TECHNISCHE UNIVERSITÄT WIEN

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

Institut für Computertechnik
Institute of Computer Technology

Final Presentation - P4 Accuracy of Approximate Circuits

FABIAN GARBER, SIMON HOWIND, KAGAN ÖZTEN, MARTIN RESETARITS, PETER TRAUNMÜLLER

Table of Content

Goals

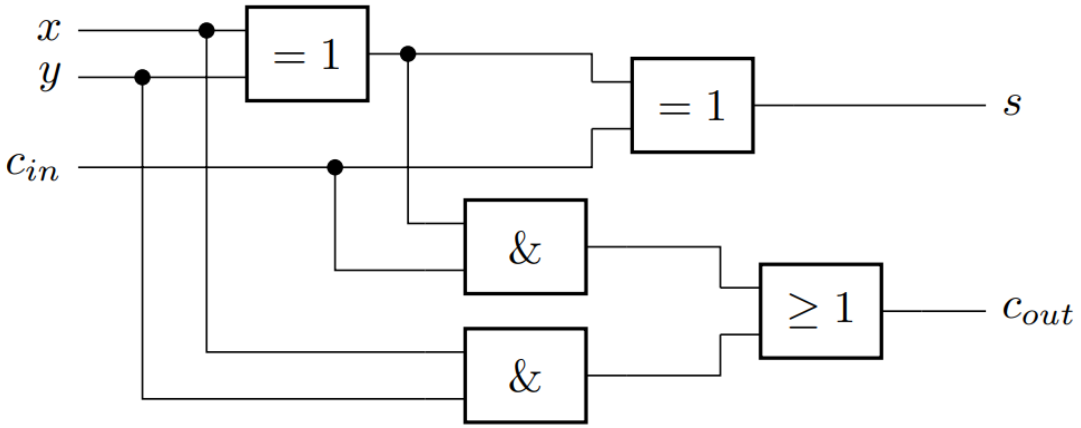
- Comparison of exact and approximate Full-Adder
- Power Consumption & Area evaluation
- Systematic error analysis
- VHDL implementation

Goals

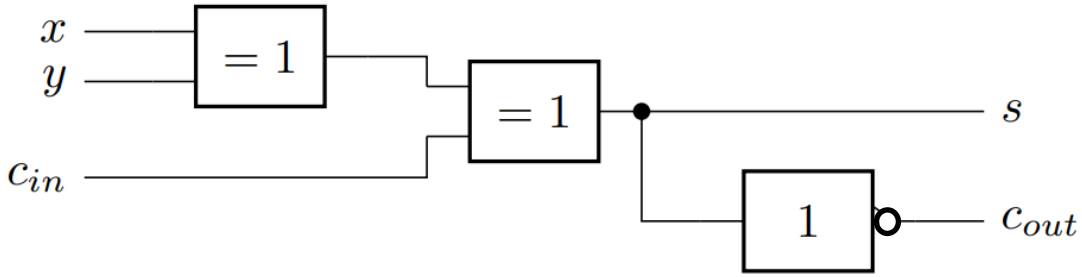
- Comparison of exact and approximate Full-Adder
- Power Consumption & Area evaluation
- Systematic error analysis
- VHDL implementation

