

پروژه درس آزمایشگاه مدارهای منطقی - پاییز ۱۴۰۳

طراحی سیستم جایابی خودرو در پارکینگ هوشمند

۱- مقدمه

در دنیای امروز، استفاده از فناوری‌های نوین در حوزه‌های مختلف زندگی روزمره، به ویژه در بخش حمل‌ونقل، به یکی از ارکان اصلی بهبود کیفیت و بهره‌وری تبدیل شده است. یکی از مهم‌ترین چالش‌ها در شهرهای بزرگ و مناطق پرجمعیت، مسئله پارکینگ خودروها و مدیریت بهینه فضاهای پارک است. سیستم‌های پارکینگ سنتی معمولاً دارای محدودیت‌های بسیاری در زمینه مدیریت فضا، زمان و هزینه هستند. به همین دلیل، استفاده از فناوری‌های هوشمند برای طراحی و پیاده‌سازی پارکینگ‌های مدرن که بتوانند به‌طور خودکار و با کمترین دخالت انسانی، خودروها را به مکان‌های مناسب هدایت کنند، به یکی از نیازهای اساسی شهرهای هوشمند تبدیل شده است.

پروژه طراحی سیستم جایابی خودرو در پارکینگ هوشمند با استفاده از FPGA به دنبال ارائه یک راهکار هوشمند برای بهینه‌سازی فرآیند پارک خودرو با استفاده از فناوری تراشه‌های قابل برنامه‌ریزی است. FPGA ها به دلیل قابلیت‌های پردازشی بالا و توانایی انجام پردازش‌های موازی، یک انتخاب مناسب برای پیاده‌سازی سیستم‌های سریع و کارآمد در محیط‌های پیچیده هستند. در این پروژه، هدف این است که سیستم دیجیتالی طراحی شود که به‌طور هوشمند خودروها را در پارکینگ‌های بزرگ و پیچیده هدایت کرده و با تشخیص مکان‌های خالی بر اساس اولویت، زمان پارک را به حداقل برساند. دانشجویان با انجام این تمرین با نحوه طراحی و پیاده‌سازی سیستم‌های دیجیتال مبتنی بر FPGA آشنا می‌شوند و از طراحی سطح پایین (گیت‌ها و فلیپ‌فلاپ‌ها) تا پیاده‌سازی‌های پیچیده‌تری مانند مدل‌سازی رفتاری الگوریتم‌های ترتیبی را تجربه می‌کنند.

