ECE 111 Final Project

Amaan Singh
Peter Wang

SHA-256 Description

- SHA-256 is a encryption algorithm that takes up to 2^64 bit of “message” and generates a 256-bit output.
- Breaks the “message” into block size of 512 bit (16 words). The last block will contain the length of the message in the last 64 bits.
- Each message will generate an almost guaranteed unique hash code output.

Design objectives and constraints

Objectives:

- Successful implementation of SHA-256 with block size of 16 word.
- Optimization for lower area delay product.

Constrains:

- Each block only consists 16 words, however, the compression process takes 1 word for each cycle and each block takes 64 cycles.
- Potential sizing constraints due to the FPGA technology.

Design Challenges

- Memory access (Reading memory address offset)
- Picking structure for the FSM implementation (single always block vs. separate always_ff and always_comb).
- Calculating the number of blocked needed
- Reducing the delay while maintaining a reasonable resource utilization.
- Padding the message correctly and saving the message length.

Optimization

- Implementation using single always_ff block.
- Reallocate existing registers to reduce area.
- Reduce the number of registers needed as a whole.
- Parallelism during the BLOCK state to reduce the number of cycles.

Result:SHA-256

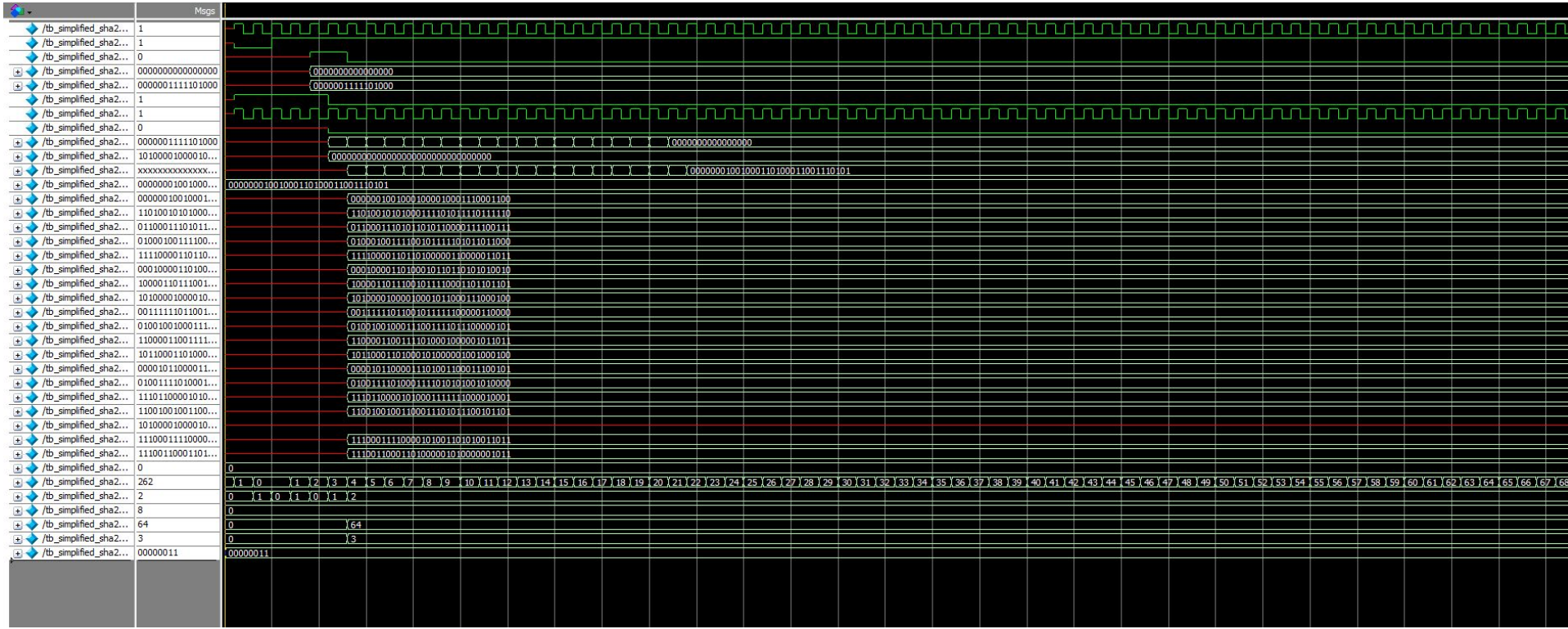
	Fmax	Restricted Fmax	Clock Name	Note
1	100.18 MHz	100.18 MHz	clk	