Methods

Methods – Design Comparison



Conventional Full Adder Circuit Design



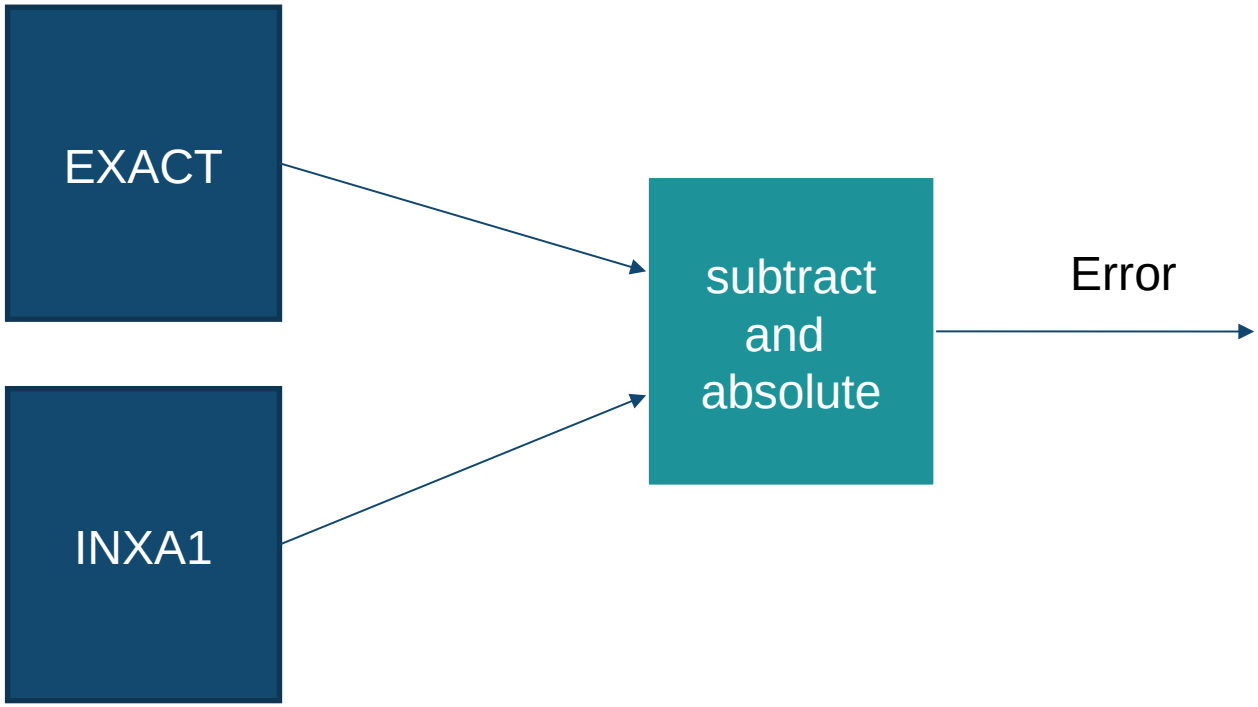
Approximate Full Adder Circuit Design as proposed by Priyadharshni et al

Methods – Result Comparison

Input			Full Adder		INXA1	
x	y	c_{in}	c_{out}	s	c_{out}	s
0	0	0	0	0	0 ✓	0 ✓
0	0	1	0	1	1 ✗	1 ✓
0	1	0	0	1	0 ✓	1 ✓
0	1	1	1	0	1 ✓	0 ✓
1	0	0	0	1	0 ✓	1 ✓
1	0	1	1	0	1 ✓	0 ✓
1	1	0	1	0	0 ✗	0 ✓
1	1	1	1	1	1 ✓	1 ✓

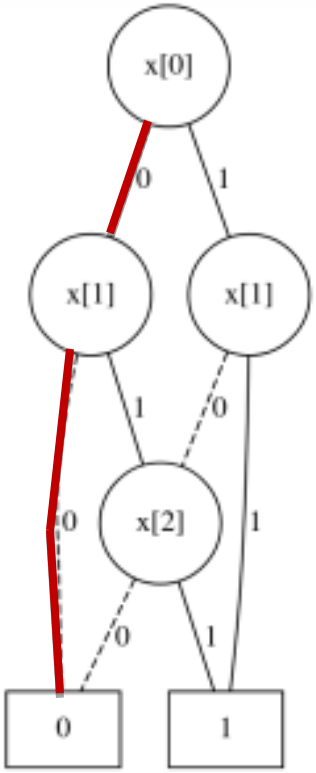
Methods - Error Analysis

- For multi bit inputs, the error is not Hamming distance
- Error must be interpreted as number

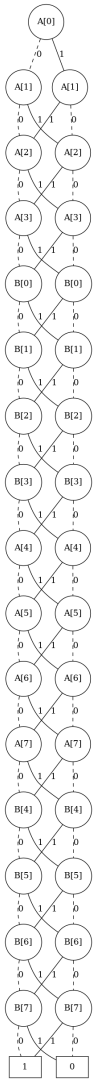


Methods - Binary Decision Diagram

Input			Output
X[0]	X[1]	X[2]	out
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Methods – BDD Complexity

