Department of Electronic and Computer Engineering
The Hong Kong University of Science and Technology
EESM5000 Design Project

Design a 32-bit Brent Kung Adder

Submitted to:
Prof. Volkan Kursun

From:
Chan Kei Hong
Cheung Pak Hey
Lam Yiu Fai

Contents

Group Information

The following is the personal information from group members:

CHAN Kei Hong Student number: 20743438

CHEUNG Pak Hey Student number: 20656075 (Group Representative)

LAM Yiu Fai Student number: 20737257

Abstract

To design a 32-bit brent kung adder, a comprehensive stepwise process is going to be taken. Four procedures are required for the design project, including setting up the cell module using transmission gate, design of 32-bit adder schematic design, layout design of the transistor logic and simulation of adder on both schematic level and layout level.

First, transistor logic with transmission gate is used to design each of the complex cell logic in 32-bit brent-kung adder. As using a transmission gate for specific functional design can provide better performance on power and reduce the number of transistors, the area of the design is reduced so that more margin is provided to speed up the adder. Compared with Fully- Complementary CMOS logic transmission gate is a better choice for the design. Combine the above cells are able to make a functional 32-bit brent-kung adder in an optimized way. More details are discussed in the second part of the report.

Second, the full schematic of the 32-bit brent-kung adder is designed. It contains 9 stages of operation, which follows the equation of 2log2N -1. The 32-bit brent-kung adder is made from three different logic, which is Bitwise PG logic, Group PG logic and Sum logic. It will be discussed in the third part of the report.

Lastly, all simulations are done after finishing the schematic and layout of the 32-bit brent-kung adder.

Standard Cell Library

A standard library is made to illustrate our design on 32-bit brent-kung adder. This library consists of four different components, including transistor logic with transmission gate, different types of cell logic composed of transmission gate, full 32-bit brent-kung adder and all layout of logic gate.

Three basic logic gates are designed using transmission gate, including AND gate, OR gate and XOR gate.

AND logic gate

Figure 2.1 is the schematic of the AND gate. For this gate, a CMOS inverter, which is used to invert the A to A' for the usage of transmission gate. So, A is used to determine the transmission gate to turn on or off and the transmission gate is used to pass B to the output when the transmission gate is on. If A is zero, it would not pass anything. Moreover, a NMOS at the output, which is used to discharge the output, is used to pull down the output when A · B switches to A' · B'. As a result, the Boolean function is defined as follow:

$$Out = A \cdot B$$

Also, the layout of AND gate is illustrated in Figure 2.2.

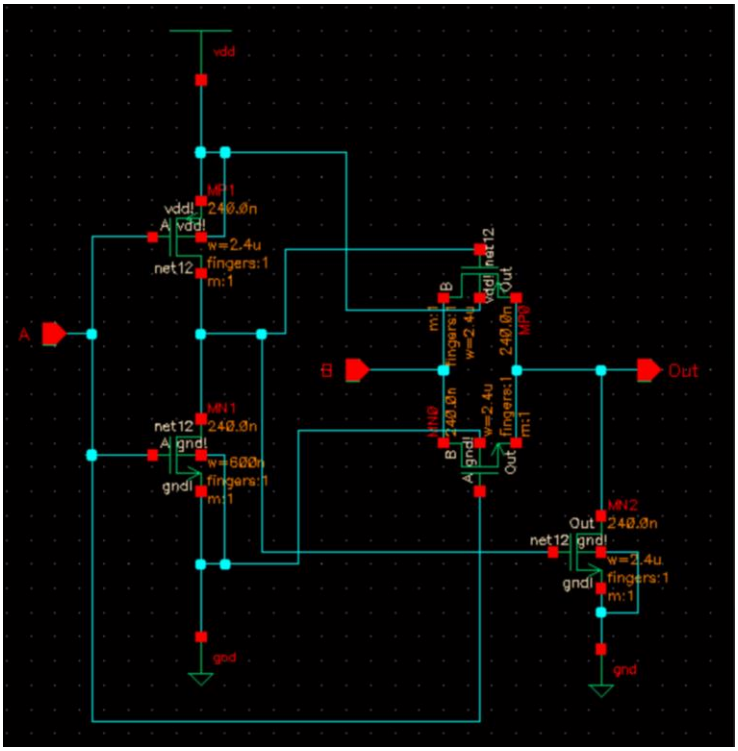

Figure 2.1. Schematic of AND gate (Cadence)

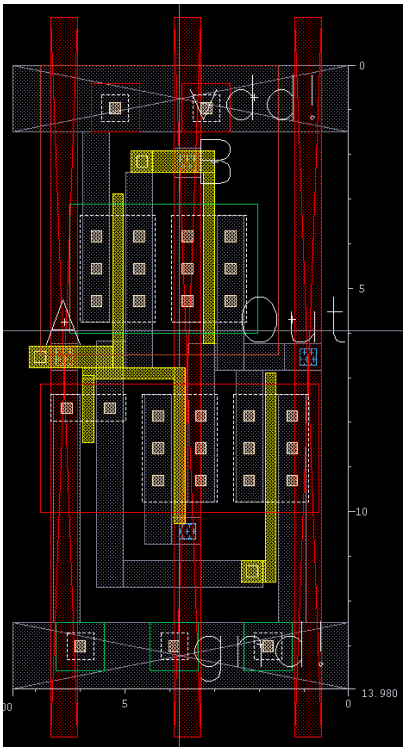

Figure 2.2. Layout of AND gate (Cadence)

OR logic gate

For OR gate, a CMOS inverter, a transmission gate and a PMOS are included in the logic gate. When A is zero, the transmission gate turns on and passes B to output. When A is one, the transmission gate turns off and the PMOS is on for passes A to output. Therefore, the Boolean function is defined as follow:

$$Out = A + B$$

The schematic of OR gate based on CMOS transmission gate is illustrated in Figure 2.3. And the layout of the OR gate is illustrated in Figure 2.4.

