

به نام آن که جان را فکرت آموخت



دانشگاه صنعتی شریف

دانشکده‌ی مهندسی کامپیوتر

دستور کار

# آزمایشگاه طراحی سیستم‌های دیجیتال

تهیه و تدوین:

دکتر علیرضا اجالالی، مهندس امیرعلی حبیبی

بهار ۱۳۸۹

## سپاسگزاری

در اینجا جای دارد از آقایان دکتر شاهین حسابی، دکتر مازیار گودرزی و دکتر امیرمسعود قره‌باغی برای تهیه و تدوین نسخه‌های پیشین دستور کار آزمایشگاه تشکر و قدردانی کنیم. تقریباً نیمی از آزمایش‌هایی که در نسخه فعلی دستور کار وجود دارند از نسخه‌های پیشین اقتباس شده و با اعمال تغییراتی اندک به روز رسانی شده‌اند. همچنین جای دارد که از آقای دکتر امیرحسین جهانگیر که از نظرات ایشان در تهیه و تدوین نسخه فعلی دستور کار استفاده بسیار نمودیم تشکر و قدردانی کنیم. در پایان از کلیه اعضای محترم گروه معماری کامپیوتر دانشکده مهندسی کامپیوتر که با ارائه نظرات و راهنمایی‌های خود ما را در تهیه این مجموعه یاری نمودند تشکر می‌کنیم.

## فهرست مندرجات

عنوان	صفحه
<b>۱ معرفی</b>	<b>۳</b>
۱-۱ هدف	۳
۲-۱ پیش‌نیازهای نظری و عملی	۳
۳-۱ تجهیزات و نرم‌افزارهای لازم	۳
۱-۴ منابع علمی	۴
۵-۱ مقررات آزمایشگاه	۴
<b>۲ آزمایشها</b>	<b>۶</b>
۱-۲ مقدمه	۶
۲-۲ آزمایش اول: طراحی مدارهای ترکیبی با استفاده از امکانات شماتیک	۷
۲-۳ آزمایش دوم: طراحی مدارهای ترتیبی با استفاده از امکانات شماتیک	۸
۲-۴ آزمایش سوم: توصیف جریان داده	۹
۲-۵ آزمایش چهارم: توصیف رفتاری	۱۰
۲-۶ آزمایش پنجم: طراحی ضرب کننده	۱۱
۲-۷ آزمایش ششم: طراحی یک انکوباتور	۱۲
۲-۸ آزمایش هفتم: UART	۱۴
۲-۹ آزمایش هشتم: ALU اعداد مختلط	۱۵
۲-۱۰ آزمایش نهم: پیاده سازی حافظه های شرکت پذیر نوع سه گانه	۱۶
۲-۱۱ آزمایش دهم: پیاده سازی یک پردازنده ساده	۱۷

# ۱ معرفی

## ۱-۱ هدف

- هدف از این آزمایشگاه آن است که دانشجویان تجربیات عملی در دو زمینه زیر به دست آورند.
- طراحی سیستم‌های دیجیتال با استفاده از ابزارهای خودکار طراحی دیجیتال (CAD Tools)
  - پیاده‌سازی سیستم‌های دیجیتال با عناصر برنامه‌پذیر همچون CPLD و FPGA

## ۲-۱ پیش‌نیازهای نظری و عملی

دانشجویانی که در این آزمایشگاه ثبت نام نموده‌اند لازم است که مطالب دروس مدارهای منطقی و معماری کامپیوتر را به خوبی بدانند و آزمایشگاه مدارهای منطقی را گذرانده باشند. همچنین این آزمایشگاه به صورت هم نیاز با درس طراحی سیستم‌های دیجیتال ارائه می‌گردد و دانشجویان باید در طی نیم‌سال تحصیلی مطالب درس طراحی سیستم‌های دیجیتال را به خوبی مطالعه نموده و بیاموزند تا بتوانند از مطالب آموخته شده در آزمایشگاه استفاده کنند.