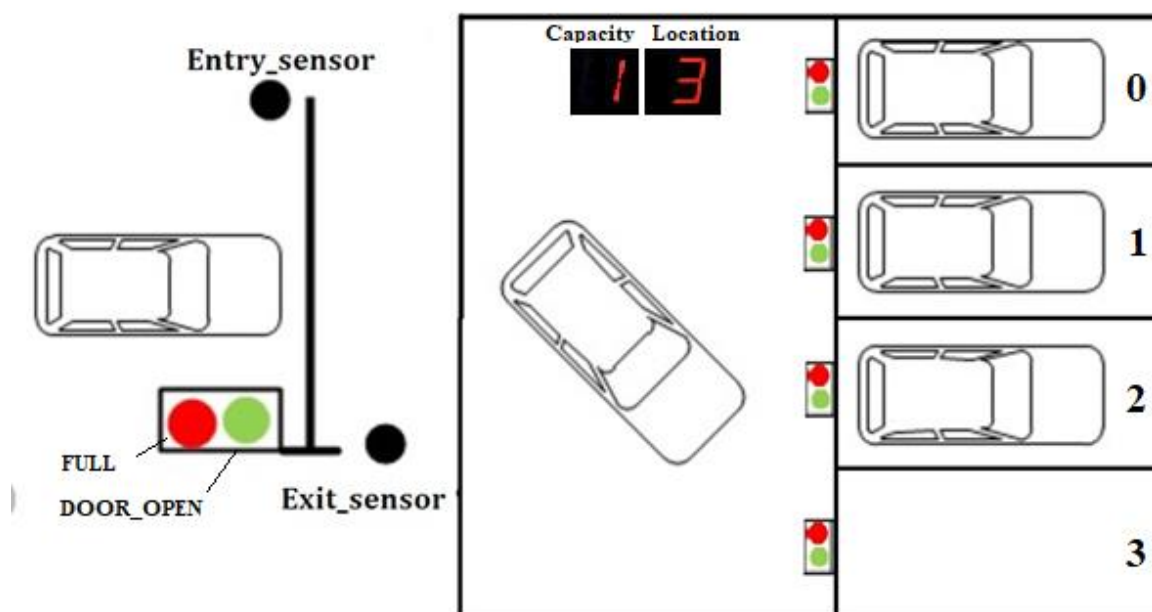


شکل ۱ - سیستم جایابی در پارکینگ هوشمند

۲- صورت مسئله

در این بخش سعی شده است با ساده سازی یک سیستم جایابی خودرو، پروژه ای قابل طراحی با مدارهای ترتیبی و قابل پیاده سازی در آزمایشگاه مدارهای منطقی برای دانشجویان طراحی شود.

فرض کنید در یک پارکینگ خودرو کوچک فقط ۴ محل برای پارک خودرو وجود دارد (شکل ۲). در این پارکینگ ساده ورود و خروج خودروها توسط دو سنسور **Entry_sensor** و **Exit_sensor** که در ورودی پارکینگ تعبیه شده اند قابل تشخیص است. در هر محل پارک یک چراغ وجود دارد که سبز بودن آن نشان دهنده جای پارک خالی و قرمز بودن نشان دهنده پر بودن (یا رزرو بودن) آن محل برای پارک است. همچنین این پارکینگ یک تابلو نمایشگر شامل ۴ رقم دیجیتال (7-segment) دارد که دو رقم سمت چپ آن تعداد مکانهای پارک خالی را نشان می دهد و دو رقم سمت راست هنگام ورود خودرو بهترین محل پارک را به راننده نشان می دهد. علاوه بر این امکانات دو چراغ سبز و قرمز در جلوی درب ورودی به ترتیب پر بودن پارکینگ (**FULL**) و باز شدن درب پارکینگ (**DOOR_OPEN**) را نشان می دهند.



شکل ۲- پارکینگ مسئله با ۴ محل پارک

می خواهیم یک سیستم دیجیتال با عملکرد زیر برای پارکینگ فوق طراحی کنیم:

- ۱) با ورود یک خودرو، در صورتی که جای پارک خالی وجود داشته باشد درب پارکینگ (یا راهبند) را باز کند. باز شدن درب پارکینگ با چشمک زدن چراغ **DOOR_OPEN** مدل می شود. چراغ **DOOR_OPEN** با فرکانس ۲ هرتز به مدت ۱۰ ثانیه روشن و خاموش شود. در صورتی که جای خالی وجود نداشته باشد نباید درب باز شود و چراغ **FULL** به نشانه پر بودن پارکینگ باید سه بار با فاصله زمانی یک ثانیه روشن و خاموش شود.

۲) با ورود خودرو بهترین جای خالی برای پارک روی نمایشگر نشان داده شود، اگر چند جای پارک خالی باشد اولویت با جای پارک با شماره کمتر است (فرض کنید خودرو در همان محل پیشنهادی پارک خواهد کرد) و اطلاعات تابلو نمایشگر و چراغ محل پارک پس از آن به روز رسانی شود.

۳) در هنگام خروج خودرو نیز باید محل خالی شدن شناسایی شده، درب پارکینگ باز شده (چراغ DOOR_OPEN با فرکانس ۲ هرتز به مدت ۱۰ ثانیه روشن و خاموش شود) و اطلاعات نمایشگر و محل پارک بروزرسانی شود.

برای راحتی در ابتدای کار به جای سنسورهای Entry_sensor و Exit_sensor دو کلید فشاری (Button) در نظر بگیرید (در بخش ۵ استفاده از یک سنسور التراسونیک برای تشخیص ورود و خروج پیشنهاد خواهد شد). همچنین فرض کنید در هنگام خروج خودرو محل پارک شده آن توسط یک ورودی دو بیتی (با Switch) به سیستم داده خواهد شد (در یک پارکینگ هوشمند واقعی باید در هر محل پارک یک سنسور تعبیه کرد یا اینکه می توان برای تشخیص اینکه کدام خودرو در حال خروج است و کدام محل خالی شده است، از سیستم تشخیص پلاک خودرو استفاده کرد).

