

دانشگاه صنعتی خواجه نصیرالدین طوسی

تابس ۱۳۰۷

دانشکده مهندسی برق و کامپیوتر

پایان نامه‌ی کارشناسی ارشد در رشته مخابرات-سیستم

موضوع پایان نامه:

تحقق عملی حمله‌ی تحلیل تفاضلی توان (DPA) روی پیاده‌سازی FPGA

الگوریتم رمزنگاری AES

استاد راهنما:

دکتر محمود احمدیان

استاد مشاور:

دکتر مسعود معصومی

نگارش:

مهدی معصومی

شهریور ماه ۱۳۸۹

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

تأییدیه هیأت داوران

هیأت داوران پس از مطالعه پایان نامه و شرکت در جلسه دفاع از پایان نامه تهیه شده تحت عنوان:
"تحقق عملی حمله‌ی تحلیل تفاضلی توان (DPA) روی پیاده‌سازی FPGA الگوریتم رمزنگاری
AES" توسط آقای مهدی معصومی، صحت و کفایت تحقیق انجام شده را برای اخذ درجه کارشناسی
ارشد در رشته: مهندسی برق گرایش مخابرات-سیستم در تاریخ / / ۱۳ مورد تأیید قرار می‌دهند.

امضاء	جناب آقای دکتر محمود احمدیان	۱- استاد راهنما
امضاء	جناب آقای دکتر مسعود معصومی	۲- استاد مشاور
امضاء	جناب آقای دکتر حسین شمسی	۳- ممتحن داخلی
امضاء	جناب آقای دکتر علی پاینده	۴- ممتحن خارجی
امضاء	جناب آقای دکتر خالوزاده	۵- نماینده تحصیلات تکمیلی

اظهار نامه دانشجو

عنوان پایان نامه:

تحقق عملی حمله‌ی تحلیل تفاضلی توان (DPA) روی پیاده‌سازی FPGA الگوریتم رمزنگاری AES

استاد راهنما: دکتر محمود احمدیان

استاد مشاور: دکتر مسعود معصومی

نام دانشجو: مهدی معصومی

شماره دانشجویی: ۸۶۰۰۸۸۴

اینجانب مهدی معصومی دانشجوی دوره کارشناسی ارشد مهندسی برق گرایش مخابرات-سیستم دانشکده مهندسی برق و کامپیوتر دانشگاه صنعتی خواجه نصیرالدین طوسی گواهی می‌نمایم که تحقیقات ارائه شده در این پایان‌نامه توسط شخص اینجانب انجام شده و صحت و اصالت مطالب نگارش شده مورد تأیید می‌باشد، و در موارد استفاده از کار دیگر محققان به مرجع مورد استفاده اشاره شده‌است. به‌علاوه گواهی می‌نمایم که مطالب مندرج در پایان‌نامه تاکنون برای دریافت هیچ نوع مدرک یا امتیازی توسط اینجانب یا فرد دیگری در هیچ جا ارائه نشده‌است و در تدوین متن پایان‌نامه چارچوب (فرمت) مصوب دانشگاه را به‌طور کامل رعایت کرده‌ام.

امضاء دانشجو:

تاریخ:

فرم حق طبع و نشر و مالکیت نتایج

۱- حق چاپ و تکثیر این پایان نامه متعلق به نویسنده آن می‌باشد. هرگونه کپی برداری بصورت کل پایان‌نامه یا بخشی از آن تنها با موافقت نویسنده یا کتابخانه دانشکده مهندسی برق دانشگاه صنعتی خواجه نصیرالدین طوسی مجاز می‌باشد.

ضمناً متن این صفحه نیز باید در نسخه تکثیر شده وجود داشته باشد.

۲- کلیه حقوق معنوی این اثر متعلق به دانشگاه صنعتی خواجه نصیرالدین طوسی می‌باشد و بدون اجازه کتبی دانشگاه به شخص ثالث قابل واگذاری نیست.

هم‌چنین استفاده از اطلاعات و نتایج موجود در پایان‌نامه بدون ذکر مرجع مجاز نمی‌باشد.

چکیده:

