

بسم الله الرحمن الرحيم

پروژه ۳ مدار منطقی دکتر نوابی

مهدی وجهی

۸۱۰۱۰۱۵۵۸

فهرست

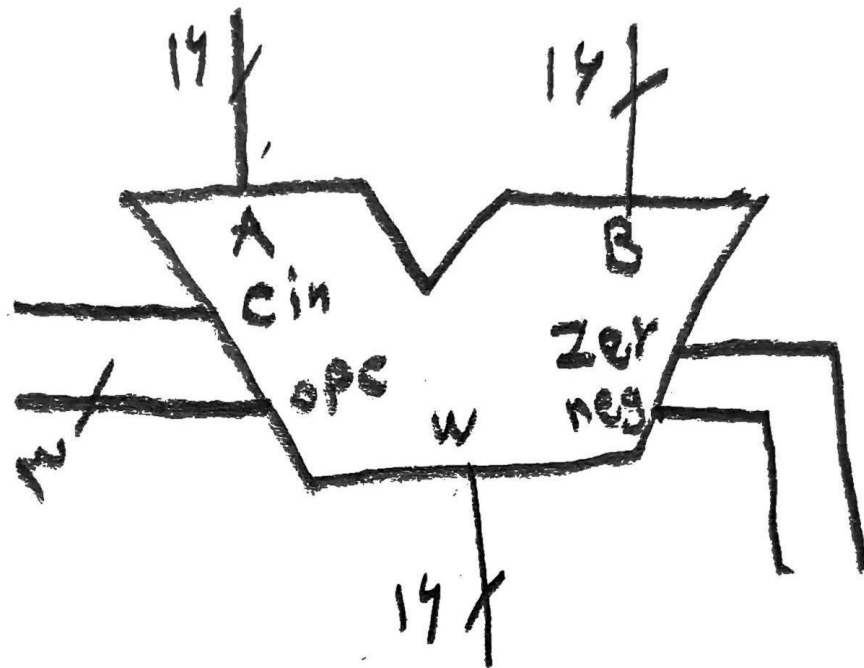
<u>3</u>	<u>سوال ۱</u>
<u>8</u>	<u>سوال ۲</u>
<u>12</u>	<u>سوال ۲+۲</u>
<u>15</u>	<u>مقایسه و نتیجه گیری</u>

سوال ۱

در این سوال باید ALU با استفاده از always statement و جدول زیر ساخت.

Opcode	Function
000	$\text{outW} = 2s\text{Comp}(\text{inA})$
001	$\text{outW} = \text{inA} + 1$
010	$\text{outW} = \text{inA} + \text{inB} + \text{inC}$
011	$\text{outW} = \text{inA} + \text{inB} \times 0.5$
100	$\text{outW} = \text{inA} \& \text{inB}$ (Bitwise)
101	$\text{outW} = \text{inA} \text{inB}$ (Bitwise)
110	$\text{outW} = \{\text{inA}[7:0], \text{inB}[7:0]\}$
111	No operation

ابتدا مدار آن را می کشیم:



کد آن به صورت زیر است:

```
module q1 (output reg [15:0] w, output reg zer,neg,
    input [2:0] opc, input signed [15:0] ina, inb, input inc);
    always @(opc,ina,inb,inc) begin
        w = 16'b0;
        zer = 1'b0;
        neg = 1'b0;

        case(opc)
            3'b000: w = ~ina + 1;
            3'b001: w = ina + 1;
            3'b010: w = ina + inb + inc;
            3'b011: w = ina + (inb >>> 1);
            3'b100: w = ina & inb;
            3'b101: w = ina | inb;
            3'b110: w = {ina[7:0], inb[7:0]};
        endcase

        neg = w[15];

        if (w == 16'b0)
            zer = 1'b1;
        else
            zer = 1'b0;
    end
endmodule
```

حال با استفاده از yosys کد را synthesize می کنیم.

