8-bit Microprocessor Data-path Using L-Edit Tool For Layout Custom Design

Mohamed Alaa Issa

Mohamed Isa El Dasher

Marawan Moutsafa Moustafa

Mustafa Mahmoud Khairalah

Noha El Sha'arawy

VLSI Course Poroject

Dr. Mohamed Morsy

Electronics and Communications Department

Alexandria University

Summary

1 Project Summary

Our project involves the design of an 8-bit microprocessor data-path including 8-byte dual port memory, ALU and barrel shifter using CMOS VLSI technology on Tanner EDA toolchain. The project summarizes thelayout design techniques we havelearnt throughout the course.

2 Overall Layout

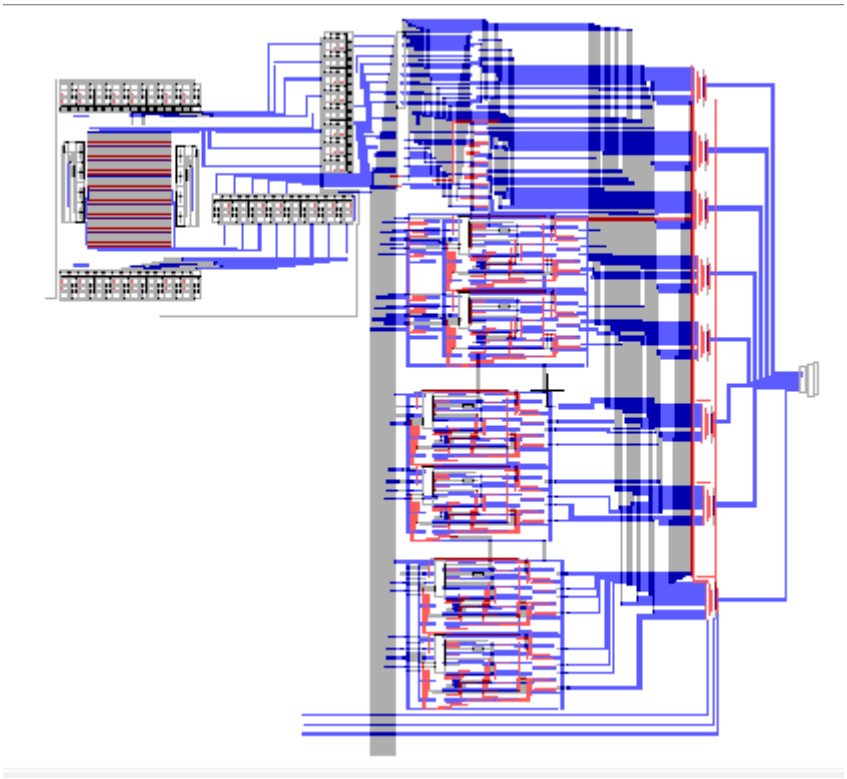


Figure 2.1. Overall Layout

3 Datapath overview

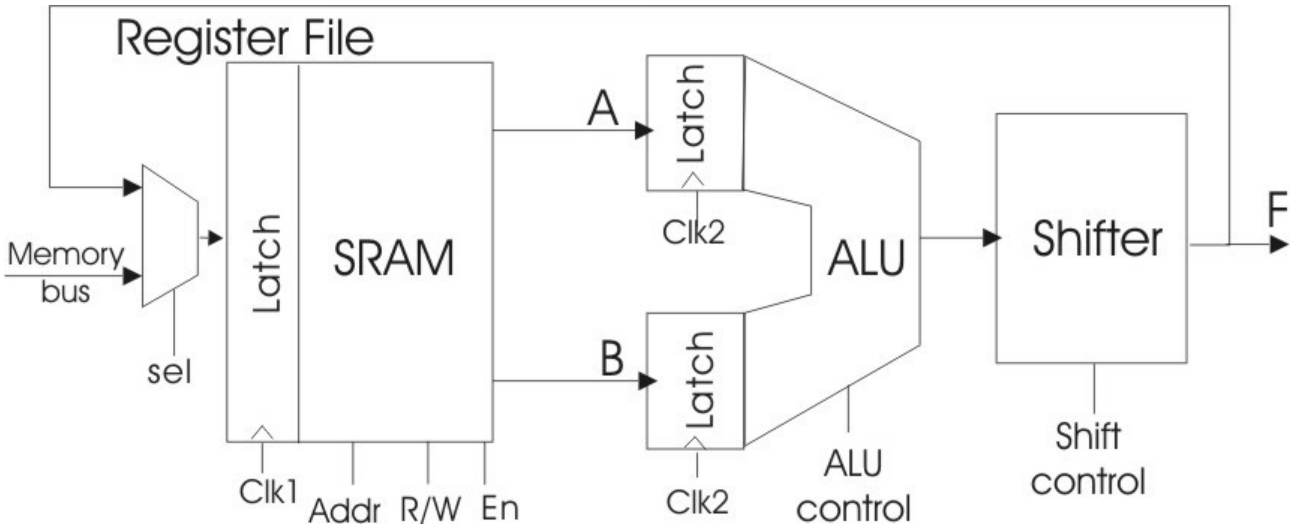


Figure 3.1. Datapath Overview

4 Registers

In this project, we use three 8-bit registers; one used with the SRAM in the Register file, & another two for the ALU. The 8-bit register is constructed using 8 positive edge 1-bit D-flipflops as shown in figure 4.1.

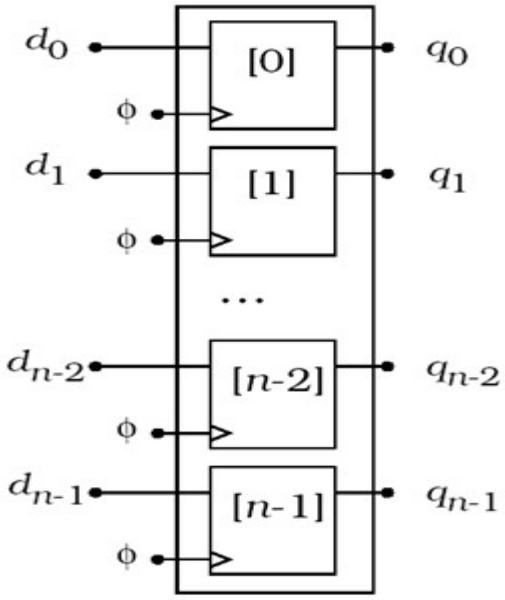


Figure. 4.1 An n-bit register

We started by constructing the building block, the positive edge D-flipflop using 6 NAND gates, as shown in figure 4.2. The layout is shown in figure 4.3 & a test simulation for the D-flipflop is shown in figure 4.4.