الگوریتم رمزنگاری Rijndael که در سال ۲۰۰۰ میلادی از سوی موسسه ملی استاندارد و فن‌آوری ایالات متحده (NIST) به‌عنوان الگوریتم استاندارد رمزنگاری داده مورد پذیرش قرار گرفت الگوریتم استاندارد پیشرفته رمزنگاری (Advanced Encryption Standard Algorithm) یا الگوریتم AES نام دارد و از سوی ISO و IEEE نیز به‌عنوان الگوریتم استاندارد رمزنگاری داده شناخته شده‌است. از سال ۲۰۰۰ تاکنون نحوه‌ی پیاده‌سازی سخت‌افزاری هرچه بهتر و مؤثرتر این الگوریتم و مقاومت آن در برابر حملات سخت‌افزاری و بخصوص حملات کانال جانبی از جمله مباحث مورد توجه صاحب‌نظران رمزنگاری بوده‌است. با توجه به اهمیت AES در صنعت و ارتباطات و کاربردهای متنوعی از قبیل کارت‌های هوشمند، وب سرورها، تلفن‌های سلولی، شبکه‌های ATM و ... به‌عنوان رمزکننده‌ی داده‌ها تلاش برای شکستن این الگوریتم از سوی افراد غیرمجاز و دسترسی به اطلاعات امنیتی صورت می‌پذیرد. دشمن می‌تواند با استفاده از روش‌های مختلفی امنیت سیستم‌های رمزنگاری را تهدید کند. یکی از این تهدیدات که موسوم به حملات کانال جانبی است از اطلاعاتی مثل توان مصرفی، زمان اجرای الگوریتم، تشعشعات الکترومغناطیسی و ... تراشه حین اجرای الگوریتم رمزنگاری استفاده کرده و کلید رمزنگاری را به‌دست می‌آورد. یک نوع حمله‌ی کانال جانبی موسوم به حمله‌ی تحلیل تفاضلی توان (DPA) از قوی‌ترین حملات سخت‌افزاری بوده و می‌تواند در زمان کوتاهی کلید رمز یک الگوریتم پیچیده‌ی رمزنگاری را فاش کند. با استفاده از تحلیل تفاضلی توان (DPA) می‌توان با اندازه‌گیری جریان تغذیه‌ی یک دستگاه، بخشی از کلید رمز یا تمام آن را کشف کرد. اگر شکل موج حاصله از جریان با آنچه که از مدل فرضی مصرف توان یک مدار به‌دست می‌آید شباهت داشته باشد، امنیت سیستم رمزنگاری به خطر می‌افتد. وقتی کلید رمزنگاری یک الگوریتم فاش شود به راحتی می‌توان آن سیستم را هک کرد و به اطلاعات آن دسترسی پیدا کرد. در این پایان‌نامه ابتدا یک پیاده‌سازی سخت‌افزاری از الگوریتم Rijndael روی FPGA انجام داده‌ایم. سپس از اطلاعاتی که این تراشه در هنگام اجرای الگوریتم AES می‌دهد استفاده کرده و حمله‌ی تحلیل تفاضلی توان را روی الگوریتم اعمال کردیم و موفق به بازیابی کلید رمزنگاری در مدت ۲ ساعت شدیم. بنابراین اگر الگوریتم و یا تراشه‌ی رمزنگاری غیرامن باشد می‌توان به راحتی و با استفاده از حملات سخت‌افزاری سیستم مذکور را به اصطلاح شکست و موفق به بازیابی کلید رمز آن شد. این پژوهش می‌تواند زمینه‌های جدیدی را در حوزه‌ی رمزنگاری و امنیت اطلاعات ایجاد کند. پیاده‌سازی دیگر حملات سخت‌افزاری یا حمله‌ی DPA از چشم‌اندازهای آینده‌ی این پژوهش است.

واژه‌های کلیدی

الگوریتم استاندارد رمزنگاری پیشرفته (AES)، حملات کانال جانبی، تحلیل تفاضلی توان (DPA)، تحلیل هم‌بستگی

۱	فصل اول: مقدمه
۹	فصل دوم: الگوریتم رمزنگاری AES
۱۱	۲-۱ مقدمه
۱۲	۲-۲ مروری بر برگزیدگان نهایی AES
۱۵	۲-۳ معیارهای انتخاب AES
۱۵	۲-۴ انتخاب Rijndael به عنوان AES
۱۸	۲-۵ مروری مختصر بر مفاهیم پایه ریاضی در Rijndael
۱۹	۲-۵-۱ میدان‌های متناهی
۱۹	۲-۵-۲ چند جمله‌ای‌ها روی میدان‌های متناهی
۲۰	۲-۵-۳ عملیات روی چند جمله‌ای‌ها در میدان‌های متناهی
۲۱	۲-۵-۴ ضرب در چند جمله‌ای‌های ثابت
۲۱	۲-۵-۵ توابع بولی
۲۲	۲-۵-۶ ترانهش
۲۳	۲-۵-۷ توابع آجرچین
۲۳	۲-۵-۸ تبدیلات بولی تکراری
۲۴	۲-۶ رمزنگاری قالبی
۲۵	۲-۶-۱ رمزنگاری قالبی تکراری
۲۷	۲-۶-۲ حالت‌های عملکرد سیستم‌های رمزنگاری قالبی
۲۷	۲-۶-۲-۱ حالت‌های رمزکنندگی
۲۸	۲-۶-۲-۲ حالت تولید رشته کلید
۲۸	۲-۶-۲-۳ حالت تصدیق پیام
۲۹	۲-۶-۲-۴ Hashing با استفاده از رمزنگاری قالبی
۲۹	۲-۷ مشخصات Rijndael
۲۹	۲-۷-۱ ورودی و خروجی RIJNDAEL
۳۰	۲-۷-۲ ساختار RIJNDAEL
۳۰	۲-۷-۲-۱ تبدیل دور در Rijndael
۳۲	۲-۷-۲-۲ تبدیل Sub-Bytes
۳۳	۲-۷-۲-۳ تبدیل Shift-Rows
۳۴	۲-۷-۲-۴ تبدیل Mix-Columns
۳۵	۲-۷-۲-۵ تبدیل Add-Round Key
۳۶	۲-۷-۲-۶ KeySchedule
۳۷	۲-۷-۳ رمزگشایی در RIJNDAEL
۳۹	فصل سوم: پیاده‌سازی سخت‌افزاری الگوریتم رمز Rijndael روی FPGA
۴۱	۳-۱ مقدمه
۴۲	۳-۲ مقدمه‌ای بر FPGA
۴۷	۳-۳ روش‌های طراحی و پیاده‌سازی سخت‌افزاری الگوریتم‌های رمزنگاری روی FPGA
۴۸	۳-۴ ملاحظات کلی در انتخاب FPGA برای پیاده‌سازی الگوریتم‌های رمز قالبی