۳- روش طراحی

- برای طراحی یک سیستم دیجیتال با مشخصات فوق موارد زیر را باید مدنظر داشته باشید:
- ابتدا ورودی ها و خروجی های مسئله و ارتباط بین آنها را دقیقاً مشخص کنید. سعی کنید سیستم را بصورت یک یا چند بلوک مرتبط با هم رسم کنید.
 - سیستم پردازشی مرکزی را بدون توجه به پیچیدگی های ورودی و خروجی (مثلاً سنسور ها و نمایشگر در این مسئله) طراحی کرده و با فرض های ساده ای برای ورودی و خروجی ها تست کنید.
 - سیستم های دیجیتال شامل دو بخش مجزای مسیر داده (شامل بلوک های انتقال، پردازش یا نگهداری داده) و بخش کنترل (مدیریت ماشین حالت و تولید کننده سیگنال های کنترلی بخش مسیر داده) هستند، می توانید هر بخش را جداگانه طراحی کنید و در نهایت به هم ارتباط دهید.
 - برای طراحی بخش کنترلی، تمام حالت های ممکن سیستم را در نظر بگیرید، دیاگرام حالت ماشین حالت متناهی (FSM) سیستم را رسم کرده و پس از ساده سازی مدار ترتیبی متناظر با آن را با استفاده از D فلیپ فلاپ طراحی و تست کنید. در این مرحله باید برای کدگذاری حالت های سیستم، نوع طراحی مثلاً ماشین Mealy یا Moore بودن سیستم تصمیم گیری کنید.

۴- نکات عملی پیاده سازی

رعایت موارد زیر برای پیاده سازی توصیه می شود:

- هر ماژول وریلاگ را که توصیف می کنید به صورت جداگانه با ورودی های مناسب تست کنید و از صحت عملکرد آن اطمینان حاصل کنید.
- خروجی نمایشگر روی برد آزمایشگاه شامل ۴ عدد 7-segment کاتد (Cathode) مشترک است و در مدار آن خطوط داده همه ارقام یکسان و کاتدها مجزا هستند. برای نمایش عدد چهار رقمی بصورت همزمان روی این نمایشگر، باید از روش مالتی پلکس ارقام (پیوست ۱) استفاده کنید، این قسمت را به صورت یک ماژول نوشته و تست کنید، سپس آن را به سیستم اصلی متصل نمایید.
- ورودی های Button و Switch نیاز به مدار لرزش گیر (Debouncing) دارند، در غیر اینصورت همراه ورودی ها، داده های تصادفی وارد سیستم خواهد شد. در پیوست ۲ یک مدار لرزش گیر معرفی شده است.
- برای کل سیستم و قسمت های مختلف، سیگنال بازنشانی (Reset) برای بازگشت به حالت اولیه تعبیه کنید.
- چون سیستم مد نظر نیاز به کلاک های متفاوت دارد (مثلاً برای زمانبندی تایمر پارکینگ، برای نمایشگر دیجیتال، برای آلارم باز شدن درب و ..) بهتر است یک ماژول تقسیم فرکانسی برای تولید کلاک های مورد نیاز طراحی کنید (فرکانس کلاک اصلی روی برد آزمایشگاه 40Mhz روی پایه P184 است).

۵- موارد پیشرفته تر (اختیاری و امتیازی)

۵-۱- استفاده از سنسور التراسونیک

می توان برای تشخیص ورود و خروج خودروها، از سنسورهای فراصوتی (Ultrasonic) استفاده کرد. این سنسورها با ارسال و سپس دریافت امواج فراصوت (فرکانس بالاتر از محدوده شنوایی انسان) می توانند فاصله تا اجسام مقابل را تشخیص دهند. با استفاده از این سنسورها می توان با تشخیص مسافت در دو زمان نزدیک به هم و مقایسه آنها با هم تشخیص داد که یک شیء (مثلاً خودرو) در حال نزدیک شدن به سنسور است یا دور شدن از آن. به این ترتیب در این پروژه به جای کلیدهای Entry_sensor و Exit_sensor می توان از سنسور التراسونیک برای تشخیص خودکار عبور و مرور خودروها استفاده نمود.



