5LIH0 Project Report : 16-bit Sklansky Adder

Burak Ergül(1280104)
Tamas Oliver Kocsis(1281887)
Keyshav Mor(1237978)
For the course : 5LIH0 Digital Integrated Circuit Design(Q2)
Department of Electrical Engineering
Eindhoven University Technology
Academic Year 2018-2019

I. Introduction

For the course 5LIH0: Digital Integrated Circuit Design at Eindhoven University of Technology, we have implemented a 16-bit Sklansky Adder.

The Sklansky Adder is a Tree Adder which are a family of Full Adders intended to reduce delays associated with Carry Look-Ahead, Carry-Bypass and Carry-Select Adders. These adders are used to make addition arithmetic circuits faster, consume lower power and area. However, for wide Adders where Number of Input Bits are greater or equal to 16, the delay of the Carry Look-Ahead, Carry-Bypass and Carry-Select adders increases as the Carry goes through the look ahead stages. This delay can be reduced by looking ahead across the look ahead blocks, which essentially gives rise to Tree Adders.

The Sklansky Adder is one such adder proposed by Jack Sklansky in his paper titled "Conditional-Sum Addition Logic" in the year 1960. The main advantage of the Sklansky Adder is that is has lower numbers of logic-levels and relatively easier routing which makes it a popular choice while implementing Tree Adders adding numbers greater than or equal to 16-bits.

II. Problem Formulation

A. Objectives of the Project

The primary objective of the project is to understand CMOS VLSI Design Process for 45nm CMOS technology using the FreePDK45 process design kit in conjunction with Cadence Virtuoso Analog Design Environment. To achieve this objective, we implement a 16-bit Sklansky Adder. The design process included schematic drawing, drawing layout of the schematic and post-layout simulation in prescribed conditions. The final design must demonstrate 16-bit adding functionality during post-layout simulation.

B. Specifications of the Project

2 16-bit numbers A [A15:A0] and B [B15:B0] along with a Carry-Input signal CIN are inputs to the Adder. The power supply to the adder and all its individual components is 1 V. The adder is tested under two different loads: 10f Farad and 100f Farad.

The adder must be implemented with static CMOS circuits which could be fully complementary and may have pass transistors or transmission gates. The transistors used in the low-level design should be standard threshold voltage transistors. The design should be optimized for size, power-consumption and delay. The rise-time (10 % - 90 %) of output voltage swing and fall time (90 % - 10 %) of the output voltage swing needs to be under 100 pico Seconds at a simulation temperature of 90º Celsius. Moreover, the circuit should function at input frequencies of 500 MHz or greater.

III. CMOS Circuit Design

A. 16-bit Sklansky Adder

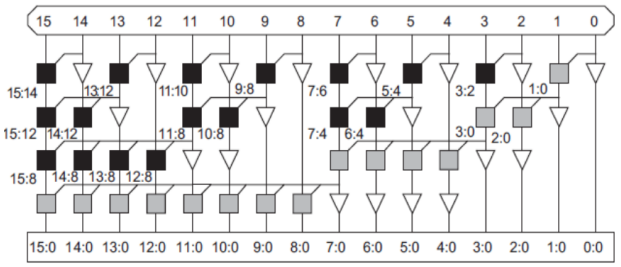

Fig. 1 The 16-bit Sklansky Adder PG Network

The 16-bit Sklansky Adder is a tree adder as we have mentioned previously. To understand the functioning of this adder, we need to understand the P(Propagate) and G(Generate) signals. The adder generates a carry when C_{out} is true independent of C_{in}, so $G = A \cdot B$. Whereas, the adder propagates a carry; i.e., it produces a carry-out if and only if it receives a carry-in, when exactly one input is true: $P = A \oplus B$. This carry generation and propagation is central to functioning of the adder.

We can generalize these signals to describe whether a group spanning bits i to j, inclusive, generate a carry or propagate a carry. A group of bits generates a carry if its carry-out is true independent of the carry-in; it propagates a carry if its carry-out is true when there is a carry-in.

These signals can be defined recursively for $i \geq k > j$ as

$$G_{i:j} = G_{i:k} + P_{i:k} \cdot G_{k-1:j} \quad \text{.....(1)}$$
$$P_{i:j} = P_{i:k} \cdot P_{k-1:j} \quad \text{.....(2)}$$

These two signals form the basic blocks of our 16-bit Sklansky Adder Schematic. The Sum for bit i can be then represented as

$$S_i = P_i \oplus C_{i-1} = P_i \oplus G_{i-1:0} \quad \text{.....(3)}$$

The adder consists of 3 building blocks, Black Cells, Gray Cells and Buffers. Black Cells generate and propagate logic to future stages. Gray cells generate logic and calculate sums which are ultimately used at the end. Buffers are used to ensure that the signal is propagated successfully to the

output to generate a full output voltage swing and minimize the load on critical paths.

