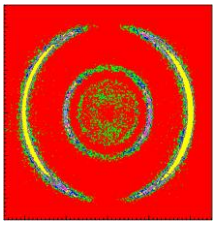
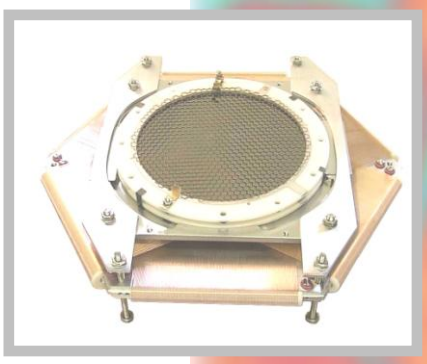
RoentDek
Handels GmbH

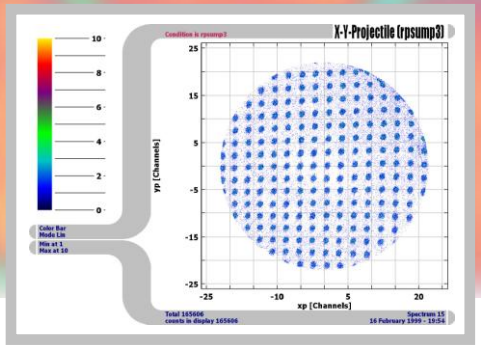
Supersonic Gas Jets
Detection Techniques
Data Acquisition Systems
Multifragment Imaging Systems



CAMAC Crate Controller (CCC1) Manual

(Version 9.3.907.1)





Mail Addresses:

Headquarter

RoentDek Handels GmbH
Im Vogelshaag 8
D-65779 Kelkheim-Ruppertshain
Germany

Frankfurt subsidiary

RoentDek Handels GmbH
c/o Institut für Kernphysik
Max-von-Laue Str. 1
D-60438 Frankfurt am Main
Germany

Web-Site:

www.roentdek.com

WEEE:

DE48573152

Product names used in this publication are for identification purposes only and may be trademarks of their respective companies.

All rights reserved. Technical changes may be made without prior notice. The figures are not binding.

We make no representations or warranties with respect to the accuracy or completeness of the contents of this publication

Table of Contents

1 INTRODUCTION 5

1.1 FEATURES 5
1.2 APPLICATIONS 5
1.3 COMPONENTS 5
1.4 OPERATING MODES 5
1.5 POWER REQUIREMENTS 5

2 HARDWARE INSTALLATION 7

3 PROGRAMMING THE CAMAC SYSTEM 9

3.1 FUNCTION CODE 9
3.2 SUBADDRESS 9
3.3 STATION NUMBER 9
3.4 STATUS INFORMATION 9
3.5 COMMON CONTROL SIGNALS 9
3.6 EVENT INFORMATION 11
3.7 LAM CONTROL 11
3.8 SELECTING LAM SOURCE 12
3.9 SELECTING INTERNAL LAM SOURCE 13

4 EVENT CONTROL 15

5 SELECTING THE RESET SOURCE 17

6 SELECTING THE IO ADDRESS OF THE IO CARD (ISA IO-CARD ONLY) 19

7 CCC1 COMMANDS AND ADDRESSES 21

7.1 INTERNAL FUNCTION CODES 21
7.2 I/O CARD CONTROL - REGISTERS 21
7.3 I/O CARD COMMAND 21
7.4 I/O CARD STATUS REGISTER INFORMATION BYTE 21

8 USING THE CCC1 IN COBOLDPC 23

8.1 THE DAQ MODULE 23
8.1.1 The CAMAC Information File (CAMAC.CIF) 23
8.1.2 Parameters used by the Standard CCC1-DAQ module 24
8.1.3 The Header Information of the ListMode File 24
8.1.3.1 The Standard CoboldPC Header 24
8.1.3.2 The CCC1 User Defined Header 25
8.2 THE DAN MODULE 25

9 FINAL NOTES: 27

LIST OF FIGURES 29

1 Introduction

You have purchased **RoentDek** CAMAC Crate Controller (**CCC1**) product.

The **RoentDek**-**CCC1** is a double-width CAMAC Crate Controller. This controller conforms to all applicable CAMAC specifications, except that it controls only station number 1-16. The controller comes with an integrated simple Event-Controller and a PC-IO card.

If you need additional information please contact service@roentdek.com. We are permanently improving our documentations.

1.1 Features

- Provides an interface between a PC and a CAMAC Crate
- Acts as main crate controller
- Controls modules in station numbers 1-16
- Data Readout Rate up to 120kB/s
- Support LAM
- Support Q and X response
- Simple Event-Controller implemented

1.2 Applications

- Laboratory automation
- Process control
- CAMAC I/O in a PC-controlled system

1.3 Components

- **CCC1** Module
- I/O board for PC (ISA)
- Connection cable (**CCC1** to I/O card)

1.4 Operating Modes

All commands and data information are passed between the IO-card and the **CCC1** in binary form, not ASCII characters.

1.5 Power requirements

+6V → up to 700mA
−6V → up to 100mA

Figure 1.1: CCC1 Front Panel

2 Hardware Installation

Locate a free ISA slot in your PC.

Make sure the power cord is removed from the computer to verify that the main board of your PC is really without any power!

