Function Sequence

This sequence is used to function a node. A 50-53 instruction will initiate this sequence. Once the instruction is issued the Channel Busy Flag sets and a function pulse will be sent out on the channel loop. Four consecutive parcels of data will also be sent out on the data lines. These four parcels contain the following information; node designator, node function, interrupt address, and parameters. The node designator will select only one node to be functioned. The node function is the function being performed. The interrupt address (sometimes referred to as call pointer or response address) is loaded in the selected node. It will be used in a call sequence which is described later. (Note: Not all functions use the interrupt address.) The parameters are used by the node function. See specific node write ups for further descriptions.

The function pulse along with the four consecutive parcels will go from node to node in a daisy chained fashion back to the foreground processor. One node will recognize its node designate and latch up the parcels. The functioned node has the option of using the channel loop or not using it which is dependent on the function type. Once the node is done using the loop a response pulse is sent from the node to the foreground via the channel loop. Two parcels of data are also sent with the pulse and the foreground latches it up. The channel busy flag in the foreground is then cleared. (Note: The two parcels are not used by all functions.)

There are basically two different types of functions that can be sent to a node. A function that can be completed immediately or a function that takes time to complete. Let's look at a few examples using a disk controller node.

An example of a function that can be done immediately is, read the status register of a disk controller node. The node will receive the function from the foreground and respond immediately with a response pulse and two parcels of data back to the foreground via the channel loop. The two parcels will contain the status register value. The response pulse clears the channel busy flag on the foreground which frees the channel to do other sequences.

An example of a function that takes much time to complete is a read cylinder address register of the disk drive. This operation functions the node to function the disk drive to read the register back to the node. This operation takes microseconds to complete. The main objective is to use the channel loop as efficiently as possible. This means the functioned node will respond immediately with a response pulse back to the foreground before the node tries to read the cylinder register of the drive. This will clear the channel busy flag, thus allowing the foreground to function other nodes on the channel. Eventually the previously functioned node will be done reading the drive cylinder address register. The foreground will be calling for interrupt. (See call sequence) The node will respond to the call sequence. The Foreground reconizes the node and functions the node to return the cylinder address value. Sequence is complete.