KU LEUVEN

FACULTEIT INDUSTRIËLE INGENIEURSWETENSCHAPPEN

CAMPUS GROEPT

CDD: Report final project

08/05/2024

Joran VLEUGELS – r0850643

Rohan BHATTARAM - r0921656

Academiejaar 2023-2024

1 SUMMARY

For the final project we designed Carry Select Adder + Carry Lookahead Adder. The maximum Adder Width tolerated by the design is 32 bits. It takes 18 cycles to complete a 512 bit addition.

2 TECHNICAL DESCRIPTION

We designed a 32-bit adder. In the bottom layer of our design you can find the 8 bit carry lookahead adder. Each 8 bit CLA is broken down into several 1 bit partial full adders which compute the carry, generate and propagate. The propagate signal indicates whether a carry should be passed through from a lower to a higher bit position, while the generate signal indicates that a carry will definitely be produced for the next higher bit.

The generate and propagate bits help the carry lookahead adder to predict the carry across multiple bits rather than waiting.

This is then connected to carry lookahead logic where the partial full adders are called in a loop to compute the carry outputs across all bits.

The structure of a Carry Lookahead adder is shown below, in our case the adder would have 8 PFAs instead of 4 shown in this figure.

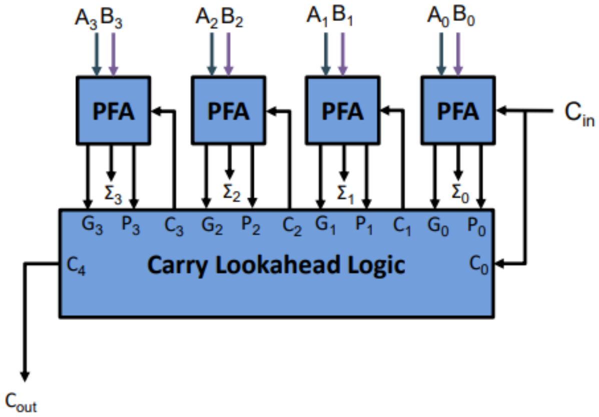


Figure 1: 4 bit Carry lookahead adder

We create our 32 bit adder (for a 32 bit addition) by using these carry lookahead adders in our carry select adder.

The carry select adder enhances calculation speeds by pre-computing two possible outcomes for each set of bits—one assuming no carry-in (0) and the other assuming there is a carry-in (1). When the actual carry-in from the previous set is confirmed, it instantly selects the accurate sum through a multiplexer.

Each CLA block in the picture design will be a 8 bit lookahead adder and thus will receive 8 bits. If we do this we get the following structure with N = 32 and M =8:

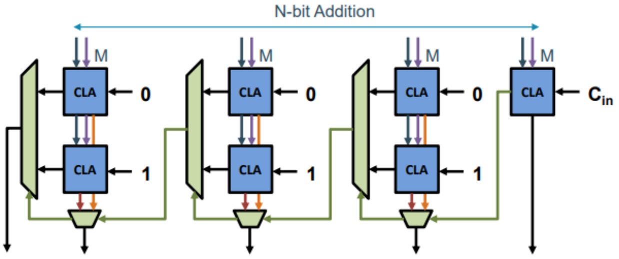
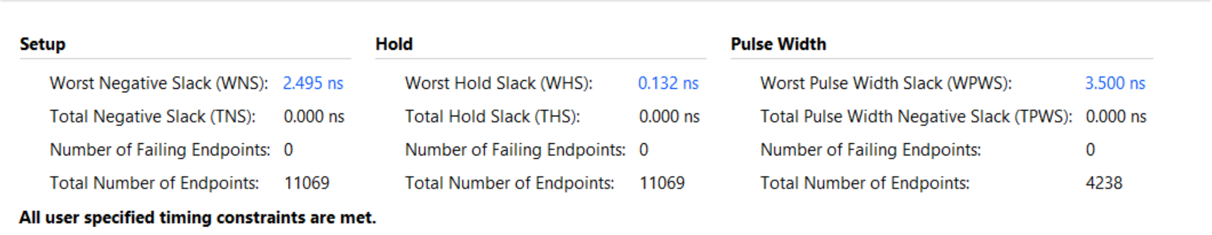


Figure 2: Carry Select adder + Carry lookahead adder

3 PERFORMANCE EVALUATION

3.1 Timing evaluation

The figure below shows the timing report. Our adder design shows good performance, with a worst negative slack (WNS) of 2.495 ns, ensuring that the slowest path meets the timing requirement perfectly. The worst hold slack (WHS) and worst pulse width slack (WPWS) being 0.132 ns and 2.459 ns respectively, it not only meets but exceeds the user specified requirements.