Before inserting the **CCC1**-IO card please select a proper IO address (see chapter 6). Usually the default IO address (Hex 340) will function well. If the IO address is occupied by other ISA cards you have to change it. How to change the IO address is described in Chapter "Selecting the IO Address of the IO Card". After applying the IO address insert the card smoothly into the free ISA slot of your PC. Close the PC and apply the power to the computer. If the computer will not start the usual way, please select an other IO address!
Usually the card will not be detected by the "Hardware Wizard" of Windows 2000 or Windows XP and will also not be displayed as a device in the "Device Manager" but even though the card will work correct in your PC especially with the CoboldPC software.
Please be aware that it is not possible to access IO addresses directly in Operating System like Windows NT, Windows 2000 or Windows XP. In Dos, Windows 9x and WindowsMe it is possible. The CoboldPC Software handles this issue correct. If you are writing your own DAQ-System software please contact RoentDek to help you handling this IO subject on the specified operating systems.

3 Programming the CAMAC System

The following provides a brief introduction to the parameters involved in programming a CAMAC system. A more detailed description is available in the IEEE Standard 583-1975, Molecular Instrumentation and Digital Interface System.
The CAMAC standard provides guidelines for control of the 25-station CAMAC crate by a crate controller residing in station 24 and 25 (or usually the two highest stations numbers in the crate).
Communication between the crate controller and the modules is accomplished via dataway commands. Three parameters common to the dataway commands are: FUNCTIONCODE, SUBADDRESS and STATIONNUMBER. The commands can be separated into three major groups: the read commands, the write commands and the control commands. The crate controller generates the necessary timing signals to execute the commands.
Status information is conveyed from the module to the crate controller via the LookAtMe (LAM) line and the Q response and command accepted busses. Each station in the crate has a LAM line. The module may use the LAM to asynchronously inform the crate controller that it requires attention. The Q and X bus are used by the addressed module to convey status during the dataway command cycle.
There are three common control signals on the CAMAC dataway. These are INITIALIZE, CLEAR and INHIBIT.

3.1 Function Code

The Function code (F), ranging from 0 to 31 (decimal), determines which of three command groups a particular command belongs to.
The read command group includes commands with F ranging from 0 to 7 (decimal). Execution of a read command initiates the transfer of data from the addressed module to the crate controller. Data are transferred on the 24 bit read bus (R1-R24). Modules which do not use the full 24 bit word, such as a 12 bit ADC, will normally use the lower segment of the word, and the unused segment will be read as zero.
The write command group consists of those commands with F greater than 15 (decimal) and smaller than 24 (decimal). Execution of a write command initiates the transfer of data from the crate controller to the addressed module. Data is transferred on the 24 bit write bus (W1-W24).
The control command group function code includes: $7 < F < 16$ (decimal) and $23 < F < 32$ (decimal). No data are transferred on the read or write bus during control command executions, however, one bit of status information can be sent to the crate controller via the Q response line. An example is the F(8) Test LAM command. The addressed module may return the state of its LAM request to the crate controller via the Q response line. Other commands in this group include the F(24) Disable, the F(25) Initiate and the F(26) Enable command.

3.2 Subaddress

The subaddress (A), ranging from 0 to 15 (decimal) are used to access different sections of an addressed module. An example is an eight-channel TDC which is read by the F(0) A(i) command, where i can range from 0 to 7 (decimal). This provides access to all eight channels data registers with the F(0) read group 1 register function code.

3.3 Station Number

The station number (N) specifies which slot is intended to receive the command being executed. N can range from 1 to 23 (decimal). With the **RoentDek** CCC1 N can range from 1 to 16 (decimal). Internal features of crate controllers typically use N=30.

3.4 Status Information

During a dataway command execution cycle, the addressed module uses the Q response (Q) and X command accepted (X) bus lines to convey status information to the crate controller. X=1 is generated by the addressed module if it recognizes the command being executed as one that it is equipped to perform. The Q response may reflect the status of any selected feature in the module.
The LAM signal may be used by a module to inform the crate controller that it needs attention. The LAM signal reflects the status of the LAM request in the module. The LAM request reflects the state of the LAM status and whether or not LAM's are enabled in the module. LAM's are enabled within a module by an enable command or by setting a bit in a LAM mask register. LAM requests are disabled at power up.

3.5 Common Control Signals

The Initialize (Z) sets registers or control functions in a module to an initial state, particularly when power is turned on.

The Inhibit (I) disables features for duration of the signal. The Inhibit line can be set by the crate controller or a module. An example of this is using a Timing Pulse Generator to provide a time base for a programmed time, disabling the counters.
The Clear (C) clears the registers or resets flip-flops.
The use of the Z, C and I is dependent on the individual module.

CCC1 Front Panel

On the front panel of the CCC1 is divided into 3 areas of interest.

3.6 Event Information

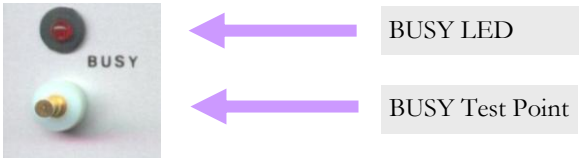


Figure 3.1: Busy LED and Test Point

- ***BUSY LED***
  is indicating the CAMAC Busy bus activity.
- ***BUSY Test Point*** (TTL)
  is for oscilloscope monitoring of the CAMAC busy bus line.

