Gelişmiş Şifreleme Standardının - AES - FPGA Üzerinde Gerçeklenmesi

K. V. Dalmışlı, B. Örs

İstanbul Teknik Üniversitesi
Elektrik-Elektronik Fakültesi
Maslak, Istanbul
dalmisli@itu.edu.tr
Siddika.Ors@itu.edu.tr

Özet

Gelişmiş Şifreleme Standardı (AES) yayınlandığından beri güvenilirliğini korumuş ve bilişim teknolojisinde güvenliği sağlamak için kullanılagelmiştir. Günümüzde teknoloji ürünlerinin boyutları küçülmekte, hızları ise artmaktadır. Bu sebeple AES tüm devrelerinin hem çok küçük alan kaplamaları, hem de yeterli hıza sahip olmaları gerekmektedir. Bu makalede 8-bit mimarisinde, alan olarak küçük ve hız olarak yeterli olabilecek, ortak kullanılan bir s-kutusu ve sütun karıştırma modülü sayesinde 680 saat işaretinde şifreleme işlemleri yapabilen ve alan olarak da 7500 eşdeğer kapı (Gate equivalent-GE) değerinde olan AES tüm devresi tasarımlarını sunuyoruz.

Abstract

Advanced Encryption Standard (AES) maintains safety and used for providing security to tech products since publish date. At the present day, tech devices are produced smaller and faster. So, AES chips must not only use very small area, but also be enough fast on chip. In this paper, we present an 8-bit structural AES cipher which encrypts plaintext in 680 clock cycles and lays on 7500 GE on FPGA by using only one S-box and a quarter mix column modules. This can be accepted as enough performance for very small are requirements.

1. Giriş

Kriptografik algoritmalar, yazılımsal ve donanımsal olarak gerçeklenebilmektedir. Yazılımsal yapılan gerçeklemeler daha ucuza mal olmakta ve daha esnek bir yapıya sahip olmaktayken, donanımsal gerçeklemeler daha hızlı çalışmakta ve daha yüksek güvenlik sağlamaktadır. Bu ikilemi aşmak için yazılım-donanım paylaşımlı sistemler de kullanılmaktadır. Özellikle donanım uygulamalarında, alan kullanımını azaltmayı ve hızı arttırmayı sağlayacak gerçeklemeler yapmaya yönelik çalışmalar yaygın olarak sürdürülmektedir. Gelişmiş Şifreleme Standardı (Advanced Encryption Standard (AES)) Kasım 2001'de elektronik verinin şifrelenmesi için kullanılmak üzere federal bilgi işleme standardı (Federal Information Processing Standards (FIPS)) [2] olarak Amerikan Ulusal Standartlar ve Teknoloji Enstitüsü (National Institute of Standards and Technology (NIST)) tarafından yayınlanmıştır. AES algoritması bilginin şifrelenmesi ve şifrelenmiş bilginin çözülmesi için kullanılan bir simetrik blok kodlayıcıdır. Bu çalışmada AES algoritmasının alan olarak efektif bir gerçeklemesi olan Küçük AES tüm devresinin tasarımı yapılmıştır. AES algoritmasının ana operasyonları FIPS-197'de gösterildiği gibi [2] Bayt Değiştirme, Satır Kaydırma, Sütun Karıştırma, Anahtar Toplama ve Anahtar Üretme işlemlerinden oluşur. Bu işlemlerin her biri bir modül olarak tasarımımızda yer almaktadır.

2. AES Algoritması

AES algoritmasının 128, 192 ve 256 bitlik versiyonları mevcuttur. Bizim çalışmamızda 128 bitlik AES Algoritması için 128 bit anahtar ve 10 tur sırasıyla, Bayt-değiştirme, Satırları Kaydırma, Sütunları Karıştırma, Tur Anahtarının Toplanması işlemleri yapılır.