۴۹ ۳-۵ معیارهای ارزیابی کارآیی پیاده‌سازی سخت‌افزاری یک الگوریتم رمزنگاری قالبی
۵۰ ۳-۶ معماری‌های مناسب برای پیاده‌سازی الگوریتم‌های رمزنگاری قالبی
۵۲ ۳-۷ گلوگاه‌های پیاده‌سازی Rijndael بر روی FPGA
۵۳ ۳-۸ مقایسه‌ی نتایج گزارش‌شده از پیاده‌سازی الگوریتم Rijndael روی FPGA
۵۴ ۳-۹ معماری‌های مناسب پیشنهادشده برای پیاده‌سازی سخت‌افزاری الگوریتم Rijndael
۵۵ ۳-۹-۱ پیاده‌سازی ضرب در میدان محدود $GF(2^8)$
۵۶ ۳-۹-۲ پیاده‌سازی MIX-COLUMNS/INV MIX-COLUMNS
۶۰ ۳-۹-۳ پیاده‌سازی SUB-BYTES/INV SUB-BYTES
۶۰ ۳-۹-۳-۱ استفاده از ریاضیات میدان‌های مرکب برای پیاده‌سازی S-Box
۶۵ ۳-۹-۳-۲ پیاده‌سازی Inv Sub-Bytes
۶۵ ۳-۹-۴ پیاده‌سازی KEY SCHEDULE
۶۹ فصل چهارم: حملات سخت‌افزاری روی سیستم‌های رمز قالبی
۷۱ ۴-۱ مقدمه
۷۳ ۴-۲ مروری بر ساختار کارت هوشمند
۷۴ ۴-۳ دسته‌بندی حملات کانال جانبی
۷۵ ۴-۴ حمله‌ی پروب‌گذاری
۷۶ ۴-۴-۱ حمله‌ی پروب‌گذاری نسبت به AES
۷۷ ۴-۵ حمله‌ی تحلیل زمانی
۷۸ ۴-۵-۱ مدل حمله‌ی تحلیل زمانی
۷۸ ۴-۵-۲ حمله‌ی زمانی نسبت به RIJNDAEL
۸۰ ۴-۶ حمله‌ی القاء خطا
۸۰ ۴-۶-۱ انواع خطا
۸۱ ۴-۶-۲ حمله‌ی تحلیل تفاضلی خطا
۸۲ ۴-۶-۳ حمله‌ی DFA نسبت به RIJNDAEL
۸۳ ۴-۷ حمله‌ی تشعشعات الکترومغناطیسی
۸۴ ۴-۸ حمله‌ی تحلیل توان
۸۴ ۴-۸-۱ حمله‌ی تحلیل توان برای شکستن رمز کارت‌های هوشمند
۸۵ ۴-۸-۲ حمله‌ی تحلیل ساده‌ی توان (SPA)
۸۷ ۴-۸-۳ حمله‌ی تحلیل تفاضلی توان (DPA)
۹۰ ۴-۸-۴ حمله‌ی DPA روی RIJNDAEL
۹۰ ۴-۸-۴-۱ DPA برای شکستن S-Box
۹۱ ۴-۸-۴-۲ DPA برای شکستن Add-Round Key
۹۳ فصل پنجم: نتایج پیاده‌سازی حمله‌ی DPA روی Rijndael
۹۵ ۵-۱ مقدمه
۹۵ ۵-۲ پیاده‌سازی سیستم رمزنگاری Rijndael روی FPGA Spartan-II
۹۵ ۵-۲-۱ ابزار جمع با کلید دور
۹۶ ۵-۲-۲ مرحله‌ی تعویض بایت‌ها
۹۶ ۵-۲-۳ پیاده‌سازی مرحله‌ی شیفت‌دهی سطرها
۹۶ ۵-۲-۴ ابزار ترکیب ستون‌ها