## ۳-۱ تجهیزات و نرم‌افزارهای لازم

برای انجام آزمایش‌ها در این آزمایشگاه دانشجویان لازم است که کار با نرم افزار مخصوص تراشه‌های برنامه‌پذیر شرکت ALTERA موسوم به Quartus II و همچنین شبیه‌ساز Modelsim را به خوبی بیاموزند. همچنین آموختن ابزار سنتز Leonardo می‌تواند برای دانشجویان در این آزمایشگاه بسیار مفید باشد. در آزمایشات این آزمایشگاه دانشجویان باید طرح‌های خود را به روی بردهای FPGA که در آزمایشگاه در اختیار ایشان قرار داده می‌شود پیاده‌سازی کنند و لذا لازم است که دانشجویان مستندات مربوط به این بردهای آزمایشگاهی که به ایشان داده می‌شود را به دقت مطالعه نموده و فراگیرند. به همراه این گزارش منابع زیر در اختیار دانشجویان قرار می‌گیرد که مطالعه آن‌ها برای موفقیت در انجام آزمایشات ضروری است:

الف- راهنمای برد FPGA مورد استفاده در آزمایشگاه طراحی سیستم‌های دیجیتال (ارائه شده توسط شرکت سازنده بردهای FPGA) مأخذ: شرکت سازنده برد FPGA

ب- یک راهنمای سریع برای کار با نرم افزار Quartus II مأخذ: شرکت ALTERA

پ- یک راهنمای سریع برای کار با نرم افزار شبیه‌ساز Modelsim

## ۴-۱ منابع علمی

برخی از منابع که مطالعه آن‌ها برای دانشجویان این آزمایشگاه مفید بوده و توصیه می‌شوند عبارتند از:

- 1) S. Palnitkar, *Verilog® HDL: A Guide to Digital Design and Synthesis*, Second Edition, Prentice Hall, 2003.
- 2) ACEX 1K Programmable Logic Family Data Sheet, [www.altera.com/literature/ds/acex.pdf](http://www.altera.com/literature/ds/acex.pdf)
- 3) ModelSim User's Manual, [www.actel.com/documents/modelsim\\_ug.pdf](http://www.actel.com/documents/modelsim_ug.pdf)
- 4) Introduction to the Quartus II Software, [www.altera.com/literature/manual/intro\\_to\\_quartus2.pdf](http://www.altera.com/literature/manual/intro_to_quartus2.pdf)

## ۵-۱ مقررات آزمایشگاه

- با توجه به اینکه آزمایشگاه طراحی سیستم‌های دیجیتال هم‌نیاز درس طراحی سیستم‌های دیجیتال است و لازم است که برای برخی از آزمایشات این آزمایشگاه دانشجویان دانش مربوطه را در درس طراحی سیستم‌های دیجیتال فرا گرفته باشند. لذا این آزمایشگاه در هر ترم حدود دو هفته (چهار جلسه) از سایر آزمایشگاه‌های دانشکده دیرتر آغاز به کار می‌کند.
- حضور دانشجویان در کلیه جلسات آزمایشگاه الزامی است و به ازای هر جلسه غیبت ۵ نمره از نمره نهایی دانشجو کسر خواهد شد.
- با توجه به محدود بودن تعداد ساعات در هر جلسه آزمایشگاه، امکان این که دانشجویان طراحی‌های خود را در جلسات آزمایشگاه انجام دهند وجود ندارد و لذا هر دانشجو باید طراحی مربوط به هر آزمایش را پیش از شروع آن جلسه انجام داده باشد و در ابتدای جلسه طرح خود را در قالب یک پیش‌گزارش به مدرس آزمایشگاه ارائه دهد. همچنین در ابتدای هر جلسه لازم است که دانشجویان گزارش آزمایشات انجام شده در جلسه قبل را به مدرس آزمایشگاه ارائه دهند.
- ارزشیابی هر دانشجو در این آزمایشگاه براساس فعالیتی که در طول نیم‌سال تحصیلی و در هر جلسه آزمایش داشته است مشخص می‌گردد. هیچ آزمون یا پروژه‌ای وجود ندارد که نمره غالب را داشته باشد و دانشجویان بتوانند عدم مشارکت خود در جلسات آزمایشگاه را با آن جبران کنند.