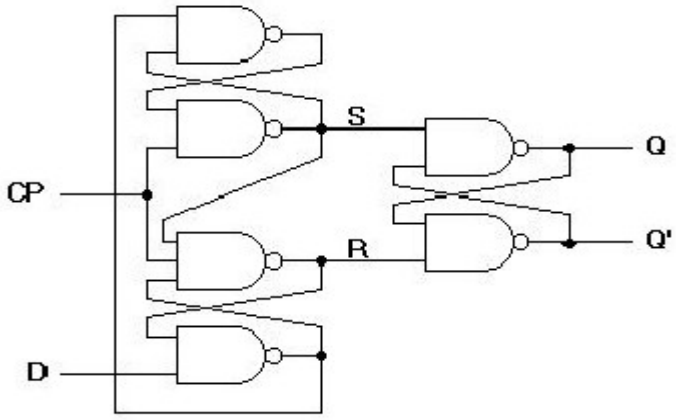


Figure 4.2. D-Type positive-edge triggered flip flop

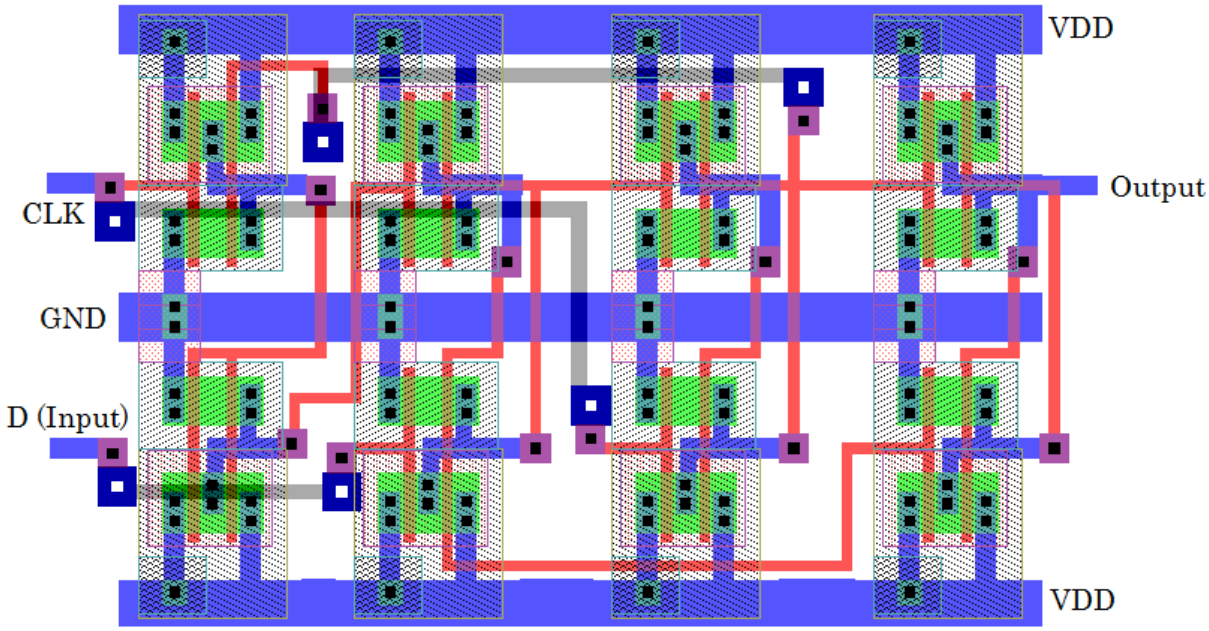


Figure 4.3. Layout of D flip flop

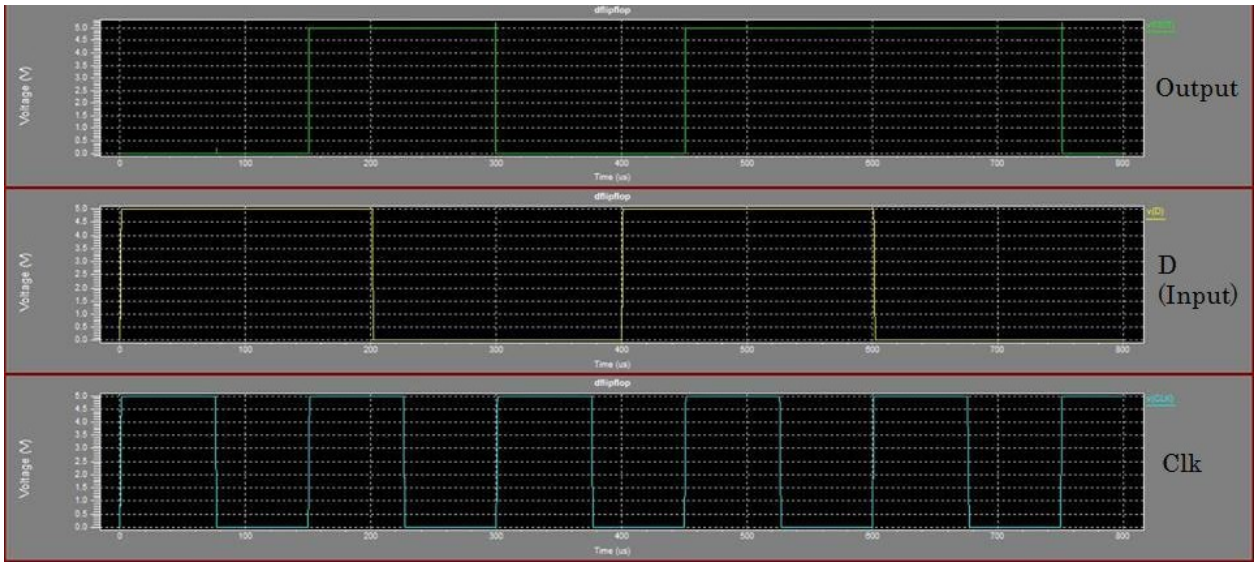


Figure 4.4. Simulation Results

Using the D-flipflop as the building block, We use 8 D-flipflops and connect their clock inputs, their VDDs & their GNDs together. The whole layout of the register is shown in figure 4.5.

