

نکته 1: انتخاب پروژه لازم است تا دوشنبه 22 آذر انجام و به دستیاران محترم آزمایشگاه اعلام شود. هر کدام از پروژه هایی که سخت افزاری نیست تنها توسط یک گروه قابل انجام است. لذا اگر پروژه ای سخت افزاری نیست و بیش از یک گروه متقاضی دارد، انتساب پروژه به صورت رندوم توسط دستیاران محترم انجام خواهد شد. (در این صورت هر گروه لازم است لیستی از پروژه های مورد نظر را به دستیاران محترم ارائه دهد).

نکته 2: گزارش کار برای تمام پروژه ها لازم است حداکثر تا تاریخی که اعلام می شود در درس افزار آپلود شود. پروژه های دو نفره می توانند یک گزارش کار داشته باشند. در گزارش کار باید به مشکلات کار و طریقه برطرف کردن آنها پرداخته شود.

نکته 3: تعلق گرفتن نمره تشویقی به کیفیت انجام پروژه، کیفیت گزارش کار و انجام قسمتهای اضافه بستگی دارد. تنها پروژه های شماره 1 و 2 در صورتی که توسط دو نفر انجام شوند قابلیت گرفتن نمره تشویقی را دارند.
نکته 4: لازم است هر پروژه (به همراه گزارش کار آن) توسط دو دستیار آموزشی ارزیابی شود.

1) پیمانه 2: (پروژه 2 نفره - اگر دو نفره انجام شود هم قابلیت گرفتن نمره تشویقی را دارد.)
مدار شما باید یک ورودی 8 بیتی دریافت کند و اگر ورودی فرد بود عبارت ODD به نحو مقتضی بر روی 3 سون سگمنت نشان داده شود و اگر ورودی زوج بود، قسمت فرد آن را بر روی 3 سون سگمنت نشان دهد (توضیح اینکه قسمت فرد هر عدد بزرگترین مقسوم علیه فرد آن است!!)

پیاده سازی : به صورت سخت افزاری

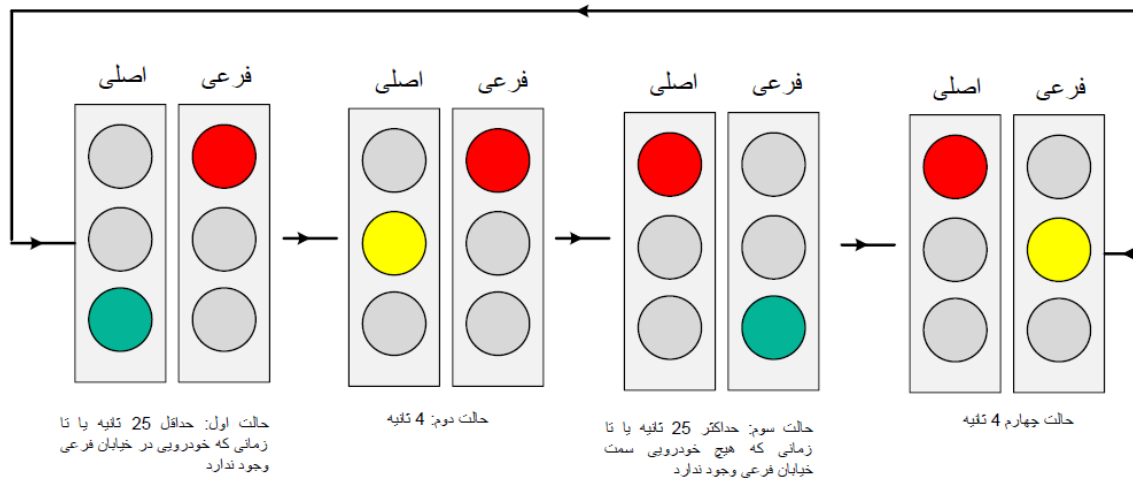
2) اندازه گیری زمان پاسخ : (پروژه 2 نفره - - اگر دو نفره انجام شود هم قابلیت گرفتن نمره تشویقی را دارد.)
توسط گیت های منطقی ، سیستمی را طراحی کنید که ابتدا یک LED را روشن می کند و سپس بعد از گذشت زمانی رندوم این LED را خاموش می کند .بعد از آن کاربر باید دکمه را در سریع ترین زمان ممکن فشار دهد .بعد از فشردن دکمه، سیستم زمان واکنش فرد را روی 3 سون سگمنت نمایش می دهد .در صورتی که کاربر دکمه را زودتر از خاموش شدن LED فشار داد، نباید زمانی برای او محاسبه شود.
پیاده سازی : به صورت سخت افزاری

