Physics Department
BROOKHAVEN NATIONAL LABORATORY
Associated Universities, Inc.
Upton, New York 11973

EXPERIMENT 787
Technical Note No. 149

E787 FASTBUS I/O Register

Maged Atiya, Mark Ito, and Chu Ng

January 20, 1988

1. Introduction

The E787 FASTBUS I/O Register is a single width module with the following features:

- 64 inputs.
- 16 outputs, selectable separately as a level or pulse.
- A free running timer with 20 ns least count.

The board has many uses in the experiment, including reading information from the trigger for data events, registering the end-of-spill event, clearing the trigger system, timing SSP programs, and registering the relative time of data events within the spill.

2. Operating Instructions

2.1 Logic Levels

All inputs to the board are assumed to be differential ECL. The input impedance is a nominal 100 Ω. Signals that have no input connection will go to logical low. Output signals are also differential ECL. Each end of the signal has a pull down resister to V_{EE}, but the line must be terminated by the user on the receiving end.

2.2 Connectors

All connectors used at the front panel of the board have the standard 100 mil pin spacing, 25 mil square posts. Their arrangement is shown in figure 1.

2.3 Inputs

The 64 input channels are received on four 17×2 pin connectors. One pair of each connector is not used. The data is read using a block transfer of two words from data space, address 0.

2.4 Input Valid

The input valid is the signal that causes all 64 bits of data present on the input connectors to be latched on the board. This signal can be generated in two ways: via a signal to the front panel of the board or by a write to a location in CSR space.

The front panel valid is a differential signal received on a two-pin connector. Data is latched on a low to high transition.

The software valid is produced by writing a single word to CSR space, address C0000002 (hex). The datum used in this operation is ignored.

Figure 1 Front panel.

2.5 Outputs

The 16 output channels are produced on one 17×2 pin connector. One pair of the connector is not used.

The channels are turned on and off by setting bits 0 through 15 of the mask register, location C0000001 (hex) of CSR space. A value of 0 turns the channel off, 1 turns it on.

Each channel is selectable separately as a level or a pulse. If the channel is selected as a level, it puts out logical low if it is off and logical high if it is on, at all times. If selected as a pulse, and if the channel is turned on, a pulse is issued on a write to the width register, CSR space location C0000000 (hex). If the channel is off, no pulse is issued. Bits 16 through 31 of the mask register select level or pulse. A value of 0 selects pulse, a value of 1 selects level.

The width of the pulse is given by

$$\text{width} = 20 \text{ ns} \times \text{value}$$

where value is the number written to the width register. All channels issue a pulse with this width. The maximum value allowed is 255 (5.10 μs). A value of 0 gives no pulse.

2.6 Output Valid

The output valid is a single signal that can be used to latch output data at their destination. The valid is a pulse, generated on a write to the width register, with the same width as given in section 2.5. The output valid signal always generates a pulse. It cannot be "turned off" and it cannot be selected as a level. There is no built-in time delay between the pulses coming from the output port and the output valid.

2.7 Timer

The timer has a least count of 20 ns and 32 bits giving a full-scale of 86 seconds. It is read using a block transfer of one word from data space, address 1. The timer is not affected by input valids or reads and writes to other registers on the board. There is an interruption of several counts when the timer itself is read, to insure a stable value during the read operation.

Upon a reset to the timer, its value is set to zero. The reset can also be generated either from the front panel, or with a FASTBUS operation. The front panel reset is received on a standard 2-pin connector. The value of the timer will be held at zero as long as the reset signal is held high. The software reset is achieved by a single-word write to CSR space, address C0000003 (hex), the datum being ignored. After the FASTBUS operation, the timer begins incrementing, without any subsequent action by the user.

2.8 Operation Modes

Some features of board operation are controlled by bits in CSR0. These involve the way in which the board handles the valids it receives. The operation modes described make no distinction between software valids and front panel valids.

2.8.1 First-Valid / Last-Valid Mode

After the board is set in first-valid mode, only the first valid received will latch the input data. All subsequent valids are ignored until first-valid mode is reset (see below).

When the board is in last-valid mode, every input valid received causes the data to be latched, therefore the board reports the state of the inputs when the last valid was received.

Bit 2 of CSR0 controls this mode. A value of 0 indicates last-valid mode, a value of 1 indicated first-valid mode. First-valid mode is reset by toggling the bit to 0 and back to 1 again.

2.8.2 Post-Busy / No-Busy Mode

After the board is set in post-busy mode, SS=1 will be posted on a block read of data space, unless a valid has been received. After a valid is received, the board will post SS=0 (normal completion), until post-busy mode is reset (see below). This mode can be used to insure that new data has been sent to the board since it was last read.

In no-busy mode, SS=1 is never posted.

Bit 7 of CSR0 controls this mode. A value of 0 indicated no-busy mode. A value of 1 indicates post-busy mode. Post-busy is reset by rewriting 1 to bit 7 (the user need not toggle the value to 0 to reset this mode).

2.9 Summary of FASTBUS Operations

The board supports the following FASTBUS operations:

1. Random read and write to CSR space.
2. Block read from data space.
3. Broadcast, Case 4: All devices seeing the Broadcast assert TP during following Read Cycle.

The board does not support the following FASTBUS features:

1. Assertion or response to WT.
2. Random transfers to/from data space (SS=6 will be posted).

Table 1 gives a description of CSR space, Table 2 gives a description of data space, and Table 3 gives a map of CSR0.

