İSTANBUL TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ

GÜVENLİ RFID SİSTEMLERİ İÇİN BİR KİMLİK DOĞRULAMA PROTOKOLÜNÜN FPGA ÜZERİNDE GERÇEKLENMESİ

BİTİRME ÖDEVİ

GÖKHAN ULUTAŞ
040080318

Bölümü: Elektronik ve Haberleşme Mühendisliği Bölümü
Programı: Elektronik Mühendisliği

Danışmanı: Doç. Dr. Sıddıka Berna ÖRS YALÇIN

MAYIS 2013

ÖNSÖZ

Bitirme çalışmamın başından sonuna kadar tüm anlayışıyla bilgi ve desteğini paylaşan saygıdeğer hocam Doç. Dr. Sıddıka Berna Örs Yalçın'a çok teşekkür ederim.

Gömülü Sistem Tasarım Laboratuvarındaki arkadaşlarıma yardımlarından ötürü ayrıca teşekkür ederim.

Gökhan Ulutaş

Mayıs 2013

İÇİNDEKİLER

KISALTMALAR

RFID : Radio Frequency Identification

FPGA : Field Programmable Gate Array

TEA : Tiny Encryption Algorithm

ISE : Integrated Software Environment

EDK : Embedded Development Kit

SDK : Software Development Kit

SOF : Start of Frame

EOF : End of Frame

CRC : Cyclic Redundancy Check

IP : Intellectual Property

XPS : Xilinx Platform Studio

RTL : Register Transfer Level

SPI : Serial Peripheral Interface

ŞEKİL LİSTESİ

TABLO LİSTESİ

TEA İLE GÜVENLİ RFID SİSTEMLERİ İÇİN BİR KİMLİK DOĞRULAMA PROTOKOLÜNÜN FPGA ÜZERİNDE GERÇEKLENMESİ

ÖZET

RFID (radyo frekansı ile tanımlama) sistemlerinin kullanıldığı alanlar her geçen gün genişlemektedir. Bu yaygınlaşmanın artması ile bilgilerin güvenliğinin önemi artmaktadır. Ayrıca Sahada Programlanabilir Kapı Dizileri(FPGA-Field Programmable Gate Array)'nin tasarım ve programlama konusunda sağladığı avantajlar sayesinde geliştirilen bir sistemin denenmesi ve oluşturulması hızlanmaktadır. Bitirme projesi kapsamında bir güvenli RFID sistemi tam olarak temsil edecek, bir kimlik doğrulama protokolü FPGA üzerinde gerçeklenmiştir.

Protokolün nasıl gerçeklendiğini anlamak amacıyla öncelikle RFID sistemlerin fiziksel unsurları anlatılmıştır. Gerekli ön bilgiler oluşturulduktan sonra tasarım ortamlarından bahsedilmiştir. Bu ortamlar sayesinde oluşturulan donanım ve yazılım parçalarından ayrıca behsedilmiştir. Şifreleme, şifre çözme ve rastgele sayı üreteci donanımları Verilog HDL ile hazırlanmıştır. Şifreleme ve şifre çözme donanımları için Küçük Şifreleme Algoritması(Tiny Encryption Algorithm-TEA) kullanılmıştır. Rastgele sayı üreteci için Gömülü Sistem Tasarımı Labaratuvarı bünyesinde yapılan çalışmalarla tasarlanan özgün rastgele sayı üreteci kullanılmıştır. Donanımların test aşamalarını geçmesinin ardından protokolün yazılım aşamalarıyla devam edilmiştir. Protokol gerçeklenirken MicroBlaze işlemcisi ve C dili kullanılmıştır. Sistemin kablosuz haberleşmesini gerçekleştirmek üzere radyo frekansı alıcı-verici modülü C dili ile kontrol edilmiştir. Böylelikle, güvenli RFID sistemleri için bir kimlik doğrulama protokolü FPGA üzerinde küçük şifreleme algoritması ile gerçeklenmiştir.

IMPLEMENTATION OF AN AUTHENTICATION PROTOCOL FOR SECURE RFID SYSTEMS WITH TEA ON FPGA

SUMMARY

The usage areas of Radio Frequency Identification(RFID) systems are enlarging everyday. Security of informations' importance is raising by increasing widespread usage. Thanks to the FPGA's advantages at design and programming issues, generating and testing a design is accelerated. An authentication protocol was confirmed on FPGA to completely represent an secure RFID system within graduation project.

First of all, physical parts of RFID systems are described to understand how the protocol was implemented. After having essential foreknowledge, it was mentioned about design tools. It was also mentioned about hardware and software parts of design which is produced thanks to theese tools. Encryption, decryption and random number generator hardwares were prepared with Verilog HDL. Tiny Encryption Algorithm was used as hardware of encryption and decryption. An original random number generator was used for random number generator which is designed with studies in Embedded System Design Laboratory. After passing hardware test phases, it was continued with software phase of protocol. MicroBlaze processor and C programming language were used when protocol is confirmed. Radio Frequency Transceiver was controlled with C language to realize the wireless communication of the system. Hereby, the authentication protocol for secure RFID systems was implemented on FPGA with TEA.

1. GİRİŞ

RFID sistemleri genel olarak iki parçadan oluşur [1]. Bunlar okuyucu ve etiket olarak adlandırılır. Bu bitirme çalışmasında güvenli bir RFID protokolünde kullanılan okuyucu ve etiket yapıları FPGA üzerinde gerçeklenmiştir. Aslından ikisi de aynı dış görünüş ve parçalara sahip olmuştur. Okuyucu ve etiket yapılarının içine bakıldığında FPGA kullanıldığı için aynı donanımlar kullanıldığı görünmektedir. Gerçeklenen protokol gereğince sistem parçaları yazılım bölgesinde birbirinden ayrışmakta ve böylece güvenli RFID sistemleri için kimlik doğrulama protokolü FPGA üzerinde küçük şifreleme algoritması (Tiny Encryption Algorithm - TEA) kullanılarak gerçeklenmiştir [2].

RFID sistemleri gerçeklenirken ISO/IEC 18000 standardına uyulması gerekmektedir [2]. Bu standartda işlemlerin nasıl yapılacağı belirtilmektedir. Tüm RFID sistemlerinin uyumlu çalışabilmesi için bilginin hangi şablonda olacağı belirlenmiştir. Sistemin güvenliği ile ilgili bir standart bulunmamaktadır. Kullanacağımız kimlik doğrulama protokolü gereğince güvenliği artırmak amacıyla gönderilecek bilgiler TEA ile şifrelenerek etiketin doğru anahtara sahip olup olmadığı belirlendi [2]. Anahtarın doğruluğunu ölçmek için sabit bir bilgi göndermek yerine rastgele sayılar kullanıldı.

Kimlik doğrulama protokolü ise MicroBlaze işlemcisi ile kontrol edilmiştir [2]. İşlemcinin sağladığı avantajlarla Verilog donanım tanımlama dili (Hardware Description Language – HDL) ile gerçeklenmiş TEA algoritmasının gerçeklemeleri ve rastgele sayı üreteci kontrol edilmiştir. FPGA üzerinde bulunan seri çevresel birim ara yüzü (Serial Preperial Interface – SPI) giriş-çıkışları ve RF alıcı-vericiler kullanılarak sistemin kablosuz olarak uzun mesafelerde çalışması sağlanmıştır.

Tüm yapılanlar aşağıdaki gibi bölümlerde ayrıntılı bir şekilde nasıl yapıldığı, test ve gerçekleme aşamaları, tüm donanım-yazılım birleşimi anlatılmıştır. Bölümler ise şöyledir:

- *Ön Bilgiler* (**Bölüm 2**): RFID sistemler ile ilgili ayrıntılı bilginin verildiği ayrıca gerçeklenen kimlik doğrulama protokolünün anlatıldığı bölümdür.

- *Gerçekleme Ortamları* (**Bölüm 3**): Spartan-3E kartı ve onunla birlikte kullanıma sunulan geliştirme ortamlardan bazıları (ISE, EDK, SDK) incelenmiştir.

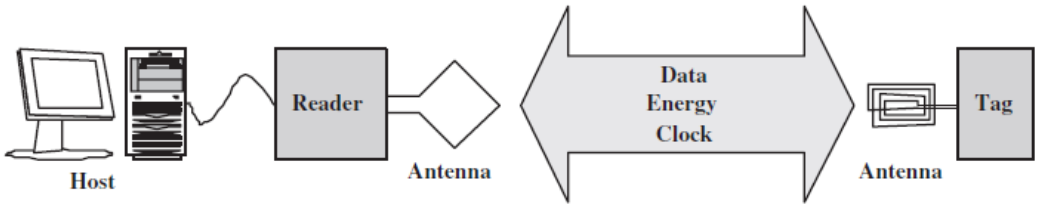
- *Donanım Gerçeklemeleri* (**Bölüm 4**): Rastgele sayı üreteci, TEA ile şifreleme ve şifre çözme donanımlarının Verilog HDL ile gerçeklemeleri ve test aşamaları anlatılmıştır.
- *Yazılım Gerçeklemeleri* (**Bölüm 5**): Rastgele sayı üreteci, TEA algoritmaları ve RF alıcı-verici yapılarının microblaze işlemcisine ve SDK ortamına eklenmesi ve yazılım ile kontrolü incelenmiştir.
- *Sistemin gerçeklenmesi* (**Bölüm 6**): Sisteme ait tüm parçaların, FPGA'lerin, rf alıcı-vericilerin, donanımların ve yazılımların birleştirilmesi ve çalıştırılması anlatılmıştır.

