoincidence with the channel numbers 10, 11, 12 and 13. The histogram name is "F_cnr". The first, time-zero and last good bins are 720, 750, and 50230, respectively. This histogram will be added to the histogram formed by the channel 2 and the final histogram will have the name "Forw".

– Coincidence or Veto (anticoincidence) Channels (for muons or positrons):

```
Channel_Number; "Name";      Type; Time_offset;
```

- * The parameter "Channel_Number" is the channel number (starting from 0 and max. 31) of the TDC.
- * "Name" represents the name given to this channel (max. 31 characters, use double quotes when a space is included in the name).
- * The parameter "Type" is either the letter "K", "A" or "V".
 "K" indicates that we have to deal with a channel which has to be in coincidence with a muon or a positron channel.
 "A" is similar to the normal coincidence and might be used if there is a single coincidence. However in this case the event time will be the averaged TDC event time between both events in coincidence.
 "V" indicates that we have to deal with a channel which has to be in anticoincidence with a muon or a positron channel.
- * The "Time_offset" represents a time offset in TDC channel units. This number differs from the value of 2000 set for the muon channel. The difference corresponds to the time difference between a same event recorded by this channel and by the muon channel. For example, if this channel records an event 20 ns after that it has been recorded by the muon channel, a value of 1898 (=2000-int(20/.1953125)) has to be set for the "Time_offset" parameter of this channel.

- * **Example:**

```
0; "M up";           M; 2000; 1 -2;
1; "M down";        K; 2005;
2; "Veto";          V; 2020;
```

The TDC channel 0 is a muon channel (see also above). We have a good event for this channel if it is in coincidence with the channel number 1 and in anticoincidence with the channel number 2.

The TDC channel 1 is therefore a coincidence channel (note the letter "K").

The TDC channel 2 is therefore a veto (anticoincidence channel (note the letter "V")).

Note also the different time offset to allow proper software coincidences.

– On/Off Channels

```
ON_Channel_Number; "ON_Name";      0; Time_offset; OFF_Channel_Number
OFF_Channel_Number; "OFF_Name";    0; Time_offset
```

- * The parameters "ON_Channel_Number" and "OFF_Channel_Number" are the corresponding channel numbers of the ON and OFF signals (starting from 0 and max. 31) of the TDC.
- * "ON_Name" and "OFF_Name" represent the names given to these channels (max. 31 characters, use double quotes when a space is included in the name).

- * The letter "O" indicates that we have to deal with either an ON or OFF channel.
- * The "Time_offset" represents a time offset in TDC channel units. As the detector signals and the ON/OFF signals are not correlated, this parameter should be kept to a value of 2000.
- * Finally, the corresponding OFF channel number must be specified as last parameter of the ON channel. **-Timing scheme:**

```
On channel.....^.....
Off channel.....^.....
On interval.....:<-SwTi[On]->|start.. ..end|.....
Off interval...end|.....:<-SwTi[Off]->|start..
```
- * The switching time interval of each On/Off channel must be specified on a separate line in the TDC setup file.

```
SWTI [<Ch>]=Number_TDC_Channels
```

• Clock Channel:

The TDC software need a periodic clock signal (usually 10 kHz) to find roll-over events, which has to be listed of the list of channels in the setup file.

```
Channel_Number; "Name"; C;
```

- The parameter "Channel_Number" is the channel number (starting from 0 and max. 31) of the TDC.
- "Name" represents the name given to this channel (max. 31 characters, use double quotes when a space is included in the name).
- The letter "C" indicates that we have to deal with a clock channel.
- **Example:**

```
15; Clock; C;
```

The TDC channel 15 is a clock channel with the name "Clock".

Channel of Rejected Events:

The TDC software needs a pseudo channel where it transfers the rejected TDC events (for example events which are not fulfilling required coincidence logics).

Note that this channel cannot be used physically to feed the TDC (i.e. it is disabled).

```
Channel_Number; "Name"; R;
```

- The parameter "Channel_Number" is the channel number (starting from 0 and max. 31) of the TDC.
Note that this channel number must correspond to a physically empty or disabled TDC channel
- "Name" represents the name given to this channel (max. 31 characters, use double quotes when a space is included in the name).
- The letter "R" indicates that we have to deal with a channel of rejected events.
- **Example:**

```
16; Rejected; R;
```

The TDC channel 16 is allocated to the rejected events with the name "Rejected".

Disabled and Optional Channels:

A "!" character at the beginning of the line or an event of type "N" indicate that the channel is not used.

For Positron channels of type "P", a "!" may be used to disable optionally selectable channel.

A "!" character before "Name_his" disables the histogramming of this channel.

A "!" character before "Add_to" disables the addition of this channel to the "Add_to" channel.

Examples:

```
!17; Veto_1;          V;  
18; Chan_18;        N;
```

The first line indicates that the channel #17 (which is a veto -- note the type "V") with name "Veto_1" is disabled.

The second line indicates that the channel #18 (whose type is undefined) with name "Chan_18" is disabled.

• End of Section:

\$

This character signals the end of the section.

Examples:

The first example is a logic file for GPS with the "without Veto" configuration.

