

به نام یگانه معمار هستی

سمینار درس: DSP

نام استاد: جناب آقای دکتر مهدی اسلامی

نام دانشجو: مریم میرزایی فرد

بهار ۱۴۰۴

عنوان مقاله:

An FPGA-Based Open-Source Hardware-Software Framework for Side-Channel Security Research

Davide Zoni, Andrea Galimberti, Davide Galli

Abstract—Attacks based on side-channel analysis (SCA) pose a severe security threat to modern computing platforms, further exacerbated on IoT devices by their pervasiveness and handling of private and critical data. Designing SCA-resistant computing platforms requires a significant additional effort in the early stages of the IoT devices' life cycle, which is severely constrained by strict time-to-market deadlines and tight budgets. This manuscript

device's normal operation, many passive side-channel attacks, such as differential power analysis (DPA), remain undetectable by the system under attack.

While SCA attacks increasingly become serious security threats to IoT devices, and in particular to those that operate in public spaces and are accessible by anyone, the computing

نام نویسندگان:

Davide Zoni, Andrea Galimberti, Davide Galli

سال انتشار: 2025

رتبه علمی (Quartile/Q): Q1



معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی



معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی

این مقاله

یک کار نوآورانه در حوزه ی امنیت سخت افزار است.

(یک چارچوب سخت افزاری-نرم افزاری متن باز مبتنی بر FPGA برای
پژوهش در امنیت کانال جانبی)

تمرکز آن بر **مقابله** با یکی از پیچیده ترین تهدیدات امنیتی در دنیای
امروزی است:

حملات کانال جانبی (Side-Channel Attacks – SCA).

حملات SCA چیست؟

حملات کانال جانبی (Side-Channel Attacks - SCA)

- ♦ حملاتی هستند که نه به الگوریتم رمزنگاری بلکه به نحوه اجرای آن در سخت افزار حمله می کنند.
- ♦ از اطلاعات جانبی **نشت شده** هنگام اجرای واقعی استفاده می کنند.



در رمزنگاری سنتی، حمله کننده سعی می کند **ساختار** الگوریتم رمزنگاری را از نظر **ریاضی بشکند** یا رمز را با **امتحان کردن همه حالت ها** پیدا کند.

✓ این جور حملات به **منطق** یا **ریاضی پشت الگوریتم** حمله می کنند.

معرفی مقاله

طراحی و ساخت یک SoC

طراحی زیرساخت اشکال زدایی

توسعه ابزارهای نرم افزاری پشتیبان

پیاده سازی حملات کانال جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد امنیتی



معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی

در مقدمه، موارد زیر مطرح می شود:

۱- افزایش نقش دستگاه های IoT در جمع آوری و پردازش داده های حساس

۲- ناتوانی روش های رمزنگاری سنتی (مثل AES و RSA) در مقابله با حملات کانال جانبی (SCA)

۳- بی توجهی پلتفرم های رایج IoT به امنیت در برابر SCA به دلیل هزینه و پیچیدگی

۴- نیاز به یک چارچوب جامع سخت افزار-نرم افزار که هم اجرای حملات را آسان کند و هم طراحی مقابله گر ها را ممکن سازد

۵- مزایای RISC-V و کمبود پلتفرم های امنیت محور بر پایه آن

۶- تأکید بر امکان شناسایی دقیق منابع نشتی در سطح سیگنال سخت افزار

راه حل مقاله چیست؟

پژوهشگران این مقاله **چارچوبی به نام JARVIS** طراحی کرده‌اند
برای:

- اجرای واقعی برنامه‌های رمزنگاری روی FPGA
- انجام حملات SCA روی آن‌ها
- ارزیابی روش‌های دفاعی مختلف.

