Circuit Simulation Project

Abstract

---

1 Name of the Participant

Juned Pinjari

2 Affiliation / Institution

Department of Electronics and Telecommunication Engineering, Government College of Engineering, Nagpur, Nagpur, Maharashtra, India

3 Title of the Circuit

Circuit-Level Design and Verification of AMBA AHB-Lite (Single Master, Single Slave) Protocol Using eSim

4 Theory / Description

AMBA AHB-Lite is a simpler version of the AMBA AHB protocol, made for single-master systems. It keeps the pipelined address and data phase structure of full AHB, but removes arbitration, multiple masters, split transactions and burst transfers.

- This project models a single-master, single-slave AHB-Lite interface with an 8-bit address bus and 8-bit data bus. It is scaled down from the full 32-bit version so the design stays simple enough to simulate in eSim, while still showing the pipelining behaviour that makes this protocol interesting.
- The AHB Master and AHB Slave are each modelled in Verilog and converted into Ngspice-compatible XSPICE digital blocks using eSim's Makerchip-NgVeri interface.
- The two digital blocks are wired directly to each other in the eSim schematic for the internal bus signals: HADDR, HWDATA, HRDATA, HTRANS and HREADY. These signals stay in the digital domain the whole time, so they do not need ADC or DAC bridges.
- ADC bridges are used only where there is an actual analog to digital boundary, that is, where the PULSE source signals HCLK (clock) and HRESETn (reset) enter the digital domain at the Master's input.
- DAC bridges are used at a few output points so HREADY and HRDATA can be plotted as waveforms in Ngspice.

5 Reason to Simulate with eSim

- eSim's Makerchip-NgVeri flow allows combining Verilog-modelled digital logic with Ngspice simulation in one open-source tool.
- This helps verify the timing behaviour of a real bus protocol at the circuit level, not just at RTL level.

- This project also connects with VLSI coursework (RTL design and the OpenLane RTL-to-GDS flow) and an interest in open-source EDA tools.
- Developed and simulated using **eSim 2.5**.

6 Expected Outcome / Outputs

The proposed outcome was a working eSim project demonstrating a single-master, single-slave AHB-Lite transfer. All of the following were achieved and verified against the raw Ngspice transient data:

- A single-cycle write, with HADDR and HWDATA lined up correctly across the address and data phase.
- A single-cycle read, with HRDATA correctly read back during the data phase.
- HREADY-based handshaking for a normal case and for a one-wait-state case.

7 Circuit Diagram(s)

As-built eSim schematic, showing the AHB Master (U1) and AHB Slave (U2) NgVeri blocks, ADC bridges (U3, U4) converting the HCLK and HRESETn pulse sources into the digital domain, and DAC bridges (U5 to U7) converting HRDATA and HREADY back to analog waveforms for plotting. Global labels are placed on both the internal AHB digital bus (HCLK, HRESETn, HRDATA7, HRDATA3, HREADY) and on the analog stimulus/probe wires (HCLK_AIN, HRESETN_AIN, HRDATA7_AOUT, HRDATA3_AOUT, HREADY_AOUT) so every signal referenced in this report exists as a named net in the actual design.

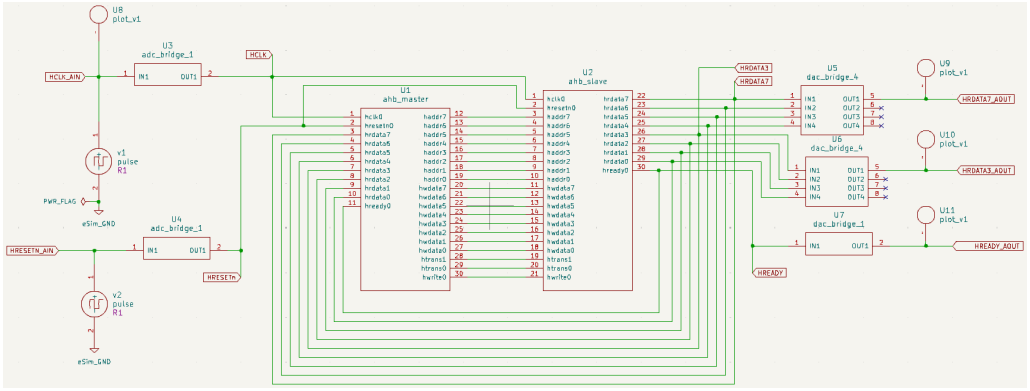

Figure 1: Final eSim schematic (`AMBA_AHB_Sim_8bit.kicad_sch`)