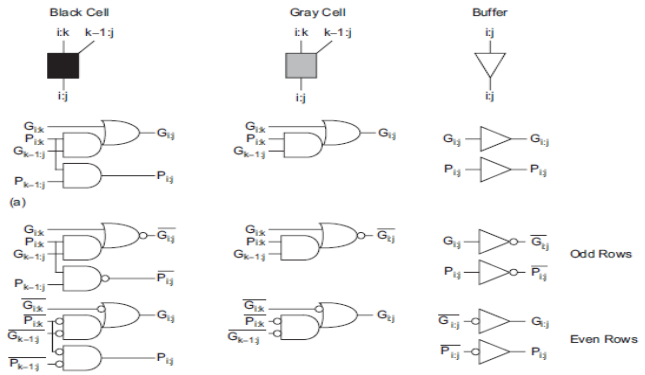

Fig.2 Black Cell, Gray Cell and Buffers

The critical path of this adder circuit is from the input point of Carry-in signal to the point where Carry-out signal is obtained along with the sum of input bits. This can be illustrated in Fig.1 by the concatenation of Gray cells from bit 0 till bit 15. The Sklansky Adder, with appropriate sizing reduces the delay on the critical path to

$$t_{tree} = t_{pg} + [\log_2 N]t_{AO} + t_{xor} \quad \text{.....(4)}$$

where t_{pg} is delay of propagate-generate cells, t_{AO} is delay of AND and OR gates in the gray cell and t_{xor} is the delay of XOR gates required for final summation.

B. Schematic of the Circuit

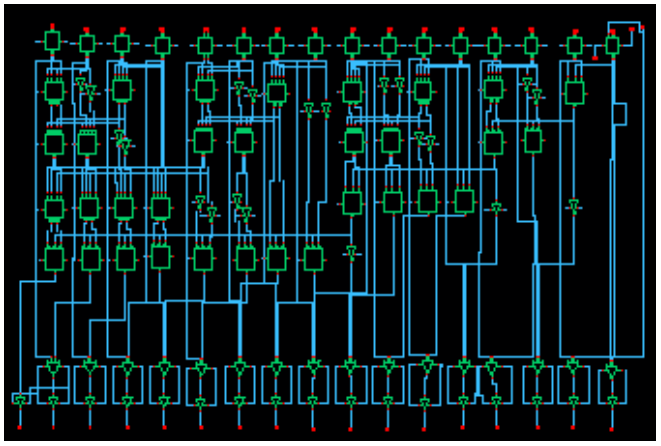

Fig. 3 Schematic of the 16-bit Sklansky Adder (LSB on the right, MSB on the left)