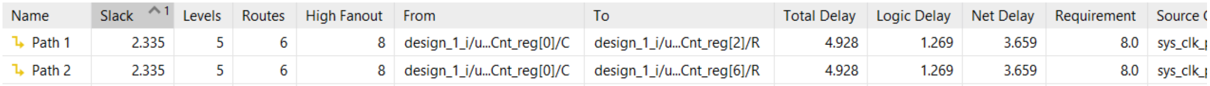


Setup		Hold		Pulse Width	
Worst Negative Slack (WNS):	2.495 ns	Worst Hold Slack (WHS):	0.132 ns	Worst Pulse Width Slack (WPWS):	3.500 ns
Total Negative Slack (TNS):	0.000 ns	Total Hold Slack (THS):	0.000 ns	Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0	Number of Failing Endpoints:	0	Number of Failing Endpoints:	0
Total Number of Endpoints:	11069	Total Number of Endpoints:	11069	Total Number of Endpoints:	4238

All user specified timing constraints are met.

Figure 3: Timing Closure of our design

The worst delay is 4.928 ns, as shown in the figure below.

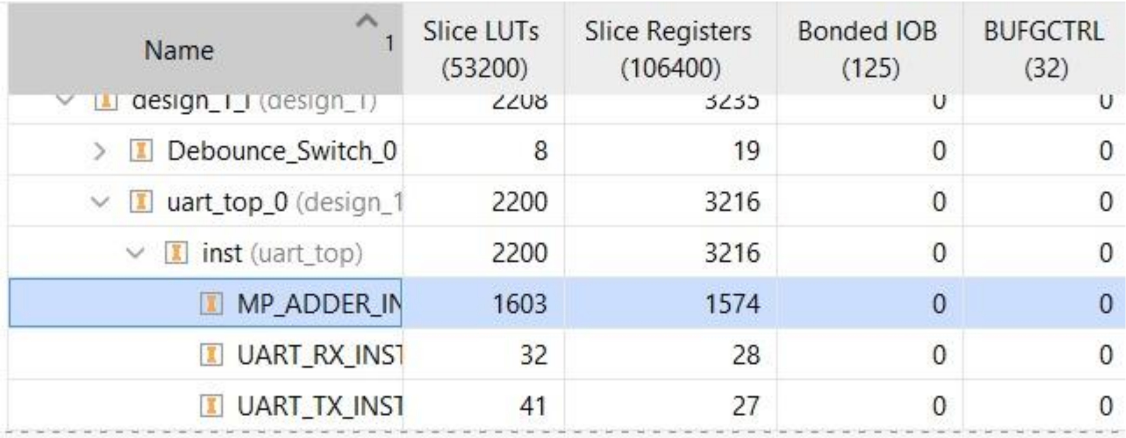


Name	Slack ^1	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Source (
Path 1	2.335	5	6	8	design_1_i/u...Cnt_reg[0]/C	design_1_i/u...Cnt_reg[2]/R	4.928	1.269	3.659	8.0	sys_clk_
Path 2	2.335	5	6	8	design_1_i/u...Cnt_reg[0]/C	design_1_i/u...Cnt_reg[6]/R	4.928	1.269	3.659	8.0	sys_clk_

