بسم الله الرحمن الرحیم

سازمان اطلاعات و امنیت کشور
جمهوری اسلامی ایران

٤٠٨١٠

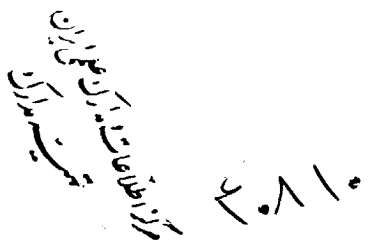
دانشگاه صنعتی اصفهان

دانشکده برق و کامپیوتر

طراحی یک کنترلر باس PCI و پیاده‌سازی آن بر روی FPGA

پایان‌نامهٔ کارشناسی ارشد در رشتهٔ مهندسی برق-کنترل

۱۳۸۱ /۴/ ۲۶



حسین نیک‌آیین

استاد راهنما:

دکتر سید مسعود سیدی

۱۳۷۹

دانشگاه صنعتی اصفهان

دانشکده برق و کامپیوتر

پایان‌نامهٔ کارشناسی ارشد رشتهٔ مهندسی برق-کنترل آقای حسین نیک‌آیین

تحت عنوان

طراحی یک کنترلر باس PCI و پیاده‌سازی آن بر روی FPGA

در تاریخ ۱۳۷۹/۱۲/۲۰ توسط کمیتهٔ تخصصی زیر مورد بررسی و تصویب نهایی قرار گرفت.

۱ـ استاد راهنمای پایان‌نامه — دکتر سید مسعود سیدی

۲ـ استاد مشاور پایان‌نامه — دکتر شادرخ سماوی

سرپرست تحصیلات تکمیلی دانشکده — دکتر علی‌محمد دوست‌حسینی

هوالحق

ای قادر متعال که دست توانایت از مافوق پهنهٔ گیتی سلسله جنبان هستی است. و شگفتی هر پدیدهٔ تازه‌ای وامدار خالقیت بی‌همتای توست. ای خدای مهربان که بدون یاریت قدمی برنداشته و برنخواهم داشت، برای همهٔ مهربانی‌هایت تو را سپاس!

اکنون که به لطف پروردگار این تحقیق را به اتمام می‌رسانم جای دارد که از زحمات تمامی سرورانی که انجام این کار با مدد دانش و محبت‌های ایشان میسر شده قدردانی کنم.

لذا از جناب آقای دکتر سیدی به خاطر راهنمایی‌ها و زحمات فراوان ایشان در طی انجام این پروژه و از جناب آقای دکتر سماوی به خاطر محبت‌ها و عنایت‌های ایشان در طول دوران تحصیل و در طی انجام این پروژه و از جناب آقای مهندس اخوان‌صراف و سایر مسئولین محترم پژوهشکدهٔ برق و کامپیوتر به خاطر فراهم آوردن زمینهٔ این تحقیق کمال امتنان و سپاسگزاری را دارم. همچنین از جناب آقای دکتر سعیدی به خاطر دقت نظر و انتقادهای به جایشان تشکر می‌نمایم.

کلیهٔ حقوق مادی مترتب بر نتایج مطالعات،

ابتکارات و نوآوری‌های ناشی از تحقیق موضوع

این پایان‌نامه (رساله) متعلق به دانشگاه صنعتی

اصفهان است.

فهرست مطالب

عنوان **صفحه**

فهرست مطالب ..... شش
فهرست اشکال ..... نه
چکیده ۱

فصل اول مقدمه

فصل دوم مروری مختصر بر باس‌های مختلف

۲ـ۱ انواع باس ۵
۲ـ۲ ویژگیهای باس PCI ۱۰
۲ـ۳ اتصال به باس PCI ۱۱

فصل سوم عملکرد باس PCI

۳ـ۱ مؤلفه‌های باس PCI ۱۴
۳ـ۲ عملیات انتقال بر روی باس PCI ۱۶
۳ـ۳ عملیات توزیع باس ۲۰
۳ـ٤ عملیات خواندن و نوشتن ۲٤
۳ـ٥ بررسی صحت انتقال اطلاعات ۳۰
۳ـ٦ فضای Configuration ۳۲
۳ـ۷ قطع عملیات انتقال ٤٤