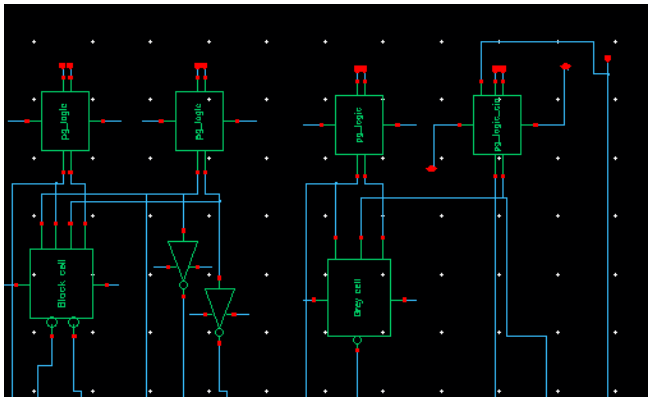

Fig.4 Interface of PG Logic, Black Cell and Gray Cells

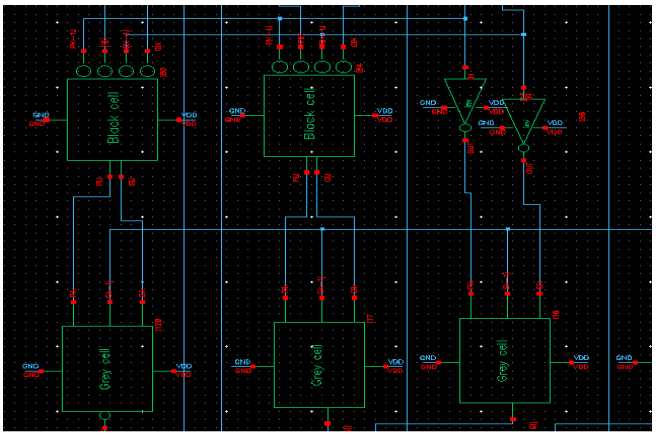

Fig.5 Interface of Black Cells (inverted inputs) with Gray Cell

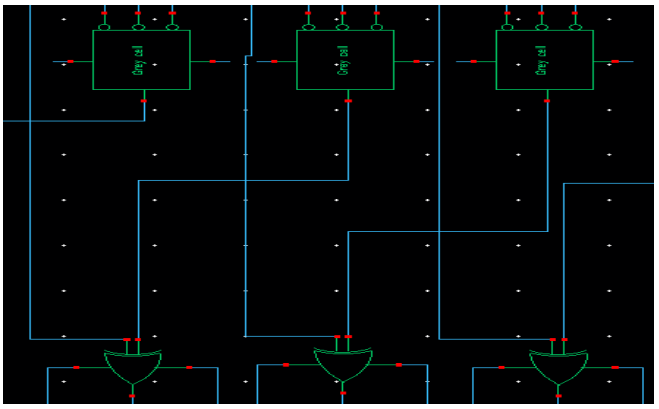

Fig.6 Interface of Gray Cells with final summation XOR gates.

We chose to make a full complementary static CMOS design for all the aforementioned gates. Our design includes different versions of black and grey cells with either an inverted input, an inverted output or non-inverted ports such that each gate in the sequence complements the next with only a few inverters required. This design reduces the area and makes the overall design process easier.

The choice of design for the xor gate can be mentioned here as well since there are several designs that can be compared. The pass transistor logic design has 6 transistors but it doesn't have a full output voltage swing and so it requires a buffer at the output, which means 4 more transistors. The full complementary design requires 8 transistors and has a full output voltage swing although it also requires a buffer at the end to be able to drive the capacitive load. We chose to use the full complementary design since we thought it would be more robust and easier to design. However, in hindsight, the pass transistor logic design is smaller and would have been the better choice (considering it also has a full output voltage swing with a buffer at the output). The difference is two transistors per xor gate.

C. Sizing of Transistors

The sizing of transistors and gates is done along the critical path to achieve minimum propagation delay between the input signal and the output carry signal. For achieving this

goal , we used the inverter of NMOS width = 90nm, PMOS width = 145nm and Length =50nm as our base reference.

We did the sizing of the critical path as per the stipulated output load capacitance of C_{load}=100f Farad. With the help of the reference inverter we were able to figure out the Logical effort(g) of each gate along the critical path, which in turn gave us the logical effort along the whole path(G). Similarly, since we calculated the branching effort of gates along the path which gave us the combined Branching effort(B) along the whole path.

The challenging part was to figure out the input load capacitances of the CMOS gates at the PG Logic stage which was found through the Cadence Virtuoso software. These combined capacitances gave us the value of C_{in}, which in turn helped us figure out the Electrical effort F of the critical path($F=C_{load}/C_{in}$). Multiplying F,G and B gave us path effort H($H=F\cdot G\cdot B$). Including the levels contributed by each gates within the Black and Gray Cells, we figured out that there are total N=10 stages, which in turn gave us the value 'h' which is the optimum stage effort($h^N=H$). Working backward from the Buffer at the C_{out} stage, we figure out the sizing of gates and individual CMOS ratios for each stage along the critical path. This sizing of the gates is only along the critical path as we have used minimum sized gates and buffers on the path other than the critical path. This sizing was done keeping in mind the load of 100f Farad which gave us a 67.49 ps Rise time and 64 ps Fall time.

However, since the post-layout simulation was only for a load of 10f Farad, we have used minimum sized elements along the critical path and only the buffer at the output is sized. This decision was taken since minimum sized elements were enough in driving the 10f Farad Load.

The speciality of our design is that we have managed to fulfill the timing requirements, delivering a fast-computing adder while also managing to use limited area for the layout.

D. Layout

We started the layout design, buy planning the transistor placement of each separate cell. This is called stick-diagrams, but it was not exactly the same style as visible on figure 5. The program used is PowerPoint.