- ارزیابی هر دانشجو توسط مدرس آزمایشگاه و در طی نیم‌سال تحصیلی انجام می‌شود و در پایان نیم‌سال نمرات نهایی دانشجویان پس از مشورت مدرس آزمایشگاه و مسئول آزمایشگاه تعیین می‌گردد.
- دانشجویان موظف هستند که نهایت تلاش خود را در استفاده صحیح از تجهیزات و کامپیوترهای موجود در آزمایشگاه بکار گیرند. هر نوع آسیب که به تجهیزات و کامپیوترهای آزمایشگاه وارد شود و از روی بی‌دقتی، عدم مطالعه مستندات مربوطه، عدم رعایت راهنمایی‌های ارائه شده توسط مدرس آزمایشگاه و ... باشد باعث دریافت نمره مردودی در آزمایشگاه خواهد شد و همچنین دانشجو موظف خواهد بود که مطابق زمانبندی که مسئول آزمایشگاه مشخص می‌کند خسارت وارده را جبران کند. تشخیص اینکه خرابی یکی از تجهیزات یا کامپیوترهای آزمایشگاه آیا ناشی از قصور یک دانشجو بوده یا خیر فقط بر عهده مدرس آزمایشگاه است.
- برای آن دسته از آزمایش‌ها که توصیف Verilog از دانشجو خواسته شده است صرف اینکه توصیف مربوطه در ابزار ModelSim صحیح عمل می‌کند و توسط ابزارهای سنتز مربوطه همچون Quartus II سنتز شود کافی نیست و باید حتماً بر روی برد FPGA ریخته شده و توسط مدرس آزمایشگاه صحت آن بر روی برد تأیید گردد.
- دانشجویان ملزم به رعایت اخلاق و آداب مهندسی و دانشجویی هستند و از انجام کارهای خلاف همچون کپی کردن طرح دیگران، ارائه نتایج ساختگی برای آزمایش‌ها، استخدام دیگران برای انجام بخشی از کارهای مربوط به آزمایش‌ها و ... اکیداً پرهیز کنند. در صورت مشاهده این موارد ضمن دریافت نمره مردودی، برابر با مقررات دانشگاه و دانشکده برخورد خواهد شد.

## ۲ آزمایش‌ها

### ۱-۲ مقدمه

در این آزمایشگاه دانشجویان موظف هستند که در طی یک نیم‌سال تحصیلی ۱۰ آزمایش را انجام دهند. با توجه به اینکه این آزمایشگاه به صورت هم‌نیاز با درس طراحی سیستم‌های دیجیتال ارائه می‌شود لازم است که برای برخی از آزمایش‌ها دانشجویان دانش مربوطه را در درس طراحی سیستم‌های دیجیتال از زبان Verilog فرا گرفته باشند. به همین دلیل دو آزمایش اول به نحوی در نظر گرفته شده‌اند که نیاز به دانش زبان Verilog ندارد و هدف اصلی آن‌ها آشنایی دانشجویان با FPGA و ابزارهای CAD مربوطه است. آزمایش‌های ۳ و ۴ اولین آزمایش‌هایی هستند که دانشجو برای انجام آن‌ها نیاز به دانش زبان Verilog دارد و برای این دو آزمایش لازم است که توصیف قابل سنتز از طرح مورد نظر ارائه گردد. لازم به تأکید است که توصیف‌های Verilog مربوط به این آزمایش‌ها باید پیش از جلسه آزمایشگاه توسط دانشجویان تهیه شده و درستی آن با ابزارهای CAD همچون ModelSim و Quartus II ارزیابی شده باشد و در ابتدای جلسه در قالب پیش‌گزارش تحویل مدرس آزمایشگاه گردد. در آزمایش ۵ دانشجویان باید یک واحد ضرب‌کننده را طراحی و پیاده‌سازی کنند و برای انجام این آزمایش نیاز است که دانشجویان دانش مربوط به طراحی مدارهای حسابی را در حدی که در درس معماری کامپیوتر ارائه می‌شود داشته باشند. در آزمایش ۶ هدف آن است که دانشجویان تجربه‌ای در مورد طراحی یک سیستم دیجیتال داشته باشند که در درون یک سیستم غیر دیجیتال (دستگاه انکوباتور) قرار دارد و مجبور به تعامل با پیرامون غیر دیجیتال است. در آزمایش ۷ دانشجویان باید سخت‌افزار لازم برای برقراری ارتباط سریال آسنکرون را طراحی کنند دانش مقدماتی مربوط به انجام این کار معمولاً در دروس معماری کامپیوتر و انتقال داده‌ها به دانشجویان آموخته شده است. مهمترین دانش مورد نیاز برای انجام آزمایش ۸ آشنایی با خط لوله است که معمولاً دانشجویان در درس معماری کامپیوتر با آن آشنا شده‌اند. در آزمایش ۹ دانشجویان باید یک حافظه TCAM را پیاده‌سازی کنند که در برخی از زمینه‌ها مانند بانک‌های داده، سیستم‌های هوشمند و .. موارد کاربرد مهمی دارند. سر انجام آخرین آزمایش (آزمایش ۱۰) به پیاده‌سازی یک پردازنده ساده می‌پردازد.