3.7 LAM Control

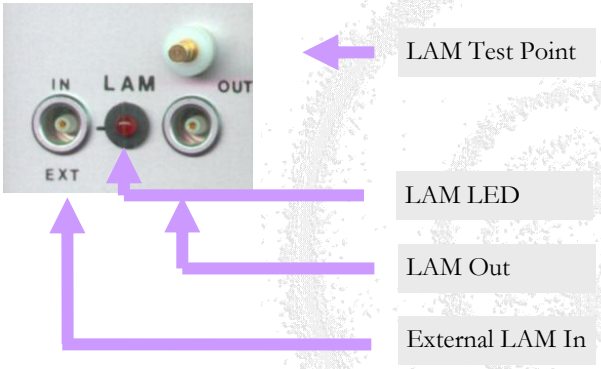


Figure 3.2: LAM Input/Output, LED and Test Point

- ***External LAM*** in (TTL)
  is equivalent an external LAM. That means this is not a pulse-signal input, it's a level-signal input. After receiving the Reset-Signal (RES-OUT) this level should be removed until the next event.
- ***LAM LED***
  will go on if an Event is detected from all sources (EventIn, External LAM in and Internal LAM). The LED will go out, when a Reset Event is sent by the computer or the LAM condition is no longer valid from CAMAC LAM.
- ***LAM Test Point*** (TTL)
  is for oscilloscope monitoring of the LAM signal
- ***LAM Out*** (TTL)
  displays the state of the LAM signal (can be used for different triggering methods)
  The state is the "ORed" ***external LAM*** **in** and the internal LAM (from CAMAC)

3.8 Selecting LAM source

To select the LAM source you have to open the side panel of the crate controller. You should see the following picture. Please note that the marked jumpers are two 3pin jumpers. Two of them are closed.

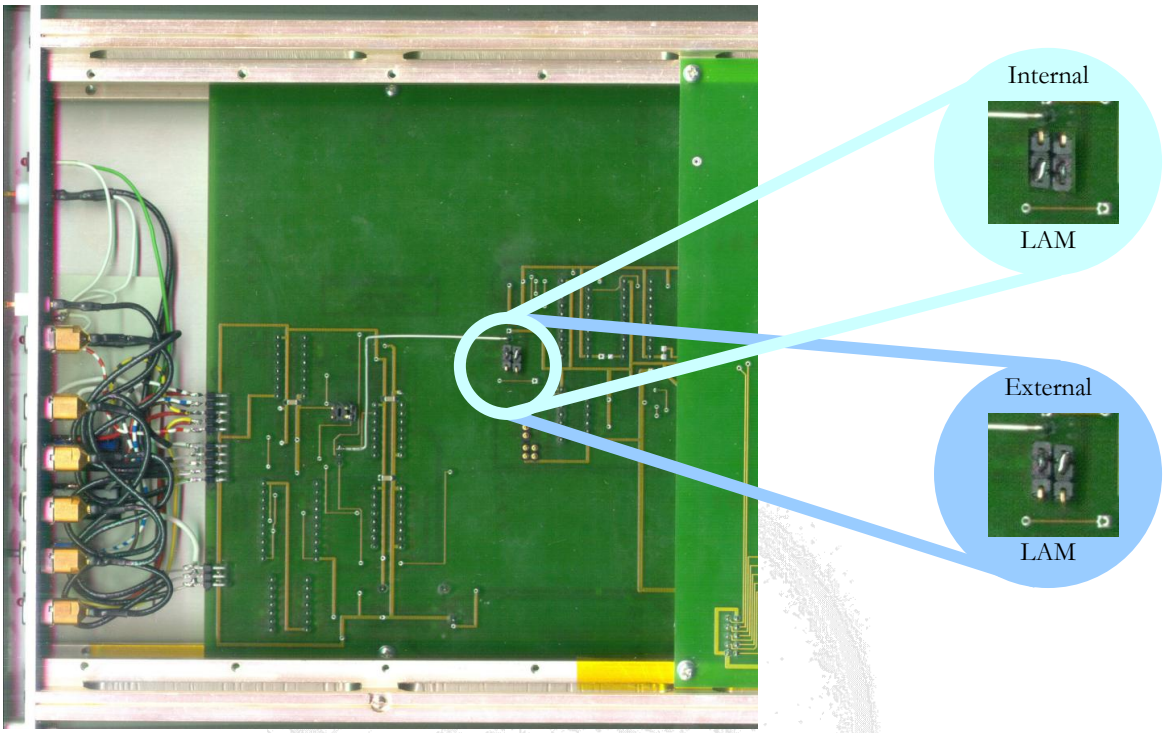


Figure 3.3: Jumper setting for selecting LAM source

3.9 Selecting internal LAM source

To select an internal LAM make sure you've selected the internal LAM as described in "Selecting LAM source". Beside this you can select up to two internal LAM sources. The LAM sources should be placed each in one of the two blocks (Q1, Q2). The source is selected by applying a switch to the needed station number. (N=1 is CSL1, N=2 is CSL2…). As mentioned earlier, the crate controller can only address the first 16 station numbers. The status bit of the CCC1 will indicate which block has initiated the LAM.

