Bus Master NOC

The Bus Master NOC connected CRC engine expands on the existing CRC design. This project is to enhance the NOC controller, (Not the CRC block) to be a bus master. The bus master logic will use the NOC interface to fetch and store data to a memory. This memory is in the test bench.

The CRC needs to be updated to pass the latest test bench. (I think the XOR is finally fixed)

The bus master adds the following registers to the NOC block. These act as a slave.

Reg	Addr	Function
Chain	32'hffff_fff0	The address of a chain block. A chain block instructs the bus master how to process information through the CRC module. This register is updated as chain blocks are processed. It may be read at any time.
Start_Chain	32'hffff_fff4	Writing a non zero value to this register starts the bus master operation. The register is cleared when all chains are processed.

The bus master device assumes a 4 byte access unit (The same as the CRC block), and a 32 bit memory address. A chain block consists of the following data:

Field	Offset	Function
Link	h'0	Provides a forward pointer to the next chain block. The engine loads this value into the chain register when the current chain is completed. A value of zero 32'h0 indicates the chain processing is complete. This causes the start_chain register to be cleared.
Seed	h'4	Memory location containing the initial seed value for the CRC calculation The WAS bit will be set in the control register for a cycle while writing the seed value
Ctrl	h'8	The control register contents. The WAS bit is ignored
Poly	h'c	The initial value of the polynomial register
Data	h'10	The first data memory location. Assumed to be a multiple of 4
Len	h'14	The number of data bytes processed
Result	h'18	Results (crc_data) placed at this Address.
Message	h'1c	The message to send when this chain block is completed. (Note that a value of zero is not sent)

Each memory location is accessed by using the value in the chain register plus the offset in the table above.