در تمامی آزمایش‌های فوق به ویژه آزمایش‌های ۳ الی ۱۰ فرض بر آن است که کدهای Verilog مربوطه پیش از جلسه آزمایشگاه توسط دانشجویان تهیه شده است و در جلسه آزمایشگاه تلاش اصلی دانشجویان باید بر پیاده‌سازی طرح بر روی برد FPGA متمرکز باشد.

## ۲-۲ آزمایش اول: طراحی مدارهای ترکیبی با استفاده از امکانات شماتیک

شرکت‌های تولید کننده تراشه‌های FPGA و CPLD همواره علاوه بر تولید تراشه‌های FPGA و CPLD نرم افزارهایی ارائه می‌کنند که این نرم افزارها به عنوان ابزارطراحی به کمک کامپیوتر (CAD) به کاربران امکان می‌دهند که از محصولات تولید شده توسط آن شرکت‌ها استفاده کنند. این ابزارهای CAD نه تنها از زبان‌های استاندارد توصیف سخت افزار همچون Verilog و VHDL استفاده می‌کنند بلکه معمولاً این امکان را می‌دهند که برای انجام سریع طراحی با هدف پیاده‌سازی مدارهای کوچک بدون آنکه نیاز به استفاده از زبان‌های Verilog یا VHDL باشد بتوان طرح مدار را برای ابزار CAD تعریف نمود که از این امکان معمولاً با نام امکان طراحی شماتیک یاد می‌شود. هدف از این آزمایش این است که دانشجویان از امکانات شماتیک استفاده نموده و یک مدار ترکیبی را طراحی و پیاده‌سازی کنند.

### شرح آزمایش ۱

مدار ترکیبی طراحی کنید که یک خروجی یک بیتی داشته باشد و خروجی آن وقتی ۱ شود که ورودی BCD چهاررقمی آن مضربی از ۳ باشد. همین آزمایش را برای اعداد مضرب ۱۱ نیز تکرار کنید. در طراحی این مدار فقط مجاز به استفاده از گیت‌های پایه هستید.



## ۳-۲ آزمایش دوم: طراحی مدارهای ترتیبی با استفاده از امکانات شماتیک

همانند آزمایش قبل می‌خواهیم از امکانات شماتیک استفاده کنیم ولی این بار هدف طراحی یک مدار ترتیبی است.

### شرح آزمایش

اتاق انتظار یک اداره ظرفیت ۱۵ نفر را دارد و دارای یک در ورودی و یک در خروجی است. در کنار هر یک از این درها، یک حسگر نصب شده است که با عبور فرد، سیگنالی به مدت یک Clock تولید می‌نماید (برای ورودی سیگنال IN و برای خروج سیگنال OUT). فردی که قصد ورود به اتاق را دارد، برای باز شدن در ورودی باید تکمه‌ی Ent را فشار دهد. در این صورت، این در به شرطی باز می‌شود که در زمان فشردن تکمه تعداد حاضرین در اتاق کمتر از ۱۵ نفر باشد.