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

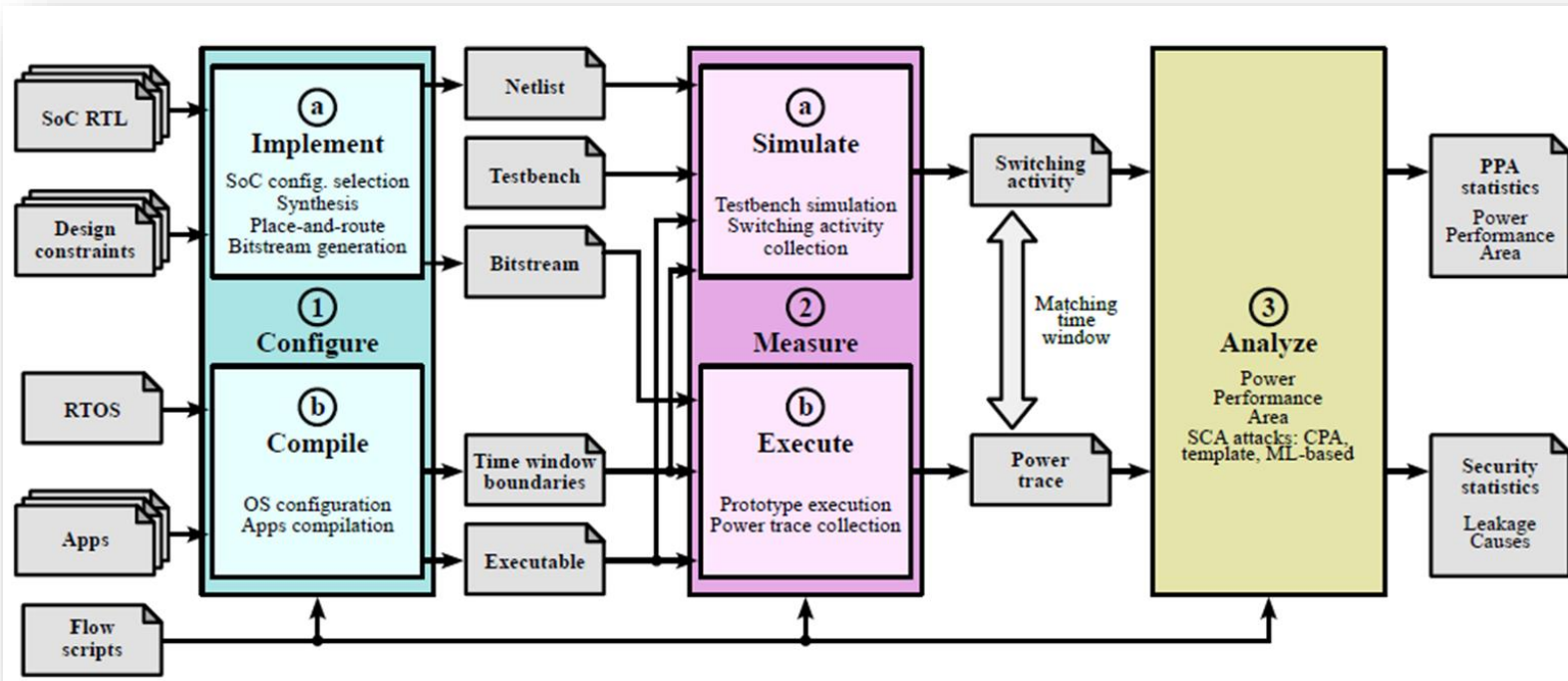
توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

این تصویر، نمای کلی از جریان کاری چارچوب **JARVIS** را نشان می‌دهد؛ سیستمی که برای **طراحی، اجرا و تحلیل** امنیت سخت‌افزار در برابر حملات کانال جانبی ساخته شده است.



