



Brazilian Journal of Physics

ISSN: 0103-9733

luizno.bjp@gmail.com

Sociedade Brasileira de Física  
Brasil

Fagundes, A.N.; Sá, W.P. de; Dantas, A.L.  
TCABR data acquisition system update  
Brazilian Journal of Physics, vol. 32, núm. 1, marzo, 2002, pp. 50-53  
Sociedade Brasileira de Física  
São Paulo, Brasil

Available in: <http://www.redalyc.org/articulo.oa?id=46413503010>

- How to cite
- Complete issue
- More information about this article
- Journal's homepage in redalyc.org

redalyc.org

Scientific Information System  
Network of Scientific Journals from Latin America, the Caribbean, Spain and Portugal  
Non-profit academic project, developed under the open access initiative

# TCABR Data Acquisition System Update

A.N. Fagundes\*, W.P. de Sá† and A.L. Dantas‡

*Institute of Physics, University of São Paulo*

*R. Matão, Trav R, 187 — CEP 05508-900, São Paulo, Brazil*

Received on 3 July, 2001

We describe recent proposed updates for TCAqs[1], the TCABR data acquisition system. Stable and in regular use, it shall be enriched in new future for better performance. Recent acquisitions widen the TCAqs capabilities in hardware and software, including new VME stations and special GPIB cards that will allow for the creation of local instrumentation network.

## I Introduction

TCAqs may be roughly divided in devices necessary for data acquisition; devices control and operation and devices for data analysis. Acquisition and control devices stay in experimental room and are based primarily upon VME[2] instrumentation standard. Personal computers with hardware action are also used. System is shown in Fig. 1. Data inspection and analysis is made in control room by the use of PC-like computers using software, either freely available or originally included in the computers when they were bought.

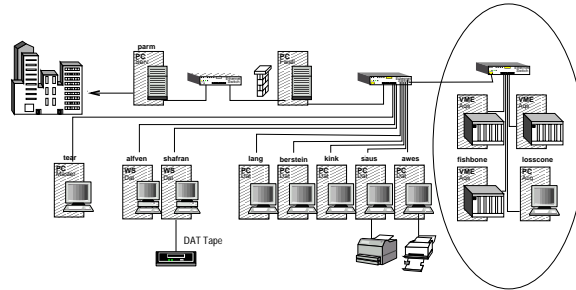


Figure 1. Computational devices for data following and support of TCAqs. Computers in oval are located in experimental room, connected by optical fiber to control room. The upper line suggests that, in its final version, TCAqs will have controlled access through a firewall. Data and public services will be available in a server before the firewall.

The development of software has been done in a slow pace, for it acts on elements connected in a local network, including the computer in experimental room that houses the trigger board that fires the entire

TCABR plasma shot. This poses safety issues, that has been solved in the safest way: The local network has only one user, the technician in charge of TCABR daily tasks, when the machine is in scientific operation.

## II Present capabilities.

TCAqs is based upon the VME[2] with one crate which has 21 slots available. Table 1 describes available modules.

TCAqs control action is limited to production of light pulses for firing the intervening processes that constitutes the tokamak action and proper recording interval. Several side actions, like sound and light warnings that precedes tokamak shots are done manually by the operator, including the necessary action of power line reservation that feeds main transformer for toroidal current. As for data acquisition, 18 ADCs modules are available, of which about 10 are in use permanent, generating around 1.3 MB of data (compressed) *per* shot, originated by around 40 sources of diagnostics. Table 2 summarizes observed diagnostics. The operational routine for programming the TCAqs involves direct writing of configuration files, that are conveyed to proper hardware housing computer.

TCABR inherited CAMAC systems, comprising crates and ADCs from TBR [5], the former tokamak at the Physics Institute. These resources are not in full use for the lack of hardware and software suport.

\*fagundes@if.usp.br

†pires@if.usp.br

‡dantas@if.usp.br

Qt	Unit	Func	STD	Chan	Mem (kb)	Res (bits)	Sp.Rate (ksp/s)	
1	Crate		vme					
2	Crate		vme					Upg
1	68060	Cont	vme					
2	68040	Cont	vme					Upg
3	Pulse gen	trig	vme	16				
1	Pulse gen	trig	vme	16				Upg
18	ADC	wfr	vme	8	512	12	1300	
9	ADC	wfr	vme	8	512	12	3000	Upg
4	Crate		cam					Upg
2	ADC	wfr	cam	2	64	10	2000	Upg
6	ADC	wfr	cam	8	4	8	4000	Upg
2	ADC	wfr	cam	4	8	10	2000	Upg
2	ADC	wfr	cam	32	1	10	5	Upg

Table 1. TCAqs hardware for data acquisition. “Upg” stands for *upgrading*, meaning that it is to be available in a short period of time.

