Project-1: Designing Basic Logical Gates

Omar Ali Elayat

Electrical and Computer Engineering Department

University of Waterloo

Abstract— **This project focuses on design exploration, analysis, and optimization of 16-bit adders architectures using 65nm technology. The objectives include detailed circuit design, and performance evaluation through functional and corner simulations. This study aims to enhance our understanding of full adder circuit behavior in modern semiconductor technology by investigating the impact of power-supply voltage on power-delay and energy-delay products.**

Keywords—CMOS Technology, Arithmetic Units, Corner Simulation, RCA, CSA_{sq}, KSA, CBA, PDP, EDP.

I. INTRODUCTION

This project explores and optimizes four 16-bit adder architectures—Ripple Carry Adder (RCA), Square Root Carry Select Adder (CSAsq), Carry Bypass Adder (CBA), and Kogge-Stone Adder (KSA) using 65nm technology. The selected architectures offer unique trade-offs in terms of speed, area, and power consumption. The primary objectives of this study encompass four key aspects. First, the project entails the detailed transistor-level design and optimizations of the 16-bit adder circuits. Subsequently, We set an objective function as a benchmark and the most optimized circuit is chosen and functionally verified against three test vectors that puts the adders in extreme propagate (P) and/or generate (G) conditions. The chosen adder delay/power/PDP/EDP vs power supply is also assessed. The rest of the report is organized as follows: Chapter II describes in detail the design and optimization of the adders. Chapter III discusses functional simulations, Chapter IV discusses global mismatch simulations and power-supply sweep for the chosen adder, and finally Chapter V is the conclusion.

II. ADDERS DESIGN AND OPTIMIZATION

A. 1-Bit Full adder Design and Optimization

To design a highly optimized 16-bit adder, CMOS and PTL FA adders were designed and optimized to fit our design requirements.

1.1. CMOS Mirrored Full Adder

A Mirrored Logic Full Adder is considered advantageous compared to a normal CMOS adder due to its symmetric structure. The mirrored design enhances performance by ensuring balanced signal paths, resulting in improved speed and reduced power consumption. The symmetry in layout contributes to better signal propagation, making mirrored logic full adders suitable for high carry propagation chains.

The pull-down network (PDN) and pull-up network (PUN) of the adder was mapped to minimum-size inverters to provide equal drive strengths, symmetric rise and fall times, and power efficiency while simplifying the design and reducing sensitivity to manufacturing variations. In this design, W_n was fixed at 200 nm and using ADE L parametric analysis, W_p was swept in the range from 1 to 5 to find the ratio $\frac{W_p}{W_n}$ that equates TpLH to TpHL. According to the sweep, $\frac{W_p}{W_n} \cong 2$ yield a TpLH = TpHL ≅ 5.4ps. Accordingly, $\frac{W_{PUN}}{PDN}$ of the adder is fixed at 2:1.

After sizing the PDN and PUN to min. sized inverter, the critical path ($C_{in} \rightarrow C_{out}$) was sized up for minimum path effort of Eq. (1) & Eq. (2) hence, minimizing the critical delay adder.

$$PE = \prod LE * B * F_o \qquad (1)$$

$$D = t_{p0}(\textstyle\sum_{i=1}^{N}(p_i + N\sqrt[N]{PE})) \qquad (2)$$

For maximum performance, the path effort (PE) should be close to 4 [1]. Since the logical effort (LE) is 2, $B * F_o = 2$. For a two cascaded minimum-sized 1-bit FA of Fig. (2), the total load on carry gate, $C_L = C_{ci} + (6 + +6 + 9) = 2C_{ci}$. Where C_{ci} is the input capacitance of carry-in signal. Solving for C_{ci} yields $PUN_{ci} : PDN_{ci} = 14:7$.

The critical path is further minimized by eliminating inverters in the carry chain, as in Fig. (1), thus, reducing inverting stages through exploiting the adder's inversion property.

