Universidad
Carlos III de Madrid
www.uc3m.es

TESIS DOCTORAL

Evaluación de la Seguridad de Sistemas Embebidos ante Ataques EMA

Autor:

Roberto Martínez Bejarano

Director:

Juan Vázquez Martínez

DEPARTAMENTO DE TECNOLOGÍA ELECTRÓNICA

Leganés, Noviembre 2013

Universidad Carlos III de Madrid
www.uc3m.es

TESIS DOCTORAL

Evaluación de la Seguridad de Sistemas Embebidos ante Ataques EMA

Autor: **Roberto Martínez Bejarano**

Director: **Juan Vázquez Martínez**

Firma del Tribunal Calificador:

Firma

Presidente: (Nombre y apellidos)

Vocal: (Nombre y apellidos)

Secretario: (Nombre y apellidos)

Calificación:

Leganés, de de

A mi abuelo

Agradecimientos

En primer lugar debo agradecer a mi tutor, Juan Vázquez Martínez, su apoyo a la hora de desarrollar este proyecto que ve su fin ahora.

Muchos han sido los que a lo largo de estos 7 años han colaborado de manera directa o indirecta poniendo su granito de arena en este proyecto: Jose Manuel Algaba Alonso, Michael García Lorentz, Luis Mengibar Pozo, Raúl Sánchez Reíllo, Belén Fernández Saavedra, Raúl Alonso Moreno, Andrés Barrado Bautista, Jesús Rubio Serrano, Ángel Villanueva de la Hermosa, Celia López Ongil, Jesús Peña Rodríguez, Agustín Pulido Domínguez… A todos ellos gracias.

No puedo olvidar a todos mis compañeros del departamento de Tecnología Electrónica; Me han ayudado, me han aconsejado y han hecho más agradable esta etapa de mi vida.

También debo agradecer la ayuda al grupo de Sistemas Electrónicos de Potencia de la Universidad Carlos III de Madrid, por el soporte y material proporcionado así como a sus secretarios, ya que realizan una labor notable en la sombra, muchas veces sin reconocimiento.

Y por último, gracias a toda mi familia. A Bego y Ana por su paciencia, a mi hermana por sus imprescindibles correcciones, a mis padres por su apoyo… Todos han estado siempre ahí en los buenos y malos momentos, con su apoyo incondicional y su ánimo a seguir siempre adelante.

"Si planificas por un año, siembra trigo;
si planificas por una década, planta árboles;
si planificas por una vida, educa personas"

(Kwan - Tzu 300 a.C.)

Índice general

Resumen

Los sistemas embebidos de bajo consumo y alto rendimiento, cuya principal aplicación son los dispositivos portátiles tales como: teléfonos móviles, tabletas, consolas de juego, reproductores de música, lectores de libros etc. han experimentado un tremendo auge en los últimos años. Estos dispositivos, además de contener información confidencial (contraseñas, fotos, números de teléfono…) permiten, en su gran mayoría, realizar operaciones bajo redes inalámbricas poco seguras: como transacciones, envío de datos, acceso a cuentas personales etc. Por tanto, se hace imprescindible el análisis del nivel de seguridad alcanzado por estos dispositivos. Sin embargo, a la espera de futuros desarrollos de la estadística, todavía no existe un marco de evaluación de la seguridad totalmente satisfactorio e internacionalmente reconocido.

Así por primera vez en este trabajo se evalúa la seguridad relativa de varios microprocesadores representativos del mercado de aplicaciones embebidas de bajo consumo, comparando su respuesta ante un ataque por canal lateral electromagnético.

Los dispositivos seleccionados para su evaluación son:

- 8051 con arquitectura de 8 bits evolucionada (C8051F303 de Silicon Labs).
- ARM7TDMI-S de 32 bits (LPC2124 de NXP).
- Dos ARMCortexM3 de 32 bits nunca antes analizados ante ataques por canal lateral: con diseño de alto rendimiento (LPC1769 de NXP) y bajo consumo (STM32L152 de STMicroelectronics).

Para la realización de los experimentos se desarrolla un setup propio de medida, altamente automatizado, robusto ante vibraciones y con una capacidad de muestreo superior a lo publicado hasta ahora en la bibliografía.

También se propone una nueva métrica para comparar la respuesta de los dispositivos ante ataques por canal lateral, y que se apoya en la correlación estadística.

Uno de los elementos cruciales en un ataque por canal lateral electromagnético es el dispositivo o sonda de medida. Las publicaciones de autores que sugieren la utilización de algún tipo de sonda, no aportan datos concluyentes. Este trabajo compara de forma novedosa la respuesta de tres tipos de sondas: dos fabricadas y comercializadas por Electrometrics EM6995 y Langer MFA-R y una tercera fabricada *ad-hoc*, y manualmente. Como resultado se concluye que cualquier tipo de sonda es factible de ser usada en un ataque electromagnético, aunque son mejores aquellas de alta precisión como la MFA-R de cabeza milimétrica y preamplificador integrado, que sin embargo requieren una preparación y un setup más elaborado.

Como resultado final del estudio, se concluye que los dispositivos actuales ARM Cortex M3, ofrecen una seguridad inherente muy superior a la de otros microprocesadores de diseño menos elaborado, y en consecuencia es recomendable usarlos para aquellas aplicaciones cuyos requisitos de seguridad sean elevados.

Abstract

The low power and high performance embedded systems used in mobile devices like mobile phones, tablet computers, music readers, handheld game consoles, book readers… have achieved a great success in the last years. These devices contain confidence information (keys, photographs, telephone numbers…) and usually let us doing operations over unsafe wireless networks: banking transactions, sending data, accessing to personal accounts etc. In consequence, the analysis of the security level reached by these devices is indispensable. However, there isn't a satisfactory and internationally recognized methodology to assess security.

For first time, this work assesses the relative security of several representative low power embedded microprocessors, comparing their response against Electromagnetic Side Channel Attack.

The selected devices for this evaluation are:

- 8051 with new 8 bits architecture (Silicon Labs C8051F303).
- ARM7TDMI-S of 32 bits (NXP LPC2124).
- Two 32 bits ARMCortexM3 never before analysed against Side Channel Attacks: with high performance (NXP LPC1769) and low power specifications (STMicroelectronics STM32L152).

A measurement setup has been developed to carry out this study. It's highly automatized, robustly against vibrations and with a higher sampling rate than rest of setups showed in bibliography.

Also, a new metric is proposed. It allows to compare device response against correlation side channel attack using statistical correlation.

One of the essential elements of an electromagnetic side channel attack is the near field probe. The authors, whose studies suggest the use of some type of probe, do not include conclusion results. This work compares the response of three probes in a new way:

Electrometrics EM6995, Langer MFA-R and ones handmade. It concludes that any type of probe is useful in an electromagnetic attack, although the use of high precision probes is recommended. For example, the MFA-R with tiny head and integrated preamplifier. Nevertheless, it requires more training and a precise setup.

This study finds out that the updated devices ARM Cortex M3, have a very high security, higher than traditional ones. Therefore, the use of this type of devices in sensitive applications is advisable.

Lista de Acrónimos

3DES	(Triple *Data Encryption Standard*) Triple Estándar de Encriptación de Datos
AES	(*Advanced Encryption Standard*) Estándar Avanzado de Encriptación Americano
ASCII	(*American Standard Code for Information Interchange*) Código estadounidense estándar para el intercambio de información
C.C.	(*Correlation Coefficient*) Coeficiente de Correlación
CEMA	(*Correlation Electromagnetic Analysis*) Ataque por Análisis de Correlación Electromagnética
CMOS	(*Complementary Metal Oxide Semiconductor*) Tecnología de fabricación de chips basada en Semiconductor Complementario Óxido-Metálico
CPA	(*Correlation Power Analysis*) Ataque por Análisis de Correlación de Consumo
CPFA	(*Correlation Power Frequency Analysis*) Ataque por Análisis Frecuencial de Correlación de Consumo
CPU	(*Central Processing Unit*) Unidad Central de Proceso
DEMA	(*Differential Electromagnetic Analysis*) Ataque por Análisis Electromagnético Diferencial
DES	(*Data Encryption Standard*) Estándar de Encriptación de Datos
DFA	(*Differential Frequency Analysis*) Ataque por Análisis de Frecuencia Diferencial
DPA	(*Differential Power Analysis*) Ataque por Análisis de Consumo Diferencial

DSA	(*Differential Spectrogram Analysis*) Ataque por Análisis de Espectrograma Diferencial
EM	Electromagnetismo / Electromagnético
EMA	(*Electromagnetic Analysis*) Ataque por Análisis Electromagnético
EMC	(*Electromagnetic Compatibility*) Compatibilidad Electromagnética
EUA	(*Equipment Under Attack*) Equipo Bajo Ataque
EUT	(*Equipment Under Test*) Equipo Bajo Test
FFT	(*Fast Fourier Transform*) Transformada Rápida de Fourier
FPGA	(*Field Programmable Gate Array*) Matriz de puertas programable
HD	(*Hamming Distance*) Distancia de Hamming
HTTPS	Protocolo Seguro de Transferencia de Hipertexto
HW	(*Hamming Weight*) Peso de Hamming
IPv6	(*Internet Protocol*) Protocolo de Internet versión 6
MIPS	Millones de Instrucciones por Segundo
PA	(*Power Analysis*) Análisis Consumo Energético
PLL	(*Phase-Locked Loop*) Lazo de seguimiento de fase
SCA	(Side Channel Attack) Ataque por Canal Lateral
SEMA	(*Simple Electromagnetic Analysis*) Ataque por Análisis Electromagnético Simple
SMTP	(*Simple Mail Transfer Protocol*) Protocolo de Transferencia Simple de Correos Electrónicos
SNR	(*Signal-to-Noise Ratio*) Relación señal a ruido
SPA	(*Simple Power Analysis*) Ataque por Análisis de Consumo Simple
TCP	(*Transmission Control Protocol*) Protocolo de Control de Transmisión
TLS	(*Transport Layer Security*) Capa de Conexión Segura
WPA	(*Wi-Fi Protected Access*) Acceso Wi-Fi Protegido

Índice de figuras

Capítulo 1

INTRODUCCIÓN

La seguridad de los dispositivos electrónicos empotrados[1] ha sido siempre un tema que ha preocupado tanto a usuarios como a fabricantes. Sin embargo, en los últimos años su importancia ha crecido de manera exponencial. El hecho de que la mayoría de estos dispositivos dispongan de conexión a internet, les permite realizar tareas hasta hace poco exclusivamente reservadas a equipos informáticos, como enviar mensajes, almacenar datos en la red, acceder a la cuenta de ahorros del banco, realizar compras, etc. Es decir, operaciones que requieren el intercambio seguro de datos bajo entornos potencialmente peligrosos. Esto ha provocado que temas como la seguridad de los datos y las comunicaciones, muy presentes en ambientes informáticos desde la revolución de internet, se conviertan en una de las principales preocupaciones para usuarios y fabricantes de este tipo de dispositivos, incluso desde la etapa de diseño. Así, elementos como los sistemas de encriptación, tales como RSA, IDEA, DES, AES…, y protocolos y estándares de comunicación segura, como por ejemplo: HTTPS, TLS, WPA, SMTP, TCP, IPv6… han pasado a formar parte fundamental del entorno de estos equipos.

Tradicionalmente, la seguridad se ha definido en el contexto de las comunicaciones, donde dos entes intercambian información sensible a través de un medio potencialmente accesible. En este caso, la seguridad se fundamenta en tres pilares:

- Integridad de datos: Asegurar que la información no es modificada ilegítimamente.

[1] También llamados embebidos (embedded).

- Confidencialidad de datos: Proteger la información de terceras personas.
- Autentificación: Asegurar que la comunicación se produce entre las partes deseadas.

Además, se deben dotar otras funciones de seguridad, como permitir la limitación del acceso al dispositivo a determinadas personas mediante controles de acceso basados en claves o técnicas biométricas, asegurar la confidencialidad e integridad de la información almacenada en el dispositivo, certificar conexiones y accesos de red seguras, o asegurar tanto el correcto funcionamiento del dispositivo bajo entornos hostiles, como su resistencia a ser forzado [Koc'04].

Para tratar de conseguir la seguridad básica requerida en un dispositivo, habitualmente se utilizan tres tipos de algoritmos criptográficos:

- *Simétricos o de clave secreta*, que requieren la misma clave para la encriptación y desencriptación. Ej. RC4, IDEA, Blowfish, 3DES, AES…
- *Asimétricos o de clave pública*, que utilizan dos claves; una para cifrar los datos y otra para descifrar. Normalmente la clave de cifrado se hace pública, de esta forma cualquier persona puede enviar al poseedor de la clave privada un mensaje cifrado que sólo éste puede descifrar. Se evita así el problema de tener que intercambiar la clave. Ej. DSA, RSA, Curva elíptica, Diffie-Hellman…
- *Hash*, que permiten confirmar la integridad de los datos, generando un valor único de longitud fija para cada dato. Ej. SHA, MD5…

El sistema asimétrico tiene la ventaja de que la distribución de claves es más segura, en cambio sus algoritmos son más complejos, por lo que requieren una capacidad de procesamiento superior. Además, requieren claves de mayor tamaño y el mensaje cifrado ocupa más espacio que el original. Por ese motivo, se suelen utilizar los algoritmos simétricos para cifrar la información a transmitir y los asimétricos para las claves. No obstante, el rendimiento de los nuevos sistemas asimétricos basados en curvas elípticas ha mejorado notablemente [Kam'11].

Por tanto, se hace indispensable el uso y fabricación de dispositivos empotrados seguros. Además, factores específicos que normalmente les afectan, como la limitación de recursos, tanto a nivel energético como de almacenamiento y procesamiento, la posibilidad de operar en entornos poco seguros o el hecho de que sus comunicaciones se puedan realizar de forma inalámbrica, hacen que su seguridad deba analizarse de forma rigurosa. Asimismo, se ha demostrado que los algoritmos de encriptación presentan vulnerabilidades en función de su

implementación física, que pueden ser explotadas mediante los Ataques por Canal Lateral (SCA), que más adelante se analizarán en profundidad.

El tremendo auge experimentado en los últimos años por los SCA, ha propiciado que gobiernos y organismos de estandarización se interesen más por el tema de la seguridad. Es el caso del NIST (National Institute of Standards and Technologies), que en el año 2011 desarrolló un seminario para intentar establecer una metodología que permita determinar la vulnerabilidad de los dispositivos criptográficos a este tipo de ataques [Nist'11].

En el caso de que el dispositivo electrónico maneje información clasificada, el concepto de seguridad alcanza una importancia superlativa. Se habla entonces de *Tempest* [[CCN-Tem'06], [CCN-NTem'12]], de aplicación en todos los países de la OTAN, y que trata de evitar o reducir la posible filtración de información a través de las emisiones de energía radiadas involuntariamente por los dispositivos electrónicos que manejan información clasificada.

Durante los últimos años se ha dado a conocer públicamente la gran preocupación del gobierno de los Estados Unidos por el tema Tempest. Dicho país ha estado desarrollando normativas desde los años setenta para reducir los riesgos producidos por estas radiaciones. Estas normas han sido recientemente desclasificadas parcialmente y son accesibles para el gran público [NSA'03].

Por consiguiente, los fabricantes deben considerar la seguridad desde el diseño, como un factor más a considerar junto con el rendimiento, el precio o la potencia, no como un añadido [Koc'04]. Algunos fabricantes así lo han entendido, como es el caso de Maxim que ha desarrollado el MAX32590. Según el fabricante, supone el microcontrolador más seguro y resistente a los ataques del mercado. Éste incluye numerosos sistemas de seguridad integrados en un único chip: detector dinámico de fallos, memoria encriptada, soporte a RSA, DSA y ECDSA, control de temperatura y voltaje, aceleradores de hardware para AES, DES y SHA etc. Para más información véase [MaxDSM'13]. Igualmente Samsung, está dedicando importantes recursos a la investigación de la seguridad, creando infraestructuras de evaluación como el JHAS, proponiendo soluciones de seguridad como el CRYPTO IP etc. [Par'13].

1.1 Motivación

Esta es la primera Tesis Doctoral que se desarrolla en el campo de la seguridad de sistemas criptográficos y los campos electromagnéticos, dentro del Departamento de Tecnología Electrónica de la Universidad Carlos III de Madrid. Se abre así una nueva línea de investigación que amplía la diversidad de temas del departamento.

En 2006 C.H. Gebotys [Geb'06] ya recalcaba la importante necesidad de estudiar la seguridad de sistemas embebidos de bajo consumo orientados a aplicaciones sobre dispositivos portátiles inalámbricos como PDA's y teléfonos móviles. El hecho de que las tecnologías inalámbricas móviles se hayan hecho con un importante hueco en el mercado, llegando a desplazar en muchas tareas a los dispositivos informáticos tradicionales, hace imprescindible su estudio.

A pesar de ello, la popularidad de este tipo de dispositivos entre los investigadores no alcanza cotas elevadas, y pocos han sido los autores que se han dedicado a analizar y acreditar su nivel de vulnerabilidad ante un ataque no invasivo o por canal lateral. Existe pues, un importante vacío que esta tesis ha tratado de explorar.

Además, en la literatura no existe ningún estudio que detalle y compare el comportamiento de varios dispositivos ante los SCA. La mayoría de los autores se han dedicado a implementar ataques o contramedidas, y a su comprobación práctica sobre un único dispositivo.

1.2 Objetivos

El objetivo principal de este trabajo, es analizar la seguridad relativa de varios dispositivos embebidos de bajo consumo, y que son representativos de la oferta actual de microprocesadores/microcontroladores presentes en el mercado, ante los SCA.

Dado que todavía no está clara una metodología que analice de forma objetiva y fiable la vulnerabilidad de un sistema, se comparará la respuesta de los dispositivos ante un SCA y se propondrá una métrica o parámetro que permita su evaluación.

Para ello, se deberá desarrollar un setup de medida específico para su ejecución, con todas las herramientas necesarias: sondas, soportes, software de adquisición etc.

Aprovechando la oportunidad, se evaluará el comportamiento de varias sondas EM de campo cercano, uno de los elementos más críticos de la cadena de medida, y se estudiarán distintos aspectos importantes de los SCA, que la literatura solo contempla de manera somera.

1.3 Estructura de la Tesis

La tesis que aquí se propone, se ha descrito dividiendo el documento en ocho capítulos y tres anexos. Los dos primeros introducen inicialmente al lector en temas relacionados con la seguridad de dispositivos electrónicos, como son el criptoanálisis y los SCA, necesarios para el entendimiento del resto de capítulos, para a continuación centrarse en dos ataques concretos: por análisis electromagnético y consumo eléctrico. En el capítulo tres

se describe la metodología elegida para analizar la seguridad de los dispositivos embebidos bajo estudio. En el cuarto apartado se hace una descripción detallada del setup de medida desarrollado para llevar a cabo las medidas. Los resultados experimentales obtenidos son mostrados y analizados en los capítulos cinco, seis y siete. El último capítulo presenta a modo de resumen, las conclusiones más relevantes obtenidas, así como también, los trabajos no tratados que se consideran pendientes de realizar en un futuro. Por último, los anexos incluyen información menos relevante, pero de posible utilidad para el lector. Así el anexo uno se centra en el algoritmo de encriptación AES, el dos muestra los códigos implementados en los dispositivos analizados y el tres presenta un resumen de todos los resultados experimentales obtenidos.

La siguiente lista detalla la información contenida en cada capítulo y anexo que componen este documento:

Capítulo 2. Hace una pequeña introducción al criptoanálisis y los ataques por canal lateral, para centrarse en las variantes por análisis electromagnético y energético, base de este proyecto. Se revelan sus fundamentos y las diferentes técnicas aparecidas en la literatura.

Capítulo 3. Se explica el método seguido para evaluar la seguridad relativa de los dispositivos embebidos bajo análisis mediante los ataques por canal electromagnético, argumentando cada una de las decisiones realizadas y se describe cada uno de los dispositivos testeados.

Capítulo 4. Descripción completa del setup y procedimiento de medida típicos usados para llevar a cabo un ataque por canal lateral electromagnético, y a continuación lo compara con el utilizado en esta disertación. Detalla uno a uno los elementos que conforman el setup implementado, con especial hincapié en la sonda de medida.

Capítulo 5. Presenta los resultados experimentales obtenidos con cada uno de los dispositivos, conclusiones provisionales y algunos experimentos realizados para la puesta a punto y comprensión de los ataques.

Capítulo 6. Se analizan los resultados de forma global y se obtienen conclusiones a partir de ellos. El trabajo realizado es comparado con lo publicado en la literatura.

Capítulo 7. Describe otros resultados experimentales adicionales efectuados para el desarrollo de este trabajo.

Capítulo 8. Resume las principales conclusiones del estudio y detalla las líneas de investigación abiertas que se constituyen como posible trabajo futuro.

Anexo 1. Explicación en detalle del algoritmo de cifrado AES.

Anexo 2. Muestra los códigos implementados del estándar AES en los dispositivos analizados.

Anexo 3. Contiene todos los resultados experimentales relativos al estudio principal de este trabajo que evalúa la seguridad de los cuatro dispositivos embebidos.

Capítulo 2

INFORMACIÓN PRELIMINAR Y ESTUDIOS PREVIOS

En 1996, Kocher y otros autores [Koc'96] desvelaron públicamente que los algoritmos criptográficos podían presentar debilidades por el hecho de ejecutarse sobre un dispositivo físico, debido a la fuga de información a través de alguna magnitud física propia del equipo. Hasta ese momento, la criptografía era una ciencia prácticamente exclusiva de investigadores matemáticos. A partir del descubrimiento de Kocher, distintos profesionales como físicos, ingenieros, informáticos… se interesaron por el tema y pasaron a formar parte de tan distinguido grupo. La criptografía sufrió entonces un auténtico boom.

En este capítulo se hace una introducción general a la criptología y los Ataques por Canal Lateral (SCA), para a continuación centrarse en dos modelos íntimamente relacionados: los ataques que utilizan como canal lateral el consumo de energía y los basados en la radiación electromagnética emitida por el dispositivo. Por último, se estudian las diferentes técnicas disponibles en la literatura sobre dichos ataques.

2.1 El Criptoanálisis y los Ataques por Canal Lateral

El Criptoanálisis tiene su origen en las palabras Griegas kryptós "escondido" y analýein, "aflojado" o "desatado" y se define como el estudio de los posibles métodos para descifrar información encriptada sin tener acceso a la información secreta que es normalmente

necesaria para ello: típicamente una clave secreta. Aunque también se usa para referirse a cualquier intento de eludir la seguridad de algoritmos criptográficos y protocolos en general.

Un algoritmo criptográfico puede considerarse desde dos puntos de vista. Por un lado, según la visión clásica, puede tratarse como un objeto matemático abstracto o caja negra, es decir, una transformación parametrizada o no por una clave, cuyos únicos parámetros para su criptoanálisis son los datos de entrada y salida (Figura 2.1).

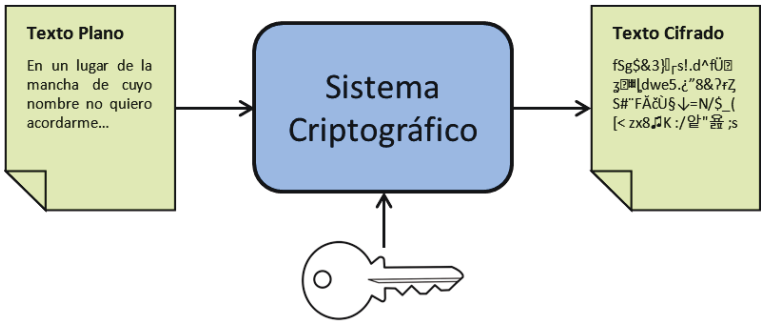


Figura 2.1: Modelo Criptográfico Clásico

Por otro lado, puede contemplarse como un objeto que tiene que implementarse en un programa que se ejecuta en un determinado procesador, en un entorno dado y que presenta características específicas propias de la implementación. Este es el punto de vista de la Seguridad Física, que se hizo público por primera vez a mediados de los años noventa [Koc'96] (Figura 2.2).

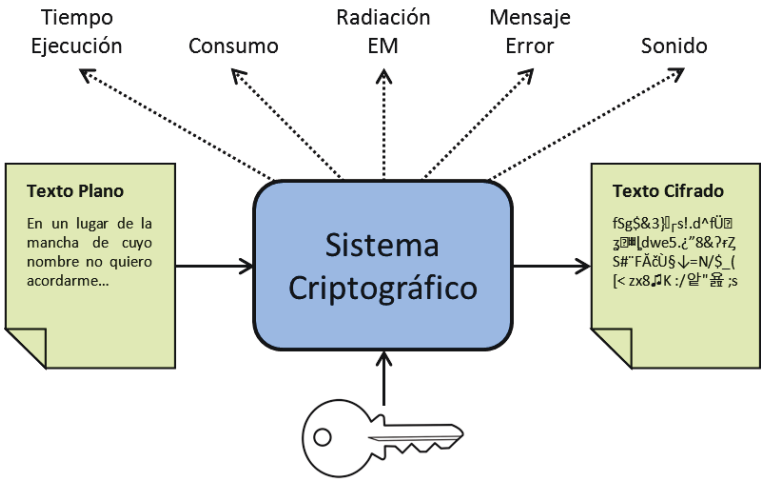


Figura 2.2: Modelo Criptográfico con Seguridad Física

La visión clásica que concibe que todos los ataques se realizarán tratando de romper las "complejas" primitivas criptográficas empleadas en dispositivos de seguridad, es poco realista. Se puede hacer una analogía muy interesante entre un potente algoritmo criptográfico y una puerta de alta seguridad en la parte delantera de una casa. Es poco probable que unos ladrones intenten, por todos los medios posibles, entrar en la casa a través de la puerta

blindada; En su lugar, intentarán acceder por alguna ventana, por una puerta trasera, cavando un foso, o robando la llave de acceso.

La seguridad, por tanto, se debe cuantificar no solo en términos de propiedades matemáticas del algoritmo criptográfico y tamaño de la clave, sino además en función de su implementación.

Los Ataques Físicos a dispositivos criptográficos, intentan explotar distintas vulnerabilidades causadas por la propia naturaleza física de los dispositivos empleados para el cifrado de la información. Esto implica que son muy específicos de la implementación realizada, pero con la gran ventaja de que pueden ser varios órdenes de magnitud más efectivos que un ataque convencional [[Kel'98], [Koe'05]].

La diversidad de los ataques físicos es muy grande. Se puede hacer una primera clasificación basada en tres parámetros:

a) Invasivo – No Invasivo: Los ataques invasivos requieren el desencapsulado del chip para tener acceso a los componentes internos. Mientras que un ataque no invasivo únicamente explota la información disponible externamente, tales como el tiempo de ejecución, el consumo de energía, la radiación EM emitida…

b) Activo – Pasivo: Los ataques activos someten al dispositivo a condiciones físicas extremas, para intentar forzar un estado erróneo en el dispositivo mientras éste se ejecuta. En el lado opuesto se encuentran los ataques pasivos, que únicamente observan el dispositivo durante el procesamiento sin tratar de provocar una alteración.

c) Simple – Diferencial: Un ataque simple analiza directamente una o varias medidas de información filtrada para intentar determinar la instrucción ejecutada y los datos utilizados. Por oposición, el ataque diferencial analiza numerosas medidas utilizando técnicas estadísticas para intentar explotar la correlación con los datos procesados. En este caso, el adversario no requiere un conocimiento previo de la implementación, como sí ocurre en el simple.

Por otro lado, los ataques se pueden organizar en tres grandes grupos:

Ataques por Sondeo

Consiste en abrir el dispositivo quitando su encapsulado para observar directamente su comportamiento: señales en los buses, celdas de memoria… utilizando tanto microscopios, como sondas microscópicas. Son, por tanto, Pasivo-Invasivo.

Ataques por Inducción de Fallos

Tratan de influenciar el comportamiento del dispositivo, sometiéndolo a condiciones de funcionamiento extremas para tratar de producir errores forzados que impliquen la fuga de información secreta. Se trata de un ataque Activo y puede ser Invasivo o no.

Ataques por Canal Lateral

Son un tipo de ataque no invasivo y pasivo, donde el adversario trata de explotar las fugas físicas de información observables durante el procesamiento. En función del parámetro físico analizado, se pueden clasificar a su vez en:

- **Ataques por Análisis de Tiempos**: Explotan el tiempo de ejecución del dispositivo criptográfico.
- **Ataques por Análisis de Consumo**: Observan el consumo eléctrico.
- **Ataques por Análisis Electromagnético**: Analizan el campo electromagnético (EM) emitido por el equipo.

La eficiencia, y en algunas ocasiones, la simplicidad [Scc'09] y el hecho de que puedan ser llevados a cabo con un equipo relativamente asequible, han contribuido a su popularidad, haciendo que este tipo de ataques representen un serio problema a la seguridad para la mayoría de dispositivos criptográficos hardware. Por tal motivo, la comunidad criptográfica se ha volcado en esta nueva disciplina, que a diferencia de la clásica usa mecanismos puramente físicos y no matemáticos, dedicando importantes recursos para su desarrollo.

La viabilidad de los SCA se debe a la no idealidad de los sistemas computacionales actuales, lo que provoca la existencia de correlación entre las medidas físicas tomadas durante el procesado de un algoritmo criptográfico y el estado interno del dispositivo, el cual está relacionado con la clave secreta. Esta correlación entre la información filtrada a través del canal lateral y la operación relacionada con la clave secreta, es lo que el SCA quiere descubrir.

2.2 Ataques por Análisis de Consumo y Electromagnético

Los SCA son posibles porque la mayoría de los sistemas electrónicos dejan escapar información observable, que está correlacionada con las instrucciones y datos que ejecutan.

Por ejemplo, los microprocesadores al realizar una computación consumen tiempo y energía, generan radiación EM, ruido sonoro [Sha'04], incluso disipan calor [Bro'09]; Y si disponen de monitor para mostrar la información, radian luz [[Lou'02], [Kuh'02]]. Existen, por tanto, gran cantidad de fuentes de fuga de información que pueden ser explotadas por atacantes. En general, cualquier magnitud física que sea accesible, pueda ser medida y esté relacionada de un modo u otro con la configuración interna o actividad de un dispositivo criptográfico, puede ser utilizada por un enemigo como fuente de información para un ataque de este tipo.

Dentro de los SCA, y atendiendo al número de referencias bibliográficas existentes, el que más interés ha despertado entre los investigadores es el que usa como fuente de información el consumo de energía: Ataque por Análisis de Consumo (PA: Power Analysis Attack). A pesar de ello, este trabajo se va a centrar en los ataques por Análisis Electromagnético, pero dado que consumo de energía y radiación electromagnética están íntimamente ligados, lo dicho para uno se puede extrapolar al otro en la mayoría de los casos.

Los sistemas criptográficos, por el hecho de consumir una energía para su funcionamiento, dejan escapar información relacionada con la instrucción y dato que están procesando. La realización de un ataque normalmente requiere relacionar los datos procesados con la energía consumida, por lo que suele ser necesaria la utilización de un modelo de consumo. Éste, se encarga de deducir el consumo a partir del dato que va a ser computado y constituye uno de los principales elementos de un ataque. La calidad de su deducción influye de forma determinante en su efectividad.

Cada sistema tiene un consumo característico, por lo que existen multitud de modelos de consumo que describen este comportamiento de forma más o menos precisa [[Pee'07], [Sta'06], [Bri'04], [Cha'02], [Li'10], [Gui'04], [Man'06]]. Dado que los atacantes normalmente tienen poca información acerca del dispositivo criptográfico a atacar, y mayor precisión implica mayor complejidad, se suelen usar modelos genéricos más sencillos, pero menos precisos. Es importante recalcar, que en un ataque PA no importa tanto el consumo absoluto, sino la diferencia relativa.

Así, en multitud de estudios realizados se establece que los sistemas criptográficos dejan escapar información relacionada con el Peso de Hamming[2] (HW) del dato que está siendo procesado [[Koc'99], [Mes'99], [Cor'00], [May'00]]. De forma que un dato procesado con un alto HW consumirá menos que uno con HW bajo. Sin embargo, últimamente la gran mayoría de investigadores asumen la existencia de una relación lineal entre el consumo y la

[2] Peso de Hamming (Hamming Weight): Número de bits que están a uno.

Distancia de Hamming[3] (HD) existente entre un estado y el siguiente [[Bri'04], [Kel'98], [Joy'05]]. Según [Joy'05], el modelo energético basado en el HW sólo es cierto para aquellos dispositivos con precarga lógica. Además el HW se puede considerar un caso especial del HD en el que el estado anterior o de referencia es cero:

$$HD(e_0, e_1) = HW(e_0 \oplus e_1) \tag{2.1}$$

$$HD(0, e_1) = HW(0 \oplus e_1) = HW(e_1)$$

Dado que la mayoría de los circuitos actuales están construidos usando puertas lógicas basadas en tecnología CMOS, veamos por qué existe fuga de información en éstos, es decir, por qué se produce correlación entre dato computado y energía disipada.

2.2.1 Consumo de Potencia en dispositivos CMOS

Los circuitos digitales actuales se construyen mayoritariamente a base de celdas lógicas básicas implementadas con transistores. Existen diversas tecnologías para construir estos bloques básicos, aunque en la actualidad la inmensa mayoría se fabrica con tecnología CMOS.

El inversor CMOS es el bloque más básico de esta tecnología. Dado que, normalmente, las celdas lógicas CMOS se diseñan a partir de estructuras complementarias similares, las disquisiciones hechas para este bloque serán representativas de la lógica digital CMOS.

Se puede decir que un inversor CMOS prácticamente sólo consume energía cuando se produce un cambio de su estado lógico tras el pulso de reloj. Es decir, su consumo depende casi de forma exclusiva del número de conmutaciones. Mantener un estado estático requiere un consumo prácticamente despreciable. Ese es el motivo por el cual, el consumo se incrementa aproximadamente de forma lineal con la frecuencia. Otros parámetros como la temperatura, la pendiente del ciclo de reloj, la tensión de alimentación, los glitches[4], el ciclo de trabajo o efectos de acoplamiento también influyen [Cha'99], aunque se suelen ignorar para tener un modelo de consumo lineal más sencillo.

Se puede considerar que las puertas CMOS tienen dos tipos de fuentes de disipación de energía distintas [Rab'03]:

[3] Distancia de Hamming (Hamming Distance): Número de bits que cambian de un estado anterior, también llamado de referencia, al siguiente. Es decir, indica el número de transiciones requeridas para pasar de un estado al siguiente.

[4] Fallos internos breves que se producen debido a la desincronización en la llegada de las señales de entrada a una celda combinacional.

- La primera, de origen estático, es debida a las pequeñas corrientes de fuga de los transistores (típicamente de 10nA a 10μA, no obstante esta corriente es función del tamaño, por lo que en los últimos años está aumentando significativamente debido al uso de estructuras cada vez más pequeñas).

- La segunda, de origen dinámico, se produce durante las conmutaciones debido a dos motivos:

  - La carga de las impedancias capacitivas: intrínsecas[5] y extrínsecas[6], representadas por el condensador C_L[7] en la Figura 2.3.

  - Las corrientes de cortocircuito producidas por la conducción simultánea durante un periodo de tiempo muy pequeño de los dos transistores nMOS y pMOS.

Si analizamos estos consumos de forma cuantitativa, en general se puede considerar, sin cometer un error considerable, que los circuitos CMOS únicamente consumen energía durante la conmutación, debido al consumo dinámico. En estado estático (transiciones 0→0 y 1→1), las celdas únicamente disipan corriente de fuga, mientras que en estado dinámico (transiciones 1→0 y 0→1), además, disipan las corrientes de carga, descarga y cortocircuito. Véase la Figura 2.3.

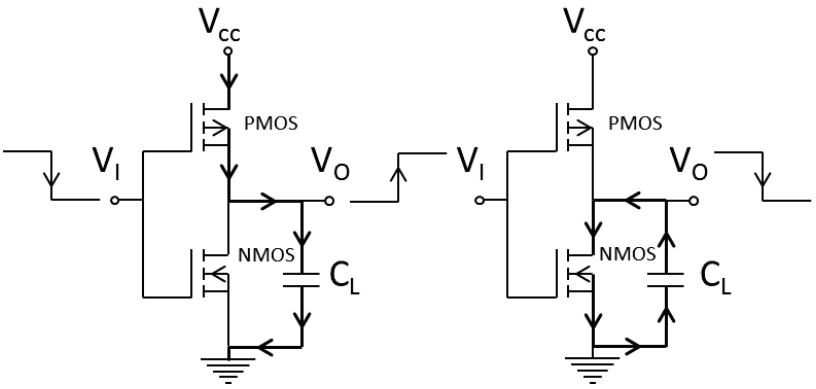


Figura 2.3: Corriente Carga Descarga inversor CMOS con transiciones 0-1 y 1-0

Si se mide el consumo de energía de un dispositivo CMOS mientras conmuta, el mayor pico de energía se produce durante la carga del condensador, es decir, durante la conmutación 0 → 1. Durante el evento 1 → 0 el condensador se descarga a través del

[5] Capacidades internas conectadas a la salida del bloque.
[6] Capacidades parásitas de los cables y capacidades de entrada.
[7] Se usa el modelo denominado C concentrada (lumped-C) para describir este consumo, el cual agrupa todas las capacidades presentes en un único condensador: C_L.

transistor NMOS, y la única corriente que se puede medir en la fuente de alimentación es la corriente de cortocircuito, que es menor. Si bien, se debe tener en cuenta, que la importancia relativa de las fuentes de disipación depende de la escala de integración de la tecnología usada [Sta'09a].

Para un chip CMOS típico estándar se cumple [Cor'00]:

$$P = P_{Estático} + P_{Din\_Corto} + P_{Din\_C_L} \cong 0.05 \cdot P + 0.15 \cdot P + 0.8 \cdot P$$

Este comportamiento del consumo dinámico dependiente del valor del dato es lo que origina las fugas de información y permite los ataques por Análisis de Consumo y Electromagnético.

Curiosamente este proceder, a priori simple y ampliamente admitido por la mayoría de los investigadores, no ha podido ser modelado hasta la fecha de forma genérica y satisfactoria. Cada dispositivo tiene un modelo de consumo característico que debe ser caracterizado, y lo que para unos funciona, para otros no. Únicamente los diseñadores hardware disponen de herramientas de simulación precisas que modelan de forma exacta los circuitos, incluyendo todos los elementos parásitos, y permiten determinar de forma precisa el consumo de los dispositivos electrónicos. El problema es que éstas son muy complejas y requieren una gran cantidad de recursos, por lo que típicamente se usan para los bloques más críticos. En su lugar, se suelen utilizar modelos genéricos de consumo, más simples, pero que igualmente permiten relacionar con una cierta precisión las conmutaciones con las trazas de consumo.

2.2.1.1 Modelo Consumo Distancia de Hamming (HD)

Dado que el consumo depende del número de transiciones 1→0 y 0→1, la idea de este modelo es contar el número de transiciones que ocurren en un circuito digital durante un intervalo de tiempo, y relacionar este valor con el consumo.

Si se asume que el consumo está relacionado con el HD del dato procesado, como suponen [Bri'04], [Bev'03] y [Cha'99], entonces éste depende de la energía necesaria para permutar los bits que cambian de un estado al siguiente. Esto implica que se requiere la misma energía para permutar un bit de 1 a 0 que de 0 a 1, y que el consumo depende del estado previo o de referencia, normalmente desconocido, que no tiene por qué ser cero, y que será el mismo siempre que se produzca la misma manipulación de datos al mismo tiempo.

Basado en esto, se puede decir que el HD es un modelo muy apropiado para describir el gasto de energía de buses, tanto de datos como de direcciones, así como de registros en implementaciones hardware [Man'07]. Por consiguiente, en aquellos sistemas donde exista un bus largo conectado a distintos componentes, el HD obtendrá buenos resultados, pues en

estos casos el consumo total estará influenciado de forma significativa por el bus, debido a su alta impedancia capacitiva. También relacionará bien el consumo de registros en dispositivos como FPGAs y ASICs, si se conoce los valores guardados en ciclos consecutivos de la señal de reloj.

La simplicidad y rapidez de procesado del HD, hacen que, a pesar de sus estimaciones poco precisas, sea un modelo ampliamente usado para simulaciones de consumo.

Como hemos dicho, el HD es un modelo genérico que considera a todos los bits iguales desde el punto de vista de su consumo energético. Igualmente, considera a las transiciones $1 \rightarrow 0$ y $0 \rightarrow 1$ energéticamente equivalentes. En cambio, si el atacante conoce la contribución de cada uno de los bits o transiciones, podrá adaptar el modelo genérico creando un modelo específico de su implementación, sin más que aplicar pesos a cada uno de los bits o transiciones para representar su contribución al consumo total [[Pee'07], [Li'10]].

2.2.1.2 Modelo Consumo Peso de Hamming (HW)

Este modelo establece que el consumo de potencia de un dispositivo al procesar un determinado dato, es proporcional al número de bits que son fijados a uno en el dato procesado.

Si se establece que el consumo depende del HW del dato procesado, indirectamente se está asumiendo que las conmutaciones $0 \rightarrow 1$ y $1 \rightarrow 1$ consumen la misma energía y que el consumo no depende del estado anterior.

De todo lo anterior se puede deducir que el modelo HW simula el consumo de manera muy burda. A pesar de ello, existen casos en los que este modelo es aplicable, como aquellos sistemas donde existe una gran diferencia de consumo entre los eventos $0 \rightarrow 1$ y $1 \rightarrow 0$, o donde existen buses con precarga a cero[8]. En general, este modelo se suele utilizar para aquellos casos en los que el atacante no tiene información acerca de la configuración del equipo bajo ataque (EUA), o cuando se desconocen dos valores consecutivos del procesado.

2.2.2 Radiación EM en dispositivos CMOS

Dado que el consumo de energía en un circuito CMOS es dependiente del dato computado, es fácil demostrar que su radiación electromagnética también lo es. Al realizar las medidas de campo EM a una distancia menor que aproximadamente la longitud de onda λ de

[8] En este caso el HW es equivalente al HD.

la fuente partido por seis[9] (zona de campo cercano), las señales pueden ser consideradas cuasi-estáticas, por lo que la ley de Biot-Savart es aplicable:

$$dB = \frac{\mu I dI \times \hat{r}}{4\pi r^2} \tag{2.2}$$

Con la ayuda de ésta, se demuestra que el campo EM depende del dato procesado, debido a que campo e intensidad de corriente lo son, y que la orientación del campo depende de la dirección de la corriente.

Por otro lado, con la ley de Faraday, se demuestra que cualquier cambio en el campo alrededor de una sonda de medida producirá un voltaje en la misma:

$$emf = -N\frac{d\Phi}{dt} \tag{2.3}$$

$$d\Phi = \int_{superficie} \vec{B} \cdot \overrightarrow{dS} \tag{2.4}$$

2.2.3 Técnicas de Análisis de datos

Los ataques criptográficos hacen uso de pequeños desequilibrios estadísticos en el estado interno de la clave. Incluso las medidas obtenidas por canales laterales con una pequeña correlación con algún estado interno de un bit, pueden ser usadas para romper un sistema. Es en estos procesos donde las técnicas de análisis de datos son más útiles, llegando a ser en algunos casos imprescindibles.

Las técnicas de procesado y análisis de datos permiten mejorar de forma considerable la eficiencia de los SCA conocidos, y en algunos casos los hacen factibles. Según [Schi'05], pueden llegar a aumentar la eficiencia de un ataque hasta en un factor 50.

Estas técnicas se suelen aplicar en los ataques diferenciales, donde se analizan un gran número de señales, normalmente de pequeño valor, que suelen estar debilitadas por errores de medida y otros ruidos. Su función es eliminar este ruido y descubrir la información oculta. De hecho, la mayoría de los SCA implementados hacen uso de diferentes herramientas estadísticas para intentar explotar al máximo la información contenida en los datos filtrados y capturados.

[9] Exactamente a $\lambda/2\pi$, siendo $\lambda = \frac{c}{f\cdot\sqrt{\varepsilon_r}}$, donde: c=$3\cdot10^8$m/s, f=frecuencia (Hz) y ε_r=Permitividad relativa (1 en el vacío).

El ruido es el principal obstáculo a tratar a la hora de implementar un SCA, y en general, en cualquier aplicación de procesado de señal. En el contexto de los SCA se tienen en cuenta varios tipos de ruido:

- Físico → Presente en el propio ambiente (EMI conducida o radiada, ruido cósmico etc.).
- De Medida → Debido al procesado de la señal (muestreo) y a las herramientas utilizadas (ruido generado por la fuente de alimentación, glitches, ruido de cuantificación del osciloscopio, señal de reloj inestable, tensión de alimentación variable etc.).
- De Ajuste al Modelo → Consecuencia de que el modelo de consumo definido para el EUA no es totalmente adecuado.
- Del dispositivo → Generado por el EUA y no dependiente del dato u operación realizada (ruido conmutación, ruido corrientes fuga etc.).
- De algoritmos → Producido por valores no tenidos en cuenta en una implementación.

Todas estas perturbaciones afectan a la eficiencia del SCA y como consecuencia, se reduce la cantidad de información filtrada y aumenta el número de medidas necesarias para llevarlo a cabo con éxito.

La relación señal a ruido: SNR, comúnmente usada en procesado de señales, extrapolada a los SCA, permite cuantificar la cantidad de información que se escapa por un canal lateral. Se define como la relación entre la señal útil que queremos medir y el ruido presente en una medida:

$$SNR = \frac{Var(Señal)}{Var(Ruido)} \tag{2.5}$$

Cuanto mayor sea el SNR, más información se podrá extraer de un dispositivo.

La mayoría de las técnicas utilizadas para el análisis de las medidas provienen de la teoría de procesado de señales. Ésta se encarga del modelado, detección, identificación y utilización de patrones y estructuras en una señal. Para ello utiliza como herramienta base la teoría de procesado estadístico de datos, que modela la distribución aleatoria de señales y el entorno en el que se propagan. El fin es extraer información de una señal que puede tener ruido, estar distorsionada y/o incompleta, aplicando modelos estadísticos.

Los métodos se pueden clasificar en cuatro grupos:

- No paramétricos → Se caracterizan por no utilizar un modelo paramétrico de la generación de la señal o de la distribución estadística. En su lugar, la señal es procesada como una forma de onda o una secuencia de dígitos. Algunos ejemplos de este tipo de análisis son:
  - Filtrado Digital.
  - Análisis de Fourier.
  - Interpolación.
  - Estimación del Espectro de Energía.
  - Etc.
- Basado en Modelos → Utilizan un modelo paramétrico del proceso de generación de la señal para describir su estructura y patrón previsibles, y así, intentar predecir valores futuros de la señal. Un ejemplo es la Predicción Lineal.
- Estadísticos Bayesianos → Permiten modelar la distribución de señales aleatorias, las cuales solo pueden ser descritas en términos estadísticos y de probabilidad de funciones.
- Redes Neuronales → Son combinaciones de procesado adaptativo no lineal y procesado de señal en neuronas.

2.3 Estudios Previos

En este capítulo se van a pormenorizar las principales técnicas de ataque por Análisis Electromagnético (EMA) publicadas hasta la fecha de redacción de este documento. Dado que, como ya se ha demostrado, existe una estrecha relación con los ataques por consumo energético (PA), y las técnicas implementadas para uno, pueden ser adaptables al otro en la mayoría de los casos, se van a exponer también las técnicas basadas en el análisis del consumo de energía.

Por otro lado, al ser el ataque por consumo eléctrico, PA, muy popular entre los investigadores, la mayoría de las técnicas documentadas tienen su origen en este canal lateral. Por este motivo se presentarán en primer lugar las técnicas PA y a continuación las EMA.

2.3.1 Ataques por Análisis de Consumo (PA)

El consumo de potencia de un dispositivo criptográfico puede proporcionar mucha información acerca de las instrucciones ejecutadas y los datos involucrados. De hecho, los PA han resultado ser efectivos y de probada eficacia en la mayoría de los dispositivos:

Microcontrolador 8 bits [Schu'04], 32 bits [Geb'06], DSP [Geb'03], FPGA [Sta'04a], ASIC [Ors'04], Microprocesador soft-core [Lum'13], Tarjeta Inteligente [Mes'02] etc. y algoritmos criptográficos: DES [Koc'99], AES [Ors'04], SEED [Yoo'04], ARIA [Ha'05], RSA [Jaf'06], ECC [Cor'99] etc. Este ataque se basa en el hecho de que el consumo de energía necesario para manipular un bit a 1 es distinto del necesario para manipularlo a 0.

La idea fue introducida por Kocher y otros en 1999 con su **Ataque Diferencial por Consumo** (Differential Power Analysis: **DPA**) [Koc'99], donde se conseguía descifrar la clave usada en el algoritmo de encriptación DES sin necesidad de ningún tipo de conocimiento acerca del dispositivo usado. Éste, consiste en realizar la encriptación de una serie larga de N datos (textos planos) aleatorios conocidos: P_i., i=1…N y capturar las correspondientes curvas de consumo asociados a cada uno de ellos $T_i = W(P_i)$, obteniendo así una serie de duplas Texto – Traza Consumo: $\{P_i , T_i\}$. Por otro lado se selecciona una función de selección D perteneciente al algoritmo de encriptación, que tome como parámetros de entrada un texto P_i conocido y una parte de la clave estimada K_s, y genere como salida un bit b: $D(P_i , K_s) = \{1 , 0\}$. El adversario hace una suposición de parte de la clave K_s y usa la función de selección D para dividir la serie de N trazas en dos grupos: uno que contiene las trazas de los textos que hacen la función de selección $D(P_i , K_s) = 1$ y otro con el resto de trazas cuyos textos asociados hacen que $D(P_i , K_s) = 0$:

$$G_{0,K_s} = \{T_i, i = 1 \dots N ., \quad D(P_i, K_s) = 0\}$$

$$G_{1,K_s} = \{T_i, i = 1 \dots N ., \quad D(P_i, K_s) = 1\}$$

Por último se realiza la media de cada uno de los grupos y se restan ambas:

$$\Delta_D(b) = \frac{\sum_{i=1}^{N} D(P_i, K_s) \cdot W(P_i)}{\sum_{i=1}^{N} D(P_i, K_s)} - \frac{\sum_{i=1}^{N} \big(1 - D(P_i, K_s)\big) \cdot W(P_i)}{\sum_{i=1}^{N} D(P_i, K_s)} \qquad \textbf{(2.6)}$$

Si la suposición hecha para la clave: K_s es correcta y el número de textos aleatorios es suficiente y siguen una distribución normal, la diferencia $\Delta_D(b)$ producirá un pico en el instante en el que el bit b es manipulado, puesto que la función está reflejando algo que realmente está ocurriendo en el dispositivo. Si la suposición es incorrecta, la diferencia tenderá a 0, ya que los textos escogidos son aleatorios según una distribución normal, y las aportaciones de unos textos se compensarán con las de otros. No obstante, en la práctica puede ocurrir que aparezcan varios picos a distintos tiempos siendo la clave elegida K_s correcta, o producirse algunos picos aunque la suposición sea incorrecta debido a la existencia de correlación entre el bit y la salida de alguna función. Por ejemplo, en el caso del AES la función Sbox (véase

Anexo 1 para más información) suele tener una mayor predisposición a ello. Es lo que algunos autores, como [[Bri'04], [Can'05], [Le'06]], denominan picos fantasmas.

A partir del estudio original de Kocher, han surgido diversos artículos sobre DPA aplicado a diversos algoritmos criptográficos, incluyendo AES y DES, y sobre diversos dispositivos criptográficos. Por ejemplo, Golic y otros en [Gol'02] describen un ataque sobre AES. En [Sta'04b] Standaert y otros, definen un ataque sobre una FPGA que ejecuta AES. En [Cor'99] se aborda un criptosistema basado en la Curva Elíptica etc.

Por otro lado, algunos autores se han centrado en mejorar el DPA. De esta forma, han surgido los multi-bit DPA. Así, en [Mes'02], Messerges y otros se introducen los d-bit DPA, que consisten en dividir las trazas de consumo en dos grupos usando una función de selección que tiene como salida *d* bits en lugar de un bit. Por consiguiente, para una función de selección que genera una serie de *d* bits: $D(P_i, K_j) = [b_1, b_2, \ldots b_d]$ se crean dos grupos. Uno que contiene las trazas de los textos que generan a la salida de la función D series con menos de la mitad de los bits a 1 y otro con los textos que generan la mitad o más de bits a 1:

$$G_{0,K_s} = \left\{ T_i, i = 1 \ldots N ., \qquad HW\big(D(P_i, K_s)\big) = HW(b_1, b_2, \ldots b_d) < \frac{d}{2} \right\}$$

$$G_{1,K_s} = \left\{ T_i, i = 1 \ldots N ., \qquad HW\big(D(P_i, K_s)\big) = HW(b_1, b_2, \ldots b_d) \geq \frac{d}{2} \right\}$$

Bevan y otros en [Bev'03], implementan un 4-bit DPA, que se obtiene como resultado de realizar cuatro mono-bit DPA, como el original de Kocher, a cada uno de los 4 bits de la serie elegidos. La decisión final se hace en función de la suma de los resultados obtenidos en los cuatro bits. El método es eficiente sólo si los valores de los 4 bits influyen en el consumo de energía al mismo tiempo y del mismo modo. Algo que no puede ocurrir en implementaciones software.

La generalización de este ataque no se consiguió hasta que en [Le'06], Le y otros, proponen el denominado DPA por división (Partitioning Power Analysis: PPA), a partir de una idea mencionada en [Akk'00]. En éste, las trazas de consumo son divididas en d+1 grupos de acuerdo a su HD respecto al estado anterior, o estado de referencia R:

$$G_j = \left\{ T_i, i = 1 \ldots N_j \; ; \; HD\big(R, D(P_i, K_s)\big) = HD(R, b_1, b_2, \ldots b_d) = j \right\}$$

La decisión final se obtiene de acuerdo a la suma y ponderación de los valores medios de cada grupo:

$$\sum_{HD(b_1 b_2 \dots b_d)} D = \sum_{j=0}^{d} a_j \frac{\sum_{G_j} T_i}{N_j} \qquad \textbf{(2.7)}$$

Donde a_j son las ponderaciones determinadas mediante una función de selección basada en bits conocidos como los textos de entrada. En función de estas ponderaciones se pueden obtener los diferentes tipos de ataques multi-bit.

Coron y otros investigadores, mejoraron el ataque reduciendo la posibilidad de obtener falsos positivos con la normalización de la diferencia de medias mediante la desviación estándar [Cor'00]:

$$G_{0,K_s} = \{T_i, i = 1 \dots N_0 ., \quad D(P_i, K_s) = 0\}$$
$$G_{1,K_s} = \{T_i, i = 1 \dots N_1 ., \quad D(P_i, K_s) = 1\}$$
$$\Delta_D(b) = \frac{\overline{G_{0,K_s}} - \overline{G_{1,K_s}}}{\sqrt{\frac{sd\left(G_{0,K_s}\right)^2}{N_0} - \frac{sd\left(G_{1,K_s}\right)^2}{N_1}}} \qquad \textbf{(2.8)}$$

De forma similar, Gebotys define la clave correcta como aquella que maximiza la diferencia de medias absoluta menos dos veces su desviación estándar [Geb'06]:

$$\Delta_D(b) = \left|\overline{G_{0,K_s}} - \overline{G_{1,K_s}}\right| - 2\sqrt{\frac{sd\left(G_{0,K_s}\right)^2}{N_0} - \frac{sd\left(G_{1,K_s}\right)^2}{N_1}} \qquad \textbf{(2.9)}$$

En su estudio [Agr'03], Agrawal y otros sugieren usar el principio de máxima probabilidad, en lugar de la diferencia de medias propuesto por Kocher, para determinar la clave correcta. Este procedimiento minimiza la posibilidad de cometer un error, si bien, no es concluyente puesto que depende de unos datos obtenidos aleatoriamente.

En [Bev'03], Bevan y Knudsen proponen simular un DPA para todas las posibles claves, y compararlos con un DPA real mediante un test de mínimos cuadrados. Aquella simulación que se ajuste más a la real, será la de la clave correcta.

Jaffe, en [Jaf'06], muestra que una selección de los textos de entrada permite atacar al AES mediante una función de selección que tiene como objetivo bits de salida de la función MixColumns (véase Anexo 1), en lugar de bits de la primera o última ronda[10].

[10] La función de selección usada para un DPA, debe tener como salida un valor intermedio del algoritmo criptográfico que dependa de un dato conocido no constante y de una parte de la clave. Para el caso concreto del AES, esto solo se produce en la primera ronda y en la última (véase Anexo 1).

En el año 2000, Messerges y otros muestran un ataque diferencial mejorado denominado **DPA de alto orden** (High-order DPA) [Mes'00a], que permite atacar sistemas protegidos contra los ataques de primer orden[11]. Éste consiste en buscar relación entre el consumo de dos o más puntos de la traza (según el orden del ataque) a distintos instantes de tiempo y los bits de la clave bajo ataque. Es decir, mientras que en un DPA de orden uno se analizan las propiedades estadísticas individuales de cada uno de los puntos de la traza, en un orden superior se investigan las propiedades estadísticas conjuntas de varios puntos. Esto demanda, como es lógico, una mayor capacidad de procesamiento y almacenamiento; Además de un conocimiento preciso del algoritmo de encriptación para conocer en qué puntos puede ser aplicada la estadística conjunta.

Teóricamente, un DPA puede ser de cualquier orden deseado, pero al aumentar éste, la complejidad aumenta rápidamente con él. Por ese motivo, la mayoría de los autores que han seguido el camino de Messerges optan por un orden dos. Así Waddle y otros autores en su artículo [Wad'04], definen un ataque de segundo orden denominado FFT 2DPA, al que se le ha incluido una etapa de preprocesado, que consiste en realizar una autocorrelación ($FFF^{-1}(FFT(Traza)^2)$) a las trazas. Joye, en [Joy'05], analiza de forma más extendida los DPA de segundo orden, incluyendo diversas generalizaciones etc.

Chari y sus compañeros, revelaron en [Cha'02] un nuevo ataque al que titularon **Ataque por Modelado** (Template Attacks), a partir de la idea inicialmente propuesta en [Fah'99]. Esta clase de ataque se compone de dos etapas: Una primera fase de caracterización o modelado, en la que se construye una base de datos con los modelos del ruido generado por el equipo bajo test, o EUT, para cada una de las posibles claves de acuerdo a una distribución Multivariable-Gaussiana, utilizando para ello un dispositivo experimental idéntico o parecido al EUA. Y una segunda etapa, en la que, a partir de los modelos obtenidos en la primera fase, se detecta la clave mediante una función de detección basada en probabilidad. Según los autores, supone el SCA más potente en un sentido teórico, ya que permite atacar a un dispositivo realizando una única medida. En contra, es un ataque poco práctico, ya que requiere el modelado previo de todas las posibles claves y textos de acuerdo a su perfil estadístico de fuga de información. Lo que implica calcular, para cada posible par de dato y clave (p_i , k_j), un vector de medias y una matriz de covarianza[12]. Por ejemplo, para el caso del

[11] Protegidos mediante contramedidas como máscaras, duplicación, suma de curvas, interrupciones aleatorias, variación frecuencia operación etc. Para más detalle véase: [[Akk'01], [Geb'06], [Cha'99], [Mes'00b], [Mes'01], [Gol'02], [Cor'99], [Gou'99], [Char'05]].

[12] La matriz de covarianza, crece cuadráticamente con el número de puntos de la traza y además debe ser invertida, por ese motivo es necesario, seleccionar algunos puntos denominados comúnmente en la literatura "*de interés*", que contengan la máxima información para definir el modelo.

algoritmo DES supone 2^{12}= 4096 modelos, pues existen 2^6 opciones de textos y 2^6 opciones de claves. Así mismo, este método presupone que el consumo del EUA, se distribuye según una función Multivariable-Gaussiana.

A la vista del potencial de los ataques por modelado, muchos autores han orientado sus esfuerzos a estudiar este ataque, surgiendo aplicaciones y desarrollos nuevos. Así, Agrawal y otros proponen extender el concepto de modelado a los DPA, desarrollando el denominado **Ataque DPA Basado en Modelado** (Template-Based DPA Attack)[13] [Agr'03]. Éste consiste en un DPA, en el que la selección de la clave correcta se realiza calculando las probabilidades de que las trazas medidas, $T = \{T_i,\ i = 1,2, \dots, N\}$, estén asociadas a cada una de las posibles claves $k_i,\ i = 1,2, \dots, K$. Para ello, modela el consumo del dispositivo al computar el valor intermedio objetivo para cada una de las posibles claves, y a partir de estos modelos calcula las probabilidades. La clave que presente una mayor probabilidad, será la correcta. Este método, continúa padeciendo el problema de que su efectividad depende de la selección de los puntos de interés a la hora de calcular los modelos.

Siguiendo una línea de investigación similar al DPA por modelado, Schindler y otros formulan el ataque **Estocástico** (Stochastic Attack), [Schi'05]. Básicamente se diferencia del DPA por modelado en que, en lugar de usar la media y la matriz de covarianza de las señales medidas para modelar o estimar el consumo de energía, utiliza funciones predefinidas moduladas con un peso, que varía de un dispositivo a otro y que se obtienen mediante su caracterización. En la fase previa se determinan los pesos de las funciones predefinidas y la distribución de ruido y en la segunda se detecta la clave. El inconveniente de este método es que exige la selección de una función óptima que describa el consumo de energía, por lo que en el mejor de los casos, será tan potente como un DPA por modelado.

En [Hos'11], los autores desarrollan la misma idea, pero apoyándose en otra técnica de clasificación y selección basada en el algoritmo de clasificación: Least Squares Support Vector Machines. En general, los resultados obtenidos no son mejores y éstos dependen del Kernel elegido.

Para hacer el DPA por modelado más práctico y efectivo, otros autores han realizado mejoras. Así, Agrawal y otros en [Agr'05], exponen un ataque que se centra únicamente en cuatro bits de salida. De esta forma el número de posibles modelos es mucho menor: 2^4=16 modelos. En cambio, se requiere más de una medida para obtener la clave. Mangard, Oswald y Popp plantean el uso de modelos reducidos, en los que se prescinde de calcular la matriz de

[13] El ataque por modelado presentado por Chari y otros, se implementa utilizando una única traza, por lo está englobado dentro de los ataques Simples. En otras palabras, se podría definir como un SPA basado en modelado.

covarianza. Ésta, simplemente se sustituye por una matriz identidad. También proponen estrategias para reducir el número de modelos, como centrarse en los bits de salida de una función intermedia del algoritmo o basarse en el modelo de consumo asociado al dispositivo [Man'07]. Otros, en cambio, se centran en el método para elegir los puntos de interés de la señal a la hora de calcular la matriz de covarianza [[Rec'04], [Gie'04], [Arc'06]] o en la simplificación de la función de detección de la clave correcta [Schi'05].

En otro término se sitúan los autores que se han centrado en la combinación de diversos ataques: los ataques **Multicanal** (Multi-channel Attacks) como Agrawal y otros, que combinan un DPA con un DEMA [Agr'03].

Schramm y otros autores desarrollan una nueva técnica: el Ataque por **Colisión** (Collision-Attack) [Schr'03]. Un algoritmo sufre una colisión interna, si al procesar diferentes argumentos de entrada se genera un mismo valor intermedio. Como consecuencia, el dispositivo genera dos trazas parciales de consumo idénticas o muy similares, que pueden ser detectadas mediante herramientas estadísticas de comparación como la correlación cruzada. Según los autores, para el caso concreto del AES, las colisiones dependientes de la clave pueden ocurrir a la salida de la operación MixColumns [Schr'04] (véase Anexo 1). Una vez son detectadas varias colisiones, se pueden establecer relaciones entre los textos que las generan y la clave, permitiendo al atacante deducirla. La ventaja principal de este ataque, sin considerar su eficiencia, es que de forma general se puede aplicar a la mayoría de los algoritmos criptográficos y su detección es relativamente más fácil, puesto que la colisión afecta a una secuencia de instrucciones[14], en lugar de a un instante de tiempo particular. Por contra, en algunos casos requiere de un setup complejo, es fuertemente dependiente del ruido y su fase de obtención de la clave es complicada.

Otros autores que han seguido los pasos de Schramm, intentando mejorar el ataque han sido: Moradi [Mor'10] donde se presenta un método de identificación más eficiente basado en la media de las trazas aplicado sobre una implementación de AES resistente a los ataques de primer orden; Bogdanov [Bog'07], desarrolla un método más eficaz, que se sustenta en la idea de la colisión interna generalizada, que se produce en cualquiera de las operaciones Sbox del AES al introducir dos bytes iguales; También centrados en AES, Ye y Eisenbarth [Ye'12], introducen las colisiones grandes, que son aquellas que se producen en cuatro bytes

[14] El valor intermedio se procesa, después normalmente se almacena en memoria, y más adelante es leído para, si se da el caso, ser sometido a más transformaciones. Como consecuencia, se producen varias colisiones internas que facilitan su detección.

consecutivos del texto, eligiendo para ello, textos con todos los bytes constantes excepto los de la diagonal principal del estado (bytes 1, 6, 11 y 16, véase Anexo 1).

En [Bri'04], Brier y otros autores proponen otra nueva variante de ataque más eficiente, bautizada con el nombre de **Análisis de Correlación de Energía** (Correlation Power Analysis: CPA). Éste se basa en la dependencia existente entre el consumo de energía y el HW o HD del dato procesado. Según Brier y sus compañeros, la relación entre el consumo de energía y el HD del dato procesado es lineal, de forma que la clave correcta es aquella que maximiza el factor de correlación. La decisión final se tomará, por tanto, en función del factor de correlación $\hat{\rho}_{WH}$:

$$\hat{\rho}_{WH} = \frac{N\sum T_i \cdot HD(R,P_i) - \sum T_i \cdot \sum HD(R,P_i)}{\sqrt{N\sum T_i^2 - (\sum T_i)^2} \cdot \sqrt{N\sum HD(R,P_i)^2 - (\sum HD(R,P_i))^2}} \quad \textbf{(2.10)}$$

A partir de la aparición del CPA, gracias a su eficiencia y facilidad de aplicación [Li'10], muchos autores han seguido su corriente, intentando mejorarlo. Así, en [Sch'10], Schimmel y otros autores exponen un CPA mejorado al que denominan Análisis Frecuencial por Correlación de Energía (CPFA). Consiste en un CPA al que se le añade un preprocesado previo que consiste en calcular la Densidad Espectral de Potencia de las trazas. La Correlación se calcula entre los consumos de energía teóricos calculados mediante HW y cada uno de los componentes frecuenciales de las trazas. Dado que la FFT es una operación lineal, y un cambio de amplitud en el dominio del tiempo produce el mismo cambio en el dominio de la frecuencia, la correlación es posible entre el dominio del tiempo y de la frecuencia.

Algunos investigadores se centran en mejorar el modelo de consumo, como en [Li'10], donde se presenta un CPA mejorado en el que se utiliza un nuevo modelo de consumo basado en el HD con la distribución de probabilidad del HW: $HD_{Li} = \sqrt{HW + 2}$.

Hutter y otros, en [Hut'12], muestran un ataque que utiliza la diferencia de consumo de energía de dos dispositivos criptográficos idénticos al procesar textos elegidos. Con este sistema, el ruido ambiental y el consumo estático y constante del dispositivo son cancelados, mejorándose así el ataque: señales con menos ruido, más sensitivas y con una relación señal a ruido menor. Según los autores, en relación a un CPA, supone un ataque más eficiente y con una relación señal a ruido mejor. Tiene el inconveniente que utiliza dos dispositivos criptográficos idénticos que se deben ejecutar exactamente de forma síncrona (con un reloj externo común o un reloj interno muy estable) y que requiere de un proceso de calibrado y selección de textos que permitan obtener valores intermedios complementarios en ambos equipos.

Otros autores han dedicado sus esfuerzos a estudiar sistemas protegidos contra ataques de primer orden. Es el caso de Waddle y otros [Wad'04], que desarrollan un ataque para equipos protegidos con máscara[15], denominado Zero-offset DPA, en el que se define el consumo de energía como: $P(t) = \epsilon(W(M) + W(Y)) + N$, donde ϵ es una constante de proporcionalidad, W(M) es el HW de la máscara aleatoria, W(Y) es el HW del dato dependiente de la clave enmascarado por M y, finalmente, N es el ruido gaussiano. También, en el año 2005, Mangard y otros autores, descubren que en los circuitos convencionales CMOS, el número total de conmutaciones de la línea de datos de puertas lógicas con máscara, está correlacionado con las señales de entrada y salida sin máscara. A partir de este dato desarrollan el Toggle-Count DPA [Man'05a]. El mismo autor junto a Schramm, proponen en [Man'06] el Zero-Input DPA, que se basa en el hecho de que la salida de la Sbox del AES, tiene un consumo mínimo si la entrada *x* es cero. Definen así el siguiente modelo básico de consumo:

$$P(x) = 0 \quad if\ x = 0$$
$$P(x) = 1 \quad if\ x \neq 0$$

2.3.2 Ataques por Análisis Electromagnético (EMA)

Quisquarter y otros en [Qui'01], fueron los primeros en presentar un estudio sobre los ataques EMA. En su publicación, basada en tarjetas inteligentes, propusieron que un procesador puede dejar escapar información, no sólo por consumo de potencia, sino también por radiación electromagnética. Según ellos, desde el punto de vista de un ataque no intrusivo, un EMA puede ser más preciso que un PA. También sugirieron que los EMA son fuertemente dependientes de la arquitectura del chip y el conocimiento del circuito interno del procesador facilita el trabajo. Para medir la radiación EM usaron tres espiras planas, dos iguales y una de mayor tamaño situada sobre el chip. A su juicio, los mejores resultados se obtienen con sondas de bajo factor de calidad, pues permiten medir el campo global. Los autores no presentaron resultados experimentales en este trabajo.