شکل ۳- سنسور التراسونیک مدل HC-SR04

سنسور التراسونیک مدل HC-SR04 (موجود در آزمایشگاه) یک سنسور رایج و پرکاربرد در پروژه‌های الکترونیکی و رباتیک برای اندازه‌گیری فاصله استفاده است. این سنسور با ولتاژ کاری ۵ ولت و جریان مصرفی حدود ۱۵ میلی‌آمپر قادر به اندازه‌گیری فاصله اشیاء بین ۲ سانتی‌متر تا ۴ متر با زاویه تشخیص ۱۵ درجه است. نحوه استفاده از سنسور HC-SR04 برای محاسبه فاصله شامل ۴ مرحله زیر است:

(۱) ارسال پالس: به پایه Trig (پایه تحریک) یک رشته پالس داده می‌شود، این پالس یک سیگنال اولتراسونیک را از فرستنده سنسور منتشر می‌کند.

(۲) بازتاب پالس: امواج اولتراسونیک از جسم یا مانع پیش رو بازتاب می‌شود و به گیرنده سنسور می‌رسد.

(۳) دریافت سیگنال: گیرنده سنسور امواج بازتابی را دریافت کرده و یک پالس دیجیتال به پایه Echo ارسال می‌کند.

(۴) محاسبه فاصله بر اساس مدت زمان رفت و برگشت سیگنال از فرمول زیر:

$$\text{فاصله} = \frac{\text{زمان رفت و برگشت}}{2} \times \text{سرعت صوت}$$

می‌توانید برای پیاده‌سازی این بخش از مازول‌های آماده در اینترنت برای HC-SR04 در پروژه خود استفاده کنید (به شرطی که دانشجو بتواند عملکرد قسمت‌های مختلف کد را توضیح دهد). یک نمونه از کدهای موجود در لینک زیر قابل دسترس است.

<https://github.com/suoglu/HC-SR04/blob/master/Sources/hc-sr04.v>

همانطور که بیان شده با استفاده از مازول فوق باید در چند بازه زمانی نزدیک هم فاصله خودرو را محاسبه کرده، سپس با مقایسه آنها ورود یا خروج خودرو را تشخیص دهید.

۵-۲ - نمایش مدت زمان توقف خودرو

قسمتی را به پروژه اضافه کنید که در هنگام خروج هر خودرو مدت زمان توقف آن خودرو در پارکینگ را به مدت ۱۵ ثانیه روی نمایشگر به صورت دو رقم سمت چپ ساعت و دو رقم سمت راست دقیقه نمایش دهد.

برای این کار می‌بایست برای هر محل پارک یک تایمر/شمارنده در نظر بگیرید (در کل ۴ شمارنده) که با ورود یک خودرو به یک محل پارک شمارنده آن فعال شده و در هر ثانیه یک عدد افزایش یابد و با خروج آن مقدار آن در نمایشگر نشان داده شده و سپس صفر گردد.

۶- ارزیابی و زمانبندی تحویل

انجام پروژه را می‌توان به سه مرحله (فاز) تقسیم کرد:

فاز اول: در این مرحله باید طرح پیشنهادی به صورت بلوک دیاگرام رسم شده و عملکرد بلوک‌های مختلف (ماژول‌ها) توضیح داده شود. ماشین حالت (FSM) بخش کنترل را رسم کرده و پس از ساده سازی، توصیف وریلاگ ماژول‌های مورد نیاز را بنویسید.

فاز دوم: برنامه ماژول‌های فاز اول را در محیط ISE نوشته، برای هر ماژول یک برنامه آزمون (Testbench) بنویسید و آنها را شبیه‌سازی کنید. در صورت عدم عملکرد صحیح، ماژول‌ها را اصلاح و دوباره تست کنید. برنامه آزمون نوشته شده باید قابلیت خواندن سناریوهای آزمون از فایل **txt** و نوشتن نتایج شبیه‌سازی در فایل را داشته باشد.

