CAMAC SOFTWARE FOR TJ-I AND TJ-IU

B.Ph. van Milligen
Asociación EURATOM-CIEMAT
May 1993

CONTENTS

1. Introduction

The CAMAC software described in this report provides a user-friendly interface to control an unlimited number of CAMAC modules in synchronization with the existing STORM VME-based data-acquisition system, as well as automatic data management. Further, user subroutines for easy recovery of data are provided and programs for inspection of data file contents and plotting and analysis of data.

The CAMAC control program uses a single settings file that contains the setup parameters for all CAMAC modules. After termination of the data-acquisition process, the data from all active CAMAC channels are stored in a single CAMAC data file. These data files are stored temporarily on the local VAX 4000 ('FUEXP1') and transferred automatically each evening at 20:00 h to the central VAX ('CIEVX1'). The data are available to the user immediately after termination of the data-acquisition process for inter-shot analysis. To this purpose, information, plot and analysis software is available. Also, the user may write his own software for more detailed analysis.

In the following sections, the function of each of the programs and routines is briefly described. Section 2 describes the control program that is used to set up the CAMAC modules and control the (automatic) data-acquisition process. Section 3 provides the definition of a CAMAC data file. Section 4 describes the (automatic) file management. Section 5 describes the user facilities for inspection of data files, plotting, analyzing and recovery of data. The Appendices provide user manuals for the most important programs.

2. CAMAC control software

The CAMAC control software performs the following tasks:

1) Update the settings file.
2) Setup and arm the CAMAC modules (initialization of CAMAC acquisition).
3) After reception of the triggers and termination of the CAMAC acquisition process, read out all active CAMAC modules and store the data in a CAMAC file.

These processes are synchronized with the STORM (VME) data-acquisition system. In particular, the discharge numbers are taken from the STORM system such that CAMAC data and STORM data for a particular discharge are identified by the same number. The CAMAC_CONTROL program is called from the STORM control menu and performs tasks 1) and 2):

1) Adding or deleting modules in the settings file and modification of existing module settings (sampling rates, start times etc.). These functions are performed in a user-friendly way, such that the user's knowledge of the particularities of the distinct modules can be minimal, and entering settings that are incompatible with the characteristics of a module is almost impossible (this to prevent delays due to errors in the setup procedure as much as

possible). Also, for modules that can only be set up manually, the knob settings can be read from the modules and stored in the settings file.

2) manual execution of the Initialize, Load Setup and Arm commands. Although normally the control cycle will be executed automatically (see Appendix A) when the command Autorizar Disparo is given from the STORM menu, the manual control is useful for taking between-shots calibration data.

Task 3) is performed by an independent subprocess called CAMAC_ACQUIRE. This process is launched when authorizing a discharge or giving the Arm command manually. It performs a wait loop until all CAMAC modules have taken their data, and then reads the data and stores them on disk. In case of error, the subprocess can be aborted by the operator from the CAMAC_CONTROL main menu.

3. CAMAC data file structure

All data traces for a single discharge are stored in a single datafile. The CAMAC datafiles provide compact data storage for an unlimited number of signals, while the size (number of points) for each signal is unlimited in principle, although at present limited to 65536 samples (64 kSamples) per signal. Data file structure is as follows:

CAMAC data file: Recordlength = 256, Unformatted, Direct access

file contents: {*header*, [*signal_1*,] [*signal_2*,]...}

where

- *header* is a character*256 string starting with the character '+'. The header fills exactly one record. The header may contain, e.g., discharge date and time and/or operator comments. At present only one header is assumed, but in principle an unlimited number of headers is possible.
- *signal_i* is the complete information for a single data trace. This information can be summarized as follows:

signal_i: {*mnemonic, module_type, number_of_samples(1-3),*
sampling_frequency(1-3), time_start, sensitivity, offset, fullscale, data(1-N)}

Data types for the variables in signal_i are as follows:

character*12	mnemonic
character*4	module_type
integer*4	number_of_samples(3)
real	sampling_frequency(3)
real	time_start, sensitivity, offset
integer*4	fullscale
integer*2	data(N)

The variables {*mnemonic, ... , fullscale*} are stored in the first record of signal_i. The array *data* occupies the next INT((N-1)/256)+1 records. Minimum size of a signal is

therefore 2 records. The length of array *data* is determined as $N = \Sigma_{j=1}^{3}$ *number_of_samples(j)*. For data with variable sampling rates, up to three different rates *sampling_frequency(j)* are possible. The time array corresponding to the data is reconstructed from *time_start* (time in ms.), *sampling_frequency(j)* (frequency in Hz) and *number_of_samples(j)*, assuming that the data segments with different sampling rates are contiguous. The real data value corresponding to *data(k)* is calculated as *value(k) = offset + sensitivity*(data(k)-fullscale/2)/fullscale*. The variable *fullscale* is equal to 2^n, where n is the number of bits of the data in array *data*, and the *sensitivity* and the *offset* of the signal are in V. The variable *mnemonic* is the signal name, used for identification of the signal (and should therefore be unique in the data file), and *module_type* is the module type with which the measurements were made (see Appendix A for more details).

4. CAMAC file management

As mentioned previously, the CAMAC data for a single discharge are stored in a single file on the FUEXP1 computer. The file name is DISCO0:[STORM.TJx]CAM_TJx_nnnnnn.DAT, where TJx can be TJ1 of TJU, and nnnnnn is the discharge number.

The user has the possibility to change the file extension (from 'DAT' to, e.g., 'CAL') by means of the CAMAC rename facility RENAME_CAMAC. Thus, it is possible to store data that were taken in between shots for test or calibration purposes. Should the user not rename his between-shot data, the data will be lost when the next discharge has been executed and the data directory has been purged.