Figure 4: Worst delays

3.2 Utilization and Power Report

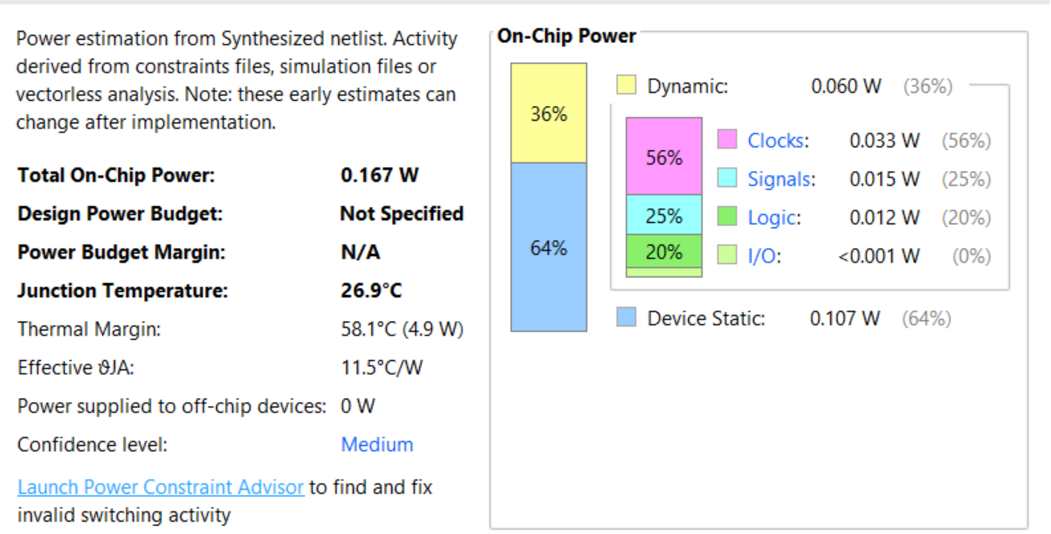
Our design used 1603 Slice LUTs and 1574 FFs which is smaller than the number of Slice LUTs and FFs we have which is 53200 and 106400 available on the FPGA. We can see the information about the utilization in the figure below:



Name	Slice LUTs (53200)	Slice Registers (106400)	Bonded IOB (125)	BUFGCTRL (32)
design_1_i (design_1)	2208	3235	0	0
Debounce_Switch_0	8	19	0	0
uart_top_0 (design_1	2200	3216	0	0
inst (uart_top)	2200	3216	0	0
MP_ADDER_IN	1603	1574	0	0
UART_RX_INST	32	28	0	0
UART_TX_INST	41	27	0	0

Figure 5: Utilization report of our design

The total on-chip power is 0.167 W with a medium confidence level, this is shown in the power report below.



Power estimation from Synthesized netlist. Activity derived from constraints files, simulation files or vectorless analysis. Note: these early estimates can change after implementation.

Total On-Chip Power:	**0.167 W**
Design Power Budget:	**Not Specified**
Power Budget Margin:	**N/A**
Junction Temperature:	**26.9°C**
Thermal Margin:	58.1°C (4.9 W)
Effective ϑJA:	11.5°C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Medium

Launch Power Constraint Advisor to find and fix invalid switching activity

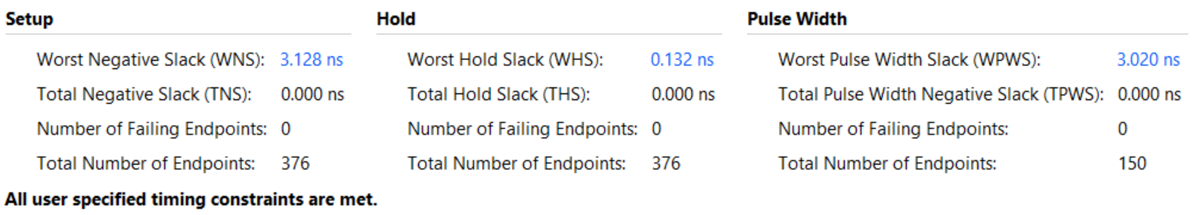
On-Chip Power

Dynamic:	0.060 W	(36%)
Clocks:	0.033 W	(56%)
Signals:	0.015 W	(25%)
Logic:	0.012 W	(20%)
I/O:	<0.001 W	(0%)
Device Static:	0.107 W	(64%)

Figure 6: Power Report of our design

4 COMPARISON

As shown in the figure below, the WNS for the ripple carry adder is 3.128 ns. This is higher than the 2.498 ns we got for the adder we designed. It indicates that our adder meets its timing requirements more efficiently and is likely to perform better in terms of speed.



