Frequently Asked Questions

Title: 1553Px module [BC mode] How to simultaneously monitor messages
Date: 20 May 2015
Card/Board/Module: Exc1553Px family (e.g., M4K1553Px module)
Operating System: all

Question:

Can a single M4K1553Px device be used to simultaneously behave as a Bus Controller and to capture data returned from an external RT? The Test Equipment M4K1553Px needs to be the Bus Controller and is also needed for simultaneously monitoring data returned from the RT.

Do I need two M4K1553Px modules installed on the EXC-4000PCI card - one for the BC and one for the Monitor ? Or can I get by with only one module ?

Answer:

1. As per the 1553 spec, the BC is the active player in the protocol, creating a bus list of messages which it transmits over the 1553 bus. The messages are of multiple types including BC2RT (BC sends the data words to the RT), RT2BC (BC requests that the RT transmit back data), RT2RT (BC requests that one RT receive data, and a different RT transmit the data).

Since the BC must verify that the message was properly completed, the protocol rules include that the BC will verify (for RT2BC & RT2RT messages) that it detected the receipt of the requested number of words from the RT.

So, by nature of the protocol, the BC will receive the data transmitted by the RT. And, our API supplies functions for retrieving any message in the bus list, at any given time.

2. To help retrieve messages, we added a feature called **Internal Concurrent Monitor** (**ICM**) in both BC and RT modes. This sets aside a section of memory on the module to act as a quasi-Monitor (within the confines of the BC or RT).

In the **ICM** memory area, we store a record of all message traffic on the bus. Effectively, we have a Monitor stored within the memory area of the BC. All data words that the BC sends over the bus to an RT (BC2RT message), and all data words transmitted by some RT over the bus (RT2BC or RT2RT messages) are recorded in the ICM, along with the command words & status words, and an attached timetag stamp (plus an additional internally built message status word).

Each message is stored sequentially in the buffer. Buffer size is defaulted to 409 blocks.

You can read more about this memory area in the User's Manual (hardware), chapter 5.

You can read up on the API functions used for monitoring in the Programmer's reference (software). use functions Get_Next_Message_BCM_Px or Get_Next_Mon_Message_Px.

Here is a snippet from the software manual:

Get_Next_Mon_Message_Px reads the message block following the message block read in the previous call to Get_Next_Mon_Message_Px.

This function can be used in all modes, instead of:
Get_Next_Message_Px in Monitor mode,
Get_Next_Message_BCM_Px in BC mode or
Get_Next_Message_RTM_Px in RT mode.

3. Bottom line: If you want to be a BC and still be able to get all of the message contents as individual messages (aka Monitor), you should be able to do this with a single module in BC mode, reading messages from the ICM.

Please see our demo program demo_cmon.c which uses this methodology.