- Waveform, resouce utilization, and Area*Delay

<<Filter>>

Fitter Status	Successful - Fri Dec 15 14:34:47 2023
Quartus Prime Version	22.1std.2 Build 922 07/20/2023 SC Lite Edition
Revision Name	1004
Top-level Entity Name	simplified_sha256
Family	Arria II GX
Device	EP2AGX45DF29I5
Timing Models	Final
Logic utilization	23 %
Total registers	2672
Total pins	118 / 404 (29 %)
Total virtual pins	0
Total block memory bits	0 / 2,939,904 (0 %)
DSP block 18-bit elements	0 / 232 (0 %)
Total GXB Receiver Channel PCS	0 / 8 (0 %)
Total GXB Receiver Channel PMA	0 / 8 (0 %)
Total GXB Transmitter Channel PCS	0 / 8 (0 %)
Total GXB Transmitter Channel PMA	0 / 8 (0 %)
Total PLLs	0 / 4 (0 %)
Total DLLs	0 / 2 (0 %)

	Resource	Usage
1	Estimated ALUTs Used	3684
1	-- Combinational ALUTs	3684
2	-- Memory ALUTs	0
3	-- LUT_REGs	0
2	Dedicated logic registers	2672
3		
4	Estimated ALUTs Unavailable	2153
1	-- Due to unpartnered combinational logic	2153
2	-- Due to Memory ALUTs	0
5		
6	Total combinational functions	3684
7	Combinational ALUT usage by number of inputs	
1	-- 7 input functions	2
2	-- 6 input functions	2151
3	-- 5 input functions	168
4	-- 4 input functions	161
5	-- <=3 input functions	1202
8		
9	Combinational ALUTs by mode	
1	-- normal mode	3142
2	-- extended LUT mode	2
3	-- arithmetic mode	396
4	-- shared arithmetic mode	144
10		
11	Estimated ALUT/register pairs used	7802
12		
13	Total registers	2672
14		
15		
16	I/O pins	118
17		
18	DSP block 18-bit elements	0
19		
20	Maximum fan-out node	sta...UTE
21	Maximum fan-out	2812
22	Total fan-out	32711
23	Average fan-out	4.96


Msgs
/tb_simplified_sha2...
1
1
0
0000000000000000
0000001111101000
1
1
0
0000000111101000
0000000000000000
10100001000010...
0000000000000000000000000000000000
xxxxxxxxxxxxxxx...
0000000100100011...
00000001001000110100011001110101
0000001001000100...
1101001010100001...
0110001110101101...
0100010011110010...
1110000110110100...
0001000011010010...
1000011011100111...
1010000100001000...
0011111101100111...
0100100100011100...
1100001100111101...
1011000110100010...
0000101100001110...
0100111101000111...
1110110000101000...
1100100100110001...
1010000100001000...
1110001111000010...
1110011000110100...
0
262
2
8
64
3
00000011

Bitcoin Description

- A simplified version of bitcoin blockchain hashing with 16 nonce (unique seeds).
- Hashing implemented using previously built SHA-256

Design objectives and constraints

Objectives:

- Improvise area delay product for both serial/parallel implementation.
- Efficient usage of limited FPGA resource

Constraints:

- Physical area limitation of FPGA
- Optimization for delay and area

Design Challenges

- Reducing area consumption when using the parallel implementation
- Not a

Optimization/Parallelization

- Serial implementation.
- Greatly educed the number of cycles by hashing the first 16 words once and the last 4 words 16 times.
- Restructuring the SHA-256 for less area consumption.