2. ÖN BİLGİLER

Martin Feldhofer'ın güvenli RFID sistemler için hazırlamış olduğu "An Authentication Protocol in a Security Layer for RFID Smart Tags" adlı kimlik doğrulama protokolü kullanılmıştır. Protokol okuyucu ile etiket arasındaki konuşmaları düzenlemekte ve nasıl olacağını belirlemektedir. Protokolde ayrıca güvenliği artırıcı özellikler bulunmaktadır. Bu ön bilgiler ile birlikte RFID sistem parçaları, özellikleri ve çalışmaları ek olarak bu bölümde anlatılmıştır.

2.1 Radyo Frekansı Tanımlama Sistemleri

Radyo Frekansı Tanımlama verinin etiketlerle RFID okuyuculara taşınmasında kullanılan otomatik kimliklendirme teknolojisine denir[1]. Etiket ve okuyucu olmak üzere iki temel yapı kullanarak oluşturulmuş bir sistemdir. Temel bir RFID sistemin şeması Şekil 2.1'de verilmiştir. Etiket ve okuyuculara ait ayrıntılı bilgiler bölümün ilerleyen kısımlarında anlatılmıştır.



Şekil 2.1 : RFID sistem yapısı[3]

Sistemin kablosuz haberleşmesi antenlerle sağlanmaktadır. Etiketin türüne göre etiketin enerjisi de okuyucudan gönderilebilmektedir. RFID sistemlerde veri iletimi farklı radyo frekansı bandlarında yapılabilmektedir. Frekans badlarına ait bir tablo aşağıda verilmiştir.

Frekans Bandı	Uygulamalar
433.5 - 434.5 MHz	Avrupa'da ISM bandı kullanılmaktadır. Japonya ve Kore'de uygulanmak üzere ayrılmıştır.
865 - 868 MHz	ETSI 302–208 düzenlemeleri sonucunda tanımlanan spektrum
869.4 - 869.65 MHz	Avrupa'da RFID ve diğer uygulamaları için lisans gerektirmeyen bandda ayrılmış 250 kHz lik spektrumdur.
902 - 928 MHz	Kuzey Amerika'da yayılı spektrum haberleşmesi için ayrılmış lisans gerektirmeyen banddır..
918 - 926 MHz	ERP 1 Watta kadar müsaade edilen Avustralya standardıdır.
950 -956 MHz	Japonya'da RFID uygulamaları için ayrılmıştır.
2.4 GHz (mikrodalga)	Dünyanın birçok ülkesinde lisans gerektirmeden yayılı spektrum uygulamaları için tanımlanmış olan banddır. Bluetooh ve WLAN uygulamaları için de kullanılmaktadır.(IEEE 802.11b ve 802.11g)

Tablo 2.1 : Dünya genelinde kullanılan RFID frekansları[4]

Bizim çalışmamızda kullanacağımız RF alıcı-verici modülleri ise 869.5MHz'de çalışmaktadır.

RFID kullanım alanları her geçen gün genişlemektedir. Yaygınlaşma sonucunda güvenlik eksikliklerini de beraberinde getirmektedir. Bu nedenle kimlik doğrulama, güvenlik ve gizlilik gerektiren RFID uygulamalarında önemli rol almaktadır[5]. Güvenliği ve gizliliği tam olarak sağlayacak kimliklendirme protokolleri oluşturmak için dünya genelinde çalışmalar yapılmaktadır. Bu çalışmalardan çoğu anonimleşme konusunda başarılı olamamakta ve zayıf kalmaktadır[6]. Bizim sistemimizde bu nedenle Martin Feldhofer'in yazmış olduğu kimlik doğrulama protokolü kullanılmıştır.

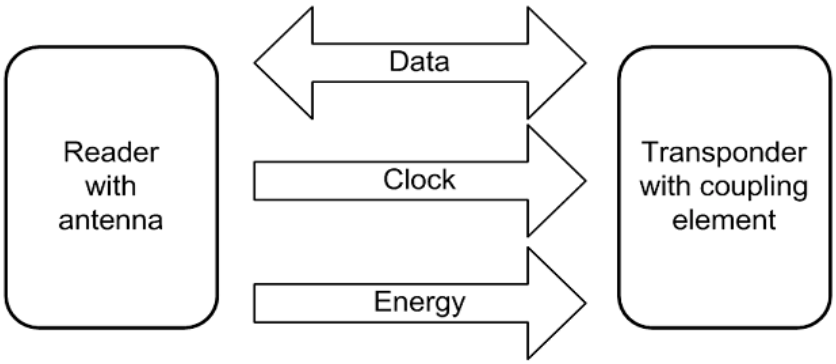
2.1.1 Etiket

Güç kaynağına sahip olup olmama durumuna göre RFID etiketleri ikiye ayrılır. Bunlar aktif etiket ve pasif etiket olarak adlandırılır. Aktif etiketler içerisinde güç kaynağı barındırı ve onunla çalışır. Pasif etiketler ise enerjisini okuyucudan sağlar.

Genel olarak pasif etiketler kısa okuma mesafesine sahiptir fakat uzun ömürlüdürler. Aktif etiketler ise uzun okuma mesafesine, yüksek hafızaya ve daha iyi gürültü korumasına sahiptir. Yalnız pahalı ve kısa ömürlüdürler[7]. sahip oldukları anten aracılığıyla kablosuz olarak veri iletir ve alırlar.

2.1.2 Okuyucu

Okuyucu elde taşınabilir bir bilgisayara bağlı olabileceği gibi, daha büyük daha uzak mesafeli çalışabilecek bir şekilde büyük bir bilgisayara da bağlı olabilir. Görevi etiketi sorgulamak ve enerji göndermektir. Etiketten gelen bilgiye göre bir tanımlama yapar. Etiket ile okuyucu arasında Şekil 2.2'deki bağlantılar vardır.



Şekil 2.2 : Etiket ile okuyucu arasındaki hatlar [8]

Hatların nasıl bir çalışmaya sahip olacağı gerçeklenecek protokol uyarınca ISO/IEC 18000 standardlarına bağlı olarak belirlenir. Ayrıca RFID sistem bir okuyucu ve birden çok etiketten oluşabilmektedir.

2.2 Kimlik Doğrulama Protokolü

Martin Feldhofer'ın yazmış olduğu protokol ile RFID sistemimizin kimliklendirme işleminin nasıl olacağı, sorgulama işleminin nasıl yapılacağı ve hangi standartlara ve kurallara uyulacağı açıklanmaktadır. Kimlik doğrulama tek yönlü olabileceği gibi çift yönlü de olabilir. İki yönlü kimlik sorma-yanıt(challenge-response) yapısı ile güvenlik daha da artmaktadır. Bu yapı ile 2 tarafın aynı anahtara sahip olup olmadığı belirlenmektedir. Bölüm 2.2 de kimliklendirme işleminin sırası ve nasıl olacağı ayrıntılı anlatılmıştır.

2.2.1 Kimliklendirme Mekanizması

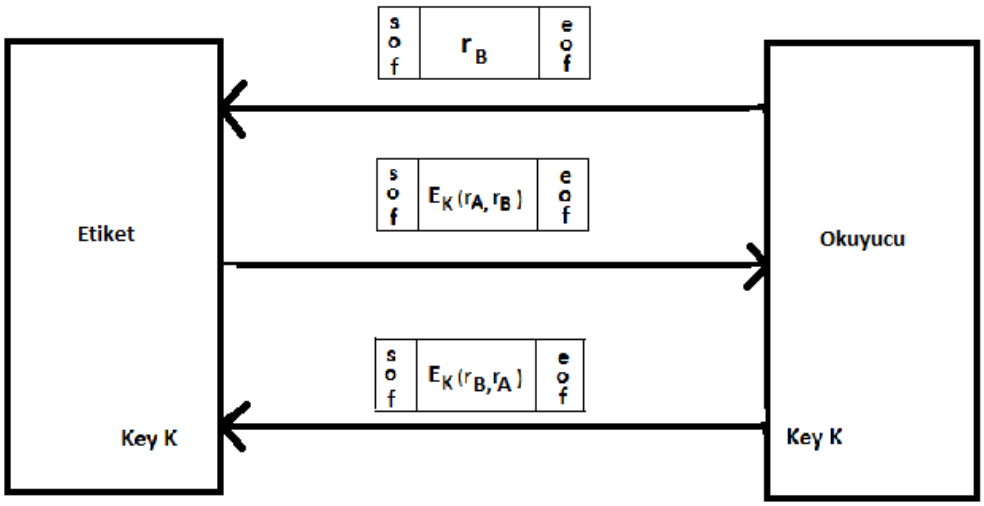
Kimliklendirme mekanizması iki yönlü kimlik sorma-yanıt yapısına sahiptir. Önce okuyucu konuşur. Öncelikle okuyucu ve etiket A ve B olarak simgelenmiştir. B'den A'ya bir rastgele sayı gönderilmekte, bu sayıya ek olarak A da kendi bir rastgele sayı üretmektedir. Bu sayılar şifrelenerek B'ye geri gönderilmektedir. B sayıların şifresini çözer, kendi rastgele sayısını elde ederse bir taraf doğrulanmış olur. Sonra A'ya sayılar yerleri değiştirilip şifrelenerek gönderilir. A'da aldığı veriyi çözer ve doğruluğuna bakar. Böylece okuyucu ve etiket birbirini tanımış olur. Bu protokol Martin Feldhofer'ın makalesinde Şekil 2.3'teki gibi verilmiştir.