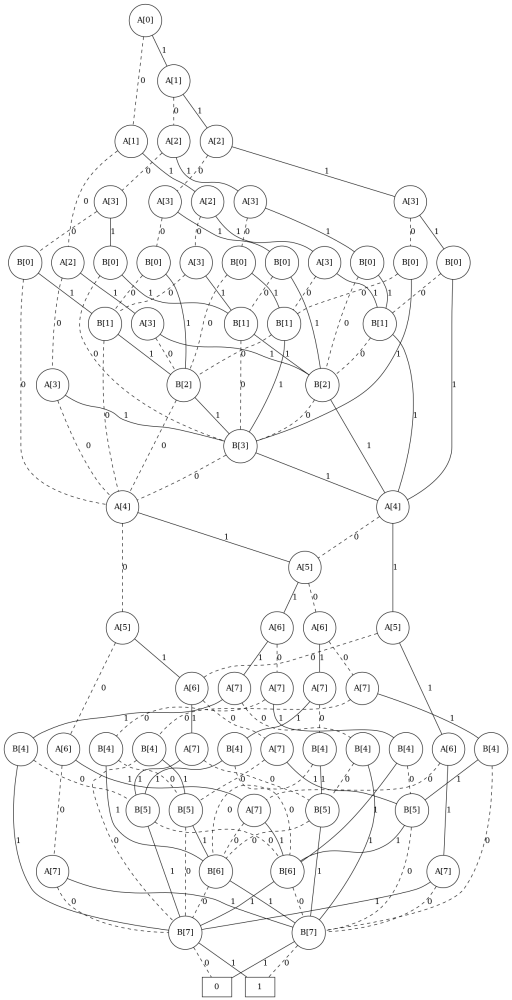


Example:

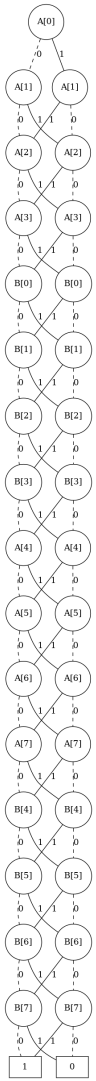
- 8 bit full adder
- 2^(8*2) = 65536 input combinations
- BDDs for output bit „1“

<= Approximate: 31 nodes

Exact: 49 nodes =>



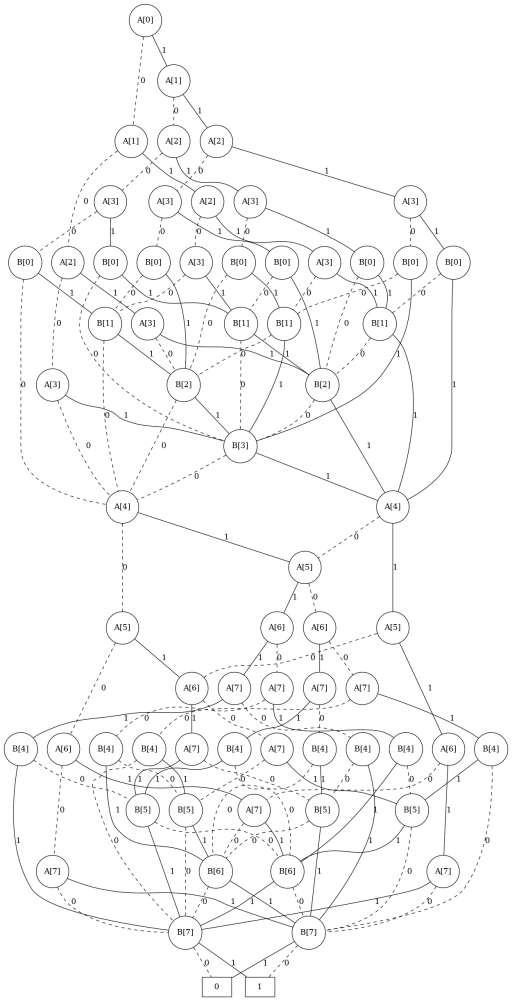
Methods – BDD Complexity



Example:

- 8 bit full adder
- 2^(8*2) = 65536 input combinations
- BDDs for output bit „1“