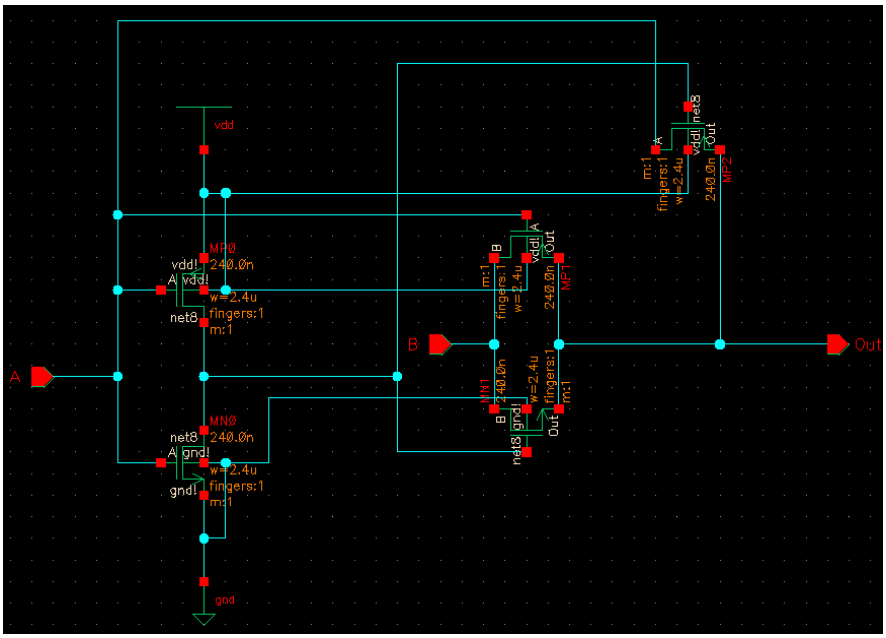

Figure 2.3. Schematic of OR gate (Cadence)

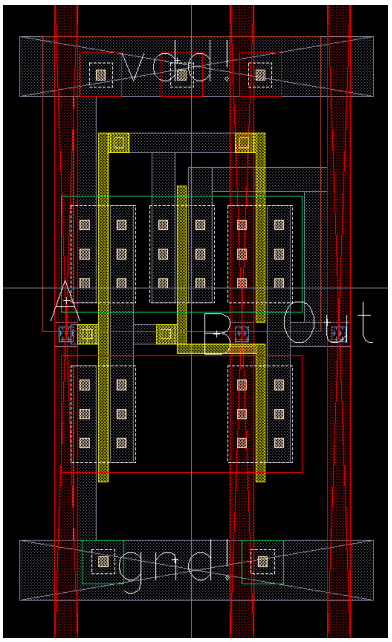

Figure 2.4. Layout of OR gate (Cadence)

XOR logic gate

For XOR gate, one CMOS inverter, a transmission gate, a PMOS and a NMOS are included in the logic gate. The Boolean function is defined as follow:

$$Out = A' \cdot B + A \cdot B'$$

The schematic of XOR gate based on CMOS transmission gate is illustrated in Figure 2.5. And the layout of XOR gate is illustrated in Figure 2.6.

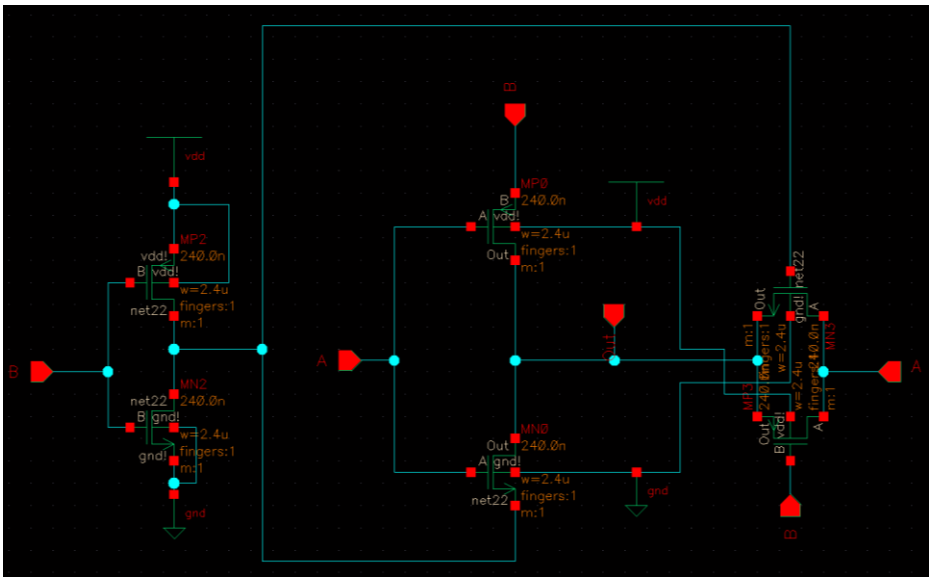

Figure 2.5. Schematic of XOR gate (Cadence)

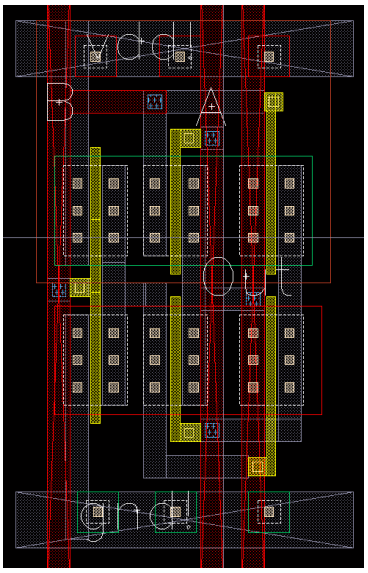

Figure 2.6. Layout of XOR gate (Cadence)

As there is a critical relationship between delay time and extrinsic load capacitance, different sizes of capacitor are used for simulating the delay time. In order to emphasize this relationship, capacitance with 15fF, 120fF, 960fF are used on every basic logic gate in order to simulate the performance of delay time of the logic gate. Three tables which show the size of each transistors, output low-to-high propagation delay and output high-to-low propagation delay are shown in Table 2.1, 2.2 and 2.3 for AND gate, OR gate, XOR gate respectively.