۹۸	۵-۲-۵ نتایج حاصل از سنتز الگوریتم
۹۹	۵-۲-۶ مقایسه‌ی نتایج حاصل از این پیاده‌سازی با کارهای مشابه
۱۰۰	۵-۳ مصرف توان مدارهای CMOS
۱۰۲	۵-۳-۱ توان مصرفی ایستا
۱۰۲	۵-۳-۲ توان مصرفی پویا
۱۰۴	۵-۴ شبیه‌سازی توان و مدل‌های توان برای حمله به سیستم‌های رمزنگاری
۱۰۵	۵-۴-۱ مدل فاصله‌ی همینگ
۱۰۶	۵-۴-۲ مدل وزن همینگ
۱۰۷	۵-۴-۳ مدل‌های دیگر توان
۱۰۸	۵-۵ تنظیمات آزمایشگاهی برای اعمال حملات تحلیل توان
۱۱۱	۵-۵-۱ مدار اندازه‌گیری توان
۱۱۲	۵-۵-۲ اسیلوسکوپ دیجیتال
۱۱۴	۵-۵-۳ تنظیمات آزمایشگاهی برای پیاده‌سازی حمله‌ی توان روی FPGA SPARTAN-II
۱۱۶	۵-۵-۴ معیار کیفیت برای تنظیمات آزمایشگاهی
۱۱۷	۵-۵-۴-۱ نویز الکترونیکی
۱۱۹	۵-۵-۴-۲ نویز سوئیچینگ
۱۲۰	۵-۶ تحلیل تفاضلی توان
۱۲۵	۵-۷ نتایج پیاده‌سازی حمله‌ی تحلیل تفاضلی توان روی FPGA
۱۲۷	۵-۷-۱ حمله‌ی DPA مبتنی بر ضریب همبستگی
۱۲۹	۵-۷-۱-۱ حملات DPA با استفاده از مدل وزن همینگ
۱۳۱	۵-۷-۱-۲ حملات DPA با استفاده از مدل فاصله‌ی همینگ
۱۳۱	۵-۷-۲ دیگر روش‌های آماری برای تعیین رابطه‌ی بین ستون‌های دو ماتریس
۱۳۲	۵-۷-۲-۱ اختلاف میانگین‌ها
۱۳۴	۵-۷-۲-۲ فاصله‌ی میانگین‌ها
۱۳۴	۵-۸ حمله‌ی DPA با استفاده از ترکیب روش‌های آماری
۱۳۷	فصل ششم: نتیجه‌گیری و پیشنهادات
۱۳۹	۶-۱ مقدمه
۱۳۹	۶-۲ نتایج کلی پروژه
۱۴۱	۶-۳ پیشنهادات
۱۴۴	لیست مقالات ارائه شده
۱۴۵	منابع و مراجع

شکل ۱-۱. مدل حمله به سیستم‌های رمزنگاری کلیدمتقارن.	۴
شکل ۲-۱. یک دور از ساختار Feistel کلاسیک	۱۳
شکل ۲-۲. مثالی از ترانهش	۲۲
شکل ۲-۳. مثالی از یک تبدیل آجرچین [۷].	۲۳
شکل ۲-۴. تبدیل بولی تکراری.	۲۴
شکل ۲-۵. شمای ساده‌ای از سیستم رمزنگاری قالبی کلید متقارن [۱].	۲۵
شکل ۲-۶. سیستم رمزنگاری قالبی تکراری با ۳ دور [۷].	۲۶
شکل ۲-۷. شمای دو دور از Rijndael [۷].	۲۷
شکل ۲-۸. نمایش State و آرایه‌ی کلید در شروع الگوریتم Rijndael.	۳۰
شکل ۲-۹. ساختار رمزنگاری در Rijndael [۱].	۳۱
شکل ۲-۱۰. ساختار رمزنگاری و رمزگشایی در Rijndael و تشابه آنها با یکدیگر.	۳۱
شکل ۲-۱۱. اعمال S-Box بر روی هر بایت State در تبدیل SubBytes [۱].	۳۲
شکل ۲-۱۲. جدول S-Box در الگوریتم Rijndael [۱].	۳۲
شکل ۲-۱۳. بایت‌های State در ShiftRows به‌طور حلقوی شیفت داده می‌شوند [۱].	۳۴
شکل ۲-۱۴. چگونگی عملکرد تبدیل Mix-Columns روی ستون‌های State [۱].	۳۵
شکل ۲-۱۵. در Add-Round Key کلیدهای هر دور با ماتریس State بایت به بایت XOR می‌شوند [۱].	۳۵
شکل ۲-۱۶. الگوریتم Key Expansion به‌صورت کدهای pseudo-C [۷].	۳۷
شکل ۲-۱۷. نمایش شباهت ساختار رمزنگاری و رمزگشایی در Rijndael [۱].	۳۸
شکل ۳-۱. شمای کلی معماری FPGA همراه با بلوک‌های اساسی سازنده‌ی آن [۱۹].	۴۳
شکل ۳-۲. یک مالتی پلکسر با دو ورودی (چپ) و یک فلیپ‌فلاپ (راست).	۴۴
شکل ۳-۳. بلوک منطقی پایه در Xilinx Spartan-II [®] [۱۲].	۴۴
شکل ۳-۴. Xilinx Spartan-II [®] Slice [۱۲].	۴۶
شکل ۳-۵. نحوه‌ی قرارگیری مالتی پلکسرهای F5 و F6 در CLB [۱۲].	۴۶
شکل ۳-۶. ساختار یک سلول SRAM [۱].	۴۷
شکل ۳-۷. شمای کلی معماری حلقه. دور با p و کلید با β نشان داده شده‌است [۱۹].	۵۱
شکل ۳-۸. شمای کلی ساختار معماری Unrolled [۱۹].	۵۲
شکل ۳-۹. پیاده‌سازی بلوک 'XTime' با استفاده از گیت XOR [۱۶].	۵۶
شکل ۳-۱۰. تحقق مؤثر تبدیل Mix-Columns [۲۷].	۵۷
شکل ۳-۱۱. تحقق مؤثر تبدیل Inv Mix-Columns [۲۷].	۵۹
شکل ۳-۱۲. پیاده‌سازی تبدیل Sub-Bytes [۱۹].	۶۲
شکل ۳-۱۳. پیاده‌سازی بلوک‌های شکل ۳-۱۲ (a) ضرب‌کننده در $GF(2^4)$; (b) ضرب‌کننده در $GF(2^2)$; (c) مربع‌کننده در $GF(2^4)$; (d) و (e) ضرب‌کننده‌های ثابت [۱۹].	۶۳
شکل ۳-۱۴. پیاده‌سازی محاسبه‌ی عنصر معکوس ضربی روی $GF(2^4)$ با روش square and multiply [۲۷].	۶۳
شکل ۳-۱۵. پیاده‌سازی محاسبه‌ی عنصر معکوس ضربی روی $GF(24)$ با روش multiple decomposition approach	۶۴
شکل ۳-۱۶. بلوک دیاگرام تبدیل Inv Sub-Bytes	۶۵