این جریان کاری از سه مرحله‌ی اصلی تشکیل شده:
1-Configure (پیکربندی)
2-Measure (اندازه‌گیری)
3-Analyze (تحلیل)

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

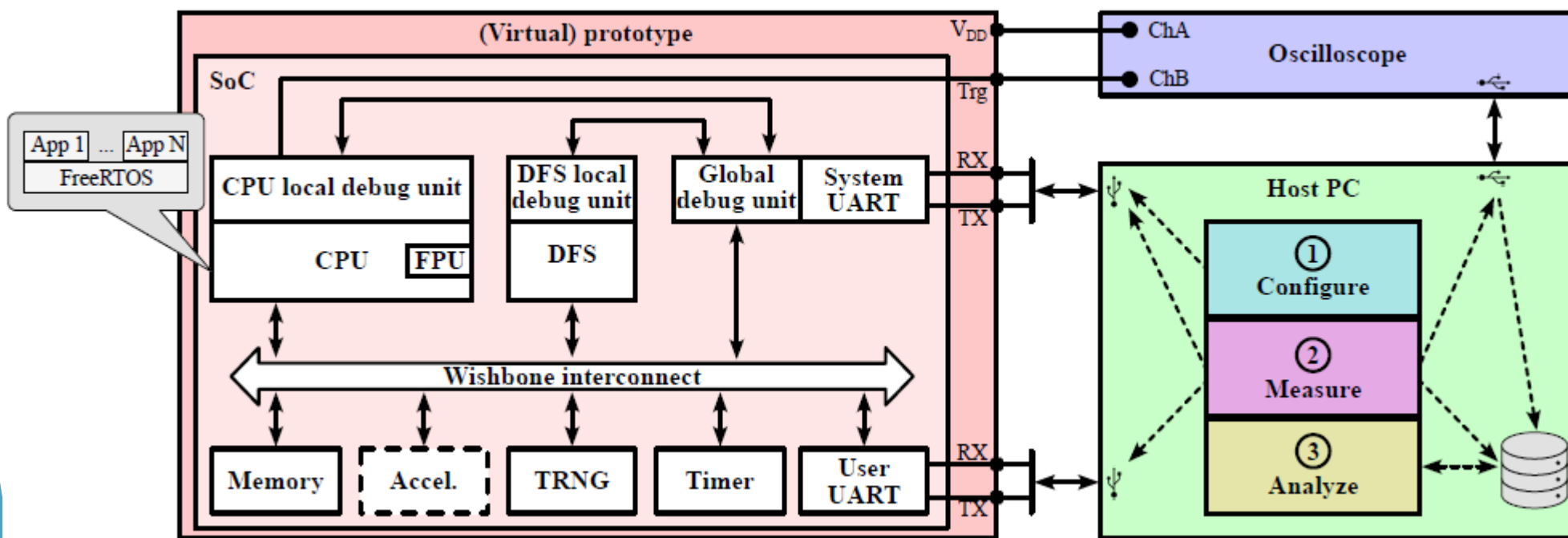
آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی

مهم ترین کارهای انجام شده در این پروژه به شرح زیر است:

- 1 طراحی و ساخت یک SoC (سیستم روی تراشه)
- 2 طراحی زیرساخت اشکال زدایی (Debug Infrastructure)
- 3 توسعه ابزارهای نرم افزاری پشتیبان
- 4 پیاده سازی حملات کانال جانبی
- 5 آزمایش روش های دفاعی (Countermeasures)
- 6 ارزیابی دقیق عملکرد امنیتی

این معماری، کل فرآیند اجرای برنامه، جمع آوری داده توان، و تحلیل امنیتی را به صورت یکپارچه مدیریت می کند.



معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی

1 طراحی و ساخت یک SoC (سیستم روی تراشه)

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی

در این مقاله، پژوهشگران یک SoC طراحی کرده اند که **هسته ی اصلی چارچوب JARVIS** را تشکیل می دهد. این SoC شامل یک **پردازنده ی ساده**، سبک و قابل برنامه نویسی با **معماری RISC-V 32 بیتی** است که به گونه ای طراحی شده تا هم برای **اجرای الگوریتم های رمزنگاری مناسب** باشد و هم **بتوان رفتار آن را به دقت بررسی کرد**.



معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی

به منظور افزایش کارایی و انعطاف پذیری سیستم، مجموعه ای از اجزای جانبی به صورت ماژولار و قابل پیکربندی در طراحی گنجانده شده اند.