It is possible to delete signals from the CAMAC data files in order to save disk space. However, no facility is available to the general user for this purpose, the CAMAC philosophy being that it is impossible to delete discharge data. Signals can be deleted on request to the data manager (program CAMAC_DELSIG).

The CAMAC data, initially stored on the FUEXP1 computer, are transferred automatically every evening at 20:00 h to the central CIEVX1 computer. This action is performed by a detached process called CAMAC_COPY, which not only copies the CAMAC files, but also the STORM files. The detached process is launched by the program AUTOCOPY that is run automatically when logging in on the STORM account on the FUEXP1 computer. The process CAMAC_COPY carefully checks whether the copying has terminated successfully before deleting the source files, thus avoiding the possibility of losing data in the process.

5. CAMAC facilities

Some user facilities for inspection of data files, plotting and analyzing data and user subroutines are described below. More detailed information can be found in the Appendices.

5.1 CAMAC Information program

The CAMAC Information program CAMAC_INFO provides a listing of the file contents of a CAMAC data file. An example:

```
CAMAC INFORMATION FOR SHOT NR.:      3   (FILE EXTENSION: DAT  )

  N MNEMONIC     N_SAMP FREQ.(KHZ) TSTART(MS) SENS.(V) OFF. (V) VMIN(V) VMAX(V)

  1 TEST1          8192   1000.000     0.0000   0.5000   0.0000 -0.2305  0.2285

  2 TEST2          8192   1000.000     0.0000   0.5000   0.0000 -0.1300  0.1380
```

Here N_SAMP is the number of samples of the signal, SENS is the maximum possible voltage excursion, OFF is the signal offset, and VMIN and VMAX are the actual minimum and maximum signal values in the data. The shot information can be displayed on screen or saved in a data file.

5.2 CAMAC Plot program

The CAMAC plot program CPLOT is described in detail in Appendix B. The program can retrieve up to 4 signals simultaneously from a single shot. The program is capable of performing some basic data-analysis functions: filtering, taking spectra, following the dominant frequency of a signal in time, and calculating cross correlations or autocorrelations.

The plotting section of the program provides (CIEVX1 version) interactive plot manipulation by means of single-key commands and mouse operations: plotting the traces separately or within a single frame, zooming in, moving the zoom window, moving the vertical position of the traces, undoing arbitrary signal offsets, getting signal values and sending a copy of the plot to a laser printer.

5.3 CAMAC User subroutines

The CAMAC software provides three user routines for easy retrieval of CAMAC data. Thus the user is able to write his own data analysis and/or plot software. These routines are described in more detail in Appendix A.

Subroutine CAMAC_SIGNAL retrieves a signal trace from a given CAMAC data file. The signal is identified through its mnemonic name. The routine returns the number of points of the signal, a time array (times in ms) and a data array (data in V).

Subroutine CONSTRUCT_FNAME allows easy construction of a full CAMAC data file name from the device type ('TJ1' or 'TJU') and the shot number. The returned file name is CAM_TJx_nnnnn.DAT.

Subroutine EXTENSION_FNAME allows the alteration of the default file extension ('DAT') to another one (e.g. 'CAL') in order to access calibration or test data files.

A. Appendix: CAMAC Control Manual

CONTENTS

A.1 Introduction

The CAMAC_CONTROL program was developed at CIEMAT for easy handling of a large number of CAMAC modules. The basic idea is to control the setup of a large number of CAMAC modules distributed over several CAMAC crates from a single settings file that specifies all adjustable quantities (sampling frequencies, start times etc.). Single commands issued by the user will use this information to initialize and set up all installed modules and start data acquisition. After receiving the stop trigger, the data is automatically collected from all modules and stored. The latter task is carried out by a subprocess called CAMAC_ACQUIRE.

The progam CAMAC_CONTROL is a menu-driven program that performs basically two tasks:

Firstly, it facilitates the setup of the modules, taking account of the peculiarities of each module type, such that the user need not worry about these. Some modules, however, cannot be set up through software commands, but only by hardware switches on the modules themselves. For these modules, the switch settings can be read by the program.

Secondly, it controls the execution of the subprocess CAMAC_ACQUIRE that carries out the actual data-acquisition. This subprocess, when started by the ARM command (or automatically when authorizing a discharge), enters a wait loop that terminates (1) upon time-out (presently the time-out time is 1 hour), (2) when CAMAC_ACQUIRE is aborted by the operator or (3) when all modules with active channels have taken their data. In the latter case, it then automatically reads the data, stores them on disk and terminates.

A.2 Program CAMAC_CONTROL operation

The program is called from the STORM data-acquisition menu on the machine FUEXP1. Upon starting the program, the CAMAC CONTROL MASTER MENU appears. Most options are self-evident, but here follows a description of some of them:

A.2.1 CAMAC CONTROL MASTER MENU

CHANGE SETTINGS FILE: Calls other menus to modify the settings file (explained below). This option is blocked when CAMAC_ACQUIRE is running (see the section on CAMAC_ACQUIRE).

CAMAC: READ KNOB SETTINGS: Reads CAMAC hardware switch settings (only for those modules that have switches to be read). This option is blocked when CAMAC_ACQUIRE is running.

CAMAC: INITIALIZE: Initializes the CAMAC crates and resets all modules. This option is blocked when CAMAC_ACQUIRE is running.

CAMAC: LOAD SETUP: Loads the settings from the settings file into the CAMAC modules (only for those modules that allow software setup). This option is blocked when CAMAC_ACQUIRE is running.

CAMAC: ARM: Arms all CAMAC modules (ready for data acquisition). This option will only work when preceded by a call to INITIALIZE. This option is also blocked when CAMAC_ACQUIRE is running. When effectuated, the subprocess CAMAC_ACQUIRE will be launched.

CAMAC: ABORT: Terminate CAMAC_ACQUIRE without reading the CAMAC modules or saving the data.