AND gate	MP0	MP1	MN0	MN1	MN2
Width	2.4um	2.4um	2.4um	0.6um	2.4um
Length	0.24um	0.24um	0.24um	0.24um	0.24um
Output low-to-high propagation delay (15fF)		172.17ps	Output high-to-low propagation delay (15fF)		146.999ps
Output low-to-high propagation delay (120fF)		257.72ps	Output high-to-low propagation delay (120fF)		250.95ps
Output low-to-high propagation delay (960fF)		1041.9ps	Output high-to-low propagation delay (960fF)		1014.35ps

Table 2.1 Size specification and propagation delay of AND gate

OR gate	MP0	MP1	MP2	MN0	MN1
Width	2.4um	2.4um	2.4um	2.4um	2.4um
Length	0.24um	0.24um	0.24um	0.24um	0.24um
Output low-to-high propagation delay (15fF)		160.2415ps	Output high-to-low propagation delay (15fF)		115.395ps
Output low-to-high propagation delay (120fF)		405.0443ps	Output high-to-low propagation delay (120fF)		194.972ps
Output low-to-high propagation delay (960fF)		2365.483ps	Output high-to-low propagation delay (960fF)		810.69ps

Table 2.2 Size specification and propagation delay of OR gate

XOR gate	MP0	MP2	MP3	MN0	MN2	MN3
Width	2.4um	2.4um	2.4um	2.4um	2.4um	2.4um
Length	0.24um	0.24um	0.24um	0.24um	0.24um	0.24um
Output low-to-high propagation delay (15fF)		141.116ps		Output high-to-low propagation delay (15fF)		118.855ps
Output low-to-high propagation delay (120fF)		362.3204ps		Output high-to-low propagation delay (120fF)		280.563ps
Output low-to-high propagation delay (960fF)		2164ps		Output high-to-low propagation delay (960fF)		1550ps

Table 2.3 Size specification and propagation delay of XOR gate

Logic Cell Block

Four logic cell blocks are built in this adder, including buffer cell, pre-processing cell, grey cell and black cell.

Buffer cell is used to boost up the signal before going into the buffer block for dealing with the timing of propagation between lower bit and upper bit. It is made by two CMOS inverters. The Boolean function of the buffer cell can be illustrated as:

$$Out = In"$$

The schematic of buffer cell based on CMOS transmission gate is illustrated in Figure 2.7. And the layout of buffer cell is illustrated in Figure 2.8.

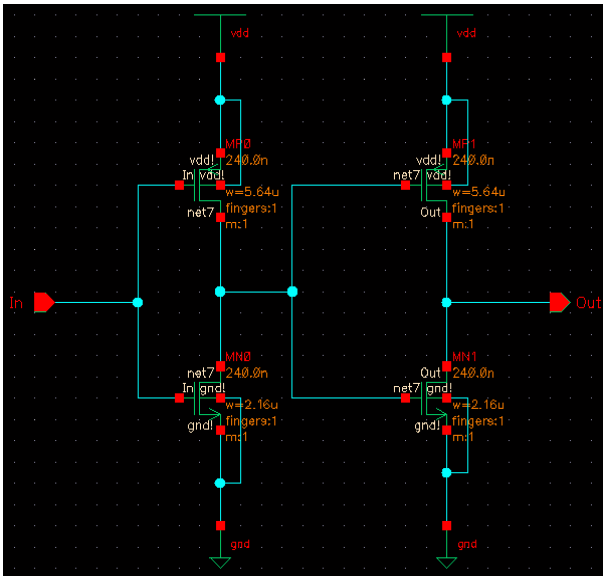

Figure 2.7. Schematic of buffer cell (Cadence)

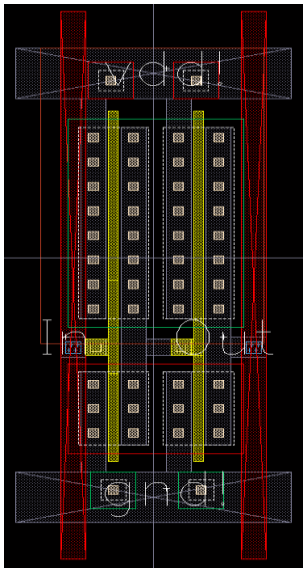

Figure 2.8. Layout of buffer cell (Cadence)

Pre-processing cell is used to compute the generate and propagation bit. It is a combination of an AND gate and XOR gate. The Boolean function of the pre-processing cell can be illustrated as:

$$P = A' \cdot B + A \cdot B' \quad ; \quad G = A \cdot B$$

The schematic of the pre-processing cell based on CMOS logic is illustrated in Figure 2.9. And the layout of the pre-processing cell is illustrated in Figure 2.10.

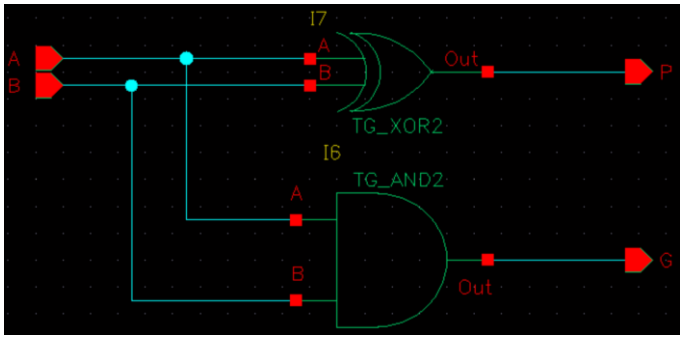

Figure 2.9. Schematic of pre-processing cell (Cadence)

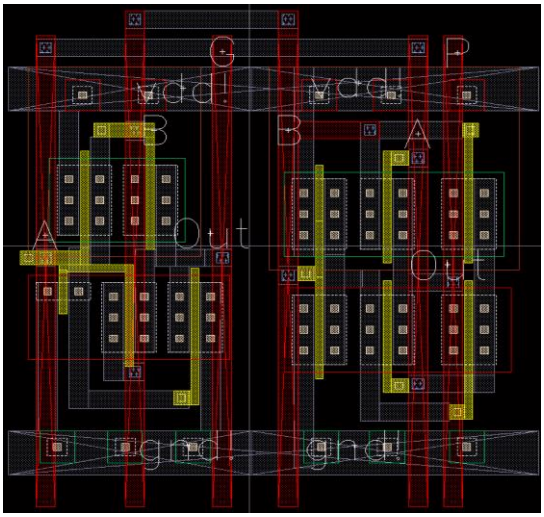

Figure 2.10. Layout of pre-processing cell (Cadence)

Gray cell is used to provide logical calculation. It is a combination of a AND gate and OR gate. The Boolean function of the grey cell can be illustrated as:

$$G_j = (P_k \cdot G_{k-1}) + G_k$$

The schematic of gray cell based on CMOS transmission gate is illustrated in Figure 2.11.

The layout of gray cell is illustrated in Figure 2.12. And it is obvious that the OR cell inside the gray cell is different from Figure 2.4.

