

**پروژه درس طراحی سیستم های دیجیتال**

**استاد: دکتر فصحتی**

**صادق محمدیان**

**شماره دانشجویی:۴۰۱۱۰۹۴۷۷**

**تیرماه ۱۴۰۳**

**سیستم مدیریت پارکینگ:**

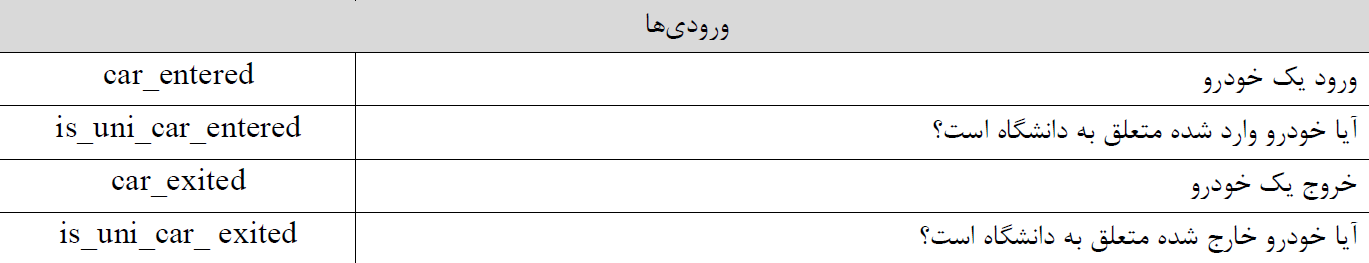
می خواهیم یک مدار برای مدیریت یک پارکینگ طراحی کنیم. این مدار باید ویژگی های زیر را داشته باشد:

۱) اولویت فضاي پارکینگ با اساتید و کارمندان دانشگاه است و این ظرفیت بر اساس آمار حداکثر 500 خودرو تعیین گردیده است.

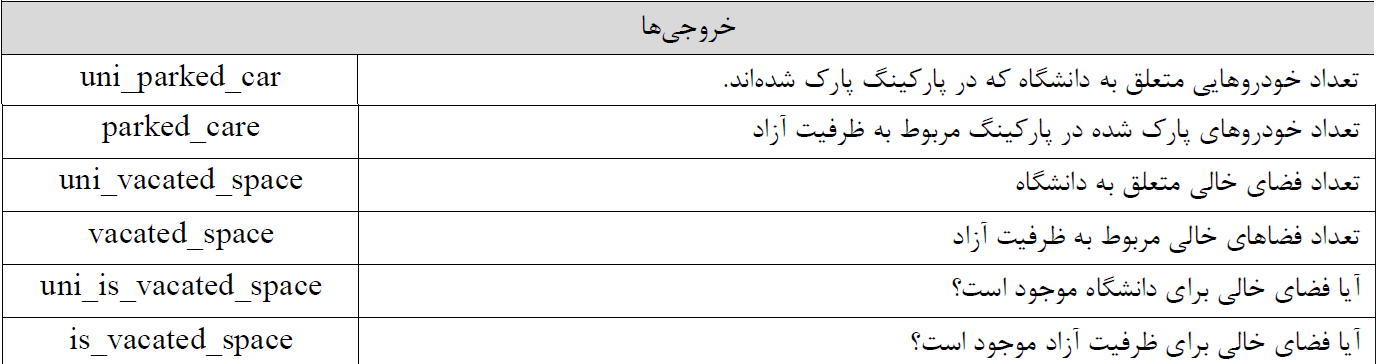
۲) باتوجه به اینکه فضاي کل پارکینگ 700 خودرو است از ساعت 8 ت ا 13 فقط 200 ظرفیت خالی براي ورود آزاد موجود است.

۳) از ساعت 13 تا 16 به ازاي هر ساعت ظرفیت ورود آزاد 50 خودرو افزایش مییابد و در ساعت 16 ظرفیت ورود آزاد به 500خودرو می رسد.

ماژول اصلی ما باید ورودی های زیر را به همراه کلاک و سیگنال شروع داشته باشد:



همچنین باید خروجی های زیر را نیز داشته باشد:



ابتدا بهتر است برای زمان و ساعت و ارتباط آن با ظرفیت پارکینگ یک ماژول جداگانه طراحی کنیم:

module timer (input start,clk,

              output reg [11:0] Time);

        reg [12:0] count;

        always @(posedge clk)

        begin

        if (~start)

        begin

            Time <= 0;

            count <= 0;

        end

        else

        begin

            if (count >= 60)

            begin

                Time <= Time + 1;

                count <= 0;

            end

            if (Time >= 23)

            begin

                Time <= 0;

            end

            else

            begin

                count <= count + 30;

            end

        end

    end

endmodule

همانطور که مشاهده میکنید از یک متغیر count استفاده می شود که با رسیدن به عدد ۶۰ صفر می شود و متغیر Time زیاد می شود . این ماژول به طور مداوم در حال کار است پس باید بعد از به پایان رسیدن روز صفر شود.

