

گزارش آزمایش طراحی سیستم های دیجیتال

آزمایش چهارم:

توصیف رفتاری

نگارش :

سید ابوالفضل رحیمی ۹۷۱۰۵۹۴۱

امیرحسین باقری ۹۸۱۰۵۶۲۱

استاد :

دکتر علیرضا اجاللی

دستیار آموزشی :

آقای روزبه سیادت زاده

Contents

2 مقدمه
2 ابزار پیاده سازی مدار
2 شرح خروجی ها و ورودی ها
3 خروجی ها :
3 ورودی ها:
4 گزارش کد stack.v
6 Test bench
6 حالت push ها متوالی و پر شدن
6 بررسی درستی ترتیب pop ها و حالت خالی شدن
7 بررسی حالت Error و RstN
7 Synthesis report

مقدمه

هدف از این آزمایش طراحی یک پشته است که یک داده ساختار برای حفظ و بازیابی داده ها در یک سیستم دیجیتال یا یک نرم افزار از آن استفاده می شود. عناصر این داده ساختار به صورت LIFO یا همان last in first out است.

عملیات های مورد استفاده در داده ساختار استک:

- 1 - عملیات push : یک داده را از Data_In وارد استک می کند.
- 2 - عملیات pop : یک داده را از بالای استک برداشته و آن خانه آزاد می شود.
- 3 - عملیات RSTN : خانه های استک آزاد می شوند. (صفر کردن آنها اختیاریست و الزامی برای آن وجود ندارد.)

ابزار پیاده سازی مدار

مدار به کمک ابزار مادلسیم کشیده شده اند و همچنین کد stack در ابزار کوارتوس سنتز گشته است.

شرح خروجی ها و ورودی ها

module stack

```
# (parameter WordSize = 4 , AddressSize = 3)
```

```
(  
    output reg [3:0] Data_Out,  
    output reg Full,  
    output reg Empty,  
    output reg Error,  
    input Clk,  
    input RstN,  
    input [3:0] Data_In,  
    input push,  
    input pop  
);
```

خروجی ها :

Data_Out خروجی پشته در هنگام عملیات pop را داراست.

Full سیگنالی که زمانی که استک پر باشد یک می شود.

Empty سیگنالی که زمانی که پشته خالی (تهی) باشد یک می شود.

Error زمانی که سیگنال های ورودی push , pop هر دو یک باشند یک می شود بدان معنا که عملیات های خواسته شده به صورت همزمان نمی توانند انجام شوند. و در این حالت هیچ کاری صورت نمی گیرد.

ورودی ها:

Clk ورودی کلاک پالس کلاک را به ماژول می دهد.

RstN مقادیر پشته را پاک می کند و پشته را خالی می کند.

Data_In مقداری که باید در هنگام push وارد پشته شود را به ماژول می دهد.

Push سیگنالی که مشخص کننده عمل push به پشته است.

Pop سیگنالی که مشخص کننده عمل pop از پشته است.


```

43         else begin
44             Full <= 0;
45             Empty <= 0;
46         end
47     end
48     else if (push == 1 && pop == 0 && Full!=1) begin
49         Mem[counter[AddressSize-1:0]]<= Data_In;
50         counter <= counter +4'b0001;
51         Error <= 0;
52         if (counter == 4'b0111) begin
53             Full <= 1;
54             Empty <= 0;
55         end
56         else begin
57             Full <= 0;
58             Empty <= 0;
59         end
60     end
61     else if (push == 1 && pop == 1) begin
62         Error <= 1;
63     end
64     else begin
65         Error <=0;
66     end
67     if (counter[AddressSize] ==1) begin
68         Full <= 1;
69         Empty <= 0;
70     end
71     else if (counter == 4'b0000) begin
72         Full <= 0;
73         Empty <= 1;
74     end
75     end
76 end
77 endmodule
--

```

مدار با کلاک کار می‌کند بنابراین یک مدار synchronous است.

با توجه به رفتاری بودن کد می‌توان گفت هنگامی که لبه بالارونده کلاک صورت می‌گیرد فرایند زیر توسط کد بالا سنتز می‌گردد و به یک مدار دیجیتال تبدیل می‌شود.