Gandolfi y otros en [Gan'01], demostraron experimentalmente, que los puntos de mayor intensidad de radiación son aquellos situados cerca de la CPU, buses de datos y líneas de alimentación; Siendo la más importante, desde el punto de vista de la información que contiene (la más correlacionada con el dato procesado), la que proviene de la CPU. Los autores realizaron las medidas de radiación EM sobre un Microcontrolador CMOS de 8 bits

[15] Es un valor aleatorio que aplicado a un valor intermedio del algoritmo mediante alguna operación matemática, intenta conseguir que el consumo no esté correlacionado.

que ejecutaba DES, utilizando de forma simultánea varias sondas pequeñas fabricadas por ellos mismos, compuestas de espiras de hilo de cobre de varios diámetros entre 150 y 500 micrómetros y una antena de campo lejano, demostrando así, que es posible realizar ataques puramente no invasivos. En el estudio se recalca la importancia de situar la sonda lo más cerca posible del EUT, y utilizar sondas muy pequeñas de tamaño similar al área del chip bajo estudio, para de este modo, disponer de un ancho de banda superior y capturar la mayor cantidad de campo posible. Por otro lado, también compararon los DPA y DEMA, llegando a la conclusión de que el DEMA es más efectivo (requiere un menor número de medidas), puesto que aunque es más ruidoso, genera una relación señal a ruido mayor.

En [Qui'02a], Quisquarter y Samyde presentaron un diccionario de instrucciones con sus trazas de consumo eléctrico / electromagnético y usaron técnicas de correlación y redes neurales, para reconocer las instrucciones ejecutadas por un procesador.

Agrawal y otros en [Agr'02], consideran la señal captada por un sensor EM como un conglomerado de señales de distintos tipos, amplitudes y contenido de información. En este estudio categorizaron a las radiaciones EM de Voluntarias o directas e Involuntarias. Las voluntarias son aquellas que se generan como consecuencia de flujos de corriente, mientras que las involuntarias son señales moduladas AM y FM que tienen su origen en acoplamientos con componentes cercanos. De acuerdo a los autores, las emanaciones involuntarias son más fáciles de atacar y más efectivas que las directas, y dentro de las involuntarias, las AM más que las FM. Además, algunas de ellas tienen mejor propagación que las directas, lo que permite ser captadas sin la necesidad de realizar un ataque invasivo. Según ellos, los mejores resultados para la captura de la radiación EM se obtienen utilizando sondas de campo cercano o antenas colocadas lo más cerca posible del EUT o al menos sin superar el campo cercano (aproximadamente un sexto de la longitud de onda de distancia), recalcando el uso de metales altamente conductores como plata o cobre. No obstante, también se pueden captar emanaciones a distancias superiores. Además, se demostró que con un equipo de bajo coste se podían realizar medidas bastante eficaces, y presentaron un ataque práctico a una tarjeta inteligente con DES y reloj externo a 3.68Mhz, utilizando, a diferencia de Gandolfi en [Gan'01], una única sonda y demodulando la señal EM, captada a diferentes frecuencias intermedias correspondientes a armónicos de la frecuencia del reloj.

En [Car'04] Carlier y otros, realizaron un ataque sobre una implementación hardware de AES en FPGA, donde todos los bytes del texto se procesan en paralelo. Para ello utilizaron sondas fabricadas con hilo de cobre bobinado, compuestas por una serie de espiras de un mm. de diámetro, situadas lo más cerca posible de la FPGA. Comprobaron que para una posición

específica de la sonda, sólo algunos bits podían ser atacados. Esto explica por qué el DEMA no es afectado por el procesamiento en paralelo, y sí el DPA.

En [Geb'05a] Gebotys y otros, introducen un nuevo tipo de ataque denominado Análisis Diferencial de Frecuencia (Differential Frequency Anlysis: DFA) y lo aplican a dos dispositivos criptográficos que ejecutan AES: una PDA y un Microcontrolador ARM Integrator/C7TDMI. Éste, es una variación del DPA original presentado por Kocher, y consiste en realizar los cálculos diferenciales sobre las señales en el dominio de la frecuencia, calculando para ello la densidad espectral diferencial de potencia de las señales. La principal ventaja de este tipo de ataque respecto al DPA o DEMA, es que no requiere un alineamiento previo de las señales a tratar, uno de los principales problemas a la hora de realizar medidas sobre dispositivos reales. En cambio, tiene como elementos en contra que no permite determinar el instante de tiempo concreto en el que se produce la correlación, por lo que no se pueden identificar las operaciones dependientes de la clave. Por otra parte, si la traza captada es demasiada larga algunos picos correspondientes a eventos rápidos pueden pasar desapercibidos [Sch'10].

La misma autora en [Geb'05b], propone un ataque al que denomina Análisis por Espectrograma Diferencial, que es básicamente un DFA, con la salvedad de que, en lugar de calcular en el preprocesado la densidad espectral diferencial de potencia, obtiene el espectrograma[16] de las trazas, pero no se especifican detalles de cómo seleccionar una ventana apropiada. También en [Geb'08a], analiza las emisiones de una PDA y realiza un DEMA utilizando la Densidad Espectral de Potencia. Éste, requiere una fase de preprocesado en el que se obtienen patrones de emisión, utilizando un dispositivo modelo y técnicas de truncamiento.

En su estudio, Hodgers y otros, [Hod'11], presentan un ataque similar al DFA, denominado Ataque por Densidad Espectral de Potencia con Ventana Variable (Variable Window Power Spectral Density Attack) el cual permite evitar contramedidas que varían el tamaño de las trazas captadas.

[16] Espectrograma: Permite conocer la energía del contenido frecuencial de la señal a lo largo del tiempo

Capítulo 3

EVALUACIÓN DE LA SEGURIDAD

La caracterización de un dispositivo en función de su seguridad, es algo que todavía no se ha logrado de forma óptima y robusta. Este hecho puede no resultar significativo en determinados equipos. Sin embargo, en el caso de los dispositivos embebidos de bajo consumo implementados en equipos portátiles, sí tiene importancia. Éstos suelen almacenar, enviar, recibir y manipular información sensible, por lo que el hecho de que el dispositivo carezca de una mayor o menor seguridad contrastada, es un factor a tener en cuenta a la hora de su adquisición y uso.

Por tal motivo, esta tesis se centra en el análisis de la seguridad inherente ante los SCA de cuatro microcontroladores actuales de propósito general, cuya principal aplicación son los dispositivos embebidos de bajo consumo.

Todos los microcontroladores a estudiar son de 32 bits, a excepción del primero, que es de 8 bits y está basado en una arquitectura 8051 mejorada denominada CIP-51. El segundo, el popular ARM7TDMI, con la arquitectura ARMv4T. Y los dos últimos son dos ARM Cortex M3, uno de ellos con especificación de bajo consumo. Éstos, pertenecen a la familia Cortex M de ARM, una de las tres familias que utiliza la arquitectura ARMv7 con distinta especificación:

- Cortex A (ARMv7-A): diseñada para aplicaciones complejas, como sistemas operativos que requieren altas capacidades de procesamiento y memoria.
- Cortex R (ARMv7-R): diseñada para aplicaciones embebidas en tiempo real.

- Cortex M (ARMv7-M): diseñada para aplicaciones de bajo coste, rendimiento eficiente y bajo consumo.

Hasta ahora, en la literatura no se había analizado de forma comparativa la seguridad de microcontroladores de bajo consumo orientados a aplicaciones embebidas. En [Geb'06] se estudia un microcontrolador ARM7, que en ese momento se considera un dispositivo de bajo consumo pues su alimentación se realiza a 3.3V, en lugar de los habituales 5V. Y en [Tiu'05], los autores se dedican a analizar una PDA, aunque no dan referencias acerca de su configuración interna ni modelo y/o fabricante. En [Ken'12] se analizan tres dispositivos: dos Smartphone y una PDA, también sin dar detalles acerca de los mismos. Sin embargo, en el artículo, únicamente se verifica la vulnerabilidad de los equipos a los SCA sin realizar un estudio más exhaustivo ni comparativo.

La evaluación de la vulnerabilidad, o resistencia de un dispositivo a los SCA, es un tema muy en boga, sobre todo desde la propuesta del NIST para el desarrollo de una metodología estándar [Nist'11]. Existen por un lado los métodos que se basan en comprobar la resistencia del equipo ante una serie de ataques conocidos, como el Common Criteria [CC'12]. Y por otro lado, se encuentran las técnicas que intentan detectar la presencia de cualquier tipo de información en las trazas que pueda ser utilizada por un SCA, sin necesidad de ningún tipo de conocimiento acerca del dispositivo [[Goo'11a], [Jaf'11], [Chat'10], [Chot'11]]. Tal es el caso de la metodología propuesta por Goodwill [[Goo'11a], [Goo'11b]], que utiliza la técnica basada en la diferencia de medias originalmente utilizada por Kocher para el DPA [Koc'99]. Aun así, todavía no hay una conclusión clara. La primera estrategia resulta ser poco práctica y tediosa, pues exige comprobar la respuesta del dispositivo ante diversos ataques y su actualización constante con los nuevos ataques descubiertos. En el caso de la segunda, las metodologías propuestas todavía tienen limitaciones, a la espera de que las técnicas estadísticas evolucionen (para más detalle véase [Math'13]).

Por ello, en este caso para realizar en este trabajo una evaluación de la seguridad inherente a los SCA de los dispositivos seleccionados, se ejecutará un código de encriptación en ellos sin ningún tipo de contramedida y se verificará la cantidad de información filtrada a través de alguna magnitud física; O dicho de otro modo, se comprobará la resistencia que ofrecen a un ataque, que más adelante seleccionaremos.

A continuación, se describen los distintos elementos seleccionados para llevar a cabo la evaluación propuesta.

3.1 Algoritmo de Encriptación

El código de encriptación que se utilizará para la evaluación, es el Estándar Avanzado de Encriptación: AES, principalmente por tres motivos. Primero, es el sistema de cifrado de información más usado [[Ska'11], [WikAES'13], [Man'06]]. De hecho, numerosos programas lo incorporan para encriptar datos, tales como WinZip, WinRAR, Windows XP, Vista, 7, 8, Phone 8, Server 2012, Mozilla Firefox, Google Chrome etc. [Nist'13]. Incluso el gobierno de EE.UU. desde el 2003, permite su uso para cifrar información no clasificada súper secreta. Por otro lado, su diseño está orientado a la optimización de los recursos [Dae'99], por lo que es perfecto para una aplicación embebida donde los recursos están limitados. Y por último, su descripción matemática es sencilla, así como su implementación en cualquier entorno. En el Anexo 1 se detalla más información acerca de este algoritmo de cifrado.

En cuanto a qué versión de AES utilizar, 128, 192 o 256 bits, se ha decidido implementar la versión más ampliamente utilizada 128/10, que recurre a una clave de longitud 128 y 10 rondas. En principio, lo lógico sería pensar que el nivel de seguridad del algoritmo es proporcional a la longitud de la clave utilizada, de forma que el AES/256/14 sería más seguro que el 192/12 y éste a su vez más que el 128/10. Sin embargo, en este caso la lógica no impera y algunos autores así lo demuestran. Es el caso de Biryukov y Khovratovich [Bir'09], los cuales consiguen reducir con su ataque el número de claves posibles del algoritmo AES/192 a 2^{123}, mientras que para el AES/256 obtienen 2^{119}, es decir, un valor inferior a las 2^{128} del AES/128. En [Bir'10] también se exponen algunas debilidades del estándar AES con longitud de clave 256, pero en este caso el algoritmo analizado no tiene 14 rondas.

3.2 Ataque por Canal Lateral: Análisis Electromagnético vs. Consumo vs. Tiempo

Llegados a este punto, la pregunta que surge es por qué elegir un EMA y no un PA o un ataque análisis de tiempos.

En primer lugar, descartamos los ataques por análisis de tiempo. En teoría y por definición, los resultados que se pueden llegar a obtener con este tipo de ataque no superan los de los otros dos, ya que se sirven de una magnitud física, el tiempo, que contiene menos información. Se puede considerar que este tipo de ataque explota una información unidimensional, mientras que los PA aprovechan una información bidimensional, la energía consumida por unidad de tiempo. En cambio los EMA disponen de una información tridimensional, el campo EM en una posición del espacio y en un tiempo específico, puesto que la sonda de medida se puede desplazar en el espacio. Por tanto, los PA y EMA pueden

proporcionar más información acerca de las instrucciones ejecutadas y los datos utilizados. Aunque es de rigor decir, que los ataques por análisis de tiempo están continuamente mejorando (Ej. Cache Timing Attacks [Osv'06]), en ciertos casos constituyen la única opción posible para extraer información y además cuentan con la ventaja de que se pueden acometer a través de la red de forma remota [[Bru'03], [Lev'04], [Cro'09]].

Según Gandolfi y otros [Gan'01], los EMA son más efectivos que los PA, pues permiten medir el consumo individual de determinados componentes del EUA[17] y consiguen una SNR mejor.

En [Agr'02], además se indica que las señales EM contienen más información que las medidas equivalentes de consumo de energía. Esto se debe a que las señales EM pueden ser captadas con un ancho de banda superior a las señales de consumo. En efecto, para medir el consumo de energía normalmente se conecta una resistencia de bajo valor óhmico en serie con la línea de alimentación o tierra. Este camino que recorre la señal a medir contiene muchos elementos parásitos que limitan su ancho de banda, como la capacidad de la fuente, la inductancia de los cables, etc. Esto provoca que la señal captada en un PA siempre esté filtrada. Por este motivo, el campo EM contiene más información, especialmente a altas frecuencias.

Por otra parte, este tipo de ataque es en esencia no invasivo, por lo que es posible su ejecución a distancia sin necesidad de manipular el EUA [[Gan'01], [Mey'10]]. No obstante, se puede realizar de manera más eficiente desencapsulando el chip, de modo que las medidas se puedan realizar con la mayor proximidad posible para tener una señal de mayor intensidad y evitar las posibles perturbaciones de la capa de pasivación, lo que redunda en una SNR mayor.

También está el hecho de que los EMA han sido estudiados en mucho menor grado que los PA [Pop'07] y por otro lado, el grupo de Sistemas Electrónicos de Potencia de la Universidad Carlos III de Madrid, en cuyo seno se ha realizado este trabajo, dispone de una cámara anecoica donde realizar las medidas en un ambiente aislado EM del exterior, aunque no sea estrictamente necesaria su utilización [Pee'07].

Existen también estudios con otra visión, como en [Tiu'05] donde los autores comparan los ataques DEMA y DPA sobre un ARM7TDMI, llegando a la conclusión, a diferencia de [Gan'01], que las trazas de consumo generan un mayor SNR que las trazas EM.

[17] En los ataques por consumo se mide el consumo global de todos los elementos del dispositivo criptográfico, independientemente de si contienen información relevante para el ataque o no.

Analizadas todas las impresiones, se decide aplicar este tipo de análisis en la presente tesis.

3.3 Ataque por Canal Lateral Electromagnético

Una vez seleccionado el ataque EMA, queda por definir el tipo a utilizar. Como se ha demostrado en el epígrafe 2.2, los PA y EMA están íntimamente ligados, de forma que, en general, la información que se filtra a través de consumo de energía también lo hace a través del campo EM. Así, en la mayoría de los casos, lo aportado por uno se puede aplicar al otro.

Dada la idiosincrasia de los SCA, llegar a la conclusión de que un ataque es mejor que otro, es complejo. Como ya se ha comentado, los SCA están intrínsecamente ligados al circuito electrónico en el cual se implementan. Por ese motivo, son complejos de evaluar, modelar, prevenir o comparar. Varios autores han hecho intentos de desarrollar un marco o herramienta común con el que poder evaluar los ataques de manera neutral [[Gen'04], [Mic'04], [Köp'07], [Dvam'08], [Sta'09b]], pero por diversos motivos[18], hasta la fecha no se ha llegado a una conclusión indiscutible.

Analizando el estado del arte, se puede llegar a la conclusión que los ataques más óptimos e interesantes para la realización de un estudio comparativo, son los DPA por modelado y CPA.

En [Man'07] los autores consideran los DPA por Modelado como los más potentes. Según ellos la forma óptima de describir el consumo de energía característico de un dispositivo es mediante la utilización de los modelos (*templates*). Éstos, permiten atacar al dispositivo utilizando pocas medidas y reducen la probabilidad de error. A cambio, tienen algunos inconvenientes que dificultan su aplicación. Como ya se ha comentado anteriormente, su éxito depende de la elección de los puntos de interés necesarios para calcular la matriz de covarianza. Esta selección debe hacerse de forma efectiva y cuidadosa, eligiendo el menor número de puntos posibles, ya que el tamaño de la matriz de covarianza crece de forma cuadrática con dicho número y ésta debe invertirse durante el proceso, lo que normalmente provoca problemas numéricos. Además en general, requieren la obtención de considerables modelos, aunque se han desarrollado técnicas que permiten reducir notablemente su número, como por ejemplo, centrarse en la salida de una función intermedia, utilizar el modelo de consumo asociado o fijarse en solo 4 bits [[Osw'07], [Agr'05]]. Por último, se asume que el

[18] Por ejemplo, las hipótesis hechas son muy fuertes, no queda muy claro cómo se aplica en la práctica, o bien no se analizan dispositivos más actuales [Sta'09b].

ruido generado por el dispositivo sigue una distribución Multivariable Gaussiana, algo que no siempre tiene por qué cumplirse.

En su artículo, Li y sus colegas [Li'10] establecen que los CPA proporcionan más robustez y eficiencia de análisis que los DPA. El coeficiente de correlación C.C. constituye la primera y más común herramienta cuando se trata de determinar si existe relación entre dos grupos de datos. Por consiguiente, es una opción excelente cuando se intenta aplicar un DPA [Man'07]. En [Koe'05], Koeune y otros indican que los DPA no hacen un uso óptimo de las medidas adquiridas. Los CPA suponen un avance respecto al DPA, aun así, tampoco son óptimos. Pero dado que los modelos de consumo tienen una relación lineal con las trazas, los CPA constituyen en la práctica, un ataque simple y eficiente. En ambos casos, su efectividad depende de la calidad y precisión del modelo de consumo elegido.

A la vista de los datos ofrecidos por otros autores, se decide el uso del ataque CPA. Como se ha expresado, este ataque supone un avance con respecto a un DPA clásico, ofrece unos resultados razonables con un coste computacional relativamente bajo y además, su descripción e implementación son relativamente sencillas. Dado que en este estudio se pretende realizar una comparación relativa entre varios dispositivos, la utilización de un ataque más potente, pero menos práctico, no supone un beneficio. La aplicación del DPA por modelado a nuestro estudio, exigiría en el mejor de los casos, determinar para cada una de las cuatro implementaciones los correspondientes modelos y puntos de interés. En nuestro caso, su mayor efectividad no compensa el coste extra de desarrollo.

3.4 Ataque por Correlación Electromagnética: CEMA

Como ya se ha mencionado, el ataque DPA por correlación o CPA, fue presentado en el año 2004 por Brier y otros autores [Bri'04]. Éste se basa en el hecho de que la clave correcta es aquella que maximiza el coeficiente de correlación entre la traza de consumo y el modelo de consumo usado.

Tras su publicación, varios autores han extrapolado este ataque al canal electromagnético [[Mul'05], [Rea'09], [Din'09], [Mat'10], [Mey'10], [Hor'12]].

A continuación se muestran en detalle los pasos a seguir para realizar un CEMA, o CPA sobre el algoritmo AES. En este punto, es recomendable consultar el Anexo 1, donde se explica en detalle dicho algoritmo:

Paso 1. Selección de una función intermedia del algoritmo

Esta función *f*, se corresponde con una o varias operaciones del algoritmo criptográfico, que generan como resultado un valor intermedio, que es función de una parte de la clave k_i y un dato variable conocido.

En general, en la mayoría de algoritmos criptográficos simétricos, se pueden determinar básicamente dos funciones que cumplen con estas premisas. Una pertenece a la primera ronda del algoritmo y otra a la última, puesto que lo más habitual es conocer dos datos, el texto plano de entrada *p* y/o el texto cifrado de salida *c* si se analiza la encriptación, o al contrario si se estudia la desencriptación.

Respecto al AES, se puede decir que existen principalmente tres funciones posibles[19], ya que la correspondiente a la ronda 1 se puede definir de dos formas. Se puede considerar la función *f* como la operación AddRoundKey de la ronda inicial, o incluir además la operación SubBytes de la ronda 1. En ambos casos, los bytes de salida de la función dependen del texto plano de entrada *p*, variable y conocido, y de una parte de la clave k_i. La otra función, se corresponde con la operación AddRoundKey de la fase final, que toma los bytes resultantes de la operación ShiftRows *d* y los añade a la ronda 10 de la clave k_{R10}, y tiene como resultado los bytes de salida cifrados *c* del algoritmo, variables y conocidos.

$$\text{Ronda Primera} \begin{cases} f(p_i, k_i) = p_i \oplus k_i = a_i \\ f(p_i, k_i) = Sbox_1(p_i \oplus k_i) = b_i \end{cases}$$

$$\text{Ronda Final} \begin{cases} f(c_i, k_i) = c_i \oplus k_{R10_i} = d_i \end{cases}$$

A la hora de seleccionar *f*, lo más conveniente es escoger una función perteneciente a la primera ronda, pues manipulará los bytes de la clave directamente sin realizar ninguna transformación. En cambio, si se escoge la función perteneciente a la última ronda, ésta trabajará con la ronda 10 de la clave, por lo que si el ataque tiene éxito, al finalizar no se obtendrán los bits de clave directamente, sino que se deberán determinar a partir de los obtenidos.

Por otro lado, las operaciones no lineales incrementan la eficiencia de los ataques estadísticos, por lo que, en el caso del AES, el ataque será más eficiente y requerirá menos

[19] Realmente, en la práctica, es posible atacar al AES con otras funciones, aunque exige realizar una selección de los textos de entrada o los textos cifrados de salida. Por ejemplo, en [[Car'05], [Jaf'06]], se demuestra cómo atacar al AES utilizando la función MixColumns.

medidas si se selecciona como objetivo los bytes de salida de la función Subbytes [[Pra'04], [Man'07]].

Paso 2. Cálculo del consumo hipotético

Una vez hecha la selección de una función intermedia *f* con las especificaciones indicadas en el paso 1, será posible calcular, a partir de una serie de textos planos aleatorios dados $p_1, p_2 \ldots p_P$, los valores intermedios generados por la misma para cada una de las posibles subclaves $k_1, k_2 \ldots k_K$. Se obtendrá así un valor intermedio para cada texto y para cada posible valor de la subclave. Es decir, $P \cdot K$ valores, que expresados en forma de matriz constituyen la matriz I_{PxK}, formada por *P* filas (una por cada texto plano) y *K* columnas (una por cada posible subclave).

Llegados a este punto, uno puede pensar que la realización de un CPA requiere una cantidad ingente de cálculos, pues exige calcular el valor intermedio para cada una de las posibles subclaves y textos. Aquí aplica el principio clásico de divide y vencerás. En el caso de elegir la función *f* de la primera ronda del algoritmo, cada una de las subclaves estará formada por 8 bits, pues la función Subbytes trabaja con bytes. Se tendrán, por tanto, 256 posibles subclaves, que es un valor relativamente bajo. En cambio, si se elige la función de la última ronda, será posible fijar como objetivo un bit, puesto que la función AddRoundKey trabaja bit a bit y, así se tendrán solo 2 opciones. Pero, esto exigiría repetir el proceso 8 veces por cada byte.

Una vez se ha obtenido la matriz *I*, el atacante debe determinar los consumos hipotéticos asociados a la computación de cada uno de los valores intermedios calculados, utilizando técnicas de simulación o modelos de consumo más o menos precisos, que predigan el gasto de energía del EUA. Se obtiene así una matriz W_{PxK}.

En esta fase se deben tener en cuenta dos datos importantes:

1. Cuanto más se aproxime el consumo hipotético al real, más efectivo será el ataque.
2. El objetivo no es obtener valores absolutos de consumo, sino relativos con los que poder comparar.

Es por eso que esta fase es una de las más importantes del ataque. En general, cuanta más información se tenga acerca del EUA, más complejas y precisas serán las técnicas de simulación que se podrán aplicar. No obstante, aquí también entran en juego otras variables importantes, como son la sencillez, rapidez, coste etc. En la práctica, los modelos de consumo más utilizados son los modelos genéricos HD y HW.

Paso 3. Registro de las trazas electromagnéticas

Esta fase consiste en medir el campo EM emitido por el dispositivo criptográfico al encriptar o desencriptar los P textos planos $p_1, p_2 \ldots p_P$. Se obtendrán así, P trazas: $t_1, t_2 \ldots t_P$, una por cada texto, cuya longitud o número de elementos N, dependerá de la tasa de muestreo utilizada y del tiempo de ejecución del algoritmo. Expresándolo en formato matricial, se puede decir que se obtiene una matriz T_{PxN} con P filas y N columnas. Una fila por cada texto y una columna por cada registro de la traza, correspondiente al campo EM medido en un instante de tiempo.

En este paso se deben tener en cuenta varios aspectos:

- La sonda utilizada debe tener un ancho de banda y sensibilidad suficientes para poder captar la señal íntegra. En general, será conveniente usar un preamplificador que aumente la ganancia de la señal, pues ésta es de baja intensidad y la sonda relativamente pequeña.
- Las trazas deberán estar correctamente alineadas, para que los puntos de la traza correspondan siempre a las mismas instrucciones, de forma que se puedan comparar adecuadamente en el tiempo. Para este fin, será necesario sincronizar el dispositivo criptográfico con el equipo registrador de las señales, que normalmente es un osciloscopio y/o utilizar técnicas de alineación como las expuestas en [[Tiu'05], [Geb'08b], [Sch'10]].
- En general, no será necesario captar la radiación emitida por el dispositivo durante toda la encriptación o desencriptación, sino sólo de aquellas instrucciones incluidas en la función seleccionada en la fase 1. De hecho, cuanto más grande sea la longitud de la traza, mayor será el procesado informático y la potencia requerida al equipo informático.

Paso 4. Comparación de las trazas electromagnéticas con los consumos hipotéticos

En esta última fase se realiza una comparación del consumo real del dispositivo criptográfico al encriptar los P textos $p_1, p_2 \ldots p_P$ medido a través de su radiación electromagnética, con el consumo hipotético teórico determinado en el paso 2. En concreto, se comparan las columnas de las matrices W y T. Es decir, cada columna de la matriz W, que contiene el consumo hipotético del dispositivo criptográfico al encriptar los textos $p_1, p_2 \ldots p_P$ suponiendo una subclave k_i, se compara con cada una de las columnas de la matriz T, que contienen las registros EM medidos en cada instante de tiempo determinado. Se obtiene así una matriz C_{KxT}, con K filas correspondientes a las posibles subclaves y T columnas, una para cada uno de los registros de las trazas de consumo EM.

La comparación de las columnas se puede realizar utilizando diferentes técnicas estadísticas. En este caso, dado que se utilizará el coeficiente de correlación $\hat{\rho}_{WT}$, se obtendrá

un valor numérico comprendido entre 0 y 1 que indicará la relación lineal existente entre ambas columnas. Cuanto mayor sea este valor, más similares serán ambas.

$$\hat{\rho}_{WT} = \frac{P\sum t_i \cdot w_i - \sum t_i \cdot \sum w_i}{\sqrt{P\sum t_i^2 - (\sum t_i)^2} \cdot \sqrt{P\sum w_i^2 - (\sum w)^2}} \qquad \textbf{(3.1)}$$

Una vez calculado el coeficiente de correlación para los *K*x*T* casos, si el número de textos planos elegidos es suficiente y el modelo de consumo aplicado es correcto, se podrá determinar la subclave correcta y el instante, o instantes de tiempo, en el que se procesa el valor intermedio resultante de la función *f* seleccionada, con solo verificar la fila y columna del mayor coeficiente de correlación calculado en la matriz *C*.

En ciertas ocasiones puede ocurrir que se obtengan valores altos de correlación para claves incorrectas. Es lo que en la literatura, en algunas ocasiones, se define como "picos fantasma" [[Bri'04], [Can'05], [Le'06]], inevitables, aunque dependientes en cierta medida de la función *f* seleccionada. Así, por ejemplo, la operación AddRoundkey del AES, genera picos fantasmas de mayor correlación (más altos) que la operación SubBytes [Man'07].

Veamos esquemáticamente el desarrollo completo de este ataque, Figura 3.1:

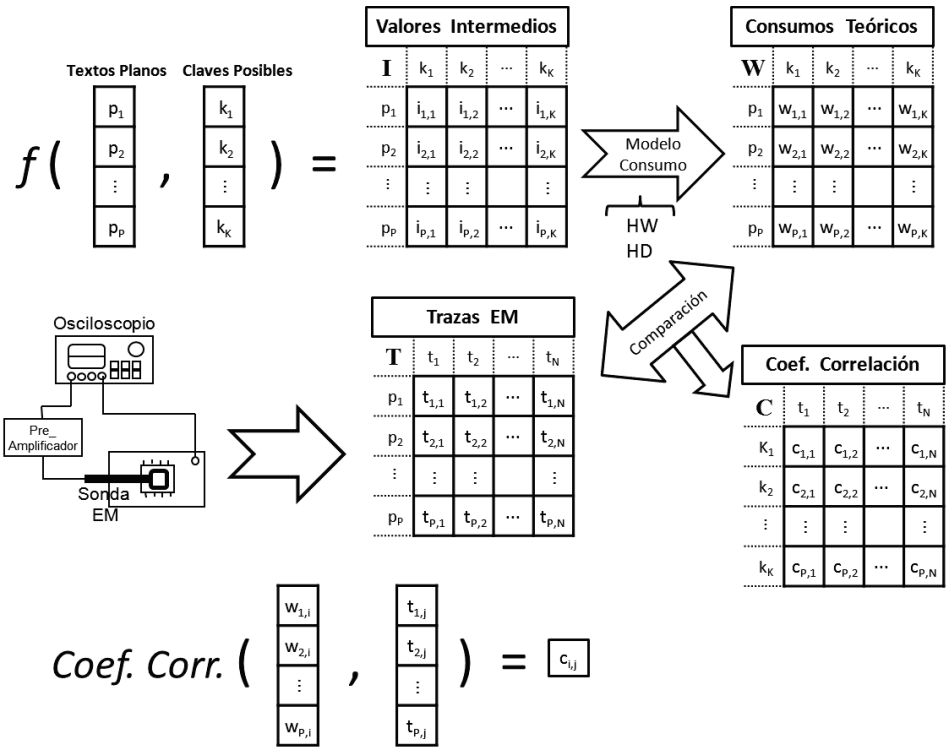


Figura 3.1: Esquema Ataque CEMA sobre AES

3.5 Dispositivos Criptográficos Bajo Test

Se han analizado cuatro microcontroladores distintos que ejecutan una implementación software del mismo algoritmo AES sin ningún tipo de contramedida. El primero es un modelo mejorado del conocido 8051 de 8 bits, el segundo es un ARM 7 de 32 bits, y los dos últimos son dos modelos ARM Cortex M3, también de 32 bits, uno de ellos con especificación de bajo consumo. Estos tres últimos procedentes de dos fabricantes distintos bajo licencia ARM.

3.5.1 Microcontrolador Silicon Labs C8051F303 8 bits

Este microcontrolador utiliza una arquitectura 8051 de 8 bits mejorada, denominada CIP-51. Ésta se caracteriza, entre otras cosas, por alcanzar una velocidad de procesamiento de instrucciones de hasta 25 MIPS a su frecuencia máxima de reloj (25 MHz), suponiendo instrucciones de un ciclo máquina. Esto representa una capacidad de procesado muy superior a un 8051 estándar, debido principalmente, a que en este equipo un ciclo de reloj equivale a un ciclo máquina[20] [SL'08].

Otra característica importante del dispositivo es su simplicidad. Además de la derivada de la utilización de una arquitectura 8051 de 8 bits, éste incluye muy pocos periféricos y está implementado sobre un chip de solo 11 patas[21] (véase la Figura 3.2). Otras características a considerar son: una memoria flash de 8 Kb, 256 bytes de RAM, emulación en tiempo real y alimentación entre 2.7 y 3.6 V.

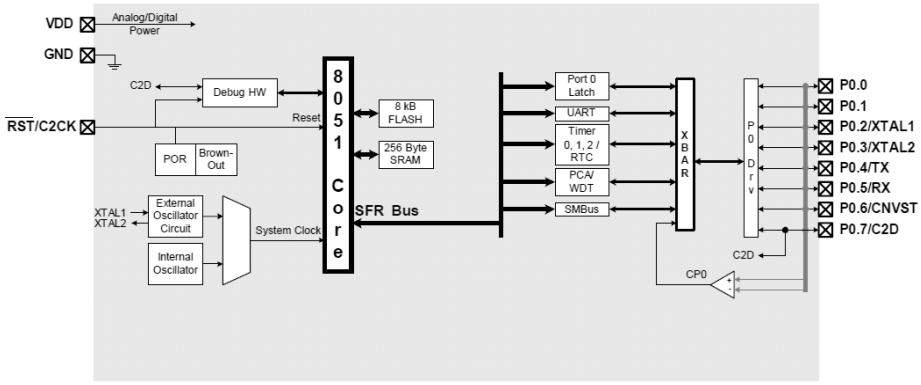


Figura 3.2: Esquema Microcontrolador 8 bits Silicon Labs C8051F303 (fuente [SL'08])

La elección de este microcontrolador se hizo en base a varios factores, como son su sencillez, robustez y experiencia previa con otros equipos del mismo fabricante, aunque sobre

[20] En una implementación 8051 clásica, un ciclo máquina utiliza 12 ciclos de reloj. En general, este valor suele variar entre 2 y 12 en función del dispositivo y fabricante.
[21] El encapsulado es de 12 patas, con una nula.

todo pesaron dos características que lo hacen distinto a las implementaciones de 8 bits analizadas previamente en la literatura. Por una parte su relativa alta capacidad de procesamiento y por otra su alimentación a una tensión de 3.3V, menor que los entonces habituales 5V. Además, aunque este tipo de microcontroladores de 8 bits están tecnológicamente desfasados, siguen usándose en diversos sectores, como automoción, aeroespacial, tarjetas inteligentes etc.

El circuito en el que se implementó el microcontrolador se diseñó y desarrolló ad hoc para los ensayos en una placa de circuito impreso utilizando el programa de layout PsPice (Figura 3.3). Con objeto de evitar posibles interferencias provenientes de otros dispositivos instalados, se decidió simplificar al máximo su diseño e incluir el mínimo de elementos posibles.

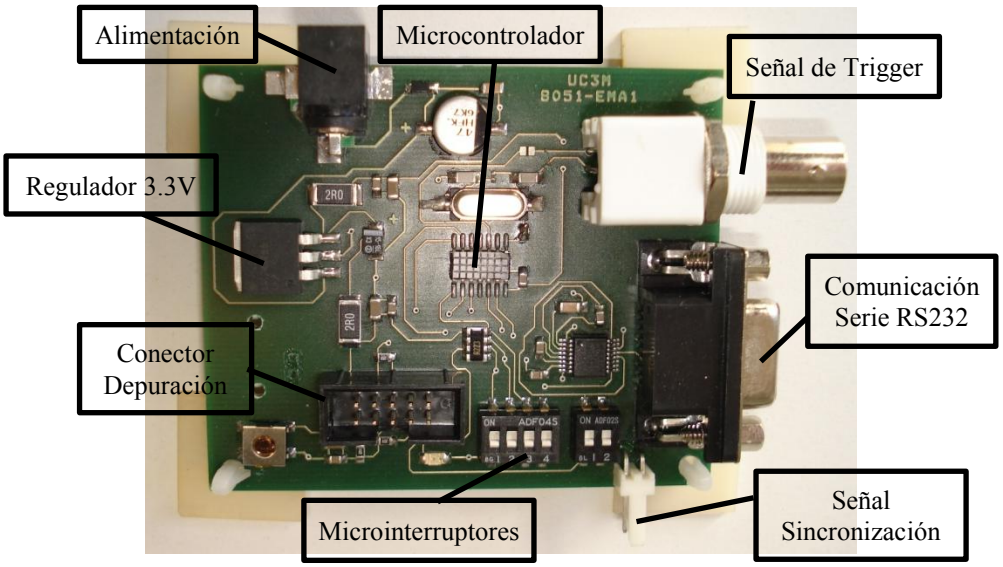


Figura 3.3: Placa EMA 1: Microcontrolador 8 bits C8051F303

La placa EMA 1 cuenta con un cristal externo de 11.0592 MHz, que proporciona la señal de reloj del microprocesador de forma más precisa que el oscilador RC interno, algo muy importante a la hora de realizar medidas consecutivas que deben estar perfectamente sincronizadas. Asimismo, contiene un regulador de tensión para alimentar el microprocesador a 3.3V, una serie de microinterruptores que permiten realizar variaciones en el programa sin necesidad de una reprogramación, como variar el dato introducido a cifrar o la clave utilizada, un conector para la programación y depuración y dos conectores de sincronización, uno que envía la señal de trigger al osciloscopio y otro al Pc que guarda las trazas capturadas por dicho osciloscopio.

Dado que fue la primera placa que se estudió, también se incluyeron algunos elementos en previsión, que finalmente no se utilizaron. Como un interfaz RS232 con su

correspondiente conversor de niveles de tensión para una posible comunicación serie con un Pc. Esta sección de la placa no llegó a alimentarse para evitar interferencias EMI indeseables.

3.5.2 Microcontrolador ARM7TDMI-S NXP LPC2124FBD64 32 bits

El LPC2124FBD64 es un microcontrolador de bajo coste fabricado por NXP (antes Philips), destinado a equipos empotrados, basado en la CPU de ARM ARM7TDMI-S, el cual ofrece altas prestaciones y un consumo bajo (0.28mW/MHz para la CPU). Entre otras características, dispone de 256 Kb de memoria flash de alta velocidad, 16Kb de RAM, permite la posibilidad de emulación en tiempo real, soporta frecuencias de reloj externas de entre 1 y 30 MHz, aunque gracias a un PLL interno, su frecuencia máxima interna puede llegar a 60 MHz y su alimentación es dual. Entre 1.65 y 1.95 V para la CPU y de 3.0 a 3.6V para los periféricos de entrada-salida [NXP'08].

Además de las mencionadas características de procesamiento, el LPC2124 dispone de numerosos puertos de entrada/salida que permiten disponer de comunicaciones, así como de otras capacidades. De ahí que el chip disponga de 64 patas. En la Figura 3.4 se muestra su esquema.

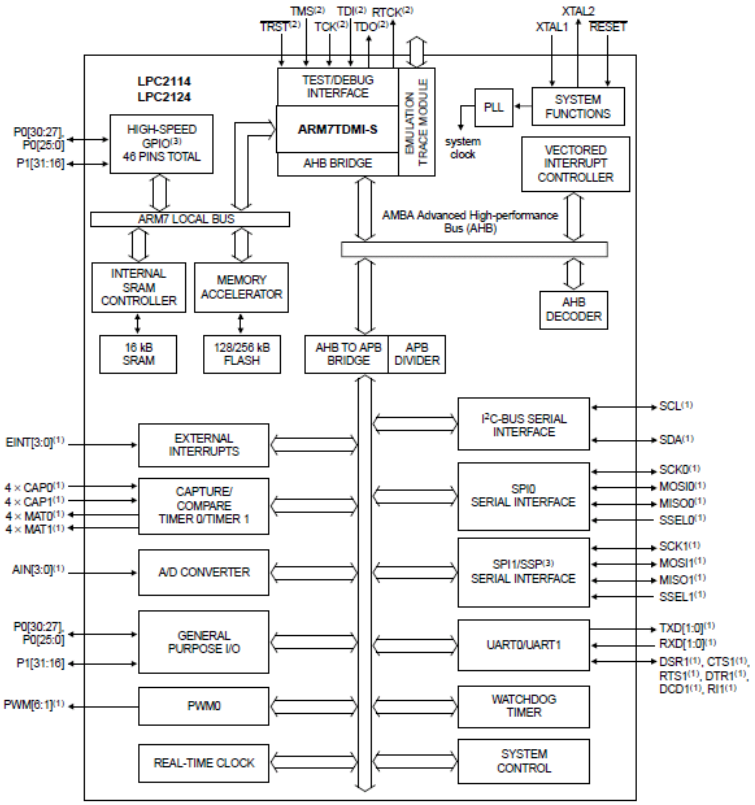


Figura 3.4: Esquema Microcontrolador NXP LPC2124 (fuente [NXP'08])

La elección de este dispositivo se hizo por dos motivos principalmente. Primero, es un microcontrolador diseñado específicamente para equipos embebidos de bajo consumo. Y en

segundo lugar, gozaba de un gran éxito, siendo ampliamente utilizado por la industria. De hecho, el ARM7 ha sido el procesador embebido de 32 bits más usado en la historia, con más de un billón de unidades producidas al año [Yiu'10]. Si bien, en estos momentos ha sido desbancado por otros dispositivos más potentes y eficientes, como los Cortex M0 y M3, aunque aún se sigue utilizando en equipos básicos [ARM7'13].

La tarjeta de desarrollo utilizada para los ensayos ha sido desarrollada íntegramente para los mismos y su sencillez es máxima con objeto de identificar claramente las emisiones EM provenientes del EUT y evitar el posible ruido generado por otros dispositivos que pudiesen estar instalados en dicha placa (Figura 3.5).

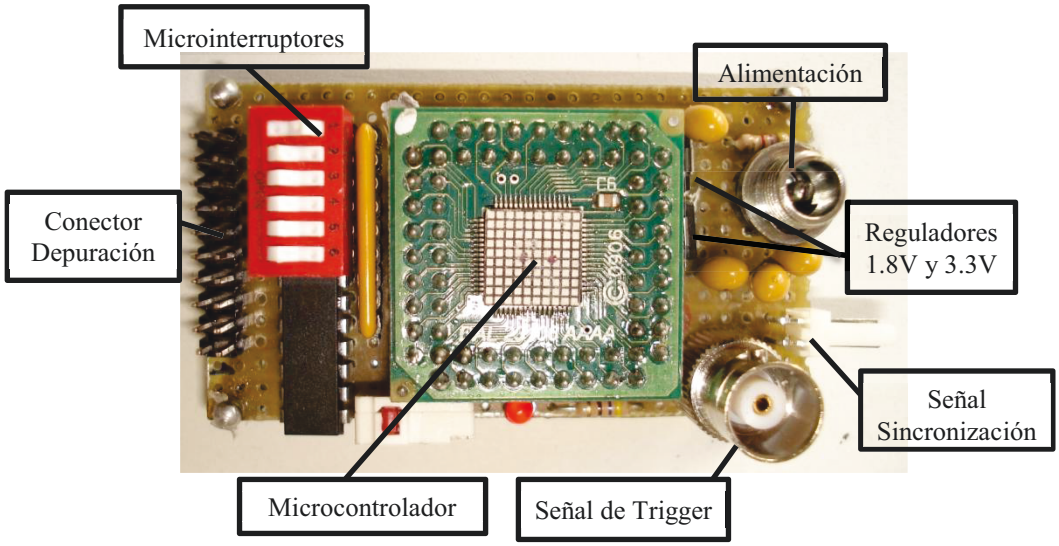


Figura 3.5: Placa EMA 2: Microcontrolador 32 bits ARM7 LPC2124

La placa EMA 2 dispone de un cristal de 12 MHz que proporciona la señal de reloj externa, aunque el dispositivo se ha configurado para trabajar a su frecuencia máxima de operación: 60 MHz. También, contiene como en la placa EMA1, los correspondientes reguladores de 1.8V y 3.3V de tensión para alimentar el microcontrolador según especificaciones del fabricante, una serie de microinterruptores que permiten, entre otras cosas y desde el punto de vista práctico de la investigación, poder variar el dato introducido para cifrar, así como la clave utilizada y diversos conectores para alimentación y sincronización.

3.5.3 Microcontrolador ARMCORTEXM3 NXP LPC1769FBD100 32 bits

Al igual que en el caso anterior, el LPC1769 de 32 bits de NXP, incluye bajo licencia la CPU de ARM CortexM3 basada en la arquitectura ARMv7-M. Éste, consiste en un microcontrolador de alto rendimiento, bajo coste, y bajo consumo (CPU de 0.19mW/MHz), idóneo para aplicaciones embebidas. Supone, por tanto, la evolución natural del ARM7TDMI-S y una alternativa a microprocesadores de 8 y 16 bits con menos rendimiento, pero precio

similar. Sus principales características son: 512 Kb de memoria flash, 16Kb de RAM, frecuencia de operación de hasta 120 MHz, posibilidad de generar la señal de reloj con un cristal externo con frecuencias comprendidas entre 1 y 25Mhz o una red RC interna, alimentación entre 2.4 y 3.6V y 100 pines [NXP'10]. Véase su esquema en la Figura 3.6.

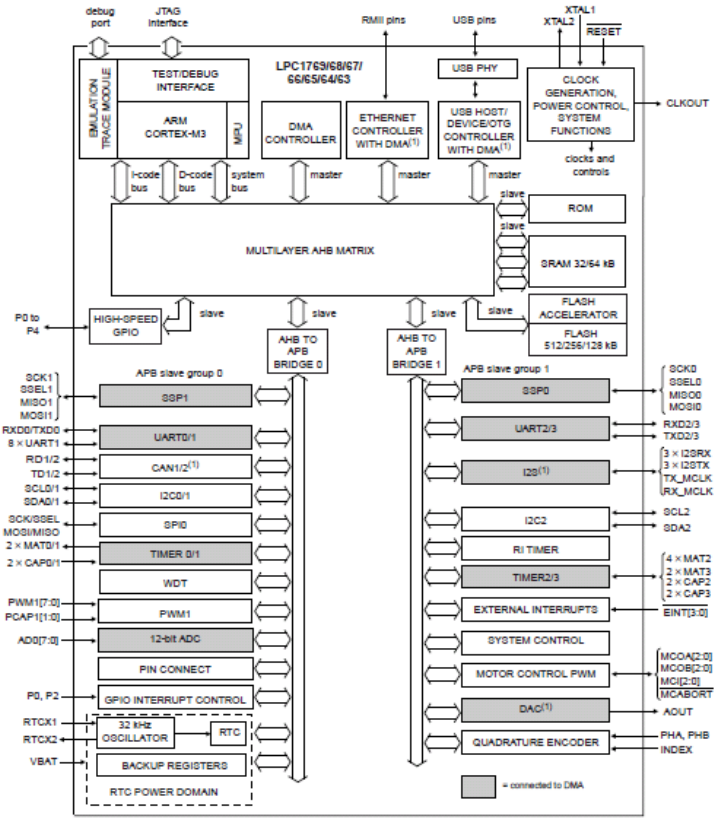


Figura 3.6: Esquema Microcontrolador NXP LPC1769 (fuente [NXP'10])

El equipo utilizado para las medidas, se compone de una placa de desarrollo comercial LPCXpresso LPC1769 a la que se le ha añadido un circuito con distintos conectores para alimentación y sincronización, como se puede apreciar en la Figura 3.7.

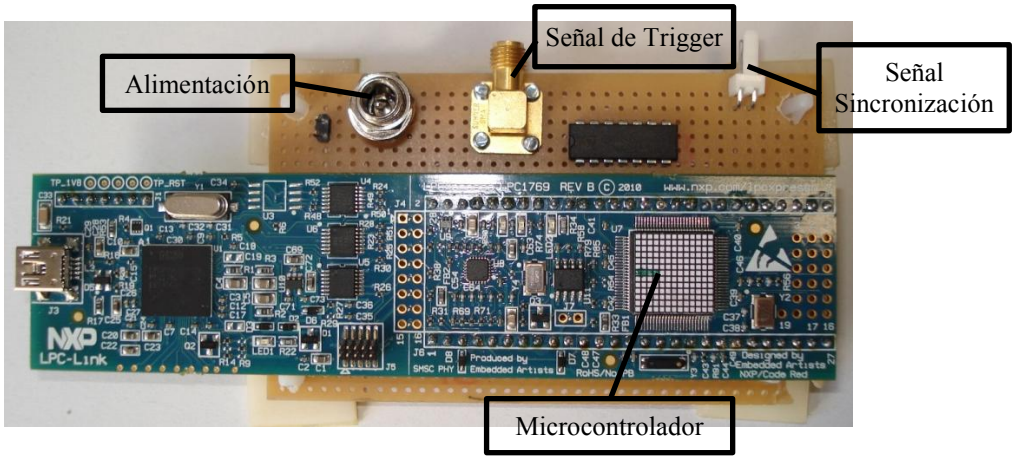


Figura 3.7: Placa EMA 3: LPCXpresso LPC1769

Al igual que EMA2, esta placa dispone de un cristal externo de 12 MHz. La frecuencia de operación se ha configurado internamente a 120 MHz, límite máximo.

Dado que este micro supone la evolución del ARM7 se vio lógico su estudio y análisis. Con el avance de dispositivos como smartphone y tablet, las necesidades de procesamiento han aumentado exponencialmente, por lo que el Cortex M3 se ha orientado a aplicaciones específicas embebidas de bajo coste, bajo consumo y rendimiento eficiente, como sistemas de automoción (control de ventanas, sensores, sistemas de navegación…), sistemas de control industrial y comunicaciones inalámbricas (3G, Bluetooth, WiFi, WiMax, Zigbee…), soluciones de almacenamiento, subsistemas de gestión de energía, sistemas GPS etc. La serie Cortex-A se ha elegido para dar soporte a aplicaciones sofisticadas que demandan un alto rendimiento y bajo consumo, como smartphone, tablet y PDA. A pesar de ello, ambos dispositivos utilizan la misma arquitectura ARMv7, pero con distintas especificaciones adaptadas al producto final.

3.5.4 Microcontrolador ARMCORTEXM3 STM32L152RBT6 32 bits

Este último microcontrolador es, al igual que el modelo anterior, un ARM Cortex M3 con arquitectura ARMv7-M, pero en este caso con especificación de bajo consumo. Ésta, consiste en una serie de mejoras que consiguen disminuir el ya de por sí bajo consumo de la arquitectura. Por ejemplo, 7 modos distintos de bajo consumo, uno de los cuales consigue un consumo mínimo de entre 0.3 y 1 μA, tensión de alimentación reducida entre 1.65 y 3.6V o núcleo con tensión dinámica y reducción proporcional. Además, dispone de 128 Kb de memoria flash, 16Kb de RAM, 4Kb de memoria EEPROM, su frecuencia máxima está limitada a 32 MHz con posibilidad de suministrar la señal mediante un cristal externo entre 1 y 24 MHz y el chip consta de 64 pines [ST'11]. En la Figura 3.9 se muestra su esquema.

El equipo utilizado para las medidas se compone de una placa de desarrollo comercial STM32L-DISCOVERY a la que se le ha añadido un circuito con distintos conectores para alimentación y sincronización, Figura 3.8.

La placa EMA 4, como se puede observar en la Figura 3.8, dispone entre otras cosas, de un LCD de 24 segmentos y una zona táctil, aunque para el estudio no se han utilizado. El micro se ha configurado para trabajar a la frecuencia máxima de operación de 32 MHz, a partir de un cristal externo de 8 MHz.

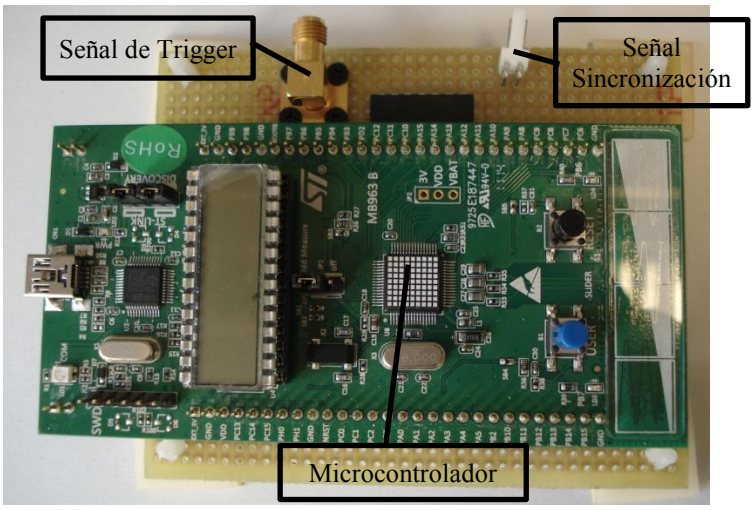


Figura 3.8: Placa EMA 4: STM32L-Discovery

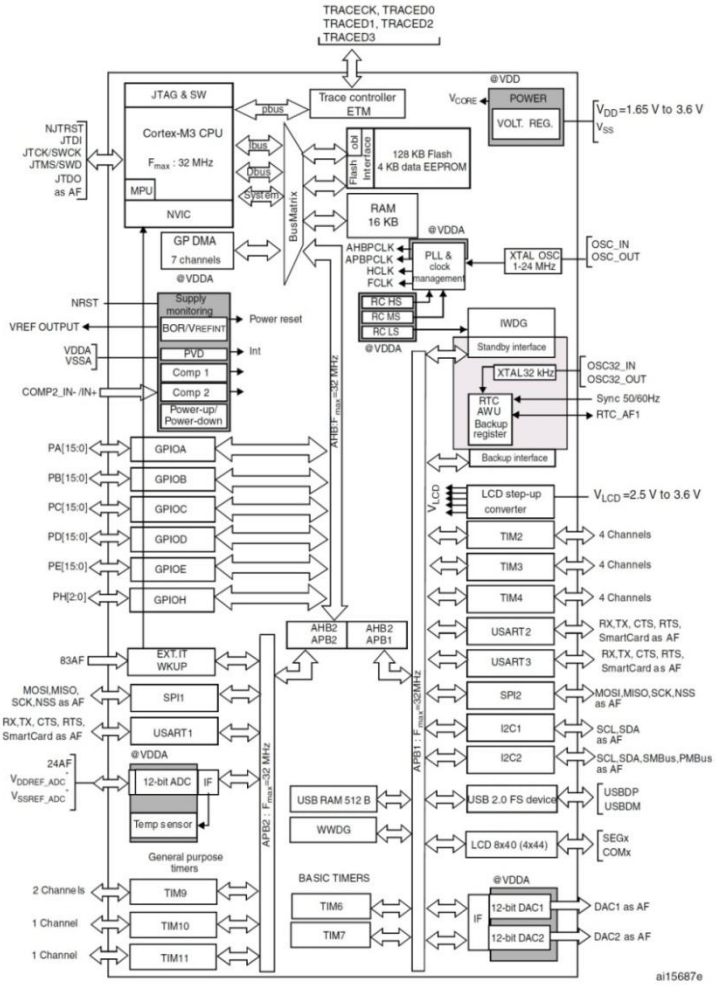


Figura 3.9: Microcontrolador STM32L152RBT6 (fuente [ST'11])

Capítulo 4

SETUP EXPERIMENTAL DE MEDIDA

La realización de un SCA persigue la captura y digitalización de una magnitud física observable, relacionada con un parámetro interno del dispositivo a atacar, para su tratamiento posterior en busca de información secreta. En particular, un ataque CEMA diferencial, requiere medir la radiación EM generada por el EUA mientras ejecuta un algoritmo criptográfico sobre un determinado número de textos planos y posteriormente analizar estadísticamente las trazas medidas, para intentar extraer la información que contienen. Para ello, es necesario tener en cuenta una serie de consideraciones de carácter práctico y un buen test setup de medida específico con distintos elementos que se describirá a continuación. La utilización de un setup de medida idóneo, es de vital importancia a la hora de realizar un SCA, influyendo notablemente en la eficiencia, e incluso en el éxito del mismo [[Sta'09a], [Hut'12]].

Como ya se ha comentado, los SCA electromagnético y de consumo, tienen muchas similitudes en cuanto a técnicas y métodos. En el caso del setup, ocurre igual. El setup utilizado para un PA generalmente se puede extrapolar a un EMA, sin más que sustituir el circuito medidor de consumo, que normalmente suele ser una resistencia de bajo valor óhmico, por una sonda electromagnética.

4.1 Setup de Medida Típico de un ataque EMA

En general, en un setup típico de medida suelen estar presentes diversos equipos, que trabajan aliados con el objetivo común de obtener un dato secreto. En la Figura 4.1 se muestra el diagrama de un setup típico.

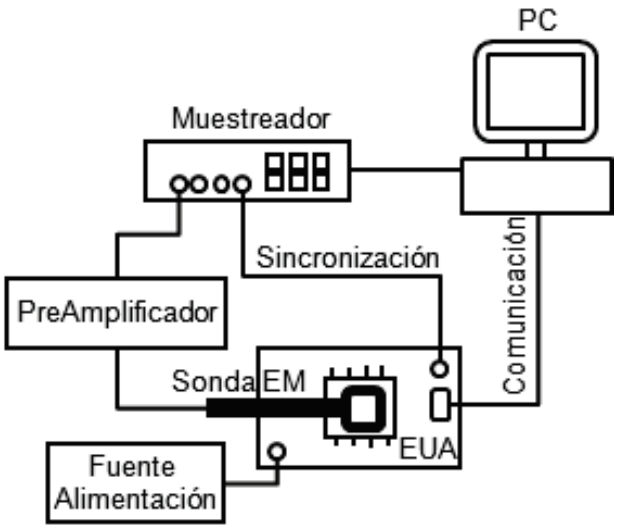


Figura 4.1: Setup Típico de Medida ataque EMA

Veamos los dispositivos que generalmente lo conforman:

- **Equipo Bajo Ataque (EUA)**: Éste consiste en un dispositivo electrónico que contiene la información confidencial que se quiere descubrir y con la capacidad de ejecutar un algoritmo criptográfico, como puede ser un microprocesador, tarjeta inteligente, FPGA, etc. Normalmente este dispositivo dispone de varios interfaces:

    a) Interfaz de comunicación. Normalmente es una interfaz serie que permite la comunicación con un PC. Suele ser usado para el envío de comandos, la recepción del texto a cifrar y el envío del texto cifrado al PC.

    b) Interfaz de sincronización. Permite coordinar la ejecución del algoritmo criptográfico con el equipo medidor de las señales. Es decir, proporciona información acerca del inicio y fin del proceso criptográfico.

    c) Interfaz de Reloj. En algunos casos la señal de reloj puede ser generada externamente, bien a través de un cristal de cuarzo o bien mediante algún tipo de generador de señales, lo que permite una rápida variación de la frecuencia de ejecución del dispositivo y el aseguramiento de una señal perfectamente estable. Según [Man'07] la opción más recomendable es usar una señal de reloj sinusoidal, pues su perfil de ruido es mucho menor que una onda cuadrada.

- **Fuente de Alimentación**: Proporciona la tensión de alimentación necesaria para el funcionamiento del EUA.
- **Sonda Electromagnética**: Dispositivo transductor que convierte el campo EM en una corriente eléctrica proporcional medible. Puede ser una sonda de campo cercano de tamaño similar al del dispositivo criptográfico o una antena receptora de campo lejano para medir a mayores distancias.
- **Preamplificador de banda ancha**: Amplifica la señal captada por la sonda electromagnética para aumentar la SNR de la medida.
- **Equipo muestreador digital de la señal**: Este equipo digitaliza la señal captada por la sonda electromagnética y la transfiere al PC a través de una conexión USB, GPIB o Ethernet. Una señal de sincronización, proporcionada por el dispositivo criptográfico, le indica los momentos justos en los que comenzar y finalizar la captura. Puede consistir en una tarjeta de adquisición de datos con alta resolución y varios canales o un osciloscopio de alto ratio de muestreo, en general con menor resolución, aunque de uso más común en un laboratorio.
- **PC**: Su principal función es almacenar la señal digitalizada por el equipo muestreador. Si bien, también hace las funciones de coordinador de la cadena de medida. En general, un CEMA requiere la realización de un alto número de medidas, por lo que es necesario un equipo con una alta capacidad de almacenamiento. El PC manda el texto a cifrar o descifrar[22] al EUA, y cuando éste ha sido cifrado o descifrado, lo recibe y lo almacena en memoria junto con la traza electromagnética capturada por el equipo muestreador.

4.1.1 Procedimiento de Medida

En general, el método de captura de las trazas EM necesarias para realizar un ataque EMA es el siguiente. En primer lugar el PC envía al equipo bajo ataque o EUA el texto a cifrar. Éste lo recibe, le envía un comando al muestreador para que comience a grabar la traza proporcionada por la sonda electromagnética y acto seguido cifra el dato recibido haciendo uso de la clave secreta almacenada. El muestreador captura la corriente proporcionada por la sonda EM durante el cifrado, previamente amplificada por el Preamplificador de banda ancha, y envía el registro al PC. Por último, el EUA envía el texto cifrado al PC, el cual lo almacena

[22] En general, es posible realizar un ataque tanto en el cifrado del texto, como en el descifrado. Esto depende del dato disponible en cada caso.

junto a la traza electromagnética. Así este proceso se repite el número de veces necesario para realizar el ataque.

4.2 Setup de Medida Experimental Implementado

Para la realización de este trabajo ha sido necesario el desarrollo de un setup de medida prácticamente automatizado, puesto que el número de repeticiones y medidas necesarias, hacía imposible su ejecución en modo manual. Éste, ha sido desarrollado a partir de los recursos limitados disponibles en el momento de la realización del trabajo, y su efectividad ha quedado demostrada experimentalmente.

El setup está basado en el diseño usado por la mayoría de investigadores con algunas modificaciones que se han considerado oportunas. Es recomendable usar el menor número de componentes posibles para evitar posibles perturbaciones y, en caso de que sean imprescindibles, separarlos del EUA. El esquema se muestra a continuación:

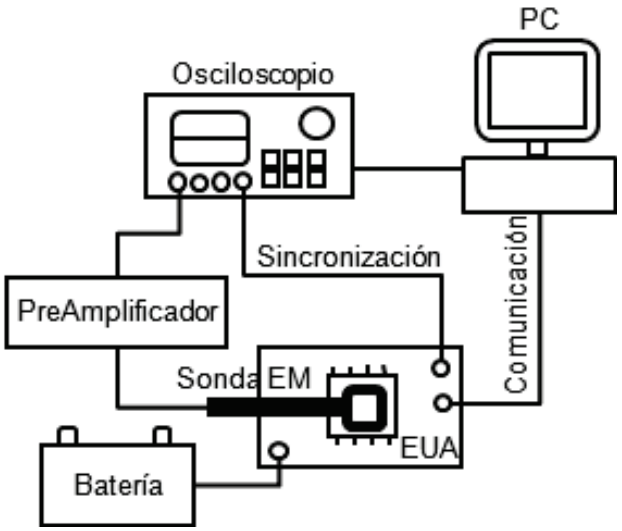


Figura 4.2: Setup de Medida utilizado

4.2.1 Procedimiento de Medida

En nuestro caso, el procedimiento de medida es el siguiente. El EUA, que será uno de los microcontroladores a analizar descritos en el epígrafe anterior, contiene los textos a cifrar almacenados en su memoria de programa. Éste se encuentra a la espera de recibir la orden de inicio del proceso de encriptación proveniente del PC. En el momento de recibirla, envía una señal de disparo al osciloscopio para que comience a capturar la señal proveniente de la sonda EM. Acto seguido, encripta el primer texto y se mantiene a la espera de recibir la orden de encriptación del siguiente texto plano a cifrar. A continuación, el PC solicita la traza medida al osciloscopio y la almacena en memoria. Una vez se ha realizado este paso, el más lento de la cadena, el PC vuelve a enviar al EUA la señal de inicio para que encripte el siguiente texto.

Los elementos del setup de medida utilizado se describen seguidamente.

4.2.2 Fuente de Alimentación

La calidad de la fuente de alimentación es uno de los principales parámetros que se deben cuidar a la hora de realizar un EMA o PA. De hecho, la presencia de ruido en la fuente reduce la precisión de las medidas seriamente [Aig'00]. Es por ello que algunos autores, como [Ben'03], proponen técnicas de adición de ruido a la fuente como contramedida para evitar los SCA.

En general, todo el ruido generado por la fuente se acoplará a las medidas realizadas, por lo que el uso de una fuente de calidad puede significar la diferencia entre descifrar la clave secreta o no. Por consiguiente, es imprescindible disponer de una tensión de alimentación completamente estable y sin ruido.

En el test setup implementado, se ha utilizado una batería recargable de 6V como fuente (Figura 4.3). De esta forma se asegura la ausencia de ruido en la señal de alimentación. A partir de esta tensión, y con la ayuda de reguladores lineales, se obtienen los voltajes necesarios para cada uno de los dispositivos a estudiar.

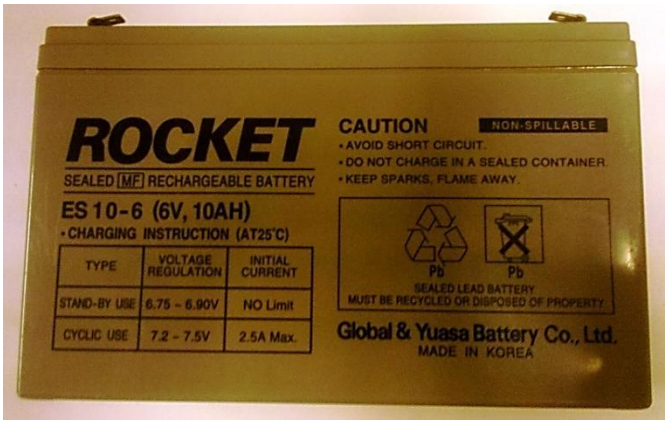


Figura 4.3: Batería 6V usada como fuente de alimentación

4.2.3 Osciloscopio

Es el dispositivo que se encarga de medir, digitalizar y guardar provisionalmente la tensión proporcionada por la sonda EM de medida, para su posterior envío al PC coordinador, que se encarga finalmente de almacenar los datos. Debe ser, por consiguiente, un equipo con un ratio de muestreo, ancho de banda, resolución y sensibilidad adecuados para poder llevar a cabo las medidas. Además, debe tener la capacidad de poder controlarse de forma remota a través de algún puerto de comunicaciones: USB, Ethernet, GPIB etc.

En general, en la literatura, los setup de medida utilizados suelen disponer de un equipo muestreador compuesto por un osciloscopio con hasta 1 GHz de ancho de banda, 1 Gmuestras/segundo de ratio de muestreo y 8 bits de resolución [Man'07]. En este caso se ha utilizado un osciloscopio Tektronix MS04104 (Figura 4.4), con las siguientes especificaciones [Tek'03]:

Tabla 4.1: Especificaciones técnicas osciloscopio Tektronix MSO4104

Tektronix MSO4104	
Ancho Banda	1 GHz
Ratio Muestreo	5 Gmuestras/sg.
Resolución	8 bits
Longitud medida	10 Mmuestras
Comunicación PC	USB 2.0 Ethernet

Por tanto, sus prestaciones son más que suficientes. Hay que tener en cuenta que la frecuencia de operación del EUA va a ser, al menos en un caso, superior a la de los dispositivos estudiados hasta ahora, donde no suele ser habitual superar los 50 MHz[23] y que un sobremuestreo no aporta ningún beneficio, sino que más bien dificulta el análisis de las muestras, al presentar las trazas un mayor número de registros y aumentar los recursos necesarios del equipo informático.

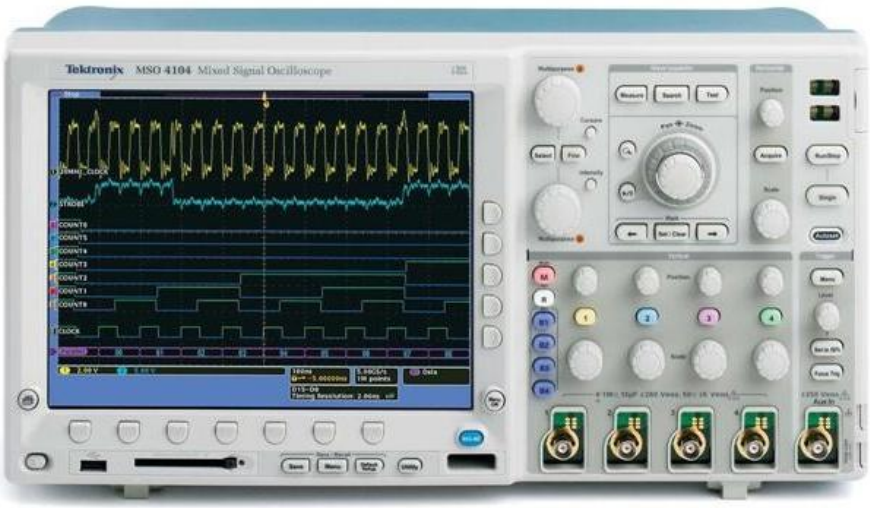


Figura 4.4: Osciloscopio MSO4104 utilizado como digitalizador (fuente Tektronix Inc.)

[23] Por ejemplo: 40 MHz en [Geb'05a]; 50 Mhz en [Car'04]; En [Geb'08a] se estudia una PDA, pero no se dan datos acerca de la frecuencia de operación.