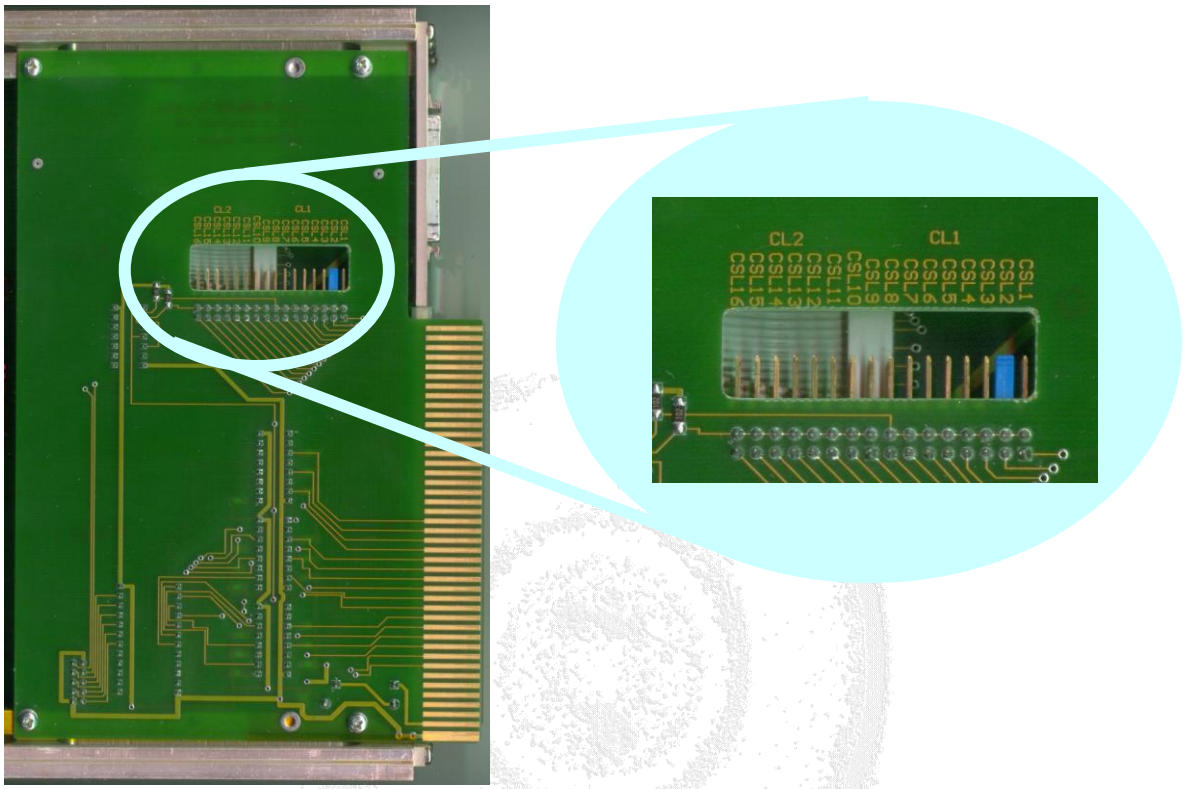


Figure 3.4: Jumper settings for selecting the internal LAM source

4 Event Control

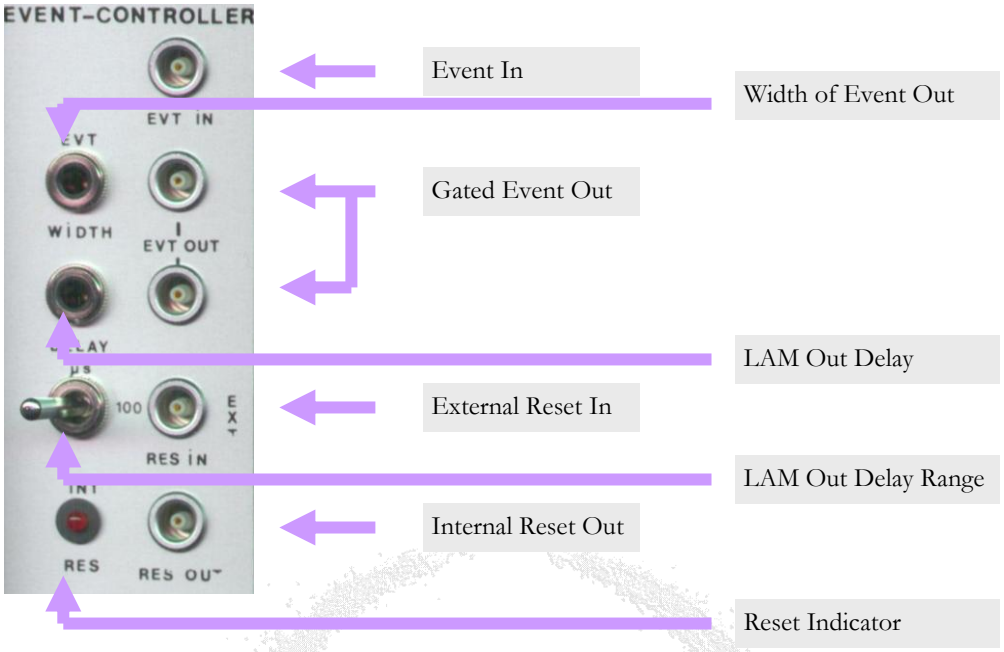


Figure 4.1: CCC1 front view for the Event-Controller

- ***Event In*** (NIM)
  signals the start of an event. After receiving an event the input will be blocked till a Reset is sent either by External Reset In or from the computer.
- ***Event Out*** (NIM)
  are regenerated ***Event In*** signals (gated by the internal event controller!). The signal is delayed about 100ns. The width of the signal can be adjusted by the ***Event Width Out*** potentiometer.
- ***Event Width Out***
  (potentiometer) controls the width of the ***Event Out*** signals.
  Range is from about 50ns up to 4.5µs.
- ***LAM OUT Delay***
  (potentiometer) controls the fine pitch of the delay between ***Event In*** and ***LAM Out***.
- ***LAM Out Delay Range***
  (switch) controls the range of the delay between ***Event In*** and ***LAM Out***
  Range: 1.0µs – 10.0µs
  10.0µs – 100.0µs
  Set this delay to the slowest conversion time of your inserted CAMAC modules to make sure that the modules are not read out before conversion has finished!
- ***External Reset In*** (TTL)
  can reset the ***Event Control*** unit if the internal switch is set to "External Reset".
- ***Internal Reset Out*** (TTL)
  State of the Reset of the Event Controller, no matter if the Reset will be initiated by ***External Reset In*** or the internal Reset (PC initiated).
- ***Reset Indicator***
  LED is on when the internal Bridge is set for internal (PC initiated) reset of the Event Controller.

5 Selecting the Reset source

To select the source of the Reset signal, you have to open one side panel of the CCC1. If the displayed switch is in “closed” position, the Reset is set to internal (means computer initiated) Reset. In this state the Reset-Indicator LED is on.
To select external reset, disconnect the switch. The Reset-Indicator LED is off.