فصل چهارم بلوک دیاگرام کلی کنترلر

٤ـ۱ پیش‌زمینهٔ طرح ٥۰
٤ـ۲ شرح بلوک دیاگرام کنترلر ٥٥
٤ـ۳ شرح عملکرد کنترلر ٥۸

فصل پنجم طراحی اجزاء کنترلر

٥ـ۱ بلوک Initiator State Machine ٦٤

ه-۲ بلوک Target State Machine ۶۸
ه-۳ بلوک Data Flow ۷۳
ه-٤ بلوک Address Decoder ۷٤
ه-٥ بلوک Parity Generator ۷٤
ه-٦ بلوک Configuration ۷٥
ه-۷ بلوک Control Box ۷۸
ه-۸ بلوک I/O ۸۱
ه-۹ بلوک Fifo ۸۲
ه-۱۰ بلوک Add_On Bus Sequencer ۸۳

فصل ششم شبیه‌سازی و پیاده‌سازی کنترلر طراحی شده

۶-۱ حالت Target ۸۵
۶-۲ حالت Master ۹۲
۶-۳ سنتز مدار ۹۷
۶-٤ شبیه‌سازی زمانی ۹۷
۶-٥ تست سخت‌افزاری ۱۰۲

فصل هفتم نتیجه‌گیری و پیشنهادات

ضمیمهٔ A

A-۱ بلوک Initiator State Machine ۱۰۷
A-۲ بلوک Target State Machine ۱۱۲
A-۳ بلوک Data Flow ۱۱۶
A-٤ بلوک Address Decoder ۱۱۸
A-٥ بلوک Parity Generator ۱۱۹
A-٦ بلوک Configuration ۱۲۰
A-۷ بلوک Control Box ۱۲۷
A-۸ بلوک I/O ۱۳۱
A-۹ بلوک Fifo ۱۳۳
A-۱۰ بلوک Add_on Bus Sequencer ۱٤۱
A-۱۱ برنامهٔ VHDL اصلی کنترلر ۱٤٥
A-۱۲ برنامهٔ VHDL تست کنترلر ۱٥٤

ضمیمهٔ B

B-۱ ایجاد پروژه ۱٦٤
B-۲ مدیریت پروژه ۱٦٦
B-۳ ابزارهای ورودی طرح ۱٦۷
B-٤ ابزار سنتز ۱٦۸
B-٥ ابزار شبیه‌سازی ۱٦۹
B-٦ ابزار پیاده‌سازی ۱٦۹
B-۷ ابزار برنامه‌ریزی ۱۷۰
B-۸ شبیه‌سازی سخت‌افزاری ۱۷۰

فهرست اشکال و جداول

شکل ۲ـ۱: اتصال وسایل مختلف به باس PCI ۹
شکل ۲ـ۲: استفاده از Standard PCI Chipset برای دو حالت کاربرد خاص و کاربرد عمومی. ۱۲
شکل ۲ـ۳: نمایش یک Fixed PCI Cores ۱۳
شکل ۳ـ۱: ساختار باس PCI ۱۵
شکل ۳ـ۲: دیاگرام زمانی یک عملیات انتقال بر روی باس PCI ۱۶
جدول ۳ـ۱: انواع عملیات انتقال در روی باس PCI ۱۷
شکل ۳ـ۳: توزیع‌کنندهٔ باس PCI ۲۰
شکل ۳ـ٤: توزیع باس بین دو آغازکننده. ۲۳
شکل ۳ـ۵: یک نمونه عملیات خواندن. ۲۷
شکل ۳ـ٦: یک نمونه عملیات نوشتن. ۲۸
شکل ۳ـ۷: عملیات نوشتن بهبود یافته از نظر سرعت انتقال. ۲۹
شکل ۳ـ۸: نحوهٔ ایجاد و چک کردن Parity در عملیات خواندن و نوشتن. ۳۱
شکل ۳ـ۹: تصویر فضای Configuration ۳۲
شکل ۳ـ۱۰: محتوای پورت 0CF8 هنگام عملیات خواندن یا نوشتن در فضای Configuration. ۳٤
شکل ۳ـ۱۱: Command Register ۳۵
شکل ۳ـ۱۲: Status Register ۳۹
شکل ۳ـ۱۳: base address register برای دو حالت فضای حافظه و فضای I/O ٤۲
شکل ۳ـ۱٤: قطع عملیات انتقال به صورت disconnect نوع A ٤٦
شکل ۳ـ۱۵: قطع عملیات انتقال به صورت disconnect نوع B ٤۷
شکل ۳ـ۱٦: قطع عملیات انتقال به صورت Retry ٤۷
شکل ۳ـ۱۷: قطع عملیات انتقال به صورت Target Abort ٤۸
شکل ٤ـ۱: ساختار یک سیستم رادار. ۵۰
شکل ٤ـ۲: نمودار زمان‌بندی نوشته شدن اطلاعات ورودی در حافظهٔ کارت. ۵۱
شکل ٤ـ۳: کارت PCI بر اساس کنترلر طراحی شده. ۵۳
شکل ٤ـ٤: بلوک دیاگرام کلی کنترلر. ۵٤
شکل ٤ـ۵: زمان‌بندی نوشتن اطلاعات داخل Fifo و انتقال آن به حافظهٔ داخلی سیستم. ۵۹
شکل ٤ـ٦: نحوه قرار گرفتن اطلاعات دو پالس تریگر داخل حافظهٔ سیستم. ٦۱
شکل ٤ـ٦: نحوه قرار گرفتن اطلاعات دو پالس تریگر داخل حافظهٔ سیستم. ٦۳
شکل ۵ـ۲: نمودار Initiator State Machine ٦۷
شکل ۵ـ۳: نمودار Target State Machine ۷۰