ساعت مجاز ورود به اتاق سپری نشده باشد (فرض کنید که یک ساعت در اختیار داریم که خروجی آن T در زمان مجاز ۱ بوده و هرگاه زمان مجاز سپری شده باشد، برابر ۰ باشد).

برای باز بودن این در، باید سیگنال Open را به مدت لازم (تا عبور فرد از در) ۱ نگه داشت. در خروجی همواره باز است، مگر وقتی که تعداد حاضرین در اتاق به ۰ برسد، در این صورت باید سیگنال خروجی Close برابر ۱ شود تا این در بسته شود. در طرح مدار می‌توانید از یک Up/Down Counter با مشخصات زیر استفاده کنید. نمودار بلوکی شمارنده را در طرح خود بگنجانید و ارتباط ورودی‌ها و خروجی‌های آن را با سایر قسمت‌های مدار مشخص کنید. دقت کنید که در یک Clock ممکن است فردی از در ورودی وارد شده و همزمان فردی از در خروجی خارج شود. فرض کنید حرکت در خلاف جهت تعیین شده‌ی درها امکان‌پذیر نباشد.

U	Clk	Clr	Enable	Function
X	X	0	X	Reset counter to 0
X	X	1	0	Hold previous number
1	↑	1	1	Up count
0	↑	1	1	Down count

فرکانس کاری مدار را محاسبه کنید.

## ۲-۴ آزمایش سوم: توصیف جریان داده

۱) با استفاده از دستور assign (توصیف جریان داده) یک Cascadable 1-bit comparator بسازید. با اتصال چهار عدد از این مقایسه‌کننده‌ها به یکدیگر یک مقایسه‌کننده‌ی چهار بیتی بسازید. در این آزمایش بایستی طراحی سلسله مراتبی انجام دهید و مدار به دست آمده باید مدار ترکیبی شود.

۲) با استفاده از دستور assign (توصیف جریان داده) یک مقایسه‌کننده سریال بسازید. این مقایسه‌کننده یک مدار ترتیبی است که با استفاده از ورودی reset در اول کار reset می‌شود و پس از آن از دو ورودی خود بیت‌های دو عددی که باید مقایسه شوند را بیت به بیت دریافت نموده و در هر پالس ساعت حاصل مقایسه را تا جایی که مقایسه کرده (تا بیتی که مقایسه انجام شده) در خروجی سریال خود تحویل می‌دهد. برای این مدار طراحی سلسله مراتبی انجام ندهید و فقط باید یک پیمانه (module) داشته باشید که آن پیمانه نیز فقط باید توصیف جریان داده شده باشد و استفاده از هیچ نوع توصیف دیگری مجاز نیست.

## ۵-۲ آزمایش چهارم: توصیف رفتاری

یک پشته با عمق ۸ و پهنای ۴ بیت طراحی کنید که دارای ورودی‌ها و خروجی‌های زیر باشد :

<b>Inputs:</b>	Clk	Clock signal
	RstN	Reset signal
	Data_In	4-bit data into the stack
	Push	Push Command
	Pop	Pop Command
<b>Outputs:</b>	Data_Out	4-bit output data from stack
	Full	Full=1 indicates that the stack is full
	Empty	Empty=0 indicates that the stack is empty

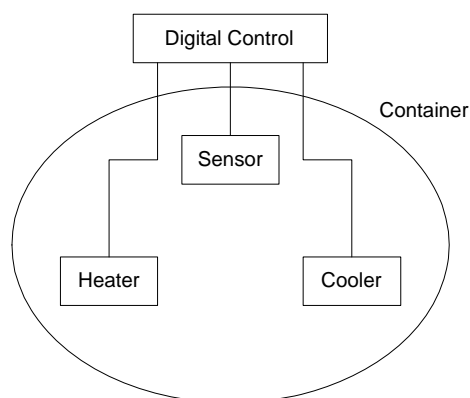
## ۶-۲ آزمایش پنجم: طراحی ضرب کننده

هدف از این آزمایش طراحی یک واحد ضرب کننده است که برای انجام عمل ضرب از روش ضرب بوث (Booth) استفاده می‌کند. برای انجام این آزمایش مسیره داده و واحد کنترل را جداگانه طراحی کنید و سپس با اتصال آن‌ها به یکدیگر ضرب کننده را ایجاد کنید.

