A CAMAC–USB Crate Controller

Yu. V. Tuboltseva, Yu. V. Chichagova, E. M. Khilkevitchb, and V. D. Simutkinb

a Ioffe Physicotechnical Institute, Russian Academy of Sciences, ul. Politekhnicheskaya 26, St. Petersburg, 194021 Russia
e-mail: tuboltsev@mail.ioffe.ru
b Uppsala University, Marmorvägen 11C, Uppsala, 75244 Sweden

Received April 13, 2009; in final form, July 16, 2009

Abstract—A controller providing communication between a computer and a CAMAC crate via the USB bus is described. For this purpose, the controller includes a DLP-USB245M module, which allows a programmer to work with the controller through a virtual COM port and, at the same time, provides all the advantages of the USB standard. We consider versions of interactions of the DLP-USB module with controller registers on a programmable logic array and on the microcontroller.

DOI: 10.1134/S0020441210010112

In spite of the fact that the CAMAC standard was designed in 1969, it still finds wide application in automated physical experiments in our country and abroad. Without analyzing all the reasons for this lengthy application, we can point to one of them—the availability of a great number of modules in laboratories acquired and created over many years. Experiments based on these modules are still being performed [1]. The connecting link between the CAMAC crate and a control computer is a controller.

There exist controllers providing communications between the CAMAC crate and a computer via parallel data transmission channels (through adapters of the ISA and PCI buses or through a parallel port of a computer [2–7]). Some of them communicate with the computer via serial data transmission channels, operating through a COM port, Ethernet, or USB. Each controller is intended for one or another purpose and has its own advantages and disadvantages.

Thus, when the controller communicates with the computer via the PCI bus, the maximal data transmission rate is ensured. As a rule, communication is provided via a multicore cable (a loop circuit) with a length of ≤2 m. For this purpose, a special conversion board (an adapter) is inserted into a free slot of the computer. In this case, in contrast to the ISA bus, the controller should be addressed via the adapter of the PCI bus by means of special program drivers, writing of which is a complicated problem [8].

In some cases, operation with standard external ports of the computer is required, e.g., while working with portable computers (lately, this has been often used in field experiments or when the user cannot or does not want to install a controller–computer adapter on the PCI bus). Sometimes it is necessary to collect data from equipment located in the crate far away from the computer, e.g., in the experimental room. In these cases, it is possible to use crate controllers operating with computers via standard serial ports, such as Ethernet or USB. To tell the truth, in the latter case, USB–RS422/485 adapters are required for operation at long distances [9]. The controllers operating via the Ethernet are, as a rule, based on PC/104 inner single-board computers [10] or high-power microprocessors and Ethernet interface cards [11].

In our opinion, the cheapest and simplest method of the hardware and program embodiment is to construct a controller with the USB port.

Figure 1 shows a block diagram of the controller. The controller contains ten 8-bit registers connected to each other by local data bus D, via which information is read out of some registers and written into the other registers. Four-bit address bus BA is used to address the registers. The registers are used for storing CAMAC commands (N, A, F, Z, and C), data (R and W), and status information (X, Q, and J). Signals L1–L23 are read directly from crate modules. There is common signal L. The functions of the registers of the controller and their addresses are summarized in the table.

Thus, to prepare a command for writing data into any module, it is required that the module number, subaddress, function, and data be written into registers N, A, F, and Wh–Wm–Wl, respectively. The CAMAC cycle starts operating in response to the Start command. Preparation and execution of read commands differs only in the fact that, while executing the cycle, data, instead of being read from registers W, will be written in response to strobe SI into registers Rh, Rm, and RL from the CAMAC dataway. The control commands are executed with recourse to W and R registers. It should be noted that data written into registers are stored in them and do not require re-writing in the case of their repetition.

To write/read registers of the controllers and execute other commands via the USB bus, we checked
three simple versions of constructing the interface between the USB port and an 8-bit data bus, as well as with a 4-bit address bus.

A 32-Kbyte flash memory with a possibility of programming via the USB is provided for programs. To operate with the microcontroller via the USB bus, it is possible to use the computer cross-platform library with libusb open initial code, avoiding needing to create a driver for it.

The drawback of using the AT89C5131A microcontroller is the need to realize the protocol of communication with the USB port. For this purpose, an AT89C5131A microcontroller with an MSC-51 architecture is used [12]. It has a 48-MHz clock frequency, a 256-byte cache memory, and 1-Kbyte built-in extended RAM.

Fig. 1. Block diagram of the CAMAC–USB crate controller.