Learning Experience C.3 Project Report
ECE 2544

Objective and Requirements

The objective of Learning Experience C was to create two key components of a simple computer: an 8-bit function unit and a register transfer system, where the function unit is capable of implementing 13 distinct operations on two 8-bit operands as well as outputting four status bits, and there are four 8-bit registers capable of storing data from switches, storing data from the function unit, and sending data to the function unit. The function unit had to accept a four bit function select code to choose the operation, each operation had to have a unique code, the entire circuit had to be designed and simulated in Quartus Prime Lite Edition using structural and/or dataflow Verilog, and the circuit had to be designed to minimize FPGA resource usage. In addition to designing the circuit, part 3 of Learning Experience C, also referred to throughout this document as LE C.3, required the creation of a network diagram for the main deliverables of the assignment. This network diagram can be found in Diagram 1, where the main deliverables are colored in yellow.

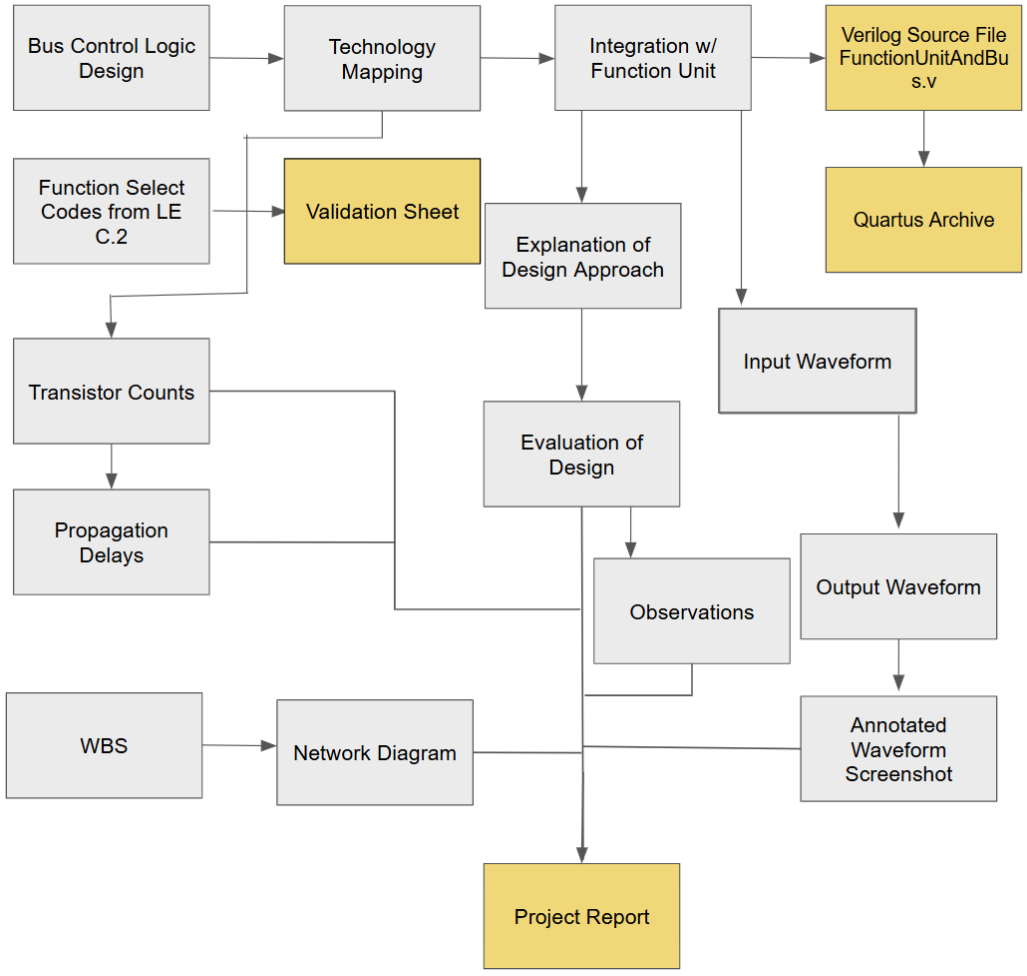

Diagram 1: Learning Experience C.3 Network Diagram

Circuit Design

The same function unit from LE C.2 was used in this learning experience. The function unit was designed with a combination of ripple carry adders, muxes, decoders, and XOR gates. The function unit's 13 operations were divided into three separate units, involving an arithmetic unit, a logic unit, and a shifting unit. Given the operands and the function select code, each functional block computed an 8-bit output, though only one of the functional blocks produced an output corresponding to the actual operation. Each bit of the three outputs was sent to a 3 x 1 mux along with the first three bits of the function select code to choose which bit would be passed through to the final output. Three bits were used instead of two since both the arithmetic block and logic block contained more than 4 operations. If only two bits from the function select code were preserved to indicate the proper functional block, then each operation would only be distinguished by the remaining two bits, which would only allow each block to contain four operations. By using three bits, the part of the function select code that corresponded to the functional block could overlap with the remaining code, allowing for 13 unique operation codes and the ability to select between 3 distinct functional blocks. To simplify the control logic, codes corresponding to the arithmetic operations ended in a 1 if the carry-in bit of the operation needed to be a 1, otherwise they ended in a 0. To produce the status bits, the N bit was set to equal the most significant bit of the output, the Z bit was equivalent to putting every output bit through one large NOR gate, and the arithmetic block was modified to output the carry out and carry in bits of the most significant bit to get the values for V and C. C was set equal to the value of the carry out bit, and a XOR gate was used between the carry out and carry in bit to get the value of V. The register transfer system was designed such that a 2 x 4 decoder was used to