توجه کنید که در این طراحی هنگامی که مضروب فیه (Multiplier) را به سمت راست شیفت می‌دهید لازم است که واحد شیفت دهنده توان انجام شیفت بیش از یک بیت در یک پالس ساعت را داشته باشد تا بتواند نسبت به ضرب عادی (Shift & ADD) تسریع داشته باشد. لذا حتماً در طراحی خود از چنین واحد شیفت دهنده‌ای استفاده کنید.

## ۷-۲ آزمایش ششم: طراحی یک انکوباتور

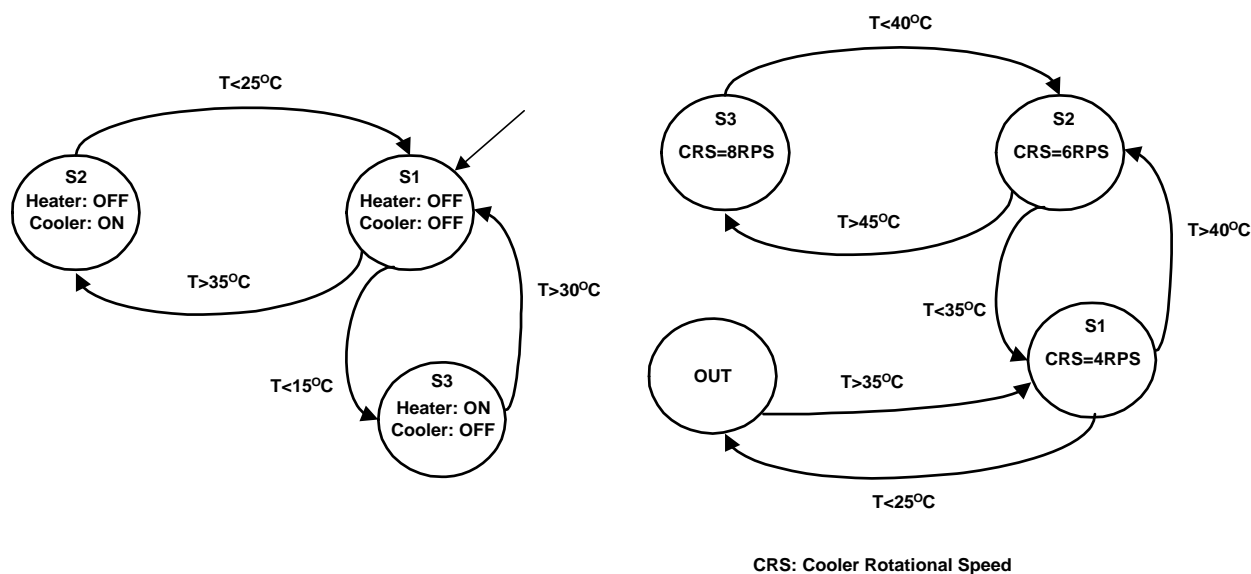
در این آزمایش هدف آن است که واحد کنترل دیجیتال یک سیستم انکوباتور (Incubator) را طراحی و پیاده‌سازی کنید. در این سیستم مطابق شکل زیر یک حسگر دما، یک واحد خنک کننده (Cooler) مجهز به پنکه (Fan) و یک واحد گرم کننده (Heater) وجود دارد.



در این سیستم یک حسگر دما وجود دارد که دمای محفظه را که میان 10- تا 60+ درجه سانتیگراد متغیر است می‌خواند و در قالب یک عدد ۸ بیتی به سیستم شما تحویل می‌دهد. دما هر دقیقه یک بار از حسگر دریافت می‌شود و براساس آن واحد کنترل دیجیتال تصمیم می‌گیرد که:

- ۱- چگونه واحدهای گرم کننده و سرد کنند را فعال و غیر فعال کند.
- ۲- چگونه در صورت فعال بودن واحد سرد کننده دور Fan آن را تنظیم کند.