1.2. PTL Full adder

CBA and KSA adders utilized the outputs of the setup block (P, G) and C_i signals to compute the sum (S) and carry-out (C_o) instead of A and B according to Eq.(3) & Eq.(4).

$$C_o = G + PC_i \qquad (3)$$

$$S = P \oplus C_i \qquad (4)$$

The used gates are non-inverting by nature. Thus, PTL designs in Fig. (2) were proven to be cheaper in silicon than their CMOS counterparts by eliminating the output inversion. Transmission gates were used to ensure full output swing, and low propagation delay. Moreover, appropriate buffering was placed along the PTL carry chains to break the exponential growth of the Elmore chain every M switch. Obviously, the number of switches per segment grows with increasing values of t_{buff}. In current technologies, M_{opt} is typically around 3 [1].

B. Ripple Carry Adder

A 16-bit RCA, Fig. (1), was constructed by cascading three 6-bit RCA, Fig. (2), where every 6-bit RCA is composed of six CMOS FA chained in series. As the inversion property was used, all the odd S outputs {1,3,5, … ,15} were inverted. According to Eq. (5) & Eq. (6), the worst cade delay of RCA is typically linear with the number of bits (N).

$$t_d = (N - 1)t_{carry} + t_{sum} \qquad (5)$$

$$t_{adder} = O(N) \qquad (6)$$

Where t_{carry} is the carry delay, and t_{sum} is the sum delay.

C. Carry Bypass Adder

Similar to RCA, a 16-bit CBA, Fig (3), is constructed from a 4-unit series chain of 4-bit CBA, Fig (2). The 4-bit RCA is composed of 3 stages, from top to bottom: Setup, addition, and Multiplexing. The inputs of the 4-bit CBA are the Cin and A[3:0] and B[3:0] vectors. Then, the next block takes an offset of M bits from both vectors, where M is 4 in our case. We will use the notation $A_i|B_i$ pair to refer to the A and B inputs of a given block$_i$. The working principle is to directly forward the C_{in} if all the $A_i|B_i$ to $A_{m-1}B_{m-1}$ pairs propagates the Cin block

input. Thus, block$_{i+1}$ can start the addition immediately without waiting for the C$_o$ from A$_{m-1}$B$_{m-1}$ of block$_{i-1}$. Initially, the setup block of each A$_i$|B$_i$ generates a propagate (P) and generate (G) pair (P$_i$|G$_i$) according to Eq. (7) & Eq. (8). Then, P$_i$|G$_i$ and C$_{in}$ is fed to the PTL P|G adder as discussed earlier. The vector S[0:3] is driven directly from the adders, however, the C$_o$ has to be multiplexed with C$_{in}$ to decide whether to bypass the carry or kill/generate according to the mux signal Sel from Eq. (9).

$$P_i = A_i \oplus B_i \qquad (7)$$

$$G_i = B_i B_i \qquad (8)$$

$$sel = P_0 P_1 P_2 P_3 \qquad (9)$$

According to Eq. (10) & Eq. (11), the worst cade delay of CBA is typically linear with the number of bits (N), however it should have a lower slope than RCA because of division by M stages.

$$t_d = t_{setup} + M * t_{carry} + \left(\tfrac{N}{M} - 1\right) * t_{bypass} + (M - 1)t_{carry} + t_{sum} \quad (10)$$

$$t_{adder} = O(N) \qquad (11)$$

Where t$_{setup}$ is the setup block delay of each stage, and t$_{bypass}$ is the Multiplexing delay.

D. Square Root Carry Select Adder

RSA$_{sq}$ is another approach to beat the adders linear dependency on N. It's architecture is simply constructed from the optimized RCA discussed earlier in addition to an extra multiplexing stage. The working principle behind a 16-RSA$_{sq}$ is, similar to CBA, to divide the N bits into M blocks, where each addition unit in each block computes the S$_i$ and Co$_i$ of both C$_i$ = 0 and C$_i$ = 1 in parallel. Then, the two S$_i$ and two Co$_i$ output of each addition unit is multiplexed based on the signal C$_{in}$. Accordingly, C$_{in}$=0, S$_{i/1}$ and Co$_{i/1}$ is flushed, C$_{in}$=1, S$_{i/0}$ and Co$_{i/0}$ are propagated, and vice versa. In a sense, the adder will still linearly dependent on N, just like CBA. However, RSA$_{sq}$ overcomes this linear dependency by accounting for the multiplexing delay when constructing the arithmetic blocks by progressively increasing the number of bits/group. In other words, assuming the first block has an M=4, the critical delay will be t$_{delay}$=RCA$_4$+t$_{mux}$. Thus, the next block will have an M=5, so by the time the carry is multiplexed, an extra bit will be computed. Similarly, the third block will have an M=6 and so on.

