



نکات مهم:

- (1) پروژه ها برای گروه دو نفره (ارشدها بصورت تک نفره) است. اعضای گروه و پروژه انتخابی را تا تاریخ دوشنبه 11 آذر ایمیل کنید.
- (2) پروژه باید توسط نرم افزار *Altera Quartus* (برای بورد *Altera DE0-CV*) یا *Xilinx ISE* (برای بورد *Spartan6*) پیاده سازی شود.
- (3) کل پوشه های مربوط به پیاده سازی و گزارش را در یک فایل zip شده به نام "CADProject_Name1_Name2" قرار دهید.
- (4) قبل از زمان تحویل حضوری، فقط یک فایل zip شده به آدرس زیر ارسال کنید:
<https://www.dropbox.com/request/3knDRN3iOOxdTWABo7Nh>
- (5) زمان تحویل فاز اول پروژه، (احتمالا بصورت حضوری)، تا تاریخ سه شنبه 26 آذر خواهد بود.
- (6) زمان تحویل نهایی پروژه، بصورت حضوری، در تاریخ احتمالی یکشنبه 29 دی خواهد بود (تاریخ دقیق متعاقبا اعلام خواهد شد).
- (7) توجه مهم: تمام اعضای گروه باید به تمام بخش های پروژه مسلط باشند و ندانستن بخشی به بهانه تقسیم کار پذیرفته نیست.

در این پروژه می توانید از فایل *CAD_VGA_Quartus* (برای *Quartus*) یا *CAD_VGA_Xilinx* (برای *Xilinx*) که روی دراپ باکس (در پوشه *CAD_Tool*) قرار داده شده است کمک بگیرید. قالب اصلی این فایل را تغییر ندهید تا بتوانید Pin-Assignment مربوط به بوردها را استفاده نمایید. در این فایل، کنترلر VGA پیاده سازی شده است تا بتوان داده را از طریق پورت VGA روی مانیتور نشان داد. همچنین، برای مثال اولیه، یک مربع متحرک نیز روی مانیتور نمایش می دهد که هرگاه به انتهای صفحه برسد باز می گردد. چند خروجی نمونه نیز روی 7Segment ها نمایش داده شده است. ابتدا کد قرار داده شده را بصورت کامل نگاه کنید و کاملا رفتار کد را تحلیل کنید. سپس این کد را باید بگونه ای تکمیل کنید که بازی زیر را پیاده سازی کند.

فاز اول: در فاز اول پروژه کفایت صفحه بازی (نمایش صفحه بازی، مهره بازیکن و چند مانع بصورت رندوم در صفحه بازی و با فاصله) را آماده کنید.

Atari 2600: River Raid

هدف از این پروژه پیاده سازی بازی River Raid است. همانند شکل زیر، یک هلی کوپتر (در اینجا یک مربع) در صفحه بازی قرار دارد که قابلیت حرکت به چپ و راست را دارد. صفحه بازی که دارای موانعی نیز هست بصورت مداوم در حال حرکت به سمت پایین است و هلی کوپتر باید از برخورد به این موانع اجتناب کند. بازیکن باید به کمک کلیدهای بازی (Button ها) و حرکت هلی کوپتر به سمت چپ و راست موانع را رد کند.

کد قرار داده شده را باید به گونه ای تکمیل کنید که صفحه نمایش بازی بصورت متحرک، مهره بازیکن و موانع را بصورت رندوم در محدوده مجاز بازی نمایش دهد. بعد از ریست که بازی شروع می شود، یک مربع (جایگزین هلی کوپتر) در وسط صفحه نمایش قرار می گیرد و با هر بار فشار دادن Button ها مربع به سمت چپ یا راست حرکت کند. صفحه مجاز بازی و موانعی که به صورت رندوم در محدوده صفحه بازی قرار گرفته اند بصورت دایم در حال حرکت به سمت پایین هستند. بازیکن باید با کنترل مربع (هلی کوپتر) از برخورد به موانع اجتناب کند.



