Buffer Manager – Buffer Controller Communications

27 July, 2000

1. Introduction

Storage/readout of L3 data to/from buffers in the L3 Data memory in the Buffer Controllers (BC) on all the cards in the system is coordinated by the Buffer Manager (BM) on the FRC. Communication is accomplished by a dedicated bus on J3[1] consisting of 4 command lines, 8 message lines and a message strobe for sending signals from the BM to all the BCs simultaneously. The status of the BCs is returned to the BM on 12 status lines, which are the wire AND or OR of all the BCs. This protocol is adapted from that used by the VRB-VRBC[2], but uses different definitions for commands, messages and statuses. The changes have been made to allow a more complete handshake to be done between the BM and BCs on both L1 and L2 actions.

2. BM to BC Communications (messages)

See the *STT J3 Backplane* [*2*] document for a mapping of command, message and strobe lines to J3 pins.

cmd [0-3]	***mess [0-7]***	**L1/L2**	**Purpose**
0			free
1	*PUT_BUFF*	L1	Number of next buffer in BC Mem to put data from DB
2	*PUT_BX*	L1	BX for cross-check with data put to BC Mem
3	*PUT_END*	L1	All BCs done putting data from DB to BC Mem
4	*GET_BUFF*	L2	Number of next buffer in BC Mem to get data
5	*GET_BX*	L2	BX for cross-check with data got from BC Mem
6	*GET_END*	L2	All BCs done getting data from BC Mem to Output fifo
7-15			free

Table 1: Command numbers and messages for BM to BC communications.

3. BC to BM Communications (status)

The BCs return status information to the BM on 12 lines on J3. Each status line has a specific meaning (see Table 2) and is the result of a wire AND or OR on the backplane of the individual signals from each BC. See the *STT J3 Backplane* [*2*] document for a mapping of these lines to J3 pins. Note also that two lines labeled as *reserved* in the standard J3 backplane layout are added to the status lines for the STT.

stat	**Name**	**L1/L2**	**Logic**	**Purpose**
0	*PUT_DONE*	L1	AND	Status BC putting data to Mem (see Fig. 1)
1	*PUT_DONE**	L1	OR	*put_done* inverted for wire OR
2	*GET_DONE*	L2	AND	Status of BC getting data from Mem
3	*GET_DONE**	L2	OR	*get_done* inverted for wire OR
4	*STT_L1BUSY**	L1	OR	L1 Busy (request holdoff of L1 Accepts)
5	*STT_L2BUSY**	L2	OR	L2 Busy (request holdoff of L2 Accepts)
6	*STT_L1ERROR**	L1	OR	L1 Error (request SCL_INIT)
7	*STT_L2ERROR**	L2	OR	L2 Error (request SCL_INIT)
8-12				free

Table 2: Status line definitions for BC to BM communications.

4. BM-BC Handshaking for L3 Data Transfers

Signals passed between the BM and BCs during transfer of L3 data into and out of the BC Memory buffers are slightly more complicated than those used in the VRB-VRBC communications. In the VRB-VRBC system, one message from the VRBC initiates the storage/retrieval of L3 data into / out of VRB buffers (*READOUT BUFFER NO.* / *SCAN BUFFER NO.*) and one status line (ORed on the backplane) marks the action as complete by all VRBs (*READOUT_BUSY** / *SCAN_BUSY**). Under this protocol, one of the VRBs can be broken such that its *xxx_BUSY** signal is stuck low, and the VRBC will not recognize this. While this scenario is unlikely, it makes us more comfortable to do a full handshaking between the BM and the BCs, ensuring that all BCs start and finish any action.

Shown in Fig. 1 is the sequence of signals for handshaking between the BM and the BCs when L3 data is transferred to (put) or from (get) the BC Memory. In this proposed scheme the BM sends the *get/put* message followed by the *BX* message to all the BCs. It then waits for all BCs to finish the get/put operation as signaled by the OR of all *xxx_DONE*[*i*] going high. The BM then sends the *end* message to all BCs, which respond by setting their /*xxx_DONE*[*i*] high. The BM checks to see that the wire-OR of all /*xxx_DONE*[*i*] (negative logic) goes high signaling that all BCs are now ready for the next get/put operation. Note that two status lines, which are the inverse of each other, are necessary for this scheme to work because both an AND and an OR are required.

Since the BC Memory is dual port and since separate status lines exist for *put* (L1 accept) and *get* (L2 accept) operations a get and a put can be performed simultaneously. Individual gets and puts cannot be pipelined however.

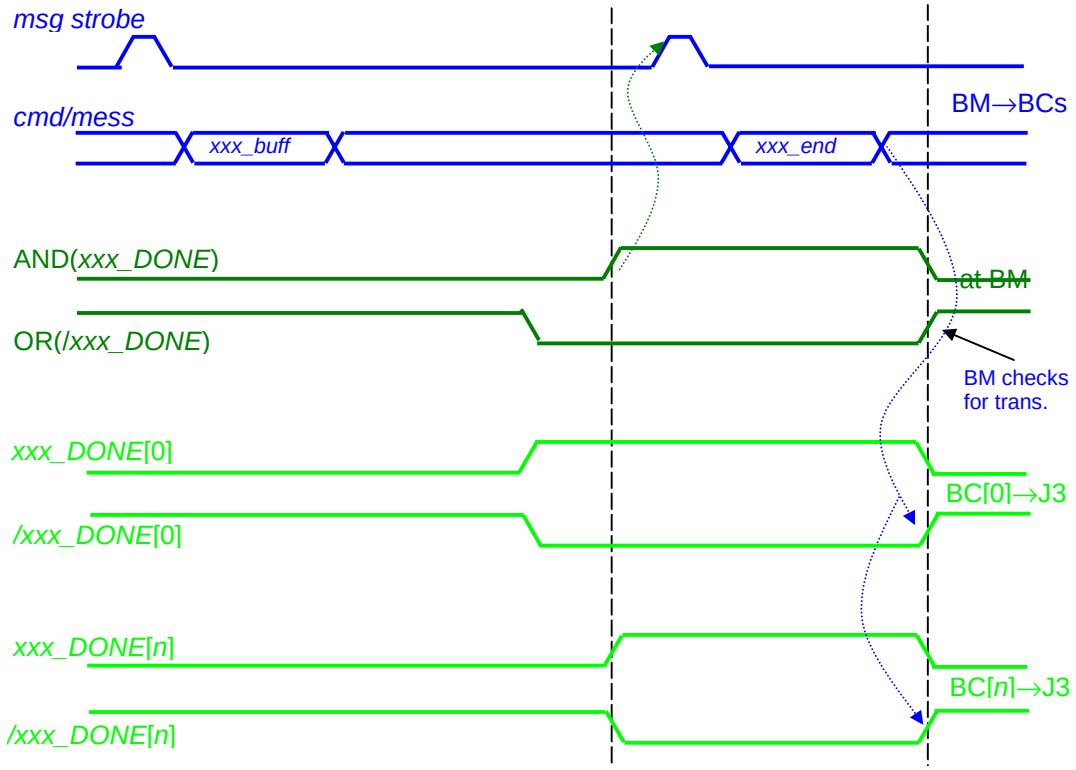

Figure 1: Timing of signals between the BM and BCs for a "put" or "get" operation.

References

[1] *STT J3 Backplane* (19 July, 2000) http://www.nevis.columbia.edu/~evans/stt/sb_0700/J3_STT.pdf

[2] M. Bowden, et al, *VME Readout Buffer (VRB)* (22 October, 1999) http://www-ese.fnal.gov/SVX/Production/SVX_Web/VRB/vrbspec.pdf