4.2.4 Sonda Electromagnética de Campo Cercano

Es quizás el elemento más crítico de la cadena de medida, ya que es el encargado de captar el campo magnético generado por el EUA y transformarlo en una tensión proporcional medible por el osciloscopio.

Generalmente, cuando se ataca a un dispositivo, sólo una parte del mismo radia el campo magnético útil para deducir la clave secreta. Es decir, en un EMA, y en general en cualquier SCA, lo ideal es captar sólo la parte de la magnitud física filtrada que es dependiente del dato; El resto se considera ruido.

Actualmente, medir en un PA únicamente la energía consumida por la parte del circuito cuya energía consumida está correlacionada con el dato procesado, es una tarea pendiente. Sólo es posible evaluar el consumo global de todos los componentes a la vez. En cambio, en un EMA teóricamente es posible, debido a que la sonda se puede situar en aquellas zonas del EUA que generan el campo correlacionado. Sin embargo, en la práctica esto resulta prácticamente imposible. Con el nivel de integración actual, siempre van a existir acoplamientos inevitables con componentes cercanos.

4.2.4.1 Características de una Sonda Electromagnética

Una sonda EM debe tener la propiedad de:

a) Medir el mayor espectro posible del campo EM manteniendo su ganancia constante. Es decir, debe tener un *ancho de banda* suficiente.

b) Captar campos de bajo nivel, por lo que debe disponer de una alta *sensibilidad*.

c) Discernir el origen de las radiaciones. Es decir, tiene que poseer una buena *resolución espacial*.

De la literatura publicada se pueden obtener una serie de directrices a la hora de seleccionar/fabricar la sonda EM de campo cercano más apropiada:

- Forma: Según [Smi'98] es aconsejable utilizar sondas cuadradas para realizar medidas de tensión o corriente en un circuito, siendo indiferente la forma (redonda o cuadrada) para medidas en campo abierto, Figura 4.5.

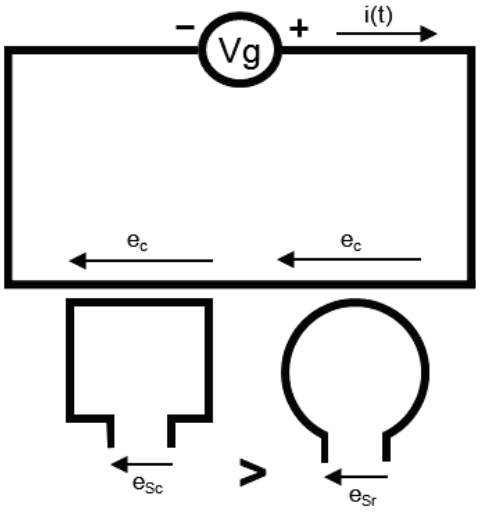


Figura 4.5: Influencia Forma sonda

- Tamaño: El análisis EM de dispositivos de pequeño tamaño, requiere sondas muy pequeñas con dimensiones similares a las del área bajo estudio [Gan'01]. De esta forma, se podrá obtener una resolución espacial suficiente que permita discernir el origen de las radiaciones EM pero a cambio tendrá el inconveniente de que su sensibilidad será muy pequeña. Una sonda grande captará más señal y, por tanto, tendrá mayor sensibilidad, algo bastante útil cuando la señal a detectar es muy débil, pero tendrá el inconveniente de que no podrá discriminar si el origen de la misma está situado en una zona u otra.

  Por otra parte, al analizar la respuesta en frecuencia de una sonda en función de su tamaño, se comprueba que el comportamiento a altas frecuencias mejora conforme disminuye el tamaño (Véase la Figura 4.6). Según [[Smi'98], [Smi'99]], una espira de 2,5 cm posee una frecuencia de corte inferior de 100 MHz. Si ésta es de 1 cm, la frecuencia de corte se producirá entre 200 y 300 MHz. Esto es debido al filtro paso bajo LR formado por la impedancia de 50 Ω del instrumento de medida y la reactancia inductiva L de la sonda, la cual se hace igual a 50 Ω aproximadamente a esas frecuencias.

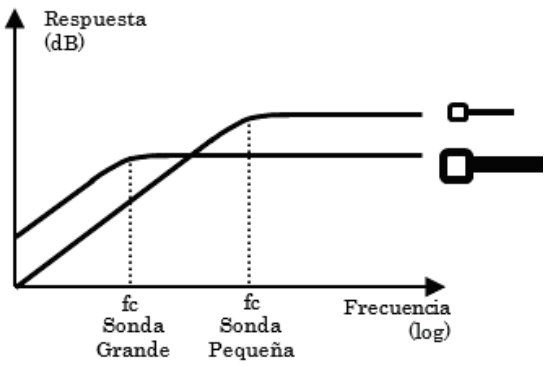


Figura 4.6: Respuesta en frecuencia sondas en función tamaño

En cuanto al comportamiento de las sondas EM a alta frecuencia, éstas se caracterizan por presentar una frecuencia de resonancia dependiente de la inductancia propia de la sonda y la capacidad parásita existente. Así, cuanto menor sea el diámetro de la sonda, mayor será la frecuencia de resonancia. No obstante, otros parámetros como el diámetro del cable y el método de construcción también influyen [Edi'09]. Por ejemplo, una sonda de 2.5 cm tiene una inductancia de aproximadamente 80 nH y su frecuencia de resonancia se produce poco después de 1 Ghz [Smi'98].

- Número de espiras: En principio las leyes de Maxwell estipulan que a mayor número de espiras, mejor es la sensibilidad de la sonda. Sin embargo, debido al efecto de la terminación de 50Ω presente normalmente en la mayoría de las sondas y a su propia inductancia, que se incrementa en un factor que depende del cuadrado del número de espiras, el comportamiento difiere de la teoría [Smi'98]. En la Figura 4.7 se muestra la respuesta en frecuencia de dos sondas en función de su número de espiras. Como se puede comprobar, a bajas frecuencias la respuesta de la sonda con dos espiras es superior, pero esto cambia a medida que aumenta la frecuencia, donde la sonda con una espira se impone.

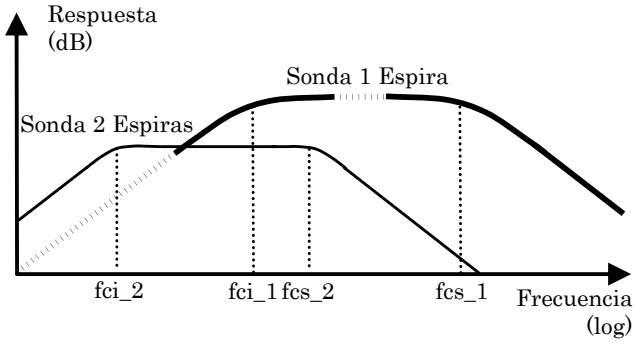


Figura 4.7: Respuesta en frecuencia sondas en función número espiras

En conclusión, será recomendable una sonda multiespira para aplicaciones donde la sensibilidad a bajas frecuencias sea un parámetro imprescindible y monoespira si se desea un gran ancho de banda a costa de una menor sensibilidad a bajas frecuencias.

4.2.4.2 Sondas Electromagnéticas utilizadas

En la práctica, y pese a que sensibilidad y resolución espacial resultan ser propiedades opuestas, ya que al mejorar una empeora la otra, fabricar un dispositivo con unas propiedades suficientes para llevar a cabo un EMA, no resulta demasiado complejo. Además, su coste económico es notablemente inferior al de una sonda comercial y los resultados que se obtienen, son similares e incluso mejores a los de sondas comerciales. Por ejemplo, en su estudio, Gandolfi y sus compañeros compararon la respuesta de varios sensores EM, llegando

a la conclusión de que los mejores resultados se obtienen con sondas hechas a mano con hilo de cobre de diámetros comprendidos entre 150 y 500 micras [Gan'01]. Se suma también el hecho de que es un dispositivo al cual se le exigen una serie de propiedades concretas necesarias para resolver un problema en cuestión, como tamaño, resolución, ganancia, ancho de banda… y que en la gran mayoría de los casos sólo se consigue con un diseño y fabricación ex profeso.

Por ese motivo, se tomó la decisión de fabricar varios tipos de sondas y comparar su comportamiento con sondas comerciales. A continuación, en las siguientes figuras se muestran las sondas implementadas:

Figura 4.8: Sonda 1: Monoespira con un clip y termorretráctil según [Smi'99]

Figura 4.9: Sonda 2: Monoespira cobre esmaltado 0.8 mm según [Smi'99]

Figura 4.10: Sonda 3: Monoespira blindada cobre rígido y termorretráctil según [Smi'00]

Figura 4.11: Sonda 4: Multiespira con semiferrita Fair-Rite TN 9/6/3-4C65 según [Joh'99]

Figura 4.12: Sonda 5: Multiespira con semiferrita Fair-Rite TX 10/6/3-4C65 según [Joh'99]

Figura 4.13: Sonda 6: Multiespira (8 espiras) cobre esmaltado 0,6 mm según [Pee'07]

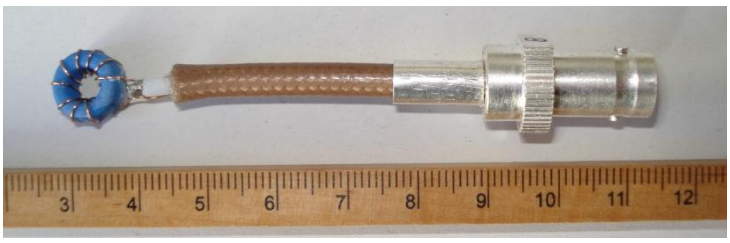


Figura 4.14: Sonda 7: Sonda multiespira con ferrita TN10/6/4-3F3 según [Rid'99]

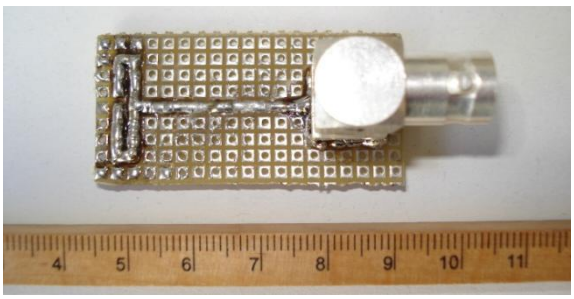


Figura 4.15: Sonda 8: Monoespira circuito protoboard según [Ost'03]

Como se puede advertir, las tres primeras sondas son muy similares en cuanto a su forma, monoespira cuadrada. Las diferencias aparecen en cuanto al material y aislante usado para su fabricación. La primera está construida a partir de un clip aislado con termorretráctil, la segunda con cobre esmaltado y la tercera con cobre rígido blindado con cinta de cobre adhesiva a la que se le ha practicado un gap y finalmente se ha aislado externamente con termorretráctil. Las sondas 4 y 5, consisten en una multiespira de cobre esmaltado, terminadas en una semiferrita toroidal. El tipo de ferrita usado supone la única diferencia sustancial entre las dos. En cuanto a la sonda 6, es también una multiespira de 8 vueltas con cobre esmaltado. La 7 es similar en concepto a 4 y 5, pero en este caso dispone de una ferrita toroidal entera. Por último la sonda 8 está fabricada mediante estaño sobre una placa protoboard de puntos.

En cuanto a las sondas comerciales con las que comparar y poder reproducir los experimentos, se han utilizado dos. La primera es una sonda monoespira EM-6995 (Figura 4.16) incluida en el kit de Electro-Metrics EM-6992 con las siguientes características [EM'07]:

Figura 4.16: Sonda EM6995 de Electro-Metrics

Tabla 4.2: Características técnicas EM-6995

EM-6995	
Impedancia	50 Ω
Diámetro	1 cm
Ancho Banda	100 Khz – 1 GHz

La otra sonda comercial es la novedosa[24] MFA-K 0,1-12 (Figura 4.17), incluida en el set MFA 01 de Langer EMV-Technik. Ésta, se caracteriza por disponer de una micro-cabeza activa con una resolución espacial de 300 μm, específicamente diseñada para realizar medidas sobre elementos muy pequeños como componentes, circuitos y pines. La sonda incluye un pequeño amplificador integrado situado al lado de la cabeza sensora, alimentado a través de un dispositivo denominado Bias-tee [LanMFA'10]. Sus características técnicas se incluyen en la siguiente tabla:

[24] Se comenzó a comercializar en 2010

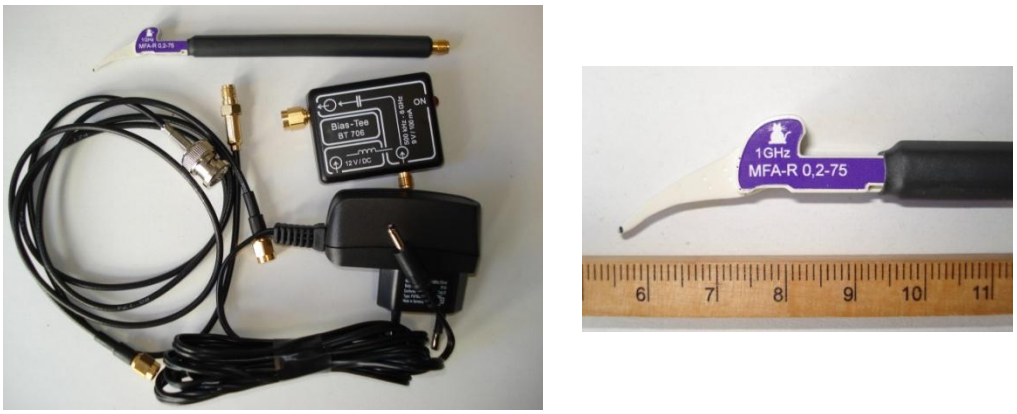


Figura 4.17: Sonda MFA-R 0,2-75 de Langer EMV-Technik

Tabla 4.3: Características técnicas MFA-K 0,1 – 12

MFA-R 0,2-75	
Impedancia	50 Ω
Resolución	300 μm
Ancho Banda	1 Mhz – 1 Ghz

También se estudió la posibilidad de usar cabezas lectoras de disco duro como sondas, tal como propone Gandolfi y otros en su artículo [Gan'01]. Para ello, se analizaron varias cabezas lectoras de discos duros vetustos, puesto que en general el tamaño de su sonda lectora, ya de por sí pequeño, es mayor al de dispositivos novedosos. Aun así, los resultados obtenidos no fueron satisfactorios.

4.2.4.3 Caracterización Sondas Electromagnéticas

Para la realización del trabajo se decidió utilizar las dos sondas comerciales descritas, con un comportamiento contrastado, y dos sondas fabricadas artesanalmente. Para seleccionar las sondas más adecuadas se realizaron dos pruebas.

La primera prueba es similar al test de susceptibilidad EMI que deben superar los equipos electrónicos que son comercializados (norma UNE-EN 61000-4-3 [UNE'07]). Consiste en crear en una zona del espacio, un campo EM constante de valor 0.1 V/m para un rango de frecuencias comprendido entre 5 MHz y 2 GHz.

Para conseguir tal fin, se conecta un generador RF a una antena logoperiódica, y se barre el espectro de frecuencias, generando un campo magnético constante de 0,1 v/m en un punto del espacio situado a una distancia de 60cm. de la antena. En ese punto se sitúa la sonda bajo estudio conectada a un Receptor EMI y se registra el campo captado durante el barrido. La Figura 4.18 detalla el setup utilizado. Dado que la antena usada para generar el campo no tiene una respuesta lineal, es necesario realizar previamente una calibración del setup.

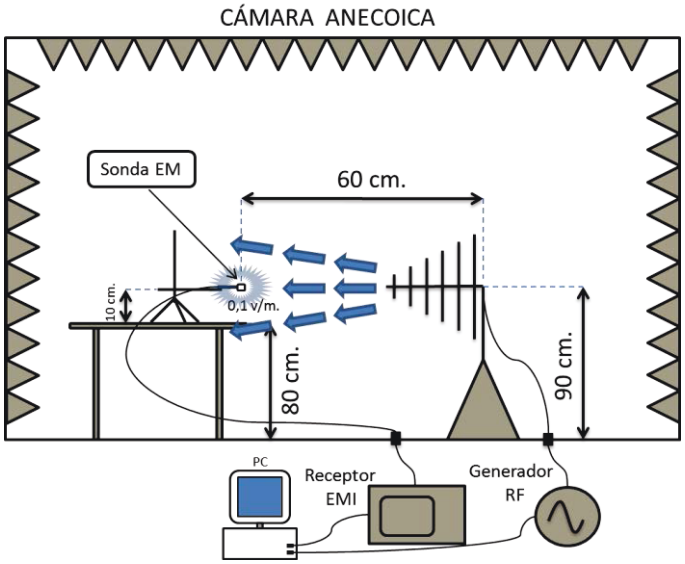


Figura 4.18: Test setup para la caracterización de sondas campo cercano

El resultado de esta prueba es una gráfica que determina la respuesta en frecuencia de la sonda en el espectro medido. A continuación se muestran los resultados obtenidos para cada una de las sondas. Se ha incluido la sonda EM6995 para comparar su comportamiento con el del resto de sondas fabricadas, aunque se debe tener en cuenta que, según especificaciones del fabricante, su respuesta en frecuencia alcanza 1 GHz, y el test se realiza hasta 2 GHz.

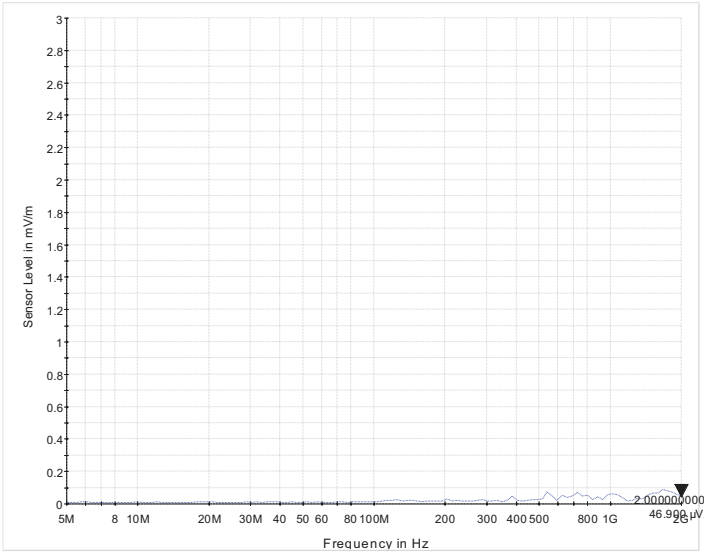


Figura 4.19: Respuesta en frecuencia Sonda EM6995

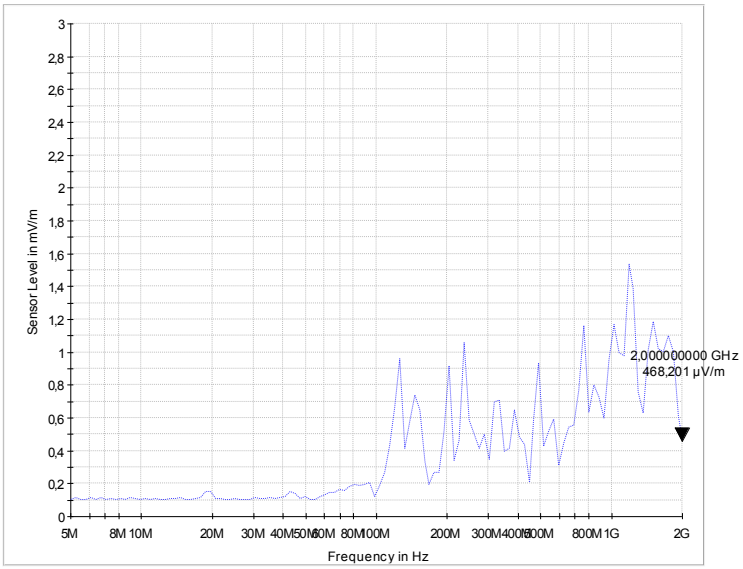


Figura 4.20: Respuesta en frecuencia sonda 1

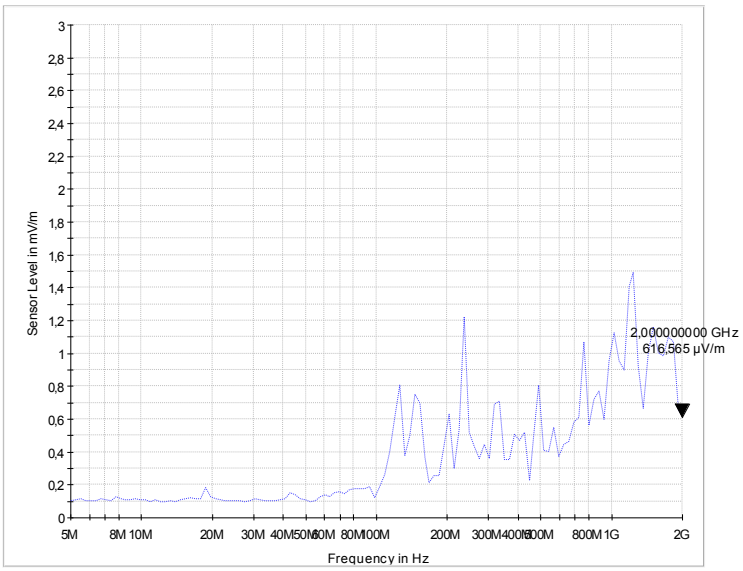


Figura 4.21: Respuesta en frecuencia sonda 2

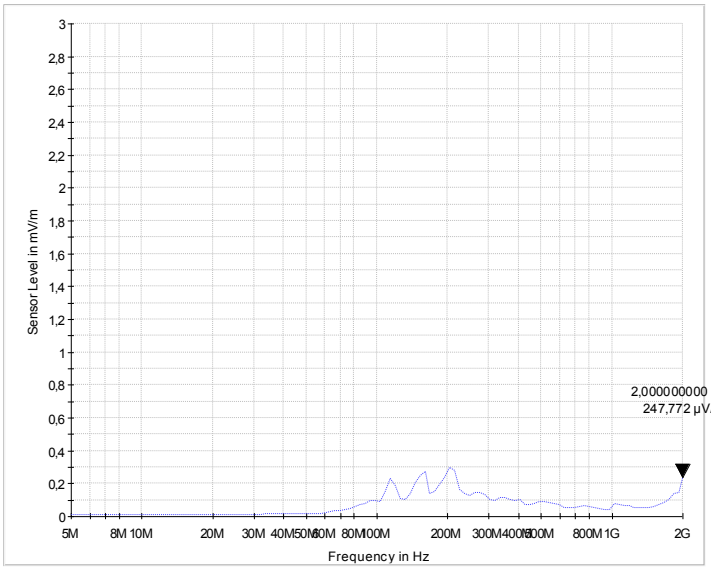


Figura 4.22: Respuesta en frecuencia sonda 3

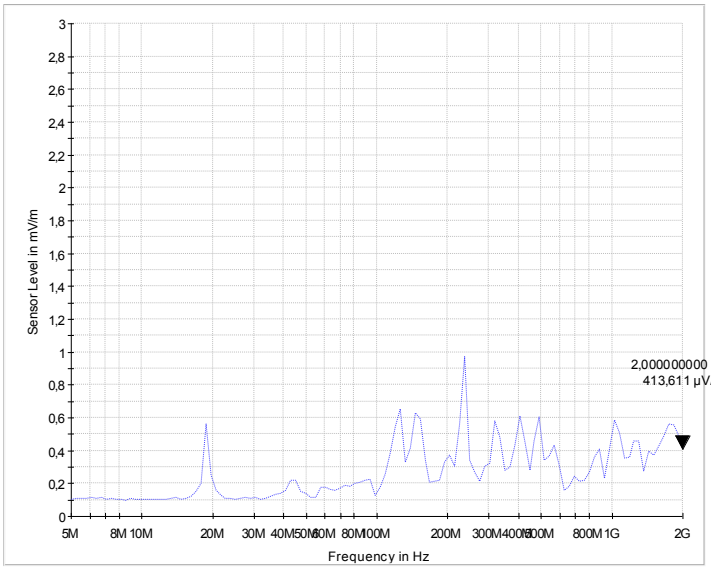


Figura 4.23: Respuesta en frecuencia sonda 4

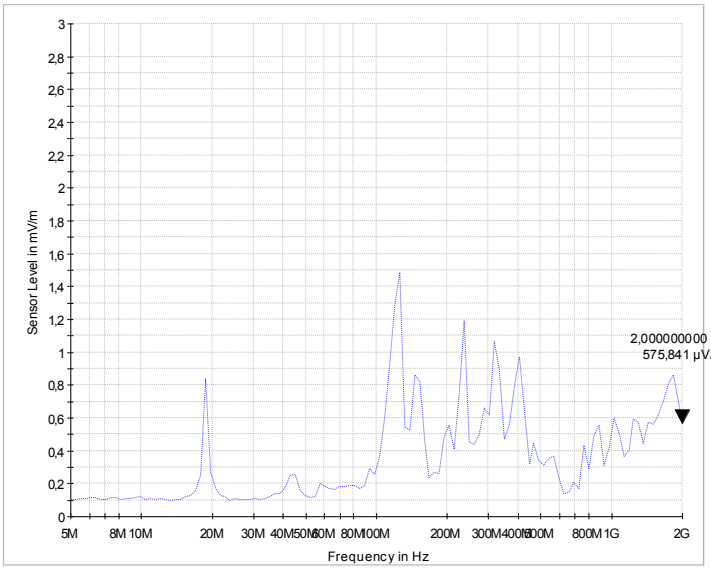


Figura 4.24: Respuesta en frecuencia sonda 5

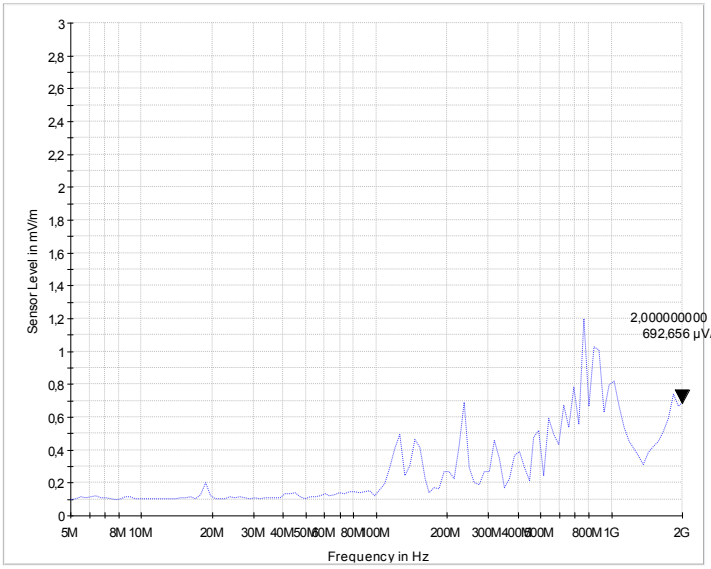


Figura 4.25: Respuesta en frecuencia sonda 6

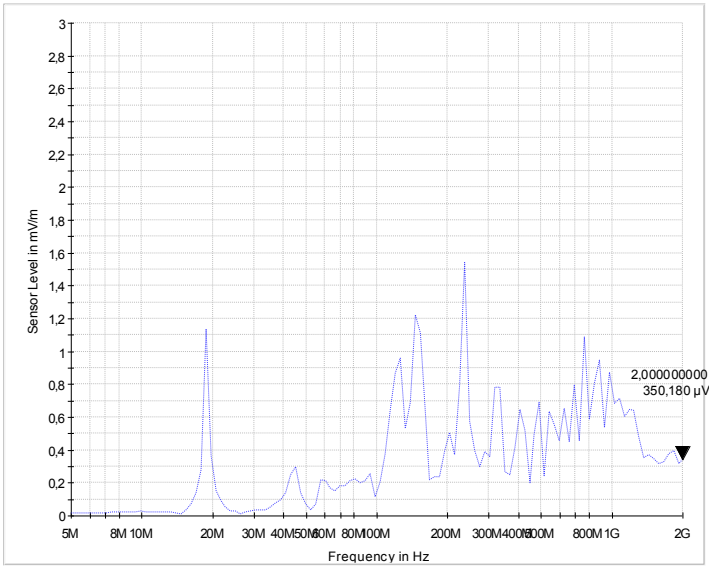


Figura 4.26: Respuesta en frecuencia sonda 7

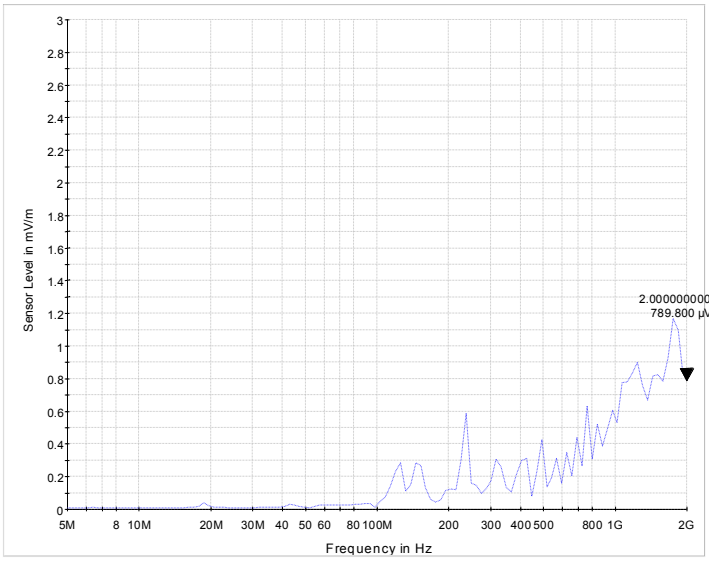


Figura 4.27: Respuesta en frecuencia sonda 8

Como se puede verificar, la respuesta de las sondas, incluida la EM6995 comercial, no es notable. En términos generales, exceptuando resonancias, se miden valores 1000 veces menores y las resonancias comienzan a aparecer a partir de los 100 MHz. Algunas sondas muestran una resonancia en torno a 18 MHz, si bien el origen no está claro.

De este test se desprenden varias conclusiones:

- Las sondas EM6995, 3, 7 y 8 disponen de una ganancia muy pequeña.
- Las sondas 4, 5 y 7, tienen comportamiento más desfavorable con la presencia de una resonancia a 20M.

- Las sondas 1 y 2 tienen una respuesta muy parecida, ya que básicamente solo se diferencian por el material con el que están fabricadas.

Dado que los resultados obtenidos no arrojaron una conclusión definitiva sobre qué sondas utilizar, se decidió realizar un segundo test. Éste consiste en medir el campo generado por la placa EMA 1 del estudio, al ejecutar el algoritmo de encriptación AES de forma cíclica. Para ello, se sitúa la sonda bajo estudio sobre la placa y con la ayuda de un receptor EMI se mide el espectro de la señal captada para el rango de frecuencias 1 MHz - 2 GHz, tal como se muestra en la Figura 4.28.

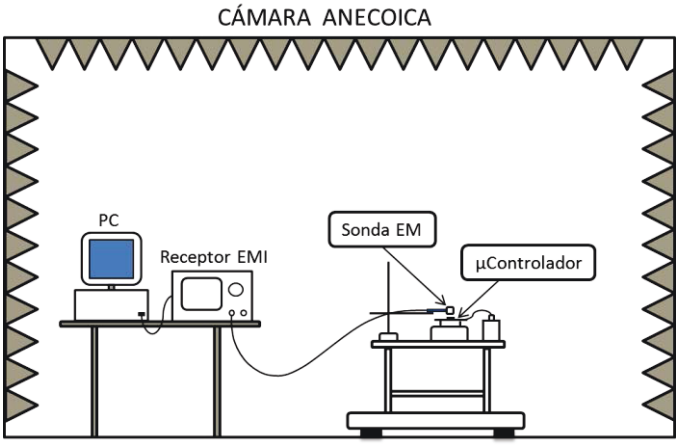


Figura 4.28: Setup medida espectro EM EMA1

Estos son los resultados obtenidos:

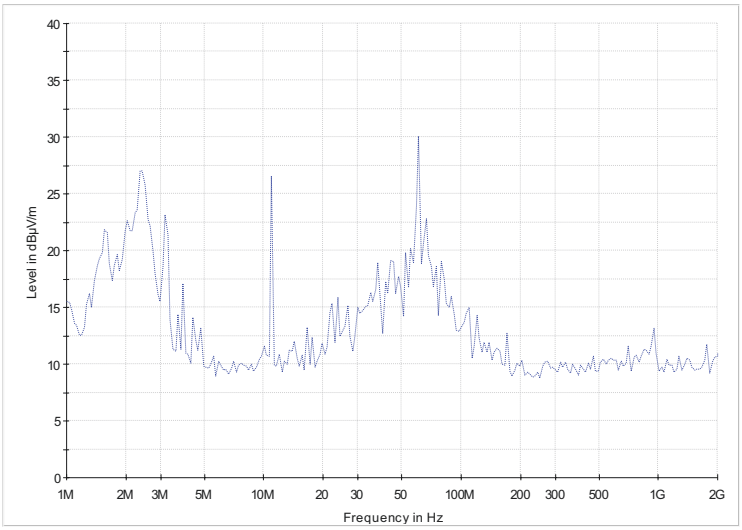


Figura 4.29: Medida Espectral Sonda 1

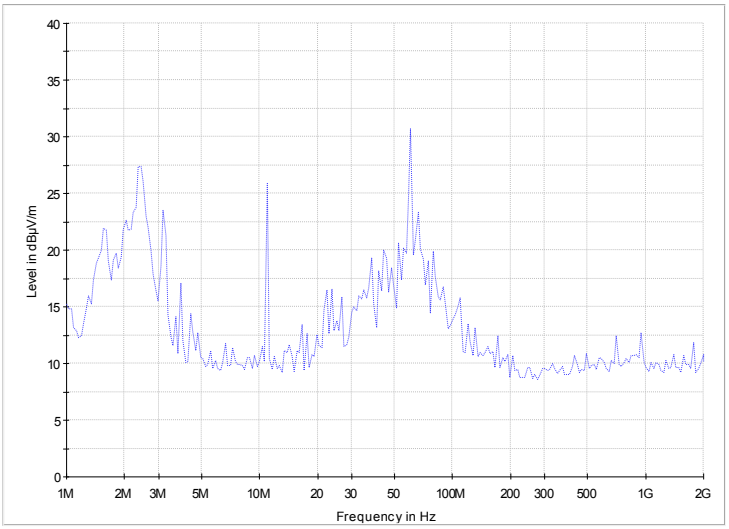


Figura 4.30: Medida Espectral Sonda 2

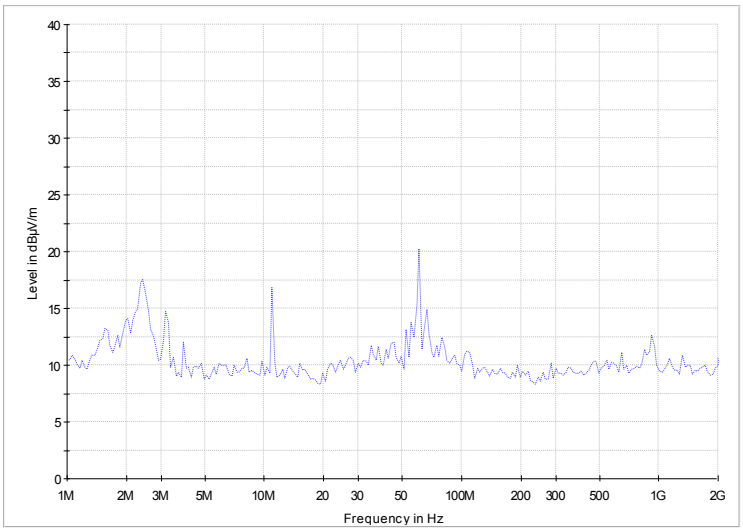


Figura 4.31: Medida Espectral Sonda 3

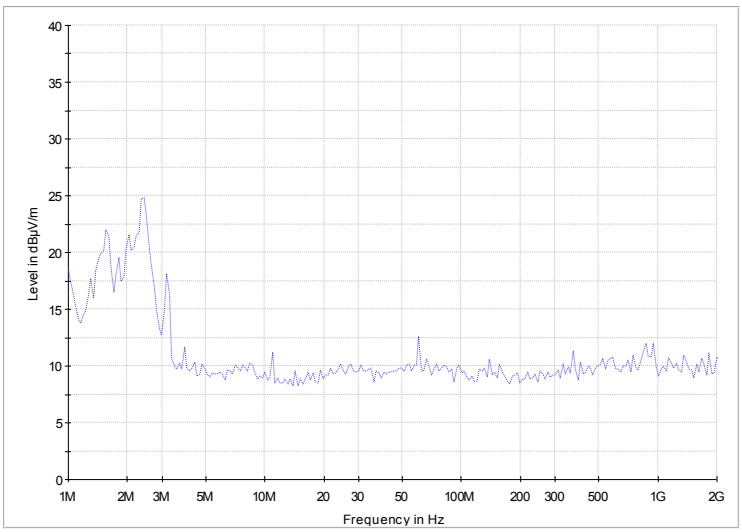


Figura 4.32: Medida Espectral Sonda 4

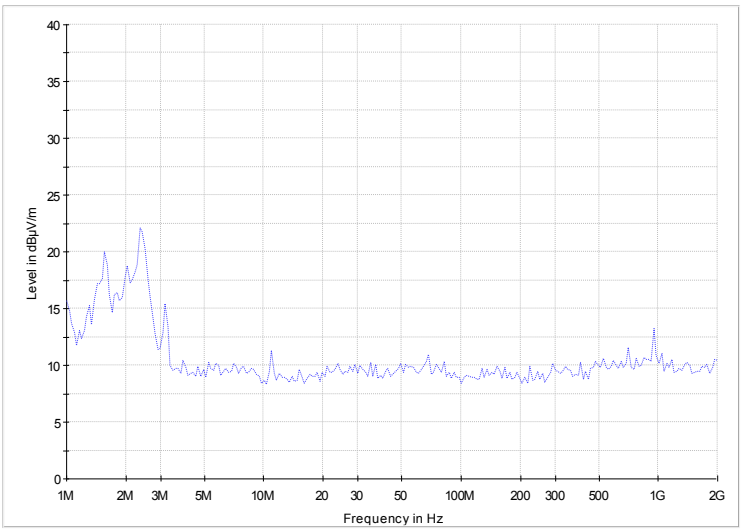


Figura 4.33: Medida Espectral Sonda 5

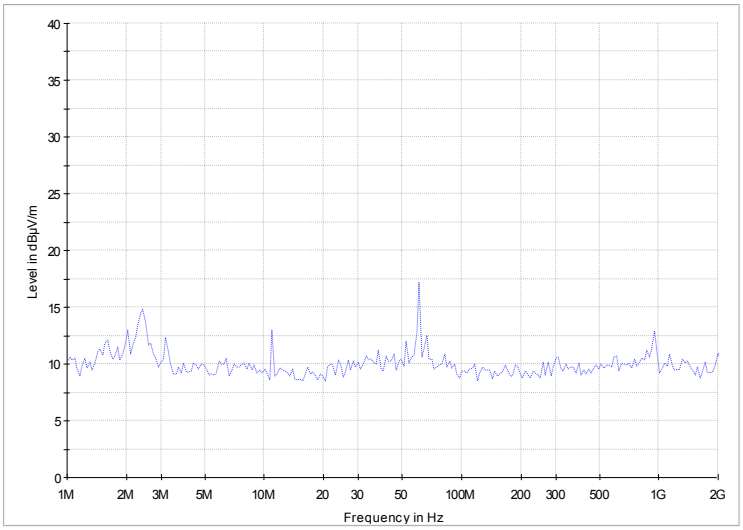


Figura 4.34: Medida Espectral Sonda 6

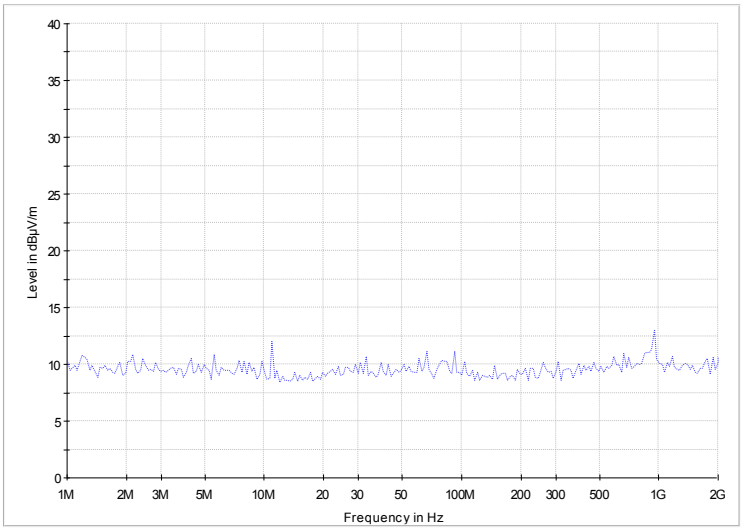


Figura 4.35: Medida Espectral Sonda 7

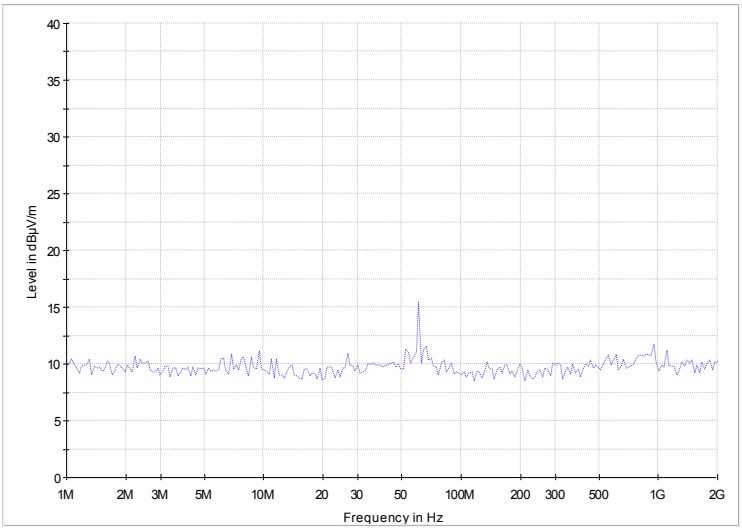


Figura 4.36: Medida Espectral Sonda 8

Como se constata, la mayoría de las sondas captan la frecuencia principal de la placa, 11 MHz, y un armónico de aproximadamente 60 MHz. Las sondas 1 y 2, consiguen captar un nivel superior, seguidas por la 3 y 6.

Analizando todos los resultados, se llegó a la conclusión de que las sondas más apropiadas son la 1 y 2. Al disponer de una mayor ganancia, es previsible que las señales captadas tengan una SNR mayor. No obstante, dado que ambas tienen un comportamiento muy similar, se decidió utilizar únicamente la sonda 2 fabricada con hilo esmaltado de cobre, pues su ganancia es un poco superior, debido a la utilización de cobre como conductor y, en parte, al reducido grosor de su aislante, lo que permite acercar más la sonda al EUA.

4.2.4.4 Soporte Sondas

Un elemento a tener en cuenta es el soporte de las sondas para la realización de las medidas. La elección ideal, es utilizar un posicionador milimétrico, que permita una situación precisa de la sonda y asegure una repetitividad de las medidas. Pero dado su alto coste y el hecho de que esté fabricado en metal, se decide no utilizarlo. En su lugar, se diseña y fabrica un soporte novedoso (Figura 4.37), utilizando materiales no metálicos como madera y plástico. Con la ayuda de una ventosa se consigue situar la sonda en el lugar deseado.

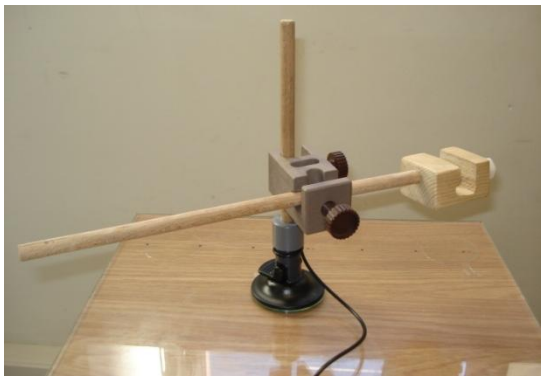


Figura 4.37: Soporte sondas setup medida

4.2.5 PreAmplificador

El amplificador de banda ancha es también uno de los elementos principales en cualquier cadena de medida de un EMA. De él depende la calidad de la señal que es posible detectar con el osciloscopio.

Como se ha indicado previamente, a su entrada llega la señal captada por la sonda EM de medida; En este caso, a través de un cable BNC 50Ω de 1.8 metros de longitud y doble apantallamiento (modelo Pomona 2249-Y-72 [Pom'10]), que asegura la ausencia de ruido; Y su salida se conecta directamente a una de las entradas del osciloscopio.

En este trabajo se ha utilizado un preamplificador Langer PA303 [LanPA'08], Figura 4.38. Los tres parámetros más importantes que caracterizan a un amplificador son: ganancia, ancho de banda y figura de ruido.

La ganancia debe ser suficiente para excitar el equipo que va a efectuar la digitalización. Un valor típico suele ser 20 dB. En este caso, el amplificador seleccionado dispone de 30 dB.

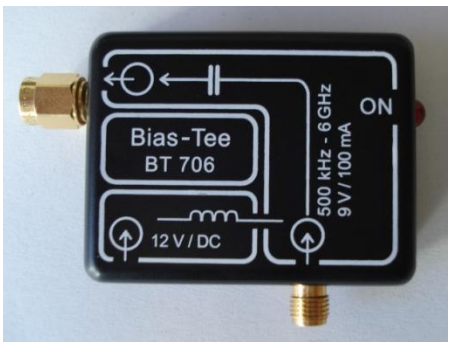


Figura 4.38: Preamplificador Langer PA 303

El ancho de banda debe ser suficientemente grande para poder trabajar en un amplio margen de frecuencias. Para un EMA o PA, al menos debe ser capaz de llegar hasta una frecuencia de 1 Ghz. El dispositivo utilizado cubre ampliamente ese rango como se puede

verificar en su respuesta en frecuencia de la Figura 4.39, y permite medir señales desde 100 KHz hasta 3 GHz con una respuesta prácticamente plana.

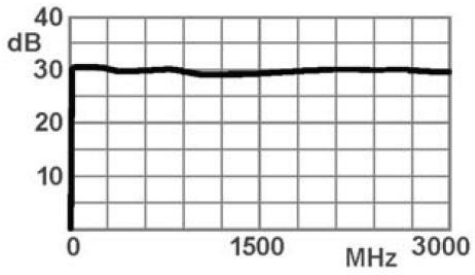


Figura 4.39: Respuesta en frecuencia amplificador Langer PA 303

La figura de ruido indica la calidad del amplificador. En otras palabras, la capacidad para discernir señales de muy bajo nivel. Cuanto más bajo sea este indicador mejor será el amplificador. El Langer PA 303 usado posee una figura de ruido de 4.5 dB.

4.2.6 PC

Como ya se ha indicado, un CEMA se compone de dos fases. Una primera de captura de trazas EM y una segunda fase de análisis estadístico. El PC es el coordinador o centro neurálgico de este tipo de ataques, prácticamente imprescindible, debido al alto número de trazas a registrar y la notable cantidad de datos a computar.

En el setup de medida implementado desempeña las siguientes funciones:

- Fase 1: Captura de Datos
  - Genera previamente los textos aleatorios a encriptar.
  - Coordina la cadena de medida, controlando remotamente el osciloscopio y enviando la señal de inicio de encriptación al EUA.
  - Almacena las trazas capturadas por el osciloscopio.
- Fase 2: Análisis de Datos:
  - Realiza los cálculos necesarios para llevar a cabo el ataque.
  - Muestra los resultados gráficamente.

4.2.6.1 El PC en la fase de captura de datos

Previamente a la captura de datos, el PC se utiliza para generar los textos aleatorios de 16 bytes que el EUA cifrará durante el ataque y que serán almacenados en su memoria ROM.

En cuanto al setup para la captura de datos, inicialmente configura el osciloscopio según las necesidades específicas del ataque a realizar (nivel y tipo de trigger, longitud de medida, número de medidas etc.) y lo mantiene a la espera de la señal de trigger proveniente

del EUA que le indique la traza EM a capturar. A continuación, envía una señal de comienzo de encriptación al EUA. Éste proporciona la señal de disparo al osciloscopio, realiza el cifrado del primer texto almacenado en su memoria de programa y se mantiene a la espera de la siguiente señal de cifrado para encriptar el siguiente texto. Después de estos pasos, el Pc solicita la señal capturada por el osciloscopio y la almacena en memoria. Aumenta el contador de número de textos cifrados y repite el procedimiento. Este proceso lo repite hasta cifrar el número de textos almacenados en la memoria del EUA que depende de la capacidad de su memoria ROM.

Para realizar estas funciones es necesario implementar un programa específico que las realice. Para este fin, se valoraron tres software de control de equipos:

- Instrumentation Control ToolBox de Matlab R2010 [MathIns'10].
- Labview 8.5 [NILab'07].
- TekVisa Connectivity v3.3.3 [Tek'09][25].

Dado que las posibilidades de control y personalización que permitían los software de Matlab y TekVisa, en el momento de la realización del estudio, eran muy inferiores a las de Labview, se tomó la decisión de utilizar este último, junto con los drivers de control proporcionados por el fabricante para dicho software: Instrument Driver Tektronix DPO MSO 2000 4000 Series v.4.1.1 [NILab'09c].

4.2.6.1.1 Generación de textos aleatorios

Para la generación de los textos aleatorios de 16 bytes se ha desarrollado una aplicación en Labview con tres variantes, una para cada tipo de microcontrolador, pues el código a generar es distinto en cada caso, ya que depende del lenguaje de programación (ensamblador o ANSI C) y del entorno de desarrollo utilizado:

8051:

```
DB     0E3H, 0D4H, 0EAH, 06EH, 091H, 02CH, 053H, 0E1H, 0E3H, 011H, 091H, 056H, 00AH, 042H, 0B8H, 01AH, \
       0F7H, 0BEH, 0F4H, 043H, 03EH, 092H, 00FH, 059H, 0A9H, 029H, 07CH, 0AAH, 068H, 0BBH, 0B8H, 071H, \
```

ARM7 y Cortex LPC1769:

```
       {0xE4, 0x57, 0x89, 0x74, 0x2D, 0x8A, 0x01, 0xFA, 0x15, 0x4E, 0xA0, 0x4A, 0xA8, 0xBD, 0x61, 0x62},
       {0xA3, 0xBF, 0x16, 0xCA, 0x7A, 0x69, 0xC8, 0x45, 0x90, 0xEB, 0xA8, 0x4B, 0x33, 0x64, 0xB4, 0xA2},
```

ARM Cortex STM32:

```
       0x47, 0xF6, 0xAF, 0xF3, 0x8B, 0xB8, 0xF4, 0x96, 0x56, 0x82, 0xEB, 0x9B, 0x26, 0xAA, 0x7B, 0xA7,
       0xAB, 0xEB, 0x87, 0xD6, 0x05, 0xEB, 0x24, 0x6A, 0xFA, 0x13, 0x86, 0xCF, 0x38, 0xA6, 0xCF, 0xBF,
```

[25] Consiste en un software de control específico del fabricante del osciloscopio.

La interfaz de la aplicación es muy sencilla (Figura 4.40). Únicamente se indica el número de textos a generar, la dirección donde se creará el archivo con los textos y el formato del archivo: "txt" o "csv".

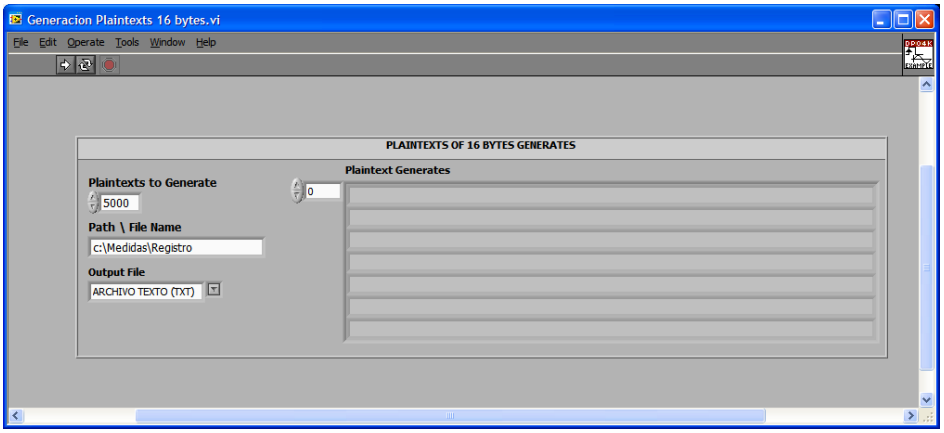


Figura 4.40: Interfaz aplicación Labview para la generación de textos planos

4.2.6.1.2 Control de Cadena Medida

En cuanto al software de control de la cadena de medida del CEMA, consta de una interfaz gráfica a través de la cual es posible controlar distintos parámetros para la captura de la trazas EM:

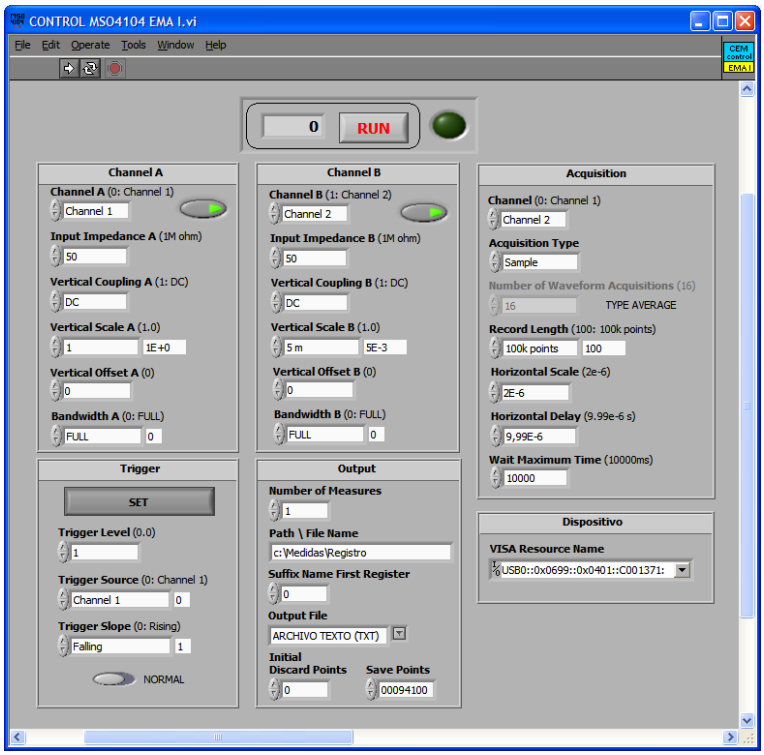


Figura 4.41: Interfaz aplicación Labview para el control de la cadena de medida

Esta aplicación permite configurar dos de los cuatro canales disponibles del osciloscopio, estableciendo diversos parámetros, como su impedancia de entrada, escala

vertical, acoplamiento, ancho de banda y offset. Uno de ellos recibirá la señal de trigger proveniente del EUA, y el otro la señal EM capturada por la sonda previamente amplificada, por lo que la aplicación permite establecer otros parámetros:

- Señal de Trigger: Se debe definir el tipo de disparo (normal o automático), el nivel, la detección por flanco de subida o bajada y el canal que actuará como tal.
- Señal EM: Se debe seleccionar el canal y tipo de adquisición (muestreo, detección de picos, alta resolución, envolvente o promediado), longitud de registro (1k, 10k, 100k, 1M o 10M), escala horizontal, tiempo de retardo y tiempo de espera máximo.

Por otro lado, también es posible configurar todos los aspectos relacionados con los datos de salida: número de medidas[26], nombre y dirección de los archivos resultantes, tipo de archivo (.txt o .csv), puntos iniciales de la traza a descartar, número de puntos de la traza que se guardarán y sufijo del primer archivo creado[27].

El proceso de configuración requiere conocer previamente el tiempo de ejecución del AES en el microcontrolador bajo ataque, para en función de este parámetro, establecer otros como la longitud del registro, escala horizontal, tiempo de retardo, puntos a salvar etc. Para ello, se puede modificar provisionalmente el código del EUA e incluir una señal de trigger adicional que indique el fin de la ejecución del algoritmo de encriptación y permita medir su duración.

4.2.6.1.2.1 Señal de Encriptación

Como se ha explicado previamente, durante el ataque el EUA permanece a la espera, y únicamente cifra el siguiente texto almacenado en su memoria cuando recibe una señal de inicio de encriptación del PC. Para la generación de esta señal se recurrió a un cable comercial conversor de USB a TTL, denominado TTL-232R [FTDI'10], Figura 4.42.

Utilizando dos de sus pines (RQS y tierra) y los drivers proporcionados por el fabricante para el software Labview, se consigue generar una onda cuadrada, que el EUA interpreta como señal de inicio de la encriptación.

[26] El número de medidas deberá concordar con el número de textos almacenados en el EUA.
[27] Por defecto, el programa nombra a los archivos con un sufijo que indica el número de medida, empezando por 0. Esta opción, permite establecer un sufijo inicial distinto de cero, para así poder concatenar medidas sucesivas.

Figura 4.42: Cable TTL-232R para la señal de encriptación

4.2.6.2 El PC en la fase de análisis de datos

Una vez almacenadas las trazas EM en memoria, es necesario realizar un estudio estadístico y presentar los resultados gráficamente. Se requiere, por tanto, un software matemático. En este caso, se sopesaron dos alternativas:

- Matlab R2010 [MathMat'10].
- R 2.11.1 [R'10].

Finalmente, se decidió usar Matlab, pues además de conocer su gran eficiencia, se contaba con experiencia previa de uso, al contrario que R.

Para tal fin, se han creado dos aplicaciones gráficas con Matlab GUI [MathGui'10], incluida en el paquete Matlab R2010. La primera, cuya interfaz se muestra en la Figura 4.43, permite leer las trazas de memoria y ejecutar un CEMA utilizando los modelos de consumo HW y HD, tal como se describe en el epígrafe 3.4.

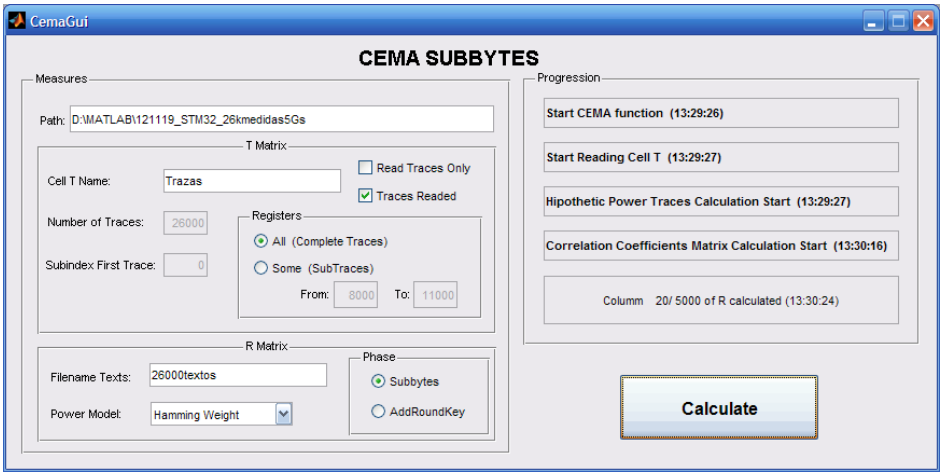


Figura 4.43: Interfaz aplicación CemaGui para la realización de un ataque CEMA

La ejecución de la aplicación requiere indicar varios parámetros, como la dirección que contiene los archivos con las trazas EM, nombre de los archivos sin el subíndice, número de

archivos o trazas EM, textos que han sido encriptados durante el ataque, modelo de consumo a utilizar (HW o HD) y fase del algoritmo AES que se desea atacar.

Aparte de ello, la aplicación tiene implementadas varias funcionalidades como son:

a) Leer la totalidad o parte de los registros de las trazas guardadas en memoria del osciloscopio[28].

b) Seleccionar la función o funciones que se desean ejecutar: leer las trazas de memoria para obtener la matriz T, realizar el análisis estadístico a partir de la matriz T o el test completo. Al seleccionar una de las funciones varían los datos solicitados, y los que no son necesarios se deshabilitan, tal como se muestra en la Figura 4.43. En ese caso se ha seleccionado realizar únicamente el análisis estadístico, y por esa razón solicita el nombre de la matriz T, y parámetros como el número de trazas o el subíndice de la primera traza se han deshabilitado.

c) Informar al usuario de la evolución del ataque durante su ejecución, puesto que, en función de la longitud del registro, puede demorar varias horas.

Al finalizar, el programa muestra la clave que genera una mayor correlación y el registro en el que se produce, lo que permite conocer el momento en el que la clave es usada en el algoritmo. Así mismo, presenta el C.C. obtenido en cada uno de los registros para la clave con mayor C.C., con una gráfica como la siguiente:

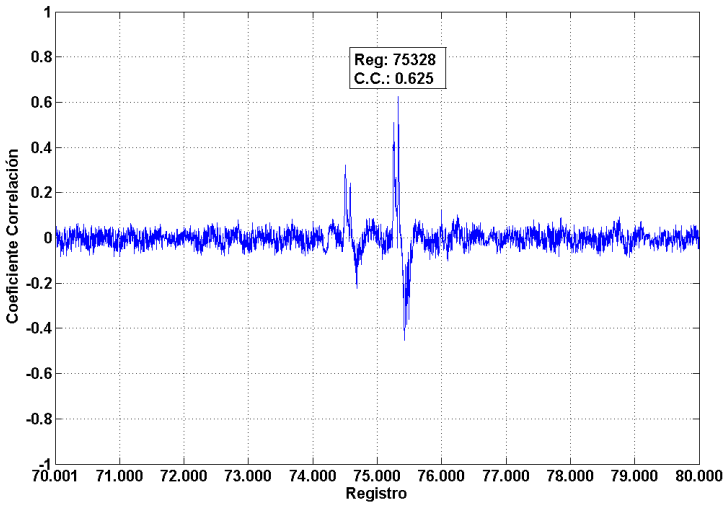


Figura 4.44: Gráfica C.C. – Registro para la clave con mayor Coeficiente de Correlación

[28] En algunos casos, en función del número de registros y trazas, se puede dar el caso de que el PC no tenga suficiente capacidad para computar la totalidad de los registros de las trazas, por lo que será necesario dividir la traza en varias partes.

El objetivo de la segunda aplicación desarrollada, Figura 4.45, es la obtención de la gráfica Coeficiente de correlación – número de medidas, que normalmente se calcula para el registro que genera mayor C.C. obtenido con la aplicación anterior CemaGui. Pese a ello, se ha implementado la posibilidad de determinar la gráfica para varios puntos si las prestaciones del equipo informático y el número de medidas a analizar lo permiten.

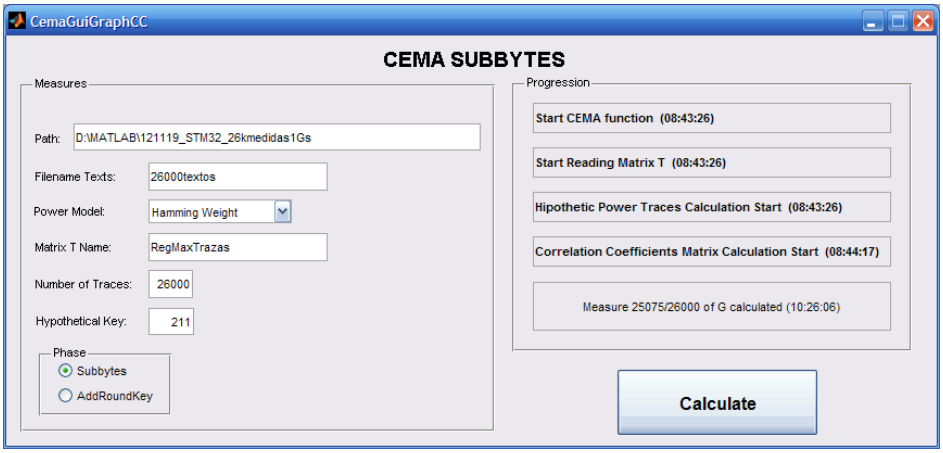


Figura 4.45: Interfaz aplicación CemaGuiGraphCC para la obtención de la gráfica CEMA-Medidas

La gráfica obtenida permite conocer la evolución del C.C. en función del número de textos o trazas capturadas para su cálculo, para cada una de las 256 claves posibles. Con ella es posible visualizar de forma gráfica el número de medidas necesarias para poder deducir la clave. Por ejemplo en la Figura 4.46 se aprecia cómo a partir de la traza 286 aproximadamente, el C.C. de la clave correcta comienza a desmarcarse del resto y el ataque comienza a ser efectivo.

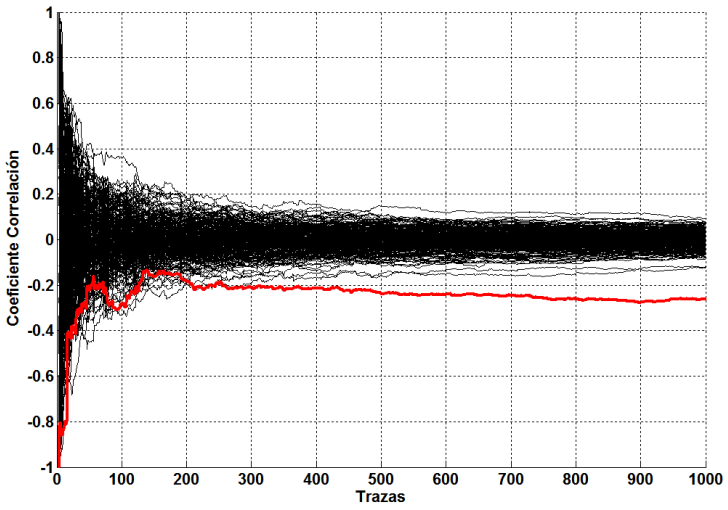


Figura 4.46: Gráfica Coeficiente Correlación – Número de Trazas

En este punto cabe decir, que la repetitividad de las medidas realizadas para un CEMA, depende directamente del ruido presente. Puesto que éste no se puede eliminar de

forma total, siempre existirá cierta variabilidad. En otras palabras, dos ataques consecutivos sobre un mismo dispositivo en igualdad de condiciones, darán resultados parecidos pero nunca exactamente iguales. Por ese motivo, a la hora de determinar un valor, éste se deberá fijar con un cierto nivel de confianza para asegurar que se cumplirá en circunstancias similares.

4.2.7 Equipo Bajo Ataque (EUA)

El equipo bajo ataque en este estudio va a consistir en uno de los microcontroladores detallados en el epígrafe 3.5. Respecto a éste, se han analizado varios elementos importantes.

4.2.7.1 Comunicación Equipo Bajo Ataque - PC

En la literatura es común el uso de una comunicación RS232 para interconectar PC y EUA [[Aig'00], [Pop'07], [Man'07], [Rea'09], [Mat'10], [Hay'12]]. Este sistema se utiliza como medio para transmitir al PC el texto cifrado previamente, recibir el siguiente texto a cifrar e indicar el momento de comienzo de la encriptación del siguiente texto.

En este estudio se sopesó la idea de utilizar una interfaz serie 232 como medio de comunicación entre PC y EUA, y por ese motivo se implementó en la placa EMA 1, aunque finalmente no se utilizó. La idea se desechó principalmente porque el conversor de niveles es un elemento muy ruidoso que puede dificultar las medidas[29] y, dado que se dispone de memoria suficiente en el EUA para almacenar los textos, no se utilizó. En su lugar se almacenaron los textos a encriptar en la memoria ROM del EUA y se utilizó un cable conversor USB-TTL para la sincronización con el PC.

4.2.7.2 Señal de Reloj

En [Man'07] los autores recomiendan la utilización de una señal de reloj sinusoidal, para reducir la cantidad de ruido presente en las medidas. Esto exige el uso de un generador de onda de alta calidad y la adición de un conector extra. En el afán de hacer el sistema lo más eficiente y sencillo posible, se decidió usar una señal cuadrada a partir de cristal de cuarzo más estable que la red RC interna de algunos microcontroladores.

4.2.7.3 Señal de Trigger

A la hora de formalizar un ataque, tan importante es que la señal de reloj sea estable como que las señales captadas estén completamente alineadas en el tiempo. Una desalineación de las trazas supone una disminución notable de la efectividad del mismo, tanto es así, que supone la base de numerosas contramedidas software implementadas [Geb'08b].

[29] En [Man'05a] los autores aíslan ópticamente la parte del circuito dedicada al interfaz RS232 para evitar posibles acoplamientos.

La sincronización de las medidas se puede conseguir a través de una referencia o utilizando diferentes técnicas, como:

- Identificación de patrones [Hom'06],
- Análisis en el domino de la frecuencia [Tiu'05],
- Integración [Cla'00] etc.

En este estudio se ha utilizado una referencia consistente en una señal de trigger cuadrada generada por el propio EUA. Con esta técnica, las desalineaciones presentes tienen su origen principalmente en tres factores, la arquitectura del EUA, el ruido presente y el mecanismo de trigger del osciloscopio [Tiu'05].