3) زنگ ساعت: (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)
یک ساعت دیجیتال که دقیقه و ساعت را در 4 سون سگمنت نمایش می دهد طراحی کنید. دقیقه ها میتوانند با کلاکی که سرعت نسبتا بالایی دارند، افزایش یابند (گویی زمان با سرعت در حال گذر است). حال این ساعت را میتوان برای زمان خاصی تنظیم کرد که در آن ساعت یک چراغ روشن شود. به این منظور باید یک زمان خاصی را انتخاب کرده و یک دکمه (دکمه ی Set) برای لحظه ای زده شود تا زمان در مدار ذخیره شود طوری که با تغییر زمان انتخابی، تا وقتی که دکمه ی Set زده نشده، زمان زنگ تغییر نکند. حال با رسیدن زمان به زمان تنظیم شده، چراغ مورد نظر روشن خواهد شد. یک دکمه هم برای reset کردن زنگ قرار دهید تا با فشردن آن، دیگر در زمان set شده زنگ نزنند.

پیاده سازی : به صورت سخت افزاری

4) چراغ راهنمایی: (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)

یک تقاطع بین یک خیابان اصلی با ترافیک سنگین و یک خیابان فرعی با ترافیک سبک را در نظر بگیرید. زمان بندی های مربوط به چراغ های راهنمایی این دو خیابان به صورت زیر می باشد.



در کنار عملکرد خودکار سیستم، باید امکان انتقال به حالت اول و سوم به صورت دستی (توسط پلیس) وجود داشته باشد. همچنین پلیس می تواند وضعیت را در یکی از حالت های اول و سوم نگه دارد. علاوه بر این برای ساعات خلوت شب، چراغ های راهنمایی باید بتواند در حالت چشمک زن قرار گیرند.

پیاده سازی: FPGA

(5) بازی سرعت عمل (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد).

ابتدا بازیکن عددی را بین 0 و 9 وارد میکند.

یک دکمه داریم. به محض فشار دادن آن توسط کاربر، اعدادی با سرعت پایین روی 7segment شروع به چرخش بین 0 تا 9 میکنند و وقتی کاربر دستش را از روی دکمه برداشت می ایستند. اگر عددی که روی آن ایستاده با عدد وارد شده برابر بود کاربر 1 امتیاز میگیرد در غیر اینصورت هیچ.

سپس اگر کاربر درست وارد کرده بود وارد مرحله بعدی میشود به این صورت که سرعت چرخش اعداد روی 7segment دو برابر میشود و این روند ادامه میابد تا جایی کاربر بسوزد (یعنی در مرحله ای عدد روی عدد وارد شده نایستد).

در ضمن در هر مرحله امتیاز درست وارد کردن 2 برابر مرحله قبل است.

فرکانس ورودی در اختیار خود شماست.

در آخر وقتی کاربر سوخت باید امتیاز روی 7segment نمایش داده شود.

پیاده سازی: FPGA وبا استفاده از 7seg های برد

(6) کرنومتر دیجیتال (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد).

یک کرنومتر دیجیتال بسازید که از لحظه Reset شدن، صفر شود و شروع به شمارش دهم ثانیه، ثانیه، دقیقه و ساعت کند. مدار یک دکمه دارد که با فشردن آن شمارش متوقف می شود و با فشردن مجدد، مدار Reset می شود. تنها مجاز به استفاده از یک کریستال با فرکانس دلخواه (به عنوان اسیلاتور)، فقط یک نوع آی سی شمارنده دلخواه، آی سی 74LS74 (D-FF) و گیت های منطقی هستید. از تعداد لازم 7-Segment برای نمایش اعداد استفاده کنید و اعداد را با علامت "." روی 7-Segment جدا کنید.