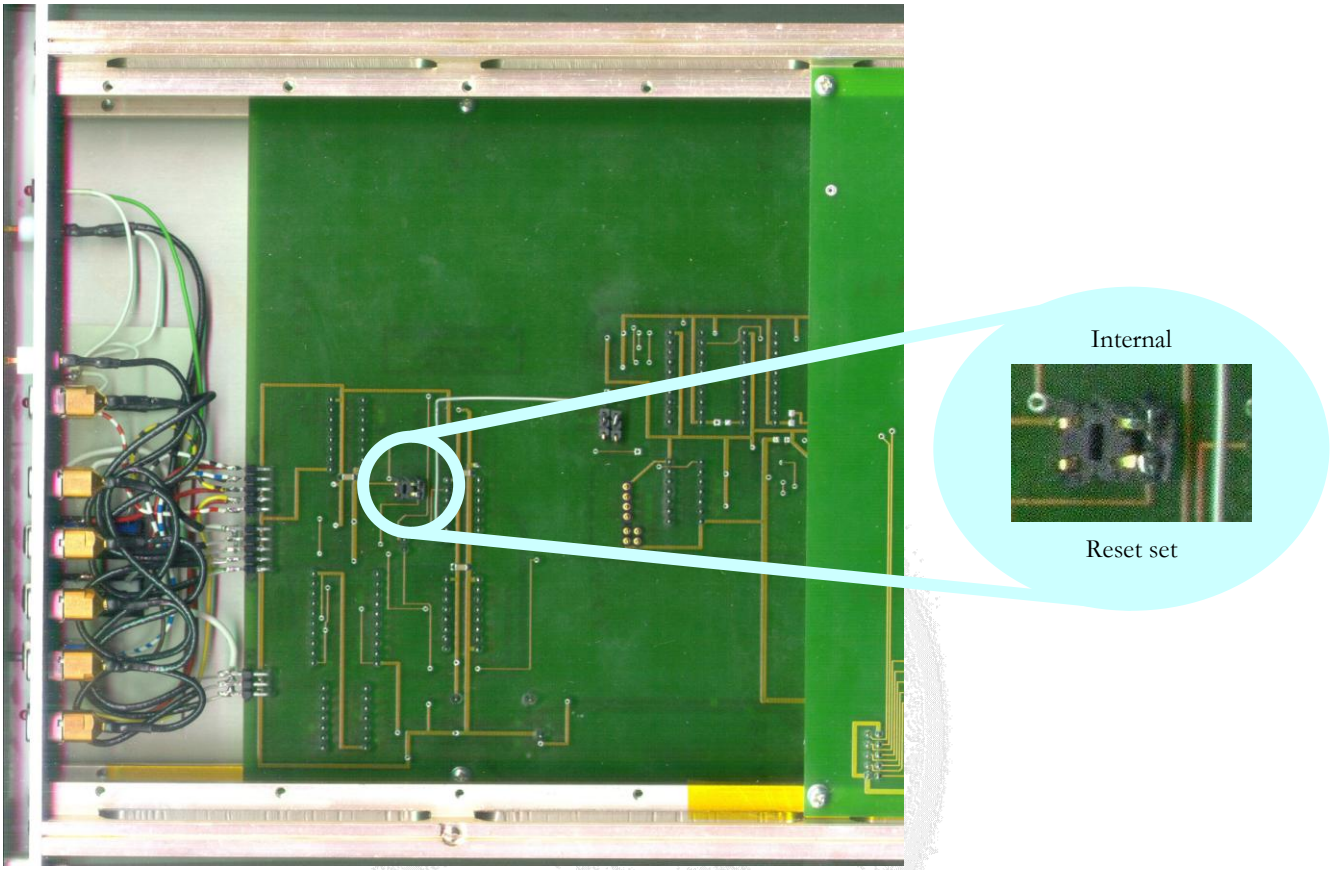
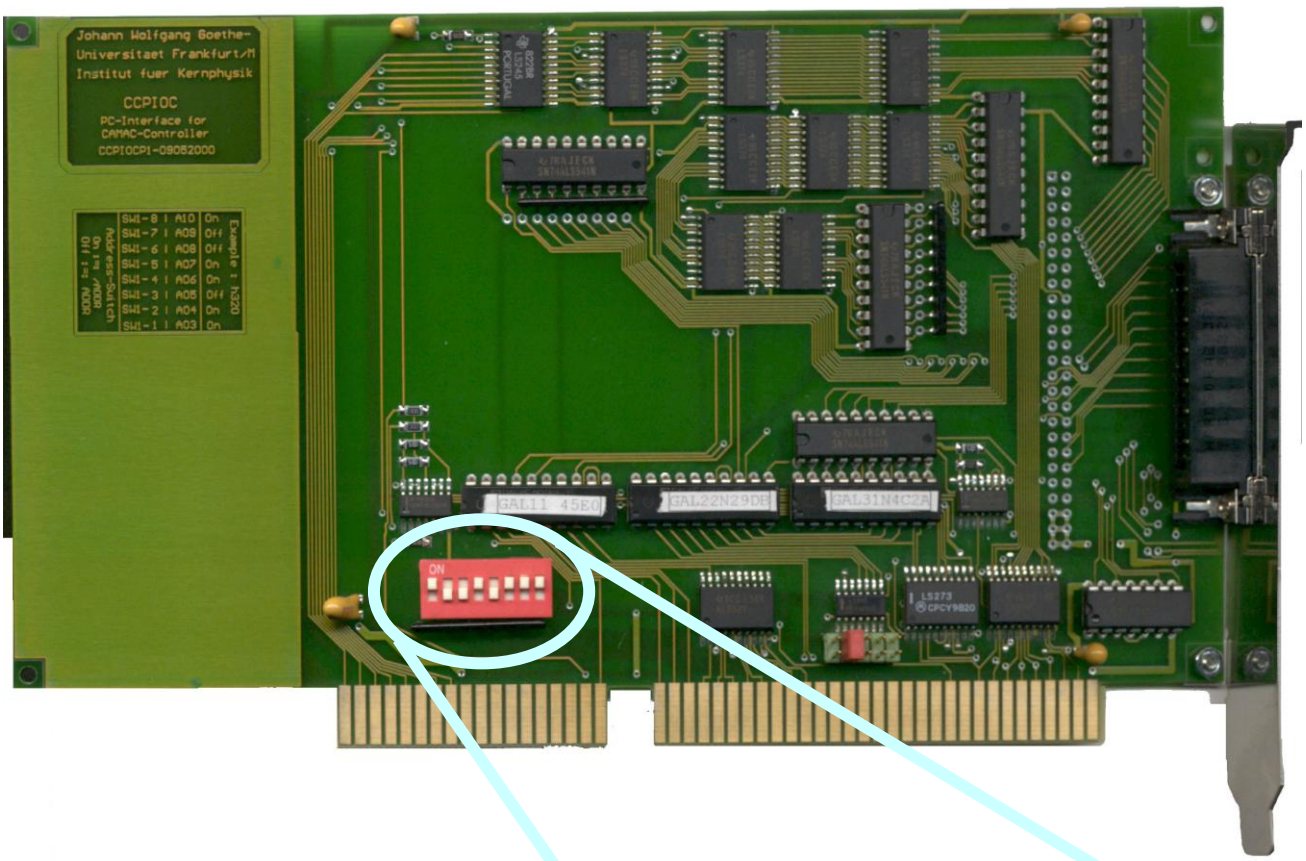
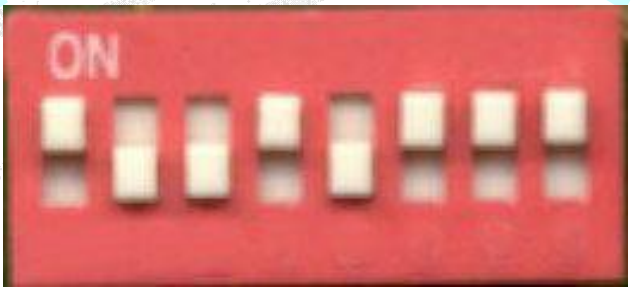


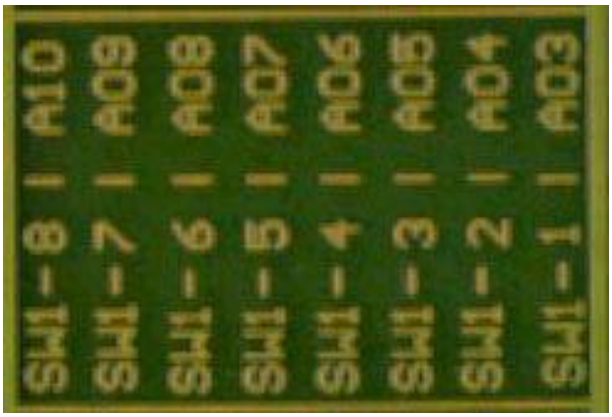
Figure 5.1: Jumper settings for selecting the Reser source

6 Selecting the IO Address of the IO Card (ISA IO-Card only)



Please ignore any description on the switch.
The right description is the one you can see here!





