KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ
ANABİLİM DALI

DOKTORA TEZİ

RSA ALGORİTMASI GERÇEKLEMELERİ İÇİN GÜÇ TÜKETİMİ SALDIRI YÖNTEMLERİ

EBRU AKALP KUZU

KOCAELİ 2021

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ELEKTRONİK VE HABERLEŞME MÜHENDİSLİĞİ
ANABİLİM DALI

DOKTORA TEZİ

RSA ALGORİTMASI GERÇEKLEMELERİ İÇİN GÜÇ TÜKETİMİ SALDIRI YÖNTEMLERİ

EBRU AKALP KUZU

Prof. Dr. Ali TANGEL **Danışman, Kocaeli Üniv.**	……
Prof. Dr. Sıddıka Berna ÖRS YALÇIN **Jüri Üyesi, İTÜ**	……
Dr. Öğr. Üyesi Anıl ÇELEBİ **Jüri Üyesi, Kocaeli Üniv.**	……
Dr. Öğr. Üyesi Engin AFACAN **Jüri Üyesi, Kocaeli Üniv.**	……
Dr. Öğr. Üyesi Erdinç ÖZTÜRK **Jüri Üyesi, Sabancı Üniv.**	……

Tezin Savunulduğu Tarih: 02.03.2021

ÖNSÖZ VE TEŞEKKÜR

Bu çalışmaya başlamamda ve sürdürmemde desteklerini esirgemeyen Sevgili eşim Ahmet Kuzu ve yaşam kaynağım canım oğullarım, Aybars Kuzu ve Timuçin Kuzu sizleri çok seviyorum. Rahmetli babam Şerafettin Akalp ve rahmetli ablam Sibel Akalp Selvi'yi sevgi, saygı ve özlemle anıyorum. Tüm yaşamım ve uzun süreli eğitim hayatım boyunca destek, sevgi ve şefkatini esirgemeyen, fedakâr annem Şükran Akalp, sevgili ablam Sevilay Akalp Özmen ve sevgili abim Selahattin Akalp, sizleri çok seviyorum, saygı ve şükranlarımı sunuyorum. Doktora çalışmalarım boyunca destek ve sevgilerini esirgemeyen, aile büyüklerim kayın pederim Mustafa Kuzu ve kayın validem Aliye Kuzu, sizleri çok seviyorum, saygı ve şükranlarımı sunuyorum.

Doktora eğitimim boyunca sabrı ve titizliği ile desteğini hep hissettiğim değerli danışmanım Prof. Dr. Ali Tangel'e sevgi, saygı ve şükranlarımı sunuyorum. Bana verdiğiniz tüm emekler için çok sağ olun.

Doktora tezime katkılarından dolayı değerli tez jürisi hocalarım Prof. Dr. Sıddıka Berna Örs Yalçın ve Dr. Öğr. Üyesi Anıl Çelebi'ye sevgi, saygı ve şükranlarımı sunuyorum. Bana verdiğiniz tüm emekler için çok sağ olun.

Yan kanal analizi konusunda çalışma ortamımı sağlayan, çalışmalarıma destek olan, her zaman evim gibi hissettiğim iş yerim TÜBİTAK/BİLGEM'e çok teşekkür ediyorum.

Mart - 2021 Ebru AKALP KUZU

İÇİNDEKİLER

ŞEKİLLER DİZİNİ

TABLOLAR DİZİNİ

SİMGELER VE KISALTMALAR DİZİNİ

Kısaltmalar

ASIC:	: Application Specific Integrated Circuit
BGA:	: Basit Güç Analizi
SPA	: Simple Power Analysis
CMOS:	: Complementary metal-oxide-semiconductor logic
ÇİA:	: Çapraz İlinti Analizi
FÇİA:	: Frekans Çapraz İlinti Analizi
FGA	: Farksal Güç Analizi
DPA	: Differential Power Analysis
EM:	: Elektro Manyetik
EEŞ:	: Eliptik Eğri Şifreleme
ECC	: Elliptic Curve Cryptography
FFT:	: Fast Fourier Transform
DFT:	: Discrete Fourier Transform
FGA:	: Farksal Güç Analizi
FPGA:	: Field Programable Gate Array
HD:	: Hamming Distance
HW:	: Hamming Weight
İGA:	: İlintisel Güç Analizi
MontMul	: Montgomery Multiplication
ML	: Montgomery Ladder
RSA:	: Rivest Shamir Adelman
DES	: Data Encryption
SCA:	: Side Channel Anakysis
SPA:	: Simple Power Analysis
PMOS	: P-type metal-oxide-semiconductor logic
NMOS	: N-type metal-oxide-semiconductor logic
SEMD:	: Single-exponent-multiple-data
MESD:	: Multiple-exponent single-data
VCD:	: Value Change Dump
YKA:	: Yan Kanal Analizi
ZEMD:	: Zero-exponent multiple-data

RSA ALGORİTMA GERÇEKLEMELERİ İÇİN GÜÇ TÜKETİMİ SALDIRI YÖNTEMLERİ

ÖZET

Yan Kanal Analizi (YKA), bir şifreleme gerçeklemesinin doğası gereği sahip olduğu, işlem süresi, güç tüketimi, elektromanyetik yayınımı, ısı ve ses yayınımı gibi istemsiz giriş çıkışlarının kullanılarak anahtar değeri hakkında bilgi edinilmesini sağlayan bir analiz yöntemidir. Güç analizi, hedef şifreleme gerçeklemesinin standart çalışması sırasında ortaya çıkan güç tüketiminin analizine dayanmaktadır. Basit Güç Analizi (BGA), Farksal Güç Analizi (FGA), İlintisel güç Analizi (İGA) ve Çakışma Analizi (ÇA) gibi türleri olan güç analizinin temelinde sayısal bir devrede oluşan anlık güç tüketiminin, o anda işlenen veri ve gerçekleşen işlem adımına bağımlı olması yatmaktadır. RSA (Rvest-Shamir-Adelman) algoritması pratik olarak kullanılmış olan ilk nesil asimetrik kriptolojik algoritma olup, açık anahtar sistemlerinde, özel anahtar dağıtımı, imzalama ve imza doğrulama gibi amaçlarla kullanılmaktadır. Günümüzde geliştirilen şifreleme cihazları, FIPS-140, ISO/IEC 19790 gibi, kriptolojik modüllerin güvenlik analizine yönelik uluslararası standartlar kapsamında test edilmektedir. Bu tez çalışmasının amacı, şifreleme cihazlarında bulunan RSA gerçeklemelerinin güç analizine yönelik özgün yöntemler geliştirerek ilgili literatüre ve dolayısıyla güvenlik test süreçlerine katkı sağlamaktır. Bu doğrultuda tez sürecinde, yatayda çapraz ilinti analizi tabanlı iki yöntem ve hem yatay hem dikeyde eğrileri işleyen ilintisel güç analizi tabanlı iki yöntem literatüre kazandırılmıştır. Uygulamalarda saldırı hedefi olarak hem hazır bir ASIC devresi, hem de FPGA tabanlı RSA devreleri kullanılmıştır. Geliştirilen analizler gerçek güç tüketim eğrilerinin yanı sıra FPGA devresine ait benzetim eğrilerine de uygulanmıştır. Benzetim eğrileri ile çalışılması, tasarlanacak şifreleme modüllerinin güvenlik açıklarının önceden tespit edilebilmesi açısından önemlidir. Gerçek güç tüketim eğrilerini kaydetmek için osiloskop ve hedef cihazlarla haberleşen uygulama programları geliştirilmiştir. Benzetim güç eğrilerini elde etmek için ise FPGA benzetim yazılımlarının bir çıktısı olan ve devrede anlık sinyal değişim bilgilerini içeren davranışsal seviye ve yerleştirme-bağlama (post place route) seviyesindeki VCD (Value Change Dump) dosyalarını ayıklayan yazılımlar geliştirilmiştir. Bu tez çalışmasının literatüre sunduğu asıl katkılar ise güç eğrilerini hem yatayda hem dikeyde kullanan İGA yaklaşımı, ilinti eğrilerinin sadece tepe değerlerini değil birbirleri ile olan ilişkisini kullanan şablon İGA yaklaşımı, yataydaki çapraz ilinti değerlerini birden çok eğride ilintiler toplamı ve sayaç yöntemleri ile birleştiren çapraz ilinti yaklaşımı, saldırıda aynı anda işlem gören anahtar bitlerinin alt işlemleri yerine birden fazla anahtar bitine ait alanın kullanılarak böylece aynı anda işlem görmeyen birden çok bitin elde edilebilmesi yaklaşımı şeklinde sıralanabilir.

Anahtar Kelimeler: Çakışma Analizi, Çapraz İlinti Analizi, İlintisel Güç Analizi, RSA, Yan Kanal Analizi.

POWER CONSUMPTION ATTACK METHODS FOR RSA ALGORITHM IMPLEMENTATIONS

ABSTRACT

Side Channel Analysis (SCA) is a method that, by the nature of an encryption device, allows you to obtain information about the key value using unvanted input outputs such as processing time, power consumption, electronmagnetic emission, heat and sound emission. Power analysis is based on the analysis of power consumption that occurs during the standard operation of target encryption implementation. At the heart of power analysis, such as Simple Power analysis (SPA), Differential Power Analysis (FGA), Correlation Power Analysis (CPA), and Collision Analysis (CA), is the fact that instantaneous power consumption in a digital circuit depends on the data we are currently processing and the process step that is taking place. RSA (Rvest-Shamir-Adelman) algorithm is the first generation asymmetric cryptological algorithm that has been used practically and is used in public key systems for purposes such as private key distribution, signing and signature verification. Encryption devices are tested under international standards for security analysis of cryptological modules, such as FIPS-140, ISO/IEC 19790. The aim of this thesis is to contribute to the relevant literature and therefore security testing processes by developing original methods for power analysis of RSA implementations in encryption devices. In this direction, during the thesis process, two methods based on horizontal cross-correlation analysis and two methods based on correlation power analysis that process curves in both horizontal and vertical direction were introduced to the literature. Both a ready-made ASIC circuit and FPGA-based RSA circuits were used as attack targets in applications. The developed analyses were applied to actual power consumption curves as well as to simulation curves of FPGA circuit. Working with simulation curves is important in terms of detecting vulnerabilities of the encryption modules to be designed in advance. Application programs that communicate with oscilloscopes and target devices have been developed to record actual power consumption curves. In order to obtain simulation power curves, software has been developed that extracts VCD (Value Change Dump) files at the behavioral level and Post place route level, which is an output of FPGA simulation software and contains instantaneous signal change information in the circuit. The main contributions of this thesis to the literature are the IGA approach which uses power curves both horizontally and vertically, the template IGA approach which uses not only the peak values of the correlation curves, but also the relationship between each other, the cross-correlation approach which combines horizontal cross-correlation values in multiple curves, uses of trace field belonging to multiple key bits instead of sub operations of key bits that are not treated at the same time.

Keywords: Collision Analysis, Cross Correlation Analysis, Correlation Power Analysis, RSA, Side Channel Analysis.

GİRİŞ

Günümüzde, sağlık, bankacılık ve askeri sistemlerden günlük hayata kadar her alanda sayısal elektronik cihazlar kullanılmakta olup, bu cihazlarda önemli miktarda veri işlenmektedir. İşlenen verinin güvenliğinin sağlanması ise her zaman önemli bir konu olmuştur. Sayısal bir cihazda işlenen verinin güvenliğinin sağlanması, içerdiği bilginin gizliliğinin, bütünlüğünün korunması, doğru kaynaktan geldiğinin kanıtlanması ve taklit edilmezliğinin garantilenmesi anlamlarına gelmektedir. Bilgi güvenliğinin bu dört ayağını sağlamak amacıyla, veri işleyen cihazlarda ve sistemlerde kriptolojik algoritma ve protokoller kullanılmaktadır. "Kerckhoffs" ilkeleri [1] gereği, şifreleme algoritmalarında, algoritmanın matematiksel akışı bellidir, ya da gizli olmasının bir önemi yoktur ancak algoritmaya bir parametre olarak verilen "gizli anahtar" değerlerinin korunması gerekir. Şifreleme yapan sistemler, tek çıkışı şifreli veri olan bir kara kutu olarak düşünüldüğünde, kara kutunun dış dünyayla olan bağlantıları sadece giriş çıkış kanallarından ibaret olmaktadır. Bu kara kutunun, kurcalama tipi çeşitli fiziksel saldırılara ve giriş çıkış ara yüzlerinden gelecek çeşitli mantıksal saldırılara karşı dayanıklı bir yapıya sahip olması, anahtar değerini doğrudan ya da dolaylı şekilde dış dünyaya vermemesi gerekir. Bu kabuller ışığında kripto analiz disiplini, şifreleme algoritmalarının, standart giriş çıkışlarını kullanarak, matematiksel gücünü ölçmeye çalışır. Eğer matematiksel olarak güçlü bir algoritmanın, anahtar boyu deneme yanılma yöntemini elverişsiz yapacak kadar uzun seçilmişse ve uygun kullanım koşullarına göre kullanılmışsa kriptolojik açısından güvenli kabul edilmektedir.

Son yıllarda ortaya çıkan ve Yan Kanal Analizi (YKA) olarak adlandırılan saldırı türü, güvenlik açısından, şifreleme algoritmalarının matematiksel gücünün yanı sıra, nasıl gerekçelendiğinin de önemli olduğunu, aksi halde kara kutu sistemlerden dış dünyaya sanılandan çok daha fazla sistem bilgisi sızabileceğini ortaya koymuştur. Sayısal devreler, doğası gereği, üzerinde koşan algoritmalarla ilişkili şekilde zaman harcar, güç tüketir, ısınır, ses çıkarır veya elektromanyetik yayınım yaparlar. YKA türü saldırılar, algoritmaların standart giriş çıkışlarının yanı sıra, koşturumu sırasında

oluşan ve yan kanal bilgisi olarak isimlendirilen bu fiziksel değişimleri de analiz etmektedir. Hedef sistemlerden ölçülen yan kanal değişimleri, algoritma adımlarında ortaya çıkan ara değerler hakkında da bilgi verdiğinden, yan kanal analizi ile klasik kripto analizin ötesine geçilmekte, matematiksel açıdan güçlü olan şifreleme algoritmalarının anahtarları bile deneme yanılma yönteminden çok daha hızlı bir şekilde parçalar halinde elde edilebilmektedir. YKA türü saldırılar, hedef algoritmanın ne şekilde gerçeklendiği ile ilgilendiğinden "gerçekleme saldırısı" olarak nitelendirilmekte ve hedef sistemlerin fiziksel değişimlerini kullandığı için de "fiziksel saldırılar" sınıfına girmektedir.

YKA saldırıları bilim dünyasına Kocher ve arkadaşları tarafından tanıtılmıştır [2]. YKA saldırılarının etken ve edilgen olmak üzere iki temel çeşidi bulunur. Edilgen YKA türleri, Zaman Analizi [3], Güç Analizi [3 - 6], Elektromanyetik Yayınım Analizi [8 - 9], Ses Tabanlı Analiz [10] olarak adlandırılan ve yan kanal çıkışlarının gözlenmesine dayanan yöntemleri kapsar. Algoritma işleyişi sırasında devrenin güç ve saat hattına temaslı/temassız olarak çentik ve sinüs dalgaları uygulayarak, devre yüzeyine EM, lazer ya da iyon vs. atışları yaparak algoritma ara adımlarında hatalar oluşturan ve bu hatalı çıktıları analiz eden Hata Analizi yöntemleri [11 - 12] ise etken YKA sınıfını oluşturur.

RSA (Rvest-Shamir-Adelman) algoritması [13] pratik olarak kullanılmış olan ilk nesil asimetrik kriptolojik algoritma olup, açık anahtar sistemlerinde, özel anahtar dağıtımı, imzalama ve imza doğrulama gibi amaçlarla kullanılmaktadır. YKA saldırılarının gerçeklenebilir olduğu ortamlarda çalışan şifreleme modüllerinde, bu algoritmanın güç analizi açısından güvenliğinin sağlanması oldukça önemlidir. Günümüzde geliştirilen şifreleme cihazları, FIPS-140-3 [14] , ISO/IEC 19790:2012 [15] gibi, kriptolojik modüllerin güvenlik analizine yönelik uluslararası standartlar kapsamında test edilmektedir. Bu süreçlerde YKA saldırılarına karşı dayanıklılık testleri de önemli bir yer tutmaktadır. Bu tez çalışmasının amacı, şifreleme cihazlarında bulunan RSA gerçeklemelerinin güç analizine yönelik özgün yöntemler geliştirerek ilgili literatüre ve dolayısıyla güvenlik test süreçlerine katkı sağlamaktır. Bu doğrultuda tez kapsamında, yatayda çapraz ilinti analizi tabanlı iki yöntem [16 - 17] ve hem yatay hem dikeyde eğrileri işleyen ilintisel güç analizi tabanlı [18 , 19] iki yöntem literatüre kazandırılmıştır. Güç analizi çalışmalarında klasik dikey analiz yöntemlerinin yanı sıra

son yıllarda güç eğrisi alanlarını yatayda da işleyebilen, böylece eğrileri daha etkin bir şekilde kullanabilen yöntemler ön plana çıkmaktadır. Tez kapsamında geliştirilen çalışmalar, güç eğrilerini hem yatayda hem dikeyde kullanabiliyor olmaları, birden çok eğride yatayda gerçekleştirilen analiz sonuçlarını birleştirme yaklaşımları, aynı anda işlem görmeyen birden fazla anahtar bitini elde ediyor olmaları gibi açılardan literatüre özgün yaklaşımlar sunmuştur. Bunun yanı sıra YKA testlerinde kullanılan güç analizi yöntemlerinin, hedef şifreleyiciden elde edilebilecek anahtar bitlerinin oranını artırması, saldırıyı daha az eğri ile gerçekleştirilebilir hale getirmesi, gereken işlem gücünde iyileşme sağlaması, saldırıyı daha az gerçekleme ayrıntısına gereksinim duyar ve daha uygulanabilir hale getirmesi önemlidir. Geliştirilen her bir yöntemde bu alanların en az birinde ilerleme sağlanması hedeflenmiştir. Bunun yanı sıra, kriptolojik modüllere yönelik güvenlik testi süreçlerinde, güç analizi yöntemlerinin kaç eğri kullanarak başarıma ulaşılabileceğinin kestirilebilmesi de önemlidir. Bu bağlamda, geliştirilmiş olan yöntemlerin kaç eğri ile anahtar bitlerini elde edebileceğinin yani gerekli eğri sayısının kestirimine yönelik istatistiksel modeller oluşturulmuştur.

Geliştirilen bu yöntemler hazır ASIC RSA devresi ve tez çalışmaları amacıyla geliştirilen FPGA tabanlı RSA devresinden toplanan gerçek güç eğrilerine uygulanmıştır. Yöntemlerin bir kısmı FPGA tabanlı devrenin VCD dosyalarının ayıklanması ile elde edilen benzetim eğrilerine de uygulanmıştır. Benzetim eğrileri ile çalışılması, henüz donanımsal olarak ortaya çıkmamış cihazlarda konacak şifreleme modüllerinin güvenlik açıklarının önceden tespit edilebilmesi ve tasarım-gerçekleme-test süreçlerinin hızlanması açısından önemlidir.

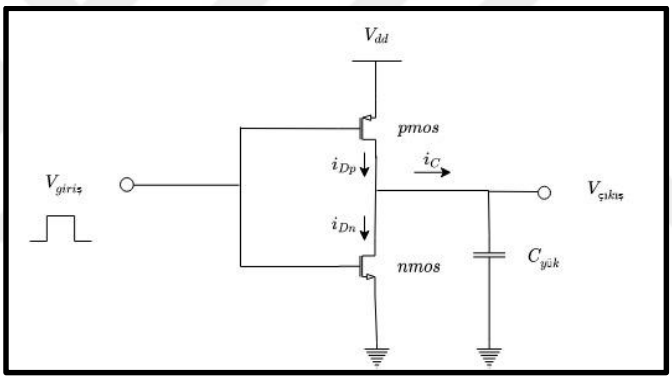
Tez çalışması şu bölümlerden oluşmaktadır: Bölüm 1’ de, kriptolojik devrelerde güç analizine yönelik temel bilgilerin verilmesi hedeflenmiştir. Bu bölümde öncelikle CMOS devrelerdeki güç kaçağının kaynağı, güç tüketiminin ölçülmesi ve istatistiksel analizine yönelik temel bilgiler verilmiş, sonrasında ise bu güç kaçaklarını kullanan YKA türleri tanıtılmıştır. Yine aynı bölümde RSA algoritması tanıtılmış ve literatürdeki RSA üs alma adımlarına uygulanabilecek türdeki analizi yöntemleri incelenmiştir. Bölüm 2’de ise öncelikle saldırılarda kullanılmak üzere tez sürecinde geliştirilmiş ya da hazır olarak kullanılmış RSA devrelerinin özellikleri tanıtılmıştır. Daha sonra saldırılarda kullanılan bir kısmı hazır bir kısmı tez sürecinde geliştirilmiş yazılım ve düzenekler tanıtılmıştır. Bölüm 3’de ise geliştirilen özgün güç analizi

yöntemleri tanıtılarak yöntemlerin deneysel sonuçları gösterilmiştir. Son bölümde ise sonuçlar özetlenmiştir.

1. GÜÇ ANALİZİ

1.1. Sayısal Devrelerde Güç Tüketimi

Sayısal devreler çoğunlukla CMOS teknolojisine dayanır. CMOS devrelerde güç tüketimi, devrenin içerdiği bütün lojik kapılara ait güç tüketimleri toplamından oluşur. CMOS devrelerin güç tüketim davranışı, en küçük CMOS hücre olan eviricilerin çalışma mantığı ile anlaşılabilir [20 - 21]CMOS devrelerde, durağan durum (Static Power consumption) ve geçici durum (Transient Power Consumption) olarak adlandırılan iki çeşit güç tüketimi bulunmaktadır.



Şekil 1.1. CMOS evirici lumped-C modeli

Durağan durumda, yani CMOS hücresindeki sinyaller konum değiştirmiyorken, "PMOS" ve "NMOS" türü transistorlar hiçbir zaman aynı anda iletime geçmez. Bu nedenle durağan durumdaki güç tüketimi yaklaşık 0'dır. Geçici (dinamik) durum güç tüketimi ise ancak hücre iç sinyallerinde, ya da hücre çıkışında durum değişikliği olduğu zaman gerçekleşen anlık güç tüketimidir.

Geçici durumda çekilen akım değeri, "dolma akımı (kapasite akım)" ve "doğrudan yol akımı" (bazı kaynaklarda kısa devre akımı olarak tanımlanmaktadır) olmak üzere iki bileşenden oluşur. Eviricilerin güç tüketimini basitleştirmek amaçlı kullanılan lumped-C modeli Şekil 1.1.' de görülmektedir. Bu model kapasite kaynaklı akım karakterini anlamak için oldukça uygundur. Bu modelde, bir hücrenin iç ve dış kapasitelerinin toplamı $C_{yük}$ olarak adlandırılan toplamsal bir çıkış kapasitesi ile temsil edilir. Burada

içsel kapasiteler, hücrenin çıkışına bağlı iç kapasiteleridir. Dışsal kapasiteler ise hücrenin çıkışına bağlı olan elektrik hatları ve sonraki katta bulunan lojik hücrelerin giriş kapasitesinden oluşur. Kapasite akımı, evirici çıkışı 0-1 geçişi yaparken $C_{yük}$ kapasitesinin VDD kaynağından çekip yüklendiği akımdır. Evirici çıkışı 1-0 geçişi yaparken $C_{yük}$kapasitesi toprağa doğru yükünü kaybeder. Bu sırada, devredeki besleme kaynağı VDD'den akım çekmez. Bir lojik hücrede, T süresi boyunca kapasite akım bileşeninden kaynaklanan ortalama güç tüketimi (1.1) eşitliğindeki gibi hesaplanabilir [20 - 21] Burada "f" devrenin saat frekansı, α ise her bir saat vurusunda hücre çıkışında oluşan değişim sayısını gösteren "etkinlik katsayısıdır". Örneğin bir lojik hücre her saat vurusunda çıkış bir kez değişiyorsa $\alpha = 1$ olacaktır.

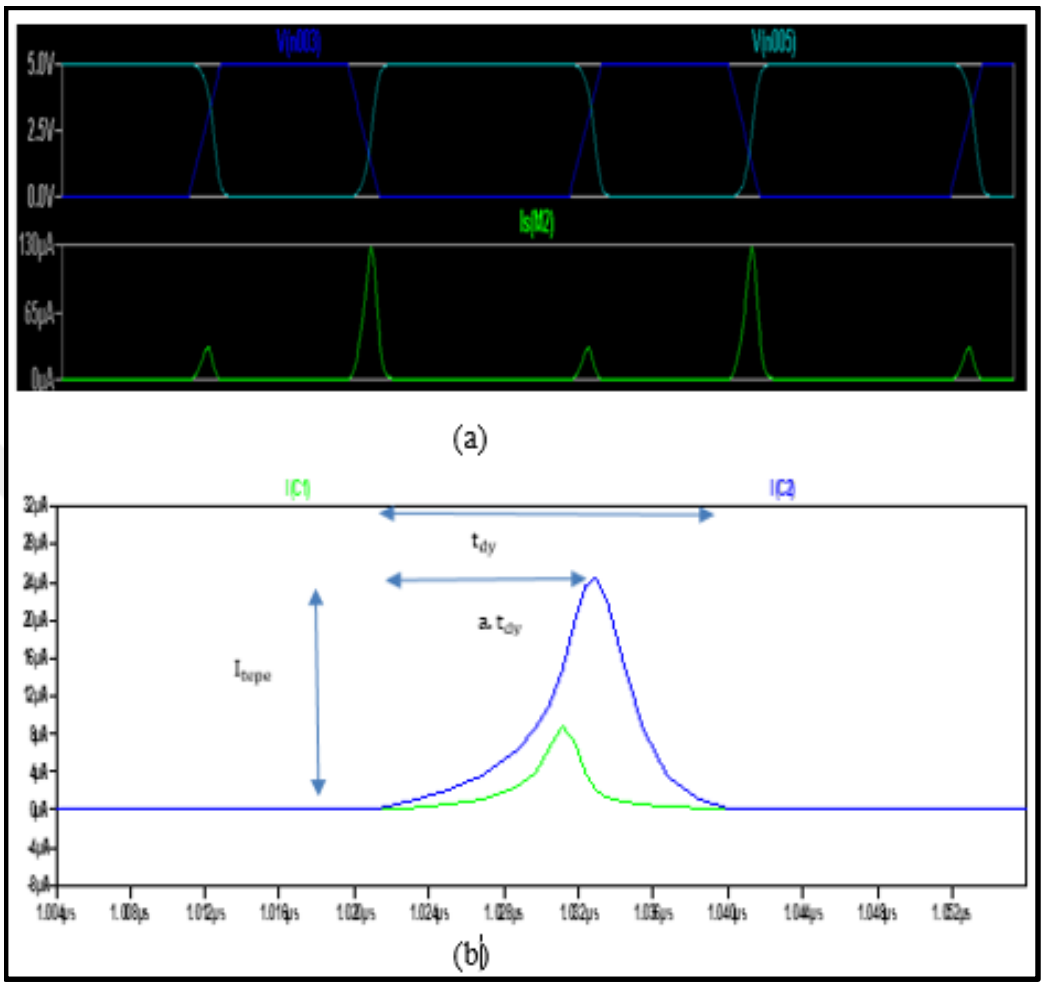
$$P_{kap} = \frac{1}{T}\int_0^T p_{kap}(t)\,dt\alpha * f * C_L V_{DD}^2 \tag{1.1}$$

Doğrudan yol akımı ise çıkış kapısı durum değiştirirken, P ve N transistorlarının ikisinin birden iletimde olduğu ve VDD'den toprağa doğru gerçekleşen geçiş durumunda oluşan akımıdır. Bazı kaynaklarda kısa devre akımı olarak da adlandırılmaktadır [20]. Lumped-C modeli ile modellenmiş bir eviricinin durum değiştirmesi sırasında oluşturduğu toplam doğrudan yol akımı, yaklaşık olarak genliği "I_{tepe}", taban genişliği "t_{dy}" , I_{tepe} değerine erişim süresi ise $a.t_{dy}$olan bir üçgen darbe şeklindedir (Şekil 1.2). T süresi boyunca gerçekleşen ortalama doğrudan yol güç tüketimi P_{dy} 'yi hesaplamak için, bu üçgen dalga biçiminin alanını kullanmak yaygın olarak başvurulan bir yöntemdir [21]

$$P_{dy} = \frac{1}{T}\int_0^T p_{dy}(t)\,dt = \alpha * f * I_{tepe} * V_{DD}^2 * t_{dy} \tag{1.2}$$

Şekil 1.2'de , "lumped-C" ile modellenmiş bir evirici için, (a) giriş çıkış sinyalinin lojik değişimi ile (b) farklı $C_{yük}$ (yük kapasitesi) değerleri ile çekilen akımın değişimini gösteren "SPICE" benzetimi görülmektedir. Şekilden anlaşılacağı gibi çıkışta gerçekleşen 0-1 değişimi sırasında, 1-0 değişimine göre VDD'den daha fazla akım çekilmektedir. Bu durum dinamik güç tüketiminin kapasite bileşeninden kaynaklanmaktadır. Çıkışta 0-1 değişimi olurken $C_{yük}$kapasitesi VDD'den çektiği

akımla dolup, diğer durumda VDD' den akım çekmeyip depoladığı yükü toprağa boşaltmaktadır.



Şekil 1.2. Evirici lumped-C modeli güç tüketim benzetimi a) giriş çıkış değişimi b) Doğrudan yol akımının kapasite ile değişimi

Kombinezonsal devrelerde peş peşe bağlı lojik kapılarda, kapı girişleri hücreye aynı anda erişemezse, kapı çıkışında ara durumlar oluşur. Bu değişimler çentik (gliç) olarak adlandırılır. Sayısal devrelerdeki bu çentiklerin sayısı, işledikleri veri ile bağlantılı olup devrenin dinamik güç tüketiminde oldukça büyük bir etkiye sahiptir.

1.1.1. Anlık güç tüketimi benzetimi

YKA açısından da sayısal devrelerin modelleme ve benzetim yolu ile anlık güç tüketim davranışlarının kestirilebilmesi oldukça önemlidir. Sayısal devre tasarımcılarının kullandıkları güç tüketimi modellemesi ve benzetimi, analog, lojik ve davranışsal olmak üzere üç seviyeye ayrılır [21].

“Analog seviye benzetim, gerçek güç tüketimine en yakın değerleri veren seviyedir. Analog benzetim, devredeki tüm transistorları, bağlantıları ve devrenin parazitlik elemanlarını da gösteren “bağlantı listesi” (netlist) üzerinden çalışır. Parazit devre elemanları, transistorlar arası elektriksel bağlantılardan ve transistorların iç parazit kapasitelerinden kaynaklanır. Devredeki parazit kapasitelerin sayısı oldukça fazla olabilir ve bu durum benzetimin karmaşıklığını artırır. Bu karmaşıklığı azaltmak üzere birtakım yaklaşımlar kullanılmaktadır. Örneğin daha önce bahsedilen ve bir hücrenin tüm parazit kapasitelerini tek bir çıkış kapasitesi olarak modelleyen “lumped-C” modeli bunlardan biridir. Sonuç olarak parazit bileşenleri de göz önünde tutarak oluşturulan bağlantı listesi (netlist), fark denklemleri kullanılarak çözülür ve devrede oluşan akım ve gerilim değerleri hesaplanır.

Lojik seviye benzetim, kesinlik açısından analog seviyeden sonra gelir. Bu seviye benzetim, kapı seviyesindeki “bağlantı listesi” üzerinden çalışır. Bu listeler kapı gecikmelerini ve kapı çıkışlarının yükselme ve düşme zamanlarını da (back annotation) içerebilir. Lojik benzetim iki aşamadan oluşur: Önce devredeki kapı geçişleri benzetime sokulur. Sonra bu geçişlerle güç tüketim eşleştirmesi yapılır. Burada her bir hücre için, sinyal değişimi olduğu zaman nasıl bir güç tüketiminin gerçekleşeceği hesaplanırken, çıkışın kapasite yüklenmesi ve çıkış değişiminin zamanlaması yani geçiş süreleri de göz önünde tutulabilir. Genelde standart hücre kütüphanelerinde hücrenin güç tüketim modeli de yer alır ve benzetim araçları bu veriyi kullanır.

Lojik benzetimde standart hücre kütüphane modellerinin yanı sıra, her bir lojik kapının, güç tüketimine eşit seviyede katkı yaptığını varsayan “Hamming Distance” (HD) ve Hamming Weight (HW) modelleri de kullanılmaktadır. HD tabanlı modellemede, her bir hücre için güç tüketiminin basitçe, belli bir anda kapıların çıkışında oluşan 0-1 ya da 1-0 geçişi yani değişim sayısı kadar olduğu varsayılır. HW modelinde ise hücrenin güç tüketiminin, belli bir anda kapıların çakışlarındaki 1’lerin sayısı kadar olduğu varsayılır. İleriki bölümlerde görüleceği gibi YKA saldırılarında çoğunlukla bu basit ama etkili modeller temel alınarak devrelerin teorik güç tüketimi hesaplanmaktadır.

Davranışsal seviye güç tüketim benzetimi ise CMOS devrelerin yüksek seviye güç tüketimi tanımlarına dayanmaktadır. Genelde davranışsal seviye güç benzetimi, devrelerin ortalama güç tüketimini hesaplamak amacıyla kullanılmaktadır. YKA kapsamında sadece devrenin veri ve işlem bağımlı güç modellerini de kullanan davranışsal benzetim yaklaşımları önemlidir.

1.1.2. Anlık güç tüketiminin ölçümü

Bir kriptolojik cihaza güç analizi uygulayabilmek için devrenin şifreleme algoritmasını çalıştırması esnasında harcadığı gücün ölçülmesi gerekmektedir. Tipik bir ölçüm düzeneği hedef şifreleme cihazı, saat üreteci, güç kaynağı, güç ölçüm devresi ya da EM probu, sayısal örnekleme osiloskobu ve bir bilgisayardan oluşur. Saat üreteci ve güç kaynağı şifreleme devresinin dış saat üreteci ve besleme kaynağı ile sürülmesi durumunda gerekmektedir.

Osiloskoplar sadece gerilim ölçebildiği için, bir devrenin güç ya da akım değişimini ölçmek için bu büyüklüklerle orantılı gerilim değişimi oluşturacak düzenekler gerekmektedir. Bu tür bir gerilim sinyali, basitçe güç kaynağı ve devrenin arasına ya da devrenin toprak hattına seri bağlı ve değeri 1-50 ohm arasında değişen bir direnç üzerinden, gerilim probuyla ölçülebilir. Bir diğer yol da devrenin oluşturduğu elektromanyetik alanın, EM probu kullanılarak ölçülmesidir. Bu yöntemlerden ilkinde, besleme kaynağının sabit bir gerilim verdiği düşünülürse, direnç üzerinden ölçülen gerilim düşümü devrenin çektiği akımla, dolayısıyla da güç tüketimi ile orantılı olacaktır. EM alanının ölçümüne dayanan diğer yöntemde ise tüm devrenin yaydığı manyetik alanın ölçümü, belli bir alandaki hücrelerin yaydığı EM alanın ölçümü ve kablolar üzerinden ölçüm alan “Manyetik prob yöntemi ” gibi yaklaşımlar bulunmaktadır. Ayrıca basitçe güç kablosunun bir akım probunun içinden geçirilmesi ile de kablodan geçen akım, dolayısıyla güç tüketimi temassız olarak ölçülebilmektedir.

Devre üzerinden problarla alınan ölçümler, sayısal osiloskopla kaydedilmektedir. Bir osiloskop için önemli parametreler, giriş bant genişliği, örnekleme hızı ve çözünürlüktür. Bir sinyalin bant genişliği, sinyaldeki en düşük ve en yüksek frekans bileşenleri arasındaki fark olarak tanımlanmaktadır. Bir osiloskobun giriş bant genişliği ise osiloskobun bozulma (distortion) yapmadan işleyebildiği en yüksek

frekans bileşenidir. Tipik olarak yan kanal sinyallerinin bant genliği 1 GHz ler civarındadır. Modern osiloskopların bant genişliği bu değerlerin çok üstünde olabilmektedir. Osiloskoplarda bir diğer önemli parametre de örnekleme hızıdır. Örnekleme hızı, belli bir analog sinyalin saniyede kaç örnek alınarak sayısallaştırıldığını gösterir. "Nyquist-Shanon" örnekleme teoremine göre, örnekleme hızının en azından ölçülen en yüksek frekans genişliğinin iki katı değerde olması gerekir. Yan kanal analizinde genelde gürültü yüksek frekans bileşenlerine sahiptir. Ayrıca bir yan kanal sinyalinde, anlamlı en yüksek frekans bileşeni yerine en baskın frekans bileşenin ölçülmesi önemlidir. Osiloskobun çözünürlüğü ise ölçüm kalitesini belirleyen bir diğer önemli parametredir ve her bir örnek noktasının kaç bit ile temsil edilebildiğini gösterir. Osiloskoplar genelde 8 bit çözünürlüğe sahip olmakla birlikte yeni nesil, yüksek bitle örnekleme yapan osiloskoplar mevcuttur. Sonuç olarak 8 bitlik bir osiloskop her bir noktayı $2^8 = 256$ farklı değe ile temsil edebilmektedir. Sonsuz aralıklara bölünebilecek analog bir değerin belli bir çözünürlükte kaydedilmesinden kaynaklanan hata, "kuantalama gürültüsü" olarak adlandırılır. Osiloskop bit sayısı arttıkça kuantalama gürültüsü azalacaktır.

1.1.3. Anlık güç tüketimi ölçüm kalitesi

Kriptolojik cihazların güç sinyalleri, yüksek frekanslı (HF) analog sinyallerdir. Bir kriptolojik cihazdaki lojik hücreler, GHz'ler mertebesinde frekans bileşenleri olan güç tüketimi gerçekleştirirler. Ancak bu sinyaller, oluşturdukları hücreden osiloskoba ulaşana kadar, pek çok gürültü kaynağından etkilenirler. Temel olarak ölçüm düzeneğinden ve devrenin iç etkinliklerinden kaynaklanan bu gürültüler, "elektronik gürültü" ve "anahtarlama (switching) gürültüsü " olarak iki sınıfa ayrılır.

Elektronik gürültü, tüm ölçüm düzeneklerinde bulunur ve bu gürültü türünden kurtulmak mümkün değildir. Devredeki elektronik gürültüden dolayı, aynı girdilerle gerçekleştirilen sabit bir işleme ait her bir ölçüm birbirinden farklı olacaktır. Elektronik gürültünün temel bileşenleri, "güç kaynağı gürültüsü, saat üretecinin gürültüsü, temaslı yayınım gürültüsü (conducted emssion), temassız yayınım gürültüsü ve kuantalama gürültüsü" olarak sıralanabilir.

Anahtarlama gürültüsü, güç ölçümlerinde bulunan ve lojik hücrelerin kendi etkinliklerinden kaynaklanan bir diğer gürültüdür. Bir şifreleme algoritmasının

çalışması sırasında, içerdiği lojik kapıların her birinin çıkışında hızlı değişimler gerçekleşir. Lojik hücrelerde gerçekleşen bu etkinliklerin her biri, yüksek güç tüketimine yol açar. Genel olarak YKA' da, bu değişimlerden kaynaklanan güç tüketimi önemlidir. Ancak ilgilenilen esas güç tüketimi devredeki tüm hücrelere değil sadece belli bir hücre grubuna ait olan değişimlerden kaynaklanmaktadır Bir sayısal devrede, ilgilenilen lojik hücreler dışındakilerin durum değiştirmesinden kaynaklanan güç tüketimi, "anahtarlama gürültüsü" olarak adlandırılır.

Kullanılan YKA ölçüm düzeneklerinde tipik olarak, kriptolojik cihaz ile VDD hattı arasına bağlanan bir devre üzerinden, toplam güç tüketimi ölçülmektedir. Bu tür bir senaryoda, temel olarak ölçümde bulunan anahtarlama gürültüsü, devredeki iki bileşenden etkilenmektedir: Bunlardan ilki kriptolojik cihazdaki lojik hücrelerle osiloskop arasındaki sinyal bant genişliğidir. Normalde lojik hücrede oluşan sinyal bandı GHz'ler seviyesindedir ancak bu bant genişliği, aradaki parazit bileşenlerden dolayı tam olarak osiloskoba ulaşamamaktadır. Bu parazit etkenlerden ilki VDD ve GND hatları arasında bulunan "eşleştirme kapasitelerinden (decaupling capacitances)" kaynaklanır [22 - 23]. Eşleştirme kapasiteleri özellikle yüksek saat frekanslarında çalışan CMOS devreler için yüksek frekanslı akım bileşenlerinin lojik bloklara erişebilmesi amacıyla kullanılır. Bu tür devrelerde VDD hattına paralel olarak bağlı, farklı büklüklerde birden çok eşleştirme kapasitesi bulunur. Güç ölçümünün sadece VDD hattını besleyen kaynak tarafından yapılabildiği durumlarda ise YKA açısından en önemli akım bileşenleri, eşleştirme kapasiteleri üzerinden sağlandığı için ölçüm kalitesi önemli ölçüde azalacaktır. Bu durumun önüne geçmek amacıyla hedef devrelerden eşleştirme kapasitelerinin sökülmesi yönünde yaklaşımlar bulunmakla birlikte [23], güç ölçümünü hedef lojik bloklara en yakın besleme pinine bağlı eşleştirme kapasitesi üzerinden alarak, bu durumu saldırgan lehine kullanan çalışmalar da bulunmaktadır [22]. Bunun yanı sıra, kriptolojik cihazlarda VDD ve GND hatları, giriş/çıkış (G/Ç) hatları üzerinden devreye bağlanır. Bu G/Ç bağlantıları, güç sinyaline parazit bir indüktans eklenmesine neden olur. Sonuç olarak bir şifreleme cihazının pinleri üzerinden ölçülen güç tüketimi, filtrelenmiş bileşenlerden oluşmaktadır. Bu filtrenin bant genişliği ise temel olarak VDD ağına bağlanan eşleştirme kapasiteleri ve indüktans değerleri ile belirlenir. Sonuç olarak YKA çalışmalarında ölçülebilen bant

genişliği, orijinal anahtarlama etkinliklerinden kaynaklanan gerçek güç tüketiminin bant genişliğinden çok daha düşük olabilmektedir.

Güç ölçümlerinde bulunan anahtarlama gürültüsünü etkileyen bir diğer parametre de kullanılan saat frekansıdır. Örneğin bir devrede çok yüksek saat frekansları kullanılırsa, ölçülen güç eğrilerinde komşu saat vuruları arasında girişimler gözlenir. Normalde düşük bir saat frekansı kullanıldığında, ölçümde saatin yükselen kenarıyla tetiklenen hücresel sinyal değişimi ile orantılı tepeler gözlenebilmektedir. Ancak saat frekansı yükseldikçe, komşu saat vuruları birbirini etkilemeye ve bu tepe değerleri kaybolmaya başlar. Normalde bir sayısal tüm devreye, devrenin desteklediği en yüksek saat frekansına ulaşılana kadar saldırı uygulanabilir ancak devrenin çalışma frekansı yükseldikçe, bir saat vurusunun sonrakileri etkilemesinden dolayı çok daha fazla anahtarlama gürültüsü oluşur [21]. Genel olarak bu gürültüyü azaltmak için, düşük saat frekanslarında çalışmak tercih edilir ama bu durum tüm hedef cihazlar için uygulanabilir olmayacaktır.

1.1.4. Anlık güç ölçümlerinin istatistiksel özellikleri

Sayısal CMOS devrelere güç tabanlı yan kanal kaçaklarının oluşmasının temel nedeni, çekilen gücün devrede gerçekleşmekte olan işlem ve işlenmekte olan veriye bağımlılık göstermesidir. Tıpkı [21]'da tanımlandığı gibi, bir saat vurusu T kadar olan bir sayısal devrede, bu T süresi boyunca çekilen ortalama gücün işlem bağımlı parçasını P_{islem}, ve veri bağımlı parçasını P_{veri} olarak adlandıralım. Her bir güç ölçümünde, bu iki etkinin yanı sıra daha önce sözü edilen elektronik gürültü ($P_{el-gür}$) ve sabit bir bileşen de (P_{sabit}) bulunmaktadır. Bu bileşenlerin her biri, zamanın birer fonksiyonudur ve her bir saat vurusunda değişiklik gösterirler.

Güç eğrilerinde, belli bir anda gerçekleşen işlem ve işlenen veri sabit tutulduğunda bile, alınan her ölçümde genlik değerlerinin birbirinden farklı olduğu görülür. Bunun temel nedeni, ölçüme etki eden elektronik gürültüdür. Gerçekleştirilen işlem ve işlenen veri sabit tutularak yapılan ölçümlerde, bir saat vurusu olan T süresi boyunca P_{islem}, P_{veri} ve P_{sabit}'den kaynaklanan ortalama güç değişimi "0" olacaktır. Bu koşullar altında, gerçekleştirilen pek çok ölçüm için, belli bir örnek anına ait değerlerin histogramı, normal dağılıma benzeyecektir. Bu durum, güç ölçümlerinde, her bir

noktaya etki eden elektronik gürültünün, normal dağılıma sahip olduğunun göstergesidir.

Normal dağılım, temel parametreleri ortalama değer "μ" ve varyans değeri" σ^2 "olan aşağıdaki fonksiyonla tanımlanır:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma}\left(\frac{x-\mu}{\sigma}\right)^2} \quad (1.3)$$

Dağılım parametreleri μ ve σ^2 ise aşağıdaki eşitliklerle hesaplanır. Burada E() beklenen değer fonksiyonudur.

$$\mu = E(x) \quad (1.4)$$

$$\sigma^2 = \mathrm{Var}(x) = E((-E(x))^2) \quad (1.5)$$

Bir x değişkeninin normal dağılıma sahip olduğunu belirtmek için x ~N(μ, σ^2) gösterimi kullanılır. Dağılım parametrelerinin μ=0 ve $\sigma^2 = 1$ değerlerine sahip olduğu dağılımlar, standart normal dağılım olarak adlandırılır. Standart normal dağılımın toplamsal dağılım fonksiyonu (cumulative distribution) ise Φ(x) ile gösterilir.

Alınan ölçümlerde bir noktaya ait normal dağılım parametreleri μ ve σ^2'nin kestirimi olan $\bar{x}$ ve s^2 değerleri aşağıdaki eşitliklerle hesaplanır.

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (1.6)$$

$$s^2 = \frac{1}{n-1}\sum_{i=1}^{n} (x_i - \bar{x})^2 \quad (1.7)$$

Hedef noktada ölçülen ortalama değer P_{sabit} bileşeni olarak kabul edileceğinden, güç ölçümlerinde belli bir saat vurusundaki elektriksel gürültü $P_{el-gür}$, parametreleri $N(0, \sigma^2)$ olan normal dağılıma sahip olur. Dağılımın varyansı ise cihazdan cihaza değişkenlik gösterir.

1.1.5. Güç ölçümlerinde veri ve işlem bağımlı dağılım parametreleri

Daha önce de belirtildiği gibi, bir güç eğrisine ait her bir noktada P_{veri} ve P_{islem} olarak adlandırdığımız veri ve işlem bağımlılığından kaynaklanan güç tüketim bileşenleri bulunur. Bir işlem anına ait veri bağımlı güç dağılımını bulmak için, o anda işlenen verinin değiştirilip, işlemin ise sabit tutulduğu pek çok ölçümüm alınması gerekir. İşlemde kullanılacak veri değerleri de eş dağılımlı olmalıdır. Sekiz bitlik bir işlemcinin bir bellek gözünden diğerine veri taşıması sırasında ölçüm alınan böyle bir deneyde [21] veri bağımlı güç tüketiminin karakteristiği gösterilmiştir. Deneyde elektriksel gürültü $P_{el-gür}$'nin varyansı, pek çok ölçümün ortalaması alınarak azaltılmıştır. Bu nedenle deneyde, her bir veriye ait ölçüm 200' er kere tekrarlanmıştır. Bu deneyde işlem anına ait belli bir ölçüm noktanın histogramı çıkarıldığında, sabit veri için yapılan önceki deneye benzerliklerin olduğu gözlenmiştir. Ancak burada tek bir normal dağılım yerine, 9 alt normal dağılım bulunmaktadır. 8 Bitlik bir giriş verisinde, toplamda 9 farklı HW değeri bulunur. Söz konusu deneyde oluşturulan dağılımda da "9" adet alt normal dağılım gözlenmiştir. Yani tüm HW değerlerinin bulunduğu bir ölçüme ait dağılım, her bir HW değerine ait dağılımların üst üste binmesinden (süper position) oluşmaktadır. Her bir alt dağılımdaki ortalama genlik değeri, HW değeri ile orantılı olarak artmaktadır. Dağılımların her birinde elektronik gürültüden kaynaklanan ve yaklaşık olarak aynı değerlere sahip varyans değerleri gözlenmiştir. Böylece bu deneyle, aynı HW değerine sahip her bir giriş için alınan ölçümlerin aynı dağılıma sahip olduğu görülmüştür. Eş dağılıma sahip bir verideki HW değerleri "Binom" dağılımına sahiptir [21] Bu nedenle oluşan histogramda HW=4'e karşılık düşen ortalama genlik değeri en sık görülürken, HW=0 ve HW=9 değerinden kaynaklananlar sıklığı en az olan değerlerdir.

Sonuç olarak çoğu elektronik cihazda, eş dağılıma sahip bir veri kullanılarak ölçülen veri bağımlı güç bileşeni P_{veri}, yaklaşık olarak normal dağılıma sahip olur. Bunu yanı sıra, işlem bağımlı güç değişimi P_{islem} de yaklaşık olarak normal dağılıma sahiptir.

YKA amacıyla alınmış bir güç tüketim eğrisinde $P_{el-gür}$, P_{veri} ve P_{islem}, en fazla ilgilenilen ölçüm bileşenleridir. Bir ölçümden alınabilecek bilgi kalitesini ölçmek için [21]'de anlatılan sinyal/gürültü oranının tespit edilmesi yaklaşımı kullanılabilir.. Bir güç eğrisinde her bir noktada ölçülen işlem ve veri bağımlı bileşenler olan P_{veri}ve

P_{islem}'in tamamı, saldırgan tarafından kullanılabilecek bir içeriğe sahip değildir. Bu bileşenlerin saldırgan tarafından kullanılabilecek türde bilgi içeren kısmını tıpkı [21]'deki gibi P_{bilgi} olarak adlandıralım. O zaman P_{islem}ve P_{veri}'de bulunan ve saldırgan tarafından kullanılamayacak olan diğer değişimler ise daha önceki bölümde bahsedilen $P_{anh-gür}$ gürültüsü olacaktır.

$$P_{anh-gür}+P_{bilgi}=P_{islem}+P_{veri} \tag{1.8}$$

Bu gösterim ışığında, güç eğrisindeki bir saat vurusuna ait ortalama güç değeri, birbirinden bağımsız aşağıdaki bileşenlerin toplamından oluşur.

$$P_{toplam}=P_{bilgi}+P_{anh-gür}+P_{el-gür}+P_{sabit} \tag{1.9}$$

Burada saldırgan tarafından bilgi alınabilecek tek bileşen, P_{bilgi} olarak adlandırılan ve saldırı türüne göre veri ve işlem bağımlılığı içeren bileşendir. Bu değer, sadece ilgilenilen işlem ve veri ile ilgili değerleri kapsadığından, yine veri ve işlem bağımlılığı olan $P_{anh-gür}$ değerinden de bağımsızdır. Bu durumda bir güç eğrisinde belli bir saat vurusunda ölçülen sinyal/gürültü oranı (SNR) aşağıdaki formülle ifade edilebilir:

$$SNR = \frac{Var(P_{bilgi})}{Var(P_{anh-gür}+P_{el-gür})} \tag{1.10}$$

Burada $Var(P_{bilgi})$, ölçümdeki bir noktanın, YKA açısından değerli olan bilgiden dolayı ne kadar değişime sahip olduğunu gösterir. $Var(P_{anh-gür}+P_{el-gür})$ değerleri ise, analiz açısından bilgi taşımayan ve ilgilenilen anda devrede gerçekleşen diğer kapı geçişleri yani anahtarlama gürültüsü ve devrede oluşan elektronik gürültüden kaynaklanan değişimleri kapsamaktadır. Güç tüketimindeki sabit bileşenin varyansı "0" olduğundan bu formülde yer almamaktadır. Basit güç analizinde, $Var(P_{bilgi})$ hem veri hem işlem bağımlı bileşenler içerirken, FGA türü saldırılarda, hep aynı işlem hedef seçildiğinden, işlem bağımlı değişim "0" olup değerli tüm bilgi, veri bağımlı değişimden kaynaklanmaktadır.

Önceki bölümlerde açıklandığı gibi, bir güç eğrisinin bileşenleri olan elektronik gürültü $P_{el-gür}$, veri bağımlı güç tüketimi P_{veri}ve işlem bağımlı güç tüketimi P_{islem} yaklaşık olarak normal dağılıma sahiptir [21]. Normal dağılım ise ortalama değer "μ"

ve varyans değeri "σ^2" parametreleri ile tanımlanır. Normal dağılımı ifade edebilmek için gerekli olan bu parametreler pratikte belli sayıda ölçüm kullanılarak (1.6) ve (1.7) eşitliklerindeki gibi kestirilebilir.

Örnekleme dağılımı, kestirimi yapılan bir parametreye ne kadar iyi yaklaşım yapıldığını gösteren bir kavramdır. İlgili parametrenin kestirimi için belli sayıda ölçüm kullanılarak gerçekleştirilecek her farklı deneyde, kestirim sonuçları, aslında birbirine yakın olmakla beraber farklı değerlere sahip olur. Bu durum ise ortalama değerin kestirimi olan $\bar{x}$ ve varyans değerinin kestirimi olan s^2'nin de aslında birer rastgele değişken olduklarını göstermektedir. Bu kestirim değerlerinin dağılımı da yine bir normal dağılımdır ($\bar{x} \sim N(\mu, \frac{\sigma^2}{n})$) ve parametreleri aşağıdaki eşitliklerle hesaplanır:

$$E(\bar{x}) = \mu \tag{1.11}$$

$$Var(\bar{x}) = \frac{\sigma^2}{n} \tag{1.12}$$

Eşitlik (1.12)'de görüldüğü gibi $var(\bar{x})$ değeri, ölçülen eğri sayısı "n" ile azalmaktadır. Bu eşitlik, bir ölçümdeki elektronik gürültüyü azaltmak amacıyla, neden pek çok ölçüm ortalamasının kullanıldığını teorik olarak da açıklamaktadır.