$$
\begin{aligned}
A \leftarrow B: &\quad r_B \\
A \rightarrow B: &\quad E_K(r_A, r_B) \\
A \leftarrow B: &\quad E_K(r_B, r_A)
\end{aligned}
$$

Şekil 2.3 : Kimliklendirme Protokolü [3]

2.2.2 Doğrulama Mekanizması

Doğrulama protokolü ile A ve B'nin ISO/IEC 18000 standardı uyarınca okuyucu mu etiket mi olacağı belirlenir. Veri transferinin hangi format çerçevesinde olacağını belirler. Burada A ile okuyucu, B ile etiket tasvir edilmiştir. Protokolün detaylı şeması Şekil 2.4'teki gibidir.



Şekil 2.4 : Doğrulama protokolü

Doğrulama protokolünde gösterilen rastgele sayılar 128 bitlik ikili sayılardır. Buradan iki tarafta da rastgele sayı üreteci gerekliliği ve etikette şifreleme ve şifre çözme yapılarının gerekliliği görülmektedir. Şekilden ayrıca bilgilerin temel anlamda nasıl olacağı anlaşılmaktadır.

Gönderilecek paketlerin hangi çerçevede olacağı ISO/IEC 18000 standardı ile belirlenmiştir. Bölüm 2.4'de ISO/IEC 18000-3 standardı ile okuyucu ile etiket arasındaki kısıtlamalar ve özellikler anlatılmıştır.

2.2.3 ISO/IEC 18000 Standardı

Etiket ile okuyucu arasında veri taşıma sırasını, verinin hangi çerçeve içerisinde olacağı, çerçeve içinde bulunan diğer bilgilerin ne olması gerektiği, verinin hangi modülasyonla gönderileceği gibi konular ISO/IEC 18000 Standardı ile belirlenmiştir. Bu standarda göre, RFID okuyucu ve etiket 13.56 Mhz frekansında haberleşmektedir [9].

Standarda göre okuyucu ile etiket arasında haberleşme belirli modülasyonla sağlanır. Okuyucu haberleşmek için endeksi %10 ve %100 olan Genlik Kaydırmalı Anahtarlama (Amplitude Shift Keying, ASK) modülasyonu kullanmaktadır. Veri şifrelemesi “256’da 1” ya da “4’te 1” veri şifreleme biçimi ile mümkün olur. Veri şifreleme biçimine göre çıkış yolu oranı (uplink rate) saniyede 26.69 kbit hızına ulaşabilmektedir [9].

Ayrıca standart "okuyucu önce konuşur" prensibi ile oluşturulmuştur. Öncelikle okuyucudan bir istek gönderilir. Buna karşılık etiketten bir yanıt beklenir. Tüm paralo sorma- yanıt sistemi bir çerçeve düzeninde olmalıdır. Sorgulama ve yanıt çerçeveleri içerisinde verilecek komutlar ve gönderilecek veriler bir düzene koyulmaktadır. Şekil 2.5'te okuyucudan etikete gönderilen sorgulama çerçevesi görülmektedir.

SOF	Flags	0xA0	IC Mfg code	UID	Random number r_R	CRC	EOF
	8 bit	8 bit	8 bit	64 bit	128 bit	16 bit	

Şekil 2.5 : İstek çerçevesi [8].

Çerçeveler SOF ile başlayıp EOF ile bitmektedir. Bunları 8 bitlik bayraklar izlemektedir. Çerçevede ayrıca 8 bitlik 0xA0 komutu, 8bitlik üreticinin verdiği ürün kodu, 64 bitlik kullanıcı numarası(her birine özel), 128 bitlik rastgele sayı ve 16 bitlik Çevrimsel Hata Denetimi(CRC) içermektedir.

Şekil 2.6'da etiketten okuyucuya gönderilen yanıt çerçevesi görülmektedir.

SOF	Flags	UID	Signed data $E_K(r_R)$	CRC	EOF
	8 bit	64 bit	128 bit	16 bit	

Şekil 2.6 : Yanıt çerçevesi[8]

Yanıt çerçevesinde de sorgulama çerçevesine benzer olarak bayraklar kişisel numara ve CRC bulunmaktadır. Burada gönderilen veri okuyucudan gelen rastgele sayının şifrelenmiş halidir.

Yanıt ve sorgulama çerçevelerinde veri uzunlukları 0-8 byte arası değişebilmektedir. Aşağıda çerçeve parçalarının detayları verilmiştir.

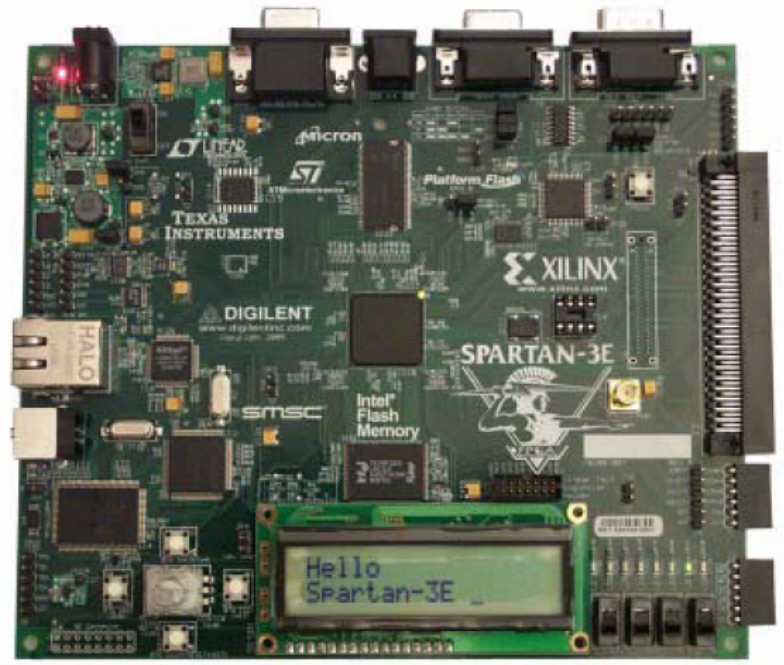
- Bayraklar: Bir ya da iki alt taşıyıcı frekansını ve yanıt için hangi veri oranının kullanılması gerektiğini göstermektedir. Uygun görülen etiketleri adreslemek için ekstra bilgiler sunulmaktadır. Etiketin yanıtı bayrakları kullanarak haberleşme sırasında oluşan hataları göstermektedir.
- Komut Kodu: Bir baytlık sabit, hangi isteğin gönderildiğini göstermektedir. Üç adet temel komut mevcuttur. Bunlardan birincisi olan zorunlu komutlar etiket tarafından gerçekleştirilmelidir. Seçmeli komutlar uygulama için gerekli ise etiket tarafından gerçeklenebilir. Özel komutlar ise kendi komutlarını protokole eklemek isteyen üreticiler tarafından kullanılabilir.
- Parametreler ve Veri Alanları: İstek ve Yanıtı işlemek için gerekli bilgileri barındıran özel komutlardır.
- CRC: Çevrimsel Hata Denetimi kendi hariç SOF'den sonra gelen bütün baytların belli bir algoritma içerisinde hesaplanmasıyla oluşturulmakta ve haberleşme esnasında herhangi bir hata olup olmadığını ortaya çıkarmak için kullanılmaktadır [8].

3. GERÇEKLEME ORTAMLARI

Kimlik doğrulama protokolü Spartan-3E kartındaki MicroBlaze işlemcisi ile kontrol edilecektir. Protokol içerisinde şifreleme, şifre çözme ve rastgele sayı işlemleri yapılması gerekmektedir. Bunun için gerekli işlemler ISE aracında Verilog HDL ile oluşturulan donanımlarla gerçekleştirilmiştir. Bu donanımlar MikroBlaze işlemcisine EDK ve SDK araçları ile bağlanmıştır. Bu araçlar bu bölümde incelenecektir.