ابتدا فایل پروژه را می خوانیم:

```
yosys> read_verilog q1.v
1. Executing Verilog-2005 frontend: q1.v
Parsing Verilog input from `q1.v' to AST representation.
Generating RTLIL representation for module `q1'.
Note: Assuming pure combinatorial block at q1.v:4 in
compliance with IEC 62142(E):2005 / IEEE Std. 1364.1(E):2002.
Recommending
use of @* instead of @(...) for better match of synthesis and
simulation.
Successfully finished Verilog frontend.
```

حال باید ماژول مورد نظر را اعلام کنیم:

```
yosys> synth -top q1
...
2.26. Printing statistics.

=== q1 ===

Number of wires:          433
Number of wire bits:      480
Number of public wires:   7
Number of public wire bits: 54
Number of memories:       0
Number of memory bits:    0
Number of processes:      0
Number of cells:          443
    $_ANDNOT_              93
    $_AND_                 19
    $_AOI3_                30
    $_AOI4_                20
    $_NAND_                36
    $_NOR_                 13
    $_NOT_                 27
    $_OAI3_                35
    $_OAI4_                15
    $_ORNOT_               8
    $_OR_                  53
    $_XNOR_                31
    $_XOR_                 63

2.27. Executing CHECK pass (checking for obvious problems).
checking module q1..
found and reported 0 problems.
```

سپس باید کتابخانه ی داده شده را به برنامه بدهیم:

```
yosys> dfflibmap -liberty ./yosys/mycells.lib

yosys> abc -liberty ./yosys/mycells.lib

4.1.2. Re-integrating ABC results.
ABC RESULTS:      NAND cells:      261
ABC RESULTS:      NOR cells:       331
ABC RESULTS:      NOT cells:        88
ABC RESULTS:      internal signals: 426
ABC RESULTS:      input signals:    36
ABC RESULTS:      output signals:   17
Removing temp directory.
```

موارد اضافه را حذف می کنیم:

```
yosys> clean
Removed 0 unused cells and 479 unused wires.
```

در آخر نتیجه را در فایل مربوطه ریخته:

```
yosys> write_verilog syq1.v

5. Executing Verilog backend.
Dumping module `q1'.
```

حال باید با testbench مدار را راستی آزمایی کنیم.

```
`timescale 1ns/1ns

module tbq1();
    reg [15:0] a,b;
    reg cin;
    reg [2:0] opc = 3'b0;
    wire [15:0] w, wsy;
    wire zer, zersy, neg, negsy;

    q1 my_ALU (.w(w), .zer(zer), .neg(neg), .opc(opc), .ina(a), .inb(b),
    .inc(cin));
    syq1 my_syALU (.w(wsy), .zer(zersy), .neg(negsy), .opc(opc), .ina(a),
    .inb(b), .inc(cin));

    initial {a, b, cin} = $random;
    initial repeat (20) #1000 {a, b, cin} = $random;
    initial repeat (20) #1000 opc = opc + 3'b001;
endmodule
```

نتیجه تست به صورت زیر است:

tbq1/a	-15207	-2314	-8124	-15806	-9992	860	9071	-9887	-15205	121	876	7569	3010	15210	8882	15998	3695	3717	5495	14679	-8777
tbq1/b	27401	-25986	-20672	27396	11057	-17018	-13114	16946	-22263	-3712	-6522	-1861	-5498	11254	1082	2820	28061	15714	5803	1038	14651
tbq1/cin	0																				
tbq1/opc	100	000	001	010	011	100	101	110	111	000	001	010	011	100	101	110	111	000	001	010	011
tbq1/w	16393	-2314	-8123	11591	-4484	260	-4113	24882	0	-121	876	5708	661	11106	-1066	32508	0	3717	-5494	13642	-1402
tbq1/zer	S10																				
tbq1/neg	S10																				
tbq1/wsy	16393	-2314	-8123	11591	-4484	260	-4113	24882	0	-121	876	5708	661	11106	-1066	32508	0	3717	-5494	13642	-1402
tbq1/negsy	S10																				
tbq1/zersy	S10																				

برای سنجش سرعت شبیه سازی تنها کافیست خط مربوط به مدار دیگر را کامنت کرد و سپس سنجید.
با دستور زیر می توان زمان شبیه سازی را سنجید.

