+Jefferson Lab Tech Notes

TITLE: Application of the PCOS 4 system at JLAB	TN#: JLAB-TN-00-031 DATE:December 11, 2000

AUTHORS: I.A. Rachek

KEYWORD(S):

☐ Accelerator Physics	☐ Diagnostics	☐ Magnets
☐ Arc	☐ Environment, QA	☐ Nuclear Physic
☐ Beam Dynamics	☐ Experimental Equipment	☐ Operations
☐ Beam Transport	☐ Extraction	☐ RF
☐ BSY	☐ Failure Mode/Tests	☐ Safety and Health Physics
☐ Civil Construction	☐ Free Electron Laser (FEL)	☐ Schedule
☐ Commissioning	☐ Front End	☐ SRF
☐ Controls	☐ Injector	☐ Test Results
☐ Cost	☐ Installation	☐ Vacuum
☐ Cryogenics	☐ Integration	☒ Other
☐ DC Power	☐ Linac	

ABSTRACT: Over 5000 wire chamber readout channels are available for detectors instrumentation at Hall-A with PCOS4 front – end cards and readout modules delivered to JLAB by Yale University. Here we report on the experience obtained while getting this system operating.

STANDARD DISTRIBUTION

Dir. Office
MS 12C
H. Grunder

Accel. Div.
MS 12A1
J. Boyce
Y. Chao
F. Dylla
R. Li
W. Schneider
B. Yunn
MS 12A2
J. Bisognano
J. Delayen
D. Douglas
A. Hutton
C. Leemann
C. Rode
C. Sinclair
AccD File
MS 16A
E. Feldl
A. Guerra
J. Kam
H. Liu
D. Machie
E. Martin
W. Oren
M. Wiseman
S. Witherspoon
MS 35
P. Hunt
B. Moss
MS 52A
M. Washington
MS 58
R. Campisi
J.-C. Denard
C. Dong
L. Doolittle
M. Drury
J. Fugitt
L. Harwood
C. Hovater
P. Kneisel
J. Mammosser
L. Merminga
J. Musson
G. Myneni
V. Nguyen
L. Phillips
C. Piller
P. Piot
T. Powers
J. Preble
C. Reece
M. Tiefenback
D.X. Wang
Test Lab Library

MS 85A
J. Benesch
M. Bickley
D. Bryan
A. Grippo
T. Hassler
A. Hofler
R. Kazimi
R. Lauzé
S. Schaffner
M. Spata
S. Suhring
J. Tang
K. White
MCC File
MS 87
R. Johnson
MCC Annex File
MS 90A
R. Nelson
MS 89
D. Arenius
S. Benson
B. Bevins
G. Biallas
C. Bohn
J. Coleman
J. Creel
R. Ganni
K. Jordan
G. Krafft
K. Mahoney
R. May
G. Neil
H. Robertson
J. Wilson
ATSB File
MS 10
B. Chronis

AD Tech Perf.
MS 12A3
R. Sundelin
MS 28H
C. Ficklen

Physics Div.
MS 12H
H. Ashley
W. Brooks
V. Burkert
L. Cardman
R. Carlini
K. de Jager
J. Grames
B. Mecking
A. Saha
Physics Div. File
MS 16B
P. Brindza
D. Kashy
Library (3 copies)
(MS ARCIA)

S. Thomas
Rev. 11/18/98

Application of the PCOS 4 system at JLab

I.A.Rachek

Budker Institute of Nuclear Physics, 630090 Novosibirsk, Russia

December 11, 2000

Abstract

Over 5000 wire chamber readout channels are available for detectors instrumentation at Hall–A with PCOS4 front-end cards and readout modules delivered to JLab by Yale University. Here we report on the experience obtained while getting this system operating.

1 Introduction

PCOS 4 stands for Proportional Chamber Operating System, the 4th generation. The system has been developed in 1992 by LeCroy Research Systems in conjunction with Yale University. The main components of the system are:

- chamber mounted card containing the circuitry to amplify, discriminate, delay, latch and encode signals from 16 wires;
- a custom backplane providing the bus and daisy chain wiring to connect up to 16 cards together to form one readout stream, and distributing power, the Gate and Fast Clear signals to the cards;
- Model 2748 controller (CAMAC or VME module) handling four readout streams, or a total of 1024 wires;
- Model 2749 driver module (CAMAC or VME) providing the interface to the experiment's trigger system, controlling the delay calibration mode and synchronizing the operation of a set of readout controllers in one VME or CAMAC crate.