Varyans değerinin kestirimi olan s^2'nin örnekleme dağılımı ise "chi-square" türündedir [20 - 21 , 24] ve parametreleri aşağıdaki gibi kestirilir.

$$E(s^2) = \sigma^2 \tag{1.13}$$

$$Var(s^2) = \frac{\sigma^4}{n-1} \tag{1.14}$$

Standart normal dağılımda bir Z değişkeninin z_α değerinden küçük olma olasılığı "α" olarak alındığında, $p(Z < z_\alpha) = \Phi(\alpha)$ özelliği bulunmaktadır. Burada $\Phi(x)$ "toplamsal dağılım fonksiyonu (qumulative distrubution function)" olarak adlandırılır. Normal dağılımda sık kullanılan bir diğer özellik te $z_\alpha = -z_{1-\alpha}$ eşitliğidir. Daha önce bahsedildiği gibi, x değişkeninin ortalama değeri $\bar{x}$ 'in örnekleme dağılımı $\bar{x} \sim N(\mu, \frac{\sigma^2}{\sqrt{n}})$ normal dağılımdır. Burada $z = (\bar{x} - \mu)\frac{\sigma}{\sqrt{n}}$ değişken dönüşümü yapıldığında, z

değişkeni standart normal dağılıma sahip olur. Yukarıdaki açıklamalar aşağıdaki eşitlikle ifade edilmektedir.

$$P(z_{\alpha/2} \leq z \leq z_{1-\alpha/2}) = 1 - \alpha \quad (1.15)$$

Bu eşitlikteki α değeri "hata olasılığı", $[z_{\alpha/2}, z_{1-\alpha/2}]$ ise "güven aralığıdır". (1.13) denklemi $\bar{x}$ türünden yazıldığında ise, ortalama değer μ nün kestirimine ait $1-\alpha$ güven aralığı aşağıdaki gibi olur.

$$[\bar{x} - \frac{\sigma}{\sqrt{n}} . z_{1-\alpha/2} , \bar{x} + \frac{\sigma}{\sqrt{n}} . z_{1-\alpha/2}] \quad (1.16)$$

Hipotez testi, belli bir ölçüm setinden değeri kestirilmeye çalışılan bir μ değerinin, gerçek ortalama değeri μ_0 olan belli bir dağılıma ait olup olmadığını anlamada kullanılan istatistiksel bir kavramdır. Bu istatistiksel kavramların ayrıntıları [24]'de bulunabilir. (1.16) eşitliğinde gösterildiği gibi $c=\frac{\sigma}{\sqrt{n}}$ olarak alındığında, $\bar{x}$ kestirim değerinin, gerçek değer olan "μ_0" ın "c" komşuluğunda olması, yani kestirim hatasının "c" den küçük olması olasılığı "$1-\alpha$" olarak verilir.

$$P(|\bar{x} - \mu_0|) > c=1-\alpha \quad (1.17)$$

Burada tanımları verilen güven aralığı ve hipotez testi kavramları, belli bir YKA saldırısında başarıya ulaşmak için, ölçülen eğrilerden kaç tane kullanılması gerektiği ve belli koşullarda sürülen hipotezlerin doğruluğuna karar verilip verilmemesi bağlamında kullanılabilecek ölçütlerdir [20 - 21 , 24]. İleriki bölümlerde tanıtılacak olan yenilikçi yöntemlerde, bu istatistiksel kavramlar kullanılarak, analizlerin başarımında kaç eğri kullanılması gerektiği teorik olarak hesaplanmış ve pratik olarak ta gösterilmiştir.

1.2. Güç Analizi Yöntemleri

Literatürde yer alan temel güç analizi türleri, Basit Güç Analizi (BGA) [3], Farksal Güç Analizi (FGA) [3 - 4], İlintisel Güç Analizi (İGA) [5 - 6 , 18 - 19], Çakışma Analizi [16 - 17 , 25 - 32] ve Şablon Analizi [33 - 35] olarak sıralanmakla birlikte sadece bu yöntemlerle sınırlı değildir. Aslında işaret işlemedeki "k-ortalamalar" [36] ya da "yapay sinir ağları" [37] gibi pek sınıflandırma yöntemi de, doğrudan güç analizi

amacıyla ya da diğer YKA türlerine destekleyici yöntem olarak kullanılabilmektedir. Literatürdeki temel YKA yöntemlerinin her biri eğrileri kullanış tarzına göre de dikey ve yatay analiz olarak iki temel sınıfa ayrıştırılabilir [26]. Dikey analiz yöntemlerinde birden fazla eğride aynı işlem anına ait alanların değerlendirilmesi yapılır ve her bir anahtar parçası için farklı girdilerle oluşturulmuş eğri parçalarına gereksinim duyulur. Yataydaki çalışmaların dikeydeki çalışmalara en önemli farkı ve üstünlüğü ise belli bir anahtar parçasını elde etmek amacıyla aynı eğri üzerindeki diğer işlem anlarına ait alanların da analizde kullanılabiliyor olmasıdır. Yatay analizde tek bir eğri ile sonuca gidilebilmesi mümkün olabilmektedir.

Yukarıda sıralanan güç analizi yöntemlerinin tümü, güç eğrilerinin zaman örnekleri yerine, bazı üstünlükler sunan frekans uzayına dönüştürülmüş değerleri üzerinde de gerçekleştirilebilir [38 - 46].

1.2.1. Basit güç analizi

Basit güç analizi (BGA), "bir kriptolojik işlemin gerçekleşmesi sırasında ölçülen güç tüketim eğrilerinin, kullanılmakta olan anahtar değeriyle doğrudan ilişkilendirilmesi" olarak tanımlanmaktadır [3]. Basit güç saldırısında genel olarak az sayıda güç eğrisi kullanılarak anahtar değerinin elde edilmesi hedeflenir. CMOS tabanlı sayısal devrelerde, güç tüketim eğrilerinin şekilsel olarak gerçekleştirilen işleme bağımlılık göstermesi BGA kaçaklarının temel nedenidir [3]. Örneğin bir mikro işlemcide belli bir komut seti kullanılarak algoritmalar gerçeklenir. Kullanılan komutların her biri, işlemcinin iç ya da dış belleği, lojik birimi, ya da çevre birimleri gibi farklı fiziksel parçaları üzerinde işlem yaparlar. Her biri işlemcideki fiziksel olarak ayrı alanlar üzerinde çalışan bu komutların da kendine özgü farklı bir güç izi bulunur. Algoritma koşturumu sırasında oluşan eğriler incelenerek, hangi komutların kullanıldığı dolayısıyla gerçekleşen işlem sırası hakkında bilgi edinilmesi mümkündür. Eğer algoritmanın ara adımlarında anahtar parçaları ile ilişkili olarak farklı işlem yoluna dallanma söz konusu ise, bu durum güç eğrilerinden tespit edilerek anahtar değeri parça parça elde edilebilir. BGA saldırısının başarılı olabilmesi için bu işlem farkının güç eğrisi üstünde önemli bir etkiye sahip olması gerekir. BGA saldırısı uygulamak için, gerçekleşen algoritmaların işlem ayrıntıları hakkında da bilgi sahibi olunması gerekir. Örneğin DES algoritmasına ait "tur anahtarı oluşturma (key schedule)"

adımlarında, anahtar kütüğünün döndürüldükten sonra, en düşük anlamlı bitin değerine göre farklı işlem yollarına gidilmesi BGA kaçaklarına yol açabilmektedir. Bunun yanı sıra simetrik algoritmalarda sıkça kullanılan permutasyon işlemelerindeki şartlı dallanmalar da BGA kaçaklarının nedenleri arasında yer alır. Asimetrik algoritmalarda ise kullanılan modüler çarpıcıların işlem girdilerinin HW değerlerine göre de BGA kaçakları oluşabilmektedir. Yine RSA ya da Eliptik Eğri gibi asimetrik algoritmalarda, temel işlem adımı olan üs alma ya da katlama-ekleme döngülerinde anahtar bitinin değerine göre çarpma ya da ekleme işlemlerinin şartlı olarak gerçekleştirilmesi ve bu iki işlemin bir birinden ayrıt edilebilmesi durumlarında BGA kaçakları oluşabilmektedir.

BGA kaçaklarını engellemek için genel olarak algoritma işleyişinde gizli ara değerlere göre şartlı bir şekilde işlem farkının oluşturulmaması gerekir. Algoritmaların işlem akışında bu durum söz konusu olsa bile gerçeklemede aynı sonucu verecek şekilde işlem yollarının düzenlenmesi, bazı sahte işlemlerin eklenmesi gibi çözüm yöntemleri bulunmaktadır.

1.2.2. Farksal güç analizi

Farksal güç analizi (FGA) [3 - 4] bir kriptolojik işlemin koşturulması sırasında ölçülen çok sayıda güç tüketim eğrisinin istatistiksel analizi ile işlem ara değerlerinin dolayısıyla bu değerlerin oluşmasına katkı sağlayan anahtar parçalarının elde edilmesinde kullanılan güç analizi türüdür. CMOS devrelerde güç tüketiminin genliği işletilmekte olan verinin değerine bağımlılık gösterir. FGA sadırlarında eğrinin şekli değil de veri bağımlı bu genlik değişimleri önem taşımaktadır. FGA yöntemlerinde genel olarak işletilen algoritmanın gerçekleme ayrıntılarının bilinmesi çok önem taşımamsa da, algoritma girdi/çıktı değerlerinden en az birinin bilinmesi ya da kontrol edilmesi gerekebilmektedir.

Klasik yani dikeyde gerçekleştirilen FGA yönteminde, temel olarak sabit bir “k” anahtarını elde etmek üzere, pek çok farklı değerde verinin kullanıldığı şifreleme ya da çözme işlemine ait güç eğrilerinde, aynı işlem anına denk gelen alanların fark eğrileri kullanılır. Bu yöntemde öncelikle giriş/çıkış verisinden biri ile anahtar parçasının bir fonksiyonu olan bir algoritma ara değeri hedef olarak seçilir. Seçilen ara değerin, verilen her bir giriş değeri ve elde edilmesi hedeflenen anahtar parçasının tüm

olası değerleri için, alabileceği değerler hesaplanır. Daha sonra hesaplanan bu olası ara değerin işlem gördüğü ya da oluştuğu anda tüketilen gücün kestirimine dayanan teorik güç modeli oluşturulur. Bu teorik model, tüketilen gücün, hedef ara değerde, değeri "1" olan toplam sinyal sayısı yani Hamming Weight (HW) değeri, ya da ara değerin işlenmesi sırasındaki toplam 1-0, 0-1 sinyal değişimi sayısı yani Hamming Distance (HD) değerleri ile doğru orantılı olduğu varsayımına dayanır. Bu teorik modele dayanan bir seçim fonksiyonu oluşturularak güç eğrileri gruplandırılır. Eğrileri gruplamada kullanılan en temel seçim fonksiyonu ise eğrileri HW ya da HD' si yüksek olanlar ve düşük olanlar diye iki gruba ayırmaktır. Oluşturulan her bir eğri grubun ortalamasının farkı hesaplanarak, belli bir anahtar parçası kestirimine ait fark eğrisi elde edilir. Bu eğride bir tepe değer oluşup oluşmadığına göre gruplamanın, dolaysıyla kestirimin doğru yapılıp yapılmadığına karar verilir. Tepe değerin bulunması, gruplandırmanın dolaysıyla da kestirimin doğru yapıldığını gösterdiğinden, ilgili anahtar parçası elde edilmiş olmaktadır. Aksi durumda, modele sonraki olası değerler konarak, fark eğride bir tepe değer bulunana kadar işlemlere devam edilmektedir. Bazen sahte tepe değerlerinin oluşması da söz konusu olduğundan, tüm olası değerlere göre işlem yapılıp, tepe değeri oluşturan fark eğrinin bütün uzayda aranması daha az yanıltıcı olabilmektedir.

Yatayda farksal güç analizi, klasik ya da dikeyde farksal güç analizinin bir türevi olarak düşünülebilir. Her iki yöntemde de saldırgan algoritma işleyişindeki ara değerlere dayanan HW ya da HD tabanlı bir güç tüketim modeli oluşturur. Ancak aradaki temel fark yatay fark analizi yönteminde güç tüketim modelinin aynı eğri üzerindeki farklı işlem anlarına ait eğri parçaları için oluşturulmasıdır.

FGA saldırılardan korunmak için, üs körleştirme [49 , 62 - 64], mesaj körleştirme [3 , 49] ve modülüs körleştirme [62] gibi maskeleme tarzında karşı önlemler kullanılabilmektedir.

1.2.3. İlintisel güç analizi

İlintisel Güç Analizi (İGA) [5 - 6 , 18], tıpkı FGA'da olduğu gibi güç eğrilerindeki veri bağımlı bileşenlerin istatistiksel analizine dayanır. İGA türü yöntemlerin FGA yöntemlerinden temel farkı, algoritma adımlarında işlem gören verideki "1" lerin sayısı (HW) ya da "1-0, 0-1" geçiş sayısına (HD) göre oluşturulan teorik güç tüketim

modellerinin, gerçek güç eğrileri ile olan ilintisinin kullanılmasıdır. Hesaplanan bu ilinti değerinin seviyesine göre anahtar parçasının doğru kestirilip kestirilmediğine karar verilir.

Yatayda İGA klasik İGA yönteminin bir türevidir. Her iki yöntemde de algoritma işleyişindeki ara değerlere dayanan HW ya da HD tabanlı bir güç tüketiminin gerçek eğrilerle ilintisi kullanılır. Ancak aradaki temel fark yatay yöntemde güç tüketim modelinin aynı eğri üzerindeki farklı işlem anlarına ait eğri parçaları ile ilintisinin göz önüne alınmasıdır.

İGA türü saldırılar için de FGA karşı önlemleri olarak kullanılan üs körleştirme [49 , 63 - 66] mesaj körleştirme [3 , 49], modülüs körleştirme [62] sahte işlemler ekleme [64 , 67] türünde yöntemler kullanılabilmektedir.

1.2.4. Çakışma analizi

Karşılaştırma ya da çakışma analizi, bir algoritma işletilirken gerçekleşen aynı türdeki ara işlem adımlarının, aynı veri değerini kullanıp kullanmadığının tespitine dayanır [16 - 17 , 25 - 32]. Bir veya birden fazla eğride bu tür bir çakışmanın olması durumunda, bu anlara ait güç eğrisi parçalarının benzemesi, örneğin aralarında hesaplanacak çapraz ilinti değerinin yüksek olması beklenir. Çakışma analizinde, aynı eğrideki farklı işlem anlarına ait parçalar karşılaştırıldığında analiz yatayda gerçekleştirilmiş olurken, farklı girişlere ait birden çok eğride aynı işlem anları karşılaştırılmakta ise bu yöntem dikey çakışma analizi olarak adlandırılır [26 , 28].

Çakışmanın tespitinde fark eğri, ilinti ve çapraz ilinti analizi yöntemleri kullanılabilmekte ve çakışma analizi kullanılan bu yönteme göre “çapraz ilinti çakışma analizi”, “farksal çakışma analizi” gibi isimler alabilmektedir [26]. Bunun yanı sıra çakışmaların tespitinde “Support Vector Machine ” kullanan çalışmalar da bulunmaktadır [68].

FGA ve CPA türü saldırılarda, üs körleştirme [49 , 62 - 64], mesaj körleştirme [3 , 49] ve modülüs körleştirme [62] yöntemleri karşı önlem olarak kullanılabilirken, tek bir eğri ile de gerçekleştirilebilen yatay çakışma analizi türü yöntemlere karşı koymada bu önlemler yetersiz kalmaktadır. Tek eğri ile gerçekleştirilebilen çakışma analizi türlerine karşı çarpıcı seviyesinde, rastgele körleştirme ve yine çarpıcı seviyesinde

işlem sırasını rastgeleleştirmeye dayanan ve gerçekleme açısından daha fazla çaba gerektiren yöntemler kullanılabilmektedir [26].

1.2.5. Şablon tipi güç analizi

Literatürdeki bir diğer yan kanal saldırı yöntemi de Şablon Analizi (ŞA) olarak bilinmektedir. FGA ve İGA saldırıları, basitçe HW ya da HD değerlerine dayanan tahmini güç modellerini kullanırlar. Tahmini modellemenin doğru şekilde yapılamadığı ya da yetersiz kaldığı durumlarda, bu yöntemler başarıya ulaşamazlar. Şablon türü yöntemlerde ise hedef cihaza saldırırken, bu hedefle özdeş ve daha önce güç tüketim karakteristiği çıkarılmış başka bir cihazın güç tüketim modeli kullanmaktadır. Buradan anlaşılacağı üzere şablon saldırıları [33 - 35], hedef cihaz hakkında bilgi sahibi olmayı sağlayan güç modeli çıkarma evresi ve bu modeli kullanarak anahtarı elde etmeyi hedefleyen eşleştirme evrelerinden oluşur. Karakter çıkarma evresi, çok fazla ölçümle gerçekleştirilebilecek iken, şablon eşleştirme yani anahtarı elde etme evresinde çok daha az ölçüm kullanılmaktadır.

1.2.6. Frekans uzayında güç analizi

Yukarıda anlatılan güç analizi yöntemleri aslında zaman uzayı yerine frekans uzayında da gerçekleştirilebilir. Bu yöntemlerin zaman uzayında uygulanmasında, algoritma koşturumlarından elde edilen her bir güç eğrisinin, hedef işlem etrafında iyi bir şekilde yanaştırılmış olması gerekmektedir. Ölçüm düzeneğindeki tetikleme ya da gerçeklemede bulunabilecek sahte işlem eklenmesi gibi özelliklerden kaynaklanan yanaştırma sorunları, analiz başarımını olumsuz yönde etkilemektedir. Bununla birlikte, hedef işlemle aynı anda gerçekleşen diğer işlemlerden kaynaklanan anahtarlama gürültüsü ve her devrede bulunan elektronik gürültü de başarımda olumsuz etkilere sahiptir. Tüm bunların üstesinden gelmek üzere, zaman uzayı güç analizine bir alternatif olarak frekans uzayı çalışmaları literatüre girmiştir [38 - 46] . Zaman uzayındaki yanaştırma sorunları, frekans uzayında sadece bir faz kayması olarak görünmekte [44] ve zaman uzayı kadar olumsuz sonuçlara yol açmamaktadır. Ayrıca eğer hedef işlemden kaynaklanan YKA kaçağı ile gürültü farklı frekans bantlarında ise, zaman yerine frekans uzayında çalışılarak her ikisini birbirinden yalıtmak mümkün olmaktadır. Tüm bu avantajlarından dolayı frekans uzayı üç analizi önem kazanmaktadır.

1.3. RSA Algoritması

RSA algoritması, bilinen ilk "Açık Anahtarlı (Public Key)" asimetrik şifreleme algoritmasıdır. Bir algoritmanın asimetrik olması, şifreleme ve çözme işlemleri için iki farklı anahtar değerinin kullanımı anlamına gelmektedir. "Açık anahtar" ile kastedilen ise anahtarlardan birinin herkesçe bilinen bir değer alabilmesidir. Paylaşılan anahtar "açık anahtar" adını alırken, gizli olması gereken ise "özel anahtar" olarak adlandırılır. Açık anahtarla şifrelenmiş bir verinin gizli anahtarla çözülmesi gerekir. RSA algoritması kullanılarak iletişimin şifrelenmesi-çözülmesi mümkün olsa da bu iş için daha hızlı çalışan özel anahtarlı, diğer bir değişle simetrik türdeki algoritmalar kullanılmaktadır. DES, AES gibi simetrik algoritmaların şifreleme ve çözmede kullanılan anahtarlarının gizli tutulması gerekir. Bu algoritmaların kullanımındaki en önemli sorunlardan biri olan güvenli anahtar paylaşımı ise RSA ve Elliptic Eğri Şiferleyici (EEŞ) gibi asimetrik algoritmalar kullanan "Diffie-Helman Anahtar Dağıtımı" [47] türündeki protokollerle sağlanır. Bunun yanı sıra bir verinin doğru kaynaktan geldiğini anlamak amacıyla kullanılan DSA [48] gibi imzalama ve imza doğrulama protokolleri de RSA ve EEŞ gibi asimetrik algoritmalarla çalışmaktadır.

RSA algoritması matematiksel olarak modüler üs alma döngüsünden oluşur. Büyük n-bitlik bir değer olan ve modülüs olarak kullanılan m değeri, n/2 bitlik iki asal değer olan p ve q'nun çarpımıyla elde edilsin yani m=p*q eşitliği sağlasın. Φ (m)=(p-1)(q-1) olarak tanımlandığında "e" ve "d" iki tamsayı olmak üzere, "ed.mod Φ(m)=1" koşulunu sağlayan "e" ve modülüs değeri olan "m" açık anahtar, d değeri ise özel anahtar olarak adlandırılır. Şifreleme işlemi özel üs değeri "d" kullanılarak S=Y^dmod(m) şeklinde, şifre çözme işlemi ise açık üs değeri "e" kullanılarak, C=S^e mod(m) şeklinde gerçekleştirilir. $(S^e)^d$=S eşitliği her zaman sağlanır. RSA algoritmasının güvenliği, büyük bir değer olan modülüsün m=q.p asal çarpanlarına ayrılmasının, "e" ve "m" açık anahtar çifti bilindiğinde "d" kapalı anahtarının bulunması kadar zor bir işlem olmasına dayanmaktadır. RSA algoritmasında modülüsün bit sayısı, algoritmanın bit sayısı olarak adlandırılır ve bu değer n=256-4096 Aralığında olabilmektedir.

Bilgi işleyen cihazlardaki gelişen hesaplama gücüyle orantılı olarak RSA algoritmasının düşük anahtar boylarında kullanılmaması gereği doğmaktadır. Bu

durum daha az anahtar boyunda aynı güvenliği sağlıyor olması nedeni ile RSA algoritmasının yerini EEŞ algoritmasına bırakmasına neden olmaktadır.

1.4. Literatürdeki Üs Alma Adımlarına Güç Analizi Uygulamaları

RSA gerçeklemelerinin tümünde, anahtarla ilgili en temel işlem olan üs alma adımları YKA saldırılarının ana hedeflerinden biridir. EEŞ algoritmasındaki skaler çarpım işlemlerinde yer alan katla-ekle döngüsü, RSA'da gerçekleştirilen üs alma döngüsünün özdeşidir. EEŞ katla-ekle döngüsüne uygulanan pek çok saldırı türü RSA algoritmasının üs alama adımlarına da uyarlanabilir. Bu nedenle üs alma saldırıları incelenirken EEŞ katla-ekle döngüsüne uygulanan saldırı türleri de göz önüne alınmıştır.

Bu bölümde özellikle standart ikilik üs alma ya da Montgomery merdiveni üs alma yöntemlerine uygulanabilecek türdeki YKA saldırıları incelenmiştir Daha sonra tez kapsamında geliştirilen özgün yöntemlerin literatürdeki hangi yöntemlerle benzeştiği, hangi saldırı ailesine dâhil olduğu ve bu yöntemlerden ne yönde farklılaştıkları incelenmiştir.

1.4.1. Basit güç analizi

RSA algoritmasında karşılaşılabilecek en basit BGA saldırısı, klasik kare al-çarp yöntemi gerçeklemelerindeki kare alma ve çarpma işlemlerinin oluşturduğu farklı güç izlerinden tespit edilmesine dayanan yöntemdir. Bu yöntemde anahtarda değeri bir olan her bir bit için kare alma ve çarpma işlemlerinin her ikisi de gerçekleştirilirken sıfır olanlar için sadece kare alma işlemi yapıldığından çarpma ve kare almanın ayrıt edilebilmesi anahtar birilerinin elde edilmesi anlamına gelmektedir. Bu saldırı türüne karşı koymak amacıyla geliştirilen kare al- hep çarp yöntemi [49], işlem yolunu anahtar değerinden bağımsız hale getirerek BGA'ya karşı koruma sağlamaktadır. Ancak sahte işlem eklemeye yönelik önlemler diğer YKA türleri [32 - 50] ile devre dışı bırakılabilmektedir. Joe ve Yen tarafından BGA saldırılarına karşı, önerilen bir diğer üs alma algoritması ise [51] Montgomery Merdiveni (Montgomery Ladder -ML) üs alma [52] yöntemidir. Bu yöntemde her bir anahtar biti için aynı işlem dizisi gerçekleştiğinden BGA açıklığı oluşmamaktadır. Ancak ML yöntemine karşı veri

bağımlılığını kullanan diğer İGA-FGA tipi saldırılar [53] etkili olarak uygulanabilmektedir.

1.4.2. Farksal güç analizi

RSA gerçeklemesine uygulanan klasik yani dikeyde FGA tipindeki saldırı yöntemlerinden ilki Messenger ve arkadaşları tarafından [4]'de önerilen 3 yöntemden biri olan Zero-Exponenet, Multiple-Data (ZEMD) 'dir. Bu yöntemde aynı üs değeri ile pek çok verinin işlem görmesi gerekmektedir. Bu yöntemin uygulanması için 0-i arası anahtar bitleri biliniyorken i+1 nolu bitin elde edilmesi için bu bite ilişkin bir kestirim yapılarak tüm girişlerin, i+1 nolu bit için oluşacak ara değerleri hesaplanır. Bu değerlerin HW'sine göre güç eğrilerine ait fark eğrileri oluşturulur. Eğer ilgili bitin kestirimi doğru ise fark eğride tepe değer görülecektir. Bu şekilde bitler belli bir sırada elde edilebilmektedir.

Yatayda gerçekleştirilen farksal güç analizi yöntemlerinden ilk örneklerinden biri Walter tarafından geliştirilen ve "Big Mac" olarak isimlendirilen [25] çalışmadır. Bu çalışmada üs alma adımlarında gerçekleşen uzun çarpma işlem girdilerinden biri olan pencere değerleri, bu girdilere ait ortalama eğri alanlarının birbirinden olan "Euclid" uzaklığı ile ayırt edilmeye çalışılmaktadır. Walter'ın "kayan pencere türündeki" bir RSA gerçeklemesine uyguladığı bu yöntem, aslında diğer üs alma yöntemlerine de uyarlanabilir türde bir çalışmadır.

1.4.3. İlintisel güç analizi

Klasik dikey İGA yöntemi ilk kez Amiel ve arkadaşları tarafından [6] asimetrik kriptolojiye uygulamıştır. Bu çalışma ZEMD FGA [4]saldırısının İGA türevi olarak nitelendirilmektedir. Çalışmada RSA koşturumlarının üs alma sırasında oluşan kare alma ya da çarpma işlemlerine ait çıktıların, HW ya da HD'sinin güç eğrileri ile olan ilintisi kullanılmaktadır. Bu yöntemin birden fazla sayıda anahtar biti üzerinde eş zamanlı arama yapacak şekilde geliştirilebileceği de gösterilmiştir. Tıpkı FGA yönteminde olduğu gibi saldırganın giriş değerleri hakkında bilgi ya da kontrol sahibi olması gerekir.

Yatayda ilintisel güç analizi çalışmasının ilk örneklerinden biri [28] ikilik kare-al, çarp türündeki RSA gerçeklemesine uygulanmıştır. Bu çalışmada değeri 1 olan bitleri tespit

etmeye yönelik her bir bite ait işlem alanında "x" veri değeri ile çarpma yapılıp yapılmadığı anlaşılmaya çalışılmaktadır. Bu amaçla "x" değerinin uzun çarpmanın alt adımlarında işlenen parçalarının HW değeri ile her bir işlem adımına ait güç eğrisi bölütlerinin ilintisi hesaplanmaktadır. Bu çalışma aslında Walter'a ait yataydaki FGA yönteminin [25] ilinti analizi ile gerçekleştirilen şekli olarak değerlendirilebilir. Bu yöntem "kare-al, her zaman çarp" ve "Montgomery merdiveni" gibi diğer üs alma yöntemlerine de uyarlanabilir. Mesaj boyunun büyük olması ve uzun çarpma işlemine giren kelime boyunun küçüklüğü, kullanılabilecek eğri parçası sayısını artırdığından bu saldırı türünün başarımını da artırmaktadır.

Tez kapsamında geliştirilen ve Montgomery Merdiveni kullanan RSA gerçeklemesine uygulanan [18] yöntemi, aslında güç eğrilerini hem yatayda hem dikeyde kullanarak ilintisel güç analizi geçekleştirmektedir. Bu çalışmada anahtar bitlerinden oluşturulan belli sayıdaki bit gruplarına ait tip vektörlerinin, çeşitli RSA koşturumlarından elde edilen güç eğrisi bölütleri ile hesaplanan ilinti değerleri kullanılmaktadır. Çalışma, hem aynı eğri üstündeki hem de farklı eğrilerde aynı anlara denk gelen eğrileri kullanması ile hem yatay hem dikey yönde çalışan "karma" türdeki ilintisel güç analizi çalışmalarının özgün örneklerinden biridir.

1.4.4. Çakışma analizi

1.4.4.1. Farksal çakışma analizi

Fouque ve Valette tarafından "Doubling attack" [54] olarak literatüre tanıtılan ve "hep çarpma" türündeki üs alma gerçeklemesine uygulanan yöntem, birden fazla eğride belirli işlem anlarına ait alanlar karşılaştırdığından dikeyde çakışma analizinin ilk örneklerinden biridir. Yöntemin uygulanmasında, üs alma gerçeklemesine "x" ve "x^2" veri değerleri girdi olarak verilip, oluşan güç eğrileri karşılaştırılmaktadır. Hesaplamalarda üs değerindeki "i" nolu bit 0 değerini aldığı zaman x^2girdisi için yapılan "i." kare alma işlemi ile "x" girişi için gerçekleştirilen "i+1." kare alma işlemlerinde aynı değerlerin karesi alındığından bu adımın tespit edilmesi ile anahtar bitleri elde edilmiş olur. Örtüşmenin tespiti için bu çalışmada [54] basitçe eğri bölütlerinin farkı alınsa da, gürültünün olması durumunda çapraz ilinti analizinin kullanılması daha iyi sonuçlar verecektir. Bu yöntem daha sonra Yen ve arkadaşları tarafından [27] "Montgomery Merdiveni" gerçeklemelerine uyarlanmıştır. Bu

uyarlamada da “x” ve “x^2” gibi iki farklı giriş değerine ait ölçümlerde, değeri bir sonraki bit ile aynı olan bitler için bir çakışma durumu oluşmaktadır. Bu yöntemde anahtar bitlerinin doğrudan değeri değil sonraki bit ile aynı olup olmadıkları tespit edilmektedir.

Bir diğer çakışma türü saldırı da giriş değeri x’e modülüs türünden x=m-1 değeri verilerek üs alma adımlarında belli ara değerlerin oluşturulduğu yöntemdir [55]. Algoritmaya modülüsle ilişkili bu özel giriş verildiğinde oluşan ara değerler ya “1” ya da “m-1” olup, ilgili güç tüketim alanları kolayca birbirinden ayırt edilebilir hale gelmektedir. Bu yöntem özellikle hep çarp türündeki çarpma-kare alma işlemleri düzenli olarak seyreden ikilik üs alma algoritmasında bütün anahtar bitlerinin elde edilmesinde etkili bir şekilde kullanılabilmektedir.

Messengers ve arkadaşları tarafından [4] ‘de tanıtılan “single-exponent-multiple-data (SEMD)” ve “multiple-exponent single-data (MESD)” yöntemleri, tek bir eğrinin farklı bölümlerini işlediğinden yatayda basit çakışma analizi olarak sınıflandırılabilir. SEMD yönteminde bilinen ve bilinmeyen iki üs değeri ile pek çok rastgele veri işleme konup her bir üs değerine ait eğrilerin ortalaması birbirinden çıkarılır. Elde edilen bu fark eğride anahtar değerinin birbiri ile aynı olduğu alanların sıfıra yakın, farklı olanların ise sıfırdan farklı bir değer alması beklenir. Bu yaklaşım asılında doğrudan anahtar bağımlı işlemleri tespit etmeye yönelik olduğundan “adres-bit-FGA” türü açıklıkların tespitinde de kullanılabilir. MESD yönteminde ise bilinmeyen bir üs değerine ait anahtar bitleri, aynı mesajla işlem gören ve her bir bitin tespiti için, yeniden ilgili bitleri ayarlanan üs değerlerine ait ölçümlerle karşılaştırılır. Örneğin standart ikilik üs alma yönteminde anahtarın ilk i-1 tane biti bulunmuşken, i nolu biti tespit etmek için i=0 ve i=1 olasılıklarından her ikisini de alan üs değeri ile hesap yapılır. Her iki hesap için elde edilen ölçümlerde i. bite ait işlemler karşılaştırılarak hangi ölçümle çakışma olduğu gözlenir.

Yataydaki adrese bağlı çakışmaları farksal analiz ile tespit etmeye yönelik [56] çalışmasında ise öncelikle özel anahtar ile pek çok veri değeri işleme konarak ortalama eğriler oluşturulmaktadır. Veri bağımlılığı bu şekilde ortadan kaldırıldıktan sonra, sadece anahtar bitine bağlı olarak farklı adreslerden okuma yapılıp yapılmadığını tespit etmek için her bir bite ait ortalama eğri parçaları birbiri ile karşılaştırılmaktadır. Bu

çalışmada karşılaştırma için kullanılan ortalamaların farkı yöntemi yerine, bu değerlerin çapraz ilintisinin kullanılması da mümkündür.

Dikeyde farksal çakışma analizine örnek verilebilecek [57] çalışmasında, anahtar bitlerinin değerine göre işlem girdilerinin farklı belleklerden okunup okunmadığı ayırt edilmeye çalışılmıştır. Bu çalışmada pek çok farklı rastgele veri öncelikle hem bilinen hem de özel üs değeri ile işleme konarak ortalama eğriler oluşturulmaktadır. Bu şekilde oluşturulan eğrilerde veri bağımlılığı ortadan kalkarken sadece anahtar bitine göre farklı adreslerden okuma işleminden kaynaklanan güç değişimleri ortaya çıkarılmaktadır. Oluşturulan ortalama eğrilerin farkı alınarak da özel ve açık anahtarda her bir bit için aynı ya da farklı adreslerden okuma yapılıp yapılmadığı anlaşılmaya çalışılmaktadır.

1.4.4.2. Çapraz ilinti çakışma analizi

Hee Seok Kim ve arkadaşları tarafından geliştirilen ve binary-with-random-initial-point (BRIP) karşı önlemi içeren RSA gerçeklemesine uygulanan yöntem [29], birden fazla eğride aynı anlara ait örnek değerlerinden oluşturulan vektörlerin çapraz ilinti değerlerinin karşılaştırmasına dayandığından, dikeyde çapraz ilinti tabanlı çakışma analizi olarak sınıflandırılabilir. Bu çalışmada, anahtar bitinin değerine göre ön hesaplanmış iki değerden birini kullanarak gerçekleştirilen üs alma adımlarında, çarpım girdilerinden kaynaklanan çakışmaların yakalanması hedeflenmiştir. Yöntemi uygulamak için öncelikle farklı eğriler üzerinde referans olarak kullanılan ilk bit değerine ait pek çok eğrinin alt alta konmasıyla oluşturulmuş güç örneği değerlerinin, hedef bit işlemine ait örnek değerler ile oluşturulan vektörlerle olan çapraz ilintisi kullanılmaktadır. Referans bit ile aynı değere sahip bitlere ait vektörlerin, referans bite ait vektörle yüksek, farklı türde olanların ise düşük ilintili değerlerine sahip olması beklenmektedir. Hesaplanan çapraz ilinti değerleri toplandıktan sonra değerlendirmesi yapılmaktadır. Yöntemin Montgomery Merdivenine uygulanışına ait bir yöntem önerisi de aynı çalışmada yer almaktadır. Dikeyde çapraz ilinti tabanlı bir diğer çakışma analizi ise Witteman ve arkadaşlarının [30] her zaman çarp türü ikilik üs alma yöntemindeki sahte çarpma işlemlerini ayırt etmeye yönelik çalışmasıdır. Bu çalışmada her zaman çarp yöntemindeki sahte çarpmanın tespit edilmesi için ardışık bitlerde oluşan ara değer çakışmaları tespit edilmeye çalışılmaktadır. Değeri ”0” olan

bitler için yapılan sahte çarpma sonuçları sonraki bit için kullanılmadığından, çakışmaların tespit edildiği bitlerin değerinin "1" olduğu anlaşılabilmektedir. Bu çalışmada her bir bite ait işlem alanı toplanarak tek bir değer ile ifade edilmiş ve pek çok güç eğrisi için hesaplanan bu değerler alt alta sıralanarak çapraz ilintileri hesaplanmıştır. Eğer hesaplanan çapraz ilinti değerleri çoğunlukla yüksek değerler almakta ise, bu ardışık işlemler arasında çakışmanın olduğu, dolayısıyla ilgili bit değerinin 0 olduğu anlaşılır. Sıfıra yakın ilinti değerlerinde ise çakışmanın olmadığı dolayısıyla bit değerinin 1 olduğu sonucuna varılmaktadır.

Yatayda çapraz ilinti tabanlı çakışma analizi olarak sınıflandırılabilecek ilk çalışma Messengers ve arkadaşları [5] tarafından gerçekleştirilmiştir. Bu çalışmada çapraz ilinti kullanılarak RSA'deki çarpma ve kare alma işlemleri ayırt edilmeye çalışılmış anacak sadece bu işlemlerin gerçekleştiği zaman dilimleri tespit edilebilmiştir. İki işlem başarılı bir şekilde birbirinden ayırt edilememiştir. Yatayda çapraz ilinti tabanlı bir diğer çalışmada Bauer ve arkadaşları tarafından [58] EEŞ algoritmasındaki skaler çarpım adımlarına uygulanmıştır. Skaler çarpımda en az bir ortak girdi kullanan katlama işlemlerini tespit etmek üzere ekleme işlemi sırasında gerçekleştirilen uzun çarpma işleminin alt adımlarına ait eğri alanları birbiri ile karşılaştırılmıştır. Böylece aynı girdilerin kullanılıp kullanılmadığına ve işlenen anahtar bitinin değerine karar verilmektedir. Gizli skaler değerin toplamsal olarak maskelenmesi, nokta körleştirme, atomik işlem kullanımı gibi karşı önlemler tek bir eğri kullanılarak yatayda gerçekleştirilen bu tür bir çapraz ilinti analizini engelleyememektedir.

Tez kapsamında geliştirdiğimiz yatayda çapraz ilinti tabanlı çakışma analizi türündeki yöntemde ise [16] Montgomery merdiveni türündeki RSA gerçeklemesinde ardışık olarak gelen anahtar bitlerinin birbiri ile aynı ya da farklı türde olmasına göre gerçekleştirilen işlem farklarının yakalanması hedeflenmektedir. Bu çalışmada referans bir anahtar bitine ait güç eğrisi bölütlerinin diğer bitlere ait bölütler ile çapraz ilintisi hesaplanarak, bu işlemlerin birbiri ile aynı ya da farklı türde olduklarına karar verilmektedir. Bu yöntem [16] çapraz ilinti değerlerini öncelikle yatay yönde kullanmakta ve her bir eğri için hesaplanan çapraz ilinti değerlerini toplayarak güçlendirmektedir. Wunan ve arkadaşları ise [31], bu yöntemi [16] birbiri ile daha ilintili olan güç eğri parçalarını kullanacak şekilde geliştirmiştir. Wang ve arkadaşları [59 , 16] yönteminin "seçilmiş mesaj" değerleri kullanılarak ve ilk bit referans

alınarak, daha yüksek başarımlarda RSA üs alma adımlarına uygulanabildiğini göstermiştir.

Tez kapsamında geliştirilen [16] ‘deki yöntemin daha gelişmiş bir şekli olan ve yine tez kapsamında geliştirilen [17] çalışmasında, farklı tipteki anahtar bitlerinin çapraz ilinti davranışının farklı olmasına dayanarak, tek referans bit yerine tüm bitlerin birbiri ile olan çapraz ilinti değerlerinin toplanarak kullanılabileceği gösterilmiştir. Bu yöntemle güç eğrisi bölütleri çok daha etkin bir şekilde kullanmakta olup [16] deki yöntemden daha az eğri ile sonuca ulaşılabilmektedir. Wunan ve arkadaşları tarafından gerçekleştirilen [36] çalışmasında ise benzer bir yaklaşımla, çift mesaj körleştirme önlemine sahip RSA algoritmasında, üs alma adımlarında 4 çeşit ortak modüler çarpma girdisinden bir ya da ikisinin ortak kullanan anahtar bitleri tespit edilebilmiştir. Bu çalışmada da tıpkı [16 , 17] olduğu gibi çapraz ilinti değerleri toplanarak kullanılmıştır. İlinti değerlerini hesaplamada kullanılması gereken önemli noktaların tespitinde varyans değerleri dikkate alınmış, toplamsal çapraz ilinti değerleri ise k-ortalamalar yöntemi ile sınıflandırılarak anahtar bitleri elde edilmiştir.

1.4.5. Şablon tipi güç analizi

Literatürdeki bir diğer yan kanal saldırı sınıfı da Şablon Analizi (ŞA) olarak bilinmektedir. FGA ve İGA saldırıları, basitçe HW ya da HD değerlerine dayanan tahmini güç modellerini kullanırlar. Bu modellemenin doğru şekilde yapılamadığı ya da yetersiz kaldığı durumlarda, bu yöntemler başarıya ulaşamazlar. Şablon türü yöntemlerde ise hedef cihaza saldırırken, bu hedefle özdeş ve daha önce güç tüketim karakteristiği çıkarılmış başka bir cihazın güç tüketim modeli kullanılmaktadır. Buradan anlaşılacağı üzere şablon saldırıları [33] hedef cihaz hakkında bilgi sahibi olmayı sağlayan güç modeli çıkarma evresi ve bu modeli kullanarak anahtarı elde etmeyi hedefleyen eşleştirme evrelerinden oluşur. Karakter çıkarma evresi, çok fazla ölçümle gerçekleştirilebilecek iken, şablon eşleştirme yani anahtarı elde etme evresinde çok daha az ölçüm kullanılmaktadır.

1.4.6. Frekans uzayında güç analizi

Frekans uzayında gerçekleştirilen ilk FGA-İGA tipi saldırılarda [38 , 40 - 41] frekans uzayının, yanaştırma, gürültüyü yalıtma gibi tüm avantajlarını onaylayan çalışmalar

gerçekleştirilmiştir. Bununla birlikte [38]'da frekans uzayında gerçekleştirilen güç analizi saldırılarının hedef sistemin saat frekansından bağımsız olduğu gösterilmiştir. [42 , 43]'de gerçekleştirilen saldırılarda ise frekans uzayı FGA-İGA yöntemlerinde, eğri genliklerinin yanı sıra şeklinin de önemli olabileceği tespitinde bulunulmuştur.[45 , 46]'de ise güç eğrilerinde saat frekansı ve harmoniklerine filtreme uygulanarak FGA–İGA saldırılarının başarımı artırılmaya çalışılmıştır. [39]'de frekans uzayı güç ve EM kaçakları için analitik bir model oluşturulmuş ve kaçağın aslında çalışılan saat frekansı ile doğrudan ilişkili olmadığı, devrenin destekleyebileceği en yüksek frekansa ise dolaylı bir bağımlılığın olduğu gösterilmiştir.

Tez kapsamında frekans uzayında gerçekleştirilen çalışmada önceki çapraz ilinti tabanlı güç analizi yöntemi, zaman uzayı yerine frekans uzayında gerçekleştirilmiştir. Bu çalışmada, zaman uzayı ile karşılaştırıldığında frekans uzayında daha hızlı sonuca ulaşılabilmektedir. Bunun nedenlerinden biri, tüm önceki literatürde belirtildiği gibi [38 , 40 - 41] zaman uzayında gerçekleşebilecek yanaştırma sorunlarının frekans uzayında daha az olumsuz etkilere yol açmasıdır. Bunun yanı sıra çalışmada, çapraz ilintisi hesaplarında tüm frekans bileşenleri yerine daha düşük frekans bileşenlerinin kullanılmasının başarımı artırdığı gösterilmiştir. Bu durumun en önemli nedenlerinden biri tıpkı [39]'da gösterildiği gibi FGA ve İGA tipi saldırılar için geçerli olan frekans uzayı güç tüketim modelinin ÇİA analizi için de geçerli olmasıdır. Literatürdeki çalışmalarda pratik [38] ve teorik [39] olarak gösterildiği gibi düşük frekans bantları saldırılar açısından daha önemli olabilmektedir. Sonuç olarak gerçekleştirilen çalışma ile frekans uzayı FGA ve İGA saldırıları için geçerli olan avantajlar, frekans uzayındaki ÇİA tipi saldırılar için de geçerli olmaktadır.

1.5. Geliştirilen Yöntemlerin Literatüre Katkıları

Bu bölümde tez kapsamında geliştirilen ve yayınlaştırılan "yenilikçi İGA", "şablon tipi İGA", "tek bit çapraz ilinti analizi" ve "tüm bitler çapraz ilinti analizi" çalışmalarının literatüre kazandırdığı yenilikçi yaklaşımlar yani özgün tarafları, literatürdeki mevcut yöntemlere olan üstünlükleri ve bu yöntemlerden zayıf tarafları incelenmiştir.

1.5.1. Yenilikçi İGA yöntemi

Tez kapsamında geliştirilen ve "yenilikçi İGA" olarak adlandırılan [18] yönteminde, RSA algoritmasındaki Montgomery Merdiveni üs alma adımlarında, belli bir anahtar bitinin kendinden sonra gelen bit ile aynı değeri taşıyıp taşımamasına göre işlem girdilerinin farklı belleklerden okunmasına dayanan güç kaçakları ayırt edilmeye çalışılmaktadır. Saldırıda temel olarak her biri farklı veri değerleri ile işlem görmüş pek çok RSA koşturumundan elde edilen ve yan yana belli sayıdaki bit gruplarına ait güç eğrilerinin, ilgili anahtar tiplerini içeren "tip kestirim vektörleri" ile olan ilintisi kullanılmaktadır. Gerçek güç tüketimine ait eğri matrisleri, yan yana belli sayıda eğrinin her bir ölçümden alınmış örneklerinin alt alta dizilimi ile oluşturulur. Teorik güç tüketimini temsil eden güç tüketim vektörleri ise aynı sayıda anahtar bitinin tekrarı ile oluşturulur. Her iki tipte ve doğru kestirim değerlerine sahip tip vektörlerinin, güç eğrileri ile yüksek ilintiye sahip olması beklenir. Ancak, sadece tek tip bit içeren vektörlerde bu durumun gözlenmeyip (ilinti değerlerini hesaplamak için birden fazla tip içeren vektörler gerektiğinden), tüm kestirim değerleri ile güç eğrilerinin düz bir ilinti eğrisine sahip olması beklenmektedir.

Bu yöntem aynı eğriye ait farklı işlem anlarını kullanması açısından yatay analiz grubuna girerken birden çok işlem anını farklı eğrilerde birleştirmesi açısından da dikey analiz özelliklerine sahiptir. Anahtar bitlerine ait tip vektörleri her bir anahtar bitinin kendinden önceki bit ile aynı değere sahip olup olmamasına göre 0 ya da 1 olarak temsil edilmesiyle oluşturulduğundan aslında bu değer aynı zamanda yan yana gelen anahtar bitlerinin HD değeri ile de özdeştir. Hedef Montgomery Merdiveni gerçeklemesinde, ardışık bitlerin değerinin aynı olup olmadığına dair bir kontrol de yapılmakta olup, bu tür bir kontrolden kaynaklanabilecek bir kaçak da bu yöntemle yakalanabilir. Saldırı bu yönüyle de bir tür "adres bit İGA" olarak da değerlendirilebilir

Geliştirilen bu yöntemin literatüre kazandırdığı yenilikler ve mevcut yöntemlerle karşılaştırılması şu şekildedir:

i) Güç eğrilerini hem yatay hem de dikey yönde kullandığından karma tipte bir ilintisel güç analizi olma özelliği taşımaktadır. Çalışma bu yönüyle hem

dikeydeki ilintisel güç analizi çalışmalarından [6 - 7] hem de yataydaki [28] İGA çalışmalarından ayrışmakta ve yenilikçi bir yaklaşım sunmaktadır.

Bu çalışmanın literatürdeki diğer dikey ve yatay çalışmalarla karşılaştırıldığında üstün tarafları şu şekilde sıralanabilir: Doğrudan anahtar bitlerine saldıran bu yöntem sadece yatay yönde kullanıldığında, tek bir ölçüm için anahtar bitlerindeki dörtlü grupların sayısı kadar eğri alanı mevcut olduğundan anahtar bitlerinin büyük bir yüzdesini elde etmek mümkün olmayacaktır. Ölçümlerde bulunan elektronik gürültü ve anahtarlama gürültüsü bu durumun önüne geçmektedir. Aynı anahtar bitleri ile alınmış pek çok ölçümde ise aynı işlem anlarında aynı anahtar bitleri işlenmiş olacağından, sadece dikey yönde ve doğrudan anahtar bitleri ile ilişkili bir İGA saldırısı gerçekleştirmek mümkün değildir. Eğrilerin hem yatay hem dikeyde kullanılması, her bir bit grubu için çok daha fazla eğri alanının kullanımına olanak tanıyarak verimliliği artırmaktadır. Bunun yanı sıra dikeydeki çalışmaların [6 - 7] aksine bu yöntem, aynı anahtar değeri ile elde edilmiş eğrilerde anahtar bitlerine giriş verisi üzerinde herhangi bir bilgi ya da kontrol sahibi olmayı gerektirmemektedir.

Bu çalışma karma bir yöntem olarak dikey yöndeki çalışmalarla karşılaştırıldığında, eğrileri yatayda da kullanabilmesi açısından avantajlı iken, tek bir eğri ile sonuca ulaşabilen türdeki yatay saldırılarla karşılaştırtıldığında da dez avantajlı görülmektedir. Buradaki en büyük olumsuz taraf ise tek bir eğriyle sonuca ulaşabilen yöntemlerde üs körleştirme türü önlemler devre dışı bırakılabilirken, bizim yöntemimizin üs körleştirme önlemine karşı kullanılamayacak olmasıdır. Ancak yataydaki saldırıların büyük bir kısmı da elektronik gürültüyü azaltabilmek için tamamen aynı girişler ile oluşturulmuş birden fazla eğriye gereksinim duyabilmektedir.

ii) Yatayda tek bir anahtar biti ya da bit grubuna ait işlemin alt adımlarına dallanmak yerine, birden çok eğrideki birden çok anahtar bitine ait işlem alanları doğrudan kullanılarak İGA uygulanmıştır. Aynı anda işlem görmeyen anahtar bitleri gruplar halinde elde edilmeye çalışılmıştır.

Literatürdeki yatayda İGA türü çalışmalarda genel olarak, aynı anda işlem gören anahtar bitine ya da bit gruplarına ait alt işlem alanları kullanılmaktadır. Örneğin geliştirdiğimiz bu yönteme en yakın çalışmalar olan yatayda FGA türündeki [25]

yöntem ve bu yöntemin İGA uyarlaması olarak değerlendirilen [28] yönteminde, bu tür bir yaklaşım vardır. Bu çalışmalarda üs alma döngüsünde anahtar parçası için gerçekleşen temel işlemin alt adımları kullanılmaktadır. Bizim çalışmamızda ise aynı anda işlem gören anahtar parçalarına ait alt alanlar yerine, farklı anlarda işlem gören birden fazla temel bit işlemi bir arada hedef seçilmiş ve karşılaştırmalar bu işlemler arasında gerçekleştirilmiştir. Örneğin [28] çalışması, pencere tipi bir üs alma yöntemine uygulanmış olup pencere değeri ile gerçekleştirilen uzun çarpma işlemlerine ait alt adımların HW ya da HD değerleri ile ilgili eğrilerin ilintisi hesaplanırken, bizim çalışmamızda anahtar bitlerinin doğrudan değerine bağlı değişimler yakalanmaya çalışılmıştır. Bu türdeki uzun çarpma işlemi alt adımlarına odaklanmaktaki temel zorluklardan biri bu alt alanların doğru şekilde bölütlenebilmesi ve yanaştırılmasından kaynaklanmaktadır. Bizim çalışmamızda ise [18] sadece anahtar bitlerine karşı düşecek şekilde alanların alt alta getirilmesi yeterlidir. Uzun çarpma alt adımlarının ayrıştırılması anahtar bitlerine göre ayrıştırmaya oranla daha zor bir işlemdir ve bu alt işlemlerin ayırt edilmesi her zaman mümkün olmayabilir. Ayrıca uzun çarpma işlemlerinde zaman farklarının olması da işi daha da zorlaştıracaktır. Modüler uzun çarpma işlemlerinde kelime boyu 1 bit ile 64 bit arasında değişebilmektedir. Kelime boyunun büyümesi ayırt ediciliği artırmış olsa da bu kez de kullanılabilecek alan sayısını azaltarak alt alanları kullanan yöntemlerin gücünü zayıflatırken bizim yöntemimiz bu durumdan etkilenmemektedir.

1.5.2. Şablon tipi İGA yöntemi

Tez kapsamında geliştirilen İGA türü bir diğer özgün yöntemde [19] ise, [18]'de geliştirilen yönteme güçlendirici bir karar mekanizması eklenmiştir. Bu çalışmada da [18]'de olduğu gibi, gizli anahtar bitlerinden oluşan grupların, işlem gördükleri güç eğri parçaları ile ilintisi hesaplanmakta, ancak doğru tip vektörüne karar verilirken bu ilinti değerlerinin tümü arasındaki ilişki kullanılmaktadır. Bu yöntemin arkasındaki temel fikir ise, her bir bit grubunun gerçek güç eğrisi ile olan ilintisinin, aslında bu grubun gerçek bit penceresi ile olan ilintisi ile orantılı olması gerekliliğidir. Bu yeni yönteme, ilinti eğrilerinin bir tür şablonunu kullanmasından dolayı şablon tipi İGA ismi verilmiştir. Bu yöntemin [19], anahtar bitlerini elde etmek için [18]'de önerilen yöntemden daha az eğri gerektirdiği hem teorik hem de pratik olarak gösterilmiştir. Bu çalışmanın yenilikçi ve avantajlı yönleri aşağıdaki gibi sıralanabilir:

i) Anahtar gruplarının kestirimlerine ait ilinti değerlerinde, en yüksek ilinti değeri yerine, gerçek anahtar değeri ile olan yakınlıklarına göre tüm olasılıklar için beklenen ilinti değerlerini içeren bir şablonun var olup olmadığı araştırılmakta ve doğru sonuca buna göre karar verilmektedir.

Literatürdeki yatay ya da dikey İGA çalışmaları genel olarak güç modeli ve gerçek eğrilerin ilinti değerlerini tek tek değerlendirmekte, ilinti değerinin büyüklüğüne göre doğru güç tüketim modeline karar verilmektedir. Bizim çalışmamamızda ise, doğru ve yanlış tip kestirim vektörleri arasında var olan ilinti değerlerinin, güç eğrileri arasında da bulunması gerekliliğinden yola çıkılarak, hem yanlış hem doğru kestirimlere ait ilinti eğrilerinden gelen bilgiler birleştirilmiştir. Geliştirilen bu yaklaşım işaret işlemenin farklı alanlarında da uygulanabilir olsa da [60 - 61] YKA konusunda bu tür bir İGA analizi henüz bulunmamaktadır. Ayrıca hem teorik hem pratik olarak da gösterildiği gibi bu yöntem, klasik anlamdaki kestirimlere ait eğrilerde bir tepe değer aramayla karşılaştırıldığında daha az eğri kullanılarak saldırı başarımını artırmaktadır.