En cuanto a la arquitectura poco se puede hacer para mejorar la sincronización. En general, los lenguajes de alto nivel introducen más desalineaciones que los de bajo nivel. En cuanto al ruido, como ya se ha visto, se han tomado diversas medidas para atenuarlo: el uso de una batería, cámara anecoica, cables apantallados etc. Tan solo queda mejorar la sincronización por la vía del mecanismo de trigger. Por tal motivo se decidió profundizar en dicho elemento, analizándolo en detalle.

4.2.7.3.1 Estudio de la generación de la Señal de Trigger

El objetivo de este análisis es tratar de conseguir una sincronización lo más ideal posible, que permita captar siempre la misma porción de la señal EM.

Por tanto, en primer lugar, se plantea la necesidad de generar una señal de trigger con la mayor calidad posible, esto es, lo más afín a una señal cuadrada perfecta. Para ello, se toma la placa EMA 2 y se programa un software cíclico que genera dos señales cuadradas de 100 µs cada una, a través de uno de sus puertos de salida. El flanco de bajada del primero indicaría el comienzo de la zona de interés y el flanco de subida del segundo pulso el fin. Posteriormente se diseña una aplicación en Labview que permite analizar estadísticamente la posición de disparo del osciloscopio Tektronix MSO4104 para cada uno de los pulsos, Figura 4.47. Ésta almacena la posición de los disparos generados para el número de iteraciones seleccionado, y los analiza estadísticamente calculando los puntos máximos y mínimos, variancia, desviación típica y media.

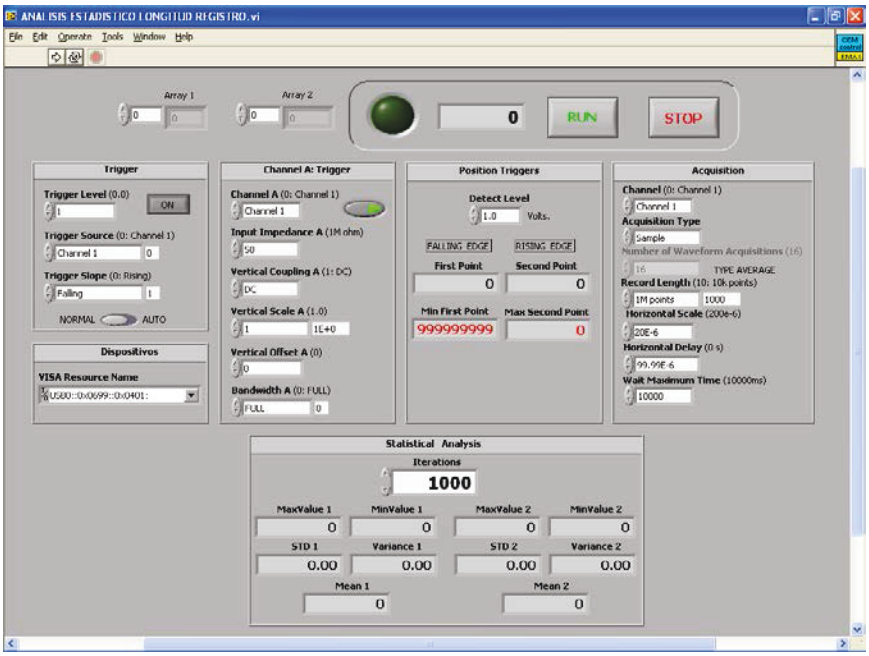


Figura 4.47: Aplicación Labview para el análisis de la señal de trigger

Con la ayuda de esta aplicación se estudia la calidad de la señal de trigger generada con diversas configuraciones hardware que se implementaron:

Configuración 1: Consiste en conectar al puerto de la placa EMA 2 la base de un transistor, tal como aparece en la figura, de forma que se puede controlar la saturación y corte del transistor en función de la salida del puerto:

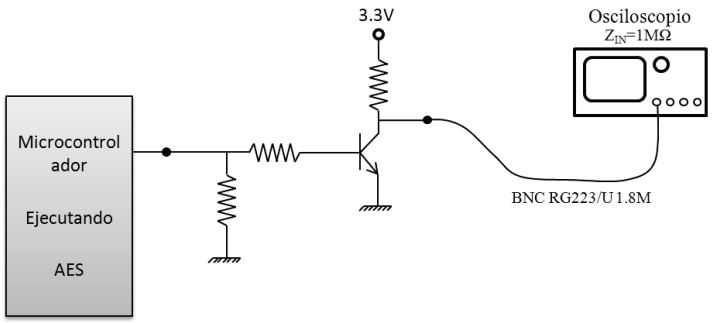


Figura 4.48: Configuración 1 para la generación de trigger

Configuración 2: Se añade a la configuración 1 un inversor schmitt trigger M74HC14 de alta velocidad [ST'01].

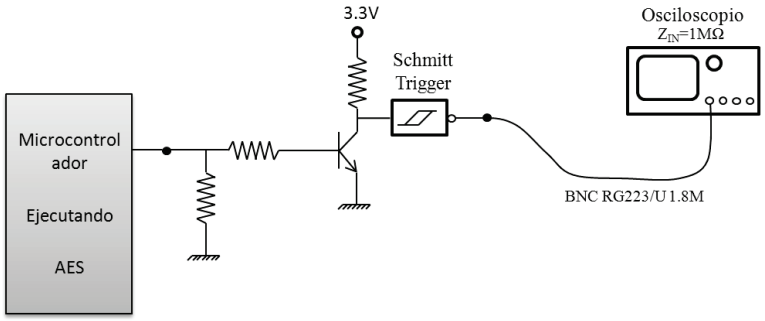


Figura 4.49: Configuración 2 para la generación de trigger

Configuración 3: Se basa en conectar un inversor schmitt trigger M74HC14 de alta velocidad, a la salida del puerto del microcontrolador. Si se analiza el sistema, se observa que éste se conecta a un osciloscopio a través de un cable BNC cuya impedancia característica es 50Ω. Por otro lado, el osciloscopio está configurado por defecto con una impedancia de entrada de 1MΩ. Existe, pues, una desadaptación entre las impedancias que puede provocar distorsiones en la señal, por lo que se investiga también el efecto que tiene la impedancia de entrada seleccionada en el osciloscopio.

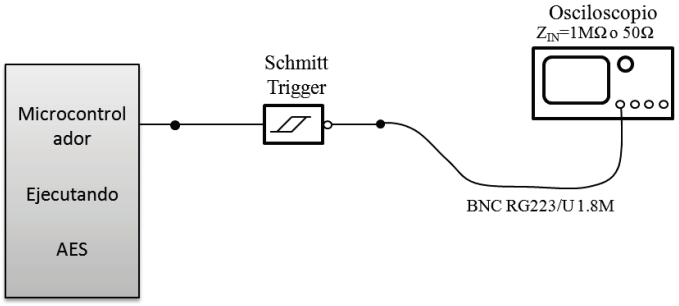


Figura 4.50: Configuración 3 para la generación de trigger

Configuración 4: Se analiza el efecto que tiene usar varios schmitt trigger M74HC14 en paralelo.

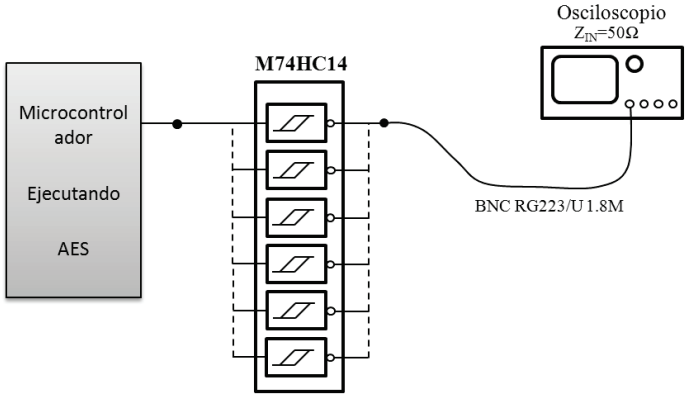


Figura 4.51: Configuración 4 para la generación de trigger

Configuración 5: Se conecta un schmitt trigger en serie con el microcontrolador y a continuación 5 en paralelo. Con este setup se intenta analizar el posible efecto de carga sobre el puerto de microcontrolador, que tiene el uso de varios schmitt trigger en paralelo.

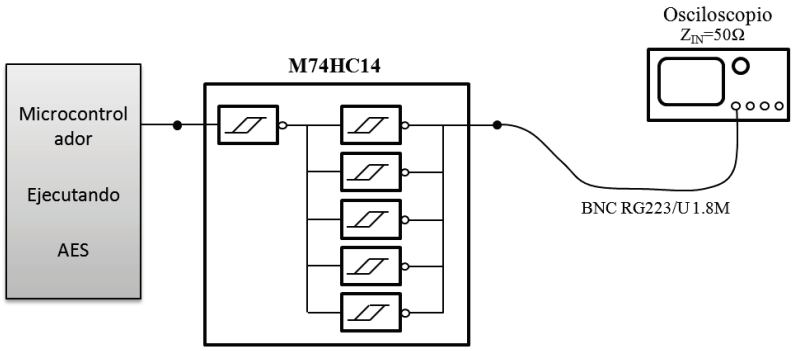


Figura 4.52: Configuración 5 para la generación de trigger

A continuación se muestran los resultados obtenidos del análisis de las distintas configuraciones implementadas. En la Tabla 4.4 se comparan los tiempos y pendientes de subida y bajada o Slew Rate, de los pulsos y en la Tabla 4.5 y Tabla 4.6 se presentan los resultados estadísticos obtenidos tras realizar 1000 disparos con cada una de las configuraciones. Se indica la diferencia máxima entre los registros máximo y mínimo en el que se ha producido el disparo, la desviación típica y la varianza.

Tabla 4.4: Análisis físico señales trigger

Config.	Flanco Bajada		Flanco Subida		Tensión Flanco (v)
	Tiempo (ns)	Pendiente (v/µs)	Tiempo (ns)	Pendiente (v/µs)	
1	31.5	81	$5.2 \cdot 10^3$	0.5	3.2
2	22.1	174	22	175	4.8
3 (1MΩ)	22	171	22.1	170	4.7
3 (50Ω)	2.25	604	2.65	513	1.7
4(1 s.t.)	2.25	604	2.65	513	1.7
4(2 s.t.)	1.85	1168	2.89	747	2.7
4(3 s.t.)	1.70	1506	2.93	874	3.2
4(4 s.t.)	1.66	1639	2.92	932	3.4
4(5 s.t.)	1.65	1745	3.07	938	3.6
4(6 s.t.)	1.53	1987	3.00	1013	3.8
5	2.52	1143	4.38	658	3.6

Donde:

- *Tiempo*: Se define como el tiempo necesario para que el pulso se eleve/descienda desde el valor de referencia bajo/alto, hasta el valor de referencia alto/bajo, siendo el valor de referencia bajo el correspondiente al 10% del pulso y valor de referencia alto el 90% del pulso.

– *Pendiente*: Definida como la medida entre los puntos en los que la tensión toma el valor de nivel bajo y el valor de nivel alto, sin tener en cuenta las posibles oscilaciones posteriores, tal como se muestra en el ejemplo de la Figura 4.53:

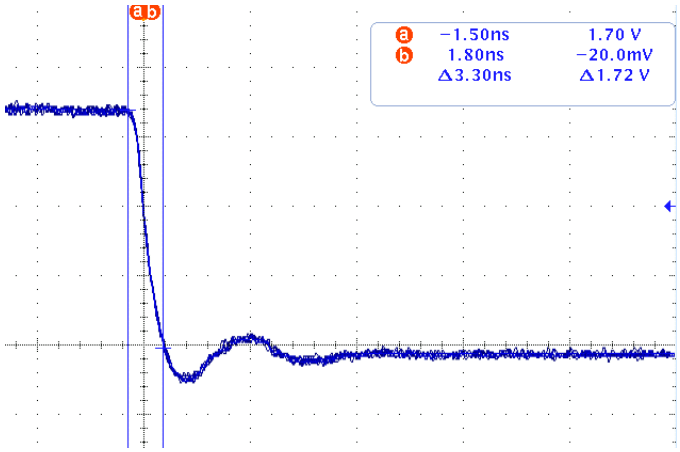


Figura 4.53: Determinación de la pendiente del flanco

Tabla 4.5: Análisis estadístico flanco bajada 1000 señales trigger

Flanco Bajada			
Configuración	**Dif. Max.**	**Desv. Est.**	**Varianza**
1	17	3	7
2	51	4	18
4(1 s.t.)*	1	0.46	0.21
4(2 s.t.)	1	0.48	0.23
4(3 s.t.)	1	0.47	0.22
4(4 s.t.)	1	0.47	0.22
4(5 s.t.)	2	0.45	0.20
4(6 s.t.)	2	0.39	0.15
5	2	0.46	0.21

* Obsérvese que la configuración 3 (50Ω) es igual a la 4 (1 s.t.), por ese motivo, no se ha añadido a la tabla.

Tabla 4.6: Análisis estadístico flanco subida 1000 señales trigger

Flanco Subida			
Configuración	**Dif. Max.**	**Desv. Est.**	**Varianza**
1	526	85	7180
2	64	5	23
4(1 s.t.)	2	0.45	0.18
4(2 s.t.)	2	0.45	0.18
4(3 s.t.)	1	0.50	0.20
4(4 s.t.)	1	0.49	0.24
4(5 s.t.)	2	0.47	0.06
4(6 s.t.)	2	0.45	0.20
5	2	0.50	0.25

Analizando los resultados se llega a la conclusión que el uso del schmitt trigger mejora de forma notable la calidad de la señal de trigger generada, aunque la configuración del osciloscopio es también un elemento a considerar, como se desprende de los resultados obtenidos de la configuración 3. En la siguiente imagen se puede apreciar el flanco de bajada generado con las dos impedancias de entrada del osciloscopio. Como se aprecia, las distorsiones presentes con la impedancia de 1MΩ desaparecen al seleccionar 50Ω. Por ese motivo, el resto de medidas se realizaron con esta última configuración.

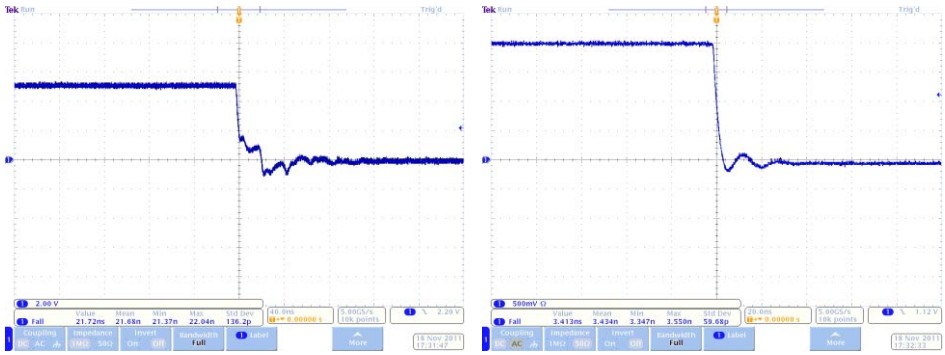


Figura 4.54: Flanco bajada Z_{IN}=1MΩ (izquierda) Z_{IN}=50Ω (derecha)

Durante la realización de las pruebas también se manifestó una pequeña influencia de la longitud de los cables en la pendiente de los flancos. Ésta es poco significativa, no obstante se intentó disminuir acortándolos al máximo.

Tras concluir el estudio se decidió utilizar la configuración 4, con seis schmitt trigger en paralelo. Respecto del pulso de bajada (realmente el más importante puesto que es el que indica el inicio de la traza a medir), se comprueba que, aunque curiosamente la distancia máxima es superior a la obtenida con menos inversores en paralelo, consigue mejores valores de desviación típica y varianza. Algo fundamental cuando se precisa una sincronización estable durante un alto número de repeticiones.

4.2.8 Mesa sujeción

Un elemento que en la mayoría de los setup pasa desapercibido y que en esta disertación ha demostrado su importancia, es la mesa de realización de los ensayos. Durante la ejecución de un ataque, es primordial que la sonda EM no varíe su posición. Si la posición de ésta cambia durante el proceso de captura de los registros EM, los resultados no serán satisfactorios o en el mejor de los casos, se verá aumentado el número de medidas necesarias para su realización.

En este estudio fue necesaria la fabricación de una mesa que aislara el test de las vibraciones presentes en el edificio. Para ello, se construyó una base o peana con elastómeros

de caucho adaptable a la mesa, lo suficientemente pesada para realizar la función de amortiguador. Véase las siguientes figuras:

Figura 4.55: Mesa test setup

Figura 4.56: Base mesa test setup: detalle elastómeros

4.2.9 Cámara Anecoica

La cámara anecoica elimina toda perturbación externa, tanto radiada como conducida, ya que trabaja como una caja de Faraday que evita la entrada de componentes radiadas. Además, en la mayoría de los casos incluye un filtro EMI que asegura una tensión de alimentación libre de ruido.

Por tanto su uso es muy recomendable [[Vua'09], [Mey'11]], aunque no obligado. Por ejemplo, en su estudio Peeters y otros autores [Pee'07], recomiendan el uso de una cámara de Faraday para eliminar el posible ruido ambiental, pese a ello, llevaron a cabo sus experimentos de forma suficientemente precisa sin la ayuda de una cámara.

Otros autores como [[Qui'01], [Sou'10]], utilizan métodos alternativos para evitar perturbaciones externas. Por ejemplo, Soussi y sus compañeros desarrollan una pequeña caja de Faraday de acero recubierta de aluminio, donde introducen el EUA y la sonda para así evitar el ruido externo, tanto magnético como electrostático. El inconveniente de este método es que puede producir reflexiones, sobre todo a bajas frecuencias [Sau'10] y el aislamiento de dos elementos de la cadena de medida no asegura la ausencia de ruido en la medida si el resto se encuentran sometidos a un ambiente EM contaminado [Gan'01].

En este trabajo, dado que se disponía de ella, se realizaron todos los ensayos dentro de una cámara semianecoica con filtro EMI de 220v, Figura 4.57 y Figura 4.58:

Figura 4.57: Filtros EMI alimentación cámara anecoica

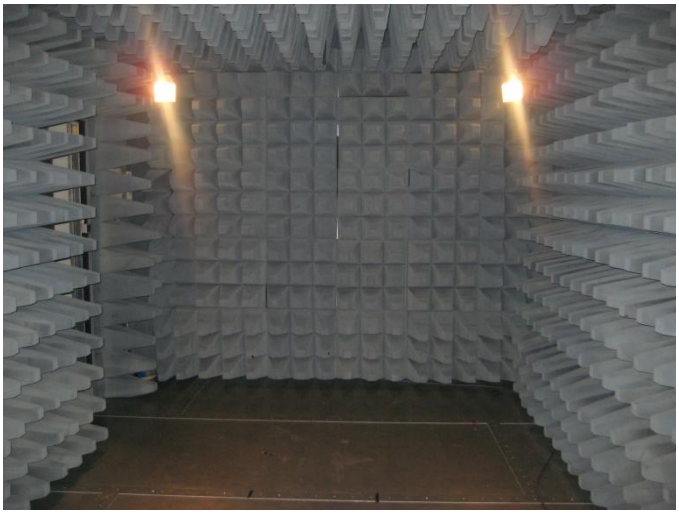


Figura 4.58: Cámara Semianecoica laboratorio CEM Universidad Carlos III de Madrid donde se realizaron las medidas

Capítulo 5

RESULTADOS EXPERIMENTALES

En este epígrafe se van a mostrar los resultados experimentales obtenidos para intentar determinar la seguridad inherente relativa a los ataques por canal lateral electromagnético EMA, de los cuatro microcontroladores de propósito general bajo estudio. En cada uno de ellos se programó el estándar de encriptación AES en su versión de 128 bits, sin ningún tipo de contramedida software ni hardware, y se realizaron varios ataques CEMA a cada uno de ellos utilizando tres sondas EM distintas de campo cercano preamplificadas, situadas en varias posiciones. La diferencia de efectividad del ataque en cada uno de ellos, ha proporcionado un indicativo de su seguridad congénita.

5.1 Placa EMA 1: C8051F303 8 bits

5.1.1 Desarrollo Experimental del Ataque

Paso 1. Programación AES

El primer paso es realizar la programación del estándar de encriptación en la placa bajo estudio. Este proceso se ha realizado utilizando el entorno de desarrollo facilitado por el fabricante del equipo: "*Silicon Laboratories IDE*" [SL'09] y el adaptador "*USB Debuger Adapter*" [SL'06], que permite la conexión del PC con el microcontrolador.

El algoritmo AES para este dispositivo se ha escrito en lenguaje ensamblador, tomando como base la versión publicada en lenguaje ANSI C en [SL'07], con algunas modificaciones que optimizan el uso de memoria. Éste, realiza la encriptación de los textos

aleatorios almacenados en su memoria de programa de forma consecutiva utilizando una clave prefijada. Dado que la memoria interna del C8051F303 es relativamente limitada (8 Kb), sólo permite el almacenamiento de alrededor de 300 textos de 16 bytes. Por ese motivo, el proceso de captura de trazas se ha realizado en bloques de 250 medidas, hasta obtener las 1000 necesarias para cada análisis.

La obtención de los textos aleatorios para su cifrado con el formato adecuado para ensamblador, se ha realizado mediante una aplicación desarrollada ex profeso en Labview, como ya se ha revelado en el capítulo anterior.

Paso 2. Captura de Trazas

Para realizar este paso es necesario montar el setup completo, tal como se explicó en el epígrafe anterior y configurar el osciloscopio de forma que capture la señal asociada a la ejecución del algoritmo de encriptación.

La captura de los registros EM se ha realizado con tres sondas distintas, dos comerciales y una fabricada de forma artesanal:

1. EM6995: Sonda comercial Electrometrics EM6995, conectada al osciloscopio a través de un cable BNC macho-macho 2249-Y-72 y preamplificador Langer PA303.
2. MFA-R: Sonda comercial Langer EMV MFA-R 0.2-75 milimétrica conectada al osciloscopio a través de cable SMA macho-macho, preamplificador Bias-Tee BT 706 y cable BNC macho - SMA hembra.
3. Homemade: Sonda artesanal fabricada con hilo de cobre esmaltado, conectada al osciloscopio a través de un cable BNC macho-macho 2249-Y-72 y preamplificador Langer PA303.

El posicionamiento de las sondas sobre el dispositivo se ha hecho en función del nivel de la señal captada. Es decir, se han buscado los puntos en los que la señal captada es máxima y sobre esos puntos se han realizado las capturas. Para el caso concreto de la sonda EMV de cabeza milimétrica, se ha dividido la superficie del microcontrolador con una cuadrícula milimétrica, lo que permite una identificación clara de la zona de medida y cierta repetitividad de las medidas.

En cuanto a la configuración del osciloscopio digitalizador para realizar la captura, es necesario medir previamente el tiempo de ejecución del AES sobre este dispositivo. Una vez conocido este dato, y sabiendo que el muestreo se va a realizar a su frecuencia máxima (5

Gmuestras/segundo), se tienen que definir los parámetros de configuración necesarios, como la longitud del registro, escala y posición horizontal, retardo, número de medidas… para que de este modo, la parte de la señal a capturar se muestre íntegramente y la captura comience en el momento adecuado.

El tiempo de ejecución de las dos primeras rondas del AES: "*AddRoundKey*" y "*Subbytes*"[30] incluido el preprocesado con este dispositivo ejecutándose a una frecuencia de 11.0592MHz, es de:

$$18.82\mu s\ @5Gs/sg \Leftrightarrow 94.100\ Registros$$

A partir de este dato se establece la configuración del osciloscopio y se realizan las medidas oportunas, haciendo uso de la aplicación desarrollada en Labview tal y como muestra la siguiente imagen:

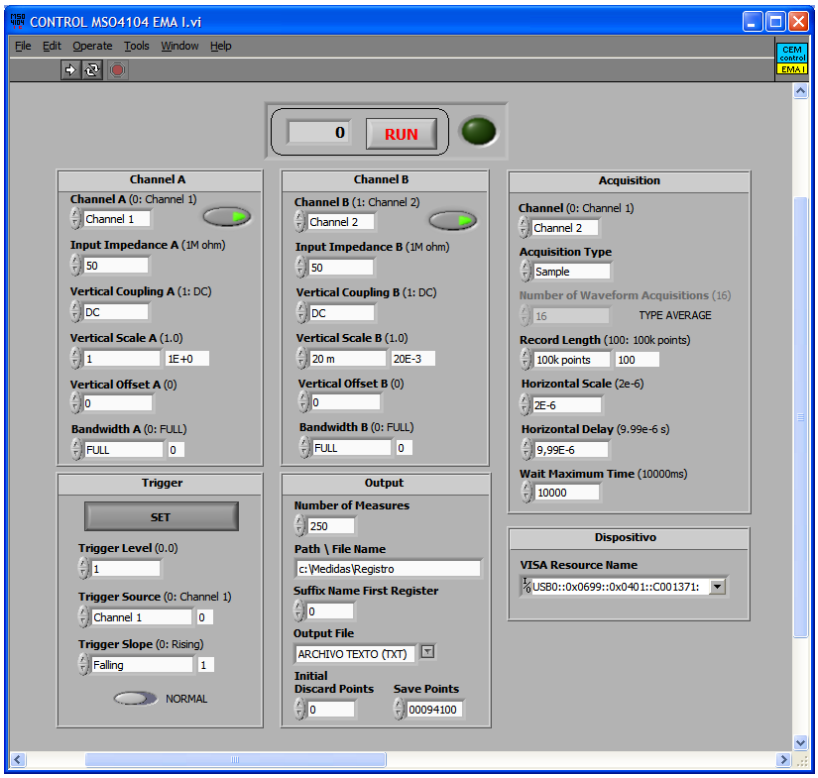


Figura 5.1: Configuración Setup Medida Placa EMA 1

Paso 3. Análisis de Resultados

Una vez capturadas las trazas EM, con la ayuda del software diseñado en Matlab, se determina la clave supuesta utilizando la metodología del CEMA detallada en el epígrafe 3.4.

[30] Para llevar a cabo el ataque CEMA, únicamente es necesario captar la señal asociada a la etapa de Subbytes, pero en este estudio se captó también la fase inicial AddRoundKey para la realización de otras pruebas. Después en el procesado se descartaron.

En este trabajo, a diferencia de lo expuesto en la bibliografía, se han analizado varios modelos de consumo, siendo el HW el que genera mejores resultados y en algunos casos el único que genera resultados satisfactorios. Por este motivo, el consumo teórico se ha calculado utilizando únicamente este algoritmo.

Los programas implementados proporcionan los siguientes datos:

a) **Presunta clave**: aquella que genera el mayor C.C.

b) **Coeficiente de correlación máximo correspondiente a la presunta clave**.

c) **Registro donde se genera la mayor correlación**: Permite determinar, a partir del ratio de muestreo (5 Gmuestras/sg), el instante o instantes de tiempo en el que el byte de la clave es computado en el algoritmo de encriptación AES.

d) **Gráfica "Coeficiente de Correlación – Registro" para la clave supuesta**: Representa el coeficiente de correlación obtenido a partir de la supuesta clave para cada uno de los registros de la traza. Si la suposición hecha es correcta, permite conocer gráficamente los momentos en los que el algoritmo opera con el byte bajo ataque. La Figura 5.2 muestra una gráfica de este tipo, resultado de un CEMA sobre el byte 6 de la clave.

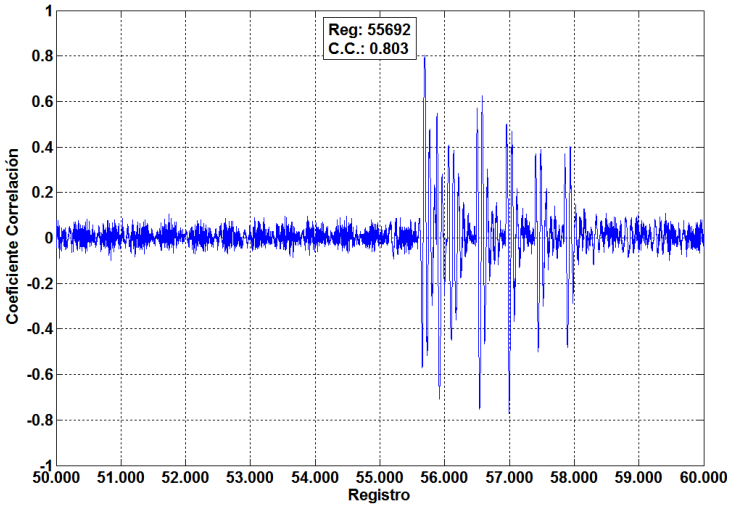


Figura 5.2: Gráfica C.C. – Registro para la clave hipotética con mayor coeficiente de correlación

En este ejemplo aparecen varios picos[31], signo de que el byte de la clave objeto del ataque es usado en varias instrucciones del algoritmo. En concreto en dos instrucciones de 5 ciclos de reloj de duración, que concuerda con el número de grupos de picos de la gráfica:

[31] Muy típico en implementaciones software, donde primeramente el byte es calculado y luego movido a una posición de memoria a través de un registro [Man'07].

INSTRUCCIÓN	DURACIÓN
MOVC A , @A+DPTR	3 ciclos
MOV State+06H , A	2 ciclos

e) **Gráfica "Coeficiente de Correlación – Número de Trazas"**: Muestra la evolución del coeficiente de correlación de cada una de las 256 claves posibles, en función del número de textos cifrados para el registro con mayor C.C. El coeficiente asociado a la clave hipotética se representa en color rojo, para su diferenciación del resto como se puede apreciar en la Figura 5.3.

A partir de esta gráfica se puede determinar de forma visual, el número mínimo de medidas o textos a cifrar para poder determinar la clave con suficiente confianza, lo que permite comparar numéricamente la efectividad relativa del ataque en las distintas implementaciones del algoritmo.

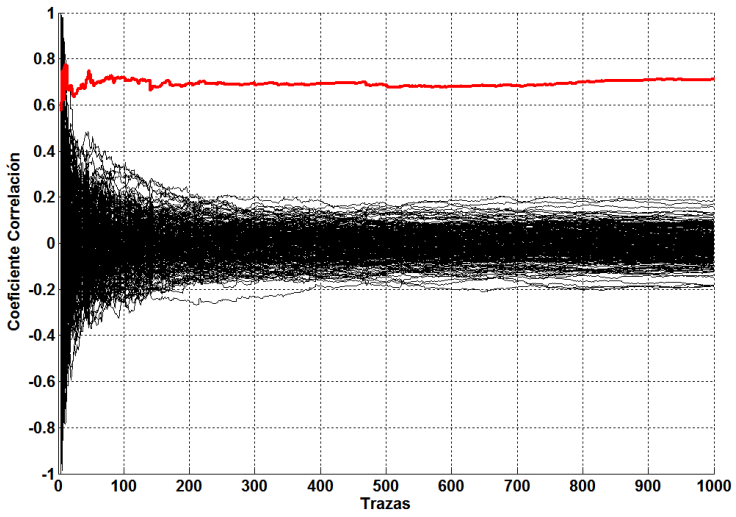


Figura 5.3: Gráfica Coeficiente Correlación - Número Trazas

Dado que las medidas realizadas en un EMA, y por ende PA, no son deterministas[32] debido a la presencia incondicional de ruido, se ha decidido fijar el número mínimo de medidas necesarias para el éxito de un ataque con un nivel de confianza alto. Así, en este estudio se ha establecido que éste es aquel en el que se cortan la envolvente de las curvas y la diferencia de la curva asociada a la clave hipotética con la envolvente de las curvas. Dicho de otro modo, el número mínimo de trazas, se corresponde con el punto en el que el C.C. de la clave supuesta es doble del de la envolvente.

[32] Dos medidas consecutivas nunca dan exactamente los mismos resultados.

En la siguiente gráfica referente a otro CEMA, se muestra visualmente su determinación. En rojo se muestra la evolución en valor absoluto del C.C. asociado a la clave supuesta, en verde la envolvente de las curvas, también en valor absoluto y en azul la diferencia de la curva roja y verde. El punto en el que se cruzan las curvas verde y azul, indica el número mínimo de medidas necesarias para realizar un CEMA sobre dicha placa con suficiente confianza:

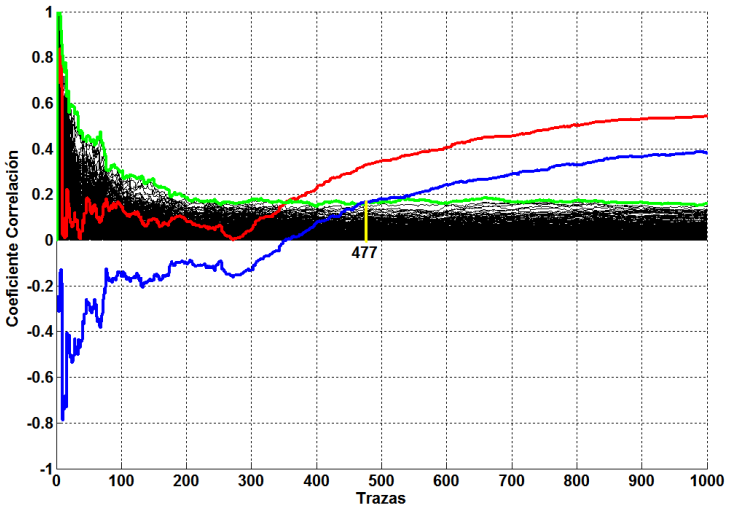


Figura 5.4: Determinación del número mínimo de medidas ataque CEMA

5.1.2 Resultados Experimentales

A continuación se muestran los resultados obtenidos al realizar el ataque por correlación electromagnética CEMA al microcontrolador C8051F303 de la placa EMA 1, utilizando tres sondas distintas.

Tanto la clave mostrada, como el byte 6 objeto del ataque han sido definidos de forma totalmente aleatoria:

```
0C8H, 00AH, 04AH, 0BFH, 078H, 0D3H, 0B2H, 079H, 098H, 0A0H, 0B6H, 036H, 0B4H, 05EH, 09BH, 0D1H
```

El ataque se ha centrado en la fase 'Subbytes' de la primera fase, debido a que es la que ha proporcionado mejores resultados. El número de medidas realizadas y, por ende, de textos encriptados, por cada posición y sonda es 1000, muy por encima del valor mínimo de medidas necesarias para realizar el ataque sobre este dispositivo.

Los resultados obtenidos se muestran en la Tabla 5.1.

Como conclusiones provisionales a partir de estos resultados se puede decir que cualquier sonda es factible de ser usada para un CEMA sobre la placa EMA 1 y la sonda

MFA-R es la que consigue mejores resultados, pues obtiene un C.C. mayor y hace el ataque más efectivo ya que requiere un menor número de medidas.

Tabla 5.1: Resultados ataque CEMA sobre placa EMA 1

SONDA	POSICIÓN	COEF. CORRELACIÓN	
		VALOR	Nº TRAZAS
EM6995	EM1	**0.759**	**65**
	EM2	0.711	82
	EM3	0.541	477
	EM4	0.621	195
MFA-R	MF1	0.803	104
	MF2	**0.880**	58
	MF3	0.840	**46**
	MF4	0.844	93
HOME	HO1	0.678	90
	HO2	**0.793**	**46**
	HO3	0.723	60

5.1.3 Resultados Experimentales Adicionales

En este apartado, para intentar esclarecer algunos aspectos de los ataques por canal EM en general y de los CEMA en particular, se van a detallar los experimentos realizados sobre la placa EMA 1.

5.1.3.1 Número mínimo de medidas ataque CEMA

Como ya se ha explicado, el número mínimo de medidas necesarias para que un CEMA tenga éxito, se ha definido como aquel para el cual el C.C. de la clave correcta se hace doble del de la envolvente del resto de claves para el registro que genera mayor correlación. En la siguiente figura por ejemplo, éste se corresponde con la traza número 1413:

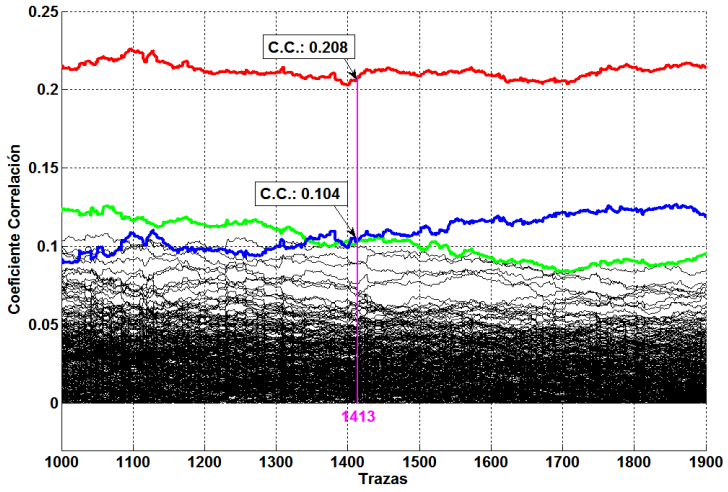


Figura 5.5: Registro que indica el número mínimo de trazas del ataque CEMA

La pregunta que surge es por qué escoger el registro con mayor C.C. para realizar dicho análisis, o dicho de otro modo, el número mínimo de trazas es inversamente proporcional al C.C. del registro analizado. En la siguiente imagen se muestra el C.C. obtenido de un CEMA sobre el byte 2 de la clave, en cada uno de los registros de la traza para la clave correcta, utilizando el modelo HW:

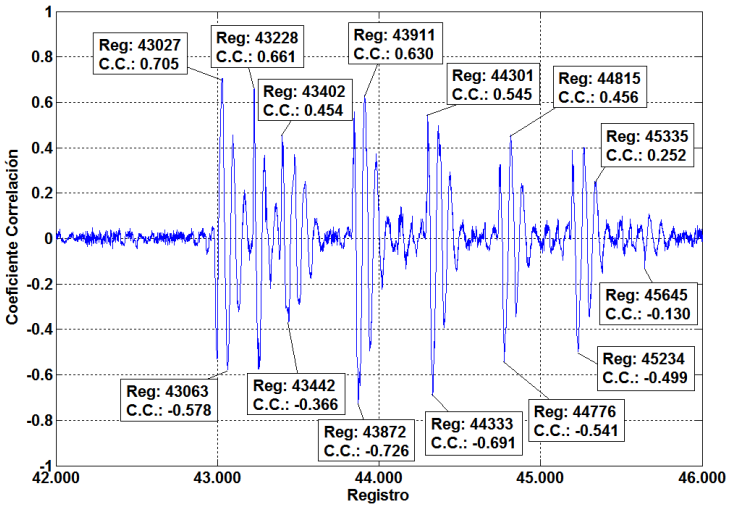


Figura 5.6: C.C. en función del registro de la traza para la clave correcta

Si ordenamos los registros en función del valor del C.C. obtenido y determinamos el número mínimo de medidas para los más correlados, se obtienen los siguientes resultados:

Tabla 5.2: Análisis número mínimo de medidas para los registros más correlados con HW

Registro	**C.C.**	**Nº Trazas**
43872	0.726	101
43027	0.707	103
43030	0.705	122
43031	0.704	143
43029	0.699	118
43873	0.699	103

Si se seleccionan algunos registros que presentan un máximo local (es decir en picos de la traza), se obtiene lo siguiente:

Tabla 5.3: Análisis número mínimo de medidas para máximos locales de coeficiente de correlación con HW

Registro	**C.C.**	**Nº Trazas**
43872	0.726	101
43228	0.661	111
43402	0.454	317
43911	0.631	103
44301	0.545	241
44333	0.691	80
43063	0.578	243
44776	0.541	103
44815	0.456	416
45234	0.499	221
43442	0.366	383
45335	0.252	745
45645	0.130	3112

Como se aprecia, en general, cuanto mayor es el C.C., menor es el número de trazas o medidas que es necesario realizar para conseguir un ataque exitoso. Este hecho se comprueba de forma más clara si se expresan los resultados gráficamente, Figura 5.7:

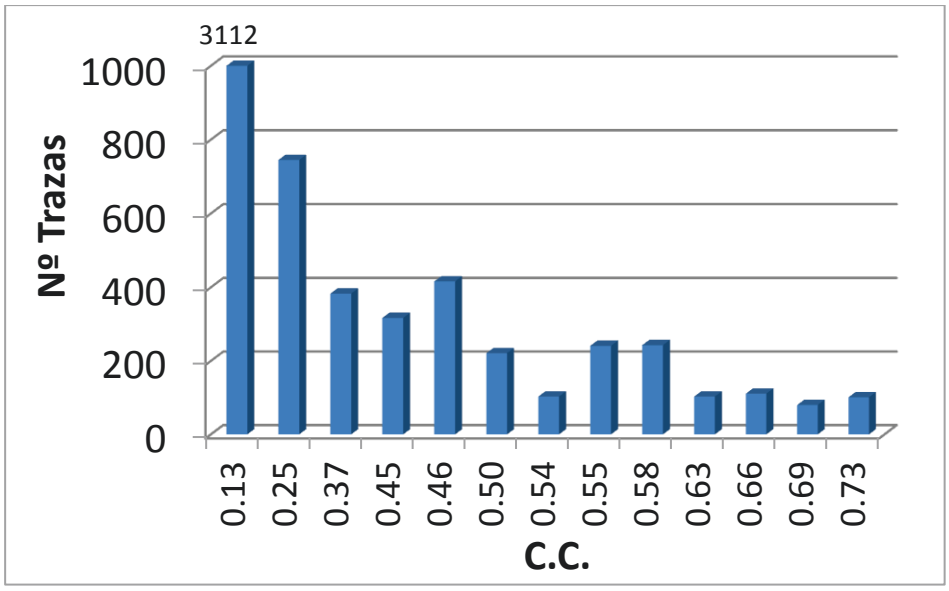


Figura 5.7: Influencia C.C. en el número de trazas necesarias para un ataque CEMA con modelo de consumo HW

En el caso de utilizar el modelo de consumo HD, se obtienen resultados similares:

Tabla 5.4: Análisis número mínimo de medidas para los registros más correlados utilizando HD

Registro	**C.C.**	**Nº Trazas**
43837	0.364	263
43838	0.345	485
43833	0.241	2270
43904	0.232	3488
43836	0.231	1414
43906	0.229	ERROR

Por consiguiente, como se desprende del análisis, no existe una relación lineal entre el C.C. obtenido y el número mínimo de trazas. Sin embargo, salvo algún caso concreto, suele ser necesario un menor número de trazas en los ataques en los que se obtiene un C.C. más alto.

5.1.3.2 Ataque CEMA sobre la fase AddRoundKey del AES

Como se ha mencionado anteriormente, los CEMA implementados en este estudio se han realizado sobre la fase Subbytes de la primera ronda del estándar de encriptación AES, porque es donde se obtienen mejores resultados. Según la literatura, las operaciones no lineales como Subbytes, aumentan la eficiencia de los ataques [[Pra'04], [Man'07]], aunque no se demuestra el por qué.

En la Figura 5.8 se muestra el resultado de realizar un CEMA utilizando el modelo de consumo HW sobre la operación lineal AddRoundKey de la ronda inicial (como ya se ha visto, también susceptible de ser atacada). En rojo se ha destacado la evolución del C.C. asociado a la clave correcta con valor $10_D = 0A_H$.

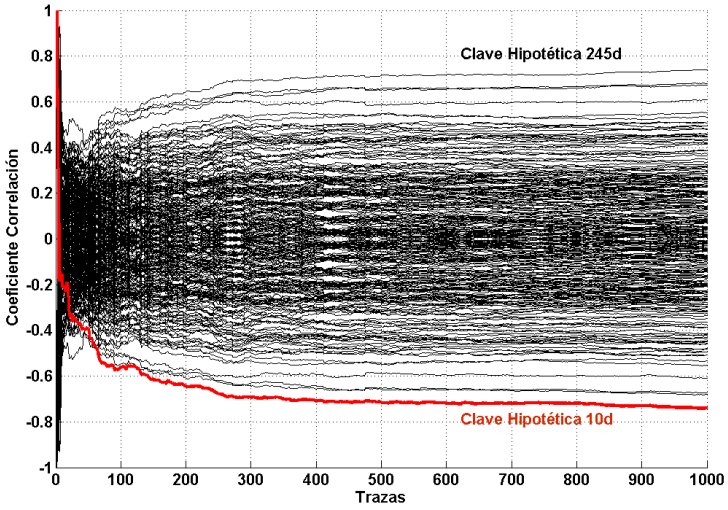


Figura 5.8: Gráfica Coeficiente Correlación – Trazas fase AddRoundKey

Como se puede advertir, los resultados obtenidos son peores. La gráfica obtenida es simétrica respecto del eje de abscisas. Esto es debido a que las claves con valores complementarios presentan un mismo valor absoluto de C.C., algo lógico, pues si se calculan sus valores teóricos de consumo aplicando HW, generan valores también complementarios. Por ese motivo, la clave 10_D=$0A_H$=00001010_B genera una correlación igual pero con distinto signo que 245_D=$F5_H$=11110101_B.

De esta forma, el resultado del ataque no es totalmente satisfactorio. Dos claves generan el máximo C.C.: 0.739 y el nivel de incertidumbre es superior, puesto que aparecen claves incorrectas con un valor de correlación muy similar al de la clave correcta. En el ejemplo de 0.680.

En este caso, el algoritmo HD no puede aplicarse. Genera resultados incorrectos debido a que su solución únicamente depende de la clave supuesta:

$$HD(AddRoundKey) = HD(Ptext, Ptext \oplus Key) = HW(Key)$$

5.1.3.3 Análisis modelos de consumo

En este trabajo se han aplicado varios tipos de modelos de consumo:

a) Hamming Weight

$$HW(Subbytes_i) = HW(\Sigma_i)$$

Donde: $\Sigma_i = Ptext_i \oplus Key_i$

b) Hamming Distance

$$HD(Subbytes_i) = HD\big(\Sigma_i, Sbox(\Sigma_i)\big)$$

c) Modelo propio implementado que simula el consumo de la fase Subbytes al computar el byte *i* basado en HW:

$$HW(Subbytes_i) = HW(\Sigma_i) + 2 \cdot HW(Sbox(\Sigma_i))$$ [33]

d) Igual a c) pero basándose en HD:

$$HD(Subbytes_i) = HD(\mathrm{Sbox}(\Sigma_{i-1}), \Sigma_i) + 2 \cdot HD\big(\Sigma_i, Sbox(\Sigma_i)\big)$$

De estos modelos, los dos últimos resultan ser inválidos, planteando como solución claves incorrectas. Esto se debe a que simulan el consumo de una fase del AES, no un consumo instantáneo.

En cuanto a los dos primeros, el HW resultó ser más efectivo, pues consigue unos valores de C.C. superiores a los obtenidos con HD y descifra la clave con un menor número de trazas. En la siguiente tabla se muestran los C.C. máximos obtenidos de un CEMA realizado sobre la placa EMA 1, utilizando la sonda comercial EM6995:

Tabla 5.5: Análisis C.C. en función del modelo de consumo usado

Nº Textos	Subbytes	
	HW	**HD**
5000	0.717	0.3353
1000	0.718	0.2776
500	0.729	-*
100	0.768	-

***** El guion '-', indica que el ataque generó un valor incorrecto de clave.

Si se analiza el número de trazas necesarias para realizar el ataque de forma satisfactoria, se constata que con HD es necesario un número casi diez veces superior que con HW: Tabla 5.6, Figura 5.9 y Figura 5.10.

Tabla 5.6: Análisis número de trazas en función del modelo de consumo

Nº Mínimo Trazas	Subbytes	
	HW	**HD**
	70	690

[33] Esta ecuación modela el consumo de la fase Subbytes de un byte, implementada en el C8051F303:

```
MOV   A,State+05H
MOVC  A,@A+DPTR
MOV   State+05H,A
```

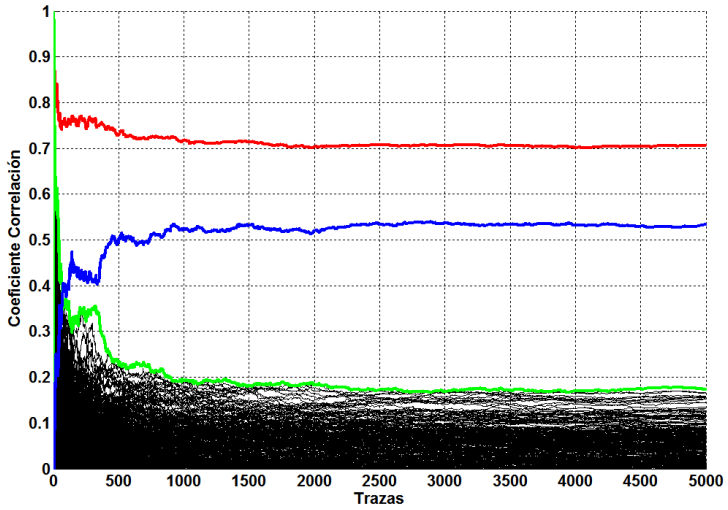


Figura 5.9: Número mínimo trazas aplicando HW

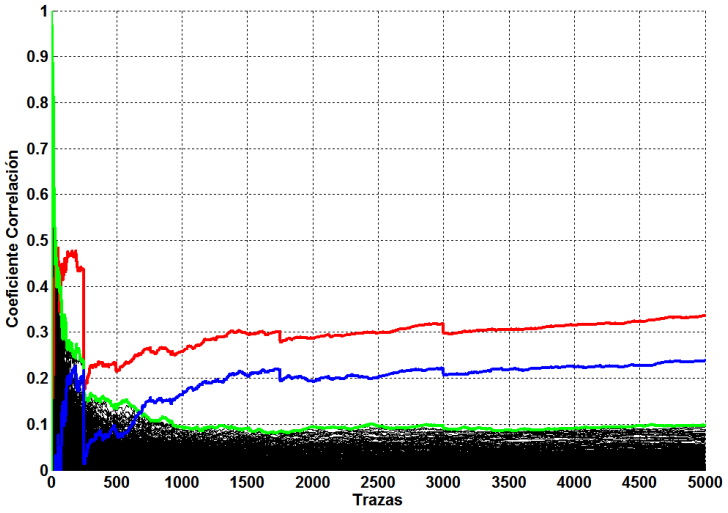


Figura 5.10: Número mínimo trazas aplicando HD

Se debe tener en cuenta que los dos ataques se realizaron sobre las mismas trazas y con las mismas condiciones. La única diferencia existente, es el modelo de consumo aplicado. No obstante, si se descartan las primeras 1000 trazas EM, para intentar evitar la caída brusca de correlación obtenida con el modelo HD (véase la Figura 5.10), se consiguen mejores resultados para éste último, aunque siguen siendo inferiores a los alcanzados con HW:

Tabla 5.7: Análisis número de trazas en función del modelo de consumo para 4000 trazas

4000 Trazas	**C.C.**	**Nº Trazas**
HW	0.726	101
HD	0.361	263

Por consiguiente, se puede llegar a la conclusión de que el HW resulta ser un modelo de consumo, que en el caso concreto de la placa EMA 1, genera mejores resultados en cuanto a nivel de correlación obtenido y número de trazas necesarias. Además, se muestra más robusto, al ser afectado en menor grado por posibles errores de medida.

5.1.3.4 Análisis comportamiento sondas bajo estudio

Para estudiar más en detalle el comportamiento de las sondas utilizadas en este estudio, se han llevado a cabo otros ensayos que intentan esclarecer su comportamiento.

Para ello, se ha realizado un experimento compuesto por dos fases, que permiten analizar la idoneidad de un SCA EMA/PA. En una primera etapa se ha medido la radiación EM emitida por el dispositivo EMA 1 al ejecutar 1000 veces las dos primeras fases del algoritmo de encriptación AES (AddRoundKey y SubBytes), sobre un texto fijo en el que todos sus bits están a 1. Se obtienen así, 1000 vectores de 94100 registros cada uno, que almacenan el campo EM asociado a la encriptación de los textos. La segunda fase es igual a la primera, pero cambiando el valor de uno cualquiera de sus bits. De forma aleatoria se ha elegido el bit 66, que tomará el valor 0, dejando el resto a 1. Una vez guardadas las 2000 trazas correspondientes a estas dos fases, se realizan dos análisis, cuyo objetivo es la detección de los puntos de influencia del bit 66 en las tramas medidas, es decir, los instantes correlacionados con dicho bit. Dado que las pruebas se van a realizar con las tres sondas, los resultados permitirán su comparación relativa.

- **Análisis 1**: En este primer análisis se utiliza la "Relación Señal a Ruido" o SNR. De forma general se define como el cociente de la varianza del dato, dividido por la varianza del ruido presente en la medida. Este parámetro permite determinar la cantidad de información que se está escapando en un punto de la traza. Para este caso particular, dado que se dispone de dos grupos de datos, la fórmula se ha expresado como se detalla a continuación:

$$SNR = \frac{Var(Dato)}{Var(ruido)} = \frac{Var(S_1, S_2)}{\frac{(n_1 - 1) \cdot Var(S_1) + (n_2 - 1) \cdot Var(S_2)}{(n_1 + n_2 - 2)}} \quad \textbf{(5.1)}$$

  Se obtiene así una tabla, con la SNR en cada uno de los puntos de las trazas, que se puede representar gráficamente, Figura 5.11.

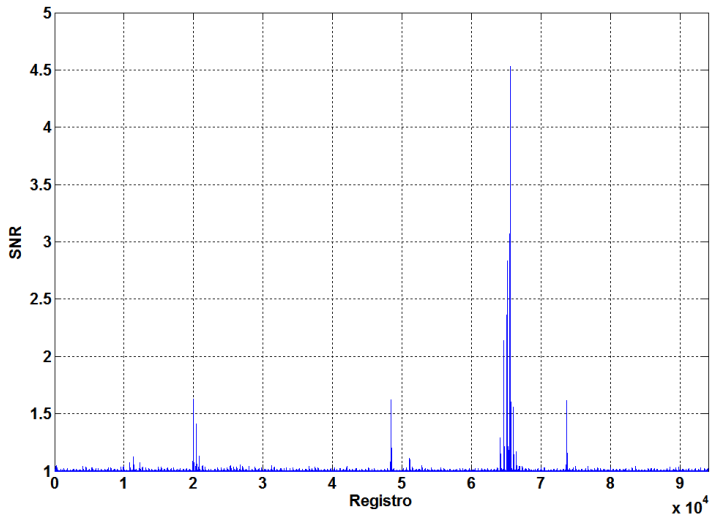


Figura 5.11: SNR obtenido con la sonda EM6995 en la posición EM3

- **Análisis 2**: En este caso se realiza un análisis simple pero efectivo, que consiste en realizar para cada grupo de medidas y para cada uno de los registros de las trazas, la media de las 1000 medidas. Se obtienen así dos vectores con las medias de cada uno de los 94100 registros. Por último, se calcula la diferencia de los dos vectores. Al igual que en el caso anterior este estudio mostrará los puntos en los que el bit 66 es procesado:

$$Dif.Medias = \overline{S_1} - \overline{S_2} \tag{5.2}$$

Gráficamente se obtiene lo siguiente:

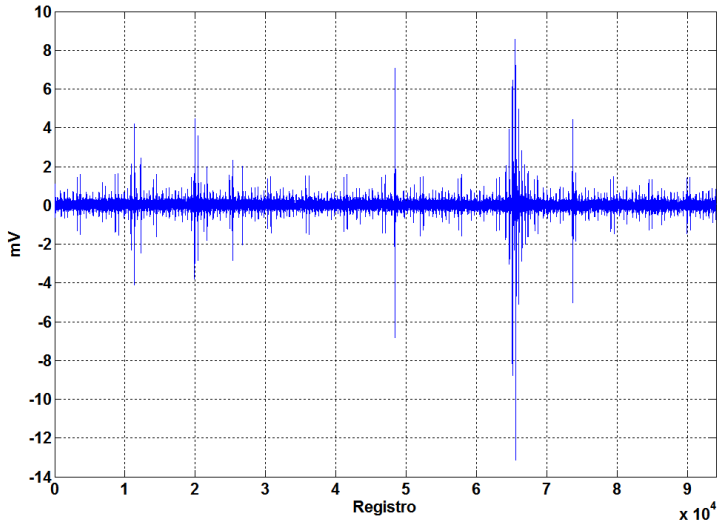


Figura 5.12: Diferencia Medias obtenido con sonda MFA-R en la posición MF4

A continuación se muestran los resultados obtenidos con las tres sondas. En la tabla se muestra el registro que genera el mayor valor para cada una de las sondas y en cada una de las posiciones de medida.

Tabla 5.8: Análisis comparativo sondas EM medida

		SNR		DIFERENCIA MEDIAS	
		MaxValor	REGISTRO	MaxValor (mv)	REGISTRO
EM6995	EM1	**5.33**	65.598	**7.5**	65.563
	EM2	3.71	65.598	6.8	65.598
	EM3	4.53	65.598	3.98	65.597
	EM4	2.34	65.598	2.33	65.580
MFA-R	MF1	7.48	65.575	10.61	65.538
	MF2	**9.89**	65.575	**13.51**	65.575
	MF3	3	65.578	5.69	65.578
	MF4	3.71	65.577	6.9	65.576
HOME	HO1	4.63	65.604	11.75	65.603
	HO2	**8.2**	65.602	**13.89**	65.602
	HO3	7.46	65.602	13.28	65.534

A la vista de los resultados se pueden obtener algunas conclusiones:

1. Las medidas SNR y Diferencia de Medias permiten determinar las zonas de medida más susceptibles a un ataque. En general, un nivel alto de estos parámetros implica un ataque más eficiente.
2. La medida SNR es más precisa pues ejerce la función de filtro, eliminando el ruido presente y facilitando el análisis.
3. Las tres sondas son factibles de ser usadas en un EMA. Por tanto, no es necesario un equipo muy costoso para su implementación.
4. La sonda MFA-R, al ser más pequeña, tiene un tiempo de respuesta inferior a las otras, dado que detecta la posición correlada varios registros antes que las otras dos sondas. Por consiguiente, proporciona una precisión mayor en términos temporales.
5. La sonda MFA-R tiene una resolución espacial mejor, pero también requiere un tiempo de preparación de la medida mayor, puesto que hay que seleccionar el punto exacto donde medir con una resolución grande. Como se puede observar en las siguientes dos gráficas, la posición de medida influye más en el caso de la sonda MFA-R que en el caso de la sonda HOME:

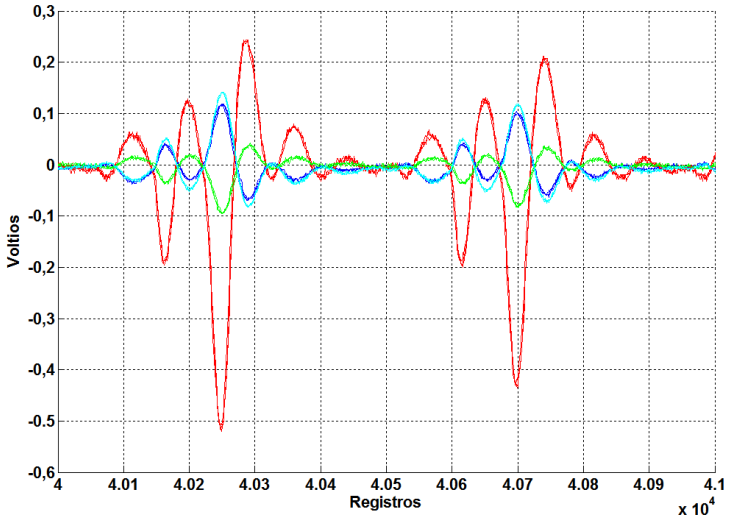


Figura 5.13: Medida Sonda EMV en cuatro posiciones distintas

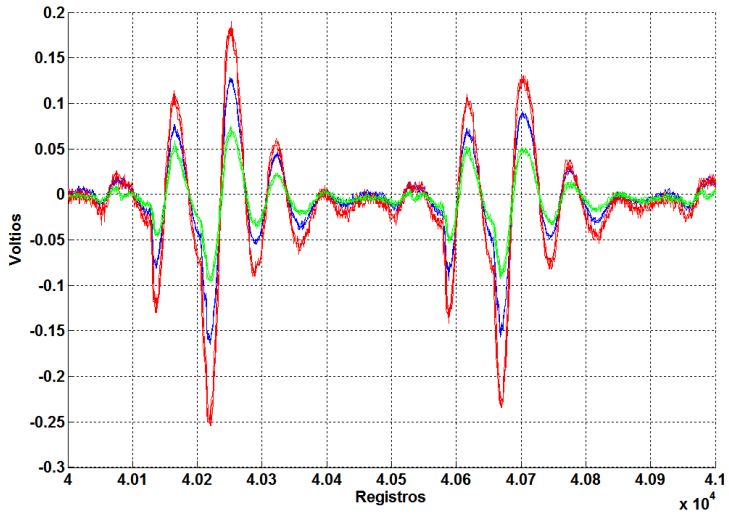


Figura 5.14: Medida Sonda HOME en tres posiciones distintas

5.2 Placa EMA 2: ARM7TDMI-S LPC2124FBD64 32 bits

5.2.1 Desarrollo Experimental del Ataque

Paso 1. Programación AES

Para este equipo el proceso de programación se ha realizado utilizando el entorno de desarrollo: "*Keil uVision*" [Kei'08] y el adaptador "Ulink" [Kei'04], para la conexión USB Pc – conector JTAG del microcontrolador.

El algoritmo AES para este dispositivo se ha escrito en lenguaje ANSI C a partir de la versión [SL'07], con distintas mejoras para reducir su tiempo de ejecución. Al igual que con la placa EMA1, los textos aleatorios que serán encriptados durante el ataque se han almacenado en la memoria de programa. El LPC2124FBD64 dispone de 256 Kb de ROM, una capacidad

suficiente para guardar más de 1000 textos de 16 bytes, por lo que la captura de trazas se puede realizar en un único ciclo.

Los textos aleatorios según una distribución normal han sido determinados mediante una aplicación en Labview desarrollada para ello.

Paso 2. Captura de Trazas

El test setup de medida utilizado es el mismo que en el caso de la placa anterior.

El tiempo de ejecución que emplea este microcontrolador configurado a una frecuencia de reloj de 60 MHz, para las dos primeras operaciones del AES: "*AddRoundKey*" y "*Subbytes*", incluido el preprocesado, es de:

$$23.8\mu s\ @5Gs/sg \Leftrightarrow 119.000\ Registros$$

Superar la franja de los 100.000 registros exige configurar el osciloscopio con una longitud de registro de 1 M, lo que ralentiza la captura de forma acusada. A pesar de ello, se decidió su captura en previsión de futuros estudios.

A partir de este dato, se configura el osciloscopio y se realizan las medidas con la aplicación de Labview implementada, tal como se muestra en la siguiente figura:

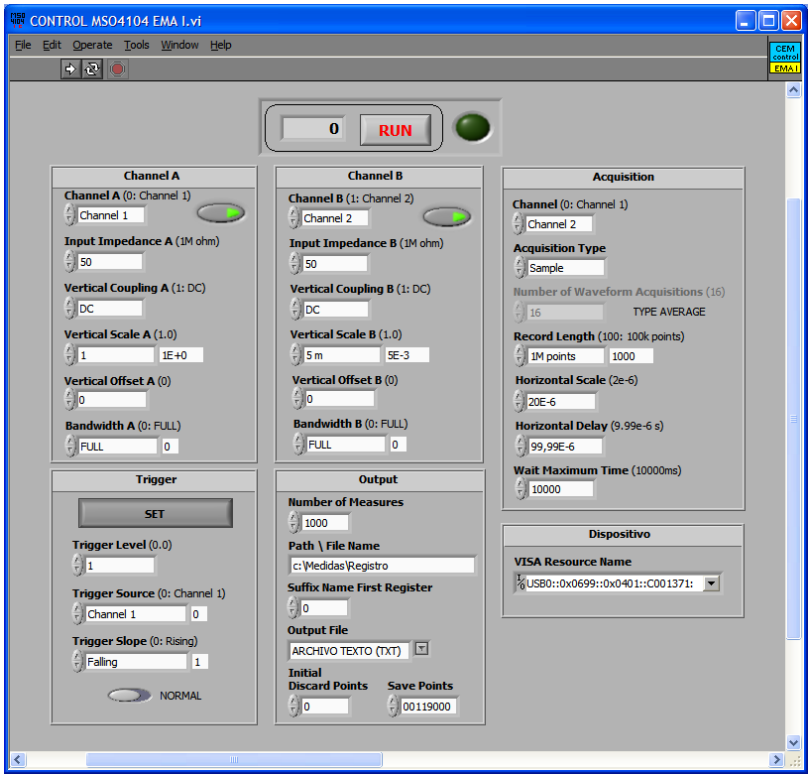


Figura 5.15: Configuración Setup Medida Placa EMA 2

Paso 3. Análisis de Resultados

A partir de las trazas capturadas, y utilizando el programa implementado en Matlab, se determina la clave supuesta utilizando HW; Único modelo de consumo que generó resultados positivos.

5.2.2 Resultados Experimentales

A continuación se muestran los resultados obtenidos del CEMA sobre la placa EMA 2, que tiene como objetivo el byte 6 de la clave:

```
{0xC8, 0x0A, 0x4A, 0xBF, 0x78, 0xD3, 0xB2, 0x79, 0x98, 0xA0, 0xB6, 0x36, 0xB4, 0x5E, 0x9B, 0xD1}
```

El ataque se ha centrado en la fase 'Subbytes' y se han realizado 1000 medidas por cada sonda y posición de medida.

Estos son los resultados obtenidos:

Tabla 5.9: Resultados ataque CEMA sobre placa EMA 2

SONDA	POSICIÓN	COEF. CORRELACIÓN	
		VALOR	Nº TRAZAS
EM6995	EM1	0.186	987
	EM2	**0.625**	**77**
	EM3	0.453	119
	EM4	0.438	471
MFA-R	MF1	0.221	752
	MF2	0.308	289
	MF3	**0.413**	**258**
	MF4	0.291	492
HOME	HO1	0.353	552
	HO2	**0.518**	**206**
	HO3	0.479	301
	HO4	0.423	206

A tenor de los resultados, se puede decir que cualquier sonda es útil para realizar un CEMA sobre la placa EMA 2, si bien los mejores resultados se han obtenido de forma global con la Homemade. La EM6995 ha conseguido el mejor ataque individual, pero sus resultados globales no superan los obtenidos con la sonda Homemade.

5.3 Placa EMA 3: ARMCORTEXM3 LPC1769 32 bits

5.3.1 Desarrollo Experimental del Ataque

Paso 1. Programación AES

El software utilizado para la programación de la placa EMA 3 ha sido el entorno de desarrollo: "LPCXpresso" [NXP'11], que utiliza el compilador libre "Eclipse" [Ecl'04]. En este caso no ha sido necesario un adaptador independiente, como el "Ulink" o "USB debugger", debido a que la placa incluye el interfaz denominado "LPC Link".

El algoritmo AES para este dispositivo se ha escrito en lenguaje ANSI C a partir de la versión implementada para la placa EMA 2. Con respecto a ésta, toda la parte del algoritmo referida al AES es prácticamente igual, mientras que la parte de configuración y control del hardware del microcontrolador es nueva. El LPC2124FBD64 dispone de 512 Kb de ROM, una capacidad muy superior a la necesaria, que le capacita para almacenar más de 30.000 textos aleatorios de 16 bytes.

Los textos aleatorios han sido obtenidos mediante la misma aplicación Labview utilizada para el micro ARM7.

Paso 2. Captura de Trazas

El test setup de medida utilizado es el mismo que en el caso de la placa anterior.

El tiempo de ejecución que emplea este microcontrolador a 120 MHz, para las dos primeras rondas del AES: "*AddRoundKey*" y "*Subbytes*", incluido el preprocesado, es de:

$$15.7\mu s\ @5Gs/sg \Leftrightarrow 78.500\ Registros$$

A partir de este dato se configura el osciloscopio y se realizan las medidas con la aplicación de Labview implementada, Figura 5.16.

Paso 3. Análisis de Resultados

A partir de las trazas capturadas y utilizando el programa implementado en Matlab, se determina la clave supuesta utilizando el modelo de consumo HW. Para este equipo, el modelo HD no proporcionó resultados positivos.