$$t_d = t_{setup} + M * t_{carry} + \left(\sqrt{N}\right) * t_{mux} + t_{sum} \quad (12)$$

$$t_{adder} = O(\sqrt{N}) \qquad (13)$$

According to Eq. (12) & Eq. (13), RSA$_{sq}$ has a square dependency on N.

E. Radix-2 Kogge-Stone Adder

KSA is a parallel prefix form carry look ahead where carries are computed fast by computing them in parallel at the cost of increased area. KSA is composed of 3 stages: setup, look ahead network, and post-processing. The setup stage is the same as in CBA. The look-ahead network stage involves computation of carries corresponding to each bit. A hierarchy carry chain is built by dividing the addend into two parts, a higher part (H) and a lower part (L). The G|P function is expressed in Eq. (15) & Eq. (16).

$$P_{HL} = P_H P_L \qquad (14)$$

$$G_{HL} = G_H + P_H G_L \qquad (15)$$

$$C_H = G_L + P_L C_{in} \qquad (16)$$

$$P_{HL} = P_L \oplus C_{in} \qquad (17)$$

Given the value of the carry-in of the least significant bits, we also generate the carries for every adder by considering the G and P of all the least significant bits as in Eq. (16). Finally, in the post-processing phase, the sum vector and C$_o$ is computed at one shot according to Eq. (17) & Eq. (3) respectively. Appropriate 2 stage buffers were placed intra-stages to handle the stages high-fanout. There are only 4 stages in carry generation tree, the worst-case propagation is typically four times than a 1-bit adder. Thus, t$_d$ ~ log$_2$(N).

III. FUNCTIONAL SIMULATIONS

Functional simulations were performed on the adders using the test vectors provided. The vectors assert different P|G combinations. All the simulation settings adhered to the provided project specifications. As shown in Fig. (5), All the adders are functioning as expected. The Input/Output vectors had to be grouped in vector buses, with appropriate labels, to meet the report space constraints. A summary of the worst-case delay, average power, power-delay-product (PDP), and energy-delay-product of all adders across the three test vectors can be found in Table (1) & Table (2). None of the vectors causes a falling Co edge. Thus, results were collected in the form of only the worst-case TpLH$_{co}$, TpHL$_{s(15)}$, and TpLH$_{s(15)}$. Where worst-case delay is based on TpLH$_{co}$ only since it will be the one scaling with N. According to Table (2), RCA exhibits the highest worst-case delay, followed by CBA. Surprisingly, KSA had higher delay than CSA$_{sq}$, although t$_{KSA}$ ≈ t$_{CSA}$ ~ t$_d$, which can be explained by the effect of high branching overhead of KSA. Since we are optimizing for performance, EDP was proven to be the best metric for choosing an adder since it puts more emphasis on the delay part of the equation. Thus, CSA$_{sq}$ was chosen as the best architecture for our application.

IV. GLOBAL MISMATCH AND POWER-SUPPLY SWEEP

CSA$_{sq}$ performance was assessed under two extreme process corners (ff & ss) and extreme temperature (-25°C and 85°C). The results were compared to the typical condition (tt) under typical temperature (27°C). Table (3) and Fig. (6) provides a comprehensive summary for the critical delay, average power, PDP, and EDP for all three vector bitstreams. As expected CSA$_{sq}$ had 21.9% delay, 2.6% power, 17.7% PDP, 35.8% EDP performance improvement at ff/-25°C, and 39.5% delay, 2.6% power, 27.5% PDP, and 67.4% EDP overhead at ss/85°C. At high temperatures, carrier mobility decreases and effective resistance increases, thus delay decreases and power dissipation increases. CSA$_{sq}$ design has shown its corner process and temperature tolerance, as it abides by the timing constraints thoroughly.