CAMAC: LOAD OR SAVE STANDARD SETTINGS: Load standard settings from a previously saved settings file, or save the current settings in a file with arbitrary name for later recovery.

A control cycle (preparing a discharge, discharge, read acquired data) normally requires execution of these menu options one after the other in the following sequence: CHANGE SETTINGS FILE, READ KNOB SETTINGS, INITIALIZE, LOAD SETUP, ARM. If none of the software settings will be changed, the option CHANGE SETTINGS FILE may be skipped; if none of the hardware switches is moved, the option READ KNOB SETTINGS may be skipped (*but not the first time! Otherwise the settings file may not contain the correct settings*); all other options necessarily must be executed in a full control cycle.

NOTE: Although the CAMAC CONTROL MASTER MENU allows "manual" execution of the initialization, load and arm commands, these commands are executed *automatically* when authorizing a discharge (option 3, "Autorizar Disparo", in the TJ1 or TJU menus). Thus, in general, it will *not* be necessary to execute these commands using the CAMAC CONTROL MASTER MENU (except when taking, e.g., calibration data inbetween shots).

CAUTION: It is not recommended to change any hardware settings after the ARM command (or the "Autorizar Disparo" command) but before termination of CAMAC_ACQUIRE, as this may (and, in some cases, certainly will) corrupt the data. After each change in the hardware settings the command READ KNOB SETTINGS *must* be issued in order to have the data stored in the settings file correspond to reality!

A.2.2 The settings file

The settings of a particular module are divided into MAIN and BASE settings. The BASE settings are settings that (normally) only need to be set when installing the module (crate number, trigger slope etc.), whereas the MAIN settings are liable to change every few discharges (sampling frequencies, start times etc.). This division is only made for administrative purposes. The settings parameters are explained in more detail below.

A.2.3 CAMAC: CHANGE SETTINGS menu

CHANGE CHANNEL: MAIN SETTINGS: Gives an overview of all available channels. A menu will be shown that allows selection of a specific channel attribute for display (e.g., mnemonic name or sampling frequency). Selection of a particular channel will result in a display of the MAIN settings for the corresponding module, which may then be altered. An exception to this is the CHANNEL ACTIVE option, that allows direct toggling between ACTIVE and INACTIVE for the channels.

CHANGE MODULE: MAIN SETTINGS: Gives an overview of all available modules. Upon selection of a module the MAIN settings for the corresponding module will be shown, which may then be altered.

CHANGE MODULE: BASE SETTINGS: Gives an overview of all available modules. Upon selection of a module the BASE settings for the corresponding module will be shown, which may then be altered.

ADD MODULE: Creates a new module entry in the settings file.

DELETE MODULE: Deletes a module entry from the settings file.

SWAP MODULES: Swaps one module with another (this has no other objective than esthetics; it may be nice to have modules measuring similar quantities listed together. It does not affect the functioning of the program).

A.2.4 Settings Parameters: BASE settings

As explained above, the BASE settings are those settings of a module that (generally) only have to be set upon installing the module. The meaning of some parameters (e.g. trigger slope) depends on the module type. This is explained in the listing below:

MODULE NAME: An arbitrary 40-character string to help identify the module. It may be useful to choose the name such that it explains the use of the module (e.g. "Reflectometry channels 1-4").

MODULE TYPE: A 4-character string that can only be one of the supported module types (see Section 4).

CRATE NUMBER: The crate address of the physical location of the module.

STATION NUMBER: The station number of the physical location of the module.

TIME STAMP: Type 6810 only: accuracy with which the time difference between two triggers is measured [2].

MEMORY SIZE: Type 2264: Number of attached 8800A memory modules
Type 6810: Number of attached 6310 memory modules + 1
Type 8210: Number of attached 8800A memory modules
Type 8837: Memory size =(n_memory_size+1)*1kSamples
Type 3232: Number of attached 7000 memory modules + 1

NR. OF SEGMENTS: Type 6810 only: Nr. of triggers accepted (usually 1 only)

TRIGGER HOLDOFF: Type 6810: 0 = off; 1 = on
Type 6840: Nr. of samples acquired before trigger accepted

TRIGGER SLOPE: Type 6810: 0 = positive, 1 = negative (see [2] for more)
Type 6840: 0 = positive, 1 = negative

TRIGGER COUPLING: Type 6810 only: 0 = DC, 2 = AC (see [2] for more)

TRIGGER UPPER LEVEL/TRIGGER TRESHOLD:
Type 6810: 0 = minus full scale, 255 = plus full scale
Type 6840: 0 = -2V, 255 = +1.984 V

TRIGGER LOWER LEVEL: Type 6810 only: 0 = minus full scale, 255 = plus full scale

TRIGGER SOURCE: Type 6810 only: 0 = external, 3 = software (see [2] for more)

A.2.5 Settings Parameters: MAIN settings

The MAIN settings of a module are those settings that require frequent adjustments. Below, a listing of adjustable and non-adjustable quantities is given:

NUMBER OF CHANNELS: Only adjustable for modules that support more than one input channel.

START TIME: Start time in ms. All times are relative to the stop trigger. Some modules only allow adjustment of the start time in discrete steps; when entering a time, the program will then automatically select the closest possible value permitted by the module.

STOP TIME: Calculated from the start time, sampling frequency and number of samples and therefore not adjustable.

SAMPLING FREQUENCY: Upon selection, a list of possible frequencies will be displayed from which a choice can be made.

NUMBER OF SAMPLES: Self-explanatory; the maximum number of samples (per channel) is determined by the memory size (see BASE settings) and/or the number of channels.

CHANNEL: Local numbering of input channels within the module (does not relate to the "channel identifier" that appears in the CHANGE CHANNEL menu).

ACTIVE: Channel attribute that can be "Y" or "N" if the channel is active or non-active, respectively. If a channel is non-active, data acquisition will be carried out but the signal from the channel will not be read and stored. If a channel is active, these actions will be performed.