حال با شرایط گفته شده اقدام به طراحی ماژول مورد نظر می پردازیم:

در صورتی سیگنال شروع صفر شود تغییرات زیر در خروجی های ماژول ایجاد می گردد:

            uni\_car\_parked <= 0;

            parked\_care <= 0;

            uni\_vacated\_space <= universityCapacity;

            vacated\_space <= outCapacity;

            uni\_is\_vacated\_space <= 1'b1;

            is\_vacated\_space <= 1'b1;

در صورت ورود خودرو جدید منطق آن را در این قسمت پیاده سازی کرده ایم:

            if (car\_entered)

            begin

                if (parked\_care + uni\_car\_parked < MAX\_CAP\_PARKING)

                begin

                    if (is\_uni\_car\_entered)

                    begin

                        if ( 1 <= uni\_is\_vacated\_space && $signed(uni\_vacated\_space))

                        begin

                            uni\_car\_parked = uni\_car\_parked + 1;

                            uni\_vacated\_space = universityCapacity - uni\_car\_parked;

                            if (uni\_car\_parked >= universityCapacity)

                            begin

                                uni\_is\_vacated\_space = 1'b0;

                            end

                        end

                    end

                    else

                    begin

                        if (is\_vacated\_space && $signed(vacated\_space) >= 1)

                        begin

                            parked\_care = parked\_care + 1;

                            vacated\_space = outCapacity - parked\_care;

                            if (parked\_care >= outCapacity)

                            begin

                                is\_vacated\_space = 1'b0;

                            end

                        end

                    end

                end

            end

در صورت خروج خودرو منطق زیر را پیاده سازی کرده ایم:

            else if (car\_exited)

            begin

                if (is\_uni\_car\_exited)

                begin

                    if (uni\_car\_parked > 0)

                    begin

                        uni\_car\_parked = uni\_car\_parked - 1;

                        uni\_vacated\_space <= universityCapacity - uni\_car\_parked;

                        if (!uni\_is\_vacated\_space)

                        begin

                            uni\_is\_vacated\_space = 1'b1;

                        end

                    end

                end

                else

                begin

                    if (parked\_care > 0)

                    begin

                        parked\_care = parked\_care - 1;

                        vacated\_space <= outCapacity - parked\_care;

                        if (!is\_vacated\_space)

                        begin

                            is\_vacated\_space = 1'b1;

                        end

                    end

                end

            end

        end

در طراحی خود را این را در نظر گرفته ایم که ممکن است در ساعات مختلف که ظرفیت پارکینگ تغییر می کند تغییر ظرفیت خودروهای گروه خودرو های عمومی یا خودروهای مربوط به دانشگاه عددی منفی(!) باشد که در این صورت خودرو ها نمی توانند ورود داشته باشند و این تازمانی ادامه پیدا می کند که همه ی خودرو های اضافه بر ظرفیت خارج شوند.

و این قسمت نیز مربوط به تغییر ظرفیت های پارکینگ با توجه به ساعت است که باید به صورت زیر باشد:

**۱)** باتوج هبه اینکه فضاي کل پارکینگ 700 خودرو است از ساعت 8 ت ا 13 فقط 200 ظرفیت خالی براي ورود آزاد موجود است.

**۲)** از ساعت 13 تا 16 به ازاي هر ساعت ظرفیت ورود آزاد 50 خودرو افزایش مییابد و در ساعت 16 ظرفیت ورود آزاد به 500خودرو می رسد.