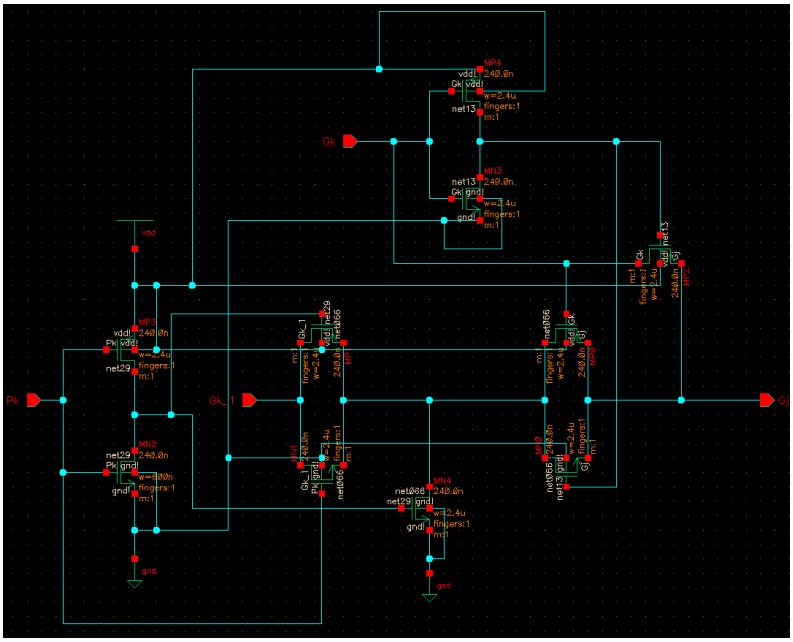

Figure 2.11. Schematic of grey cell (Cadence)

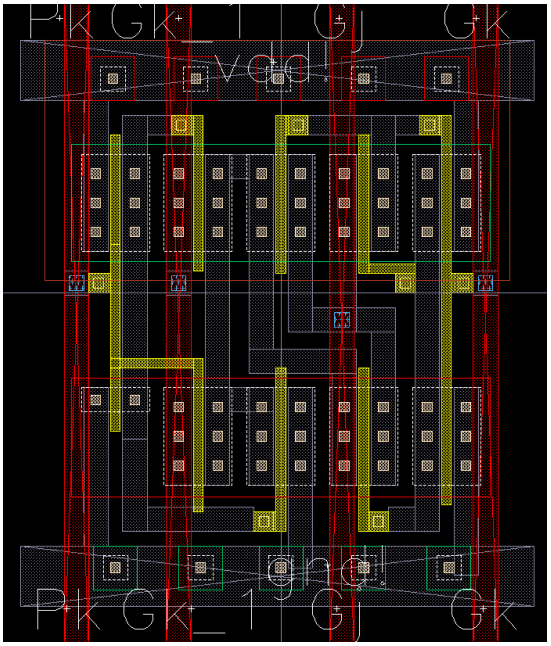

Figure 2.12. Layout of grey cell (Cadence)

Black cell is used to provide both logical calculation and propagation. It is a combination of two AND gates and a XOR gate. The Boolean function of the grey cell can be illustrated as:

$$P_j = P_k \cdot P_{k-1} \quad ; \qquad G_j = (P_k \cdot G_{k-1}) + G_k$$

The schematic of black cell based on CMOS transmission gate is illustrated in Figure 2.13. And the layout of black cell is illustrated in Figure 2.14.

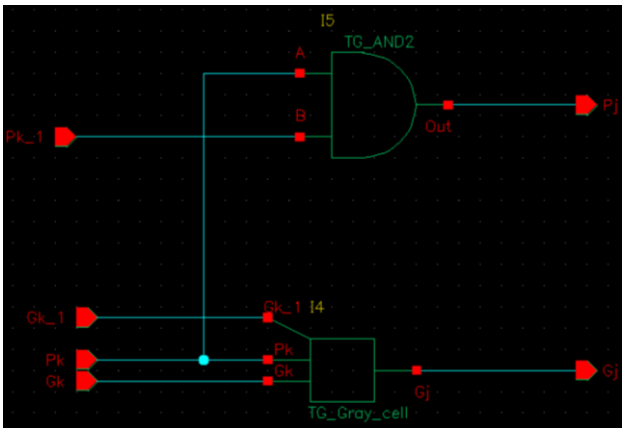

Figure 2.13. Schematic of black cell (Cadence)

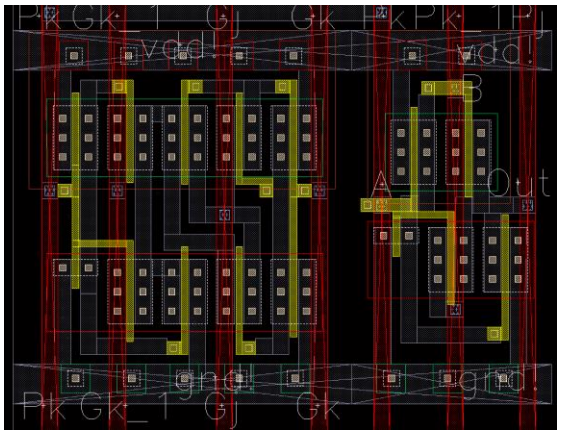

Figure 2.14. Layout of black cell (Cadence)

Two tables which show the size of each transistors, output low-to-high propagation delay and output high-to-low propagation delay are shown in Table 2.4 and 2.5 for buffer cell and grey cell respectively. As black cell is the combination of a grey cell and a AND gate, the specification of transistor and the corresponding propagation delay is just the summation of the data from the grey cell and the AND gate. Therefore, the table of specification for black cell is not included in the report.