MNEMONIC: 12-character signal name, can freely be chosen but should make sense. This name will appear in the data file and on the plots of the corresponding signal.

IDENT: Channel identifier; integer number, can freely be chosen (as long as there are no duplicates). This number allows identification and modification of a channel through the CHANGE CHANNEL menu.

OFFSET: Offset of a channel (in Volts) (see Note 1 below).

SENSIT: Sensitivity of a channel. By definition, this is the maximum peak-peak voltage excursion that can be sampled without truncation by the channel (see Note 1 below).

INPUT: Type 6810 only: Upon selection, a list of possible input couplings is displayed from which a selection can be made.

TIME BASE MODE: Type 6810 only: A value of time_base_mode larger than zero will allow operation of the module in dual time base mode. The acquired signal will be divided into three sections according to the value of time_base_mode as follows:

time_base_mode	*Pre-trigger data*	*Post-trigger data*	
		Near	*Far*
0	f1	f1	f1
1	f1	f2	f1
2	f1	f2	f2
3	f1	f1	f2

where f1and f2 are two *different* sampling frequencies that are to be defined by the user using the option SAMPLING FREQUENCY. The number of pre-trigger samples can be 0, 1/8, 2/8,... ,8/8 of the total number of samples taken. The total number of samples is defined as usual using the option NUMBER OF SAMPLES. The number of pretrigger samples is determined indirectly from the start time entered by the user under the option START TIME. The number of "near" post-trigger samples is also entered under the option START TIME. The number of "far" post-trigger samples is the remainder of total minus pre-trigger minus post-trigger "near" number of samples.

After modification of the settings the settings are *not* automatically saved; this is only done when (in the CAMAC CONTROL MASTER MENU) the options SAVE, END or LOAD SETTINGS are chosen.

Note 1: *Modules should be calibrated before use in order to be able to deduce the correct voltages from the voltages as produced by the CAMAC software!*

Note 2: Only *saved* settings are communicated to the subprocess CAMAC_ACQUIRE that carries out the data-acquisition!

The settings are checked for internal consistency (e.g. to make sure that you haven't specified more samples to be taken than there is memory available to a module) before the settings file is saved, and also when you select LOAD SETUP in the main menu. If an inconsistency is detected, the command is aborted and a message will appear.

A.2.6 Program CAMAC_ACQUIRE

CAMAC_ACQUIRE is a subprocess that is launched upon giving the ARM command, or automatically when a discharge is authorized. When it's running, the "busy" LEDs on the CAMAC crate controllers should flash on regularly (every 5 seconds).

CAMAC_ACQUIRE checks the presence of LAM's at regular time intervals. If the number of LAM's set corresponds to the number of modules that have active channels, the program reads all active channels and saves the data in a datafile. The name of the data file is CAM_TJx_nnnnnn.DAT, where TJx is TJ1 or TJU where applicable and nnnnnn is the shot number.

When doubt about the performance of CAMAC_ACQUIRE is raised, one may consult the log file CAMAC_ACQUIRE.LOG that reports on all actions of the program.

A.3 Directories, input and output; CAMAC_COPY

The program executables (CAMAC_CONTROL, CAMAC_ACQUIRE and AUTO_CAMAC_LOAD) reside on directory DISCO1:[BOUDI.CAMAC] on the machine FUEXP1.

The program can only be run successfully on the machine FUEXP1, as the CAMAC software is installed on that machine.

The directories on which the programs reside and to which the data are written are defined in command file DISCO0:[STORM]MENU.COM: e.g. at present the data directory (DIR_TJx_B) is set to DISCO0:[STORM.TJx], where TJx can be TJ1 or TJU.

The directory [STORM] contains the CAMAC settings file SETTINGS.DAT, input to and output of the program. Standard settings files can be saved (with a different name), using the facility in the CAMAC CONTROL MAIN MENU described above, and when required copied to SETTINGS.DAT for use.

CAMAC_COPY is a detached process, launched by STORM when entering the STORM menu, that automatically copies the CAMAC and DAS files to the STORM directory on the machine CIEVX1 every evening at 20:00 h. In order to have access to *all* the CAMAC files for TJ1 regardless on which machine you are working, define

a) in your login.com file on machine FUEXP1:

```
$ define    DIR_TJ1_B  DISCO0:[STORM.TJ1], -
            CIEVX1::DUA9:[FUSION.STORM.TJ1]
```

b) in your login.com file on machine CIEVX1:

```
$ define    DIR_TJ1_B  DUA9:[FUSION.STORM.TJ1], -
            FUEXP1::DISCO0:[STORM.TJ1]
```

A.4 Supported modules

Below follows a list of supported module types:

Module type	*Abbreviation*	*Software/Hardware setup*
LeCroy Model 2264 Waveform Digitizer	2264	H
LeCroy Model 6810 Waveform Recorder	6810	S
LeCroy Model 6840 Waveform Recorder	6840	S
LeCroy Model 8210 10-bit Transient Digitizer	8210	H
LeCroy Model 8837F Transient Recorder	8837	S
Aeon Model 3232 Transient Recorder	3232	S/H

The last column indicates whether the module setup is controllable through the program (S) or by means of front panel switches (H). Model 3232 has hardware switches *on its sides* for controlling offset and sensitivity, while the rest of the setup can be controlled through software.

A.4.1 Module characteristics

Module type	*Bits resolution*	*Maximum sensitivity*	*Maximum sampling rate*	*Maximum channels*	*Input impedance*
2264	8	0.512 V	4 MHz	8	50 Ω
6810	12	102.4 V	5 MHz	4	1 MΩ, 40 pF
6840	8	0.61 V	40 MHz	2	?
8210	10	10 V	1 MHz	4	1 MΩ
8837F	8	0.512 V	32 MHz	1	50 Ω
3232	12	20.48 V	250 kHz	32	?