3.1 Spartan-3E Kartı

Spartan-3E başlangıç kiti kartı kimlik doğrulama protokolünü gerçeklemek için gerekli konfigüre edilebilir kontrolörü, çeşitli çevreselleri ve FPGA'sı ile ihtiyacımızı karşılamaktadır. Ayrıca, kart MicroBlaze gömülü işlemcisinin temel özelliklerini ve Xilinx Gömülü Geliştirme Kiti(EDK)'ni sağlamaktadır[10]. 10.000 in üzerinde mantık hücresiyle MicroBlaze ve mantıksal devrelerimizi gerçeklemek için yeterli olmaktadır. Kart Şekil 3.1'de görüldüğü gibidir.



Şekil 3.1 : Spartan-3E kartı[10]

3.1.1 MicroBlaze İşlemcisi

MicroBlaze kişiye tasarımı doğrultusunda çeşitli özellikler seçmesine izin veren yüksek konfigüre edilebilirlikli sanal işlemci çekirdek işlemcidir[11]. Bu işlemci

zorunlu bazı özellikler dışında kullanıcıya istediği çevreseli ekleme ve çıkarma olanağı sağlayanbir işlemcidir. Bu işlemci fiziksel olarak kartın üzerinde bulunmaz. EDK tasarım aracı sayesinde istenilen doğrultuda FPGA' da oluşturulur. Gerçeklenecek kimlik doğrulama protokolü için TEA donanımları ve rastgele sayı üreteci, MicroBlaze'in konfigüre edilebilirlik özelliğinden faydalanılarak sisteme bağlanmıştır. MicroBlaze ayrıca tasarımımızın tüm kontrolünden ve akışından sorumludur. Kontrolü sağlamak için SDK aracında C dili ile program yazılmıştır.

3.2 ISE Ortamı

ISE ortamı genel olarak HDL tasarımının tüm aşamalarının yapılabildiği tasarım aracıdır. Başlangıçta seçilen tasarım dili ve kartın büyüklüğü doğruldusunda sayısal donanım tasarımları buradan yapılabilmektedir. Oluşturulan tasarımların benzetimlerine de olanak tanımaktadır. Donanımın, sentezleme ve yerleştirme aşamalarını tamamlamasıyla birlikte tasarım karta yine bu araç ile aktarılabilmekte ve fiziksel olarak denemeler yapılabilmektedir. Araç ayrıca tasarımımızın kullanacağımız kartta ne kadar yer kapladığı gibi ayrıntı bilgileride içermektedir.

Gerçeklediğimiz kimlik doğrulama protokolü çerçevesinde ISE aracı TEA donanımlarının ve rastgele sayı üreteci donanımının tasarımı ve denenmesi sırasında sıkça kullanılmıştr. Bu süreçler ile ilgili ayrıntılı bilgiler Bölüm 4'te verilecektir.

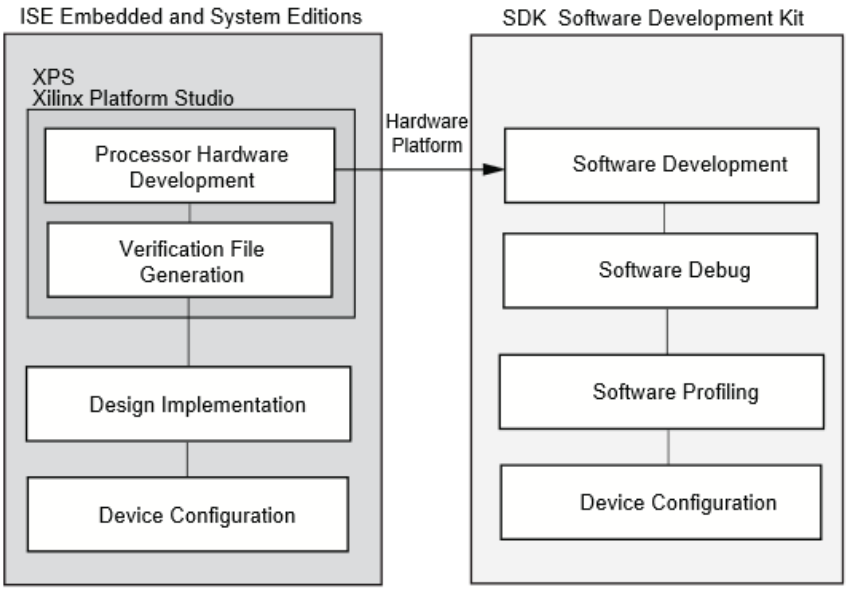
3.3 EDK Ortamı

EDK çeşitli araçlardan ve IP'lerden oluşan Xilinx FPGA'larında komple bir gömülü işlemcili sistem tasarımı yapılabilen bir süittir[12]. EDK, ISE aracı olmadan sentezleme, yerleştirme ve bağlantı aşamalarını yapamayacağı için ISE'ye ihtiyaç duyar. Oluşturulacak sistem EDK içindeki XPS ve SDK gibi iki önemli araç ile şekillendirilmektedir.EDK kapsamında:

- XPS arayüzü,
- Gömülü sistem araçları süiti,
- işlemcilerden ve çevresellerden oluşan gömülü işlemler akıllı ürünleri(IP),
- gömülü yazılım uygulamaları geliştirmek için açık kaynak kodlu Eclipse

arayüzü temelli Yazılım Geliştirme Kiti(SDK)

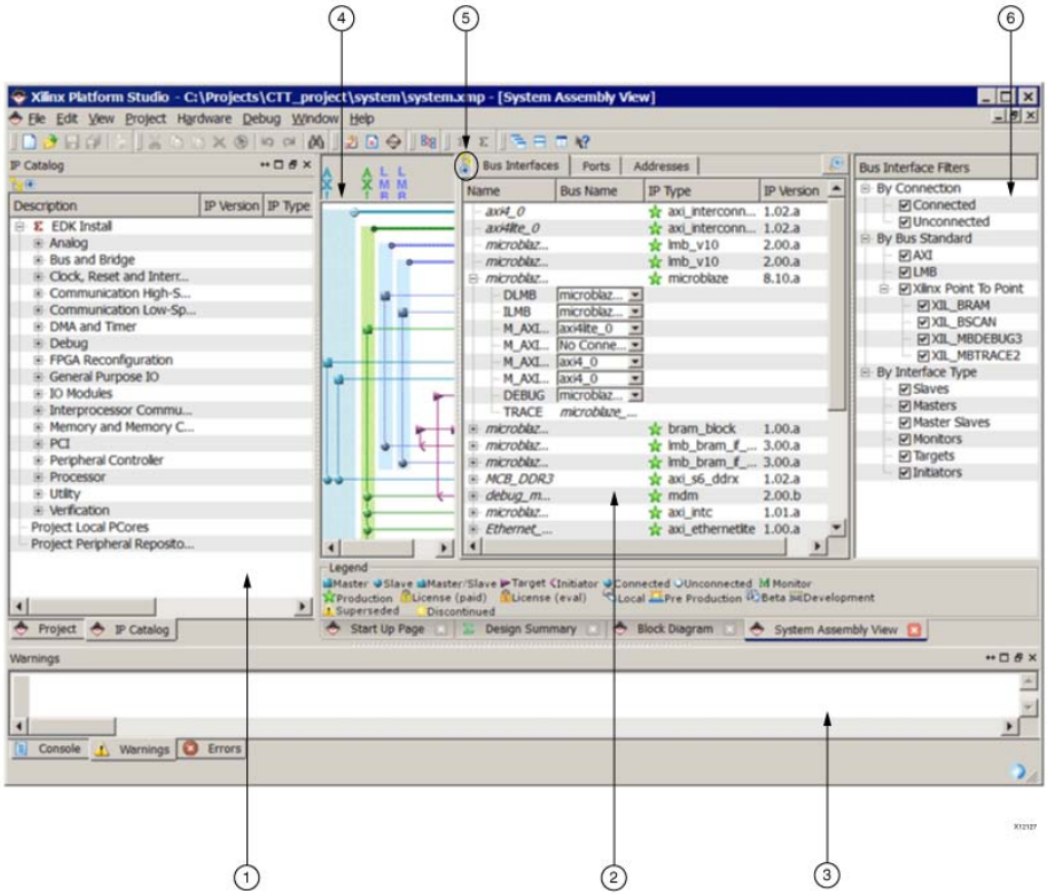
bulunmaktadır[13]. EDK ile yapılan tasarım aşamaları aşağıdaki Şekil 3.2'de görülmektedir.



