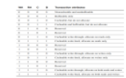
Design requirmennts					Verfication Requirments			Coverage Links				Tracability Matrix
Section	**Title**	**Description**	**Issues/Concerns/ Info**	**Setup**	**Generate**	**Check**	**Cover**	**Link**	**Type**	**Weight**	**Goal**	
1	*Asyncronus RST*	Reset the entire design	N/A	ANRST == 0 eveything else can be randomized	ANRST == 0	ARVALID, AWVALID, WVALID == 0, these signals can only be driven high if RST is high at a clock edge	Cover point SVA			1	100	
2.1	*Hand shake compliance*	When valid is high and ready is low, signals that are accicated with valid do not change	N/A	**set internal signals to cause** YXVALID = 1 and YXREADY = 0 (X can be R or W and Y can be A or null)	Random stimulus with contraint AXVALID = 1 and AXREADY = 0	Signals being associated with valid field are constant even when ready goes low	Cover point SVA			1	100	
2.2	*Hand shake values change*	valid goes low and ready is high, values can change that are planning to be sent	N/A	**set internal signals to cause** YXVALID low and YXREADY high	Random stimulus	Check that signals wither change or do not change	SVA			1	100	
3.1	*Master Write simple (1 byte 1 transfer)*	Write 1 byte to slave	N/A	**set internal signals to cause** AWLEN = 4'b0 Burst lenth is 1 AWSIZE = 4'b0 each transaction is 1 byte	Random stimulus	Check AWADDR is as expected WLAST transfer must complete before BVALID is asserted Check WDATA sent is correct Check if data sent is correct	Cover point			1	100	
3.2	*Master Read simple (1 byte, 1 transfer)*	Read 1 byte from slave	N/A	**set internal signals to cause** ARLEN = 4'b0 and ARSIZE = 0	Random stimulus	RVALID cannot be asserted until ARADDR has been trasnfered Check RDATA read is correct	Cover point			1	100	
4.1	*Master Write FULL*	write up to full AXI througput metrics	N/A	**Randomize internal signal** to cause random AWLEN and fix AWSIZE = Max value	Rand stimulus	Master writes we check BRESP is ok on each write up to the full 2048 bytes of data (max AXI throughput)	Cover point for max throughput of 2048			1	100	
4.2	*Master Read FULL*	read up to full AXI througput metrics	N/A	**Randomize internal signal** to cause random ARLEN and fix ARSIZE = Max value	Rand stimulus	Max throughput of read data is correct	Cover pointmax throughput of 2048			1	100	
5.1	Burst modes Writing	Each mode upholds its specific standards	N/A	**Randomize** the Burst signal with constraint	Rand stimulus constraint that AXBURST != 2'b11	**FIXED:** AXADARR no change if read RDATA as expected **INCR:** AXADRR increments by value I expect if read RDATA should be as expected **WRAP:** AXADDR wraps around after hitting max address range for if read RDATA should be as expected	Cover point			1	100	
5.2	*Burst modes Reading*	Each mode upholds its specific standards	N/A	**randomize** burst signal	Rand stimulus	**FIXED:** AXADARR no change if read RDATA as expected **INCR:** AXADRR increments by value I expect if read RDATA should be as expected **WRAP:** AXADDR wraps around after hitting max address range for if read RDATA should be as expected	Cover point SVA			1	100	
6	Cache Compliance	All cache values are tested and specifications are met	I do not think I can test this without an interconnect	**randomize** cache signals	Rand stimulus with constrainst shown in photo	Check every possible combination making sure that cache signals are abided by	Cover point			1	100	
7	Previlage	Check all possible combinations, privileged vs unprevliged,secure vs unsecure,data vs instruction(Still confused on what this entails)	Do not know how to test exactly	**IDK**	Rand stimulus	Not sure yet	Cover point			1	100	
8	Unalligned transfers	Start the AW or AR addr at a weird spot so not the entire data is received (example: 0x1, size is 32 bits, on the 1st transfer only 1-3 bytes are written or read and then from there each word should be fully seen	N/A	**Randomize internal signals so** AXADRR is unalligned	Rand stimulus	Make sure data read back during reads makes sense and the unalligned transfer is correct	Cover point			1	100	
9.1	***ERROR*** *Unsupported address*	Send an unsupported address to slave	N/A	Set AXADDR to a value that Is not available	Rand stimulus	DECRR signal should go high	Cover point			1	100	
9.2	***ERROR*** *FIFO overrun*	Send too much data to slave causing a SLVRRR	N/A	Send/ read more data then size of fifo	Rand stimulus	SLVRR should go high	Cover point			1	100	
10	*Exculsivity*	Make sure RMW is supported and check accordingly	N/A	Set internal signals such that AXLOCK = exclusive transfer	Rand stimulus	Preform a Exclusive write to a location and read value later to make sure the read was exclusive	Cover point			1	100	
11	*Out of order*	Send transfers with ID's and make sure OOO is working as expected	Last of my list will be amazing if I do it but everything above this is more important	Set different BID's	Rand stimulus	I think it's the same as everything above but now I need to check ID's	Cover point			1	100	
12	*Read and Write to the same address*	Send read and write at the same time to the same address	N/A	**randomize such that** ARADDR and AWADRR are equal	Rand stimulus	Depending on IP either the read or write will stall and one will occur before the other	Cover Point			1	100	



Value	Burst type	Usage notes	Length (number of transfers)	Alignment
0x00	FIXED	Reads the same address repeatedly. Useful for FIFOs.	1-16	Fixed byte lanes only defined by start address and size.
0x01	INCR	Incrementing burst. The slave increments the address for each transfer in the burst from the address for the previous transfer. The incremental value depends on the size of the transfer, as defined by the AxSIZE attribute. Useful for block transfers.	AXI3: 1-16 AXI4: 1-256	Unaligned transfers are supported.
0x10	WRAP	Wrapping burst. Similar to an incrementing burst, except that if an upper address limit is reached, the address wraps around to a lower address. Commonly used for cache line accesses.	2, 4, 8, or 16	The start address must be aligned to the transfer size.
0x11	RESERVED	Not for use.	-	-