Buffer cell	MP0	MP1	MN0	MN1
Width	5.64um	5.64um	2.16um	2.16um
Length	0.24um	0.24um	0.24um	0.24um
Output low-to-high propagation delay (15fF)	122.5698ps	Output high-to-low propagation delay (15fF)		126.87ps
Output low-to-high propagation delay (120fF)	222.159ps	Output high-to-low propagation delay (120fF)		237.73ps
Output low-to-high propagation delay (960fF)	990.30806ps	Output high-to-low propagation delay (960fF)		1098.65ps

Table 2.4 Size specification and propagation delay of buffer cell

Grey Cell	MP0	MP1	MP2	MP3	MP4
Width	2.4um	2.4um	2.4um	2.4um	2.4um
Length	0.24um	0.24um	0.24um	0.24um	0.24um
/	MN0	MN1	MN2	MN3	MN4
Width	2.4um	2.4um	0.6um	2.4um	2.4um
Length	0.24um	0.24um	0.24um	0.24um	0.24um
Output low-to-high propagation delay (15fF)		119.257ps	Output high-to-low propagation delay (15fF)		203.149ps
Output low-to-high propagation delay (120fF)		314.347ps	Output high-to-low propagation delay (120fF)		342.66ps
Output low-to-high propagation delay (960fF)		2188.7ps	Output high-to-low propagation delay (960fF)		1422.6ps

Table 2.5 Size specification and propagation delay of grey cell

Based on the result of the simulation result, it is obvious that when the value of extrinsic load capacitance increases, the delay time of the basic logic gate increases. The simulation result can be supported by the equation shown below. As the delay time of a logic gate can be calculated by the following equation:

$$T_d = 0.69R\ (C_{in} + C_{ext})$$

Equation 2.1. Equation between time delay, resistance and capacitance

Td is the time delay, the C_{in} is the intrinsic Capacitance, C_{ext} is the extrinsic Capacitance R is the resistance of the circuit.

It is obvious that when a larger value of extrinsic load capacitance is used, a longer delay time is resulted. Therefore, a conclusion can be drafted with the above equation. When a more fan-in gate is used, this equation is applicable.

Structure of 32-bit brent-kung adder

The Brent-Kung adder is formed by three different parts, Bitwise PG Logic, Group PG Logic and Sum Logic. For the overall design, it can be reviewed from the figure 3.2-3.4. The input A[0:31] and B [0:31] were connected with bitwise PG logic , C_{in} was connected to the sum logic input with the G0 and the S[0:31] and C_{out} were connected to the Sum logic as an output.

Bitwise PG Logic

For the Bitwise PG Logic, it has been mentioned from the above logic description, which is the pre-processing cell. The function of this logic gate is used to process the A and B signal to be P and G signals for further processing.

Group PG Logic

After the P and G signal are generated, the signal passes to the Group PG Logic for further processing. The processing is based on the bit number, which different group cells are used.

The Brent Kung adder is designed to get the prefix computation with the usage of the current and previous P and G signal. In other words, the gray cell is used to capture the Generate bit from the previous one and compute the carry bit. It means that no carry bit is needed after computation is done in the first bit. Therefore, the performance of Brent-Kung adder is faster than a normal carry ripple adder.

To perform the prefix computation, the black cell and the gray cell are used in this Group PG logic. For example, the Black cell under input bit 3 uses the Propagation bit and the Generation bit computed from the previous input bit (ie. bit 2) and the target input bit (ie. bit 3) for computing the Propagation bit and the Generation bit of the output bit (ie. bit 3). Next, the black prefix is computed by the carry bit from the bit before (G4:3 and P4:3). Meanwhile, the gray cell is used to get the prefix bit from bit 1, which is the prefixed Generate bit G2:0. When the signal from bit 3 is computed by the black cell, the output of the grey cell is the Carry out bit of the next sum bit (ie. bit 4). From the result of the output of Group PG logic, Brent-Kung adder uses less stages to calculate a 32-bit value (ie. $2\log_2(32-1)$ = 9 stages) compared with the Carry ripple adder, which uses 32 stages to propagate the carry out signal.

Compared with the Carry ripple adder, the Brent-Kung adder uses more logic gates in the design. From the fact above, the buffer is necessary to maintain a low load capacitance. Also, the buffer requires to be discovered after the critical Propagate and Generate bit connection. Meanwhile, another functionality of the buffer is reducing heavy loading of the circuit, which is defined as a large number of usages of black cells and gray cells.

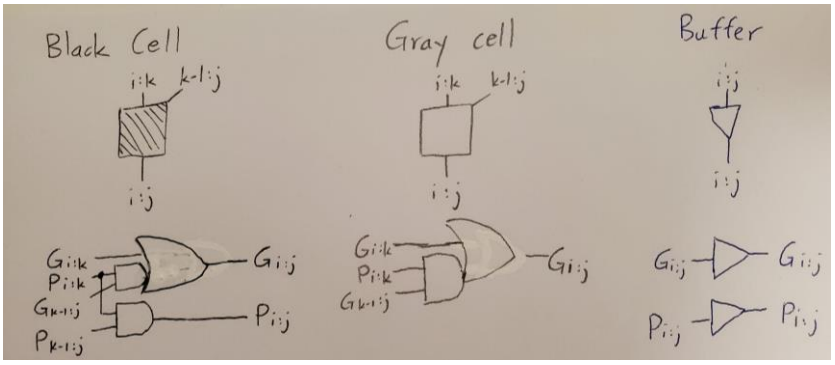

Figure 3.1: Group PG Logic cell (self-drawn)

Sum Logic

After the computation process in Gray cell, the bit G_j becomes the Carry bit. It is used for the sum logic or act as the carry of the higher bits in the circuit for the Gray cell for computing another G_j. In order to compute the sum bit, the sum logic is used, which is a XOR gate with input G_{n-1} and P_n in the bit n. P_n is the output of the Bitwise PG Logic and G_{n-1} is the output computed from a bit before the sum bit. In exclusion from the normal calculation of sum logic, the bit C_{in} and P_0 are the first two input bits of XOR gate and calculate the output sum bit S_0.