Result:Bitcoin

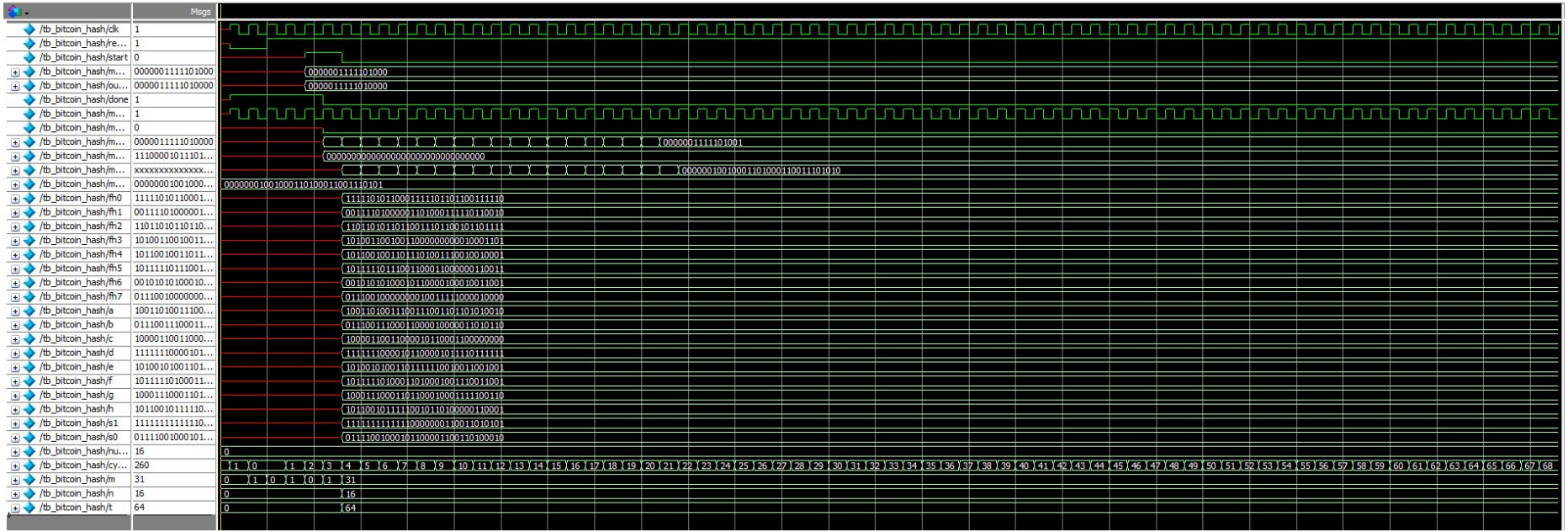
- Waveform, resouce utilization, and Area*Delay

	Compilation Hierarchy Node	Combinational ALUTs	Dedicated Logic Registers	Block Memory Bits	DSP Elements	DSP 9x9	DSP 12x12	DSP 18x18	DSP 36x36	Pins	Virtual Pins
1	bitcoin_hash	6153 (6153)	2732 (2732)	0	0	0	0	0	0	118	0

Fitter Status	Successful - Sat Dec 16 23:54:49 2023
Quartus Prime Version	22.1std.2 Build 922 07/20/2023 SC Lite Edition
Revision Name	1004
Top-level Entity Name	bitcoin_hash
Family	Arria II GX
Device	EP2AGX45DF29I5
Timing Models	Final
Logic utilization	29 %
Total registers	2732
Total pins	118 / 404 (29 %)
Total virtual pins	0
Total block memory bits	0 / 2,939,904 (0 %)
DSP block 18-bit elements	0 / 232 (0 %)
Total GXB Receiver Channel PCS	0 / 8 (0 %)
Total GXB Receiver Channel PMA	0 / 8 (0 %)
Total GXB Transmitter Channel PCS	0 / 8 (0 %)
Total GXB Transmitter Channel PMA	0 / 8 (0 %)
Total PLLs	0 / 4 (0 %)
Total DLLs	0 / 2 (0 %)