        //TIME AND CAPACITY

        if (Time < 8)

        begin

            universityCapacity = 0;

            outCapacity = 0;

        end

        else if (Time < 13)

        begin

            universityCapacity = 500;

            outCapacity = 200;

        end

        else if (Time < 14)

        begin

            universityCapacity = 450;

            outCapacity = 250;

        end

        else if (Time < 15)

        begin

            universityCapacity = 400;

            outCapacity = 300;

        end

        else if (Time < 16)

        begin

            universityCapacity = 350;

            outCapacity = 350;

        end

        else

        begin

            universityCapacity = 200;

            outCapacity = 500;

        end

صورت کلی ماژول به صورت زیر می باشد:

module Parking(input start, clk, car\_entered, is\_uni\_car\_entered, car\_exited, is\_uni\_car\_exited,

               output reg [15:0] uni\_car\_parked, reg [15:0] parked\_care, reg [15:0] uni\_vacated\_space, reg [15:0] vacated\_space, reg uni\_is\_vacated\_space, reg is\_vacated\_space, wire [11:0] Time);

    integer MAX\_CAP\_PARKING = 700;

    TIMER timer (clk,start,Time);

    integer universityCapacity;

    integer outCapacity;

        always @(car\_entered or car\_exited or start or Time)

        begin

        if (~start)

        begin

            uni\_car\_parked <= 0;

            parked\_care <= 0;

            uni\_vacated\_space <= universityCapacity;

            vacated\_space <= outCapacity;

            uni\_is\_vacated\_space <= 1'b1;

            is\_vacated\_space <= 1'b1;

        end

        else if (start)

        begin

            vacated\_space = outCapacity - parked\_care;

            uni\_vacated\_space = universityCapacity - uni\_car\_parked;

            if (car\_entered)

            begin

                if (parked\_care + uni\_car\_parked < MAX\_CAP\_PARKING)

                begin

                    if (is\_uni\_car\_entered)

                    begin

                        if ( 1 <= uni\_is\_vacated\_space && $signed(uni\_vacated\_space))

                        begin

                            uni\_car\_parked = uni\_car\_parked + 1;

                            uni\_vacated\_space = universityCapacity - uni\_car\_parked;

                            if (uni\_car\_parked >= universityCapacity)

                            begin

                                uni\_is\_vacated\_space = 1'b0;

                            end

                        end

                    end

                    else

                    begin

                        if (is\_vacated\_space && $signed(vacated\_space) >= 1)

                        begin

                            parked\_care = parked\_care + 1;

                            vacated\_space = outCapacity - parked\_care;

                            if (parked\_care >= outCapacity)

                            begin

                                is\_vacated\_space = 1'b0;

                            end

                        end

                    end

                end

            end

            else if (car\_exited)

            begin

                if (is\_uni\_car\_exited)

                begin

                    if (uni\_car\_parked > 0)

                    begin

                        uni\_car\_parked = uni\_car\_parked - 1;

                        uni\_vacated\_space <= universityCapacity - uni\_car\_parked;

                        if (!uni\_is\_vacated\_space)

                        begin

                            uni\_is\_vacated\_space = 1'b1;

                        end

                    end

                end

                else

                begin

                    if (parked\_care > 0)

                    begin

                        parked\_care = parked\_care - 1;

                        vacated\_space <= outCapacity - parked\_care;

                        if (!is\_vacated\_space)

                        begin

                            is\_vacated\_space = 1'b1;

                        end

                    end

                end

            end

        end

        //TIME AND CAPACITY

        if (Time < 8)

        begin

            universityCapacity = 0;

            outCapacity = 0;

        end

        else if (Time < 13)

        begin

            universityCapacity = 500;

            outCapacity = 200;

        end

        else if (Time < 14)

        begin

            universityCapacity = 450;

            outCapacity = 250;

        end

        else if (Time < 15)

        begin

            universityCapacity = 400;

            outCapacity = 300;

        end

        else if (Time < 16)

        begin

            universityCapacity = 350;

            outCapacity = 350;

        end

        else

        begin

            universityCapacity = 200;

            outCapacity = 500;

        end

    end

endmodule

حال ماژول اصلی است و باید آن را در حالت های مختف تست کنیم:

ابتدا ورودی ها و خروجی های مدار را میسازیم و از ماژول خود اینستنس میگیریم:

    reg clk;

    reg start;

    reg car\_entered;

    reg is\_uni\_car\_entered;

    reg car\_exited;

    reg is\_uni\_car\_exited;

    wire [15:0] uni\_car\_parked;

    wire [15:0] parked\_care;

    wire [15:0] uni\_vacated\_space;

    wire [15:0] vacated\_space;

    wire uni\_is\_vacated\_space;