فاز سوم: ابتدا ماژول‌های ورودی/خروجی (سنسورها و نمایشگر دیجیتال) را تست کرده سپس ماژول‌های تست شده بخش قبلی را به ماژول‌های ورودی/خروجی متصل و کل پروژه را روی **FPGA** سنتز کنید. در صورتی که جواب صحیح نمی‌گیرید باید فازهای یک و دو را اصلاح کنید.

جدول ۱- زمان بندی انجام و تحویل پروژه

مرحله	تاریخ	فعالیت کلاس	موارد تحویلی	نمره از ۱۰۰
فاز اول: طراحی	هفته سوم آذر*	اصلاح طراحی FSM در صورت نیاز کد نویسی ماژول‌های اصلی	بلوک دیاگرام طراحی رسم دیاگرام حالت FSM (بارگذاری در کورسز)	۳۰
فاز دوم: شبیه سازی	هفته چهارم آذر*	شبیه سازی و اصلاح ماژول‌های اصلی تست ماژول‌های نمایشگر و لرزشگیر	شبیه‌سازی ماژول‌های اصلی (تحویل حضوری)	۲۰
فاز سوم: سنتز (تحویل نهایی)	هفته سوم دی*	سنتز مدار	سنتز پروژه (تحویل حضوری) پاسخ به سوالات (بارگذاری فایل ۷. در کورسز)	۵۰

+ زمان تحویل حضوری برای هر گروه آزمایشگاه در ساعات تشکیل کلاس آن گروه در طول هفته است.

انجام بخش اختیاری/امتیازی

با انجام موارد اختیاری بخش پنج و تحویل آن همراه با ستنز پروژه، ۲۰ نمره (هر قسمت ۱۰ نمره) به نمره پروژه اضافه خواهد شد.

نکات مهم

- پروژه ها به صورت تیم های حداکثر دو نفره انجام می شوند ولی به صورت تک نفره تحویل گرفته خواهند شد (هر دو نفر باید آمادگی کامل برای پاسخگویی به سوالات را داشته باشند).
- در صورت مشابه بودن پروژه با دیگر تیم ها یا کلاس های دیگر، نمره پروژه هر دو تیم صفر خواهد شد.
- در صورت عدم اشراف دانشجو به بخشی از پروژه تحویلی و عدم توانایی پاسخگویی به سوالات مدرس آزمایشگاه، نمره کل پروژه برای آن دانشجو صفر خواهد شد.
- گزارش فاز اول می تواند دست نویس یا تایپی باشد، گزارش در یک فایل pdf با نام **Ggr_names.pdf** که در آن **gr** شماره گروه آزمایشگاه و **names** نام افراد تیم است، در سامانه کورسز بارگذاری شود (مثال **G3_Alavi_Rezae.pdf**). به ازای هر روز تاخیر تا حداکثر ۵ روز، ۱۰ درصد نمره کسر خواهد شد.
- برنامه نهایی فاز سوم در یک فایل وریلاگ **v**. با همان اسم (**Ggr_names.v**) در سامانه کورسز بارگذاری شود (همه مازول ها، به جز برنامه های آزمون باید در یک فایل **v**. باشد). برای این فاز هیچ تاخیری در بارگذاری فایل قابل قبول نیست.
- در برنامه مازول ها باید خوانا باشد (اسامی شناسه ها با معنی انتخاب شود و برنامه توضیحات (comment) کافی داشته باشد) برنامه های دریافتی از همه گروه های آزمایشگاه با ابزارهای هوش مصنوعی مشابهت سنجی خواهند شد. در صورت مشابهت بالا در ساختار یا منطق برنامه با تیم های دیگر نمره پروژه صفر خواهد شد.