<= Approximate: 31 nodes



Exact: 49 nodes =>

64 bit adder ?
2^(64*2) = 3.4E38 input combinations

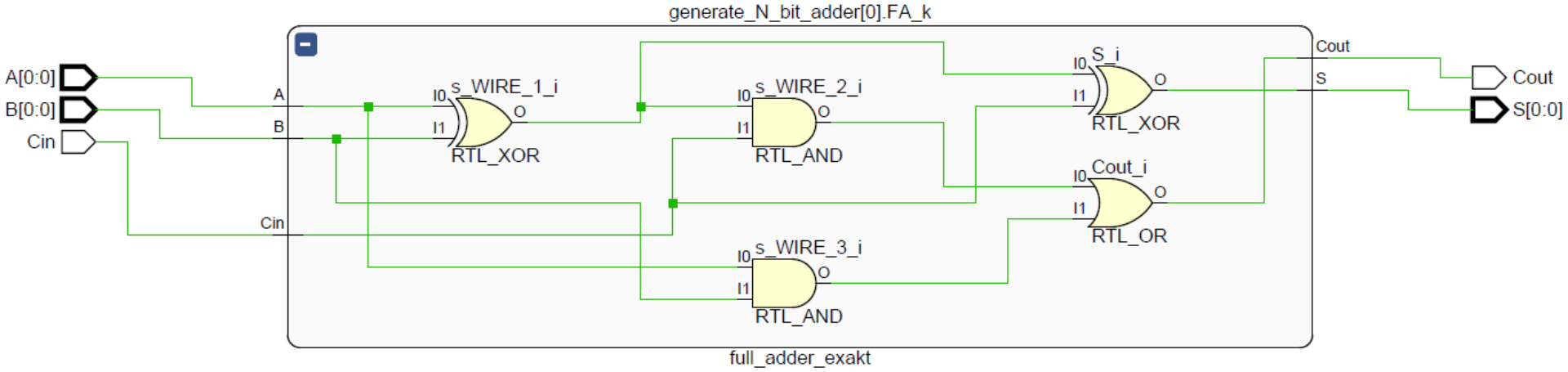
Methods - VHDL Implementation

- Vivado used for:
  - Power consumption analysis
  - Area analysis
  - Timing analysis

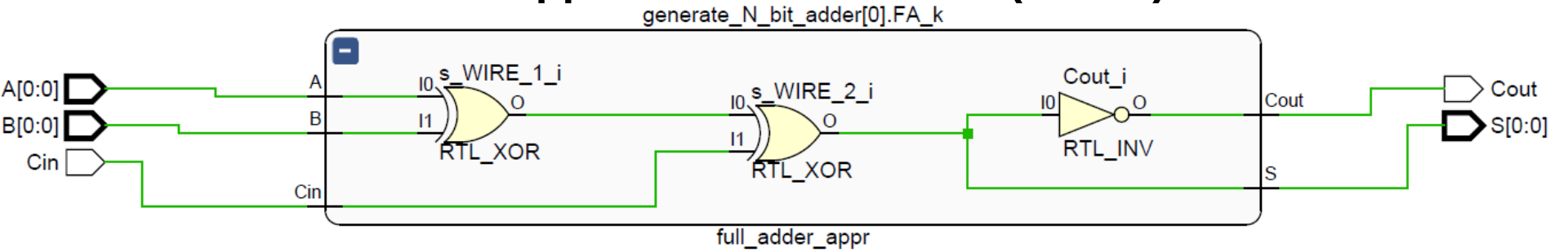
Methods - VHDL Implementation

- Exact vs. Approximate: reduced basic logic gate number

1-Bit Full Adder:

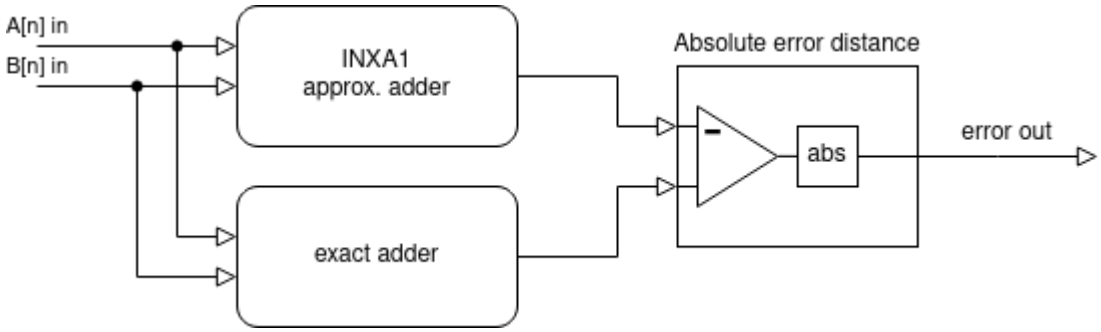


1-Bit Approximate Full Adder (INXA1):