1.5.3. Tek referans bit çapraz ilinti yöntemi

Tez kapsamında geliştirilen çapraz ilinti türündeki özgün yöntemlerden ilki [16] yatayda çapraz ilinti analizini toplamsal olarak kullanan öncü çalışmalardan biridir. Yöntem, referans bir anahtar bitine ait güç eğri bölütünün diğer anahtar bitlerine ait alanlar ile çapraz ilintisini değerlendirerek, her bir hedef bitin referans ile aynı ya da farklı türde olduğuna karar vermektedir. Çalışmada, çapraz ilinti değerlerini pek çok eğri kullanarak daha iyi kestirmek amacıyla biri ilintiler toplamı diğeri ise bir oylama mekanizmasına dayanan iki farklı yol önerilmiştir. Bu yöntem [16] çapraz ilinti değerlerini yatay yönde kullanıyor olması açısından [29 - 30] yöntemlerinden ayrışmaktadır. Yöntemimiz her bir eğri için hesaplanan çapraz ilinti değerlerini toplayarak güçlendirmesi açısından [5 , 58] yöntemlerinden ayrışırken, ilinti katsayılarını oylama yöntemi ile kullanması açısından da [5] yönteminden farklılaşmaktadır. Bu yöntemin literatüre getirdiği yenilikler ve üstün yönleri şu şekilde sıralanabilir:

i) Yatay yönde çapraz ilinti analizi ile ara değerlere bağlı bir çakışma değil doğrudan anahtar bitlerinden kaynaklanan işlem benzerlikleri yakalanmaya çalışılmıştır.

Önceki yatay ilinti çalışmalarında genel olarak belli bir işlem girdisinin ortak olarak kullanıldığı işlem anlarının benzerliği yakalanmaya çalışılmıştır. Bu tür bir çakışma analizi için ya aynı girdinin bir koşturumu boyunca kullanılması ya da birbiri ile belli bir ilişkisi olan girdi değerleri kullanılarak oluşturulan iki farklı eğrideki alanların birbiri ile karşılaştırılması gerekmektedir. Bizim çalışmamızda ise, bit işlemlerinde kullanılan bir ortak ara değer bulunmayıp, bir şekilde anahtar bitlerinin kullanımından kaynaklanacak işlem/adres benzerliklerinin yakalanması hedeflenmiştir. Bu tür bir yöntem referans değer ile ortak ara değer kullanımının bulunması halinde zaten bu durumları da tespit edebilecektir. Örneğin yöntemin Montgomery Merdiveni üs alma gerçeklemesine uygulanmasında anahtar bitlerinin tipinin tespit edilmesi için, anahtar bitlerini kontrol eden işleminin kendisi ya da bu kontrol sonucuna göre gerçekleştirilecek olan farklı bellek alanlarından veri okumadan kaynaklanan farklar yakalanabilmektedir. Bununla birlikte, hep çarpma yapmaya dayanan ikilik üs alma yöntemine yapılan uygulamada ise değeri sıfır olan anahtar bitlerden sonra gelen bit için gerçekleştirilen kare alma işlemine ait girdi değerlerinin değişiminin yani HD değerinin “0” olmasından kaynaklanan bir güç tüketim farkı yakalanabilmektedir.

Sonuç olarak bir şekilde anahtar bitlerine doğrudan bağlı işlem farkları referans bir işlem ile benzerlik kurularak anlaşılmaya çalışılmakta ve kullanılan işlem girdisi için herhangi bir kontrol gerekmemektedir.

ii) Pek çok eğri için yatayda hesaplanan çapraz ilinti değerleri, bir sayaç mekanizması kullanılarak ya da doğrudan toplanarak birleştirilmiştir.

Literatürdeki yataydaki çapraz ilinti saldırıları genel olarak tek bir eğri ile çalıştırılmaktadır. Tek bir eğri ile çalışıldığında sinyal gürültü oranını azaltabilmek için daha fazla alt eğri bölütü kullanımına gerek duyulmaktadır. Ancak bu işlem için alt işlem parçalarının doğru bir şekilde ayrıştırılabilmesi gerekmektedir. Bu durum da zaten gürültü nedeni ile her zaman sağlıklı bir şekilde gerçekleştirilemez. Üstelik bu yöntemler çarpıcı boyundan ve anahtar boyundan çok fazla etkilenmektedir. Bu yöntemlerin avantajı üs körleştirme karşı önlemi uygulansa bile çalışabilecek olmalarıdır. Ancak üs körleştirme olmayıp modülüs ya da mesaj körleştirme gibi önlemlerin kullanıldığı durumda eğer tek bir eğri ile elde edilebilecek sinyal gürültü oranı yeterli olmazsa bu yöntemler işe yaramayacaktır. Bunun yanı sıra bu

yöntemlerde doğrudan anahtar bitine bağlı kısa sürede gerçekleşen işlemlere odaklanılması mümkün değildir. Bizim yöntemimizde ise mesaj ya da modülüs körleştirme gibi önlemler bulunsa bile, her bir bit için istenildiği kadar eğri değerleri kullanılarak çapraz ilinti değerleri hesaplanmakta ve hesaplanan bu değerler bir oylama mekanizması ya da her bir değerin toplanması ile birleştirilerek güçlendirilmektedir. Sinyal gürültü oranını artırmada kullanılan temel mekanizma ise birden fazla işlem koşturumundan elde edilen ilinti değerlerinin birleştirilebilmesidir.

1.5.4. Tüm bitler çapraz ilinti yöntemi

Tez kapsamında çapraz ilinti türünde geliştirilen bir diğer çalışmada ise [17], farklı bit türlerinin birbirinin tersi çapraz ilinti davranışı göstermelerine dayanarak [16] teki yöntem, tüm bitlerin birbiri ile olan çapraz ilinti değerlerini kullanacak şekilde geliştirmiştir.

i) Literatürde, yatayda tüm anahtar bitlerinin çapraz ilinti değerlerinin toplanarak kullanımına dair bir örnek bulunmamaktadır. Çalışma bu yönüyle hem özgün hem de eğrileri daha verimli şekilde değerlendiren bir çalışma özelliğindedir.

Aslında tüm bitler ÇİA yönteminin kullandığı temel davranış özelliği, pek çok üs alma algoritmasında bulunabilir. Örneğin çapraz ilinti kullanımına olanak sağlayan bu davranış hem Montgomery Merdiveni tabanlı hazır ASIC devresinde hem de "her zaman çarp" türündeki "ikilik üs alma devresinde" bulunmaktadır. Bu yöntemde belli bir eğride bulunan tüm anahtar bitlerine ait eğri parçaları kullanılmış olduğundan ve toplanan güç eğrilerinde kullanılabilen toplam eğri bölütü sayısı artmış olmaktadır. Bu durum hem elektronik gürültü hem anahtarlama gürültüsünün etkisi azaltacağından, kullanılan eğri sayısı ve değeri kestirilebilen toplam bit sayısı açısından bu yöntem daha iyi bir performansa sahiptir. .

Bu yöntem ASIC devreye uygulanan tek referans bite dayalı yöntem [16] ile karşılaştırıldığında, pratikte %75 oranında daha az eğri kullanımını gerektirmektedir. FPGA uygulamalarında ise tek ve tüm bitler yöntemlerinin eğri sayısı açısından başarımları bu kadar farklılık içermese de tüm bitler çapraz ilinti yöntemi ile daha fazla anahtar bitinin değeri elde edilebilmektedir. FPGA devresinde tüm bitler yönteminin eğri sayısı açısından çok önemli bir fark yaratmamasının temel nedeni, bu devreye ait

güç eğrilerinin daha az gürültü içerecek koşullarda alınmasından kaynaklanabilir. Sonuç olarak gürültünün yüksek olduğu eğrilerde sonuca ulaşmak için tüm bitler yöntemi daha önemli olmaktadır.

1.6. Geliştirilen yöntemlerin literatürle karşılaştırılması

Tez kapsamında geliştirilen çalışmalar, sahip oldukları özgün yönlerinin ve önceki çalışmalara göre güçlü ve zayıf yönlerinin daha iyi anlaşılabilmesi açısından Tablo 1.1.'de literatürdeki temel yöntemlerle karşılaştırılmıştır. Karşılaştırmada, saldırının özgünlüğü, karşı önlemlere olan etkinliği ve uygulama kolaylığı açısından önemli olduğu düşünülen ve her bir birinin ne anlama geldiği aşağıdaki gibi açıklanan kıstaslar kullanılmıştır.

- Kullanılan Temel Analiz Yöntemi: Gerçekleştirilen saldırı yönteminde çakışma analizinin alt türleri ya da İGA, FGA gibi yöntemlerden hangisinin kullanıldığı açıklanmaktadır.
- Saldırı Yönü: Literatür özetinin verildiği 1.4 bölümünde de bahsedildiği gibi saldırı türleri aynı güç eğrisindeki farklı işlem analarını kullandığında yatay, birden çok eğride aynı işlem anlarını kullandığında dikey analiz adını almaktadır. Eğrileri her iki yönde kullanan çalışmalar ise karma çalıma sınıfına girmektedir.
- Hedef alan seçim hassasiyeti: Bazı YKA yöntemlerinde hedef eğri alanı, kullanılan yöntem gereği kendiliğinden ortaya çıkarken, bazı saldırılarda sınırlarının hassas bir şekilde belirlenmesi önemli olabilmekte, hatta eğri alanı gerekenden geniş ya da dar alındında, saldırı uygulanamaz hale gelebilmektedir. Yatayda gerçekleştirilen yöntemler genel olarak eğri parçasının kapsamı konusunda daha hassas olurken, dikey yöntemlerde bu durum daha az önem taşımaktadır. Saldırıda örneğin yataydaki uzun çarpma adımlarının çıkarılarak yanaştırılması gerekmekte ise alan hassasiyeti "çok önemli", daha geniş kapsamlı bir işlem olan "çarpma – kare alma" alanlarını ayırt etmek yeterli ise önemli, dikey saldırılarda olduğu gibi aslında alanların çıkarılması değil sadece belli seviyede yanaştırılması gerekmekte ise alan hassasiyeti az önemli olarak sınıflandırılmıştır.
- Açık veri bilgisi /kontrolü: Saldırıyı gerçekleştirmek için, algoritmaya verilen açık veride saldırganın bilgi ya da kontrol sahibi olması gerekebilir. Örneğin birden fazla eğrinin kullanıldığı bazı çakışma saldırılarında, her bir koşturuma ait girdiler

arasında bir ilişki olması (ör. x ve x^2) söz konusu olabileceği gibi, klasik türdeki dikey İGA saldırılarında, girişlerin bilinip ara değerlere ait olası kestirimlerin yapılması gerekmektedir. Analizin gerçekleştirilmesi için bu tür ön koşullarının olması, özellikle açık verinin kendisinin de hedef olduğu, ya da veri girişine rastgele dolgulama yapıldığı durumlarda saldırının uygulanabilirliğini ortadan kaldıracaktır.

- Çapraz ilinti birleştirme yöntemi: Bu özellik birden fazla eğri kullanan çapraz ilinti tabanlı yöntemler için geçerli olup, her bir eğriden elde edilen çapraz ilinti değerinin birleştirme şekline göre bir sınıflandırma yapmaktadır. Yatay ya da dikeydeki eğri parçalarından elde edilen her bir ilinti değeri, tüm ilinti değerlerinin ortalamasının bir eşikle karşılaştırılması, her bir ölçüme ait ilintilerin ayrı ayrı değerlendirerek bir sayaç değerinin artırılması gibi yöntemlerle birleştirilebilmektedir.
- Eğri sayısı: Klasik FGA ve İGA türü yöntemler, aynı anahtar değeri için birden çok ve farklı veri girişi ile işlem görmüş eğri kullanımı gerektirirken yatay saldırıların bir kısmında tek ya da karşılaştırma yapmak amaçlı iki eğrinin kullanımı yeterli olabilmektedir. Ancak anahtar bitlerine ait alt işlem alanlarının kullanılmadığı yatay saldırılarda birden fazla eğri kullanımı zorunlu olmaktadır. Doğrudan adres bitlerine saldıran yatay analiz türleri de en hem elektronik gürültü hem veri değişiminden kaynaklanan anahtarlama gürültüsünü azaltmak adına pek çok eğriye gereksinim duyabilmektedir. Bu bağlamda tek ölçümle sonuca varabilen yöntemler özellikle üs körleştirme olarak adlandırılan karşı önlemleri geçersiz kılmaları açısından önemli olmaktadır.
- Etkisiz kıldığı karşı önlemler: Geliştirilen çakışma, FGA, İGA türü güç analizi türlerine karşı literatürde mesaj körleştirme [3 , 49] , modülüs körleştirme [62] ve üs körleştirme [49 , 62 - 64] olarak adlandırılan ve algoritma seviyesinde çalışan karşı önlemler kullanılabilmektedir. Açık verinin bilinmesi ve kontrolünü gerektiren önlemler genel olarak mesaj/modülüs körleştirme yöntemleri karşısında başarısız olurken, doğrudan anahtar bitine, anahtar adresine odaklanan yöntemler bu önlemlerden etkilenmemektedir. Yatay yönde çalışan ve tek ölçümle sonuca gidebilecek türdeki yöntemler ise mesaj/modülüs ve üs körleştirme karşı önlemini etkisiz kılmaları açısından ön plana çıkmaktadır.

Tablo 1.1. Geliştirilen yöntemlerin literatürle karşılaştırılması

	Tek Referans Çapraz İlinti[16]	Tüm Bitler Çapraz İlinti[17]	Wunan [31]	Wang [59]
Kullanılan Temel Analiz Yöntemi	Çakışma/çapraz ilinti	Çakışma/ çapraz ilinti	Çakışma/ çapraz ilinti	Çakışma/ çapraz ilinti
Saldırı Yönü	Yatay	Yatay	Yatay	Yatay
Alan Hassasiyeti	Önemli	Önemli	Az önemli	Önemli
Açık veri kontrolü/ Bilgisi	Gereksiz	Gereksiz	Gereksiz	Gerekli
Aynı anda işlem görmeyen hedef bit	Yok	Yok	Yok	Yok
Çapraz ilnti birleştirme yöntemi	Toplama/ sayaç	Toplama	Toplama/ sayaç	Toplama/ sayaç
Eğri sayısı	Birden fazla	Birden fazla	Birden fazla	Birden fazla
Etkisiz kıldığı karşı önlemler	Mesaj/ modülüs körleştirme	Mesaj/ modülüs körleştirme	Mesaj/ modülüs körleştirme	-
	Witterman [30]	Bauer [58]	Clavier [28]	Walter [30]
Temel Analiz Yöntemi	Çakışma/Çapraz ilinti	Çakışma/İGA	İGA	Çakışma/FGA
Saldırı Yönü	Dikey	Yatay	Yatay	Yatay
Alan Hassasiyet	Önemli	Önemli	Çok önemli	Çok önemli
Açık veri kontrolü/ Bilgisi	Gereksiz	Gerekli	Gerekli	Gereksiz
Aynı anda işlem görmeyen hedef bit	Yok	Yok	Yok	Yok
Çapraz ilnti birleştirme yöntemi	Sayaç	Yok	Yok	Yok
Eğri sayısı	Birden fazla	Tek/birden fazla	Tek	Tek
Etkisiz kıldığı karşı önlemler	Mesaj, modülüs körleştirme	Mesaj, modülüs körleştirme	Mesaj, modülüs, üs körleşirme	Mesaj, modülüs, üs körleşirme
	Yenilikçi İGA [18]	Şablon İGA[19]	E. Brier[6]	Messenger[5]
Temel Analiz Yöntemi	Çakışma/İGA	Çakışma/İGA/Şablon	İGA	FGA
Saldırı Yönü	Yatay ve Dikey	Yatay ve Dikey	Dikey	Dikey
Alan Hassasiyeti	Önemli	Önemli	Az Önemli	Az Önemli

Tablo 1.1. (devam)

Açık veri kontrolü/ Bilgisi	Gereksiz	Gereksiz	Gerekli	Gerekli
Aynı anda işlem görmeyen hedef bit	4 bit	Yok	Yok	Yok
Eğri sayısı	Birden fazla	Birden fazla	Birden fazla	Birden fazla
Etkisiz kıldığı karşı önlemler	Mesaj,Modülüs körleştirme	Mesaj,Modülüs körleştirme	-	-

2. KULLANILAN ÖLÇÜM VE ANALİZ DÜZENEKLERİ

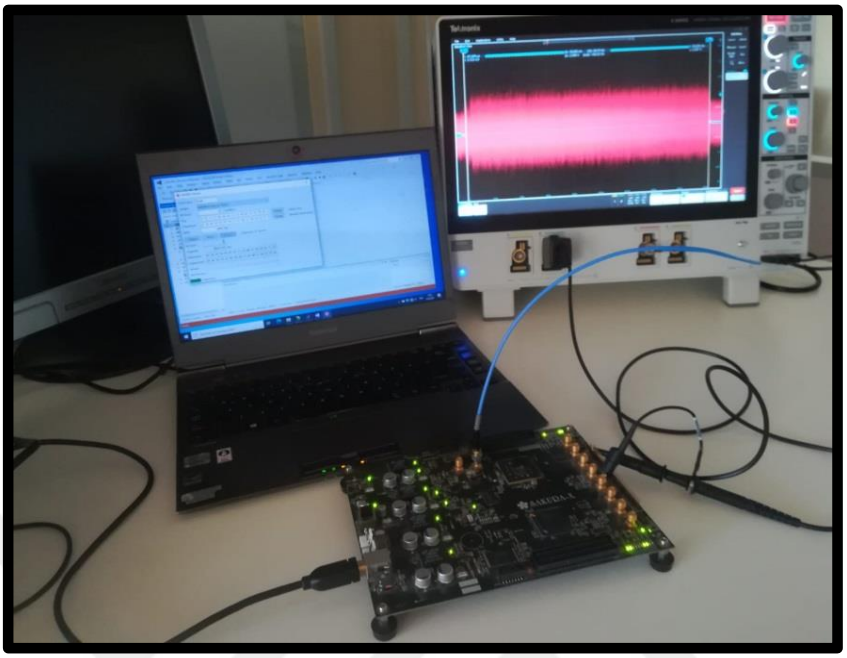
Bu bölümde hedef devrelerden ölçüm almak için ne tür yazılım ve donanımlardan oluşan düzeneklerin kullanıldığı açıklanmıştır. Ayrıca RSA gerçeklemelerinin bulunduğu hedef devrelere ait ve YKA açısından önemli olan özelliklerden de yine bu bölümde bahsedilmiştir. Analizlerde kullanılan gerçek devreler ve güç ölçümlerinin yanı sıra, benzetim tabanlı eğrilerin nasıl elde edildiği de bu bölümde ayrıntılandırılmıştır. Düzeneklerde kullanılan yazılım, donanım ve hedef develerin bir kısmı hazır olup bir kısmı tez kapsamında geliştirilmiştir.

2.1. Ölçüm Düzenek ve Yazılımları

Hedef devrelerden ölçümleri kaydetmek için öncelikle bu devrelere komut yollayarak RSA işlemini çalıştıran ve bu sırada da osiloskoba komut göndererek ölçümleri kaydeden programlar gerekmektedir. Hedef devrelerden biri olan Sakura-X kart üzerindeki RSA modüllerinin çalışması için, kart geliştiricisi tarafından sağlanan ve aslında AES devrelerini çalıştırmak üzere C# dilinde geliştirilmiş ‘Sakura Checker’ isimli programda [68] değişiklikler yapılmıştır. Yapılan eklenti ile program RSA gerçeklemesini çalıştırabilir hale getirilmiştir. Algoritmanın koşturumu sırasında oluşan güç ve EM ölçümleri ise Tektronix DPO 7064 Model 8 bit osiloskop ve Tektronix MSO64 model 12 bit sayısal osiloskoplar ile alınmıştır. Hedef ASIC devresinden ölçüm almak için VDD hattına bağlı 1 ohm’luk direnç ve farksal gerilim probu kullanılmıştır. Sakura karttaki FPGA devresi ölçümleri ise kart üstüne konmuş ölçüm noktasına bir SMA kablo bağlanarak ölçülmüştür. İlk gerçekleme için 100 MHz, ikinci gerçekleme için ise 1.562 GHz örnekleme hızları kullanılmıştır. Şekil 2.1’de Sakura kartından ölçüm almak üzere kullanılan düzenek görülmektedir.

Bu devrelerden ölçüm almak ve kaydetmek amacıyla bir Visual C++ uygulaması geliştirilmiştir. Bu uygulama osiloskop ile haberleşmekte, osiloskoba “tetik bekle”, “aldığın veriyi dosyaya yaz” vs. gibi komutlar göndererek Sakura kart ve ASIC devre üzerindeki RSA algoritmaları koşarken oluşan güç eğrilerini bir bilgisayara

kaydetmektedir. Şekil 2.2.'de Sakura kartı üzerinden, MSO64 model osiloskop ile toplanmış bir güç ölçümünü gösteren osiloskop ekran alıntısı görülmektedir.



Şekil 2.1. Sakura kartından ölçüm alınan düzenek

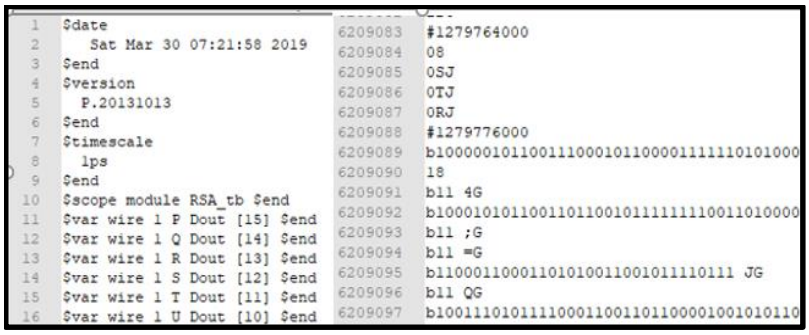
Şekil 2.2. RSA işlemine ait osiloskop ekran görüntüsü

Bu resimde mavi ile görülen kanal (ch2) tetik amacıyla kullanılan sinyale ait olup bir RSA işlem süresi boyunca mantıksal 1 değerinde kalmaktadır. Pembe ile görülen sinyal ise (ch3), kart üzerindeki SMA bir bağdaştırıcıdan ölçülen güç tüketim eğrisidir.

Şekilden anlaşılacağı üzere RSA işleminin olduğu anlarda güç eğrisinde belirgin bir şekilde yükselme oluşmaktadır.

2.2. FPGA Güç Tüketimi Benzetim Yazılımları

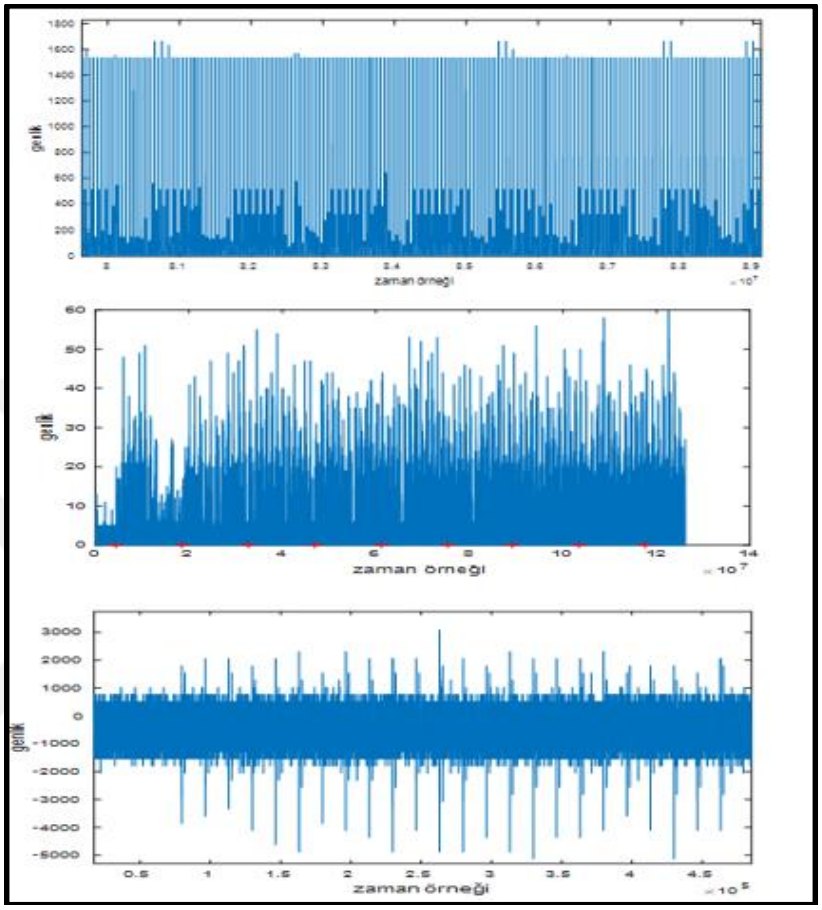
Bu bölümde, analiz girdilerinden biri olan Verilog modüllerinin, davranışsal ya da yerleştirme-bağlama (place-route) seviyesi benzetimi ile oluşturulan VCD dosyalarından, bu devreye ait güç tüketim eğrilerinin nasıl elde edildiğinden bahsedilecektir. ISIM ya da Modelsim gibi HDL benzerim programları kullanılarak, Verilog modüllerinin davranışsal seviyesinden en son FPGA üzerine yerleştirilmesi sonrasına ait olan yerleştirme-bağlama (place-route) "seviyesine kadar olan tüm ara seviyelerine ait "sinyal değişimini (switching activity)" gösteren benzetim (simülasyonlar) işlemleri yapılabilmektedir. Bu benzetim işlemleri sonucunda, eğer ilgili komutlar test betiğine eklenirse, "VCD" uzantılı bir dosya elde edilebilmektedir. Bu dosya ASCII biçiminde olup, başlık bilgisi, değişken tanımları ve her bir benzetim adımında sinyallerin hangi değerlere sahip olduğu bilgisini içermektedir. Hedef gerçeklemede bulunan sinyal sayısına göre bu dosya oldukça büyük boyutlara ulaşabilmektedir.



Şekil 2.3. Örnek vcd dosyası alanları

CMOS devrelerdeki anlık güç tüketimi, bir anda değişen sinyallerin sayısı ile orantılı olduğundan, çalışma kapsamında, hedef RSA devresinin benzetim tabanlı anlık güç tüketimini elde etmek için VCD dosyaları kullanılmıştır. VCD dosyasının içeriği Matlab betikleri ile ayıklanarak, her bir benzetim adımındaki sinyal değişimleri hesaplanmış ve bu değişime dayanan anlık güç tüketimini temsil eden güç eğrisi benzetimleri elde edilmiştir. VCD dosyasını oluşturmak için Üs alma devresine ilgili girişleri vermek üzere geliştirilen test betiği içerinde `timescale 1ns/1ps" değeri örnekleme değeri olarak verilmiştir. Saat darbesi periyodu "parameter PERIOD = 50"

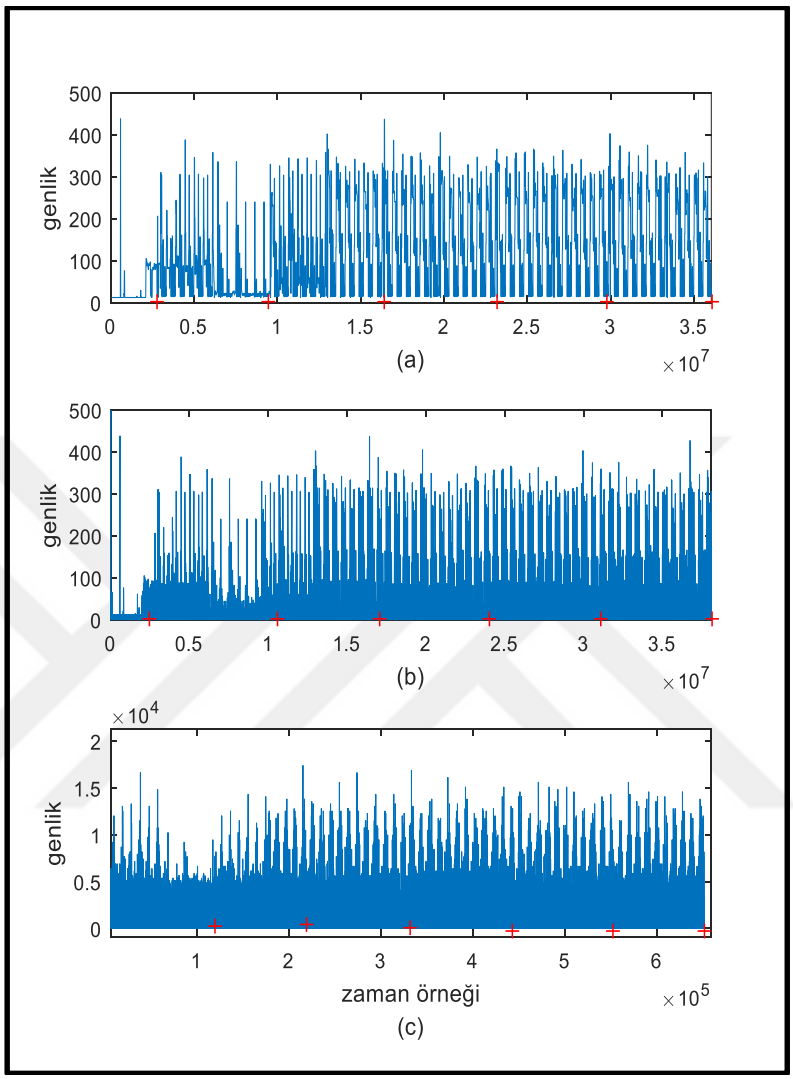
ile belirtilmiştir. Test betiğinin bulunduğu en üst modülden itibaren tüm sinyallerin dosyaya kaydedilmesi için. “$dumpfile ("Dumpdosya.txt") ve $dumpvars (0, RSA_tb.RSA_girismodulu1)” komutları dosyaya eklenmiştir.



Şekil 2.4. RSA 24 bit gerçeklemesine ait benzetim ve gerçek güç eğrileri

Şekil 2.3.’de VCD dosyasının başlık ve gövdesinden örnek alanların içeriği görülmektedir. VCD dosyasında, en başta her bir sinyal ve kütüğe verilen isimler yer almaktadır. Bu dosyada bir boyutlu sinyallerin her biri ve çok boyutlu sinyallerin her bir biti için ayrı bir isim kullanılmaktadır. Kütüklerde ise bu durum söz konusu değildir, çok boyutlu ve tek boyutlu kütüklerin, tüm bitlerini içerecek şekilde tek bir ismi bulunmaktadır. VCD dosyasını işlerken, önce değişkenlerin isimleri dosyadan elde edilerek bir hücre dizisine atılmış, sonrasında ölçülen iki örnek anı arasındaki tüm değerler ilgili sinyallere eklenmiştir. Sonrasında tüm sinyal ve kütükler göz önüne alınarak, her bir örnek anının önceki örnek anına göre değişim sayısı yani “Hamming

Distance” değerleri hesaplanmıştır. Bu değerler, ilgili örnek anına ait anlık güç tüketim değeri kabul edilerek benzetim ortamı güç eğrisi oluşturulmuştur.



Şekil 2.5. RSA 192 bit gerçeklemesine ait benzetim ve gerçek güç eğrileri

Benzetim eğri oluşturma işine öncelikle ISE’de ISIM benzetim ortamı kullanılarak başlanmıştır. Şekil 2.4’de sırasıyla ISE ve Vivado’dan post-route seviye benzetimle elde edilen 24 bit RSA devresine ait VCD eğrilerinin işlenmesiyle oluşturulmuş güç eğrileri ve gerçek güç eğrisi görülmektedir. Gerçek güç eğrilerinin 20 ortalaması kullanılmıştır ve gerçek eğride hemen hemen hiçbir ayrıntı görülememektedir. Şekil 2.5‘de ise 192 bit RSA’nın Vivado benzetim uygulamasından elde edilmiş davranışsal, “post place-route” seviye benzetim güç eğrileri ve geçek güç eğrisi görülmektedir. Davranışsal seviye eğri ve gerçek güç eğrisi şekilsel olarak benzemektedir ve saldırılarda da tamamen olmasa da benzer davranışlar göstermişlerdir.

2.3. Hedef RSA Devreleri

Bu bölümde, saldırılarda kullanılan biri hazır ASIC devre diğeri FPGA tabanlı devrenin YKA açısından önem taşıyan gerçekleme özellikleri tanıtılmıştır

Modüler çarpım işlemlerini hızlı bir şekilde gerçekleştirmek için geliştirilmiş etkili yöntemlerden birisi Montgomery çarpımıdır [70]. Hedef devrelerde de, aşağıda ayrıntıları verilen bu yöntem kullanılmıştır. RSA algoritmasının temel işlem döngüsü olan üs alma adımlarında kullanılabilecek, soldan sağa ikilik üs alma, sağdan sola ikilik üs alma, Montgomery merdiveni [52], sabit genişlikli ve kayan pencere yöntemleri gibi yöntemler mevcuttur. Aşağıdaki bölümlerde hedef devrelerde kullanılan iki temel üs alma yöntemi olan Montgomery Merdiven ve ikilik üs alma yöntemlerinin ayrıntıları verilmiştir.

2.3.1. Geliştirilen FPGA RSA Devresi

Saldırı hedefi olarak kullanılan devrelerden biri Sakura-X isimli kartta bulunan Kintex7 FPGA'inde çalıştırılmak üzere tez kapsamında geliştirilmiş devredir. Sakura kartı YKA analizlerini gerçeklemek amaçlı geliştirilmiş, üzerinde kripto işlemleri ve ara yüz işlemlerini koşturmak için iki ayrı FPGA barındıran bir karttır. Sakura kart kripto FPGA'i görevi gören Kintex7'de koşmak üzere Verilog dilinde geliştirilen bu gerçekleme, "Montgomery Çarpıcısı" ile modüler indirgeme yapmaktadır. Bu devrede üs alma yöntemi olarak "Hep çarp türü ikilik üs alma yöntemi" kullanılmaktadır. Bu gerçeklemelere ait işlem akışları aşağıdaki alt bölümlerde açıklanmıştır.

2.3.1.1. Montgomery modüler çarpıcısı

Montgomery çarpımı [70], modüler üs alma döngülerinin hızlı şekilde gerçekleştirilebilmesi için geliştirilmiş modüler çarpım yöntemlerinden biridir. Montgomery çarpımı "Montgomery kalanı" üzerinde işlem yapmayı gerektirir. Çarpım girdilerinin, işlemin başında Montgomery uzayına alınması, işlem sonunda da Montgomery uzayından çıkarılması gerekmektedir. Ancak Montgomery çarpımı, üs alma algoritmasında kullanıldığı zaman, bu işlemlerin üs almadan önce ve sonra birer kez yapılması yeterlidir. Bunun yanı sıra, modüler indirgeme adımları için gereken modülüse bölme işlemini (normalde modülüsün pek çok kez çıkarılması ile

gerçekleştirilir) modülüsün belli bir katı ile toplama ve basit bir sola kaydırma işlemine dönüştürmektedir. .

Algoritma 2.1. Montgomery çarpıcısı ile Montgomery kalanını hesaplama

GİRDİLER r=2^n m = $m_n \dots m_2 m_1 m_0$:, $2^{n-1} < m < 2^n$ $x = x_n \dots x_2 x_1 x_0$,taban, x< m ÇIKTILAR $\overline{x}$,
$\overline{x}$ =MontMul(x,1) $= x * r \pmod m$ Return $\overline{x}$

Algoritma 2.2. Bir kerede 1-bit indirgeyen n-bitlik Montgomery çarpımı

GİRDİLER $r = 2^n$, m = $m_n \dots m_2 m_1 m_0$, $2^{n-1} < m < 2^n$ $r^{-1} = 2^{-n} \bmod m$, $x = x_n \dots x_2 x_1 x_0$ $y = y_n \dots y_2 y_1 y_0$, ÇIKTILAR: $z = z_n \dots z_2 z_1 z_0$ $z = x * y * r^{-1} \pmod m$
z= 0 for i= 0 : n-1 z = z + x_i*y If(z mod 2 ==1) z=z+m $z = z \div 2$ If z≥ m then z=z-m Return z

Boyu "n" bit olan bir "a" sayısının, m modülüs değeri, ve r=. 2^n için m-Montgomery kalanı, $\bar{a} = a.r \bmod m$ olarak tanımlanır. Burada 0 < a < m-1 olmak üzere, tam sayılar ve m-Montgomery kalanı arasında birebir eşleşme söz konusudur. Montgomery çarpımı ise x, y değerlerinin Montgomery kalanının giriş olarak verilmesi durumunda, $z = x.y$ değerinin Montgomery kalanını üreten $\bar{z} = \text{MontMul}(\bar{x}, \bar{y}) = \bar{x}.\bar{y}.r^{-1} \bmod m$ İşlemi olarak tanımlanır. Montgomery kalanının hesaplanması Algoritma 2.1.'de verilmiştir. İkilik tabanda n-bitlik girdilerle her döngü adımında bir bit indirgeme yapan temel Montgomery algoritması Algoritma 2.2.' de verilmiştir. Montgomery algoritması, kelime üzerinde işlem yapacak şekilde de çalışabilmektedir. Algoritma 2.3.'de n- bitlik k-kelimelik girdiler üzerinde indirgeme yapan Montgomery çarpım adımları görülmektedir. Tez kapsamında geliştirilen "Montgomery" modülünde de bu

yöntem kullanılmıştır. Burada hesaplanan n=32 bitlik kelime boyu, k=6 kelime sayısı olmak üzere 192 bitlik giriş değerleri ile işlem yapılmaktadır. Modülüs "M" değeri de k kelimeden oluşmaktadır. İşlemlerde elde edilen 32*192 bitlik ara basamak değeri olan $x_i * y'$değerlerinin $2^{-32} \pmod M$ 'ye bölünmesi ile Montgomery çarpımı gerçekleştirilmektedir. Her bir adım 6 kez tekrarlanarak, aslında $x * y * R^{-1} \pmod M = x * y * 2^{-192}$ değeri hesaplanmış olmaktadır.

Algoritma 2.3. Bir kerede n-bit indirgeyen k-kelimelik Montgomery çarpımı

GİRDİLER: $M = M_{k-1} \dots M_2 M_1 M_0$, $M_i = m_{n-1} \dots m_2 m_1 m_0$, $2^{nk-1} < M < 2^{nk}$ $r = 2^n$, $R = r^k = 2^{nk}$ $R^{-1} = 2^{-nk}\ mod\ M$ Mu= -(1/M_0) (mod R) $X = X_{k-1} \dots X_2 X_1$, $X_i = x_{n-1} \dots x_2 x_1 x_0$ $Y = Y_{k-1} \dots Y_2 Y_1 Y_0$, $Y_k = y_{n-1} \dots y_2 y_1 y_0$ ÇIKTILAR: $Z = X * Y * R^{-1}\ mod(M)$
Z = 0 for i= 0 : k-1 Z = Z + X_i*Y $T = Z[31:0] * Mu$ $T_0 = T[31:0]$ $Z = Z + T_0 * M$ $Z = Z * 2^{-n}$ If $Z \geq M$ then $Z = Z - M$ Return Z

2.3.1.2. Soldan sağa ikilik üs alma devresi

RSA algoritmasındaki üs alma adımlarını gerçekleştirmek amacıyla yaygın olarak kullanılan yöntemlerden biri, anahtar bitlerini en yüksek anlamlıdan en düşük anlamlıya doğru tek tek işleyen "soldan sağa ikilik üs alma algoritmasıdır". Bu algoritmada her anahtar biti için kare alma işlemi yapılmakta olup, eğer anahtar bitinin değeri "1" ise ek olarak da çarpma işlemi gerçekleştirilmektedir. Bu haliyle algoritma anahtar bitine bağlı işlem farkı içermesinden dolayı kalıtsal bir BGA kaçağına sahiptir. Algoritmaya ait akış Algoritma 2.4.'de görülmektedir. Burada M, boyu p bit olan modülüs değeri, Y veri girişini temsil eden taban değeri, d ise anahtar değerini temsil eden üs değeridir. "MontMul" kısaltması Algoritma 2.3.' de verilen Montgomery çarpımını ifade etmektedir.

Algoritma 2.4. Soldan sağa ikilik üs alma döngüsü

GİRDİLER:
$M = m_{p-1} \dots m_2 m_1 m_0$
$R = 2^p$
$R^{-1} = 2^{-p} \bmod M$
$Y = y_{p-1} \dots y_2 y_1 y_0$
$d = d_{p-1} \dots d_2 d_1 d_0$
ÇIKTILAR:
$S = Y^d \bmod(M)$

$\overline{Y} = \text{MontMul}(Y, R^2)$
$\overline{S} = \text{MontMul}(1, R^2)$
for i= p-1 :1 : 0
  $\overline{S}$ = MontMul ($\overline{S}$, $\overline{S}$)
  If($d_i == 1$) $\overline{S}$ = MontMul ($\overline{S}$, $\overline{Y}$)
S= MontMul ($\overline{S}$,1)
Return S

Algoritma 2.4.'de verilen standart ikilik üs alma yönteminde, normalde tüm anahtar bitleri için kare alma işlemi ve sadece anahtar değerindeki bir değerli bitler için çarpma işlemi yapılmakta ve bu durum BGA türü kaçaklara neden olmaktadır.

Algoritma 2.5. Soldan sağa hep çarp ikilik üs alma döngüsü

GİRDİLER:
$M = m_{p-1} \dots m_2 m_1 m_0$
$R = 2^p$
$R^{-1} = 2^{-p} \bmod M$
$Y = y_{p-1} \dots y_2 y_1 y_0$
$d = d_{p-1} \dots d_2 d_1 d_0$
ÇIKTILAR:
$S = Y^d \bmod(M)$

$\overline{Y} = \text{MontMul}(Y, R^2)$
$\overline{S} = \text{MontMul}(1, R^2)$
for i= p-1 :1 : 0
  Skare = MontMul ($\overline{S}$, $\overline{S}$)
  Scarp=MontMul(Skare , $\overline{Y}$)
  If($d_i == 0$) $\overline{S}$=Skare
  Else $\overline{S}$=Scarp
S= MontMul ($\overline{S}$,1)
Return S

Geliştirilen devrede BGA kaçaklarını engellemek amacıyla, standart ikilik üs alma algoritmasından türetilmiş ve her anahtar biti için hem çarpma hem kare alma yapan "hep çarpma türü ikilik üs alma algoritması" kullanılmıştır. Bu algoritmada, değeri 0 olan anahtar bitleri için yapılan gereksiz çarpma işleminin sonucu kullanılmamaktadır Bu şekilde oluşan güç eğrisinde BGA açıklığının oluşması engellense de FGA ve ÇİA türü açıklıkların önüne geçilmiş olmamaktadır. Bu algoritmanın matematiksel ifadesi,

gerçekleme ayrıntılarını da verecek şekilde Algoritma 2.5.'de görülmektedir. Bu algoritmada da yine M modülüs, d anahtarı temsil eden üs değeridir. $\overline{Y}$ değeri ise veri girişine karşılık düşen taban değeri Y'nin Montgomery kalanını ifade etmektedir.

Algoritma 2.5.'de verilen üs alma akışında, öncelikle "veriyi" temsil eden Y taban değerinden, Y'nin M-Montgomery kalanı olan $\overline{Y}$ değerinin bulunması, bir diğer değişle de verinin Montgomery uzayına alınması işlemi yapılmalıdır. Bu işlem Montgomery çarpıcısı kullanılarak Algoritma 2.1.'de verilen algoritma ile gerçekleştirilmektedir. Algoritmadaki döngü adımlarında, Montgomery çarpıcısına giriş olarak mesajın ve "1 değerinin" "Montgomery kalanı" verilmektedir. Böylece döngünün her bir "i numaralı" adımında $Y^i \bmod(R)$ ara değerleri hesaplanmaktadır.

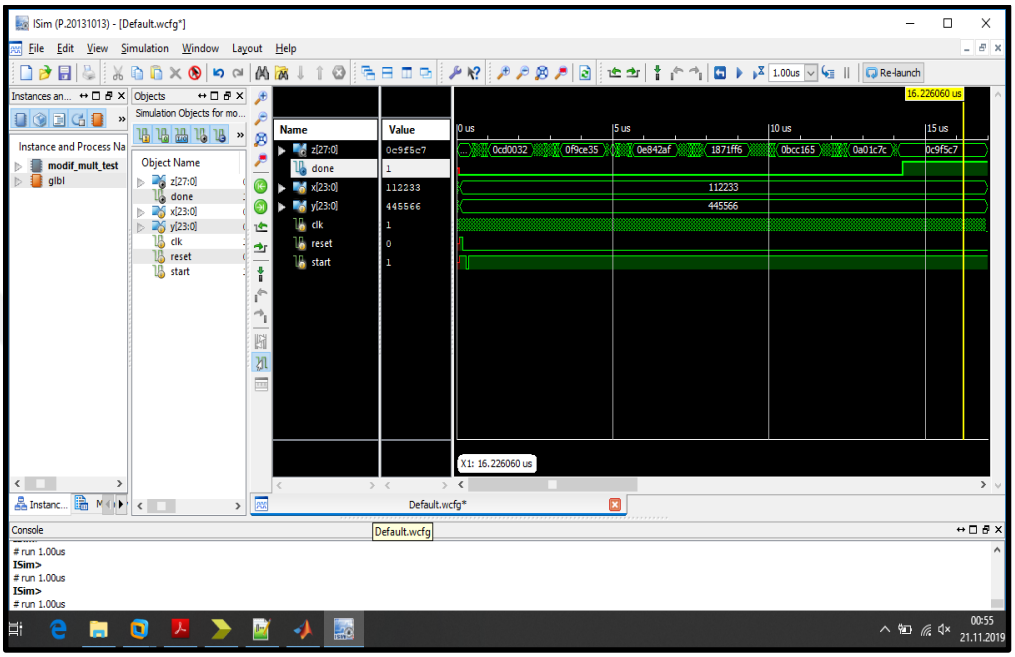
Algoritma 2.6. Montgomery çarpıcısı kullanarak Montgomery uzayından çıkış

GİRDİLER: $R = 2^{n*k}$ $M = M_k \dots M_2 M_1 M_0$, $2^{nk-1} < M < 2^{nk}$ $\overline{X} = \overline{X}_k \dots \overline{X}_2 \overline{X}_1, \overline{X}_0$, $\overline{X}_i = \overline{x}_n \dots . \overline{x}_1 \overline{x}_0$ $\overline{X}$=X*R mod(M) ÇIKTILAR: X, X< M
$X = \text{MontMul}(\overline{X}, 1)$ $= \overline{X} * 1 * R^{-1}$ (mod M) = X * R*1*R^{-1} (mod M) Return X

Döngünün son adımda $\overline{S} = Y^d * R \bmod(R)$ değeri hesaplanmaktadır. Asıl sonuç olan $Y^d \bmod M$' nin elde edilmesi için bu değerin Montgomery uzayından çıkarılması gerekir. Bu işlem ise Algoritma 2.6.'da verilen Montgomery çarpıcısı kullanılarak gerçekleştirilir. Sonuç olarak MontMul($\overline{S} * 1$) =($Y^d * R \bmod(M) * 1 * R^{-1}$) mod M) = $Y^d \bmod M$ değeri elde edilir. Tez çalışmasında $R^{-1} \bmod M$ değerinin algoritmaya dışarıdan verildiği varsayılmaktadır. Normalde bu değer de devre içerisinde hesaplanabilir, ancak işlem kolaylığı açısından bu durum tercih edilmemiştir.

Sakura kartında çalışmak üzere iki ayrı boyda RSA işlemi gerçekleştiren devre geliştirilmiştir. Bunlardan birincisi n=192 bit uzunluğunda anahtar boyu içeren ve Montgomery çarpıcısı 32 bit uzunluğunda olan bir devredir. 192 bitlik anahtar boyu

normalde kriptolojik açısından zayıf bir değer olsa da YKA analizleri açısından bir fark yaratmamaktadır. YKA analizinde gerekli temelleri sağlaması ve yapılacak işlemlere kolaylık sağlaması bakımından bu boyut tercih edilmiştir. Bu boyda RSA gerçeklemesinden yapılan saldırı türleri aynı şekilde daha büyük anahtar boylarına da uygulanabilir.



Şekil 2.6. 192 bit RSA devresi tepe modül sinyalleri

Geliştirilen bir diğer RSA işlemcisi ise n=24 bit uzunluğundadır ve 4 bit Montgomery indirgeyicisi içermektedir. VCD dosyasının pars edilerek benzetim tabanlı güç tüketim eğrisini elde etmek üzere n=24 bit uzunluğunda ve Montgomery indirgeyicisi 4 bit olan bir türevi daha gerçeklenmiştir. Bu devre özellikle benzetim tabanlı güç eğrilerini daha hızlı elde etmek amacıyla geliştirilmiştir. Özellikle ISE kullanılarak yerleştirme-bağlama (place –route) seviyesinde benzetimden elde edilen VCD dosyaları, çok fazla farklı sinyal içermektedir. 192 bit ya da daha yüksek boylarda, bu dosyanın çözümlenmesi ve sinyal değişim sayılarının elde edilmesi işlemi çok uzun sürmektedir. Buna çözüm olarak böyle bir temsili RSA boyu tercih edilmiştir.

RSA işlemini yapmak için geliştirilen Verilog modülleri, ana işlem modülleri ve arayüz modülleri olmak üzere iki temel kısımdan oluşmaktadır. RSA algoritmasını gerçeklemek üzere temel olarak, “Montgomery çarpımı” ve “Üs alma ” işlemlerini yapan Verilog modülleri geliştirilmiştir. Ara yüz modülü olarak da yerel veri yolu

(local bus) üzerinden ara yüz FPGA'i ile haberleşmesini sağlayan mevcut modüllerde düzenleme yapılmıştır.

Şekil 2.6.'da kripto FPGA'i olarak isimlendirilen Kintex7 tipi FPGA için hazırlanmış olan Verilog modüllerinin tamamının en üst modül sinyalleri görülmektedir. Ayrıca bu modülün RTL şeması Ek-A 'da verilmiştir. Her bir modülün işlevi ve ne oranda hazır olarak/ne oranda temelinde geliştirilerek gerçeklendiği aşağıda açıklanmıştır.

Lbus_If: şifreleme modüllerinin arayüz FPGA'i üzerinden bilgisayar ile haberleşmesini sağlayan ara yüz modülüdür. Sakura Board geliştiricisi tarafından sağlanan Verilog kodu [69] üzerinde değişiklik yapılarak geliştirilmiştir.

Mont_expo**:** Bu modül "Soldan sağa hep çarp ikilik üs alma" algoritmasını gerçekleştirmektedir. Gerçeklemede her bir anahtar biti için gereksiz bir kare alma ve çarpma işlemi yapılmakta ve bu işlem sonucu bir kütüğe kaydedilmektedir. Eğer anahtar değeri 0 ise, sonraki işlem adımında bu kütük değeri kullanılmaktadır. Aksi halde bu değer hiçbir işlem görmemektedir. Sonuç olarak bu şekilde anahtardaki 0-1 bit değerleri için mümkün olduğunca özdeş işlemlerin gerçekleştirilmesi ve Basit Güç Analizi (BGA) kaçağının oluşması engellenmektedir. Buna rağmen geliştirmede bulunan diğer açıklıklardan dolayı, çeşitli zaman ve frekans uzayı FGA-İGA türü saldırılarla anahtar bitlerinin elde edilmesi mümkün olabilmektedir.

Montmul_blokcarpici: Bu modül, Montgomery indirgemesi yöntemi ile modüler çarpım işlemlerini gerçekleştirmektedir. Modül 192 bitlik girişleri 32 bitlik parçalar şeklinde uzun modüler çarpma işleminden geçirerek 192 bitlik uzunluktaki değerleri hesaplamaktadır. 24 bitlik RSA gerçeklemesi için ise, aynı mantıkça çalışan 4 bitlik çarpıcı geliştirilmiştir.

RSA_girismodulu: Bu modül, normalde FPGA'e konmak amacıyla değil, simülasyonlarda kullanılmak üzere geliştirilmiştir. Üs alma devresi, dışarıdan 192 bitlik anahtar/açık veri alacak şekilde tasarlanmış olduğundan, doğrudan bu modülün post-place-route benzetimi yapılamamaktadır. Bu kadar pin FPGA kartında bulunmadığından davranışsal seviyenin üstündeki benzetimlerde hata alınmaktadır. Üs alma modülü üzerinde yer alan ve "Local Bus" üzerinde arayüz FPGA'i ile haberleşen modül de karmaşık bir yapıda olduğundan bu modüle test betiği geliştirmek

daha zordur. Bu nedenle benzetimlerde kullanmak üzere, üs alıcı devreyi kapsayacak şekilde dış ara yüzlerinden 16 bitlik bloklar halinde anahtar ve açık veriyi alıp RSA işlemlerini başlatması için üs alma modülüne veren ve gerekli diğer sinyalleri sağlayan bu ek modül geliştirilmiştir.

2.3.2. Hazır ASIC RSA Devresi

Saldırı hedefi olarak kullanılan bir diğer devre ise ASIC olarak gerçeklenmiş bir RSA kripto işlemcisidir. Bu işlemci "Montgomery Çarpıcısı" ile modüler indirgeme yapmakta ve üs alma adımlarında "Montgomery Merdiveni" yöntemini kullanmaktadır. Devrede standart FGA saldırılarına karşı önlem olarak üs alma işlemindeki ara değerleri tahmin etmeyi engelleyici "taban ya da mesaj körleştirme adı verilen mekanizmalar bulunmaktadır. Ancak tez kapsamında gerçekleştirilen saldırılarda, işlem ara değerleri değil de doğrudan anahtar bitlerinin kendisi hedeflendiğinden bu önlemin bir önemi yoktur.

2.3.2.1. Montgomery merdiveni üs alma devresi

Algoritma 2.7. Montgomery merdiveni üs alma yöntemi

GİRDİLER M = $m_{p-1} \dots m_2 m_1 m_0$ $R = 2^p$ $R^{-1} = 2^{-p}\ mod\ M$ $Y = y_{p-1} \dots y_2 y_1 y_0$ $d = d_{p-1} \dots d_2 d_1 d_0$ ÇIKTILAR S = $Y^d\ mod(M)$

A= MMul (Y, R^2), B=MontMul(A,A)
For i= p-1 to 1 do
  If (d_i=0)
  B= MontMul (A,B)
  A= MontMul (A,A)
  If (d_i=1)
  A= MontMul (A,B)
  B= MontMul (B,B)
S= MontMul (A,1)
Return S

Montgomery Merdiveni yöntemi aslında Peter Montgomery tarafından [52] eliptik eğri işlemlerindeki skaler çarpım adımlarını hızlandırmak amacıyla önerilmiş bir yöntem olmakla birlikte Joe ve Yen tarafından [51] yeniden keşfedilip BGA kaçağı içermeyen üs alma algoritması olarak önerilmiştir.

