طراحی مبتنی بر PLD با VHDL

طراحی، سنتز و پیاده سازی RTL

وایبهاو تاراته

مترجم:

دکتر موسی یوسفی

www.ketab.ir

سرشناسه : تاراته، وایببهاو
Taraate, Vaibbhav
عنوان و نام پدیدآور : طراحی مبنی بر PLD با VHDL/ [وایببهاو تاراته ترجمه] موسی یوسفی.
مشخصات نشر : تبریز، دانشگاه شهید مدنی آذربایجان، ۱۳۹۸.
مشخصات ظاهری : ۴۰۵ ص
شابک : ۹۷۸-۶۲۲-۶۷۳۷-۰۲-۹
وضعیت فهرست‌نویسی : فیپا
یادداشت : عنوان اصلی: PLD Based Design with VHDL RTL Design, Synthesis and Implementation.
موضوع : دستگاه‌های منطقی قابل برنامه‌ریزی
موضوع : Programmable logic devices
موضوع : وی. اچ. دی. ال (زبان توصیفی سخت‌افزار)
موضوع : (VHDL) Compouter hardware description language
شناسه افزوده : یوسفی، موسی، ۱۳۵۹-، مترجم
شناسه افزوده : دانشگاه شهید مدنی آذربایجان
رده بندی کنگره : TK ۷۸۸۲
رده بندی دیویی : ۶۲۱/۳۹۵
شماره کتابشناسی ملی : ۵۸۳۵۴۸۲

انتشارات دانشگاه شهید مدنی آذربایجان

- عنوان کتاب : طراحی مبتنی بر PLD با VHDL
- تألیف : تاراته، وایببهاو
- ترجمه : دکتر موسی یوسفی
- ویراستار علمی : دکتر خلیل منفردی
- ویراستار ادبی : دکتر رضا ذاکر
- نوبت چاپ : اول- تبریز- ۱۳۹۸
- قطع : وزیری- ۴۰۵ ص
- نشر : انتشارات دانشگاه شهید مدنی آذربایجان
- شابک : ۹۷۸-۶۲۲-۶۷۳۷-۰۲-۹
- شمارگان : ۵۰۰ جلد
- قیمت : ۶۰۰.۰۰۰ ریال
- چاپ : سایلان

www.ketab.ir

فهرست مطالب

پیشگفتار ۱۷

فصل ۱: مقدمه‌ای بر HDL

چکیده ۲۱
۱-۱ تاریخچه HDL ۲۲
۱-۲ تجرید طراحی منطقی و سیستمی ۲۳
۱-۳ نمونه‌سازی ASIC ۲۵
۱-۴ اصول و طراحی مدار مجتمع ۲۹
۱-۴-۱ برنامه‌نویسی RTL ۳۰
۱-۴-۲ تصدیق رفتار (تایید عملکرد) ۳۱
۱-۴-۳ سنتز ۳۱
۱-۴-۴ طراحی فیزیکی ۳۲
۱-۵ مقایسه زبان برنامه‌نویسی با HDL ۳۲
۱-۵-۱ فراگیرشدن و سیر تکاملی VHDL ۳۲
۱-۶ توصیف طرح با VHDL ۳۷
۱-۶-۱ طراحی ساختاری ۳۷
۱-۶-۲ طراحی رفتاری ۳۸
۱-۶-۳ طراحی با RTL قابل سنتز ۳۹

www.ketab.ir

۱-۷ موارد برجسته و ساختارهای کلیدی VHDL ۴۱
۱-۸ خلاصه ۴۴

فصل ۲: مدارهای منطقی پایه و توصیف VHDL

۲-۱ مقدمه‌ای برای منطق ترکیبی ۴۶
۲-۲ دروازه‌های منطقی و RTL قابل سنتز با استفاده از VHDL ۴۷
۲-۲-۱ NOT یا منطق معکوس کننده ۴۸
۲-۲-۲ منطق OR دو ورودی ۵۰
۲-۲-۳ منطق NOR دو ورودی ۵۱
۲-۲-۴ منطق AND دو ورودی ۵۳
۲-۲-۵ منطق NAND دو ورودی ۵۵
۲-۲-۶ منطق XOR دو ورودی ۵۶
۲-۲-۷ منطق XNOR دو ورودی ۵۹
۲-۲-۸ منطق سه حالته (بافر سه حالته) ۶۱
۲-۳ جمع‌کننده ۶۳
۲-۳-۱ نیم‌جمع‌کننده ۶۳
۲-۳-۲ تمام‌جمع‌کننده ۶۵
۲-۴ مبدل‌های کد ۶۷
۲-۴-۱ مبدل کد دودویی به گری ۶۷
۲-۴-۲ مبدل کد گری به کد دودویی ۶۸
۲-۵ خلاصه ۶۹