Note, for example, that the "F left" must be in coincidence with the "F right" but in anticoincidence with all the Backward vetos (channel 10, 11, 12 and 13). The resulting histogram is "F_out". The Channel 14 ("F center"), which is also in anticoincidence with all the Backward vetos builds the histogram "F_cntr" which is then added to the "F_out" histogram to finally form the histogram "Forw".

```
# TDC V1190 Set up file generated by Deltat : 16-Apr-2008 10:30:27
# bins are always in TDC channel units (195.3125 ps/channel)

INSTRUMENT=GPS

DESCRIPTION=No Veto -- lport
TYPE=TDCV1190
RESOLUTION=200

MDELAY=0
PDELAY=800

MCOINCIDENCEW=50
PCOINCIDENCEW=50
VCOINCIDENCEW=100

0; "M up";           M; 2000; 1 -10 -11 -12 -13;
1; "M down";        K; 2005;
2; "F left";         P; 1306; 3 -10 -11 -12 -13; F_out; 1485; 1515; 50995;
3; "F right";        K; 1263;
4; "B left";         P; 1311; 5 -10 -11 -12 -13; Back; 1485; 1515; 50995;
5; "B right";        K; 1262;
6; Up;               P; 1306; -10 -11 -12 -13; Up; 1490; 1520; 51000;
7; Down;             P; 1264; -10 -11 -12 -13; Down; 1525; 1555; 51035;
8; "R int";          P; 1309; -10 -11 -12 -13; R_int; 1480; 1510; 50990;
9; "R ext";          P; 1267; -10 -11 -12 -13; R_ext; 1520; 1550; 51020; 8; Rite
10; "Bveto up";      V; 2085;
11; "Bveto down";    V; 2089;
12; "Bveto left";    V; 2080;
13; "Bveto right";   V; 2079;
14; "F center";      P; 2073; -10 -11 -12 -13; F_cntr; 720; 750; 50230; 2; Forw
15; Clock;           C;
16; Rejected;        R;
!17; Ch17;           N;
!18; Ch18;           N;
!19; Ch19;           N;
!20; Ch20;           N;
!21; Ch21;           N;
!22; Ch22;           N;
!23; Ch23;           N;
!24; Ch24;           N;
!25; Ch25;           N;
!26; Ch26;           N;
!27; Ch27;           N;
!28; Ch28;           N;
!29; Ch29;           N;
!30; Ch30;           N;
!31; Ch31;           N;

$
```

The second example is a logic file for GPS with an ON/OFF condition (Channels 17 and 18) and with the "without Veto" configuration.

Note the SWTI[NN] lines at the end of the file defining the switching time interval of each ON/OFF channel in TDC bins.

```
# TDC V1190 Set up file generated by Deltat : 21-Nov-2010 19:07:22
# bins are always in TDC channel units (195.3125 ps/channel)

INSTRUMENT=GPS

DESCRIPTION=No Veto -- lport -- ON-OFF
TYPE=TDCV1190
RESOLUTION=200

MDELAY=0
PDELAY=800

MCOINCIDENCEW=50
PCOINCIDENCEW=50
VCOINCIDENCEW=100

 0; "M up";           M; 2000; 1 -10 -11 -12 -13;
 1; "M down";        K; 2005;
 2; "F left";        P; 1306; 3 -10 -11 -12 -13;      F_out;      1485; 1515; 24524;
 3; "F right";       K; 1263;
 4; "B left";        P; 1311; 5 -10 -11 -12 -13;      Back;       1485; 1515; 24524;
 5; "B right";       K; 1262;
 6; Up;              P; 1306; -10 -11 -12 -13;      Up;         1490; 1520; 24524;
 7; Down;            P; 1264; -10 -11 -12 -13;      Down;      1525; 1555; 24524;
! 8; "R int";        P; 1309; -10 -11 -12 -13;      !R_int;     0; 0; 0;
! 9; "R ext";        P; 1267; -10 -11 -12 -13;      !R_ext;     0; 0; 0;
10; "Bveto up";     V; 2085;
11; "Bveto down";  V; 2089;
12; "Bveto left";  V; 2080;
13; "Bveto right"; V; 2079;
!14; "F center";    P; 2073; -10 -11 -12 -13;      !F_center;  0; 0; 0;
15; Clock;          C;
16; Rejected;       R;
17; Ch17;           O; 2000; 18;
18; Ch18;           O; 2000;
!19; Ch19;          N;
!20; Ch20;          N;
!21; Ch21;          N;
!22; Ch22;          N;
!23; Ch23;          N;
!24; Ch24;          N;
!25; Ch25;          N;
!26; Ch26;          N;
!27; Ch27;          N;
!28; Ch28;          N;
!29; Ch29;          N;
!30; Ch30;          N;
!31; Ch31;          N;

SWTI[17]=25000
SWTI[18]=25000

$
```

4.2.3 Setting Up a Template Logic File

The Instrument Scientist can set up a **template** for a logic file, which can be loaded by a user, modified, but not saved under the same name. Hence, this allows us to have "**master copies**" of the logic files, which can be loaded in case that the normal logic files have been screwed up.

These files have an extension: **v1190_template**. In the **deltat** application, they are available on the drop-down menu which appears by pressing the **Modify Settings** button on the **TDC Settings** tab.

Note that these files cannot be saved on the same name by a normal users. When using the "Instrument Scientist Flag", they can be overwritten, although they do not appear on the **List of existing TDC Set-Up Files**.