

گزارش آزمایش ششم



دانشگاه صنعتی شریف - بهار ۱۴۰۰

آزمایشگاه طراحی سیستم‌های دیجیتال - دکتر اجلالی

نویسندگان:

سروش جهانزاد - ۹۸۱۰۰۳۸۹

علی حاتمی تاجیک - ۹۸۱۰۱۳۸۵

۱ | مقدمه

در مستند پیش رو گزارشی بر روند طراحی انکوباتر ارائه شده است. فایل های کد این مدار (گرچه کدها درون این مستند نیز موجودند) به همراه کد تست بنچ آن نیز در کنار گزارش قابل دسترسی اند. سعی شدم تا در صورت لزوم توضیحات کافی برای بخش های مختلف آورده شود. از صبر و بردباری شما در مطالعه این گزارش سپاسگزاریم.

با احترام

سروش جهانزاد، علی حاتمی تاجیک - بهار ۱۴۰۰

۲ ساخت انکوباتر

انکوباتری که قرار است سیستم دیجیتال آن پیاده سازی شود از دو بخش کلی تشکیل شده است. یک بخش مربوط به کنترل فن و بخشی مربوط به قسمت انرژی (پاور) سیستم است.

سیستم اینطور کار می کند که با نرخ ثابتی (کلاک) دما را دریافت کرده، در صورت نیاز و وضعیت (استیت) دو بخش را تغییر داده و خروجی های برنامه نیز با توجه به وضعیت دو بخش آپدیت خواهد شد.

ورودی های سیستم به شرح زیر خواهد بود:

- T: دمای ورودی گرفته شده از سنسور - ۸ بیتی علامت دار
 - clk: کلاک سیستم
 - rstN: سیگنال آسنکرون ریست
 - خروجی های سیستم نیز بدینگونه خواهد بود:
 - Cooler: روشن/خاموش کولر - تک بیتی
 - Heater: روشن/خاموش هیتر - تک بیتی
 - CRS: تعداد دور کولر (بین صفر تا ۸) - بدون علامت
 - OUT: روشن/خاموش فن (در نظر گرفته شده است که برای صرفه جویی در مصرف برق نه تنها سیگنال مربوط به دور فن صفر خواهد شد بلکه سیگنال خاموش نیز یک خواهد شد تا فن خاموش و در حالت استندبای قرار بگیرد)
- در صفحات بعد سه ماژول کنترلر فن، کنترلر هیتر و کولر و کنترل کننده این دو بخش به همراه توضیحات آن آمده است.

Power Control

این ماژول به روش One-Hot کار می‌کند. برای هر حالت که در نمودار دستور کار آزمایش آمده بود یک رجیستر در نظر گرفته شده است. زمانی که در استیت S3 باشیم هیتر روشن است و اگر در استیت S2 باشیم کولر روشن است. پس خروجی این رجیسترها را به خروجی‌های ماژول متصل می‌کنیم.

با استفاده از یک بلاک `always` در هنگام کلاک یا زدن سیگنال ریست وارد این بلاک خواهیم شد. اگر ریست فعال شده بود مدار را به حالت اولیه خودش می‌بریم. در غیر این صورت، با استفاده از حالت بندی کردن روی حالت فعلی و ورودی دما (همانند تعویض شدن استیت‌ها در نمودار) حالت‌های بعدی را مشخص می‌کنیم.

```
module power_control (
    T,
    Heater,
    Cooler,
    clk,
    rstN
);
    input signed [7:0] T;
    input clk, rstN;
    output Heater, Cooler;

    reg S1, S2, S3; // one-hot state indicators

    assign Heater = S3;
    assign Cooler = S2;

    always @(posedge clk or negedge rstN) begin
        if (~rstN) begin
            S1 <= 1;
            S2 <= 0;
            S3 <= 0;
        end
        else begin
            if (S1) begin
                if (T < 15) begin
                    S1 <= 0;
                    S3 <= 1;
                end
                if (T > 35) begin
                    S1 <= 0;
                    S2 <= 1;
                end
            end
            else if (S2) begin
                if (T < 25) begin
                    S2 <= 0;
                    S1 <= 1;
                end
            end
            else if (S3) begin
                if (T > 30) begin
                    S3 <= 0;
                    S1 <= 1;
                end
            end
        end
    end
end
endmodule
```

Fan Control

این ماژول نیز مانند ماژول قبلی به صورت One-Hot عمل می‌کند. برای این ماژول علاوه بر خروجی دور فن که یک عدد چهاربیتی بدون علامت است، یک خروجی یک بیتی برای روشن یا خاموش بودن فن نیز ارائه می‌شود تا بتوان فن را در حالت استندبای قرار داد.