فصل ۳: VHDL و ساختارهای کلیدی

۳-۱ نمونه طراحی VHDL ۷۲
۳-۲ پیکربندی و معماری‌های چندگانه ۷۶
۳-۲-۱ پیکربندی و معماری چندگانه ۷۶
۳-۳ انواع داده و اشیاء ۷۸
۳-۳-۱ انواع داده عددی ۷۸

www.ketab.ir

۳-۳-۱-۱ مجموعه داده شمارشی ۷۸
۳-۳-۱-۲ داده صحیح ۷۹
۳-۳-۱-۳ انواع داده فیزیکی ۷۹
۳-۳-۱-۴ داده طبیعی ۸۰
۳-۳-۲ داده مرکب ۸۰
۳-۳-۲-۱ آرایه‌ها ۸۱
۳-۳-۲-۲ رکوردها ۸۲
۳-۳-۳ [illegible] داده ۸۲
۳-۳-۴ [illegible]ت‌ها ۸۲
۳-۳-۴-۱ م[illegible]یر ۸۳
۳-۳-۴-۲ [illegible]گنال ۸۳
۳-۳-۴-۳ فایل ۸۴
۳-۴ انتساب‌های سیگنال ۸۴
۳-۴-۱ مثال‌های انتساب سیگنال ۸۵
۳-۵ انتساب متغیر ۸۵
۳-۵-۱ مثال انتساب‌های متغیر ۸۷
۳-۶ ساختارهای همزمان ۸۸
۳-۶-۱ When Else ۸۸
۳-۶-۲ With Select ۹۰
۳-۶-۳ پروسس ۹۰
۳-۷ ساختارهای ترتیبی ۹۴
۳-۷-۱ If then Else ۹۴
۳-۷-۲ If Then Else تودرتو ۹۷
۳-۷-۳ Case ۹۹
۳-۸ مدل‌سازی منطق ترتیبی ۱۰۰
۳-۸-۱ ثبات چهاربیتی ۱۰۱
۳-۸-۲ لچ چهار بیتی ۱۰۲

www.ketab.ir

۳-۹ عبارت‌های Wait ۱۰۲
۳-۹-۱ Wait on ۱۰۲
۳-۹-۲ Wait for ۱۰۴
۳-۹-۳ Wait Until ۱۰۵
۳-۱۰ حلقه ۱۰۶
۳-۱۰-۱ Loop ۱۰۶
۳-۱۰-۲ While Loop ۱۰۷
۳-۱۰-۳ For Loop ۱۰۷
۳-۱۱ خلاصه ۱۰۸

فصل ۴: طراحی منطق ترکیبی با استفاده از ساختارهای VHDL

۴-۱ منطق و تأخیرهای ترکیبی ۱۱۲
۴-۱-۱ منطق ترکیبی زنجیره‌ای ۱۱۴
۴-۱-۲ منطق ترکیبی موازی ۱۱۵
۴-۲ مدارهای محاسباتی ۱۱۵
۴-۲-۱ جمع‌کننده چند بیتی ۱۱۶
۴-۲-۲ تفریق‌کننده – جمع‌کننده چند بیتی ۱۱۷
۴-۲-۳ ضرب‌کننده ۱۱۹
۴-۲-۴ مقایسه‌کننده‌ها ۱۲۰
۴-۳ مبدل کد ۱۲۳
۴-۳-۱ مبدل کد دودویی به افزودنی -۳ ..... ۱۲۴
۴-۳-۲ مبدل کد BCD به ۷ قسمتی ۱۲۶
۴-۴ مالتی‌پلکسرها ۱۲۸
۴-۴-۱ مالتی‌پلکسر به عنوان منطق جامع ۱۳۰
۴-۴-۱-۱ MUX 2:1 ..... ۱۳۱
۴-۴-۱-۲ MUX با استفاده از 'if then else' تو در تو ۱۳۳
۴-۴-۱-۳ MUX 4:1 با استفاده از ساختار 'case' ۱۳۵
۴-۵ دیکدر ۱۳۵

