



گزارش آزمایش چهارم طراحی سیستم‌های دیجیتال

گروه شش

اعضا:

احمد سلیمی

همیلا میلی

درنا دهقانی

شرح آزمایش

هدف این آزمایش طراحی پشته‌ای (LIFO) به عمق ۸ و پهنای ۴ بیت است.

ورودی‌ها:

- Data_In: مقدار ورودی ۴ بیتی که می‌خواهیم وارد پشته کنیم.
- Push: دستور وارد کردن داده به پشته.
- Pop: دستور خروج داده از پشته.
- Clk
- RstN

خروجی‌ها:

- Data_Out: مقدار خروجی پشته در صورت وجود و اعمال ورودی Push.
 - Full: خروجی که مقدار یک آن بیانگر پر بودن پشته است.
 - Empty: خروجی که مقدار صفر آن بیانگر خالی بودن پشته است.
- در این آزمایش تنها از دو ماژول استفاده می‌کنیم که یکی از آنها test bench می‌باشد.

ماژول Stack

این ماژول، ماژول اصلی آزمایش است و ورودی‌ها و خروجی‌های آن قبلاً ذکر شده‌اند.

چون این مدار بر پایه کلاک کار می‌کند، برای مقداردهی‌ها باید از always block استفاده کنیم که به دو سیگنال حساس است:

- لبه بالارونده کلاک.
- لبه پایین‌رونده RstN که همان reset کردن مدار است.

از دو متغیر برای شبیه‌سازی پشته کمک می‌گیریم که چون در always block استفاده می‌شوند، از جنس reg هستند:

- a که نشان‌دهنده آخرین خانه خالی پشته است و مقدار اولیه آن برابر با ۰ است.
- mem که یک آرایه به طول ۸ است که نشان‌دهنده عمق پشته می‌باشد. هم‌چنین هر عضو آرایه ۴ بیت است که بیانگر پهنای پشته است. این متغیر، همان پشته است.

در صورت صفر شدن مقدار RstN، به شکل آسنکرون تمام مقداردهی‌ها به حالت اولیه باز می‌گردند.

در غیر این صورت، با بالا رفتن لبه کلاک مدار شروع به کار می‌کند:

- اگر ورودی Push فعال شود، در صورت پر نبودن پشته (صفر بودن Full)، مقدار Data_In در اولین خانه خالی پشته که با اندیس a مشخص می‌شود، قرار می‌گیرد و در صورت برابر بودن a با ۷، یعنی آخرین خانه پشته، خروجی Full برابر با ۱ می‌شود. سپس این اندیس به‌علاوه یک و مقدار خروجی Empty برابر صفر می‌شود.
- اگر ورودی Pop فعال شود، در صورت خالی نبودن پشته (صفر بودن Empty)، مقدار آخرین خانه پر پشته که با اندیس a-1 مشخص می‌شود در خروجی Data_Out قرار می‌گیرد و در صورت برابر بودن a با ۱، یعنی خالی شدن اولین خانه پشته، خروجی Empty برابر با ۱ می‌شود. سپس این اندیس منهای یک می‌شود و مقدار خروجی Full برابر با صفر می‌شود.
- در غیر این دو صورت، اتفاق خاصی نمی‌افتد و خروجی‌ها ثابت می‌مانند.

در سنتز مدار، چون می‌خواهیم به ازای هر یک از بیت‌های Full, Empty, Data_Out یک فلیپ‌فلاپ داشته باشیم، از مقداردهی non-blocking استفاده می‌کنیم.

کد وریلاگ این ماژول به شرح زیر است:

```

module stack (
    input Clk,
    input RstN,
    input [3:0] Data_In,
    input Push,
    input Pop,
    output reg [3:0] Data_Out,
    output reg Full,
    output reg Empty
);

reg [3:0] mem [7:0];
reg [3:0] a = 0;

always @(posedge Clk or negedge RstN) begin
    if (~RstN) begin
        a <= 0;
        Empty <= 1;
        Full <= 0;
    end
    else if (Push & ~Full) begin
        if (a == 7) Full <= 1;
        mem[a] <= Data_In;
        a <= a + 1;
        Empty <= 0;
    end
    else if (Pop & ~Empty) begin
        if (a == 1) Empty <= 1;
        Data_Out <= mem[a-1];
        a <= a-1;
        Full <= 0;
    end
end
endmodule