	Fmax	Restricted Fmax	Clock Name	Note
1	100.18 MHz	100.18 MHz	clk	

	Resource	Usage
1	Estimated ALUTs Used	6153
1	-- Combinational ALUTs	6153
2	-- Memory ALUTs	0
3	-- LUT_REGs	0
2	Dedicated logic registers	2732
3		
4	Estimated ALUTs Unavailable	2138
1	-- Due to unpartnered combinational logic	2138
2	-- Due to Memory ALUTs	0
5		
6	Total combinational functions	6153
7	Combinational ALUT usage by number of inputs	
1	-- 7 input functions	4
2	-- 6 input functions	2134
3	-- 5 input functions	2272
4	-- 4 input functions	189
5	-- <=3 input functions	1554
8		
9	Combinational ALUTs by mode	
1	-- normal mode	5530
2	-- extended LUT mode	4
3	-- arithmetic mode	475
4	-- shared arithmetic mode	144
10		
11	Estimated ALUT/register pairs used	8444
12		
13	Total registers	2732
14		
15		
16	I/O pins	118
17		
18	DSP block 18-bit elements	0
19		
20	Maximum fan-out node	clk...put
21	Maximum fan-out	2733
22	Total fan-out	44360
23	Average fan-out	4.86


Msgs
/tb_bitcoin_hash/clk
/tb_bitcoin_hash/re...
/tb_bitcoin_hash/start
/tb_bitcoin_hash/m...
/tb_bitcoin_hash/ou...
/tb_bitcoin_hash/done
/tb_bitcoin_hash/m...
/tb_bitcoin_hash/m...
/tb_bitcoin_hash/m...
/tb_bitcoin_hash/m...
/tb_bitcoin_hash/m...
/tb_bitcoin_hash/m...
/tb_bitcoin_hash/fh0
/tb_bitcoin_hash/fh1
/tb_bitcoin_hash/fh2
/tb_bitcoin_hash/fh3
/tb_bitcoin_hash/fh4
/tb_bitcoin_hash/fh5
/tb_bitcoin_hash/fh6
/tb_bitcoin_hash/fh7
/tb_bitcoin_hash/a
/tb_bitcoin_hash/b
/tb_bitcoin_hash/c
/tb_bitcoin_hash/d
/tb_bitcoin_hash/e
/tb_bitcoin_hash/f
/tb_bitcoin_hash/g
/tb_bitcoin_hash/h
/tb_bitcoin_hash/s1
/tb_bitcoin_hash/s0
/tb_bitcoin_hash/nu...
/tb_bitcoin_hash/cy...
/tb_bitcoin_hash/m
/tb_bitcoin_hash/n
/tb_bitcoin_hash/t
1
1
0
0000001111101000
0000011111010000
1
1
0
0000001111101001
00000000000000000000000000000000
0000001001000110100011001110101
0000000100100011010001100111010101
111110101100011111011011001111110
001111010000011010001111101100010
110110101101100111011001011011111
101001100100110000000000100011101
101100100110111010011100100100001
101111011100110001100000011001 1
001010101000101100001000100111001
011100100000000100111110000100000
100110100111001110011011010100010
011100111000110000100000011010110
100001100110000101100011000000000
111111100000101100001011110111111
101001010011011111100100110011001
101111010001101000100111001 1001
100011100011011000100011111001110
101100101111100101101000001 0001
111111111111000000001100110101011
011100100010110000110011010010
0
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
0
1
0
1
0
1
31
0
16
0
64

Conclusion

- In this project, we learned about the SHA-256 encryption and its applications in the form of bitcoin chain block. We begin by implementing the 16 work version of SHA-256 (originally 64 word) using word expansion function. After throughly testing the SHA-256 and making sure it works for various message length, we combined 16 SHA-256 block to implement a 16 nonce bit coin block chain. The block chain generation can either be done by serial or parallel implementation. Both implementation have its unique advantages as trade off between area and delay/number of cycles. Thus this project provides an exposure of the real life design problem where trade-offs are made to satisfy various design requirements.