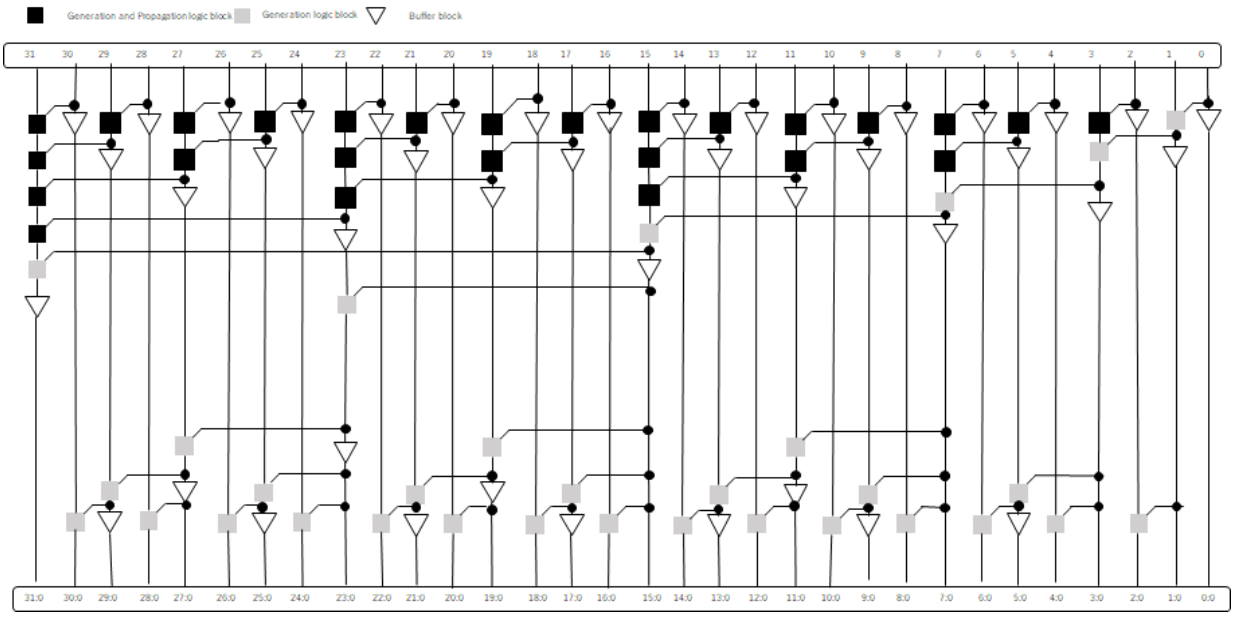

Figure 3.1. 32-bit Brent-Kung adder in cell block level (Drawn by PowerPoint)

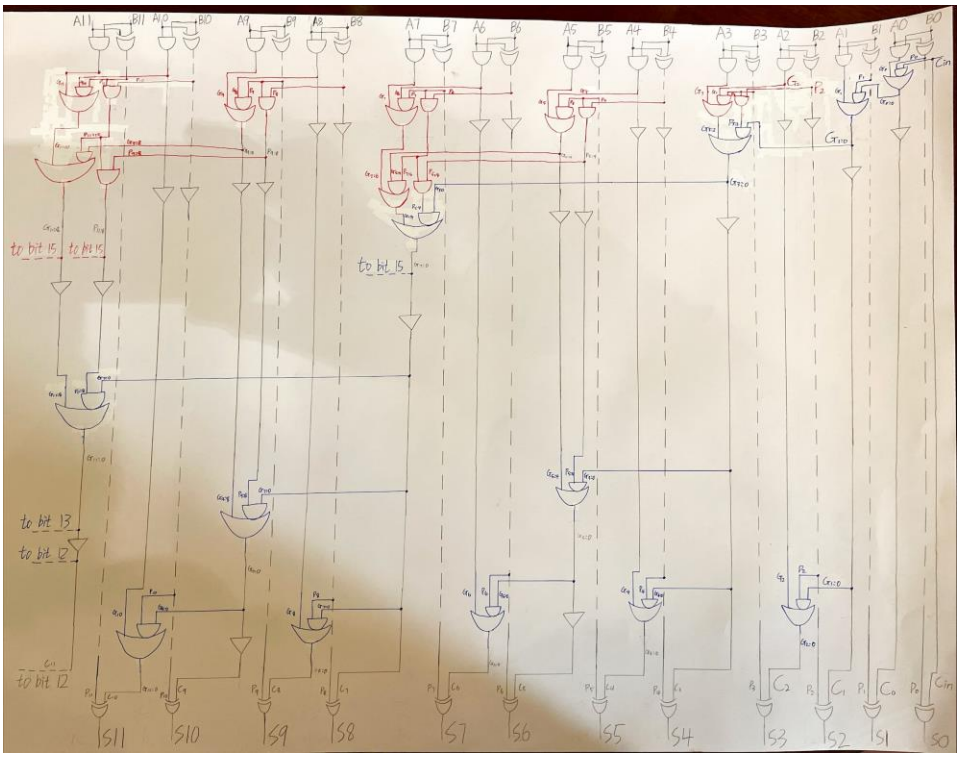

Figure 3.2. bit 0 – bit 11 of 32-bit Brent-Kung adder in basic logic gate level (Self-drawn)

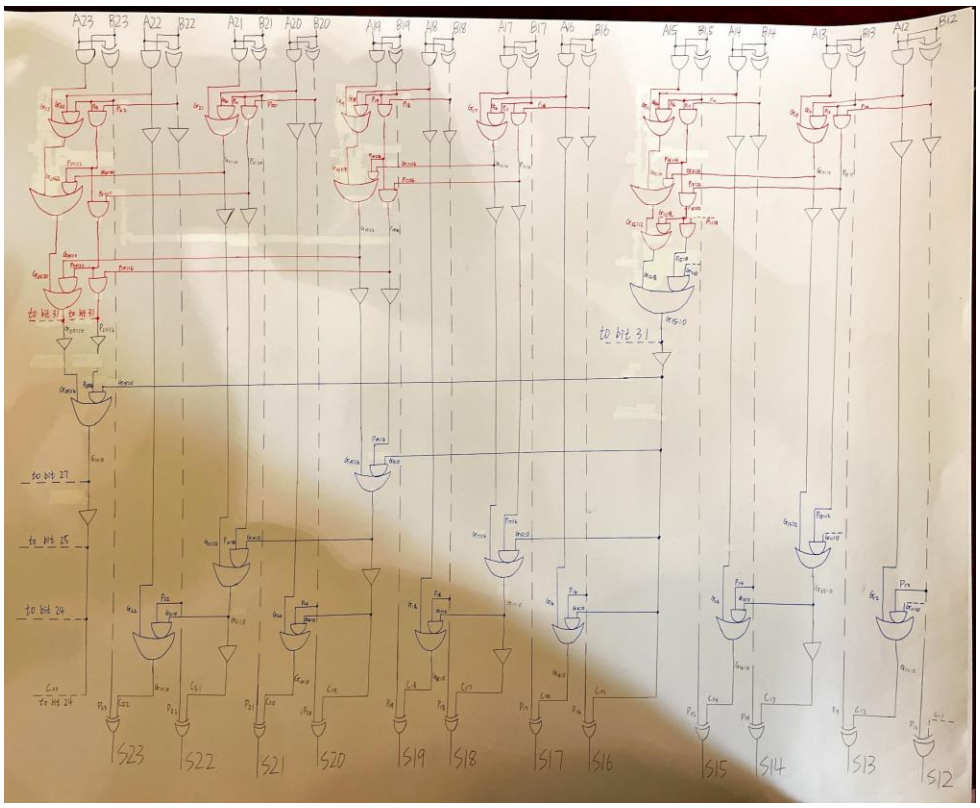

Figure 3.3. bit 23 – bit 12 of 32-bit Brent-Kung adder in basic logic gate level (Self-drawn)