شکل ۱۷-۳. معماری Key Expansion مناسب برای ساختار sub pipelined الگوریتم AES ۱۲۸ بیتی [۲۷].	۶۷
شکل ۱-۴. دیاگرام نحوه‌ی انجام حمله‌ی تحلیل زمانی بر ضد پیاده‌سازی یک سیستم نوعی رمزنگاری.	۷۸
شکل ۲-۴. القاء خطا در بایت دوازدهم از کلید دور نهم [۵۵].	۸۳
شکل ۳-۴. نمایش نحوه‌ی اندازه‌گیری انرژی مصرفی یک کارت هوشمند.	۸۵
شکل ۴-۴. توان مصرفی یک کارت هوشمند هنگام انجام امضای دیجیتال [۴۱].	۸۵
شکل ۵-۴. مشاهده‌ی توان مصرفی یک کارت هوشمند نوعی هنگام انجام عملیات DES [۴۱].	۸۶
شکل ۶-۴. تابع f در الگوریتم DES.	۸۹
شکل ۷-۴. موفقیت‌آمیز بودن حمله‌ی DPA و پدیدار شدن همبستگی در مشاهده‌ی تفاضلی توان [۴۱].	۹۰
شکل ۱-۵. زمان‌بندی سیگنال‌های Reset, پالس ساعت و متن رمزشده‌ی خروجی.	۹۸
شکل ۲-۵. مصرف توان مدارهای CMOS.	۱۰۱
شکل ۳-۵. مدار یک معکوس‌کننده‌ی CMOS.	۱۰۲
شکل ۴-۵. بلوک دیاگرام تنظیمات آزمایشگاهی برای پیاده‌سازی حمله‌ی تحلیل توان.	۱۰۸
شکل ۵-۵. تصویری از اعمال تنظیمات آزمایشگاهی برای حمله به سیستم رمزنگاری روی FPGA.	۱۱۵
شکل ۶-۵. نمودار توان مصرفی FPGA حین اجرای الگوریتم رمزنگاری Rijndael.	۱۱۶
شکل ۷-۵. بلوک دیاگرامی از مراحل ۳ تا ۵ حمله‌ی DPA.	۱۲۴
شکل ۸-۵. نمونه‌برداری توان مصرفی FPGA حین اجرای الگوریتم AES توسط اسیلوسکوپ.	۱۲۶
شکل ۹-۵. نمودار همبستگی بین ستون‌های ماتریس H و T برای حدس صحیح از کلید.	۱۲۹
شکل ۱۰-۵. نمودار تفاضل نمونه‌های توان مصرفی برای حدس صحیح از زیرکلید.	۱۳۰
شکل ۱۱-۵. نمودار تفاضل نمونه‌های توان مصرفی برای یک حدس غلط از زیرکلید.	۱۳۰
شکل ۱۲-۵. بازبایی زیرکلید صحیح با استفاده از یک حمله‌ی ترکیبی که با مقادیر واقعی اندازه‌گیری شده‌است.	۱۳۵
شکل ۱۳-۵. بازبایی زیرکلید صحیح 0x4F با استفاده از روش مذکور.	۱۳۶

- جدول ۳-۱. مقایسه‌ی ویژگی‌های FPGA های خانواده‌ی Xilinx Spartan-II [۱۳]. ۴۷
- جدول ۳-۲. مقایسه‌ی برخی نتایج گزارش شده از پیاده‌سازی Rijndael روی FPGA ۵۴
- جدول ۳-۳. تعداد گیت مورد نیاز برای پیاده‌سازی و گیت‌های موجود در مسیر بحرانی سه روش محاسبه‌ی معکوس عناصر در $GF(2^4)$ ۶۵
- جدول ۵-۱. مقایسه‌ی برخی نتایج پیاده‌سازی Rijndael روی FPGA ۱۰۰
- جدول ۵-۲. حالت‌های گذار خروجی یک سلول CMOS و توان مصرفی متناظر با آنها. ۱۰۳

AES	Advanced Encryption Standard Algorithm
ASIC	Application Specific Integrated Circuit
CBC	Cipher Block Chaining Mode
CFB	Cipher Feedback Mode
CLB	Configurable Logic Block
CMOS	Complementary Metal Oxide Semiconductor
DES	Data Encryption Standard Algorithm
DFA	Differential Fault Analysis Attack
DPA	Differential Power Analysis Attack
DSA	Digital Signature Algorithm
DSP	Digital Signal Processor
ECB	Electronic Code Book Mode
EEPROM	Electrically Erasable Programmable Reed-Only Memory
FIPS	Federal Information Processing Standard
FPGA	Field Programmable Gate Array
GF	Galois Field
HDL	Hardware Description Language
IPsec	Internet Protocol Security
LUT	Look-Up Table
MAC	Message Authentication Codes
NIST	U. S. National Institute of Standard and Technology
OFB	Output Feedback Mode
RSA	Rivest-Shamir-Adleman Cryptosystem
S-Box	Substitution Box
SCA	Side-Channel Attack
SPA	Simple Power Analysis Attack
SRAM	Static Random Access Memory
VLSI	Very Large Scale Integrated Circuit
XOR	Exclusive OR