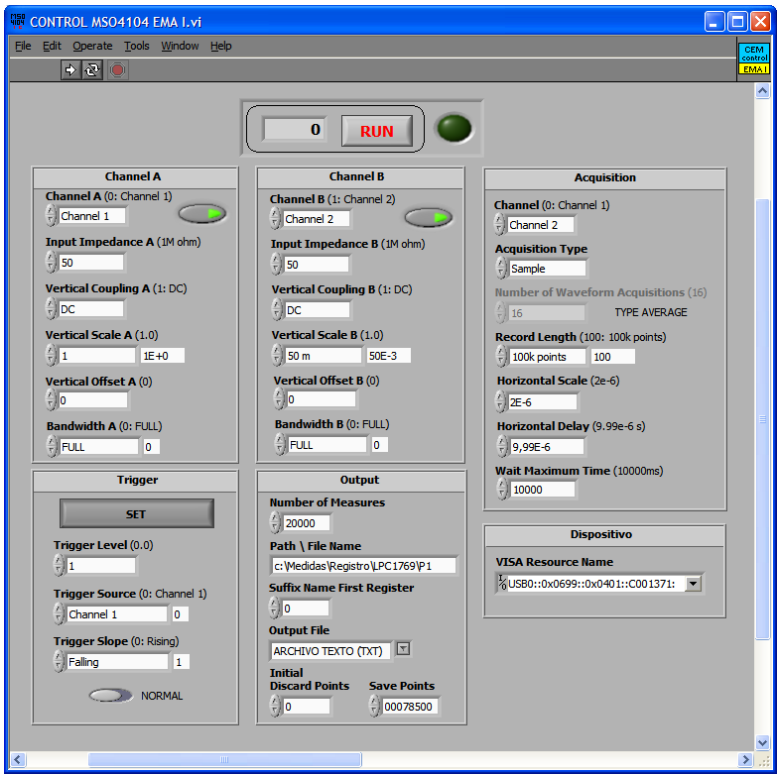


Figura 5.16: Configuración Setup Medida Placa EMA 3

5.3.2 Resultados Experimentales

A continuación se muestran los resultados obtenidos del CEMA sobre la placa EMA 3, que tiene como objetivo el byte 6 de la clave:

```
{0xC8, 0x0A, 0x4A, 0xBF, 0x78, 0xD3, 0xB2, 0x79, 0x98, 0xA0, 0xB6, 0x36, 0xB4, 0x5E, 0x9B, 0xD1}
```

El ataque se ha centrado en la fase 'Subbytes' y se han realizado 20.000 medidas por cada sonda y posición de medida.

Estos son los resultados obtenidos:

Tabla 5.10: Resultados ataque CEMA sobre placa EMA 3

SONDA	POSICIÓN	COEF. CORRELACIÓN	
		VALOR	Nº TRAZAS
EM6995	EM1	-	-
	EM2	-	-
	EM3	**0.086**	**12774**
	EM4	-	-
MFA-R	MF1	-	-
	MF2	**0.178**	**2093**
	MF3	-	**-**
	MF4	0.074	18467
HOME	HO1	**0.059**	**16637**
	HO2	0.047	*
	HO3	-	-
	HO4	0.038	*

*****: En este caso el coeficiente de correlación no alcanzó nunca un nivel doble del valor de la envolvente; Y si lo hizo en algún punto, volvió a descender, como se puede comprobar en la siguiente imagen:

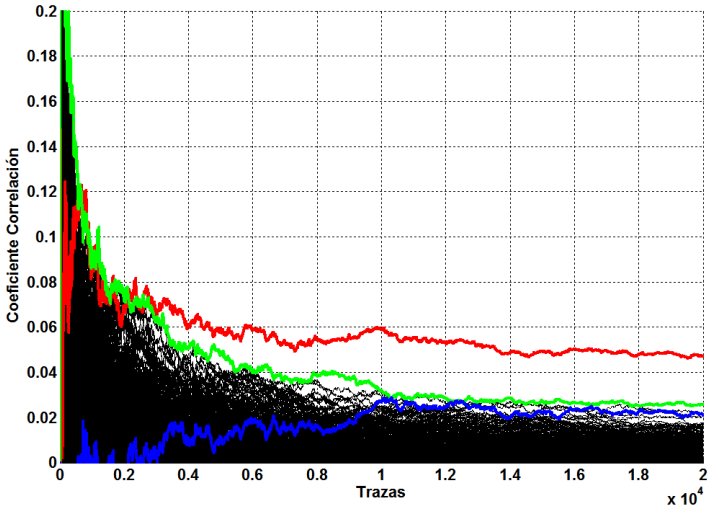


Figura 5.17: Placa EMA 3 sonda Homemade posición HO2 modelo HW

Como consecuencia del análisis anterior se puede deducir que la sonda más ideal para realizar un ataque sobre la placa EMA 3 es la MFA-R. Todas las sondas consiguen obtener la clave en algún intento, si bien, es la MFA-R la que consigue mejores resultados.

5.4 Placa EMA 4: ARMCORTEXM3 STM32L152RBT6 32 bits

5.4.1 Desarrollo Experimental del Ataque

Paso 1. Programación AES

El entorno de desarrollo utilizado para la programación de la placa EMA 4 ha sido: "Atollic TrueSTUDIO" [Ato'12], que al igual que para el caso anterior, se basa en el compilador "Eclipse". Al ser ésta, igual que EMA 3, una placa de desarrollo comercial, incluye el denominado "ST-Link/v2" para la conexión con el PC.

El algoritmo AES para este dispositivo se ha escrito en lenguaje ANSI C a partir de la versión implementada para la placa EMA 2. Las variaciones más importantes con respecto a este código se han realizado en la parte de configuración y control del hardware, mientras que la parte referente al código de encriptación no se ha variado sustancialmente. El STM32L152RBT6 dispone de 128 Kb de ROM, con dicha memoria es posible almacenar alrededor de 7000 textos de 16 bytes. Dado que van a ser necesarios más textos, se decidió modificar el programa, de forma que sólo almacenara el byte bajo ataque, ya que es el único que debe ser conocido para realizar el ataque. El resto de bytes, son obtenidos de forma aleatoria en tiempo real durante el mismo. En consecuencia, el ahorro de memoria es considerable.

Los textos aleatorios correspondientes al byte bajo ataque han sido obtenidos mediante una aplicación Labview, variación de la utilizada para el ARM7.

Paso 2. Captura de Trazas

El test setup de medida utilizado es el mismo que en el caso de la placa anterior.

Configurando el dispositivo a su frecuencia máxima de operación: 32 MHz, el tiempo que dedica para ejecutar las dos primeras rondas del AES: "*AddRoundKey*" y "*Subbytes*", incluido el preprocesado, es de:

$$57.4\mu s\ @5Gs/sg \Leftrightarrow 287.000\ Registros$$

En este caso, dado que el número de registros hace su procesado difícil, se decidió almacenar únicamente la fase Subbytes, que tiene un tiempo de procesado de:

$$26.2\mu s\ @5Gs/sg \Leftrightarrow 131.000\ Registros$$

A partir de este dato se configura el osciloscopio y se realizan las medidas con la aplicación de Labview implementada, Figura 5.18:

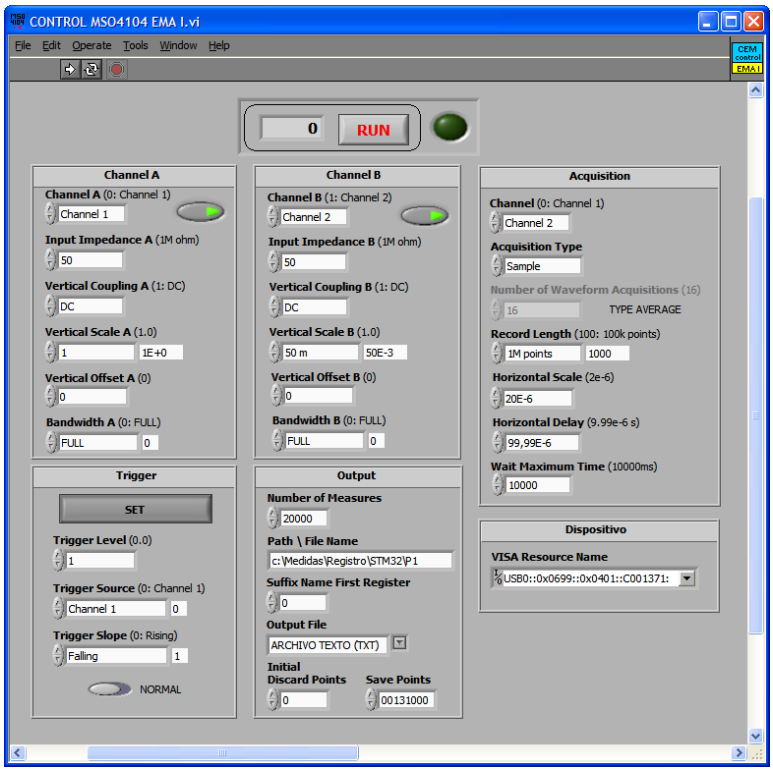


Figura 5.18: Configuración Setup Medida Placa EMA 4

Paso 3. Análisis de Resultados

A partir de las trazas capturadas y utilizando el programa implementado en Matlab, se determina la clave supuesta utilizando únicamente el modelo de consumo HW. El HD proporcionó claves incorrectas.

5.4.2 Resultados Experimentales

A continuación se muestran los resultados obtenidos del CEMA sobre la placa EMA 4, que tiene como objetivo el byte 6 de la clave:

```
{0xC8, 0x0A, 0x4A, 0xBF, 0x78, 0xD3, 0xB2, 0x79, 0x98, 0xA0, 0xB6, 0x36, 0xB4, 0x5E, 0x9B, 0xD1}
```

El ataque se ha centrado en la fase 'Subbytes' y se han realizado 20.000 medidas por cada sonda y posición de medida.

Estos son los resultados obtenidos:

Tabla 5.11: Resultados ataque CEMA sobre placa EMA 4

SONDA	POSICIÓN	COEF. CORRELACIÓN	
		VALOR	Nº TRAZAS
EM6995	EM1	-	**-**
	EM2	-	-
	EM3	**0.044**	**17866**
	EM4	-	-
MFA-R	MF1	-	-
	MF2	**0.391**	**655**
	MF3	0.138	3537
	MF4	0.088	5197
HOME	HO1	0.038	*
	HO2	**-**	**-**
	HO3	**0.039**	*
	HO4	-	-

*****: El coeficiente de correlación no alcanzó nunca un nivel doble del valor de la envolvente.

Del análisis de estos resultados se concluye que la mejor opción para realizar un CEMA sobre la placa EMA 4, es utilizar la sonda MFA-R, pues genera mejores resultados y su efectividad es superior.

Capítulo 6

DISCUSIÓN DE RESULTADOS

En este epígrafe se analizan en profundidad los resultados experimentales obtenidos, para tratar de obtener conclusiones acerca del comportamiento de los distintos microcontroladores embebidos de bajo consumo ante los ataques CEMA y se comparan con los estudios publicados en la bibliografía.

6.1 Análisis de Resultados

En la siguiente tabla se muestran, a modo de resumen, todos los resultados obtenidos con las cuatro placas bajo estudio, a fin de analizarlos de forma conjunta:

Tabla 6.1: Resultados Experimentales

SONDA	EMA 1		EMA 2		EMA 3		EMA 4	
	C.C.	TRAZAS	C.C.	TRAZAS	C.C.	TRAZAS	C.C.	TRAZAS
EM6995	0.759	65	0.186	987	-	-	-	-
	0.711	82	0.625	77	-	-	-	-
	0.541	477	0.453	119	0.086	12774	0.044	17866
	0.621	195	0.438	471	-	-	-	-
MFA-R	0.803	104	0.221	752	-	-	-	-
	0.880	58	0.308	289	0.178	2093	0.391	655
	0.840	46	0.413	258	-	-	0.138	3537
	0.844	93	0.291	492	0.074	18467	0.088	5197
HOME	0.678	90	0.353	552	0.059	16637	0.038	*
	0.793	46	0.518	206	0.047	*	-	-
	0.723	60	0.479	301	-	-	0.039	*
	x	x	0.423	206	0.038	*	-	-

*****: No se alcanzó el nivel de coeficiente de correlación mínimo con respecto al resto de coeficientes obtenidos para poder asegurar el éxito del ataque.

-: Se generó un resultado erróneo.

x: La disposición de los componentes de la placa no permitió situar la sonda en otra posición con resultados distintos.

Se pueden obtener varias conclusiones:

1. De las placas analizadas, EMA 3 y EMA 4 resultan ser las más seguras ante un CEMA, seguidas a continuación por EMA2 y EMA1 en este orden.

   A la vista de los resultados mostrados en la Tabla 6.1, no se puede determinar de manera fehaciente, una clasificación correlativa que permita distinguir la placa más segura ante un CEMA.

   Si se tienen en cuenta los mejores valores obtenidos con cada una de las sondas[34], dos realizan el ataque de forma más efectiva sobre la placa EMA3 y una sobre la placa EMA4, como se puede comprobar en la Tabla 6.2 o de forma gráfica en la Figura 6.1.

[34] El análisis de los valores medios por sonda genera conclusiones que no representan fidedignamente los valores obtenidos. Además, las medidas se hicieron en puntos con nivel de radiación máximo, donde los resultados deben ser mejores, aunque esto no se puede asegurar. Por otra parte, en el caso del número de trazas necesarias, no es posible el cálculo del valor medio, puesto que algunas medidas correspondientes a EMA3 y EMA4 no arrojaron un valor de número de trazas.

Tabla 6.2: Valores Máximos de Coeficiente de Correlación y Número de Trazas en función de la sonda utilizada

SONDA	EMA 1		EMA 2		EMA 3		EMA 4	
	C.C.	TRAZAS	C.C.	TRAZAS	C.C.	TRAZAS	C.C.	TRAZAS
EM6995	0.759	65	0.625	77	0.086	12774	0.044	17866
MFA-R	0.880	46	0.413	258	0.178	2093	0.391	655
HOME	0.793	46	0.518	206	0.059	16637	0.039	*

En el caso de realizar el análisis comparativo teniendo en cuenta los valores medios de los valores máximos, determinar qué placa es más segura, EMA3 o EMA4, resulta complicado, pues con EMA4 se consigue un mejor valor de C.C. y con EMA3 el número de trazas necesarias es menor (Tabla 6.3, Figura 6.2):

Tabla 6.3: Valores Medios de Coeficiente de Correlación y Número de Trazas a partir de los valores máximos

EMA 1		EMA 2		EMA 3		EMA 4	
$\overline{C.C.}$	$\overline{TRAZAS}$	$\overline{C.C.}$	$\overline{TRAZAS}$	$\overline{C.C.}$	$\overline{TRAZAS}$	$\overline{C.C.}$	$\overline{TRAZAS}$
0.811	52	0.519	180	0.108	10501	0.158	>12840

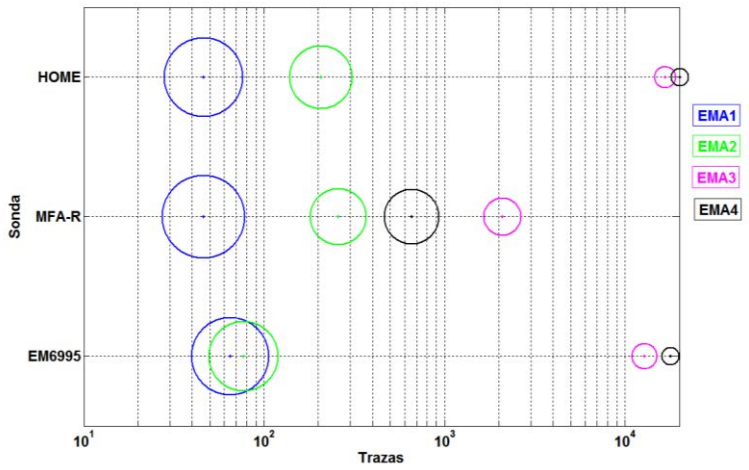


Figura 6.1: Valores máximos ataques
En abscisas se representa el número de trazas en escala logarítmica y en ordenadas la sonda utilizada. El tamaño de los círculos representa el coeficiente de correlación obtenido.

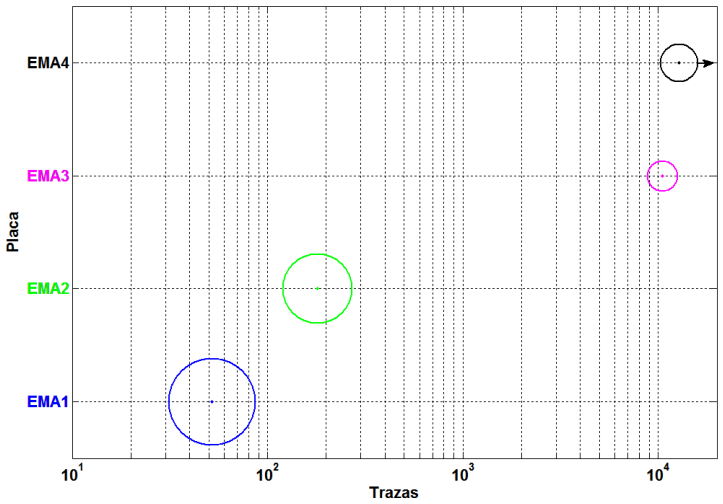


Figura 6.2: Valores medios ataques a partir de valores máximos
En abscisas se representa el número de trazas en escala logarítmica y en ordenadas la placa analizada. El tamaño de los círculos representa el coeficiente de correlación obtenido.

De igual modo, si analizamos la efectividad de los ataques, se obtienen dos grupos, las placas EMA 1 y 2 por un lado, que tienen un 100% de efectividad y, por otro, las placas 3 y 4 que consiguen un 50% de efectividad (Tabla 6.4). En cambio, si se tiene en cuenta la sonda utilizada no se llega a ninguna conclusión (Tabla 6.5).

Tabla 6.4: Efectividad de los Ataques

PLACA	EMA 1		EMA 2		EMA 3		EMA 4	
Efectividad	11/11	100%	12/12	100%	6/12	50%	6/12	50%

Tabla 6.5: Efectividad de los Ataques en función de la sonda utilizada

PLACA	EMA 1		EMA 2		EMA 3		EMA 4	
EM6995	4/4	100%	4/4	100%	1/4	25%	1/4	25%
MFA-R	4/4	100%	4/4	100%	2/4	50%	3/4	75%
HOME	4/4	100%	4/4	100%	3/4	75%	2/4	50%

Por tanto, es necesario determinar un parámetro que permita clasificar los dispositivos analizados en función de su resistencia a los CEMA. Así, se ha definido como novedad el denominado *Coeficiente de efectividad del ataque* C_{EfAt} que como su propio nombre indica, permite calcular el grado de efectividad de un ataque sobre un dispositivo. Éste, tiene en cuenta tanto el coeficiente de correlación obtenido *C.C.*, como el número de trazas necesarias para que el ataque sea exitoso (*Trazas*, calculado tal como se propone en el epígrafe 5.1.1.) y el número de trazas totales utilizadas en el ataque *n*, que en este estudio, en concreto, han sido 20.000:

$$C_{EfAt}(\%) = \frac{\left(C.C. + \frac{(n+1) - Trazas}{n}\right)}{2} \cdot 100 \quad \textbf{(6.1)}$$

Cuanto menor sea este valor, más seguro será el dispositivo, o lo que es lo mismo, menos vulnerable a un ataque.

Aplicando este factor a los resultados de las medidas y hallando la media para cada una de las placas se obtiene que la placa EMA3 es la más segura, como se puede constatar en la Tabla 6.6:

Tabla 6.6: Valor de efectividad del ataque CEMA sobre las placas analizadas

PLACA	EMA 1	EMA 2	EMA 3	EMA 4
V_{EfAt} (%)	86.9	68.6	8.27	14.1

Tras analizar los resultados obtenidos no cabe duda de la gran diferencia de comportamiento existente entre las placas modernas con ARM Cortex, y el resto (ARM7 y 8051). Si se toma como referencia la placa más simple EMA1 con el microprocesador 8051 de 8 bits, y los valores medios obtenidos con cada una de las placas (Tabla 6.3), se obtiene que al menos son necesarias un número de medidas 200 veces superior, dato nada despreciable:

Tabla 6.7: Valores relativos de Coeficiente de Correlación y Número de Trazas respecto de la placa EMA 1 a partir de los valores medios

EMA 1		EMA 2		EMA 3		EMA 4	
$C.C._{Ref}$	$TRAZAS_{Ref}$	$C.C._{Rel}$	$TRAZAS_{Rel}$	$C.C._{Rel}$	$TRAZAS_{Rel}$	$C.C._{Rel}$	$TRAZAS_{Rel}$
1	1	0.64	3.5	0.13	202	0.19	>247

En principio, los resultados mostrados son específicos de las implementaciones realizadas en este estudio. No obstante, dado que las placas utilizadas han sido diseñadas de la forma más simple posible para que influyan mínimamente en los resultados, las conclusiones obtenidas pueden ser extrapoladas a los microcontroladores utilizados. Así ha quedado demostrado que los dispositivos ARM Cortex M3 LPC1769FBD100 y STM32L152RBT6, resultan ser menos prácticos de atacar, que los ARM7TDMI-S LPC2124FBD64 y C8051F303.

Por otro lado, la ejecución del algoritmo AES en los distintos microcontroladores no requiere prácticamente del uso de periféricos[35], por lo que los resultados son en su mayoría atribuibles a la arquitectura del núcleo del microprocesador y la memoria. Por este motivo, extrapolando los resultados a las arquitecturas aquí usadas, se puede concluir que las arquitecturas CIP-51 de Silicon Labs y ARMv4T del ARM7TDMI-S, son más factibles de atacar que la arquitectura ARMv7-M del Cortex M3.

En cuanto al supuesto origen de las diferencias de comportamiento existentes entre arquitecturas, poco se puede decir, dado que su diseño no es accesible al público. Únicamente se pueden hacer suposiciones a partir de los datos publicados. Por ejemplo, el alto rendimiento de la arquitectura Cortex M3 (alrededor de 13 veces superior a un 8051 novedoso y 1.3 veces superior a un ARM7TDMI), su alta integración (con 33000 puertas lógicas en 0.30mm^2 de núcleo, 0.62mm^2 en el caso del ARM7TDMI), la nueva arquitectura de instrucciones Thumb 2 que simplifica el código y lo hace más eficiente, el uso de un bus de datos multicapa denominado matricial o su bajo consumo (0.19mW/MHz, ARM7 0.28mW/MHz) [[Yiu'10], [Sad'06]]. Todos estos elementos pueden favorecer la aparición de ruido en las medidas, entendiendo como tal todas aquellas emisiones no correlacionadas con el dato ejecutado, provocando las diferencias de comportamiento descubiertas. Los acoplamientos con componentes cercanos, el funcionamiento de varios componentes al mismo tiempo o la rapidez de ejecución de instrucciones, pueden favorecer la generación de un ambiente ruidoso, y por tanto, disminuir la eficiencia de los ataques.

Respecto a la comparativa de los dos microcontroladores ARM Cortex M3 estudiados, uno, el LPC1769, con una alta velocidad de procesamiento y otro, el STM32L152, con un diseño orientado a la minimización del consumo de energía, no se ha podido establecer una diferenciación clara en cuanto a su comportamiento ante un CEMA. El hecho de disponer de un microprocesador con menor consumo supone captar señales de menor intensidad, por lo que en este caso, el ruido presente en las medidas debido a diversas fuentes[36], adquiere mayor importancia. Dicho de otro modo, la SNR disminuye. En el caso de aumentar la frecuencia de funcionamiento, la presencia de ruido en las medidas aumenta, puesto que como ya es sabido, todo conductor se convierte en antena a longitudes superiores a $\lambda/30$ metros.

[35] Únicamente se han utilizado los puertos de salida para la comunicación con el PC y el osciloscopio.
[36] En toda medida, ya sea de campo EM para un ataque EMA o de corriente para un PA, siempre existirá ruido procedente de diversas fuentes: físico, de medida o del dispositivo (véase epígrafe 2.2.3).

2. Los resultados obtenidos, refrendan el análisis realizado en el epígrafe 5.1.3.1, donde se estudia la relación existente entre el número de trazas necesarias y el coeficiente de correlación obtenido. Si se ordenan los resultados en función de la sonda y el C.C., queda claro que la obtención de un valor mayor genera un ataque más efectivo, Tabla 6.8.

Tabla 6.8: Resultados Experimentales ordenados en función del C.C.

SONDA	EMA 1		EMA 2		EMA 3		EMA 4	
	C.C.	TRAZAS	C.C.	TRAZAS	C.C.	TRAZAS	C.C.	TRAZAS
EM6995	0.759	65	0.625	77	0.086	12774	0.044	17866
	0.711	82	**0.453**	**119**	-	-	-	-
	0.621	195	0.438	471	-	-	-	-
	0.541	477	0.186	987	-	-	-	-
MFA-R	0.880	58	0.413	258	0.178	2093	0.391	655
	0.844	**93**	0.308	289	**0.074**	**18467**	0.138	3537
	0.840	46	0.291	492	-	-	0.088	5197
	0.803	**104**	0.221	752	-	-	-	-
HOME	0.793	46	0.518	206	0.059	16637	0.039	*
	0.723	60	0.479	301	0.047	*	0.038	*
	0.678	90	0.423	206	0.038	*	-	-
	-	-	**0.353**	**552**	-	-	-	-

*****: No se alcanzó el nivel de coeficiente de correlación mínimo con respecto al resto de coeficientes obtenidos para poder asegurar el éxito del ataque.

-: Se generó un resultado erróneo

Existen valores que se desmarcan del resto, y no siguen la tendencia general (resaltados en negrita en la tabla 6.8). En estos casos, la variación inesperada del C.C. de alguna clave errónea o de la propia clave correcta, provoca estas excepciones. Por ejemplo, las 104 trazas obtenidas para la placa EMA1 con la sonda MFA-R se deben a un aumento del C.C. de algunas claves erróneas durante el intervalo de Trazas 80-140, como se puede verificar en la siguiente figura:

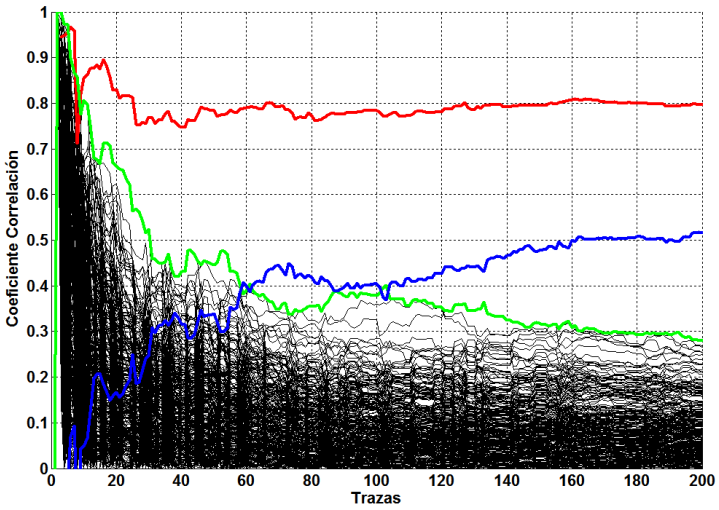


Figura 6.3: Número de trazas EMA1 sonda MFA-R posición MF1

Igualmente, en la siguiente figura se puede comprobar que la desviación producida con la misma sonda pero con la placa EMA3 y las 18467 trazas, se debe al C.C. alcanzado por una clave errónea que se desmarca del resto:

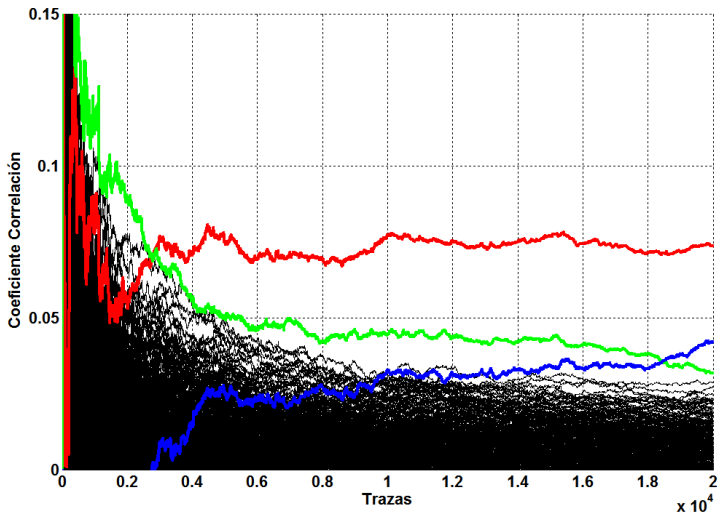


Figura 6.4: Número de trazas EMA3 sonda MFA-R posición MF4

3. Con respecto a las sondas utilizadas se concluye que cualquiera es factible de ser utilizada en un CEMA de forma satisfactoria. Sin embargo, el uso de sondas de calidad de cabeza milimétrica y preamplificador integrado, como la MFA-R utilizada en este estudio, mejoran los resultados, haciendo de media el ataque 10 veces más efectivo, como se desprende de los resultados de la Tabla 6.2, Tabla 6.9 y Tabla 6.10. Por contra está el hecho de que requieren un tiempo de preparación muy superior. En primer lugar se debe seleccionar el punto exacto sobre el que realizar la captura. En el caso concreto de la MFA-R con una resolución espacial de 300 μm, lo que supone ensayar la sonda al menos 1

vez por milímetro cuadrado. De este modo, para atacar un dispositivo como el STM32L152 con encapsulado de 64 pines y 100mm^2, se deberán realizar un mínimo de 100 medidas previas. En segundo lugar, debido a su alta precisión, requieren de unas condiciones de medida muy estables, algo que en determinados escenarios como un ataque real sobre un dispositivo, no siempre es posible.

Tabla 6.9: Valores medios de Coeficiente de Correlación y número de Trazas para cada una de las sondas a partir de los valores máximos

SONDA	C.C.	TRAZAS
EM6995	0.379	7696
MFA-R	0.466	763
HOME	0.352	>9222

Tabla 6.10: Efectividad de los Ataques en función de la sonda

PLACA	EM6995		MFA-R		HOME	
Efectividad	10/16	62%	13/16	81%	13/16	81%

Además, el disponer de una cabeza milimétrica más precisa, que a priori resulta ser una ventaja, pues permite discernir el origen de las emanaciones y/o captar la radiación deseada, en otros casos puede suponer un inconveniente. Si la posición de medida no es adecuada, la captura puede generar únicamente ruido, originándose un ataque nulo como ya indicaba Carlier en [Car'04]. En este sentido el uso de sondas de calidad más grandes supone cierta ventaja. Es decir, si se diera el hipotético caso de que un ataque solo se pudiera realizar con la sonda en una posición y se desconociera el origen de la radiación objetivo, el uso de una sonda milimétrica jugaría en contra.

6.2 Comparación con Estudios Previos

Hasta ahora, en la literatura existente no se había analizado la seguridad relativa de equipos criptográficos actuales de bajo consumo orientados a aplicaciones embebidas ante ataques por canal EM.

La mayor parte de los estudios publicados se centran en FPGAs, Tarjetas Inteligentes o microcontroladores de 8 bits con frecuencias de operación relativamente bajas o sin dar detalles acerca de la velocidad. Por ejemplo, en [Car'04] se experimenta sobre una FPGA Altera Cyclone a 50 MHz; Los autores en [Rea'09] utilizan una FPGA Altera Stratix sin dar detalles de la frecuencia de operación; En [Mat'10] se estudia un AT89C51ED con una señal de reloj externa con frecuencias comprendidas entre 1 y 40 MHz; en [Mey'10] se analiza una

FPGA Virtex II de Xilinx a 24 MHz, en [Li'10] se centran en los CPA sobre una tarjeta inteligente de 8 bits sin indicar más detalles etc.

En [Geb'06] se estudia la efectividad de una contramedida basada en máscaras sobre un microcontrolador de 32 bits ARM7TDMI configurado, en este caso, a una frecuencia de reloj de 40 MHz, que en ese momento suponía uno de los dispositivos de menor consumo del mercado, pues además de otros elementos, su alimentación se realizaba a 3.3V en lugar de los habituales 5V.

En [Tiu'05] los autores desarrollan un nuevo ataque eficaz contra la desincronización de las trazas, denominado Ataque por análisis de frecuencia diferencial: DFA, y demuestran su potencial aplicándolo sobre un ARM7TDMI a 40 MHz y una PDA que ejecutan AES/128. Sin embargo, este estudio no tiene un carácter comparativo, sino más bien de demostración de la efectividad del ataque desarrollado. Así mismo, en el estudio se utiliza una única sonda EM de 1cm sin justificar adecuadamente su elección.

En [Ken'12] se analizan tres dispositivos, un smartphone 4G LTE que ejecuta RSA, una PDA que realiza una multiplicación sobre una curva elíptica (elliptic curve point multiplication) y un teléfono móvil con el AES/128 en su interior. En este caso también sin dar detalles acerca de los mismos. En el artículo, únicamente se verifica la vulnerabilidad de los equipos a los SCA sin realizar un estudio más exhaustivo ni comparativo.

Así pues, en la literatura no existe un estudio como el que aquí se expone, que compare la seguridad de equipos criptográficos embebidos destinados a aplicaciones de bajo consumo ante ataques por correlación EM.

Capítulo 7

MEDIDAS EXPERIMENTALES ADICIONALES

7.1 Ataque DEMA

Previamente a la realización del estudio principal de este trabajo se analizaron los ataques diferenciales DPA/DEMA, originalmente propuestos por Kocher y otros en [Koc'99] para el canal lateral por consumo y el algoritmo DES, y posteriormente extrapolados al canal EM y los algoritmos: COMP128, DES y RSA por Gandolfi y otros en [Gan'01].

Para la realización del mismo se utilizó la placa EMA 1 con el microcontrolador 8051 evolucionado de 8 bits y el estándar AES/128 ejecutándose en su interior. El setup utilizado es el mismo que el de las medidas principales de este trabajo, con la sonda EM de campo cercano EM6995.

En primer lugar, se escogió como función de selección D la salida de la Sbox de la primera ronda, y se determinó de forma aleatoria como objetivo el bit 2 del byte 2. Dado que la clave es conocida (se ha programado en el dispositivo), se determinaron con ayuda de una aplicación Labview desarrollada para ello, 600 textos aleatorios[37], tales que a la salida de la Sbox el bit objetivo está a 1 y otros 600 en los que el bit está a 0. A continuación se ejecutó el AES sobre los textos y se guardaron las trazas asociadas. Se calculó la media de cada uno de

[37] La memoria del C8051F303 permite almacenar 300 textos de 16 bytes, de forma que para la realización del ataque se programaron dos bloques de 300 textos.

los bloques de trazas asociados a los textos en cada uno de los registros y se restaron dando como resultado la Figura 7.1, en la que se aprecia el momento en el que el bit 2 de la clave es computado en el algoritmo. Por último, se comprobó la validez del ataque repitiendo el proceso anterior suponiendo claves incorrectas, obteniéndose gráficas similares a la de la Figura 7.2.

Así, quedó comprobada la validez del método propuesto por Kocher y otros y verificado el setup de medida, que posteriormente sería utilizado para el resto de pruebas.

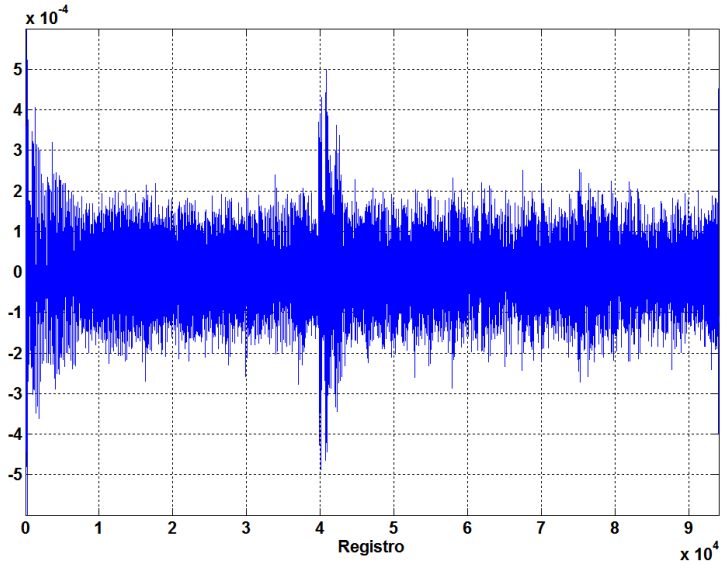


Figura 7.1: Resultado ataque DEMA clave supuesta correcta

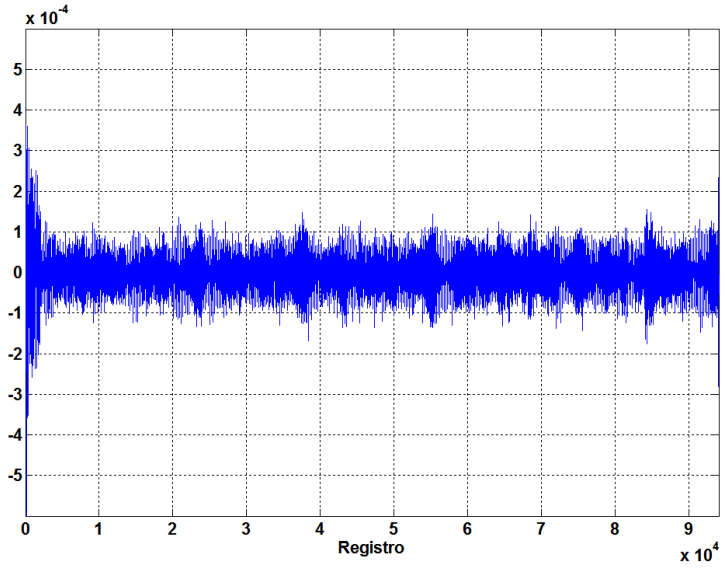


Figura 7.2: Resultado ataque DEMA clave supuesta incorrecta

7.2 Análisis de la EMI generada por las placas

Otro de los puntos posibles que se puede estudiar para intentar comparar los distintos equipos, es analizar la señal EM que generan. Durante la realización de las medidas de este trabajo, se observó en la pantalla del osciloscopio digitalizador una clara discrepancia entre las trazas de las distintas placas, factor que este análisis trata de evaluar.

7.2.1 Análisis estadístico

En la Tabla 7.1 se muestra el análisis estadístico realizado a 1000 trazas EM captadas durante la ejecución de las dos primeras fases del AES, AddRoundKey y Subbytes, sobre 1000 textos distintos, con cada uno de los equipos estudiados en este trabajo.

El estudio consiste en calcular en cada uno de los puntos de las trazas, la desviación típica y varianza de las 1000 trazas y a continuación determinar para cada grupo su media, varianza y desviación típica. Como el tiempo de ejecución es distinto para cada placa y la frecuencia de digitalización es constante e igual a 5 Gmuestras/segundo, la longitud de las trazas varía de un equipo a otro. Así por ejemplo, si representamos de forma matricial el procedimiento para la placa EMA 1, éste se puede expresar como en el esquema de la Figura 7.3.

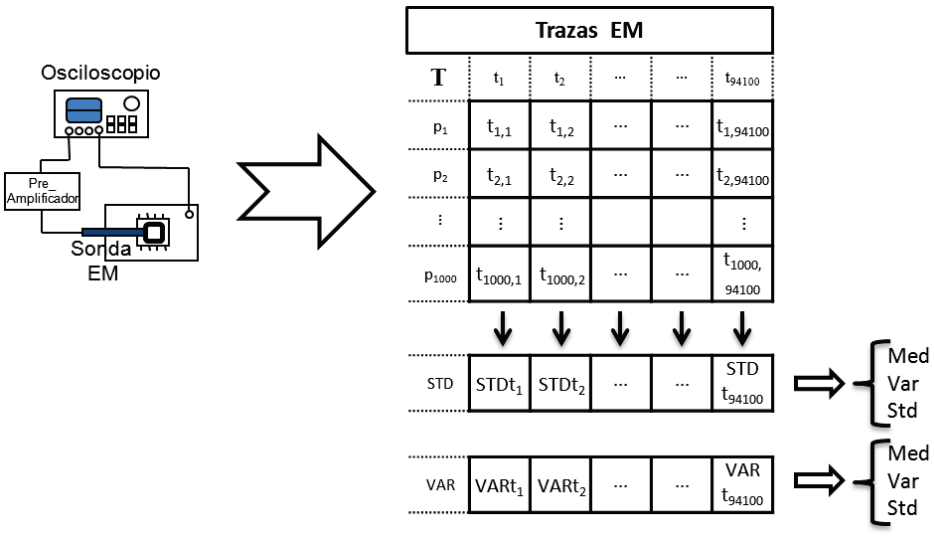


Figura 7.3: Análisis estadístico trazas EM placa EMA 1

Tabla 7.1: Análisis estadístico trazas EM

1000 Trazas EM		DESVIACIÓN TÍPICA			VARIANZA		
PLACA	REGISTROS	MEDIA ($\cdot 10^{-3}$)	VAR ($\cdot 10^{-6}$)	STD ($\cdot 10^{-3}$)	MEDIA ($\cdot 10^{-6}$)	VAR ($\cdot 10^{-9}$)	STD ($\cdot 10^{-6}$)
EMA 1	94.100	4.5	3.9	2	24.1	0.98	31.3
EMA 2	119.000	6.3	4.5	2.1	44.3	3.77	61.4
EMA 3	78.500	70.1	19.5	4.4	4900	427.3	653.7
EMA 4	287.000	29.0	2.3	1.5	844.7	7.6	87.4

En la siguiente figura se muestra de forma gráfica la desviación típica estándar de las 1000 trazas en cada uno de los registros para las cuatro placas:

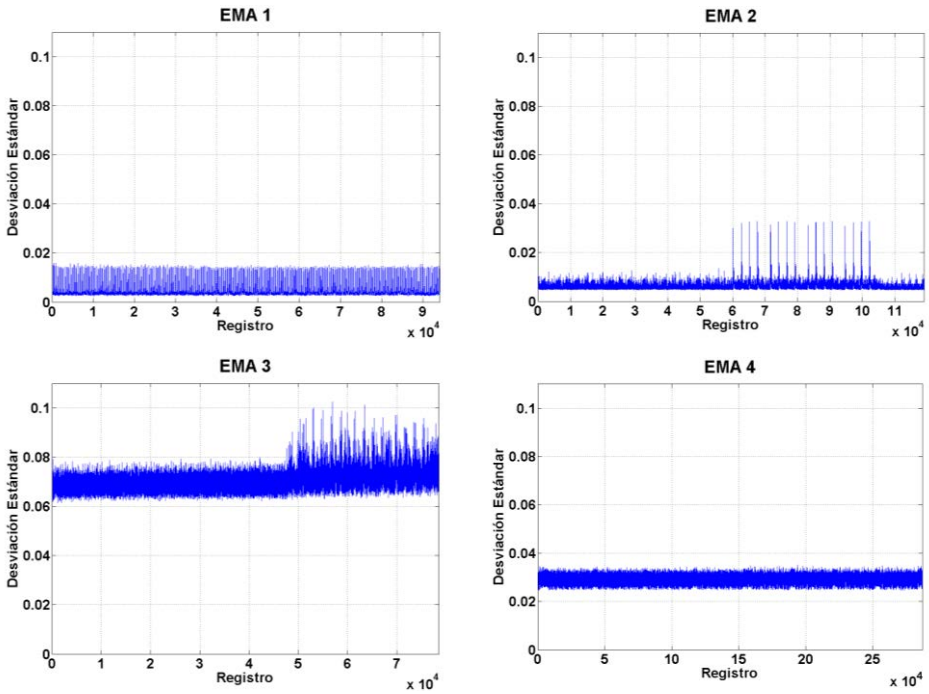


Figura 7.4: Desviación típica estándar 1000 medidas placas bajo estudio

Como se puede apreciar, la desviación típica y, por ende, la varianza, que presentan las trazas EM generadas por las placas EMA4 y sobre todo EMA3, es muy superior a la de las otras dos placas, con una relación de 4 a 1 en el mejor de los casos. Esto significa que existe una mayor inconsistencia en la radiación emitida por estas placas, con unas variaciones de campo magnético superiores. Este hecho puede deberse a dos factores:

a) La componente de ruido presente en las trazas es mayor.

b) La emisión EM depende en gran medida del dato encriptado.

En el caso de que la emisión dependiera del dato encriptado (caso b), los resultados obtenidos en un CPA con estas placas serían mejores que con las otras dos placas, cosa que, como ya se ha comprobado, no ocurre. Por tanto, estos resultados son un indicativo de la presencia de radiación EM no correlacionada con el dato en las trazas, que se considera ruido.

En principio, el hecho de que la variabilidad de las trazas provenientes de la placa EMA 3 sea alta, es lógico, puesto que su frecuencia de funcionamiento de 120 MHz es muy superior a la del resto. Sin embargo, la placa EMA 4 genera trazas con alta desviación típica y, en cambio, trabaja a menor velocidad que EMA 2. Esto pone en evidencia que la arquitectura Cortex M3 tiene un perfil EM más ruidoso, que la hace a priori más resistente frente a los EMA y PA.

Otro hecho a destacar a partir de los resultados obtenidos, es la gran variación presente al final de las trazas EM originadas por EMA2 y 3. Estos registros, se corresponden con la ejecución de la fase Subbytes, única operación no lineal del AES cuyo principal cometido es la adición de ruido, y que además tiene un consumo mayor respecto a la fase AddRoundKey previa, debido a que la instrucción *load* es menos eficiente energéticamente que *xor* [Ito'02], de ahí los picos presentes.

7.2.2 Análisis frecuencial

Por otro lado, se ha realizado un análisis de Fourier de las trazas EM captadas de cada una de las placas con la ayuda de Matlab y la transformada rápida de Fourier o FFT [MathFFT'10]:

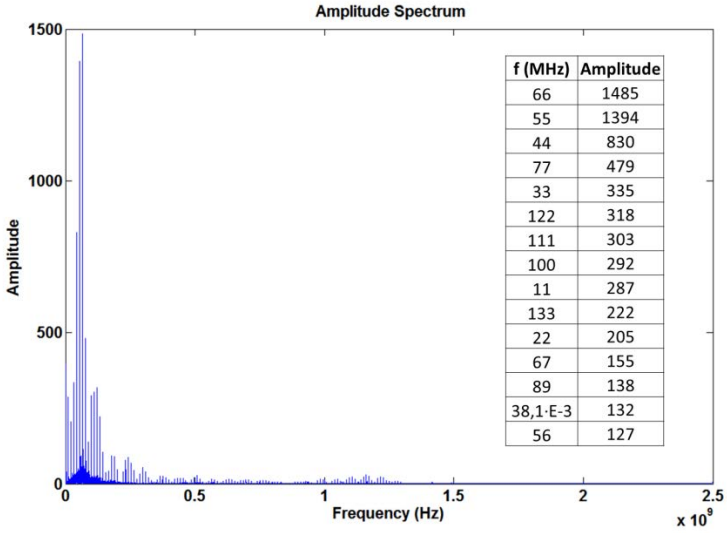
$$X(k) = \sum_{j=1}^{N} x(j) w_N^{(j-1)(k-1)}$$
$$x(j) = \frac{1}{N} \sum_{k=1}^{N} X(k)\, w_N^{-(j-1)(k-1)} \qquad \textbf{(7.1)}$$

Donde:

$$w_N = e^{(-2\pi i)/N}$$

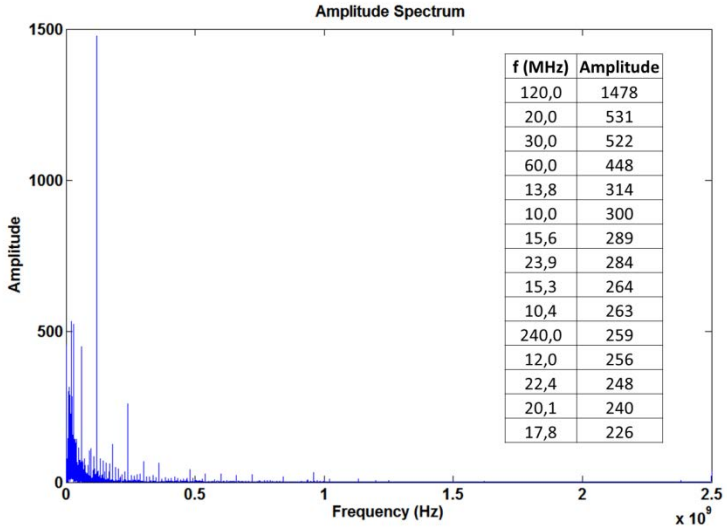
En las siguientes gráficas se detallan los resultados obtenidos[38]:

[38] Téngase en cuenta que la amplitud del espectro mostrada en las gráficas se ha calculado a partir de la función FFT de Matlab sin dividir el resultado por el número de registros de la traza, para de esta forma evitar trabajar con valores muy pequeños. El propósito no es la comparación de los valores absolutos de los armónicos, sino la distribución del espectro en cada una de las placas.



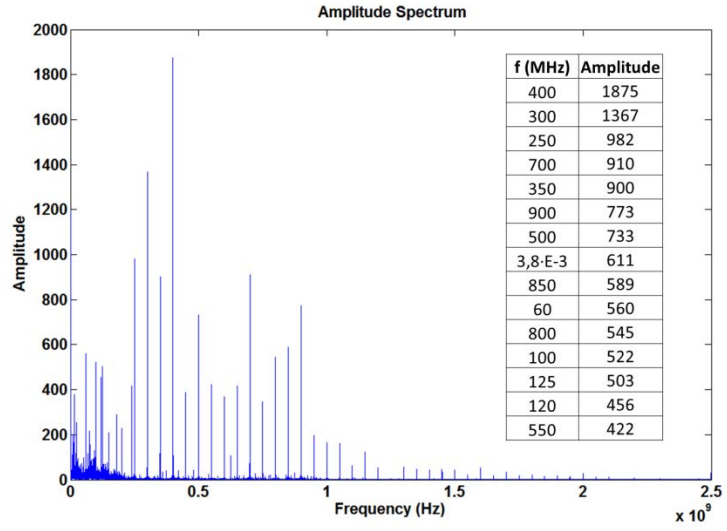
f (MHz)	Amplitude
66	1485
55	1394
44	830
77	479
33	335
122	318
111	303
100	292
11	287
133	222
22	205
67	155
89	138
38,1·E-3	132
56	127

Figura 7.5: Análisis Fourier traza EM placa EMA 1 (11.0592MHz)



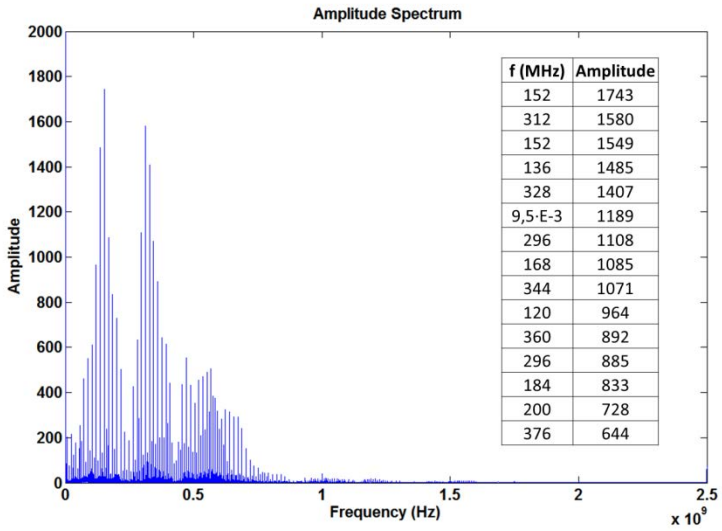
f (MHz)	Amplitude
120,0	1478
20,0	531
30,0	522
60,0	448
13,8	314
10,0	300
15,6	289
23,9	284
15,3	264
10,4	263
240,0	259
12,0	256
22,4	248
20,1	240
17,8	226

Figura 7.6: Análisis Fourier traza EM placa EMA 2 (12x5 = 60 MHz)



f (MHz)	Amplitude
400	1875
300	1367
250	982
700	910
350	900
900	773
500	733
3,8·E-3	611
850	589
60	560
800	545
100	522
125	503
120	456
550	422

Figura 7.7: Análisis Fourier traza EM placa EMA 3 (12x10 = 120 MHz)



f (MHz)	Amplitude
152	1743
312	1580
152	1549
136	1485
328	1407
9,5·E-3	1189
296	1108
168	1085
344	1071
120	964
360	892
296	885
184	833
200	728
376	644

Figura 7.8: Análisis Fourier traza EM placa EMA 4 (8x4 = 32MHz)

A partir de estos resultados se pueden obtener algunas conclusiones:

- Las trazas EM captadas provenientes de las placas bajo estudio tienen un espectro que prácticamente no supera 1 GHz, salvo en EMA 3, que llega hasta los 2 GHz.
- Las trazas de las placas EMA 1 y 2 contienen principalmente armónicos de la frecuencia del reloj. En el caso de las originadas por las placas EMA 3 y 4, el origen no está tan claro, pues existen armónicos que no son submúltiplos de la frecuencia de funcionamiento. Esto puede ser debido a su alta integración que favorece acoplamientos con componentes cercanos y al modo en que está diseñada su CPU, que incluye elementos como un bus matricial, nueva arquitectura de instrucciones thumb-2… [Sad'06]. Estos elementos pueden hacer que la señal EM generada sea más aleatoria e impredecible.
- La placa EMA 4 emite armónicos de frecuencia superior a los generados por EMA 2, siendo su frecuencia de funcionamiento inferior, 32MHz respecto a 60MHz.

7.3 Ataque SEMA

En este trabajo también se ha contemplado la investigación de los ataques "*Simple EM Analysis*" [[Koc'99], [Man'03], [Gan'01]][39]. Aquellos en los que únicamente se realizan varias medidas sobre el EUA ejecutando el algoritmo de encriptación, en este caso AES, y se intenta obtener información a partir de ellas, como instrucciones ejecutadas, datos computados etc. (véase epígrafe 2.1).

[39] Esta disertación se ha realizado en colaboración con Jose Manuel Algaba Alonso, como parte del trabajo final del Máster en Sistemas Electrónicos Avanzados de la Universidad Carlos III de Madrid [AlMa'09].

El equipo necesario para llevar a cabo este análisis ha sido el mismo que el utilizado para la implementación de los CEMA previamente descrito. Únicamente, en algunas medidas se añadió a la cadena de medida un receptor con el fin de captar sólo parte del espectro e intentar eliminar el ruido presente en las medidas. Al ser este equipo altamente sensible, hace innecesario el uso del preamplificador, por lo que en los casos en que se utilizó el receptor, éste no formó parte del setup. Véase la Figura 7.9.

El resultado obtenido con este ataque, ha sido la obtención de información relativa al algoritmo ejecutado en el microcontrolador como tiempo de ejecución total y parcial de cada fase, número de fases implementadas etc. La variación de un bit de la clave o del dato, produce variaciones en las trazas captadas, sin embargo éstas no permiten la identificación de dicho bit.

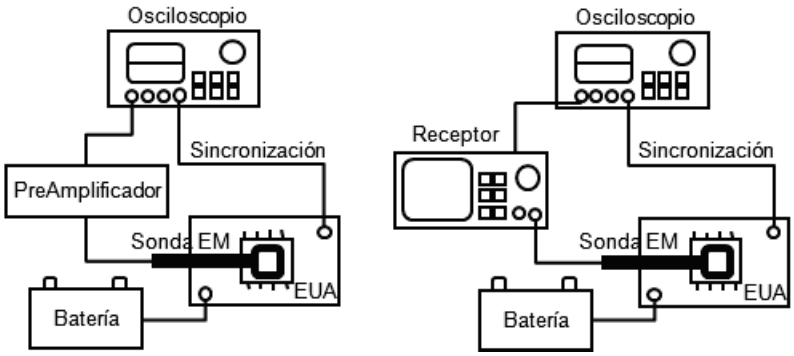


Figura 7.9: Setups de medida utilizados ataque SEMA

Por ejemplo, en la Figura 7.10 se pueden apreciar de forma relativamente clara las 11 fases del estándar de encriptación AES con su tiempo de ejecución. La señal azul representa la señal de trigger, que indica el inicio y fin de la trama a captar:

Figura 7.10: SEMA sobre placa EMA 1 con receptor centrado en 44MHz y RBW=1MHz

Las siguientes imágenes denotan las diferencias presentes en el espectro captado al variar un bit del texto o un bit de la clave sobre la placa EMA 2. El ARM7 TDMI de esta placa se ha programado de forma que realiza continuamente la encriptación de un texto y se varía de forma consecutiva un bit del mismo o un bit de la clave. Así, la diferencia entre dos trazas consecutivas captadas será debida a la variación del bit del texto o clave, ya que el resto de parámetros permanecen inalterados. En este caso se han generado dos pulsos a nivel bajo para identificar las zonas de análisis. El de mayor duración indica el comienzo de la fase de preparación de los datos (trama del espectro más corta) y el menor, la fase de encriptación del texto (trama del espectro de mayor duración).

La primera imagen muestra dos encriptaciones consecutivas en las que no se han realizado modificaciones en texto y clave. Mientras que las siguientes reflejan el efecto que tiene la variación de un bit del texto y clave en la traza captada:

Figura 7.11: Espectro de dos encriptaciones consecutivas sin modificación

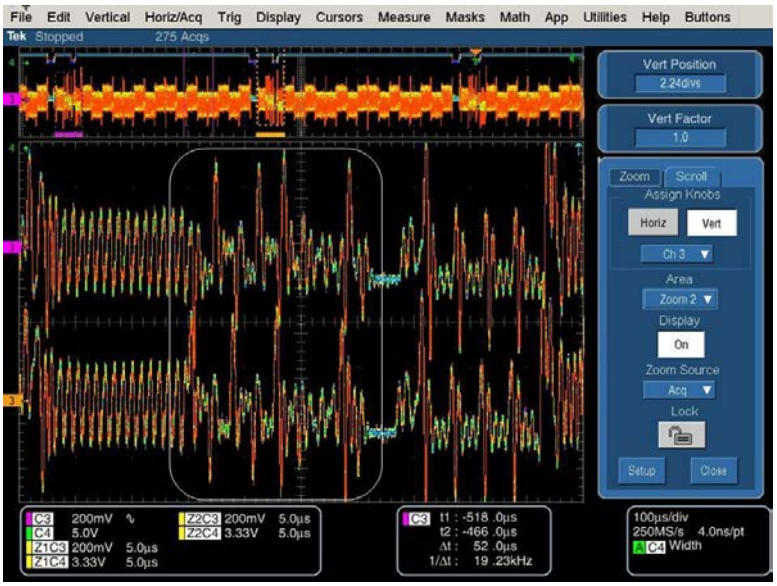


Figura 7.12: Variación espectro captado al variar un bit del texto

Figura 7.13: Variación espectro captado al variar un bit de la clave

Obsérvese que la variación se produce en ambos casos, en el preprocesado (fase del algoritmo AES donde el texto plano a encriptar es guardado en el estado y las subclaves o RoundKeys son generadas y almacenadas, véase Anexo 1), que es la zona entre los pulsos largo y corto. El resto de la traza correspondiente a la ejecución de las distintas fases del AES, que es donde se realiza la encriptación, no ha generado diferencias visuales apreciables significativas.

7.4 Filtrado de Señales

Un método para tratar de hacer más efectivo un ataque EM, consiste en filtrar la señal EM captada por la sonda procedente del EUA. Dado que los métodos estadísticos tienen la

misión de eliminar el ruido presente en la medida, entendiendo ruido como aquellas señales no correladas con el dato a atacar. Se va a tratar de realizar este proceso a través de un filtro, de forma que la fase de procesado sea más liviana y sencilla. Este preprocesado se debe efectuar de forma cuidadosa para no eliminar las componentes correlacionadas de interés.

Algunos autores han implementado esta idea de diversas formas: con filtrado North o de coincidencia [Mes'02], con análisis acumulante [Le'07], con filtros Kalman [Sou'10] o filtros Wavelet [Char'05]. En este caso se ha intentado llevar a la práctica con ayuda de la placa EMA 3 configurada en este caso a 100 MHz, pues teóricamente, al ser su frecuencia de funcionamiento mayor, debe ser la más ruidosa.

Se han aplicado dos métodos:

1. **Filtrado con Receptor**: El primer método aplicado para filtrar la señal ha consistido en utilizar un receptor, tal como propone Agrawal en [Agr'02]. El test setup utilizado es igual al utilizado para llevar a cabo los SEMA de este estudio, Figura 7.14:

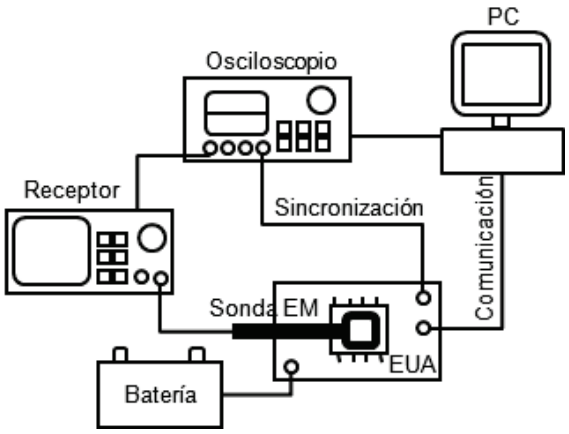


Figura 7.14: Setup medida con receptor trabajando como filtro

El receptor utilizado ha sido un Rohde&Schwarz ESCI [RS'04], el cual posee un conector de salida "20.4 MHz Out", que superpone la señal captada con una señal fija de frecuencia 20.4 MHz correspondiente a su IF (Intermediate Frequency) [Rau'03].

La metodología seguida fue la siguiente. Primeramente, para decidir qué configuración establecer en el receptor, se realizó una medición espectral de la placa EMA3 que determinara las frecuencias más importantes, dando como resultado la gráfica de la Figura 7.15:

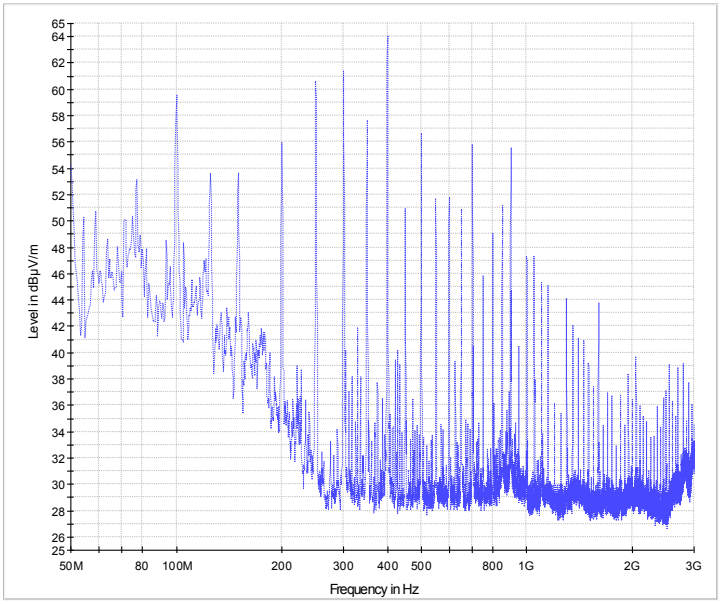


Figura 7.15: Medida Espectral EMA 3 RBW=1MHz, BW=400KHz, ST=50ms

A partir de estos resultados se realizaron diversos CEMA posicionando el receptor a distintas frecuencias detectadas en el paso anterior. Algunas de estas frecuencias resultaron ser armónicos de la frecuencia de reloj de la placa EMA 3, como 400, 300, 250, 100 MHz etc. En todos los casos se utilizó un Resolution Bandwidth RBW=1MHz, detector Max Peak y Sweet Time ST=1ms, puesto que resultó ser la configuración más adecuada[40].

A pesar de todo, los resultados no fueron satisfactorios. Los ataques realizados utilizando este método dedujeron claves incorrectas. El hecho de que la señal analizada tenga superpuesta la señal IF a 20.4 MHz parece influir de manera notable en los resultados.

2. **Filtrado digital en el procesado mediante Matlab**: Otra táctica que se llevó a la práctica fue el filtrado digital de las trazas captadas. Para ello se utilizó Matlab y algunos de los filtros implementados en la aplicación, como Butterworth, Notching, Chevyshev etc. y se establecieron distintas configuraciones como grado, frecuencia de corte etc.

   También en este caso los resultados fueron insatisfactorios.

7.5 Propuesta Ataque Experimental "Bidimensional"

Esta idea es un intento de desarrollar un DEMA novedoso. Se fundamenta en el CEMA, y básicamente consiste en realizar una comparación gráfica de los consumos teóricos

[40] Con valores superiores de RBW la señal captada contenía muchos armónicos y con inferiores la señal se filtraba en exceso.

y reales, en lugar de una comparación apoyada en el Coeficiente de Correlación. El procedimiento se resume a continuación:

1) Se ejecuta el AES sobre una serie de *n* textos aleatorios y se guardan las *n* trazas EM asociadas a la ejecución de la fase bajo ataque.

2) Se selecciona el byte a atacar, y con la ayuda de un modelo de consumo se simula el gasto de energía asociado a ese byte durante la fase bajo ataque para cada uno de los *n* textos aleatorios y para cada una de las 256 posibles claves. Se obtienen así 256 bloques de *n* elementos, que contienen el consumo asociado a la serie de textos aleatorios.

3) Una vez obtenidos estos datos, se disponen en formato gráfica, con el valor del texto: 0 - 255 en abscisas y como ordenada el valor de consumo.

4) Se hace una comparación de la gráfica de consumo real con cada una de las gráficas de consumo simulado para cada uno de los puntos de las trazas EM captadas. Aquella gráfica que presente una mayor similitud determinará la clave hipotética y el tiempo hipotético en el que el byte de la clave es utilizado.

Para la verificación de este nuevo método de ataque propuesto, se realizaron diversos experimentos utilizando la placa EMA 1 con el 8051 mejorado y el test setup desarrollado para llevar a cabo los CEMA previos. En este caso solo se utilizó la sonda EM6995.

En primer lugar, se ejecutó el AES sobre 1000 textos aleatorios de 16 bytes y se captó el campo EM generado por el microcontrolador durante la fase Subbytes de la ronda 1 con un ratio de muestreo de 5 Gs/s. Se obtuvieron 1000 trazas EM (una por cada texto), de 58.600 registros de longitud. A continuación se representó de forma gráfica el valor de cada uno de los registros para las 1000 trazas, obteniéndose, como es lógico, 58.600 gráficas similares a la siguiente:

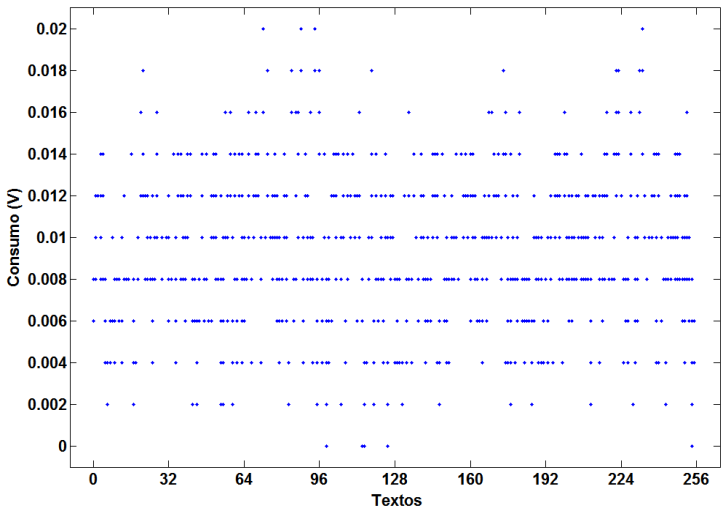


Figura 7.16: Campo EM real captado registro 43027 de las 1000 trazas captadas

Después se calculó, con ayuda del modelo HW, el consumo teórico del microcontrolador al ejecutar la fase Subbytes sobre el byte 4 de los textos aleatorios para cada una de las 256 claves posibles. Estos datos se expresaron de forma gráfica, dando como resultado 256 gráficas HW-Texto, como la Figura 7.17.

Por último, se intentó comparar, sin éxito, las 256 gráficas de consumo teórico con las 58.600 gráficas de consumo real utilizando diversas herramientas estadísticas de comparación como la "Distancia de Mahalanobis o Distancia cuadrática" [[Mah'36], [Mae'00]], "Distancia de Hausdorff" [Hutt'93] y Distancia de Correlación [Sze'07].

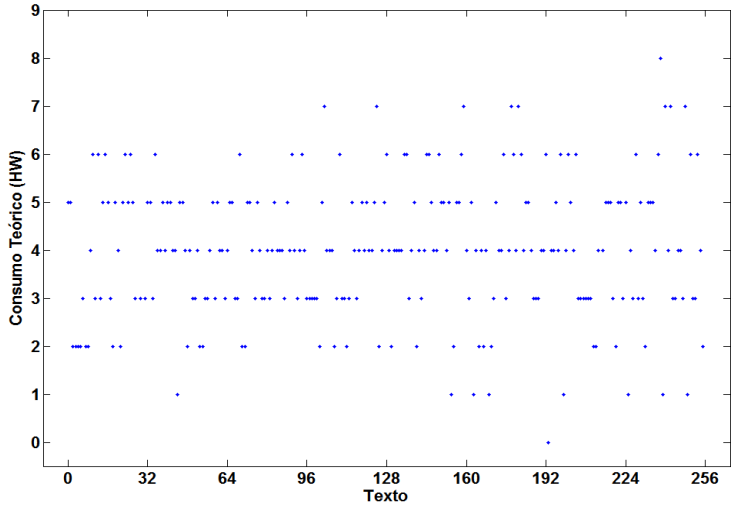


Figura 7.17: Consumo teórico byte 4 de 1000 textos aleatorios para la clave 147_D

7.6 Análisis Wavelet

En los últimos tiempos las Wavelets, [[Tor'98], [Pol'01], [Kai'10]], se han erigido como una herramienta poderosa para el análisis de señales no estacionarias en el dominio tiempo-frecuencia. Su ventaja radica, principalmente, en que la descomposición tiempo-frecuencia que realizan permite separar las componentes de una señal de forma más efectiva y flexible que otros métodos, como la transformada rápida de Fourier [Add'09]. Este aspecto posibilita la detección de señales de baja duración y alta frecuencia al mismo tiempo que señales de larga duración y baja frecuencia.