Algoritma 2.8. Hedef Montgomery merdiveni algoritma akışı

GİRDİLER M = $m_{p-1} \dots m_2 m_1 m_0$ $R = 2^p$ $R^{-1} = 2^{-p} \ mod \ M$ $Y = y_{p-1} \dots y_2 y_1 y_0$ $d = d_{p-1} \dots d_2 d_1 d_0$ ÇIKTILAR S = $Y^d \ mod(M)$

A0= MontMul (Y, R^2),
A1= MontMul (A0,A0)
For i= p-1 to 1 do
If (d_i=0 && d_{i+1}=0)
A1= MontMul (A1,B)
A0= MontMul (A0,B)
Else If (d_j=0 && d_{j+1}=1)
A1= MontMul (A1,B)
A0= MontMul (A0,B)
Else If (d_i=1 && d_{i+1}=0)
A0= MontMul (A0,B)
A1= MontMul (A1,B),
Else If (d_j=1 && d_{j+1}=1)
A0= MontMul (A0,B)
A1= MontMul (A1,B)
S= MontMul (A0,1)
Return S

Montgomery çarpıcısı kullanılarak gerçekleştirilen Montgomery merdiveni işlem akışı Algoritma 2.7.'de görülmektedir. Burada M değeri p bit büyüklüğünde modülüs, Y taban değeri görevi gören giriş verisi, d üs değeri olarak işlem gören anahtar değeridir. Saldırı hedefi olarak kullanılan Montgomery Merdiveni devresinin, bazı üst düzey gerçekleme ayrıntılarını içeren işlem akışı Algoritma 2.8.' de verilmiştir.

3. GELİŞTİRİLEN ÖZGÜN GÜÇ ANALİZİ YÖNTEMLERİ

Bu bölümde, tez kapsamında geliştirilen ve çapraz ilinti analizi (ÇİA), tüm bitler çapraz ilinti analizi, yenilikçi ilinti analizi ve şablon tipi ilinti analizi ve frekans uzayı çapraz ilinti analizi isimleri verilen özgün yöntemler tanıtılarak uygulama sonuçları gösterilmiştir. Geliştirilen yöntemlerin tamamı öncelikle ASIC ML tabanlı devreye, bir kısmı da FPGA üzerinde ikilik üs alma yöntemi kullanan RSA devresinin gerçek ve benzetim tabanlı güç eğrilerine uygulanmıştır. Hedef devrelerde değeri elde edilmeye çalışılan gizli anahtar değerine ait bitler iki temel tipe ayrılmaktadır. ML tabanlı üs alma kullanan gerçeklemede, anahtar tiplerinden kendinden sonraki ile değeri aynı olan bite (0-0, 1-1 örüntülerinin ilk bitine) tip0 ismi verilmiştir. Kendisinden sonra gelen bitten değeri farklı olan bitlere (0-1, 1-0 örüntülerinin ilk biti) tip 1 ismi verilmiştir. İkilik üs alma yöntemi kullanan gerçeklemelerde ise, değeri "0" olan bitler tip0, değeri "1" olan bitler ise "tip1" olarak adlandırılmaktadır.

3.1. Tek Referans Bit Çapraz İlinti Analizi

Bu bölümde, geliştirilen çapraz ilinti tabanlı, iki özgün analiz yöntemi tanıtılmıştır. Bu yöntemler hem ASIC ML tabanlı RSA gerçeklemesine hem de FPGA üzerinde ikilik üs alma yöntemi kullanan devrenin gerçek ve benzetim eğrilerine uygulanmıştır. Ayrıca yöntemi tam başarımla sonuçlandırmak için kullanılması gereken eğri sayısının hesabına ilişkin istatistiksel bir model oluşturulmuştur.

3.1.1. Yöntemin tanıtımı

Tez kapsamında geliştirilen ve "tek referans bit çapraz ilinti analizi " olarak nitelendirilen yöntemde, RSA üs alma döngüsündeki her bir anahtar bitinin tipine göre gerçekleştirilen farklı işlem adımlarının, bu işlem adımlarına ait güç eğrisi bölütleri arasındaki "çapraz ilinti değerine göre" sınıflandırılması hedeflenmektedir. Yöntemin temel dayanağı, belli bir bit referans olarak alındığında, bu bite ait güç eğrisi bölütünün, tüm diğer bitlere ait bölütleri ile çapraz ilintisinin, bitlerin referans ile aynı ya da farklı oluşuna göre değişiklik göstereceğidir. Bu sınıflandırma işlemi doğru

olarak gerçekleştirildiğinde anahtar değeri bit bit elde edilmiş olmaktadır. Yöntem, her bir eğri bölütünde bulunan elektronik gürültü ve anahtarlama gürültüsü nedeni ile tek bir güç eğrisi kullanılarak sonuca ulaşamamakta, birden fazla güç eğrisinden elde edilen ilinti değerlerinin birleştirilmesi gerekmektedir. Çalışmada, bu ilinti değerlerini birleştirmek üzere iki farklı yaklaşım sunulmuştur. Bu yaklaşımlardan ilkinde her bir eğri için elde edilen çapraz ilinti değerleri bir eşikle karşılaştırılmakta ve bu karşılaştırma sonucuna göre iki sınıftan birine ait sayaç değeri artırılmaktadır. Tüm eğriler kullanıldığında her bir bite ait sayaç değerleri hangi sınıf için en yüksek değeri almışsa bitin o sınıftan olduğuna karar verilmektedir. İkinci yöntemde ise her bir eğri için elde edilen ilinti değerleri toplanmakta ve tüm eğriler toplandıktan sonra hesaplanan eşik değer ile karşılaştırılarak her bitin hangi sınıftan olduğuna karar verilmektedir.

Saldırının başarılı olması için aynı anahtar değeri ile farklı açık veri değerlerinin işlem gördüğü pek çok güç tüketim eğrisinin kullanılması gerekmektedir, aksi halde yöntem işe yaramamaktadır ya da başarımı çok düşmektedir. Bunun nedenlerinden biri odaklanılan her bir işlemde bulunan anahtarlama gürültüsüdür. Yukarıda ayrıntılarının verildiği gibi, çapraz ilinti tabanlı saldırının kullandığı gerçekleme zayıflıkları, anahtar bitlerinin tipine göre farklı bellek alanlarından veri okunması, ya da anahtar bitinin değerine göre önceki ile aynı ya da farklı veriyi okumaktan kaynaklanan değişimlerdir. Odaklanılan bu tür işlemler sırasında, belli bir bellekten taşınan verinin HW ya da HD değişimleri de "veri bağımlı" bir güç tüketimine neden olmaktadır. Bu güç tüketimi, saldırı açısından kullanılabilecek bir parçaya sahip olmayıp, hedef işlemle aynı anda gerçekleştiğinden, güç ölçümleri üzerine "anahtarlama gürültüsü" eklemektedir. Rastgele HW ya da HD değerlerinin taşınması sırasında ölçülen pek çok eğri kullanılarak, bu değişimlerden gelen etkinin sıfırlanması hedeflenmiştir. Pek çok eğri kullanılmasının bir diğer nedeni ise her bir güç ölçümünde toplamsal bir elektronik gürültünün de bulunmasıdır. Birden çok eğri kullanılarak bu gürültüden gelen bileşenlerin de tıpkı anahtarlama gürültüsünden kaynaklananlar gibi azaltılması hedeflenmiştir. Böylece, hesaplanan ilinti değerlerinden ortak bir ilinti değeri üretilirken, sadece anahtar değerine bağlı etkilerin birbirini güçlendirerek, olması gereken ilintinin elde edilmesi sağlanmaktadır.

Çapraz ilinti analizinde, çapraz ilintisi hesaplanacak güç eğrisi alanlarının mümkün olduğunca ilgilenilen işleme ait bölgeleri içermesi, fazlalık alanlardan arındırılmış olması önemlidir. Eğri bölütlerinde yer alan ilgisiz alanlar yine gürültü etkisi yapacağından başarımı düşürecektir. Bu nedenle her bir anahtar grubu için, ilgilenilen işlem alanlarının tüm eğri üzerinden elde edilmesi yani eğrinin birim işlem adımlarına bölütlenmesi gerekmektedir. Her bir Montgomery çarpımında bulunan zaman farklarından dolayı, her bir alt adım eşit sürelerde tamamlanmadığından, bölütleme işlemi basitçe işlem yapılan tüm eğri boyunun birim işlem sayısına göre eşit alanlara bölünmesi ile gerçekleştirilememektedir. Uygulamalarda eğri parçalarını bölütlemek için iki temel yöntem kullanılmıştır. Birincisinde eğriler "kayan ortalamalar" tipinde bir sayısal filtreden geçirilerek her bir filtre çıktısının en yüksek değeri aldığı alanlar merkez kabul edilecek şekilde eşit aralıklara bölünmüştür. İkinci yaklaşımda ise referans bir eğri parçası yaklaşık pencere değerleri üzerinde gezdirilerek örtüşmenin en yüksek olduğu indis, örnek merkezi olacak şekilde eşit alanlara ayrıştırılmıştır.

Algoritma 3.1. Çapraz ilinti değerlerinin hesaplanması

GİRDİLER P1, P2 ,..., PM , Pi ={Pi_1, Pi_2, ….Pi_{w-1} } ÇIKTILAR $C_r = C1_r$, $C2_r$,..., CM_r, $Ci_r = \{Ci_{r1}, Ci_{r2} \ldots., Ci_{rw-1}\}$
For i= 1 to M For j=w-1 to 1 do Ci_{rj}=Corr(Pi_r, Pi_j) $Ci_r = \{Ci_{r1}, Ci_{r2} \ldots., Ci_{rw-1}\}$ $C_r = \{C1_r, C2_r \ldots., CM_r\}$ Return C_r

Pi, i inci RSA koşturuşundan elde edilen güç tüketim eğrisi olsun. Bu eğri üzerinde ilgilenilen işlem adımlarının yer aldığı alanlar bölütlenerek Pi_1, Pi_2, …Pi_{w-1} adı verilen ve her bir anahtar bitine ait ilgilenilen işlemin gerçekleştiği zamana ait güç eğrisi bölütleri elde edilir. Elde edilen her bir eğri bölütünün, referans bit ile çapraz ilinti değerinin hesaplanması gerekmektedir. Bu işlem için öncelikle sabit bir referans bit değeri "r" seçilir. Pi numaralı güç tüketim eğrisinde, r numaralı referans bite ait güç eğrisi bölütü Pi_r olsun. Öncelikle Algoritma 3.1.' de gösterildiği gibi, Pi_r bölütünün, tüm diğer j indisli bitlere ait olan Pi_1, Pi_2, ….Pi_{w-1} bölütleriyle, "Pearson" yöntemi kullanılarak hesaplanan Ci_{rj} çapraz ilinti değerleri hesaplanır. Sonuç olarak her bir güç eğrisi Pi için 0 - 1, 1 - w arasındaki tüm anahtar bitlerine ait ve her bir i eğrisi

kullanılarak r referans biti için hesaplanan $Ci_r = \{Ci_{r1}, Ci_{r2}, \ldots, Ci_{rw-1}\}$ M tane çapraz ilinti değerini içeren $C_r = \{C1_r, C2_r, \ldots, CM_r\}$ matrisi elde edilmiş olur.

Birinci yönteme göre anahtar bitinin ait olduğu grubun tespit edilmesi için, daha önce hesaplanmış olan çapraz ilinti değerlerinin nasıl kullanıldığı Algoritma 3.2.'de verilmiştir. Bu yöntemde öncelikle tüm güç ölçümleri için her bir anahtar bitine ait çapraz ilinti değeri, hesaplanan eşik değeri ile karşılaştırılır. Eşik değeri olarak belli sayıda örneğin komşu anahtar bitine ait ortalama ilinti değeri ya da bu değerlerin kayan ortalaması kullanılabilir. Çalışmada, ilk bitten itibaren yan yana 50 şer tane komşu bitin ortalaması kullanılmıştır. Karşılaştırma sonucunda ilintisi eşik değerden küçük olan anahtar bitleri için tip0 sayaç değeri, aksi durumlar için ise tip1 sayaç değeri artırılır. Tüm güç eğrileri bu şekilde değerlendirildikten sonra, ilgili bit için hangi tip sayacı en yüksek değere sahipse bit tipinin o olduğuna karar verilmektedir.

Tüm güç ölçümleri için her bir anahtar bitine ait çapraz ilinti değeri, hesaplanan eşik değeri ile karşılaştırılır. İlinti değeri eşikten küçük olan bitler, referans bit ile aynı sınıfa, düşük olanlar ise farklı sınıfa atanır. Bu algoritma sonucunda anahtar bitlerine ait "w" bitlik ve her bir bitin tip0 ya da tip1 türlerinden hangisine ait olduğunu gösteren tip kestirim vektörü "dt" elde edilmektedir.

Algoritma 3.2. Sayaç yöntemine göre anahtar bit tiplerinin elde edilmesi

GİRDİLER
$C1_r$, $C2_r, \ldots, CM_r$, $Ci_r=\{Ci_{r1}, Ci_{r2}, \ldots, Ci_{rw-1}\}$
ÇIKTILAR
dt={ $dt_{w-1}, \ldots,$ $dt_r, \ldots dt_1$ }

sayaç = $[s_{w-1}, s_{w-2}, \ldots, s_1]$=0;
eşik = 0 ;
For i= 1 to M
  For j=w-1 to 1 do
    eşik = EşikGüncelle1 (Ci_{rj})
    If Ci_{rj} > eşik
      s_j ++;
    If s_j > i/2
      dt_j = tip0
    Else
      dt_j = tip1
dt={dt_{w-1} , $dt_{w-2}, \ldots,$ $dt_r, \ldots dt_1$ }
Return dt

Algoritma 3.3.'de ikinci yönteme göre anahtar bitlerinin nasıl sınıflandırıldığı görülmektedir. Bu yöntemin ilkinden temel farkı, bit tipine karar verirken her bir bit için hesaplanan ilinti değerlerinin ortalamasının kullanılmasıdır. Belli bir bit için, her bir güç eğrisinden hesaplanan ilinti değerlerinin ortalaması hesaplanır.

Algoritma 3.3. İlintiler toplamı yöntemine göre anahtar bit tiplerinin elde edilmesi

GİRDİLER
$C1_r$, $C2_r, \ldots\ldots, CM_r$, $Ci_r=\{Ci_{r1}, Ci_{r2}, \ldots\ldots, Ci_{rw-1}\}$
ÇIKTILAR
$dt=\{ dt_{w-1}, \ldots\ldots, dt_r, \ldots dt_1 \}$

eşik = 0
For j=w-1 to 1 do
  $C_{rj\ top} = 0$
  For i= 1 to M
    $C_{rj\ \mathbf{top}} = C_{rj\ \mathbf{top}} + Ci_{rj}$
    eşik = EşikGüncelle2 ($Ci_{rj\ \mathbf{top}}$)
    If $C_{rj\ top} >$ eşik
      dt_j = tip0
    Else
      dt_j = tip1
$dt=\{dt_{w-1}, dt_{w-2}, \ldots\ldots, dt_r, \ldots dt_1 \}$
Return dt

Eşik değeri hesaplanırken de bu ortalama ilinti değerlerinin yan yana 50'şer tanesinin tekrar yatayda ortalaması alınmaktadır.

Algoritma 3.4. Tek referans bit eşik güncelleme fonksiyonları

GİRDİLER
$C1_r$, $C2_r, \ldots\ldots, CM_r$, $Ci_r=\{Ci_{r1}, Ci_{r2}, \ldots\ldots, Ci_{rw-1}\}$
ÇIKTILAR
eşik

eşik = 0
For j=w-1 to 1 do
  $C_{rj\ top} = 0$
  k1= j/50
  k2= mod(j,50)
  For i= 1 to M
    If(EşikGüncelle1)
      eşik(j)= mean ($Ci_{r\ k1.50+1}, Ci_{r\ k1.50+2} \ldots\ldots, Ci_{r\ k1.50+50}$)
    Else If(EşikGuncelle2)
      $C_{rj\ \mathbf{top}} = C_{rj\ \mathbf{top}} + Ci_{rj}$
      eşik(j)= mean ($C_{r\ k1.50+1\ top}, C_{r\ k1.50+2\ top} \ldots\ldots, Ci_{r\ k1.50+50\ top}$)
Return eşik

Bitin tipine karar verilirken, tüm eğrilerden elde edilen ortalama ilinti değeri, eşik değeri ile karşılaştırılır ve ortalama ilintisi eşik değerinden yüksek olan bitlere, referans bit ile aynı tip yani tip0 tipi atanırken, düşük olanlar da tip1 olarak sınıflandırılmaktadır. Her iki yöntem için eşik değerinin hesaplanmasına ilişkin işlem akışı Algoritma 3.4.'de verilmiştir.

3.1.1.1. Gerekli eğri sayısını hesaplama yöntemi

Tüm YKA saldırıları gibi çapraz ilinti analizinde de, parça parça elde edilen anahtar değerinin belli bir yüzde ile doğru şekilde kestirebilmek için kaç eğri kullanılması gerektiğinin hesabı önemlidir. Önceki bölümlerde de açıklandığı gibi, her bir güç eğrisi bölütünde, hedef işlemle aynı anda gerçekleşen ve saldırı açısından kullanılabilir olmayan diğer işlemlerden kaynaklanan "anahtarlama gürültüsü" ve devre içi ve dış etkenlerden kaynaklanan elektronik gürültü bulunmaktadır. Tüm bu nedenlerden dolayı her bir RSA koşturumunda, sabit referans bitin belli bir hedef bitle olan çapraz ilintisi, rastgele bir süreçten beklendiği üzere farklı değerler almaktadır. Tüm bu gürültü etkilerinden dolayı tek bir ölçümde yeteri kadar Sinyal/Gürültü oranı (SNR) elde edilemediğinden, referans bit ile hedef bit arasında var olan gerçek ilinti değerine, aynı işleme ait birden fazla ölçümden faydalanılarak daha iyi bir yaklaşıklık yapılması gerekmektedir.

Bölüm 1.1'de, güç analizinde kullanılan güç eğrilerine ait ortalama ve standart sapma değerlerinin kendisinin de belli bir dağılıma sahip rastgele değişkenler olarak kabul edildiğinden bahsedilmişti. Kendisi de normal dağılıma sahip bir X rastgele değişkeninin örnek ortalaması kullanılarak hesaplanan $\overline{X}$ değeri, daha fazla eğri kullanıldığında daha az hata ile gerçek ortalama olan μ değerine yaklaşmaktadır. Geliştirdiğimiz çapraz ilinti analizinde, birden çok eğri kullanılarak kestirimi yapılmaya çalışılan parametre, referans ve hedef eğri bölütleri arasındaki ilinti değeridir. Aslında her biri normal dağılıma sahip süreçlerden geldiği bilinen ve rastgele olarak seçilen örnekler arasındaki çapraz ilinti değerleri, tam olarak normal dağılıma sahip değildir. Ayrıca temelde iki vektör arasındaki ilinti değeri, bu vektörler arasındaki açının kosinüs değerini vermekte olup toplamsal özelliğe de sahip değildir. Ancak bu bölümde öncelikle, aynı bit için pek çok farklı giriş değerine sahip eğrilerden elde edilen çapraz ilinti değerlerinin ortalamasını alarak, gerçek ilinti değerini tahmin

etmeye dayalı yaklaşımın geçerli bir yaklaşım olduğu literatür örnekleri ile kanıtlanacaktır. Bunun yanı sıra yöntemimizdeki koşullarda kullanıldığında bu ortalama ilinti değerinin yaklaşık olarak normal bir dağılıma sahipmiş gibi muamele görebileceği yine literatür örnekleri ile gösterilecektir. Daha sonra bu varsayım altında eğri sayısının hesabı için, ilinti katsayılarının ortalamasına ait örnekleme dağılımı kavramının nasıl kullanıldığı açıklanacaktır. Yöntemin çeşitli gerçeklemelere uygulanmasının anlatıldığı ileriki bölümlerde de teori ve pratiğin ne derece uyumlu olduğu gösterilecektir.

Literatürde birbirinden bağımsız ve k>2 olmak üzere her biri n_i tane çift içeren (n_i boyutlu) k tane rastgele örnek değişken (X_{ij}, Y_{ij}) i=1,2,… , k ve j=1,2,… n_i için hesaplanmış r_i ilinti katsayılarının kullanılarak gerçek ilinti değerine daha iyi yaklaşan ortak bir ilinti değerinin elde edilmesi konusunda çeşitli yöntemler bulunmaktadır. Bu yöntemlerden en yaygın olanı, öncelikle k tane deneyden elde edilen her bir r_i ilinti katsayısının aşağıdaki gibi Z_i değerlerine dönüştürülerek kullanılmasıdır [24].

$$Z_i = \tanh^{-1}.r_i = \frac{1}{2}\log_e\{(1+r_i)/(1-r_i)\} \quad (3.1)$$

Buradaki Z_i değerleri, ortalaması $Z_\rho = \tanh^{-1}.\rho$ ve varyansı $(n_i - 3)^{-1}$ olan ve yaklaşık olarak normal dağılıma sahip Fisher Z-dönüşüm değerleridir [24]. Bu işlemden sonra, Z dönüşümü ile elde edilen her bir Z_i değerinin ağırlıklı ortalaması olan $\overline{Z_w}$ değeri aşağıdaki gibi elde edilir.

$$\overline{Z_w} = \frac{\sum_{i=1}^{k}(n_i-3).Z_i}{\sum_{i=1}^{k}(n_i-3)} \quad (3.2)$$

Gerçek ilinti değeri ρ'nun kestirimi olan r_F'yi elde etmek için aşağıdaki eşitlik kullanılır.

$$r_F = \tanh Z_w = \{\exp(2Z_w)-1\} \quad (3.3)$$

Ortak ilinti değerini elde etmeye yönelik alternatif yaklaşımlar sunan [24 , 72 - 75] çalışmalarında da gösterildiği gibi, gerçek ilinti değeri olan ρ , normalize edilmiş X'_{ij} ve Y'_{ij} değerlerinin birinci moment çarpımı ile de kestirilebilir. Bunun için önce

X_{ij} ve Y_{ij} çiftlerinden elde edilen ortalama ve varyans değerlerinin kestirimi olan ($\overline{X_ı}\,\overline{Y_ı}$) ve ($\overline{S_{xı}}\,\overline{S_{yı}}$) değerleri kullanılarak aşağıdaki gibi X'_{ij} ve X'_{ij} 'ler elde edilir.

$$X'_{ij} = \frac{X_{ij}-\overline{X_ı}}{\overline{S_{xı}}},\ Y'_{ij} = \frac{Y_{ij}-\overline{Y_ı}}{\overline{S_{yı}}} \tag{3.4}$$

Son olarak tıpkı her bir gruba ait ilinti değerlerinin hesaplanmasında yapıldığı gibi X'_{ij} ve Y'_{ij} değerlerinin birinci moment çarpımı alınarak ortak ilinti değeri hesaplanır.

$$r_s=\frac{\sum_{i=1}^{k}\sum_{j=1}^{n_i} X'_{ij}.Y'_{ij}}{\sum_{i=1}^{k}\sum_{j=1}^{n_i} X'^{2}_{ij}\cdot.\sum_{i=1}^{k}\sum_{j=1}^{n_i} Y'^{2}_{ij}}=\frac{\sum_{i=1}^{k}\sum_{j=1}^{n_i} X'_{ij}.Y'_{ij}}{\sum_{i=1}^{k}(n_i-1)\sum_{i=1}^{k}(n_i-1)}=\frac{\sum_{i=1}^{k}\sum_{j=1}^{n_i} X'_{ij}.Y'_{ij}}{\sum_{i=1}^{k}(n_i-1)} \tag{3.5}$$

Yukarıdaki denkleme $r_i = \frac{\sum_{i=1}^{k}\sum_{j=1}^{n_i} X'_{ij}.Y'_{ij}}{\sum_{i=1}^{k}(n_i-1)}$ değerleri konduğu zaman ortak ilinti değeri r_s, aslında basitçe tüm r_i değerlerinin ağırlıklı ortalaması alınarak aşağıdaki gibi hesaplanabildiği görülecektir [24 , 72].

$$r_s=\frac{\sum_{i=1}^{k}(n_i-1).r_i}{\sum_{i=1}^{k}(n_i-1)} \tag{3.6}$$

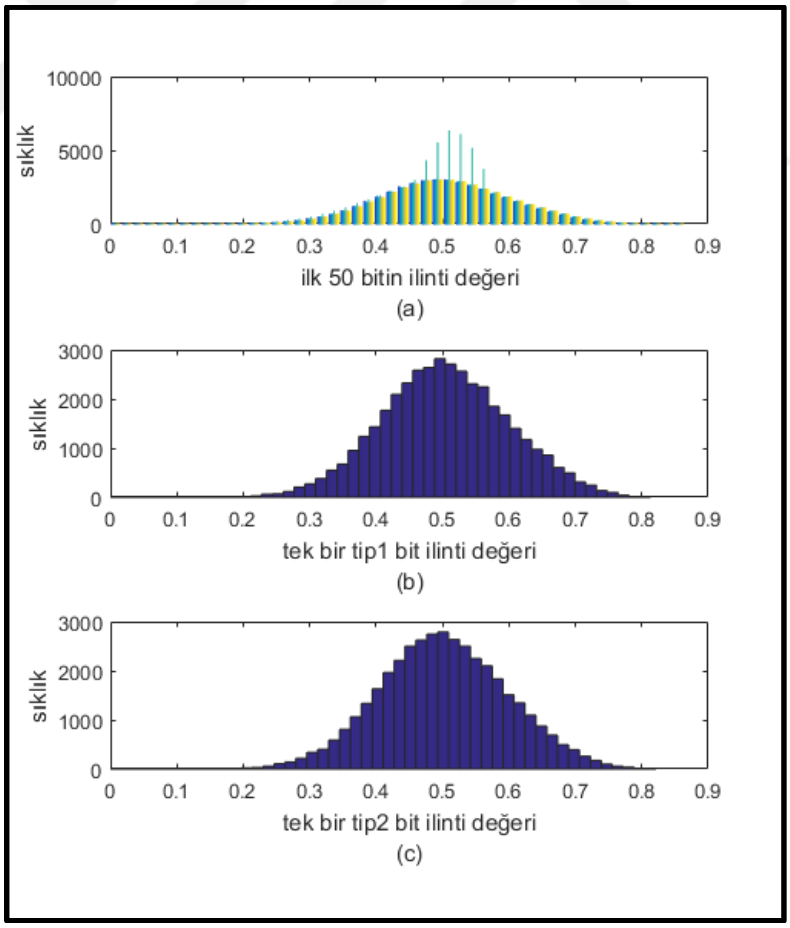
Burada n_i değerleri, her r_i değerinin hesaplanmasında kullanılan rastgele değişken çiftlerinin sayısı yani serbestlik derecesi, k ise kullanılan toplam ilinti sayısıdır. Bizim çalışmamızda ise, her bir eğri için kullanılan nokta sayısı yani serbestlik derecesi aynı olduğundan (n_i=2000), pek çok örnek eğri çiftinden elde edilen r_i çapraz ilinti katsayılarının basitçe toplanıp, bu değerin kullanılan eğri sayısı "k'ya" bölünmesi ile ortalama değeri aşağıdaki gibi elde edilmektedir.

$$r_s=\frac{(n_i-1)\sum_{i=1}^{k}.r_i}{(n_i-1)\sum_{i=1}^{k}1}=\frac{\sum_{i=1}^{k}.r_i}{k} \tag{3.7}$$

Sonuç olarak doğrudan ilinti değerlerinin ortalaması alınarak hesaplanan ortak ilinti katsayısı [24 , 72] çalışmasında anlatılan ikinci yaklaşıma özdeş olmaktadır. Çeşitli çalışmalarda [72 - 75] bizim çalışmamızda olduğu gibi doğrudan ilinti değerlerinin ortalamasının alınarak kullanılmasının, aslında büyük serbestlik derecelerinde yani $n_i > 20$ için her bir ilinti değerinin Fisher-Z dönüşümünü kullanan yöntemle karşılaştırıldığında çok da fazla bir hata içermediği belirtilmektedir. Fisher-Z değişkenlerine dayanan yöntemin özellikle küçük n_i değerleri için önemli olduğu [72 -

75] . Sonuç olarak, basitçe ilinti katsayılarının ortalaması alınarak elde edilen ortak ilinti değerinin standart hatasının hesaplanmasında küçük n_i değerleri için Student-T dağılım tablolarının kullanılabileceği savunulmaktadır. Student-T dağılımı $n_i > 100$ serbestlik dereceleri için normal dağılıma yaklaştığından çalışmamızda Student-T tablosu yerine Z -Tablosu değerleri kullanılmıştır.

İstatistikteki güven aralığı ya da hata oranı olarak isimlendirilen kavramın, belli bir rastgele değişkenin kestirim değerinin, değişkenin gerçek değerine ne kadar yakınsadığını ölçmede kullanıldığından bahsedilmişti. Bu kavramla yakından ilgili olan hipotez testi ise belli bir hipotezin ne oranda doğru olduğunun test edilmesi anlamına gelmektedir. Bir parametrenin, kestirimine dayanan bir hipoteze ait güven aralığı, o hipotezin geçerli olacağı değerleri içeren aralık olarak tanımlanır. YKA'da kaç eğri kullanıldığı zaman, kestirimi yapılan parametrenin güven aralığı içerisinde yer alacağı tahmin edilerek [21] saldırılar için gereken eğri sayısı hesaplanabilir.



Şekil 3.1. İlk 50 bite ait çapraz ilinti değerlerinin dağılımı

Şekil 3.1,'de 40000 eğri ve sabit bir referans bit kullanılarak elde edilen ilinti değerlerinden ilk 50 bit için ve tek bir bit için hesaplanan histogramları görülmektedir. Bu histogramlardan görüldüğü gibi ilinti değerlerinin örnek dağılımı normal dağılıma çok benzemektedir ve yaklaşık olarak r=0.5 değeri etrafında oldukça simetrik bir yapı sergilemektedir. İlk 50 bite ait bu değerlerin ortalama değeri ve varyansı hesaplandığında, Tip1'ler, ortalaması en yüksek varyansı en düşük, Tip2'ler, ortalaması en düşük varyansı en yüksek grup olurken, karma durum için hesaplanan ortalama ve varyans, beklendiği gibi bu değerlerin ortasında yer almaktadır.

Hem ilinti değerlerinin dağılımına ilişkin literatür hem de pratik histogram değerleri göz önüne alındığında, çapraz ilinti değerlerinin güven aralığı hesaplamalarında, bu değerlerin yaklaşık normal dağılıma sahip olduğu varsayılabilir.

Ortalama çapraz ilinti değerini kullanmaya dayanan ÇİA yönteminde, güven aralığı ve standart hata kavramları kullanılarak gerekli eğri sayısının hesaplanabilmesi için, öncelikle 50 bitlik komşu bitlerin, ortak ilinti değerinin nasıl kestirilebileceğine dair bir model oluşturulması gerekmektedir. Bunun nedeni komşu elli bitin kestiriminde, bu bitlere ait ilintiler ortalamasının eşik olarak kullanılmasıdır Çapraz ilinti değerlerinin kullanılarak anahtar tipine nasıl karar verildiği "Algoritma 3.3.'de" görülebilir. Aslında burada hem tip0 ve tip1 bitlerine ait ilinti değerleri, hem de eşik değeri, belli bir ortalama ve standart sapma ile değişen normal dağılıma sahip rastgele değişkenler olarak düşünülmüştür. 50'bitlik bir diliminde yer alan tip0 ya da tip1 türündeki her bir bitin değerinin doğru olarak hesaplanabilmesi, bu bite ait ortalama ilintinin, ortalama eşik değerinden uzaklığının 0'dan büyük olması anlamına gelmektedir. Belli bir 50'lik dilimdeki tip0 bitlerin $N(M_{tip1}, \sigma_{tip1})$, tip1 bitlerin $N(M_{tip1}, \sigma_{tip1})$, ve bu aralıktaki tüm bitlerin ortalamasından oluşan eşik değerinin ise $N(M_{esik}, \sigma_{esik})$ dağılımına sahip rastgele değişkenler olduğunu düşünelim. Bu durumda, bu aralıktaki tip0 ya da tip1 bitlerinin eşikten farkına ait dağılım parametreleri aşağıdaki gibi ortalamaların farkı ve varyansların toplamı şeklinde hesaplanabilir [21].

$$M_{tip1-esik} = M_{tip1} - M_{esik} \quad , \quad M_{tip2-esik} = M_{tip2} - M_{esik} \tag{3.8}$$

$$\sigma^2{}_{\text{tip1}-\text{esik}} = \sigma^2{}_{\text{tip1}} + \sigma^2{}_{\text{esik}}, \sigma^2{}_{\text{tip2}-\text{esik}} = \sigma^2{}_{\text{tip2}} + \sigma^2{}_{\text{esik}} \quad (3.9)$$

Yukarıda parametreleri çıkarılan $N(M_{tip1}, \sigma_{tip1})$, $N(M_{tip2}, \sigma_{tip2})$ dağılım parametrelerini, gerçek değerlerini bilmediğimiz durum için n/2 tane eğri kullanarak kestirdiğimizi varsayalım. Bu durumda $\sigma^2{}_{\text{tip1}-\text{esik}}, \sigma_{\text{tip2}-\text{esik}}$ varyans değerleri (pooled variance) kullanılarak, ortalamaya ait varyansın örnek kestirimini temsil eden "$\overline{\sigma^2}_{\text{tip1}-\text{esik}}$ ve $\overline{\sigma^2}_{\text{tip2}-\text{esik}}$ " değerleri, aşağıdaki gibi kestirilebilir.

$$\overline{\sigma^2}_{\text{tip1}-\text{esik}} = \frac{\sigma^2{}_{\text{tip1}-\text{esik}}}{\frac{n}{2}} = \sigma^2{}_{\text{tip1}-\text{esik}} \cdot \frac{2}{n} \quad (3.10)$$

$$\overline{\sigma^2}_{\text{tip2}-\text{esik}} = \frac{\sigma^2{}_{\text{tip2}-\text{esik}}}{\frac{n}{2}} = \sigma^2{}_{\text{tip2}-\text{esik}} \cdot \frac{2}{n} \quad (3.11)$$

Burada $c_{tip1} = M_{tip1-esik}$ ve $c_{tip2} = M_{tip2-esik}$ değerlerini $N(M_{tip1-esik}, \sigma_{tip1-esik})$, $N(M_{tip2-esik}, \sigma_{tip2-esik})$ dağılımlarının ortalama değerlerinin 1-α olasılıkla olması gereken aralık sınırını, yani çift taraflı güven sınırı olarak alırsak, gerekli eğri sayısı aşağıdaki gibi hesaplanabilir:

$$n_{\text{tip1}-\text{esik}} = 2.z^2_{1-\alpha} \frac{\sigma^2_{\text{tip1}-\text{esik}}}{M^2_{\text{tip1}-\text{esik}}} \approx 4.z^2_{1-\alpha} \frac{\sigma^2_{\text{tip1}}}{M^2_{\text{tip1}-\text{esik}}}, \quad (3.12)$$

$$n_{\text{tip2}-\text{esik}} = 2.z^2_{1-\alpha} \frac{\sigma^2_{\text{tip2}-\text{esik}}}{M^2_{\text{tip2}-\text{esik}}} \approx 4.z^2_{1-\alpha} \frac{\sigma^2_{\text{tip2}}}{M^2_{\text{tip1}-\text{esik}}} \quad (3.13)$$

Bu alt bölümde anlatılan çapraz ilinti kullanan yöntem, hedef devrelerden hem ML üs alma algoritması kullanan ASIC RSA gerçeklemesine, hem de ikilik üs alma algoritması kullanan FPGA RSA gerçeklemesine uygulanmıştır. Uygulamalarda FPGA devresinin hem gerçek güç ölçümleri hem de benzetim tabanlı güç eğrileri kullanılmıştır. Bu hedeflere ait uygulama sonuçları aşağıdaki bölümlerde yer almaktadır. Her bir uygulamada, (3.12) ve (3.13) eşitlikleri kullanılarak, başarılı saldırı için gerekli eğri sayısının teorik ve pratik hesapları yapılmış, elde edilen sonuçlar karşılaştırılmıştır. Genel olarak, teorik olarak hesaplanan eğri gereksinimleri ile pratik değerlerin bir biri ile tutarlı olduğu tespit edilmiştir.

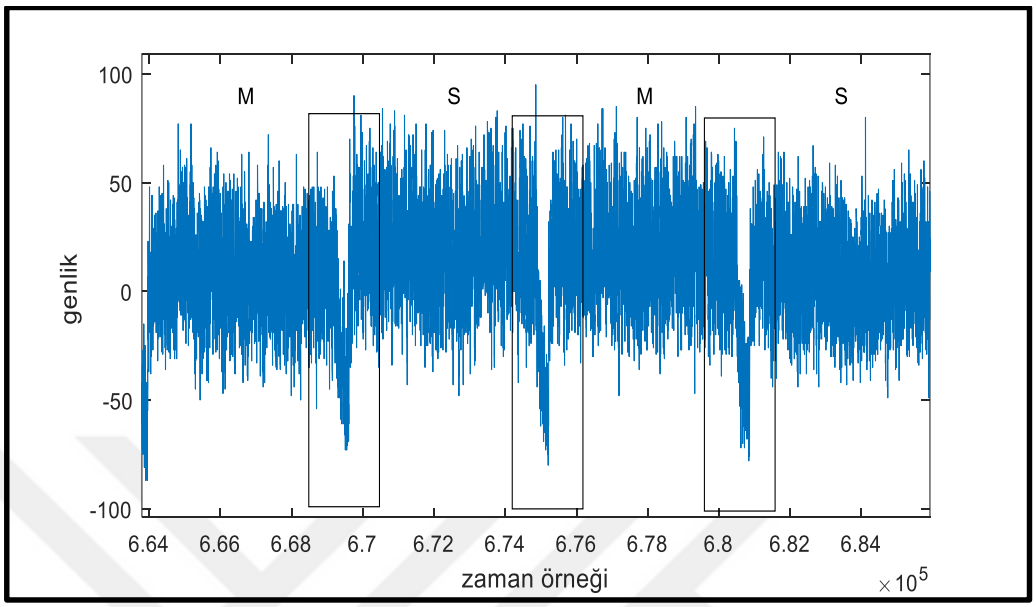
3.1.2. Yöntemin ASIC ML devresine uygulanması

Çapraz ilinti analizi yönteminin uygulandığı hedef devrelerden birisi, üs alma algoritması olarak Montgomery Merdiveni kullanan ASIC bir RSA gerçeklemesidir. Hedef devreye ait gerçekleme ayrıntıları Algoritma 2.8.'de verilmiştir. Gerçekleme ayrıntılarına bakıldığında her bir bit için gerçekleştirilen çarpma-kare alma işlemlerinde, A0 ve A1 olarak isimlendirilen iki kütükten birinin ilk işlem girdisi için, B olarak isimlendirilen bir diğer kütüğün ise ikinci işlem girdisi için kullanıldığı görülmektedir. Her bir çarpma işleminde, eğer sonraki bit işlem görmekte olan bit ile aynı değere sahip ise, B kütüğünün içeriği doğrudan mevcut işlem sonucu ile oluşturulmaktadır. Ancak eğer sonraki bit, şimdiki bitten farklı bir değere sahip ise B kütüğü içeriği başka bir kütükten okunmaktadır. Sonuç olarak anahtar bitinin değerine bağlı olarak farklı konumlardan işlem girdisi okumaya dayanan bir farklılık oluşmaktadır. Bu işlem farkı güç eğrileri kullanılarak ayırt edilebildiğinde anahtar bit değerleri de elde edilebilecektir.

Makalede önerilen yöntem bu işlem farkının ilgili güç tüketim eğrilerinin çapraz ilintisinin kullanılarak tespit edilmesine dayanmaktadır. Bu amaçla öncelikle anahtar bitleri, bit değerinin sonraki bitle aynı ya da farklı olmasına göre iki gruba ayrılmıştır. Değeri bir sonraki bit ile aynı olan yani 0-0, ya da 1-1 örüntüsüne sahip bitlerden ilkine tip0, bit değeri sonraki bitten farklı olan yani 0-1 ya da 1-0 örüntüsüne sahip bitlerden ikine de tip1 bit ismi verilmiştir. Saldırıdaki temel varsayım isetip0 türünde olan bir bitin, yukarıda bahsedilen işlem farkının gerçekleştiği zamana ait güç tüketim eğrisinin, diğer bitlere ait aynı alanlarla çapraz ilintisi hesaplandığında, bu değerin tip0 bitler için ortalama ilinti değerinden daha yüksek bir değer, tip1 bitler için ise daha düşük bir değere sahip olacağıdır. Bu şekilde bu çapraz ilinti değerinin bir eşikle karşılaştırılması ile her bir bitin tipine karar verilebilecektir.

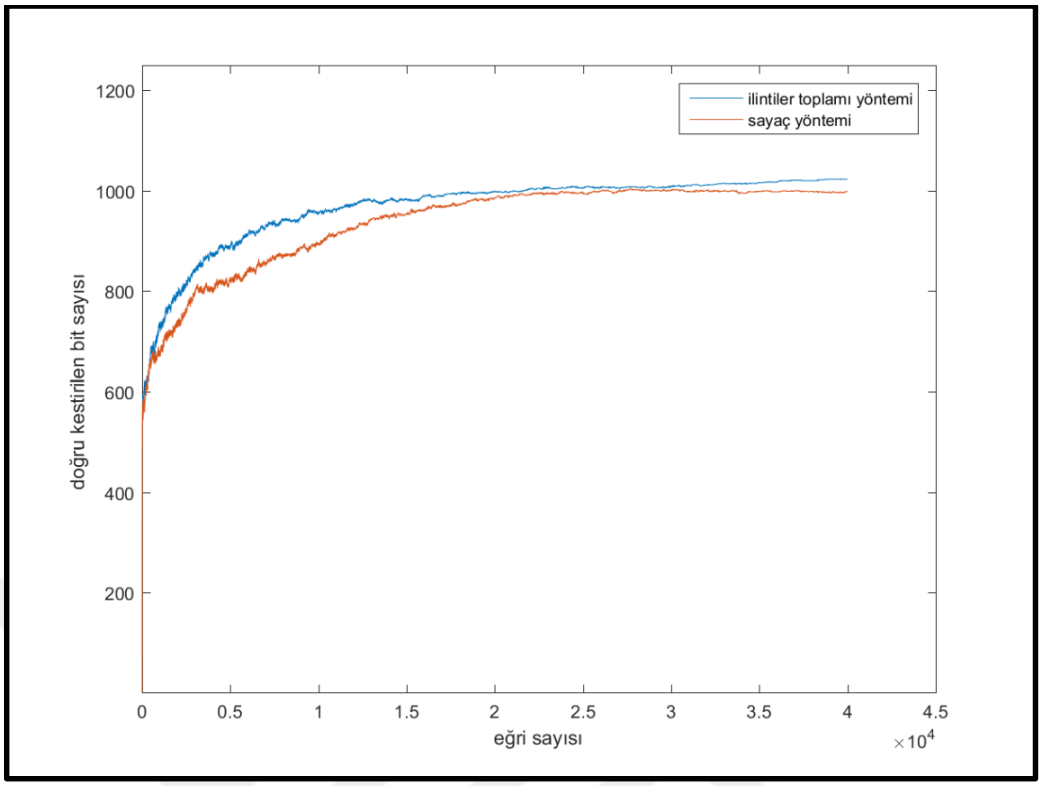
Şekil 3.2.'de çarpma ve kare alma döngü adımlarını içeren örnek bir güç eğri parçası görülmektedir. Her bir ölçümde ilgilenilen eğri parçası bölütleri, şekilde kare içine alınmış olan ve şimdiki bite ait kare alma işlemi ile sonraki bite ait çarpma işlem adımı arasında yer alan alanlardır. Bu alanlar çıkarılarak daha ileri işlem adımları bu parçalar üzerine uygulanmıştır. Burada her bir kare alma ve çarpma işlem adımı eşit sürelerde

gerçekleşmediğinden eğri bölütlerini almak için de önce eğrinin zarfını alan bir filtre uygulanıp bu filtre çıkışına göre eğriler yanaştırılmıştır.

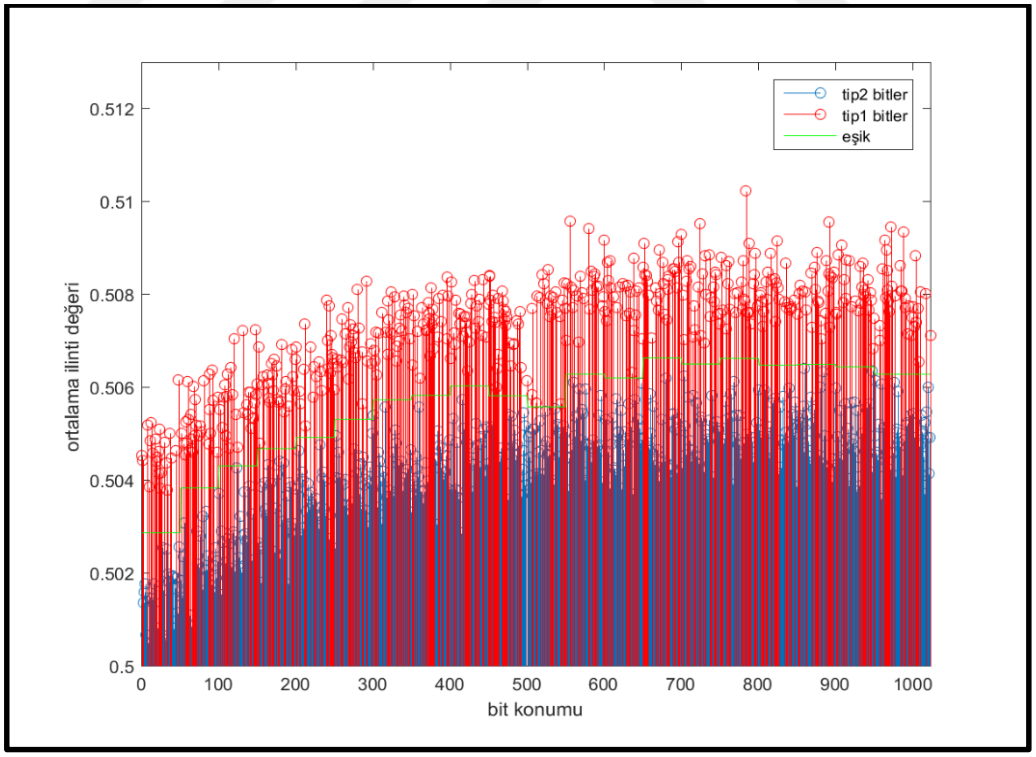


Şekil 3.2. Örnek ölçüm üzerinde bölütlenecek alanları

Oyların toplanmasına dayanan birinci yöntemde, tip0 grubundan herhangi bir bit referans olarak seçilip, bu bite ait güç eğrisi bölütünün tüm diğer bitlere ait alanlar ile çapraz ilinti değerleri hesaplanmıştır. Her bir güç eğrisinden elde edilen çapraz ilinti değerleri, bir eşik değeri ile karşılaştırılarak, eşikten yüksek olan değerler için ilgili bitin tip0 olduğuna dair bir sayaç aksi hale tip1 olduğuna dair bir sayaç değeri artırılmaktadır. Bu işlem tüm eğriler için tekrarlanmaktadır. Daha sonra bu sayaç değerlerinden hangisi daha fazla oy almışsa bitin ilgili tipe ait olduğuna karar verilmektedir. Şekil 3.3.'de, tip0 türünde seçilen bir referans bit için, birinci yönteme göre, artan eğri sayısıyla değeri doğru olarak kestirilebilen bit sayısının değişimi görülmektedir. Burada toplam 40000 eğri kullanıldığında 1024 bitten ancak 1000 tanesi kadarı doğru olarak tespit edilebilmektedir. Şekil 3.4.'de 40000 eğri kullanılarak elde edilen ortalama ilinti değerlerinin ve eşik değerinin genel görünümü verilmiştir. Şekillerde tip0-tip1ve eşik değerleri sırasıyla kırmızı-mavi-yeşil renklerde görülmektedir. Burada referans olarak 500 numaralı bit kullanılmıştır. Genel resme bakıldığında ilinti değerlerinde kullanılan referans bite olan uzaklıklarına göre de ilinti değerleri değişkenlik göstermektedir. Bu durumun [21]' de belirtildiği gibi, güç eğrisinde yan yana olan örnek noktalarda aslında elektronik gürültünün birbiri ile ilintili olmasından kaynaklandığı düşünülmektedir.



Şekil 3.3. İlintiler toplamı ve oyların toplamı yöntemleri



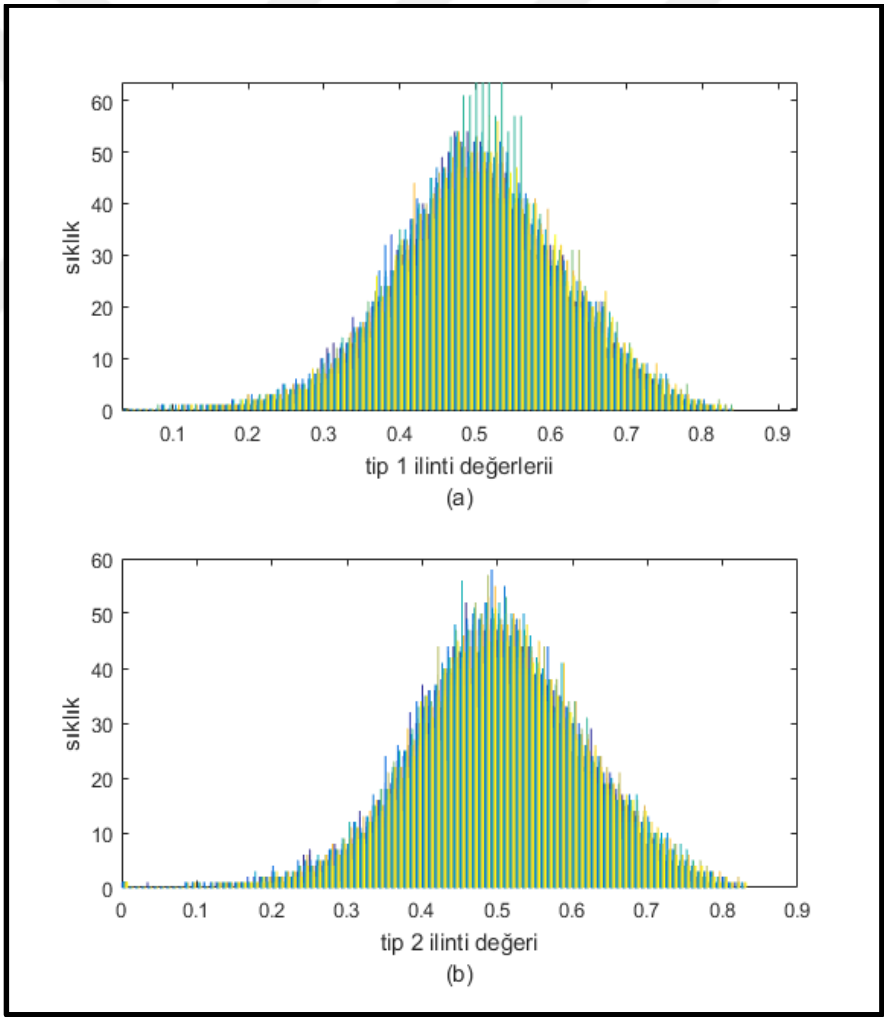
Şekil 3.4. İlintiler toplamı yönteminde 40000 eğri için ortalama ilinti değerleri

Referans bite yakın bitlerin gösterdikleri ilinti değeri, birbiriyle ilişkili gürültü bileşenlerine sahip olmalarından dolayı daha yüksek olmaktadır. Aslında frekans

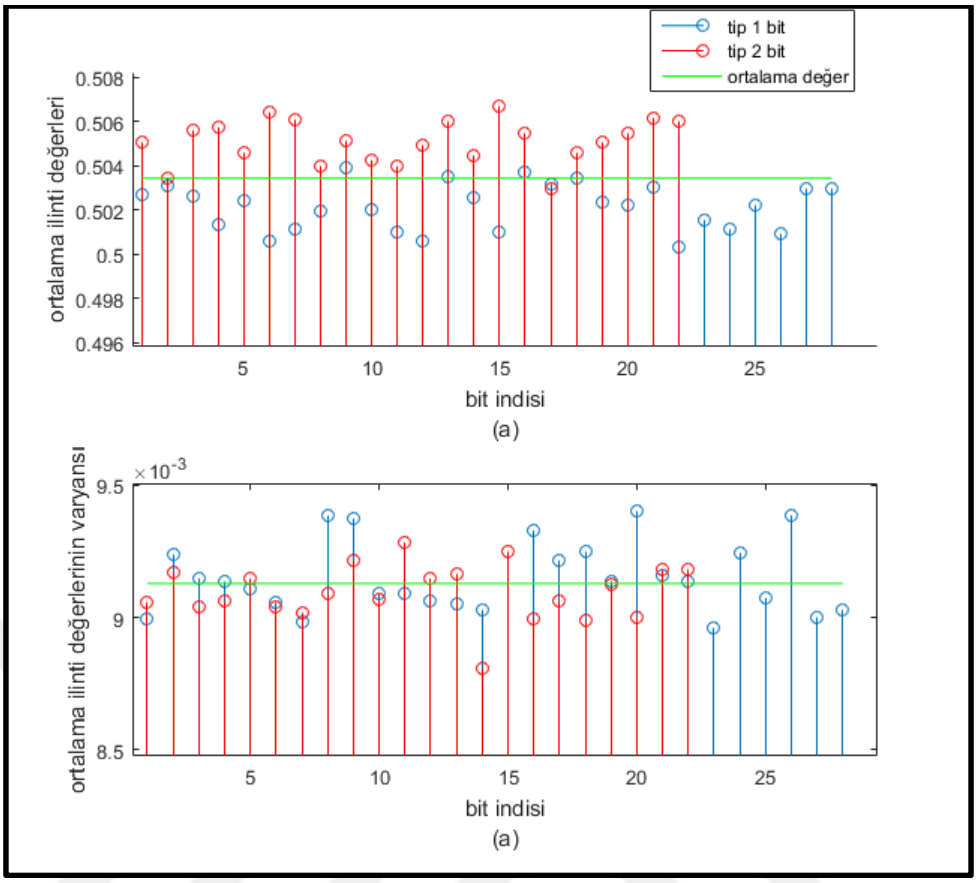
uzayında, gürültünün olduğu frekans bantlarını filtreleyerek gürültüden kaynaklanan etkiden kurtulmak mümkün olabilir. İleriki bölümlerde frekans uzayında çalışılarak ve filtreleme yapılarak bu etkinin azaldığı gösterilmiştir.