از جمله این اجزای قابل پیکربندی:

1- TRNG

2- DFS actuator

3- تایمر داخلی

1- TRNG (True Random Number Generator)

- مولد سخت افزاری اعداد تصادفی که **وظیفه تولید داده‌های کاملاً تصادفی است.**
- نقش کلیدی در اجرای مقابله‌هایی مانند **chaffing** و تصادفی‌سازی رفتار سیستم ایفا می‌کند.

اجرای چند عملیات رمزنگاری **جعلی** و **غیرواقعی** در کنار **عملیات واقعی**، برای گمراه کردن حملات کانال جانبی (مثل تحلیل توان یا زمان اجرا)

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

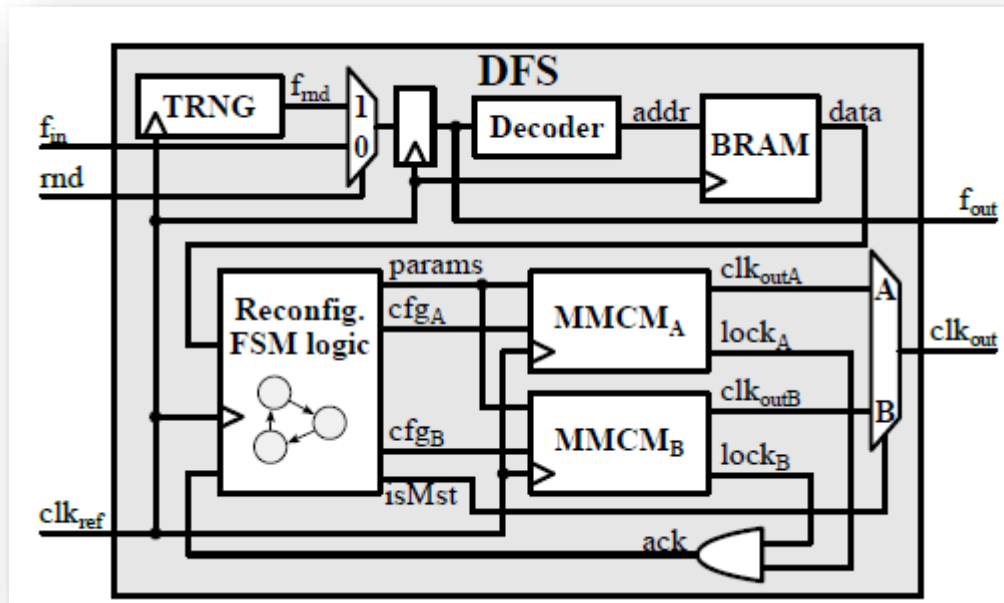
پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

2- DFS actuator (Dynamic Frequency Scaling)

- واحد کنترل فرکانس که **فرکانس کلاک پردازنده را به صورت پویا و تصادفی در زمان اجرا تغییر دهد.**
- این ویژگی کمک می کند تا حملاتی مانند تحلیل توان یا تحلیل زمان بندی دشوارتر و کم اثر شوند.



معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی



3- تایمر داخلی

با فراهم کردن قابلیت زمان‌بندی، امکان پشتیبانی از سیستم‌عامل‌های سبک‌وزن مانند FreeRTOS را فراهم می‌سازد و شرایط بررسی امنیت در محیط‌های چندوظیفگی را مهیا می‌کند.

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

2 طراحی زیرساخت اشکال زدایی (Debug Infrastructure)

به پژوهشگران امکان می دهد اجرای برنامه ها را به صورت کامل تحت کنترل داشته باشند و هم زمان بتوانند فعالیت های سخت افزاری را بررسی کنند.

در معماری سیستم، ماژول های اشکال زدایی محلی و سراسری طراحی شده اند که هر کدام وظایف خاصی را بر عهده دارند.