فصل اول:

مقدمه

همزمان با پیشرفت و فراگیری روزافزون مخابرات دیجیتال و افزایش حجم مبادله‌ی اطلاعات از طریق شبکه‌های متنوع مخابراتی داده نیاز به استانداردها و الگوریتم‌های مخابراتی با کارایی بالا از جمله ضروریات دنیای امروز محسوب می‌شود. یکی از وظایف مهم صاحب‌نظران امروز دنیای ارتباطات طراحی الگوریتم‌هایی با قابلیت و انعطاف‌پذیری بالاست که بتواند پاسخگوی تقاضا برای شبکه‌های مخابراتی پرسرعت، پرضرفیت، کم‌حجم، کم‌هزینه و در عین حال امن باشد. بدون شک امنیت داده‌های مبادله‌شده و رمزکردن آنها از مهم‌ترین مسائل در این حوزه به‌شمار می‌آید. تأمین امنیت در مبادلات مالی و بانکی و نیز ارتباطات نظامی نمونه‌هایی از اهمیت موضوع است. رمزشکن‌ها همیشه و همه جا در کمین اطلاعات مالی یا نظامی هستند که فاش شدن آنها ممکن است عواقب جبران‌ناپذیر و وخیمی به دنبال داشته باشد. معمولاً شخص کنجکاو یا رمزشکن گاه سعی می‌کند اطلاعات مبادله‌شده را به دست بیاورد و محرمانگی^۱ آن را از بین ببرد. گاهی سعی در تغییر محتوای اطلاعات و خدشه‌دار کردن اصالت^۲ آن دارد و بعضاً سعی می‌کند تا خود را به‌جای شخص دیگری معرفی کند و به مطالب مورد علاقه‌اش پی ببرد^۳. رمزنگاری^۴ دانش به‌حداقل رساندن مخاطراتی است که ممکن است از این حیث متوجه ما بشود.

یکی از روش‌های رمزکردن اطلاعات، رمزنگاری بلوکی کلیدمتقارن^۵ است. از جمله مزایای این روش سرعت، بازدهی و فراهم آوردن امنیت بالاست و واژه‌ی کلیدمتقارن به‌معنای یکسان بودن کلید رمزنگاری بین فرستنده و گیرنده است. پیشرفت سریع میکروالکترونیک به این روند سرعت بیشتری بخشیده و مهندسان را قادر ساخته تا این الگوریتم‌ها را بر روی تراشه‌هایی سریع، اما فشرده و کم-مصرف پیاده‌سازی کنند. استفاده از تراشه‌های رمزنگاری از جنبه‌های سرعت و قابل‌اعتماد بودن بر رمزکردن نرم‌افزاری اطلاعات مزیت دارد زیرا رمزشکن نمی‌تواند به آسانی اطلاعات موجود در آن را بخواند یا عوض کند. البته برخلاف بسیاری از مدارات VLSI موجود، طراحی و پیاده‌سازی چنین