Şekil 3.2 : Temel Gömülü Sistem Tasarım Akışı[12]

3.3.1 XPS Ortamı

EDK ortamının donanım ile ilgili işlemlerinden sorumlu tasarım aracıdır. Bir kullanıcı arayüzü sayesinde bir veya daha fazla MicroBlaze işlemcisine sahip sistem istenilen çevreseller eklenerek oluşturulabilmektedir. XPS ayrıca kişiye özel donanımların sisteme eklenmesine izin vermektedir. Eklemenin doğru bir şekilde tamamlanması için XPS ana ekranında adresleme ve bağlantı işlemlerinin yapılması unutulmamalıdır. XPS ana ekranına ait görüntü Şekil 3.3'te verilmiştir.



Şekil 3.3 : XPS ana ekranı[12]

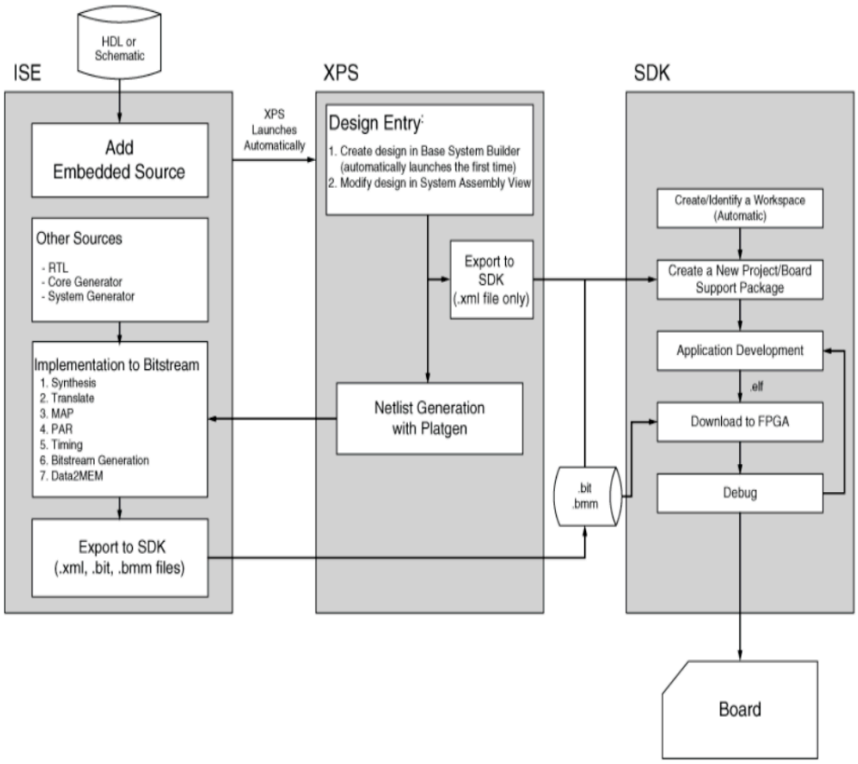
- Proje bilgi alanı (1)
- Sistem temel görüntüsü(2)
- Konsol pencere(3)
- Bağlantı paneli(4)
- Görüntüleme butonları(5)
- Filtreleme paneli(6)

Kullanılacak kart doğrultusunda gerekli donanım ekleme işlemleri tamamlandığında, artık sistem SDK ortamına geçirilmeye hazırdır.

3.3.2 SDK Ortamı

SDK ortamı XPS oluşturulan donanım sisteminin aktarılmasıyla birlikte bir *çalışma ortamı* açar. Bu çalışma ortamı açık kaynak kodu Eclipse temellidir. Burada C/C++ dilleriyle MicroBlaze temelli sistemimizi programlayabilmekteyiz. Ayrıca XPS'te eklenen kişisel donanımlar için aktarım sırasında gerekli kütüphane dosyaları oluşturulmaktadır. Bu kütüphaneler SDK oramına tanıtılarak uygulamalarda kullanma olanağı elde edilir.

Komple bir gömülü sistem tasarımının tüm akışı Şekil 3.4'te görüldüğü gibidir.



Şekil 3.4 : Sistem tasarım akışı[12]

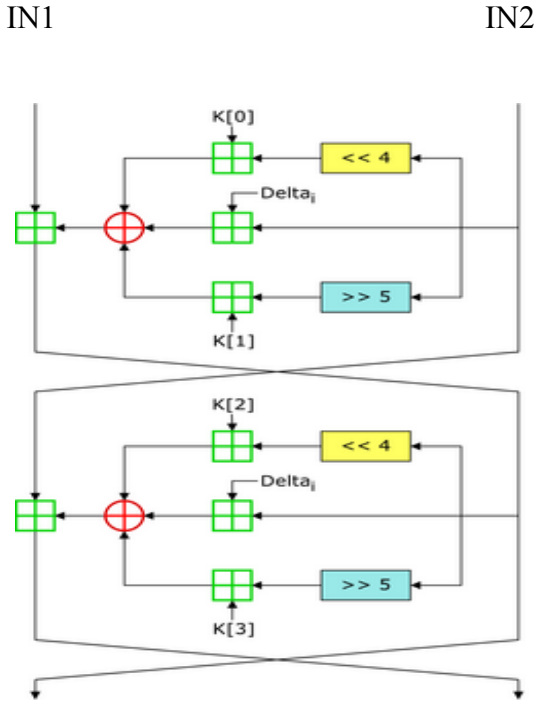
Öncelikle sistemde kullanılacak ISE'de oluşturulmalı ve kullanıma hazır hale getirilmelidir. Oluşturulan HDL, XPS'te oluşturulan MicroBlaze temelli projeye eklenir. Buradan SDK'ya aktarım yapılır. SDK'da uygulama geliştirilir. FPGA'ya yüklenip hata ayıklama yapılır. Tüm hatalarından ayıklanan sistem karta yüklenir.

4. DONANIM GERÇEKLEMELERİ

Güvenli RFID sistemleri için gerçeklediğimiz kimlik doğrulama protokolünde temel olarak kullanılan işlem şifreleme ve şifre çözmedir. Protokol gönderilecek bilginin ve tanınacak kimliğin şifre algoritmalarından geçmesine dayanır. Bu işlemi yapmak üzere MicroBlaze işlemcisine şifreleme donanımları eklenmiştir. Ayrıca rastgele sayı üreteci donanımı kullanılarak saldırılara karşı bir güvenlik ve gizlilikte sağlanmaktadır. Donanım gerçeklemeleri ISE ortamında yapılmıştır. Bu bölümün ilerleyen kısımlarında TEA algoritmasının ve rastgele sayı üretecinin ISE ortamı kullanılarak gerçeklenmesi aşamaları detaylarıyla anlatılmıştır.

4.1 Küçük Şifreleme Algoritması(TEA)

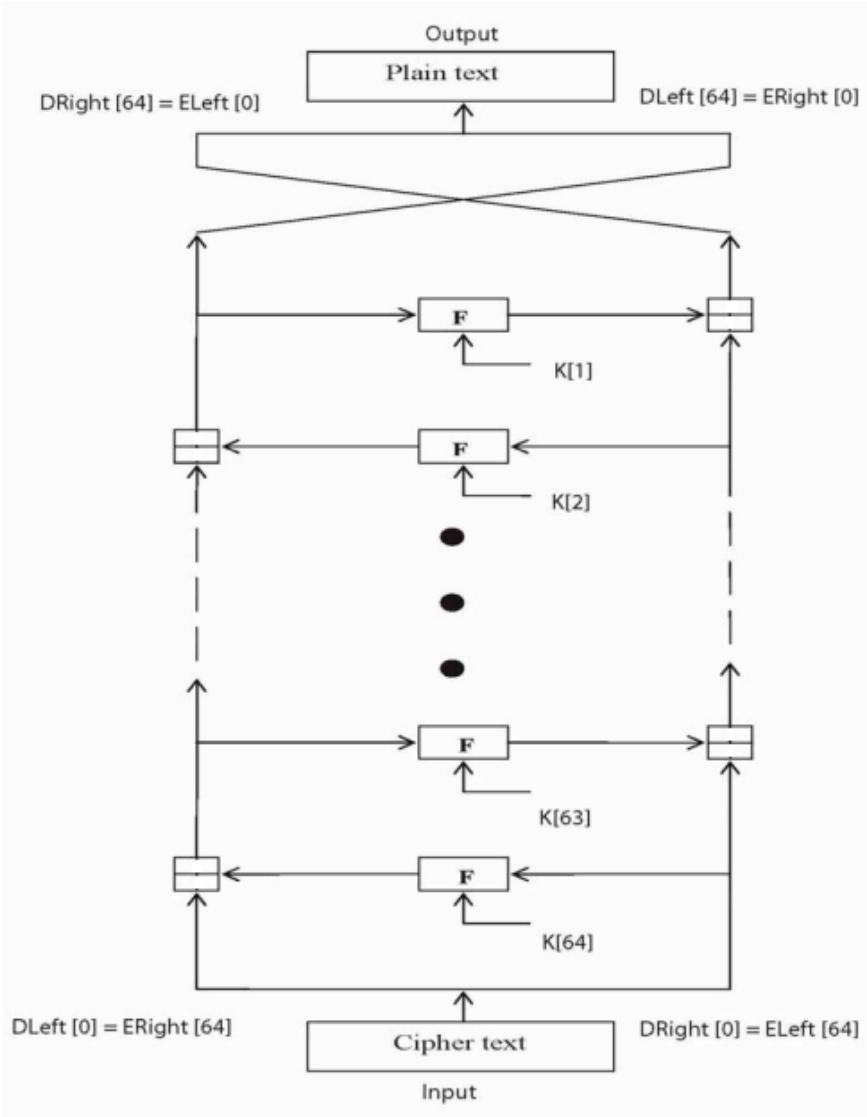
TEA kimlik doğrulama protokolü içerisinde gönderilen ve alınan rastgele sayıların şifrelenmesi ve şifre çözmesi sırasında kullanılmaktadır. Algoritma genel olarak alınan 64 bitlik bir sayıyı şifreleyerek yine 64 bitlik bir sayı vermektedir. Bu algoritmaya ait bir çevrim Şekil 4.1'deki gibidir. TEA birbiri ardına eklenmiş aynı 32 çevrimden oluşmaktadır.