Results

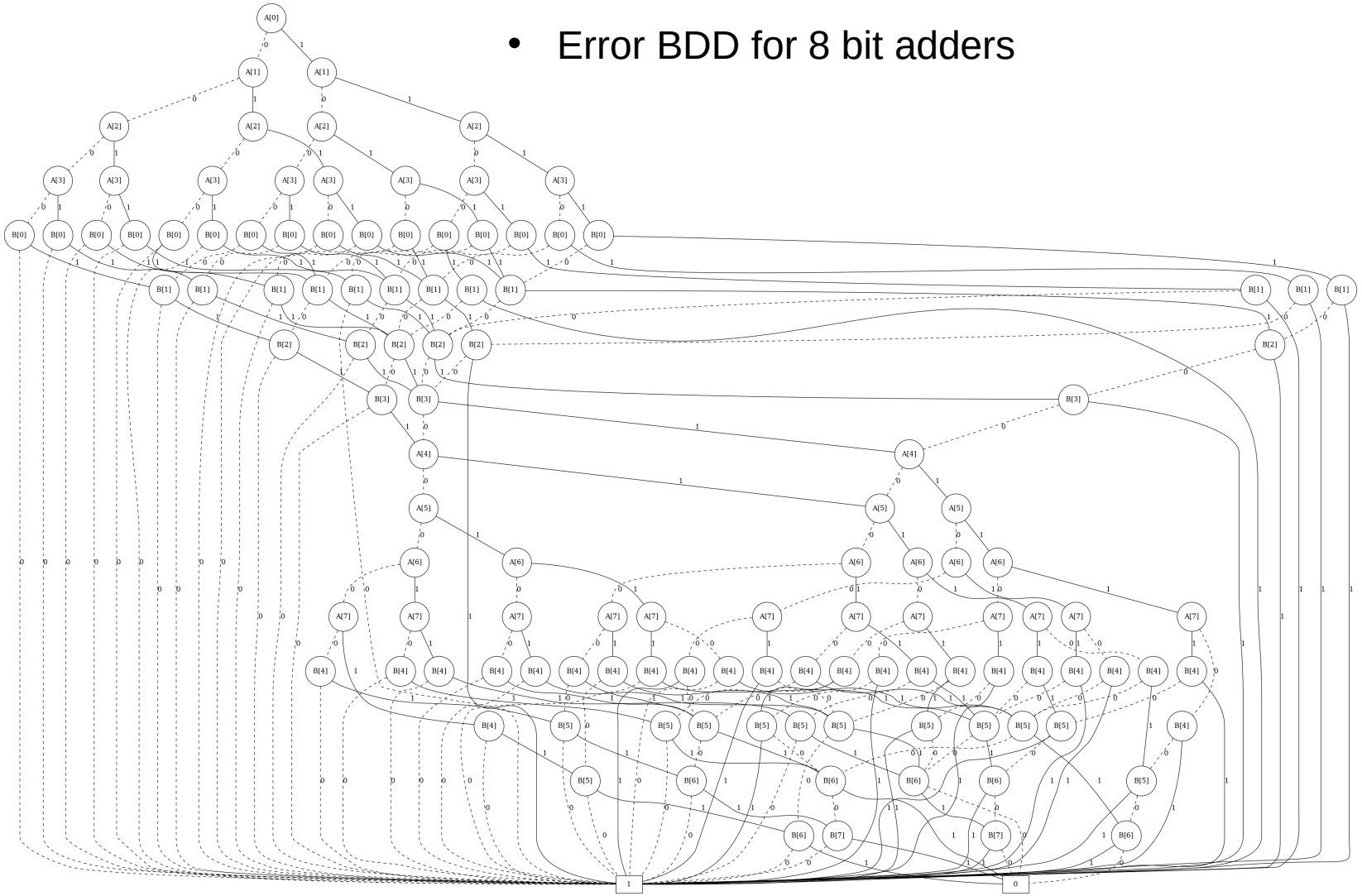
Results - BDD: Error Analysis



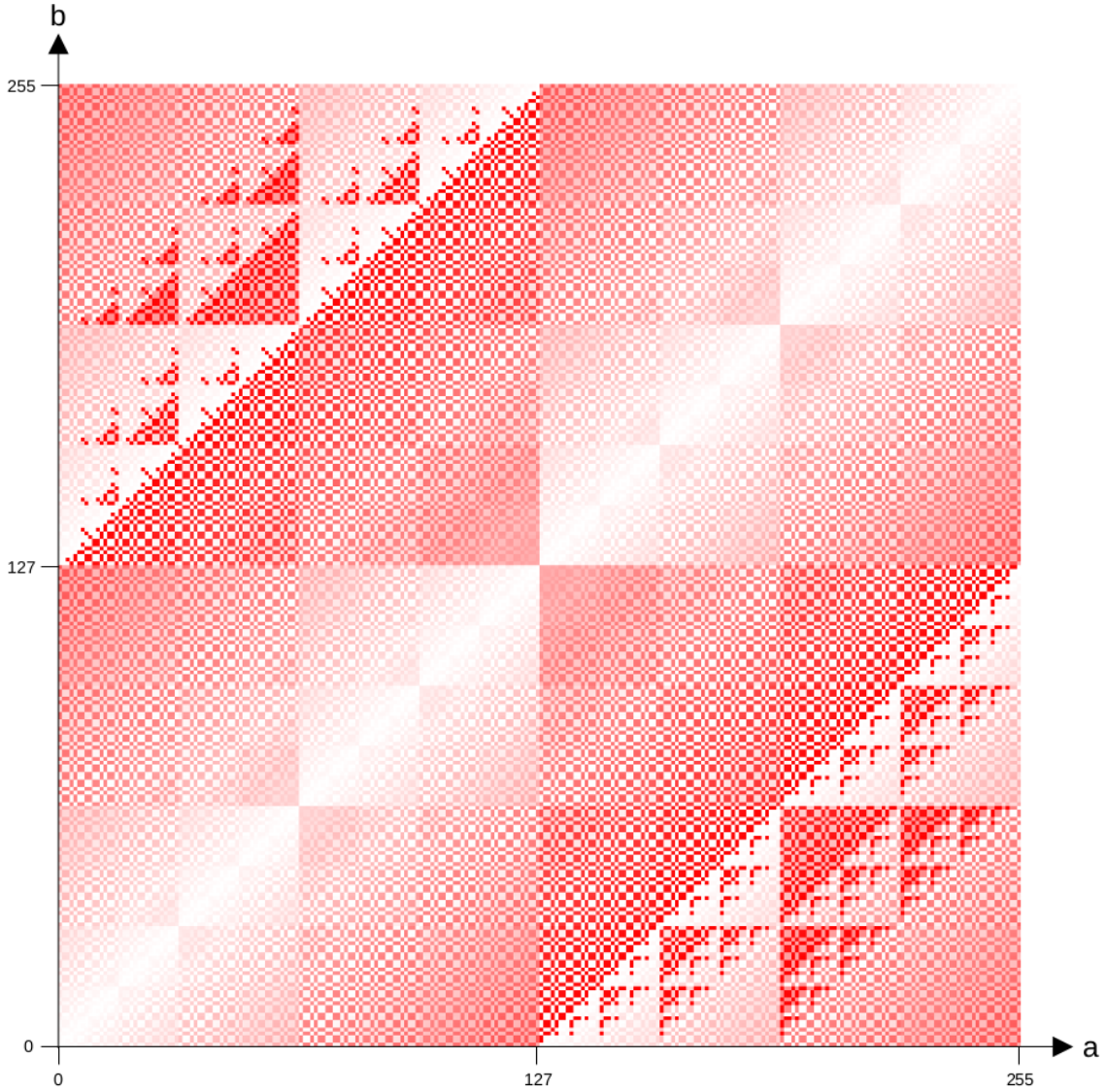

```
00 + 00 = approx: 010, exact: 000, error out: 2
00 + 01 = approx: 101, exact: 001, error out: 4
00 + 10 = approx: 100, exact: 010, error out: 2
00 + 11 = approx: 011, exact: 011, error out: 0
01 + 00 = approx: 101, exact: 001, error out: 4
01 + 01 = approx: 010, exact: 010, error out: 0
01 + 10 = approx: 011, exact: 011, error out: 0
...
```