برای پروژه مراحل زیر را پیاده‌سازی کنید:

- ✓ بعد از زدن کلید Reset، روی نمایشگرهای 7Segment روی بورد دو رقم انتهایی شماره دانشجویی اعضای گروه نمایش داده شود، و دو نمایشگر سمت چپ مقدار صفر را نشان دهند.
- ✓ روی مانیتور نیز مربع (هلی کوپتر) و صفحه بازی با موانع بصورت غیرمتحرک نمایش داده شود تا یکی از Button ها زده شوند. تمام LED ها نیز خاموش باشند.
- ✓ با زدن یکی از Button ها بازی آغاز شود و صفحه بازی همراه با موانع شروع به حرکت به سمت پایین کند و بازیکن با فشار دادن یکی از Button ها هلی کوپتر را کنترل کند.
- ✓ دو 7Segment میانی زمان گذشته از بازی را نشان دهند.
- ✓ دو 7Segment امتیاز بازیکن را نشان دهند، بدین صورت که هر بار هلی کوپتر مانعی را رد کرد یک امتیاز به بازیکن اضافه گردد.
- ✓ هر 15 ثانیه یا رد کردن 10 مانع باید سرعت حرکت موانع را افزایش دهید.
- ✓ شرط انتهایی بازی، رسیدن زمان به 99 ثانیه، یا رسیدن امتیاز یک بازیکن به 10 یا سوختن هلی کوپتر است. شرط سوختن هلی کوپتر نیز برخورد به موانع می باشد.
- ✓ در این حالت بازی متوقف شود و هلی کوپتر حرکت نکند. تمام LED های بورد نیز روشن شوند.



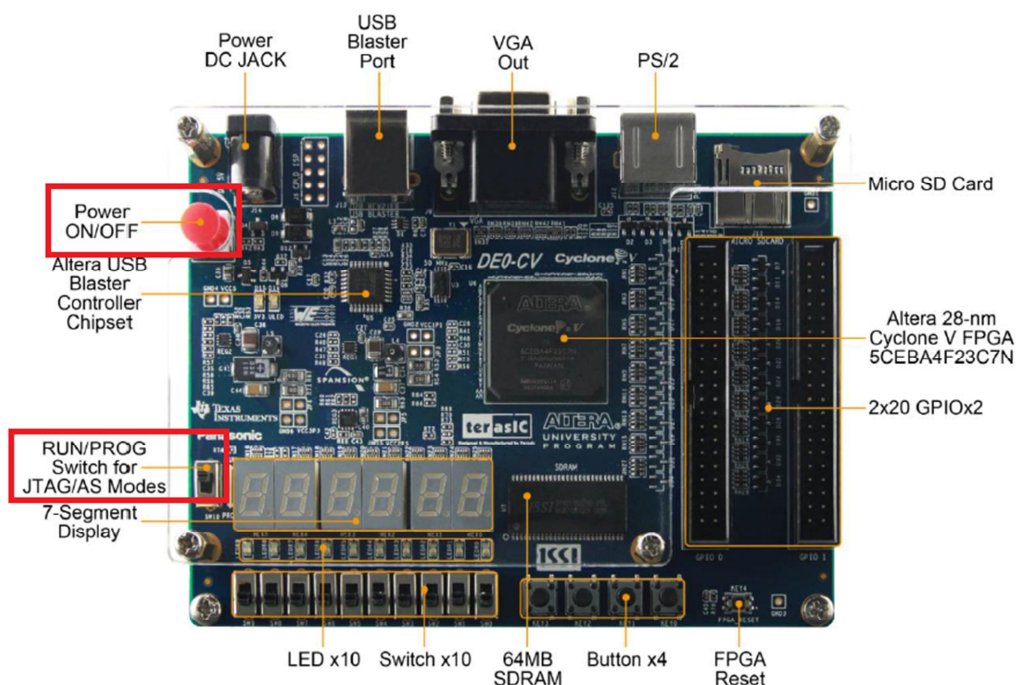
بخش اختیاری:

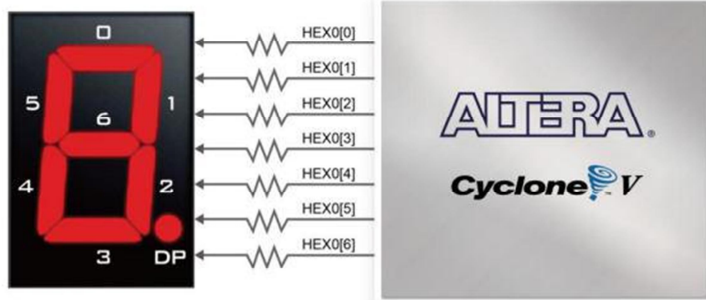
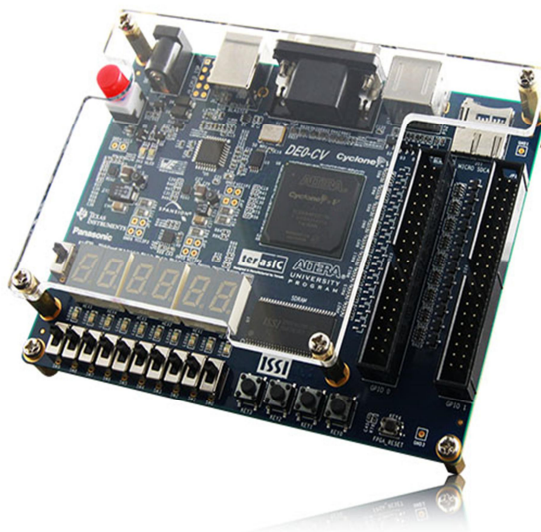
- بخش اختیاری اول (+10%): هلی کوپتر بصورت خودکار و هوشمند بازی کند.
- بخش اختیاری دوم (+20%): هلی کوپتر امکان تیراندازی داشته باشد و صرفاً اگر مانعی را با تیر زد امتیاز کسب کند (مانع نیز بعد از اصابت تیر حذف شود).
- زیبایی بازی می‌تواند 5% نمره اضافی داشته باشد.

توجه مهم: در بورد Altera-DE0، کلید Reset و Button ها، و خروجی‌های مربوط به 7Segment ها همه بصورت Active-Low هستند.

برخی نکات:

- ✓ ورودی‌های پروژک Clk، Reset_n و دو Button است.
- ✓ نمای بورد را در پایین مشاهده می‌کنید. بورد دارای شش نمایشگر، و تعدادی Button و LED و سوییچ (SW) است.
- ✓ برای **Push-Button** ها از پورت‌های KEY (بصورت Active-Low) استفاده کنید.
- ✓ برای نمایشگرهای **7Segment** (6 نمایشگر، هر کدام شامل 7 پین Active-Low) از پورت‌های HEX0 تا HEX5 استفاده کنید.
- ✓ برای **LED** ها از پورت‌های LEDR استفاده کنید.
- ✓ برای **سوییچ** ها از پورت‌های SW استفاده کنید.
- ✓ برای نمایش چهاربیت روی 7Segment می‌توانید از تابع convSEG داخل برنامه کمک بگیرید.
- ✓ برای فعال کردن **USB-Blaster**، بعد از اتصال بورد به کامپیوتر به کمک فایل "QUARTUS II" که روی درآپ‌باکس هست، مراحل نصب درایور USB-Blaster را انجام دهید.





توجه (تولید اعداد تصادفی): برای تولید اعداد تصادفی می‌توانید از کد زیر استفاده کنید و به تعداد بیت‌هایی که لازم دارید از خروجی‌های آن استفاده کنید:

```
process(clk)
-- maximal length 32-bit xnor LFSR
function lfsr32(x : std_logic_vector(31 downto 0)) return std_logic_vector is
begin
    return x(30 downto 0) & (x(0) xnor x(1) xnor x(21) xnor x(31));
end function;
begin
    if rising_edge(clk) then
        if resetn='0' then
            pseudo_rand <= (others => '0');
        else
            pseudo_rand <= lfsr32(pseudo_rand);
        end if;
    end if;
end process;
```

- برای LFSR سی و دو بیتی (LFSR-32)، بیت‌های 0 و 1 و 21 و 31 را با هم XNOR کنید.
- برای LFSR شصت و چهار بیتی (LFSR-64)، بیت‌های 59 و 60 و 62 و 63 را با هم XNOR کنید.
- برای LFSR صد و بیست و هشت بیتی (LFSR-128)، بیت‌های 98 و 100 و 125 و 127 را با هم XNOR کنید.

موفق باشید

آریان زودی، اسما ناصری‌راد
مهدی امینیان