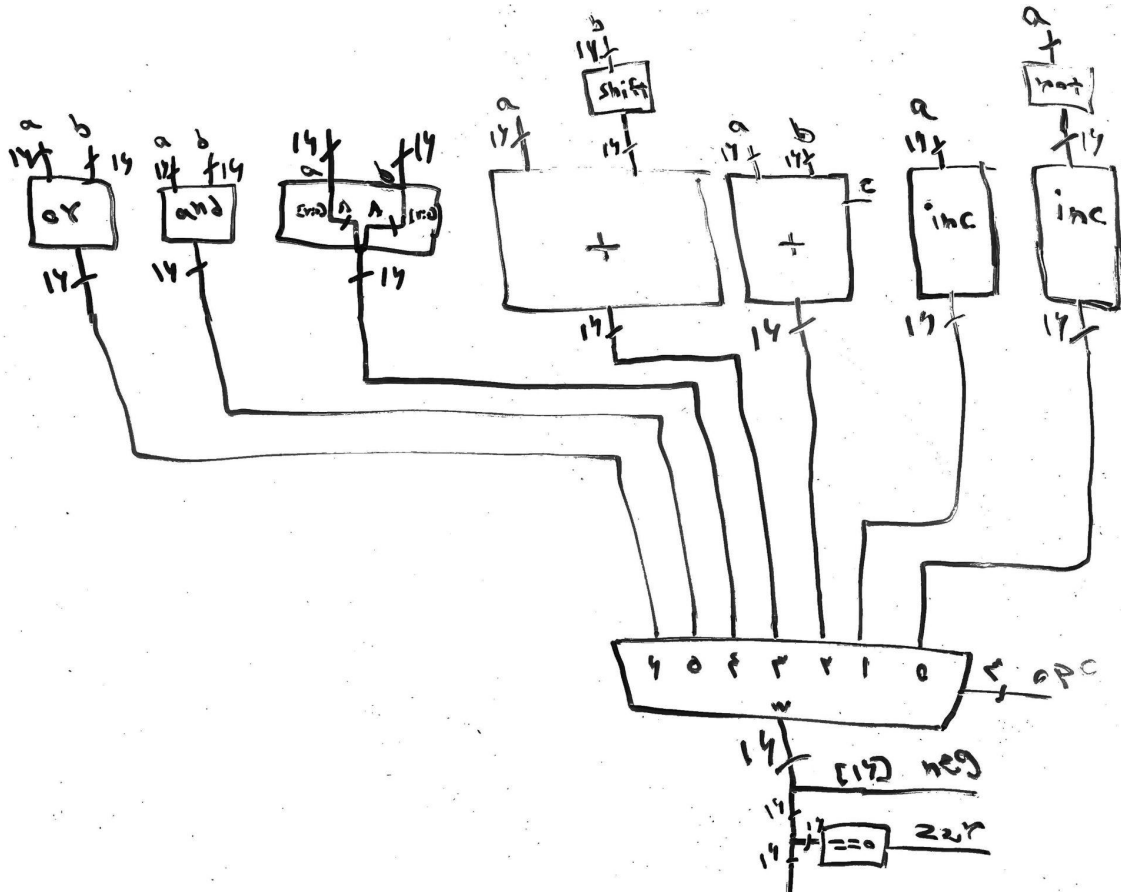
```
set sim_time [time {run -all}]
```

با سنجش چند باره زمان شبیه سازی مشاهده شد که این زمان بین دو مدار تفاوتی ندارد و برای شبیه سازی اول حدود ۸۲۰۰۰ microseconds per iteration است و برای دفعات بعدی ۴۲۰۰۰mpi است.