Similarly, a power-supply (VDD) sweep under tt (27°C) with vector A was performed to verify the adder's tolerance to VDD variations. As shown in Fig. (6), Power dissipation is increasing quadratically with VDD, while delay is decreasing exponentially when the transistors are ON (VDD > 0.6v). This

variation is translated to an exponential decay in both PDP and EDP. To ensure that the adder meets the timing constraints of 400ps, VDD has to be a minimum of 0.7v which results in a $TpLH_{Co} \approx 400ps$, Power $\approx 36.2\ \mu W$, PDP $\approx 14.5fJ$, and EDP ≈ 5.79 yJs of vector A in isolation.

V. CONCLUSION

From the analysis of VTC characteristics it can be observed that CMOS inverter has a very narrow transition zone. Therefore, high gain can be achieved when both NMOS and PMOS are simultaneously ON and operated in saturation. Thus, in transition region a small change in the input voltage results in a large output variation.

REFERENCES

[1] J. M. Rabaey, A. Chandrakasan, and B. Nikoli´c, Digital Integrated Circuits: A Design Perspective, 2nd ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2003

TABLE I TPLH AND TPHL OF THE ADDERS AT TT 27°C

Arch.	Vector A				Vector B				Vector C			
	Co		S(15)		Co		S(15)		Co		S(15)	
	TpLH(ps)	TpHL(ps)	TpLH(ps)	TpHL(ps)	TpLH(ps)	TpHL(ps)	TpLH(ps)	TpHL(ps)	TpLH(ps)	TpHL(ps)	TpLH(ps)	TpHL(ps)
KSA	233.23	120.4	16.4	198.8	118.5	124.1	29.38	42	118.5	122.5	78.5	112.3
CSA	173.2	59.4	21.9	173.7	59.09	67.29	44.63	63.95	75.41	37.5	44.66	63.86
CBA	256.9	65.38	33.6	352.2	59.25	48.05	32.31	54	59.25	48.06	73.58	41.5
RCA	269.4	44	21.92	280.7	47.02	52.5	38.2	39.13	65.4	30	37.8	54.2

TABLE II DELAY, POWER, PDP & EDP OF THE ADDERS AT TT 27°C

Output	corner			Output	corner		
	ss/85°C	tt/27°C	ff/-25°C		ss/85°C	tt/27°C	ff/-25°C
Power	338.6	348.7	367.7	TpHL_S_vecB	50.46	63.95	84.3
PDP	49.68	60.41	77	TPLH_Co_vecB	45.7	59.09	78.6
EDP	67.7	10.47	17.5	TpHL_Co_vecB	51.38	67.2	89.6
TpLH_S_vecA	15.85	21.91	30.2	TpLH_S_vecC	35.2	44.66	58.7
TpHL_S_vecA	135.7	173.7	227.5	TpHL_S_vecC	50.04	63.86	83.6
TPLH_Co_vecA	135.2	173.2	227.4	TPLH_Co_vecC	60.5	75.4	97.7
TpHL_Co_vecA	44.8	59.4	79.8	TpHL_Co_vecC	30	37.5	48.3
TpLH_S_vecB	35.3	44.63	58.4				

TABLE III DELAY, POWER, PDP & EDP OF THE ADDERS AT TT 27°C

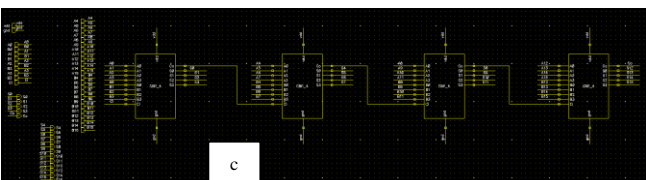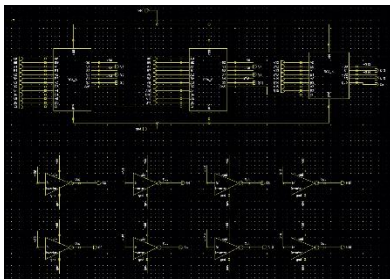| Arch.	Delay(ps)	Power(uW)	PDP(fJ)	EDP (yJs)
KSA	233.23	245.1	57.16	13.33
CSA	173.2	348.7	60.4	10.4
CBA	256.9	161.3	41.4	10.6
RCA	269.4	235.2	6.3	17.06