En este trabajo se ha tratado de aplicar el análisis Wavelet a los EMA centrados en AES para intentar mejorarlos. Hasta ahora en la literatura, las wavelets únicamente aparecían como herramienta de filtrado [Char'05]. En este caso se van a aplicar para tratar de comprobar la posible existencia de relación entre el contenido frecuencial de la señal medida y el valor de la clave y texto usados en el algoritmo de encriptación.

El procedimiento que se ha seguido para intentar demostrar esta posible correlación es el siguiente:

1. Se decide el bit de la clave que se quiere estudiar y se averigua el intervalo de tiempo en el que ese bit es computado dentro del algoritmo de encriptación. Para ello, se ejecuta el algoritmo AES con el bit de la clave a uno sobre un número suficiente de textos aleatorios, por ejemplo 1000, y se repite el proceso cambiando únicamente el bit de la clave, que pasa a valer cero. Se calcula la media de los dos grupos de 1000 medidas para cada uno de los puntos de las trazas captadas y se restan. La gráfica resultante muestra el instante de tiempo en el que el bit de la clave es procesado en el algoritmo. Por ejemplo, en la siguiente figura se pueden ver los puntos de influencia del bit 35 de la clave para la placa EMA 1:

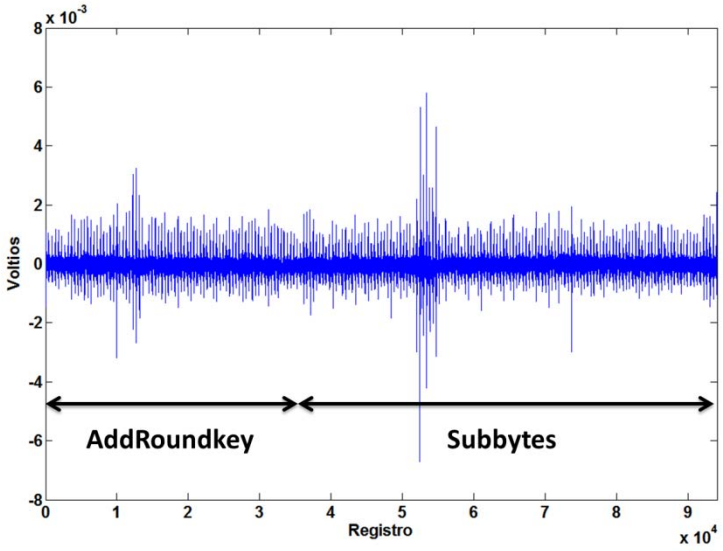


Figura 7.18: Puntos de influencia bit 35 clave algoritmo AES

2. Se toman 1000 textos con todos sus bits aleatorios, salvo el correspondiente al de la clave, que tomará el valor 1 y otros 1000 textos aleatorios también, pero con el bit a 0. Se encriptan y se capturan las trazas asociadas con el bit de la clave a 1 y con el bit de la clave a 0. Se obtienen 4 bloques de 1000 medidas.

3. En los tramos detectados en el apartado 1, se realiza un análisis wavelet, utilizando la transformada continua CWT, tratando de encontrar variaciones frecuenciales ocasionadas por la variación del texto o la clave. En este estudio se ha realizado el análisis sobre medidas individuales y sobre la media de 1000 medidas.

Este análisis se realizó con la placa EMA 1 y con ayuda de la toolbox "Wavelet" de Matlab [MathWav'10]. Se probaron distintos intervalos de señal y todos los tipos de Wavelets

madre disponibles, no obstante, se consiguió más nivel de detalle y, por tanto, una mejor comparación, con: Symlets, Coiflets y Biorthogonal.

A continuación se presentan algunos de los resultados obtenidos:

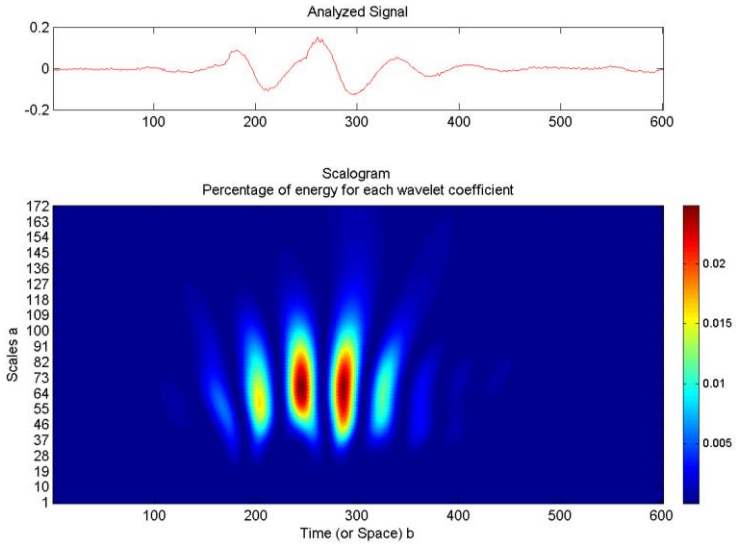


Figura 7.19: Análisis wavelet Symlets 7 bit 63 clave igual a 1

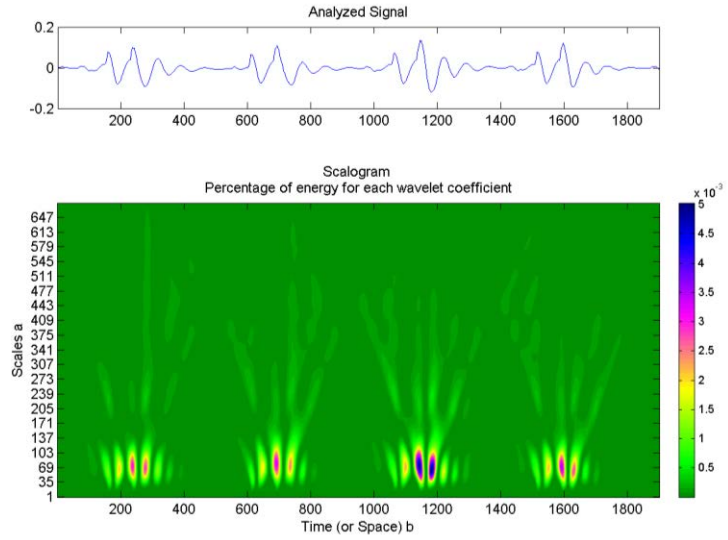


Figura 7.20: Análisis wavelet Coiflets 1 bit 19 clave igual a 0

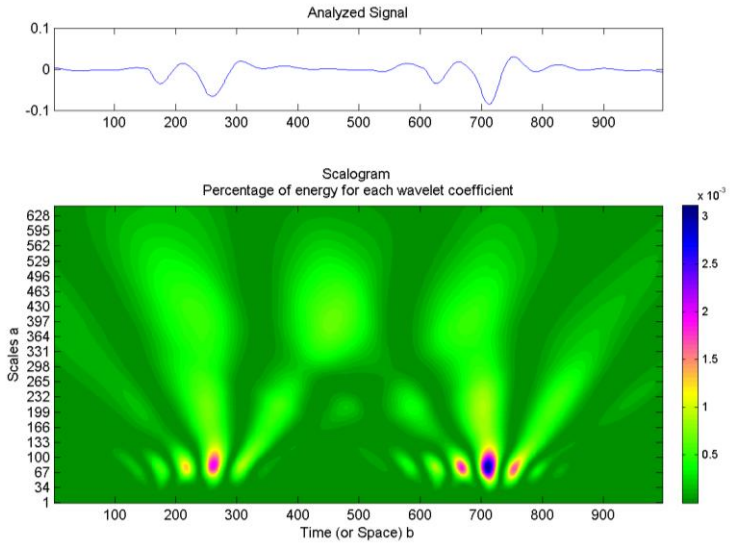


Figura 7.21: Análisis wavelet Biorthogonal 2.8 bit 89 clave igual a 1

Tras los resultados obtenidos no se pudo llegar a una conclusión sólida que relacionara el contenido frecuencial de la señal captada con el valor del bit de la clave o texto.

7.6.1 Creación y aplicación de una Wavelet Propia

También se trató de implementar una wavelet propia que permitiera detecciones de patrón más precisas. De hecho, la señal emitida por la placa EMA1 presenta cierta similitud con algunas wavelets como Daubechies6, Biorthogonal 3.9 y Cmor 1-1, como se puede confirmar en la siguiente figura:

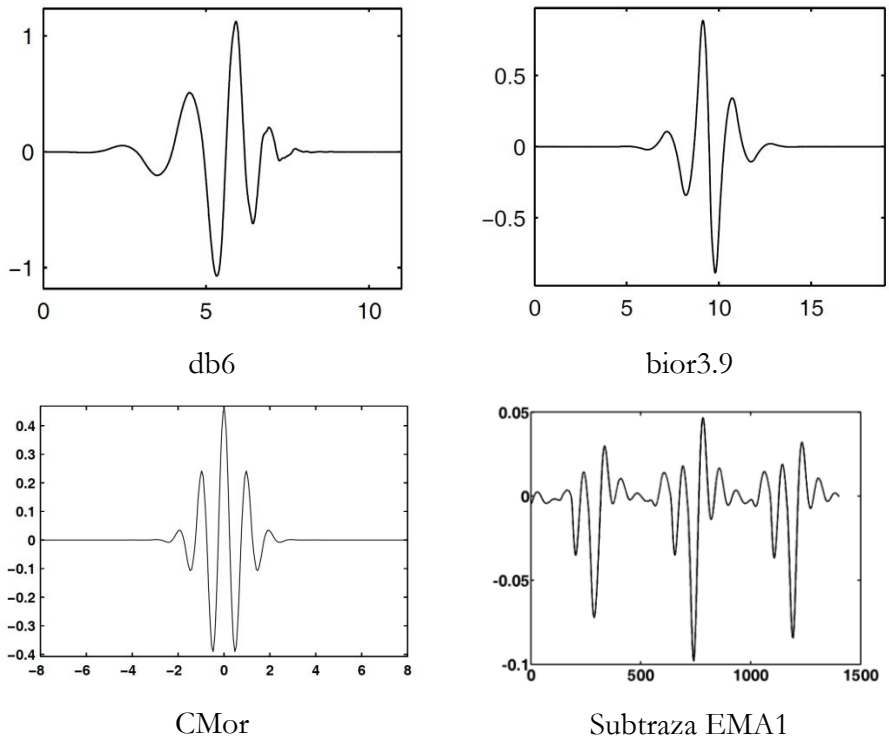


Figura 7.22: Comparación Wavelets con traza EM placa EMA 1

Para la creación de la wavelet propia se utilizó igualmente la toolbox "Wavelet" de Matlab y el método establecido en [Mis'10], el cual define una wavelet admisible a partir de una onda patrón, utilizando optimización basada en mínimos cuadrados.

La señal patrón a partir de la cual se obtiene la wavelet se obtuvo a partir de la media de 1000 trazas EM generadas por la placa EMA1 durante la encriptación textos aleatorios con AES. Dicha operación generó la onda de la siguiente imagen:

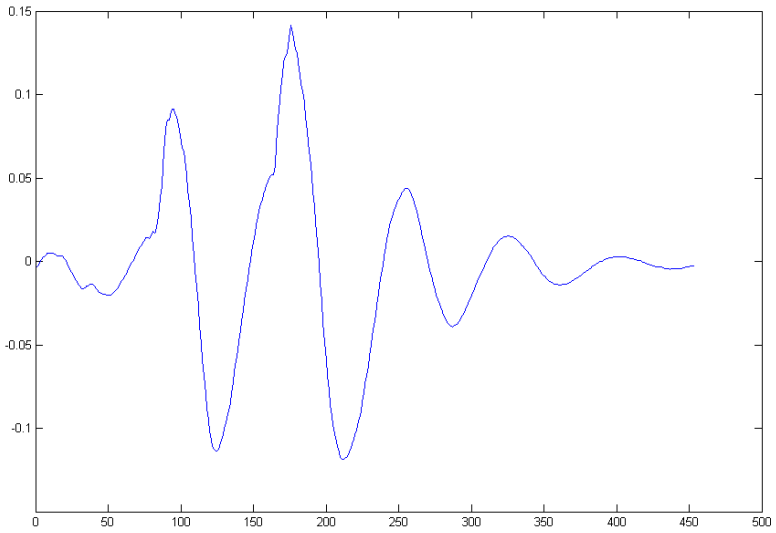


Figura 7.23: Onda patrón EM placa EMA1 para la creación de una Wavelet

A partir del patrón, y con ayuda de la interfaz gráfica de la toolbox de Matlab, se sintetiza gráficamente la wavelet adaptada, de forma que cumpla con la condición de admisibilidad necesaria para su uso con la Transformada Continua Wavelet CWT [Mis'10], Figura 7.24.

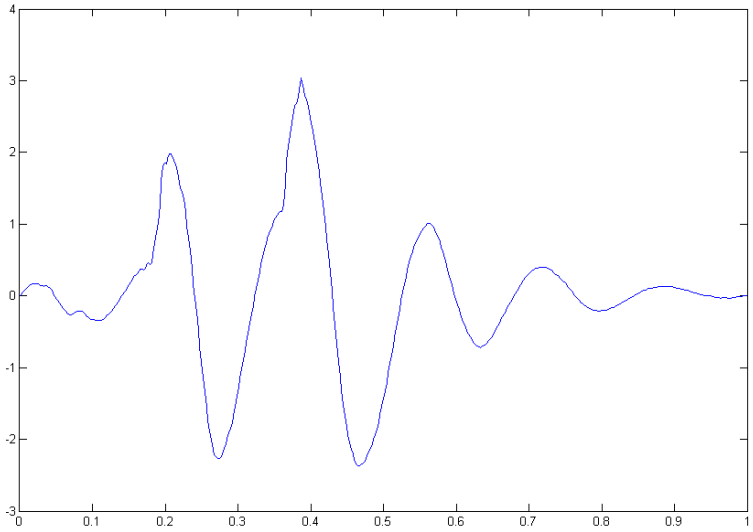


Figura 7.24: Wavelet adaptada a partir de onda base EMA1

Por último, se calcula una función de Fourier que aproxima la wavelet adaptada mediante la herramienta "cftool" de la toolbox de Matlab "fitting". El resultado es la wavelet propia que se utilizó para intentar verificar la idea propuesta:

$$f(x) = a0 + a1*\cos(x*w) + b1*\sin(x*w) + a2*\cos(2*x*w) + b2*\sin(2*x*w) + a3*\cos(3*x*w) + b3*\sin(3*x*w) + a4*\cos(4*x*w) + b4*\sin(4*x*w) + a5*\cos(5*x*w) + b5*\sin(5*x*w) + a6*\cos(6*x*w) + b6*\sin(6*x*w) + a7*\cos(7*x*w) + b7*\sin(7*x*w)$$

Siendo los coeficientes:

a0 = 0.01095 a1 = 0.06336 b1 = 0.07715 a2 = -0.129 b2 = -0.0539
a3 = 0.144 b3 = 0.07383 a4 = -0.4226 b4 = -0.01245 a5 = 0.3949
b5 = -0.6495 a6 = 0.2756 b6 = 0.8781 a7 = -0.3138 b7 = -0.2285 w = 6.086

La aplicación de la Wavelet propia a las trazas EM, tampoco permitió la obtención de una conclusión.

Capítulo 8

CONCLUSIONES Y TRABAJOS FUTUROS

Este epígrafe contiene las conclusiones obtenidas tras la realización de este trabajo, así como aquellas tareas que se considera interesante profundizar en el futuro.

8.1 Conclusiones

En este trabajo se ha realizado por primera vez un análisis comparativo de la seguridad relativa a los ataques por correlación electromagnética CEMA, de cuatro microcontroladores de propósito general, destinados a aplicaciones embebidas de bajo consumo, mediante la implementación del estándar de encriptación AES en ellos.

Los dispositivos analizados, que han sido seleccionados con el fin de obtener el mayor número de conclusiones, son los siguientes:

- ✓ C8051F303: microcontrolador evolucionado de 8 bits, con mejor capacidad de proceso y consumo que otros dispositivos de 8 bits basados en la familia Intel 8x51.
- ✓ ARM7TDMI-S LPC2124: popular microcontrolador de 32 bits, habitual en la literatura, configurado en este caso a una elevada frecuencia de reloj, de 60 MHz.
- ✓ Dos novedosos ARM Cortex M3, nunca antes analizados con respecto a los SCA PA/EMA:

- Un LPC1769, cuya característica principal es su alta capacidad de procesamiento, con una frecuencia de reloj de 120 MHz, muy superior a lo presentado en la literatura.
- Un STM32L152, con un diseño específico orientado al ahorro energético.

Paralelamente a la realización de esta investigación, se han llevado a cabo otros estudios complementarios relativos a distintos aspectos de los ataques EMA/PA. En primer lugar, se han analizado las trazas EM generadas por los dispositivos, concluyéndose que:

- Las trazas EM provenientes de las placas con arquitectura ARM Cortex M3 tienen un perfil estadístico más variable, con armónicos de dudoso origen, aparentemente debidos a acoplamientos.
- Se pone en evidencia que la arquitectura Cortex M3 tiene un perfil EM más ruidoso, que la hace, a priori, más segura frente a posibles ataques EMA que las otras arquitecturas investigadas.

Como novedad, también se ha realizado un análisis comparativo de tres sondas EM de medida de campo cercano ante un ataque CEMA: EM6995 (monoespira de 1cm), MFA-R (milimétrica con preamplificador integrado) y Homemade (fabricada a mano con cable esmaltado):

- Las tres sondas tienen prestaciones suficientes para llevar a cabo un ataque EM, aunque los mejores resultados se obtienen con la sonda milimétrica MFA-R.
- Una simple espira fabricada con hilo esmaltado de cobre es suficiente para llevar a cabo con éxito un ataque CEMA.
- La sonda MFA-R es capaz de detectar señales con una mayor precisión, tanto temporal como espacial, pero tiene el inconveniente de que requiere de un setup más elaborado y su tiempo de puesta en marcha es más lento, al tener que seleccionar los puntos exactos de medida.

Respecto al comportamiento de los dispositivos ante los ataques CEMA, objetivo principal de este trabajo, estas son las conclusiones finales:

- Se ha demostrado que los dispositivos ARM Cortex M3 LPC1769 y STM32L152, resultan ser más difíciles de atacar que los ARM7TDMI-S LPC2124 y C8051F303. Para conseguir un ataque exitoso requieren de media, en el mejor de los casos, un número de trazas unas 60 veces superior.

- Las conclusiones obtenidas con el análisis de las señales EM generadas por los dispositivos, quedan refrendadas con estos resultados.
- No está claro el origen de estas diferencias, puesto que el diseño de los dispositivos no es público y abierto. Si bien, elementos como una mayor eficiencia y rendimiento, o la incorporación de elementos más complejos, como un bus multicapa o un nuevo set de instrucciones, pueden haber sido responsables de ello.
- El C8051F303 ha resultado ser el microcontrolador más desfavorable de cara a un ataque CEMA. La combinación de la arquitectura de 8 bits y la baja frecuencia de operación influyen de manera determinante.
- En cuanto a la comparación entre los dos microcontroladores con arquitectura Cortex M3 y características diferentes, no ha generado resultados concluyentes, si bien el LPC1769 presenta cierta ventaja ante un ataque CEMA. Como consecuencia no se ha podido determinar qué aspecto resulta ser más crítico en el ataque CEMA, si la frecuencia del microprocesador o su consumo.
- A diferencia de lo publicado en la literatura, en este trabajo ha resultado ser más efectivo el modelo de consumo HW. De hecho, el HD solo ha conseguido resultados satisfactorios en el caso de la placa EMA1 de 8 bits, y en este caso se ha mostrado menos robusto y eficiente.

Otros elementos que se han desarrollado durante este trabajo han sido: aplicación del análisis Wavelet a las señales EM captadas, propuesta de un nuevo ataque bidimensional basado en la comparación de señales bidimensionales y preprocesado de las señales EM mediante su filtrado.

Como novedad en este trabajo:

- Se ha analizado por primera vez el comportamiento de la arquitectura ARM Cortex M3 ante los SCA CEMA y se ha verificado que especificaciones de bajo consumo y alta velocidad no presentan diferencias significativas.
- Se ha realizado un análisis comparativo de cuatro dispositivos representativos empotrados de bajo consumo, concluyéndose que los equipos modernos tienen un perfil EM más ruidoso y presentan un comportamiento notablemente superior ante los ataques CEMA.
- Se ha comparado la efectividad relativa de tres sondas EM de campo cercano ante un ataque CEMA, demostrando experimentalmente que una simple espira

fabricada artesanalmente con hilo esmaltado de cobre obtiene resultados similares a los de una sonda comercial. No obstante, es recomendable el uso de sondas de calidad, como la MFA-R, de cabeza preamplificada milimétrica, que permiten una mayor precisión espacial y temporal.

- Se ha construido un test setup de medida específico para la realización de ataques por canal lateral EM: prácticamente automatizado, estable ante vibraciones, económico y con un ratio de muestreo superior al utilizado hasta ahora en la literatura.

- Se ha propuesto un nuevo parámetro, denominado Coeficiente de Efectividad del Ataque, que permite caracterizar a un dispositivo en función de su vulnerabilidad o resistencia ante un ataque por correlación, ya sea EMA o PA.

En conclusión, la seguridad de los sistemas embebidos modernos ante ataques EM es superior a la de dispositivos clásicos. Sin embargo, se debe seguir trabajando en este aspecto. Como se ha demostrado, la realización de un ataque EMA/PA es bastante asequible, no requiere de un equipo excesivamente caro, se puede aplicar sobre cualquier algoritmo criptográfico y prácticamente sin necesidad de conocer el diseño interno del dispositivo bajo ataque.

8.2 Trabajos Futuros

- Analizar la seguridad relativa de dispositivos embebidos con alta capacidad de procesamiento y capaces de soportar un sistema operativo, tales como los ARM Cortex A8, A11 etc.

- Comparar los resultados obtenidos con los Intel Quark, competencia directa de los Cortex M y R, y que presumiblemente saldrán al mercado en Enero de 2014 [IntIdp'13].

- Comprobar de manera fehaciente que los resultados aquí mostrados para el canal EM tienen su réplica en el canal PA.

- En [Char'05] se aplican las wavelets como filtro para mejorar los ataques DPA. Para ello utilizan la familia de wavelets madre 'Symlet', sin justificar apropiadamente la decisión, ni indicar detalles acerca del nivel de descomposición realizado para el filtrado. Se propone analizar el uso de las wavelets como filtro y justificar su uso y configuración, tal como propone Jesús Rubio en su artículo

[Rub'10], donde utiliza el Coeficiente de Correlación de Pearson junto con las wavelets.

- Conseguir que el ataque bidimensional propuesto y estudiado en este trabajo genere resultados satisfactorios, utilizando alguna técnica de comparación bidimensional no ensayada.
- Analizar en profundidad la repercusión de la temperatura en los ataques CEMA y en extensión en la captura de trazas.
- Determinar un indicador o métrica de *benchmarking*, que refleje la seguridad relativa de los dispositivos ante cualquier tipo de ataque, para que llegue a considerarse como un factor más a tener en cuenta a la hora de la adquisición, junto a su rendimiento, consumo, memoria etc. Como ejemplos similares se pueden citar la etiqueta ecológica de los electrodomésticos o neumáticos, o el consumo medio de un automóvil.

Anexo 1

AES

El AES: Advanced Encryption Standard o Rijndael[41] es un algoritmo de cifrado simétrico[42] de bloques, basado en una red de sustituciones y permutaciones. Se obtuvo como resultado de un concurso público del Instituto Nacional de Estándares y Tecnología: NIST, que se realizó desde septiembre de 1997 a octubre de 2000. El 26 de noviembre de 2001 Rijndael fue anunciado como FIPS 197 de los EEUU [Nist'01], aunque no se adoptó como estándar efectivo hasta Mayo de 2002 [Dae'02], sustituyendo a su exitoso predecesor, el DES: Data Encryption Standard, activo desde 1977 y que fue roto en 1997 por RSA Laboratorys, usando la técnica denominada "Fuerza Bruta" con 70.000 computadoras conectadas a Internet durante aproximadamente 96 días [RSA'97]. Actualmente el AES es uno de los algoritmos de criptografía simétrica más ampliamente utilizados [[Ska'11], [WikAES'13]].

Las características que hacen del AES uno de los algoritmos más populares son:

- Rapidez: tanto en implementaciones Hardware como Software.
- Versatilidad y facilidad de implementación en distintas plataformas.
- No requiere muchos recursos.
- Es de dominio público, y por tanto gratuito, pues está libre de patentes.

[41] Derivado del nombre de sus autores: los criptólogos belgas Vincent Rijmen y Joan Daemen.
[42] También llamado de clave privada, el cual utiliza la misma clave para encriptar y desencriptar.

Descripción

El algoritmo consiste en una serie de pasos secuenciales, que modifican un bloque de datos a cifrar de 128 bits, denominado *Estado*[43], con un tamaño de clave de 128, 192 ó 256 bits. Su representación suele hacerse en forma matricial, con un byte en cada elemento y siempre con cuatro filas, para facilitar su interpretación y los cálculos.

Todos los bytes son interpretados como polinomios pertenecientes a un campo finito $GF(2^8)$[44], de forma que un byte genérico $B=\{b_7, b_6, b_5, b_4, b_3, b_2, b_1, b_0\}$ representaría el polinomio:

$$b_7x^7 + b_6x^6 + b_5x^5 + b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0$$

Este dato se debe tener en cuenta a la hora de realizar operaciones como la suma y la resta, puesto que no se realizan de la misma forma que con los números [Nist'01].

Supongamos que se quiere cifrar el texto siguiente de 16 bytes: "Electromagnetico" que en hexadecimal equivale a: "45 6C 65 63 74 72 6F 6D 61 67 6E 65 74 69 63 6F". El estado quedaría de la siguiente forma:

P_0	P_1	P_2	P_3
45	**74**	**61**	**74**
6C	**72**	**67**	**69**
65	**6F**	**6E**	**63**
63	**6D**	**65**	**6F**

Las fases del algoritmo para una longitud de clave de 128 bits son las siguientes[45], que esquemáticamente se pueden representar tal como se muestra en la Figura A1.1:

1. KeyExpansion
2. AddRoundKey
3. Rondas (x 9)
    a. SubBytes
    b. ShiftRows
    c. MixColumns
    d. AddRoundKey

[43] Es posible utilizar otros tamaños de bloque: 192 ó 256 bits, no obstante, el estándar no lo contempla.
[44] $GF(2^8)$: campo finito o Galois Field que contiene los 256 polinomios irreducibles distintos de grado menor o igual que ocho.
[45] Para un tamaño de clave distinto, el número de rondas (Rounds) varía: 128 bits → 9 rondas, 192 bits → 11 rondas y 256 bits → 13 rondas. Al final se ejecuta siempre una ronda final (Final Round).

4. Ronda Final
    a. SubBytes
    b. ShiftRows
    c. AddRoundKey

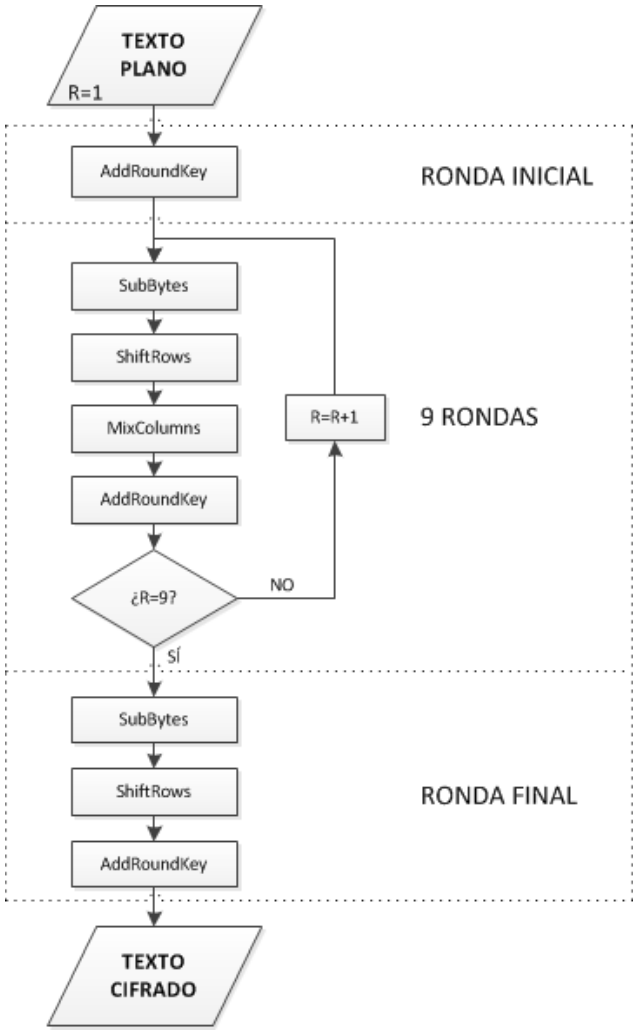


Figura A1. 1: Diagrama algoritmo AES

1. KeyExpansion: Se puede decir que es una prefase que se realiza al inicio del algoritmo, aunque suele ser habitual, por temas de economización de espacio en memoria, ejecutarla cuando el algoritmo lo requiere (just in time). A partir de la clave original se obtiene una clave expandida de mayor tamaño, que es la que luego se utiliza en el resto de pasos. Es lo que se denomina generación de las Claves de Ronda o RoundKeys.

Se parte de una clave de 128 bits en hexadecimal representada de forma matricial del mismo modo que el estado. Por ejemplo, supongamos que la clave es "Universidad_UC3M". En hexadecimal equivaldría a: "55 6E 69 76 65 72 73 69 64 61 64 5F 55 43 33 4D":

W_{i-4}	W_{i-3}	W_{i-2}	W_{i-1}
55	65	64	55
6E	72	61	43
69	73	64	33
76	69	5F	4D

Primero se toma la última columna W_{i-1} y se realiza una rotación hacia la izquierda o *RotWord*. A continuación se le realiza la operación *SubBytes*, en la que se sustituyen los bytes por otros según una tabla de sustitución, LUT o también S-Box. Por último se realiza una operación Xor entre la primera columna de la clave W_{i-4}, el resultado obtenido de la operación SubBytes y la columna Rcon[r] = $[2^{r-1}{}_{hex}; 00; 00; 00]$, donde r es el número de ronda o iteración. Se obtiene así, la primera columna de la expansión de la clave: W_i. Véase la Figura A1.2.

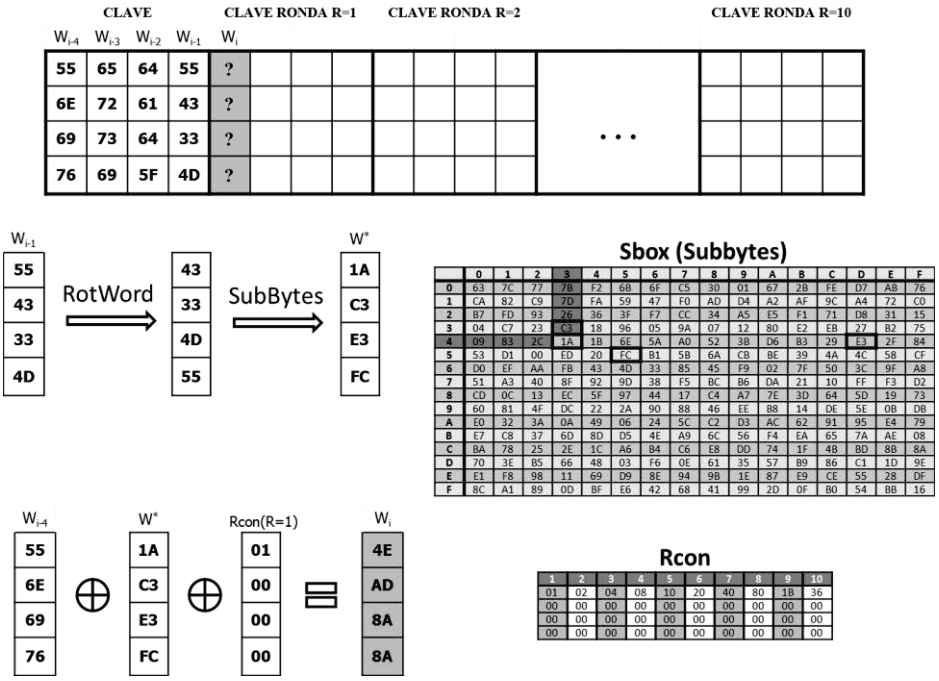


Figura A1. 2: Expansión Clave AES: Obtención columna 1 Clave Ronda 1

Las siguientes tres columnas de la ronda 1 se obtienen todas realizando respectivamente una Xor de la cuarta columna anterior W_{i-4} con la anterior W_{i-1}, Figura A1.3.

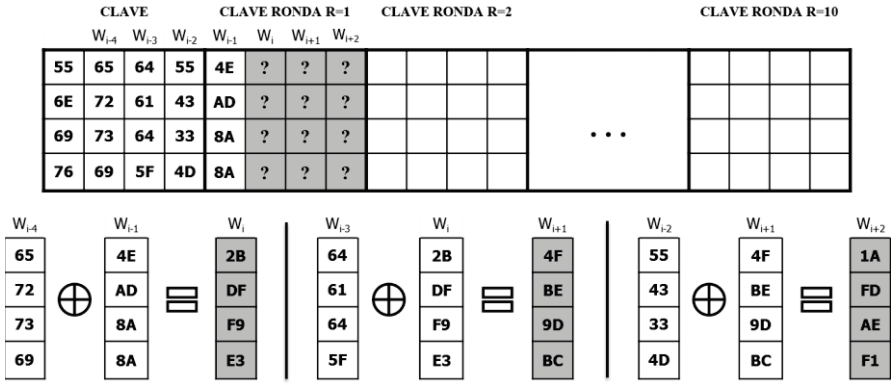


Figura A1. 3: Expansión Clave AES: Obtención columnas 2, 3 y 4 Clave Ronda 1

Este proceso se repite 9 veces más, es decir el número de rondas del algoritmo, obteniendo como resultado las RoundKeys: bloques de 128 bytes, que se aplicarán en las rondas del bloque de cifrado, Figura A1.4.

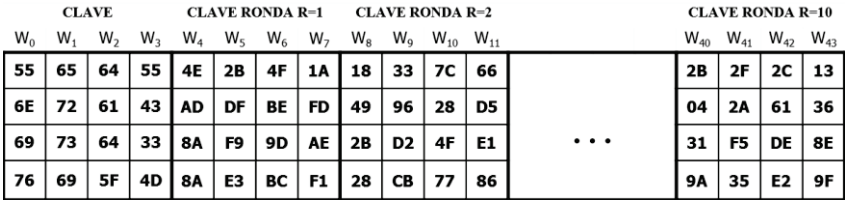


Figura A1. 4: Expansión de clave AES: Generación de Claves Rondas

2. AddRoundKey: Esta operación añade la clave al bloque de datos inicial o estado inicial, mediante una operación Xor. Consiste en realizar una Xor byte a byte, Figura A1.5.

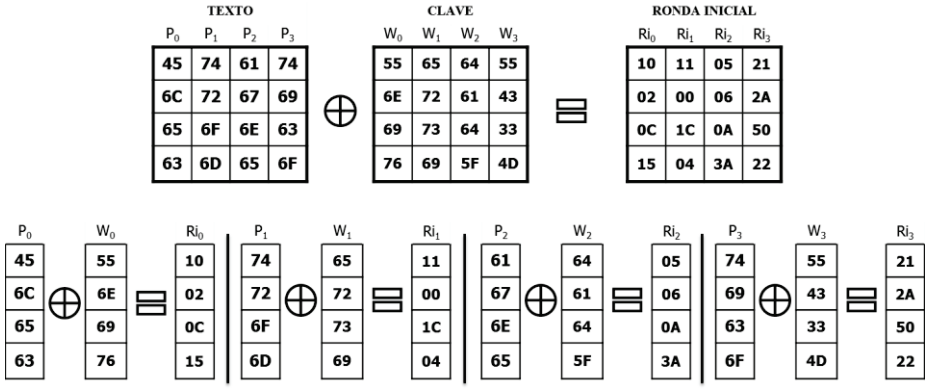


Figura A1. 5: Fase Inicial AddRoundKey AES

3. Rondas: Fase compuesta por cuatro pasos intermedios que se repiten 9 veces.

a) SubBytes → Se basa en una sustitución no lineal de cada uno de los bytes del estado anterior de acuerdo a una Lut o S-Box[46], tal como se hizo en la fase KeyExpansion, Figura A1.6.

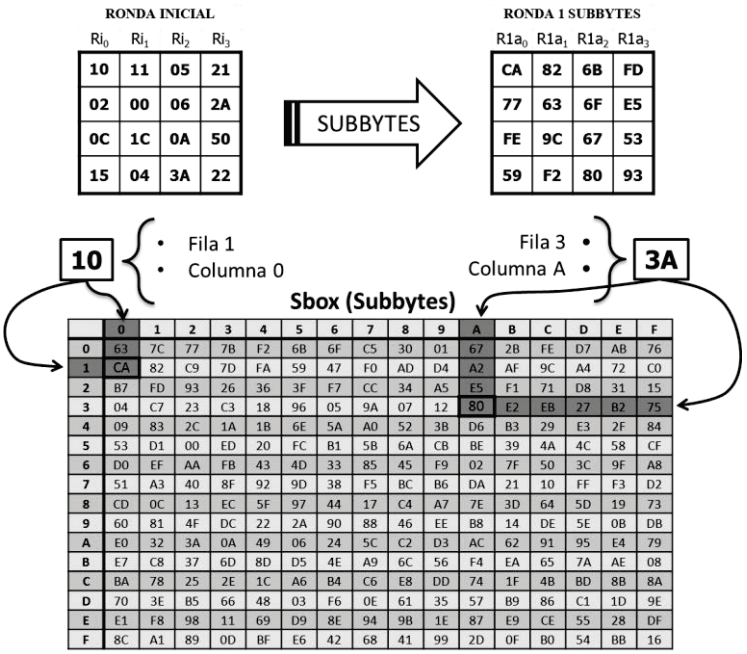


Figura A1. 6: Rondas AES, Fase Subbytes

b) ShiftRows: Como su nombre indica se trata una transformación que opera sobre las filas. Realiza un desplazamiento hacia la izquierda de cada fila del estado, un número de veces determinado. De un byte para la fila dos, de dos bytes para la tres y de tres bytes para la última fila. La primera fila no se modifica. Véase la Figura A1.7.

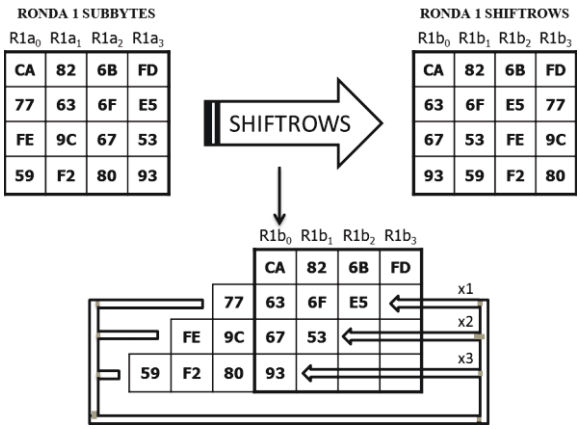


Figura A1. 7: Rondas AES, Fase ShiftRows

[46] La S-Box o LUT utilizada proviene de la función inversa alrededor del GF(2^8) famosa por sus propiedades de no linealidad.

c) MixColumns → Es una transformación de mezclado lineal invertible, que intenta difuminar el resultado. Opera con las columnas del estado, aplicándoles una transformación lineal, consistente en una multiplicación por un polinomio fijo A(x). Véase la Figura A1.8. Consigue que cualquier variación de un bit afecte a todos los resultados.

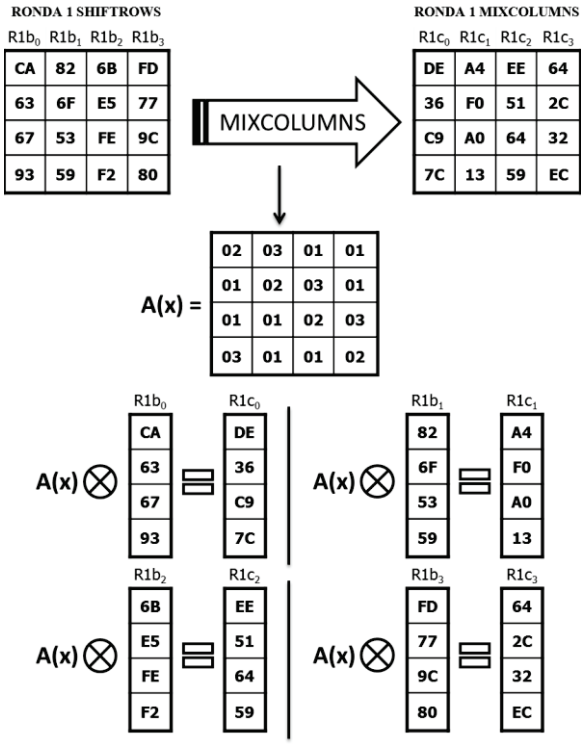


Figura A1. 8: Ronda AES, Fase MixColumns

d) AddRoundKey → El estado se combina con los bytes de la clave expandida, obtenida en el paso 1, usando la operación xor a nivel de byte, Figura A1.9. En esta primera ronda se utiliza la RoundKey 1, en siguientes rondas se irán cogiendo sucesivamente el resto de RoundKeys.

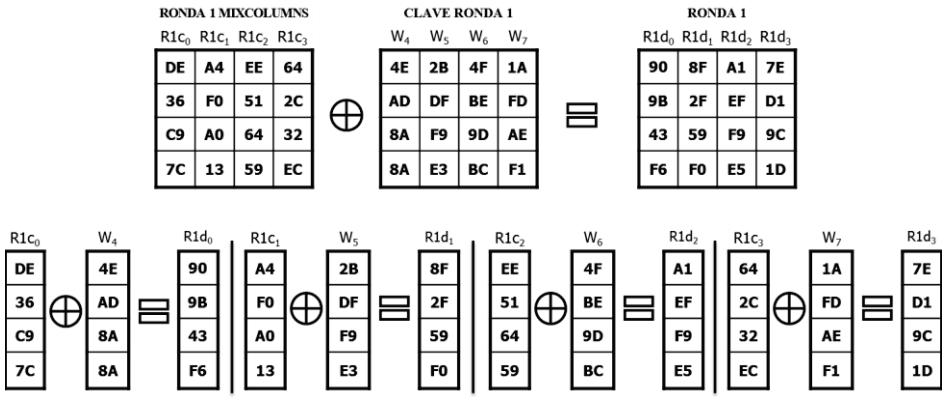


Figura A1. 9: Rondas AES, Fase AddRoundKey

Esta fase, es aplicada al estado 8 veces más. En total, dado que la clave es de 128 bits, 9 veces.

4. Final Round → Fase igual a las rondas intermedias, pero sin la etapa MixColumns. Genera el texto cifrado:

C_0	C_1	C_2	C_3
6E	**14**	**0D**	**5C**
E4	**8F**	**1F**	**80**
8A	**48**	**F3**	**9D**
75	**E5**	**32**	**2C**

Cuya representación en ASCII es: "nõèu¶ÅHÕ♪▼¾2\ÇØ,"

Anexo 2

CÓDIGOS AES IMPLEMENTADOS

En este capítulo se muestran los códigos AES implementados en las placas bajo estudio. La placa EMA 1, se ha programado en lenguaje ensamblador y el resto en ANSI C.

Para no extender en exceso este capítulo se incluirán solo las rutinas referentes al algoritmo AES; El resto se obviarán, tales como el código de configuración de la placa, rutina principal etc. Así mismo, dado que el código en lenguaje ANSI C implementado en las tres placas es muy similar (únicamente hay variaciones derivadas de la utilización de un entorno de programación distinto y por la configuración de la placa) solo se mostrará una versión y las variaciones importantes del código.

CÓDIGO EMA 2, 3 y 4: ARM 7 y CORTEX M3 (ANSI C)

```
// Número de columnas del State
#define Nb 4
// Numero de bits de la clave usada: 128, 192 o 256
#define LengthKey 128
// Numero de Textos de 16 bytes a encriptar en bucle
#define PtextsBucle 1000

// Número de rondas del AES
unsigned int Nr=0;
// Número de palabras de 32 bits de la clave
unsigned int Nk=0;

// Plaintexts a encriptar
unsigned char const Ptexts [PtextsBucle][LengthKey/8] =
{
{0xE4,0x57,0x89,0x74,0x2D,0x8A,0x01,0xFA,0x15,0x4E,0xA0,0x4A,0xA8,0xBD,0x61,0x62},
{0xA3,0xBF,0x16,0xCA,0x7A,0x69,0xC8,0x45,0x90,0xEB,0xA8,0x4B,0x33,0x64,0xB4,0xA2},
{0xAF,0xE6,0xC1,0x61,0xAC,0x4D,0x7B,0xF9,0xC6,0x40,0xFA,0x20,0x7F,0xF8,0x4B,0x54},
...

// Vector de salida con el texto encriptado
unsigned char out[16];
// Estado que almacena los resultados intermedios
unsigned char state[4][4];

// Rondas de las claves
unsigned char RoundKey[240];

// Clave
unsigned char Key[32] = {0xC8, 0x0A, 0x4A, 0xBF, 0x78, 0xD3, 0xB2, 0x79, 0x98,
                         0xA0, 0xB6, 0x36, 0xB4, 0x5E, 0x9B, 0xD1};

unsigned char const SBox[256] =   {
//0     1    2     3    4    5    6    7    8    9    A    B    C    D    E    F
0x63,0x7c,0x77,0x7b,0xf2,0x6b,0x6f,0xc5,0x30,0x01,0x67,0x2b,0xfe,0xd7,0xab,0x76,//0
0xca,0x82,0xc9,0x7d,0xfa,0x59,0x47,0xf0,0xad,0xd4,0xa2,0xaf,0x9c,0xa4,0x72,0xc0,//1
0xb7,0xfd,0x93,0x26,0x36,0x3f,0xf7,0xcc,0x34,0xa5,0xe5,0xf1,0x71,0xd8,0x31,0x15,//2
0x04,0xc7,0x23,0xc3,0x18,0x96,0x05,0x9a,0x07,0x12,0x80,0xe2,0xeb,0x27,0xb2,0x75,//3
0x09,0x83,0x2c,0x1a,0x1b,0x6e,0x5a,0xa0,0x52,0x3b,0xd6,0xb3,0x29,0xe3,0x2f,0x84,//4
0x53,0xd1,0x00,0xed,0x20,0xfc,0xb1,0x5b,0x6a,0xcb,0xbe,0x39,0x4a,0x4c,0x58,0xcf,//5
0xd0,0xef,0xaa,0xfb,0x43,0x4d,0x33,0x85,0x45,0xf9,0x02,0x7f,0x50,0x3c,0x9f,0xa8,//6
0x51,0xa3,0x40,0x8f,0x92,0x9d,0x38,0xf5,0xbc,0xb6,0xda,0x21,0x10,0xff,0xf3,0xd2,//7
0xcd,0x0c,0x13,0xec,0x5f,0x97,0x44,0x17,0xc4,0xa7,0x7e,0x3d,0x64,0x5d,0x19,0x73,//8
0x60,0x81,0x4f,0xdc,0x22,0x2a,0x90,0x88,0x46,0xee,0xb8,0x14,0xde,0x5e,0x0b,0xdb,//9
0xe0,0x32,0x3a,0x0a,0x49,0x06,0x24,0x5c,0xc2,0xd3,0xac,0x62,0x91,0x95,0xe4,0x79,//A
0xe7,0xc8,0x37,0x6d,0x8d,0xd5,0x4e,0xa9,0x6c,0x56,0xf4,0xea,0x65,0x7a,0xae,0x08,//B
0xba,0x78,0x25,0x2e,0x1c,0xa6,0xb4,0xc6,0xe8,0xdd,0x74,0x1f,0x4b,0xbd,0x8b,0x8a,//C
0x70,0x3e,0xb5,0x66,0x48,0x03,0xf6,0x0e,0x61,0x35,0x57,0xb9,0x86,0xc1,0x1d,0x9e,//D
0xe1,0xf8,0x98,0x11,0x69,0xd9,0x8e,0x94,0x9b,0x1e,0x87,0xe9,0xce,0x55,0x28,0xdf,//E
0x8c,0xa1,0x89,0x0d,0xbf,0xe6,0x42,0x68,0x41,0x99,0x2d,0x0f,0xb0,0x54,0xbb,0x16 //F
};


; ********************************************************************************
; KeyExpansion
; Obtiene las RoundKeys
; In:    Key -> Clave
; Out:   RoundKey -> Claves de las rondas
; ********************************************************************************
void KeyExpansion(void)
{
    unsigned int i,j;
    unsigned char temp[4],k;

    for(i=0;i<Nk;i++)
    {
        RoundKey[i*4]=Key[i*4];
        RoundKey[i*4+1]=Key[i*4+1];
        RoundKey[i*4+2]=Key[i*4+2];
        RoundKey[i*4+3]=Key[i*4+3];
    }

    while (i < (Nb * (Nr+1)))
    {
        for(j=0;j<4;j++)
        {
            temp[j]=RoundKey[(i-1) * 4 + j];
        }
        if (i % Nk == 0)
        {
            {
                k = temp[0];
                temp[0] = temp[1];
                temp[1] = temp[2];
                temp[2] = temp[3];
                temp[3] = k;
```

```
            }

            {
                temp[0]=SBox[temp[0]];
                temp[1]=SBox[temp[1]];
                temp[2]=SBox[temp[2]];
                temp[3]=SBox[temp[3]];
            }

            temp[0] =  temp[0] ^ Rcon[i/Nk];
        }
        else if (Nk > 6 && i % Nk == 4)
        {
            {
                temp[0]=SBox[temp[0]];
                temp[1]=SBox[temp[1]];
                temp[2]=SBox[temp[2]];
                temp[3]=SBox[temp[3]];
            }
        }
        RoundKey[i*4+0] = RoundKey[(i-Nk)*4+0] ^ temp[0];
        RoundKey[i*4+1] = RoundKey[(i-Nk)*4+1] ^ temp[1];
        RoundKey[i*4+2] = RoundKey[(i-Nk)*4+2] ^ temp[2];
        RoundKey[i*4+3] = RoundKey[(i-Nk)*4+3] ^ temp[3];
        i++;
    }
}


; ******************************************************************************
; AddRoundKey
; Aplica la función AddRoundKey sobre el State
; ******************************************************************************
void AddRoundKey(unsigned char round)
{
    unsigned int i,j;
    for(i=0;i<4;i++){
        for(j=0;j<4;j++){
            state[j][i] ^= RoundKey[round * Nb * 4 + i * Nb + j];
        }
    }
}


; ******************************************************************************
; SubBytes
; Aplica la función SubBytes sobre el State
; ******************************************************************************
void SubBytes(void)
{
    unsigned int i,j;
    for(i=0;i<4;i++){
        for(j=0;j<4;j++){
            state[i][j] = SBox[state[i][j]];
                }
    }
}


; ******************************************************************************
; ShiftRows
; Aplica la función ShiftRows sobre el State
; ******************************************************************************
void ShiftRows(void)
{
    unsigned char temp;

    temp=state[1][0];
    state[1][0]=state[1][1];
    state[1][1]=state[1][2];
    state[1][2]=state[1][3];
    state[1][3]=temp;

    temp=state[2][0];
    state[2][0]=state[2][2];
    state[2][2]=temp;
    temp=state[2][1];
    state[2][1]=state[2][3];
    state[2][3]=temp;

    temp=state[3][0];
    state[3][0]=state[3][3];
    state[3][3]=state[3][2];
    state[3][2]=state[3][1];
    state[3][1]=temp;
}


; ******************************************************************************
; xtime
```

```
; Macro que calcula el producto de x con {02} modulo {1b}
; ***************************************************************************
#define xtime(x)   ((x<<1) ^ (((x>>7) & 1) * 0x1b))


; ***************************************************************************
; MixColumns
; Aplica la función MixColumns sobre el State
; ***************************************************************************
void MixColumns(void)
{
    unsigned int i;
    unsigned char Tmp,Tm,t;
    for(i=0;i<4;i++)
    {
        t=state[0][i];
        Tmp = state[0][i] ^ state[1][i] ^ state[2][i] ^ state[3][i] ;
        Tm = state[0][i] ^ state[1][i] ; Tm = xtime(Tm); state[0][i] ^= Tm ^ Tmp ;
        Tm = state[1][i] ^ state[2][i] ; Tm = xtime(Tm); state[1][i] ^= Tm ^ Tmp ;
        Tm = state[2][i] ^ state[3][i] ; Tm = xtime(Tm); state[2][i] ^= Tm ^ Tmp ;
        Tm = state[3][i] ^ t ; Tm = xtime(Tm); state[3][i] ^= Tm ^ Tmp ;
    }
}


; ***************************************************************************
; Cipher
; Realiza el cifrado del Plaintext almacenado en state. Se debe llamar antes a
; KeyExpansion.
; In: in-> texto a cifrar
; Out: out -> texto cifrado
; ***************************************************************************
void Cipher(void)
{
    unsigned int i,j,round=0;

    // Ronda Inicial
    AddRoundKey(0);

    // Rondas
    for(round=1;round<Nr;round++)
    {
        SubBytes();
        ShiftRows();
        MixColumns();
        AddRoundKey(round);
    }

    // Ronda Final.
    SubBytes();
    ShiftRows();
    AddRoundKey(Nr);

    // Copia el state al vector de salida
    for(i=0;i<4;i++)
    {
        for(j=0;j<4;j++)
        {
            out[i*4+j]=state[j][i];
        }
    }
}


; ***************************************************************************
; DatosState
; Copia el Plaintext a cifrar en el state
; In: TextoIn-> Número de texto a cifrar de Ptexts
; Out: state -> texto a cifrar
; ***************************************************************************

void DatosState(unsigned int TextoIn)
{
        unsigned int i=0,j=0;

        Nk=LengthKey/32;
        Nr = Nk + 6;

        for(i=0;i<4;i++)
        {
           for(j=0;j<4;j++)
           {
              state[j][i] = Ptexts[TextoIn][j+4*i];
           }
        }
}
```

La modificación realizada para almacenar únicamente el byte bajo ataque de los textos a cifrar y optimizar la memoria usada, es la siguiente:

```
unsigned char const Ptexts [PtextsBucle] =
{
0xE4,0x57,0x89,0x74,0x2D,0x8A,0x01,0xFA,0x15,0x4E,0xA0,0x4A,0xA8,0xBD,0x61,0x62,
0xA3,0xBF,0x16,0xCA,0x7A,0x69,0xC8,0x45,0x90,0xEB,0xA8,0x4B,0x33,0x64,0xB4,0xA2,
...


void DatosState(unsigned int TextoIn)
{
         unsigned int i=0,j=0;

         Nk=LengthKey/32;
         Nr = Nk + 6;

         for(i=0;i<4;i++){
            for(j=0;j<4;j++){
               state[j][i] = rand() % 256; // Se generan bytes aleatorios
            }
         }
         state[1][1]=Ptexts[TextoIn]; // el byte 6 toma el valor "TextoIn" de Ptexts
}
```

CÓDIGO EMA 1 C8051F303 (ENSAMBLADOR)

CIFRADO

```
; Clave
CIPHER_KEY:      DB  0C8H, 00AH, 04AH, 0BFH, 078H, 0D3H, 0B2H, 079H, 098H, 0A0H, \
                     0B6H, 036H, 0B4H, 05EH, 09BH, 0D1H

; Plaintexts
InData: DB 0A3H,0D4H,089H,043H,070H,077H,095H,037H,0C8H,043H,040H,044H,09CH,0B6H,066H,0F4H,\
0C7H,0DDH,01AH,089H,08DH,0C5H,00AH,0D2H,031H,057H,08EH,0ABH,02FH,09DH,09BH,0F6H,\
093H,050H,0C0H,0DAH,0D0H,0EDH,0EAH,04EH,054H,0ADH,00DH,0E4H,0F8H,096H,03AH,015H,\
...

; Tabla Sbox
Sbox:
DB       063H, 07CH, 077H, 07BH, 0F2H, 06BH, 06FH, 0C5H, 030H, 001H, 067H, \         02BH, 0FEH, 0D7H,
0ABH, 076H, 0CAH, 082H, 0C9H, 07DH, 0FAH, 059H, \
         047H, 0F0H, 0ADH, 0D4H, 0A2H, 0AFH, 09CH, 0A4H, 072H, 0C0H, 0B7H, \
         0FDH, 093H, 026H, 036H, 03FH, 0F7H, 0CCH, 034H, 0A5H, 0E5H, 0F1H, \
         071H, 0D8H, 031H, 015H, 004H, 0C7H, 023H, 0C3H, 018H, 096H, 005H, \
         09AH, 007H, 012H, 080H, 0E2H, 0EBH, 027H, 0B2H, 075H, 009H, 083H, \
         02CH, 01AH, 01BH, 06EH, 05AH, 0A0H, 052H, 03BH, 0D6H, 0B3H, 029H, \
         0E3H, 02FH, 084H, 053H, 0D1H, 000H, 0EDH, 020H, 0FCH, 0B1H, 05BH
DB       06AH, 0CBH, 0BEH, 039H, 04AH, 04CH, 058H, 0CFH, 0D0H, 0EFH, 0AAH, \
         0FBH, 043H, 04DH, 033H, 085H, 045H, 0F9H, 002H, 07FH, 050H, 03CH, \
         09FH, 0A8H, 051H, 0A3H, 040H, 08FH, 092H, 09DH, 038H, 0F5H, 0BCH, \
         0B6H, 0DAH, 021H, 010H, 0FFH, 0F3H, 0D2H, 0CDH, 00CH, 013H, 0ECH, \
         05FH, 097H, 044H, 017H, 0C4H, 0A7H, 07EH, 03DH, 064H, 05DH, 019H, \
         073H, 060H, 081H, 04FH, 0DCH, 022H, 02AH, 090H, 088H, 046H, 0EEH, \
         0B8H, 014H, 0DEH, 05EH, 00BH, 0DBH, 0E0H, 032H, 03AH, 00AH, 049H, \
         006H, 024H, 05CH, 0C2H, 0D3H, 0ACH, 062H, 091H, 095H, 0E4H, 079H
DB       0E7H, 0C8H, 037H, 06DH, 08DH, 0D5H, 04EH, 0A9H, 06CH, 056H, 0F4H, \
         0EAH, 065H, 07AH, 0AEH, 008H, 0BAH, 078H, 025H, 02EH, 01CH, 0A6H, \
         0B4H, 0C6H, 0E8H, 0DDH, 074H, 01FH, 04BH, 0BDH, 08BH, 08AH, 070H, \
         03EH, 0B5H, 066H, 048H, 003H, 0F6H, 00EH, 061H, 035H, 057H, 0B9H, \
         086H, 0C1H, 01DH, 09EH, 0E1H, 0F8H, 098H, 011H, 069H, 0D9H, 08EH, \
         094H, 09BH, 01EH, 087H, 0E9H, 0CEH, 055H, 028H, 0DFH, 08CH, 0A1H, \
         089H, 00DH, 0BFH, 0E6H, 042H, 068H, 041H, 099H, 02DH, 00FH, 0B0H, \
         054H, 0BBH, 016H

RCON:    DB       001H, 002H, 004H, 008H, 010H, 020H, 040H, 080H, 01BH, 036H


; Ronda actual de la clave
ROUNDKEY:        DS   16

; Ronda actual cifrado
ROUND:           DS    1

; Estado
State:           DS   16


;*******************************************************************************
; Cifrado
; Rutina que encripta un mensaje usando AES
;*******************************************************************************
Cifrado:
         LCALL    StateIn
         LCALL    KeyRAM
         MOV      ROUND,#00H
ROUND1:
         LCALL    AddRoundKey
ROUND2:
         LCALL    SubBytes
         LCALL    ShiftRows
         LCALL    MixColumns
         LCALL    KeyExpansion
         LCALL    AddRoundKey
         INC      ROUND
ROUNDS8:
         LCALL    SubBytes
         LCALL    ShiftRows
         LCALL    MixColumns
         LCALL    KeyExpansion
         LCALL    AddRoundKey

         INC      ROUND
         MOV      A,ROUND
         XRL      A,#09H
         JNZ      ROUNDS8

         LCALL    SubBytes
         LCALL    ShiftRows
         LCALL    KeyExpansion
         LCALL    AddRoundKey
```

```
        LCALL   StateOut
RET
; Fin de Cifrado


;*******************************************************************************
; StateIn
; Copia el texto plano en la matriz State
; In: F0=1 (solo primer uso)
;     DPTR-> dirección primer plaintext a cifrar (solo primer uso)
; Out: R7
;*******************************************************************************
StateIn:
        JBC     F0,IJump ; Salta en el primer cifrado F0: User Flag 0
        MOV     A,#10H
        CLR     C
        ADD     A,DirL
        MOV     DirL,A
        JNC     FJump    ; Si no hay acarreo (CARRY) salto y no aumento DirH
        INC     DirH
      FJump:
        MOV     DPH,DirH ; Movemos al DPTR la dirección del Plaintext (Parte Alta)
        MOV     DPL,DirL ; Movemos al DPTR la dirección del Plaintext (Parte Baja)
      IJump:
        MOV     R1,#State

        LCALL  ROM2RAM   ; Rutina implementada en Expansion de Clave
RET
; Fin de StateIn

;*******************************************************************************
; StateOut
; Copia el texto encriptado de la matriz State al vector de salida
; In: State-> texto encriptado
; Out: EncryptedData
;*******************************************************************************
StateOut:
        MOV     i,#00H
        MOV     R1,#State
        MOV     R0,#EncryptedData
     MASDATOS:
        MOV     A,@R1
        MOV     @R0,A
        INC     R1
        INC     R0
        INC     i
        MOV     A,i
        XRL     A,#10H
        JNZ     MASDATOS
RET
; Fin de StateOut

;*******************************************************************************
; SubBytes
; Ejecuta la fase Subbytes sobre el State
; In: State-> texto
; Out: State-> texto modificado con Subbytes
;*******************************************************************************
SubBytes:
        MOV     A,State
        MOV     DPTR,#Sbox
        MOVC    A,@A+DPTR
        MOV     State,A
        MOV     A,State+01H
        MOVC    A,@A+DPTR
        MOV     State+01H,A
        MOV     A,State+02H
        MOVC    A,@A+DPTR
        MOV     State+02H,A
        MOV     A,State+03H
        MOVC    A,@A+DPTR
        MOV     State+03H,A

        MOV     A,State+04H
        MOVC    A,@A+DPTR
        MOV     State+04H,A
        MOV     A,State+05H
        MOVC    A,@A+DPTR
        MOV     State+05H,A
        MOV     A,State+06H
        MOVC    A,@A+DPTR
        MOV     State+06H,A
        MOV     A,State+07H
        MOVC    A,@A+DPTR
        MOV     State+07H,A

        MOV     A,State+08H
        MOVC    A,@A+DPTR
        MOV     State+08H,A
        MOV     A,State+09H
```

```
        MOVC    A,@A+DPTR
        MOV     State+09H,A
        MOV     A,State+0AH
        MOVC    A,@A+DPTR
        MOV     State+0AH,A
        MOV     A,State+0BH
        MOVC    A,@A+DPTR
        MOV     State+0BH,A

        MOV     A,State+0CH
        MOVC    A,@A+DPTR
        MOV     State+0CH,A
        MOV     A,State+0DH
        MOVC    A,@A+DPTR
        MOV     State+0DH,A
        MOV     A,State+0EH
        MOVC    A,@A+DPTR
        MOV     State+0EH,A
        MOV     A,State+0FH
        MOVC    A,@A+DPTR
        MOV     State+0FH,A
        RET
; Fin de SubBytes

;*******************************************************************************
; ShiftRows
; Ejecuta la fase ShiftRows.
; In: State-> texto
; Out: State-> texto modificado con ShiftRows
;*******************************************************************************
ShiftRows:
        MOV     R7,State+01H
        MOV     State+01H,State+05H
        MOV     State+05H,State+09H
        MOV     State+09H,State+0DH
        MOV     State+0DH,R7

        MOV     R7,State+02H
        MOV     State+02H,State+0AH
        MOV     State+0AH,R7
        MOV     R7,State+06H
        MOV     State+06H,State+0EH
        MOV     State+0EH,R7

        MOV     R7,State+0FH
        MOV     State+0FH,State+0BH
        MOV     State+0BH,State+07H
        MOV     State+07H,State+03H
        MOV     State+03H,R7
        RET
; Fin de ShiftRows

;*******************************************************************************
; MixColumns
; Ejecuta la fase MixColumns usando la rutina xtime.
; In: State-> texto
; Out: State-> texto modificado con MixColumns
;*******************************************************************************
MixColumns:
        MOV     A,State
        XRL     A,State+01H
        MOV     aux1,A
        MOV     A,State+02H
        XRL     A,State+03H
        MOV     aux3,A
        XRL     A,aux1
        MOV     aux0,A
        MOV     A,State+02H
        XRL     A,State+01H
        MOV     aux2,A
        MOV     R3,aux1
        LCALL   xtime
        MOV     aux1,R7
        MOV     R3,aux2
        LCALL   xtime
        MOV     aux2,R7
        MOV     R3,aux3
        LCALL   xtime
        MOV     aux3,R7
        MOV     A,aux0
        XRL     A,aux1
        XRL     State,A
        MOV     A,aux0
        XRL     A,aux2
        XRL     State+01H,A
        MOV     A,aux0
        XRL     A,aux3
        XRL     State+02H,A
        MOV     A,State
        XRL     A,State+01H
```

```
XRL     A,State+02H
XRL     A,aux0
MOV     State+03H,A

MOV     A,State+04H
XRL     A,State+05H
MOV     aux1,A
MOV     A,State+06H
XRL     A,State+07H
MOV     aux3,A
XRL     A,aux1
MOV     aux0,A
MOV     A,State+06H
XRL     A,State+05H
MOV     aux2,A
MOV     R3,aux1
LCALL   xtime
MOV     aux1,R7

MOV     R3,aux2
LCALL   xtime
MOV     aux2,R7
MOV     R3,aux3
LCALL   xtime
MOV     aux3,R7
MOV     A,aux0
XRL     A,aux1
XRL     State+04H,A
MOV     A,aux0
XRL     A,aux2
XRL     State+05H,A
MOV     A,aux0
XRL     A,aux3
XRL     State+06H,A
MOV     A,State+04H
XRL     A,State+05H
XRL     A,State+06H
XRL     A,aux0
MOV     State+07H,A

MOV     A,State+08H
XRL     A,State+09H
MOV     aux1,A
MOV     A,State+0AH
XRL     A,State+0BH
MOV     aux3,A
XRL     A,aux1
MOV     aux0,A
MOV     A,State+0AH
XRL     A,State+09H
MOV     aux2,A
MOV     R3,aux1
LCALL   xtime
MOV     aux1,R7
MOV     R3,aux2
LCALL   xtime
MOV     aux2,R7
MOV     R3,aux3
LCALL   xtime
MOV     aux3,R7

MOV     A,aux0
XRL     A,aux1
XRL     State+08H,A
MOV     A,aux0
XRL     A,aux2
XRL     State+09H,A
MOV     A,aux0
XRL     A,aux3
XRL     State+0AH,A
MOV     A,State+08H
XRL     A,State+09H
XRL     A,State+0AH
XRL     A,aux0
MOV     State+0BH,A

MOV     A,State+0CH
XRL     A,State+0DH
MOV     aux1,A
MOV     A,State+0EH
XRL     A,State+0FH
MOV     aux3,A
XRL     A,aux1
MOV     aux0,A
MOV     A,State+0EH
XRL     A,State+0DH
MOV     aux2,A
MOV     R3,aux1
LCALL   xtime
MOV     aux1,R7
```

```
        MOV     R3,aux2
        LCALL   xtime
        MOV     aux2,R7
        MOV     R3,aux3
        LCALL   xtime
        MOV     aux3,R7
        MOV     A,aux0
        XRL     A,aux1
        XRL     State+0CH,A
        MOV     A,aux0
        XRL     A,aux2
        XRL     State+0DH,A
        MOV     A,aux0
        XRL     A,aux3
        XRL     State+0EH,A
        MOV     A,State+0CH
        XRL     A,State+0DH
        XRL     A,State+0EH
        XRL     A,aux0
        MOV     State+0FH,A
RET
; END OF MixColumns

;*******************************************************************************
; AddRoundKey
; Ejecuta la fase AddRoundKey
; In: State-> texto
; Out: State-> texto modificado con AddRoundKey
;*******************************************************************************
AddRoundKey:
        MOV     A,ROUNDKEY
        XRL     State,A
        MOV     A,ROUNDKEY+01H
        XRL     State+01H,A
        MOV     A,ROUNDKEY+02H
        XRL     State+02H,A
        MOV     A,ROUNDKEY+03H
        XRL     State+03H,A

        MOV     A,ROUNDKEY+04H
        XRL     State+04H,A
        MOV     A,ROUNDKEY+05H
        XRL     State+05H,A
        MOV     A,ROUNDKEY+06H
        XRL     State+06H,A
        MOV     A,ROUNDKEY+07H
        XRL     State+07H,A

        MOV     A,ROUNDKEY+08H
        XRL     State+08H,A
        MOV     A,ROUNDKEY+09H
        XRL     State+09H,A
        MOV     A,ROUNDKEY+0AH
        XRL     State+0AH,A
        MOV     A,ROUNDKEY+0BH
        XRL     State+0BH,A

        MOV     A,ROUNDKEY+0CH
        XRL     State+0CH,A
        MOV     A,ROUNDKEY+0DH
        XRL     State+0DH,A
        MOV     A,ROUNDKEY+0EH
        XRL     State+0EH,A
        MOV     A,ROUNDKEY+0FH
        XRL     State+0FH,A
RET
; Fin de AddRoundKey

;*******************************************************************************
; xtime
; Realiza una multiplicación de un polinomio (entrada) por x y hace el
; modulo m(x); m(x) is (x^8 + x^4 + x^3 + x + 1).
; Esto se implementa como un desplazamiento a la izquierda y una XOR con 0x1B
; si el bit desplazado es 1.
; In: R3-> polinomio
; Out: R7
;*******************************************************************************
xtime:
        MOV     A,R3
        ANL     A,#80H
        JZ      NOX8
        MOV     A,R3
        ADD     A,R3            ;Rotación a la izquierda tb: CLR C, RLC A
        XRL     A,#1BH
        MOV     R7,A
RET
NOX8:
        MOV     A,R3
        ADD     A,R3            ;Rotación a la izquierda tb: CLR C, RLC A
        MOV     R7,A
```

```
RET
; Fin de xtime
```

EXPANSIÓN CLAVE

```
;*******************************************************************************
; KeyRAM
; Pasa la clave de Rom (CIPHER_KEY) a Ram (ROUNDKEY)
; In:  Nada
; Out: Clave en variable ROUNDKEY
;*******************************************************************************
KeyRAM:
        MOV     DPTR,#CIPHER_KEY
        MOV     R1,#ROUNDKEY
        LCALL   ROM2RAM
RET
; Fin de KeyRAM


;*******************************************************************************
; ROM2RAM
; Pasa 16 bytes de Rom a Ram
;       In:     DPTR -> Dirección de los 16 bytes de Rom: Datos origen
;       R1       -> Dirección de los 16 bytes de Ram: Datos destino
; Out: Datos ROM apuntados por DPTR guardados en variable RAM a la que apunta R1
; *******************************************************************************
ROM2RAM:
        MOV     i,#00H
      SaltoFor:
        MOV     A,i
        MOVC    A,@A+DPTR
        MOV     @R1,A
        INC     R1

        INC     i
        MOV     A,i
        XRL     A,#10H
        JNZ     SaltoFor
RET
; Fin de ROM2RAM


; *******************************************************************************
; KEYEXPANSION
; Expande la clave una ronda
; In:   Key o RoundKeyX en vector ROUNDKEY
;       Round = X-1 -> En Round se introduce el número de ronda empezando por 0 a 9
; Out:  RoundKeyX+1 en ROUNDKEY
; *******************************************************************************
KeyExpansion:
        MOV     TEMPWORD,ROUNDKEY+12
        MOV     TEMPWORD+1,ROUNDKEY+13
        MOV     TEMPWORD+2,ROUNDKEY+14
        MOV     TEMPWORD+3,ROUNDKEY+15

        MOV     R7,TEMPWORD
        MOV     TEMPWORD,TEMPWORD+01H
        MOV     TEMPWORD+01H,TEMPWORD+02H
        MOV     TEMPWORD+02H,TEMPWORD+03H
        MOV     TEMPWORD+03H,R7

        MOV     A,TEMPWORD
        MOV     DPTR,#Sbox
        MOVC    A,@A+DPTR
        MOV     TEMPWORD,A

        MOV     A,TEMPWORD+01H
        MOVC    A,@A+DPTR
        MOV     TEMPWORD+01H,A

        MOV     A,TEMPWORD+02H
        MOVC    A,@A+DPTR
        MOV     TEMPWORD+02H,A

        MOV     A,TEMPWORD+03H
        MOVC    A,@A+DPTR
        MOV     TEMPWORD+03H,A

        MOV     DPTR,#RCON
        MOV     A,ROUND          ; ROUND EMPIEZA EN 0 Y TERMINA EN 9
        MOVC    A,@A+DPTR
        XRL     TEMPWORD,A

        MOV     A,TEMPWORD
        XRL     ROUNDKEY,A
        MOV     A,TEMPWORD+1
        XRL     ROUNDKEY+1,A
        MOV     A,TEMPWORD+2
```

```
        XRL     ROUNDKEY+2,A
        MOV     A,TEMPWORD+3
        XRL     ROUNDKEY+3,A

        MOV     A,ROUNDKEY
        XRL     ROUNDKEY+4,A
        MOV     A,ROUNDKEY+1
        XRL     ROUNDKEY+5,A
        MOV     A,ROUNDKEY+2
        XRL     ROUNDKEY+6,A
        MOV     A,ROUNDKEY+3
        XRL     ROUNDKEY+7,A

        MOV     A,ROUNDKEY+4
        XRL     ROUNDKEY+8,A
        MOV     A,ROUNDKEY+5
        XRL     ROUNDKEY+9,A
        MOV     A,ROUNDKEY+6
        XRL     ROUNDKEY+10,A
        MOV     A,ROUNDKEY+7
        XRL     ROUNDKEY+11,A

        MOV     A,ROUNDKEY+8
        XRL     ROUNDKEY+12,A
        MOV     A,ROUNDKEY+9
        XRL     ROUNDKEY+13,A
        MOV     A,ROUNDKEY+10
        XRL     ROUNDKEY+14,A
        MOV     A,ROUNDKEY+11
        XRL     ROUNDKEY+15,A
RET

; Fin de KeyExpansion
```

Anexo 3

RESULTADOS EXPERIMENTALES COMPLETOS

A continuación se muestran los resultados de los ataques CEMA realizados sobre las placas bajo estudio cuya seguridad se ha tratado de estudiar en este trabajo utilizando el test setup desarrollado para ello y tres sondas EM de campo cercano. En este caso solo se han incluido los ataques que han sido satisfactorios y han proporcionado como resultado el byte 3 de la clave correctamente 0xD3. El resto al no reportar conclusión alguna, no se han incluido.