سوال ۲

در این سوال باید ALU سوال قبل را طراحی کرد با این تفاوت که باید با استفاده از عناصری که در کلاس یاد گرفته ایم مدار را طراحی کنیم.

شکل مدار به صورت زیر است:



در نتیجه کد آن به شکل زیر می شود:

```
module q2 (output reg [15:0] w, output reg zer,neg,
           input [2:0] opc, input signed [15:0] ina, inb, input inc);
  wire [15:0] out [7:0];
  assign out[0] = ~ina + 1;
  assign out[1] = ina + 1;
  assign out[2] = ina + inb + inc;
  assign out[3] = ina + (inb >>> 1);
  assign out[4] = ina & inb;
  assign out[5] = ina | inb;
  assign out[6] = {ina[7:0], inb[7:0]};

  assign w =
    (opc == 3'b000) ? out[0]:
    (opc == 3'b001) ? out[1]:
    (opc == 3'b010) ? out[2]:
    (opc == 3'b011) ? out[3]:
    (opc == 3'b100) ? out[4]:
    (opc == 3'b101) ? out[5]:
    (opc == 3'b110) ? out[6]:
    1'b0;

  assign zer = (w == 16'b0) ? 1 : 0;
  assign neg = w[15];
endmodule
```

حال با استفاده از yosys کد را synthesize می کنیم.

```
yosys> synth -top q2
...
=== q2 ===

Number of wires:          401
Number of wire bits:      463
Number of public wires:   8
Number of public wire bits: 70
Number of memories:       0
Number of memory bits:    0
Number of processes:      0
Number of cells:          410
    $_ANDNOT_             44
    $_AND_                 25
    $_AOI3_                20
    $_MUX_                 96
    $_NAND_                23
    $_NOR_                 7
    $_NOT_                 19
    $_OAI3_                17
    $_ORNOT_               8
    $_OR_                  60
    $_XNOR_                26
    $_XOR_                 65
...
yosys> abc -liberty ./yosys/mycells.lib
...
4.1.2. Re-integrating ABC results.
ABC RESULTS:      NAND cells:      272
ABC RESULTS:      NOR cells:       296
ABC RESULTS:      NOT cells:        68
ABC RESULTS:      internal signals: 393
ABC RESULTS:      input signals:    36
ABC RESULTS:      output signals:   17
Removing temp directory.
...
```

حال باید با testbench مدار را راستی آزمایی کنیم.

```
`timescale 1ns/1ns

module tbq2();
    reg [15:0] a,b;
    reg cin;
    reg [2:0] opc = 3'b0;
    wire [15:0] w, wsy;
    wire zer, zersy, neg, negsy;

    q2 my_ALU (.w(w), .zer(zer), .neg(neg), .opc(opc), .ina(a), .inb(b),
    .inc(cin));
    syq2 my_syALU (.w(wsy), .zer(zersy), .neg(negsy), .opc(opc), .ina(a),
    .inb(b), .inc(cin));

    initial {a, b, cin} = $random;
    initial repeat (20) #1000 {a, b, cin} = $random;
    initial repeat (20) #1000 opc = opc + 3'b001;
endmodule
```