¹ Confidentiality

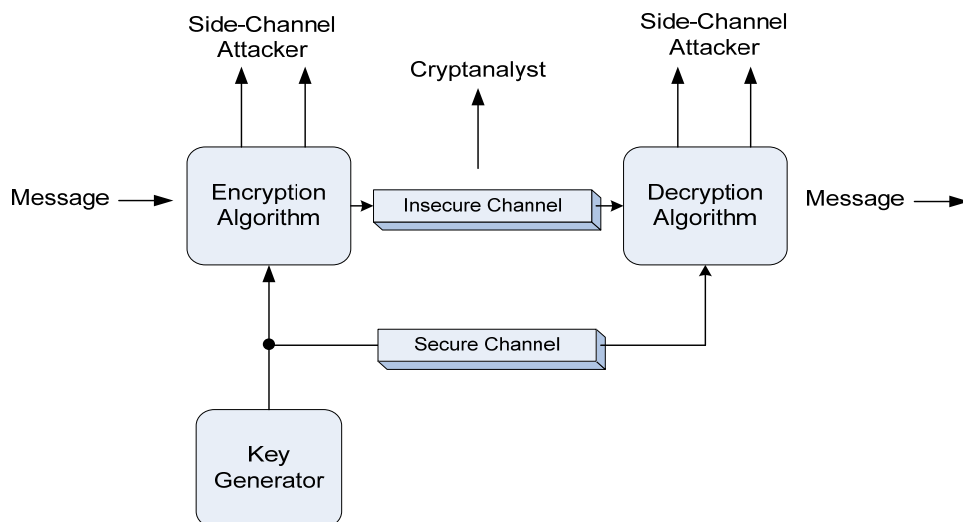
² Integrity

³ Entity Authentication

⁴ Cryptology

⁵ Symmetric Key Block Ciphering

تراشه‌هایی از جمله مقولاتی است که هرگز نمی‌تواند توسط یک غیرخودی انجام بگیرد. احتمال آنکه یک غریبه دریچه‌هایی^۱ را هنگام طراحی یا پیاده‌سازی در تراشه تعبیه کند که بتواند در موقع لزوم به اطلاعات آن دسترسی پیدا کند مانع از اعتماد به طراحی یا پیاده‌سازی توسط وی خواهد شد. علاوه بر آن طراحی چنین تراشه‌هایی باید به‌دست افرادی توانا و کاملاً آگاه به مقوله‌ی رمزنگاری انجام بگیرد زیرا چنانچه در موقع طراحی برخی مسائل و جوانب مهم در نظر گرفته نشود دشمن با استفاده از اطلاعات جانبی به‌دست آمده از تراشه‌ی در حال رمزنگاری قادر به پی‌بردن به کلید و شکستن رمز خواهد بود. چنانچه در شکل ۱-۱ نشان داده شده‌است ابتدا فرستنده و گیرنده از طریق یک کانال امن بر قسمت کوچکی از اطلاعات که معمولاً کلید رمزنگاری است توافق می‌کنند و سپس از طریق کانالی که ممکن است در معرض استراق سمع دشمن یا هر شخص دیگر قرار بگیرد اطلاعات را به‌صورت رمز شده با یکدیگر مبادله می‌کنند. رمزشکنی در این ساختار به دو دسته‌ی عمده تقسیم می‌شود: (۱) استراق سمع از کانال غیرامن و استفاده از تکنیک‌های خاص شکستن رمز که عمدتاً روش‌های نرم-افزاری، ریاضی و آماری هستند و (۲) استفاده از اطلاعاتی نظیر توان مصرفی یا زمان اجرای الگوریتم در یک تراشه‌ی رمزنگاری که به رمزشکنی سخت‌افزاری یا حملات پیاده‌سازی موسوم هستند.



شکل ۱-۱. مدل حمله به سیستم‌های رمزنگاری کلیدمقتارن.

^۱ Trapdoors

در رمزشکنی نرم‌افزاری یا کلاسیک تنها الگوریتم مورد تجزیه و تحلیل قرار می‌گیرد که رمزشکنی خطی و تفاضلی از جمله معروف‌ترین مثال‌های آن هستند.

دسته‌ی دوم حملات که برای اولین بار توسط P. Kocher در سال ۱۹۹۶ مطرح شدند به‌گونه‌ای کاملاً متفاوت از دسته‌ی اول سیستم را مورد تهاجم قرار می‌دهند. هنگامی که سخت‌افزار در حال پردازش و رمزکردن اطلاعات است می‌توان از اطلاعاتی نظیر توان مصرفی سخت‌افزار، تشعشعات الکترومغناطیسی آن یا زمان اجرای الگوریتم استفاده کرده و با کمک تحلیل‌های آماری و سایر تکنیک‌های رمزشکنی کلید رمزنگاری را به‌دست آورد. بسیاری از الگوریتم‌های معروف و شناخته‌شده نظیر RSA، DES، AES، El Gamal، Diffie-Hellman و... که در مقابل حملات کلاسیک کاملاً مقاوم هستند به‌راحتی با این‌گونه حملات موسوم به حملات کانال جانبی شکسته می‌شوند. در واقع توان مصرفی یا تشعشع الکترومغناطیسی یک ابزار رمزنگاری، یک کانال اطلاعاتی جانبی^۱ برای دشمن در کنار کانال واقعی عبور داده ایجاد می‌کند که بعضاً ممکن است اطلاعات فوق‌العاده مفیدی را در اختیار وی قرار دهد. بعضی از این حملات در زمان بسیار کوتاه و با هزینه‌ی اندک قادر به شکستن رمز هستند. مثلاً طبق گزارش منابع مربوطه حمله‌ی تحلیل ساده‌ی توان^۲ (SPA) در چند ثانیه رمز یک کارت هوشمند را می‌شکند و حمله‌ی تحلیل تفاضلی توان^۳ (DPA) ظرف چند ساعت قادر به شکستن رمز یک سیستم پیچیده است. چون این نوع حملات تحقق فیزیکی الگوریتم را هدف قرار می‌دهند و از بی‌دقتی یا ضعف هنگام پیاده‌سازی یک الگوریتم استفاده می‌کنند به آنها حملات پیاده‌سازی^۴ یا حملات سخت‌افزاری نیز گفته می‌شود. طراحان و پیاده‌سازان ابزارهای^۵ رمزنگاری چنانچه تنها به برآوردن معیارهایی چون سرعت، توان مصرفی یا کاهش ابعاد فیزیکی ابزار رمز پردازند و از مسأله‌ی محافظت از تراشه در مقابل حملات پیاده‌سازی غافل بمانند دچار خطایی بزرگ و در برخی موارد جبران‌ناپذیر شده‌اند.