با آرزوی موفقیت

عباس نیک آبادی، آذر ماه ۱۴۰۳

پیوست ۱ - مالتی پلکس ارقام

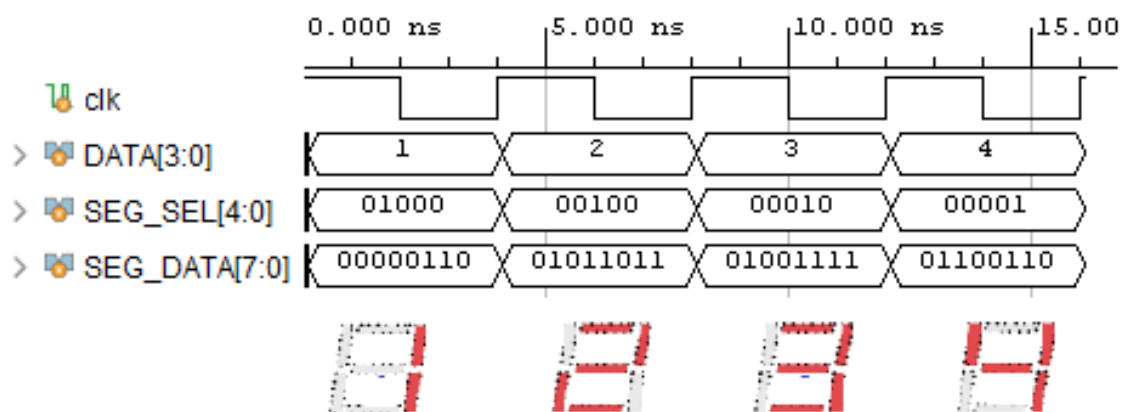
روش مالتی پلکس ارقام (Digit Multiplexing) یکی از تکنیک‌های رایج برای نمایش ارقام چند رقمی روی نمایشگرهای 7-segment است. در این روش، به جای روشن کردن همه سگمنت‌ها به طور همزمان، تنها یک رقم در هر لحظه روشن می‌شود و به سرعت بین ارقام مختلف سوئیچ می‌شود تا به چشم انسان این طور به نظر برسد که همه ارقام همزمان نمایش داده می‌شوند.

نمایشگر برد آزمایشگاه (AVA3S400) طبق شکل زیر از ۵ بخش (چهار رقم 7-segment و کاراکتر :) تشکیل شده است. خطوط داده در هر چهار 7-segment مشترک بوده و کاتدهای آنها به طور مجزا قابل مقداردهی است. با یک کردن کاتدهای ارقام می‌توانیم مشخص کنیم داده ارسالی در کدام رقم (7-segment) نمایش داده شود.



شکل ۴- نمایشگر ۴ رقمی برد AVA3S400

برای نمایش یک عدد ۴ رقمی با روش مالتی پلکس ارقام، هر کدام از ۴ رقم بصورت جداگانه با فاصله زمانی کوتاه نمایش داده می‌شوند (در هر لحظه فقط یکی از ارقام روشن است). با سوئیچینگ سریع بین ارقام با سرعت حدود ۶۰ هرتز تا ۵۰۰ هرتز (یعنی هر ۲ میلی ثانیه تا ۱۶ میلی ثانیه یک رقم تغییر می‌کند)، چشم انسان این تغییرات را به صورت یک نمایش ثابت مشاهده می‌کند. این روند برای تمام ۴ رقم به طور مکرر تکرار می‌شود (ابتدا رقم اول روشن می‌شود و کاتدهای رقم اول فعال می‌شود، سپس رقم دوم، سوم و چهارم به طور پیوسته نمایش داده می‌شوند. بعد از نمایش چهارم، دوباره به اولین رقم برمی‌گردد و این روند ادامه می‌یابد). شکل زیر زمانبندی یک دور تکرار برای نمایش عدد 1234 را نشان می‌دهد. در این شکل DATA داده اصلی، SEG_SEL سیگنال انتخاب کاتدهای 7-segment ها برای نمایش یک رقم و SEG_DATA داده‌های لازم برای روشن شدن LED های 7-segment برای نمایش رقم مورد نظر است.

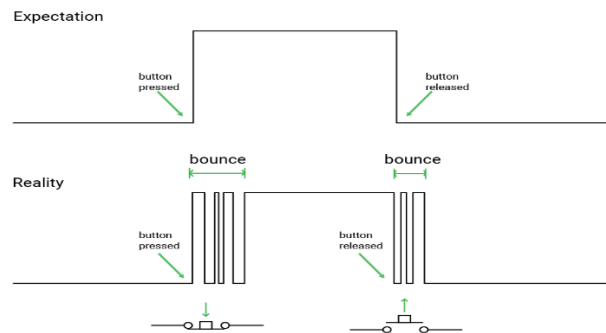


شکل ۵- زمانبندی یک دور برای نمایش عدد 1234

بنابراین برای نمایش اعداد روی نمایشگر ۴ رقمی، باید مازولی بنویسید که بصورت تکراری با فرکانس ۶۰ تا ۵۰۰ هرتز ارقام آن عدد را توسط مازول مبدل BCD به 7-segment (آزمایش ۹) به پایه‌های نمایشگر ارسال کند.