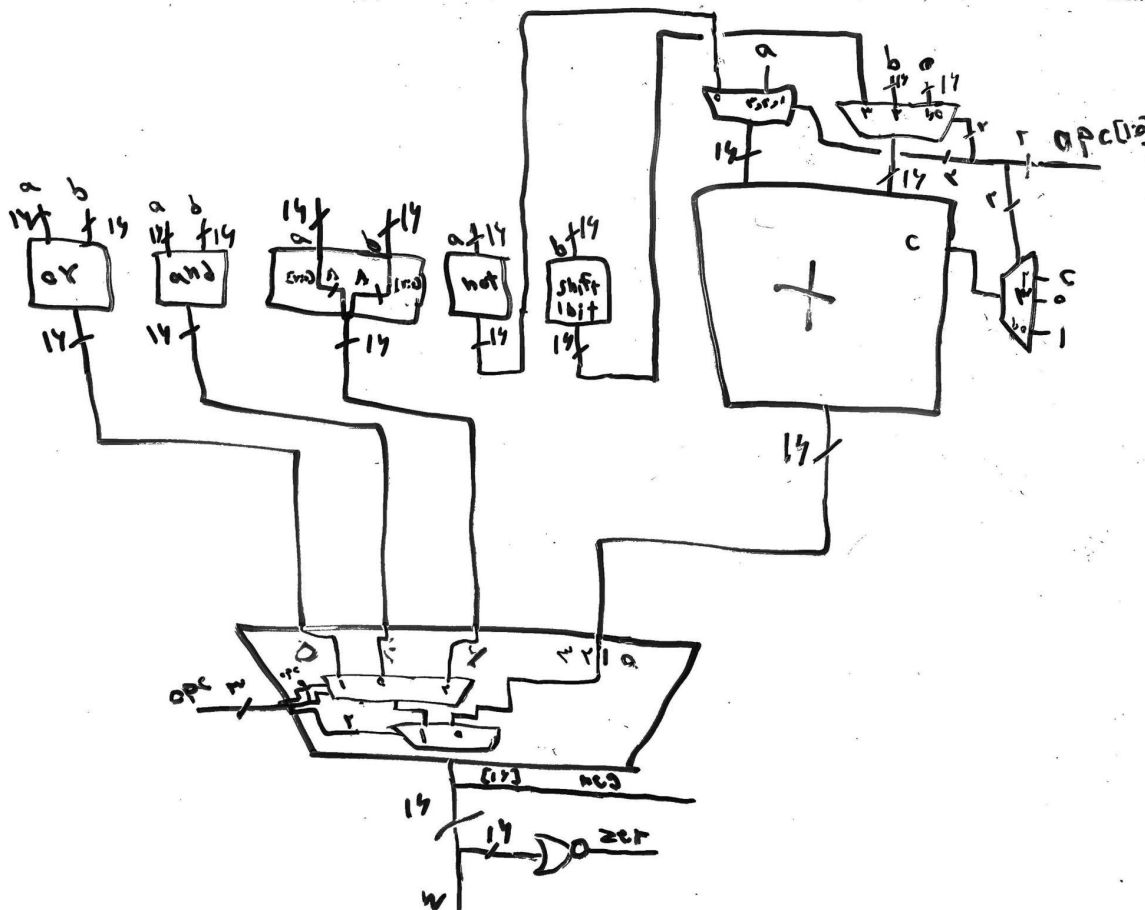
نتیجه تست به صورت زیر است:

tbq2a	2314	-3124	-11550	-4582	-250	-5112	-2887	0	-121	676	-578	661	-11105	-1066	-29207	0	-3717	-5895	-13679	-1401	
tbq2b	-2666	-20672	-27296	-11657	-17018	-18114	-16646	-32263	-3712	-692	-381	-688	-1124	-1082	-2620	-28061	-14774	-5803	-1038	-1664	
tbq2cin	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
tbq2opc	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
tbq2w	2314	-3123	-11551	-4581	-250	-5113	-2882	0	-121	676	-578	661	-11106	-1066	-29208	0	-3717	-5894	-13642	-1402	
tbq2zer	S10																				
tbq2z	S11																				
tbq2wsy	z	-3314	-8123	-11591	-4494	-250	-5113	-2882	0	-121	676	-578	661	-11106	-1066	-29208	0	-3717	-5894	-13642	-1402
tbq2zsy	S1X																				
tbq2negsy	S1X																				

با سنجش چند باره زمان شبیه سازی مشاهده شد که این زمان بین دو مدار تفاوتی ندارد و برای شبیه سازی اول حدود ۴۵۰۰۰ microseconds per iteration است و برای دفعات بعدی ۳۲۰۰۰۰ mpi است.