Results - BDD: Error

- Error BDD for 8 bit adders



Results – Visual Error Analysis



Results – Timing Analysis

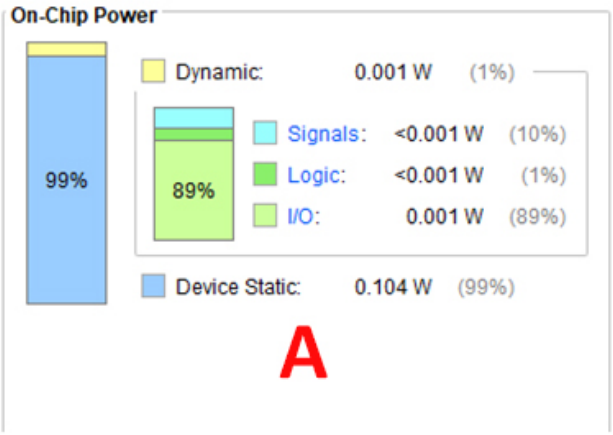
Type	A	B	C	S	C_{out}
4 Bit Exact	2,5	2,5	2,5	6,235	3,814
4 Bit Approx.	2,5	2,5	2,5	7,830	9,24
16 Bit Exact	2,5	2,5	2,5	6,460	3,847
16 Bit Approx.	2,5	2,5	2,5	9,39	9,999

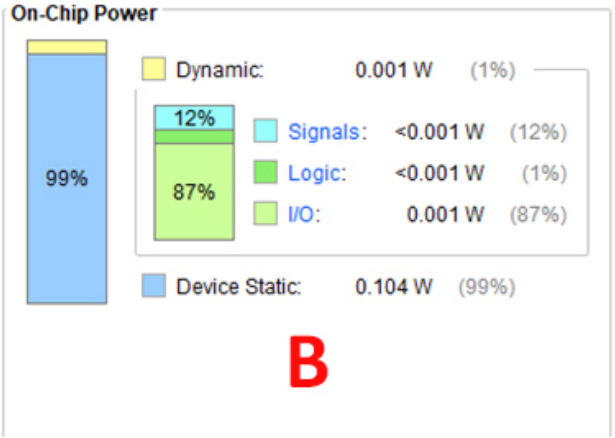
Timing Analysis – Signal Rate in Mega transactions (Mtr) per second

Results - Power Consumption

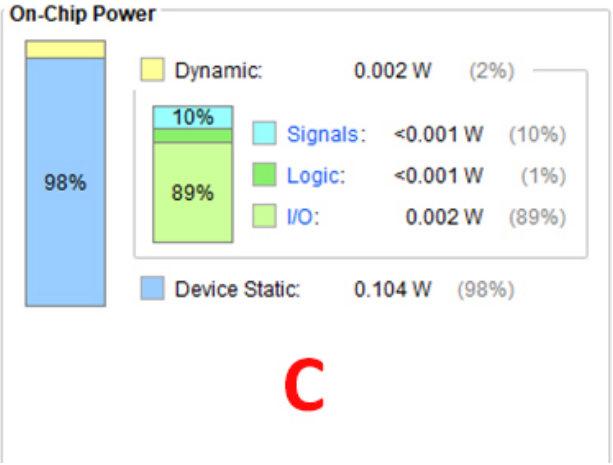
EXACT

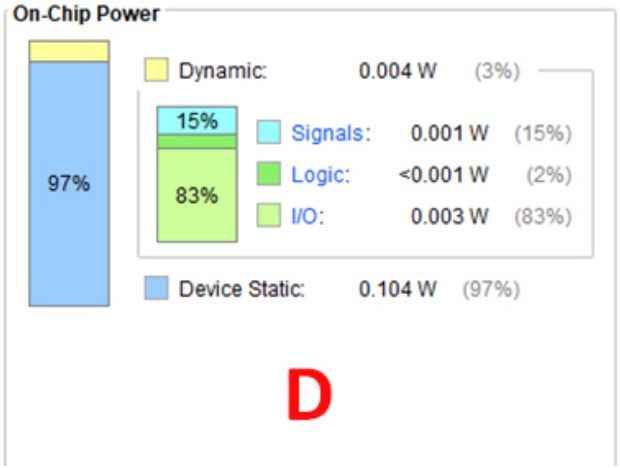
APPROXIMATE





4 BIT





16 BIT

Results - Area Analysis

Used metric for area: Gate Count in adders

TYPE	XOR	OR	AND	NOT	Basic Gates in Total
1 Bit Exact	2	1	2	X	11
4 Bit Exact	8	4	8	X	44
8 Bit Exact	16	8	16	X	88
16 Bit Exact	32	16	32	X	176
1 Bit Appro.	2	X	X	1	9 (-3)
4 Bit Appro.	8	X	X	4	36 (-8)
8 Bit Appro.	16	X	X	8	72 (-16)
16 Bit Appro.	32	X	X	16	144 (-32)

