



دانشکده مهندسی کامپیوتر

آز سیستم دیجیتال

نیم سال تابستان ۰۲-۰۱

مدرس: دکتر انصاری

سپنتا رحمانی زاده

۹۸۱۱۰۰۴۹

عطا رحیم زاده

۹۸۱۷۰۸۰۵

عماد زین اوقلی

۹۸۱۰۳۲۶۷

۱ آزمایش سوم

در این آزمایش بایستی یک مدار مقایسه کننده ۴ بیتی ترکیبی و یک مدار مقایسه کننده سریالی طراحی کنیم.

۱.۱ مدار ترکیبی

برای این قسمت دو ماژول طراحی کردیم. در ماژول one_bit_comparator یک مقایسه کننده تک بیتی آبشاری^۱ طراحی کردیم. سپس در ماژول four_bit_comparator با استفاده از ماژول one_bit_comparator یک مقایسه کننده چهاربیتی طراحی کردیم. در این بخش جزییات این دو ماژول به همراه آزمون و Waveform مدار را می آوریم.

one_bit_comparator ۱.۱.۱

از آنجا که قرار است این ماژول به صورت آبشاری کار کنند علاوه بر دو ورودی تک بیتی a, b سه سیگنال ورودی تک بیتی great_in, equal_in, less_in را نیز در نظر گرفتیم که خروجی قبلی خواهند. این مدار سه خروجی equal_out, great_out, less_out نیز دارد که به صورت زیر از ورودی ها بدست می آیند.

$$\begin{cases} \text{less_out} &= (\overline{\text{equal_in}} \cdot \overline{\text{great_in}} \cdot \text{less_in}) + (\text{equal_in} \cdot \overline{a} \cdot b) \\ \text{great_out} &= (\overline{\text{equal_in}} \cdot \overline{\text{less_in}} \cdot \text{great_in}) + (\text{equal_in} \cdot \overline{b} \cdot a) \\ \text{equal_out} &= \text{equal_in} \cdot \overline{\text{less_in}} \cdot \overline{\text{great_in}} \cdot a \oplus b \end{cases}$$

```

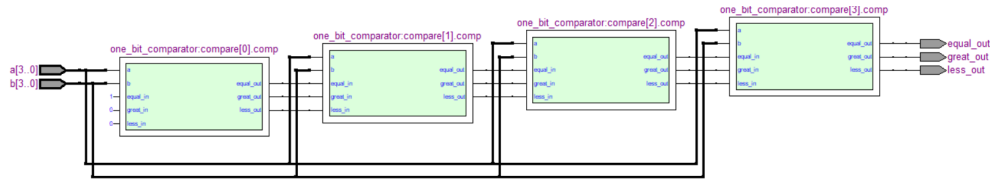
1
2
3 module one_bit_comparator (a, b, less_in, great_in,
   equal_in, less_out, great_out, equal_out);
4
5     input a, b, equal_in, less_in, great_in, equal_in;
6     output less_out, great_out, equal_out;
7
8     assign less_out = (~equal_in & less_in & ~great_in) | (equal_in
   & ~a & b);
9     assign great_out = (~equal_in & ~less_in & great_in) | (
   equal_in & a & ~b);
10    assign equal_out = (equal_in & ~less_in & ~great_in & ~(a ^ b
   ));
11
12 endmodule

```

cascaded^۱

four_bit_comparator ۲.۱.۱

در این ماژول با استفاده از ۴ ماژول one_bit_comparator یک مقایسه کننده چهار بیتی به شکل زیر ساختیم.



برای اولین ماژول ورودی‌ها را به صورت زیر قرار دادیم.

$$\begin{cases} \text{less_out} &= 0 \\ \text{great_out} &= 0 \\ \text{equal_out} &= 1 \end{cases}$$

همچنین خروجی آخرین ماژول را به عنوان خروجی کل مدار قرار دادیم.

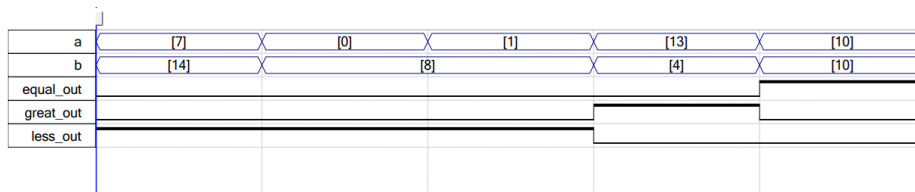
```

1
2
3 module four_bit_comparator (a, b, less_out, equal_out, great_out);
4
5     parameter          LEN = 4;
6     input  wire         [LEN-1:0] a, b;
7     output              less_out, great_out, equal_out;
8
9     wire                [LEN:0] less, great, equal;
10
11     assign less[0] = 0;
12     assign equal[0] = 1;
13     assign great[0] = 0;
14     assign less_out = less[LEN];
15     assign great_out = great[LEN];
16     assign equal_out = equal[LEN];
17
18     genvar i;
19     generate
20     for (i = 0 ; i < LEN; i = i + 1) begin : compare
21         one_bit_comparator comp(
22             .a(a[LEN - 1 - i]),
23             .b(b[LEN - 1 - i]),
24             .less_in(less[i]),
25             .great_in(great[i]),
26             .equal_in(equal[i]),
27             .less_out(less[i+1]),
28             .great_out(great[i+1]),
29             .equal_out(equal[i+1])
30         );
31     end
32     endgenerate
33
34 endmodule