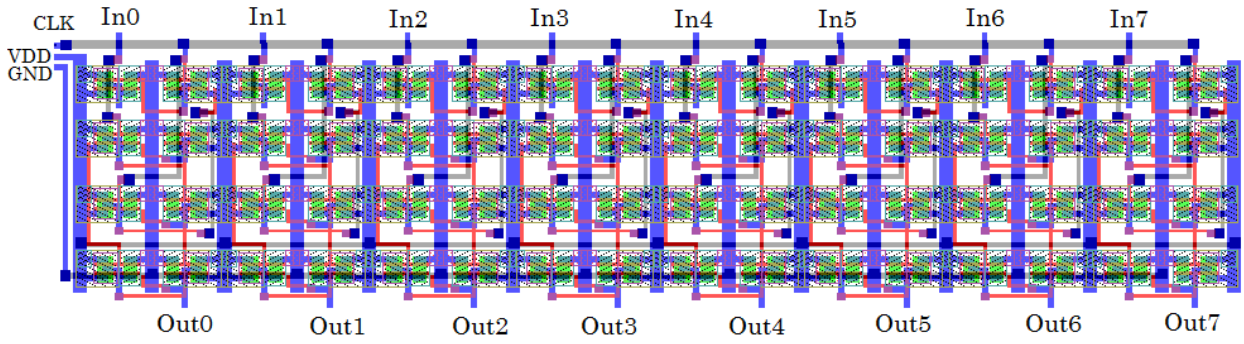


Figure 4.5. 8-Bit register layout

5 Barrel Shifter

A barrel shifter is a digital circuit that can shift a data word by a specified number of bits in one clock cycle. It can be implemented as a sequence of multiplexers (mux.), and in such an implementation the output of one mux is connected to the input of the next mux in a way that depends on the shift distance, a generic schematic of such implementation is shown in figure 5.1.

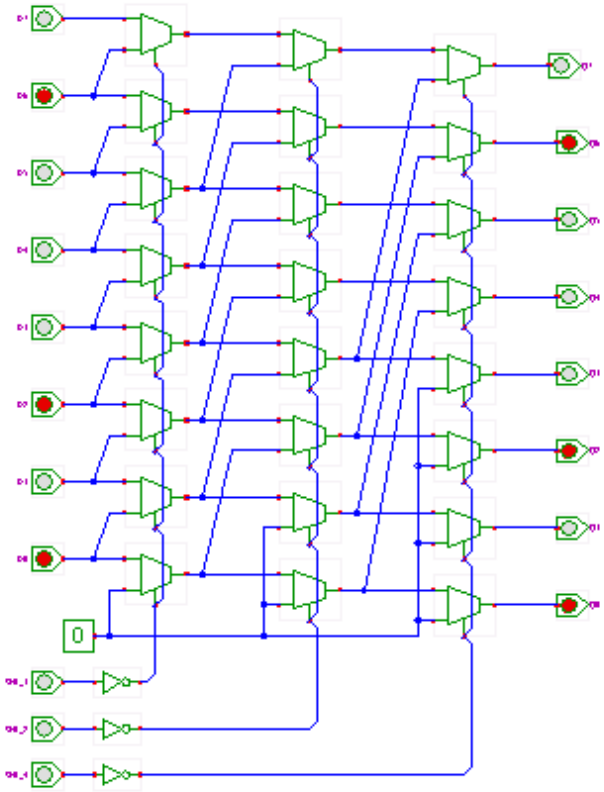


Figure 5.1. Barrel Shifter Circuit

The Barrel Shifter's function is to receive the 8-bit data from the ALU & do one of the following 5 functions according to the shift control bits:

Pass (no shift).

- Shift left by 1 bit.
- Shift left by 2 bits.
- Shift right by 1 bit.
- Shift right by 2 bits.

Our Barrel Shifter consists of 3 stages (1st stage's layout is shown in figure 5.2, 2nd or 3rd stages' layout is shown in figure 5.3):

- The 1st stage consists of 9 MUXs, taking 10 different inputs (the 8 bits from the ALU with 2 '0' bits) & giving 9 outputs. The 1st stage outputs the 8 received bits with an extra '0' bit (that will be used for the shifting).
- The 2nd stage consists of 8 MUXs, taking the outputs from the 1st stage with another 2 '0' bits & performing the second shift to the bits.
- The 3rd stage is the same as the 2nd stage, consisting of 3 MUXs also. According to its selector bit, it takes the output of either stage 1 or stage 2. This is used so that it would be possible to pass the 8 bits as it is.

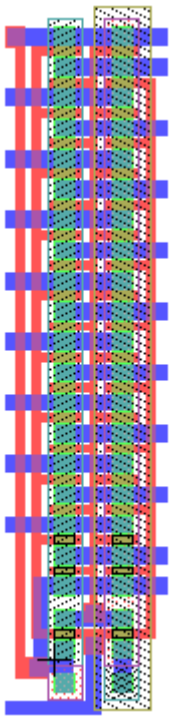


Figure 5.2.Layourt of the first stage

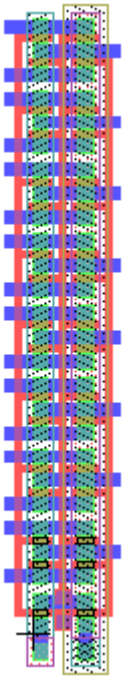


Figure 5.3. Layout of the second stage

The whole barrel shofter layout is shown in figure 5.4.