www.ketab.ir

۴-۵-۱ دیکدر ۳ در۸ دارای سیگنال فعال‌ساز با استفاده از عبارت 'case' ۱۳۵
۴-۵-۲ دیکدر ۲ به ۴ دارای فعال ساز با بکارگیری 'case' ۱۴۰
۴-۶ انکدرها ۱۴۰
۴-۶-۱ انکدر اولویت‌دار ۱۴۱
۴-۷ خلاصه ۱۴۷

فصل ۵: طراحی مدارهای منطقی ترتیبی

۵-۱ مدارهای منطقی ترتیبی ۱۵۰
۵-۱-۱ پارامترهای زمانی و ناپایداری در مدارهای منطقی ترتیبی ۱۵۳
۵-۱-۱-۱ زمان استقرار ۱۵۴
۵-۱-۱-۲ زمان نگهداری ۱۵۴
۵-۱-۱-۳ تاخیر انتشار فلیپ فلاپ ۱۵۴
۵-۲ لچ‌های D در طراحی ۱۵۶
۵-۲-۱ لچ D حساس به سطح مثبت ۱۵۷
۵-۲-۲ لچ D حساس به سطح منفی ۱۵۸
۵-۲-۳ تفریق‌کننده – جمع‌کننده چند بیتی ۱۶۰
۵-۲-۴ لچ D حساس به سطح منفی دارای ورودی نشاندن و پاک کردن ۱۶۲
۵-۲-۵ لچ D حساس به سطح مثبت با نشاندن و پاک کردن غیرهمزمان ۱۶۲
۵-۳ فلیپ فلاپ ۱۶۴
۵-۳-۱ فلیپ فلاپ D حساس به لبه مثبت ۱۶۴
۵-۳-۲ فلیپ فلاپ D حساس به لبه منفی ۱۶۶
۵-۴ بازنشانی همزمان و غیرهمزمان ۱۶۷
۵-۴-۱ فلیپ فلاپ D با بازنشانی همزمان ۱۶۷
۵-۴-۲ فلیپ فلاپ D با بازنشانی همزمان ۱۶۹
۵-۵ زمان‌بندی مدارهای ترتیبی ۱۷۱
۵-۶ شمارنده‌های همزمان ۱۷۲
۵-۶-۱ شمارنده بالا شمار ۴ بیتی ۱۷۳
۵-۶-۲ شمارنده ۴ بیتی پایین شمار ۱۷۵

www.ketab.ir

۵-۶-۳ بالا شمار BCD ۱۷۶
۵-۶-۴ پایین شمار BCD ۱۷۹
۵-۶-۵ پایین-بالا شمار BCD ۱۸۰
۵-۷ شمارنده‌گری ۱۸۲
۵-۸ شمارنده حلقوی ۱۸۴
۵-۹ شمارنده جانسون ۱۸۶
۵-۱۰ شیفت ریجسترها ۱۸۸
۵-۱۰-۱ شیفت ریجسترهای چپ و راست ۱۹۰
۵-۱۰-۲ شیفت ریجستر با ورودی خروجی موازی (PIPO) ۱۹۰
۵-۱۱ طراحی‌های غیرهمزمان ۱۹۳
۵-۱۲ خلاصه ۱۹۳