2.1. Bayt Değiştirme

Bayt değiştirme, algoritma içerisindeki tek doğrusal olmayan dönüşüm işlemidir. Bayt değiştirme dönüşümü, girişindeki durumun her bir baytını, S-kutusu (S-box) adı verilen bir değiştirme tablosu kullanarak, başka bir bayta dönüştürür. Değiştirme tablosu, S-Kutusu, iki farklı dönüşüm işleminin birleşiminden oluşmaktadır:

1. Giriş baytı $GF\left(2^8\right)$'de bir polinom olarak ele alınır ve matematiksel ters elemanına dönüştürülür. Hex(00) elemanı yine kendisine atanır.

2. Birinci dönüşüm sonucunda elde edilen değer, İlgin dönüşümünden geçirilir.

b_i ilgin dönüşüm girişindeki bayt'ın i'inci biti ve $\bar{b_i}$ de dönüşüm sonrasında elde edilen bayt'ın i'inci biti olmak üzere, ilgin Dönüşümü aşağıdaki gibi matemetiksel

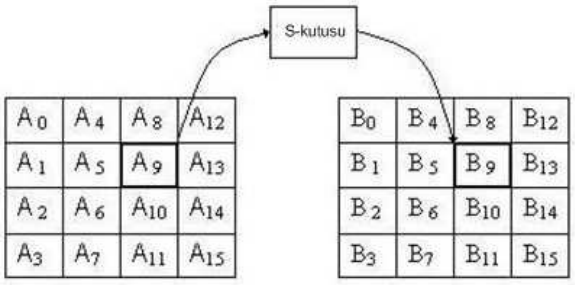

Figure 1: Bayt Değiştirme İşlemi

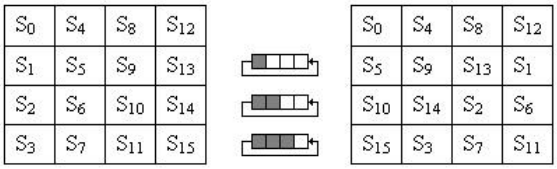

Figure 2: Satır Kaydırma İşlemi

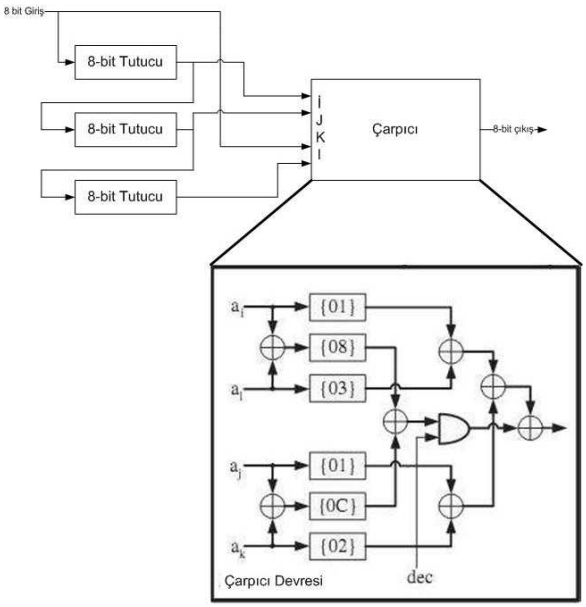

Figure 3: $1/4$ Sütun Karıştırma Devresi

olarak ifade edilebilir:

$$\bar{b_i} = b_i \oplus b_{(i+4)mod8} \oplus b_{(i+5)mod8} \oplus b_{(i+6)mod8} \oplus b_{(i+7)mod8} \oplus c_i \quad (1)$$

2.2. Satırları Kaydırma

Şifreleme için, ilk satır aynı bırakılır. İkinci satır sağdan sola doğru bir pozisyon, değiştirecek şekilde, döngüsel olarak kaydırılır. Döngüsel kaydırma nedeniyle, 1. sütuna gelen eleman kaydırıldığında 4.sütuna geçer. Üçüncü satır benzer şekilde iki pozisyon, dördüncü satır da üç pozisyon döngüsel olarak kaydırılır.