```

ماژول Stack_TB

از این ماژول برای بررسی کارکرد مدار استفاده می‌شود. کد وریلاگ آن به شرح زیر است:

```

module stack_TB();

reg        clk = 0, reset = 0, push = 0, pop = 0;
reg [3:0]   data_in;
wire       full, empty;
wire [3:0]  data_out;

stack Stack(
    .Clk(clk),
    .RstN(reset),
    .Push(push),
    .Pop(pop),
    .Data_In(data_in),
    .Data_Out(data_out),
    .Full(full),
    .Empty(empty)
);

always #10 clk = ~clk;

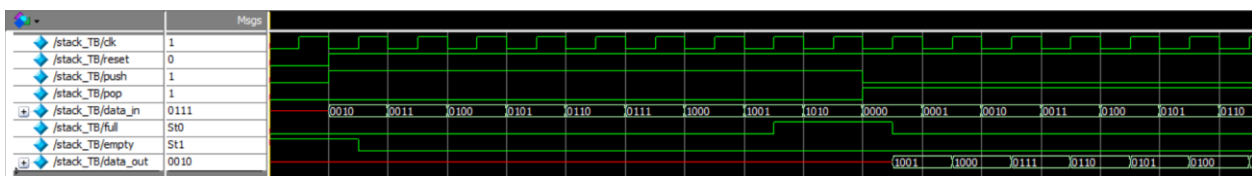
initial begin
    $monitor("data_out = %d, Full = %d, Empty = %d", data_out, full, empty);
    #20 reset = 1;
    push = 1;
    for (data_in = 2; data_in < 11; data_in = data_in + 1) begin
        #20;
    end
    push = 0;
    pop = 1;
    for (data_in = 0; data_in < 9; data_in = data_in + 1) begin
        #20;
    end
    push = 1;
    data_in = 7;
    #20;
    #20 reset = 0;
    #20;
    $stop;
end

endmodule

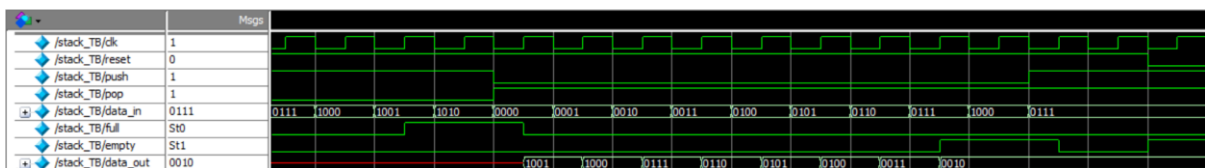
```

Waveform

حاصل waveform شبیه‌سازی test bench این مدار به شکل زیر است:



با توجه به حاصل شبیه‌سازی، به محض رسیدن تعداد داده‌های درون پشته به ۸، خروجی Full برابر با ۱ می‌شود و پس از آن دیگر مقداری وارد پشته نمی‌شود، تا زمانی که ورودی Pop برابر با ۱ شود و سپس Full برابر با ۰ شود.



در شبیه‌سازی موفق نیز با Pop کردن‌های متوالی هر بار مقدار آن در Data_Out قرار می‌گیرد. پس از ۸ بار Pop کردن، پشته خالی شده و خروجی Empty برابر با ۱ می‌شود.

سنتز

خلاصه سنتز مدار توسط کوآرتوس به شرح زیر است:

Flow Summary	
Flow Status	Successful - Mon Apr 12 20:13:56 2021
Quartus II 64-Bit Version	15.0.0 Build 145 04/22/2015 SJ Web Edition
Revision Name	lab4
Top-level Entity Name	stack
Family	Cyclone IV E
Total logic elements	59 / 6,272 (< 1 %)
Total combinational functions	45 / 6,272 (< 1 %)
Dedicated logic registers	42 / 6,272 (< 1 %)
Total registers	42
Total pins	14 / 92 (15 %)
Total virtual pins	0
Total memory bits	0 / 276,480 (0 %)
Embedded Multiplier 9-bit elements	0 / 30 (0 %)
Total PLLs	0 / 2 (0 %)
Device	EP4CE6E22C6
Timing Models	Final

Slow 1200mV OC Model Fmax Summary				
	Fmax	Restricted Fmax	Clock Name	Note
1	484.26 MHz	250.0 MHz	Clk	limit due to minimum period restriction (max I/O toggle rate)