= ~18.2% less

XOR = 2 AND + 1 OR + 1 NOT = 4 Basic Gates

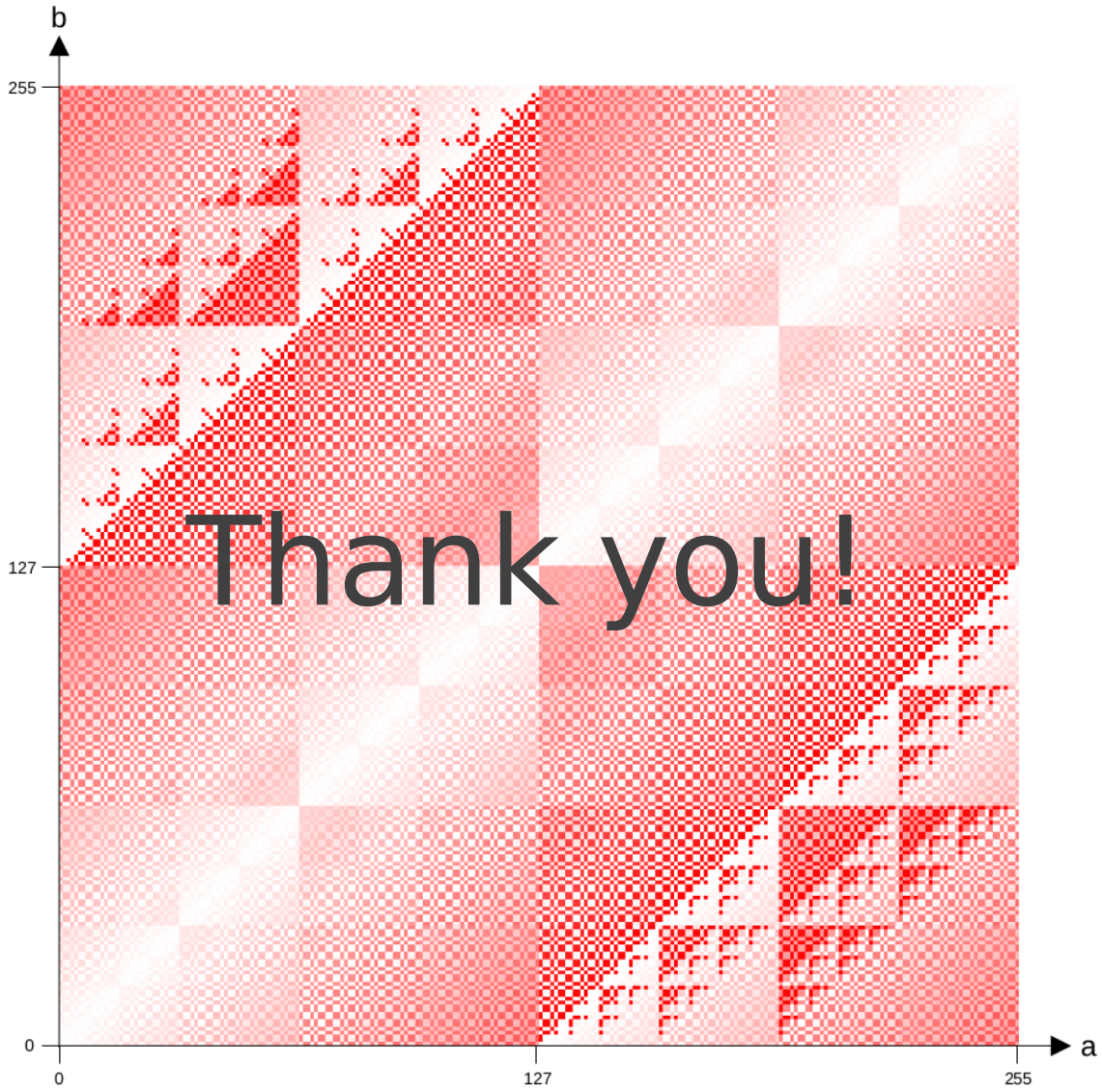
Conclusion

Achieved Goals

- ✔ Comparison of exact and approximate Full-Adder
- ✔ Power Consumption & Area evaluation
- ✔ Systematic error analysis
- ✔ VHDL implementation

Outlook

- Implementation on FPGA board
- Using an actual design



Thank you!