شکل ٥ـ٦: Command Register ٧٦
شکل ٥ـ٧: Status Register ٧٧
شکل ٥ـ٨: Control Register ٧٩
جدول ٥ـ١: شرح بیت‌های Control Register ٨٠
شکل ٥ـ٩: بلوک Fifo ٨٣
شکل ٥ـ٩: بلوک Fifo ٨٧
شکل ٦ـ٢: خواندن و نوشتن در Base Address Register0 ..... ٨٨
شکل ٦ـ٣: خواندن و نوشتن در Command Register ٨٩
شکل ٦ـ٤: عملیات خواندن و نوشتن مربوط به اولین فضای I/O (Address Register) ٩٠
شکل ٦ـ٥: عملیات خواندن و نوشتن مربوط به دومین فضای I/O (Control Register) ٩١
شکل ٦ـ٦: عملیات خواندن و نوشتن مربوط به پورت‌های خارجی کنترلر ٩٣
شکل ٦ـ٧: نمودار سیگنال‌های CLKT و 0E1-OE4 بعد از فعال شدن تریگر ٩٤
شکل ٦ـ٨: پس دادن باس توسط کنترلر به علت اتمام داده‌های Fifo ٩٦
شکل ٦ـ٩: نتایج پیاده‌سازی طرح بر روی چیپ XC4020XL با بسته بندی PQ160 ..... ٩٨
شکل ٦ـ١٠: نتایج زمانی به دست آمده از پیاده‌سازی طرح بر روی چیپ XC4020XL – 09 با
بسته بندی PQ160 ..... ٩٨
شکل ٦ـ١١: نتایج شبیه‌سازی زمانی خواندن و نوشتن در Base Address Register0 ..... ٩٩
شکل ٦ـ١٢: نتایج شبیه‌سازی زمانی مربوط به نوشتن در اولین فضای I/O (Control Box) ١٠٠
شکل ٦-١٣: نتایج شبیه‌سازی زمانی عملیات نوشتن در حافظه توسط کنترلر ١٠١
شکل ٦ـ١٤: نتایج شبیه‌سازی نرم‌افزاری به صورت منطقی ١٠٣
شکل ٦ـ١٥: نتایج شبیه‌سازی نرم‌افزاری به صورت زمانی ١٠٤
شکل B-1: ایجاد پروژه ١٦٥
شکل B-2: مشخص نمودن نام و مسیر پروژه ١٦٦
شکل B-3: نمای برنامهٔ مدیریت پروژه ١٦٧
شکل B-4: ویرایشگر برنامه‌های HDL ١٦٨
شکل B-5: تصویر برنامهٔ Simulator ١٦٩
شکل B-6: مراحل مختلف پیاده‌سازی ١٧٠
شکل B-7: کارت XS 40 متعلق به شرکت XESS ١٧١
شکل B-8: نرم‌افزار GXSLOAD ١٧٢
شکل B-9: برنامه WinXSPort متعلق به شرکت XESS ١٧٢

مرکز اطلاعات و مدارک علمی ایران

چکیده

باس PCI یک پروتکل رابط است که به صورت گذرگاه ارتباطی بین اجزای اصلی کامپیوتر و وسایل جانبی به کار می‌رود. این باس یک راه حل بسیار خوب برای کاربردهای با پهنای باند زیاد و کارآیی بالا است اما مسئلهٔ محدود کننده در استفاده از آن پیچیدگی زیاد آن است.در این میان استفاده از آرایه های قابل برنامه ریزی جای خود را باز نموده است.
در این رساله ما ابتدا به شرح مختصری از باس PCI وعملیات انتقال بر روی آن پرداخته سپس به بررسی کنترلر طراحی شده می‌پردازیم.این کنترلر از ده بلوک مجزا تشکیل شده که طراحی آنهاتوسط زبان برنامه نویسی توصیف سخت‌افزار صورت گرفته است ودر نهایت کل طرح بعد از شبیه‌سازی،سنتز و پیاده‌سازی شده است.