There are two types of chamber mounted cards: a master card and a slave card. A master card besides the 16 wire readout channels incorporates the interface to the 2748 controller module. A connection between the master card and the controller module is done with 8 twisted pairs.

2 The documentation

The documents available for the PCOS 4 components are:

- PCOS 4 manual. Available as pcos4man.pdf file from the LeCroy web site (http://www.lecroy.com/lrs/manuals/pcos4man.pdf). The manual marked as "preliminary" and all figures are missing. However this is the only source describing in detail the operation of the 2741-16 front-end card.
- Operating manual on 2748CAM and 2749CAM CAMAC modules.
- Operating manual on 2748VME and 2749VME VME modules.
- PCOS 4 description in the LeCroy Research Systems Catalog

3 The backplane

The backplane supplies power, common signals and the daisy chain wiring to form the shift register. LeCroy does not provide the backplane, it must be custom built. Instead it is claimed that exact electrical specifications for the backplane is provided by LeCroy. However it turned out to be not correct. In the manual (pcos4man.pdf) one can find the reference to the figure showing the backplane schematic, but no figures are available. No response from LeCroy customer support could have been received either. Therefore the backplane schematic had to be developed using incomplete text description and just using common sense. The scheme of the backplane is shown in fig. 1. This is for 8 front-end cards: 1 master card, which is placed into the leftmost position, and 7 slave cards. This scheme was proved to be correct.

4 The input connector transition module

Earlier version of LeCroy wire chamber front-end card uses other type of input connector - 36-pin 0.100"–pitch in contrast to 36-pin 0.159"–pitch one used in PCOS 4. In order to be able to upgrade the existing wire chamber with new readout we have developed a transition module. It consists of a transition card with 8 pairs of connectors, one with 0.100" and one with 0.159" pitch, and a number of small cards to couple a 0.100" connector on wire chamber and that on the transition card, see Fig. 2

5 No-Q response problem

The settings of chamber-mounted cards are performed during initial hardware setup (pre-start stage in CODA). This includes the tuning of thresholds, delays, pulse widths, channel selections. The settings are done via serial connection in such a sequence:

1. set mode 2 (NA[4]F[17]. W=2);