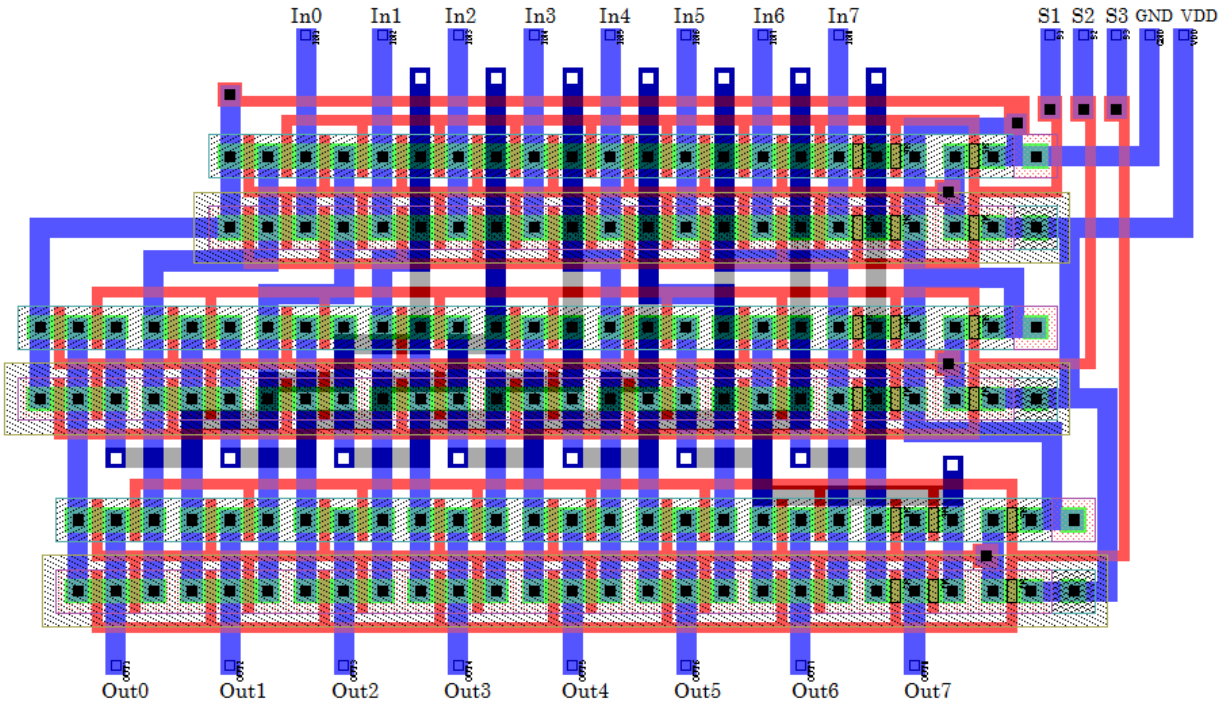


Figure 5.4. Layout of the whole barrel shifter

We simulated the Barrel Shifter to make sure it works. One of the examples is shown in figure 5.5 & figure 5.6. The input bits are 10100011 & the selectors are 110 (shift two bits to the left). The output will be 10001100.