3.1.2.1. Gerekli eğri sayısının hesaplanması

Şekil 3.1.'de, ilk 50 bitlik anahtar biti için, tip0 ve tip1 bitlerine ait çapraz ilinti değerlerinin 40.000 ölçüm için hesaplanan histogram tabloları verilmişti. Bu histogramlar genel olarak her bir bit türüne ait ortalma ilint değerinin etrafında, yaklaşık olarak simetrik bir görünüme sahip normal dağılım karakteristikleri sergilemektedir. Bununla birlikte tip0 bit türüne ait histogramda, ortalama değerin daha sağında yoğunlaşmış farklı bir alt bölgenin varlığı da görülmektedir.



Şekil 3.5. 10000 eğri için tip1 ve tip2 bitlere ait ilinti değerlerinin histogramı

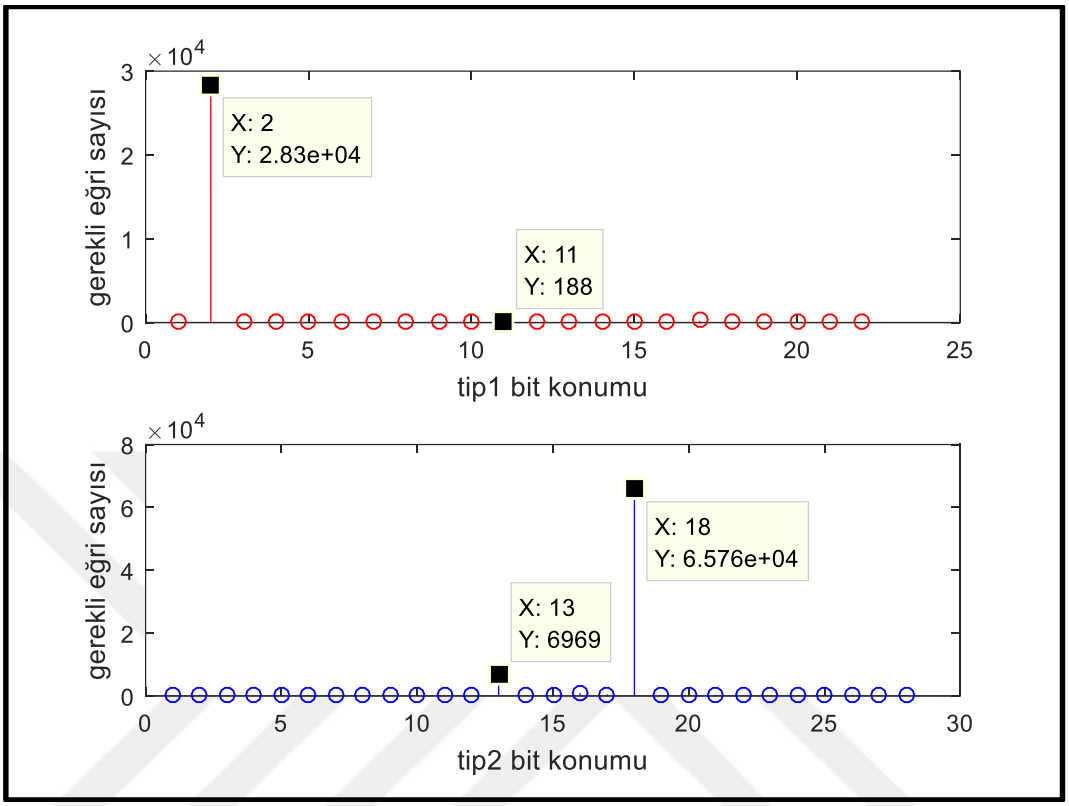


Şekil 3.6. 10000 eğri için ilk 50 bite ait ortalama ilinti ve varyans değerleri

Eğri sayısını hesaplamak, için, öncelikle bu normal dağılımlara ait parametreler olan ortalama ve varyans değerleri nispeten daha az sayıda (örneğin 10000 adet) ölçüm kullanılarak kestirilmiştir. Şekil 3.5.'de kestirimde kullanılan bu 10000 eğriye ait ilinti değerlerinin histogramları görülmektedir. Bu histogramlar da çok az sağa yanaşık olsalar da genel olarak normal dağılıma görüntüsü sergilemektedir. Şekil 3.6.'da 10000 ölçümle hesaplanan tip1-tip2-eşik değerlerine ait ortalama ve varyans değerleri sırasıyla kırmızı-mavi-yeşil renklerde verilmiştir. Ortalama ilinti değerlerinin eşikle olan ilişkilerinden anlaşılacağı üzere tip0 ve tip1 bitlerden bir kısmının değeri 10000 ölçümle doğru olarak kestirilememektedir. Ayrıca tip1 (mavi) türündeki bitlere ait ilinti değerleri daha büyük varyans değerlerine sahip olup tip1 türünde değeri doğru olarak kestirilmeyen bit sayısı daha fazladır. Aslında bu varyans değerleri de farklı tipler arasında ayırt edici bir özellik olduğundan bit tipi kestiriminde de kullanılabileceği düşünülmektedir.

Şekil 3.7.'de 10000 eğriden elde edilen istatistiksel parametreler ve (3.12) eşitliği kullanılarak, ilk 50 biti doğru olarak kestirebilmek için gerekli eğri sayılarının ne

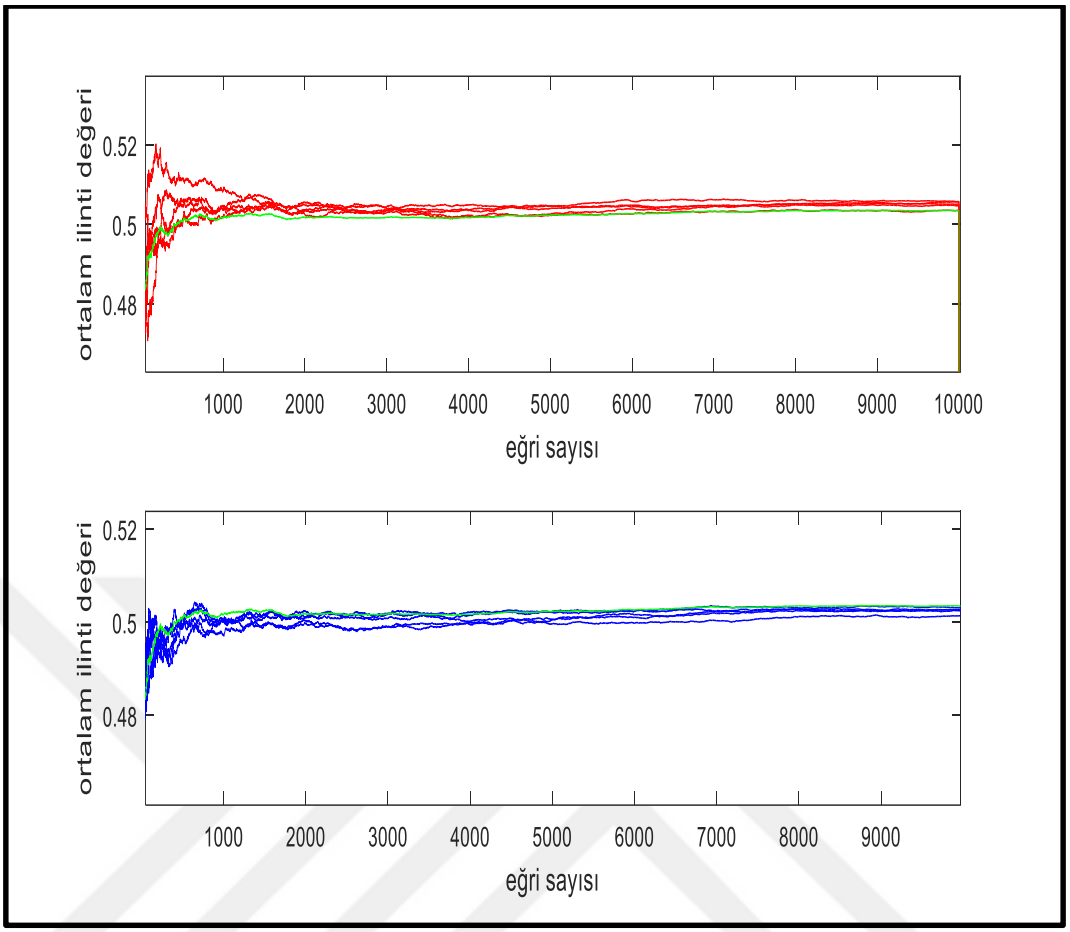
olması gerektiğine dair hesaplama sonuçları verilmiştir. Bu şekilden görüleceği gibi gerekli en yüksek eğri sayısı 65000 mertebesindedir.



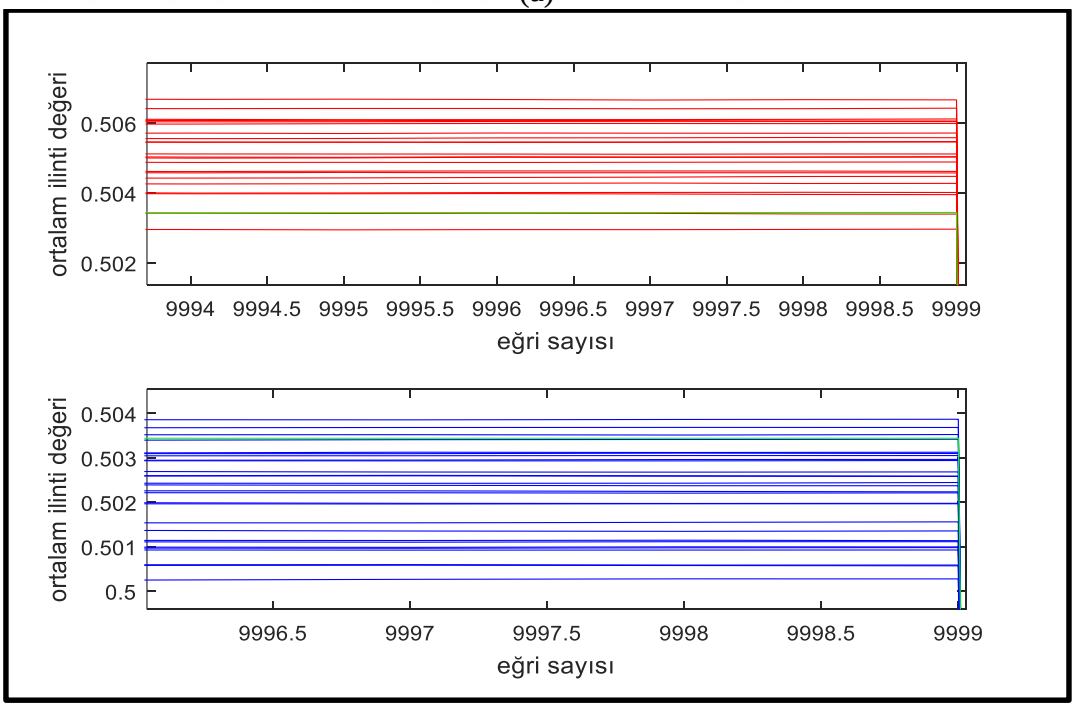
Şekil 3.7. 10000 eğri kullanılarak gerekli eğri sayısının tahmini

Şekil 3.8.'de ilk 50 bit için artan eğri sayısıyla değişen ortalama ilinti değerlerinin tüm seyri (a) ve 10000 eğri civarına ulaşıldığı zamanki yakından görünümü (b) verilmiştir. Eğri sayısı artırıldıkça ortalama ilinti değerlerinin varyansı azalmakta ve şekilde kırmızı ile gösterilen tip0'lara ait ortalama ilinti değerleri, yeşil renk ile gösterilmiş olan eşiğin üstünde yoğunlaşmaktadır. Aynı şekilde mavi renkte gösterilmiş olan tip1 bitlerine ait ilinti değerleri ise eşiğin altında yoğunlaşmaktadır. Bu resimden anlaşılacağı gibi 10000 eğri kullanıldığında tip0 türünde "1 adet", tip1 türünde ise "3" adet bitin değeri doğru olarak kestirilememiştir.

Şekil 3.9.'da ise hesaplanan eğri sayıları açısından en kötü performansa sahip yani değeri en fazla eğri kullanımı gerektiren (a) ve en iyi performansa sahip yani değeri en hızlı şekilde kestirilebilen (b) tip0 ve tip1 türünde ikişer bitin ortalama ilinti değerinin artan eğri sayısıyla değişimi görülmektedir. Hesaplanan gerekli eğri sayısı pratik ile kısmen uyumludur denebilir.

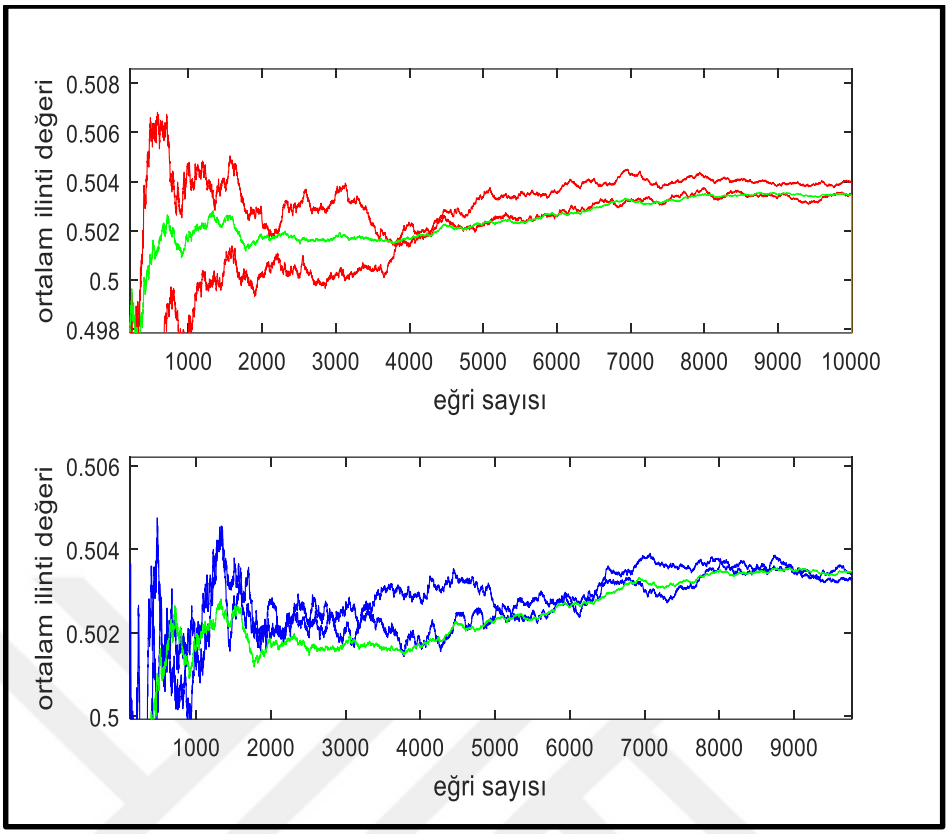


(a)

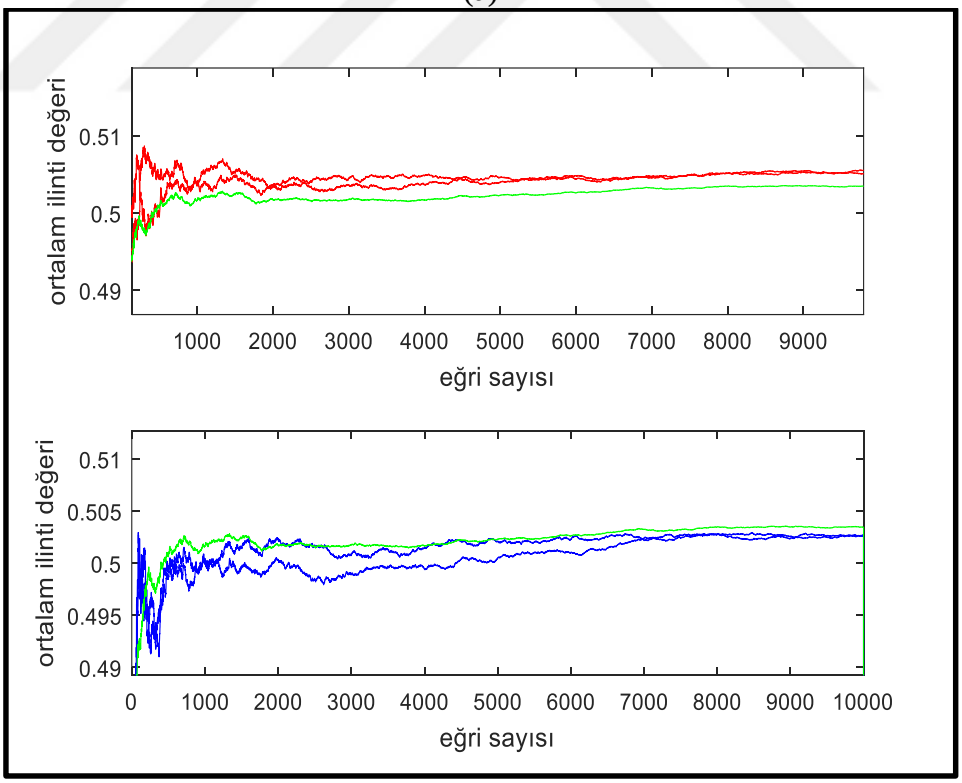


(b)

Şekil 3.8. İlk 50 bit için artan eğri sayısıyla değişen ortalama ilinti değerleri



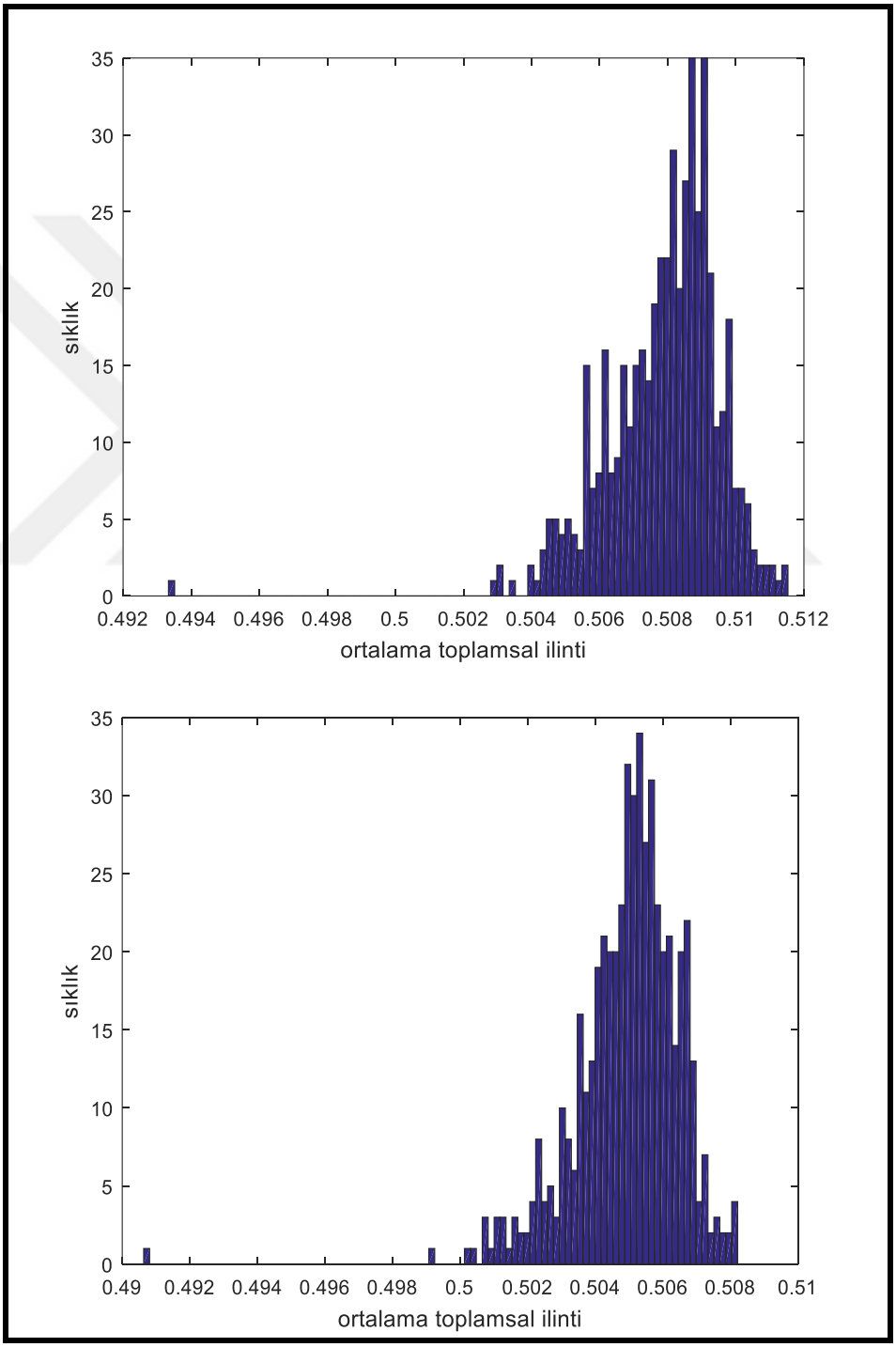
(a)



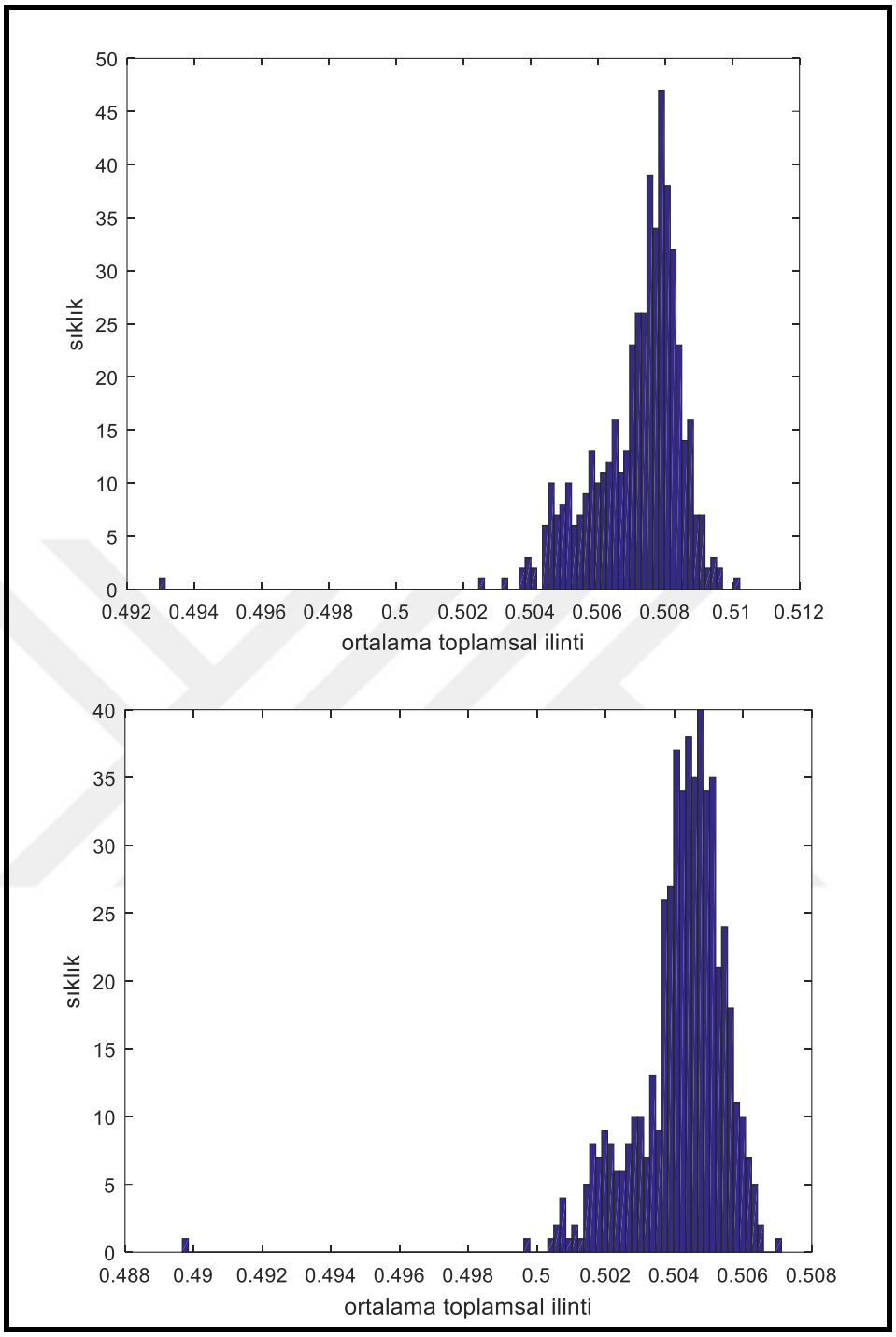
(b)

Şekil 3.9. En kötü ve en iyi performansa sahip iki bite ait ortalama ilinti değerlerinin değişimi

Şekil 3.10.'da ise 10000 ve 40000 eğri için hesaplanan tüm tip0 ve tip1 bitlere ait ortalama ilinti değerinin histogramı görülmektedir. Bu dağılımlar tam bir normal dağılıma sahip değildir ve eğri sayısı arttıkça gerçek ilinti değerinin olduğu yöne doğru dikleşme oluşmaktadır. Aslında komşu bitlere ait dağılımlardaki bu bozukluklar eşik değeri hesaplamasında komşu bitlerin ortalaması kullanıldığından bu değerlere yansımaktadır ve hesaplarda hataya neden olan etkenlerden biridir.



(a)

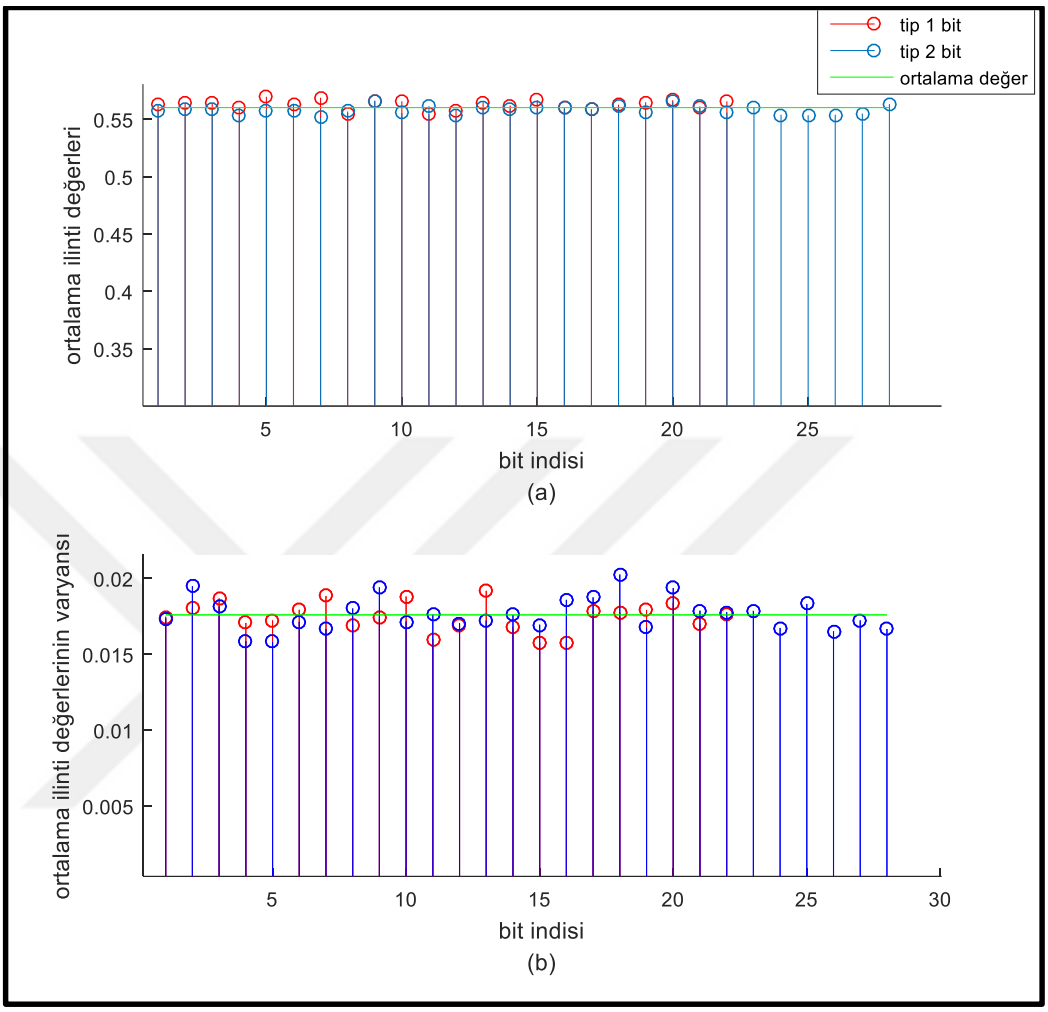


(b)

Şekil 3.10. 10000 ve 40000 eğri için ortalama ilinti değerlerinin histogramı

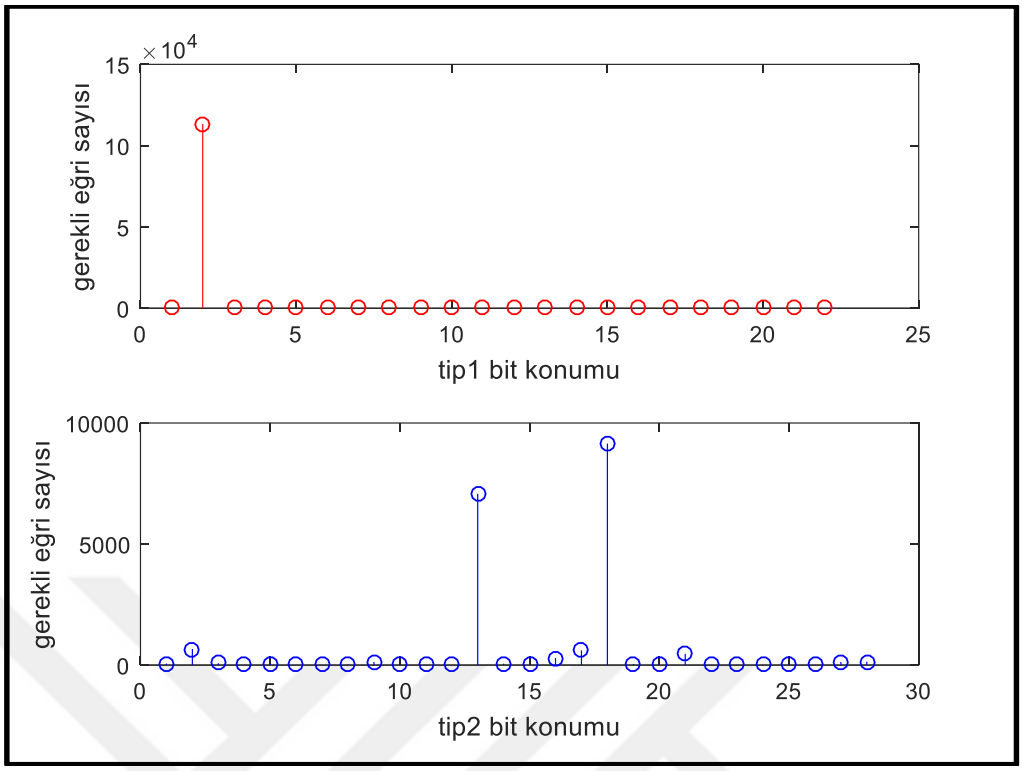
Şekil 3.11.'de ilinti değerlerinin Fisher-Z dönüşümü alındıktan sonra, bu dönüştürülmüş değerler üzerinden hesaplanan ortalama ilinti ve varyans değerleri görülmektedir. Bu sonuçlar, Şekil 3.7.'deki normal zaman örnekleri ile hesaplanan eş değerleri ile karşılaştırıldığı zaman, uygulanan dönüşümün aslında daha kötüleştirici

bir etkiye neden olduğu görülmektedir. Örneğin eşik değerine göre yanlış konumda olan bit sayıları artmış, bunun yanı sıra varyans değerlerinde bir artış olmuştur.



Şekil 3.11. 10000 eğri için Fisher-Z değerleri üzerinden ilinti değerlerinin ortalama ve varyans değerleri

Şekil 3.12.'de, 10000 eğriye ait çapraz ilinti değerlerine Fisher-Z dönüşümü uygulandıktan sonra eşitlik (3.12) kullanılarak hesaplanan gerekli eğri sayıları görülmektedir. Dönüşüm uygulanmış bu değerlerle yapılan hesaplarda her bir bitin tipini doğru olarak kestirebilmek için gerekli eğri sayısında, dönüşüm uygulanmamış duruma göre artış olduğu gözlenmektedir. Bu durum Fisher-Z dönüşümünden sonra varyanslarda gözlenen artıştan kaynaklanıyor olabilir. Sonuç olarak dönüşüm uygulanmadan çapraz ilinti değerlerinin doğrudan ortalaması alınarak yapılan kestirim daha isabetli olmaktadır.



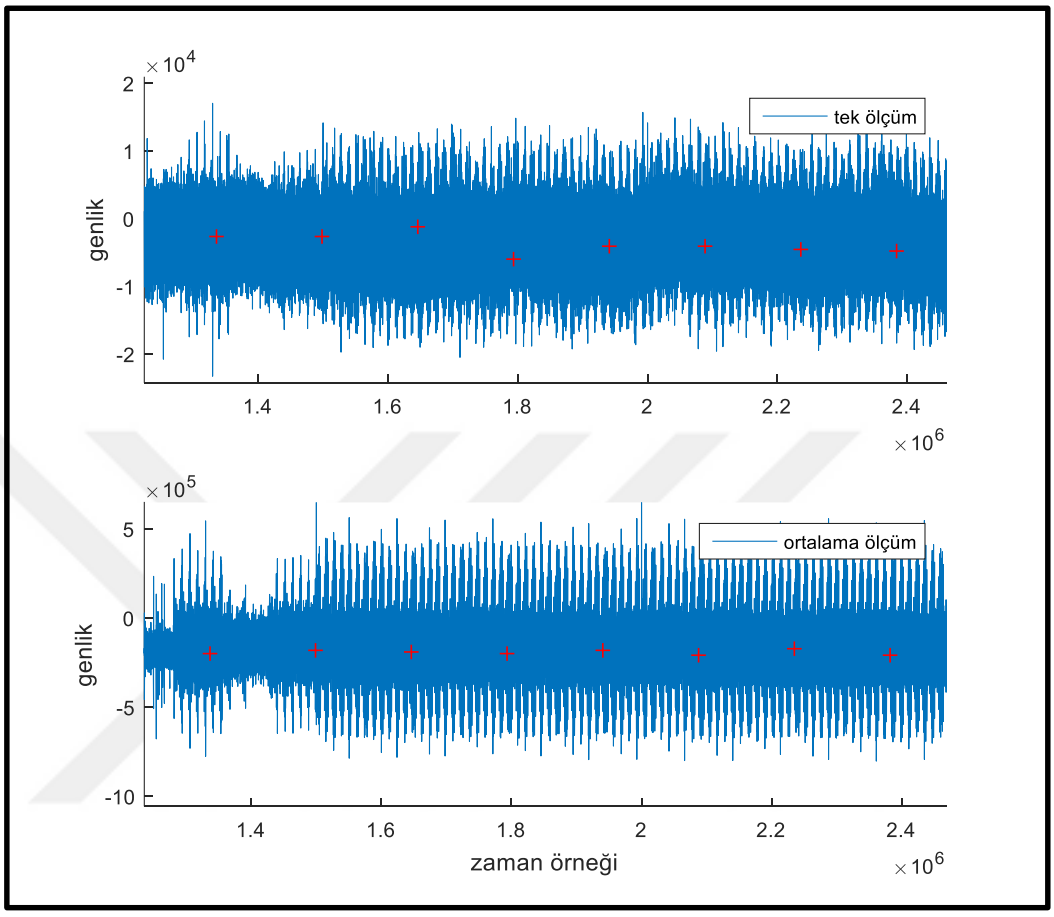
Şekil 3.12. 10000 eğri için Fisher-Z dönüşümü uygulanarak gerekli eğri sayısının kestirimi

3.1.3. Yöntemin FPGA ikilik üs alma devresine uygulanması

Çapraz ilinti analizi yöntemin uygulandığı hedef devrelerden bir diğeri, üs alma algoritması olarak, Algoritma 2.5.'de verilen ve hep çarpma türü ikilik üs alma algoritmasını kullanan FPGA tabanlı bir RSA gerçeklemesidir.

Hep çarpma türündeki bu algoritmayı gerçeklemek için, her bir bit için yapılacak çarpma -kare alma işlemlerinde, soldan sağa doğru işlem görecek bit indisini tutan bir sayaca göre anahtar bit değeri okunmakta ve Montgomery çarpıcısının yani MontMul fonksiyonunun girdileri ilgili anahtar bitine göre değişmektedir. Sayaç değerinin değiştiği anlarda yani bir sonraki bit işlemlerine geçildiği zaman, işlem görmüş en son bitin değeri "1" ise, sonraki bit için gerçekleşecek kare alma işleminde, çarpıcının ilk girdisi mevcut ara değer olmaktadır. Böylece bu kütüğün içeriği, dolayısıyla kütüğün "Hamming Distance (HD)" değeri değişmezken, ikinci girdiye de aynı ara değer yüklenmektedir. İşlem görmüş en son bitin "0"olduğu durumda ise, mevcut hesaplama sonucu kullanılmayıp daha önce kaydedilen ara değerin kullanılması gerekmekte ve böylece ilk girdiye ait kütüğünün "Hamming Distance (HD)" değeri değişmektedir. Bu durum güç eğrilerinde de bir etkiye sahip olacağından bölütlenmiş güç eğrilerinde

“0” değerli bitler ile “1” değerli bitlere ait alanların birbirinden ayırt edilmesi mümkün olmaktadır.

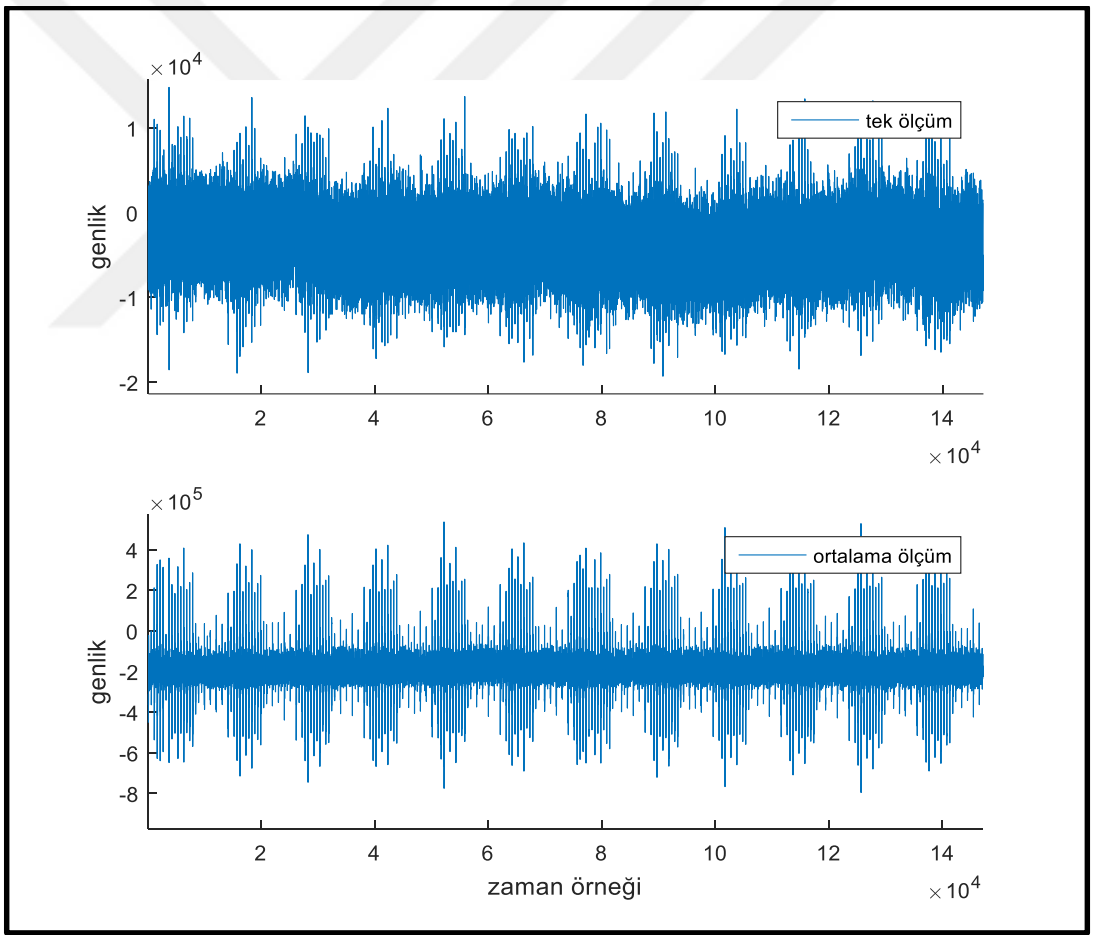


Şekil 3.13. Tek ve ortalama ölçümler üzerinde bölütlenecek alanlar

Şekil 3.13.’de bir RSA işlem adımının gerçekleşmesi sırasında, Sakura Kart üzerinden alınan güç eğrisi kesiti görülmektedir. Ölçümün ilk bölümü, RSA işleminin girdisi olan verinin Montgomery uzayına alındığı ilk uzun çarpma adımıdır. Bu işlem, tüm diğer modüler uzun çarpma ve kare alma işlem adımları gibi ince çubuklarla ayırt edilen 6 alt adımından oluşmaktadır. Daha sonra ise üs alma döngüsünün başladığı ilk kare alma işlemine ait 6 ince çubuk görülmektedir. Bu ilk kare alma adımında bir değerinin Montgomery kalanının karesi alındığından, diğer kare alma/çarpma adımlarından farklı bir güç eğrisine sahiptir. Sonraki alanlar ise standart işlem adımlarıdır ve kare alma ve çarpma arasında çok belirgin farklar görülmemektedir.

ÇİA saldırısı uygulamak amacıyla RSA işlemine ait eğrileri kaydedildikten sonra, her bir bit işlemine ait kare alma ve çarpma işleminin gerçekleştiği anların güç eğrilerinden

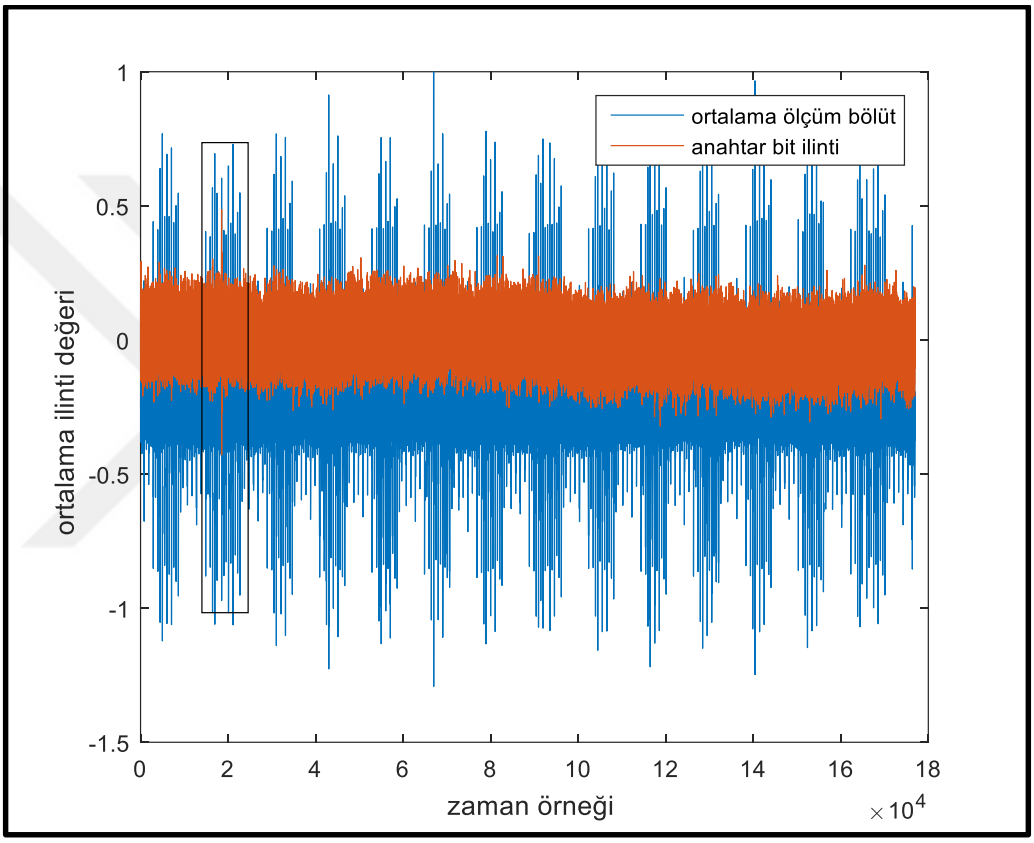
bölütlenmesi gerekir. Şekil 3.14.'de güç tek bir bit için gerçekleştirilen kare alma ve çarpma adımlarına ait güç eğrisi bölütü görülmektedir. Çapraz ilini analizinin öncesinde, anahtar bitleri ile en ilinti alanları görmek amacıyla, öncelikle anahtar bitlerinin doğrudan değeri ile eğri bölütlerinin ilintisine bakılmıştır. Şekil 3.15.'da görülen ilk resim, anahtarın 192 bitinin üs alma işlemleri sırasında gerçekleşen modüler uzun çarpma işlemine ait güç eğri bölütleri ile ilintisini göstermektedir. Çapraz ilinti analizinde, tüm eğri bölütlerini kullanmak yerine bu alanların kullanılması analizin başarımını artırmaktadır. Uygulamada ise eğer aynı gerçekleme hem açık hem özel anahtar işlemlerinde kullanılmakta ise, saldırgan açık anahtar değerini kullanarak gerçekleştirebileceği bu tür bir analiz ile çapraz ilinti uygulanacak alanı başarılı bir şekilde tespit edebilir.



Şekil 3.14. Bir bit için kare alma ve çarpma işlemi güç eğrisi

Şekil 3.16.' da rastgele veri girişleri ile oluşturulmuş ikilik üs alma tipi FPGA gerçeklemesine ait çapraz ilinti değerleri görülmektedir. Burada "kırmızı " ile işaretlenmiş olanlar, değeri 0 olan anahtar bitlerine ait çapraz ilinti değerleridir. İlk

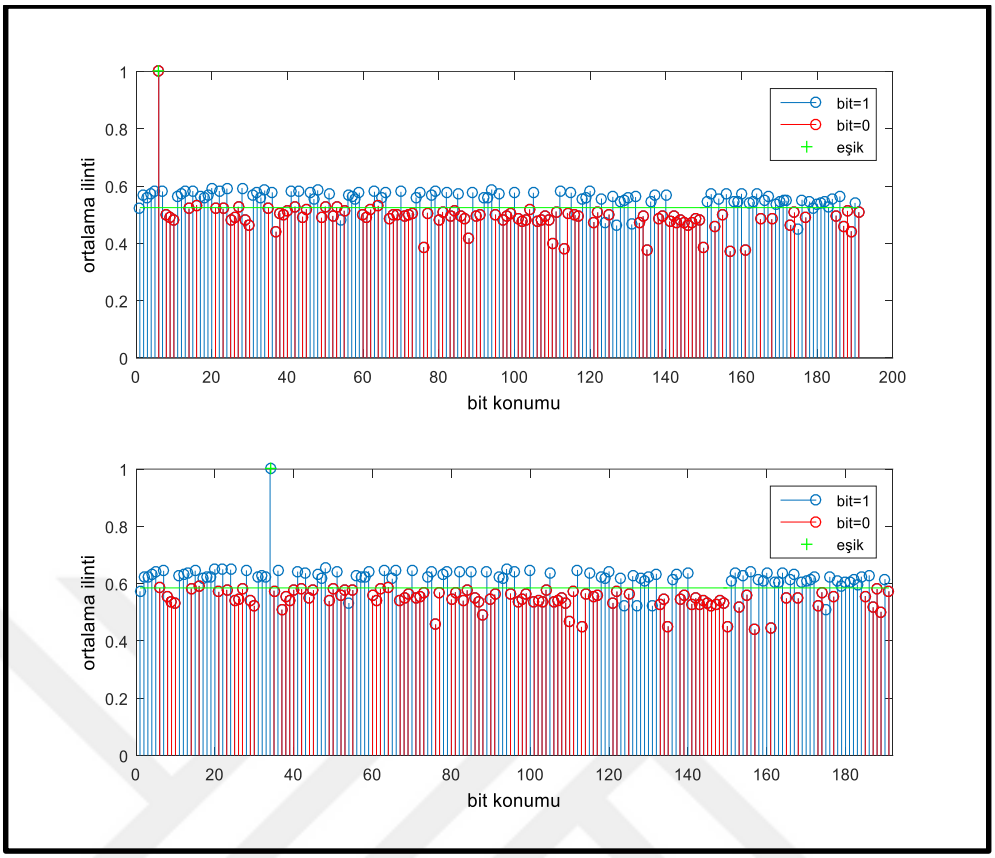
resimde, değeri ‘0’ olan ikinci resimde ise değeri ‘1’ olan referans bit kullanılmıştır. Hem referans bitin 0 hem de 1 olduğu durumlarda, değeri 0 olan bitlerin referans bit ile ilintisinin düşük, değeri 1 olanlarınkinin ise yüksek olduğu görülmektedir. Sonuç olarak tek bir referans bitin kullanıldığı durumda, farklı anahtar bitlerine ait çapraz ilinti değerleri birbirinden ayrışabilmektedir. Bundan dolayı FPGA tabanlı ikilik üs alma gerçeklemesine de “tüm bitler çapraz ilinti analizi” yönteminin uygulanabileceği sonucuna varılmaktadır.



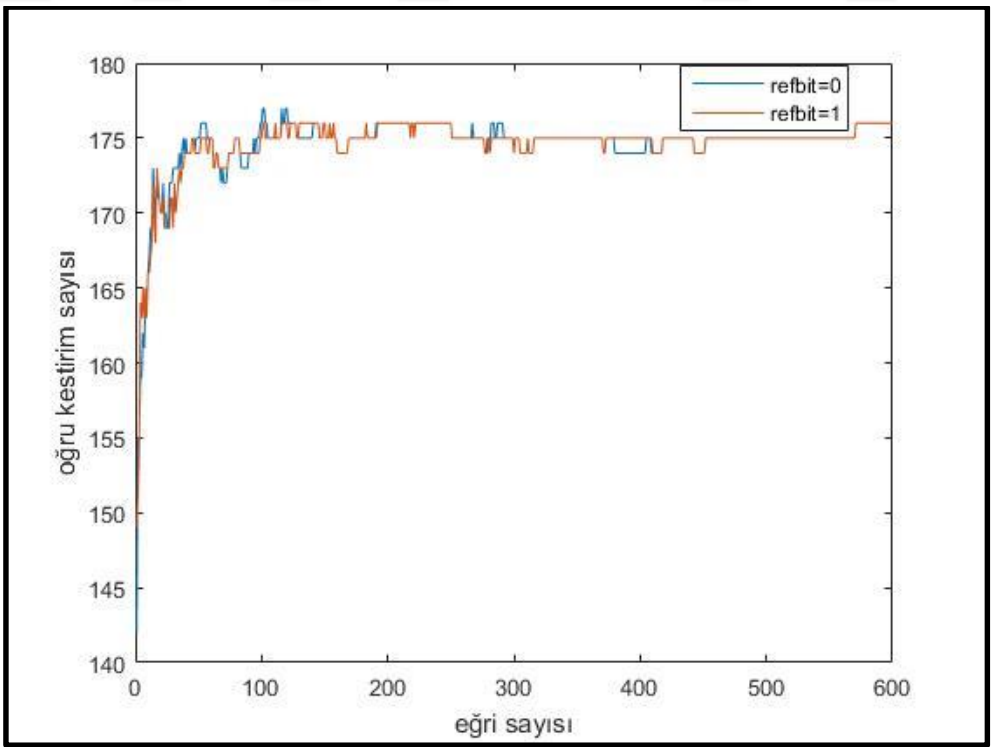
Şekil 3.15. Anahtar bitleri ile güç eğrileri ilintisi

Şekil 3.17.’de ise değeri “0” ve “1” olana referans bitler için, kullanılan ölçüm sayısı ile değeri doğru olarak kestirilen bit sayısının değişimi görülmektedir. Şekilde kırmızı ile gösterilen eğri referansın tip0 türünde, mavi ile gösterilen eğri ise referansın tip1 türünde olması durumundaki sonuçlardır. Her iki durumun da birbirine yakın sonuçlar verdiği görülmektedir.

Şekil 3.18.‘ de ise aynı hedef devrenin sabit veri için oluşturulan çapraz ilinti analizi sonuçları verilmiştir. Burada “kırmızı ” ile işaretlenmiş olanlar, değeri 0 olan anahtar bitlerine ait çapraz ilinti değerleridir.

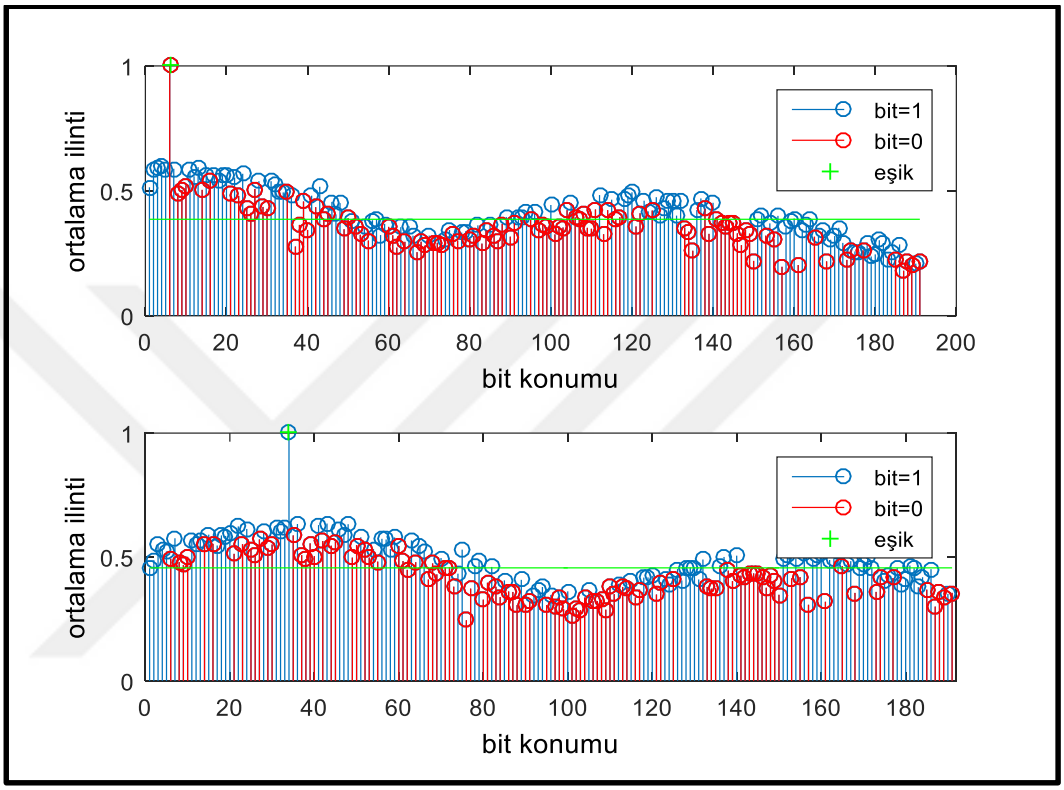


Şekil 3.16. Tek bit çapraz ilinti analizi, rastgele veri



Şekil 3.17. Artan eğri sayısıyla değeri doğru olarak kestirilebilen bit sayısı

İlk resimde, değeri ‘0’ olan ikinci resimde ise değeri ‘1’ olan referans bit kullanılmıştır. Buradan görüleceği gibi “0” değerli bitlerin yine her iki referans için de daha düşük ilinti değeri alması söz konusu olsa da yöntemin başarımı oldukça düşüktür. Aslında bitler tam olarak ayırt edilememekte ve eğri sayısını artırmanın da bir faydası olmamaktadır.