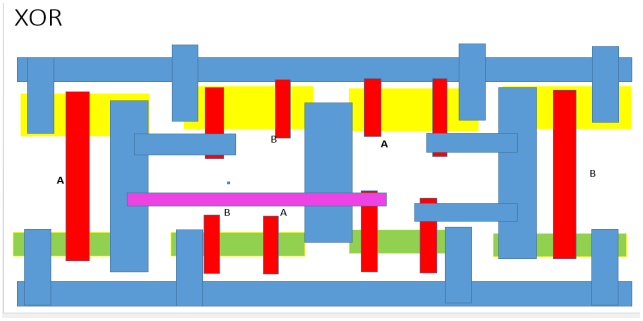

Fig.5: Layout planning of Xor gate

Although all the cells are planned, the actual layout had to follow the drc rules. We tried not to use metal 2 layer in these cells, but for more complicated cells it was inevitable.

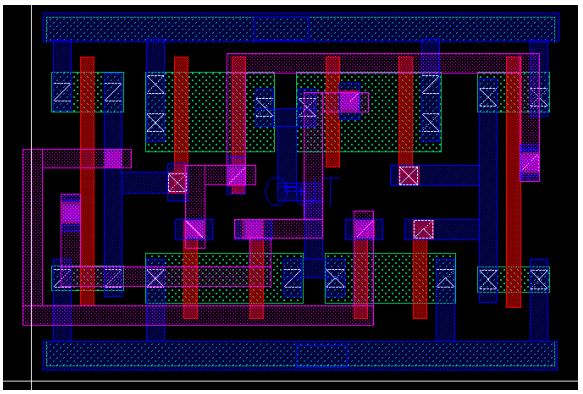

Fig.6: Actual layout of XOR gate.

The height of the cells were increased to 1.380 micrometer, to have space for increased sized activate areas, which in the end design we didn't use, as it was not a 100fF design. This increased height still helped with the routing.

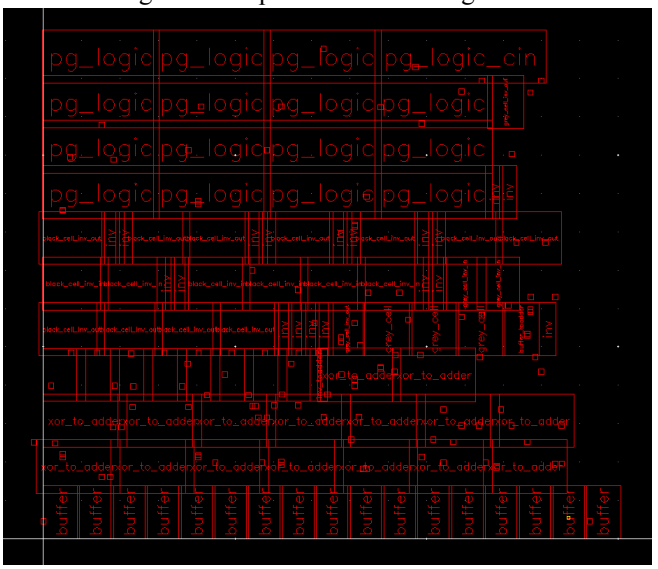

Fig.7: Layout of the adder.

For the final design, we placed in all the cells, so that their VDD and GND metal1 rails are in the same position, as shown in the lab videos. Every second line is rotated for matching rails. The biggest cells were pg_logic, the longest line is still the buffer part. The design is not perfect, better placement of the cells would have been possible. The constraint is simply time and complexity, as it is very difficult to see which component is which, if the cells are not keeping the same lines as in the schematic. Overall we spent more than 60 hours on completing the layout after all the separate cells were done, which also took considerate amount of effort. The wiring tool in layout design would have been a great help if explained earlier, and would have been save us much time, sadly we only found out about it halfway through. (ctrl+shift+w in layout design). Figure 7 shows the effort of routing, which was the most difficult in the congested grey cell / black cell area as those cells have 4/6 connections in a smaller area. Metal 2, 3, 4, and also 5 was used. After layer 4 it becomes increasingly difficult to find enough space for the vias, as metal 4 already needs spacing of 140nm.