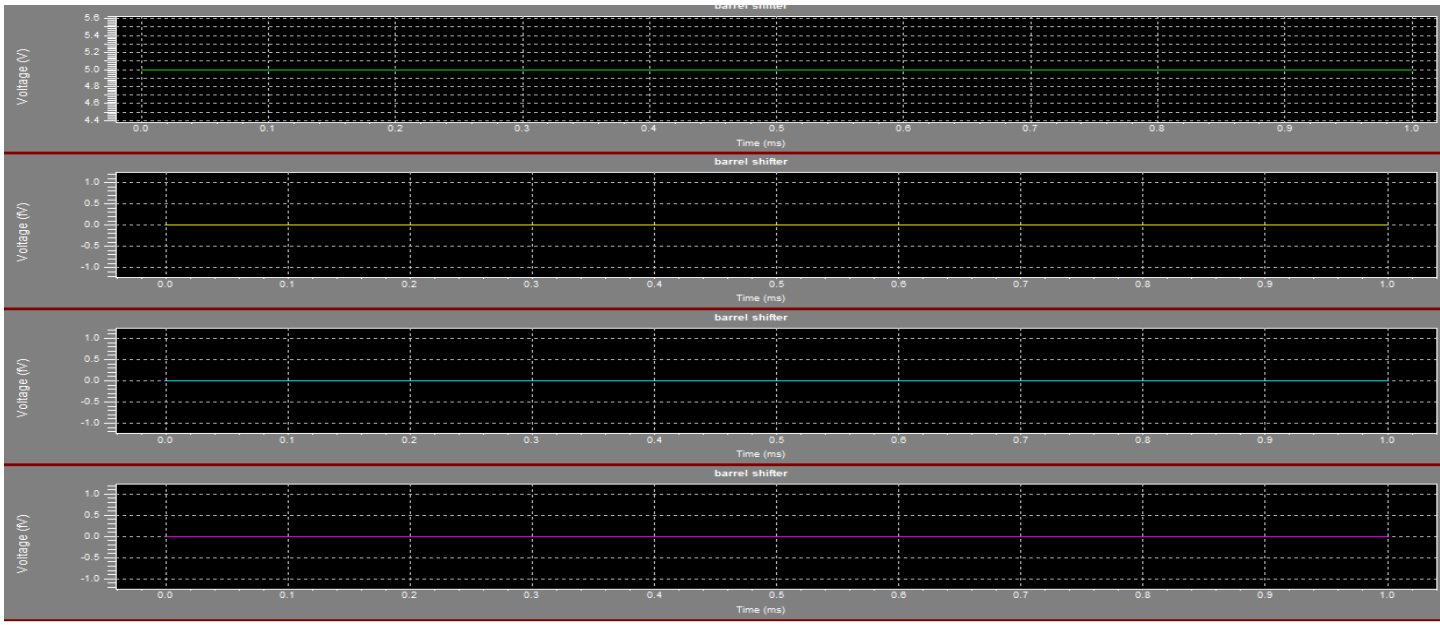


Figure 5.5. Most significant 4 bits of the output

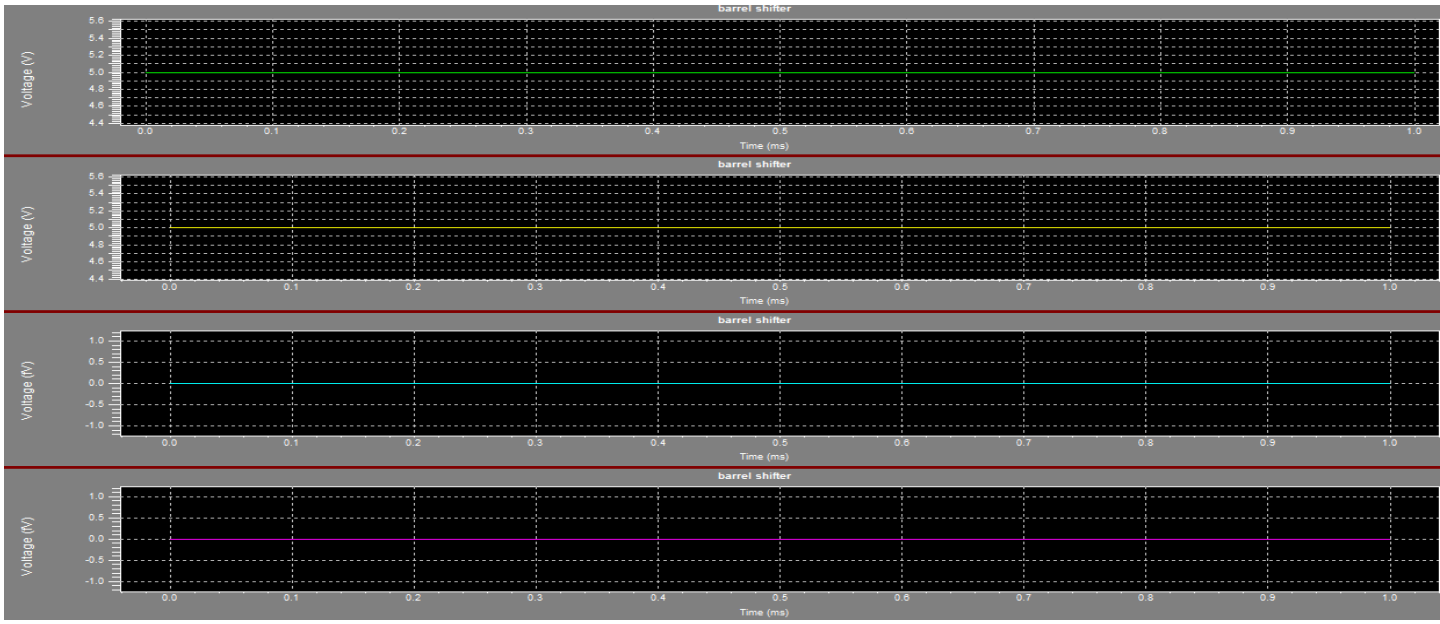


Figure 5.6. Least significant 4 bits of the output

The functionality according to the selectors is shown in the table 5.1.

S1	S2	S3	
0	0	0	shift2 right
0	0	1	shift1 right
0	1	0	shift1 left
0	1	1	shift1 right
1	0	0	shift1 right
1	0	1	pass
1	1	0	shift2 left
1	1	1	pass

Table 5.1. Shifter control bits

6 SRAM (Register File)

A register file is an array of registers inside the microprocessor. It is designed by tiling a single SRAM cell. As it is required to design a dual port register file we will use 8T SRAM Cells. However, first we will explain the modes of operation on a basuc 6T SRAM Cell as they have the same operating modes. They differ only in the number of ports.

SRAM Cell

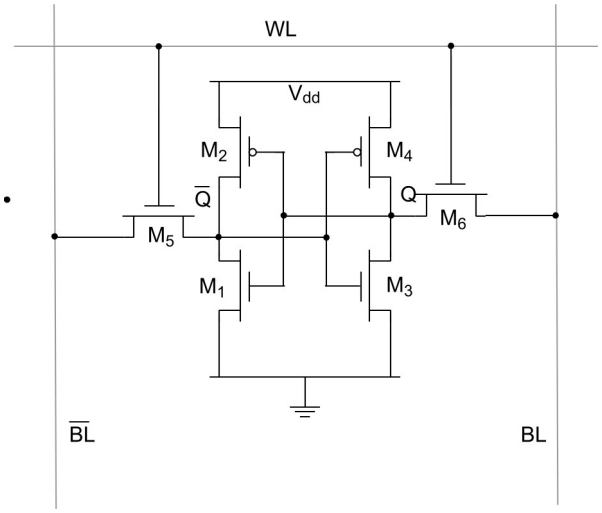


Figure 6.1. Basic 6T single port SRAM Cell

The SRAM Cell has three modes of operation:

1. Hold: both the word-line is at 0 V so the inner latch keeps its old value.
2. Read: the word-line is at VDD, as well as both bit-lines, according to the value stored in the inner latch, one of the bit-lines is pulled to 0 V.
3. Write: the word-line is at VDD. The bit-lines are driven by the value to be stored.

The transistors in the SRAM cell need to be properly sized in order to allow the latch to flip in the write mode and to hold its value in the read mode. This ratio is approximately M1:M2:M5 = 5 : 2 : 1.

The SRAM circuit consists of three main blocks:

6.1 Dual Port SRAM Cell

The basic 8T SRAM Cell consists of two cross-coupled inverters, two word-lines for each output port and four bit-lines; two for each port. It is shown in figure 6.2.

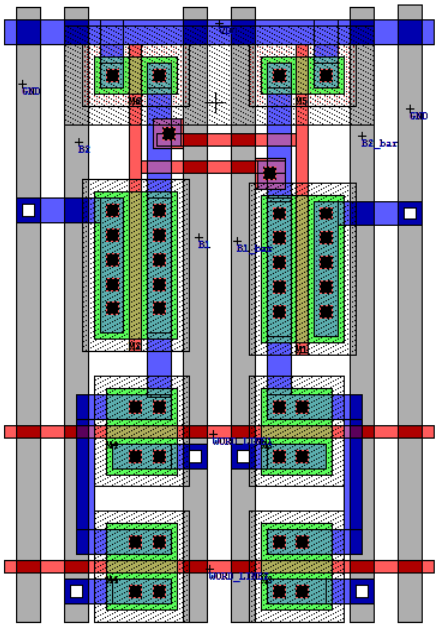


Figure 6.2. Dual-Port 8T SRAM Cell Layout

6.2 Bit-line conditioning

Consists of a pair of NMOS transistors to charge the bit-lines to VDD during the first half cycle. It is shown in figure 6.3.