فصل ۶: معرفی PLD

۶-۱ تاریخچه و سیر تکاملی PLDها ۱۹۶
۶-۲ تراشه منطقی قابل برنامه‌ریزی ساده SPLD ۱۹۹
۶-۲-۱ حافظه فقط خواندنی قابل برنامه ریزی (PROM) ۲۰۱
۶-۲-۲ منطق آرایه‌ای قابل برنامه‌ریزی PAL ۲۰۲
۶-۲-۳ آرایه منطقی قابل برنامه‌ریزی ۲۰۳
۶-۳ تراشه‌های منطقی قابل برنامه‌ریزی پیچیده ۲۰۶
۶-۴ آرایه‌های گیت قابل برنامه‌ریزی ۲۰۸
۶-۴-۱ مفهوم LUT و ایجاد منطق ترکیبی ۲۱۱
۶-۴-۲ طراحی VHDL و پیاده سازی با استفاده از CLB ۲۱۱
۶-۴-۳ بلوک IO ۲۱۸
۶-۴-۴ بلوک RAM (BRAM) ۲۱۹
۶-۴-۵ منابع ساعت (پالس دهی) ۲۲۲
۶-۴-۵-۱ مسیرهای ساعت و داده با استفاده از بافرهای ساعت ۲۲۵
۶-۴-۶ بلوک‌های DSP و ضرب‌کننده‌ها ۲۲۵
۶-۴-۷ استانداردهای IO و منابع مسیر یابی ۲۲۶

www.ketab.ir

۶-۵ رهنمون‌ها و سناریوی طراحی کاربردی ۲۲۸
۶-۵-۱ استراتژی بازنشانی ۲۲۸
۶-۵-۱-۱ بازنشانی همزمان ۲۲۹
۶-۵-۱-۲ بازنشانی غیر همزمان ۲۳۱
۶-۵-۲ طرح‌های همزمان در برابر غیرهمزمان ۲۳۲
۶-۵-۳ استراتژی تولید ساعت (پالس دهی) ۲۳۴
۶-۵-۳-۱ تک ساعت اصلی ۲۳۴
۶-۵-۳-۲ شمارنده‌های حلقوی ۲۳۴
۶-۵-۳-۳ پالس دهی لبه مخلوط ۲۳۴
۶-۵-۳-۴ ساعت‌های دروازه‌ای ۲۳۵
۶-۶ خلاصه ۲۳۵

فصل ۷: طراحی و شبیه‌سازی به کمک ساختارهای VHDL

۷-۱ شبیه‌سازی با VHDL ۲۳۸
۷-۱-۱ برنامه تست MUX 4:1 ..... ۲۳۹
۷-۱-۲ برنامه تست بالاشمار ۴ بیتی دودویی ۲۴۱
۷-۲ توابع ۲۴۱
۷-۳ پکیج‌ها ۲۴۸
۷-۳-۱ استفاده از پکیج در طراحی ۲۴۹
۷-۴ مشخصه‌ها ۲۵۱
۷-۴-۱ مشخصه سیگنال ۲۵۲
۷-۴-۲ مشخصه آرایه ۲۵۲
۷-۵ دستکاری سیگنال ۲۵۳
۷-۵-۱ استفاده از فایل‌ها در شبیه‌سازی طرح ۲۵۳
۷-۵-۲ TEXTIO ۲۵۴
۷-۶ خلاصه ۲۵۶

فصل ۸: رهنمون‌های طراحی مبتنی بر PLD

۸-۱ قراردادهای نام گذاری ۲۶۰

www.ketab.ir

۸-۲ استفاده از متغیرها و سیگنال‌ها ۲۶۰
۸-۳ دسته‌بندی در طراحی ۲۶۳
۸-۴ رهنمون‌هایی برای استفاده از منطق سه حالته ۲۶۴
۸-۵ تسهیم منابع محاسباتی ۲۶۷
۸-۶ تکثیر منطق ۲۷۰
۸-۷ انتساب‌های تحریک چندتایی ۲۷۲
۸-۸ استنتاج لچ‌ها ۲۷۷
۸-۹ استفاده از عبارت If Then Else در برابر Case ۲۷۸
۸-۱۰ استفاده از روش لولای در طراحی ۲۸۱
۸-۱۱ حوزه ساعت حداکثر و انتقال داده ۲۸۵
۸-۱۲ IO دو جهتی ۲۸۸
۸-۱۳ ساعت کنترل شده ۲۹۰
۸-۱۴ طراحی با فعال‌ساز ساعت ۲۹۱
۸-۱۵ خلاصه ۲۹۲