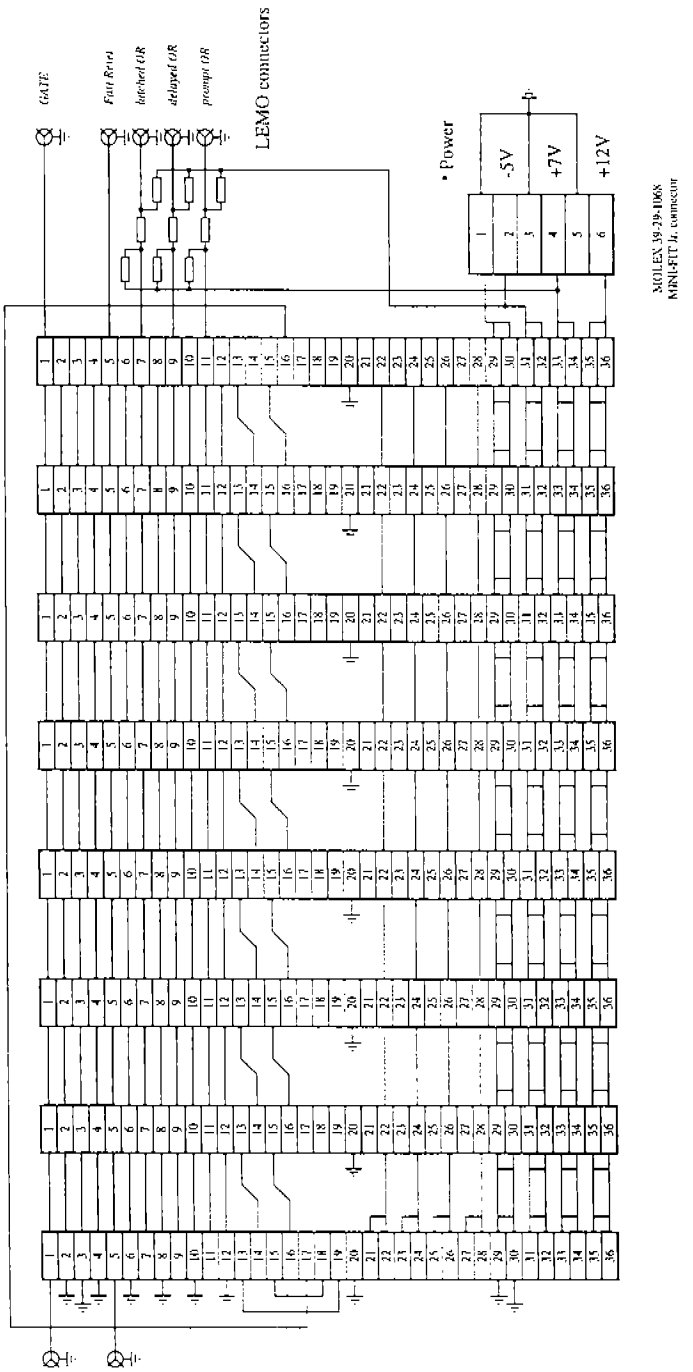

Figure 1: The schematic of the backplane for 8 cards

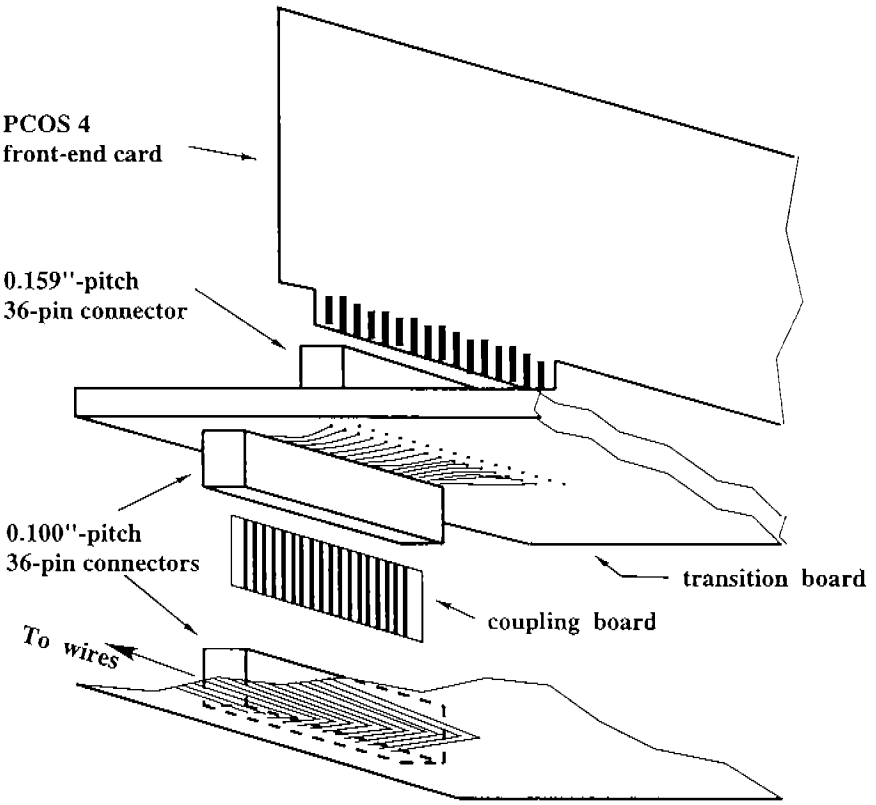

Figure 2: The transition module

2. write 16-bit data word into the shift register of the 2748CAM readout controller module, (NA[1]F[16]), data transfer over serial line will start automatically;
3. poll Q-response on NA[0]F[27] CAMAC function to detect the end of transmission;
4. repeat steps (2-3) until all settings words are transmitted to the shift registers of the front-end cards; total number of words are $16 \times N_{cards}$;
5. issue NA[1]F[25] CAMAC function to copy the shift register contents to the configuration registers inside the front-end cards[1].