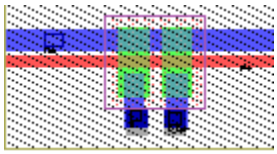


Figure 6.3. Bit-line Conditioning Circuit

6.3 Address Decoder

6.3.1 The truth table and gate-level schematic for address decoder

Before we show the layout of the, we must provide the truth table and gate-level circuit diagram we used to design it. The truth table is table 6.1. The schematic is shown in figure 6.4.

A	B	C	W0	W1	W2	W3	W4	W5	W6	W7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0

1	1	1	0	0	0	0	0	0	0	0

Table.6.1. Address Decoder Truth Table

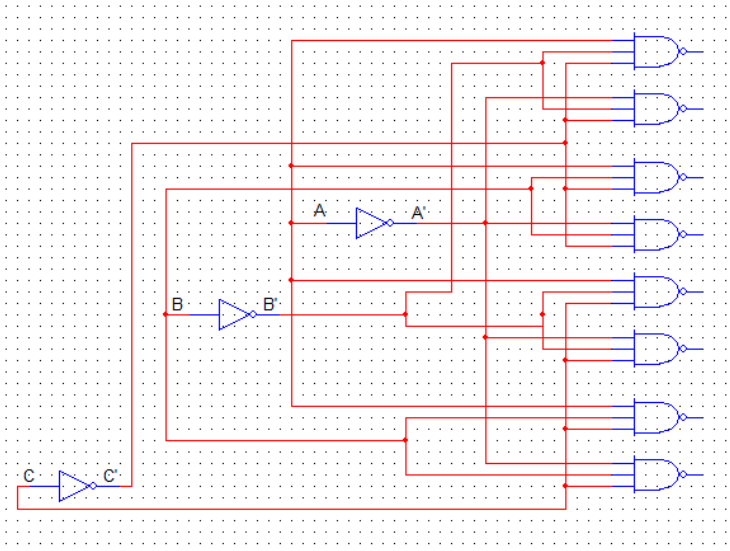


Figure 6.4. Address Decoder Schematic

6.3.2 Layout

The layout is shown in figure 6.5. It is worth mentioning that this design is duplicated, once for each output port.

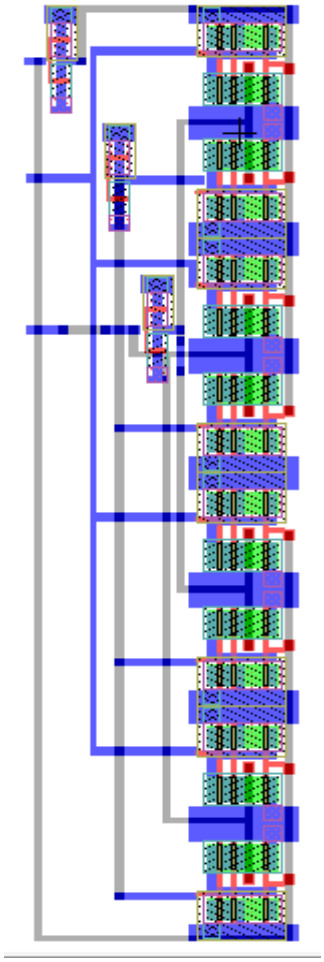


Figure 6.5. Address Decoder

Figure 6.6. represents the whole SRAM design

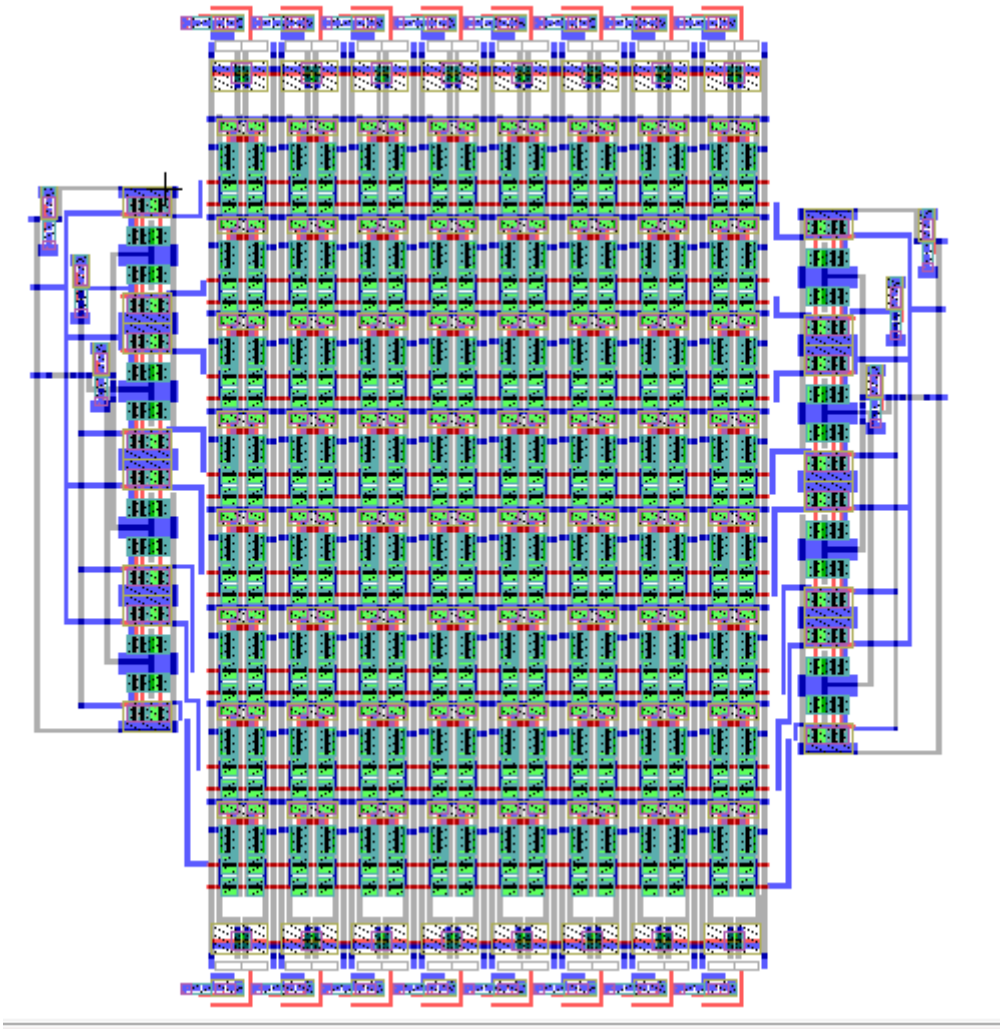


Figure 6.6. 8-Byte SRAM

7 ALU

7.1 Background and schematics

To select between the 8 functions of ALU we used eight 8x1 multiplexers, one for each bit of data as shown in figure 7.1.