Para cada ataque, se muestran dos tipos de gráficas. La primera refleja el coeficiente de correlación obtenido suponiendo la clave correcta para un intervalo de registros alrededor del máximo. El número de registro indicado en las gráficas, permite conocer el momento en el que se produce la correlación respecto del inicio de la captura que coincide con el comienzo de la fase AddRoundkey[47] pues la captura se realiza de forma constante a 5 Gmuestras/segundo, o lo que es lo mismo 0.2 nsegundos/registro.

La segunda gráfica presenta la evolución en valor absoluto del coeficiente de correlación de cada una de las claves posibles, para el registro de la traza que genera un mayor coeficiente de correlación, en función del número de trazas EM utilizadas para su cálculo. En

[47] En el caso de la placa EMA4, se captó sólo la fase Subbytes. Por ello, el número de registro se ha corregido para incluir la fase AddRoundKey previa, de forma que las cuatro placas estén en las mismas condiciones.

verde se resalta la envolvente de los coeficientes de correlación de las claves incorrectas, en rojo el coeficiente de correlación de la clave correcta y en azul la diferencia.

EMA 1: C8051F303

EM6995

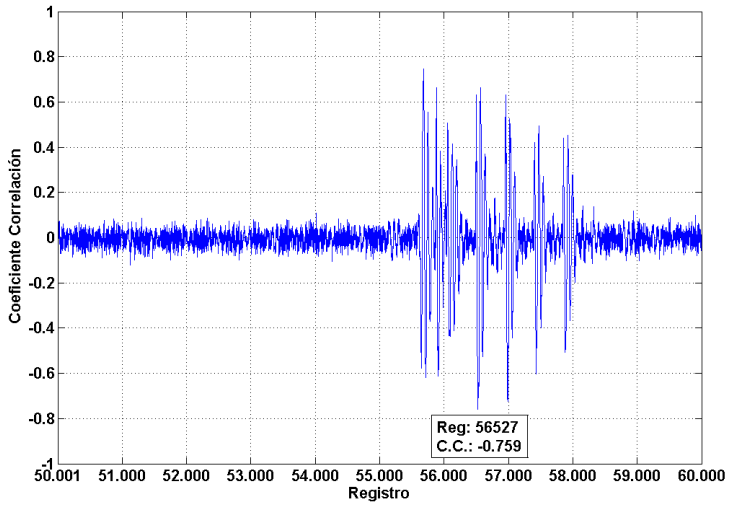


Figura A3.1: EMA 1 EM1 C.C.-Registros

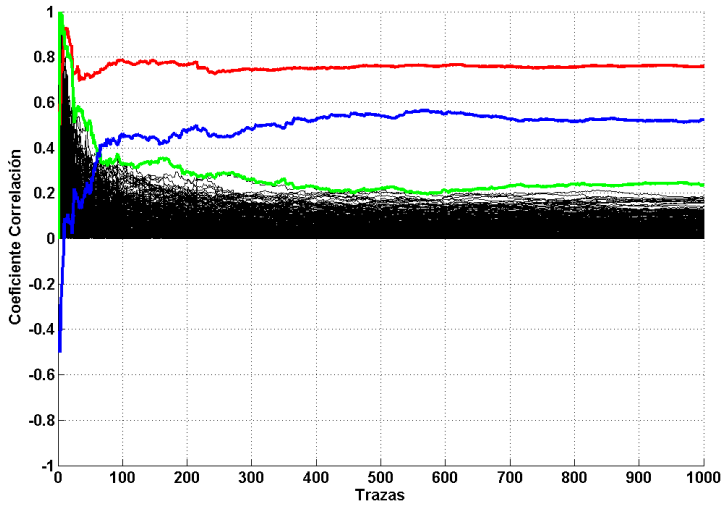


Figura A3.2: EMA 1 EM1 C.C.-Trazas Registro 56527

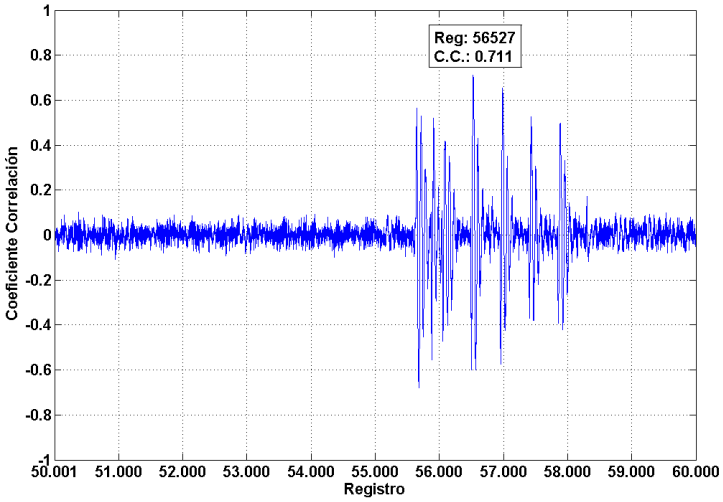


Figura A3.3: EMA 1 EM2 C.C.-Registros

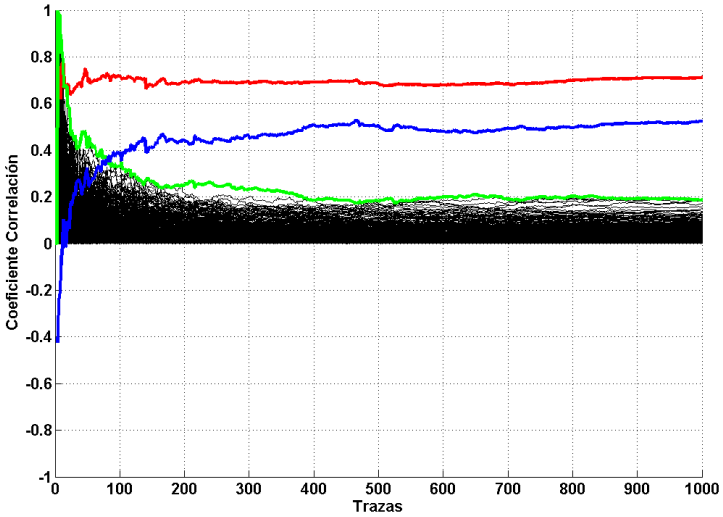


Figura A3.4: EMA 1 EM2 C.C.-Trazas Registro 56527

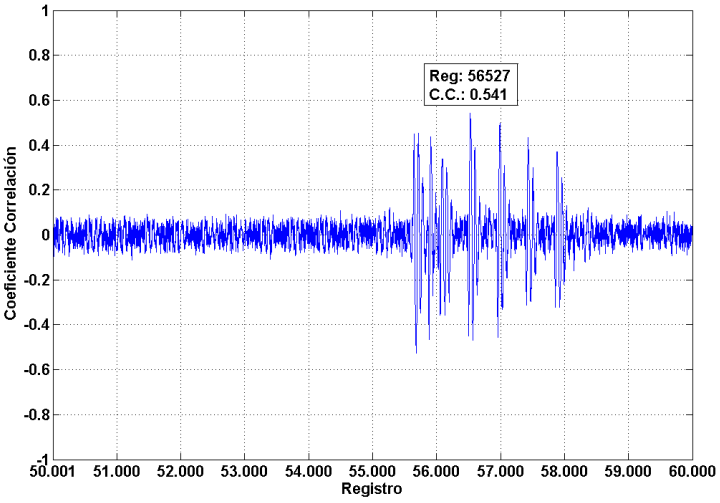


Figura A3.5: EMA 1 EM3 C.C.-Registros

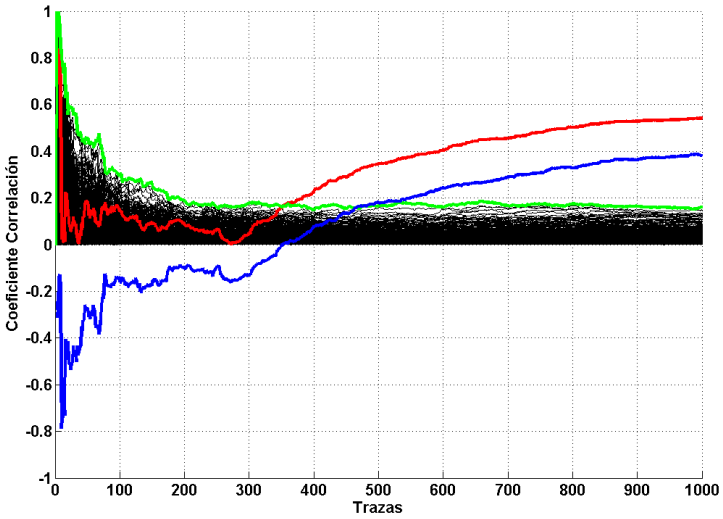


Figura A3.6: EMA 1 EM3 C.C.-Trazas Registro 56527

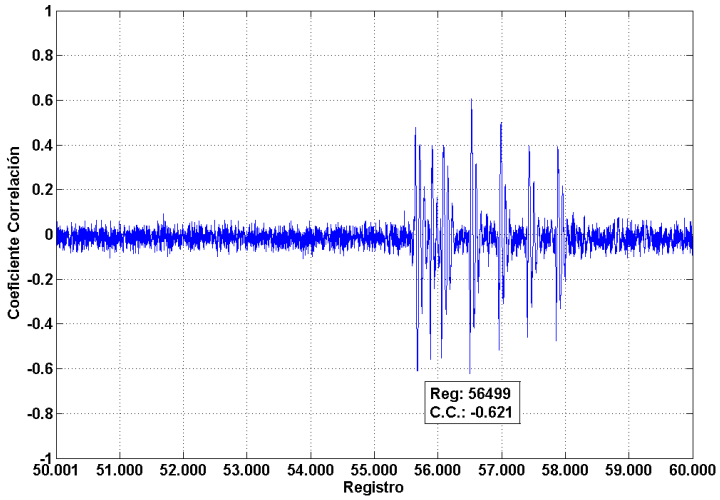


Figura A3.7: EMA 1 EM4 C.C.-Registros

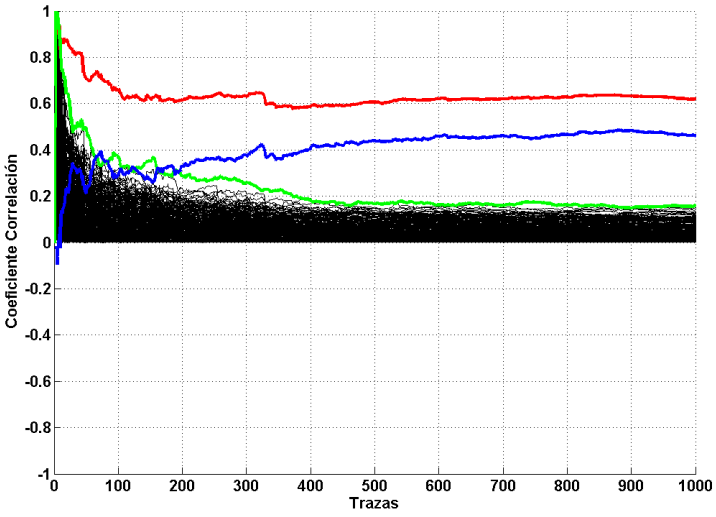


Figura A3.8: EMA 1 EM4 C.C.-Trazas Registro 56499

MFA-R

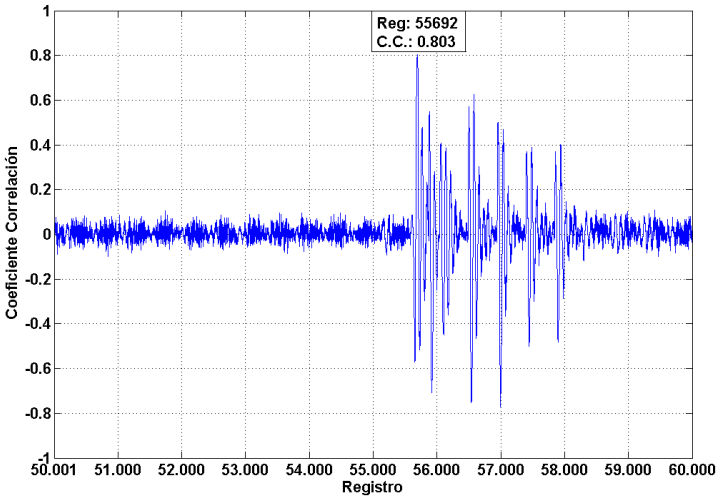


Figura A3.9: EMA 1 MFA1 C.C.-Registros

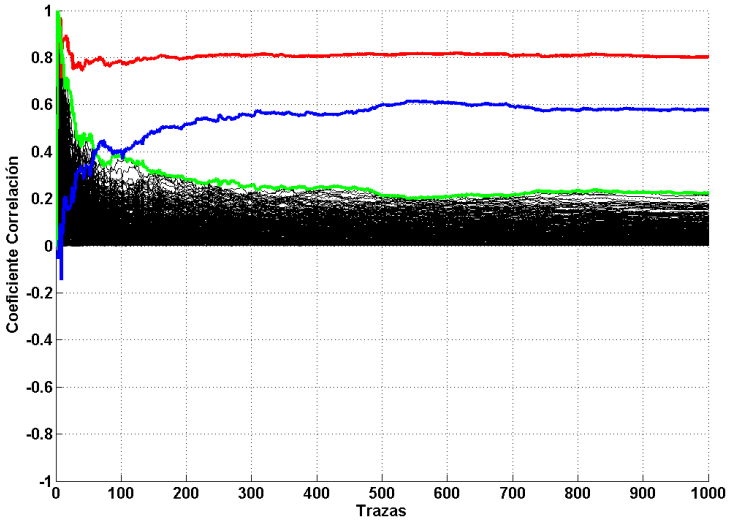


Figura A3.10: EMA 1 MFA1 C.C.-Trazas Registro 55692

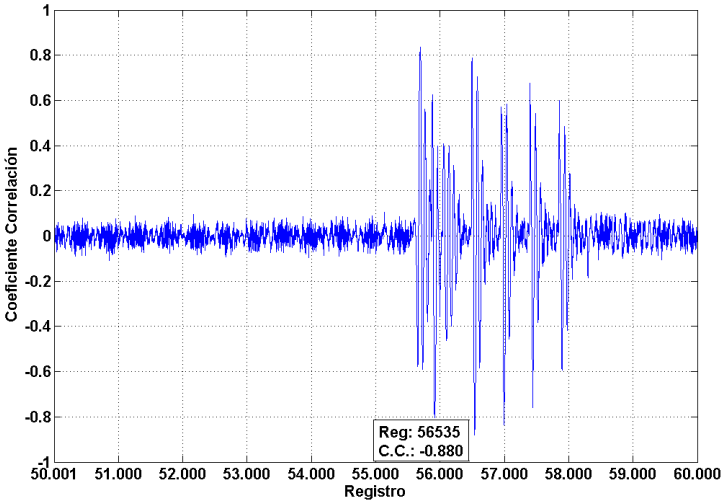


Figura A3.11: EMA 1 MFA2 C.C.-Registros

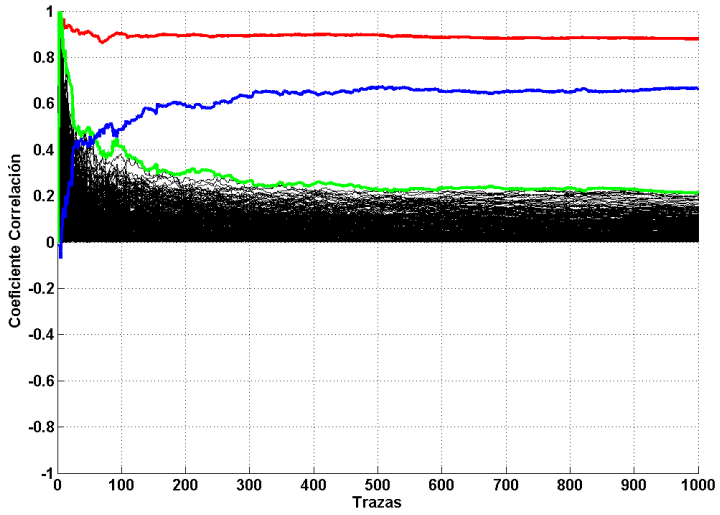


Figura A3.12: EMA 1 MFA2 C.C.-Trazas Registro 56535

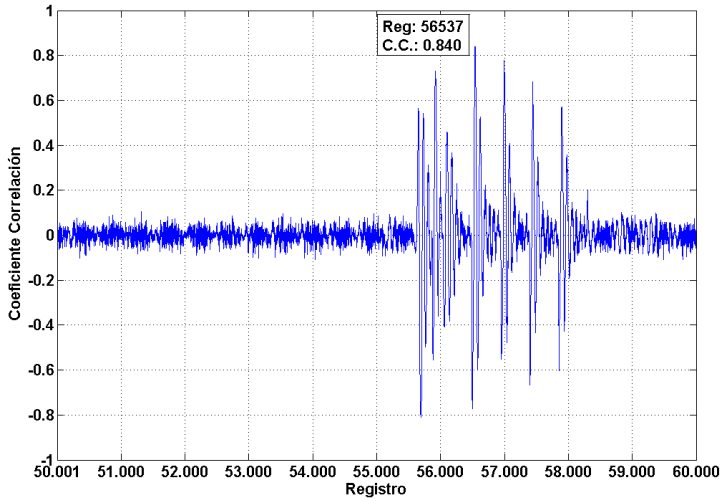


Figura A3.13: EMA 1 MFA3 C.C.-Registros

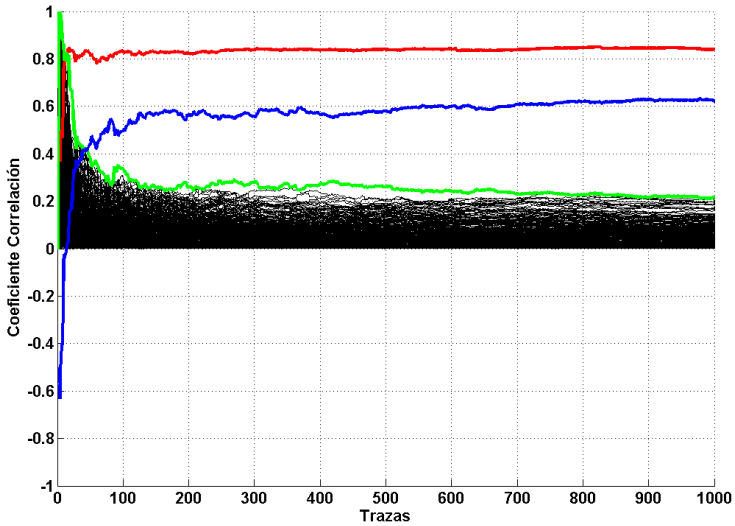


Figura A3.14: EMA 1 MFA3 C.C.-Trazas Registro 56537

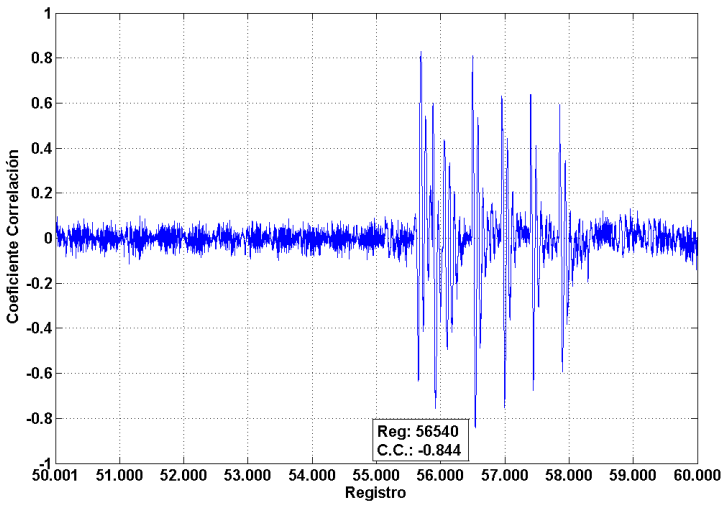


Figura A3.15: EMA 1 MFA4 C.C.-Registros

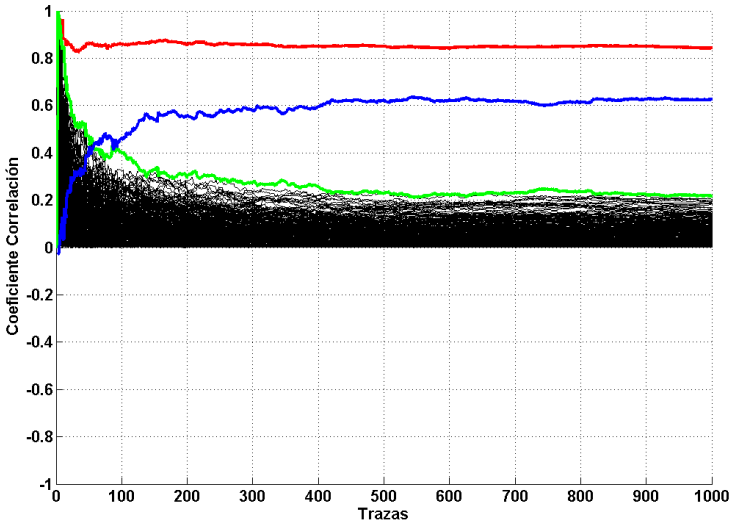


Figura A3.16: EMA 1 MFA4 C.C.-Trazas Registro 56540

HOMEMADE

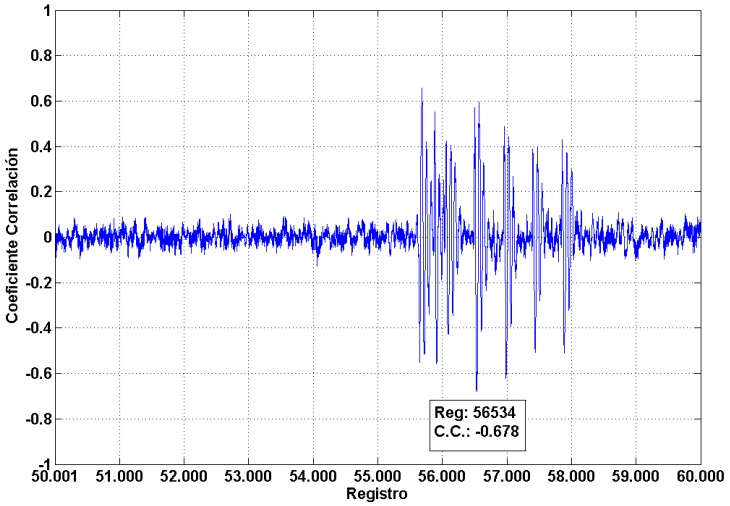


Figura A3.17: EMA 1 HO1 C.C.-Registros

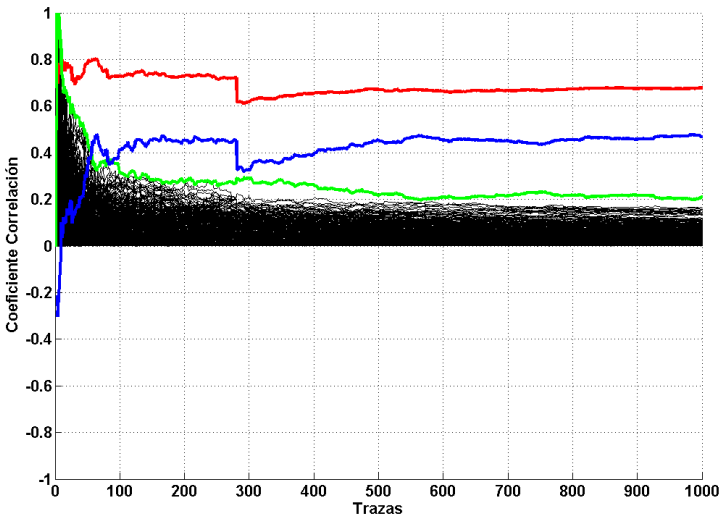


Figura A3.18: EMA 1 HO1 C.C.-Trazas Registro 56534

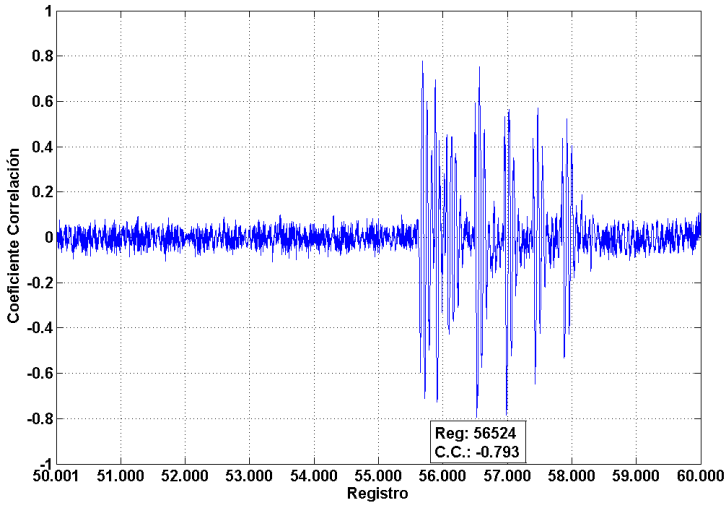


Figura A3.19: EMA 1 HO2 C.C.-Registros

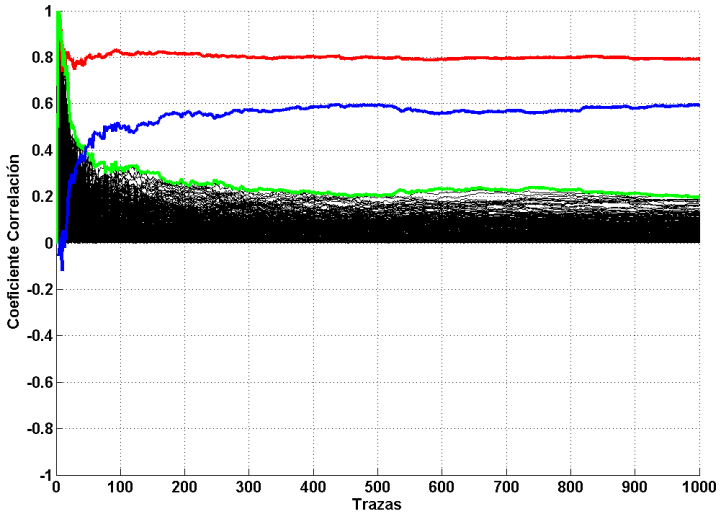


Figura A3.20: EMA 1 HO2 C.C.-Trazas Registro 56524

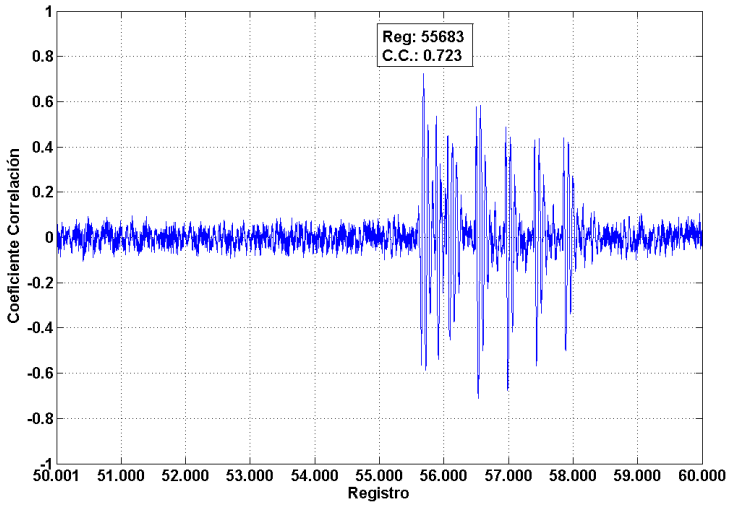


Figura A3.21: EMA 1 HO3 C.C.-Registros

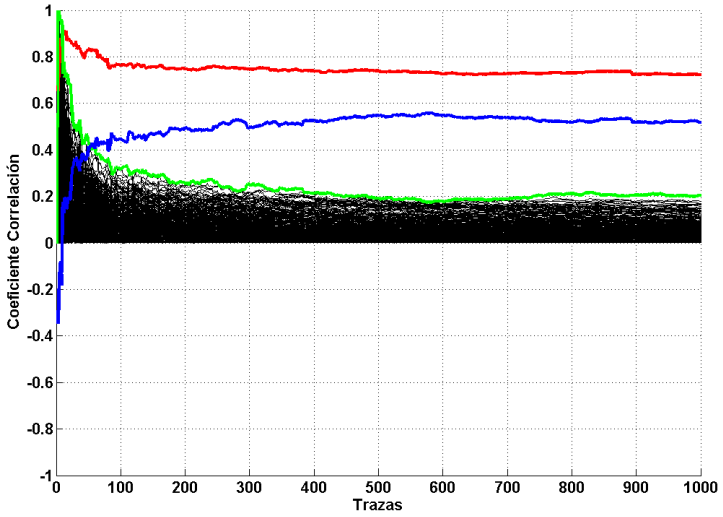


Figura A3.22: EMA 1 HO3 C.C.-Trazas Registro 55683

EMA 2: ARM7

EM6995

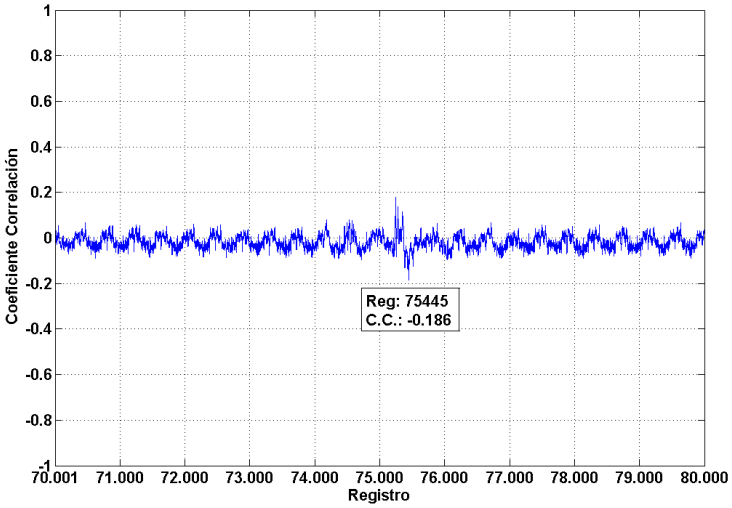


Figura A3.23: EMA 2 EM1 C.C.-Registros

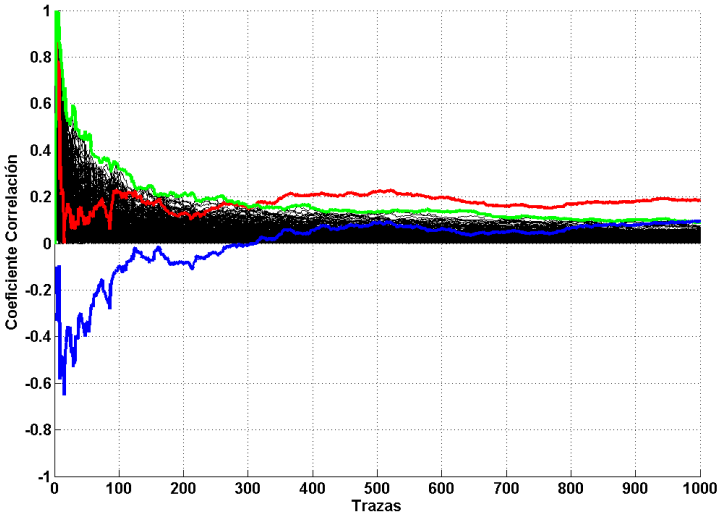


Figura A3.24: EMA 2 EM1 C.C.-Trazas Registro 75445

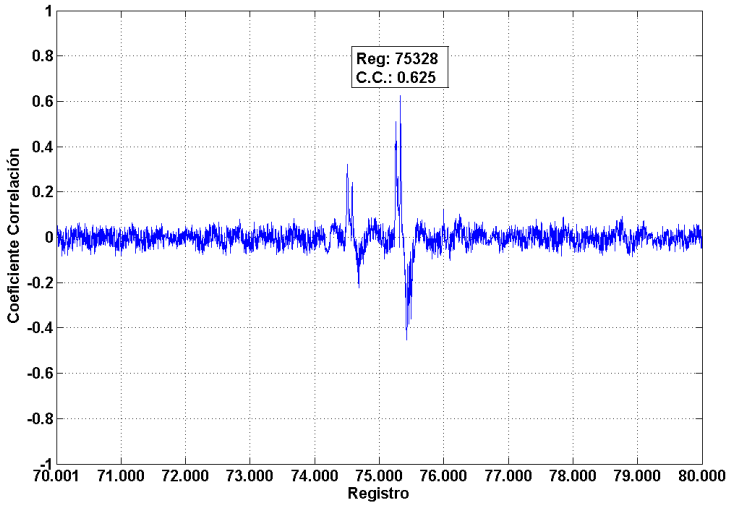


Figura A3.25: EMA 2 EM2 C.C.-Registros

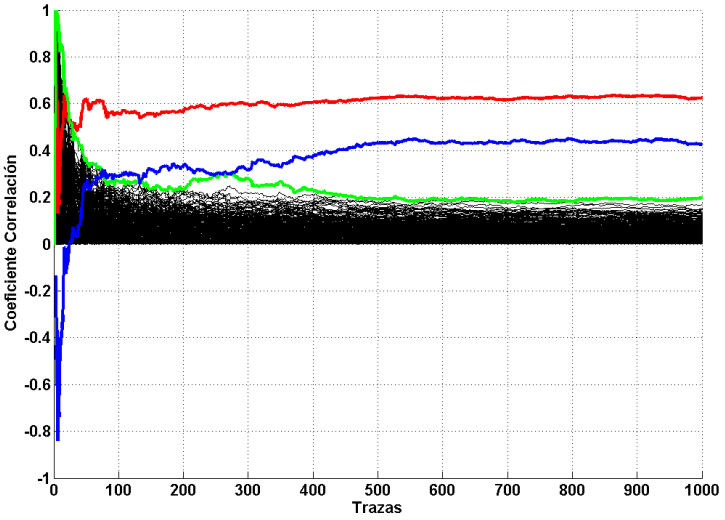


Figura A3.26: EMA 2 EM2 C.C.-Trazas Registro 75328

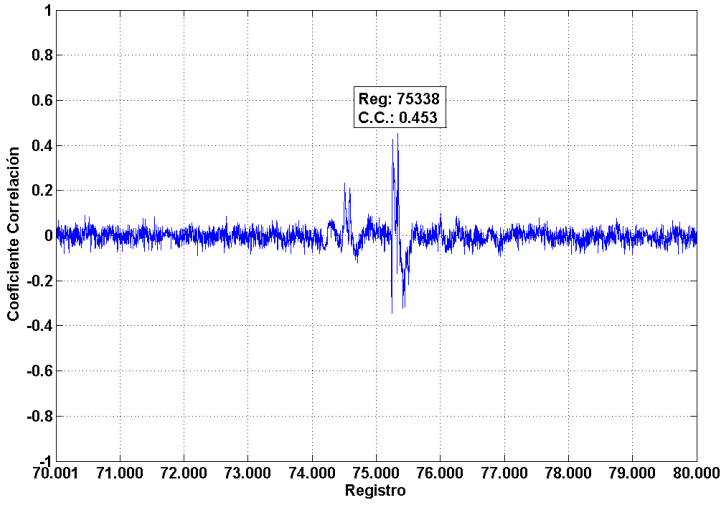


Figura A3.27: EMA 2 EM3 C.C.-Registros

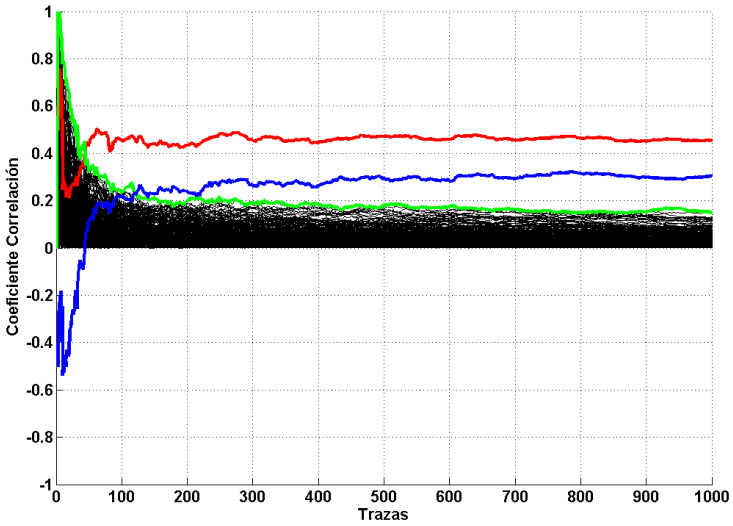


Figura A3.28: EMA 2 EM3 C.C.-Trazas Registro 75338

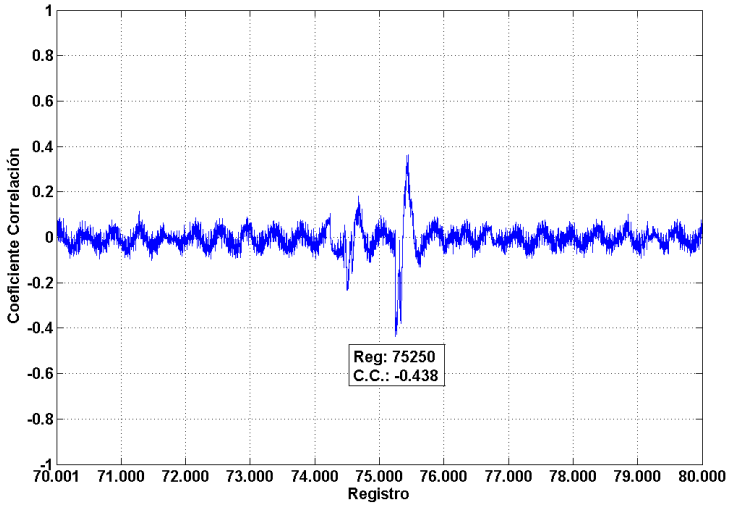


Figura A3.29: EMA 2 EM4 C.C.-Registros

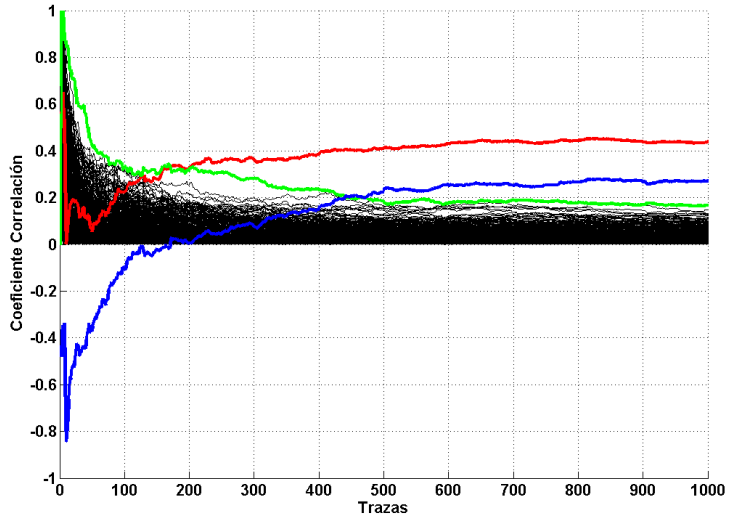


Figura A3.30: EMA 2 EM4 C.C.-Trazas Registro 75250

MFA-R

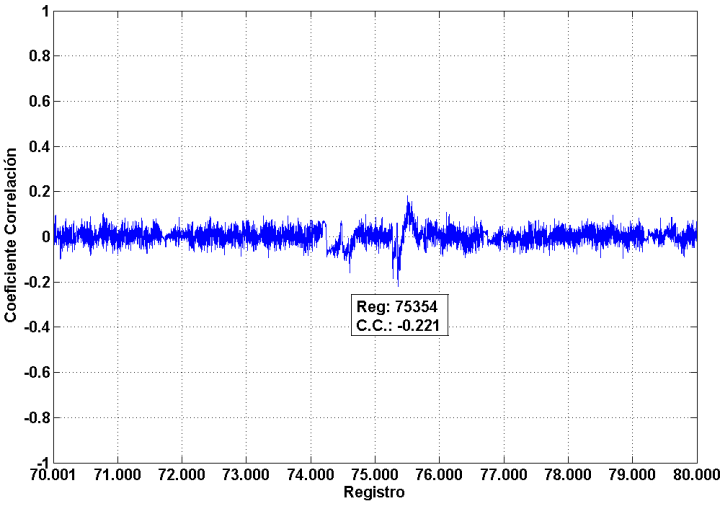


Figura A3.31: EMA 2 MFA1 C.C.-Registros

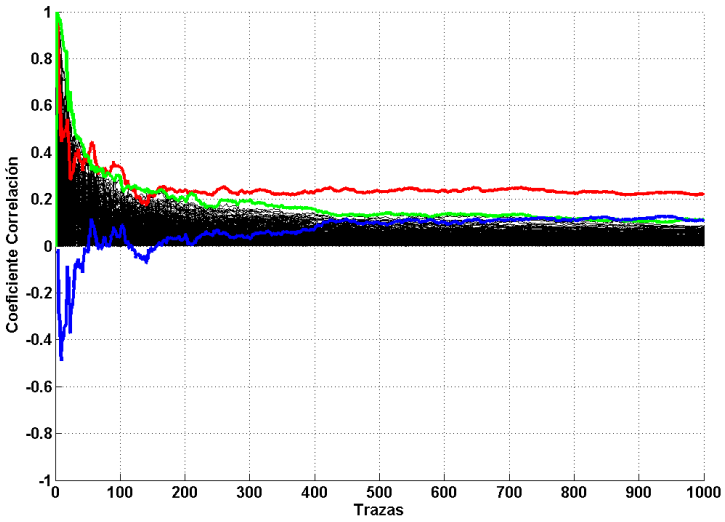


Figura A3.32: EMA 2 MFA1 C.C.-Trazas Registro 75354

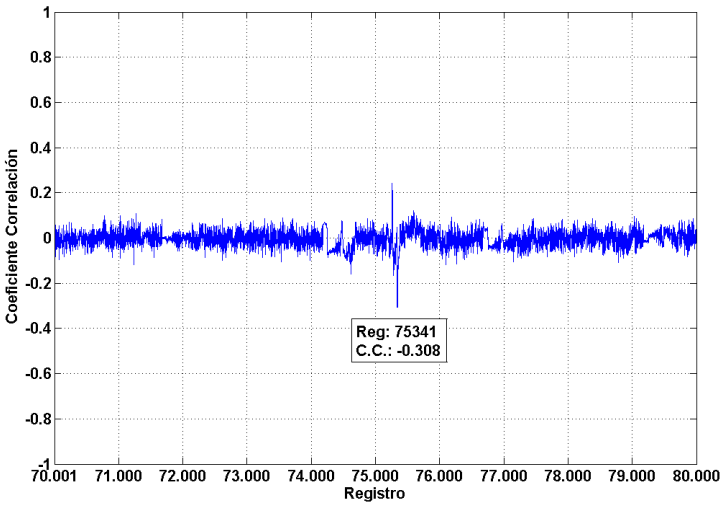


Figura A3.33: EMA 2 MFA2 C.C.-Registros

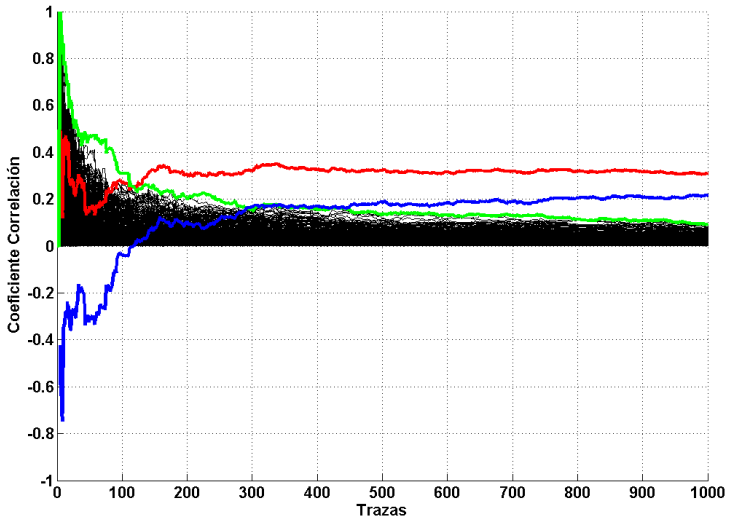


Figura A3.34: EMA 2 MFA2 C.C.-Trazas Registro 75341

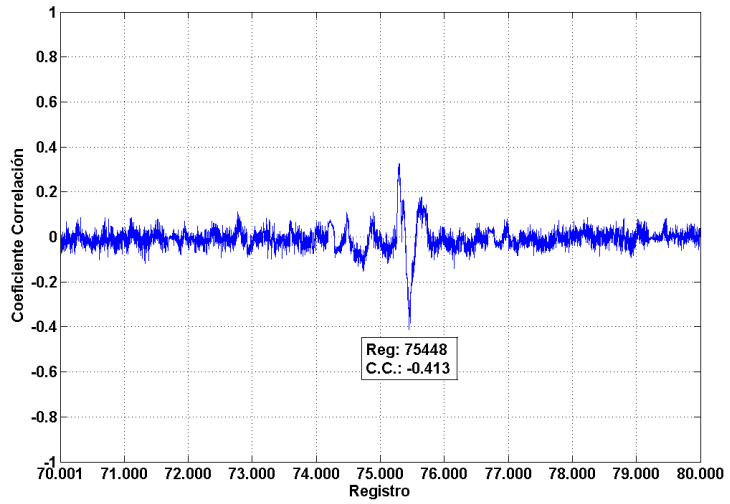


Figura A3.35: EMA 2 MFA3 C.C.-Registros

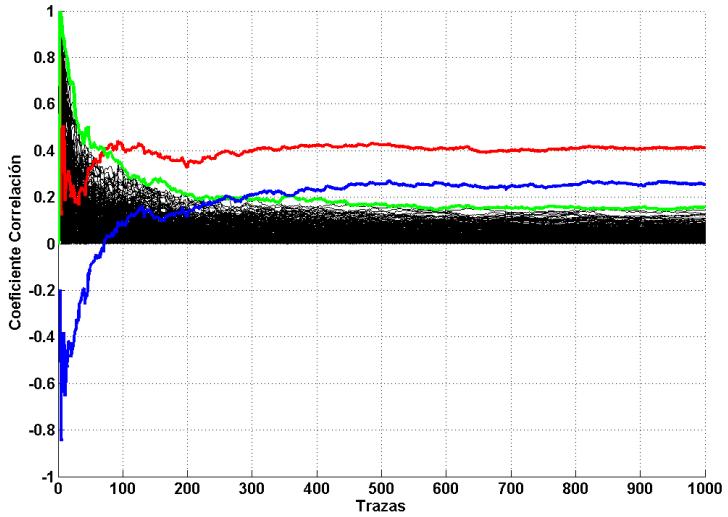


Figura A3.36: EMA 2 MFA3 C.C.-Trazas Registro 75448

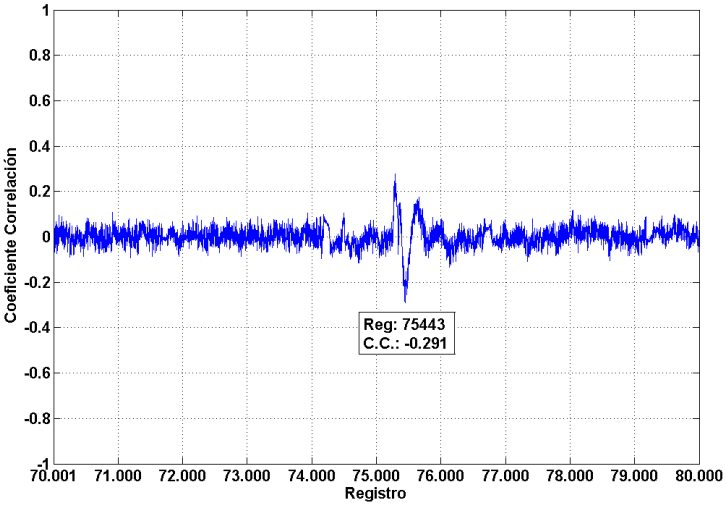


Figura A3.37: EMA 2 MFA4 C.C.-Registros

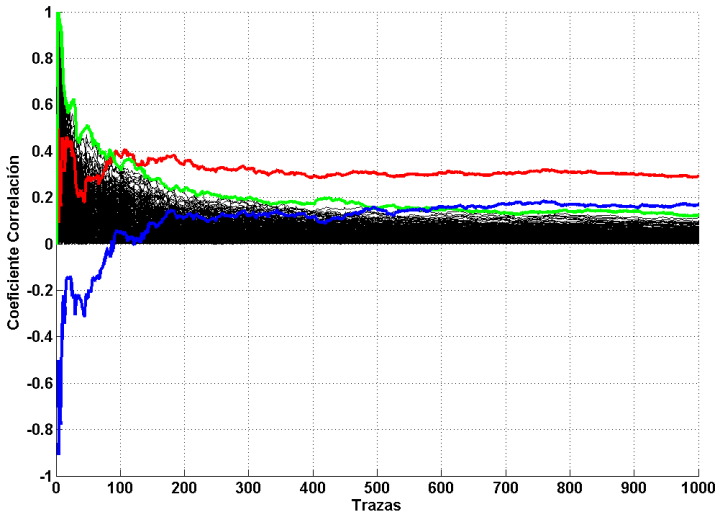


Figura A3.38: EMA 2 MFA4 C.C.-Trazas Registro 75443

HOMEMADE

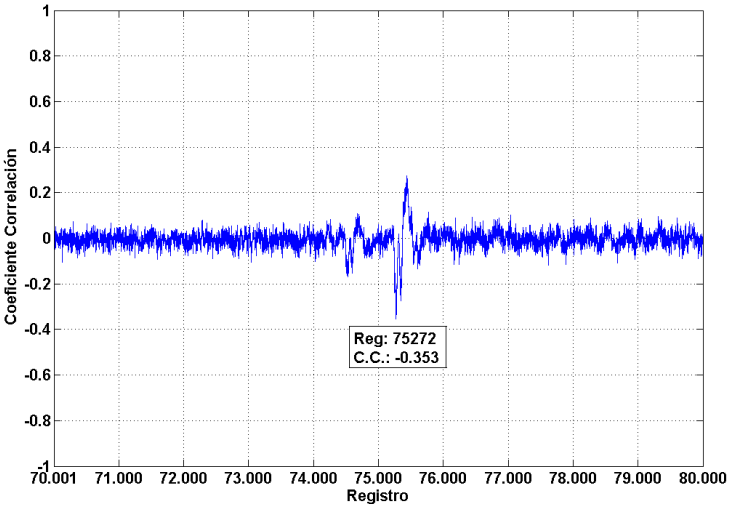


Figura A3.39: EMA 2 HO1 C.C.-Registros

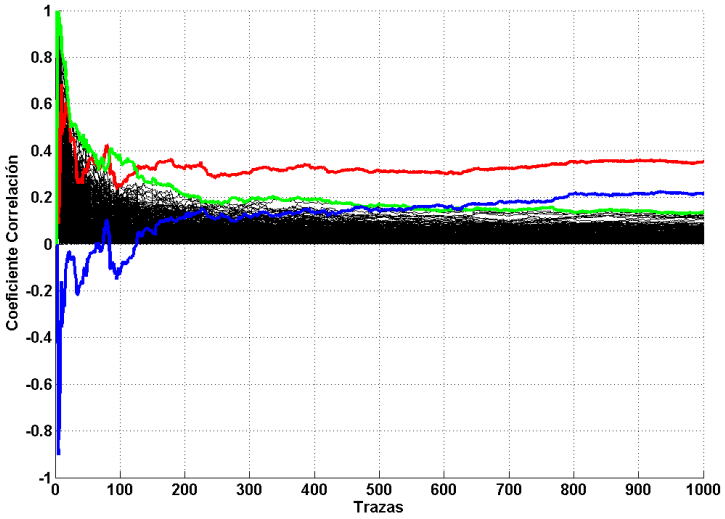


Figura A3.40: EMA 2 HO1 C.C.-Trazas Registro 75272

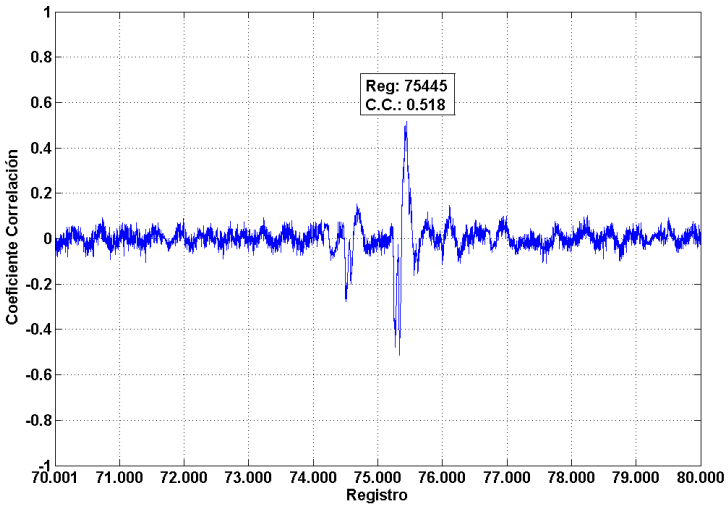


Figura A3.41: EMA 2 HO2 C.C.-Registros

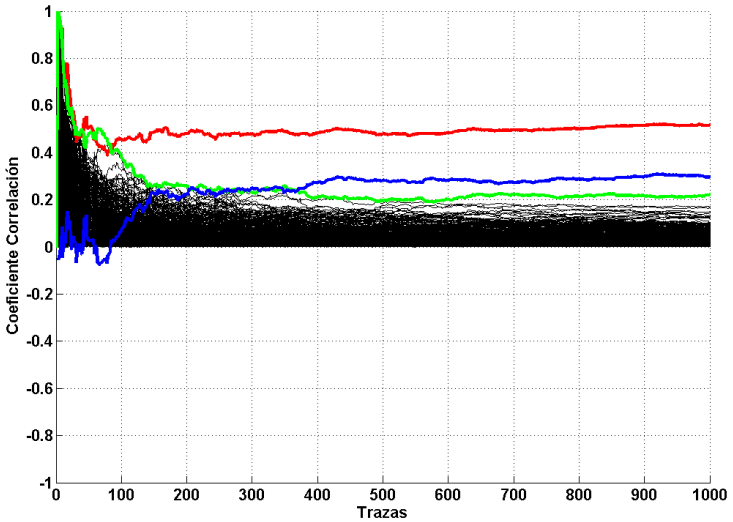


Figura A3.42: EMA 2 HO2 C.C.-Trazas Registro 75445

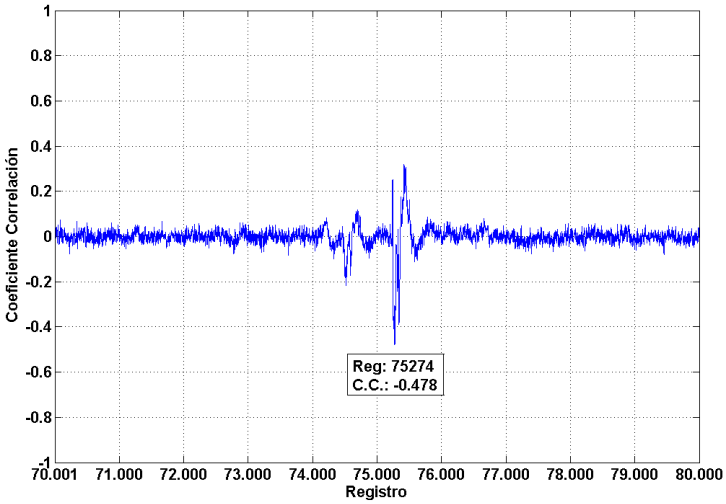


Figura A3.43: EMA 2 HO3 C.C.-Registros

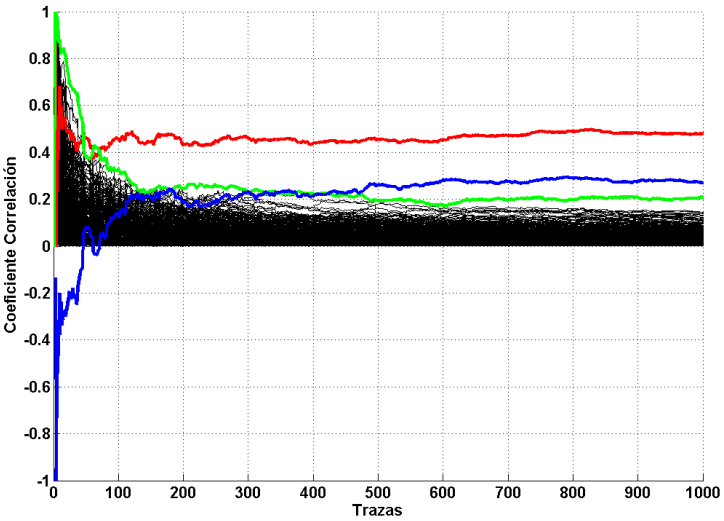


Figura A3.44: EMA 2 HO3 C.C.-Trazas Registro 75274

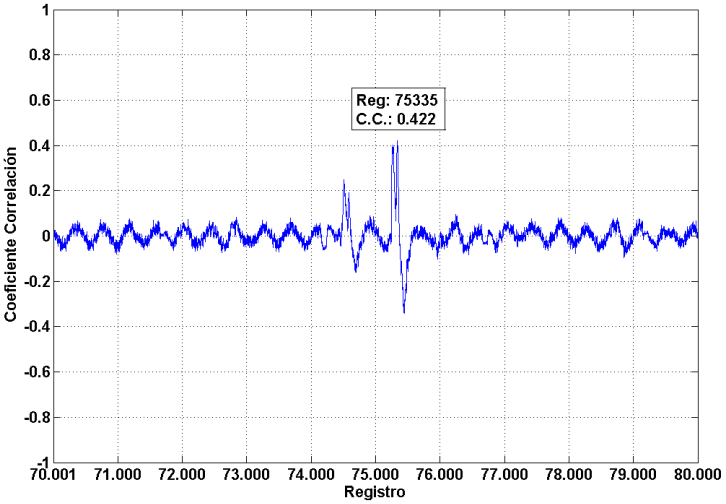


Figura A3.45: EMA 2 HO4 C.C.-Registros

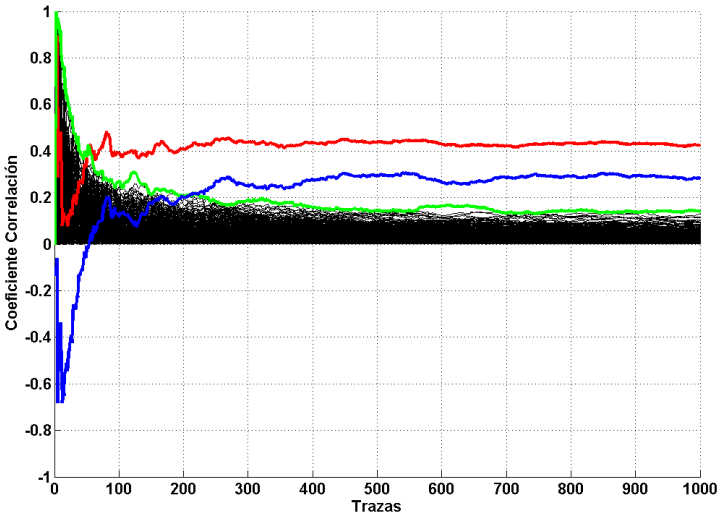


Figura A3.46: EMA 2 HO4 C.C.-Trazas Registro 75335

EMA 3: LPC1769

EM6995

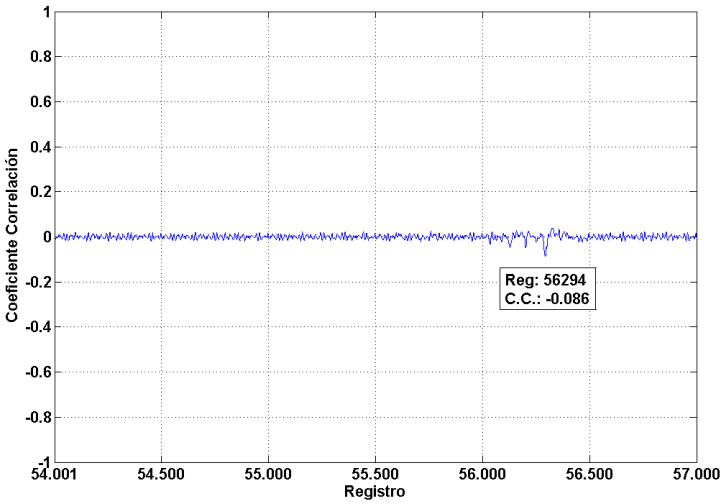


Figura A3.47: EMA 3 EM3 C.C.-Registros

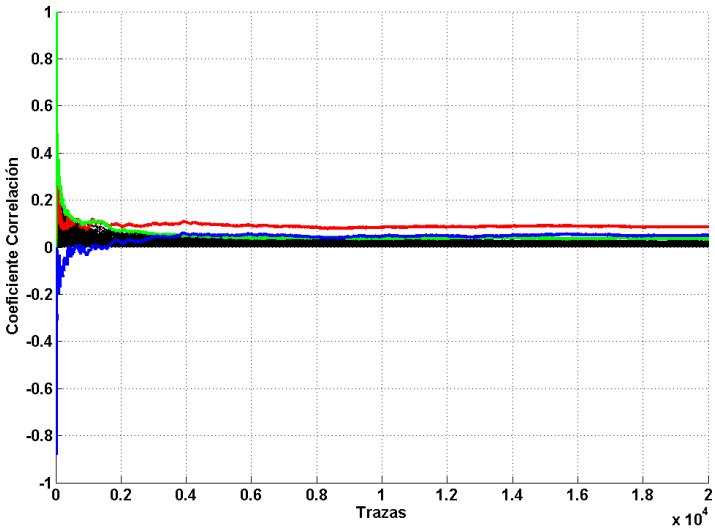


Figura A3.48: EMA 3 EM3 C.C.-Trazas Registro 56294

MFA-R

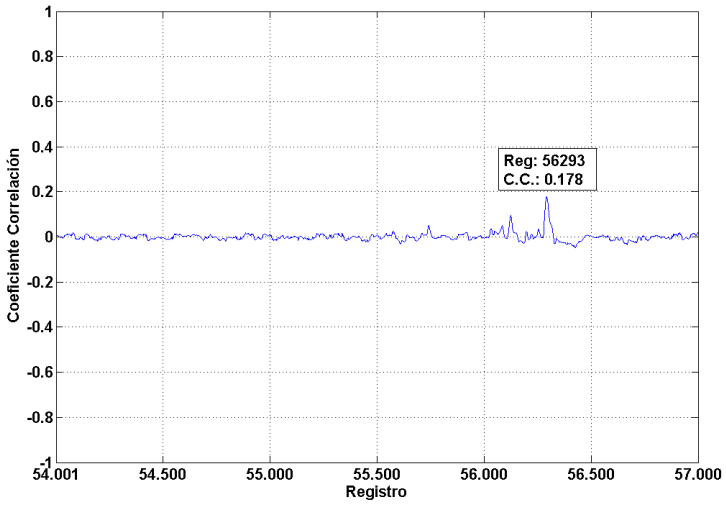


Figura A3.49: EMA 3 MFA2 C.C.-Registros

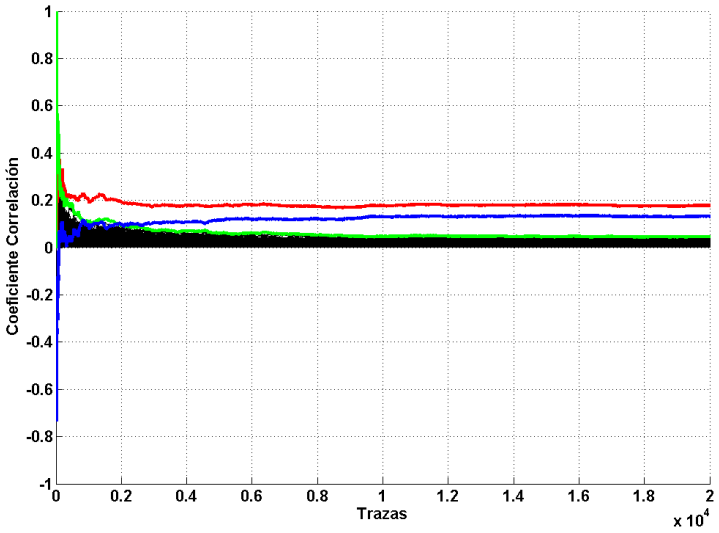


Figura A3.50: EMA 3 MFA2 C.C.-Trazas Registro 56293

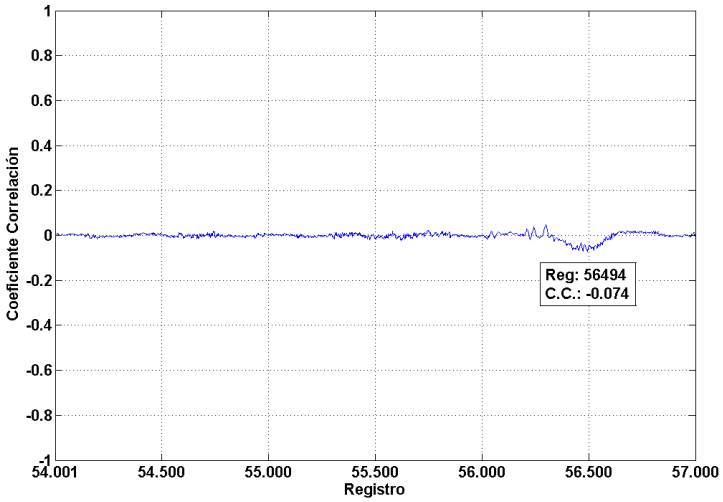


Figura A3.51: EMA 3 MFA4 C.C.-Registros

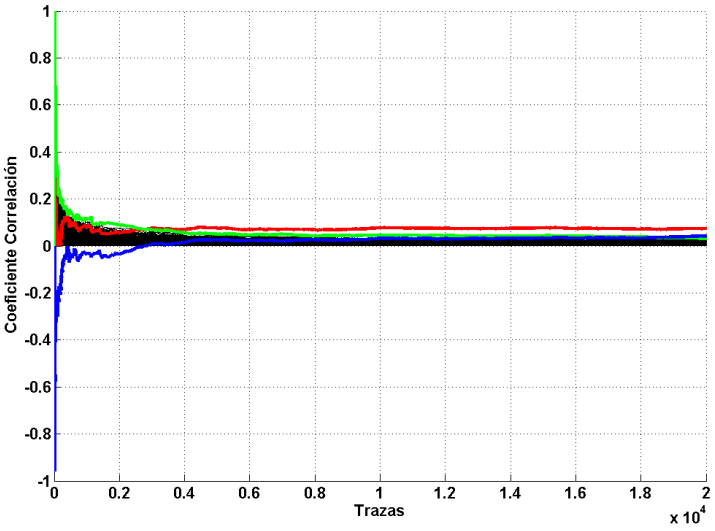


Figura A3.52: EMA 3 MFA4 C.C.-Trazas Registro 56494

HOMEMADE

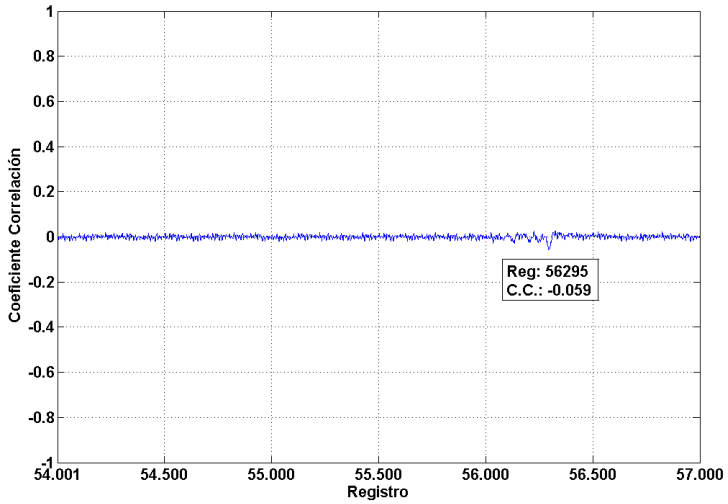


Figura A3.53: EMA 3 HO1 C.C.-Registros

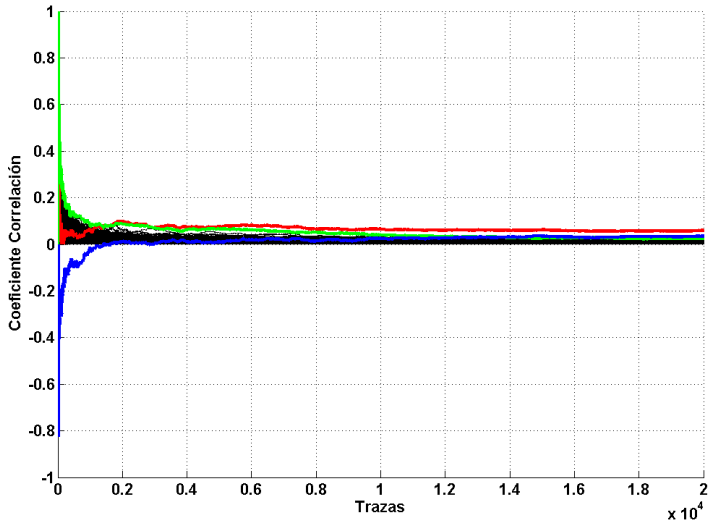


Figura A3.54: EMA 3 HO1 C.C.-Trazas Registro 56295

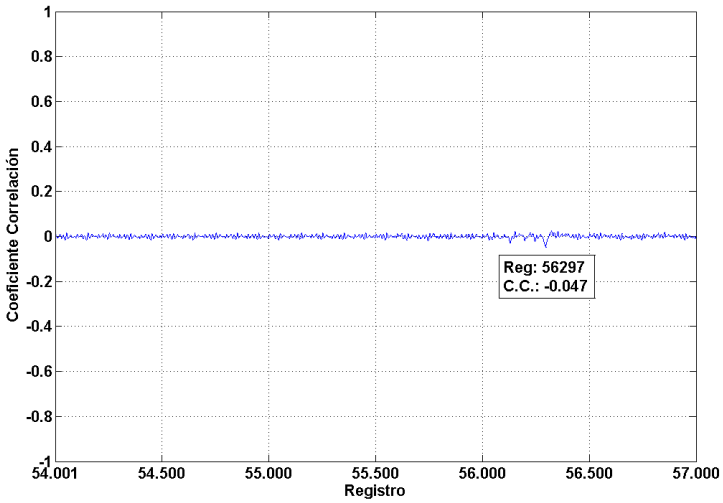


Figura A3.55: EMA 3 HO2 C.C.-Registros

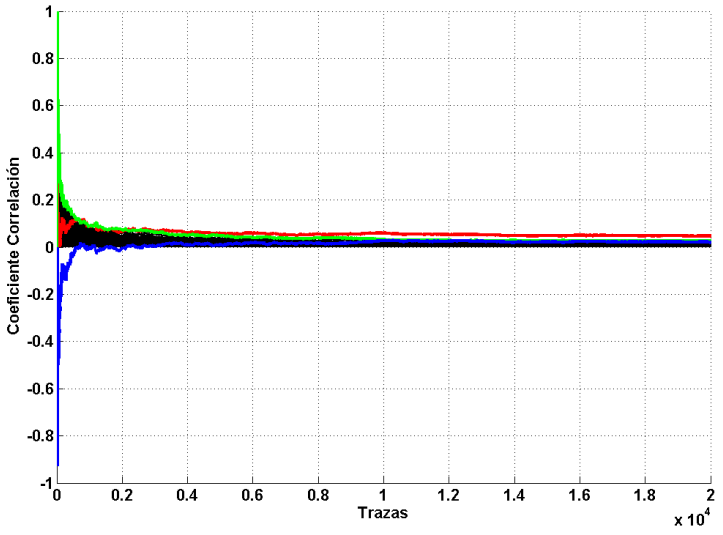


Figura A3.56: EMA 3 HO2 C.C.-Trazas Registro 56297

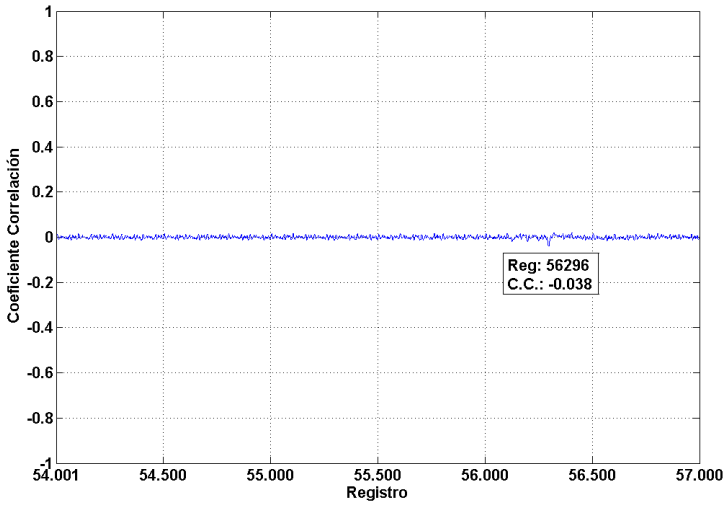


Figura A3.57: EMA 3 HO4 C.C.-Registros

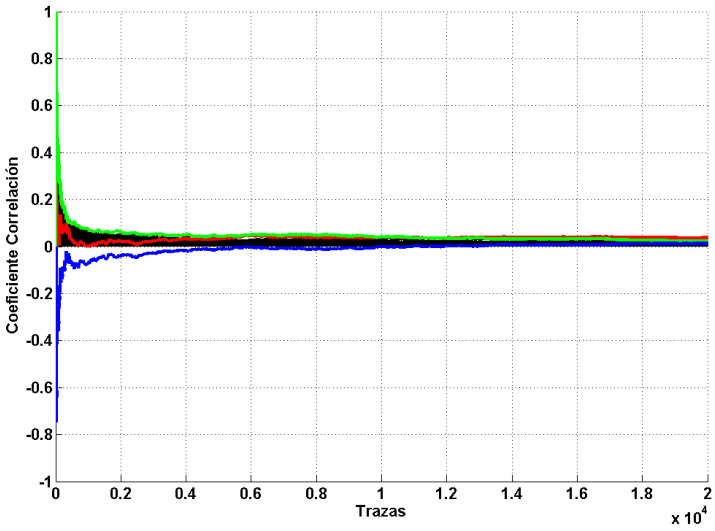


Figura A3.58: EMA 3 HO4 C.C.-Trazas Registro 56296

EMA 4: STM32

EM6995

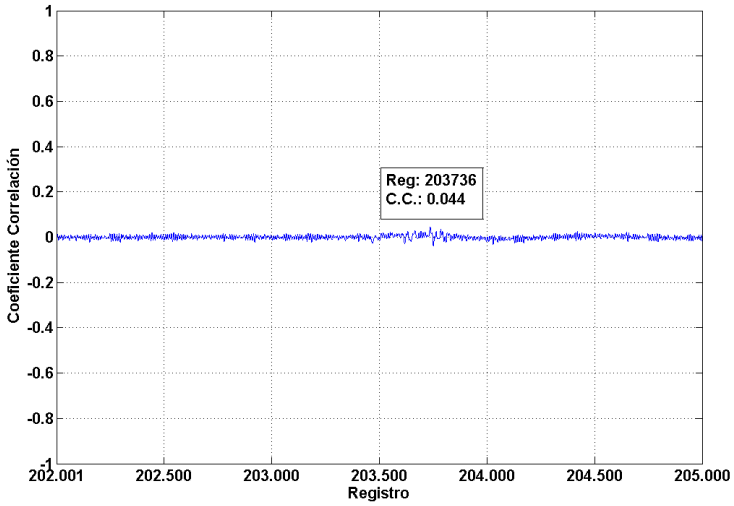


Figura A3.59: EMA 4 EM3 C.C-Registros

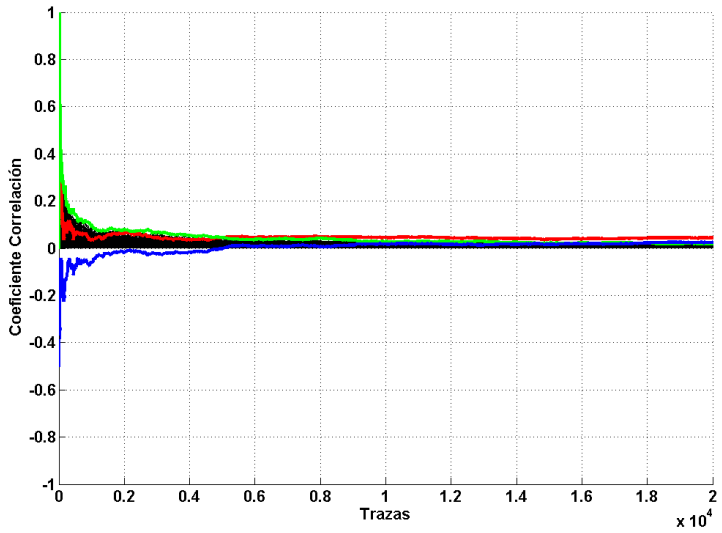


Figura A3.60: EMA 4 EM3 C.C.-Trazas Registro 203736

MFA-R

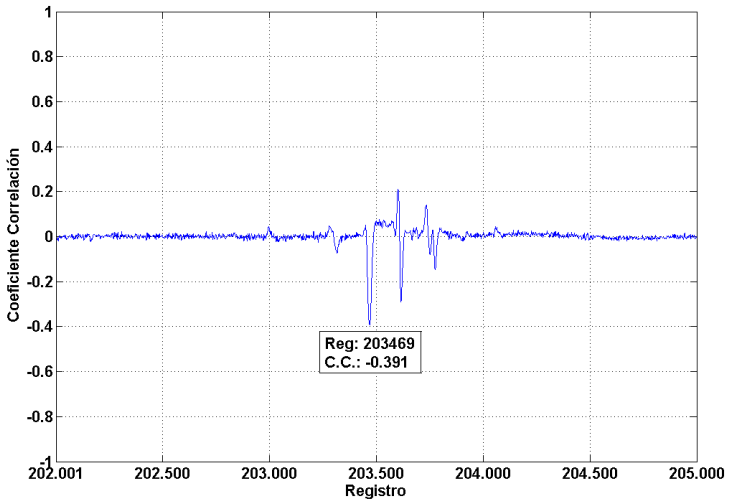


Figura A3.61: EMA 4 MFA2 C.C-Registros

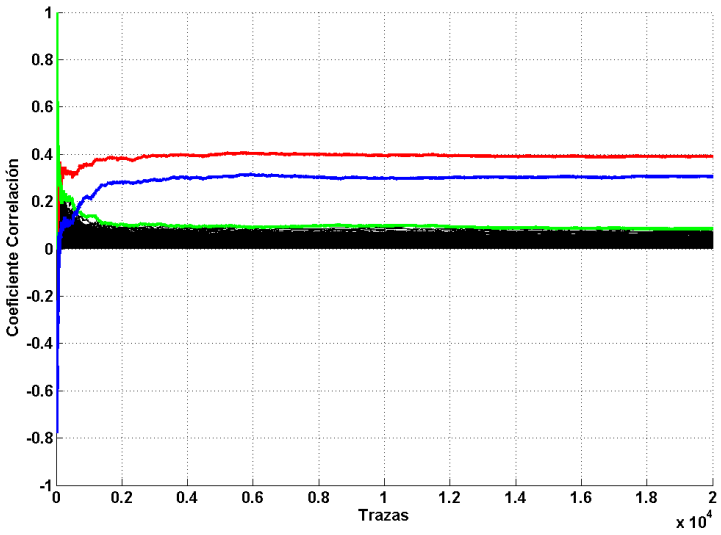


Figura A3.62: EMA 4 MFR2 C.C.-Trazas Registro 203469

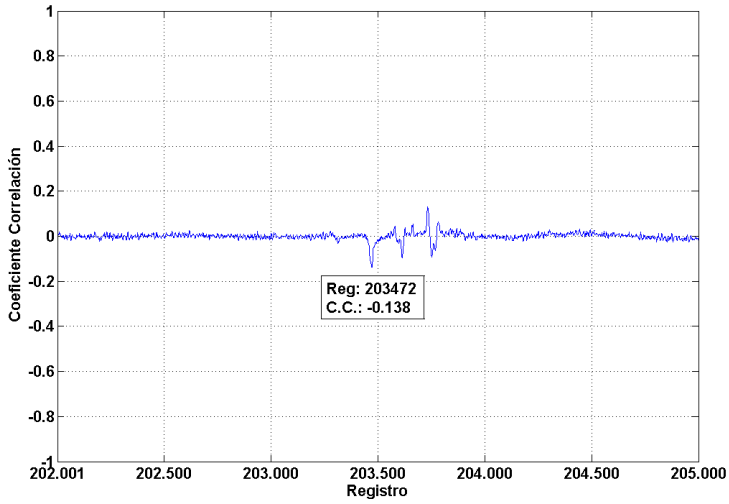


Figura A3.63: EMA 4 MFR3 C.C-Registros

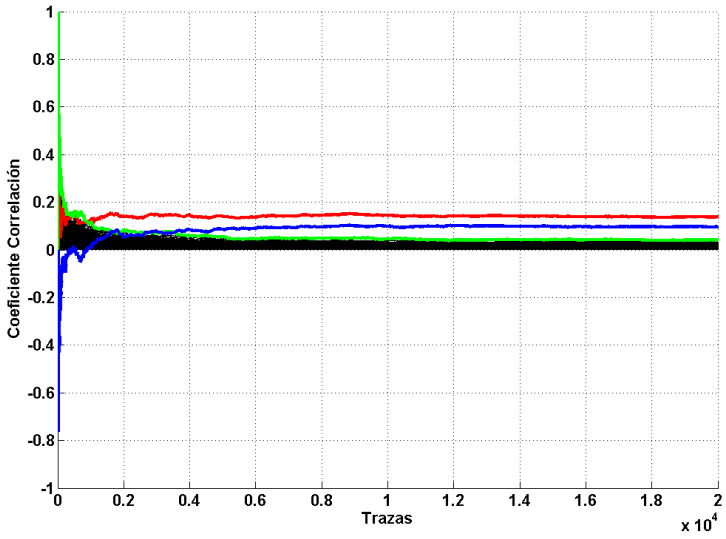


Figura A3.64: EMA 4 MFR3 C.C.-Trazas Registro 203472

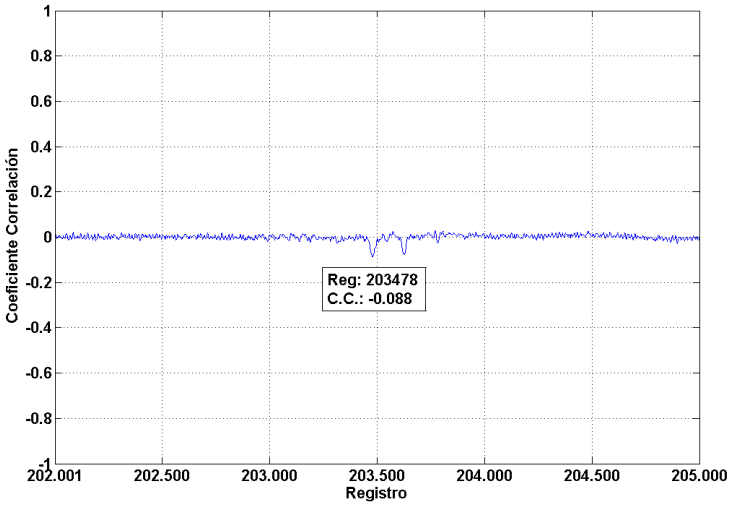


Figura A3.65: EMA 4 MFR4 C.C-Registros

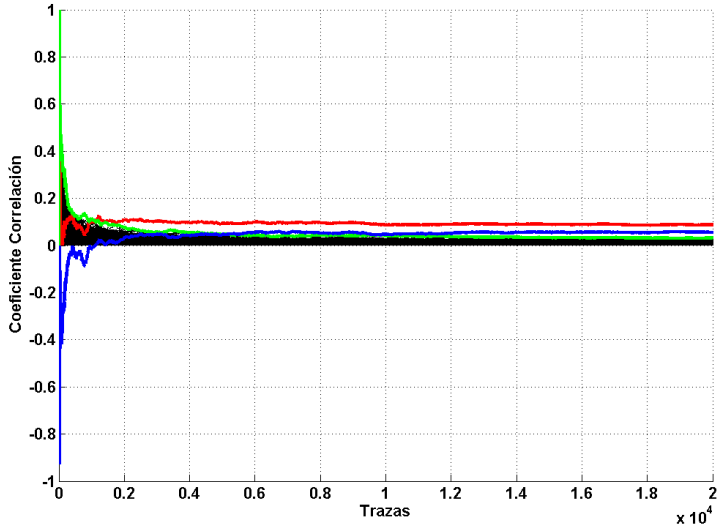


Figura A3.66: EMA 4 MFR4 C.C.-Trazas Registro 203478

HOMEMADE

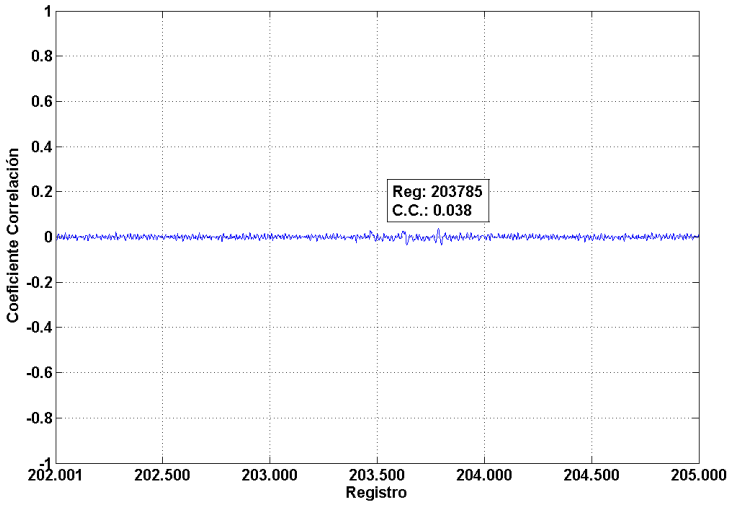


Figura A3.67: EMA 4 HO1 C.C-Registros

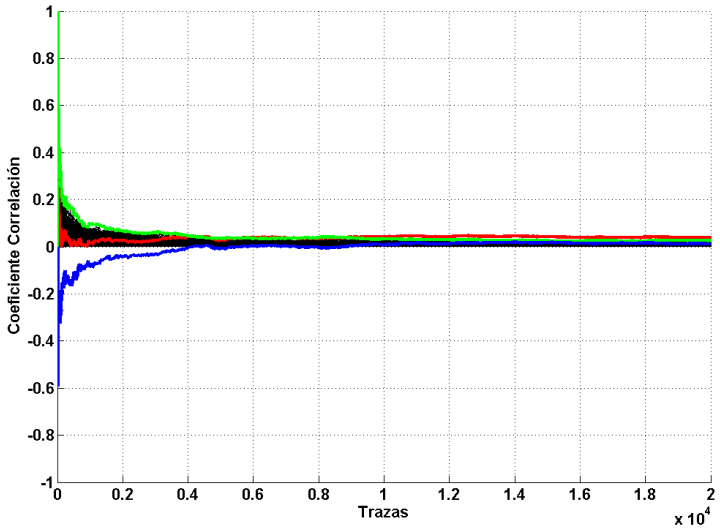


Figura A3.68: EMA 4 HO1 C.C.-Trazas Registro 203785

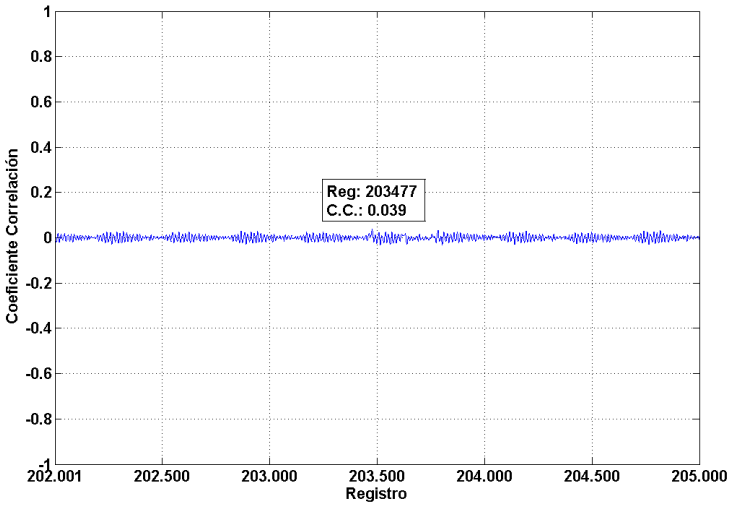


Figura A3.69: EMA 4 HO3 C.C-Registros

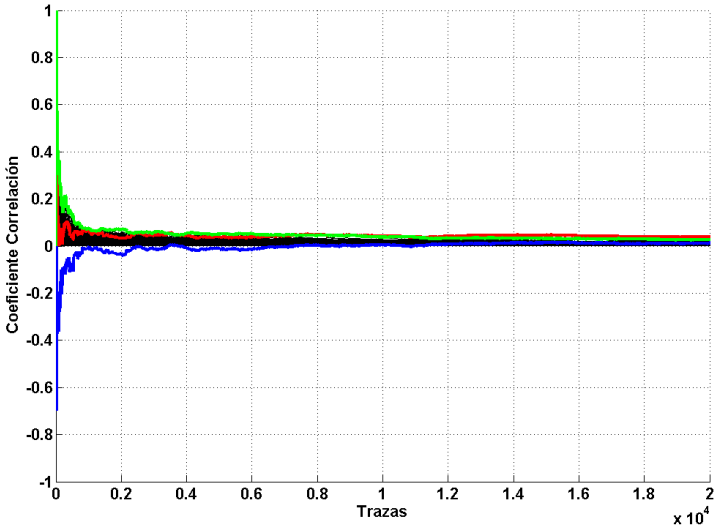


Figura A3.70: EMA 4 HO3 C.C.-Trazas Registro 203477

BIBLIOGRAFÍA

[Add'09] P. S. Addison, J. Walker, R. C. Guido: "Time-Frequency Analysis of Biosignals". *In IEEE Engineering in Medicine and Biology Magazine*, 2009.

[Agr'02] D. Agrawal, B. Archambeault, J. R. Rao, P. Rohatgi: "The EM Side-Channel(s): Attacks and Assessment Methodologies". *In proceedings of CHES 2002*, LNCS 2523, pp. 29-45, Springer-Verlag, 2002.

[Agr'03] D. Agrawal, J. Rao, P. Rohatgi: "Multi-channel Attacks". *In proceedings of CHES 2003*, LNCS 2779, pp. 2-16, Springer-Verlag, 2003.

[Agr'05] D. Agrawal, J. Rao, P. Rohatgi, K. Schramm: "Template as Master Keys". *In proceedings of CHES 2005*, LNCS 3659, pp. 15-29, Springer-Verlag, 2005.

[Aig'00] M. Aigner, E. Oswald: "Power Analysis Tutorial". *Institute for Applied Information Processing and Communication*, Technical Report, University of Technology Graz, 2000.

[Akk'00] M.L. Akkar, R. Bevan, P. Dischamp, D. Moyart: "Power Analysis, What is now Possible…". *In proceedings of ASIACRYPT 2000*, LNCS 1976, pp. 489-502, Springer-Verlag, 2000.

[Akk'01] M.L. Akkar, C. Giraud: "An Implementation of DES and AES, Secure against Some Attacks". *In proceedings* of CHES 2001, LNCS 2162, pp. 309-318, 2001.

[AlMa'09] J. M. Algaba, R. Martínez: "Electromagnetic Analysis – EMA I y II". Tesis de Máster, Dpto. Tecnología Electrónica, Universidad Carlos III de Madrid, 2009.

[Arc'06] C. Archambeau, E. Peeters, F.-X. Standaert, J.-J. Quisquater: "Template Attacks in Principal Subspaces". *In proceedings of CHES 2006*, LNCS 4249, pp. 1-14. Springer, Heidelberg, 2006.

[ARM7'13] ARM: "ARM7 Processor Family". http://www.arm.com/products/processors/classic/arm7/index [Último acceso: 20/03/2013].

[Ato'12] Atollic: "Atollic TrueSTUDIO". http://www.atollic.com/index.php/truestudio [Último acceso: 05/07/2013].

[Ben'03] L. Benini, A. Macii, E. Macii, E. Omerbegovic, M. Poncino, F. Pro: "Energy-Aware Design Techniques for Differential Power Analysis Protection". *In proceedings of DAC'03*, 2003.

[Bev'03] R. Bevan, E. Knudsen: "Ways to Enhance DPA". *In proceedings of ICISC 2002*, LNCS 2587, pp. 327-342, Springer-Verlag, 2003.

[Bir'09] A. Biryukov, D. Khovratovich: "Related-key Cryptanalysis of the Full AES-192 and AES-256". *In proceedings of ASIACRYPT 2009*, LNCS 5912, pp. 1-18, 2009.

[Bir'10] A. Biryukov, O. Dunkelman, N. Keller, D. Khovratovich, A. Shamir: "Key Recovery Attacks of Practical Complexity on AES-256 Variants With Up To 10 Rounds". *In proceedings of EUROCRYPT 2010*, LNCS 6110, pp. 299-319, 2010.

[Bog'07] A. Bogdanov: "Improved Side-Channel Collision Attack on AES". *Selected Areas in Cryptography*, LNCS 4876, pp. 84-95, 2007.

[Bri'04] E. Brier, C. Clavier, F. Olivier: "Correlation Power Analysis with a Leakage Model". *In proceedings of CHES 2004*, LNCS 3156, pp. 16-29, Springer-Verlag, 2004.

[Bro'09] J. Brouchier, T. Kean, C. Marsh, D. Naccache: "Temperature Attacks". *In Security and Privacy IEEE*, Vol. 7-2, pp. 79-82, ISSN: 1540-7993, IEEE Computer Society.

[Bru'03] D. Brumley, D. Boneh: "Remote Timing Attacks Are Practical". *In proceedings of Usenix Security Symp.*, p. 1, 2003.

[Buc'01] J. J. Buchholz: "Matlab Implementation of the Advanced Encryption Standard". http://buchholz.hs-bremen.de.

[Can'05] C. Canovas, J. Clédière: "What do S-boxes Say in Differential Side Channel Attacks?". *Cryptology ePrint Archive*, Report 20085/311, 2005.

[Car'04] V. Carlier, H. Chabanne, E. Dottax, H. Pelletier: "Electromagnetic Side-Channels of an FPGA Implementation of AES", *Cryptology ePrint Archive*, Report: 2004/145, http://eprint.iacr.org/, 2004. [Último acceso: 21/01/2013].

[Car'05] V. Carlier, H. Chabanne, E. Dottax, H. Pelletier: "Generalizing Square Attack Using Side-Channel of an AES Implementation on an FPGA". *IEEE Field Programmable Logic and Applications*, 2005.

[CC'12] Common Criteria v3.1 Release 4 CCMB-2012-09-001 http://www.commo criteriaportal.org/cc/ [Último acceso: 17/09/2013].

[CCN-Tem'06] Centro Criptológico Nacional: "Tempest" https://www.ccncert.cni.es/publico /serieCCN-STIC401/es/t/tempest.htm [Último acceso: 19/07/2013].

[CCN-NTem'12] Centro Criptológico Nacional: "Normativa TEMPEST" https://www.oc. ccn.cni.es/index.php?option=com_content&view=article&id=68&Itemid=12 7&lang=es [Último acceso: 19/07/2013].

[Cha'99] S. Chari, C.S. Jutla, J.R. Rao, P. Rohatgi: "Towards Sound Approaches to Counteract Power-Analysis Attacks". *In proceedings of CRYPTO 99,* LNCS 1666, pp. 398-412, Springer-Verlag, 1999.

[Cha'02] S. Chari, J. Rao, P. Rohatgi: "Template Attacks". *In proceedings of CHES 2002,* LNCS 2523, pp. 13-28, Springer-Verlag, 2002.

[Char'05] X. Charvet, H. Pelletier: "Improving the DPA Attack using Wavelet Transform". *In NIST Physical Security Testing Workshop*, 2005.

[Chat'10] K. Chatzikokolakis, T. Chothia, A. Guha: "Statistical Measurement of Information Leakage". *In TACAS10*, pp. 390-404, 2010.

[Chot'11] T. Chothia, A. Guha: "A Statistical Test for Information Leaks Using Continuous Mutual Information". *In CSF11*, pp. 177-190, 2011.

[Cla'00] C. Clavier, J.-S. Coron, N. Dabbous: "Differential Power Analysis in the Presence of Hardware Countermeasures". *In proceedings of CHES 2000,* LNCS 1965, pp. 252-263, Springer-Verlag, 2000.

[Cor'99] J.S. Coron: "Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems". *In proceedings of CHES 1999,* LNCS 1717, pp. 292-302, Springer-Verlag, 1999.

[Cor'00] J.S. Coron, P. Kocher, D. Naccache: "Statistics and Secret Leakage". *In proceedings of Financial Cryptography*, LNCS 1962, pp. 157-173, Springer-Verlag, 2000.

[Cro'09] S.A. Crosby, D.S. Wallach, R.H. Riedi: "Opportunities and Limits of Remote Timing Attacks". *In proceedings of ACM Trans. Information and System Security*, vol. 12-3, article 17, 2009.

[Dae'99] J. Daemen, V. Rijmen: "AES Proposal: Rijndael", 1999. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.36.640&rep=rep1&type=pdf [Último acceso: 01/03/2013].

[Dae'02] J. Daemen, V. Rijmen: "The Design of Rijndael. AES — the Advanced Encryption Standard". *Springer*, 2002.

[Din'09] G. L. Ding, J. Chu, L. Yuan, Q. Zhao: "Correlation Electromagnetic Analysis for Cryptographic Device". *In proceedings of PACCS'09*, pp. 388-391, 2009.

[Dvam'08] L. Batina, T. Eisenbarth, B. Gierlichs, T. Kasper, F. Koeune, K. Lemke-Rust, F. Macé, E. Oswald, C. Petit, S. Tillich: "Theoretical Models for Side-Channel Attacks". *UCL Crypto Group*, 2008. http://www.ecrypt.eu.org/ecrypt1/documents/D.VAM.15.pdf [Último acceso: 20/02/13].

[Ecl'04] Eclipse – The Eclipse Foundation open source community. http://www.eclipse.org [Último acceso: 02/08/13].

[Edi'09] R. Ediss: "Probing the Magnetic Field Probe". *Philips Semiconductors UK*, 2009. http://www.complianceclub.com/archive /old_archive/030718.htm [Último acceso: 22/07/13].

[EM'07] Electro-Metrics: "EM-6992 Set of E & H Probes, 100 Khz to 1000 Mhz". http://www.electro-metrics.com/product/em-6992/207 [Último acceso: 23/04/13].

[Fah'99] P. Fahn, P. Pearson: "IPA: A new class of Power Attacks". *In proceedings of CHES 1999,* LNCS 1717, pp. 173-186, Springer-Verlag, 1999.

[FTDI'10] FTDI Chip: "TTL-232R USB – TTL Level Serial Converter". http://www.ftdichip.com/Products/EvaluationKits/TTL-232R.htm [Último acceso: 23/04/13].

[Gan'01] K. Gandolfi, C. Mourtel, F. Olivier: "Electromagnetic Analysis: Concrete results". *In proceedings of CHES 2001*, LNCS 2162, pp. 251-261, Springer-Verlag, 2001.

[Geb'03] C.H. Gebotys, R.J. Gebotys: "Secure Elliptic Curve Implementation: an Analysis of resistance to power-attacks in a DSP processor". *In proceedings of CHES 2003*, LNCS 2523, pp. 114-128, 2003.

[Geb'05a] C.H. Gebotys, C.C. Tiu, X. Chen: "A Countermeasure for EM Attack of a Wireless PDA". *In proceedings of ITCC 05*, Vol. 1, pp. 544-549, 2005.

[Geb'05b] C.H. Gebotys, C.C. Tiu, S. Ho: "EM Analysis of Rijndael and ECC on a Wireless Java-based PDA". *In proceedings of CHES 2005*, LNCS 3659, 2005.

[Geb'06] C.H. Gebotys: "A Table Masking Countermeasure for Low-Energy Secure Embedded Systems". *IEEE Transactions on VLSI Systems*, Vol. 14, N7, pp. 740-753, 2006.

[Geb'08a] C.H. Gebotys, B.A. White: "EM Analysis of a Wireless Java based PDA". *In ACM Trans. on Embedded Computing Systems*, Vol. 7 Issue 4 Nº 44, 2008.

[Geb'08b] C.H. Gebotys, B.A. White "EM alignment using phase for secure embedded systems". *Springer Science+Business Media, LLC*, 2008.

[Gen'04] R. Gennaro, A. Lysyanskaya, T. Malkin, S. Micali, T. Rabin: "Algorithmic Tamper-Proof Security: Theoretical Foundations for Security Against Hardware Tampering". *In proceedings of TCC 2004*, LNCS 2951, pp. 258-277, 2004.

[Gie'04] B. Gierlichs, K. Lemke-Rust, C. Paar: "Templates vs. Stochastic Methods". *In proceedings of CHES 2006*, LCNS 4249, pp. 15-29, Springer-Verlag, 2006.

[Gol'02] J. Dj. Golié: "Multiplicative Masking and Power Analysis of AES". *In proceedings of CHES 2002*, LNCS 2523, pp. 198-212, Springer-Verlag, 2002.

[Goo'11a] G. Goodwill, B. Jun, J. Jaffe, P. Rohatgi: "A testing methodology for side-channel resistance validation". *In NIST Non-Invasive Attack Testing Workshop, 2011.*

[Goo'11b] G. Goodwill: "Defending against side-channel attacks". *The Electrochemical Society: ESC*, 2011. http://www.eetimes.com/design/military-aerospace-design/4396658/Defending-against-side-channel-attacks-Part-I [Último acceso: 11/03/2013].

[Gou'99] L. Goubin, J. Patarin: "DES and Differential Power Analysis – The Duplication Method". *In proceedings of CHES 1999*, LNCS 1717, pp. 158-172, Springer-Verlag, 1999.

[Gui'04] S. Guilley, P. Hoogvorst, R. Pacalet: "Differential Power Analysis Model and Some Results". *In proceedings of CARDIS 2004*, pp. 127-142, 2004.

[Ha'05] J. Ha, C. Kim, S. Moon, I. Park, H. Yoo: "Differential Power Analysis on Block Cipher ARIA". *In proceedings of HPCC 2005*, LNCS 3726, pp. 541-548, 2005.

[Hay'12] Y. Hayashi, N. Homma, T. Mizuki, T. Aoki, H. Sone: "Transient IEMI Threats for Cryptographic Devices". *In IEEE transactions on Electromagnetic Compatibility*, 2012.

[Hit'09] Hitex: "The Insider's Guide to the STM32 ARM Based microcontroller v.1.8". Hitex (UK) Ltd., 2009. ISBN: 0-9549988 8.

[Hod'11] P. Hodgers, K.H. Boey, M. O'Neill: "Variable Window Power Spectral Density Attack". *In WIFS'2011*, 2011.

[Hom'06] N. Homma, S. Nagashima, T. Sugawara, T. Aoki, A. Satoh: "A High-Resolution Phase-Based Waveform Matching and Its Application to Side-Channel Attacks". In proceedings of CHES 2006, LNCS 4249, pp. 187-200, 2006.

[Hor'12] Y. Hori1, T. Katashita, A. Sasaki, A. Satoh: "Electromagnetic Side-channel Attack against 28-nm FPGA Device". *In proceedings of WISA 2012*, 2012.

[Hos'11] G. Hospodar, E. De Mulder, B. Gierlichs, I. Verbauwhede, J. Vandewalle: "Least Squares Support Vector Machines For Side-Channel Analysis". *In proceedings of COSADE 2011*, 2011.

[Hut'12] M. Hutter, M. Kirschbaum, T. Plos, J.-M. Schmidt, S. Mangard: "Exploiting the Difference of Side-Channel Leakages". *In proceedings of COSADE 2012*, LNCS 7275, pp. 1-16, Springer-Verlag, 2012.

[Hutt'93] D.P. Huttenlocher, G.A. Klanderman, W.J. Rucklidge: "Comparing images using the Hausdorff distance". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15-9, pp. 850-863, 1993.

[IntIdp'13] IntelPR: "Krzanich Announces New Lower-Power Product Family". Intel Developer Forum IPD2013, San Francisco, 10/09/2013. https://intel-newsroom.jive-mobile.com:443/#jive-document?content=%2Fapi%2Fcore%2Fv2%2Fposts%2F6662 [Último acceso: 12/09/2013].

[Ito'02] K. Itoh, M. Takenaka, and N. Torii: "DPA countermeasure based on the masking method". *In ICIS2001*, LCNS 2288, pp. 440-456, Springer-Verlag, 2002.

[Jaf'06] J. Jaffe: "More Differential Power Analysis: Selected DPA Attacks". *In ECRYPT 06*, 2006.

[Jaf'11] J. Jaffe, P. Rohatgi: "Efficient Side-Channel Testing for Public Key Algorithm: RSA Case Study". *In NIST Non-Invasive Attack Testing Workshop*, 2011.

[Joh'99] Fred Johnson: "Simple "Homemade" Sensors Solve Tough EMI Problems". *Electronic Design*, pp. 109-114, 1999.

[Joy'05] M. Joye, P. Paillier, B. Schoenmarkers: "On Second-Order Differential Power Analysis". *In proceedings of CHES 2005*, LNCS 3659, pp. 293-308, Springer-Verlag, 2005.

[Kai'10] G. Kaiser: "A Friendly Guide to Wavelets". *Springer*, *Birkhauser*, 2010. ISBN: 978-0-8176-8110-4.

[Kam'11] K. Gupta, S. Silakari: "ECC over RSA for Asymmetric Encryption: A Review". *IJCSI International Journal of Computer Science Issues*, Vol. 8, Issue 3, No. 2, 2011.

[Kei'04] KEIL: "Ulink Debug Adapter" http://www.keil.com/ ulink1/ [Último acceso: 08/07/2013].

[Kei'08] KEIL: "µVision IDE". http://www.keil.com/uvision/ [Último acceso: 08/07/2013].