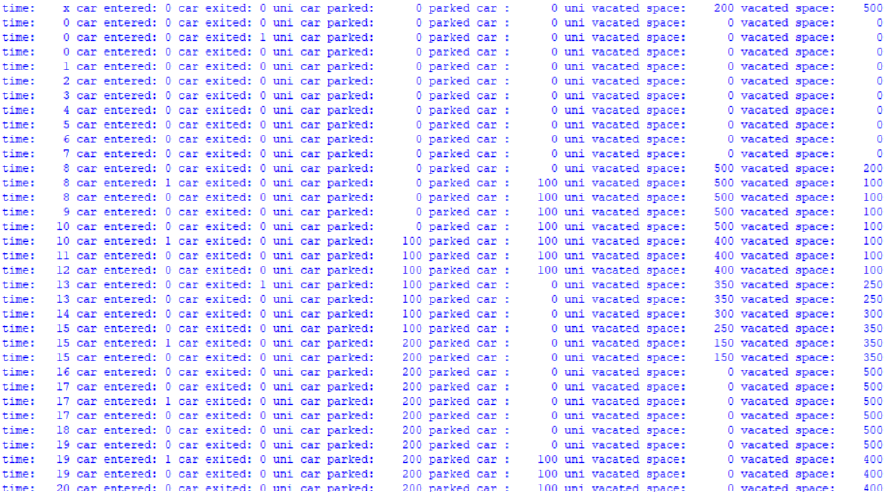
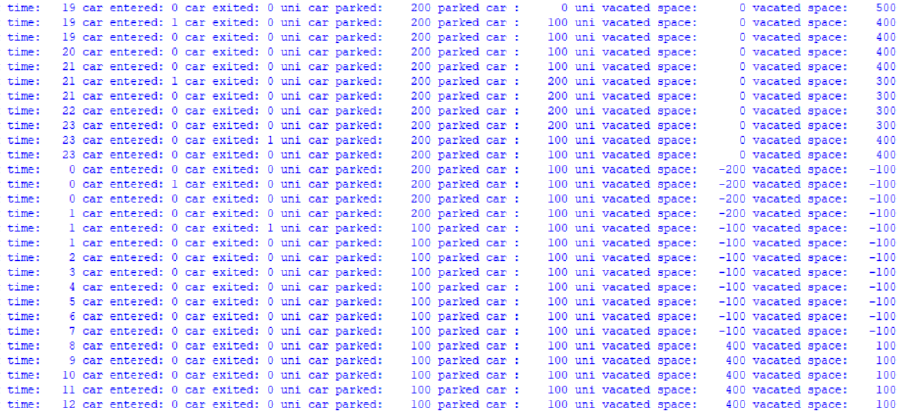
    wire is\_vacated\_space;

    wire [11:0] Time;

    Parking parking(clk,start,car\_entered,is\_uni\_car\_entered,car\_exited,is\_uni\_car\_exited,uni\_car\_parked,parked\_care,uni\_vacated\_space,vacated\_space,uni\_is\_vacated\_space,is\_vacated\_space,clock\_time);

    initial clk <= 0;

    always #5 clk <= ~clk;