2.3. Sütun Karıştırma

Sütun Karıştırma işlemi 4x4 lük bir matriste her bir sütun kendi başına ele alınarak yapılır. Her bir sütun $GF(2^8)$ elemanları olan 3. dereceden birer polinom olarak alınır. Elde edilen bu polinomlar sabit bir polinomla, $(x^4 + 1)$'e göre modülo alınarak, çarpılarak sütunları karıştırma dönüşümü gerçekleştirilir. Şifreleme işlemi için kullanılan sabit polinom $c(x)$'in ifadesi

$$c(x) = 03 \bullet x^3 + 01 \bullet x^2 + 01 \bullet x + 02 \quad (2)$$

şeklindedir.

Şifre çözme işleminde ise alınacak polinom ise

$$c(x) = 0B \bullet x^3 + 0D \bullet x^2 + 09 \bullet x + 0E \quad (3)$$

şeklindedir.

2.4. Tur Anahtarını Toplama

Bu dönüşümde, durum verisinin baytları, tur anahtarının karşı düşen baytlarıyla karşılıklı x-or işlemine sokulur. Turun son adımı olan Tur Anahtarını Toplama işlemi $GF(2)$ üzerinde doğrusal bir işlemdir. Şifreleme ve çözme işlemleri için bu modül tamamen aynıdır.

2.5. Tur Anahtarı Üreteci

AES algoritmasında her tur için bir önceki turdaki anahtardan yeni bir tur anahtarı üretilir. 128 bit AES versiyonunda bunun için bir önceki anahtar $4x4$ boyutunda birer matrise dizilir. Daha sonra aşağıdaki algoritma uygulanarak her bir tur için anahtar üretilir. Tur anahtarı üretmek için uygulanan algoritma aşağıda gösterilmiştir.

Algorithm 1 Tur Anahtarı Üretme Algoritması

Require: N_k: Anahtardaki 32 bitlik kelime sayısı,
$w(i)$: Sütun numarası,
$RCon(i) \in GF(2^8)$,
$RCon(1) = x^0 = 01$, $RCon(2) = x^1 = 02$ ve $RCon(i) = x \bullet (RCon(i-1))$

Ensure:

```
for i from 0 to N_k − 1 do
    w(i) = Key((32 × i) − 1 : 0)
end for
for i from N_k to (N_b × (N_r + 1) − 1) do
    if i = 0 mod N_k then
        w(i) = w(i − N_k) ⊕ (S − Box(Kaydır(w(i − 1)))) ⊕ Rcon(i/N_k)
    else
        w(i) = w(i − 1) ⊕ w(i − N_k)
    end if
end for
```