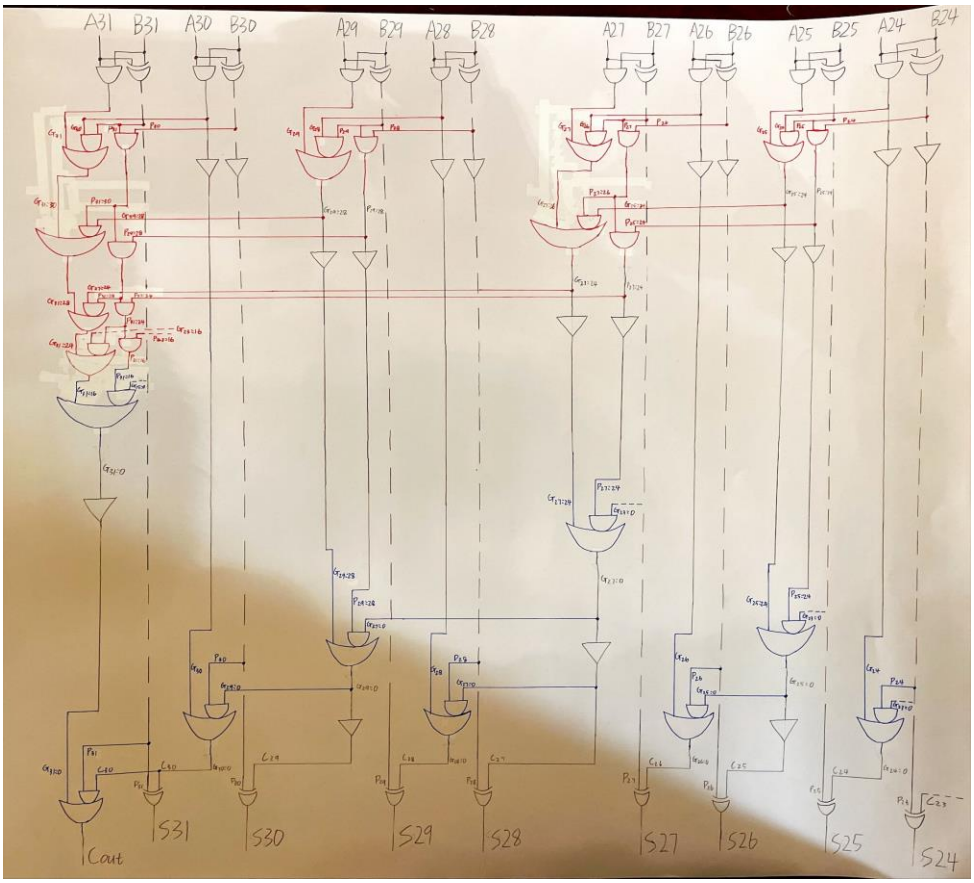

Figure 3.4. bit 31 – bit 23 of 32-bit Brent-Kung adder in basic logic gate level (Self-drawn)

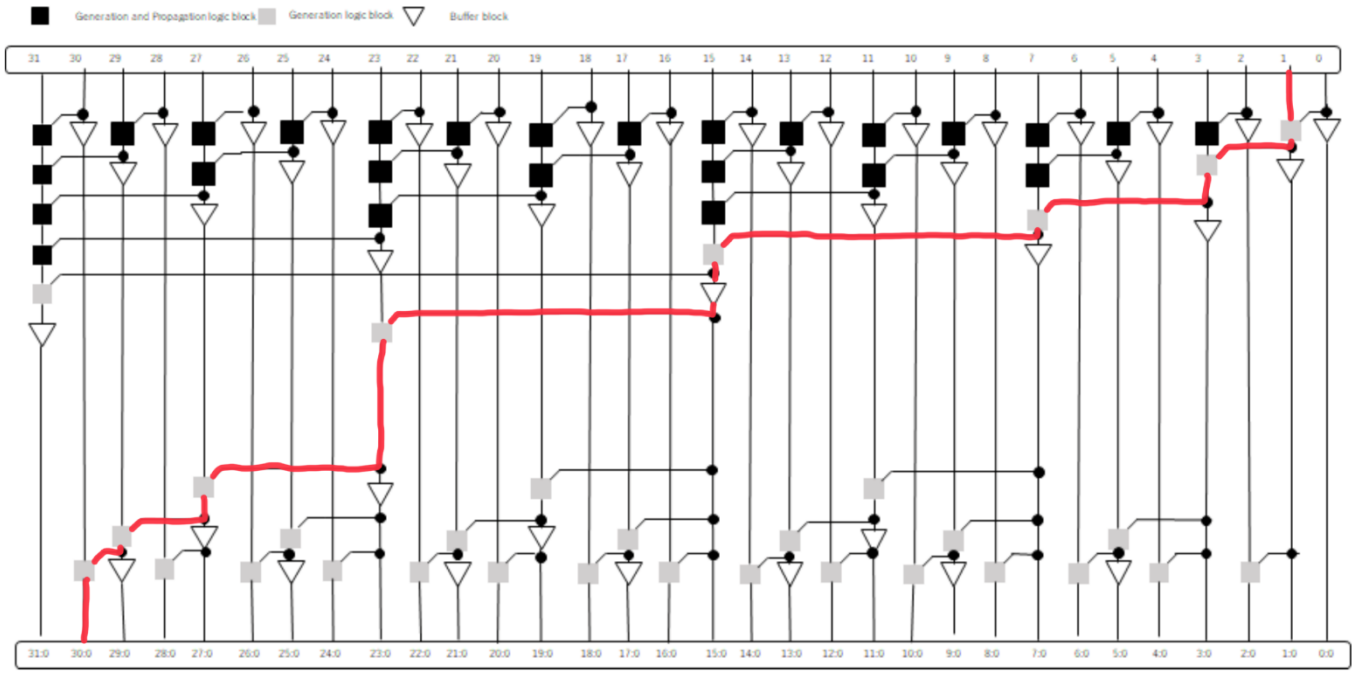

Figure 3.5. Critical path of the 32-bit Brent-kung adder (Drawn by PowerPoint)

Figure 3.5 shows the critical path of the 32-bit Brent-kung adder. The path starts from input bit 1 and terminates at the carry bit. The path can be proved by the following input cases. When bit 1 of number A is zero and bit 0 of number A is one, both bit 0 and bit 1 of number B result one from the logic computation. Therefore, the corresponding signal must pass the preprocess cell XOR, gray cell AND gate and the gray cell OR gate. After the computation of the previous three cell blocks. there are 8 gray cells for further computation to get the carry out for bit 31. Therefore, this path is the critical path for our design.