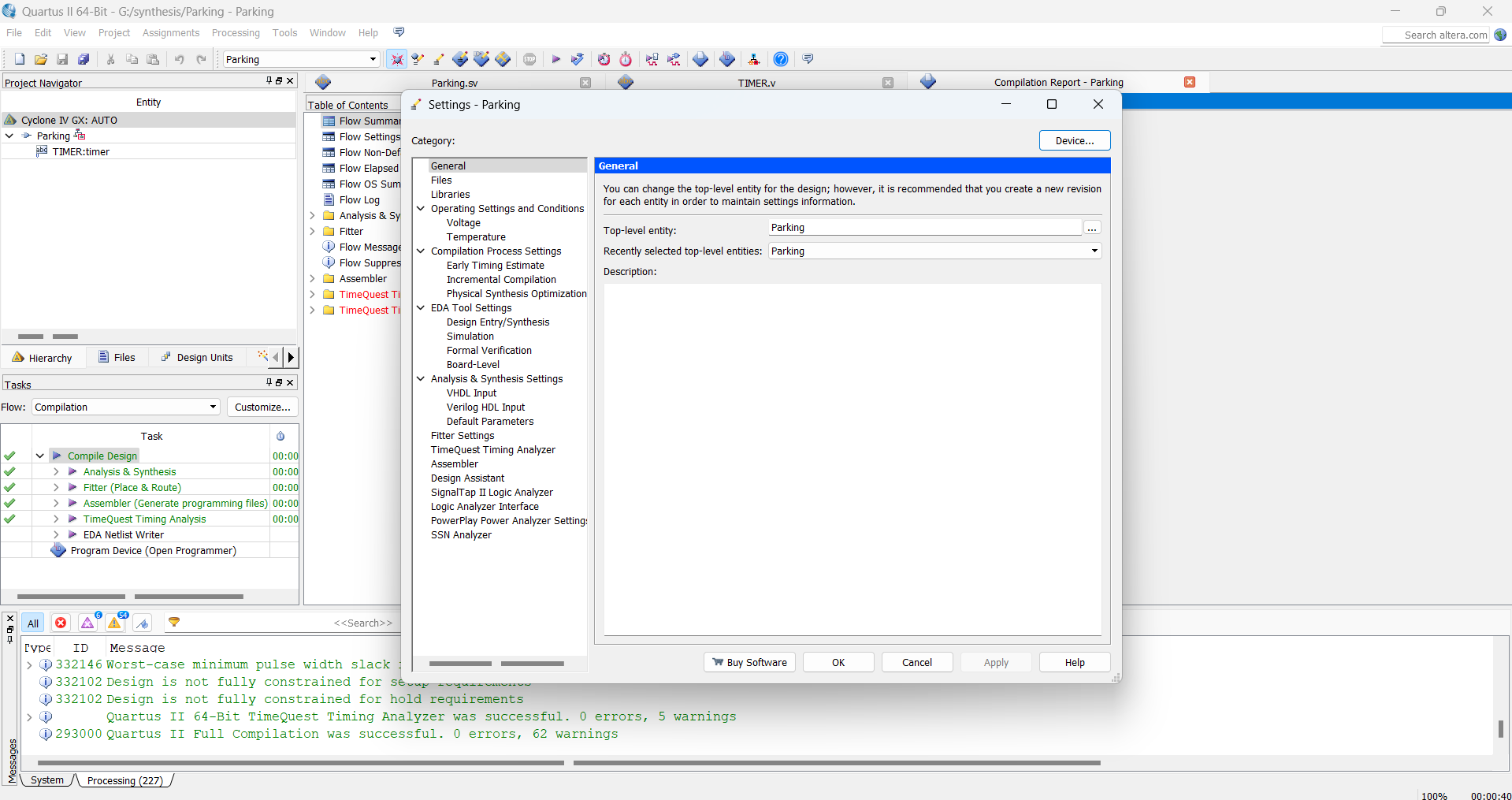
حال در یک بلاک initial تست های خودرا می نویسیم که در فایل مربوطه نوشته شده است به علت طولانی بو دن از آوردن آن صرف نظر میکنیم و نتایج آن را نشان می دهیم:.. با توجه به این نتایج میتوان نتیجه گرفت مدار ما به درستی کار می کند.