Şekil 4.1 : Temel TEA çevrimi

Şifreleme girişlere ek olarak bir anahtar yardımıyla yapılmaktadır. 128 bitlik ikili anahtar ile her bir giriş için farklı bir çıkış elde edilmektedir. Algoritma incelendiğinde Delta diye bir değişken görülmektedir. TEA $(\sqrt{5}-1)*2^{31}$ eşitliğinin sonucunda oluşan altın orandan türetilmiş sayının 32 bitlik 32 parçasından oluşur. Algoritmada ayrıca toplama, özel veya(exor) ve sağa sola kaydırma işlemleri gerekmektedir.

Şekil 4.2'de şifre çözme TEA rutini görülmektedir.

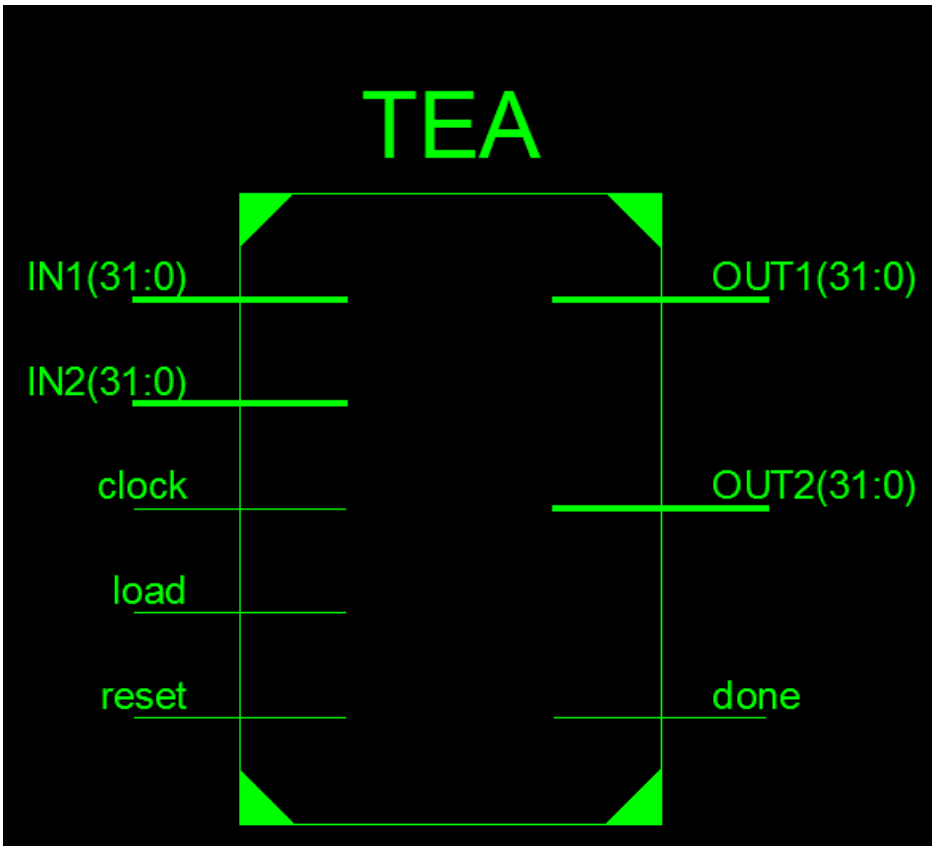


Şekil 4.2 : TEA şifre çözme rutini[14]

Şifreleme ile hemen hemen aynı yapıya sahiptir. İşlemler sırasında küçük bir fark olarak çıkarma bloğu bulunmaktadır. Şifreleme işleminde girişler baştan verilip sondan alınırken şifre çözme işleminde şifreli girişler sondan verilip baştan alınmaktadır.

4.1.1 TEA Donanımları Gerçeklemesi

ISE ortamında Verilog HDL ile TEA şifreleme ve şifre çözme algoritmaları gerçeklenmiştir. Şekil 4.1 ile verilen bir çevrim kullanılmış, aynı işlem 32 defa tekrarlanarak şifreleme işlemi tamamlanmıştır. Bu alandan kazanç sağlarken zamandan kayıp sağlamaktadır. Çünkü her bir çevrim bir sonraki yükselen saat kenarı ile tetiklenmiştir. Kullanılan işlemcimizin 50MHz ile çalıştığı düşünüldüğünde şifreleme işleminin 32*20=640ns civarı süreceği söylenebilir. Bir çevrim göz önüne alındığında burada bulunan toplama, özel-veya ve kaydırma modülleri TEA şifrelemesi boyunca bir çok kez çağırılan alt modüllerdir. Bunlarda ayrıca gerçeklenmiştir. Şekil 4.3 ile TEA şifreleme algoritmasının en üst modül RTL şematiği görülmektedir. Alınan yükle işaretiyle birlikte 32 saat darbesi sonunda şifreleme tamamşanmaktadır. Şifre çözme donanımı da aynı şematiğe sahip olacağı söylenebilir. Farklılık çıkışın giriş, girişin çıkış olacağı ve içeride bir toplama yerine bir çıkarma modülü olacağı olarak özetlenebilir.



Şekil 4.3 : TEA üst modül görüntüsü

Aynı şekilde şifre çözme donanımı içerisindeki çıkarma, toplama, özel-veya ve kaydırma donanımlarıyla birlikte ISE ortamında tasarlanmıştır. Şifreleme ve şifre

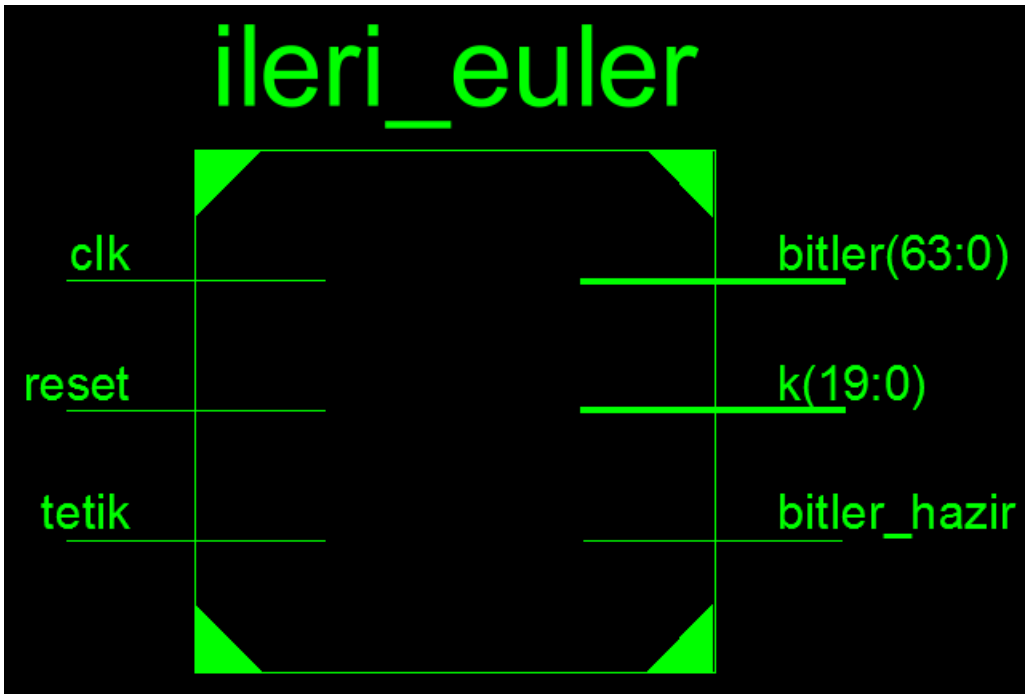
çözme donanımlarının aynı tasarım ortamında benzetimleri yapılmıştır. Şifreleme donanımına ait benzetim çıktısı EK-A kısmında şifre çözme donanımına ait benzetim çıktısı ise EK-B de görüldüğü gibidir.