control the load signal for each register, using input switches SW[1] and SW[0] to determine which register would receive the load signal. Each bit of each register was sent to a 4 x 1 mux along with the corresponding switch values to determine which register's value would get sent to the operandA bus, and a second set of muxes was used in the same way to determine which register's value would get sent to the operandB bus. The two buses were used as inputs to the function unit. The function unit then sent its output to a 2 x 1 mux along with the switch values SW[9:2], and which one was sent to the registers to be loaded in was based on which button was being pressed. All circuits and functional units were technology mapped wherever possible to minimize transistor usage, replacing most AND/OR logic with NAND gates. Circuits that relied primarily on AND gates and inverters, such as most of the decoders, were not technology mapped since there would have been no reduction in transistor usage. The transistor counts for the function unit and its functional blocks can be found in Tables 1.1 - 1.10, the transistor counts for the register transfer system can be found in Tables 1.11 - 1.17, and the total transistor count for LE C.3 can be found in Table 1.18. Note that the 8-bit 2x1 mux for the circuit was assumed to be implemented using just AND and OR gates, since no inverters were seen in its implementation in the Verilog code. The propagation delays for the critical path of the different functional blocks and the function unit can be found in Tables 2.1 - 2.5, the propagation delays for the critical path of the register transfer system can be found in Tables 2.6 - 2.9, and the propagation delay for the critical path of the whole circuit can be found in Table 2.10. Due to the looping nature of the circuit, the critical path for this circuit was defined as the path from an input

control signal to the 2x4 loading decoder of the register transfer system all the way to the output of the function unit.

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
inverter	2	4	8
4-input AND	10	5	50
4-input NOR	8	1	**8**
			66

Table 1.1: Transistors for the 4x6 decoder for the arithmetic functional block

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
inverter	2	4	8
4-input AND	10	5	50
			58

Table 1.2: Transistors for the 4x5 decoder for the logic functional block

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
inverter	2	2	4
4-input AND	10	2	20
			24

Table 1.3: Transistors for the 4x2 decoder for the shifting functional block

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
inverter	2	3	6
2-input NAND	4	3	12
3-input NAND	6	5	30
4-input NAND	8	1	8
			56

Table 1.4: Transistors for the full adder

Functional Block	Transistors in FB	Number of FBs	Total Transistors
Full Adder	56	8	448
			448

Table 1.5: Transistors for the ripple carry adder

Functional Block	Transistors in FB	Number of FBs	Total Transistors
4 x 6 Decoder	66	1	66
3 x 1 Mux	38	15	570
4 x 1 Mux	38	1	38
Ripple Carry Adder	448	1	448
			1122

Table 1.6: Transistors for the arithmetic functional block

Functional Block	Transistors in FB	Number of FBs	Total Transistors
5 x 1 mux	30	8	240
5 x 1 mux's inputs	16	8	128
4 x 5 decoder	58	1	58
			426