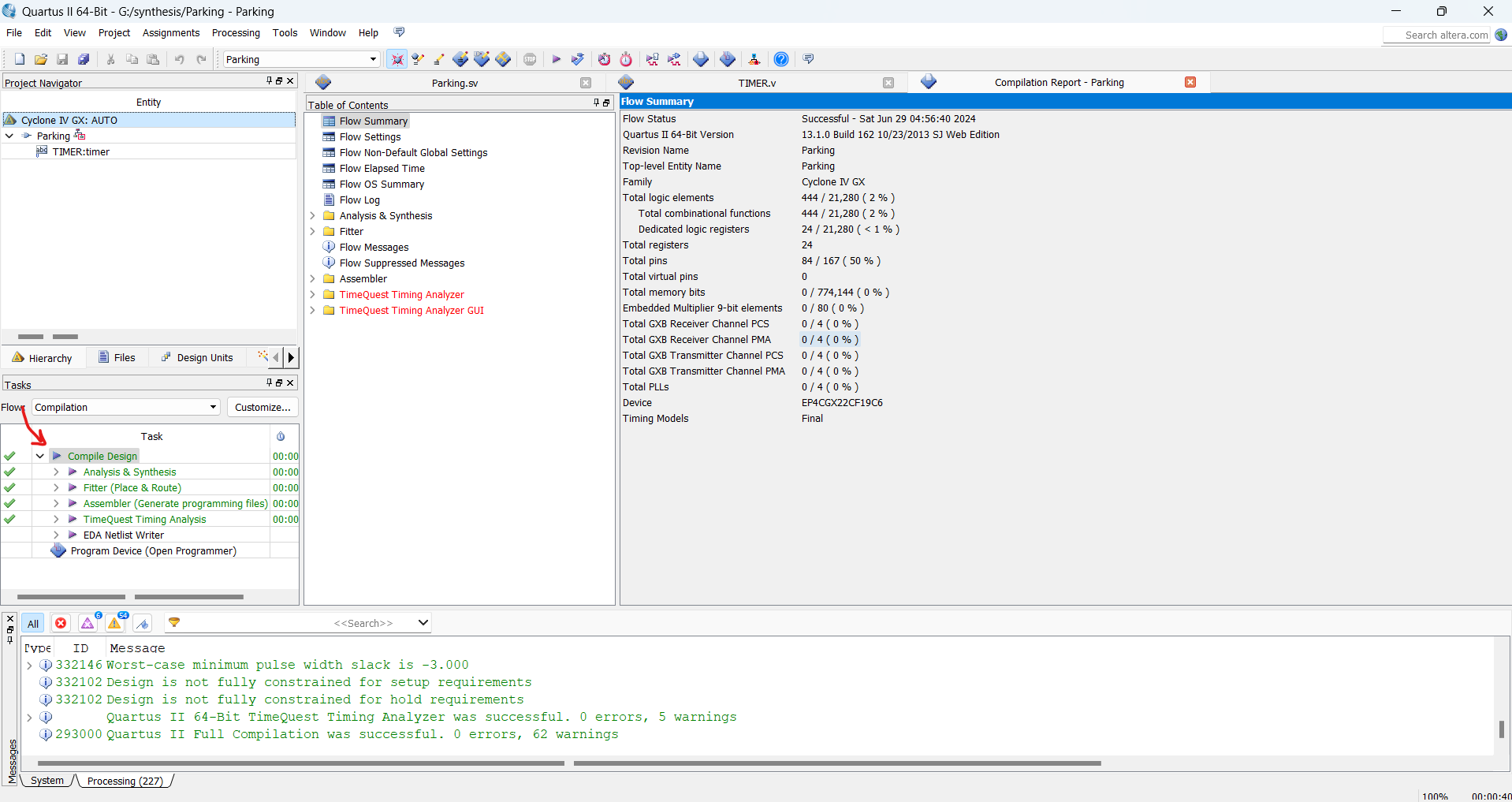
حال می خواهیم مدار خود را سنتز کنیم

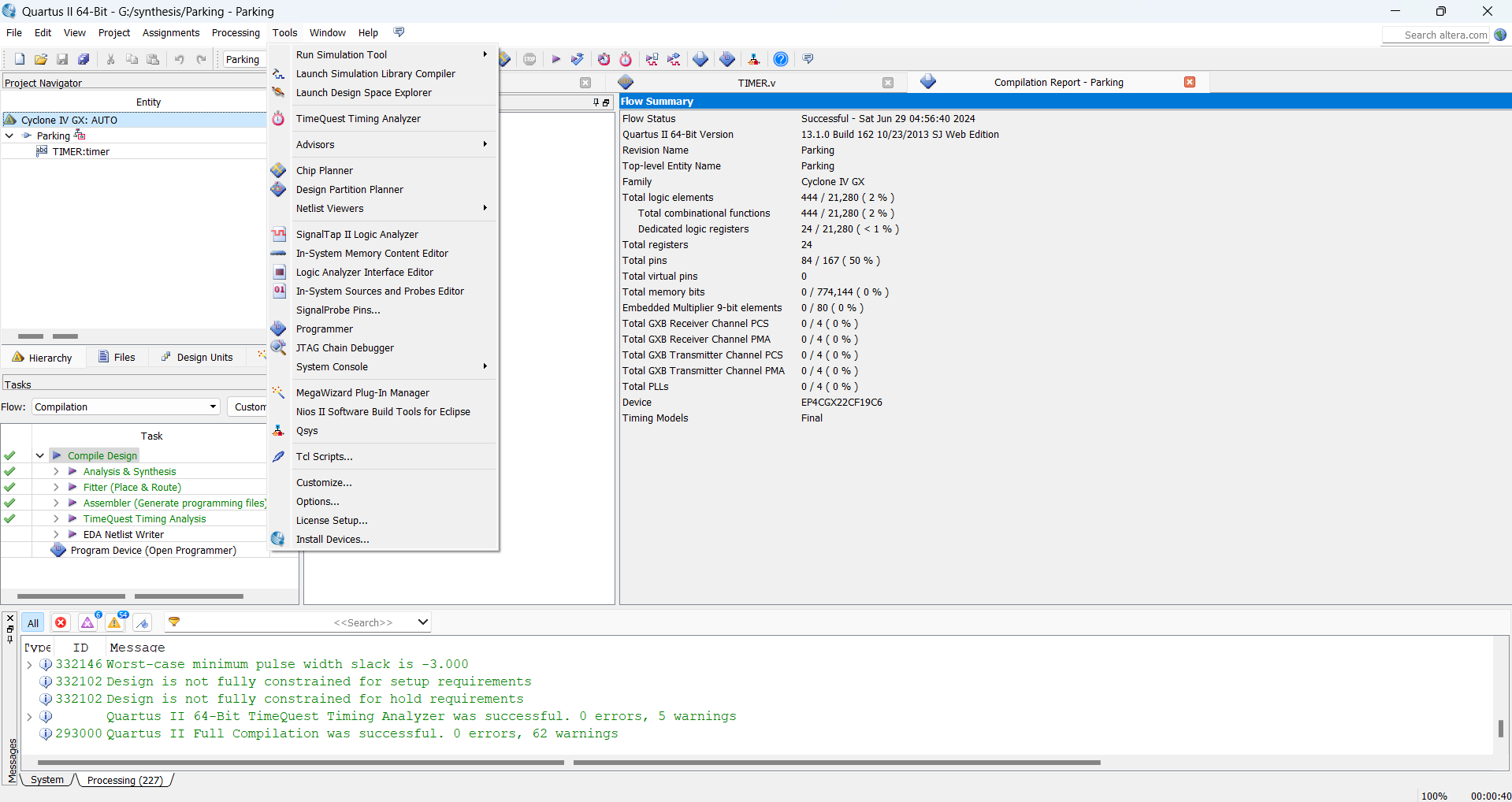
برای اینکار از نرم افزار کوارتوس کمک می گیریم و مراحل گفته شده را انجام می دهیم.