فصل اول

مقدمه

باس PCI[1] به عنوان یک پروتکل رابط که به صورت گذرگاه ارتباطی بین اجزای اصلی کامپیوتر و وسایل جانبی به کار می‌رود، روز به روز مقبولیت بیشتری پیدا می‌کند. این باس یک راه حل بسیار خوب برای کاربردهای با پهنای باند زیاد و کارآیی بالا است. قبل از ارائهٔ استاندارد PCI باس‌های مختلفی ارائه شده‌اند که عیوب هر کدام از آنها موجب به وجود آمدن استاندارد جدیدتری شده است. اما با ظهور باس PCI از سال ۱۹۹۲ تا کنون استاندارد جدیدتری که قابل رقابت با این باس باشد به وجود نیامده است. علت این امر ویژگی‌های خاص استاندارد PCI می‌باشد. مهم‌ترین ویژگی‌های این باس پهنای باند زیاد، تعداد پین کم، مستقل بودن از نوع پردازندهٔ سیستم و قابلیت تنظیم اتوماتیک وسایل متصل به این باس می‌باشد. به دلیل همین ویژگی‌ها و کارآیی زیاد، این باس مقبولیت زیادی در میان تولیدکنندگان کامپیوتر و وسایل جانبی پیدا کرده است. استفاده از این باس به کامپیوترهای شخصی محدود نشده بلکه در زمینه‌های ارتباطات، سوئیچ‌های مخابراتی، کامپیوترهای صنعتی و ابزار دقیق عمومیت زیادی پیدا نموده است. از مهمترین کاربردهای صنعتی برای باس PCI می‌توان به ساخت کارت‌های جمع‌آوری اطلاعات با نرخ انتقال بالا اشاره نمود. این کارت‌ها برای کاربردهای اتوماسیون و سیستم‌های مبتنی بر پردازش سیگنال‌های دیجیتال (مانند سیستم‌های رادار[2]، سونار[3]، دوربین‌های دیجیتال[4] و وسایل تصویربرداری پزشکی) استفاده می‌گردد.

---

1. Peripheral Component Interconnect
2. Radar
3. Sonar
4. Digital Video

هر چند به دلیل ویژگی‌های جالب باس PCI زمینهٔ استفاده از آن در بسیاری کاربردها وجـود دارد، امـا مسئلهٔ محدودکننده در این استفاده پیچیدگی بسیار زیاد آن و فرکانس کار بالای این باس می‌باشـد. بـه طـوری که بدون بهره‌گیری از تکنولوژی VLSI[1] ساخت مدارهای واسط جـهت کـار بـا ایـن بـاس تقریباً غیرممکن می‌باشد. در این میان استفاده از FPGA[2] که متعلق به خانوادهٔ آرایه‌های قابل برنامه‌ریزی می‌باشد جای خـود را در این صنعت باز نموده است. این امر به علت قابلیت پیاده‌سازی مدارات مجتمع با حجم بالا بـر روی FPGA در زمان بسیار کم در مقایسه با ساخت یک مدار مجتمع و یا استفاده از تکنولوژی ASIC[3] می‌باشد.

هدف ما در این پروژه طراحی یک PCI Controller مخصـوص کارت‌هـای جمع‌آوری اطلاعـات و پیاده‌سازی آن بر روی FPGA می‌باشد. کنترلر طراحی شده دارای قابلیت Master و Target می‌باشد.

طراحی بلوک‌های مختلف این کنترلر توسط زبان VHDL انجام پذیرفته و هر بلوک به صـورت مجـزا شبیه‌سازی شده است و در نهایت کل طرح شبیه‌سازی و سنتز گردیده است.

در فصل دوم ایـن رسـاله بـه مـرور اجمـالی بـر مفـهوم بـاس و معرفـی یـک سـری از باس‌هـای متـداول می‌پردازیم. سپس در پایان فصل به بیان خصوصیات ویژهٔ بـاس PCI و نـیز بررسـی راه‌حل‌هـای مرسـوم بـرای ساخت مدارات واسط بر روی باس PCI می‌پردازیم.