A.5 Common errors

Most errors are recognized by the program. They will result in an error message in the lower right-hand corner of the screen. The message will be accompanied by one or more beeps when the error is considered serious. No sound will be made when the message reports successful termination of a task.

CAMAC errors, when not recognized by CAMAC_CONTROL, will result in a message of the format %CAMxxx-y-ERRzzz, followed by a message text. If the error is fatal (y=F), the program will be aborted. Normally, no such errors should occur. Most errors will occur upon installing the modules. Details like trigger tresholds, memory size etc. should be set carefully to guarantee correct operation of the modules.

A.6 Reading the CAMAC data files

CAMAC data files can be read using the subroutine
CAMAC_SIGNAL(file_name, mnemo_name, dim, ndata, time, data,ierror).
The arguments are:

file_name:	filename of the CAMAC datafile (character string, input)
mnemo_name:	character string identifying the channel to be read (signal name) (character*12, input / output)
dim:	dimension of data and time arrays (integer, input)
ndata:	number of data-points read (integer, output)
time:	time vector array (real, output)
data:	data array (real, output)
ierror:	error indicator (integer, output)

The user should make sure that the time and data arrays are sufficiently large. For information on the meaning of ierror and on linking, refer to the Appendix. Wild cards ('*') are permitted in mnemo_name. E.g., calling the subroutine with mnemo_name = '*1' will result in reading the first signal with a name ending in '1'; and mnemo_name will be changed to the signal's full name. As an additional facility, when the string mnemo_name commences with the character '%', all occurrences in the data file of a signal matching the following characters of mnemo_name will be listed, but no data will be returned. E.g., '%R*' lists all signals beginning with 'R', and '%*' lists all signals in the file.

The argument file_name can be constructed using subroutine
CONSTRUCT_FNAME(device, n_discharge, file_name),
where the arguments are:

device	TJx device (TJ1 or TJU) (character*3, input)
n_discharge	discharge number (integer, input)
file_name	file name (character, output)

The constructed file name is: DIR_TJx_B:CAM_TJx_nnnnnn.DAT, with TJx replaced by TJ1 or TJU according to argument "device" and nnnnnn replaced by n_discharge. This means the directories DIR_TJ1_B and DIR_TJU_B should be defined in your login.com (see section 3).

When using the rename facility (section 8) to rename the .DAT files to, e.g., .CAL, you must use the following routine to read the CAMAC files:
EXTENSION_FNAME(file_name, file_extension), where

file_name	file name (character, input/output)
file_extension	file extension (character, input)

This routine allows you to change the file extension (e.g.: file_extension='CAL').

A.7 CAMAC plot programs

There are two CAMAC plot programs. One runs on the machine FUEXP1, and a more advanced one that is fully interactive (mouse-controlled) that runs on the machine CIEVX1. I suggest you use the CIEVX1 version and run it from your own directory on CIEVX1; only if the communication link between CIEVX1 and FUEXP1 crashes, you may be forced to use the FUEXP1 version. Both programs allow for selection of time windows and provide options for elementary data analysis (spectra, cross-correlations etc.).

The fully interactive CAMAC plot program:

In your login.com file on CIEVX1, define:

```
$ cp == "run dua7:[fusion.milligen.plot]cplot"
```

Don't forget to define the data directory (Section 3 above). The program can be run using the command 'cp'.

The non-interactive CAMAC plot program:

In your login.com file on FUEXP1, define:

```
$ cp == "run disco1:[boudi.plot]cplot"
```

Don't forget to define the data directory (Section 3 above). The program can be run using the command 'cp'.

A.8 CAMAC rename facility / deleting CAMAC signals

Data files that must be kept for special purposes, e.g. calibration, can be renamed using the CAMAC rename facility. This facility enables the user to change *only* the file extension of the CAMAC file (from, e.g. 'DAT' to 'CAL'). This facility must be used with some care, as it does not allow to change the file name back to 'DAT', in order to avoid file duplication.

To use this facility, define

a) on FUEXP1:

```
$ ren_cam == "@ disco1:[boudi.camac]rename_camac"
```

b) on CIEVX1:

```
$ ren_cam == "@ dua7:[fusion.milligen.camac]rename_camac"
```

The facility can then be invoked with the command 'ren_cam'.

It is possible to delete single CAMAC signals within a CAMAC datafile. However, no facility is available to the general user to do this, as the general philosophy of the CAMAC software is that is *impossible* to delete CAMAC data, and therefore no data can be lost ever. Upon special request, however, and in order to save disk space, CAMAC signals can be deleted by the author of this manual.

A.9 CAMAC information program

The CAMAC information facility provides a listing of the contents of CAMAC data files (discharge date, signals stored, sampling frequencies, max. and min. values of the signals etc.). To use this program, define

a) in your login.com file on CIEVX1:

$ cinfo == "run dua7:[fusion.milligen.camac]camac_info"

b) in your login.com file on FUEXP1:

$ cinfo == "run disco1:[boudi.camac]camac_info"

Don't forget to define the data directory (Section 3 above). The program can be run using the command 'cinfo'.