Donanımlar benzetim sonuçları ile doğrulandıktan sonra EDK ortamında MicroBlaze işlemcisine bağlamak için hazır hale gelmiştir.

4.2 Rastgele Sayı Üreteci Donanımı Gerçeklemesi

Kimlik doğrulama protokolümüz için gerekli olan rastgele sayı üreteci donanımı için

Gömülü Sistem Tasarımı Laboratuvarında ileri euler metoduyla oluşturulmuş rastgele ikili sayı üreteci kullanılmıştır. Verilog HDL ile hazırlanan bu üreteç sistem saat frekansından daha düşük bir hızda örneklenerek 64 bitlik bir hafızaya kaydedilmiştir[15]. Böylece gerekli rastgele sayı üretilmiştir. Şekil 4.4'te bu donanıma ait üst modül görülmektedir.



Şekil 4.4 : Rastgele sayı üreteci üst modül görüntüsü[15]

Burada görüldüğü üzere verilen tetik işaretiyle birlikte 64 bitlik sayı toplanmaktadır ve bitlerin hazır olduğu bilgisiyle birlikte dışarı verilmektedir. Rastgele sayı üretecine ait simülasyon görüntüsü EK-C de verilmiştir.

5. YAZILIM GERÇEKLEMELERİ

Kimlik doğrulama protokolümüz içerisinde kullanılacak olan şifreleme, şifre çözme ve rastgele sayı üreteci donanım olarak oluşturulmuştur. Bunların microblaze işlemcisiyle beraber kullanılması için öncelikle XPS ortamına eklenmesi ve MicroBlaze'e bağlanması gerekir. Bundan sonra SDK ortamında yazılacak C kodu ile gerekli yazılım tasarımı yapılabilmiştir. Son olarak okuyucu ve etiket olarak çalışacak FPGA'lerin kablosuz haberleşmesi için RFM22B rf alıcı-verici modülü kullanılmıştır. RFM22B'de SPI üzerinden yine C kodu ile kontrol edilerek yazılım gerçeklemeleri tamamlanmıştır.

5.1 TEA Yazılımlarının Gerçeklenmesi

MicroBlaze temelli oluşturduğumuz kimlik doğrulama protokolünde kullanmak üzere hazırlanan şifreleme donanımlarının kullanılabilmesi için SDK ortamına taşınmaları ve burada yazılacak C kodu ile kontrolü gerekir. Bunun için XPS ortamındaki projeye TEA donanımları eklenmeli ve bağlanmalıdır. Ekleme işleminden sonra tasarımın son haliyle yeni bir SDK projesi oluşturulması için SDK'ya geçişi yapılır. Bu geçiş sırasında EDK aracı sayesinde TEA donanımlarımızı SDK ortamında kullanabilmemiz için sürücü dosyaları oluşturulur. MicroBlaze işlemcisi ile kullanmak üzere oluşturulan sürücü dosyasıyla birlikte TEA kullanma kodları oluşturulmuş olur. Bunları denemek için SDK'da boş bir C kaynağı açılıp burada gerekli sürücüler eklenerek C kodu ile kontrol edilebilir.

Projede hata ayıklamak için FPGA bilgisayara bağlanır. Bu bağlantı JTAG-USB bağlantı kablosuyla yapılır. Bağlantı yapıldıktan sonra yazılım FPGA üzerinde koşturulur ve hata ayıklama yapılır.

5.2 Rastgele Sayı Üreteci Yazılımı Gerçeklemesi

Okuyucu ve etiketler için kimlik doğrulama protokolü dahilinde rastgele sayı üreteci gereklidir. Bunun için donanım oluşturulmuştur. SDK ortamında yazılım ile kontrol edilebilir olması için gerekli eklemelerin ve bağlantıların yapılması gerekir. TEA donanımlarında olduğu gibi Öncelikle XPS projesine donanımlar eklenir ve bağlanır

ardından SDK ortamına aktarım yapılır. Rastgele sayı üreteci yazılımsal olarak bir tetik girişi verilerek çıkışından hazır olduğunda 64-bitlik rastgele sayı alınır. Her sayısal gerçeklemede olduğu gibi bu rastgele sayı üreteci de bir sözde rastgele sayı üretecidir. Bu nedenle bir süre sonra kendini tekrar etmeye başlayacaktır. Gerçek bir rastgele sayı üreteci olabilmesi için doğadaki rastgele gelişen bir olaydan beslenebilir. Modülümüz bu haliyle de önceden hesaplanması kolay olmayan bir rastgele sayı üretmektedir.

SDK ortamında C diliyle hazırlanan kodlar, FPGA'i bilgisayara bağlayarak donanım üzerinde koşturulur ve hata ayıklama işlemleri yapılır.

5.3 RF Alıcı-verici Yazılımı Gerçeklemesi

RFM22B rf alıcı-verici modülü spi üzerinden seri olarak haberleşmektedir. Protokol ile ilgili rastgele sayı üreteci ve şifreleme işlemleri, yazılımla kontrol edildikten sonra haberleşmenin kablosuz olarak gerçekleştirilmesi bu modül ile sağlanmıştır. SPI'da SDK ortamına taşındıktan sonra rf alıcı-verici, yazılım ile kontrol için hazırdır. Modülün çalışma mekanizmasına ait değişiklikler yazılımla yapılmaktadır. Haberleşmenin hangi frekansta yapılacağı, veri gönderme ve alma hızının kaç olacağı, çıkış gücünün ne kadar olacağı yazılım ile ayarlanmaktadır. Tüm ayarlar yapıldıktan sonra protokolde belirtilen çerçeveye oturtulmuş veriler için 8-bitlik parçalar halinde seri olarak gönderme ve alma işlemleri yapılmıştır.

6. SİSTEMİN GERÇEKLENMESİ

Güvenli RFID sistemler için gerçeklediğimiz kimlik doğrulama protokolü donanım ve yazılım parçalarından oluşmaktadır. Donanım parçalarımız başta Spartan-3e kartı ve sanal işlemcisi MicroBlaze olmak üzere TEA şifreleme ve şifre çözme donanımları, rastgele sayı üreteci ve RF alıcı-vericiden oluşmaktadır. Tüm donanım parçaları XPS ortamına eklenmiş ve birbirine bağlanmıştır. Ardından SDK ortamına geçilmiştir ve yazılım tasarımları yapılmıştır.

XPS'ten SDK'ya geçerken oluşturulan kendi donanımlarımız için sürücü kodlar otomatik oluşturulmaktadır. Bunları kullanabilmek için C kaynak kodunda bunların çağırılması gerekir. Tüm bunların ardından bilgi alış verişi bir çerçeve düzeninde yapılır. Kimlik doğrulama protokolü ile etiketin doğru anahtara sahip olup olmadığı sınanmış olur. Sistem birden fazla etiketten oluşabilir. Bunun için bilgi çerçevesi içerisinde her bir etikete kendisine özel bir kimlik numarası verilir. Böylece etiketin kim olduğu ve doğru anahtara sahip olup olmadığı sınanır. Aynı şekilde etikette okuyucudan gelen bilgiye bakarak doğru anahtara sahip olup olmadığını sınar. Böylece karşılıklı olarak kimlik doğrulama işlemi gerçekleştirilmiş olur.

7. SONUÇLAR

Martin Feldhofer, Sandra Dominikus, and Johannes Wolkerstorfer'ın güvenli RFID sistemler için yazdığı kimlik doğrulama protokolü FPGA üzerinde gerçeklenmiştir. Sistem donanım ve yazılım olmak üzere iki parçadan oluşmaktadır. Kimlik doğrulama protokolü içerisinde kullanılacak şifreleme donanımı için TEA kullanılmıştır. Küçüklüğü ve gömülü sistemlere uygunluğuyla sistemimizde önemli rol almaktadır. Ayrıca gizliliği artıran diğer unsur ise rastgele sayı üretmekti. Rastgele sayı üreteci olarak Gömülü Sistem Tasarımı Laboratuvarında yapılan çalışmalarla oluşturulmuş sözde rastgele sayı üreteci kullanılmıştır. Sistemin kablosuz haberleşmesi için RFM22B adlı rf alıcı-verici modülü kullanılmıştır.

Sistem bir mikroişlemci tabanına kurulduğu için çoklu çalışmadan mahrumdur. Bunun giderilebilmesi için çoklu görev işlemi yerine getirecek bir gerçek zamanlı işletim sistemine oturtularak sistem daha efektif çalıştırılabilir. Bunlara ek olarak daha gelişmiş ve gerçek rastgele sayı üreteci kullanmak güvenlik açısından daha faydalı olacaktır.

KAYNAKLAR

[1] Huiyun, Li, 2009. Development and Implementation of RFID Technology, Development and Implementation of RFID Technology, Cristina Turcu (Ed.), InTech.

[2] Alparslan, S., 2012. Güvenli RFID Sistemleri için Bir Kimlik Doğrulama Protokolünün Gerçeklenmesi, *Lisans Bitirme Çalışması*, İ.T.Ü. Elektrik-Elektronik Fakültesi, İstanbul.