The Default address of the ISA-Bus IO-Card is as specified in the right picture (HEX 340)

Remember that the address lines A00-A02 can not be selected and are as default in state ON.

The example shows how to get this switch-pattern.

```
HEX          3    4     0
Binary       011  0100  0000
    or        01101000 000
```

Remove now the last 3 digits (A00-A02) and invert the pattern.

```
10010111
```

1 represents now the ON state of a switch,
0 represents OFF.

Figure 6.1: DIP switch settings for selecting the IO address on the ISA IO card

7 CCC1 Commands and Addresses

7.1 Internal Function Codes

CAMAC Command that can be sent to the CCC1

Command	Action
F=0, A=0, N=32	Send Z A CAMAC Z, initialize, will be done on the dataway following the setting of this bit. After initialization is completed the Z bit will be cleared.
F=0, A=1, N=32	Send C causes a CAMAC C, clear, to be issued. The bit is cleared immediately following the CAMAC C
F=0, A=2, N=32	Disable inhibit flag
F=0, A=3, N=32	Enable inhibit flag

Note: Station Number 32 is not an error!

7.2 I/O Card Control - Registers

BasisAddress+x	
0	High byte CAMAC write
1	Low word CAMAC write
2	Command CAMAC write
4	High byte CAMAC read
5	Low word CAMAC read
6	Status byte

7.3 I/O Card Command

Function (in hex)	
04	Reset CCC1
81	Transfer data and N,A,F to CCC1 and execute command
82	Read data from CCC1
84	Read status byte

7.4 I/O Card Status Register Information Byte

Bit Number	
0	LAM1 (external LAM) (44)
1	LAM2 (internal LAM) (43)
2	Q (42)
3	X (41)
4,5,6	Not used, typically 0
7	Executing command (if set)

8 Using the CCC1 in CoboldPC

8.1 The DAQ Module

The Standard **CCC1** DAQ Module provides almost full functionality to access CAMAC Modules in CAMAC Crate.
The **CCC1**-DAQ Module is controlled by a parameter list in **CoboldPC 2002** and by a separate ASCII file named CAMAC.CIF.

8.1.1 The CAMAC Information File (CAMAC.CIF)

Here you see a sample of the CAMAC.CIF file. It is used in this form to read out channel 1-4(6) of the Phillips TDC (Model 7187), placed in station number 6.

```
////////////////////////////////////////////////////////////////////////////
// CAMAC Information File
// ======================
//
// You must give at least 4 colon separated numbers in a row
// 1st  N       Module position
// 2nd  A       Module sub address
// 3ed  F       Function code for module
// 4th  #       positive number = number of 24BitValues to write to the module
//                          0 = no additional IO to the module
//              negative number = number of bytes to read from the module
//              for the moment only -1, 0 and 1 are valid numbers
// 5th..        24-Bit values for writing...
//
// F = 30, A = 15, N = 31 (all count start from 0) are used for wait command
//
// commands only for initialization start with capital I
//                   acquisition-loop start with capital L
//                   finalization start with capital F
// hex numbers a possible and must be entered as "0x" e.g. 0x10 = 16

////////////////////////////
// Initial CAMAC sequences
I 32, 2, 0, 0                                   // disable I (inhibit) line
//I Wait, 10                                    // wait 10us

////////////////////////////
// Loop CAMAC sequences

// Phillips-TDC:
L 06, 00, 00, -1                                // read channel 1 on pos 6
L 06, 01, 00, -1                                // read channel 2 on pos 6
L 06, 02, 00, -1                                // read channel 3 on pos 6
L 06, 03, 00, -1                                // read channel 4 on pos 6
//L 06, 04, 00, -1                              // read channel 5 on pos 6
//L 06, 05, 00, -1                              // read channel 6 on pos 6
L 06, 00, 09, 0                                 // reset TDC pos 6

////////////////////////////
// Final CAMAC sequences
F 32, 3, 0, 0                                   // enable I (inhibit) line
```

This is a standard ASCII file and can be edited by any text editor like the standard Notepad (in Win9x and WinMe the maximum size of a file that can be edited with the Notepad program is about 32kByte). The file itself is self explaining.