** قسمت اضافه: اعداد خروجی را روی یک LCD نمایش دهید.

پیاده سازی: به صورت سخت افزاری

(7) جمع‌کننده ممیز شناور ترکیبی : (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)
کد وریلاگ یک جمع‌کننده ممیز شناور 32 بیتی (استاندارد IEEE-754) را بنویسید که دو عدد ممیز شناور را به عنوان ورودی دریافت کند و خروجی 32 بیتی را با همان فرمت IEEE-754 تحویل دهد. مدار سنتز شده این کد باید کاملاً ترکیبی و بدون حافظه باشد، بنابراین نباید در کد از بلاک `always` استفاده شود. تمام `assignment` ها باید به صورت `Continuous assignment` باشند. ****** قسمت اضافه : یک `Test Bench` برای ماژول خود بنویسید و عملکرد کد وریلاگ را با آن تست کنید.

پیاده سازی : فقط شبیه‌سازی

(8) تشخیص جواب‌های مسئله 8 وزیر : (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)
ماژول یک عدد 8 رقمی (32 بیتی) `BCD` را به عنوان ورودی دریافت می‌کند و با یک `LED` در خروجی، تعیین می‌کند که آیا این عدد جوابی برای مسئله 8 وزیر هست یا خیر.
هر عدد `BCD`، نشان‌دهنده سطر قرار گرفتن هر وزیر، در هر یک از ستون‌ها است (فرض شده است وزیرها هر کدام در یک ستون قرار گرفته‌اند و یکدیگر را به صورت ستونی تهدید نمی‌کنند). یک جواب در صورتی درست است که هیچ‌کدام از وزیرها یکدیگر را به هیچ صورتی (سطری، ستونی، اریب) تهدید نکنند.
مثال :

ورودی نمونه $\rightarrow (12345678)BCD$: قرار گرفتن وزیرها به صورت قطری
یک جواب صحیح $(74286135)BCD$:

پیاده سازی : به صورت سخت‌افزاری

(9) پیاده‌سازی پروتکل ارتباطی I2C : (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)
کد وریلاگ دو ماژول فرستنده (`Master`) و گیرنده (`Slave`) برای ارتباط روی باس `I2C` را بنویسید. کدها را روی دو `FPGA` پیاده‌سازی کرده و با اتصال آن‌ها، باس `I2C` را تحقق دهید. فرستنده یک عدد بدون علامت 8 بیتی را می‌فرستد و گیرنده به وسیله بیت تأیید (`Ack`)، تعیین می‌کند که آیا این عدد بزرگ‌تر از 50 بوده است یا خیر. هر ماژول، پس از پایان ارسال هر `Data Pack`، عدد 8 بیتی یا بیت `Ack` فرستاده شده از ماژول دیگر را به وسیله `LED` نمایش می‌دهد.

****** توضیحات : `I2C` یک پروتکل ارتباطی است که روی یک باس 2 بیتی اجرا می‌شود. روی یکی از این دو خط، `Master` (که در اینجا فرستنده است)، سیگنال کلاک سنکرون را می‌فرستد (خط `SCL`). روی خط دیگر نیز داده به صورت سریال ارسال می‌گردد (خط `SDA`).
آغاز و پایان ارسال هر `Data Pack`، توسط `Master` و با انجام `Start` و `Stop` انجام می‌شود. به صورت یک لبه منفی روی `SDA` در زمان `High` بودن `SCL`، و `Stop` به صورت یک لبه مثبت روی `SDA` در زمان `High` بودن `SCL` است. بین `Start` و `Stop`، در طی 9 پریود کلاک، ابتدا 8 بیت داده توسط فرستنده ارسال می‌شود و سپس گیرنده، یک بیت `Ack` به منظور تأیید یا عدم تأیید بایت ارسال شده، می‌فرستد. اگر `Master` قصد ارسال شدن دوباره داده را داشته باشد، به جای `Stop`، دوباره `Start` می‌دهد.

در هنگام ارسال دیتا روی خط `SDA` توسط هر یک از طرفین، طرف دیگر حق درایو کردن این خط را ندارد. خواندن داده ارسال شده توسط گیرنده، در لبه مثبت کلاک انجام می‌شود؛ پس تغییر سطح ولتاژ خط `SDA` توسط فرستنده، باید بین لبه‌های مثبت کلاک انجام گیرد.

****** قسمت اضافه: قابلیت `Clock Stretching` توسط `Slave` را به مدار اضافه کنید. به این صورت که پس از پایان ارسال یک `Data Pack`، `Slave` می‌تواند خط `SCL` را به مدت دلخواه `Low` نگه دارد تا دیگر ارسال داده انجام نگیرد.