روش کار واحد کنترل دیجیتال توسط دو نمودار حالت زیر توصیف شده است:



در این نمودار حالت برچسب‌های مربوط به یال‌ها نشان دهنده تغییرات دمایی هستند که باعث تغییر حالت در واحد کنترل دیجیتال شده و باعث واکنش سیستم به شکل روشن و خاموش شدن Heater و Cooler و یا تغییر دور Fan مربوط به Cooler می‌شوند.

دقت کنید که در نمودارهای حالت فوق نمودار حالت سمت راست زمانی فعال می‌شود (از حالت out خود خارج می‌شود) که نمودار حالت سمت چپ در حالت S2 که در آن Cooler روشن است قرار گرفته باشد و اگر نمودار حالت سمت چپ در حالت S2 خود نباشد نمودار حالت سمت راست غیر فعال می‌شود (یعنی وارد حالت out خود می‌شود) چون معنی ندارد که وقتی Cooler خاموش است دور آن تنظیم گردد.

در این آزمایش هدف طراحی واحد کنترل دیجیتال است و قسمت‌های حسگر، Heater و Cooler وجود خارجی ندارند. لذا در آزمایشگاه باید دانشجویان عددی را که مثلاً توسط حسگر دما خوانده می‌شود خود با استفاده از امکانات برد FPGA به مدار اعمال کنند و واکنش سیستم (و همچنین دور Fan) را به جای آنکه Heater و Cooler واقعی روشن و خاموش شوند با روشن و خاموش شدن LED و یا واحدهای Seven Segment نمایش دهند.

## ۸-۲ آزمایش هفتم: UART

هدف از انجام این آزمایش طراحی یک Universal Asynchronous Receiver Transmitter (UART) می باشد. در قسمت ارسال کننده این دستگاه هر بار یک کد ۷ بیتی ASCII بصورت سریال ارسال می گردد. در ابتدا یک بیت شروع (Start)، سپس یک بیت توازن (Parity) و بعد ۷ بیت داده ارسال می شوند. در انتها نیز حداقل یک بیت خاتمه (Stop) ارسال می شود (در مجموع ۱۰ بیت). در قسمت گیرنده نیز پس از دریافت بیت شروع (Start)، ۸ بیت مربوط به داده و توازن (Parity) بصورت سریال دریافت شده و در یک ثبات (Register) ۸ بیتی ذخیره می شود. با اتصال دو واحد از این واحدهای UART به یکدیگر نشان دهید که طراحی شما مبادله داده را به درستی انجام می دهد.

**۹-۲ آزمایش هشتم: ALU اعداد مختلط**

این آزمایش از چند بخش تشکیل شده است. در این آزمایش بایستی پیمانه‌های مختلف را پیاده‌سازی و تست کنید. سپس با کنار هم قرار دادن این پیمانه‌ها یک کامپیوتر پایه را پیاده‌سازی کنید. این ماشین فرضی دارای حافظه‌ی ۳۲ کلمه‌ای است.

الف- پیمانه جمع و تفریق اعداد مختلط

ب- ماژول ضرب اعداد مختلط

ج- یک واحد پایپ‌لاین که دستورات را از حافظه مذکور بخواند؛ سپس اجرای عملیات مختلط ورودی را به صورت پایپ‌لاین انجام دهد.

در این آزمایش، نبایستی از واحدهای جمع کننده یا ضرب کننده بیش از یک بار استفاده شود.