In order to trigger the critical path, the input of number A is 0x1 and the input of number B is all 1's and maximum time delay is resulted under this path. Table 3.1 shows the simulation result in schematic level with the number mentioned above.

Number A	Number B	Result	Delay Time (S30)
0x1	0xFFFFFFFF	0x100000000	8.229ns

Table 3.1. Simulation result using two 32bits number for triggering critical delay path

32-bit Brent-Kung Adder Layout Area and Delay

In this part, the layout of the 32-bit Brent-Kung adder is discussed. The layout area and floor plan are illustrated. Post-layout propagation delay with capacitance, resistance and both capacitance and resistance extractions is done. Simulation results are given. Lastly, the importance of R&C extractions and the pros and cons of simulation tools on both schematic level and layout level are discussed.

The design of the floor plan is illustrated in Figure 4.1. For the floor plan design, a line shape design is used to build up the Brent-Kung Adder. Using a line shape design, the layout is flexible to wire input or output at the top or bottom without changing the length of wire. Also, bit 31 is designed to be the closest bit from bit 15 which helps the wire be short at the long path connection. VDD and GND are designed for connecting at both ends of the design. Therefore, although the power supply of the adder is fixed on the left or right of the floor plan, either side is able to connect to the design.

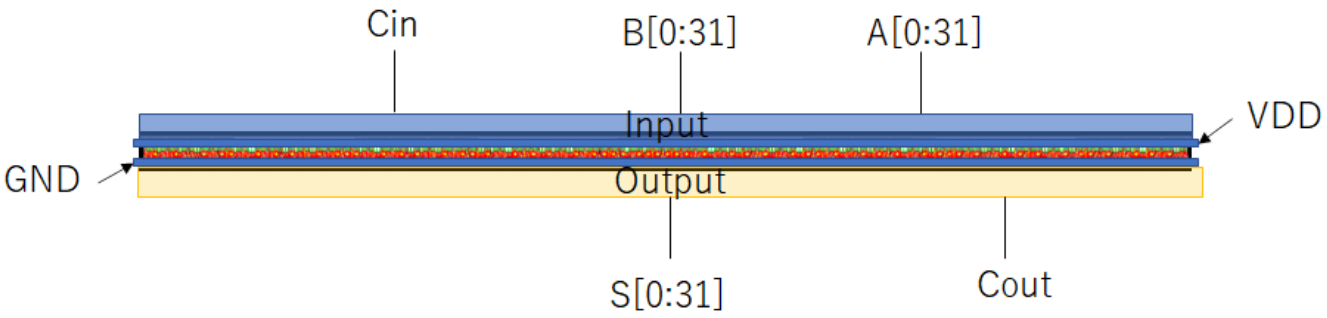

Figure 4.1. Floor plan of the 32-bit Brent-Kung adder (Cadence)

To optimize the design for shorter delay, transmission gates are used to design the logic cell. Also, the number of transistors used is minimized in the design so that the area of the design is small. Moreover, from the OR cell inside the gray cell, it is side-rotated and compressed by moving the 2.4um NMOS near to the 0.6um NMOS. Therefore, this layout of gray cell is optimized in terms of area. Furthermore, since the transmission gate is used, the width for both PMOS and NMOS only requires to be scaled with equal ratio to optimize the area used. In terms of routing, pins for each of the cell are located top or the bottom of the cell. The area of the adder is then minimized by using wire on top or bottom of the cell only. If the cell is routed inside the cell, it would take a large area to finish the adder design.

The layout of the whole adder is shown in Figure 4.2.

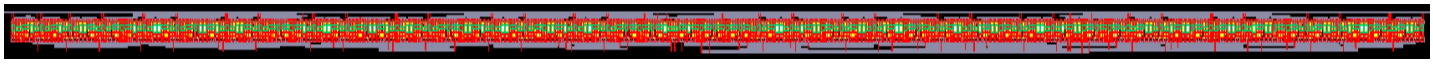

Figure 4.2. Layout of 32-bit Brent-Kung Adder (Cadence)

The area of the adder is equal to:

$$(2278.934+73.08) * (0 +29.403) =2352.014 *29.403 = 69156.267642um^2$$

In order to see the performance of the adder, the delay time of the adder must be calculated. Three different parasitic capacitance extractions, which includes R, C and R&C, are simulated with post-layout simulation. Also, the propagation delay on layout level is simulated with post-layout simulation. The results of the simulation are shown in table 4.1.

Parasitic resistor extraction	Parasitic capacitor extraction	Parasitic resistor and capacitor extraction	Output propagation delay in layout level
10397	684	19541	11860ps

Table 4.1. Simulation result of post-propagation delay

As seen from the simulation results, the propagation delay of the 32-bit Brent-Kung adder in layout level is greater than the propagation delay in schematic level. There are a number of reasons which show the result.

The major reason is that some parameters cannot be shown in schematic level. it is obvious that there is less information provided in the schematic compared with layout. For example, when the design is drawn in layout, there are a lot of connected wires and those connected wires generate extrinsic capacitance in the design so that the delay time of the design is affected by the layout design. After the whole design is drawn, the size of the design depends on the designer drawing, such as floor plan, the width of the wires, the length of connection between two logic gates. As a result, the simulation from the schematic can only provide a big picture for the designer for further development.

Performance of 32-bit Brent-Kung adder

The performance of Brent-Kung adder is computed using the formula below:

$$\text{Performance} = 1 / [\text{post-layout delay} * \text{area}]$$

$$\text{Performance} = 1 / [11860\text{ps} * 69156.267642\text{um}^2]$$

$$\text{Performance} = 1 / [0.000000011860\text{s} * 69156.267642\text{um}^2]$$

$$\text{Performance} = 1 / [0.000000011860\text{s} * 69156.267642\text{um}^2]$$

$$\text{Performance} = 1219.224\ \text{s}^{-1}\ \text{um}^{-2}$$