Table 1.7: Transistors for the logical functional block

Functional Block	Transistors in FB	Number of FBs	Total Transistors
4 x 2 decoder	24	1	24
2 x 1 mux	12	8	96
Ripple Carry Adder	448	1	448
XOR gates	96	1	96
			664

Table 1.8: Transistors for the shifting functional block

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
2-input NAND	4	3	12
3-input NAND	6	1	6
			18

Table 1.9: Transistors for the 3x1 mux for the function unit

Gate/Block Type	Transistors in Gate	Number of Gates	Total Transistors
8-input NOR	16	1	16
2-input XOR	12	1	12
3 x 1 mux	18	2	36
			64

Table 1.10: Transistors for the status bit logic for the function unit

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
inverter	2	3	6
2-input NAND	4	4	16
3-input NAND	6	2	12
2-input AND	6	1	6
			40

Table 1.11: Transistors for the 3x3 decoder for the function unit

Gate / Block Type	Transistors in Gate/Block	Number of Gates/Blocks	Total Transistors
Arithmetic FB	1122	1	1122
Logical FB	426	1	426
Shifting FB	664	1	664
3 x 1 mux	18	8	144
3 x 3 decoder	40	1	40
Status Bit Logic	64	1	64
			2460

Table 1.12: Transistors for the function unit

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
2-input AND	6	16	96
2-input OR	6	8	48
			144

Table 1.13: Transistors for the 8-bit 2x1 mux for the register transfer system

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
inverter	2	2	4
2-input OR	6	1	6
3-input NAND	6	4	24
			34

Table 1.14: Transistors for the 2x4 decoder for the load signal of the register transfer system

Gate Type	Transistors in Gate	Number of Gates	Total Transistors
inverter	2	2	4
2-input NAND	4	4	16
			20

Table 1.15: Transistors for the 2x4 decoder for the 4x1 mux of the register transfer system

Gate/Block Type	Transistors in Gate/Block	Number of Gates/Blocks	Total Transistors
2-input NAND	4	4	16
4-input NAND	8	1	8
2x4 Decoder	20	1	20
			44

Table 1.16: Transistors for the 4x1 mux for the register transfer system

Gate / Block Type	Transistors in Gate/Block	Number of Gates/Blocks	Total Transistors
8-bit 2x1 mux	144	1	144
2x4 Load Decoder	34	1	34
4x1 mux	44	16	704
			882

Table 1.17: Total transistor count for the register transfer system

Block/Circuit Type	Transistors in Block	Number of Blocks	Total Transistors
Register Transfer System	882	1	882
Function Unit	2460	1	2460
			3342

Table 1.18: Total transistor count for the LE C.3 circuit

Gate Type	Propagation Delay (ps)	Number of Gates in Critical Path	Total Propagation Delay (ps)
inverter	25	1	25
4-input AND	125	1	125
4-input NOR	100	1	100
			250

Table 2.1: Propagation delay for the 4x6 decoder of the arithmetic functional block

Gate Type	Propagation Delay (ps)	Number of Gates in Critical Path	Total Propagation Delay (ps)
inverter	25	1	25
4-input AND	125	1	125
5 x 1 mux	175	1	175
			325

Table 2.2: Propagation delay for an opcode bit to the output for the logical functional block

Functional Block Type	Propagation Delay (ps)	Number of FBs in Critical Path	Total Propagation Delay (ps)
4 x 6 Decoder	250	1	250
3 x 1 Mux	225	1	225
Ripple Carry Adder	1075	1	1075
			1550

Table 2.3: Propagation delay for the arithmetic functional block

Gate Type	Propagation Delay (ps)	Number of Gates in Critical Path	Total Propagation Delay (ps)
2-input NAND	50	1	50
3-input NAND	75	1	75
			125

Table 2.4: Propagation delay for the 3 x 1 mux used for the function unit

Functional Block Type	Propagation Delay (ps)	Number of FBs in Critical Path	Total Propagation Delay (ps)
Arithmetic FB	1550	1	1550
3 x 1 Mux	125	1	125
			1675

Table 2.5: Propagation delay for the function unit

Gate Type	Propagation Delay (ps)	Number of Gates in Critical Path	Total Propagation Delay (ps)
inverter	25	1	25
3-input NAND	75	1	75
			100