2.10 LED's

From top to bottom the LED's provided on the front panel are:

1. SLAVE: On when the board is addressed (AS/AK lock).
2. OUT: Indicates that output pulse is being generated.

Table 1

Map of CSR Space

Address (hex)	Name	Description
0	CSR0	Sets board operation modes. See Table 3. FASTBUS ID number is 7870 (hex).
C0000000	Width	Sets width of output pulses and width of output valid. Least count=20 ns. Max value=255.
C0000001	Mask	Bits 0:15 select output on/off. Bits 16:31 select pulse/level (0/1).
C0000002	Valid	Latches input data. Logically OR'ed with front panel input valid. Datum ignored.
C0000003	Reset	Sets value of timer to 0. Logically OR'ed with front panel reset. Datum ignored.

Table 2

Map of Data Space

Address (hex)	Name	Description
0	Data	Input data. Read two 32-bit words with block transfer.
1	Timer	32-bit value of timer. Use block transfer.

Table 3

Map of CSR0

Bit (0-31)	Name	Description
2	First Valid	Selects First-Valid/Last-Valid mode. 0=Last, 1=First.
7	Post Busy	Selects Post-Busy/No-Busy mode. 0=No-Busy, 1=Post-Busy.

Turn off bit n by writing 1 to bit $n+16$.

3. FIRST VALID: On when in first-valid mode, off when in last-valid mode.

4. POST BUSY: On when in post-busy mode, off when in no-busy mode.

5. VALID: Indicates that a valid signal has been received.

6. TIMER: Bit 24 of timer. Blinks on and off when timer is incrementing.

3. Functional Description

Figure 2 shows a block diagram of the I/O board. The various sections and their function are described below.

3.1 Internal Buses

There are two internal buses, the AD bus and the pipeline bus, each 32 bits wide. All registers that do so output their data onto the AD bus. All registers that receive data get it from the pipeline bus.

The FASTBUS input buffers, which pass the 32 bits of the FASTBUS AD lines, the mask register buffers, the timer buffers, the CSR0 buffers, and the input data buffers all enable their data onto the AD bus during the various operations performed by the board. The only register that receives data from the AD bus is the pipeline register.

The pipeline bus always reflects the contents of the pipeline register. The pipeline bus provides data to the FASTBUS AD lines when the board is being read, and to the internal registers: the mask register, the NTR, the width register, and CSR0. See section 3.7 for a description of the pipeline register.

3.2 Inputs Receivers and Latches

This section of the board receives the 64 differential input signals and produces single-ended signals internally, latches them, and outputs them onto the AD bus via buffers when the data is read, 32 bits while DS=1 the other 32 bits while DS=0. The both halves of the differential input signals are terminated in 56 Ω to $V_{\rm BB}$. The

Figure 2 Block diagram of the I/O board.

positive ends of the signals are tied to -5 V through 3.3 kΩ in order to give the inputs a definite state (low) in the absence of a input cable.

3.3 Mask Register, Output Logic, and Output Drivers

The mask register is 32 bits wide and receives its input from the pipeline bus. It outputs its bits on the AD bus through a set of buffers when read back. The lower order 16 bits determine whether an output channel is on or off, and the higher order 16 bits determine whether it is a pulse or level. The output logic consists of 16 identical sets of gates that take input from the mask register and the width generator in order to produce the desired combination of levels and pulses. From the single-ended signals generated from the output logic, the output drivers produce the differential signals on the output connector. Each side of the the signal is pulled down through 470 Ω to -5 V.

3.4 Width Generator

The width generator produces a negative logic pulse with a width proportional to the value written to the width register starting at the time of the write. This pulse serves as input to the output logic and directly determines the width of all output pulses. The value of the width register is not preserved and cannot be read. The width register receives its input from the pipeline register. All the input bits are inverted and broadside loaded into an 8-bit (2×4-bit) synchronous counter upon receipt of the enable signal from the NTR. The loading of the counter also initiates the pulse. The counter then counts up to its terminal count, the terminal count signaling the end of the pulse.

3.5 Timer

The timer is a 32-bit (8×4-bit) ripple counter that increments continuously, with a 20 ns least count. The counter is only inhibited while it is enabled onto the AD bus during read back, for 3 or 4 counts. The counter can also be reset to zero. The reset signal is a logical OR or the front panel reset and the software reset.

3.6 Next Transfer Register (NTR)

The NTR is responsible for decoding the FASTBUS secondary address and providing enables and clocks to the other sections of the board as needed. The secondary address is only partially decoded. In particular, only the value of bits 0, 1 and the OR of bits 30 and 31 are used. The various enable and clock pulses are actually generated by the FASTBUS logic, with different signals for operations involving CSR and data space. The NTR passes these pulses along to the required sections based on the value of the secondary address.

3.7 Pipeline Register

The pipeline register latches the value of the AD bus on both rising and falling edges of DS. This allows the AD bus to change state, in preparation for the next data cycle, while the pipeline bus preserves the data present at the previous DS edge.

3.8 FASTBUS Input and Output Buffers

The board contains two independent sets of buffers to handle the 32 FASTBUS AD lines. One set buffers the FASTBUS lines onto the internal AD bus (during FASTBUS writes) and the other enables the pipeline bus onto the FASTBUS (during FASTBUS reads). In this way, as far as its relation to the internal buses is concerned, the FASTBUS crate segment is handled like the other registers on the board.

3.9 FASTBUS Logic

Most of the logic necessary to handle FASTBUS operations is contained in three PAL's. Since these are TTL devices, ECL-TTL conversion must be performed on most of the FASTBUS control lines. The functions performed include interpretation of the MS lines, sending clocks and enables to the NTR at appropriate DS edges, clocking the pipeline register, and generating SS codes. SS codes are generated only for data cycles. In general, SS=6 is posted for operations that are not supported.