Function to be validated	Methodology Used	Further Description	Input signals	Output Signals
APB_Controller	AEP, FXP	Explore the high level implementation of the bridge and controller		
Bridge Controller FSM	FPV	Verify the FSM states and transitions given the AMBA specifications		
Output logic based on the states	FPV	Verify that each state produces the expected output according to specs		
System functions (reset, and error handling)	FPV	Verify the behavior of the different parts of the design under the reset and error conditions.	HRESP = Error, HRESET	Observe affeced signals
Nonsequential (Read and Write) transactions	FPV	For different combinations of HSIZE (byte, halfword, and word) and HBURST (Signle transfer, Incremental, 4-beat wrapping, 8 beat wrapping), different HRESP (Okay, Error) and different HTRANS (Nonsequential, Sequential)	HADDR, HWDATA, HRESET	PRDATA, HREADY, HRESP,
Sequential (Read and Write) transactions	FPV	For different combinations of HSIZE (byte, halfword, and word) and HBURST (Signle transfer, Incremental, 4-beat wrapping, 8 beat wrapping), different HRESP (Okay, Error) and different HTRANS (Nonsequential, Sequential)	HADDR, HWDATA, HRESET	PRDATA, HREADY, HRESP,
Incremental Burst (Read and Write) transactions	FPV	For different combinations of HSIZE (byte, halfword, and word) and HBURST (Signle transfer, Incremental, 4-beat wrapping, 8 beat wrapping), different HRESP (Okay, Error) and different HTRANS (Nonsequential, Sequential)	HADDR, HWDATA, HRESET	PRDATA, HREADY, HRESP,
Wrapping Burst (Read and Write) transactions	FPV	For different combinations of HSIZE (byte, halfword, and word) and HBURST (Signle transfer, Incremental, 4-beat wrapping, 8 beat wrapping), different HRESP (Okay, Error) and different HTRANS (Nonsequential, Sequential)	HADDR, HWDATA, HRESET	PRDATA, HREADY, HRESP,

Interface to be Validated	Methodology	
AHB Slave Interface	FPV	Verify the (Haddr1, Haddr2, Hwdata1, Hwdata2) signals, the Valid logic generation, and the Tempslex Logic

APB_Controller	
Use AEP and FXP to explore the design and find any obvious bugs	Automated VC Formal Check

System functions	
If Hresetn is 0	reset everything to 0
HRESP = Error	

Read Transactions
Covered in the sequential/nonsequential transactions
HRDATA should be same as PRDATA when PENABLE(Enable cycle)
HREADYOUT and PENABLE should be high at end of transaction
PWRITE should be same as HWRITE
If HWRITE is low and HREADYIN is high next cycle should have PSEL high for next 2 clocks and should be onehot and PADDR should be same as HAADR
If HWRITE is low and HREADYIN is high then PENABLE should be high after 2 clocks
If a read transfer immediately follows a write, then 3 wait states are required to complete the read
PENABLE shouldn't be high for 2 cycles continously

Write Transactions
Covered in the sequential/nonsequential transactions
HREADYOUT and PENABLE should be high at end of transaction
PWRITE should be same as HWRITE
If HWRITE is high and HREADYIN is high, 2 cycles after PSEL should be high for next 2 clocks and should be onehot and PADDR should be same as HAADR
If HWRITE is high and HREADYIN is high then PENABLE should be high after 3 clocks
Should support back to back writes for 2 different addresses
PENABLE shouldn't be high for 2 cycles continously

Bridge Controller FSM
IDLE to IDLE
IDLE to READ
IDLE to WRITE-WAIT
READ to READ-ENABLE
READ-ENABLE to READ
READ-ENABLE to IDLE
READ-ENABLE to WRITE-WAIT
WRITE-WAIT to WRITE-P
WRITE-WAIT to WRITE
WRITE-P to WRITE-ENABLE-P
WRITE-ENABLE-P to WRITE-P
WRITE-ENABLE-P to WRITE
WRITE to WRITE-ENABLE-P
WRITE to WRITE-ENABLE
WRITE-ENABLE to WRITE-WAIT
WRITE-ENABLE to IDLE
WRITE-ENABLE to READ

Sequential/Nonsequential transactions		
Single Transfer	Read	Byte
		Half word
		Word
	Write	Byte
		Half word
		Word
4 beat wrapping burst	Read	Byte
		Half word
		Word
	Write	Byte
		Half word
		Word
8 beat wrapping burst	Read	Byte
		Half word
		Word
	Write	Byte
		Half word
		Word
Incremental Burst	Read	
		Half word
		Word
	Write	
		Half word
		Word
4 beat incrementing burst	Read	Byte
		Half word
		Word
	Write	Byte
		Half word
		Word
8 beat incrementing burst	Read	Byte
		Half word
		Word
	Write	Byte
		Half word
		Word

Incremental Bursts
Covered in the sequential/nonsequential transactions

Wrapping Bursts
Covered in the sequential/nonsequential transactions

Output logic based on the states
IDLE state with (valid &&~Hwrite)
IDLE state with (valid &&Hwrite)
WAIT state wih (not valid)
WAIT state wih (valid)
READ state
WRITE state (with ~valid)
WRITE state (with valid)
WRITEP (ABP) state
RENABLE state with (valid &&~Hwrite)
RENABLE state with (valid &&Hwrite)
RENABLE state with (not valid)
WENABLE state with (~valid && Hwritereg)
WENABLE state with (valid)
WENABLEP state with (~valid && Hwritereg)
WENABLEP state with (valid)

General Asumptions
HTRANS should be either 2'b10 or 2'b11 for SEQ, NONSEQ transfer type
when HWRITE is high next clcok should have HWDATA as a valid value
HADDR should be in the range of peripherals address
when HWRITE is high HREADYIN should be high in that cycle and in following cycle
HREADYIN is high for 2 consecutive cycles later HREADYIN shouldn't be high until HREADYOUT[=2]

General Assertions
If HADDR is not in range of peripherals, PSEL should be 0
PADDR will hold its current value until the start of the next APB transfer