فصل ۹: ماشین‌های حالت محدود

۹-۱ مقدمه FSM ۲۹۴
۹-۱-۱ ماشین مور ۲۹۴
۹-۱-۲ ماشین میلی ۲۹۶
۹-۲ روش‌های رمزگذاری FSM ۲۹۸
۹-۳ چگونه برای FSM مور برنامه VHDL بنویسیم؟ ۳۰۰
۹-۳-۱ الگوی طراحی FSM برای ماشین مور ۳۰۲
۹-۴ چگونه برای FSM میلی برنامه VHDL بنویسیم ؟ ۳۰۲
۹-۴-۱ الگوی طراحی FSM برای ماشین میلی ۳۰۴
۹-۵ مثال‌هایی از FSM و کد نویسی VHDL ۳۰۴
۹-۵-۱ FSM با کدگذاری دودویی ۳۰۷
۹-۵-۲ FSM شمارنده دودویی ۳۱۰
۹-۵-۳ FSM شمارنده One-Hot ۳۱۰

www.ketab.ir

۹-۶ منطق توازن با استفاده از FSM مور ۳۱۰
۹-۱-۶ ماشین مور: FSM سه پروسسی برای بررسی توازن ۳۱۳
۹-۷ منطق توازن با FSM میلی ۳۱۳
۹-۷-۱ ماشین میلی: FSM دو پروسسی برای بررسی توازن ۳۱۹
۹-۷-۲ ماشین میلی: FSM سه پروسسی برای بررسی توازن ۳۲۰
۹-۸ ماشین میلی آشکارساز ترتیب (توالی یاب) ۳۲۴
۹-۹ توالی یاب با رمز گذاری one-hot: ماشین حالت مور ۳۲۷
۹-۱۰ توالی یاب کدگذار One-Hot : ماشین میلی ۳۲۷
۹-۱۱ بهینه‌سازی FSM ۳۲۹
۹-۱۲ خلاصه ۳۳۳

فصل ۱۰: بهینه‌سازی سنتز با VHDL

۱۰-۱ فرایند طراحی FPGA ۳۳۶
۱۰-۱-۱ ثبت طرح (وارد کردن طرح) ۳۳۶
۱۰-۱-۲ شبیه‌سازی و سنتز طرح ۳۳۷
۱۰-۱-۳ پیاده‌سازی طرح ۳۳۸
۱۰-۱-۴ برنامه‌ریزی طرح ۳۳۸
۱۰-۲ روش‌های بهینه‌سازی سنتز ۳۳۹
۱۰-۲-۱ تخصیص منابع ۳۳۹
۱۰-۲-۲ فاکتورهای مشترک مورد استفاده در بهینه سازی ۳۴۱
۱۰-۲-۳ جابجایی تکه‌ای (قسمتی) از کد ۳۴۲
۱۰-۲-۴ جاکردن ثابت ۳۴۳
۱۰-۱-۵ حذف منطقه خاموش ۳۴۳
۱۰-۱-۶ بهره‌گیر از پارانتز ۳۴۴
۱۰-۲-۷ پارتیشن‌بندی و ساختار بندی طرح ۳۴۵
۱۰-۳ طراحی ALU ۳۴۶
۱۰-۳-۱ واحد منطق پردازنده و طراحی ۳۴۶
۱۰-۳-۱-۱ واحد منطق ۸ بیتی ۳۴۹

www.ketab.ir

۱۰-۳-۱-۲ واحد منطق ریزپردازنده با IO ثباتی ۳۵۱
۱۰-۳-۲ واحد محاسباتی ۳۵۱
۱۰-۳-۳ واحد منطقی و محاسباتی ۳۵۷
۱۰-۴ شیفت دهنده بارل ۳۵۹
۱۰-۵ بررسی کننده و مولد توازن ۳۶۰
۱۰-۵-۱ بررسی کننده توازن ۳۶۱
۱۰-۵-۱ مولد توازن ۳۶۱
۱۰-۶ حافظه‌ها ۳۶۳
۱۰-۶-۱ RAM دوگاهی ۳۶۳
۱۰-۶-۲ RAM دو جهته ۳۶۷
۱۰-۷ ضرب‌کننده‌ها ۳۶۸
۱۰-۸ خلاصه ۳۸۲

