A CBLT and MCST Capable VME Slave Interface

Frank Würthwein [a,1], Charlie Strohman [b], Klaus Honscheid [c], Vitaliy Fadeyev [d], Jens Staeck [d], Igor Volobouev [d]

[a] *Caltech, Pasadena, CA 91125*

[b] *Cornell University, Ithaca, NY, 14853*

[c] *Ohio State University, Columbus, OH, 43210*

[d] *Southern Methodist University, Dallas, TX, 75275*

We report on the development of a VME slave interface for the CLEO III detector implemented in an ALTERA EPM7256 CPLD. This includes the first implementation of the chained block transfer protocol (CBLT) and multi-cast cycles (MCST) as defined by the VME-P task group of VIPA[1].

Key words: VME, CBLT, MCST, Block transfer, DAQ, VME64, CLEO

1 Overview

Within VME64[2] there is no operation that guarantees efficient readout of large blocks of data that are sparsely distributed among a series of slave modules in a VME crate. This has led the VME-P task group of VIPA to specify protocols that enable a master to address many slaves at a single address. Which slave is to drive the data bus is determined by a token passing mechanism that uses the *IACKOUT, *IACKIN daisy chain. This protocol requires no special features from the master besides conformance to VME64. Non-standard features are restricted to the VME slave interface.

The CLEO III detector comprises $\sim 400,000$ electronic channels that have to be digitized, sparsified, and stored within $20\mu s$ in order to incur less than 2% dead time at an anticipated trigger rate of 1000Hz. 95% of these channels are accounted for by only two detector subsystems, the silicon microstrip detector (125,000 channels), and the ring imaging Čerenkov detector (RICH) (230,400 channels). The occupancy after sparsification in either of these two detector subsystems is expected to be less than 1%, resulting in event fragments on the order of 10KBytes each, spread over 4, and 8 VME crates, respectively.

We developed a chip set that sparsifies, tags, and stores the incoming digital data on the data boards, and includes a VME slave interface that implements MCST and CBLT protocols. In this paper, we restrict our discussion to the VME slave interface. The VME slave interface for CLEO III (VME chip) supports

- Decode and store geographical address (GEO) after RESET or power up as specified by VME64 Extensions Draft Standard [4]
- A24/D32 interface for VME chip initialization, and A32/D32 to initialize all other components of the data board.
- Multicast (MCST) cycle as "load next event" signal
- Chained Block Transfer cycle (CBLT) for event readout
- Local Resource Management
- Control and Status Register (CSR)

These features are described in some more detail in the sections to follow.

[1] E-mail: fkw@cithe502.cithep.caltech.edu

2 Board Initialization and Geographical Address

The CLEO III VME data boards are initialized via a combination of hardware and software initialization. As part of the hardware initialization the VME slave interface chip latches the geographical address (GEO) into the lowest 5bit of the Base Address Register (BAR) with de-assertion of *RESET if GEO is a valid geographical address. Geographical addressing is implemented on pins $9-11, 13, 15, 17$ of row D of P1/J1, as described in Reference [4]. Pin 9 is a parity bit while the other pins form a 5bit address. For a GEO to be valid, we require it to lie in the range $1-21$ and have the correct parity [4]. In addition to this, a "GEO_FAIL" bit in one of the control and status registers (CSR) is set with de-assertion of *RESET should the goegraphical address be invalid. A "BOARDFAIL" light is lit, and *SYSFAIL is asserted if the data board detects an error condition during hardware initialization. For debugging purposes, *SYSFAIL can be reset in software by clearing the "sysfail enable" bit in the CSR.

Software initialization of the data boards includes setting up registers in VME chip as well as various other components on the data boards. The latter is done via A32/D32 read/write access. The former is accomplished via A24/D32 read/write cycles. In addition to AM code 0x2F that is specified for CSR accesses in VME64, we support accesses to the CSR via all other A24 AM codes.

The default settings of the CSR register after hardware initialization have been chosen such that the slave interface is fully functional except for MCST and CBLT cycles. These registers are implemented as part of VME chip. In addition, we have implemented the conventional VME64 and VME-P CROM at its nominal addresses. More details on the various CSR and CROM registers and the corresponding address map can be found in References [1] and [2].

3 Multicast (MCST)

In CLEO III, the MCST protocol is used as a "load next event" command that is broadcast to all data boards that participate in the CBLT event readout. All data boards thus receive the "load next event" command simultaneously, pass a token through the *IACKIN/*IACKOUT daisy chain, prepare the data board for readout of the next event, and latch the first data word.

There are two bits in the CSR, FIRST and LAST, that indicate the position of a given board in the readout chain. The FIRST board asserts *IACKOUT as soon as it has decoded the "load next event" command. Any of the intermediate boards assert their *IACKOUT once they have decoded the "load next event" command and have their *IACKIN line asserted (by their upstream neighbor). The LAST board asserts *DTACK once it has decoded "load next event" and has its *IACKIN line asserted. This "token" passing through the whole chain guarantees that all data boards have decoded the "load next event" command before *DTACK is given. *IACKOUT is de-asserted with the de-assertion of address and data strobe of the MCST.

Whether or not a data board participates in CBLT event readout is determined by the "CBLTEN" bit in the CSR. VME-P allows for separate MCSTEN and CBLTEN bits in this register. We have implemented this accordingly. In principle, it is possible to disable a data board in the middle of the readout chain. A disabled board will just pass on the token without any further action. Similarly, VME interrupters can be located either before or after the readout chain.