Component list:

Component	Value / Model
V1 (HCLK)	PULSE(0 3.3 0 2n 2n 48n 100n), 10 MHz, 3.3 V CMOS clock, about 50% duty cycle
V2 (HRESETn)	PULSE(3.3 0 0 2n 2n 196n 1000n), active-low, held low for first 200 ns
U3, U4 (`ADC_BRIDGE`)	in_low = 0.8 V, in_high = 2.0 V.

Component	Value / Model
U1 (AHB_MASTER)	8-bit NgVeri/Verilog block. Drives HADDR[7:0], HWDATA[7:0], HTRANS[1:0]. Reads HRDATA[7:0], HREADY.
U2 (AHB_SLAVE)	8-bit NgVeri/Verilog block with internal 256x8 register file, addressed by HADDR. Drives HRDATA[7:0], HREADY.
U5 to U7 (DAC_BRIDGE)	out_low = 0 V, out_high = 3.3 V. Used to plot HRDATA[7], HRDATA[3] and HREADY as waveforms.

Test case: write 0xA5 to address 0x10, then read it back from 0x10. A second write to address 0x20 is used to test a 1-cycle HREADY wait state.

8 Expected Results

(Input, output waveforms and multimeter readings.) All results below were obtained from a 1 us Ngspice transient simulation and cross-checked directly against the raw `plot_data_v.txt` output. Every expected behaviour from the proposal was reproduced, with exact timing:

Signal / Behaviour	Result
HCLK	10 MHz square wave, 0 V to 3.3 V, 100 ns period. Confirmed.
HRESETn	Steps from 0 V to 3.3 V at t = 200 ns, marking end of reset. Confirmed.
HADDR / HWDATA (write phase)	Write of 0xA5 to address 0x10 completes correctly across the pipelined address and data phase.
HRDATA (read transaction)	DAC-bridged output rises to the level for 0xA5 at t approx. 503 ns. HRDATA[7]=1, HRDATA[3]=0, matching 0xA5 = 1010 0101. Confirmed.
HREADY (normal case)	Stays at logic high (3.3 V) through the whole 0x10 write/read pair (0 to approx. 503 ns). Confirmed.
HREADY (wait-state case, 0x20)	Drops to 0 V for one extra 100 ns cycle (approx. 503 to 603 ns), then returns to 3.3 V, making that transfer one clock period longer. Confirmed.