این ماژول توسط دو بلاک `always` پیاده‌سازی شده است. یکی `sequential` و دیگری `combinational`. در بلاک ترکیبی با استفاده از استیت فعلی مدار مقدار دور فن تعیین می‌شود. در بلاک ترتیبی که با استفاده از لبه کلاک و لبه سیگنال روشن بودن کولر (برای روشن شدن فن بلافاصله پس از روشن شدن کولر و بدون وقفه) فعال می‌شود، با حالت بندی روی وضعیت فعلی و دمایی که از سنسور دریافت می‌شود وضعیت‌ها ممکن است تغییر کنند.

```
module fan_control (
    T,
    Cooler,
    CRS,
    OUT,
    clk
);
    input signed [7:0] T;
    input Cooler, clk;
    output reg [3:0] CRS;
    output reg OUT;

    reg S1, S2, S3; // one-hot state indicators

    always @(*) begin
        if (S1) begin
            CRS = 4;
        end
        else if (S2) begin
            CRS = 6;
        end
        else if (S3) begin
            CRS = 8;
        end
        else if (OUT) begin
            CRS = 0;
        end
    end

    always @(posedge clk or negedge Cooler or posedge Cooler) begin
        if (~Cooler) begin
            S1 <= 0;
            S2 <= 0;
            S3 <= 0;
            OUT <= 1;
        end
        else begin
            if (S1) begin
                if (T < 25) begin
                    S1 <= 0;
                    OUT <= 1;
                end
                if (T > 40) begin
                    S1 <= 0;
                    S2 <= 1;
                end
            end
            else if (S2) begin
                if (T < 35) begin
                    S2 <= 0;
                    S1 <= 1;
                end
                if (T > 45) begin
                    S2 <= 0;
                    S3 <= 1;
                end
            end
            else if (S3) begin
                if (T < 40) begin
                    S3 <= 0;
                    S2 <= 1;
                end
            end
        end
    end
end
```

```

        end
    end
    else if (OUT) begin
        if (T > 35) begin
            OUT <= 0;
            S1 <= 1;
        end
    end
end
end
endmodule

```

Digital Control

ای مازول صرفاً ورودی‌ها را گرفته و به دو مازولی که پیشتر معرفی شد می‌دهد و خروجی آن‌ها را نیز به خروجی سیستم متصل می‌کند.

```

module digital_control (
    T,
    Heater,
    Cooler,
    CRS,
    OUT,
    clk,
    rstN
);
    input [7:0] T;
    input clk, rstN;
    output Heater, Cooler, OUT;
    output [3:0] CRS;

    power_control pcu(.T(T), .Heater(Heater), .Cooler(Cooler), .clk(clk), .rstN(rstN));

    fan_control fcu(.T(T), .Cooler(Cooler), .CRS(CRS), .OUT(OUT), .clk(clk));
endmodule

```

۳ تست مدار

برای تست مدار دما از -10 بالا برده به 60 درجه رساندم و سپس از آنرا از 60 درجه به -10 درجه می‌رسانیم تا کارایی مدار را در تمام حالت‌های مدار تست کرده باشیم. نتایج تست مدار پایین‌تر آمده است (ابتدا نسخه تقطیع شده که تنها در زمان تغییر سیگنال‌ها تصاویر در کنار هم چیده شده است آمده است سپس در انتها نتیجه شبیه‌ساز از -10 تا -10).

کد تست

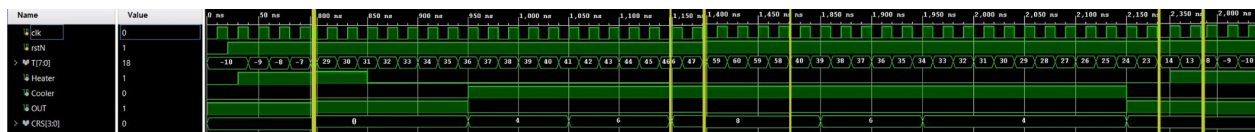
```
module digital_control_rand_tb ();
    reg clk, rstN;
    reg [7:0] T;
    wire Heater, Cooler, OUT;
    wire [3:0] CRS;

    digital_control dcu(.Heater(Heater), .Cooler(Cooler), .T(T), .clk(clk), .rstN(rstN), .OUT(OUT), .CRS(CRS));

    initial begin
        clk <= 0;
        forever #10 clk <= ~clk;
    end

    integer i = -10;
    initial begin
        T = -10;
        rstN <= 0;
        #20 rstN <= 1;
        for (i = -10; i <= 60; i = i + 1) begin
            T = i;
            #20;
        end
        for (i = 59; i >= -10; i = i - 1) begin
            T = i;
            #20;
        end
        end
        $finish;
    end
endmodule
```

نتایج تست نتیجه تقطیع شده



شکل ۱: نتیجه تقطیع شده

در ادامه نیز نتایج کامل که مربوطه به شبیه‌سازی است آمده است.