4 Chained Block Transfer (CBLT)

Chained block transfer or CBLT cycles are read operations only. The basic idea of a CBLT is to have a VME master address several slaves at once. The slaves determine who is to drive the data bus in

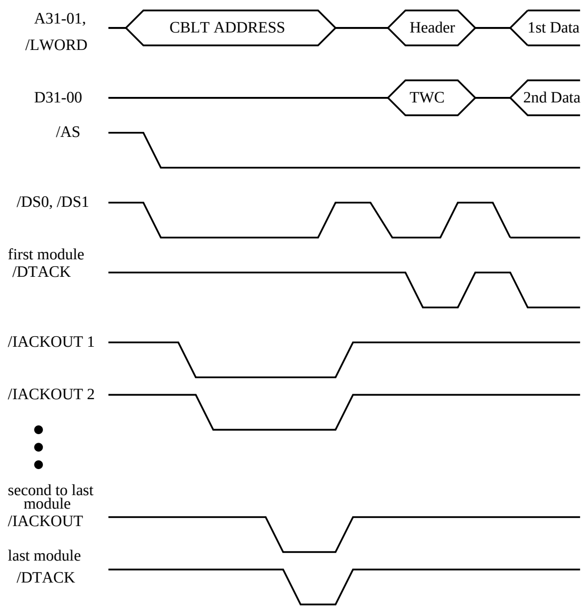

Fig. 1. Schematics of timing diagram for the address part of a D64 CBLT.

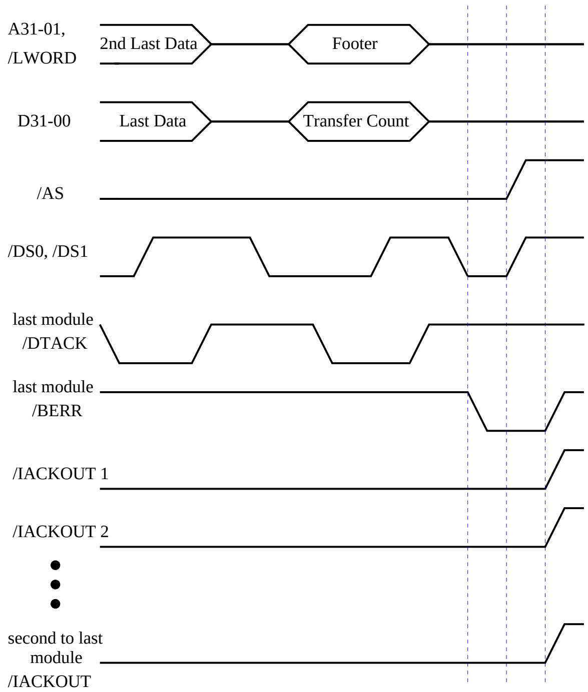

Fig. 2. Schematics of timing diagram for the end of a D64 CBLT.

response to a data strobe by passing a token via the *IACKIN/*IACKOUT daisy chain. *BERR is asserted by the last slave after all data is read out for an event. The VME master reads a CBLT status byte from the CSR of the last slave after assertion of *BERR. This is necessary to distinguish proper completion of event read-out from the assertion of *BERR by the bus timer in the crate. From the perspective of the master in a crate a CBLT is nothing more than a block transfer to the CBLT address that is terminated by a bus error. Schematics of timing diagrams of beginning and end of a CBLT are shown in Figures 1 and 2.

When to pass on the token is determined by the "PURGED" bit in CSR. This bit is set when the data strobe is de-asserted after read out of this data board for this event is completed. A PURGED data board does no longer participate in the CBLT other than passing on the token. In CLEO III, the PURGED bit is cleared only by the next MCST. The VME master can therefore pause the block transfer, e.g. to insert a page break or to service an interrupt. Once the CBLT is resumed, the token is immediately passed on to the appropriate data board by reestablishing the *IACKIN/*IACKOUT daisy chain up to the point where the CBLT was interrupted.

The last board does not assert its *IACKOUT line after its PURGED bit is set. Instead, it asserts *BERR as response to the next data strobe connected with a valid CBLT cycle.

It may be worth noting that standard VME CPU boards tend to include buffering in form of a FIFO as part of their VME interface chips. CBLT readout requires that the content of any such hardware buffer is copied to its final destination in memory even if the readout is terminated by *BERR.

During the address cycle of a D64 CBLT we require that *DTACK is not asserted unless all data boards that need to be read out have recognized the CBLT transfer. Again, this is accomplished by token passing mechanism using the *IACKIN/*IACKOUT daisy chain. This is not necessary for D32 CBLT as the VME master drives the CBLT address on the address lines throughout the D32 CBLT.

The CLEO III VME interface chip is completely designed and it has been successfully used in prototype data boards for several CLEO III detector components. We have measured a throughput of 32 MBytes/s using a 100 MHz MVME1604 CPU board.

References

[1] VME for Physics "Implementation Recommendations & Guidelines, Draft Document" (*http://www-vipa.fnal.gov/vipa/doc2.html*). VME International Physics Association (VIPA) (*http://www-ese.fnal.gov/vipa/default.htm*) VME International Trade Assn (VITA) (*http://www.vita.com/*).

[2] VME64 Standard, ANSI/VITA 1-1994.

[3] CLEO III DAQ System Technical Design Report *CLEO-CBX-95-104*.

[4] VME64 Extension Draft (*http://www.forcecomputers. com/technology/papers/v64ext.html*).