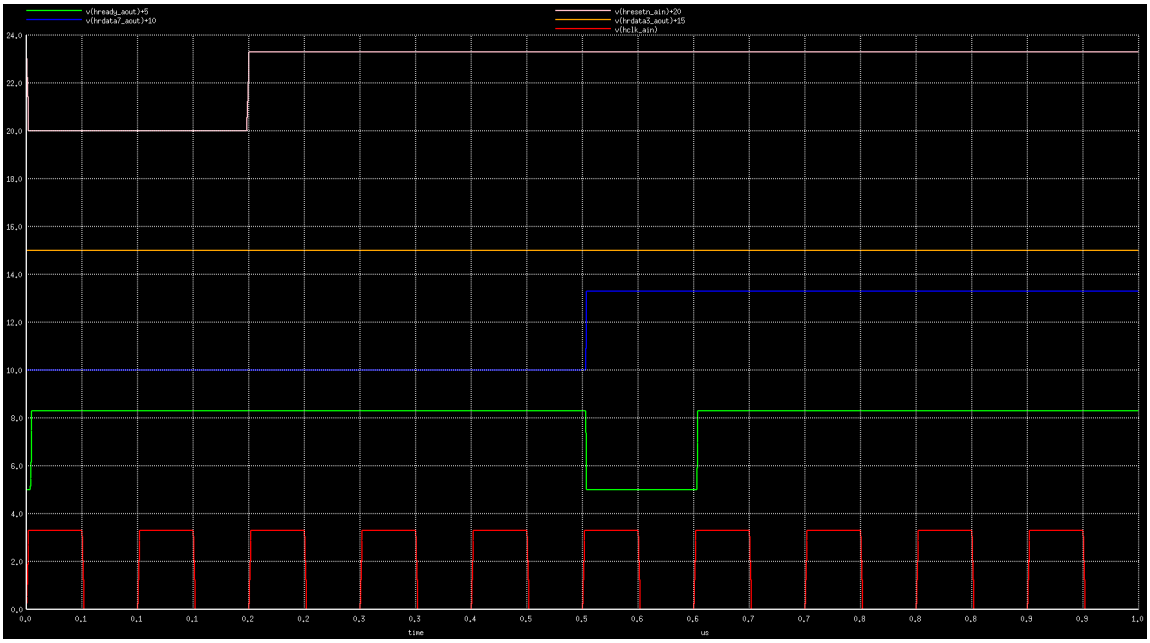

Figure 2: Ngspice composite plot. Red: HCLK. Green (+5 offset): HREADY. Blue (+10 offset): HRDATA[7]. Orange (+15 offset): HRDATA[3]. White (+20 offset): HRESETn. Traces are vertically offset for visual clarity only; actual logic levels are 0 V / 3.3 V in every trace.

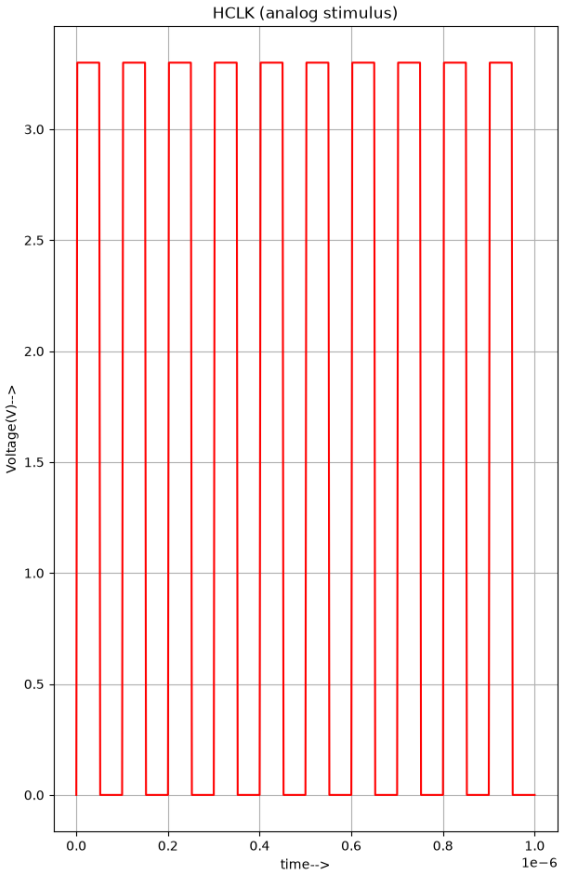

Figure 3: HCLK, 10 MHz, 0 to 3.3 V.

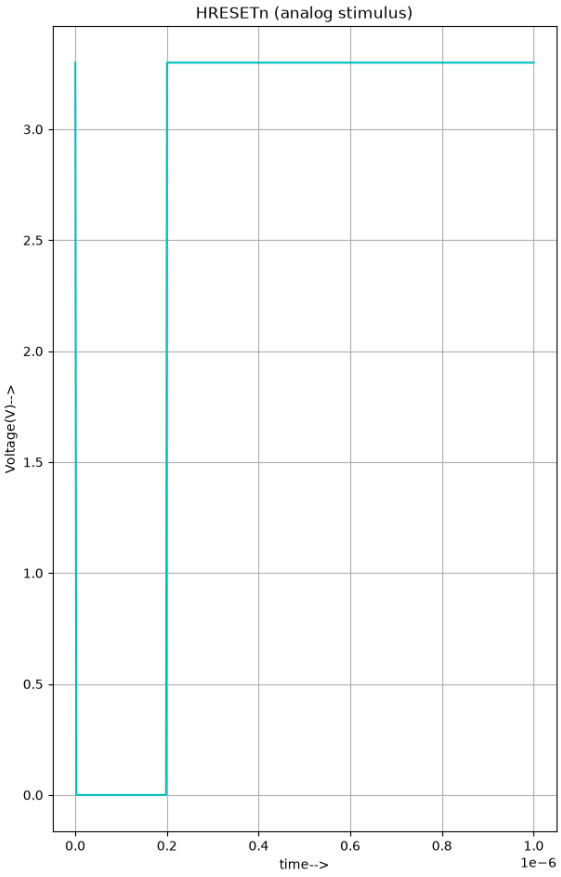

Figure 4: HRESETn, released at t = 200 ns.