فصل ۱۱: پیاده‌سازی سیستم با Xillinx Vivado

۱۱-۱ پیاده‌سازی نمونه مثال با Xilinx Vivado ۳۸۴
۱۱-۱-۱ طرح‌ریزی طراحی ۳۸۴
۱۱-۱-۲ طرح‌ریز IO و محدودیت‌های IO ۳۸۸
۱۱-۱-۳ شبیه‌سازی عملکرد طرح ۳۹۱
۱۱-۱-۴ سنتز طرح ۳۹۱
۱۱-۲ پیاده‌سازی طرح ۳۹۳
۱۱-۲-۱ شبیه‌سازی زمانی ۳۹۵
۱۱-۳ برد توسعه یافته FPGA ۳۹۵
۱۱-۴ نمونه مثال طرح FIFO ۳۹۸
۱۱-۴-۱ محاسبه عمق FIFO غیر همزمان ۳۹۹
۱۱-۴-۲ طراحی FIFO با VHDL ۴۰۲
۱۱-۵ خلاصه ۴۰۵
مراجع ۴۰۵

www.ketab.ir

پیشگفتار

در یک دهه اخیر، پیچیدگی طراحی FPGA و ASIC به شدت افزایش یافته است. بخاطر این که برای طرح‌های پیچیده به تراشه‌های پیچیده و هوشمند نیاز است، نمونه‌سازی در فضای FPGA در طی این دهه گسترده‌تر شده است.

اکثر فروشندگان FPGA همانند شرکت‌های XILINX و Altera دارای FPGAهای پیچیده برای پیاده سازی و طراحی سیستم‌های روی تراشه (SOC) هستند. در طول این دهه، عصر مینیاتوری پیش‌بینی زیادی دارد مهمترین این چالش ها طراحی محصولات هوشمندی است که هزینه تولید کم، سرعت بالا، فضای مصرفی کم و توان مصرفی پایین داشته باشند.

تحت این شرایط برای تحقق یک ایده یا زمان تولید محصول، از FPGAهای پیچیده استفاده می‌شود که معماری آنها در یک دهه گذشته پیچیده‌تر شده است. حتی برای تایید اعتبار (تأیید عملکرد) سیستم‌های SOC از طرح‌هایی با چند FPGA پیچیده استفاده می‌شود. موضوعات این کتاب بر اساس آموزش سادگی طراحی‌ها با FPGA و نمونه‌سازی ASIC با استفاده از FPGAها شکل گرفته است. این کتاب حاوی طرح‌هایی است که با تعداد گیت کم و زیاد قابل پیاده سازی است. همچنین نوشتار این کتاب حاوی مباحثی است که اطلاعاتی درباره VHDL، سنتز، FPGAها و نمونه‌سازی ASIC ارائه می‌دهد.

فصل یک این کتاب سیر تکاملی طراحی منطقی و نیاز به استفاده از HDL و تفاوت بین HDL و دیگر زبان‌های سطح بالا را توضیح می‌دهد و همچنین در این فصل روش‌های مدل‌سازی مختلف با VHDL شرح داده شده است.

فصل ۲ این کتاب عناصر اصلی مدارهای ترکیبی و استفاده از آنها در طراحی سیستم‌ها را شرح می‌دهد و همچنین چگونگی نوشتن کد RTL قابل سنتز با ساختارهای VHDL را توضیح می‌دهد. مطالب این فصل برای آموزش ساختارهای پایه VHDL و سنتز طراحی‌های با گیت اندک برای نوآموزان مفید و مناسب است.

www.ketab.ir

در فصل ۳ ساختارهای مهم و کلیدی VHDL همانند پروسس، سیگنال‌ها، متغیرها، WHEN ELSE، WITH SELECT، IF THE ELSE و CASE توضیح داده می‌شود و همچنین در این فصل سناریوهای کاربردی و استفاده از این ساختارها نیز توضیح داده شده است.

مباحث فصل ۴ چگونگی نوشتن کد RTL مناسب و موثر با زبان VHDL را توصیف می‌کند. همچنین طراحی مدارهای ترکیبی همانند جمع کننده‌های چند بیتی، مالتی‌پلکسرها، دیکدرها و انکدرها شرح داده شده است. همچنین توضیحات این فصل، سنتز کد RTL در فرم زبان VHDL با جزئیات و سناریوهای کاربردی را پوشش می‌دهد.