✓ این زیرساخت باعث می شود محقق بتواند با دقت بالا بفهمد چه زمانی و در کدام قسمت از اجرای برنامه، نشت اطلاعات رخ می دهد.

چیزی که برای مطالعه ی حملات کانال جانبی کاملاً حیاتی است.



معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی



این زیرساخت امکاناتی مانند:

A. تعیین نقاط توقف اجرای برنامه (breakpoint) تا بتوان رفتار سیستم را بررسی کرد

B. فعال سازی نقاط اندازه گیری توان (triggerpoints) دقیقاً همزمان با اجرای بخش خاصی از کد انجام می شود.

C. هماهنگ سازی دقیق بین اجرای نرم افزار و ثبت مصرف توان

معرفی مقاله

طراحی و ساخت یک
SoC

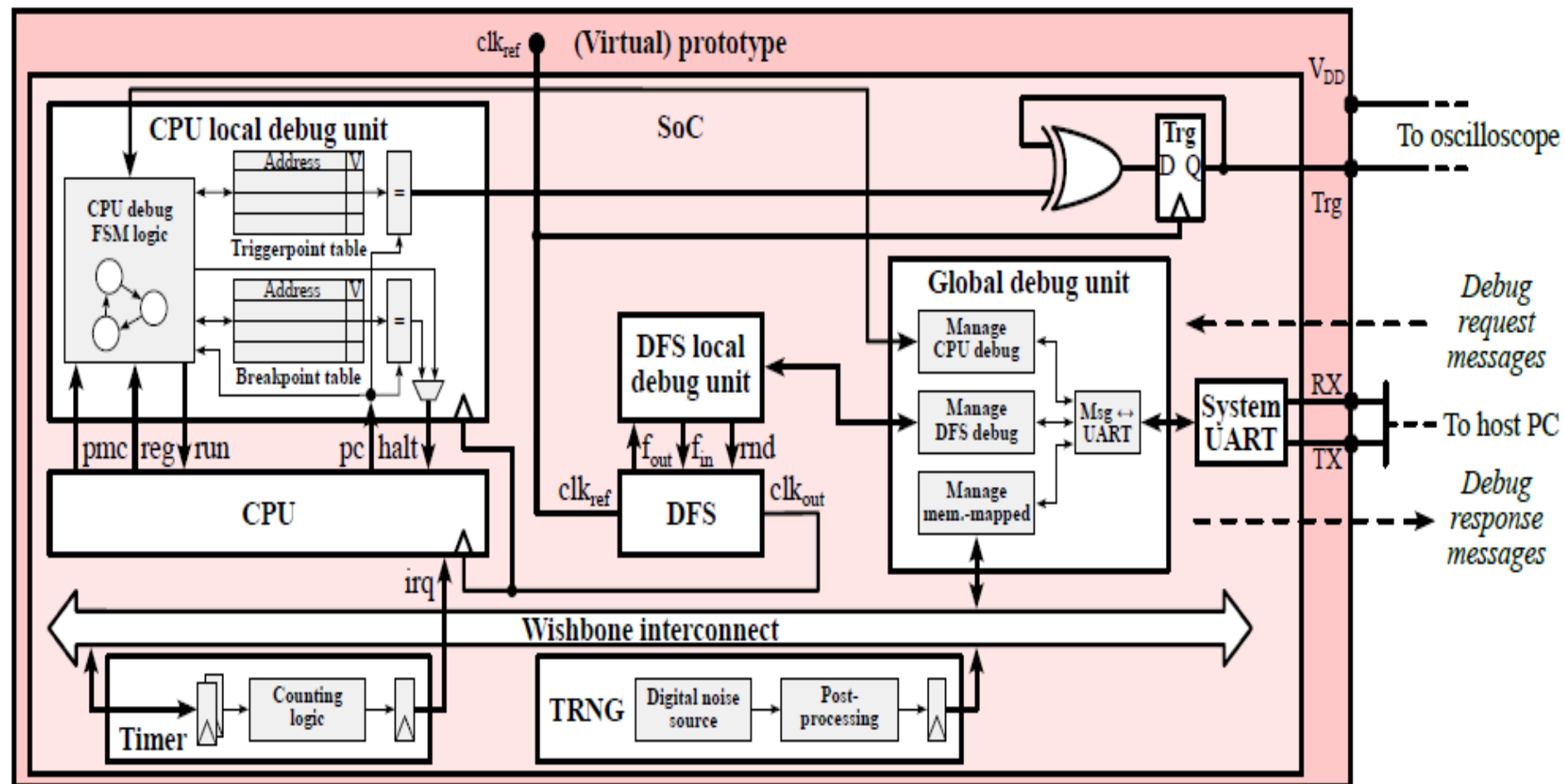
طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی



