

A programmable ISA to USB interface

R.V. Ribas

*Laboratório Aberto de Física Nuclear.
Universidade de São Paulo, Instituto de Física – CP 66318 São Paulo, Brazil*

Abstract. A programmable device to access and control ISA-standard camac instrumentation and interfacing it to the USB port of computers, is described in this article. With local processing capabilities and event buffering before sending data to the computer, the new acquisition system become much more efficient.

Keywords: Data Acquisition. Camac instrumentation. USB port.

PACS: 29.50.+v; 29.85.Ca

INTRODUCTION

This is the last part of a series of development to revival and update camac instrumentation in our Laboratory [1]. We are planning to make major changes in the DAC systems in the near future, so presently it is not convenient to invest in new camac interfaces. All but one of the crate-controllers in use presently at our laboratory, are interfaced to the computer, through the ISA bus. Mother boards with ISA slots can still be found in the market, but they are very expensive and difficult to find. With the experience in the previous development of camac interfaces to systems with few camac modules [1], we developed the ISA to USB adapter, that can be used with the two different types of crate-controllers we have in our laboratory.

THE HARDWARE

ISA, the *Industry Standard Architecture* [2] was conceived in early 1980 at the IBM Laboratories. The first generation of the IBM-PC used a 8-bit version of this bus, and the full 16-bits ISA was later implemented in the PC-AT. Since the 8-bit version is a subset of this bus, many devices developed for long time after the AT, still used only the 8-bits version. The ISA bus was later replaced by the PCI - *Peripheral Component Interconnect* bus in early 1990. For many years, most mother boards available in the market included also one or two ISA slots, to accommodate old boards. In the last 5-6 years only in special, expensive boards, usually related to industrial applications the ISA slots could be find. Presently is virtually impossible to find new mother boards with these slots.

The cost and the life-time of laboratory instrumentation are in general much larger than typical computer boards like sound or video cards, and so, in most cases, it is convenient to find solutions to adapt them to newer available interface buses.

In the case of real-time data acquisition systems like camac, local processing capability and data buffering is also necessary to improve efficiency in the event handling. We developed a simple ISA to USB interface, using a microcontroller with integrated USB capabilities, the EZ-USB FX2 from Cypress [3]. This same device, in an evaluating board from Siphec [4], was previously used in a similar interface, dedicated to up to four camac modules [1].

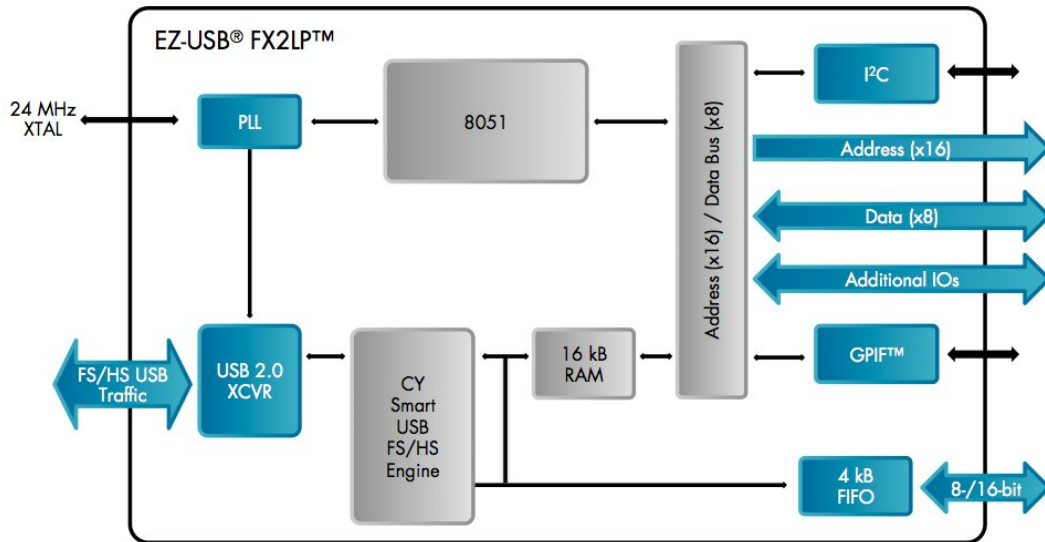


FIGURE 1. Block diagram showing the USB interface, FIFOs and the microcontroller in the Cypress EZ-USB FX2 chip [3].