Setup

Worst Negative Slack (WNS):	3.128 ns
Total Negative Slack (TNS):	0.000 ns
Number of Failing Endpoints:	0
Total Number of Endpoints:	376

Hold

Worst Hold Slack (WHS):	0.132 ns
Total Hold Slack (THS):	0.000 ns
Number of Failing Endpoints:	0
Total Number of Endpoints:	376

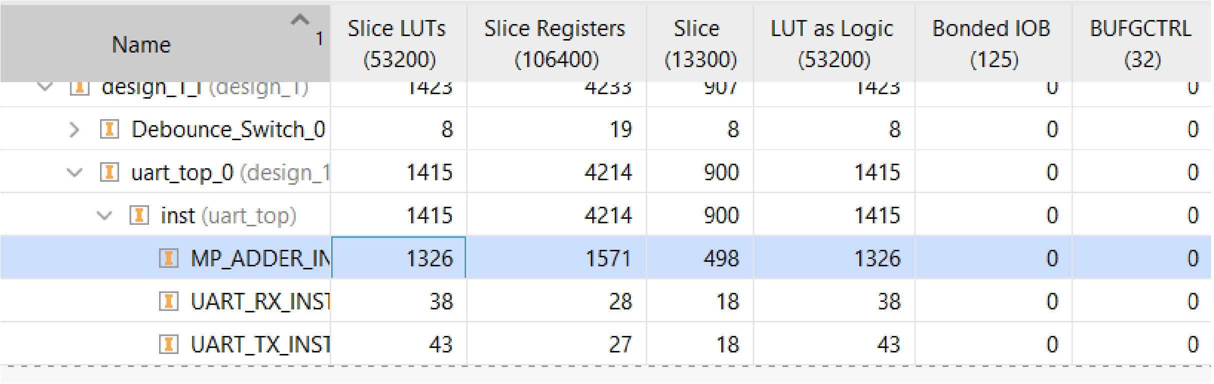
Pulse Width

Worst Pulse Width Slack (WPWS):	3.020 ns
Total Pulse Width Negative Slack (TPWS):	0.000 ns
Number of Failing Endpoints:	0
Total Number of Endpoints:	150

All user specified timing constraints are met.

Figure 7: Timing Closure of Ripple carry adder 16 bit

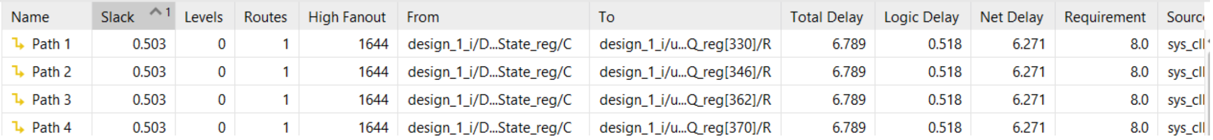
In our design we have a slight increase in the utilization, this is completely justified as we have scaled the adder to 32 bits and increased its efficiency.



Name 1	Slice LUTs (53200)	Slice Registers (106400)	Slice (13300)	LUT as Logic (53200)	Bonded IOB (125)	BUFGCTRL (32)
design_1_i (design_1)	1423	4233	907	1423	0	0
Debounce_Switch_0	8	19	8	8	0	0
uart_top_0 (design_1	1415	4214	900	1415	0	0
inst (uart_top)	1415	4214	900	1415	0	0
MP_ADDER_IN	1326	1571	498	1326	0	0
UART_RX_INST	38	28	18	38	0	0
UART_TX_INST	43	27	18	43	0	0