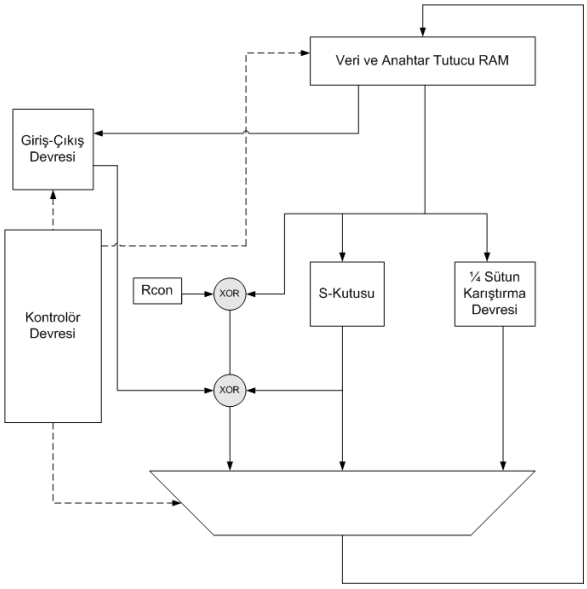

Figure 4: 8-bitlik Küçük AES-I devre şeması

3. Küçük AES I

AES gerçeklemeleri çok çeşitli tasarımlarla yapılabilir. 128 bitlik bir veriyi şifreleme işlemi bu veriyi 128 bitlik yapıdaki ve AES içerisindeki işlemleri ardı ardına yapacak bir şekilde tasarlanan gerçeklemede bir saat işaretinde yapılabilir. Bunun için paralel çalışan 16 adet 8-bit S-kutusu, 4 tane 32-bit Sütun Karıştırma, bağlama kombinasyonlarıyla Satır Kaydırma, Xor kapıları ile de tur anahtarını toplama modülü gereklidir. Bunun yanı sıra anahtar üretme işlemi için de yine S-kutusu, Rcon ve bayt döndürme işlemlerini yapacak modüllere ihtiyaç vardır. Görüldüğü gibi oldukça hızlı çalışan ama çok büyük bir alan kaplayan ve güç harcayan bir işlemci ortaya çıkmaktadır.

İhtiyaca göre çeşitlenen gerçeklemeler arasından tasarımımızda olabildiğince küçük bir tasarım yapmaya çalışıldı. Bunun için şifrelemede daha fazla saat işareti harcayan ancak alan olarak küçük bir tasarım yapmak için tur anahtarlarının ve şifrelenmiş tur verilerinin tutulduğu 2 adet 8x16 bitlik RAM, 1 adet ortak kullanılan S-kutusu, 1 adet $1/4$ Sütun Karıştırma modülü, 1 adet Rcon modülü, giriş ve çıkışları ayarlamak için 1 adet giriş-çıkış modülü ve tüm işlemleri sırayla yaptırmak için tüm modülleri kontrol eden bir adet kontrolör modülü kullanıldı [1].

Şekil 4 de görüldüğü gibi devre 4 ana modülden oluşmuştur.

RAM Blokları

2 adet 16x8 bitlik RAM kullanılmıştır. Tek giriş, tek çıkış olarak ve yazma/okuma adreslemelerine sahiptir.

Giriş-Çıkış Modülü

İşlemcinin ana girişi ve çıkışına bağlı olan, aynı zamanda işlemler sırasında RAM'lerden veri alıp, veri yoluna ileten bir modüldür. Kontrolör modülü tarafından gelen sinyaller ile yönlendirilir.

Veri yolu Modülü

İçerisinde S-kutusu, Sütun Karıştırma, Rcon modüllerini ve Xor, Mux gibi devre elemanları kullanılmıştır. S-kutusu modülü alandan tasarruf etmek için Rom kullanmak yerine matematiksel olarak gerçeklenmiştir [3]. S-kutusu modülü hem şifreleme hem de çözme için tasarlanmıştır. Sütun karıştırma modülü ise 8 bitlik tek giriş ve 8 bitlik tek çıkış olmak üzere şifreleme/çözme için tasarlanmıştır. $GF(2^8)$ de çarpma işlemleri 32 bitlik tam modüllerde çok yer kaplamaktadır. Ancak [1] de gösterildiği üzere, ters Sütun karıştırma işlemi için küçük bir sonlu alan matematiksel işlemiyle çarpma işlemleri iki adet azaltılmıştır. Bu sayede çeyrek sütun karıştırma modülü yeterli olmaktadır.

$$c(x) = 03x^3 + 01x^2 + 01x + 02 \quad (4)$$

$$c^{-1}(x) = 0bx^3 + 0dx^2 + 09x + 0e \quad (5)$$

$$c^{-1}(x) - c(x) = 08x^3 + 0cx^2 + 08x + 0c \quad (6)$$

$$c^{-1}(x) - c(x) = 08x(3 + x) + 0c(x^2 + 1) \quad (7)$$

$$c^{-1}(x) = c(x) + 08x(3 + x) + 0c(x^2 + 1) \quad (8)$$

$GF\left(2^8\right)$ de çarpma işlemi modülü büyük çarpma bloklarına gerek duyulmadan efektif olarak sadece AND kapıları ve giriş çıkış yer değiştirme kombinasyonları ile tasarlanmıştır [4]. Ayrıca büyük çarpıcı modülüne veri göndermek için girişten gelen verileri tutan 8 bitlik 3 tutucu kullanılır. Bunlara gelen veriler uygun bir sırada RAM'den çekilmek zorundadır.