And this is how the settings can be read back non-destructively to check the system integrity:

1. set mode 2 (NA[4]F[17], W=2);
2. issue NA[0]F[25] CAMAC function to transfer the chamber card configuration registers contents to their on-chip shift registers[1];
3. issue write function NA[1]F[16], W=0, to get 16-bit word to be shifted from the chamber card into the 2748CAM readout controller;
4. poll Q-response on NA[0]F[27] CAMAC function to detect the end of transmission;
5. read 16-bit word from the 2748CAM (NA[1]F[0]);
6. repeat steps (3-5) to read all configuration data words.

It was found that all the above CAMAC functions do not produce $Q = 1$ response in any state of the system. The only reasonable explanation is that the CAMAC modules are incorrectly designed. Note that the whole operation of the modules is defined by a program which is loaded into the Xilinx logic array chips during power-up. Looks like this program has "bugs".

Two consequences from this problem are:

- One can not use NA[0]F[27] function to check the end of transmission; instead one should use program delay for $\sim 16\mu s$ between each consecutive transmissions;
- When doing the read–back to verify the settings one obtains that all words are zeros. This is found to be due to the behavior of the CAMAC software library - during CAMAC read the output value is cleared if there was no Q-response. To overcome this the library has been modified. One must use this modified library, not the standard CEBAF's one.

[1]note that in the 2748CAM Operator's Manual, section 4 "Operating Instructions", incorrect function is specified for this step

6 Delay adjusting

PCOS 4 has an integrated voltage controlled 150-elements delay circuit in each readout channel, providing usable delays from 400 nsec to 800 nsec. This delay does not introduce a dead time – double hit resolution of the system is $< 80nsec$, defined mostly by amplifier pulse width. The accuracy of the delay can be as high as $1nsec$. A digital phase-locked loop is used to maintain this accuracy from chip to chip and over time and temperature. This phase-locked loop does not run continuously, it is enabled to adjust a delay if the latter is floating. This is done in the Calibration mode. To enter the Calibration mode one has either to send a NIM level at the CAL input of the 2749CAM system driver, or to set a calibration bit in the CTRL1 register of the 2749CAM (NA[1]F[17],W=1). The desirable delay is defined by a separation of pair of gate pulses during calibration mode. It is defined by a state of the CTRL0 register of the 2749CAM (NA[0][17], bits 0-5).

It is important to inhibit all other signals during the calibration mode (Trigger Gate, Fast Clear, READ, *etc*)

During the calibration mode calibration cycles (a pulse pair separated by a programmed delay), occur automatically at a rate of one every 10 μsec.

We have developed a subroutine to perform a calibration using program switching to/from the calibration mode. It basically sets the calibration mode for a specified duration and checks and reports on the results of calibration, i.e. are all channels in phase-locked state and how much DAC settings were shifted to adjust to a required delay. Trigger's GATE and READ are blocked during calibration using one of the line of the TI output register as a veto.

LeCroy recommends the following calibration sequence:

Initial startup: at least 100 msec

While warming up: at least 10 calibration per second, 1 msec each;

Stable running: a calibration every second, 1 msec each;

Actual rate and duration of calibrations are defined by the local environment and acceptable delay margin. For the test setup at TestLab a 20-msec cycle every few hours was found to be sufficient after initial 100-msec start-up adjustment.

7 Run Mode

Timing diagram for the PCOS 4 during the normal readout is shown in Fig. 3

The READ signal should be provided immediately after the end of the GATE pulse. No further gate should be issued until readout is complete. The data from front-end cards are read by Readout Controllers through the serial lines with 20 MHz clock simultaneously for all streams.. Therefore a duration of read phase is determined by the stream with largest number of cards, $T_{read} = N_{cards} \times 0.9\mu sec + 1.0\mu sec$. The