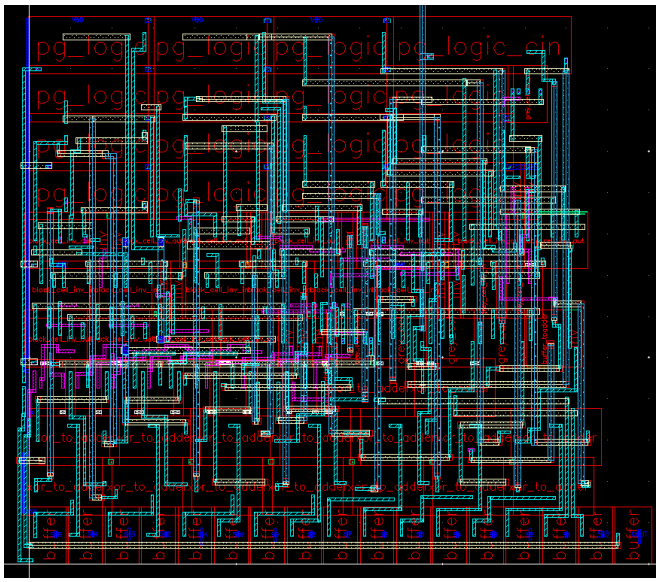

Fig.7: Routing of the cells in the final design, without displaying layers of cells.

IV. Results and Analysis

The analysis of the Sklansky adder was carried out in two phases. A pre-layout simulation with appropriate sizing of elements for the 10f Farad was done following which the post-layout simulation was carried out. Only a pre-layout simulation for 100f Farad Load was possible because of the changed load requirement and using a layout designed for a 100f Farad load to drive a 10f Farad load would result in power and area overheads. All these analyses except for the sized circuit have been done comparing the CIN with various output signals. It is observed that Rise and Fall times across the outputs remain equivalent or comparable whereas the propagation delay increases moving towards the COUT and S15 signal which is why they are at the core of our analysis.

Sr. No.	Load Capacitance	Propagation Delay (COUT)	Rise Time (COUT)	Fall Time (COUT)
1.	10f Farad	278.78 pS	19.79 pS	17.49 pS
2.	100f Farad	306.32 pS	67.49 pS	78.139pS

Table `1. Pre-Layout Simulation of the Adder with sized circuit optimized for 100f Farad Load

As can be seen from the data above, sizing of the circuit for a 100f Farad load does not produce significant reduction in propagation delay for lower loads but increases area of the layout. Also, the Rise and Fall times for 10f Farad load are significantly lower as compared to the Rime and Fall times for 100f Farad load, which is the upside of using a sized circuit for the smaller load.

Output	Load Capacitance	Propagation Delay (S4)	Rise Time (S4)	Fall Time (S4)
S4	10f Farad	163.15 pS	94.10 pS	95.56 pS
S10	10f Farad	161.53 pS	94.37 pS	94.74
S15	10f Farad	163.59 pS	94.72 pS	94.97
COUT	10f Farad	328.18 pS	95.63 pS	93.72 pS

Table 2. Pre-Layout Simulation of the Adder with sized buffer and minimum sized transistors

Output	Load Capacitance	Propagation Delay (S4)	Rise Time (S4)	Fall Time (S4)
S4	10f Farad	163.05 pS	94.17 pS	96.04 pS
S10	10f Farad	160.81 pS	95 pS	94.98
S15	10f Farad	163.65 pS	94.81 pS	94.82
COUT	10f Farad	328.08 pS	95.71 pS	93.69 pS

Table 3. Post-Layout Simulation of the Adder with sized buffer and minimum sized transistors

As can be seen from the post layout simulation the rise and fall times are approximately invariable across the different outputs. Even the propagation delays remain equivalent or comparable with the exception of the propagation delay between CIN and COUT.

This is observed probably since the XOR Gate preceding the buffer at S15 receives one of its inputs directly from the PG-logic cell adding the bits A15 & B15 without much delay. The other input to the same XOR gate is from the penultimate Gray cell. Thus, both the inputs arrive relatively earlier as compared to input of the buffer at COUT which comes from the last Gray cell which awaits inputs from the preceding block processing the last bits of the input. This leads to the huge gap between their respective propagation delays.

When the pre and post-layout simulation for 10f Farad load on the minimum sized circuit, it can be observed that the values do not vary a lot and are almost identical. However, they exhibit the same relation between propagation delay observed at S15 and COUT.

It must be also noted that the performance of the minimum sized circuit is not nearly as good as the performance of the sized circuit which gives significantly better propagation delay at COUT and also considerably optimized rise and fall times. The only drawback being that the sized circuit occupies a lot more space compared to the minimum sized circuit.

References

1. Rabaey, J. M., Chandrakasan, A. P., & Nikolić, B. (2013). Digital integrated circuits: A design perspective. Delhi: PHI Learning Private Limited.
2. Weste, N. H., & Harris, D. M. (2015). *CMOS VLSI design: A circuits and systems perspective*. India: Pearson India.

3. J. Sklansky, "Conditional-Sum Addition Logic," in *IRE Transactions on Electronic Computers*, vol. EC-9, no. 2, pp. 226-231, June 1960.