Anahtar üretme modülü ise bir Rcon modülü, xor kapıları ve muxlardan oluşur. Bayt döndürme işlemi RAMden uygun sırayla bayt çekilmesi ile anahtar bayt değiştirme işlemi ortak kullanılan S-kutusu ile gerçeklenir.

Kontrolör Modülü

Tüm modülleri uygun AES operasyonunu yapmak üzere kontrol eder. İçinde bir sonlu durum makinesi ile tüm durumlar gerçeklenir. Bu durumlara göre modüllere izin bayrakları gönderilir. Ortak kullanılan modüller bu bayrakların yönlendirmesiyle sırayla çalışır. Ayrıca kontrol modülünde adres işaretçileri ile RAM'ların giriş çıkışları da kontrol edilir.

Devrenin Çalışması

Devreye ilk önce 16 saat işareti boyunca birer bayt halinde anahtar paketleri verilir. Bu paketler Giriş-çıkış modülünden yönlendirilerek anahtarın tutulduğu RAM'a yazılır. Daha sonra 16 saat işareti boyunca birer bayt halinde şifrelenecek veri paketleri verilir. Bu paketler verinin tutulduğu RAM'a yazılmadan önce anahtar RAM'ından gelen anahtarlarla birer-birer xor işlemine tabi tutulur ve daha sonra veri RAM'ına yazılır. Daha sonra kontrol modülünde 4 bitlik sayaç saydıkça RAM'dan uygun adreslerdeki datalar çekilir ve

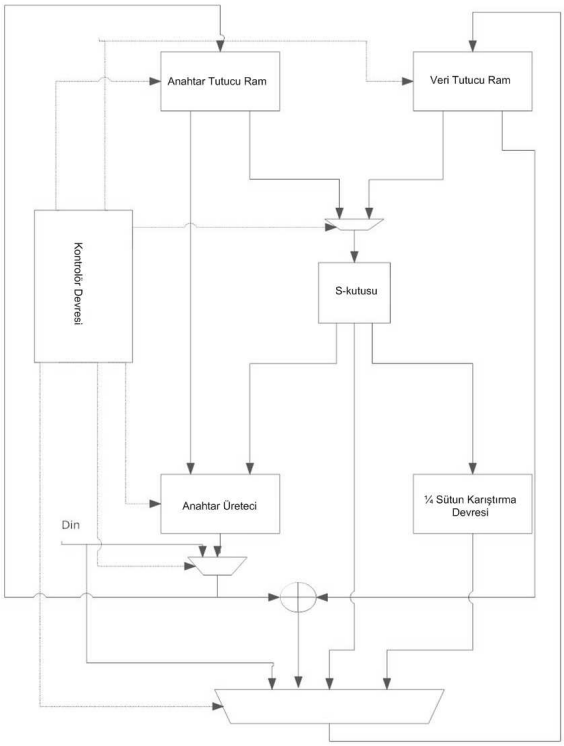

Figure 5: Küçük AES II devre şeması

sırasıyla Bayt-Değiştirme, Satırları Kaydırma, Sütunları Karıştırma işlemleri yapılır. Sonra anahtarı tutan RAM'dan uygun sırada veri okunarak anahtar üretme işlemi yapılır. En sonunda hem anahtar hem veri alınarak Xor işlemiyle tur anahtarı toplama işlemi tamamlanır.

3.1. Gerçekleme Sonuçları

10 tur sonunda bir şifreleme işlemi 1250 saat işaretinde yapılmaktadır. Sistemin maksimum çalışma frekansı yaklaşık 60 Mhz'dir. FPGA içinde kapladığı alan olarak ise toplam kapı eşdeğeri olarak 7604 GE olarak bulunmuştur. Devrenin maximum çalıştığı frekansta tahmini güç harcaması ise 267 mW olarak ölçülmüştür.

