	Project Name	Poorman's ALU (PALU)	**Test Design By**	Ryan Woodward
	Module Name	Milestone 5	**Test Design Date**	04-14-2023
	Release Version	4.0	**Test Executor**	Ryan Woodward
			Test Execution Date	04-14-2023
Pre-Condition	Development Environments and GCC Compiler Functioning, ModelSim			
Dependencies	N/A			
Test Priority	High			

Test Case #	Test Title	Test Summary	Test Steps	Test Data	Results
1	C Program GPIO functional test	Testing the C program to see if the code that interacts with the GPIO pins works with a test circuit	- Run the programmer in Quartus of the "testPALU" project - Run the C program "poormansALU_ms4' from Putty Terminal - Enter two values as prompted by the program - The system should display the values entered on the hex display	Any two single digit numbers	SUCCESS The values entered into the terminal are properly displayed on the HEX segment. This test was successful when using the DE10 Computer image from intel.
2	C Program GPIO functional test (w/ PALU circuit)	Testing the C Program to see if the the C code properly interacts with the circuit (of the PALU project) on the FPGA	- Run the programmer in Quartus of the "testPALU" project - Run the C program "poormansALU_ms4' from Putty Terminal - Enter two values as prompted by the program	PALU truth table (pg 13 of MS 4 document)	SUCCESS The values entered in the C program properly display in the 7-segment display when using

		through the GPIO pins	- The system should display the values entered on the hex display		the DE10-std Computer Image

3	Functional Simulation test of the 'poormansALU' circuit	Running simulations (in Modelsim) to verify the circuits' operation. This circuit is not the completed project but the integration of the three primary circuit to create an ALU (mux, bool logic unit, arithmetic unit)	- Open the poormansALU project
			- Create new University Waveform File
			- Set the parameters:
			  - 15us length with 1us intervals
			  - Add the inputs and output nodes
			- Run the simulation
			- Verify the results of the simulation match the truth table
				poormansALU truth table (pg 13 of MS 4 document)	SUCCESS
					The results of the simulation match the truth table

4	Hardware Test of the poormansALU circuit	This test involves testing the poormansALU circuit on the DE10-standard board itself and verifying the result from the simulation and metrics of the truth table	- Open the poormansALU project
			- Open the Assignment tab - pin planner
			- Set the inputs to the switches
			  - 8 for t 4-bit numbers
			  - 2 for selectors
			  - MUX
			  - Arith unit (+/-)
			- Set outputs to the LEDs (x5)
			  - 1 4-bit output
			  - 1 carry/borrow out
			- Compile and Synthesize the project
			- Open the programmer and ad the SoC device
			- Run the programmer
			- Verify the results with the truth table
				poormansALU truth table (pg 13 of MS 4 document)	SUCCESS
					The LEDs correspond to the truth tables outputs. The switches properly select the arithmetic operation and which result is passed through the MUX

5	Functional Simulation test of the 'PoormansALU_project'	This test involves running functional simulations of the PoormansALU_project circuit. This circuit	- Open the poormansALU project
			- Create new University Waveform File
			- Set the parameters:
			  - 15us length with 1us intervals
			  - Add the inputs and output nodes
				PoormansALU_Project truth table (pg 15 of MS4 document)	SUCCESS
					The outputs (of the seven segment) match the truth table

	circuit	is the complete project circuit that includes a poormansALU circuit and a BCD decoder circuit	- Run the simulation - Verify the results of the simulation match the truth table		

6	Hardware Test of the PoormansALU _project circuit	This test will be done to verify the circuits functionality on the DE10 standard board	- Open the poormansALU project
			- Open the Assignment tab - pin planner
			- Set the inputs to the switches
			  - 8 for t 4-bit numbers
			  - 2 for selectors
			  - MUX
			  - Arith unit (+/-)
			- Set outputs to an LED and the 2 he displays
			- Compile and Synthesize the project
			- Open the programmer and ad the SoC device
			- Run the programmer
			- Verify the results with the truth table
				PoormansALU_ Project truth table (pg 15 of MS4 document)	SUCCESS
					The outputs match those of the entered values and the selector switches properly determine operations and output from the MUX

7	Individual Verilog Circuit Tests	This test will be done to verify the results of the circuits abide the truth tables	- Similar to hardware tests and functional simulations of VHDL circuits - Abide those steps but for the poormansALU_v project	PALU truth tables MS5 document	SUCCESS Each of the outputs for the MUX, Boolean logic unit, and the arithmetic unit match those of the truth tables
8	Verilog Poormans ALU Full integrated project	This test will be done to verify the Verilog circuits can be run with the C application	- Follow the same steps as above for the pull project test in VHDL tests **2 & 3**	PALU truth table in MS5 document under PoormansALU_p roject circuit	SUCCESS Each of the tests produced the proper output for addition,subtraction, and MUX

9	ARM Assembly test	Testing the arm assembly code in the VHDL poormans ALU project	- 'Make' the poormansALU_ms5 file - Copy the binary file to the board using file zilla - Run the quartus project using the programmer - Execute the binary file in Putty - Enter two values to verify the operations	PALU truth table in MS5 document	SUCCESS Each of the tests produced the proper output for addition,subtraction, and MUX
10	IP Block Project integration test	Testing the whole PALU project with an IP Block based MUX rather than my own coded MUX	- Run the PALU project in the board using the quartus programmer - Execute the project binary in Putty - Enter two 4-bit values - Verify the results	PALU truth table in MS5 document	SUCCESS Each of the tests produced the proper output for addition,subtraction, and MUX