¹ Side Channel

² Simple Power Analysis Attack

³ Differential Power Analysis Attack

⁴ Implementation Attack

⁵ modules

در این پایان‌نامه به پیاده‌سازی نوع خاصی از حملات سخت‌افزاری موسوم به حمله‌ی تحلیل تفاضلی توان (DPA) روی یکی از مهم‌ترین الگوریتم‌های رمزنگاری قالبی کلیدممتقارن معاصر یعنی الگوریتم Rijndael می‌پردازیم. علاوه بر این جهت سهولت در پیاده‌سازی حمله‌ی DPA ابتدا الگوریتم Rijndael را روی بستر FPGA با استفاده از زبان برنامه‌نویسی سخت‌افزار Verilog پیاده‌سازی کرده‌ایم. الگوریتم Rijndael یا AES الگوریتمی است که در سال ۲۰۰۰ به دنیا معرفی شد و از سوی NIST، ISO و IEEE به‌عنوان الگوریتم استاندارد رمزنگاری داده مورد پذیرش قرار گرفت. از سال ۲۰۰۰ تاکنون بحث طراحی سخت‌افزاری این الگوریتم و جنبه‌های مختلف آن از جمله مباحث جالب توجه و مهم از سوی صاحب‌نظران رمزنگاری و نیز طراحان VLSI بوده‌است و بحث در این مورد هنوز هم ادامه دارد. افزایش سرعت، کاهش توان مصرفی و حجم تراشه‌های AES و محافظت از آن در مقابل حملات سخت‌افزاری موضوع ده‌ها مقاله، پایان‌نامه‌ی دکتری و گزارش بوده‌است و راه‌کارهای خوبی نیز در این موارد ارائه شده اما هنوز هیچ کدام از طرح‌های ارائه شده نتوانسته تمام ملاک‌های کارآیی که به آنها اشاره شد را به‌طور توأم برآورده سازد و هم‌چنان آن را در معرض حملات سخت‌افزاری قرار می‌دهد. در این پایان‌نامه ضمن بررسی اجمالی ساختار ریاضی الگوریتم Rijndael و فلسفه‌ی طراحی آن، گلوگاه‌ها و معماری‌های مناسب برای پیاده‌سازی سخت‌افزاری آن را روی FPGA ارائه داده‌ایم. ضمن تشریح انواع حملات سخت‌افزاری و نحوه‌ی عملکرد آنها به شرح مختصر برخی از این حملات نسبت به پیاده‌سازی Rijndael پرداخته‌ایم. سپس برای آشنایی کامل با حمله‌ی تحلیل تفاضلی توان و توانایی بالای آن در شکستن الگوریتم‌های رمزنگاری یک پیاده‌سازی کاملاً عملی این حمله را روی Rijndael تشریح نموده‌ایم که در آن از دو روش برای پیاده‌سازی حمله و شکستن الگوریتم استفاده شده‌است. روش‌های مذکور مدل‌های آماری هستند که برای انجام تحلیل روی نمونه‌های توان مصرفی یک تراشه‌ی رمزنگاری مورد استفاده قرار می‌گیرند و از خاصیت شباهت نمونه‌ها و اختلاف بین میانگین‌های آنها بهره می‌برند. ایده‌ی حمله‌ی تحلیل توان خیلی ساده است اما پیاده‌سازی آن مشکلات مربوط به خود را دارد. مثلاً در تنظیمات آزمایشگاهی رعایت برخی نکات

خیلی مهم است و در صورتی که رعایت نشوند یک حمله‌ی ناموفق را به همراه خواهد داشت. هم-چنین انتخاب دقیق نمونه‌هایی که بایستی مورد پردازش قرار بگیرند اهمیت زیادی دارد. لازمه‌ی یک پیاده‌سازی موفقیت‌آمیز حمله روی الگوریتم رمزنگاری این است که تمام جزئیات الگوریتم و حمله مورد مطالعه قرار بگیرد و تمامی راه‌کارها و مشکلات پیاده‌سازی بررسی شود از این رو فصل‌بندی و ترتیب مطالب این پایان‌نامه را به صورت زیر آرایش داده‌ایم تا خواننده‌ی علاقه‌مند با مطالعه‌ی آن روند کلی این پروژه را بداند و با اطلاعات ترتیبی ارائه شده بتواند کاری مشابه با این پژوهش انجام دهد.

- در فصل دوم الگوریتم AES مورد بررسی قرار می‌گیرد. ضمن معرفی و برشمردن نقاط قوت الگوریتم Rijndael در این فصل و تشریح مبانی ریاضی و ساختار رمزنگاری و رمزگشایی به بررسی اجمالی دلایل انتخاب Rijndael به عنوان AES و نیز معرفی سایر نامزدهای AES و مشخصات آنها پرداخته‌ایم. به علاوه رمزنگاری بلوکی و حالت‌های مختلف عملکرد آن را نیز تشریح نموده‌ایم.

- فصل سوم معماری VLSI الگوریتم Rijndael را ارائه می‌کند. با توجه به اینکه یکی از اهداف ما در این پایان‌نامه پیاده‌سازی الگوریتم Rijndael روی FPGA است به تشریح مختصر ساختار FPGA، ملاک‌های کارایی و معماری‌های مناسب برای پیاده‌سازی الگوریتم Rijndael روی آن پرداخته‌ایم و نکاتی که باید هنگام انتخاب یک FPGA برای پیاده‌سازی این الگوریتم مد نظر داشت ذکر کرده‌ایم. سپس گلوگاه‌های پیاده‌سازی Rijndael را شرح داده و نتایج کارهای قبلی در مورد پیاده‌سازی این الگوریتم را ارائه داده‌ایم.

- آن دسته از حملات رمزشکنی موسوم به حملات پیاده‌سازی یا حملات فیزیکی که از نقاط ضعف پیاده‌سازی یک الگوریتم رمزنگاری برای شکستن رمز استفاده می‌کنند در فصل چهارم مورد بررسی قرار گرفته‌اند. در این فصل حملات مختلف به پیاده‌سازی الگوریتم‌هایی نظیر DES، RSA و به خصوص Rijndael را بررسی می‌کنیم و روش‌های مقابله با آنها را ارائه می‌کنیم.