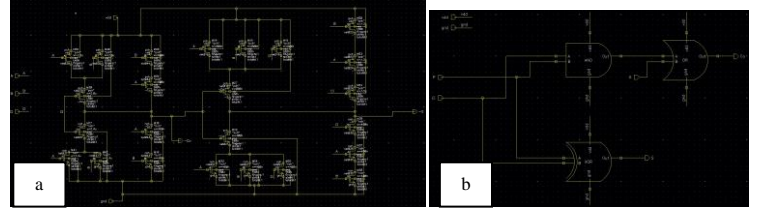Fig. 1. (a) 16-bit RCA. (b) 16-bit CSA. (c) 16-bit CBA (d) 16-bit KSA

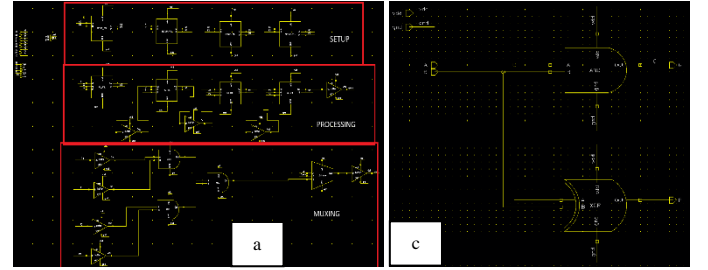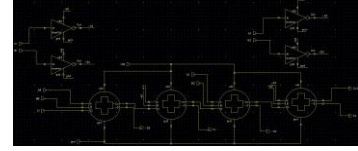Fig. 2. (a) CMOS FA. (b) PTL FA.

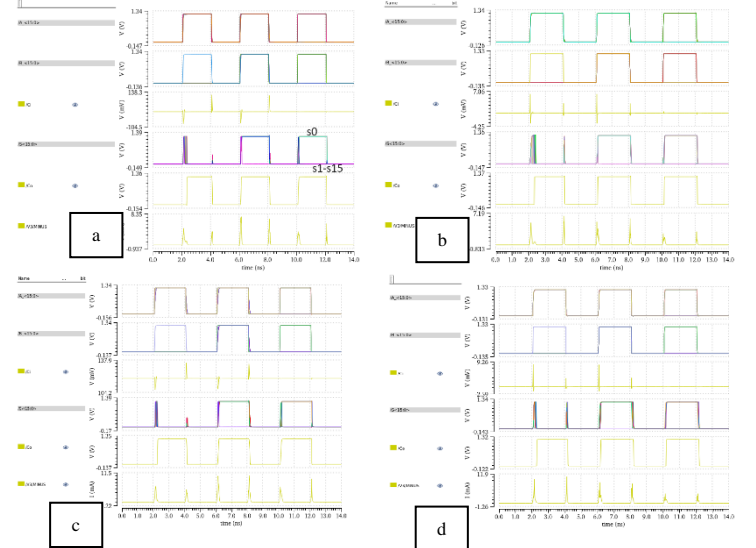Fig. 3. (a) 4-bit CBA. (b) 4-bit RCA. (c) G|P setup block

Fig. 4. Three vectors waveform of (a) 16-bit RCA. (b) 16-bit RBA. (c) 16-bit CSA. (d) 16-bit kogge-stone.
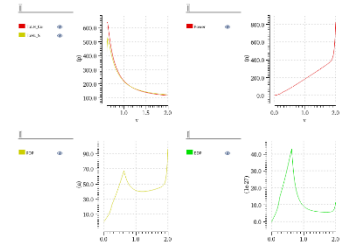

Fig. 5. Sweeping VDD versus different paramters of CSA_{sq} with test vector A.