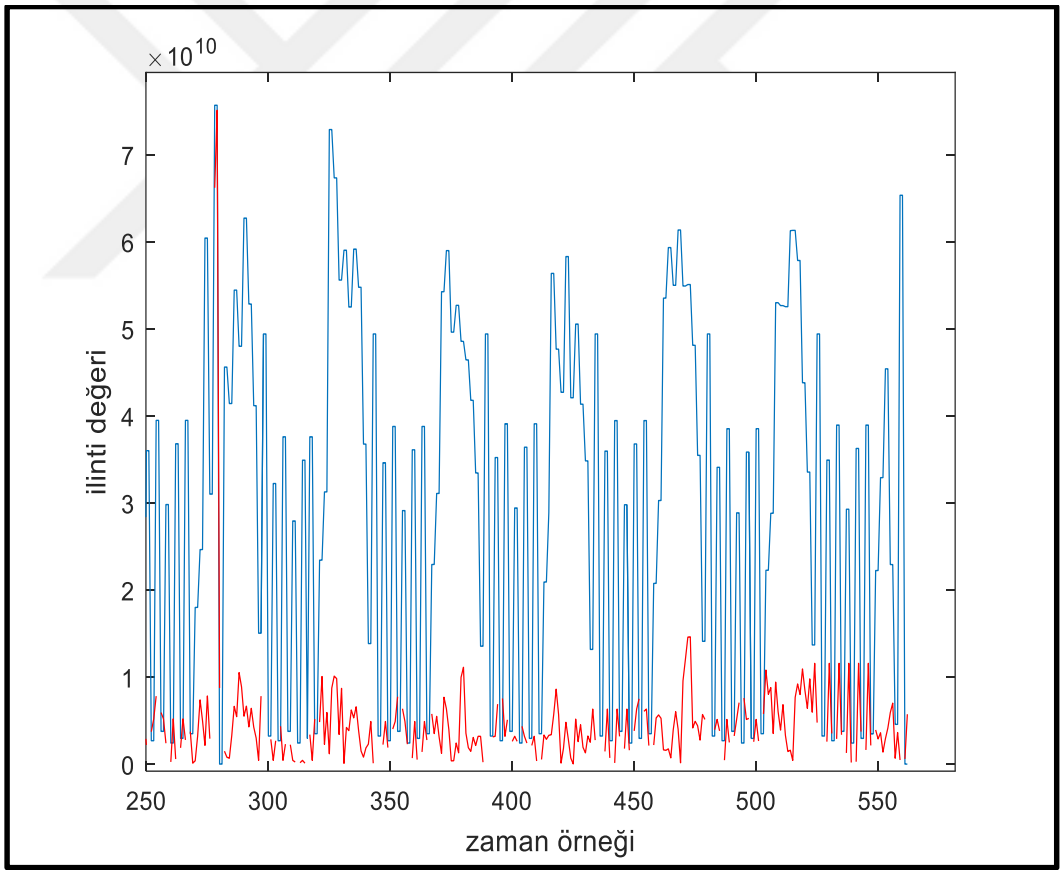
Şekil 3.18. Tek bit çapraz ilinti analizi, sabit veri

3.1.4. Yöntemin FPGA benzetim eğrilerine uygulanması

Yöntemi benzetim eğrilerine uygulamak için, tıpkı gerçek güç eğrilerinde olduğu gibi, tam bir RSA koşturumuna ait benzetim eğrisinde, her bir bit için gerçekleştirilen çarpma işlemlerine ait alanlar bölütlenmiştir. Şekil 3.19.’da, örnek bir güç eğrisi bölütü ile, tüm anahtar bitlerine ait bu eğri bölütleri ile doğrudan anahtar bitleri arasında hesaplanan ilinti eğrisi görülmektedir. Bu analiz özellikle çapraz ilinti değerlerinin güç eğrisi bölütlerinin hangi noktası etrafında hesaplanması gerektiği konusunda ipucu vermektedir. Şekil 3.20. ise yine bölütlenmiş benzetim değerleri ile gerçekleştirilen çapraz ilinti analizi sonuçları görülmektedir. Burada “kırmızı +” ile işaretlenmiş olanlar, değeri 0 olan anahtar bitlerine ait değerleridir. İlk resimde, değeri ‘0’ olan ikinci resimde ise değeri ‘1’ olan referans bit kullanılmıştır. Her iki durumda da

referans bit ile aynı tipte olan anahtar bitlerine ait örneklerin yüksek, farklı tipte olanların ise düşük ilinti değerine sahip olduğu görülmektedir. Ancak referans bitin "0" değerli bitlerden seçildiği durumda, ilinti değerlerinin daha iyi ayrıştığı görülmektedir. Bu durum, değeri 0 olan yani tip0 türündeki bitlere ait çapraz ilinti değerlerinin standart sapmasının, değeri 1 olan yani tip1 bitlerine ait değerlerden daha yüksek olmasından kaynaklanmaktadır.

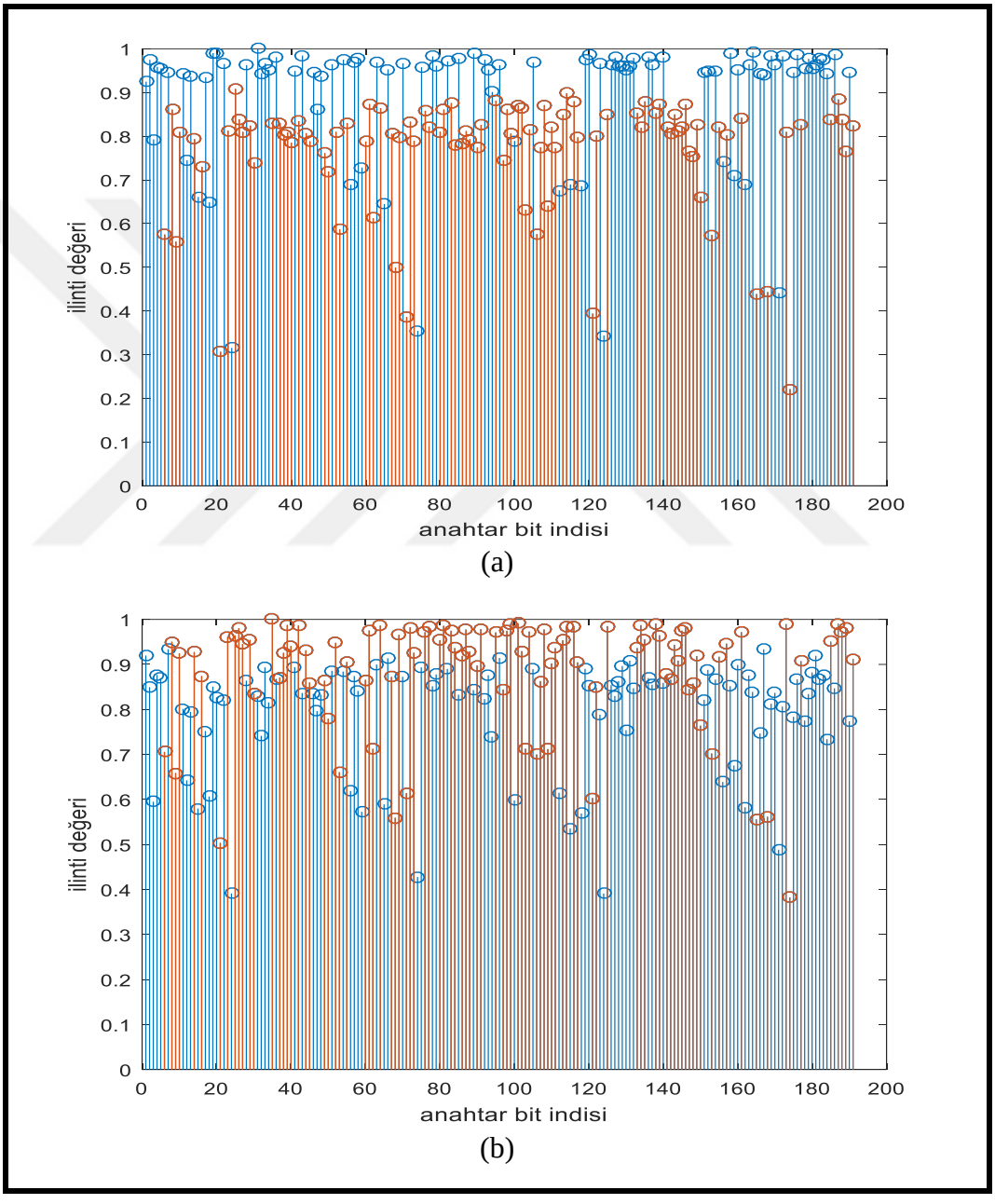
Şekil 3.21.'de "yerleştirme-bağlama (place route)" seviyesinde benzetim çıktısı olan VCD dosyasından elde edilen 24 bitlik iki ayrı RSA koşturumuna ait ölçüm bölütlerinin, doğrudan anahtar bitleri ile ilintisi verilmiştir. Şekilden görüleceği gibi ilk eğride mantıklı bir konumda bir ilinti değeri görülse de ikinci eğride ayırt edici hiçbir değer görülmemektedir.



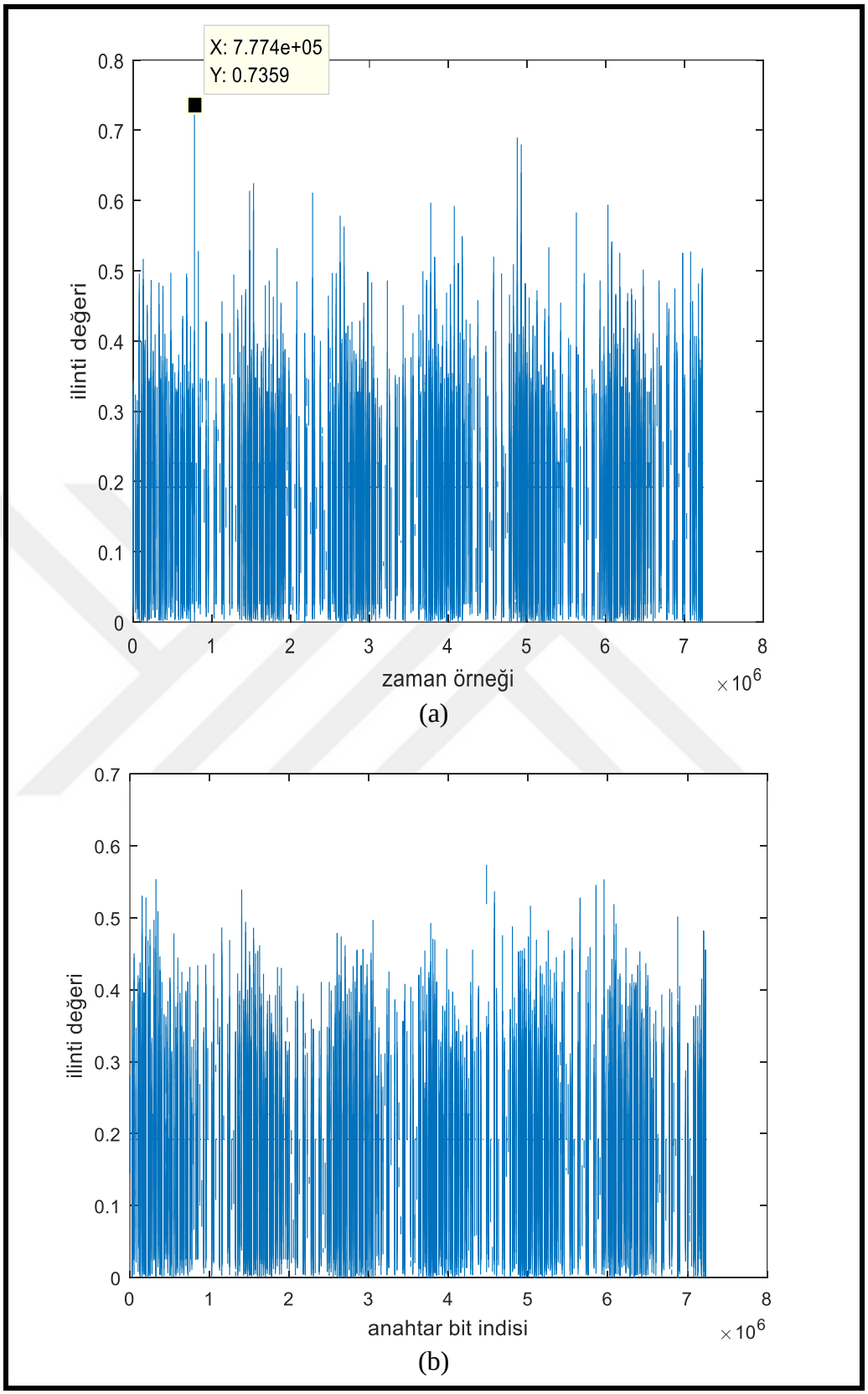
Şekil 3.19. Davranışsal benzetim eğri bölütlerinin anahtar bitleri ile ilintisi

Bu eğriler ile gerçekleştirilen çapraz ilinti analizinde de herhangi bir başarım görülmemiştir. Bunun nedeni 24 bitlik RSA işlemi güç tüketiminin çok düşük boyutlarda kalması olabilir. 24 bitlik RSA işleminde, Montgomery indirgeme

adımlarında bir kerede 4 bitlik indirgeme yapılmakta olup toplamda 6 adımda indirgeme adımları tamamlanmaktadır. Bir kerede işlem gören bit sayısı çok az olduğundan güç tüketimi de çok düşük boyutlarda kalmaktadır. Zaten osiloskopla alınan ölçümlerde de çok düşük genlikli eğriler elde edilmiştir. Osiloskop eğrilerinde tek bir ölçüm için güç tüketimi çok belirsiz olup ancak aynı işlem eğrilerinin ortalaması alınarak işlemler bir miktar görünür hale getirilebilmektedir.



Şekil 3.20. Davranışsal seviye benzetim eğrisinde tip0 ve tip1 türü referanslara ait çapraz ilinti değerleri



Şekil 3.21. Yerleştirme-bağlama (place-route) seviye benzetim eğrilerinin anahtar bitleri ile ilintisi

3.1.5. Yönteme ilişkin deneysel bulgular ve tartışma

Bu çalışmada, geliştirilen çapraz ilinti tabanlı özgün yöntem, hem ASIC ML tabanlı hem de FPGA'de ikilik üs alma yöntemlerini içeren RSA gerçeklemelerine uygulanmıştır. ASIC ML devresinde anahtar bitlerinin tipine göre işlem girdilerinin farklı bellek alanlarından okunmasından kaynaklanan farklılıkların yakalanması hedeflenmiştir. Saldırıyı gerçekleştirmek için tip0 türünde, yani kendinden sonraki bit ile aynı değere sahip olan bir bit referans seçilerek diğer anahtar bitlerine ait eğri alanlarının bu bite ait alan ile çapraz ilinti değeri hesaplanmıştır. Elde edilen çapraz ilinti değerleri iki farklı yöntemle değerlendirilmiştir: Birinci yöntemde farklı RSA koşturumlarında her bir bit için elde edilen çapraz ilinti değerleri birbiri ile toplanıp, tüm bitlerin ilinti değerlerinin kayan ortalaması ile karşılaştırılmıştır. Anahtar bitinin tipine bu karşılaştırma sonucuna göre karar verilmiştir. Buradaki önemli bir nokta referans bitin tip0 türünde seçilmiş olmasıdır. Çünkü tip0 ve tip1 referans bölgeleri arasında, sonraki bölümde açıklanacağı üzere davranış farklılıkları bulunmaktadır. İkinci değerlendirme yönteminde ise her bir RSA koşturumu için elde edilen çapraz ilinti değerleri, o koşturuma ait ilintilerin kayan ortalamasından elde edilen eşik ile karşılaştırılarak bitin tip0-tip1 sayaçlarından biri artırılmaktadır. Daha sonra her bir RSA koşturumundan elde edilen sayaçlar karşılaştırılarak bitin nihai tipine karar verilmektedir. Deneyler 40000 eğri ile gerçekleştirilip sonuç olarak birinci değerlendirme yönteminde bitlerin tamamının tipi doğru şekilde anlaşılabilirken ikinci değerlendirme yönteminde ise 1023 bitten yaklaşık 1000 tanesinin türü doğru olarak kestirilebilmiştir. Yani birinci değerlendirme yöntemi daha başarılıdır. Öncelikle ölçümü alınan her RSA koşturumunda, farklı veri girişi kullanıldığından, referans bitin hedef bit ile olan çapraz ilinti hesabı, her seferinde farklı bir HD değerini kullanmış olur. İlk bölümde anlatıldığı gibi bu durum, belli bir güç eğrisinde bulunan veri bağımlı anahtarlama gürültüsünün ve ayrıca her bir devrede bulunan anahtarlama gürültüsünün çapraz ilinti hesaplamasına yansıyan etkilerinin azalmasını sağlayacaktır. Sonuç olarak ilintiler ortalamasına dayanan yöntem ile referans bit ile hedef bit arasında var olması beklenen gerçek çapraz ilinti (cross correlation) değerine daha iyi bir yaklaşıklık yapılmaya çalışılmaktadır.

Saldırının uygulandığı bir diğer hedef is FPGA'li devrenin gerçek güç eğrilerinin yanı sıra, davranışsal seviye benzetim eğrileri olmuştur. Gerçekleştirilen saldırıda anahtar

bitlerinin tipine göre işlem girdilerinin yazıldığı kütüklerin ‘HD’ sinin değişip değişmediğinin tespit edilmesi hedeflenmiştir. Bu durum tespiti anahtar bitinin tespit edilmesi ile özdeştir.

Veri girişinin saldırı üzerindeki etkisini anlamak amacıyla FPGA’li devrede sabit bir veri değerine karşılık düşen gerçek eğrilerle yapılan analizde, çok düşük bir başarım elde edilmiştir. Yani yöntemin gerçek eğriler üzerinde işe yaraması için, eğrilerin farklı girişlere ait RSA koşturumlarından elde edilmiş olması gerekmektedir. Bu durum mesaj körleştirme karşı önleminin, saldırı üzerinde engelleyici olmak bir yana olumlu yönde, yani saldırıyı kolaylaştırıcı etkisinin olduğunun göstergesidir.

FPGA devresine ait gerçek eğriler için yapılan testlerde, ASIC ML yönteminde tip0 olarak nitelendirilen ve 0-0 ya da 1-1 örüntülerinin ilki olan bitlere ait davranışın, bu gerçeklemede değeri 1 olan bitler için geçerli olduğu saptanmıştır. Davranışsal seviye tek bir benzetim eğrisi ile yapılan analizde, değeri 0 ve 1 olan bitlerin kendi türleri ile olan çapraz ilintilerinin ortalamadan yüksek, karşıt tip ile olan ilintilerinin ise ortalamadan düşük değerlerde olduğu görülmüştür. Bu davranış gerçek eğrilerde saptanan davranıştan farklıdır. Bu durum gerçek ölçümlerden toplanmış güç eğrilerinde görülen çapraz ilinti davranışlarının, sadece algoritma girdileri ve algoritma akışından değil, alttaki elektronik devre özelliklerinden de kaynaklandığını göstermektedir.

Bunun yanı sıra FPGA devresine uygulamada, anahtar bitlerinin ancak %98’i elde edilebilmektedir. FPGA’li devrede anahtar bitlerinin tamamının elde edilememesi, ilgili anahtar parçalarına karşılık düşen devre parçalarının, sentezleme, yerleştirme-bağlama aşamalarında farklılıklara uğramış olmasından kaynaklanabilir. Bunun yanı sıra kullanılan anahtar değerine bağlı olarak belli anahtar bitleri için beklenen ilintiyi oluşturacak değerler tesadüfen ortaya çıkmıyor olabilir.

FPGA’li devrede tüm bitler elde edilememiş olsa da, ASIC devrede aynı oranda bitin elde edilmesi için kullanılan eğri sayısı ile karşılaştırıldığında çok daha azdır. Eğri sayısının az olmasının hem hedef devreden hem de kullanılan ölçüm düzeneklerinin farklarından kaynaklanan nedenleri olabilir. Sakura kartı, YKA saldırılarını gerçeklemek amaçlı bir devre olup, sağlanan ölçüm noktasından temiz bir şekilde sinyal alınabilmektedir. Bunun yanı sıra FPGA devresinden alınan ölçümlerde, hedef

devrenin saat frekansından yaklaşık 50 kat daha yüksek oranlarda örnekleme hızı kullanılmışken, ASIC devrede bu oran ancak hedefin 5 katı civarındadır. ASIC devreden alınan ölçümlerde 8 bit, FPGA'li devreninkinde ise 12 bitlik osiloskoplar kullanılmış olup, FPGA'li devreye ait eğrilerde daha düşük kuantalama hatası bulunmaktadır.

Çapraz ilinti tabanlı bu saldırı yöntemi için, hesaplamalarda kullanılan ilinti değerleri yaklaşık olarak normal dağılıma sahiptir. Bu nedenle gerekli eğri sayısının hesaplanmasında kullanılan normal dağılım parametreleri olan ortalama ve varyans değerleri, öncelikle ilinti değerlerine bir dönüşüm uygulanmadan hesaplanmıştır. Eğri sayısını hesaplamak için ise, her biri kendi ortalama ve varyans değerlerine sahip farklı tiplerdeki bitlere ait örnek ortalamasının, eşik değerine ait ortalamadan ayırt edilebildikleri güven aralığını hesaplamaya dayalı bir yöntem izlenmiştir. Hesaplamalarda standart normal dağılım değerlerini içeren Fisher-Z tablolarından faydalanılmıştır. Teorik hesaplamalardan elde edilen sonuçlar ile pratik sonuçlar kısmen uyumludur. İlinti değerlerinin ortalamasını hesaplamada kullanılan bir diğer yaklaşım ise her bir değerin Z-dönüşüm karşılıklarının hesaplanmasından sonra ortalamalarının alınmasıdır. Bu yaklaşım da önceki ile aynı ölçümlere uygulanmıştır, ancak yapılan teorik hesapların hesaplar pratik sonuçlar ile daha uyumsuz olduğu gözlenmiştir.

3.2. Tüm Bitler Çapraz İlinti Analizi

Bu çalışmada, daha önce geliştirilmiş çapraz ilinti tabanlı çalışmanın aksine [17] bir veya birden fazla referans bittin tüm diğer bitlerle olan ilintisini kullanmak yerine, tüm bitlerin birbiri ile ilintisini kullanan güçlendirilmiş bir yöntem geliştirilmiştir. Bu güçlendirilmiş yöntem, anahtara ait bitlerin daha az sayıda eğri kullanılarak elde edilmesine olanak sağlamaktadır.

3.2.1. Yöntemin tanıtımı

Tüm bitler yönteminin temel çıkış noktası, önceki çapraz ilinti analizi çalışmalarında gözlenen şu durum olmuştur: Çapraz ilinti analizinde tip0 türündeki bir referans bit seçildiğinde, bu bitin kendi türündeki tüm diğer bitlerle olan ilintileri toplamı ortalama ilinti değerlerinden yüksek, tip1 türündeki bitlerle olan ilintiler toplamı ise düşük

olmaktadır. Ancak referans bit, tip1 türündeki bitlerden seçildiğinde ise bunun tam tersi bir durum gözlenmektedir. Yani tip1 türündeki referans bit alanının, aynı türdeki diğer alanlarla ilintisi, ortalama ilinti değerinden küçük olurken tip0 türündeki referans bitlere ait alanlarla olan ilintisi yüksek olmaktadır. Tek ve birden fazla tip0 referans bit ile çalışıldığında, referansla yüksek ilinti içeren alana sahip olan bitin referansla aynı tip olduğuna karar verilirken, tip1 türündeki tek ya da çoklu referans bit ile çalışıldığında karar mekanizması tam ters yönde çalışmaktadır. İşte bu özellik, tüm bitler çapraz ilinti analizi adı verilen daha güçlü bir yöntemin geliştirilmesinde kullanılmıştır. Bu yeni yöntemde, bir veya daha fazla referans bite ait değerler yerine, aslında tüm bitlere ait eğri alanlarının birbirleri ile olan ilinti değerleri kullanılmaktadır. Eğer tipinin tespit edilmesi hedeflenen bit tip0 türünde ise, bu bite ait güç eğri alanının tüm diğer bitlerinki ile olan çapraz ilintisi, ortalama ilintiden yüksek olmaktadır. tip1 türündeki bir bit için ise, durum bunun tam tersidir ve bu bite ait eğri alanının tüm diğerlerininki ile olan ilintiler toplamı, ortalama ilintiden düşük olmaktadır. Sonuç olarak, tüm bitlerle ilintiler toplamı, ilintiler ortalamasından yüksek alanlara sahip bitler tip0, düşük olanlar ise tip1 olarak sınıflandırılmaktadır. Bu şekilde tek referans bitten gelen bilgi yerine tüm bitlerin ilintisinden gelen bilginin kullanılması ile eğri parçaları daha etkin bir şekilde kullanılmakta ve önceki yöntemden %75 daha az eğri ile anahtar bitleri elde edilmektedir.

Bu yöntemi uygulayabilmek için önceki bölümlerde anlatıldığı gibi her bir bit tipinin işlem gördüğü güç eğri alanlarının bölütlenmesi gerekmektedir. Her bir RSA koşturumuna ait Pi güç eğrisinin bu şekilde bölütlenmesi ile elde edilen alt parçalar Pi_1, Pi_2, …$\mathrm{Pi}_{\mathrm{w}-1}$ olsun. Öncelikle her bite ait eğri parçasının tüm diğerleri ile olan çapraz ilinti değerlerinin toplamından oluşan çapraz ilinti katsayıları $\mathrm{Ci} = \{\mathrm{Ci}_1, \mathrm{Ci}_2, \ldots., \mathrm{Ci}_{\mathrm{w}-1}\}$ değerleri hesaplanır ve her bir bite ait bu katsayılar toplanarak toplamsal çapraz ilinti değerleri hesaplanır. Bu işlemin akışı Algoritma 3.5. de verilmiştir. Algoritma'da verildiği gibi, her bir j bitinin tipine karar vermek için, her bir i nolu RSA koşturumundan elde edilen ve j numaralı bite ait tüm ilinti değerlerinin (Ci_j) toplamından oluşan $\mathrm{Ci}_{\mathrm{top}}$ değerleri kullanılmaktadır. Her bir $\mathrm{Ci}_{\mathrm{top}}$ değeri, tüm ilintilerin kayan ortalamasından oluşan eşik değeri ile karşılaştırılarak, ortalamadan küçük ise j bitinin tip0 büyük ise tip1 türüne olduğuna karar verilmektedir.

Algoritma 3.5. Tüm bitler çapraz ilinti değerlerinin hesaplanması

GİRDİLER
P1, P2 ,..., PM, Pi ={Pi_1, Pi_2, ….Pi_{w-1} }

ÇIKTILAR
C1 , C2,..., CM , Ci = {Ci_1, Ci_2 …. , Ci_{w-1}}

```
For i= 1 to M
    For j=w-1 to 1 do
        Cij = 0
        For k=w-1 to 1 do
            Cijk=Corr( Pij, Pik)
            Cij=Cij+Cijk
Ci = {Ci1, Ci2 …. , Ci(w-1)}
Return Ci
```

Algoritma 3.6. Tüm bitler çapraz ilinti yöntemine göre anahtar tiplerinin elde edilmesi

GİRDİLER
C1 , C2, …. , CM, Ci={Ci_1, Ci_2, …. , Ci_{w-1}}
ÇIKTILAR
dt={ dt_{w-1}, …. , dt_r, … dt_1 } i

```
eşik = 0
Cij top = 0
For i= 1 to M
     For j=w-1 to 1 do
         Cijtop = Cijtop + Cij
         eşik=EşikGüncelle3(Cijtop)
         If  Cijtop> eşik
             dti = tip0
         Else
             dti = tip1
dt={dt(w-1) , dt(w-2), …. , dtr, … dt1 }
Return dt
```

Algoritma 3.6.'da eşik güncellemek amacıyla kullanılan ve "EşikGncelle3" olarak isimlendirilen fonksiyon basitçe, tüm bitlere ait olan çapraz ilinti değerlerinin yan yana 50 tanesinin ortalamasını almaktadır. Bu fonksiyona ait işlem akışı Algoritma 3.7.'de verilmiştir.

Aşağıdaki alt bölümlerde, teorik olarak tanıtılan tüm bitler çapraz ilinti analizinin ASIC ML tipi üs alma gerçeklemesi ve FPGA'de ikilik hep çarp üs alma gerçeklemesine uygulamaları anlatılmıştır.

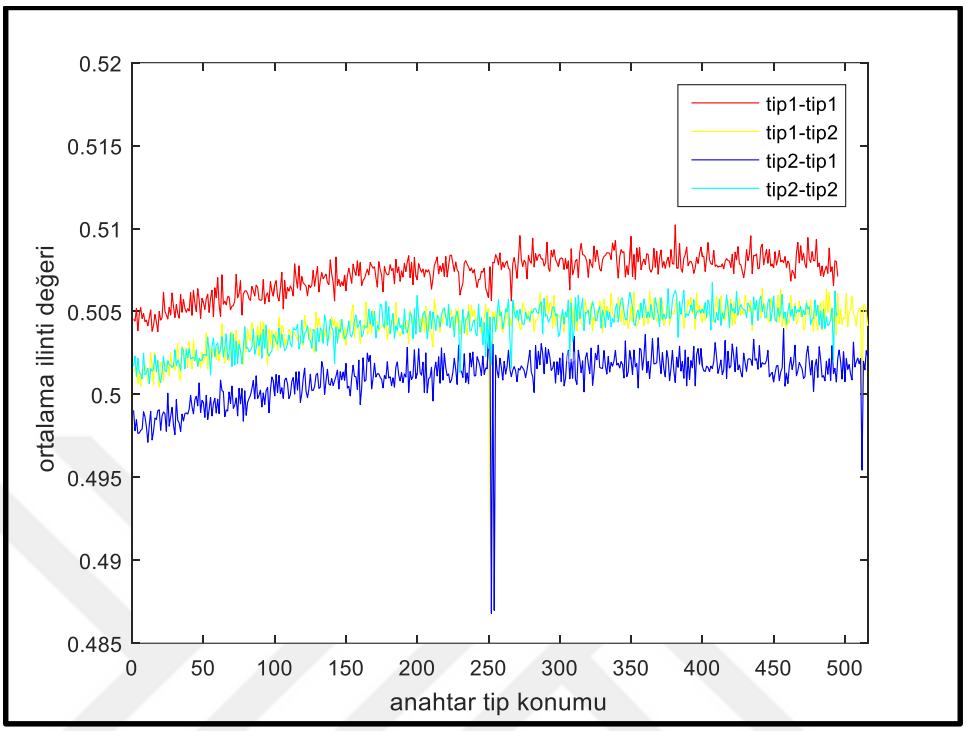
Algoritma 3.7. EşikGuncelle3 fonksiyonu

GİRDİLER C1 , C2, …. . , CM, Ci={Ci_1, Ci_2, …. , Ci_{w-1}} ÇIKTILAR eşik
eşik = 0 For j=w-1 to 1 do $C_{top\ j} = 0$ k1= j/50 For i= 1 to M $C_{top\ j} = C_{top\ j} + Ci_j$ eşik(j)= mean ($C_{top\ k1.50+1}$, $C_{top\ k1.50+2}$ …. , $C_{top\ k1.50+50}$) Return eşik

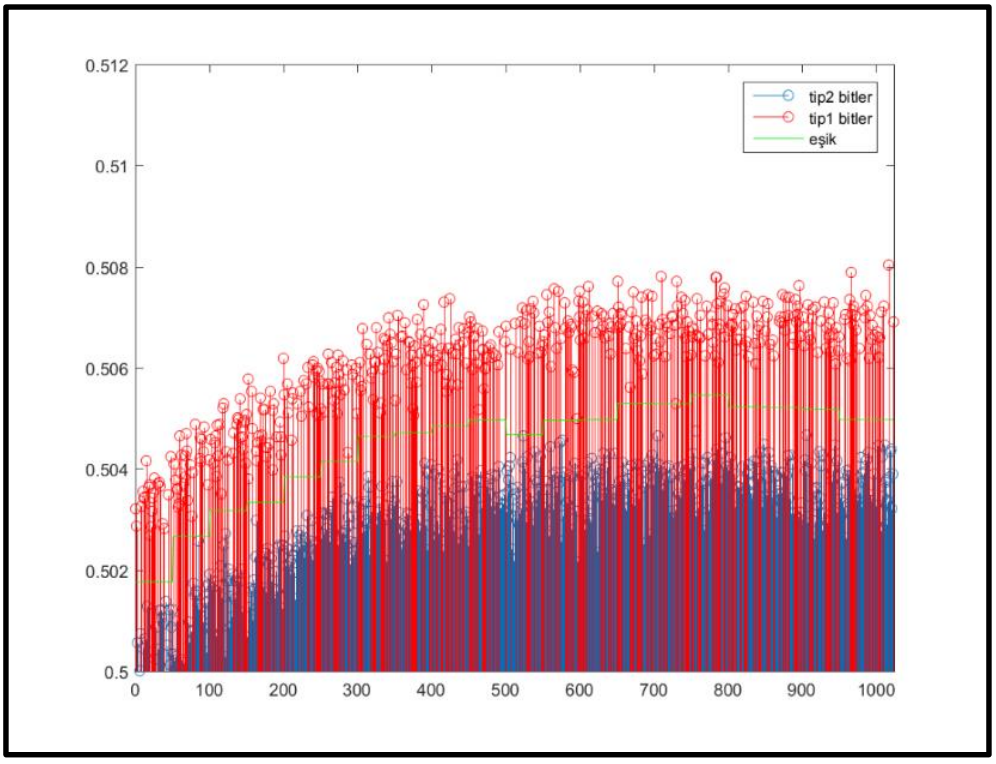
3.2.2. Yöntemin ASIC ML devresine uygulanması

Tüm bitler çapraz ilinti analizinin uygulandığı hedef devrelerden ilki ASIC ML üs alma devresidir. Bu gerçeklemede, yukarıdaki bölümlerde bahsedilen ve tüm bitler çapraz ilinti analizinin ortaya çıkmasına neden olan davranışın kaynağı, tip0 türündeki bir referans bitin kendi tipindeki bitlere ait eğri alanları ile daha yüksek ilinti değerine sahipken, tip0 türündeki bir referans bit için bu durumun tam tersinin geçerli olmasıdır. Yani tip0 bitlerin aksine tip1 türündeki referans bitler kendi türündeki diğer bitlerle düşük çapraz ilinti değerine sahip olup, karşıt tür olan tip0 bitler ile daha yüksek ilinti değerine sahiptir. Bu davranışı daha iyi gözlemek için Şekil 3.22.'nin incelenmesi gerekmektedir. Burada iki farklı tipte referans bitin kendi tipleri ve karşıt tiplerdeki tüm diğer bitlerle olan toplamsal çapraz ilinti değerleri görülmektedir. Burada açık mavi ile verilen eğri, tip0 türünde sabit bir referans bite ait eğri bölütünün tüm diğer tip0 bitlerinki ile olan, sarı ile verilen ise diğer tüm tip1 bitlerinki ile hesaplanan toplamsal çapraz ilinti değerlerini göstermektedir. Yine aynı şekilde görülen kırmızı eğri ise tip1 türünde seçilen başka bir sabit referans bit alanının tüm diğer tip1 bitlerinki ile sarı eğri ise tüm diğer tip0 bitler için hesaplanan toplamsal çapraz ilinti değerleridir. Bu eğrilerde, tek bir referans bit için gözlenen durum aslında tüm bitler için geçerlidir. Sonuç olarak, tip0 türü bitlerin kendi türü ile ortalamadan daha yüksek, tip1 türü bitlerin ise kendi türü ile ortalamadan daha düşük ilinti değerlerine sahiptir. Bununla birlikte her iki bit türü, karşıt türler ile de yaklaşık olarak ortalamaya yakın bir değerde çapraz ilinti değerlerine sahiptir. Yani tip0 ve tip1 bitler tüm diğer bitler ile gösterdikleri ilinti davranışı ile birbirlerinden ayrışmaktadır. Tüm anahtar bitlerine

ait eğri alanlarının çapraz ilinti değerleri ve bu değerler kullanılarak hesaplanan eşik değeri Şekil 3.23.'de görülmektedir.

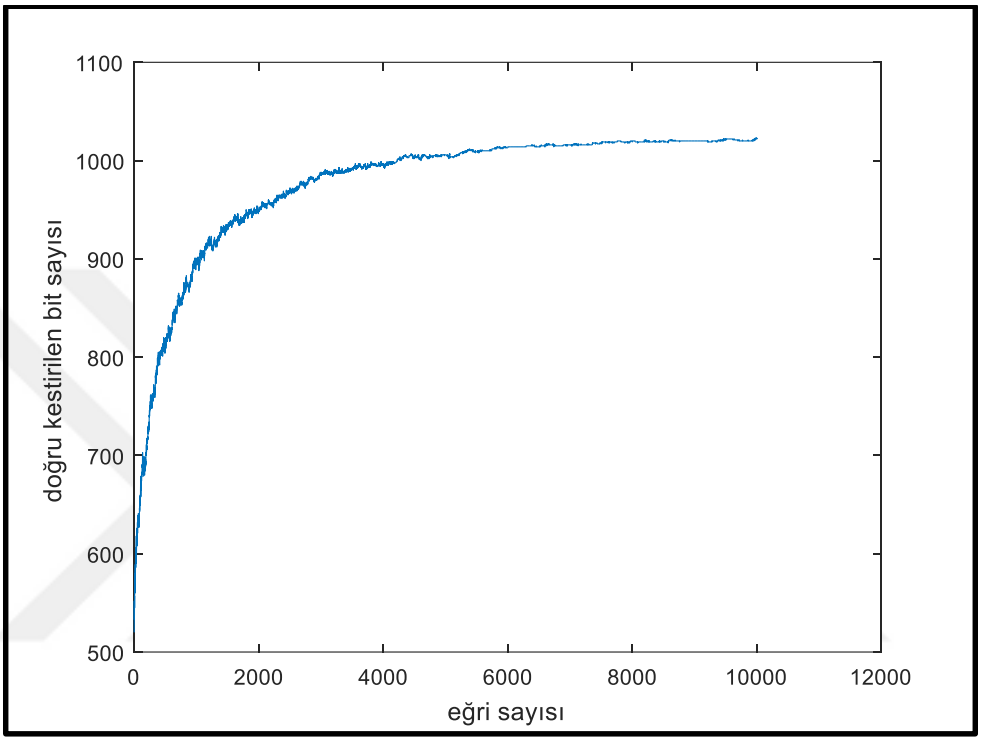


Şekil 3.22. Farklı referans bölgelerin diğerleri ile çapraz ilinti değeri



Şekil 3.23. Tüm bit değerleri için hesaplanan ortamla çapraz ilinti değerleri

Tüm bitler çapraz ilinti yönteminde, değişen eğri sayısı ile değeri doğru olarak kestirilebilen bit sayısının değişimi Şekil 3.24.'de verilmiştir. Bu şekilden de görülebileceği gibi, sabit referans bit kullanan önceki çapraz ilinti yöntemine [16] göre %75 oranında daha az eğri kullanılarak aynı başarıma ulaşılmakta, yani anahtar bitlerinin türü daha az eğri ile elde edilebilmektedir.

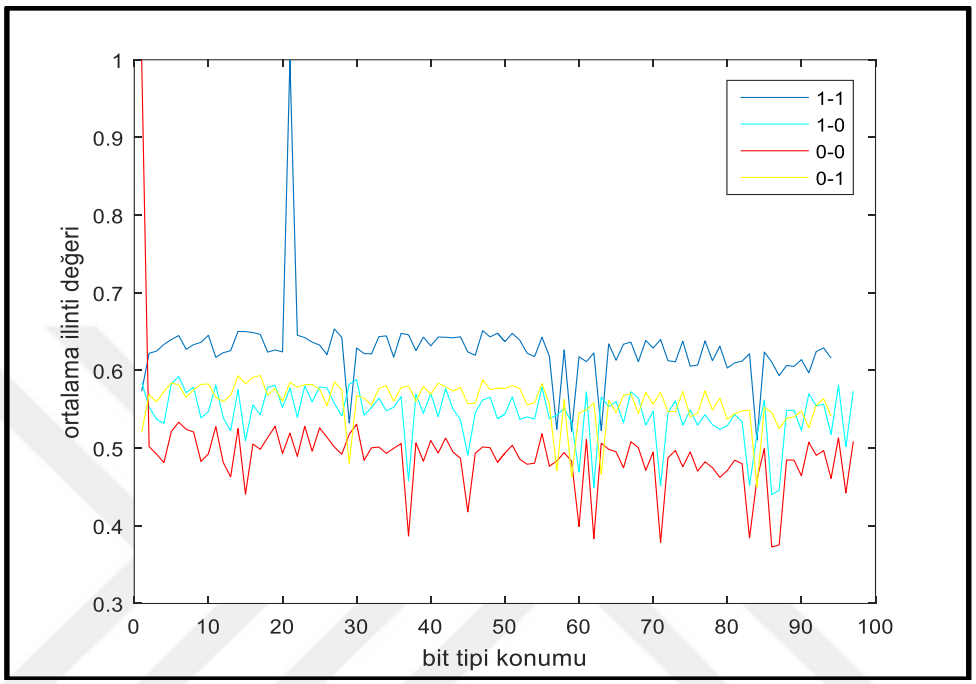


Şekil 3.24. Artan eğri sayısıyla değeri doğru olarak kestirilebilen bit sayısı

3.2.3. Yöntemin FPGA ikilik üs alma devresine uygulanması

Tüm bitler çapraz ilinti analizinin ortaya çıkmasına neden olan çapraz ilinti davranışının, FPGA tabanlı devrede nasıl gözlendiği Şekil 3.16.'da verilmişti. Tıpkı ASIC ML gerçeklemesinde olduğu gibi bu devrede de farklı tipteki referans bitlerin kendi tipi ve karşıt tipteki bit alanlarına ait çapraz ilinti değerleri farklı davranışlar sergilemektedir. Bu devrede gözlenen durum, değeri 1 olan bitlerin ASIC devredeki tip0 bitler gibi davranış göstermiş olmasıdır. Yani değeri 1 olan bitler kendi türleri ile yüksek ilintiye sahip olurken, değeri 0 olan bitlerde durum bunun tam tersidir. Bu özelliğin varlığından dolayı tüm bitler çapraz ilinti analizi bu devre üzerinde de uygulanabilecek bir saldırı türü olmaktadır. Bu davranışı daha iyi anlamak için Şekil 3.25. incelenmelidir. Burada iki farklı tipte referans bitin kendi tipleri ve karşıt tiplerdeki tüm diğer bitlerle olan ortalama (toplamsal) çapraz ilinti değerleri

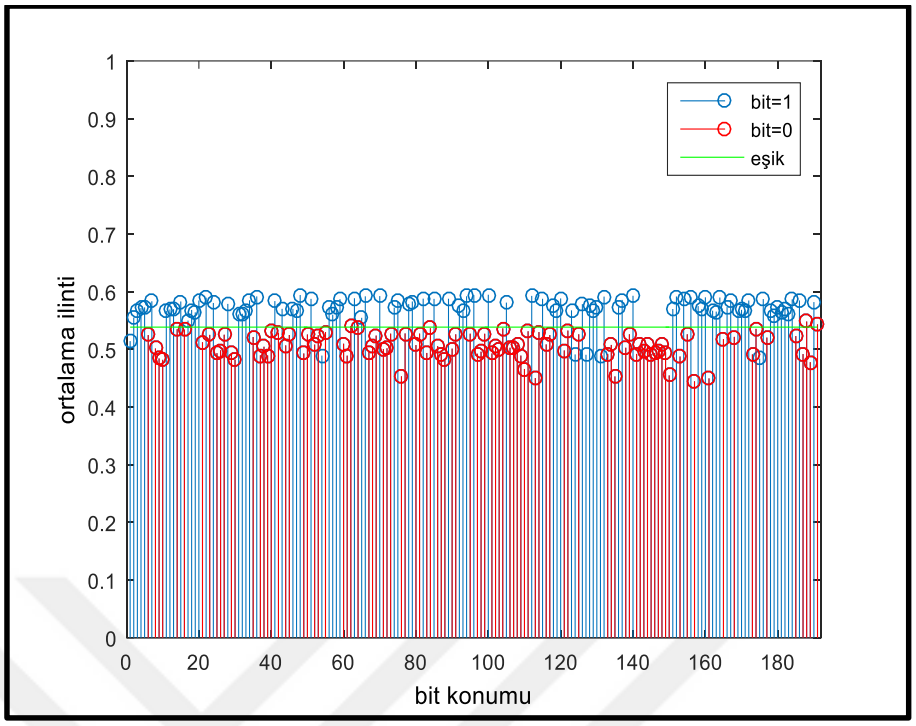
verilmiştir. Şekilden görüleceği üzere değeri 0 olan bitlerin yine 0 değerli olan bitlerle ilintisi ortalamadan düşük iken değeri 1 olanlarda tersi durum söz konusudur. Her iki bit türünde de karşıt bit türü ile olan ilinti değerleri ise yaklaşık olarak ortalama ilinti değerine yakındır.



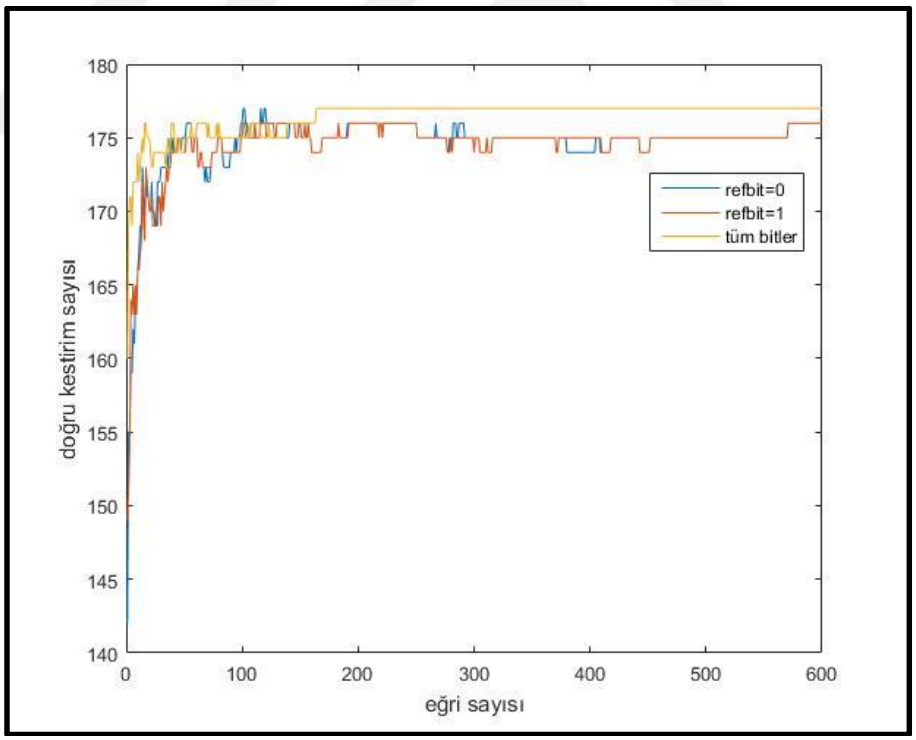
Şekil 3.25. Farklı referans bit alanlarının diğerleri ile çapraz ilinti değerleri

Tüm anahtar bitlerine ait eğri alanlarının çapraz ilinti değerlerinin toplamından oluşan eğri Şekil 3.26.'da görülmektedir. Bu şekilde kırmızı ile boyalı olan ilinti değerleri, değeri '0' olan, mavi renkliler ise değeri 1 olan anahtar bitlerine aittir. Sonuç olarak tüm bitler çapraz ilinti analizi ile 200 eğri kullanılarak bile iki farklı türdeki anahtar bitlerinin büyük oranda birbirinden ayırt edilebileceği görülmektedir.

Artan eğri sayısı ile değeri doğru olarak kestirilebilen bit sayıları ise Şekil 3.27.'de verilmiştir. Bu şekilden görüldüğü üzere tek referans bit yönteminde, değeri 0 olan (mavi çizgi) ve değeri 1 olan (kırmızı çizgi) referans değerleri kullanıldığında benzer başarımlar elde edilmektedir. Tüm bitler yöntemi (sarı çizgi) kullanıldığında ise doğru yanıt sayısı daha hızlı şekilde artmakta ve toplamda daha fazla bitin değeri doğru olarak kestirilebilmektedir.



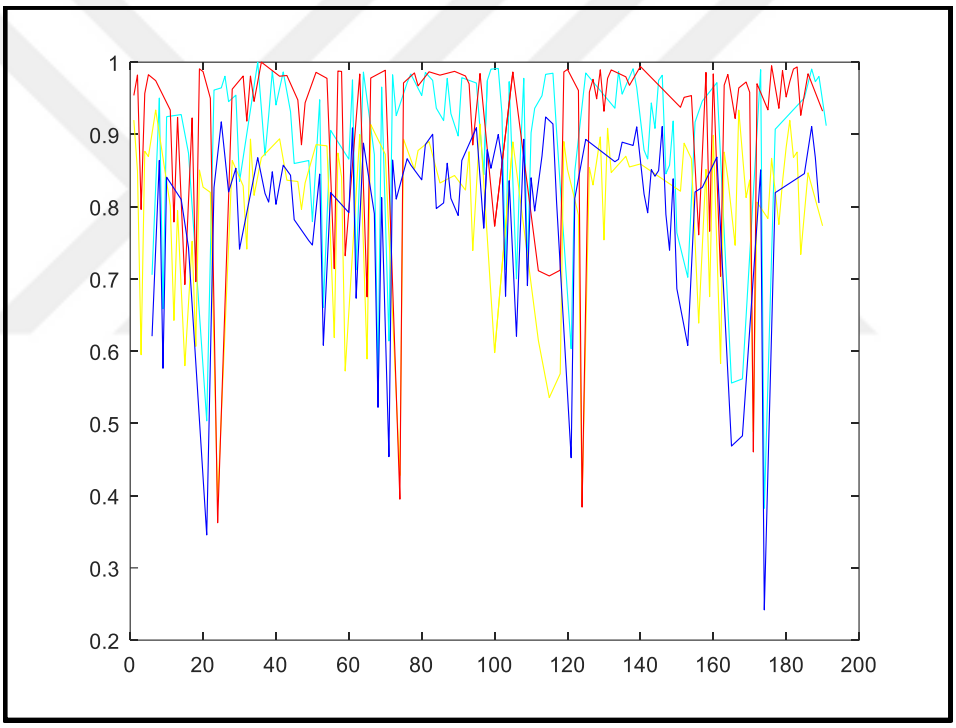
Şekil 3.26. Tüm bitlerin toplamsal çapraz ilinti değerleri



Şekil 3.27. Artan eğri sayısıyla değeri doğru olarak kestirilebilen bit sayısı

3.2.4. Yöntemin FPGA benzetim eğrilerine uygulanması

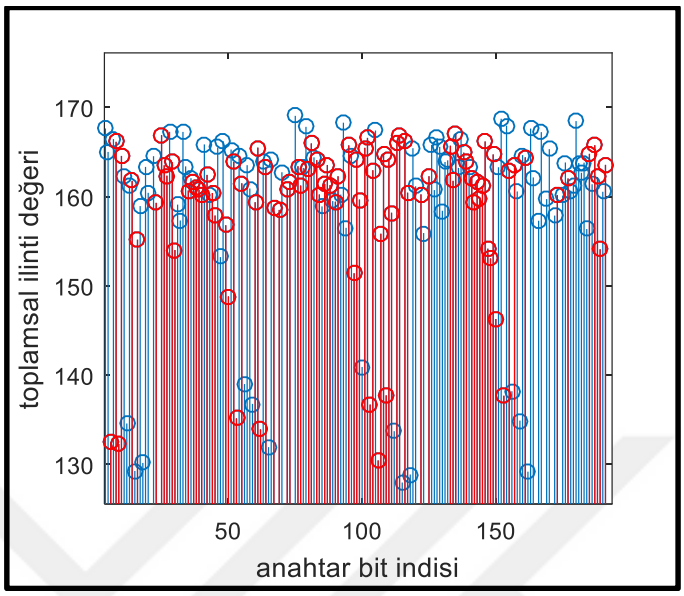
Önceki bölümde, 192 bitlik davranışsal seviye benzetim eğrileri için yapılan çapraz ilinti analizinde, gerçek güç eğrilerinin aksine, her referans türünün kendi türleri ile daha yüksek ilintiye sahip güç tüketim eğrileri içerdiği görülmüştü (Şekil 3.20). Bu duruma daha iyi gözlemek için Şekil 3.28'in incelenmesi gerekmektedir. Burada iki farklı tipte referans bitin kendi tipleri ve karşıt tiplerdeki tüm diğer bitlerle olan toplamsal çapraz ilinti değerleri görülmektedir. Şekilde açık mavi ile verilen eğri, tip0 türünde sabit bir referans bite ait eğri bölütünün tüm diğer tip0 bitlerinki ile sarı ile verilen ise diğer tüm tip1 bitlerinki ile hesaplanan toplamsal çapraz ilinti değerlerini göstermektedir.