توضیحات فصل ۵ این کتاب سناریوهای طراحی مدارهای ترتیبی و کد RTL در فرم زبان VHDL را برای لچ‌ها و فلیپ فلاپها را در بر می‌گیرد. به علاوه، مطالب این فصل حاوی شمارنده‌های BCD، شمارنده‌های دودویی، شمارنده‌های گری، شمارنده حلقوی، شمارنده جانسون و طراحی RTL و سنتز این مدارها است. پارامترهای زمانی، آنالیز زمانی سیستم‌های چند ساعتی و غیر همزمان و موضوعات ناپایداری و چگونگی فائق آمدن بر این مسایل در زمان طراحی سیستم‌ها در این فصل نیز بررسی می‌شود.

طراحی مبتنی بر PLD جزئیات طراحی سیستم‌های کاربردی به همراه مثال‌ها و سناریوهای طراحی برای SPLDها، CPLDها، و FPGAها در فصل ۶ قرار دارد. این فصل معماری FPGAهای شرکت‌های XILINX و Altera و چگونگی استفاده و نمونه‌سازی در طراحی را توضیح می‌دهد. همچنین راهنمایی خاص هر شرکت نیز در این فصل شرح داده شده است.

موضوعات فصل ۷ در برگیرنده ساختارهای VHDL و استفاده از آن برای تایید صحت عملکرد و شبیه‌سازی طرح است. مطالب قرار داده شده در این فصل برای آموزش نوشتن برنامه تست و چگونگی شبیه‌سازی طرح جهت عیب‌یابی بسیار مناسب است. مسایل کاربردی در زمان تایید صحت عملکرد طرح با استفاده از سناریوی کاربردی و مثال‌هایی در این راستا در این فصل گنجانده شده است.

مطالب فصل ۸ شامل رهنمون‌های طراحی و کدنویسی برای طرح‌های PLD است. چگونگی استفاده از VHDL برای طراحی‌های کارآمد با جزئیات کاربردی و ساختارهای VHDL قابل سنتز در این فصل قرار داده شده است همچنین تکنیک‌های مانند جزءبندی،

www.ketab.ir

منطق همزمان و موازی، تکثیر منطق و تسهیم منابع به همراه موضوعات توان پایین همانند کنترل کردن سیگنال ساعت و فعال‌سازی سیگنال ساعت در این فصل پوشش داده شده است.

مباحث فصل ۹ شامل ماشین‌های حالت محدود (متناهی) با استفاده از VHDL است. ماشین‌های میلی و مور و چگونگی استفاده از کد نویسی از آنها برای سیستم‌های آشکار ساز ترتیب (توالی یاب) و شمارنده ها در این فصل توضیح داده شده است. همچنین موضوعات همانند سنتز FSM و چگونگی بهبود کارآیی طراحی با بهره‌گیری از سناریوهای کاربردی بحث شده است. همچنین مباحث این فصل در برگیرنده رهنمون‌های سنتز FSM و تکنیک‌های بهینه‌سازی استفاده شونده در نمونه‌سازی ASIC در بستر FPGAهای پیچیده است.

در فصل ۱۰ طرح‌های پیچیده همانند ضرب‌کننده‌ها، شیفت‌دهنده‌های بارل، و منطق پردازنده همانند ALU و پاره‌ای از مدل های اساسی دیگر بررسی و مطالعه می‌شود. موضوعات مطرح شده در این فصل برای فهم مسایل سنتز در طرح‌های پیچیده و چگونگی رفع این موضوعات با استفاده از تکنیک های مطروحه در فصل ۷ مفید است.

فرایند طراحی بر اساس VIVADO و نمونه مثال با استفاده از VIVADO برای پیاده سازی طرح در فصل ۱۱ ارائه می‌شود نمونه مثال FIFO در این فصل گنجانده شده است.

کل فصول ۱ تا ۱۱ به شکلی مرتب شده است که شامل کدهای RTL در قالب زبان VHDL دارای گیت کم و سیستم‌های پیچیده‌تر به همراه سناریوهای مرتبط است. این کتاب برای نوآموزان، مهندسان طراح RTL و حرفه‌ای‌ها مفید است. امیدوارم که موضوعات مطروحه در این کتاب اطلاعات کافی برای یادگیری مناسب ساختارهای VHDL و استفاده از آن در نمونه‌سازی ASIC را به خوانندگان ارائه دهد.

وایبهاو، تاراته

www.ketab.ir