Table 2.6: Propagation delay for the 2x4 load decoder of the register transfer system

Gate Type	Propagation Delay (ps)	Number of Gates in Critical Path	Total Propagation Delay (ps)
inverter	25	1	25
2-input NAND	50	1	50
			75

Table 2.7: Propagation delay for the mux decoder of the register transfer system

Block / Gate Type	Propagation Delay (ps)	Number of Gates/Blocks in Critical Path	Total Propagation Delay (ps)
2x4 Decoder	75	1	75
2-input NAND	50	1	50
4-input NAND	100	1	100
			225

Table 2.8: Propagation delay for the 4x1 mux of the register transfer system

Block Type	Propagation Delay (ps)	Number of Blocks in Critical Path	Total Propagation Delay (ps)
2x4 Load Decoder	100	1	100
4x1 Mux	225	1	225
			325

Table 2.9: Propagation delay for the register transfer system

Circuit Type	Propagation Delay (ps)	Number of Circuits in Critical Path	Total Propagation Delay (ps)
Function Unit	1675	1	1675
Register Transfer System	325	1	325
			2000

Table 2.10: Propagation delay for the entire circuit (2x4 decoder input → function unit output)

Simulation Results

This circuit was simulated using the Quartus Prime Lite Edition software. The results from the simulation and testing can be found in Diagram 2. The simulation loads each register by setting KEY[0] to low, indicated by the text LOAD or LD, then proceeds to execute each operation the function unit is capable of and store it in one of the registers, indicated by setting KEY[1] to low. Note that two non-adjacent columns of the waveform having the same color does not indicate any special relation between them, as colors had to be reused due to the significant number of columns.

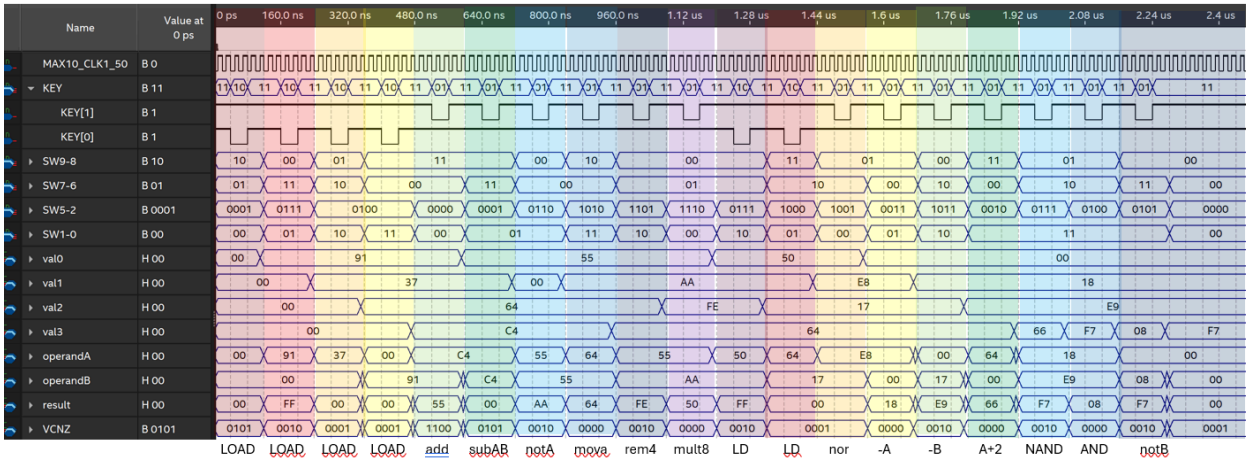

Diagram 2: The annotated waveform with the inputs and corresponding operations

Conclusion

Project LE C required the design of a function unit and a register transfer system that were combined to execute and store the results of various operations. The assignment required that the function unit could implement 13 distinct operations, the function unit had to accept two 8-bit operands and output an 8-bit result along with four status bits, the register transfer system had four registers that each stored 8 bits, any two of the registers could send their data to the function unit, and any of the registers could receive either the output of the function unit or data from the switches. The assignment also required that the circuit was verified through a

simulation in the Quartus Prime Lite Edition software. The circuit was then implemented on the DE10-Lite Board to test it physically, and was found to function as expected.