Şekil 3.28. Davranışsal seviyede farklı referans bitlerin çapraz ilinti değerleri

Yine aynı şekilde görülen kırmızı eğri ise tip1 türünde seçilen başka bir sabit referans bit eğri alanının tüm diğer tip1 bitlerinki ile sarı eğri ise tüm diğer tip0 bitler için hesaplanan toplamsal çapraz ilinti değerleridir. Gerçek güç eğrilerindeki durum burada tam olarak geçerli olmasa da, tip1 türündeki anahtar bitlerinin kendi türleri ile olan ilinti ortalamasının, tip0'ın kendi türü ile olandan biraz daha yüksek olması nedeni ile

tüm bitler çapraz ilinti analizinin aslında benzetim eğrilerine de uygulanması mümkündür. Ancak tek bir eğri için başarım çok iyi değildir.



Şekil 3.29 Davranışsal benzetim eğrisinde tüm bitlerin çapraz ilinti değerleri

Tüm anahtar bitlerine ait eğri alanlarının çapraz ilinti değerlerinin toplamından oluşan eğri, Şekil 3.29.'da görülmektedir. Şekilde kırmızı ile boyalı olanlar, değeri 0 olan, mavi renkliler ise değeri 1 olan anahtar bitlerine ait ilinti değerleridir. Burada tip1 olarak isimlendirilenlerin biraz daha yüksek ilinti seviyelerine sahip olduğu görülmektedir. Sonuç olarak tüm bitler çapraz ilinti analizi açısından, benzetim eğrisinin, gerçek güç eğrilerine yakın bir davranış gösterdiği söylenebilir. Farklı giriş değerine sahip birden çok benzetim eğrisi kullanılarak daha iyi başarımların elde edilmesi mümkün olabilir.

3.2.5. Yönteme ilişkin deneysel bulgular ve tartışma

Bu çalışmada geliştirilen güçlendirilmiş çapraz ilinti tabanlı özgün yöntem, hem ML tipi üs alma algoritması kullanan ASIC RSA devresine, hem de ikilik üs alma algoritması kullanan FPGA tabanlı RSA devresine uygulanmıştır. Her iki devrede de anahtar bitleri için yapılan tip sınıflandırması, çapraz ilinti analizine karşı benzer davranışlar göstermiş olduğundan, yani tip0 bit grupları kendi tipleri ile yüksek ilintiye sahip olurken tip1 olarak adlandırılan bitlerde bu durumun tam tersinin gözlenmesi nedeni ile tüm bitler yöntemi başarılı bir şekilde çalışmıştır. Bu çalışmaya temel

oluşturan ve referans bit kullanımına dayanan çapraz ilinti tabanlı yöntemin [16] aksine tüm bitler yönteminde, anahtar bitlerinin tümüne ait güç tüketim alanları kullanılabilmektedir. Bu durum her bir eğride kullanılan alan sayısını artırarak daha az eğri ile sonuca ulaşılmasını sağlamaktadır. Bununla birlikte kullanılan her bir eğri için daha fazla işlem gücü harcanması gerekmektedir. Yöntemin ASIC devre uygulamasında, [16] yönteminden yaklaşık olarak %75 oranında daha az eğri ile sonuca varılırken, FPGA'li devrede bu oran çok yüksek olmayıp % 10'lar civarındadır. FPGA devreden alınan güç ölçümlerinin 3.1.5 bölümünde anlatılan nedenlerden ötürü, ölçüm düzeneğinden açısından daha avantajlı olması, aradaki farkın daha az olmasının nedenleri arasında gösterilebilir. Bu yöntemde, [16] yöntemine gerçekleştirilen uygulamadan daha fazla bit değeri elde edilebilse de yine de anahtar bitlerinin tamamı elde edilememiştir. Bu durum da yine 3.1.5 bölümünde açıklandığı gibi, FPGA devresinin belli bölümlerinde oluşan sentezleme ve yerleştirme-bağlama aşamalarında oluşan farklılıklardan, kullanılan anahtar ve rastgele veri değerlerinden kaynaklanabilir.

3.3. Frekans Uzayı Çapraz İlinti Analizi

3.3.1. Yöntemin tanıtımı

Frekans uzayı çapraz ilinti analizi (FÇİA), aslında zaman uzayında uygulanan çapraz ilinti yöntemiyle aynı mantığa dayanmaktadır. Ancak burada çapraz ilintisi hesaplanan değerler eğri bölütlerine ait zaman örnekleri değil de, bu zaman dilimleri için hesaplanmış DFT (Discrete Fourier Transform) katsayılarının tamamı ya da bir kısmıdır. DFT dönüşümü, bir sinyalin içerisindeki sinüs ve kosinüs frekans bileşenlerine ayrıştırılarak ifade edilmesi anlamına gelmektedir. Her bir DFT katsayısı, ilgili frekans değerinin sinyal oluşumunda ne kadar katkısının olduğunu gösterir. Verilen bir frekanstaki DFT katsayısının genliği, aslında o frekansla ilgili bilginin çok önemli bir kısmını içerdiğinden frekans uzayı ÇİA analizinin de tıpkı FGA ve İGA yöntemlerine olduğu gibi aynı şekilde çalışması beklenir. Sonuç olarak frekans uzayı ÇİA yöntemi, her bir anahtar bitine ait güç tüketim alanlarının frekans bileşenleri anlamında birbirleri ile ne kadar örtüştüklerini dikkate almaktadır. DFT katsayılarını hesaplamak amacıyla FFT algoritması kullanılmıştır.

Gerçekleştirilen özgün FÇİA yönteminin uygulamalarında, zaman örnekleri ile aynı sayıda DFT katsayısı kullanıldığı durumda FÇİA saldırısı başarımının, zaman uzayına göre daha iyi olduğu, yani daha az eğri ile sonuca ulaşıldığı tespit edilmiştir. Bunun temel nedenleri şöyle sıralanabilir: Frekans uzayında çalışmanın avantajlarından biri, yanaştırma hatalarının bu uzayda daha az önemli olmasıdır. Zaman uzayındaki ÇİA çalışmalarında [16 - 17] işlemin gerçekleştiği her bir alan güç eğrilerinden bölütlenirken düzgün yanaştırılmaya çalışılmıştır. Yanaştırma yapabilmek için çeşitli filtreleme yöntemlerinden de faydalanılmıştır, ancak ideal bir yanaştırmanın yapılamadığı ve tetikleme sorunlarından kaynaklanan hata durumlarında ÇİA analizini uygulamak oldukça zorlaşmaktadır

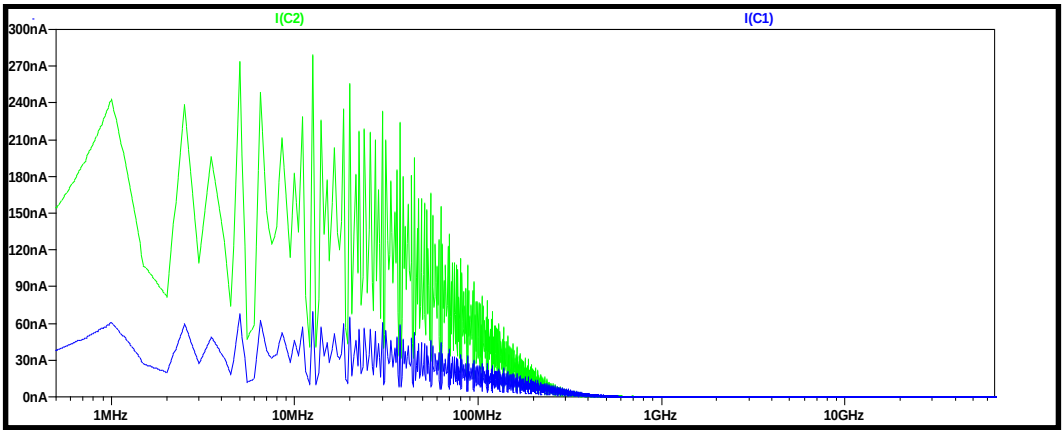
FÇİA yöntemi, tüm DFT katsayıları yerine, bu örnekler üzerinden hesaplanan katsayıların düşük frekanslara karşılık düşen belli belli bir kısmının kullanılması durumunda daha başarılı olduğu görülmüştür. Güç analizi saldırılarında zaman uzayı yerine, frekans uzayında çalışmanın en önemli avantajı, her bir frekans katsayısının birbirinden bağımsız olarak hesaplanabilmesi ve saldırı ile ilgili olmayan, ya da sinyal gürültü oranı (SNR) yeteri kadar iyi olmayan bileşenlerin elenebilmesidir. Bir hedef devrede, saldırıda kullanılan uygun frekans bileşenlerinin neler olduğunu anlamak için FGA-İGA ya da ÇİA türü saldırılarda kullanılan güç kaçağının nasıl modellendiğine bir göz atmak faydalı olacaktır. Önceki bölümlerde anlatıldığı gibi, CMOS tabanlı devrelerde veri bağımlı güç kaçağı, en basit hücre birimi olan eviricinin güç tüketim davranışı ile açıklanmaktadır. Lumped-C olarak modellenmiş bir evirici çıkışının anahtarlanması, yani durum değiştirmesi sırasında oluşan temel kaçak bileşenleri $C_{yük}$ kapasitesinin dolması ve anlık olarak oluşan doğrudan yol akımıdır. Dinamik durum güç tüketimine lojik seviyedeki en basit yaklaşım olan HW ya da HD modelleri, 0-1 ve 1-0 geçişlerinde eşit güç tüketildiğini varsaymakta ve aslında eviricilerin çıkışın durum değiştirmesine kadar olan zamandaki ara geçişlerle ilgilenmemektedir [21 , 39]. Bununla birlikte zaman ya da frekans uzayı ÇİA analizinde belli bir zaman aralığında olan değişimler önem kazanmaktadır. Bu nedenle lumped-C olarak modellenmiş bir eviricinin, durum değiştirmesi sırasında belli bir zaman aralığındaki davranışına daha yakından bakmak uygun olur. Yine önceki bölümlerde açıklandığı gibi lumped-C modelinde, bir hücrenin iç parazit kapasitelerinin hepsi ve bu hücre çıkışına bağlı olan tüm elektrik hattı ve diğer hücrelerden kaynaklanan kapasiteler,

"$C_{yük}$" olarak isimlendirilen tek bir kapasite ile temsil edilmektedir. Şekil 1.2'de lumped-C olarak modellenmiş bir evirici devreye ait analog seviye SPICE programıyla gerçekleştirilen güç tüketim benzetiminden görüleceği gibi, her iki değişim anında da oluşan akım değerleri, yaklaşık olarak, farklı yükseklik ve taban genişliğine sahip birer üçgen darbe şeklindedir. Daha önce de açıklandığı gibi 0-1 geçişinde doğrudan yol akımının yanı sıra $C_{yük}$ kapasitesinin dolmasından kaynaklı olarak VDD'den daha fazla akım çekilmektedir. Ancak 1-0 geçişinde $C_{yük}$ kapasitesi akım çekmeyip toprağa yükünü boşalttığından VDD'den çekilen akım daha azdır. Bununla birlikte $C_{yük}$k kapasitesinin değeri değiştiğinde de çekilen akımlar değişmektedir. Çekilen bu akımın frekans uzayında değişimini anlamak için [39]' dekine benzer bir yaklaşımla, değişim anına ait bu üçgen darbenin Fourier dönüşüm ifadesi belli bir fikir verecektir.

$$P(f) = \int_0^{at_{dy}} \frac{I_{tepe}}{a\,t_{dy}}\, t\,.\,e^{-j2\pi ft}\; d_t + \int_{at_{dy}}^{t_{dy}} \frac{-\,I_{tepe}}{t_{dy}-a\,t_{dy}}\,(t-t_{dy}).\,e^{-j2\pi ft}\; d_t =$$

$$\frac{-j\,I_{tepe}}{2\pi^2 f^2\, t_{dy}}\left[\frac{1}{a}\sin(a\pi f t_{dy})e^{-j\pi f a\, t_{dy}} - \frac{1}{1-a}\sin((1-a)\pi f\, t_{dy})\; e^{-j\pi f(\,(a+1)\,t_{dy})}\right] \qquad (3.14)$$

Doğrudan yol akımının, t_{dy} süresi içerisinde gerçekleştiğini ve akımın "$a.t_{dy}$" kadar sürede "I_{tepe}" yani en yüksek akım değerine ulaştığını varsayalım. Bu durumda taban değerleri "at_{dy}" ve "$t_{dy} - at_{dy}$", yüksekliği "I_{tepe}" olan iki üçgen darbe fonksiyonun Fourier dönüşümü aşağıdaki eşitliklerle hesaplanabilir.



Şekil 3.30. Lumped-C model eviricinin güç tüketiminin Fourier dönüşümü

Buradaki t_{dy}, yani üçgen darbenin taban değeri, doğrudan yol akımının oluştuğu etkin zaman aralığıdır. Bu değer devrenin saat periyodu T'den bağımsızdır ve kullanılan

lojikte işlenen veri, lojiğin içyapısı, sıcaklık ve VDD değeri gibi pek çok etkene bağlıdır [39]. Çekilen akımın tepe değere ulaşma zamanı olan "a" değeri ise devredeki saatin yükselme zamanına ve saatin farklı devre elemanlarına erişimindeki zaman farkına (skewness) bağlıdır [72]

Şekil 3.30da, lumped-C olarak modellenmiş ve zaman uzayı güç tüketimi Şekil 1.2de verilen iki eviriciye ait güç tüketimi için benzetim ortamında hesaplatılmış Fourier katsayıları görülmektedir. Hem eşitlik (3.14) hem de bu gösterim incelendiğinde, aslında YKA'da kullanılacak akım kaçağının tüm frekans bandına yayılmış olduğu ve pratikte kaçağın üst kesim frekansının ölçüm düzeneğinin kesim frekansı olduğu söylenebilir. Bununla birlikte, kaçak akımın genliği, frekans "f'nin" karesi ile ters orantılı olarak azalmaktadır. Sonuç olarak frekans yükseldikçe sinyal gücünün azalması ve [21]' de belirtildiği gibi gürültünün daha çok yüksek frekans bileşenlerinde yer alması nedeni ile ölçülen güç tüketimindeki sinyal/gürültü oranı azalmaktadır. İşte tüm bu nedenlerden dolayı, FGA ve İGA türü saldırılarda olduğu gibi frekans uzayı ÇİA türü saldırılar için de, tüm frekans bandı yerine, en önemli hatta saat frekansının altındaki bileşenlerin seçilmesi YKA başarımını artırıcı yönde etki yapabilir.

FÇİA saldırısını uygulamak için öncelikle ÇİA yönteminde belirtildiği şekilde bölütlenmiş güç eğrilerine ait zaman örneklerinden DFT (FFT) katsayılarının elde edilmesi gerekmektedir. D tane RSA koşturumundan elde edilmiş Pi güç eğrisinde, anahtar bitlerine ait işlem alanlarını içeren eğri bölütleri Pi_1,Pi_2, ...Pi_{w-1} olsun. Her birsi N tane zaman örneğinden oluşan bu $Pi_J = \{x_0\ ,\ x_1, x_2, \ldots., x_{N-1}\}$ vektörlerinde her bir örnek değer birbirinden T_s kadar uzaklıkta, yani $f_s = \frac{1}{T_s}$ örnekleme frekansına sahiptir. Bu durumda elde edilecek birim frekans çözünürlüğü " f_s/N" olup bu değerin katlarına ait $f_k = k * f_s/N$ frekans bileşenlerindeki DFT katsayıları aşağıdaki gibi (FFT algoritması kullanılarak) hesaplanabilir.

$$FFT(Pi_j(x_i)) = \sum_{n=0}^{N-1} x_k \,.\, e^{-i2\pi nk/N} \qquad (3.15)$$

Bu dönüşümden sonra N tane zaman örneğine karşılık düşen simetrik DFT katsayılarının sadece yarısının mutlak değerini içeren N/2 boyutlu $Fi_j(f_n)$ vektörleri oluşturulur.

Sonraki adım ise DFT katsayılarını içeren $\mathrm{Fi_j(f_n)}$ vektörlerinin, tüm bileşenleri ya da belli bir alt grubunun, çapraz ilinti değerlerinin hesaplanmasında kullanılmasıdır. Zaman uzayı ÇİA saldırısında olduğu gibi, tip0 türünde seçilmiş bir r referans bit için, bu bite ait DFT vektörünün diğer bitlere ait çapraz ilintisini içeren $\mathrm{FCi_{rj}} = \{\mathrm{FCi_{r1}}, \mathrm{FCi_{r2}}, \ldots., \mathrm{FCi_{rw-1}}\}$ vektörleri aşağıdaki gibi hesaplanır [24]. Burada $\mu\mathrm{i_r}$, $\mu i_j'$ ve $\sigma\mathrm{i_r}^2$, $\sigma i_j^{\,2}$ sırasıyla i nolu güç eğrisinde referans bit "r" ve hedef bit "j'ye" ait tek taraflı DFT katsayılarına ait genlik değerlerinin ortalama ve varyans değerleridir.

$$\mathrm{FCi_{jr}} = \mathrm{CorrCoeff}\left(\mathrm{Fi_j}, \mathrm{Fi_r}\right) = \frac{\mathrm{Cov}(\mathrm{Fi_j}, \mathrm{Fi_r})}{\sqrt{\sigma\mathrm{i_r}^2 . \sigma i_j^{\,2}}} = \frac{\sum_{n=1}^{N1} \left(\mathrm{Fi_j}(\mathrm{f_n}') - \mu\mathrm{i_j}'\right).\left(\mathrm{Fi_r}(\mathrm{f_n}') - \mu\mathrm{i_r}'\right)}{\sqrt{\sum_{n=1}^{N1} \left(\mathrm{Fi_j}(\mathrm{f_n}') - \mu\mathrm{i_j}'\right)^2 . \left(\mathrm{Fi_r}(\mathrm{f_n}') - \mu i_r'\right)^2}} \quad (3.16)$$

Çapraz ilinti hesabında, DFT katsayılarının 0-fs/2 bandına denk gelen tüm bileşenleri ya da 0-Ni.fs/n frekans bandını kapsayan 0-Ni arası değerleri kullanılabilir.

Algoritma 3.8. Anahtar tiplerinin elde edilmesi

GİRDİLER: $\mathrm{FC1_r}$,$\mathrm{FC2_r}$, …. , $\mathrm{CM_r}$ $\mathrm{FCi_r}$={$\mathrm{FCi_{r1}}, \mathrm{FCi_{r2}}, \ldots., \mathrm{F\ Ci_{rw-1}}$} ÇIKTILAR dt={ $\mathrm{d_{w-1}}, \ldots., \mathrm{d_r}, \ldots \mathrm{d_1}$ } anahtar bit tipleri

eşik = 0 ;
$\mathrm{FCi_{jr\ top}} = 0$;
For i= 1 to M
  For j=w-1 to 1 do
  $\mathrm{FCi_{jr\ top}} = \mathrm{FCi_{jr\ top}} + \mathrm{FCi_{jr}}$
  eşik=EşikGüncelle4($\mathrm{FCi_{jr\ top}}$);
  If $\mathrm{FCi_{jr\ top}}$> eşik $\mathrm{d_j}$ = tip0
  Else $\mathrm{dt_j}$ = tip1
dt={$\mathrm{dt_{w-1}}, \mathrm{dt_{w-2}}, \ldots., \mathrm{dt_1}, \mathrm{d_0}$ }
Return

Sonraki aşama ise w-1 tane yani tüm anahtar bitleri için oluşturulan bu çapraz ilinti değerlerine bakılarak ilgili bitin tipine karar vermektir. Algoritma 3.8.'de her bir anahtar tipinin elde edilmesini anlatan işlem akışı verilmiştir. Bu akışta kullanılan ve "EşikGüncelle4" olarak adlandırılan fonksiyon ise basitçe, yan yana 50 şer bitin frekans uzayı ilinti değerlerinin ortalamasını alarak, ilgili 50 bitlik grupta bulunan bitler için kullanılacak eşik değerini oluşturmaktadır. Bu fonksiyon Algoritma 3.4.'de

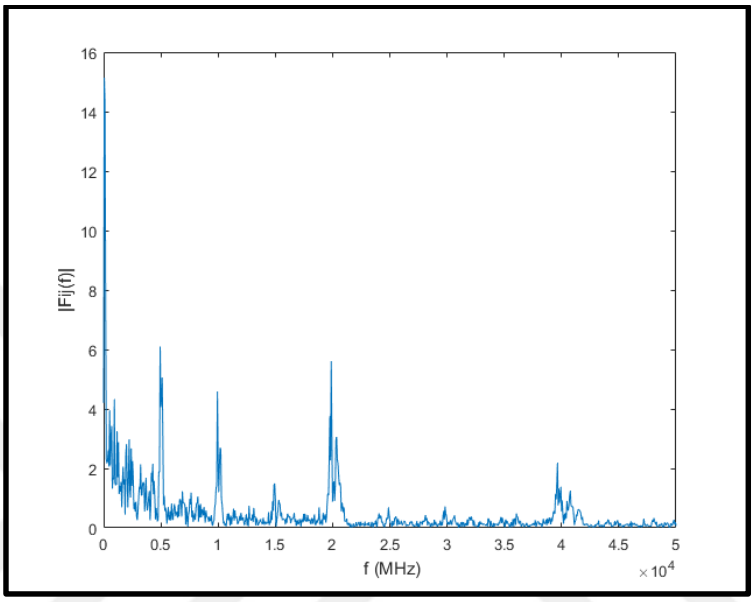
tanımlanan EşikGüncelle2 isimli fonksiyonla, zaman ilinti değerleri yerine frekans ilinti değerlerini kullanmak koşulu ile aynı şekilde çalışmaktadır.

3.3.2. Yöntemin ASIC ML devresine uygulanması

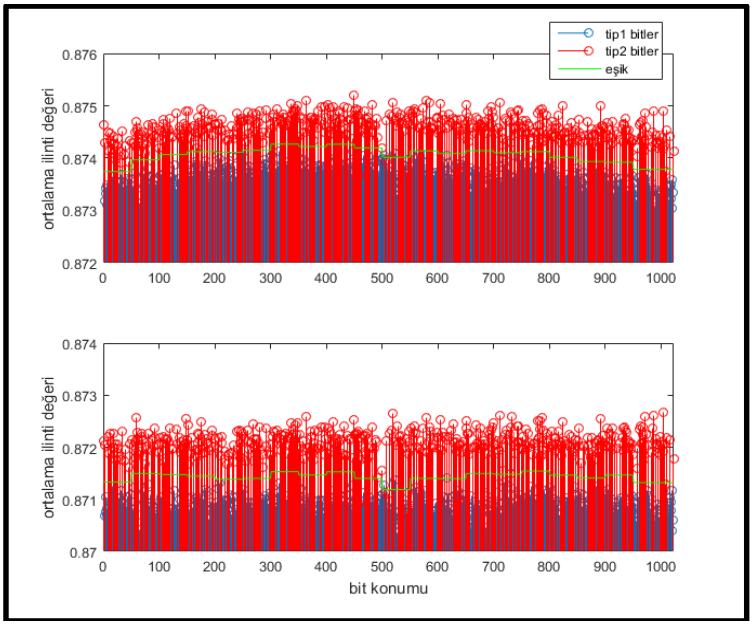
FÇİA saldırısını uygulamak için, zaman uzayı eşleniğinden farklı olarak gerçekleştirilen ilk adım, yukarıda ayrıntıları anlatılan DFT katsayılarının elde edilmesi aşamasıdır. Bu amaçla çapraz ilinti analizinde kullanılacak örnek bir ölçüm bölüntüne ait tek taraflı genlik spektrumu yani DFT katsayılarının mutlak değerleri Şekil 3.31.'de görülmektedir. Buradaki değerler, zaman uzayı ÇİA'da kullanılan zaman örneklerinin tamamının kullanılmasıyla elde edilmiş ve 0-fs/2 bandına ait frekans bileşenlerini içeren değerlerdir. Katsayıları elde etmek için FFT algoritması kullanılmıştır. Kullanılan örnekleme frekansı 100 MHz olduğu için DFT'si hesaplanabilecek en yüksek frekans bileşeni 0.5fs =50 MHz olacaktır. Buradan elde edilen DFT katsayılarının tamamı kullanıldığında, tip0 türünde belli bir referans bitin (500 numaralı bit) tüm diğerleri ile çapraz ilintisi Şekil 3.32. a)'da verilmiştir. Bu değerlerden 0-5 MHz arası bir frekans bandına denk gelecek şekilde ilk 115 tanesinin kullanılması ile elde edilen değerler ise Şekil 3.32. b)'de görülmektedir. Şekil 3.32. a)'da görüldüğü üzere, zaman uzayındaki duruma benzer şekilde, çapraz ilinti değerlerinin genliği, hedef bitin referans bite olan uzaklığına da bağlıdır. Referans bite yaklaşıldığı zaman çapraz ilinti değerlerinin ortalama genliği yükselmektedir. Bu durumun nedeni, birbiri ile daha yakın konumda olan bitlerin güç tüketimindeki gürültü bileşenlerinin birbirine daha bağımlı olmasıdır [21]. Şekil 3.32. b)'de görüldüğü gibi, çapraz ilinti hesabında düşük frekansı bileşenleri kullanıldığı zaman çapraz ilinti değerleri artık anahtar bitinin konumundan etkilenmemekte ve toplamsal ilinti değerlerinin ortalaması yaklaşık olarak sabit bir değerde seyretmektedir. Bu durumun nedeni ise yine [21]'de belirtildiği gibi, komşu işlem alanlarında birbiri ile ilintili değerler alan gürültü bileşenlerinin, daha çok yüksek frekans bantlarında bulunuyor olması ile açıklanabilir.

Şekil 3.33.'de tüm zaman örnekleri (yeşil), 0-50 MHz bandına denk gelen tüm tek taraflı DFT katsayıları ve sadece ilk 0-5 MHZ bandına denk gelen ilk 115 DFT katsayısı kullanılarak elde edilen çapraz ilinti analizi sonuçları verilmiştir. Şekilde, kullanılan toplam eğri sayısının değişimi ile ne kadar anahtar bitinin doğru olarak kestirilebildiği görülmektedir. Şekilden görülebildiği gibi en hızlı şekilde doğru

değerleri veren katsayılar az sayıda düşük frekans FFT bileşenlerinin kullanıldığı durum iken başarımı en kötü olan da zaman değerlerinin kullanılmış olduğu durumdur. Tüm FFT katsayıları yani ölçüm örnek sayısının yarısı kadar tek taraflı bileşenin kullanıldığı durum ise ortada bir yerde yer almaktadır. Sonuç olarak frekans uzayı FÇİA saldırısı, zaman uzayında çalışan eşleniğine göre daha hızlı çalışmaktadır.

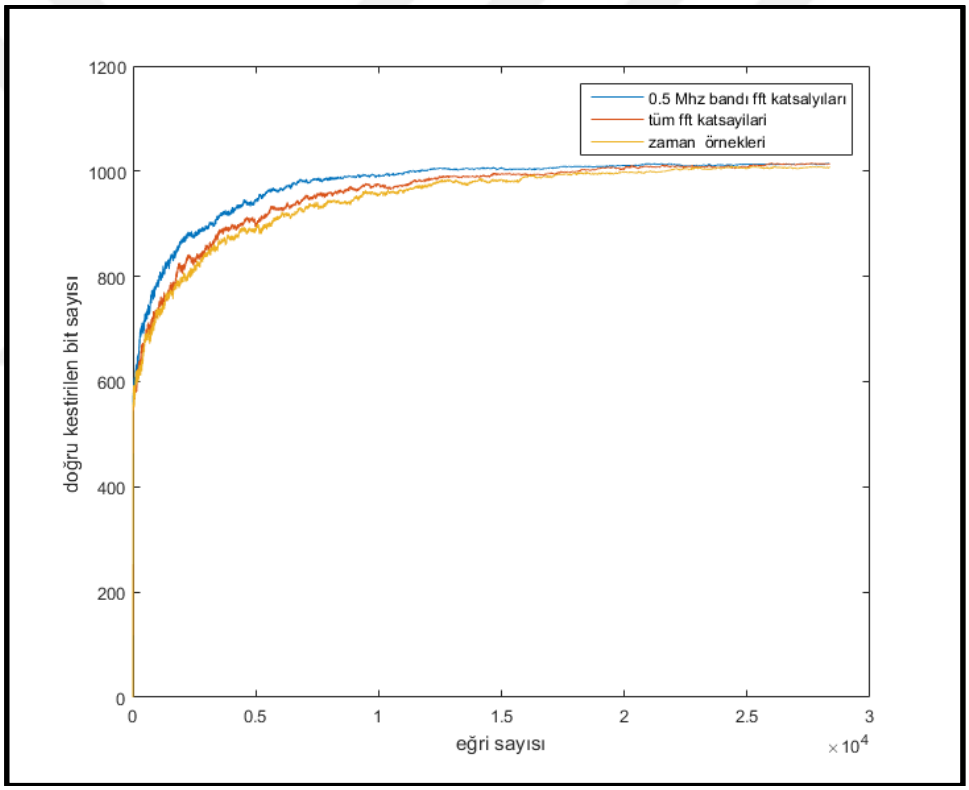


Şekil 3.31. Örnek bir ölçüm alanına ait tek taraflı genlik spektrumu



Şekil 3.32. DFT katsayılarından elde edilen çapraz ilinti değerleri

Tüm frekans bileşenleri kullanıldığı zaman başarımın hızının artması olası yanaştırma sorunlarının frekans uzayında daha az zarara yol açması ile açıklanabilir. Daha az katsayının kullanıldığı durum ise, yukarıda teorik olarak da açıklandığı gibi düşük frekanslarda sinyal/gürültü oranının daha iyi olması ile açıklanabilir. Bunun yanı sıra birbirine yakın olan bitlerdeki gürültü bileşenlerinin de birbiri ile ilintili olduğu görülmüştü. Bu durum eşik hesaplamasında da hatalara yol açmakta ve başarım hızını düşürmektedir. Önemli güç bileşenlerinin bulunduğu düşük frekans bantlarında çalışıldığı zaman, SNR iyileştiğinden, komşu referans bite yaklaşıldığı zaman ilinti değerlerinde görülen yükselme de epeyce düzelmektedir. Bu durum da düşük frekanslara ait bileşenlerle çalışıldığında başarımı artıran etkenlerden biridir.



Şekil 3.33. Artan eğri sayısı ile değeri doğru olarak kestirilen bit sayısı

3.4. Yenilikçi İlintisel Güç Analizi

3.4.1. Yöntemin tanıtımı

Tez kapsamında geliştirilen ve "yenilikçi ilintisel güç analizi" olarak adlandırılan yöntemde, İGA kullanılarak anahtardaki her bir bitin tipine göre yapılan farklı işlemlerin ayırt edilmesi yani bitlerin sınıflandırılması hedeflenmiştir. İGA tipi

yöntemlerde genel olarak anahtar parçası ile ilgili belli bir ara işlem adımına ait teorik güç tüketim modelinin, bu adıma ait gerçek güç eğrileri ile olan ilintisine göre anahtar parçasının doğru kestirilip kestirilmediğine karar verilmektedir. Teorik güç modellerinde ise hedef ara değerin HW (Hamming Weight) ya da HD (Hamming Distance) değişimi kullanılmaktadır. Bizim çalışmamızda ise herhangi bir işlem ara değerine odaklanmak yerine, hedef ML tabanlı üs alma adımlarında, belli bir anahtar bitinin kendinden sonra gelen ile aynı değeri taşıyıp taşımamasına göre işlem girdilerinin farklı belleklerden okunmasından kaynaklanan fark yakalanmaya çalışılmaktadır. Buradaki fark doğrudan anahtar bitlerinin tipinden kaynaklandığından, güç tüketim modeli olarak yan yana anahtar bitlerine ait tip kestirim vektörleri kullanılmaktadır. Dokümanın ileriki bölümlerinde, yana yana anahtar bitlerinin kestirimine ait her bir gruptan "pencere" değeri olarak bahsedilecek ve penceredeki bit sayısına da "pencere boyu =m" adı verilecektir. Bu şekilde bir kerede m bitlik pencereye ait anahtar bitlerinin tipi elde edilmesi hedeflenmektedir.

Yöntemi uygulamak için, farklı veri değerlerinin girdi olarak kullanıldığı her bir eğride, yan yana anahtar bitlerinin işlendiği anlara ait güç tüketim eğrisine ait bölütler alt alta getirilerek güç tüketim matrisleri oluşturulmaktadır. Teorik güç tüketimini temsil eden güç tüketim vektörleri ise, belli bir pencere boyu için olası tüm bit dizilimlerini içeren vektörlerin alt alta tekrarlanması ile oluşturulmaktadır. Bu şekilde hem aynı eğrideki farklı işlem anları hem de farklı eğrilerde aynı işlem anlarına denk gelen eğriler bir arada kullanılmış olmaktadır. Her iki tipte ve doğru kestirim değerlerine sahip anahtar tip vektörlerinin, güç eğrileri ile yüksek ilintiye sahip olması beklenir. Bunun yanı sıra, sadece tek tip bit içeren güç matrislerinde bu durumun gözlenmeyip (ilinti değerlerini hesaplamak için birden fazla tip içeren vektörler gerektiğinden), tüm kestirim değerleri ile güç eğrilerinin düz bir ilinti eğrisine sahip olması beklenir. Saldırılarda pencere boyu m=4 olarak seçilmiştir. Bir penceredeki her bir bitin tipi kestirilerek dörtlü "tip kestirim değerleri" olan "k_i" vektörleri k_0 = [0 … 0], k_1=[0…0 1], k_2=[0…1 1], $k_{2^{m-1}}$ =[0 1… 1] değerlerini alır. Her biri m boyutunda olan bu "k_i" vektörlerinin art arda ölçüm sayısı olan M kadar tekrarlanması ile "$K_i = [k_i\ k_i\ k_i..k_i]$" teorik sınıf kestirim vektörleri oluşturulur. Bu vektörlerin tamamı $K=\{K_0, K_1, \dots, K_{2^{m-1}}\}$ olup $K_0=[k_0\ k_0,.,k_0]$, $K_0=[k_0\ k_0,.,k_0]$, …, $K_{2^{m-1}}=[k_{2^{m-1}}\ k_{2^{m-1}}..k_{2^{m-1}}]$ değerlerini içerir. Daha sonra belli bir RSA

koşturumunda her bir anahtar bitinin işlem gördüğü güç tüketim alanlarının bölütlenmesi gerekmektedir. M tane RSA koşturumuna ait her bir eğride, hedeflenen bitlere karşılık düşen bölütlerin aynı sırada alt alta konması ile $P_i = [P1_{i1}\ P1_{i2}\ P1_{im}\ .P2_{i1}\ P2_{i2}\ P2_{i3\ ..}\ P2_{im}, \dots, PM_{i1}\ PM_{i2}\ PM_{i3\ ..}\ PM_{im}]$ güç eğrisi matrisleri oluşturulur.

Algoritma 3.9. İlinti değerlerinin hesaplanması

PARAMETRELER m= pencere genişliği w= bit sayısı GİRDİLER $P = \{P_1, P_2, .., P_{w/m}\}$ $K = \{K_0, K_1, \dots, K_{2^{m-1}}\}, K_0=[k_0\ k_0, .., k_0], .., K_{2^{m-1}}=[k_{2^{m-1}}\ k_{2^{m-1}} .. k_{2^{m-1}}]$ ÇIKTILAR C_{xk}, $0<k<2^{m-1}$, $1<x<w/m$

For x= 1 to w/ m
  For k= 1 to 2^{m-1}
  $C_{xk} = abs(corr(P_x, K_k))$
Return C_{xk}

Burada tüm m bitlik pencereler için, anahtardaki toplam w tane bit için $P = \{P_1, P_2, .., P_{w/m}\}$ olmak üzere, "w/m" tane matris oluşturulur. Daha sonra her bir "i" penceresi için, güç matrisi P_i'nin her bir sınıf kestirim vektörü K_k ile ilintisi hesaplanarak sınıf farklılığına neden olacak alanda yüksek ilinti değerinin elde edilip edilemediği gözlenir.

Algoritma 3.10. Pencere bit tiplerinin elde edilmesi

GİRDİLER C_{xk}, k_i $k_0 = [0 \dots 0\ 0]$, $k_1=[0 \dots 0\ 1]$, $k_2=[0 \dots 1\ 1]$, $k_{2^{m-1}}=[0\ 1 \dots 1]$, Length (k_i)=m ÇIKTILAR dt={ $dt_{w-1,} \dots dt_0$ }

For x= 0 to w-1 /m
  For k= 2 to 2^{m-1}
  If (C_{xk} <mean(C_{xk}))
  $dt_{mx}\ dt_{mx+1\ ..}\ dt_{mx+m-1} = k_0$
  Else if (max(C_{xk}))== C_{xl}
  $dt_{mx}\ dt_{mx+1\ ..}\ dt_{mx+m-1} = k_l$
dt={ $dt_{ax,}, dt_{ax+1,\dots,}\ dt_{ax+m}$ }
Return dt

Tip vektörlerinin oluşturulması ve her bir güç eğri matrisi ile ilinti değerlerinin hesaplanması Algoritma 3.9.' da gösterilmiştir. İlinti eğrilerinde, doğru tipleri bulunduran vektör için en yüksek ilinti değerinin elde edilmesi beklenmektedir. Yanlış tipleri içeren vektörler için de içerdiği doğru bit değeriyle orantılı bir ilinti değeri gözlenecektir. Bunun yanı sıra pencerede eğer tek tip bit bulunuyor ise farklı tipler içeren hiç bir vektör için yüksek ilinti oluşmayacaktır. Sonuç olarak tek tipli pencerelerden elde edilen en yüksek ilinti değerleri bile diğer pencerelere ait en yüksek ilinti değerlerinden oldukça düşük olacağından bu özellik de tek tip bit içeren pencerelerin diğerlerinden ayırt edilebilmesi amacıyla kullanılmıştır. Normalde her m=4 bitlik pencerede, tek tip bit içeren K_0="00...0", K_m="11..1" durumları hariç 16-2=14 tane farklı tip vektörü bulunmaktadır. Ancak sadece "tip0" ve "tip1" olmak üzere iki tip bit türü olduğundan, her tip vektörünün tam tersi (eşlenik) bitleri içeren vektörler, mutlak değeri aynı olan ilinti değerlerini verecektir. O halde 7 tane tip vektörü olasılığının göz önünde tutulması yeterli olabilmektedir. Bitlerin kesin tipi ise bir tane bitin değerinin kesin olarak bilinmesi ve her bir dörtlünün sonraki dörtlü ile ortak birer bit bulundurması sorunu ile çözülebilecektir. Tüm seçenekler arasında en yüksek ilinti değerini veren vektörün kendisi ve eşleniğinin işaretinin kontrol edilmesi yeterli olacaktır. Elde edilen ilinti değerlerinin kullanılarak m bitlik anahtar penceresinin tipine nasıl karar verildiği ise Algoritma 3.10.' da verilmiştir.

3.4.1.1. Gerekli eğri sayısının hesaplanması

Gerçekleştirilen yenilikçi İGA saldırısında, bitleri elde etmede kullanılan pencere genişliği önemli bir parametredir. Bu parametre hem hesaplama karmaşıklığı hem de bit değerlerini elde etmede kullanılması gereken eğri sayısında etkili olabilecektir. Çalışmada, güç tüketimi ile ilgili analizlerde istatistikteki güven aralığı kavramından faydalanılmıştır. Analizler [71]' de verilen doğrusal güç modeli kullanılarak gerçekleştirilmiştir. Eğer [71]' de olduğu gibi tip kestirim vektörleri ile güç tüketimi arasında basit doğrusal bir ilişki olduğu düşünülürse tip vektörü K_k ve güç eğri matrisi P_x arasında aşağıdaki gibi bir eşitlik yazılabilir. Burada b gürültüdür.

$$P_x = a * K_k + b \tag{3.17}$$

Gürültü b'nin diğer bileşenlerle ilişkisiz olduğu düşünülürse ilinti katsayısı C_{xk} aşağıdaki gibi ifade edilecektir.

$$C_{xk} = \frac{Cov(P_x,K_k)}{Var(P_x).Var(K_k)} = a.\frac{Var(K_k)}{Var(P_x)} \tag{3.18}$$

$$Cov(P_x, K_k) = \left(\frac{1}{4M-1}\sum_{i=1}^{4M}(P_x(i) - \overline{P_x})\left((K_k(i) - \overline{K_k})\right)\right) \tag{3.19}$$

$$Var(P_x) = \frac{1}{4M-1}\sum_{i=1}^{4M}(P_x(i) - \overline{P_x})^2 \tag{3.20}$$

$$Var(K_k) = \frac{1}{4M-1}\sum_{i=1}^{4M}(K_k(i) - \overline{K_k})^2 \tag{3.21}$$

Yanlış tip kestirim vektörü $K_{k'}$ için hesaplanan ilinti değeri $C_{xk'}$, eşitlik (3.18) kullanılarak doğru ilinti değeri C_{xk} türünden aşağıdaki gibi ifade edilebilir. İlinti katsayıları doğaları gereği: , -1< $C_{kk'}$<1 aralığında yer alır. Eşitlikten görüleceği gibi yanlış tip kestirim vektörü $C_{xk'}$, $C_{kk'}$ kez C_{xk} değerinden küçük olacaktır.

$$C_{xk'} = \frac{Cov(a*K_k+b,\ K_{k'})}{Var(P_x)*Var(K_{k'})} = \frac{a.Var(K_k).Cov(K_k,K_{k'})}{Var(P_x).Var(K_k)\,Var(K_{k'})} = C_{xk} * C_{kk'} \tag{3.22}$$

Brier ve arkadaşlarının çalışmasında [71] gösterildiği gibi işaret/gürültü oranı (SNR), ilinti katsayıları türünden aşağıdaki gibi ifade edilebilir.

$$SNR(P_x) = \frac{C_{xk}}{\sqrt{1 - {C_{xk}}^2}} \tag{3.23}$$

Yanlış tahminler için SNR değeri aşağıdaki gibidir:

$$SNR(P_x') = \frac{C_{xk} * C_{kk'}}{\sqrt{1 - {C_{xk}}^2 * {C_{kk'}}^2}} = \frac{C_{xk}}{\sqrt{\frac{1}{{C_{kk'}}^2} - {C_{xk}}^2}} \tag{3.24}$$

Eşitliklerden görüleceği gibi $SNR(P_x) > SNR(P_x')$ ve $C_{kk'}$ değeri SNR değerini ters yönde etkiler.

Tablo 3.1. m=3 ve m=4 için tip vektörleri ilinti katsayıları

	1	**2**	**3**
1	1	0,5	0,5
2	0,5	1	0,5
3	0,5	0,5	1

a)m=3

	1	**2**	**3**	**4**	**5**	**6**	**7**
1	1,00	0,33	0,58	0,33	0,58	0,58	0,33
2	0,33	1,00	0,58	0,33	0,58	0,58	0,33
3	0,58	0,58	1,00	0,58	0,00	0,00	0,58
4	0,33	0,33	0,58	1,00	0,58	0,58	0,33
5	0,58	0,58	0,00	0,58	1,00	0,00	0,58
6	0,58	0,58	0,00	0,58	0,00	1,00	0,58
7	0,33	0,33	0,58	0,33	0,58	0,58	1,00

b)m=4

Pencere genişlikleri m=3, m=4 ve m=5 değerlerini aldığında $C_{kk'}$'nın alacağı değerlerin mutlak değeri Tablo 3.1.'de verilmiştir. Pencere genişliği m=3, için doğru değere en yakın olan yanlış ilinti değerinin 0,5 olması gerekir. M=3 bitlik pencerede tüm yanlış tahminlerin ilinti değerlerinin aynı olması beklenir. Bu değer m=4 için 0,58'dir ve yanlış tahminlerin farklı genliklerde olması beklenir. Pencerenin m=5 genişlik değeri için ise 0,67 değeri elde edilir. Yine yanlış tahminler farklı genliklerde değer alır. Her bir pencere değeri için gerekli eğri sayısı M'yi hesaplamak için [21]' deki yöntemden yararlanılabilir. Bu yöntemi kullanırken doğru tahmin C_{xk} ' ye en yakın $C_{xk'}$ değerini veren vektörlerin 1-α olasılık değeri ile birbirinden ayırt edilebilirliği (3.25) eşitliği kullanılarak hesaplanabilir:

Örneğin m=3 için en yüksek yanlış ilinti değeri $C_{kk'} = 0.5$ değerini alır. Güç eğrileri kullanılarak doğru kestirimler için elde edilebilen en yüksek ilinti değerinin C_{xk}=0.07 olduğunu varsayalım. En yüksek ilintili yanlış kestirimi 1-α=0.8 olasılıkla elde etmek

için $Z_{1-\alpha}$=0.842 olur. Gerekli eğri sayısı M=1153 olacaktır. Burada Z değeri "Fischer's-Z Transform" değerleridir [24]. 0.8 güvenlik seviyesi için m=4 bitlik pencere kullanıldığında M=1633 eğri ve m=5 için ise M=2641 eğri gerekir. Sonuç olarak gerekli eğri sayısı seçilen pencere genişliği ile de artar.

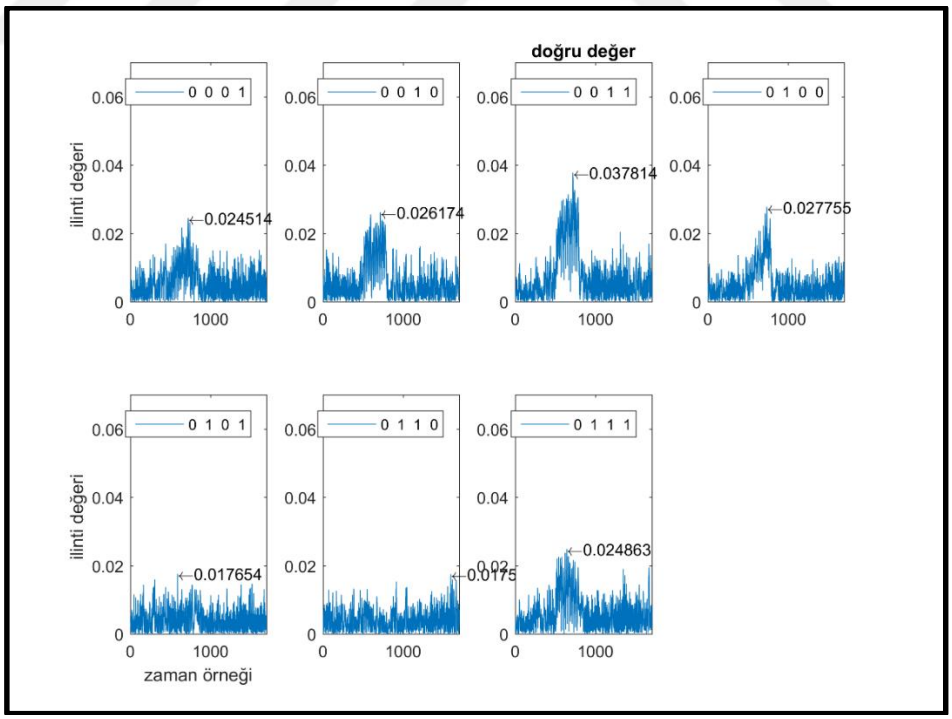
$$M = 3 + 8 * \frac{{Z_{1-\alpha}}^2}{(\ln\frac{1+\ C_{xk}}{1-\ C_{xk}} + \ln\frac{1+\ C_{xk'}}{1-\ C_{xk'}})^2} \tag{3.25}$$

Burada $C_{xk'} = C_{xk} * C_{kk'}$değerindedir.

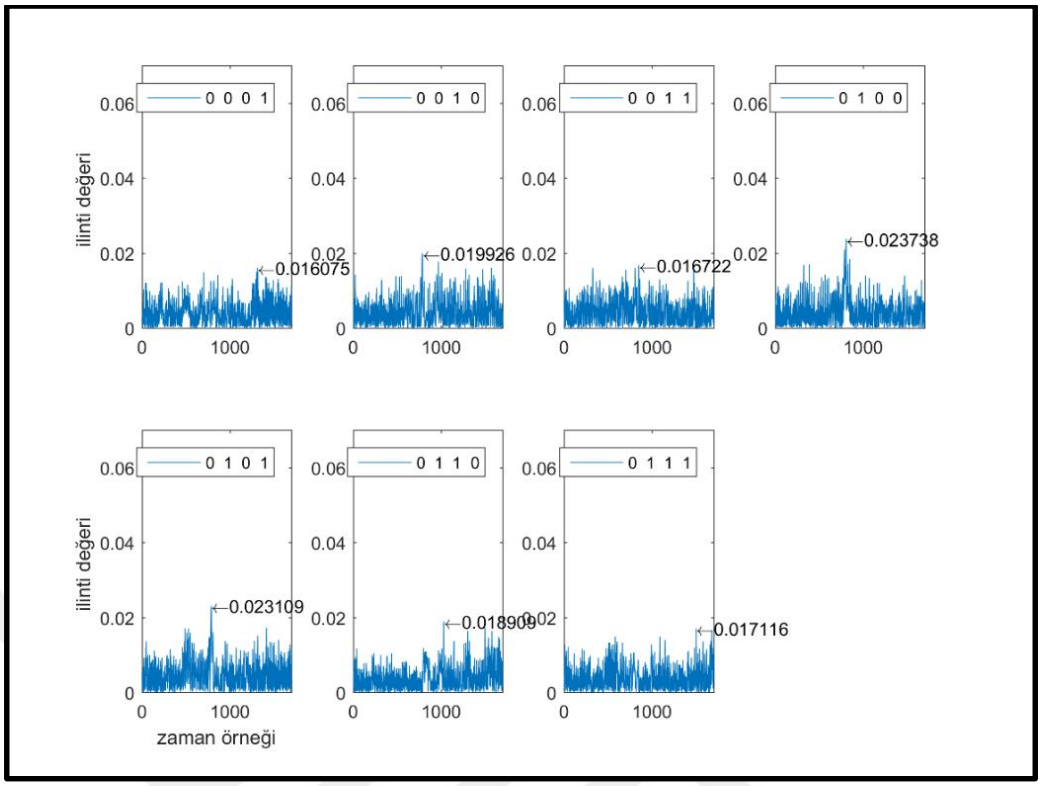
$$M = 3 + 8 * \frac{{Z_{1-\alpha}}^2}{(\ln\frac{1+\ C_{xk}}{1-\ C_{xk}} + \ln\frac{1+\ C_{xk} * C_{kk'}}{1-\ C_{xk} * C_{kk'}})^2} \tag{3.26}$$

3.4.2. Yöntemin ASIC ML devresine uygulanması

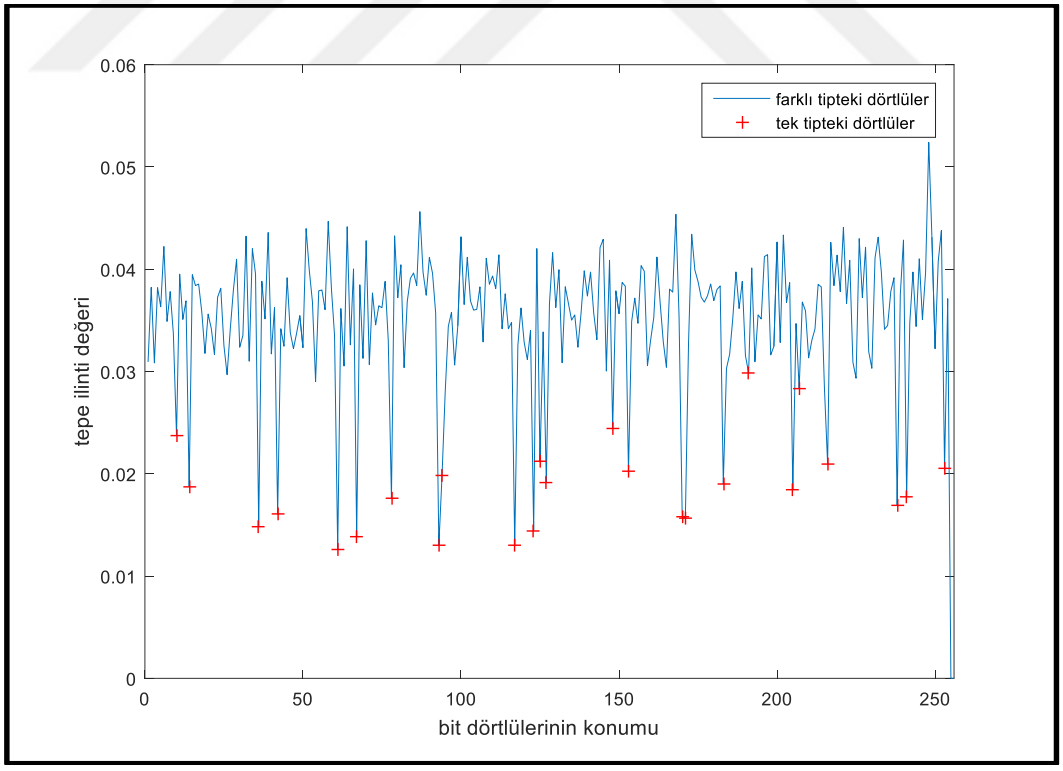
ASIC devreden alınan ölçümlere yöntemi uygulamak amacıyla güç eğrisi bölütleri ile tip kestirim vektörlerinin ilintileri hesaplanmıştır.



Şekil 3.34. Birden fazla tip içeren dörtlü kestirimlerine ait ilinti eğrileri



Şekil 3.35. Tek tip içeren dörtlü kestirimlerine ait ilinti eğrileri



Şekil 3.36. 10000 eğri için dörtlü gruplardan elde edilen en yüksek ilinti değerleri

Şekil 3.34.'de her iki tipe ait anahtar bitlerini içeren bir pencere ait güç eğrisi bölütleri ile tip kestirim vektörlerine arasındaki ilinti eğrileri görülmektedir. Hedef dörtlü pencere için "0011" ya da "1100" değerlerinden biri doğrudur ve şekilden görüleceği gibi bu değer için en yüksek ilinti elde edilmiştir. Şekil 3.35.' de ise sadece tek tip bitleri içeren bir dörtlüye ait ilinti eğrileri verilmiştir. Şekilden görüleceği gibi hiç bir bit değeri için göze çarpar bir tepe bulunmamaktadır

Şekil 3.36.'da dörtlü gruplardan elde edilen en yüksek ilinti değerleri görülmektedir. Bu şekilde"+" ile işaretlenmiş alanlar hep sıfır ya da hep 1'lerden oluşan grupları temsil etmektedir ve bu gruplara ait en yüksek ilinti değerleri, iki tip içerenlerden belirgin bir şekilde ayrılmaktadır.