ماژول دارای یک حافظه با ادرس دهی ۳ بیتی و کلمات ۴ بیتی است و یک شمارنده برای نشان دادن index در پشته نیز درون خود دارد.

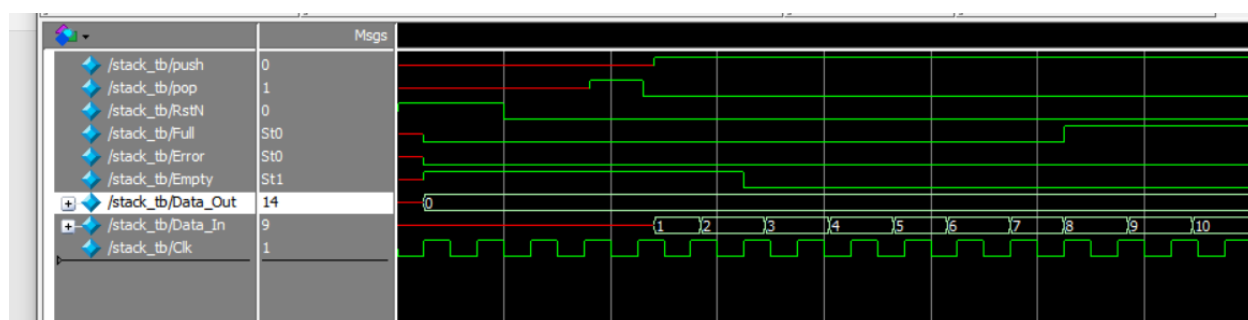
اگر تنها سیگنال pop یک باشد و سیگنال Empty نیز یک نباشد (پشته خالی نباشد) مدار مقدار روی پشته را در خروجی Data_Out میریزد و مقدار counter یکی کم می‌شود. سیگنال error نیز صفر می‌شود همچنین اگر شمارنده مقدار یک را دارا باشد با خالی شدن آن باید سیگنال empty فعال شود.

حال اگر تنها سیگنال full فعال باشد و پشته پر نباشد مقدار از Data_In وارد پشته شده و مقدار شمارنده نیز یکی اضافه می‌شود. اگر مقدار شمارنده ۷ باشد باید سیگنال full فعال شود.

Test bench

کد تست بنچ دیگر آورده نشده است و تنها ویو فرم بررسی می‌شود.
حالت های متفاوتی که ممکن است پیش بیاید را بررسی می‌کنیم.

حالت push ها متوالی و پر شدن



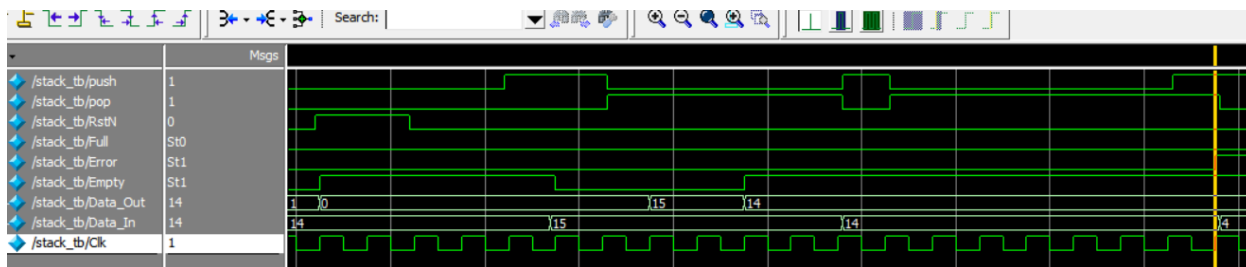
همانطور که مشاهده می‌شود مقدار هنگام پوش کردن ۸ مقدار سیگنال full یک شده است و پشته پر شده.

بررسی درستی ترتیب pop ها و حالت خالی شدن



پس از pop شدن مقادیر همانطور که مشاهده می‌شود مقادیر برعکس از پشته خارج شده اند و هنگام پاپ شدن آخرین مقدار مقدار سیگنال empty یک شده است.

بررسی حالت RstN و Error



زمانی که سیگنال RstN یک شده است مقدار خروجی صفر شده و پشته پاک شده است همچنین زمانی که هردو سیگنال pop , push یک شده اند مقدار خروجی error یک است.

Synthesis report

کد در کوارتوس سنتز شده است.