ISA EMULATION

The signals necessary to operate a ISA port IO cycle are software produced in the FX2 program. This makes the interface very simple, with only one TTL chip that is used to divide the microcontroller clock (48MHz) and produce the 12 MHz clock to the ISA bus. For a 16 bit IO read, for instance, the sequence of code (c-language) necessary is:

`IOC=0xAA; IOC=0x8A; NOP; NOP; PAT.B[0]=IOA; PAT.B[1]=IOE; IOC=0xdf;`

The initial value of the register C is always `IOC=0xdf`. The read data being stored in the two bytes of PAT. NOP are the necessary delays to conform to the ISA cycle, as shown in Fig. 2. Only the signals necessary to port I/O were implemented in the project. The ISA signals, the corresponding ports in the FX2 processor and their meaning are shown in Table 1.

Signal	Port	Meaning
RESET	E1	Reset the ISA bus
BALE	C5	Bus Address Lach Enable

CLOCK	-	Clock from EZ-FX2 is divided by four to get 12 MHz for the ISA BUS
$\overline{\text{IOW}}$	C7	I/O Write (data to ISA)
$\overline{\text{IOR}}$	C6	I/O Read (data from ISA)
IRQ3	B1	Interrupt request – set by a camac LAM
SA	C0-C4	Lower 5 bits of Port number. Higher bytes are hardwired
SD	A,D	Low and High data bytes
AEN	-	Address enable (DMA) – always zero (disabled)

TABLE 1. ISA signals used in the present implementation.

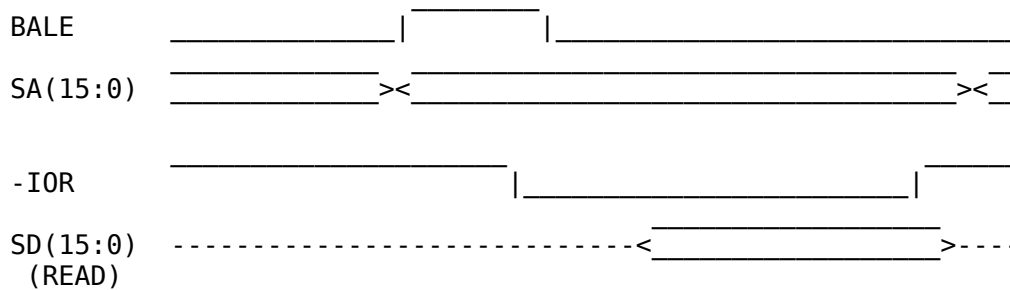


FIGURE 2. ISA bus signals for a 16 bit port IO cycle [2].

Performance

The interface has been used with two different crate controller. The DSP 6002 with a PC004 8 bit ISA interface, and the CAEN C111A, with the A151A-Turbo 16 bit ISA interface. The performance of the new interface was determined by the measure of the total time to service a LAM signal, compared to the same time in a standard ISA bus. Events generated by a pulser, consisting of eight data readout from one module, plus a single clear-module camac instruction. In the prototype board, there was no clock divider, so the microcontroller had to operate in the same clock ISA bus. For the PC004, we were able to run the clock at 24 MHz (3x the standard ISA clock), but for the A151A, it had to be reduced it to 12 MHz. For the 8 bit interface, that was tested more extensively, and the software better developed, the LAM service time was about 90 μs , compared to 190 μs in the same situation, with the standard ISA bus. For the A151A the measured time was much larger, but we believe that similar results will be obtained, using a processor clock of 48 MHz, with 12 MHz in the ISA bus. Also, the possibility of using the high speed bus clock that is available in the A151A-Turbo was not evaluated yet.

CONCLUSIONS

A simple, programmable ISA to USB interface was developed and used with two different ISA crate-controllers. The interface has been routinely used in several experiments, with good performance and stability. From the user point of view, the new interface is identical to the previous one we developed [1].

REFERENCES

1. R.V. Ribas – AIP Conf. Proc. 1425, DOI 10.1063/1.3688842, (2012)
2. ISA bus summary - <http://www.techfest.com/hardware/bus/isa.htm>
3. Cypress - <http://www.cypress.com/?id=193>
4. Siphec – <http://www.siphec.com/>