4. Küçük AES II

Bu tasarımda modüllerin çogunluğu aynı olarak kullanılmasına rağmen kontrol modülü farklı tasarlanmış olup giriş-çıkış modülü devreden çıkarılmıştır.

Bu tasarımın ilk versiyondan en önemli farkı RAM blokları birbirinden ayrılmış, S-kutusu ve sütun karıştırma modülleri ard arda çalışacak şekilde bağlanmıştır. Byte-Değiştirme, Satır Kaydırma ve Sütun Karıştırma işlemleri her byte için bir kerede yapılır. Ayrıca tur anahtarı üretilirken her tur anahtarı üretildiği anda onunla eşleşen adresteki veri ile xor işlemine sokularak tekrar verinin tutulduğu RAM adresine yazılır.

Table 1: Bazı AES gerçeklemeleri ile karşılaştırma

	Platform	Saat İşareti Sayısı	Saat Frekansı	Alan	Güç Tüketimi
Küçük AES I	FPGA (Virtex5)	1250	60 MHz	7604 GE	267 mW (60 Mhz)
Küçük AES II	FPGA (Virtex5)	680	50 MHz	7515 GE	267 mW (50 Mhz)
[1]	ASIC (0.35 um)	1032	80 MHz	3400 GE	4.5 mW (100 kHz)

Böylece bir tur için kullanılan saat işareti sayısı oldukça azalmış olur.

4.1. Gerçekleme Sonuçları

10 tur sonunda şifreleme işlemleri 680 saat işareti içinde yapılır. Sistemin maksimum çalışma frekansı yaklaşık 50 MHz dir. Alan olarak ise 7515 GE olarak ölçülmüştür. Devrenin maksimum çalıştığı frekansta tahmini güç harcaması ise 267 mW tır.

5. Karşılaştırma Sonuçları

Yapmış olduğumuz tasarımların ilkinde gerek veri işlenirken, gerek anahtar üretilirken tüm modüller aynı anda çalışmaktadırlar. Ayrıca tüm operasyonlar tek tek sırayla yapılmaktadır. İkinci tasarımımızda ise veri işleme işlemlerinin tümü (Bayt Değiştirme, Satır Kaydırma, Sütun Karıştırma) aynı anda yapılmaktadır. Bunun sonucunda diğerine oranla daha az bir saat işaretinde şifreleme tamamlanmaktadır. Tablo 1 de görüldüğü gibi bir ASIC tasarımla kıyas edildiklerinde oldukça iyi bir performansa sahip olduklarını söyleyebiliriz.

6. Sonuçlar

Alan açısından tasarruflu olan iki tasarım sunduk. Bunları yapmak için sadece birer adet S-kutusu, anahtar ve veri için RAM blokları ve çeyrek sütun karıştırma modülü kullanılmıştır. Alan limitli tasarım konusunda çalıştığımız bu tasarımın DPA ve EMA saldırılarına karşı güvenilirliği henüz ölçülmemiştir. Gelecekte bu tasarımların güvenlik testleri yapılacak ve koruma yöntemleri uygulanarak dayanıklılıkları arttırılmaya çalışılacaktır.

7. Kaynaklar

[1] M. Feldhofer, J. Wolkerstorfer, and V. Rijmen. AES implementation on a grain of sand. *IEE Proceedings - Information Security*, 152(1):13–20, October 2005.

[2] National Institute of Standards and Technology. FIPS 197: Advanced Encryption Standard, November 2001.

[3] L. Ordu. AES algoritmasının FPGA Üzerinde gerçeklenmesi ve yan kanal analizi saldırılarına karşı güçlendirilmesi. Master's thesis, Istanbul Technical University, Turkey, June 2006.

[4] G. N. Selimis, A. P. Fournaris, and O. Koufopavlou. Applying low power techniques in aes mixcolumn/invmixcolumn transformations. In *13th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pages 1089–1092. IEEE, 10-13 December 2006.