## ۲-۱۰ آزمایش نهم: پیاده‌سازی حافظه‌های شرکت پذیر نوع سه گانه

حافظه‌های شرکت پذیر (Content Addressable Memory) نوع سه گانه (Ternary) موسوم به TCAM در بسیاری از کاربردها از جمله فشرده سازی، بانک داده‌ها، سیستم‌های هوشمند و ... به کار گرفته می‌شوند. تفاوت اصلی TCAM با CAM عادی در این است که در TCAM علاوه بر ذخیره مقادیر 0، 1 می‌توان مقدار X را نیز ذخیره نمود که به این معنی است که مقایسه فقط برای محل‌هایی صورت گیرد که 0 یا 1 هستند و در محل X مقایسه‌ای صورت نگیرد و بیت‌ها برابر فرض شوند. به عنوان مثال داده 01101110 اگر در TCAM مورد جستجو قرار گیرد با هر کدام از داده‌های زیر به عنوان انطباق (Match) در نظر گرفته می‌شود: 0110XXXX، 0X1X11X0، X1101XXX

یک پیمانه TCAM به اندازه ۱۶ ثابت ۱۶ بیتی را طراحی و پیاده‌سازی کنید.

## ۲-۱۱ آزمایش دهم: پیاده‌سازی یک پردازنده ساده

یک پردازنده با معماری پشته‌ای دارای یک پشته با هشت ثابت ۸ بیتی است. این پردازنده دارای ۸ دستور در مجموعه دستورالعمل‌های خود است و دارای حافظه به اندازه 256 خانه ۸ بیتی است که ۸ خانه آخر آن (یعنی آدرس‌های F8 الی FF) برای I/O به صورت Memory Mapped I/O مورد استفاده هستند. دستورات این کامپیوتر به همراه OP CODE های خود عبارتند از:

0000      PUSHC      C

این دستور مقدار ثابت (Constant) ۸ بیتی C را در پشته PUSH می‌کند.

0001      PUSH      M

این دستور مقدار خانه حافظه (یا درگاه) که با آدرس M (آدرس ۸ بیتی) مشخص شده است را خوانده و در پشته PUSH می‌کند.

0010      POP      M

مقدار را از پشته POP کرده و آن را در خانه حافظه با آدرس M قرار می‌دهد (یا به درگاه با آدرس M ارسال می‌کند).

0011      JUMP

از پشته POP کرده و در PC قرار می‌دهد.

0100      JZ

اگر پرچم Z برابر 1 باشد از پشته POP کرده و در PC قرار می‌دهد.

0101      JS

اگر پرچم S برابر 1 باشد از پشته POP کرده و در PC قرار می‌دهد.

0110      ADD

دو داده بالای پشته را POP کرده با هم جمع کرده و حاصل را در بالای پشته PUSH می‌کند.

0111      SUB

دو داده بالای پشته را POP کرده عمل تفریق را بر روی آن‌ها انجام می‌دهد و حاصل را در بالای پشته PUSH می‌کند.

در این پردازنده همانطور که از توضیحات فوق مشخص است فقط دو پرچم S و Z موجود است. این دو پرچم فقط با اجرای دو دستور ADD و SUB تغییر مقدار می‌دهند و در اجرای سایر دستورات

مقدار آن‌ها تغییری نمی‌کند. همچنین محاسبات تماماً علامت دار و با استفاده از مکمل ۲ انجام می‌گیرند.

با استفاده از این پردازنده یک برنامه زبان ماشین بنویسید که یک عدد ۸ بیتی مثبت (مثلاً با نام  $X$ ) را با استفاده از امکانات برد FPGA از ورودی دریافت نموده و سپس پس از محاسبه مقدار:

$$Y=((X+23)*2)-12;$$

مقدار خروجی  $Y$  را با استفاده از واحدهای Seven Segment موجود بر روی برد نمایش دهد. در صورتی که ورودی عدد منفی باشد و یا مقدار خروجی از حوزه قابل نمایش خارج شود (بزرگتر از ۱۲۷ باشد) یکی از LED های روی برد به علامت خطا روشن شود.

در انجام این آزمایش باید دانشجویان ابتدا کد خود را (که باید پیش از جلسه آزمایش تهیه کرده باشند) به همراه نتایج شبیه‌سازی آن به مسئول آزمایشگاه تحویل دهند. پس از پیکربندی FPGA اجرای صحیح برنامه فوق را بر روی برد FPGA نشان دهند. همچنین لازم است دقت شود که ارتباط پردازنده با واحدهای I/O (کلیدهای و LED های روی برد) باید از طریق امکان Memory Mapped I/O پردازنده انجام گیرد.