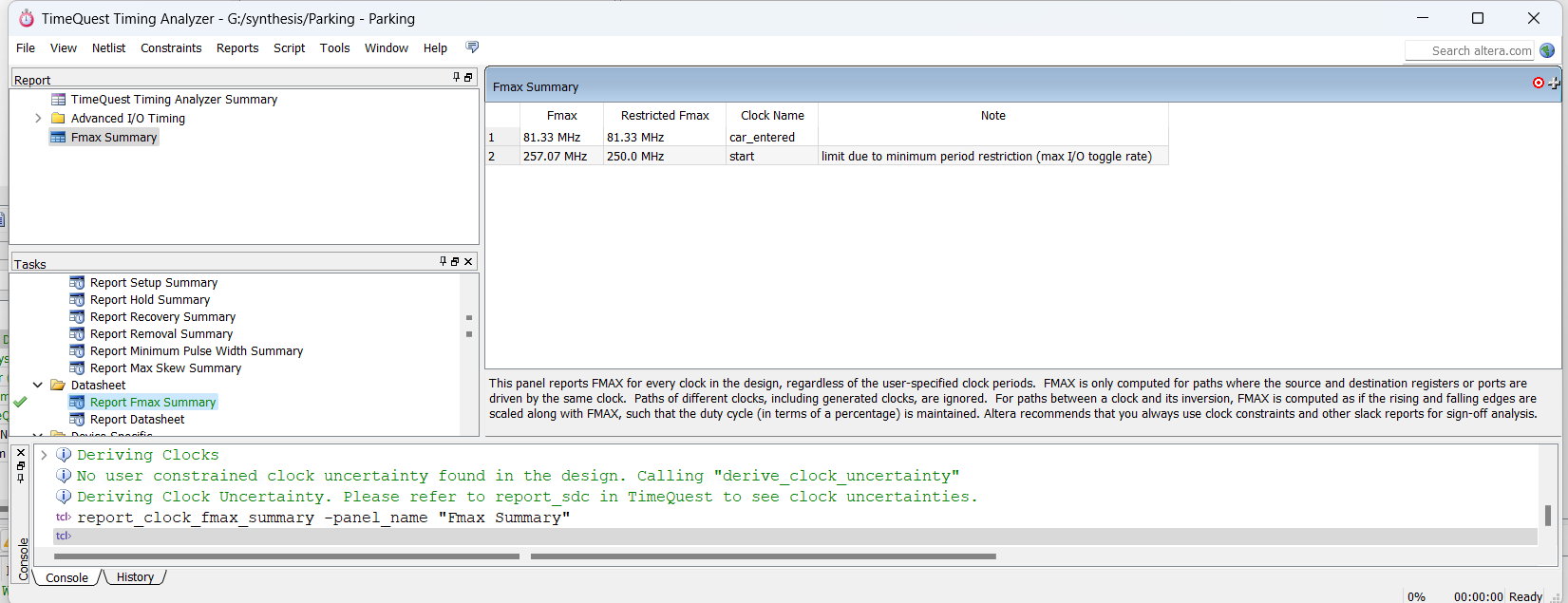
یک پروژه می سازیم و فایل های خود را آپلود می کنیم دقت کنید که باید system Verilog باشند.