A.10 References

[1] LeCroy Model 2264 Waveform Digitizer Manual

[2] LeCroy Model 6810 Waveform Recorder Manual

[3] LeCroy Model 6840 Waveform Recorder Manual

[4] LeCroy Model 8210 10-bit Transient Digitizer Manual

[5] LeCroy Model 8837F Transient Recorder Manual

[6] Aeon Systems Incorporated Model 3232 Transient Recorder Manual

[7] Kinetic Systems Corporation Model 2922 CAMAC VMS Driver 6610-1B Software Release 2.3 Manual

A.A Appendix: Example program using subroutine CAMAC_SIGNAL

```
      program demo
c*************************************************************c
c     Program to demonstrate call to subroutine camac_signal
c     with which CAMAC data can be read
c
c     Program is linked with (on FUEXP1)
c              disco1:[boudi.camac]camac_lib/lib
c     and      disco1:[boudi.genlib]genlib/lib
c     or with (on CIEVX1)
c              dua7:[fusion.milligen.camac]camac_lib/lib
c     and      dua7:[fusion.milligen.genlib]genlib/lib
c
c     Author:  B.Ph. van Milligen
c              Asociacion EURATOM-CIEMAT
c     Created: October 1992
c
c*************************************************************c
      implicit none
      integer       max_data
      parameter     (max_data=65536)
      character*80  file_name
      character*3   tj_dev
      character*12  mnemo_name
      integer       ndata,i,ierror,n_discharge,iunit
      real          time(max_data),data(max_data)
c
      write(*,*) 'Enter device (TJ1 or TJU):'
      read(*,'(a3)') tj_dev
      write(*,*) 'Enter discharge number:'
      read(*,*) n_discharge
      write(*,*) 'Enter signal name'
      read(*,'(a12)') mnemo_name
      write(*,*) 'Enter output unit ',
     +              '(1=file demo.dat, 2=screen)'
      read(*,*) iunit
      call construct_fname(tj_dev,n_discharge,file_name)
c
      call camac_signal(file_name,mnemo_name,max_data,
     +                  ndata,time,data,ierror)
c
      if (ierror.ne.1) then
          if (ierror.eq.2)
     +         write(*,*) 'File not found: ',file_name
          if (ierror.eq.3)
     +         write(*,*) 'Signal not found: ',mnemo_name
          if (ierror.eq.4)
     +         write(*,*) 'Internal workspace overflow'
          if (ierror.eq.5)
     +         write(*,*) 'Read error from data file'
          if (ierror.eq.6)
     +         write(*,*) 'Error: ndata > max_data'
          stop
      endif
c
      if (iunit.eq.1) then
          open(1,file='demo.dat',status='NEW')
```

```
   else
       iunit = 6
   endif
   write(iunit,*) 'Signal: ',mnemo_name
   write(iunit,*) 'Read: ',ndata,' data points'
   write(iunit,'(1x,3a10)') 'i','time','data'
   do 10,i=1,ndata
       write(iunit,'(1x,i10,2(f10.4))') i,time(i),data(i)
10 continue
   if (iunit.eq.1) close(iunit)
   end
```

B. Appendix: CAMAC Plot Manual

CONTENTS

B.1 Introduction

There are two CAMAC plot programs. One runs on the machine FUEXP1, and a more advanced one that is fully interactive (mouse-controlled) that runs on the machine CIEVX1. I suggest you use the CIEVX1 version and run it from your own directory on CIEVX1; only if the communication link between CIEVX1 and FUEXP1 crashes, you may be forced to use the FUEXP1 version. Both programs allow for selection of time windows and provide options for elementary data analysis (spectra, cross-correlations etc.).

B.2 Running the program

The fully interactive CAMAC plot program:

In your login.com file on CIEVX1, define:

```
$ cp == "run dua7:[fusion.milligen.plot]cplot"
```

Don't forget to define the data directory (Section 3 below). The program can be run using the command 'cp'.

The non-interactive CAMAC plot program:

In your login.com file on FUEXP1, define:

```
$ cp == "run disco1:[boudi.plot]cplot"
```

Don't forget to define the data directory (Section 3 below). The program can be run using the command 'cp'.

B.3 Data directories

In order to have access to the CAMAC data files for TJ1 regardless on which machine you are working, you should define

a) in your login.com file on machine FUEXP1:

```
$ define    DIR_TJ1_B  DISCO0:[STORM.TJ1], -
            CIEVX1::DUA9:[FUSION.STORM.TJ1]
```

b) in your login.com file on machine CIEVX1:

```
$ define    DIR_TJ1_B  DUA9:[FUSION.STORM.TJ1], -
            FUEXP1::DISCO0:[STORM.TJ1]
```

B.4 CPLOT Main Panel

Upon starting the program, the following panel will appear:

```
                    INPUT PANEL FOR CAMAC PLOT PROGRAM

PLOT DESTINATION: T                          (T = TEKTR. OR MAC, V = VT240)

DEVICE TYPE (TJ1 OR TJU): TJ1      NR. OF POINTS TO SHOW (0=ALL): 0

SHOT NUMBER: 0                     FILE EXTENSION (DAT/CAL/...) : DAT

NUMBER OF SIGNALS: 1     MNEMONIC 1:

                         MNEMONIC 2:

                         MNEMONIC 3:

                         MNEMONIC 4:

ADVANCED OPTIONS (Y/N): N

END PROGRAM      (Y/N): N                          PRESS CTRL-Z TO CONTINUE
```

and the cursor will be positioned on the field "Plot Destination". One can move from one field to the next using <TAB>, <RETURN> or the arrow keys. After entering all required data, one must enter <CTRL>-Z to proceed.

The meaning of the fields is the following:

PLOT DESTINATION:	Type of terminal; if your terminal is a MacIntosh, select 'T'.
DEVICE TYPE:	'TJ1' for TJ1 and 'TJU' for TJ1-U.
NR. OF POINTS:	Number of points plotted. Select 0 to plot all points of each signal. This may take very long, however. Since screen resolution does not allow for more than, say, 300 points, you can select 300 (or a similar number) to significantly speed up the plotting process. Be aware of aliasing effects!!
SHOT NUMBER:	Shot number
FILE EXTENSION:	Normally, 'DAT'. But some data (for, e.g., calibration purposes) may be stored in files with different names (e.g. 'CAL'). To retrieve data from such files, select the appropriate extension.
NUMBER OF SIGNALS:	Number of signals to plot (maximum: 4)
MNEMONIC:	Signal names for the four or less signals. If you select a number of signals less than four, the program assumes that you enter their names in the first mnemonics fields, and the last mnemonic fields are ignored. Note: names should be entered in upper case.
ADVANCED OPTIONS:	Enter the advanced options panel (for signal analysis) after reading the signal data.
END PROGRAM:	Leave program CPLOT.