در فصل سوم به شرح استاندارد PCI پرداخته و مفاهیمی مانند "Master" و "Target" را بیان نموده و عملیات خواندن و نوشتن در باس PCI را همـراه بـا ذکـر مثال‌هـایی بـرای آن بررسـی می‌نمـاییم. همچنیـن در مورد چک کـردن صحـت انتقـال اطلاعـات توضیحـاتی خواهیـم داد. در ادامهٔ ایـن فصـل بـه بررسـی فضـای Configuration در باس PCI پرداخته و نیز در مورد قطع عملیات انتقال بحث خواهیم نمود.

در فصل چهارم به توضیح بلوک دیاگرام کلـی کنـترلر بـر اسـاس طـرح خـاصی کـه مـورد نظـر اسـت پرداخته و شرح مختصری راجع به بلوک‌های طرح و ارتباط آنها با یکدیگر داده می‌شود. بـرای طراحـی ایـن بلوک‌ها از زبان توصیف سخت‌افزار VHDL استفاده شده است. برنامـه‌های مربـوط بـه هـر بلـوک بـه نحـوی نوشته شده‌اند که قابل سنتز بوده و طرح نهایی حـاصل از کنـار یکدیگـر قـرار دادن ایـن بلوک‌هـا بـه صـورت کامل سنتز و پیاده‌سازی گردیده است.

در فصل پنجم به شرح کامل تک‌تک بلوک‌ها پرداختـه و بـه برنامهٔ VHDL مربـوط بـه هـر بلـوک و نتایج شبیه‌سازی آن اشاره خواهیم نمود.

در فصل ششم به بررسی نتایج شبیه‌سازی کنترلر حاصل از کنار یکدیگر قـرار دادن بلوک‌هـای معرفـی شده در فصل پنجم می‌پردازیم. همچنین در این فصل به بررسـی نتـایج حـاصل از سـنتز کنـترلر طراحـی شـده پرداخته و نتایج مربوط به پیاده‌سازی عملی بلوک Initiator State Machine در آزمایشگاه ارائه می‌گردد.

---

1. Very Large Scale Integrated Circuits
2. Field Programable Gate Arrays
3. Application Specific Integrated Circuits

در ضمیمه الف برنامهٔ هر کدام از بلوک‌ها و نتایج شبیه‌سازی آنها جهت مراجعه آورده شده است. همچنین در پایان این ضمیمه برنامهٔ اصلی کنترلر که در آن، این بلوک‌ها جایگزین شده‌اند، همراه با برنامهٔ تست کنترلر آورده شده است.

در ضمیمهٔ ب نرم‌افزارهای مورد استفاده جهت برنامه‌نویسی و شبیه‌سازی بلوک‌ها و نرم‌افزارهای مربوط به سنتز و پیاده‌سازی کنترلر آورده شده است.

فصل دوم
مروری مختصر بر باس‌های مختلف

۲ـ۱ انواع باس

در سیستم‌های مبتنی بر ریزپردازنده‌ها[1] معمولاً گذرگاه‌هایی برای عبور داده، آدرس و خطوط کنترل وجود دارد. نقش این گذرگاه‌ها ایجاد پل ارتباطی بین ریزپردازنده و سایر اجزاء سیستم مثل حافظه‌ها، پورت‌ها و وسایل جانبی می‌باشد.

از آنجایی که در اکثر سیستم‌ها تمام وسایل جانبی[2] که نیاز به برقراری ارتباط اطلاعاتی با ریزپردازنده دارند در روی برد اصلی سیستم (برد مادر[3]) قرار نمی‌گیرند، به یک سری کانکتور[4] در روی برد سیستم نیاز می‌باشد تا از طریق این کانکتورها مسیر عبور اطلاعات داده، آدرس و کنترل، به وسایل جانبی فراهم گردد.

وسایل جانبی مختلف ممکن است در یک سیستم ریزپردازنده استفاده گردد. مثل دیسک سخت[5]، نمایشگر[6]، چاپگر، موس، ... . برای ارتباط با این وسایل که از دید ریزپردازنده دنیای خارج تلقی می‌گردد، به یک کنترلر که در واقع یک رابط محسوب می‌شود نیاز است. این کنترلر اطلاعات را از طریق گذرگاه‌های

---

1. Microprocessor
2. Peripheral
3. Mother-Board
4. Connector
5. Hard Disk
6. Monitor