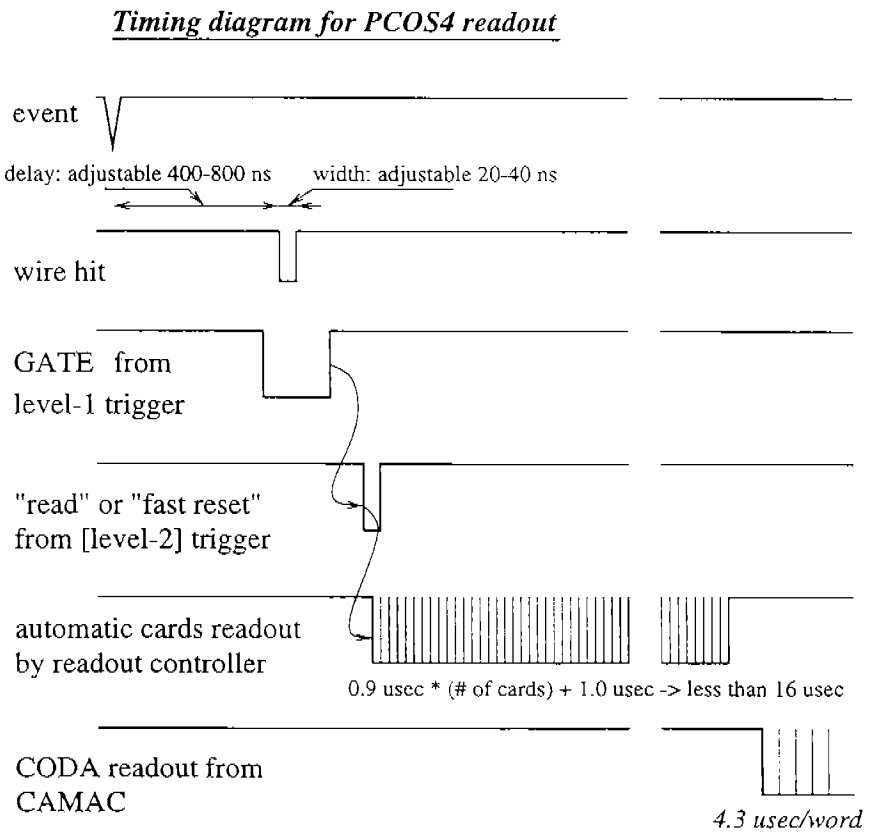

Figure 3: Timing diagram during the normal readout

longest possible read cycle is $16 \times 0.9 + 1.0 = 15.4 \mu sec$. The end of stream readout can be detected checking Q–response on NA[1]F[27] CAMAC function: Q=0 during readout and Q=1 when data is ready.

As an example a trigger scheme for the PCOS 4 readout used at the cosmic rays test setup assembled at the TestLab for Real Compton Scattering experiment (E-99-114) is shown in Fig. 4