```

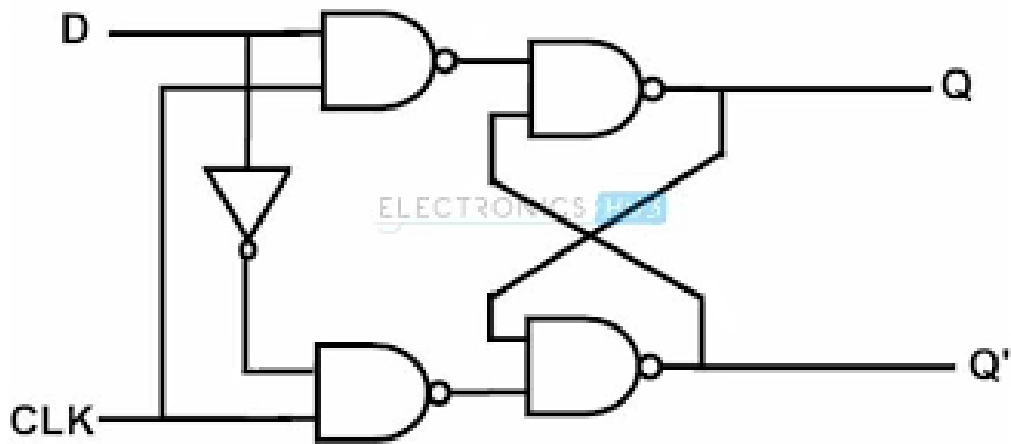
۳.۱.۱ آزمون

برای آزمایش و سنجش این مدار به صورت تصادفی اعداد زیر را تولید کردیم و مدار را بر آن‌ها اجرا نمودیم. نتیجه waveform به صورت زیر است.



۲.۱ مدار سریالی

برای هر خروجی یک فلیپ فلاپ نوع D در نظر می‌گیریم.



با توجه به این موضوع، برای هر فلیپ فلاپ مقدار ورودی در واقع خروجی مقایسه کننده تک بیتی آبشاری است (البته نیاز است که reset نیز هم در نظر گرفت). بدین ترتیب، ماژول زیر را داریم.

```

1
2 module comparator (a, b, reset, less_out, great_out,
3   equal_out, clk);
4     input a, b, reset, clk;
5     output less_out, great_out, equal_out;
6     wire less_new, great_new, equal_new;
7     wire less_up, less_down, great_up, great_down, equal_up, equal_down;
8     wire not_less, not_great, not_equal;
9
10    assign less_new = ~reset & ((less_out & ~great_out & ~equal_out) | (
11      equal_out & ~a & b));
12    assign great_new = ~reset & ((~less_out & great_out & ~equal_out) |
13      (equal_out & a & ~b));
14    assign equal_new = reset | (~reset) & (~less_out & ~great_out &
15      equal_out & ~(a ^ b));
16
17    assign less_up = ~(less_new & clk);
18    assign less_down = ~(~less_new & clk);
19
20    assign great_up = ~(great_new & clk);
21    assign great_down = ~(~great_new & clk);
22
23    assign equal_up = ~(equal_new & clk);
24    assign equal_down = ~(~equal_new & clk);
25
26    assign less_out = less_up & less_down;
27    assign great_out = great_up & great_down;
28    assign equal_out = equal_up & equal_down;
29
30 endmodule

```

```

16    assign great_up = ~(great_new & clk);
17    assign great_down = ~(~great_new & clk);
18    assign equal_up = ~(equal_new & clk);
19    assign equal_down = ~(~equal_new & clk);
20
21    assign less_out = ~(less_up & not_less);
22    assign great_out = ~(great_up & not_great);
23    assign equal_out = ~(equal_up & not_equal);
24    assign not_less = ~(less_down & less_out);
25    assign not_great = ~(great_down & great_out);
26    assign not_equal = ~(equal_down & equal_out);
27
28    endmodule

```

۱.۲.۱ آزمون

برای آزمایش و سنجش این مدار به صورت تصادفی اعداد زیر را تولید کردیم و مدار را بر آن‌ها اجرا نمودیم. نتیجه waveform به صورت زیر است.