[Kel'98] J. Kelsey, B. Schneier, D. Wagner, C. Hall: "Side Channel Cryptanalysis of Product Ciphers". *In proceedings of ESORICS 98*, LNCS 1485, pp. 97-110, 1998.

[Ken'12] G. Kenworthy, P. Rohatgi: "Mobile Device Security: The case for side channel resistance". *In proceedings of MoST2012*, 2012.

[Koc'96] P. Kocher: "Timing Attacks on Implementations of Diffie-Hellman". *In proceedings of CRYPTO 1996*, LNCS 1109, pp. 104-113, 1996.

[Koc'99] P. Kocher, J. Jaffe, B. Jun: "Differential Power Analysis". *In proceedings of CRYPTO 1999*, LNCS 1666, pp. 388-397, Springer-Verlag, 1999.

[Koc'04] P. Kocher, R. Lee, G. McGraw, A. Raghunathan, S. Ravi: "Security as a New Dimension in Embedded System Design". *In DAC 2004*, 2004.

[Koe'05] F. Koeune, F.–X. Standaert: "A Tutorial on Physical Security and Side-Channel Attacks". *In FOSAD 2004/2005*, LNCS 3655, pp. 78-108, Springer-Verlag, 2005.

[Köp'07] B. Köpf, D. Basin: "An Information-Theoretic Model for Adaptive Side-Channel Attacks". *In proceedings of CCS'07*, 2007.

[Kuh'02] M.G. Kuhn: "Optical Time-domain Eavesdropping Risks of CRT Displays". *In proceedings of the 2002 IEEE Symposium on Security and Privacy*, pp. 3–18, 2002.

[LanMFA'10] Langer EMV-Technik Gmbh: "Near Field Probe set MFA 01". http://www.langer-emv.de/en/products/disturbance-emission/near-field-probes/mfa-01/ [Último acceso: 23/04/2013].

[LanPA'08] Langer EMV-Technik Gmbh: "Preamplifier PA". http://www.langer-emv.de/fileadmin/website/dokumente/produkt_ details/PA-0508pe.pdf [Último acceso: 23/04/2013].

[Le'06] T.-H. Le, J. Clédière, C. Canovas, B. Robisson, C. Servière, J.-L. Lacoume: "A proposition for Correlation Power Analysis enhancement". *In proceedings of CHES 2006*, LNCS 4249, pp. 174-186, Springer-Verlag, 2006.

[Le'07] T.-H. Le, J. Clédière, C. Servière, and J.-L. Lacoume: "Noise Reduction in Side Channel Attack Using Fourth-Order Cumulant". *In proceedings of TIFS 2007*, vol. 2, no. 4, pp. 710–720, 2007.

[Le'08] T.-H. Le, C. Canovas, J. Clédière: "An overview of Side Channel Analysis Attacks". *In proceedings of ASIACCS 2008*, 2008.

[Lev'04] B.N. Levine, M.K. Reiter, C. Wang, M. Wright: "Timing Attacks in Low-Latency Mix Systems". *In proceedings of FC 2004*, LNCS 3110, pp.251-265, 2004.

[Li'10] H. Li, K. Wu, F. Yu: "Enhanced correlation power analysis attack against trusted systems". *Security and Communication Networks 2011*, 4:3-10, 2011.

[Lou'02] J. Loughry, D. Umphress: "Information Leakage from Optical Emanations". *ACM Transactions on Information and System Security*, Vol. 5, pp. 262-289, 2002.

[Lum'13] R. Lumbiarres-López, M. López-García, E. F. Cantó-Navarro: "Ataques por canal lateral sobre el algoritmo de encriptación AES implementado en MicroBlaze". *JCRA'2013*, 2013.

[Mae'00] R. D. Maesschalck, D. Jouan-Rimbaud, D.L. Massart: "The Mahalanobis distance Tutorial". Chemometrics and Intelligent Laboratory Systems, Vol. 50, Issue 1, pp. 1-18, 2000.

[Mah'36] P.C. Mahalanobis: "On the generalized distance in statistics". *In proceedings of the National Institute of Science of India*, 12, pp. 49–55, 1936.

[Man'03] S. Mangard: "A Simple Power-Analysis (SPA) Attack on Implementations of the AES Key Expansion". *In proceedings of ICISC 2002*, LNCS 2587, pp. 343–358, 2003.

[Man'05a] S. Mangard, N. Pramstaller, E. Oswald: "Successfully Attacking Masked AES Hardware Implementations". *In proceedings of CHES 2005*, LNCS 3659, pp. 157–171, 2005.

[Man'05b] S. Mangard, T. Popp, B.M. Gammel: "Side-Channel Leakage of Masked CMOS Gates". *In proceedings of Topics in Cryptology – CT – RSA 2005*, LNCS 3376, pp. 351-365, 2005.

[Man'06] S. Mangard, K. Schramm: "Pinpointing the Side-Channel Leakage of Masked AES Hardware Implementations". *In proceedings of CHES 2006*, LNCS 4249, pp. 76-90, Springer-Verlag, 2006.

[Man'07] S. Mangard, E. Oswald, T. Popp: "Power Analysis Attacks – Revealing the Secrets of Smart Cards". *Springer Science + Business Media*, 2007. ISBN-13: 978-0-387-30857-9.

[Mat'10] E. Mateos, C.H. Gebotys: "A New Correlation Frequency Analysis of the Side Channel". *In proceedings of WESS'10*, 2010.

[Math'13] L. Mather, E. Oswald, J. Bandenburg, M. Wójcik: "A Comparison of Statistical Techniques for Detecting Side-Channel Information Leakage in Cryptographic Devices". *Cryptology ePrint archive*, report: 2013/298, 2013. http://eprint.iacr.org/2013/298.pdf [Último acceso: 12/09/2013].

[MathFFT'10] The MathWorks, Inc: "FFT Discrete Fourier transform". *Matlab 2010a Help*, 2010.

[MathGui'10] The MathWorks, Inc: "Graphical User Interfaces in MATLAB" http://www.mathworks.es/discovery/matlab-gui.html [Último acceso: 03/05/2013].

[MathIns'10] The MathWorks, Inc: "Instrument Control Toolbox" http://mathworks.es /products/instrument/ [Último acceso: 03/05/2013].

[MathMat'10] The MathWorks, Inc: "MatLab R2010" http://www.mathworks.es /products/matlab/ [Último acceso: 07/05/2013].

[MathWav'10] The MathWorks, Inc: "Wavelet Toolbox" http://www.mathworks.es /products/wavelet/ [Último acceso: 10/07/2013].

[MaxDSM'13] Maxim: "DeepCover Secure Microcontroller Max32590" http://www. maximintegrated.com/datasheet /index.mvp/id/ 7538.

[May'00] R. Mayer-Sommer: "Smartly Analysing the Simplicity and the Power of Simple Power Analysis on Smartcards". *In proceedings of CHES 2000*, LNCS 1965, pp. 78-92, Springer-Verlag, 2000.

[Mes'99] T.S. Messerges, E.A. Dabbish, R.H. Sloan: "Investigation of Power Analysis Attacks on Smartcards". *In Usenix Workshop on Smartcard Technology*, 1999.

[Mes'00a] T.S. Messerges: "Using Second-Order Power Analysis to Attack DPA Resistant Software". *In proceedings of CHES 2000*, LNCS 1965, pp. 238-251, Springer-Verlag, 2000.

[Mes'00b] T.S. Messerges: "Securing the AES Finalists against Power Analysis Attacks". *In proceedings of Fast Software Encryption Workshop* 2000, Springer-Verlag, 2000.

[Mes'01] Thomas S. Messerges: "Securing the Rijndael finalists against Power Analysis Attacks". *In proceedings of Fast Software Encryption Workshop*, LNCS 1978, pp. 150–164, Springer-Verlag, 2001.

[Mes'02] T.S. Messerges, E.A. Dabbish, R.H. Sloan: "Examining Smart-Card Security under the Threat of Power Analysis Attacks". *IEEE Transactions on Computers*, Vol. 51, N5, pp. 541-552, 2002.

[Mey'10] O. Meynard, S. Guilley, J.-L. Danger, L. Sauvage: "Far Correlation-based EMA with a Precharacterized Leakage Model". *In EDDA 2010*, 2010.

[Mey'11] O. Meynard, D. Réal, S. Guilley, F. Flament, J.-L. Danger, F. Valette: "Characterization of the Electromagnetic Side Channel in Frequency Domain". *Information Security and Cryptology*, LNCS 6584, pp. 471-486, Springer-Verlag, 2011.

[Mic'04] S. Micali, L. Reyzin: "Physically Observable Cryptography". *In proceedings of TCC 2004*, LNCS 2951, pp. 278-296, 2004.

[Mis'10] M. Misiti, Y. Misiti, G. Oppenheim, J.-M. Poggi: "Wavelet Toolbox 4; User's Guide" The MathWorks, Inc., 2010.

[Mor'10] A. Moradi, O. Mischke, T. Eisenbarth: "Correlation-Enhanced Power Analysis Collision Attack". *In proceedings of CHES 2010*, LNCS 6225, pp. 125-139, Springer-Verlag, 2010.

[Mul'05] E.D. Mulder, P. Buysschaert, S. B. Ors, P. Delmotte, B. Preneel, G. Vandenbosch, I. Verbauwhede: "Electromagnetic Analysis Attack on an FPGA Implementation of an Elliptic Curve Cryptosystem". *In proceedings of EUROCON 2005*, Vol. 2, pp. 1879-1882, 2005.

[NILab'03] National Instruments: "LabVIEW User Manual". Part Number 320999E-01, 2003. www.ni.com/pdf/manuals/320999e.pdf [Último acceso: 07/05/2012].

[NILab'07] National Instruments: "Labview 8.5" http://www.ni.com/labview/esa/ [Último acceso: 07/05/2012].

[NILab'09a] National Instruments: "Introducción a Labview – Curso Práctico 6 horas". http://www.ni.com/academic/labview_training/esa/ [Último acceso: 07/05/2012].

[NILab'09b] National Instruments: "LabVIEW Help for step-by-step instructions". www.ni.com/pdf/manuals/323563b.pdf [Último acceso: 07/05/2012].

[NILab'09c] National Instruments: "Tektronix DPO MSO 2000 4000 Series Oscilloscope Certified LabVIEW Plug and Play (project-style) Instrument Driver". http://sine.ni.com/apps/utf8/niid_web_display.download_page?p_id_guid=6E23DB10D9F C2B05E04400144FB7D21D [Último acceso: 07/05/2012]

[Nist'99] National Institute of Standards and Technology: "FIPS 46-3 Data Encryption Standard (DES)", 1999. http://csrc.nist.gov/publications/fips/fips46-3/fips46-3.pdf [Último acceso: 21/01/2013].

[Nist'01] National Institute of Standards and Technology: "FIPS 197 Advanced Encryption Standard (AES)", 2001. http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf [Último acceso: 21/01/2013].

[Nist'11] National Institute of Standards and Technology: "Non-Invasive Attack Testing Workshop". http://csrc.nist.gov/news_events/non-invasive-attack-testing-workshop, 2011. [Último acceso: 12/09/2013].

[Nist'13] National Institute of Standards and Technology: "Advanced Encryption Standard Algorithm Validation List", 2013. http://csrc.nist.gov/groups/STM /cavp/documents/aes/aesval.html [Último acceso: 06/03/2013].

[NSA'03] NSA: "NSA Tempest Documents", 2003. http://www.cryptome.org/nsa-tempest.htm. [Último acceso: 06/03/2013]

[NXP'08] NXP Semiconductors: "UM10114: LPC21xx and LPC22xx User manual". Rev. 03 - 2 April 2008. www.nxp.com/documents/user_manual /UM10114.pdf [Último acceso: 07/05/2012].

[NXP'10] NXP Semiconductors: "UM10360: LPC17xx User manual User manual". Rev. 2 - 19 August 2010. www.nxp.com/documents/user_manual /UM10360.pdf [Último acceso: 07/05/2012].

[NXP'11] NXP Semiconductors: "LPCXpresso" http://www.nxp.com/techzones/ microcontrollers-techzone/tools-ecosystem/lpcxpresso.html [Último acceso: 05/07/2013].

[Ors'04] S.B. Örs, F. Gürkaynak, E. Oswald, B. Preneel: "Power-Analysis Attack on an ASIC AES implementation". *In proceedings of ITCC 04*, Vol. 2, pp. 546-552, 2004.

[Ost'03] T. Ostermann, B. Deutschmann: "Characterization of the EME of Integrated Circuits with the Help of the IEC Standard 61967". *In proceedings of ETW'03*, 2003.

[Osv'06] D.A. Osvik, A. Shamir, E. Tromer: "Cache Attacks and Countermeasures: The Case of AES". Topics in Cryptology—CTRSA 2006, LNCS 3860, pp. 1–20, Springer, 2006.

[Osw'07] E. Oswald, S. Mangard: "Template Attacks on Masking—Resistance Is Futile". *In proceedings of CT-RSA 2007*, LNCS 4377, pp. 243–256, 2007.

[Par'13] P. Park: "Security Attacks vs. Countermeasures". Samsung Electronics Co., Ltd. Technical Paper. http://techonline.com/electrical-engineers/education-training/tech-papers/4423661/Security-Attacks-vs-Countermeasures [Último acceso: 04/11/13].

[Pee'07] E. Peeters, F.-X. Standaert, J.-J. Quisquater: "Power and Electromagnetic Analysis: Improved Models, Consequences and Comparisons". *In Integration, the VLSI Journal*, Vol. 40, pp. 52-60, 2007.

[Pol'01] R. Polikar: "The Engineer's Ultimate Guide to Wavelet Analysis – The Wavelet Tutorial". *NDSL Scout Report for Math, Engineering and Technology*, 2001. http://Users.rowan.edu/~polikar/Wavelets/ WTutorial.html [Último acceso: 23/07/13].

[Pom'10] Pomona Electronics: "Model 2249 Cable Assembly with BNC Male on Each End" http://www.pomonaelectronics.com/pdf/d2249_100.pdf.

[Pop'07] T. Popp, E. Oswald, S. Mangard: "Power Analysis Attacks and Countermeasures". *IEEE Design and Test of Computers*, IEEE, 2007.

[Pra'04] N. Pramstaller, F.K. Gürkaynak, S. Haene, H. Kaeslin, N. Felber, W. Fichtner: "Towards an AES Cryto-chip Resistant to Differential Power Analysis". *In proceedings of ESSCIRC 2004*, pp. 307-310, 2004.

[Qui'01] J.-J. Quisquater, D. Samyde: "Electromagnetic analysis (EMA): Measures and Countermeasures for Smart Cards". *In E-smart 2001*, LNCS 2140, pp. 200-210, Springer-Verlag, 2001.

[Qui'02a] J.-J. Quisquater, D. Samyde: "Automatic Code Recognition for Smartcards using Kohonen Neural Network" *In proceedings of CARDIS 02*, Vol. 5, Springer-Verlag, 2002.

[Qui'02b] J.-J. Quisquater, F. Koeune: "State-of-the-Art regarding Side Channel Attacks". http://www.ipa.go.jp/security/enc/CRYPTREC/fy15/doc/1047_Side_Chan nel _report.pdf [Último acceso: 21/01/2013].

[R'10] The R Project for Statistical Computing: "R". http://www.r-project.org/ [Último acceso: 07/05/2013].

[Rab'03] J.M. Rabaey, Chandrakasan, B. Nikolic: "Digital Integrated Circuits – A Design Perspective". *Electronic and VLSI Series*, Prentice Hall, 2ª Ed., 2003.

[Rau'03] C. Rauscher: "Fundamentos del Análisis de Espectro". Rohde&Schwarz, 2003. ISBN: 3939837032.

[Rav'04] S. Ravi, A. Raghunathan, P. Kocher, S. Hattangady: "Security in Embedded Systems: Design Challenges". *ACM Transactions on Embedded Computing Systems*, Vol. 3-3, pp. 461-491, 2004.

[Rea'09] D. Réal, F. Valette, M. Drissi: "Enhancing Correlation Electromagnetic Attack Using Planar Near-Field Cartography". *In Proceedings of the DATE 09*, pp. 628-633, 2009.

[Rec'04] C. Rechberger, E. Oswald: "Practical Template Attacks". *In proceedings of WISA 2004*, LNCS 3325, 2004.

[Rid'99] F. P. Ridao, J. M. Carrasco, E. Galván, L. G. Franquelo: "Implementation of low cost current probes for conducted EMI interference measure in Power Systems". *In proceedings of EPE'99*, 1999.

[RS'04] Rohde&Schwarz: "ESCI EMI Test Receiver". http://www.rohde-schwarz.com/en/product/esci-productstartpage_63493 -11628.html [Último acceso: 08/07/2013].

[RSA'97] RSA Data Security Inc: "Government Encryption Standard Takes a Fall". *RSA Data Press Relace*, 1997.

[Rub'10] J. Rubio, J. E. Posada, J. A. Garcia-Souto: "Digital Signal Processing for the Detection and Location of Acoustic and Electric Signals from Partial Discharges". *In proceedings of WCE 2010*, Vol II, ISBN: 978-988-18210-7-2, 2010.

[Sad'06] S. Sadasivan: "An Introduction to the ARM Cortex-M3 Processor". http://www.arm.com/files/pdf/IntroToCortex-M3.pdf [Último acceso: 01/08/2013].

[Sau'10] L. Sauvage, O. Meynard, S. Guilley, J-L. Danger: "ElectroMagnetic Attacks Case Studies on Non-Protected and Protected Cryptographic Hardware Accelerators". *In 20109 IEEE EMC Society Symposium*, 2010.

[Scc'09] "The Side Channel Cryptanalysis Lounge – What is already Known?". Disponible en: http://www.emsec.rub.de/research/projects/sclounge/. [Último acceso: 21/01/2013].

[Sch'10] O. Schimmel, P. Duplys, E. Böhl, J. Hayek, W.Rosenstiel: "Correlation Power Analysis in Frequency Domain". *In COSADE 2010*, 2010.

[Schi'05] W. Schindler, K. Lemke, C. Paar: "A Stochastic Model for Differential Side-Channel Cryptanalysis". *In proceedings of CHES 2005*, LNCS 3659, pp. 30-46, Springer-Verlag, 2005.

[Schr'03] K. Schramm, T. Wollinger, C. Paar: "A New Class of Collision-Attack and its Application to DES". *In proceedings of FSE 2003*, LNCS 2887, pp. 206-222, Springer-Verlag, 2003.

[Schr'04] K. Schramm, G. Leander, P. Felke, C. Paar: "A Collision-Attack on AES – Combining Side Channel and Differential Attack". *In proceedings of CHES 2004*, LNCS 3156, pp. 163-175, Springer-Verlag, 2004.

[Schu'04] A. Schuster, E. Owald: "Differential Power Analysis of an AES Implementation". *SCA-Lab Technical Report Series*, TR 2004/06/25, 2004.

[Sha'04] A. Shamir, E. Tramer: "Acoustic cryptanalysis on nosy people and noisy machines". *Eurocrypt 2004 rump session*, 2004.

[Ska'11] S. Skalicky, B. Cui: "AES Side Channel Attacks". *Rochester Department of Computer Science, Institute of Technology*, 2011. http://www.cs.rit.edu/~hpb /Lectures/20112 /S_T/Src/36/AES_sidechannel_IEEE2.pdf [Último acceso: 27/02/13].

[SL'06] Silicon Laboratories Inc.: "USB Debugger Adapter" http://www.silabs.com/products/mcu/Pages/USBDebug.aspx [Último acceso: 05/07/2012].

[SL'07] Silicon Laboratories Inc.: Advanced Encryption Standard (AN324SW) v1.0, 2007. http://ww.silabs.com/Support%20Documents /Software/an324sw.zip [Último acceso: 17/06/2013].

[SL'08] Silicon Laboratories Inc.: "C8051F300/1/2/3/4/5 - Mixed Signal ISP Flash MCU Family". Rev. 2.9 12/08, 2008. www.silabs.com/Support%20Documents /TechnicalDocs/C8051F30x.pdf [Último acceso: 07/05/2012].

[SL'09] Silicon Laboratories Inc.: "Silicon Labs IDE" http://www.silabs.com /products/mcu/pages/siliconlaboratorieside.aspx [Último acceso: 05/07/2012].

[Smi'98] D.C. Smith: "Signal and Noise Measurement Techniques Using Magnetic Field Probes". *IEEE*, 1998. [Último acceso: 27/02/13].

[Smi'99] D.C. Smith: "Using a Paper Clip to Measure Signals and Noise". www.emcesd.com/tt080699.htm. [Último acceso: 27/02/13].

[Smi'00] D.C. Smith: "An easy to build shielded magnetic loop probe". http://www.emcesd.com/tt120100.htm. [Último acceso: 27/02/13].

[Sou'10] Y. Souissi, J.-L. Danger, S. Mekki, S. Guilley, M. Nassar: "Techniques for Electromagnetic Attacks Enhancement". *In proceedings of DTIS 2010*, pp. 1-6, 2010.

[ST'01] ST Microelectronics: "M74HC14 HEX SCHMITT INVERTER". http://www.st.com/web/catalog/sense_power_FM140/SC1799/PF69690 [Último acceso: 17/05/2013].

[ST'11] ST Microelectronics: "RM0038 Reference manual: STM32L151xx and STM32L152xx advanced ARM-based 32-bit MCUs". Doc ID 15965 Rev. 4 - February 2011. www.st.com/st-web-ui/static/active/cn/resource/technical /document/reference_manual/CD00240193.pdf [Último acceso: 07/05/2012].

[Sta'04a] F.–X. Standaert, S.B. Ors, B. Preneel: "Power analysis attacks against FPGA implementations of the DES". *In proceedings of FPL 2004*, LNCS 3203, pp. 84-94, Springer-Verlag, 2004.

[Sta'04b] F.–X. Standaert, S.B. Ors, B. Preneel: "Power Analysis of an FPGA – Implementations of Rijndael: Is Pipelining a DPA Countermeasure?". *In proceedings of CHES 2004*, LNCS 3156, pp. 30-44, Springer-Verlag, 2004.

[Sta'06] F.–X. Standaert, E. Peeters, F. Macé, J.-J. Quisquater: "Updates on the Security of FPGAs Against Power Analysis Attacks". *In proceedings of ARC 2006*, LNCS 3985, pp. 335-346, Springer-Verlag, 2006.

[Sta'09a] F.–X. Standaert: "Introduction to Side-Channel Attacks". *In Secure Integrated Circuits and Systems*, pp. 27–44, Springer, 2009.

[Sta'09b] F.-X. Standaert, T.G. Malkin, M. Yung: "A Unified Framework for the Analysis of Side-Channel Key Recovery Attacks". *International Association of Cryptographic Research*, Cryptology ePrint Archive, Report 2006/139, 2006.

[Sze'07] G. J. Székely, M. L. Rizzo, N. K. Bakirov: "Measuring and testing dependence by correlation of distances". *Annals of Statistics*, Vol. 35-6, 2007.

[Tek'03] Tektronix: "Manual del usuario de los osciloscopios de las series DPO4000 y MSO4000". http://www2.tek.com/cmswpt/madownload.lotr?ct=MA&cs=mur&ci=16601&lc=ES.

[Tek'09] Tektronix: "Tekvisa connectivity software" http://www.tek.com/oscilloscope/tds7054-software/tekvisa-connectivity-software-v400 [Último acceso: 07/05/13].

[Tiu'05] C.C. Tiu: "A new Frequency-Based Channel Attack for Embedded Systems". Master Thesis, Department of Electrical and Computer Engineering, University of Waterloo, Ontario, Canada, 2005.

[Tor'98] C. Torrence, G. P. Compo: "A Practical Guide to Wavelet Analysis". *Bulletin of the American Meteorological Society*, Vol. 79, pp. 61-78, 1998.

[UNE'07] UNE: "UNE-EN 61000-4-3 Ensayos de inmunidad a los campos electromagnéticos, radiados y de radiofrecuencia" http://www.aenor.es/aenor/normas/normas/fichanorma.asp?tipo=N&codigo=N0046990&pdf= [Último acceso: 07/05/13].

[Vua'09] M. Vuagnoux, S. Pasini: "Compromising Electromagnetic Emanations of Wired and Wireless Keyboards". *In proceedings of USENIX Security Symposium 2009*, 2009.

[Wad'04] J. Waddle, D. Wagner: "Towards Efficient Second-Order Power Analysis". *In proceedings of CHES 2004*, LNCS 3156, pp. 1-15, Springer-Verlag, 2004.

[WikAES'13] Enciclopedia libre Wikipedia: "Advanced Encryption Standard". http://en.wikipedia.org/wiki/Advanced_Encryption_Standard [Último acceso: 20/02/13].

[Ye'12] X. Ye, T. Eisenbarth: "Wide Collisions In Practice". *In Applied Cryptography and Network Security*, LNCS 7341, pp. 329-343, Springer-Verlag, 2012.

[Yiu'10] J. Yiu: "The Definitive Guide to the ARM Cortex-M3 Second Edition". *Elsevier Inc.*, 2010. ISBN: 978-1-85617-963-8.

[Yoo'04] H. Yoo, C. Kim, J. Ha, S. Moon, I. Park: "Side Channel Cryptanalysis on SEED". *In proceedings of WISA 2004*, LNCS 3325, pp. 411-424, Springer-Verlag, 2004.

[Zho'05] Y. Zhou, D. Feng: "Side-Channel Attacks: Ten Years after its publication and the impacts on Cryptographic Module Security Testing". *Cryptology ePrint Archive*, Report 2005/388. 2005.