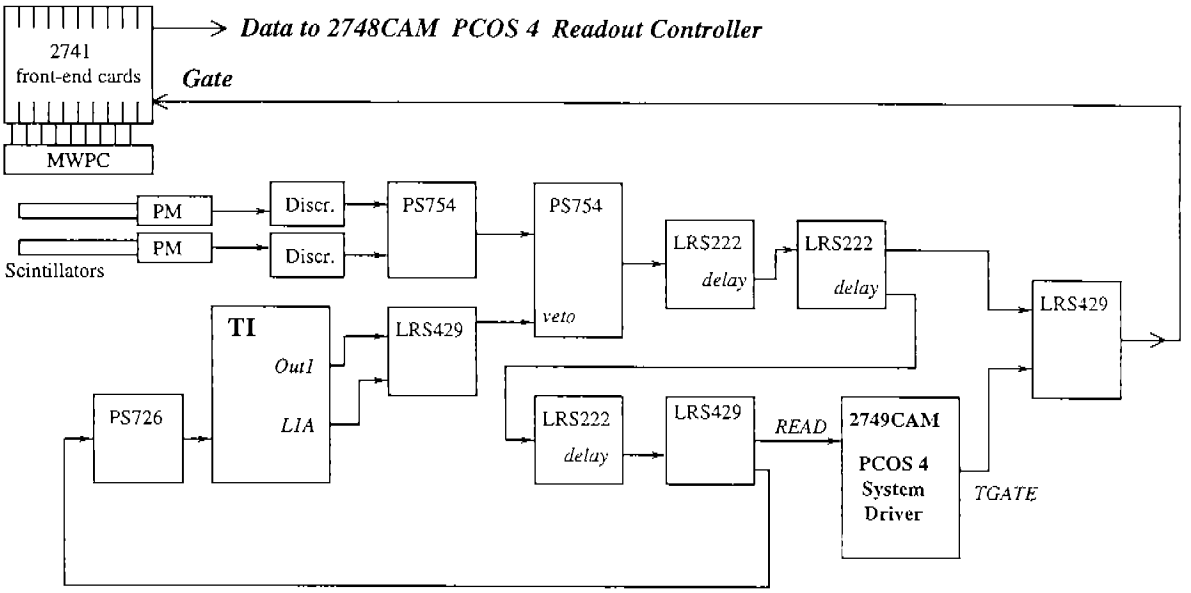

Figure 4: Example of the trigger scheme – the one used for cosmic rays calibration of MWPC for the E-99-114 experiment at Hall–A. *PS754* – Four Input Majority Logic module; *LRS222* – Gate Generator; *LRS429* – Fan–in/Fan–out; *PS726* – Level Translator; *TI* – VME Trigger Interface.

8 Data readout

Fortunately the operation of the 2748CAM controller during the readout of wire chamber data was found to be correct - exactly as described in the manual, including a proper Q-response. The correct Q-response during data readout is important to ensure a fastest readout cycle, which is achievable only if using CAMAC Q-stop readout mode.

Q–stop readout mode is implemented in the standard CEBAF's CAMAC library using DMA block transfer mode. However we could not get this subroutine working. After two CAMAC cycles the transfer interrupts (actually hangs) due to BUS ERROR. We don't know how to fix that and it was told that this subroutine has never been used in CEBAF and no-one can help us with this.

A simple decision would be to use just a loop of a CAMAC single read function and Q-response check:

1. issue single READ function NA[0]F[0]
2. get Q-status
3. if Q==1 go to step 1

However such method results in a CAMAC read speed of 15μsec per 16-bit word. Therefore we have developed a new readout mode - Program Block Data Transfer mode. Such a mode is optional for VME CAMAC interface, but it has not been implemented in the standard CAMAC library. A readout speed in this mode was measured to be 4.3μsec per 16-bit word, which is acceptable (compare to 3.5μsec per word expected for DMA operation, if it would work).

Appendix

A Software library

The following subroutines and data structures were developed to operate the PCOS 4 system.

A.1 pcos4.h

struct p4_stream – basic structure describing a single stream.

```
struct p4_stream{
   int pos,              - readout controller (2748CAM) position
   int i_stream;         - stream number [0...3]
   int na[8];            - CAMAC functions
   int n_cards;          - number of front-end cards in this stream
   int contr_data[16][2][8]; - control data to be written
                               during initialization
};
```

The array **contr_data** contains data words to be downloaded into MDL108 readout chips during the Pre-Start phase. The first index specifies a card number [0...15], the second index specifies a MDL108 on this card [0...1]. Eight 16-bit data words define the settings of each MDL108 chip:

word	default value	bits	description
0	0x200	[0-9]	Delay DAC settings
		[10-15]	Control bits for up/down counter

1	0x2C03	[0-1]	Ref. Chan. one-shot width
		[2-5]	Ref. Chan. trim delay value
		[8]	bypass
		[9]	enable prompt OR
		[10]	enable delayed OR
		[11]	enable latched OR
		[12]	select test output polarity
		[13]	PLL lock flag (read only)
2	0x110D	[0]	Chan. No. 7 (wire #1) – enable comparator output
		[1]	Chan. No. 7 (wire #1) – enable test input
		[2-5]	Chan. No. 7 (wire #1) – trim delay value
		[8]	Chan. No. 8 (wire #0) – enable comparator output
		[9]	Chan. No. 8 (wire #0) – enable test input
		[10-13]	Chan. No. 8 (wire #0) – trim delay value
3	0x110D	[0-5]	Settings for Chan. No. 5 (wire #3)
		[8-13]	Settings for Chan. No. 6 (wire #2)
4	0x110D	[0-5]	Settings for Chan. No. 3 (wire #5)
		[8-13]	Settings for Chan. No. 4 (wire #4)
5	0x110D	[0-5]	Settings for Chan. No. 1 (wire #7)
		[8-13]	Settings for Chan. No. 2 (wire #6)
6	0x3F37	[0-5]	voltage at positive input of comparators for Chans. 5–8 (wires 0–3)
		[8-13]	voltage at negative input of comparators for Chans. 5–8 (wires 0–3) *Comparator threshold is voltage difference between these two inputs*
7	0x3F37	[0-5]	voltage at positive input of comparators for Chans. 1–4 (wires 4–7)
		[8-13]	voltage at negative input of comparators for Chans. 1–4 (wires 4–7)