سوال ۲+

در سوال ۲ تمرکز روی سادگی کد بود در این بخش با طراحی مجدد مدار آن را مقداری بهینه تر کردیم.
شکل مدار به صورت زیر است:



حال مدار را پیاده سازی می کنیم:

```
module q2_plus (output reg [15:0] w, output reg zer,neg,
               input [2:0] opc, input signed [15:0] ina, inb, input inc);
  wire [15:0] adder_out, not_a, shift_b, mux_adder_a, mux_adder_b, mux_adder_c;
  assign not_a = ~ina;
  assign shift_b = inb >>> 1;

  assign mux_adder_c =
    (opc[1:0] == 2'b00) ? 1'b1:
    (opc[1:0] == 2'b01) ? 1'b1:
    (opc[1:0] == 2'b10) ? inc:
    (opc[1:0] == 2'b11) ? 1'b0:
    1'bx;

  assign mux_adder_b =
    (opc[1:0] == 2'b00) ? 16'b0:
    (opc[1:0] == 2'b01) ? 16'b0:
    (opc[1:0] == 2'b10) ? inb:
    (opc[1:0] == 2'b11) ? shift_b:
    16'bx;

  assign mux_adder_a =
    (opc[1:0] == 2'b00) ? not_a:
    (opc[1:0] == 2'b01) ? ina:
    (opc[1:0] == 2'b10) ? ina:
    (opc[1:0] == 2'b11) ? ina:
    16'bx;

  assign adder_out = mux_adder_a + mux_adder_b + mux_adder_c;

  wire [15:0] mixed_a_b, or_a_b, and_a_b, mux_4_5_6;
  assign and_a_b = ina & inb;
  assign or_a_b = ina | inb;
  assign mixed_a_b = {ina[7:0], inb[7:0]};

  assign mux_4_5_6 =
    (opc[1:0] == 2'b00) ? and_a_b:
    (opc[1:0] == 2'b01) ? or_a_b:
    (opc[1:0] == 2'b10) ? mixed_a_b:
    1'b0;

  assign w =
    (opc[2] == 1'b0) ? adder_out:
    (opc[2] == 1'b1) ? mux_4_5_6:
    1'b0;

  assign zer = ~|w;
  assign neg = w[15];
endmodule
```

حال با استفاده از yosys کد را synthesize می کنیم.

```
yosys> synth -top q2_plus
```

2.26. Printing statistics.

```
=== q2_plus ===
```

Number of wires:	279
Number of wire bits:	356
Number of public wires:	9
Number of public wire bits:	86
Number of memories:	0
Number of memory bits:	0
Number of processes:	0
Number of cells:	287
\$_ANDNOT_	40
\$_AND_	23
\$_AOI3_	8
\$_MUX_	63
\$_NAND_	8
\$_NOR_	25
\$_NOT_	19
\$_OAI3_	5
\$_ORNOT_	2
\$_OR_	46
\$_XNOR_	16
\$_XOR_	32

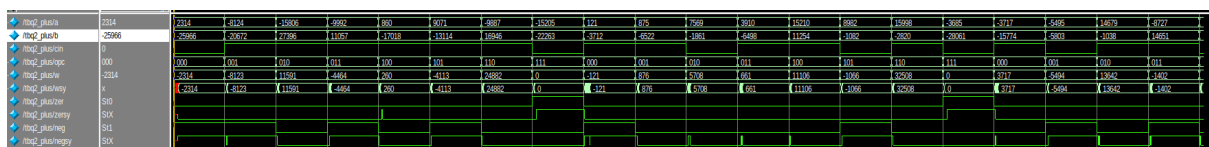
```
2.27. Executing CHECK pass (checking for obvious problems).
checking module q2_plus..
found and reported 0 problems.
```

```
yosys> abc -liberty ./yosys/mycells.lib
```

