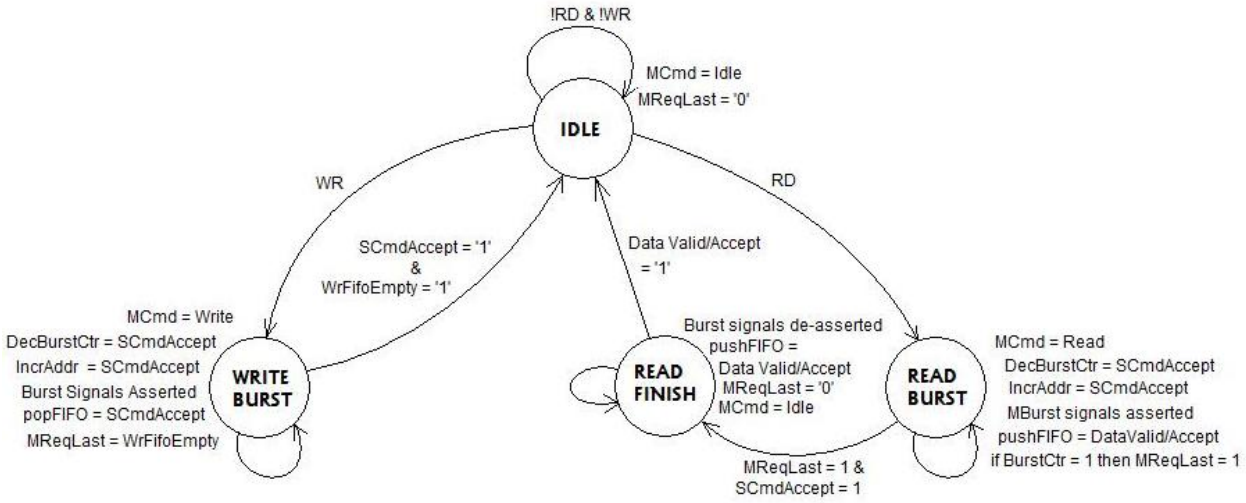

Interfacing with OCP-IP

OCP-IP, for Open Core Protocol International Partnership, intends to provide a method of interconnecting intellectual property (*IP*) core interfaces, or sockets, that facilitate "plug and play" operation. While these intentions are laudable, the specification falls somewhat short of delivering on that goal.

The specification is both broad and loose, allowing myriad interconnection schemes, signaling subsets, and ways of applying -- in some cases defining -- the available signals. For instance, the specification allows the registering of incoming signals at an interface before issuing a response. It also allows a combinational response to the same signals, if the designer so chooses. While this approach provides a great deal of flexibility, it greatly increases the chance that two OCP-IP compliant interfaces, designed by different engineers, will fail to talk to each other correctly.

The state machine diagrammed above is one of many interfaces created by the author; this one is an OCP Master, and is fully synchronous. The interface is capable of reads and writes to a single Slave. It's assumed that the Slave contains memory space, and that the Master will provide an incrementing address, on separate lines from the data, as it reads or writes to the Slave using a linear burst sequence. Nevertheless, the address information is redundant once the intial address has been transmitted, as long as both Master and Slave agree on the addressing sequence. Signals which are not obviously from the OCP-IP specification pertain to the user logic that interfaces with this Master state machine and are listed below:

IncAddr: Increments an address counter in the Master user logic

DecBurstCtr: Decrements a counter, in the Master user logic, tracking the burst length

Burst Signals Asserted: The author's shorthand for driving the correct values onto burst-related signals, like MBurstLength, MBurstSeq, and MBurstPrecise (although some may be omitted if Master and Slave agree to fixed values).

popFIFO: Pops write data from a FIFO in the Master's user logic.

WrFifoEmpty: Master user logic's source of write data is empty.

BurstCtr: The instantaneous value of the Master user logic Burst Counter during read or write activity.

pushFIFO: Signals a FIFO in the Master user logic to record data read from the Slave.

This design assumes that the signal SCmdAccept is generated by the slave to indicate acceptance of each write command, write address, and write data sent by the Master. The same signal indicates a read command and address has been accepted by the slave. However, the signal DataValid/Accept is an indication that the Slave has placed valid data on the data bus during a read operation.

The foregoing discussion only scratches the surface of OCP-IP's capabilities and of First Inversion's expertise in this area.