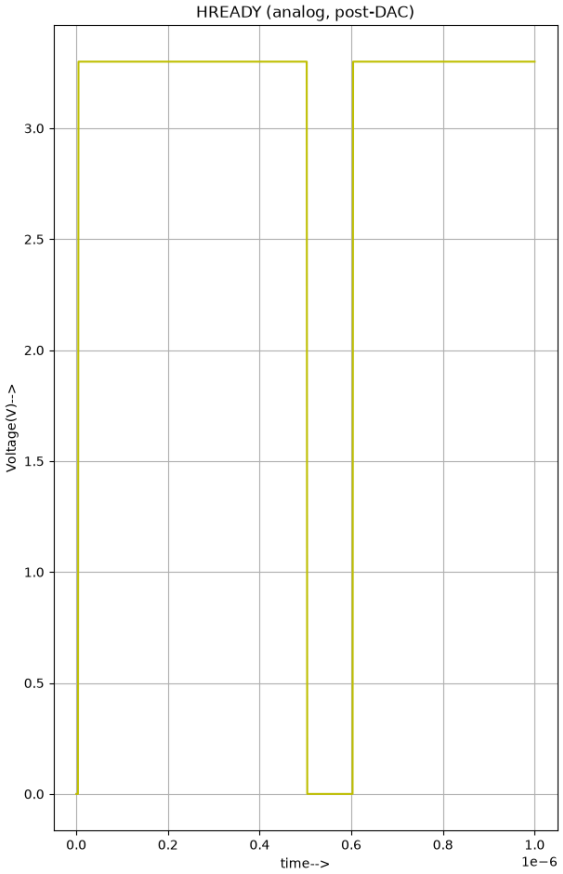

Figure 5: HREADY, high through the write/read pair, one-cycle wait-state dip at 0x20.

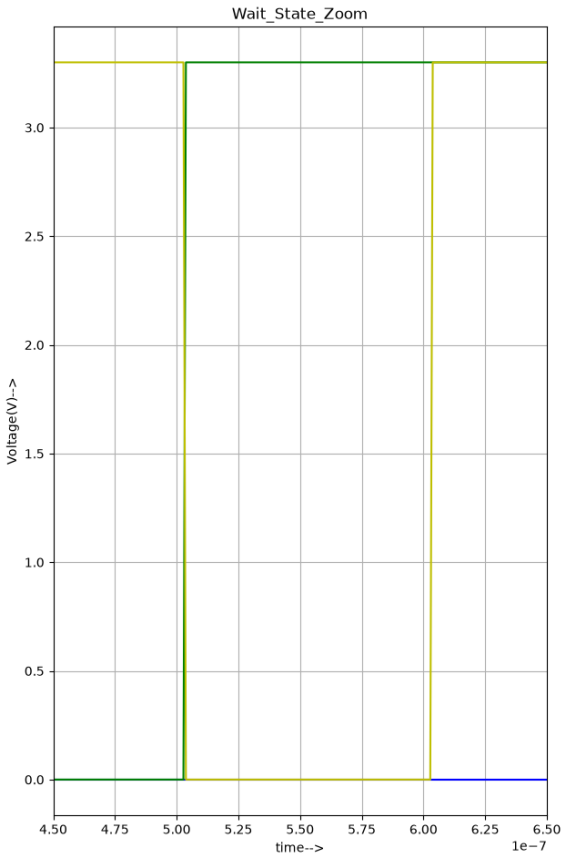

Figure 6: Zoomed view (0.45 to 0.65 us) isolating the wait state.

9 Research Paper / Journal

Title: AHB Design and Verification AMBA 2.0 using System Verilog
Author: Divya M, Dr. K. A. Radhakrishna Rao
Page No.: N/A (online publication)
Link: https://www.ijariit.com/manuscript/ahb-design-and-verification-amba-2-0-using-system-verilog/

10 Source / References

- ARM AMBA Specification: https://developer.arm.com/documentation/ihi0011/latest/
- Wikipedia, Advanced Microcontroller Bus Architecture: https://en.wikipedia.org/wiki/Advanced_Microcontroller_Bus_Architecture