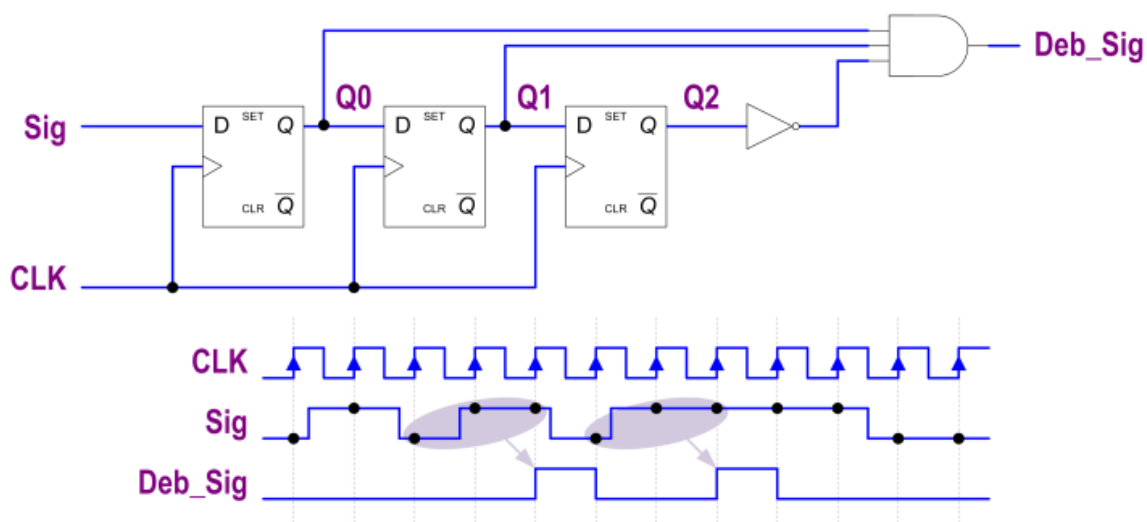
پیوست ۲ - رفع لرزش کلید

لرزش کلید (Switch Bounce) یک پدیده فیزیکی است که در هنگام فشردن یا رها کردن یک کلید مکانیکی (مانند دکمه یا سوئیچ) اتفاق می افتد. هنگامی که کلید فشرده می شود، تماس الکتریکی بین دو پین ممکن است چندین بار به صورت بسیار سریع قطع و وصل شود.



شکل ۶- اثر لرزش کلید

لرزش کلید باعث می شود که سیگنال های دیجیتال که باید تنها یک تغییر واحد (یک "۰" یا "۱") را نشان دهند، چندین بار تغییر کنند. این تغییرات ناپایدار می تواند باعث ایجاد نویز در مدار شود و ممکن است باعث اشتباهات منطقی به خصوص در در مدارهای شمارش گر (که در آنها برای هر فشار کلید باید تنها یک تغییر رخ دهد) شود. برای مقابله با اثر لرزش کلید، از روش های مختلفی می توان استفاده کرد که از جمله این روشها می توان (۱) استفاده از مدارهای RC (در کلید ورودی، ۲) استفاده از مدارهای مقایسه کننده Schmitt Trigger، (۳) تاخیر و توقف چند میلی ثانیه ای پس از ورودی و (۴) استفاده از مدارهای منطقی را نام برد. مدارهای منطقی لرزش گیر کلید معمولاً از یک فیلتر یا تاخیر دهنده استفاده می کنند تا از ثبت لرزش های سریع و موقتی جلوگیری و تنها تغییرات پایدار را شناسایی کند. شکل زیر یک نمونه از این مدارها را نشان می دهد که از ۳ فلیپ فلاپ D و یک گیت AND سه ورودی ساخته شده است.



شکل ۷- یک نمونه مدار debouncer سخت افزار