4.1.2. Re-integrating ABC results.

```
ABC RESULTS:      NAND cells:      205
ABC RESULTS:      NOR cells:       181
ABC RESULTS:      NOT cells:        77
ABC RESULTS:      internal signals: 270
ABC RESULTS:      input signals:    36
ABC RESULTS:      output signals:   17
Removing temp directory.
```

با testbench مدار را ارزیابی می کنیم.



مقایسه و نتیجه گیری

یک test bench برای ارزیابی مدار های synthesize شده می نویسیم که تاخیر ها را بسنجیم.

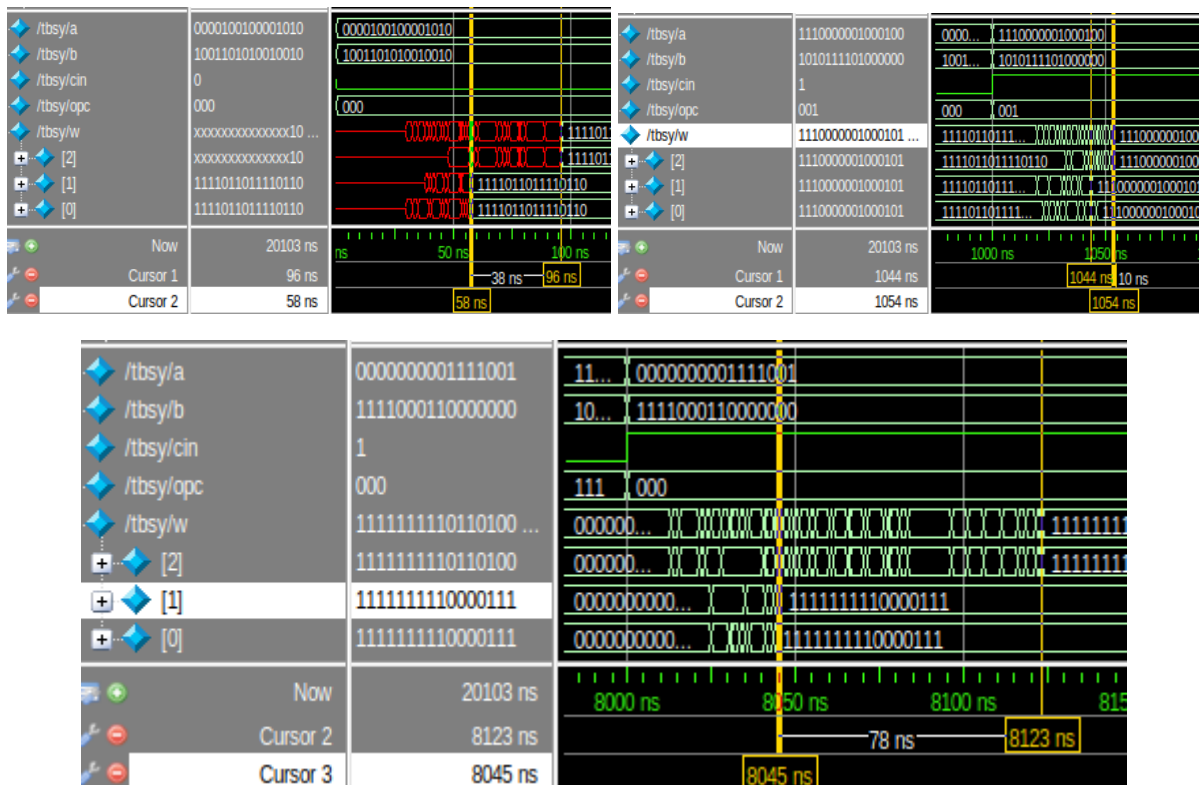
```
`timescale 1ns/1ns

module tbsy();
    reg [15:0] a,b;
    reg cin;
    reg [2:0] opc = 3'b0;
    wire [2:0][15:0] w;
    wire [2:0] zer, neg;

    syq1      q1      (.w(w[0]), .zer(zer[0]), .neg(neg[0]), .opc(opc),
    .ina(a), .inb(b), .inc(cin));
    syq2      q2      (.w(w[1]), .zer(zer[1]), .neg(neg[1]), .opc(opc),