Diagnostic	Source	Num.Points ( $\times 1000$ )	Time interval (ms)
Vertical Current	2	50	100
Plasma Current	1	50	100
Loop Voltage	1	50	100
OHT current	1	50	100
Sine coil	1	50	100
Horizontal Position	1	50	100
X-Ray	3	50	100
Soft X-Ray	2	50	100
$B_T$	1	50	100
$B_Z$	1	50	100
$H_\alpha$	12	50	100
Mirnov Coil	1	50	100
Flux Coil	1	50	100
Diamagnetic coil	1	50	100
Bolometer	1	50	100
Langmuir Probes	4	50	100
Interferometer	6	60	100
Command Voltage	1	50	100
Gas Puffing Voltage	1	50	100
ECE	1	50	100

Table 2. Normally recorded signals in TCABR in june of 2001

### III Upgrading

The TCAqs will experiment a significant expansion in the near future. Two more VME stations, with Motorola 68040 CPU based controllers are expected to arrive in few weeks, together with 9 more ADCs of the same capabilities as the present ones and a pulse gen-

erator. One of them will supply registration means for about 100 Mirnov and the other is to be installed in control room for software development, removing the present day safety risks of tampering with the crate used for firing the machine.

GPIB boards specially chosen (*cards for which there are open software support*) are to be used. These

boards will allow for the creation of a instrumentation network in experimental room, overlooked by a computer and a smooth link with TCAqs. Among other candidate devices that may be present, like oscilloscopes, voltages meters. etc. all CAMAC instrumentation resources will render available.

## IV Data compression

Daily use has determined that signals are to be sampled with intervals of  $4 \mu s$  in a extension of 50 000 points, covering 200 *ms* of time. 12-bit resolution ADCs produce numbers in the interval from 0 to 4096 ( $2^{12}$ ). Each point demands two bytes for expression. Therefore, each source of diagnostics generates 100 *kb* of data per shot. A single ADC as the ones presently used in TCAqs, may produce 0.8 *Mb* of useful data (4 *Mb* if used with full memory capacity).  $\approx 5$  *Mb* of data are to be kept as the total information acquired in the shot. Obviously this number is expected to grow as different sources of signals are aggregated into the data acquisition system.

For sake of reducing transmission errors, increase efficiency in storage and provide faster access time, data has to be compressed in order to obtain more easily manageable volumes. Unfortunately, data compression is not a well defined issue with prescribed recipes and is strongly dependent upon the kind of information at hand. The theme is studied in several books, two of which are written by K.Saywood[10] and M.Nelson and J.P.Gailly [8]. In Data Compression, two main directions are generally pursued: *lossy* methods and *lossless* methods. The former producing usually the best results in compression, at the sacrifice of information on retrieval. Lossy methods were not considered. From available methods only lossless ones were chosen for the TCAqs. That means that information restores to its entirety after decompression with no differences between the reconstructed files and the original.

The book by M.Nelson and J.P.Gailly describes several compression methods making comparisons among different methods a easy task. But in our study also other promising published methods were tried.

The methods chosen were: Huffman, Adaptive Huffman (AdHuff), Arithmetic (Arith), Statistical Modelling (ArithN), Lempel-Ziv-Welch method (lzw15M) and Dynamic Markov Chain (dmc) (as published in [9]. A test set of data was prepared, made of 27 files as obtained from TCAqs Analog-to-Digital converter devices, plus a file containing  $\pi$  expressed with 100 000 figures. Test set had size 5.34 *Mb*, comparable the present TCAqs load. To compare results we adopted the *Compression Index* defined according to [7], as

$$CI = \left(1 - \frac{\text{Compressed size}}{\text{Original Size}}\right) \times 100 \quad (1)$$

The worse compression method would produce a file compressed with size equal to the original size and a  $CI = 0\%$ . On other hand, an ideal compression scheme would have a file with compressed size of *zero* size and  $CI = 100\%$ . Results are summarized in table 3.

All methods considered reduce original files to less the half of the size and would be of significative improvement in data management. The best method applicable corresponds to statistical modelling, in which a compressed size of 1,02 *Mb* was obtained, corresponding to 81 % of data compression in 103 seconds. Time has to be considered, aside *CI*. So, the dictionary method based upon Lempel-Ziv-Welch (usually known as LZW) was chosen. A file of 1.2 *Mb* for the original 5.34 *Mb* was produced in 47 *s* with a  $CI = 77\%$ . This means that files produced by data acquisition devices had around one-fifth of their original size on the average, without no losses in contents.

Although with good perspectives, we worked upon the code for fine tuning it for our specific use. The original code by Mr.M.Nelson was done with a wide horizon in mind aiming the compression of written text, being applicable to a universe of 256 characters, for operational systems of 16 and 32 bits. In TCAqs case, we deal with different OSs but all of them based upon 32 bit and only 12 possible different bits may occur. With this boundary conditions, the code for LZW method was modified to accept only 32 bits OSs. The final product is a shorter code, running faster and shall pose no problems as long as ADCs with resolution less or equal 16 (meaning systems able to resolve 1 part in  $2^{16} = 64,536$ ) are used. The drawback is that we have a unique way of compression, instead of a known standard, making harder information exchange among research centers. An attenuant factor is that code compiles in open software code, with widely available GCC compiler and will be freely available — as also is the case with data acquired — from TCABR laboratory computers.