p4_sysdr – a structure describing a 2749CAM System Driver module.

```
struct p4_sysdr{
   int pos;          module position in CAMAC crate
   int na[8];        CAMAC functions used to operate the module
};
```

This is a list of pcos4 library entries

void **p4_sysdr_setup**(struct p4_sysdr *psdr, int crate, int pos); – sets up a control structure for System Driver Module. *crate* – crate number, *pos* – module position

void **p4_sysdr_init**(struct p4_sysdr *psdr, int delay, int pulse3); – initializes System Driver Module. *delay* – required delay: $T_{delay} = delay * 50$ nsec; *pulse3* – pulse width at output–3 of the 2749CAM module.

void **p4_stream_setup**(struct p4_stream *pstr, int crate, int pos, int i_stream, int n_cards, int *dat); – sets up a control structure for Readout Stream. *crate* – crate number; *pos* - module position; *n_cards* – number of cards in this stream; *dat* – pointer to a control data array (see above). If $dat = NULL$ then default values are used.

void **p4_stream_init**(struct p4_stream *pstr); – initializes a stream: sets up control registers of MDL108 chips in front-end cards of this stream;

int **p4_stream_check**(struct p4_stream *pstr): – checks settings of the MDL108 chips of this stream. Returns number of errors (or 0 if none)

int **p4_tune_delay**(struct p4_sysdr *psdr, struct p4_stream *pstrm, int n_streams, int ticks); – performs a computer-controlled adjustment of the delays. *psdr* – pointer to System Driver structure; *pstrm* – pointer to the list of Readout Stream structures; *n_stream* - number of streams to calibrate; *ticks* – calibration duration in units of 1/60 sec.

Returns:
0 - if everything OK
0x1ssnnnn - if in some cards PLL == 0 → something is wrong!
0x2ssnnnn - if some channels have large (> 3) deviations → more frequent calibrations needed.
ss - 8 bits containing info on which streams have a problem
nnnn - 16 bits informing which cards have problems – useful if only a single stream has a problem.

A.2 com_list.h

This is an addendum to the CEBAF CAMAC library (libcamac.o) allowing to use a Program Block Mode available in Kinetic Systems 2917 VME/CAMAC interface module, as well as perform in a more efficient way a series of single CAMAC read functions or a mixture of single/block readings. The idea is to provide a convenient access to the Command Memory of the KS2917 VME/CAMAC interface. Only 16-bit transfers are supported.

New structure:

com_list controls the filling in and access to the command list built in the KS2917 Command Memory.

The are three functions in this library:

void **init_com_list**(struct com_list *cl, int adr) performs an initial setup of a command list. **adr** – address of the command list in the KS2917 Command Memory, must be between 4096 and 8192;

void **add_to_com_list**(struct com_list *cl, int c, int n, int a, int f, int mode,int count) – adds new command to a command list.
c,n,a,f – define CAMAC function to be added to the list,
mode – CAMAC function mode. Can be selected and ORed from the following:

ks2917_qm_stop – Q-Stop mode
ks2917_qm_ignore – Q-Ignore mode
ks2917_qm_repeat – Q-Repeat mode
ks2917_qm_scan – Q-Scan mode
ks2917_tm_single – Single read function
ks2917_tm_block – Block read function
ks2917_tm_inline – Inline Write function
ks2917_ad – Abort Disable

Descriptions of these flags can be found in the KS2917 Manual.
count – number of 16-bit data words to be transfered in Block Read Operation, or a word to be written in Inline Write Operation. HALT instruction automatically placed in the end of the command list.

int **com_list_read**(struct com_list *cl, unsigned short *data, int cnt) – executes the command list specified by **cl** . **data** – pointer to input array; **cnt** – maximal number of 16-bit words to be read.