    .ina(a), .inb(b), .inc(cin));
    syq2_plus q2_plus (.w(w[2]), .zer(zer[2]), .neg(neg[2]), .opc(opc),
    .ina(a), .inb(b), .inc(cin));

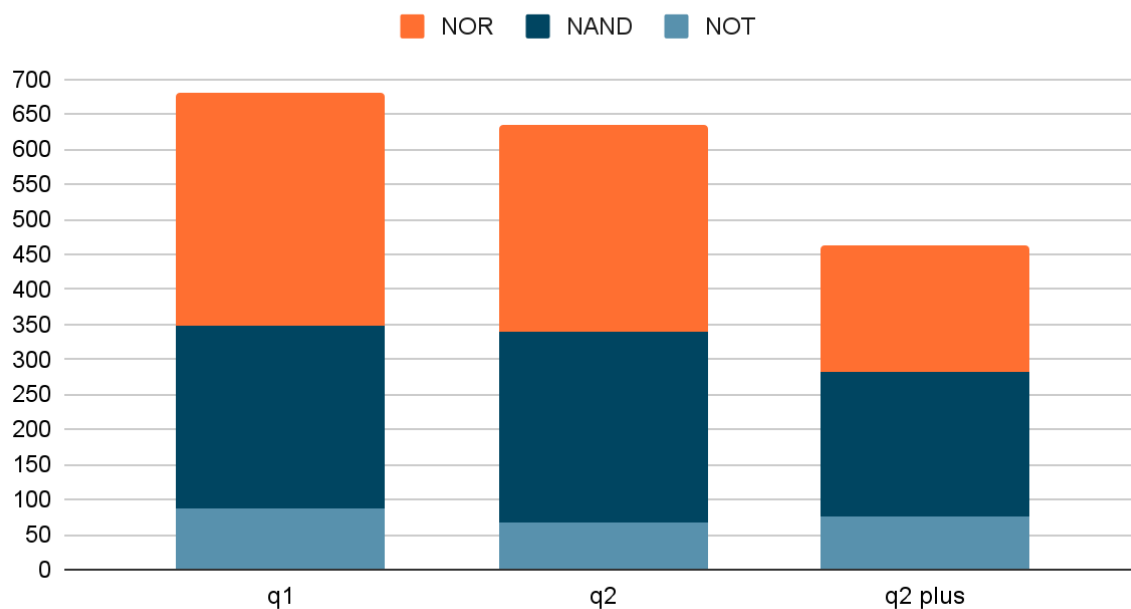
    initial {a, b, cin} = $random;
    initial repeat (20) #1000 {a, b, cin} = $random;
    initial repeat (20) #1000 opc = opc + 3'b001;
endmodule
```

نتایج به شکل زیر است:



همانطور که مشاهده می شود تاخیر مدار آخر بیشتر از دو مدار دیگر است و دو مدار دیگر زمان مشابهی نسبت به هم دارند.

حال به بررسی میزان استفاده از گیت ها می پردازیم. نمودار حاصل از اطلاعات yosys به صورت زیر است:



مشاهده می شود که تعداد گیت ها در q2 plus به صورت قابل توجهی کم تر است. می توان نتیجه گرفت که yosys تمرکز کمتری روی میزان استفاده از گیت ها دارد و در عوض مداری با تاخیر کمتر ارائه می دهد.