8.1.2 Parameters used by the Standard CCC1-DAQ module

A number of parameters have to be set to let this module function correct. How parameters are set in CoboldPC please see the CoboldPC 2002 User Manual.

```
Parameter  1 = IO Address of IO Card
Parameter  2 = Save TimeStamp
               0 = no Timestamp,
               1 = 32Bit Timestamp (Low.Low, Low.high)
               2 = 64Bit Timestamp (Low.Low, Low.high, High.Low, High.high)
Parameter  3 = System Timeout Time in s
               (after this time with no event the system will be resetted)
Parameter  5 = TimeScaling (tics per second)
               (Internally set, not to be applied by the user)
Parameter  6 = DAQ Version #
               (Internally set, not to be applied by the user)
Parameter  7 = Time Reference of DAQ Initialize (Start time of LMF)
               (Internally set, not to be applied by the user)
Parameter 40 = DataFormat
               (Internally set, not to be applied by the user)

Additionally for the "Standard DAN" (Philips TDC) we need to define

Parameter 20 = TDC resolution in ns
Parameter 21 = TDC data type information (internally set)
                  0 = Not defined
                  1 = Channel information
                  2 = Time information (in ns)
Parameter 32 = number of channels to be read out
Parameter 33 = maximum number of hits to be read out
```

8.1.3 The Header Information of the ListMode File

8.1.3.1 The Standard CoboldPC Header

The List-Mode data file consists of 2 sections. The first section is the data-header. The second is the data-area. The main header is constructed as followed.

```
unsigned int   iArchiveFlag = true;
unsigned int   LMDataFormat;
unsigned int   LMNumberOfCoordinates;
unsigned int   LMHeaderSize;
unsigned int   LMUserDefinedHeaderSize;
unsigned int   LMNumberOfEvents;
CTime          LMStartTime;
CTime          LMStopTime;
CString        LMVersionString;
CString        LMFilePathName;
CString        LMComment;
```

Unsigned int is a 32bit integer Number.

CTime and CString are classes of Microsoft Foundation Class 4.x of the Visual C++ Compiler of Microsoft.

LMDataFormat can have the following values:

```
// definitions for main program event data format
#define LM_BYTE                         1     //  8bit
#define LM_SHORT                        2     // 16bit
#define LM_LONG                         3     // 32bit
#define LM_FLOAT                        4     // 32bit
```

```
#define LM_DOUBLE                       5     // 64bit
#define LM_CAMAC                        6     // 24bit
#define LM_USERDEF                      -1    // user will handle the
                                              // reading
                                              // of file and filling of
                                              // EventData Array
                                              // except reading the main
                                              // header file
```

If the LMUserDefinedHeaderSize is not equal to zero then a user defined header will follow. The size of this data block is stored in the LMUserDefineHeaderSize variable of the main header block.
After this header block information the event data follows as described in DAQ.DLL source code.

8.1.3.2 The CCC1 User Defined Header

```
Int    *pLMUserHeaderLength   // actual header length always 1st value
int    DAQVersion             // Version is always 2nd value
int    DAQ_ID                 // DAQ_ID is always 3ed value
double Frequency              // frequency is always 4th value
int    IOAddress              // IO address (parameter 1) always 5th value
int    TimeInfo               // TimeInfo (parameter 2) always 6th value
int    DAQInfoLength          // Length of DAQInfo always 7th value
char[DAQInfoLength] DAQInfo   // DAQInfo always 8th value
int    LoadedCamacCifLength   // Length of LoadedCamacCif always 9th value
char[LoadedCamacCifLength] LoadedCamacCif
                              // LoadedCamacCif always 10th value
int    Parameter  3           // System Timeout Time in s
int    Parameter  7           // Time Reference of DAQ Initialize
                              // (Start time of LMF)
int    Parameter 40           // DataFormat (Internally set)
```

valid values for DAQ_ID are:

```
#define DAQ_ID_RAW      0x000000          // for RAW (no Data)
#define DAQ_ID_HM1      0x000001          // for HM1 (single)
#define DAQ_ID_TDC8     0x000002          // for TDC8/ISA/PCI
#define DAQ_ID_CAMAC    0x000003          // for CAMAC
#define DAQ_ID_2HM1     0x000004          // for 2 HM1
#define DAQ_ID_2TDC8    0x000005          // for 2 TDC8
```

DAQVersion is typically a date coded integer like 20020205 that means 02/05/2002 and is used to verify that you read the ListMode data with the correct DAQ-Module.

As you can see the CAMAC.CIF file is also stored in this header so you can see how your DAQ was programmed to take data.

8.2 The DAN Module

Due to the fact that CAMAC operates as a „No Standard“ in CoboldPC there are no Standard-DAN modules for CoboldPC 2002. By default the CoboldPC 2002 installation program installs a “Empty-DAN” if the CCC1 is selected in the hardware section.

But you can use the source code of the Standard-DAN module to build your own custom DAN.

9 Final Notes:

It has been observed:

If one of the two EventOut output lines are no longer working (e.g. no signal output) then the time between EventIn and EventOut can be different than expected (usually up to several hundret ns instead of only about one hundret ns).

List of Figures

FIGURE 1.1: CCC1 FRONT PANEL 5
FIGURE 3.1: BUSY LED AND TEST POINT 11
FIGURE 3.2: LAM INPUT/OUTPUT, LED AND TEST POINT 11
FIGURE 3.3: JUMPER SETTING FOR SELECTING LAM SOURCE 12
FIGURE 3.4: JUMPER SETTINGS FOR SELECTING THE INTERNAL LAM SOURCE 13
FIGURE 4.1: CCC1 FRONT VIEW FOR THE EVENT-CONTROLLER 15
FIGURE 5.1: JUMPER SETTINGS FOR SELECTING THE RESER SOURCE 17
FIGURE 6.1: DIP SWITCH SETTINGS FOR SELECTING THE IO ADDRESS ON THE ISA IO CARD 19