3.5. Şablon Tipi İlintisel Güç Analizi

3.5.1. Yöntemin Tanıtımı

Şablon Tipi İGA yöntemi ile [19], daha önce literatüre tanıtılmış olan yenilikçi İGA yönteminin [18] geliştirilmesi hedeflenmiştir. Yenilikçi İGA çalışmasında yanlış tip kestirimini içeren vektörlerin güç eğrileriyle olan ilintisinin, kestirim vektörünün içerdiği doğru bit sayısıyla orantılı bir değerde olması beklenmektedir. Bu özellik aslında daha etkin bir karar verme mekanizması geliştirilmesinde kullanılabilir

Algoritma 3.11. İlinti değerlerinin hesaplanması

PARAMETRELER
m= pencere genişliği
w=anahtar bit sayısı

GİRDİLER
C_{xk}, $0<k<2^{m-1}$, $1<x<w/m$
$K = \{K_0, K_1, \dots, K_{2^{m-1}}\}, K_0=[k_0\ k_0, . , k_0], .., K_{2^{m-1}}=[k_{2^{m-1}}\ k_{2^{m-1}} \dots k_{2^{m-1}}]$

ÇIKTILAR
C_{xk}'', $0<k<2^{m-1}$, $1<x<w/m$

```
For x= 1 to w/ m
    For k= 1 to 2^(m−1)
        If ( C_xk < eşik

            C_xk'' =abs(corr(C_xk, K_k))

Return C_xk''
```

Belli bir kestirim vektörün doğru olması durumunda, her bir yanlış adaya ait güç eğrisinin tepe değerinin, bu yanlış aday vektörü ile doğru tip vektörü arasındaki ilinti değeri ile orantılı bir genliğe sahip olması beklenmektedir. O halde doğru aday vektörü bulmak için sadece en yüksek tepe değeri veren eğriye göre karar vermek yerine, tüm adaylar için oluşturulmuş ilinti eğrilerinin tepe değerleri arasında beklenen ilişkinin olup olmadığı da incelenebilir. İleri bölümlerde teorik ve deneysel olarak da gösterildiği gibi, bu tür bir karar mekanizması kullanarak daha az eğri sayısı ile anahtar bitleri elde edilebilmektedir.

Tüm kestirim tiplerinin gerçek eğrilerle olan ilinti eğrilerindeki genlik değerlerinin kullanılarak doğru tip vektörünün diğerlerinden nasıl ayırt edilebileceği Algoritma 3.12.'de verilmiştir.

Algoritma 3.12. Pencere Bit Tiplerinin Elde Edilmesi

GİRDİLER
C_{xk}, C_{xk}'' ,
$K=\{K_0, K_1, \dots, K_{2^{m-1}}\}$
$k_0 = [0 \dots 0\ 0]$, $k_1=[0 \dots 0\ 1]$, $k_2=[0 \dots 1\ 1]$, $k_{2^{m-1}}=[0\ 1 \dots 1]$,
Length (k_i)=m

ÇIKTILAR
$dt=\{ dt_{w-1}, \dots dt_0 \}$

For x= 0 to w-1 /m
  For k= 2 to 2^{m-1}
    If(mean(Ci_{xk}))<eşik
      $dt_{mx}\, dt_{mx+1\,..}\, dt_{mx+m-1} = k_0$
    Elseif (max (C_{xk}''))== C_{xl}''
      $dt_{mx}\, dt_{mx+1\,..}\, dt_{mx+m-1} = k_l$
$dt=\{ dt_{ax,}, dt_{ax+1,\dots,}\, dt_{ax+m} \}$
Return dt

3.5.1.1. Gerekli eğri sayısının hesaplanması

Önceki bölümde anlatılan yenilikçi İGA yönteminde, tip kestirim vektörleri ile güç tüketimi arasında basit doğrusal bir ilişki olduğu varsayılarak [71]'deki yaklaşımdan faydalanarak bir model geliştirilmiştir. Tip vektörü K_k ve güç eğri matrisi P_x arasındaki doğrusal ilişkiyi aşağıdaki gibi ifade edelim.

$$P_x = a.K_k + b \tag{3.27}$$

Burada b gürültüdür ve "b" değerinin diğer bileşenlerle ilişkisiz olduğu düşünülürse ilinti katsayısı C_{xk} (3.28)'daki gibi hesaplanabilir [71]. İlinti hesabı için gerekli

kovaryans ve varyans değerlerinin hesaplanışı ise sırasıyla (3.29) ve (3.30) eşitliklerinde verilmiştir

$$C_{xk} = \frac{\mathrm{Cov}(P_x, K_k)}{\sqrt{\mathrm{Var}(P_x).\mathrm{Var}(K_k)}} = \frac{a.\sqrt{\mathrm{Var}(K_k)}}{\sqrt{\mathrm{Var}(P_x)}} \tag{3.28}$$

$$\mathrm{Cov}(P_x, K_k) = \left(\frac{1}{4M-1}\sum_{i=1}^{4M}(P_x(i) - \overline{P_x})\left((K_k(i) - \overline{K_k})\right)\right) \tag{3.29}$$

$$\mathrm{Var}(P_x) = \frac{1}{4M-1}\sum_{i=1}^{4M}(P_x(i) - \overline{P_x})^2 \tag{3.30}$$

$$\mathrm{Var}(K_k) = \frac{1}{4M-1}\sum_{i=1}^{4M}(K_k(i) - \overline{K_k})^2 \tag{3.31}$$

Yanlış tip kestirim vektörü $K_{k'}$ için hesaplanan ilinti değeri $C_{xk'}$, doğru vektöre ait ilinti değeri C_{xk} türünden, (3.28) denklemi kullanılarak (3.32) deki gibi ifade edilebilir:

$$C_{xk'} = \frac{\mathrm{Cov}(a*K_k+b,\ K_{k'})}{\sqrt{\mathrm{Var}(P_x)*\mathrm{Var}(K_{k'})}} = \frac{a*\mathrm{Cov}(K_k,K_{k'})}{\sqrt{\mathrm{Var}(P_x)*\mathrm{Var}(K_{k'})}} = \frac{a.\sqrt{\mathrm{Var}(K_k)}}{\sqrt{\mathrm{Var}(P_x)}}.\frac{\mathrm{Cov}(K_k,K_{k'})}{\sqrt{\mathrm{Var}(K_k)*\mathrm{Var}(K_{k'})}} = C_{xk} * C_{kk'} \tag{3.32}$$

İlinti katsayılarının doğası gereği -1< $C_{kk'}$<1 eşitliği sağlanır. Bu durumda, (3.32) eşitliğinden görüleceği gibi, yanlış tip kestirim vektörü $C_{xk'}$ değeri, $C_{kk'}$ kez C_{xk} değerinden daha küçük olacaktır. Ancak önerilen yeni yöntemde, doğru tip kestirim vektörüne karar vermek için, bu tip vektörünün güç eğrileriyle ilintisinin tepe değeri olan $C_{xk'}$ değerlerinin, bu tip vektörünün doğru tip vektörüyle ilintisini gösteren $C_{kk'}$ değerleriyle ne derece orantılı (yani ilintili) olduğuna bakılacaktır. Bu ilinti değerini $C_{xk''}$ olarak isimlendirirsek, C_{xk} ve $C_{kk'}$ türünden değeri aşağıdaki gibi ifade edilebilir:

$$C_{xk''} = C_{xk'}.C_{kk'} = C_{xk}.C_{kk'}.C_{kk'} \tag{3.33}$$

Burada [21]'de verilen yaklaşım kullanılarak doğru kestirim değeri C_{xk} yı $C_{xk''}$ 'dan ayırt edebilmek için kullanılması gereken en az eğri sayısı aşağıdaki gibi hesaplanabilir:

$$M = 3 + \frac{8.\ Z_{1-\alpha}{}^2}{(ln\frac{1+\ C_{xk}}{1-\ C_{xk}} + ln\frac{1+\ C_{xk''}}{1-\ C_{xk''}})^2} = 3 + \frac{8.\ Z_{1-\alpha}{}^2}{(ln\frac{1+\ C_{xk}}{1-\ C_{xk}} + ln\frac{1+(\ C_{xk}*\ C_{kk'}*\ C_{kk'})}{1-(\ C_{xk}*\ C_{kk'}*\ C_{kk'})})^2} \quad (3.34)$$

Pencere genişlikleri m=4 için $C_{kk'}$ nın ve $C_{kk''} = C_{kk'} * C_{kk'}$ 'nın alacağı değerlerin mutlak değeri Tablo 3.2.'de verilmiştir. Pencere genişliği m=4, için doğru değere en yakın olan yanlış ilinti değeri $C_{kk'}$=0.58 iken $C_{kk''}$=0,03 değerini almaktadır.

Tablo 3.2. m=3 ve m=4 için tip vektörleri ilinti katsayıları

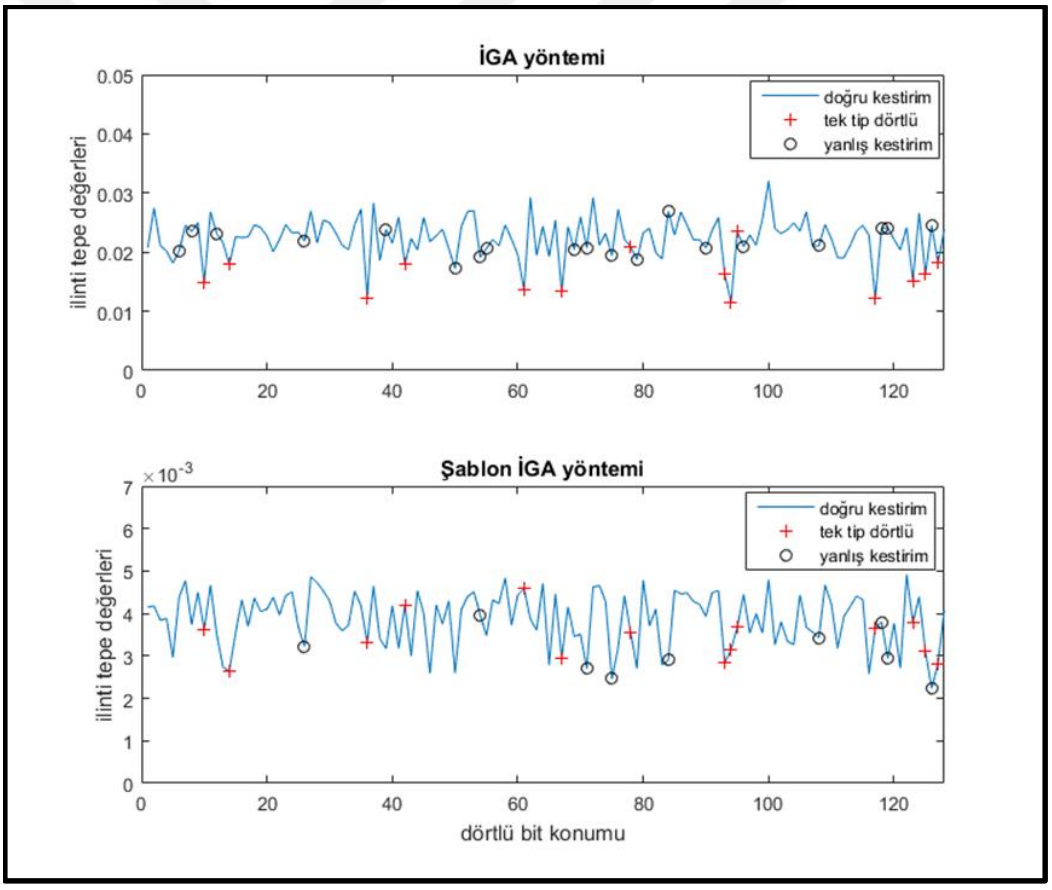
	0001	0010	0011	0100	0101	0110	0111		0001	0010	0011	0100	0101	0110	0111
0001	1,00	0,33	0,58	0,33	0,58	0,58	0,33		1	0,29	0,06	0,29	0,06	0,06	0,29
0010	0,33	1,00	0,58	0,33	0,58	0,58	0,33		0,29	1	0,06	0,30	0,06	0,06	0,30
0011	0,58	0,58	1,00	0,58	0,00	0,00	0,58		0,06	0,06	1	0,06	0,30	0,30	0,06
0100	0,33	0,33	0,58	1,00	0,58	0,58	0,33		0,29	0,29	0,06	1	0,06	0,06	0,30
0101	0,58	0,58	0,00	0,58	1,00	0,00	0,58		0,06	0,06	0,30	0,06	1	0,30	0,06
0110	0,58	0,58	0,00	0,58	0,00	1,00	0,58		0,06	0,06	0,30	0,06	0,30	1	0,06
0111	0,33	0,33	0,58	0,33	0,58	0,58	1,00		0,29	0,29	0,06	0,29	0,06	0,06	1

Eğer biz doğru tip vektörünü, ona en yakın bitleri içeren yani en yüksek ilintili olduğu yanlış kestirim değerinden $Z_{1-\alpha} = 0.842$ olasılıkla ayırt etmek istersek $C_{xk''} = 0.029 * 0.03 = 0.00087$, gereken eğri sayısı Mt=1793 olacaktır. Burada Z, "Fiser's-Z dönüşümü" olarak isimlendirilir [24] ve ayrıntılı kullanımı [21]'de bulunabilir.

Bununla birlikte daha önce önerilmiş yöntem [18] için bu eğri sayısı hesaplanırken $C_{kk'}$=0.58, alınır ve gerekli eğri sayısı 9550 değerini alır.

3.5.2. Yöntemin ASIC ML devresine uygulanması

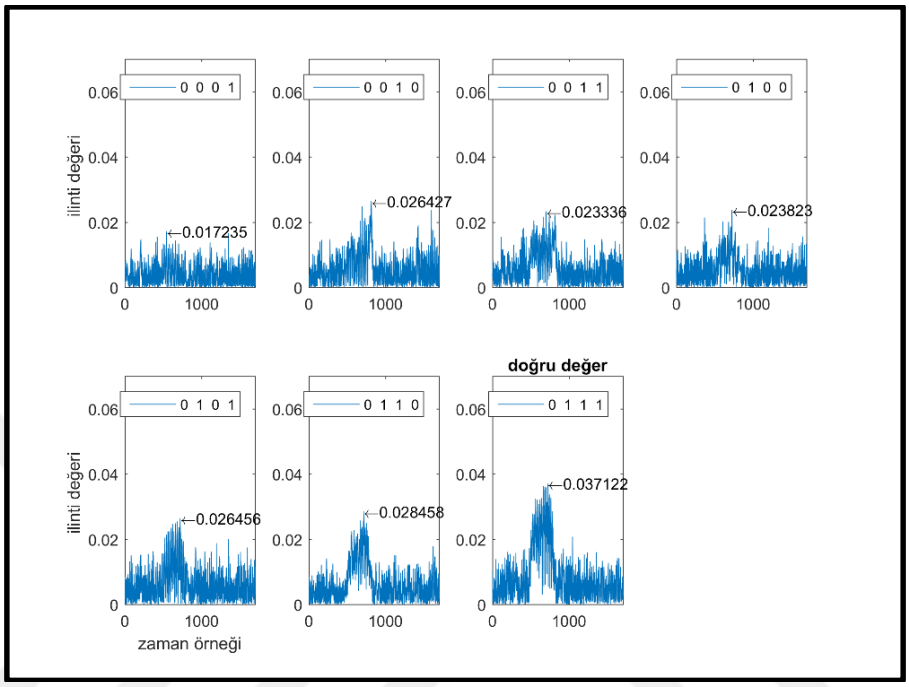
Şekil 3.37.'de İGA ve şablon tabanlı İGA için 3000 eğri kullanılarak elde edilen en yüksek ilinti değerleri yani $C_{xk'}$ *ve* $C_{xk''}$ görülmektedir. Burada siyah daireler yanlış kestirimleri "kırmızı +" lar ise özdeş bit tiplerini içeren pencereleri göstermektedir. Şekilden anlaşılacağı gibi şablon tipi İGA için yanlış kestirimlerin sayısı daha azdır. Ancak özdeş bitlerden oluşan pencereler için [18]'de anlatılan yöntem, diğer pencereler için ise şablon tabanlı İGA yöntemi daha etkin bir şekilde kullanılabilecektir.



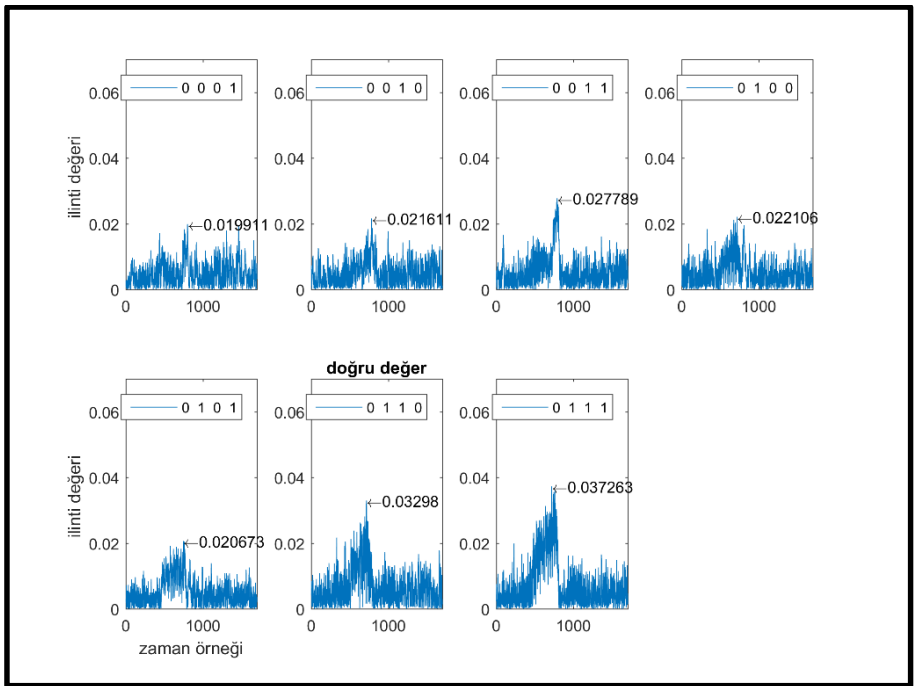
Şekil 3.37. 3000 eğri için İGA ve şablon İGA ilinti değerleri

Şekil 3.38. 'de her iki yöntemle de doğru şekilde ayırt edilebilen bir pencereye ait aday ilinti eğrileri görülmektedir. Burada en yüksek ilinti değeri doğru tip vektörüne aittir. Şekil 3.39.' da ise sadece şablon İGA ile ayrıt edilebilen bir pencereye ait ilinti

değerleri görülmektedir. Burada en yüksek ilinti değeri doğru vektöre ait olmamasına rağmen, tüm ilinti eğrilerinin arasındaki ilişki kullanan şablon yöntemi ile, doğru tip vektörü bulunabilmektedir. Bu durum pratik olarak da şablon İGA yönteminin başarımı artırdığının bir kanıtıdır.



Şekil 3.38. Her iki yöntemle doğru karar verilebilen ilinti eğrileri



Şekil 3.39. Sadece şablon İGA ile doğru karar verilen ilinti eğrileri

3.6. Geliştirilen yöntemlerin birbirleri ile karşılaştırılması

Bu bölümde ASIC ML ve FPGA ikilik üs alma algoritmalarına uygulanan yöntemlerin kendi aralarında karılaştırılması gerçekleştirilmiştir. Geliştirilen yöntemlerin uygulama başarımını değerlendirmek amacıyla saldırlar açısından önemli olan ölçütler şu başlıklar altında toplanabilir:

- Kullanılan eğri sayısı: Her bir yöntemde saldırı uygulanan anahtar bitlerinin tamamının ya da elde edilebilen en yüksek bit sayısı örneğin bitlerin %98'inin kaç eğri ile elde edildiğini göstermektedir. Eğri sayısını belirtmek üzere "N" terimi kullanılmıştır. En yüksek eğri sayısı ise "n" değeri ile gösterilmiştir. Böylece yöntemlerin gerektirdiği eğri sayılarının birbirileri ile göreceli olarak karşılaştırabilmesi sağlanmıştır.
- İşlem türü ve sayısı: Geliştirilen yöntemleri uygulamak için kullanılan temel işlem türleri, ilinti, çapraz ilinti ve DFT' den oluşmaktadır. Bu işlem türlerinde işlem girdilerinin boyu da hesap yükünü önemli derecede artıran etkenler arasındadır. İşlem sayısı, kullanılan yönteme göre anahtar bitlerini elde etmek için gerekli N tane eğri için, kaç işlem yapılması gerektiğini ifade etmektedir. Gerekli işlem sayısı özellikle saldırının geçekleştirme süresini etkilemektedir.
- Anlık bellek gereksinimi: Yukarıda belirtilen ilinti, çapraz ilinti ve DFT gibi temel işlem türlerinin her birinin bellek kullanımı birbirinden farklıdır. Bunun yanı sıra aynı tür işlemlerde, parçalara ayrılamayan en düşük işlem girdisi boyu anlık bellek gereksinimini önemli ölçüde etkilemektedir. Örneğin dikeyde ilinti hesaplanırken eğri sayısının artırılması gerektiğinde, bu işlemin alt parçalara ayrılması mümkün olmadığından anlık bellek gereksinimi de yüksek olmaktadır.

Geliştirilen yöntemlerin, ASIC devreye uygulanması ile elde edilen deneysel sonuçları da göz önünde bulundurularak, yukarıda sıralanan ölçütler kapsamında değerlendirmeleri Tablo 3.3.'de verilmiştir. Tablo 3.3.'den görüleceği gibi en az eğri ile sonuca ulaşma konusunda "Şablon İGA" yöntemi en başarılıdır. İşlem sayısı açısından "tüm bitler çapraz ilinti " yöntemi, bütün anahtar bitleri için çapraz ilinti değerinin hesaplanmasını gerekmektedir. Her ne kadar yöntem gerekli eğri sağsını azaltsa da işlem sayısını artırma oranı daha yüksek olmaktadır. Frekans uzayı analizi ise kullanılan eğri sayısını azaltmakla birlikte hesaplanacak çapraz ilinti sayısına ek

olarak FFT hesap yükü de getirmektedir. Ancak çapraz ilintide kullanılan işlem boyunu azaltarak da bir avantaj sağlamaktadır. . Anlık bellek gereksinimi açısından hem Şablon İGA hem de onun öncülü olan İGA yöntemleri, daha düşük boyutlara indirilemeyen dikey ilinti hesabı nedeni ile zorlayıcı olmaktadır.

Tablo 3.3. ASIC devreye uygulanan yöntemlerin karşılaştırılması

	Çapraz İlini	**Tüm Bitler Çapraz İlinti**	**Frekans Uzayı Çapraz İlinti**
Ölçüm sayısı (N)	N=40000	N=10000	N=28000
İşlem sayısı	Çapraz ilinti: n. d Çapraz ilinti boyu: L	Çapraz ilinti: $N.d^2 = n.0.25.d^2$ Çapraz ilinti boyu: L	Çapraz ilinti: $N.d = n.0.7.d$ Çapraz ilinti boyu:0.5L DFT: N. d= n. 0.7. d DFT boyu: 0.5L
Anlık Bellek Gereksinimi	Düşük	Düşük	Düşük
	İGA	Şablon İGA	
Ölçüm Sayısı	10000	<1000	
İşlem Sayısı	İlinti: $d*(2^{w-1}-1)$ İlinti boyu: N.w =0.25.n.w	İlinti:$d.(2^{w-1}-1)$ N.w = $d.(2^{w-1}-1)$ 0.25.n.w İlinti boyu: N*w =0.25.n.w	
Anlık Bellek Gereksinimi	Yüksek	Yüksek	

Tablo 3.4.'de hem ASIC hem de FPGA devreye uygulanan "çapraz ilinti" ve "tüm bitler çapraz ilinti" yöntemlerinin karşılaştırılması yapılmıştır. . Yöntemin ASIC devre uygulamasında tek referans bit yöntemine göre eğri sayısında yaklaşık olarak %75 oranında iyileşme sağladığı görülürken, FPGA'li devrede bu oran çok yüksek olmayıp % 10'lar civarındadır. FPGA devreden alınan güç ölçümlerinin 3.1.5 bölümünde anlatılan nedenlerden ötürü, ölçüm düzeneğinden açısından daha avantajlı daha az elektronik gürültüye sahip olması, tüm bitler çapraz ilinti analizinin getirdiği daha fazla eğri bölütü kullanımına ilişkin avantajın çok da önemli olmamasına neden olduğu düşünülmektedir. Bu yöntemde, [16] yöntemiyle gerçekleştirilen saldırıdan daha fazla bit değeri elde edilebilse de yine de anahtar bitlerinin tamamı elde edilememiştir. Bu durum da yine 3.1.5 bölümünde açıklandığı gibi, FPGA devresinin belli bölümlerinde oluşan sentezleme ve yerleştirme-bağlama aşamalarında oluşan farklılıklardan, kullanılan anahtar ve rastgele veri değerlerinden kaynaklanabilir.

Tez kapsamında geliştirilen ilk çalışma olan tek referans bit kullanımına dayalı çapraz ilinti yöntemi [16], "seçilmiş mesaj" kullanarak çok daha az eğri ile işlem yapar hale

getiren Wang ve arkadaşları tarafından [59], literatürdeki diğer temel çapraz ilinti yöntemleriyle karşılaştırılmıştır (Ek-B). Karşılaştırılması yapılan bu yöntemimiz [16], tez kapsamında geliştirilen ilk yöntem olup eğri sayısı gereksinimi açısından da en kötü performansa sahip çalışmadır. Bu nedenle [59]'da verilen karşılaştırma tablosu (Ek-B). temel alınarak tez kapsamında geliştirilen çalışmaların literatürdeki diğer çapraz ilinti tabanlı ve birden fazla eğri kullanımı gerektiren yöntemlere göre değerlendirmesi şöyle yapılabilir: Çalışmalar Kim ve arkadaşları [29] ve Witerman ve arkadaşları [30]'na ait ve temelde dikeyde çapraz ilinti hesabına dayalı yöntemlerden daha az eğri sayısıyla çalışmaktadır. Bunun yanı sıra Wan ve arkadaşları [31] ve Wang ve arkadaşlarına ait [59] ve temelde [16] çalışmasının devamı olan ve yatayda çalışan yöntemlerden ise daha kötü performansa sahiptir. Ancak [59] çalışması açık veri üzerinde kontrol sahibi olmayı gerektirdiği için tüm RSA gerçeklemelerine uygulanabilir türde bir yöntem olmamaktadır.

Tablo 3.4. FPGA devreye uygulanan yöntemlerin karşılaştırılması

	Çapraz İlini	**Tüm Bitler Çapraz İlinti**
Ölçüm sayısı (N)	200	180
İşlem türü ve sayısı	Çapraz ilinti: n. d Çapraz ilinti boyu: L	Çapraz ilinti: $N.d^2 = n.0.9.d^2$ Çapraz ilinti boyu: L
Bellek Kullanımı	Düşük	Düşük
Alan Hassasiyeti	Önemli	Önemli
Elde edilen bit sayısı	175	178

4. SONUÇLAR

Bu tez çalışması kapsamında temel olarak çapraz ilinti tabanlı 3, ilintisel güç analizi tabanlı 2 özgün yöntem geliştirilmiş ve SCI kapsamındaki dergilerde 1 adet makale yayınlanmış ve uluslararası konferanslarda 3 adet sunum gerçekleştirilmiştir. Geliştirilen çapraz ilinti tabanlı yöntemlerden ilki, belli tipteki tek bir referans bite ait güç eğrisi alanının, diğer bitlerinki ile olan ilintisini kullanmaktadır. Hesaplanan bu ilinti değerlerine göre hedef bit, referans ile aynı ya da farklı olarak sınıflandırılarak anahtar bitleri elde edilmektedir. Çalışmada, çapraz ilinti değerlerini pek çok eğri kullanarak daha iyi kestirmek amacıyla biri ilintiler toplamı diğeri ise bir oylama mekanizmasına dayanan iki farklı yol önerilmiştir. İlintiler toplamı yöntemi oylama yöntemine göre daha iyi bir performans sergilediği görülmüştür. Çapraz ilinti tabanlı yöntemin uygulanması sırasında, farklı tipteki referans bitlerin kendi türleri ile olan çapraz ilinti değerlerinin birbirinin tersi davranışlar sergilediği gözlenmiştir. Gözlenen bu özellik de aslında tek bir bit yerine tüm bitleri referans olarak kullanabilen ve tüm bitler çapraz ilinti analizi olarak isimlendirilen daha ileri bir saldırı türünün geliştirilmesini sağlamıştır. Geliştirilen bu yeni yöntem ile tek referans bitin kullanıldığı duruma göre daha az eğri ile sonuca ulaşılabilmektedir. Temelinde çapraz ilinti kullanan bu iki yöntem, hem hazır ML tabanlı ASIC RSA işlemcisine hem de Verilog dili ile geliştirilen ve ikilik üs alma yöntemi kullanan FPGA tabanlı RSA devresine uygulanmıştır. Saldırının odaklandığı RSA algoritmasının temel üs alma döngüsünde, FPGA tabanlı devrenin gerçek ve davranışsal seviye benzetim eğrileri arasında davranış farklılıkları gözlenmiştir. Bu durum gerçek ölçümlerden toplanmış güç eğrilerinde görülen çapraz ilinti davranışlarının, sadece algoritma girdileri ve algoritma akışından değil, alttaki elektronik devre özelliklerinden de kaynaklandığını göstermektedir. Bu durumun tam olarak netleştirilmesi konusu "ileri çalışmalar" kapsamında devam ettirilecektir. Bunun yanı sıra özellikle çapraz ilinti tabanlı yöntemde, ilinti değerlerinin varyans değerlerinin de her bir bitin tipini tespit etmede kullanılabileceği görülmüştür. Bu da yine bir gelecek dönem çalışması olacaktır.

Zaman uzayında gerçekleştirilen çapraz ilinti tabanlı saldırılara bir iyileştirme de bu saldırıların frekans uzayına taşınması ile gerçekleştirilmiştir. Literatürde belirtildiği gibi, eğri bölütlerinin yanaştırma sorunlarına daha bağışık olması ve sinyal/gürültü oranının daha yüksek olduğu alt bantların kullanımına olanak vermesi, frekans uzayında daha az eğri kullanarak daha başarılı sonuçların elde edilmesini sağlayan etkenlerdir. Bu durumun çapraz ilinti tabanlı yöntem için de geçerli olduğu gerçekleştirilen uygulamalar ile görülmüştür.

Geliştirilen diğer iki özgün yöntemden ilki ise güç eğrilerini hem yatayda, yani aynı güç eğrisindeki güç eğrisi alanlarını kullanan, hem de dikeyde yani farklı eğrilerdeki aynı anlara ait eğri alanlarını işleyen ilintisel güç analizi tabanlı yöntemdir. Bu yönteme bir ilerleme olarak geliştirilen şablon tipi İGA yönteminde ise, yanlış tip kestirimlerini içeren vektörlerinin de, güç eğrileri ile içerdikleri doğru bit sayısıyla orantılı bir ilintiye sahip olmaları gerektiği gerçeği kullanılmıştır. Bu özellikten de daha etkin bir karar verme mekanizması geliştirilmesinde faydalanılmıştır. Geliştirilen bu yeni yöntemde tüm ilinti eğrilerinden gelen bilginin kullanılması nedeni ile daha az eğri ile sonuca ulaşılabilmektedir.

Geliştirilen her bir yöntemde, başarılı bir saldırı gerçekleştirmek için ne kadar eğri kullanılması gerektiğinin hesabı önemlidir. Çapraz ilinti ve İGA tabanlı yöntemler için, gerekli eğri sayısını bulmaya yönelik istatistiksel modeller oluşturulmuştur. Geliştirilen modellerin teorik ve pratik sonuçları karşılaştırılmıştır. Eğri sayılarının hesaplanmasında temel olarak bir, normal dağılıma sahip bir değişkene ait örnek ortalamasının bulunacağı güven aralığı kavramından faydalanılmıştır. Çapraz ilinti tabanlı ve ilintilerin ortalamasını kullanmaya dayanan yaklaşım, aslında toplamsal özelliğe sahip olmayan ilinti değerlerinin ortalamasını kullanması açısından sorunlu görünmektedir. Ancak birden fazla ilinti değerlerinden gerçek ilinti değerinin daha iyi bir kestirimini elde etmek için, özellikle yüksek serbestlik derecelerinde, ilinti değerlerinin doğrudan ya da Fischer-Z dönüşümlerine ait ortalamalarının kullanılabileceği literatüre geçmiş çalışmalarda savunulmaktadır. Ayrıca yüksek serbestlik derecelerinde, ilinti değerlerine ait örneklerin normal dağılıma yaklaşması nedeni ile hesaplamalarda, Fisher-Z dönüşüm tabloları kullanılabilmektedir. Her iki yaklaşıma göre de yapılan hesaplamalarda, dönüşüm uygulanmayan yöntemin gerçek uygulama ile daha uyumlu sonuçlar verdiği gözlenmiştir. Bununla beraber, ilinti

değerlerini önce Z- değişkenlerine dönüştürerek ve mevcut eğrileri de parça parça kullanmaya dayanan yeni hesaplama yöntemlerinin de geliştirilebileceği düşünülmektedir. Böyle bir çalışma da yine gelecek dönem için planlanmaktadır.

İGA tabanlı yöntem de, gerekli eğri sayısının hesaplanması için, ilinti değerlerinin serbestlik derecesine göre yer alacağı güven aralığını hesaplamaya dayanan yaklaşımlar geliştirilmiştir. Bu yönteme ait teorik ve deneysel sonuçlarını birbiri ile oldukça uyumlu olduğu gözlenmiştir.

KAYNAKLAR

[1] Kerckhofes A., La cryptographic militaire, *Journal des sciences militaires*, 1883, **5**(38).

[2] Kocher P. C., Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems, *Annual International Cryptology Conference*, California, USA, 18-22 August 1996.

[3] Koecher P., Joshua J., Benjamin J., 1998 Introduction to differential power analysis and related attacks, https://www.rambus.com/wp-content/uploads/2015/08/DPATechInfo.pdf, (Ziyaret tarih: 1 Şubat 2021).

[4] Messerges T. S., Dabbish E. A.; Sloan, R., H., Investigations of Power Analysis Attacks on Smartcards, *USENIX Workshop on Smartcard Technology*, Chicago, Illinois, USA, 10–11 May 1999.

[5] Messerges T. S., Dabbish E. A., Sloan R. H., Power analysis attacks of modular exponentiation in smartcards, *International Workshop on Cryptographic Hardware and Embedded Systems*, Worcester, MA, USA, 12–13 August 1999.

[6] Brier E., Clavier C., Olivier F., Correlation power analysis with a leakage model, *International workshop on cryptographic hardware and embedded*, MA, USA, 11-13 August 2004.

[7] Amiel F., Feix B., & Villegas K., Power analysis for secret recovering and reverse engineering of public key algorithms, *International Workshop on Selected Areas in Cryptography*, Ottawa, Canada, 16-17 August 2007.

[8] Van Eck W., Electromagnetic radiation from video display units: An eavesdropping risk?, *Computers & Security*, 1985, **4**(4), 269-286.

[9] Hatun E., Kaya G., Buyukkaya E., & Yalcin, B. O., Side Channel Analysis Using EM Radiation of RSA Algorithm Implemented on Raspberry Pi, *International Symposium on Networks, Computers and Communications*, 18-20 June 2019, Istanbul, Turkey.

[10] Genkin D., Shamir A., Tromer E., RSA key extraction via low-bandwidth acoustic cryptanalysis, *Annual Cryptology Conference*, 17-21 August 2014; Santa Barbara, CA, USA.

[11] Boneh D., Demillo R. A., Lipton R. J., On the importance of checking cryptographic protocols for faults. *International conference on the theory and*

applications of cryptographic techniques, Konstanz, Germany, 11-15 May 1997.

[12] Biham E., Shamir A., Differential fault analysis of secret key cryptosystems, *Annual international cryptology conference*, Santa Barbara, California, USA, 17-21 August 1997.

[13] Rivest R. L., Shamir A., Adleman L., A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, 1983, **26**(1), 96-99.

[14] FIPS PUB. 140-3, Security Requirements for Cryptographic Module, *National Institute of Standards and Technology*, Gaithersburg, 2019.

[15] ISO/IEC 19790, Information technology -Security techniques - Security requirements for cryptographic modules, *International Electrotechnical Commission*, Berlin, 2012.

[16] Kuzu E. A., Soysal B., Şahinoğlu M., Güvenç U., & Tangel A. (2013, February). New cross correlation attack methods on the montgomery ladder implementation of RSA, *IEEE 3rd International Advance Computing Conference*, Ghaziabad, India, 22-23 February 2013.

[17] Kuzu E. A., Tangel A., All bits cross correlation attack on the Montgomery Ladder implementation of RSA, *International Conference on Digital Signal Processing*, Santorini, Greece, 1-3 July 2013.

[18] Kuzu E. A., Tangel A., A new style CPA attack on the ML implementation of RSA, *International Computer Science and Engineering Conference*, (pp. 323-328), Khon Kaen, Thailand, 30 July -1 August 2014.

[19] Kuzu E. A., Tangel A., *Correlation template matching CPA method, Electronics Letters*, 2016, **52**(15), 1306-1308.

[20] Kang S. M., Leblebici Y., *CMOS digital integrated circuits*, Tata McGraw-Hill Education, 4. ed., New York, 2003.

[21] Mangard S., Oswald E., Popp T. (2008), Power analysis attacks: Revealing the secrets of smart cards, *Springer Science & Business Media*, 1. ed., New York City.

[22] Danis A. U., Ors B., Differential power analysis attack considering decoupling capacitance effect, *IEEE European Conference on Circuit Theory and Design*, Antalya, Turkey, 23-27 August 2009.

[23] Sun S., Zijun Y., Zambreno J., Experiments in attacking FPGABased embedded systems using differential power analysis, *IEEE International Conference on Electro/InformationTechnology*, Ames, Iowa, USA, 18-20 May 2008.

[24] Fisher R. A., Statistical Methods for Research Workers (1958), *Oliver & Boyd*, 13. ed., Edinburgh London.

[25] Walter C., Sliding Windows Succumbs to Big Mac Attack, *Cryptographic Hardware and Embedded Systems (CHES)*, Paris, France, May 14-16, 2001.

[26] Kilman W., Lange T., Lochter M., Thumser W., Wicke G., Minimum requirements for evaluating side-channel attack resistance of elliptic curve implementations, *BSI, Federal Office for Information Security* , 1.0.4 01.07.11, 2016.

[27] Yen S. M., Ko L. C., Moon S., Ha J., Relative doubling attack against montgomery ladder, *International Conference on Information Security and Cryptology* , Seoul, Korea, 1-2 December 2005.

[28] Clavier C., Feix B., Gagnerot G., Roussellet M., & Verneuil, V., Horizontal correlation analysis on exponentiation, *International Conference on Information and Communications Security*, Copenhagen, Denmark, 24-26 August 2019.

[29] Kim H., Kim T. H., Yoon J. C., Hong S., Practical Second-Order Correlation Power Analysis on the Message Blinding Method and Its Novel Countermeasure for RSA, *ETRI journal*, 2010, **32**(1), 102-111.

[30] Witteman M. F., van Woudenberg J. G., Menarini F., Defeating RSA multiply-always and message blinding countermeasures, *Cryptographers' Track at the RSA Conference*, San Francisco, CA, USA, 14-18 February 2011.

[31] Wan W., Yang W., Chen J., An optimized cross correlation power attack of message blinding exponentiation algorithms, *China Communications*, 2015, **12**(6), 22-32.

[32] Fouque P. A., Valette F., The doubling attack–why upwards is better than downwards, *International Workshop on Cryptographic Hardware and Embedded Systems,* Cologne, Germany, 8-10 September 2003.

[33] Le T. H., Canovas C., Clédiere J., An overview of side channel analysis attacks, *ACM symposium on Information, computer and communications security*, 27-31 October 2008.

[34] Chari S., Rao J. R., Rohatgi P., Template attacks, *International Workshop on Cryptographic Hardware and Embedded Systems*, CA, USA, 13-15 August 2002.

[35] Oswald E., Mangard S., Template attacks on masking—resistance is futile, *Cryptographers' Track at the RSA Conference*, San Fancisco, CA, USA, 5-9 February 2007.

[36] Wan W., Chen J., Zhang S., Xia J., A cluster correlation power analysis against double blinding exponentiation, *Journal of Information Security and Applications*, 2019, **48** (102357).

[37] Weissbart L., Chmielewski Ł., Picek S., Batina L., Systematic Side-Channel Analysis of Curve25519 with Machine Learning, *Journal of Hardware and Systems Security*, 2020, **4**(4), 314-328

[38] Mateos E., Gebotys C. H., A new correlation frequency analysis of the side channel, *5th Workshop on Embedded Systems Security*, Scottsdale, AZ, USA, 24 October 2010.

[39] Tiran S., Ordas S., Teglia Y., Agoyan M., Maurine P., A Frequency Leakage Model and its application to CPA and DPA, *Journal of Cryptographic Engineering, Springer,* 2014, **4** (3), pp.197-212

[40] Bohl E., Hayek J., Schimmel O., Duplys P., Rosenstiel W., Correlation power analysisin frequency-domain, *Constructive Side-Channel Analysis and Secure Design*, Darmstadt, Germany, 2 April 2010.

[41] Gebotys C. H., Ho S., Tiu C. C., EM analysis of rijndael and ECC on a wireless java-based PDA, *International Workshop on Cryptographic Hardware and Embedded Systems,* Edinburgh, UK, 29 August-1 September 2005.

[42] Dehbaoui A., Tiran S., Maurine P., Standaert F. X., Veyrat-Charvillon N., Spectral Coherence Analysis-First Experimental Results, *IACR Cryptol. ePrint Arch.*, 2011, **2011**(56)..

[43] Tiran S., Maurine P., SCA with magnitude squared coherence, *International Conference on Smart Card Research and Advanced Appl*ications, Graz, Austria, 28-30 November 2012.

[44] Storey B., Franklin W. Olin, Computing Fourier Series and Power Spectrum with MATLAB. Retrived, College of Engineering: http://faculty.olin.edu/bstorey/Notes/Fourier.pdf, (Ziyaret tarihi: 10 Şubat 2020).

[45] Barenghi A., Pelosi G., Teglia Y., Improving first order differential power attacks through digital signal processing, I*nternational conference on Security of information and networks*, Taganrog, Russian Federation, 7-11 September 2010.

[46] Barenghi A., Pelosi G., Teglia Y., Information leakage discovery techniques to enhance secure chip design. *International Workshop on Information Security Theory and Practices*, Crete, Greece, 1-3 June, 2011.

[47] Diffie W., Hellman M.E.: New Directions in cryptography. *IEEE Transactions on Information Theory*, 1976, **22**(6), 644–654.

[48] FIPS PUB 186-3. Digital Signature Standard, *National Institute of Standards and Technology*, Gaithersburg, October 2009.

[49] Coron J. S., Resistance against Differential Power Analysis for Elliptic Curve Cryptosystems, *Cryptographic Hardware and Embedded Systems*, MA, USA, 12-13 August 1999.

[50] Yen S. M., Kim S. J., Lim S. G., Moon S. J., A Countermeasure Against One Physical Cryptanalysis May Benefit Another Attack, *Information Security and Cryptology*, Seoul, Korea, 28-29 November 2002.

[51] Joye M., Yen S. M., "The Montgomery Powering Ladder, *Cryptographic Hardware and Embedded Systems*, Cologne, Germany, 8-10 September 2003.

[52] Montgomery P. L., "Montgomery. Speeding the Pollard and elliptic curve methods of factorization." *Mathematics of Computation*, January 1987, **48**(177), 243-264.

[53] Izu T., Takagi T., A Fast Parallel Elliptic Curve Multiplication Resistant against Side Channel Attacks, *Advances in Cryptology - ASIACRYPT'98*, Beijing, China, 13 November 1998.

[54] Fouque P.-A., Valette, F.: The Doubling Attack - why upwards is better than downwards, *Cryptographic Hardware and Embedded Systems*, Cologne, Germany, 8-10 September 2003.

[55] Yen S. M., Lien W. C., Moon, S., Ha J., Power Analysis by Exploiting Chosen Message and Internal Collisions - Vulnerability of Checking Mechanism for RSA-Decryption, *International Conference on Cryptology in Malaysia*, Kuala Lumpur, Malaysia, 28-30 September 2005.

[56] Itoh K., Izu T., Takenaka M., A Practical Countermeasure against Address-Bit Differential Power Analysis, *Cryptographic Hardware and Embedded Systems*, Cologne, Germany, 8-10 September 2003.

[57] Itoh K., Izu, T., Takenaka, M., Address-Bit Differential Power Analysis of Cryptographic Schemes OK-ECDH and OK-ECDSA, *Cryptographic Hardware and Embedded Systems*, CA, USA, 13-15 August 2002.

[58] Bauer A., Jaulmes É., Prouff E., Reinhard, J. R., Wild, J., Horizontal collision correlation attack on elliptic curves, *Cryptography and Communications*, **7**(1), 91-119.

[59] Wang H., Guo W., Wei J., Practical chosen-message CPA attack on message blinding exponentiation algorithm and its efficient countermeasure, 2018, *World Wide Web*, **21**(1), 201-217.

[60] Vasantapan N., Chouvatut V., Pattern extraction from northern Thai fabrics using flexibly matching segments, Sarong Teenjok and Lanna textiles,

International Conference on Knowledge and Smart Technology, Chon Buri, Thailand, 1-4 February 2017.

[61] Heriana O., Praludi T., Wael C. B. A., Binary Template Matching for Morphological Dilation Enhancement in Navigation Radar Imaging, *Jurnal Elektronika dan Telekomunikasi*, 2018, **18**(2), 60-66.

[62] Giraud C., An RSA implementation resistant to fault attacks and to simple power analysis, *IEEE Transactions on computers*, 2006, **55**(9), 1116-1120.

[63] Ciet M., Joye M., Virtually free randomization techniques for elliptic curve cryptography, *International Conference on Information and Communications Security*, Huhehaote, China, 10-13 October 2003

[64] Clavier C., Joye M., Universal exponentiation algorithm a first step towards provable SPA-resistance, *International Workshop on Cryptographic Hardware and Embedded Systems*, Paris, France, 14-16 May 2001.

[65] Mahanta H. J., Khan A. K., Comparative modular exponentiation with randomized exponent to resist power analysis attacks, *Arabian Journal for Science and Engineering*, 2017, **42**(8), 3423-3434.

[66] Mahanta H. J., Khan A. K., Mukhopadhyay S., Modular exponentiation with inner product to resist higher-order DPA attacks, *Innovations in Systems and Software Engineering*, 2020, **16**(1), 87-97.

[67] Barman M., Mahanta H. J., A randomised scheme for secured modular exponentiation against power analysis attacks, *Cyber-Physical Systems*, 2019, **5**(4), 209-230.

[68] Hong T., Ju T., Li Y., Address Collision Attacks on ECSM Protected by ADPA, *17th International Computer Conference on Wavelet Active Media Technology and Information Processing*, Chengdu, China, 29 Jan 2021.

[69] http://www.risec.aist.go/project/sasebo/download/sasebo_gii_materials.zip, (Ziyaret tarih: 1 Şubat 2019).

[70] Montgomery P. L., Modular multiplication without trial division, *Mathematics of Computation*, 1985, Vol. **44**(170), pp. 519-521.

[71] Brier E., Clavier C. ; Olivier F., "Optimal Statistical Power Analysis", Cryptology ePrint Archive: Report 2003/152.

[72] Pandini D., Repetto G. A., Sinisi V., Clock distribution techniquesfor low-EMI design. In: Azemard, N., Svensson, L.J. (eds.) *PATMOS.Lecture notes in computer science*, 2007, **4644**, pp. 201–210.

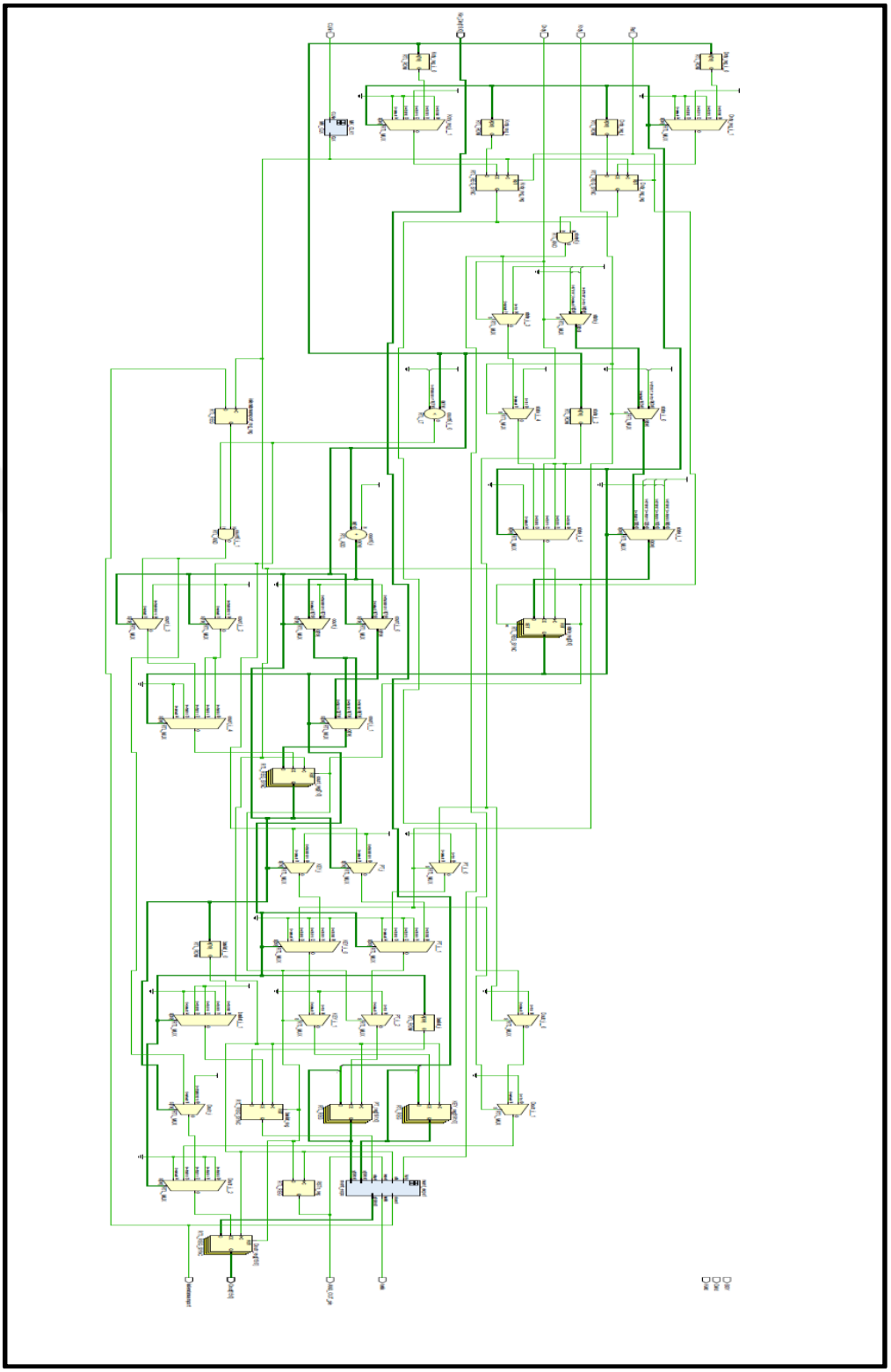
[73] Donner A., Rosner B., On inferences concerning a common correlation coefficient, *Journal of the Royal Statistical Society*, 1980, **29**(1), 69-76.

[74] Gorsuch R. L., Lehmann C. S., Correlation coefficients: Mean bias and confidence interval distortions, *Journal of Methods and Measurement in the Social Sciences*, 2010, **1**(2), 52-65.

[75] Strube M. J., Averaging correlation coefficients: Influence of heterogeneity and set size, *Journal of Applied Psychology*, 1988, **73**(3), 559.

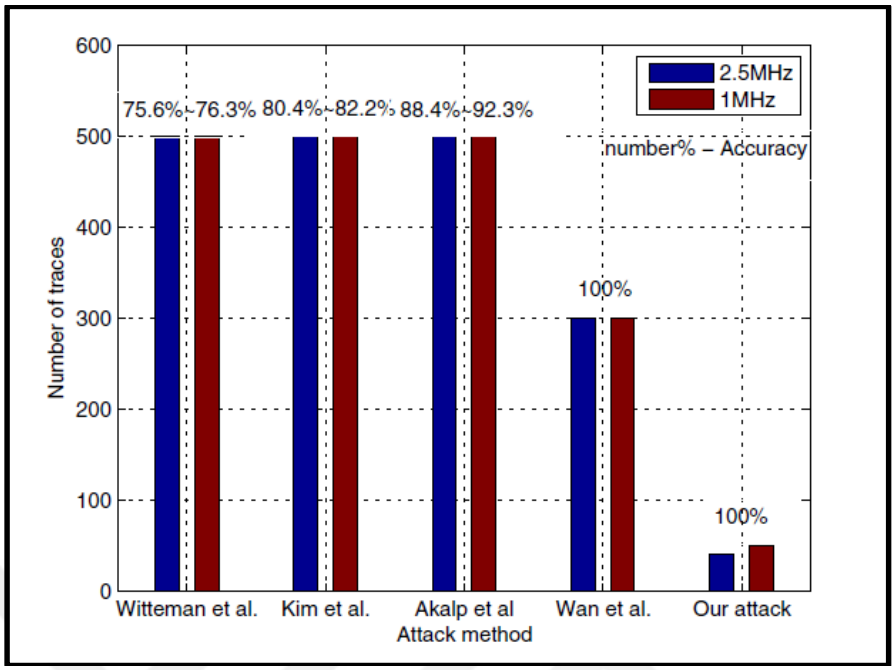
EKLER

EK-A



Şekil A.1. İkilik üs alma devresi RTL şeması

EK-B



[59] Practical chosen-message CPA attack on message blinding exponentiation algorithm and its efficient countermeasure

Şekil B.1. Literatürdeki çapraz ilinti tabanlı çakışma analizi yöntemlerinin karşılaştırılması

KİŞİSEL YAYINLAR VE ESERLER

Kuzu E. A., Soysal B., Şahinoğlu M., Güvenç U., Tangel A., New cross correlation attack methods on the montgomery ladder implementation of RSA, IEEE 3rd International Advance Computing Conference, Ghaziabad, India, February 22-23 2013.

Kuzu E. A., Tangel A., All bits cross correlation attack on the Montgomery Ladder implementation of RSA, IEEE 18th International Conference on Digital Signal Processing, Santorini, Greece, July 1-3 2013.

Kuzu E. A., Tangel A., A new style CPA attack on the ML implementation of RSA, International Computer Science and Engineering Conference, Khon Kaen, Thailand, July 30 - August 1 2014.

Kuzu E. A., Tangel A., Correlation template matching CPA method, Electronics Letters, 2016, **52**(15), 1306-1308.

.

ÖZGEÇMİŞ

Ebru Akalp Kuzu, 2001 yılında İTÜ Elektronik ve Haberleşme Mühendisliğinden lisans diploması, 2006 yılında ise aynı üniversitenin Elektronik ve Hab Müh. Biyomedikal programından yüksek lisans diploması alarak mezun olmuştur. 2010-2021 yılları arasında Kocaeli Üniveristesi Elektronk ve Hab. Müh.'de doktora eğitimine devam etmiştir. Halen TÜBİTAK BİlGEM'de araştırmacı olarak çalışmaktadır. Evli ve iki çocuk annesidir.