FROM: M[redacted] McClung, 76336,3110
TO: Dave Brown, 72103,2235
DATE: 11/11/94 5:09 PM

Re: WOSA/XMC, Draft 2, Interrupts

Hello Dave,

This correspondence will conclude my discussion of the ITimer interface and provide some feedback on interrupt support in the WOSA/XMC model.

ITimer Interface - continued ---------- Previously I indicated that there would be no need for a SetResolution() method in ITimer. However, a GetResolution() method would be useful, especially with our servo products.

In addition to the SetTimeDelay() and GetTimeDelay() methods I neglected to include the most important method, DoTimeDelay(), which invokes the time delay.

Interrupts ---------- To date most of our customers have not exploited the interrupt capability of our bus-based 6000 products. I would like the WOSA/XMC model to address this problem by providing interfaces or methods that insulate them from the low-level details of interrupt handling.

The bus-based 6000 products can interrupt the PC in 4 different ways: 1. The bus-based product can interrupt the PC when it has data in its output buffer (for interrupt driven communications). 2. The bus-based product can interrupt the PC when its input buffer is empty (for interrupt driven communications). 3. The bus-based product can interrupt the PC when a motion, I/O, timer or counter event has occurred (see INTHW command). 4. The bus-based product can interrupt the PC when the "fast status" registers have been updated.

I would imagine that the Driver Administrator would be the place to specify the kind of interrupt support that is required. For example, the user might want the bus-based product to interrupt the PC when a timer event has occurred. In the Driver Administrator application, the user would check support for the interrupt described in #3 above. In his application, the user would invoke the SetInterrupt() (or EnableInterrupt()) method to enable a timer event interrupt (INTHW.25-1). Most likely the client application needs to be immediately notified of the event so that it can respond appropriately. Or perhaps a callback function can be attached to the Motion Component so that it can handle the event. Have you worked out a general way of handling interrupts in the WOSA/XMC model?

[redacted]

RGB00051665