B.5 CPLOT Advanced Options Panel

The advanced options panel can be entered from the main panel. Its purpose is to facilitate some basic signal analysis options. The panel appears as follows:

```
                    ADVANCED SIGNAL PROCESSING OPTIONS

   TIME WINDOW FOR ANALYSIS:                START TIME (MS): 0
                                            STOP  TIME (MS): 0

   LOW -PASS DIGITAL FILTER    (Y/N): N     ==> CUTOFF FREQUENCY (KHZ): 0
   HIGH-PASS DIGITAL FILTER    (Y/N): N     ==> CUTOFF FREQUENCY (KHZ): 0

   FAST FOURIER TRANSFORM      (Y/N): N

   DOMINANT FREQUENCY VS. TIME (Y/N): N     ==> NUMBER OF TIME WINDOWS: 0

   CALCULATE CROSS CORRELATION (Y/N): N     ==> BETWEEN SIGNAL NR: 1
                                                AND     SIGNAL NR: 2




   RETURN TO MAIN PANEL (Y/N): N                      PRESS CTRL-Z TO CONTINUE
```

The meaning of the fields is the following:

TIME WINDOW: When both start and stop time are 0, the whole duration of the signal(s) as read from the datafile is assumed by default. When selecting a start and stop time, parts of the signal(s) not within that time window are discarded, and only the part of the signal(s) within that time window is retained. This option may also be useful to put all signals on the same time base.

LOW-PASS FILTER: When selecting this option, you must also enter a cutoff frequency. From the cutoff frequency and the sampling rate the program then calculates the correct number of points to be taken in an n-points cosine-bell smoothing routine, and smoothes the signal(s).

HIGH-PASS FILTER: When selecting this option, you must also enter a cutoff frequency. From the cutoff frequency and the sampling rate the program then calculates the correct number of points to be taken in an n-points cosine-bell smoothing routine, and smoothes the signal(s). The smoothed signal(s) is (are) then subtracted from the unsmoothed signal(s), such that the high-frequency part is retained.

FAST FOURIER TRANSF: Calculate the Fourier frequency spectrum for the signal(s). Note: the returned spectral estimates are logarithmic.

DOMINANT FREQ.: When selecting this option, you must also enter a number of time windows. The signal(s) is (are) divided into n_window sections, and a Fourier spectrum is determined for each section. For each section the position of the peak spectral intensity is determined, and the frequency with highest spectral intensity is plotted versus time. This option is useful for e.g. following MHD mode frequencies in time.

CROSS CORRELATION: When selecting this option, you must also enter two signal numbers (corresponding to the mnemonic names on the main input panel). The program then plots the cross correlation between these two signals. If the selected signal numbers are the same, the program returns the autocorrelation.

RETURN TO MAIN: Do not execute any advanced options and return to the main panel without plotting.

B.6 Interactive plot options (on CIEVX1 version only)

When plotting the signals for the first time in any run of the program CPLOT, the program is in the SEPARATE plot mode, i.e. it plots the signals separated, each with its own axes. As explained below, the signals may also be plotted in OVERLAY mode, in which case they are plotted all together with only one set of axes. This may be useful for comparing amplitudes of different signals and the timing of special events.

B.6.1 Separate plot mode

By default, the program is in SEPARATE plot mode. Next to the plot on the right appears a menu. Menu options are selected by entering single characters only. The menu appearing has the following options:

TOP: (enter 'T') Revert to full plot window. This option only makes sense after having zoomed in on a section of the signal, in order to view the full signal again.

ZOOM: (enter 'Z') Using the mouse, select two points on the screen by clicking. These two points are taken to be the extreme angles of a box, and the part of the signal within the box is enlarged to fill the whole plot area. Note: when zooming in, the signal may change appearance because the *same* number of points will be plotted (e.g. 300 if 300 was selected on the main input panel). Thus, the resolution of the graphic will improve each time when zooming in, and more detail may appear, until all data points of a given section are shown.

OVERLY:	(enter 'O') Change to Overlay Mode.
VALUE:	(enter 'V') Select a time point using the mouse by clicking. The program will return the time- and data values in a line of text just above the plot.
C LINE:	(enter 'C') Toggle between two line modes: 1) connect data points to form a continuous line, and 2) plot individual data points without connection.
PRINT:	(enter 'P') Send a copy of the plot to the printer (laser printer, default = LASER_66A, but you may select others).
END:	(enter 'E') End plot mode and return to main input panel.

B.6.2 Overlay plot mode

When entering Overlay Mode for the first time, the signals will be stacked (i.e. they are plotted with arbitrary offset such that they appear one above the other). The stacking can be altered or undone by some of the menu options, as explained below. Next to the plot on the right appears a menu. Menu options are selected by entering single characters only. The menu appearing has the following options:

TOP:	(enter 'T') Revert to full plot window. This option only makes sense after having zoomed in on a section of the signal, in order to view the full signal again.
ZOOM:	(enter 'Z') Using the mouse, select two points on the screen by clicking. These two points are taken to be the extreme angles of a box, and the part of the signal within the box is enlarged to fill the whole plot area. Note: when zooming in, the signal may change appearance because the *same* number of points will be plotted (e.g. 300 if 300 was selected on the main input panel). Thus, the resolution of the graphic will improve each time when zooming in, and more detail may appear, until all data points of a given section are shown.
SEPART:	(enter 'S') Change to Separate Mode. Signal offsets introduced by Overlay Mode are undone.
RIGHT:	(enter 'R') When zoomed in on a section of the signal, move the zoom window to the right to see the next signal section.
LEFT:	(enter 'L') When zoomed in on a section of the signal, move the zoom window to the left to see the previous signal section.
Mn MOVE:	(enter 'M') Enter a number from 1 to n_signals to select the signal that you want to move. The selected signal will appear within a box. Then, using the 'up' or 'down' arrow keys, move

	the box upward or downward. After pressing <RETURN> the signal will be plotted in its new position.
NO OFFS:	(enter 'N') Undo the arbitrary offsets that were summed to the signals when entering Overlay Mode.
VALUE:	(enter 'V') Select a time point using the mouse by clicking. The program will return the time- and data values in a line of text just above the plot.
C LINE:	(enter 'C') Toggle between two line modes: 1) connect data points to form a continuous line (and mark traces), and 2) plot individual data points without connection (and do not mark traces).
PRINT:	(enter 'P') Send a copy of the plot to the printer (laser printer, default = LASER_66A, but you may select others).
END:	(enter 'E') End plot mode and return to main input panel.

C. Appendix: CAMAC Source Files

C.1) CAMAC_CONTROL

Purpose: a) Modification of settings file SETTINGS.DAT
b) Control of CAMAC modules (Initialize, Read, Load, Arm, Abort); When giving the Arm command, CAMAC_ACQUIRE is launched.

Called from: The STORM menu on FUEXP1

Source files: DISCO1:[BOUDI.CAMAC]

Input files: DISCO0:[STORM]SETTINGS.DAT

Output files: DISCO0:[STORM]SETTINGS.DAT

Timeout: none

C.2) CAMAC_ACQUIRE

Purpose: Subprocess that repeats a wait loop until all active CAMAC modules have asserted their LAM; then read the modules and store the data in a CAMAC data file.

Called from: a) CAMAC_CONTROL when giving the Arm command
b) The STORM menu when giving the Autorizar Disparo command (by means of a call to AUTO_CAMAC_LOAD, see below)

Source files: DISCO1:[BOUDI.CAMAC]

Input files: DISCO0:[STORM]CAMAC.DEV (contains "TJ1" or "TJU")
DISCO0:[STORM]SETTINGS.DAT
DISCO0:[STORM.TJx]NUMDESC.TJx (contains discharge number)

Output files: DISCO0:[STORM]CAMAC_ACQUIRE.LOG (log file)
DISCO0:[STORM.TJx]CAM_TJx_nnnnnn.DAT (CAMAC data file; TJx is TJ1 or TJU; nnnnnn is the shot number)

Timeout: 1 hour (after Arm command or Autorizar Disparo command)

C.3) AUTO_CAMAC_LOAD

Purpose: Automatically initialize, load and arm the CAMAC modules and then launch CAMAC_ACQUIRE

Called from: The STORM menu when giving the Autorizar Disparo command

Source files: DISCO1:[BOUDI.CAMAC]

Input files: DISCO0:[STORM]SETTINGS.DAT

Output files: none

Timeout: none

C.4) CAMAC_COPY

Purpose: Detached process that copies CAMAC data files and SADE data files from FUEXP1 to CIEVX1 and deletes FUEXP1 files. Process hibernates, but is woken automatically at 20:00 hrs. every day

Called from: AUTOCOPY

Source files: DISCO0:[STORM]

Input files: DISCO0:[STORM.TJ1]CAM*.DAT
DISCO0:[STORM.TJ1]TJ*.TJ1

Output files: CIEVX1::[FUSION.STORM.TJ1]CAM*.DAT
CIEVX1::[FUSION.STORM.TJ1]TJ*.TJ1
DISCO0:[STORM]CAMAC_COPY.LOG (log file)
DISCO0:[STORM]CAMAC_COPY.DMP (error log file)

Timeout: none

C.5) AUTOCOPY

Purpose: Launch CAMAC_COPY and schedule wake-up at 20:00 hrs. every day

Called from: STORM (when logging in)

Source files: DISCO0:[STORM]

Input files: DISCO0:[STORM]CAMAC_COPY.COM

Output files: none

Timeout: none

C.6) CAMAC_INFO

Purpose: Provide signal information for all signals within a CAMAC datafile.

Called from: DCL (is run by the user)

Source files: DISCO1:[BOUDI.CAMAC]

Input files: DIR_TJ1_B:CAM_TJx_nnnnnn.DAT (CAMAC data file)

Output files: (optional) CAMAC_INFO.DAT

Timeout: none

C.7) CAMAC_DELSIG

Purpose: Program to delete signals from CAMAC data files.

Called from: DCL (is run by the user)

Source files: DISCO1:[BOUDI.CAMAC]

Input files: DIR_TJ1_B:CAM_TJx_nnnnnn.DAT (CAMAC data file)

Output files: DIR_TJ1_B:CAM_TJx_nnnnnn.DAT (CAMAC data file)

Timeout: none

C.8) RENAME_CAMAC

Purpose: Command file to rename file extension of CAMAC data files (in order to differentiate between 'real' data and, e.g., calibration data)

Called from: DCL (is run by the user)

Source files: DISCO1:[BOUDI.CAMAC]

Input files: DIR_TJ1_B:CAM_TJx_nnnnnn.DAT (CAMAC data file)

Output files: DIR_TJ1_B:CAM_TJx_nnnnnn.yyy (CAMAC data file)

Timeout: none

C.9) CPLOT

Purpose: Plot and analysis program for CAMAC data.

Called from: DCL (is run by the user)

Source files: CIEVX1::DUA7:[FUSION.MILLIGEN.PLOT]
FUEXP1::DISCO1:[BOUDI.PLOT]

Input files: DIR_TJ1_B:CAM_TJx_nnnnnn.yyy (CAMAC data file)

Output files: PLOT.TKF (graphics file)

Timeout: none