[3] Feldhofer, M., Dominikus, S., Wolkerstorfer, J., 2004, Strong Authentication for RFID Systems Using the AES Algorithm, *Cryptographic Hardware and Embedded Systems - CHES 2004*, pp. 357-370, 6th International Workshop Cambridge, MA, USA.

[4] Kavas, A., 2007. Radyo Frekans Tanımlama Sistemleri, **430,** s. 74-80.

[5] Lehtonen, M., Staake, T., Michahelles, F., Fleisch, E., 2008, From Identification to Authentication – A Review of RFID Product Authentication Techniques, *Networked RFID Systems and Lightweight Cryptography,* pp 169-187. Springer-Verlag Berlin Heidelberg.

[6] Chatmon, C., Le, T.V., Burmester, M., 2006. Secure Anonymous RFID Authentication Protocols. *Technical Report TR-060112*, Department of Computer Science, Florida State University, Tallahassee, Florida, USA.

[7] Shoewu, O., Badejo, O., 2006, Radio Frequency Identification Technology: Development, Application and Security Issues, *The Pacific Journal of Science and Technology*,**Vol.2**, s. 144-152.

[8] Feldhofer, M., 2004. "An authentication protocol in a security layer for RFID smart tags", Electrotechnical Conference, 2004. MELECON 2004. Proceedings of the 12th IEEE Mediterranean, **2**, 759- 762.

[9] ISO/IEC 18000-3, 2003. Information Technology AIDC Techniques - RFID for Item Management, International Organization for Standardization.

[10] Xilinx, 2006. Spartan-3E Starter Kit Board User Guide.

[11] Xilinx, 2007. MicroBlaze Processor Reference Guide.

[12] Xilinx, 2011. EDK Concepts, Tools and Techniques.

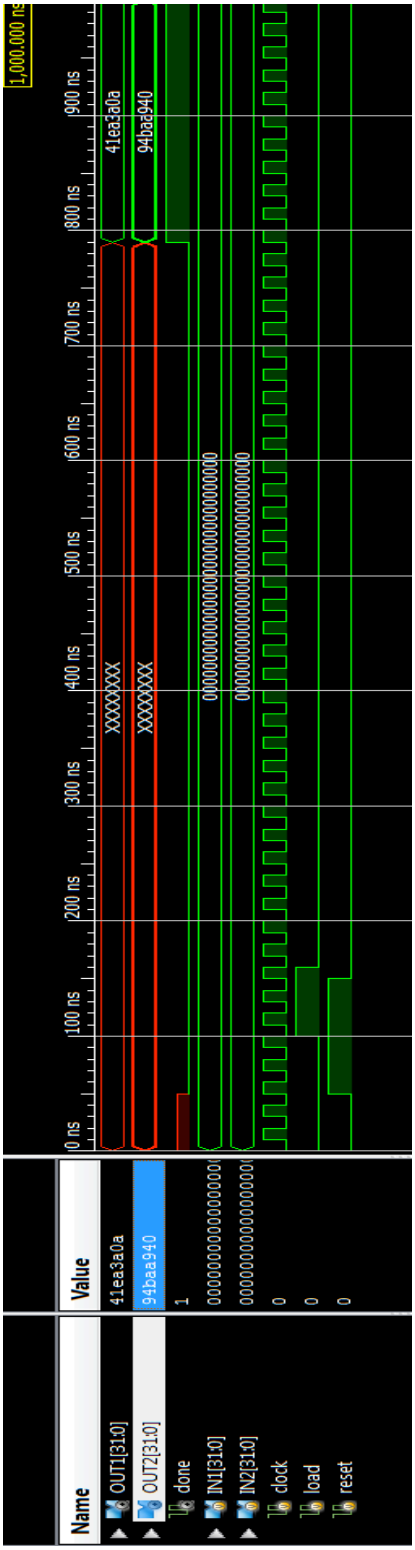
[13] Xilinx, 2007. Embedded System Tools Reference Manual.

[14] Andem, V.R., 2003. A Cryptanalysis of the Tiny Encryption Algorithm, *MSc. Thesis*, The University of Alabama, Alabama, USA.

[15] Ustaoğlu, B., 2013. Gerçek Rastgele Sayı Üreteci Tasarımı, Testleri ve Gerçeklenmesi, *Yüksek Lisans Tezi*, İ.T.Ü. Fen Bilimleri Enstitüsü, İstanbul.

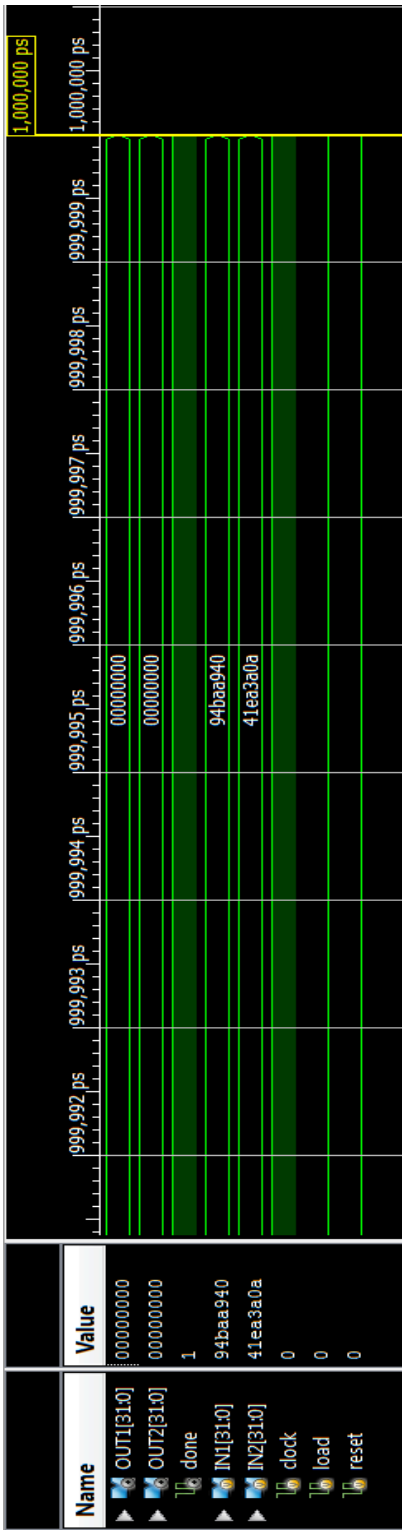
EK-A

TEA Şifreleme Bloğu Benzetimi



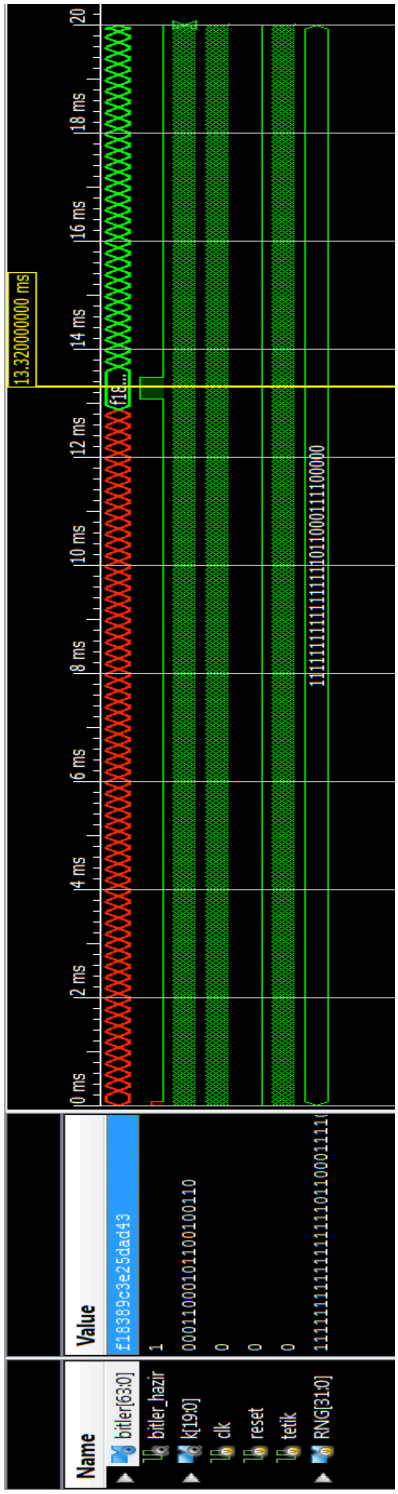
EK-B

TEA Şifre Çözme Bloğu Benzetimi



EK-C

64-bit Rastgele Sayı Üreteci Benzetimi



ÖZGEÇMİŞ

Adı Soyadı: Gökhan Ulutaş

Doğum Yeri ve Tarihi: İstanbul, 1990

Lise: Haydarpaşa Lisesi; 2004-2008

Lisans: İstanbul Teknik Üniversitesi, 2008-2013