10) کنترل کننده آسانسور (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)

مدار باید آسانسور یک ساختمان 4 طبقه را پیاده کند به طوری که یک 7segment شماره طبقه آسانسور را نشان بدهد. دو سری 4 تایی کلید داریم. یکی سری مربوط به کلید های درونی آسانسور است که مسافران فشار میدهند. دیگری مربوط به کلید های طبقات است که برای ایستادن آسانسور در آن طبقه فشرده میشود (مثل یک آسانسور واقعی). آسانسور باید همانند یک آسانسور معمولی جهت حرکت خود به بالا یا پایین را حفظ کند تا رسیدن به طبقه آخر یا همکف. آسانسور دو دکمه کنترل هم دارد که با فشردن اولی آن متوقف شده و منتظر فشرده شدن یکی از 4 کلید مربوط به طبقات میشود. سپس بعد از فشردن کلید یکی از طبقات باید منتظر دوباره فشرده شدن کلید کنترلی دوم شود تا دوباره شروع به حرکت کند با این تفاوت که دیگر در طبقه ای که کلید آن فشرده شده نباید بایستد. (در واقع در این حالت آسانسور اینطور تنظیم میشود که دکمه ی فراخوانی آسانسور در طبقه ی مورد نظر دیگر کار نمیکند و از درون آسانسور هم نمیتوان با فشردن کلید مربوط به آن طبقه، به آن طبقه رفت و گویی کلید مربوط به آن طبقه غیر فعال شده است). یک کلید ریست هم باید باشد که با زدن آن آسانسور تمام تنظیمات قبلی را پاک کند. آسانسور باید در هر طبقه به مدت لازم بایستد.

پیاده سازی : FPGA

11) فرکانس متر : (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)

مدار شما باید پریود یک سیگنال را مشخص کند و عدد بدست آمده را بر روی 2 سون سگمنت به صورت یک عدد 2 رقمی (برحسب میلی ثانیه) نمایش دهد. فرکانس ورودی بین 100 هرتز تا 10 کیلو هرتز است. دقت شود که برای فرکانس های بالای 1 کیلو هرتز، پریود به صورت \overline{xy} و برای فرکانس های کمتر از 1 کیلو هرتز، پریود به صورت \overline{xy} نمایش داده میشود. (از علامت ' ' موجود بر روی سون سگمنت استفاده کنید).

پیاده سازی : FPGA وبا استفاده از 7seg های برد

12) نمایشگر LCD : (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)

مدار شما باید با زدن دکمه RST نام شماره دانشجویی شما و همگروهیتان را در یک LCD به صورت روان نمایش دهد. با هر بار زدن دکمه ریست این فرایند از اول تکرار می شود.

پیاده سازی : به صورت سخت افزاری

13) ساعت دیجیتال : (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)

مدار شما باید ساعت و دقیقه را در مجموع روی 4 سون سگمنت نمایش دهد و با زدن یک دکمه به مدت 4 دقیقه تاریخ را نمایش دهد (ماه و روز). می توانید برای دقیقه، یک کلاک سریع تر قرار دهید طوری که گویی دقیقه ها به سرعت در حال گذر هستند. قابلیت تنظیم ساعت و تاریخ را هم در نظر داشته باشید.

پیاده سازی : به صورت سخت افزاری

14) زنگ ساعت : (پروژه 2 نفره - اگر یک نفره انجام شود قابلیت گرفتن نمره تشویقی را دارد.)

یک ساعت دیجیتال که دقیقه و ساعت را در 4 سون سگمنت نمایش می دهد طراحی کنید. دقیقه ها میتوانند با کلاکی که سرعت نسبتا بالایی دارند، افزایش یابند (گویی زمان با سرعت در حال گذر است). حال این ساعت را میتوان برای زمان خاصی تنظیم کرد که در آن ساعت یک چراغ روشن شود. به این منظور باید یک زمان خاصی را انتخاب کرده و یک دکمه (دکمه ی Set) برای لحظه ای زده شود تا زمان در مدار ذخیره شود طوری که با تغییر زمان انتخابی، تا وقتی که دکمه ی Set زده نشده، زمان زنگ تغییر نکند. حال با رسیدن زمان

به زمان تنظیم شده، چراغ مورد نظر روشن خواهد شد. یک دکمه هم برای **reset** کردن زنگ قرار دهید تا با فشردن آن، دیگر در زمان **set** شده زنگ نزنند.

پیاده سازی : به صورت سخت افزاری