نیازی به آپلود فایل تست نمی باشد.

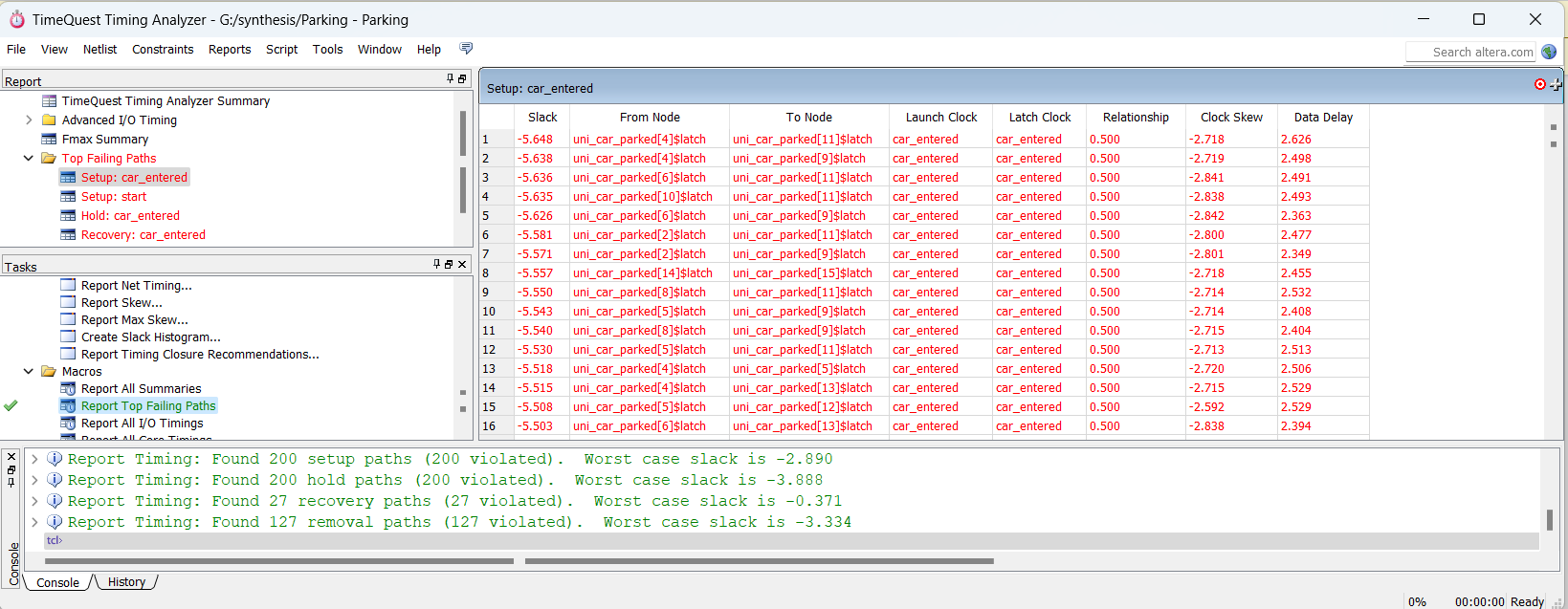
سپس ماژولtop level خود را انتخاب می کنیم:

حال برنامه خود را کامپایل میکنیم.این مرحله ممکن است کمی زمان بر باشد:

سپس در منوی tools گزینه ی timeQuest Timming Analyzer را انتخاب می کنیم:

و در منوی سمت چپ به ترتیب create timming netlistسپس read sdc fileو سپس report fmax summary را برای دیدن فرکانس ماکزیمم میزنیم.

این مقدار 257.07مگاهرتز در حالت بدون محدودیت I/Oو 250مگاهرتز در حالت با محدودیت I/O می باشد.

و سپس report top falling paths را میزنیم تا مشاهده کنیم مسیر هایمان چقدر تاخیر دارند.اگر این تاخیر دیتا را معکوس کنیم با درنظر گرفتن تقریب دهگان هم اندازه ی فرکانس ما می شود.

چون عامل موثر برآن فرکانس تاخیر مسیر بحرانی می باشد.

برای دیدن فیلم نحوه سنتز با کوارتوس می توانید فیلم از لینک زیر را مشاهده فرمایید:

https://drive.google.com/file/d/1JLdJ2Z\_U5Z84exfwDvQMOEGFVdJlFThT/view?usp=sharing