این تصویر معماری داخلی زیرساخت اشکال زدایی (Debug Infrastructure) در چارچوب JARVIS را نشان می‌دهد که داخل SoC پیاده‌سازی شده است.

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی



3 توسعه ابزارهای نرم‌افزاری پشتیبان

ابزارهایی برای:

- پیکربندی SoC
- اجرای برنامه‌ها
- ثبت ردپای توان
- تحلیل نتایج حملات
- همچنین، پشتیبانی از اجرای سیستم عامل FreeRTOS روی SoC برای تست‌های چندوظیفگی

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی



4 پیاده سازی حملات کانال جانبی

اجرای حملات مختلف برای ارزیابی امنیت سیستم:

- CPA (تحلیل همبستگی توان)
- Template Attack (حمله ی پرو فایل دار)
- CNN (حمله یادگیری عمیق)

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیر ساخت
اشکال زدایی

توسعه ابزارهای
نرم افزاری پشتیبان

پیاده سازی حملات کانال
جانبی

آزمایش روش های دفاعی

ارزیابی دقیق عملکرد
امنیتی

5 آزمایش روش‌های دفاعی (Countermeasures)

سه نوع مقابله پیاده‌سازی و آزمایش شد:

- **Morphing**: تغییر شکل کد
 - **Chaffing**: اجرای رمزنگاری‌های جعلی
 - **DFS**: تغییر تصادفی فرکانس کلاک
- کارایی هر روش بررسی شد و نتایج تجربی به دست آمد.

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی



6 ارزیابی دقیق عملکرد امنیتی

- آزمایش روی الگوریتم‌های رمزنگاری مختلف مثل AES، Clefia، Seed، Camellia
- بررسی اینکه کدام حمله موفق است و کدام روش دفاعی مؤثرتر عمل می‌کند.

معرفی مقاله

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

ویژگی‌ها	AES	Camellia	SEED	Clefia
طراح AU	(آمریکا) NIST	(ژاپن) NTT & Mitsubishi	(کره جنوبی) KISA	(ژاپن) Sony
نوع رمزنگاری	بلوکی (Block Cipher)	بلوکی	بلوکی	بلوکی
اندازه کلید	بیت 128, 192, 256	بیت 128, 192, 256	بیت 128	بیت 128, 192, 256
اندازه بلوک	بیت 128	بیت 128	بیت 128	بیت 128
کاربرد اصلی	جهانی – استاندارد صنعتی	سیستم‌ها	دولت و بانکداری کره	سیستم‌های سبک‌وزن و IoT
سرعت و عملکرد	بسیار سریع	مشابه AES	متوسط	کم‌قدرت
سطح امنیت	جهانی	بالا (تأیید ISO/IEC)	بالا (محلی)	بالا (بهینه‌شده برای حملات SCA)
جهانی	بسیار زیاد	متوسط تا زیاد	محدود (بیشتر در کره جنوبی)	محدود (در IoT و سیستم‌های خاص)



نتایج حاصل از مقاله به طور مشخص از آزمایش‌های تجربی
روی چارچوب JARVIS به دست آمده و به دو دسته کلی
تقسیم می‌شوند:

- 1- نتایج مربوط به حملات کانال جانبی (SCA)
- 2- نتایج مربوط به مقابله‌ها (Countermeasures)