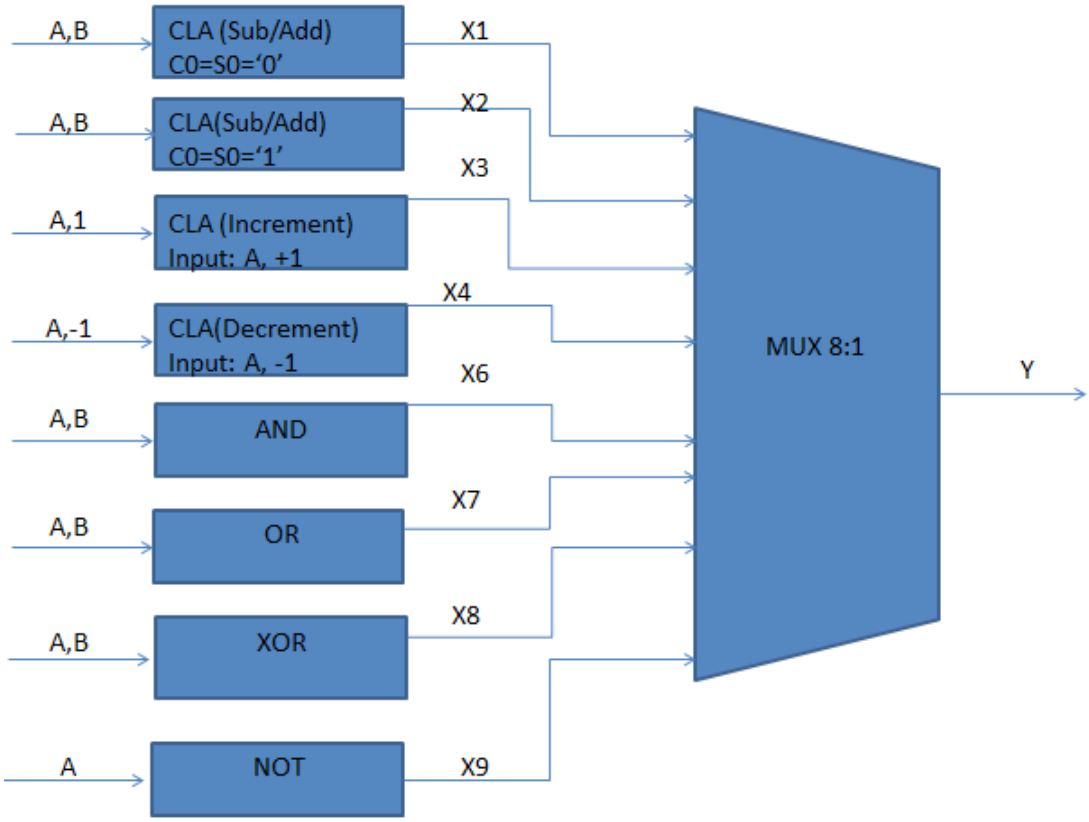


Figure 7.1. Top Level view of ALU

7.1.1 Carry-Lookahead Adder

The Adder is a basic component in any central processing unit . The speed of compute is become the most considerable condition for the designers. The Carry Look Ahead adder is the highest speed adder nowadays. It is a Fast adder which we using it to increase the performance of the application. We used CLA 4 bits with 1 carry propagate C3 and construct el CLA 8 bits by using a combinational of 2 CLA 4 bits with C0 of second CLA equal to C3 of first CLA. Its top level architecture is shown in figure 7.2.

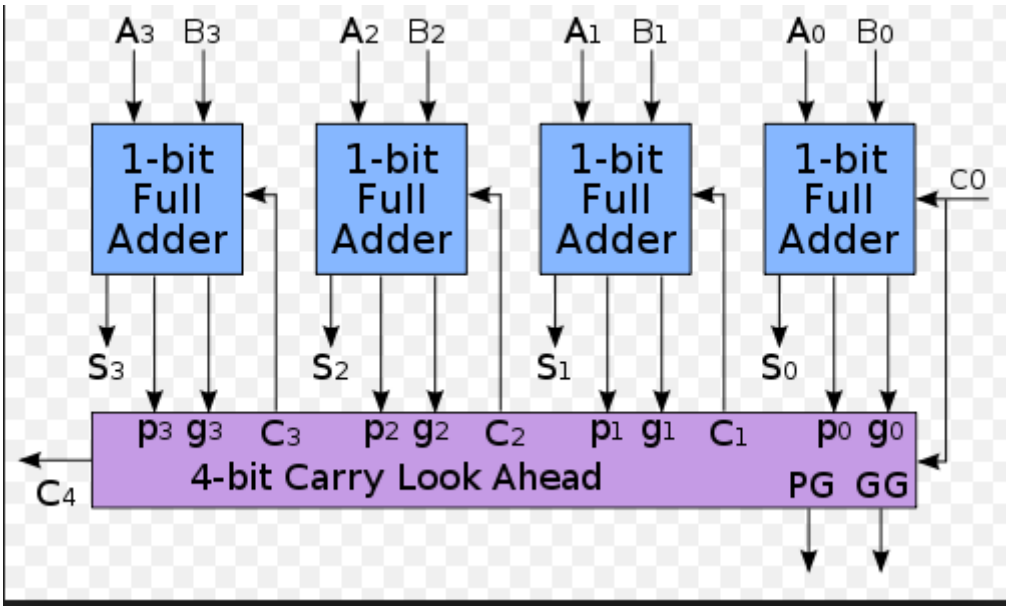


Figure 7.2. Top Level View of CLA

7.1.2 Adder/Subtractor Combined Circuit

To make one circuit can perform operation of Subtracter and Adder we use XOR gate at the second input of CLA which we XOR the input B with a signal S, this signal is connected to the carry in also so, when S=0 the Input B will XOR with 0 and then no change will happened to B and carry in will be 0 so the inputs of CLA will become (A,B,0).

and if S=1 the input B will XOR with 1 which will invert the Input B and carry in will be 1 so the invert B will add to 1 which result in 2'S complement of B and then el operation of subtracter can be performed as shown in figure 7.3.