Figure 8: Utilization report of Ripple carry adder 16 bit

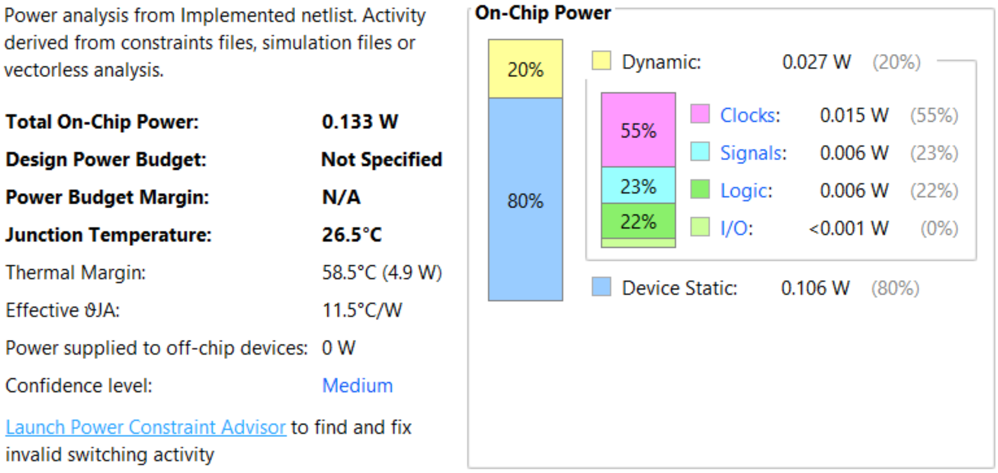
As we can clearly see that the worst delay is 6.789 for the ripple carry adder and the worst delay for our design is 4.789 which means there is a significant difference here, considerably outperforming the ripple carry adder with increased efficiency and speed showing that it is preferable for high speed computational applications. This is because The ripple carry adder operates sequentially, requiring the carry signal from one bit calculation to complete before moving on to the next bit. This design, however, can process multiple bits at once, significantly enhancing its speed.



Name	Slack ^1	Levels	Routes	High Fanout	From	To	Total Delay	Logic Delay	Net Delay	Requirement	Sourc
Path 1	0.503	0	1	1644	design_1_i/D...State_reg/C	design_1_i/u...Q_reg[330]/R	6.789	0.518	6.271	8.0	sys_cl
Path 2	0.503	0	1	1644	design_1_i/D...State_reg/C	design_1_i/u...Q_reg[346]/R	6.789	0.518	6.271	8.0	sys_cl
Path 3	0.503	0	1	1644	design_1_i/D...State_reg/C	design_1_i/u...Q_reg[362]/R	6.789	0.518	6.271	8.0	sys_cl
Path 4	0.503	0	1	1644	design_1_i/D...State_reg/C	design_1_i/u...Q_reg[370]/R	6.789	0.518	6.271	8.0	sys_cl

Figure 9: Worst delay for Ripple carry adder 16 bit

The carry-lookahead adder consumes slightly more power at 0.167 W compared to the ripple carry adder's 0.133 W, reflecting its enhanced speed and computational efficiency despite the modest increase in energy usage.



Power analysis from Implemented netlist. Activity derived from constraints files, simulation files or vectorless analysis.

Total On-Chip Power:	**0.133 W**
Design Power Budget:	**Not Specified**
Power Budget Margin:	**N/A**
Junction Temperature:	**26.5°C**
Thermal Margin:	58.5°C (4.9 W)
Effective ϑJA:	11.5°C/W
Power supplied to off-chip devices:	0 W
Confidence level:	Medium

Launch Power Constraint Advisor to find and fix invalid switching activity

On-Chip Power

Dynamic:	0.027 W	(20%)
Clocks:	0.015 W	(55%)
Signals:	0.006 W	(23%)
Logic:	0.006 W	(22%)
I/O:	<0.001 W	(0%)
Device Static:	0.106 W	(80%)

Figure 10: Power consumption of ripple carry adder 16 bit

Table 1 shows the comparison off the most important characteristics of both adders to conclude this report.

	LUTs	FFs	WNS	Total Delay	Total on-chip power
16 bit Riplle adder	1326	1571	3.128 ns	6.789 ns	0.133 W
Our design	1603	1574	2.495 ns	4.928 ns	0.167 W

Table 1: Comparison between our design and the 16 bit ripple carry adder

FACULTEIT INDUSTRIËLE INGENIEURSWETENSCHAPPEN
CAMPUS GROEP T LEUVEN
Andreas Vesaliusstraat 13
3000 LEUVEN, België
tel. + 32 16 30 10 30
iiw.groept@kuleuven.be
www.iiw.kuleuven.be