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

نتیجه

1- نتایج مربوط به حملات کانال جانبی (SCA)

هر سه حمله‌ی CPA (تحلیل همبستگی توان)، Template Attack (حمله‌ی پروفایل دار) و CNN (حمله مبتنی بر یادگیری عمیق) توانستند **در نبود مقابله‌های امنیتی**، الگوریتم AES را به راحتی **بشکنند**. حمله‌ی CNN قدرت چشم‌گیری داشت و با **۸ رد پای توان (power traces)**، کلید رمزنگاری AES را به درستی شناسایی کند. این الگو برای سایر الگوریتم‌های رمزنگاری مورد آزمایش (شامل Camellia، Clefia و SEED) نیز تکرار شد؛ نشان‌دهنده‌ی آسیب‌پذیری مشابه در آن‌ها در حالت بدون محافظت.

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

نتیجه

2-نتایج مربوط به مقابله‌ها (Countermeasures)

سه روش مقابله‌ای بررسی شدند:
DFS (Dynamic Frequency Scaling)، **Chaffing** (اجرای
رمزنگاری‌های جعلی)، **Morphing** (تغییر ساختار کدها)

نتایج تجربی نشان داد:
DFS و **Chaffing** عملکرد بسیار خوبی داشتند
و جلوی موفقیت حمله‌های پیشرفته مثل CNN را
گرفتند.

در مقابل، روش **Morphing** نسبت به دو روش
دیگر کم‌اثرتر بود، به‌ویژه در برابر یادگیری عمیق.

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی

توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

نتیجه



الگوریتم‌های رمزنگاری رایج، **بدون محافظت**، در برابر حملات کانال جانبی **بسیار آسیب پذیرند**.

اما استفاده از مقابله‌های مبتنی بر تصادفی‌سازی اجرا مثل DFS و Chaffing می‌تواند به‌طور مؤثر از نشت اطلاعات جلوگیری کند چارچوب JARVI بستری دقیق برای ارزیابی این تهدیدها و دفاع‌ها فراهم کرده است.

نتیجه:

مقاله توانسته یک بستر عملیاتی، قابل پیاده‌سازی و آزمایش را برای پژوهش در حوزه‌ی امنیت کانال جانبی فراهم کند که کاملاً متن‌باز، مستند و توسعه‌پذیر است

طراحی و ساخت یک
SoC

طراحی زیرساخت
اشکال‌زدایی


توسعه ابزارهای
نرم‌افزاری پشتیبان

پیاده‌سازی حملات کانال
جانبی

آزمایش روش‌های دفاعی

ارزیابی دقیق عملکرد
امنیتی

نتیجه



با تشکر از توجه شما
همواره برقرار و سبز باشید...

Reference:

- A Deep Learning-Assisted Template Attack Against Dynamic Frequency Scaling Countermeasures
- A Deep-Learning Technique to Locate Cryptographic Operations in Side-Channel Traces
- Security Verification of the OpenTitan Hardware Root of Trust
- GVSoC: A Highly Configurable, Fast and Accurate Full-Platform Simulator for RISC-V based IoT Processors
- A Prototype-Based Framework to Design Scalable Heterogeneous SoCs with Fine-Grained DFS
- Hound: Locating Cryptographic Primitives in Desynchronized Side-Channel Traces Using Deep-Learning
- The Impact of Run-Time Variability on Side-Channel Attacks Targeting FPGAs
- Cost-effective fixed-point hardware support for RISC-V embedded systems
- Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process
- De-RISC: the First RISC-V Space-Grade Platform for Safety-Critical Systems
- HEROv2: Full-Stack Open-Source Research Platform for Heterogeneous Computing
- An FPU design template to optimize the accuracy-efficiency-area trade-off
- Agile SoC Development with Open ESP
- On the Effectiveness of True Random Number Generators Implemented on FPGAs
- Low-Power and High-Speed Dynamic CMOS Logic Circuit Design Techniques: A Comparative Study