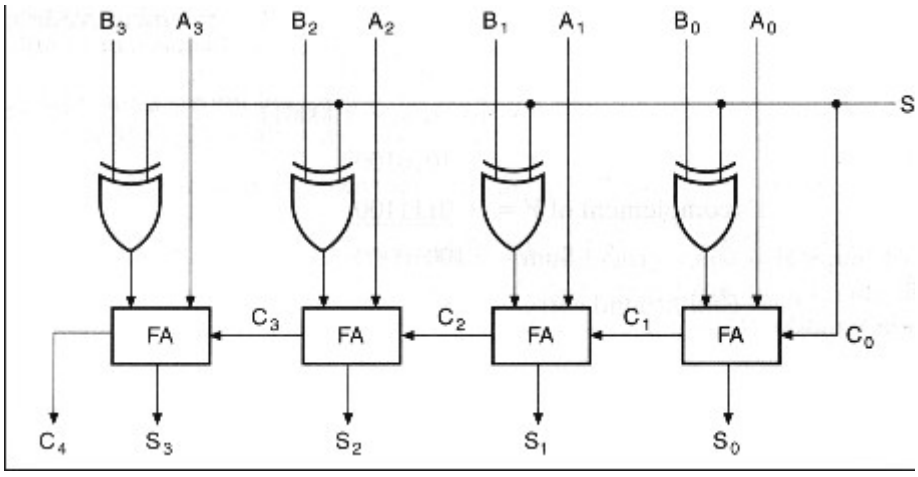


Figure 7.3. Combined Adder/Subtractor Circuit

7.2 Layout

The layout of the Adder/Subtracter CLA is shown in Figure 7.4.

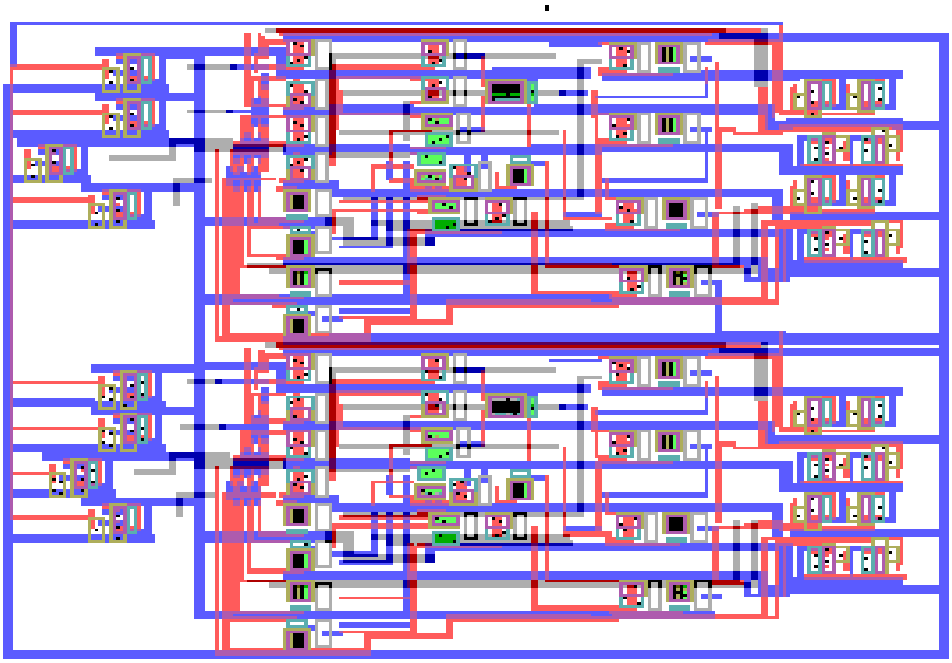
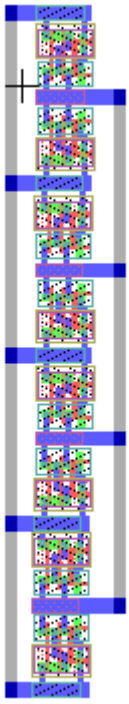


Figure 7.4. Adder/Subtracter Combined CLA Layout

The layouts of OR, AND, NOT and XOR circuits are shown in the figures 7.5, 7.6, 7.7, 7.8 respectively



Figue 7.5. OR Gates

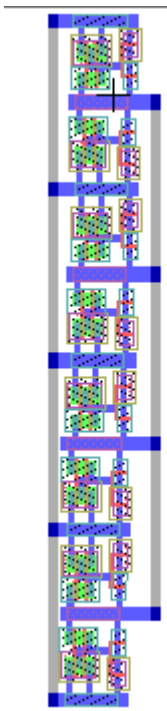


Figure 7.6. AND Gates

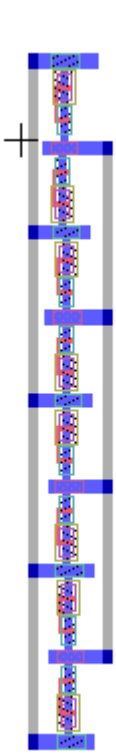


Figure 7.7. NOT Gates

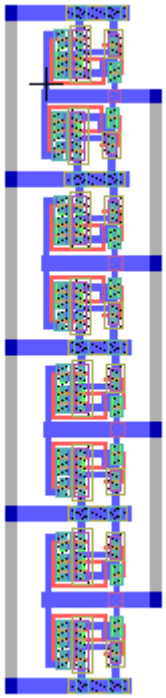


Figure 7.8. XOR Gates

8 Results

The area results are shown in table 8.1.

Module	Area	Number of Transistor
8-Bit Latch	990*216	256
Barrel Shifter	137*235	56
SRAM	964*933	704
CLA	1100*1000	496
OR	83*1000	48
AND	83*1000	48
NOT	40*1000	16
XOR	90*1000	64

Concerning simulations, we faced a problem with the tools we have as we could not extract andsimulate large design.