Method	R. File	CI	Time
	(b)	(%)	(s)
AdHuff	1885495	65	18
Arith	1914252	64	25
ArithN	1017800	81	103
lzw15M	1206563	77	16
dmc	1345600	75	121
Huffman	1948709	64	47

Table 3. Results of compression according different techniques. The same set of data was applied to every method, involving a selected collection of actual data from TCABR 29 sources of diagnostics plus  $\pi$  with 100k figures, summing up 5.34 *Mb*. "R.File" stands for *Resulting File* and "CI" stands for *Compression Index* as defined in ref.[7].

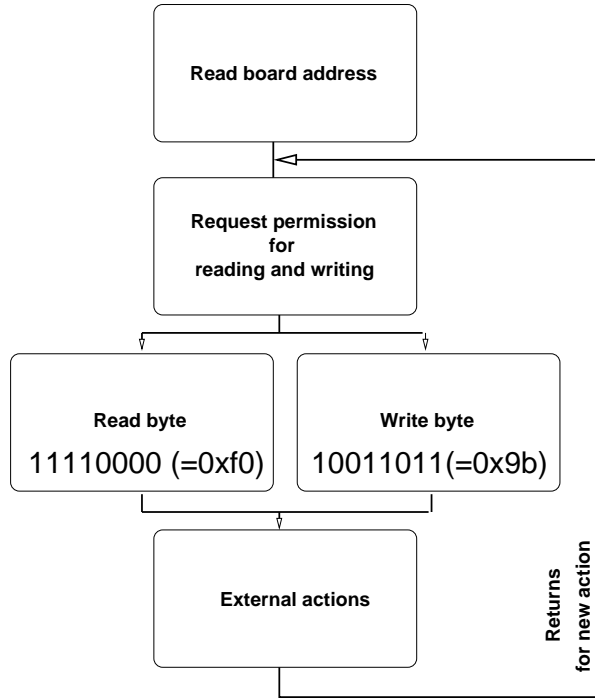


Figure 2. Board PDL-IO-16C has a simple programming cycle. We request permission from operational system for reading/writing on the board address. A byte is formed of bits, each one assuming only “1” or “0” values that will close or keep open corresponding relay in associate relay board. In the example, the first, 2<sup>nd</sup>, 4<sup>th</sup>, 5<sup>th</sup> and 8<sup>th</sup> relays would be actuated when we write 0x9b on board address. Reading action is done by *negating* byte read: the answer of 0xf0 implies that the first four relays (from right to left) are actuated.

## V Use of relay board

Development of open software controlled hardware is in progress. Initially, it is foreseen the operation of a board able to convey a word (two bytes) to the external world. This board links to another one, through optical decoupling, which has eight independent relays. The internal board is identified PLD-IO-16C; the external board, PD-INTF8, both made by the brazilian company PRODELS[6]. The internal board has two input and output independent ports, with 8 bits each. In order to access the ports, two bytes are necessary, although only one is in use. The board base address may be configured through a dip switch named SW1 in the interval from 0x200 to 0x03FF. The address is used

for both reading and writing by the use of a *control bit*. The board has a simple programming cycle, as shown in Fig. 2. All software was developed in C language, with GCC compiler, using functions *ioperm*, *inb* and *outb* under linux operational system.

This will make possible to put under automatic control some of the actions necessary to fire the TCABR, including power line reserve and warning lights, as may help in operating transient action in the side controls and safety systems, like detecting unexpected live presence in forbidden areas or unmet conditions for tokomak shot.

## VI Conclusions

Stable and dependable, having already recorded a few thousands shots, the TCAQs is about to have a significant increase of potentiality, becoming more powerful and versatile. Adding in complexity, the new hardware will increase either directly the capacity of observation and registry of physical phenomena; as indirectly, allowing for the creation of a dedicated instrumentation network around the TCABR. Software for better compression and also for hardware action under open source software has been developed and is still in progress, pursuing a higher level of automatism in a safer environment for operating TCABR.

## References

- [1] A.N.Fagundes, W.P.de S, P.M.S.A.Coelho, *Fusion Engineering and Design*, **48**, 213 (2000).
- [2] <http://www.vita.com>
- [3] <http://www.microware.com>
- [4] <http://www.tcx.se>
- [5] A.Fagundes, *Rev.Fis.Apl.Instrum.* **8**, 42 (1993).
- [6] [www.prodels.com.br](http://www.prodels.com.br)
- [7] K.Loudon, *Mastering Algorithms with C*, O'Reilly & associates, Inc. 1999
- [8] M.Nelson, J.L.Gailly *The Data Compression Book*. 2<sup>nd</sup> Ed. M & T Books, 1996
- [9] T.L.Yu, *Dr.Dobbs journal* (243) 1996.
- [10] K.Sayood *Introduction to Data Compression*, 2<sup>nd</sup> Ed. Morgan Kaufmann Publishers, 2000.