İSTANBUL TEKNİK ÜNİVERSİTESİ

ELEKTRİK – ELEKTRONİK FAKÜLTESİ

BİR RFID DOĞRULAMA PROTOKOLÜNÜN GERÇEKLENMESİ

BİTİRME ÖDEVİ

SUBUTAY GİRAY BAŞKIR

040060367

Bölümü: Elektronik ve Haberleşme Mühendisliği Bölümü

Programı: Elektronik Mühendisliği

Danışmanı: Yrd. Doc. Dr. Sıddıka Berna ÖRS YALÇIN

MAYIS 2011

ÖNSÖZ

Mensubu olmaktan onur duyduğum İstanbul Teknik Üniversitesi'ndeki tez çalışmam süresince bana değerli vaktini ayırıp sınırsız anlayışıyla yardımlarını esirgemeyen danışman hocam Sayın Yrd. Doç. Dr. S. Berna Örs Yalçın'a teşekkürlerimi sunmayı bir borç bilirim.

Ayrıca tüm çalışmalarım boyunca benim yanımda olan aileme ve başta Özen Özkaya olmak üzere arkadaşlarım; İstanbul Teknik Üniversitesi Güneş Arabası Ekibi üyelerine de teşekkürlerimi sunarım.

Mayıs 2011 Subutay Giray Başkır

İÇİNDEKİLER

Sayfa

KISALTMALAR ..... **iv**
ŞEKİL LİSTESİ ..... **v**
ÖZET ..... **vi**
SUMMARY ..... **vii**

1. GİRİŞ **1**

2. RADYO FREKANSI İLE TANIMLAMA **3**
2.1. Pasif Etiketler İçin Veri Transferi 3
2.2. ISO/IEC 18000 Standardı 4

3. GELİŞMİŞ ŞİFRELEME STANDARDI **6**
3.1 Gelişmiş Şifreleme Standardının Aşamaları 6
3.1.1. Bayt Değiştirme 6
3.1.2. Satırları Kaydırma 8
3.1.3. Sütun Karıştırma 8
3.1.4. Tur Anahtarı Ekleme 9
3.1.5. Gelişmiş Şifreleme Standardının Gerçeklenmesi 9

4. DENEY DÜZENEĞİ **10**
4.1. Kullanılan Mikrodenetleyiciler 10
4.1.1. MSP430G2352 Mikrodenetleyicisi 11
4.1.2. STM32F103R6T6 Mikrodenetleyicisi 13
4.2. Kullanılan Kartlar 15
4.2.1. MSP430 Deneme Kiti 15
4.2.2. STM32 Haberleşme ve Görüntüleme Modülü 17

5. DOĞRULAMA PROTOKOLÜNÜN GERÇEKLENMESİ **18**
5.1. MSP430 LaunchPad ile Etiket Benzetimi 18
5.2. STM32 Haberleşme ve Görüntüleme Modülü İle Okuyucu Benzetimi 21
5.3. Cihazların Birbirleri İle Haberleştirilmesi 23
5.4. Doğrulama Protokolünün Gerçeklenmesi 24

6. SONUÇLAR VE TARTIŞMA **27**

KAYNAKLAR **28**

ÖZGEÇMİŞ **29**

KISALTMALAR

DES : Data Encryption Standard
RFID : Radio Frequency Identification
AES **:** Advanced Encryption Standard
MCU **:** Microcontroller Unit
SOF : Start of File
EOF : End of File
CRC **:** Cyclic Redundancy Check
ARM **:** Acorn RISC Machine
RF **:** Radio Frequency
UART **:** Universal Asynchronous Receiver/Transmitter
ID **:** Identification
DCO **:** Digitally Controlled Oscillator
RAM **:** Random - Access Memory
IDE **:** Integrated Development Enviroment
PWM **:** Pulse Width Modulation
I^2C **:** Inter-Integrated Circuit
SPI **:** Serial Peripheral Interface
USB **:** Universal Serial Bus
CAN **:** Controller Area Network
RISC **:** Reduced instruction set computer
TTL **:** Transistor Transistor Logic
CMOS **:** Complementary Metal Oxide Semiconductor

ŞEKİL LİSTESİ

Sayfa

Şekil 2.1 : RFID Siteminin Öğeleri. 4
Şekil 3.1 : AES blok diyagramı [8]. 7
Şekil 3.2 : AES Turu İç Yapısı [8]. 7
Şekil 3.3 : Durum Matrisi [8]. 8
Şekil 3.4 : Satırları Kaydırma İşlemi Blok Diyagramı 8
Şekil 3.5 : Sütun Kaydırma İşlemi Blok Diyagramı 9
Şekil 3.6 : Tur Anahtarını Ekleme İşlemi Blok Diyagramı [8]. 9
Şekil 4.1 : MSP430G2352 Fonksiyon Blok Diyagramı. 12
Şekil 4.2 : Code Composer Studio Ekran Görüntüsü. 13
Şekil 4.3 : STM32F103xx Fonksiyon Blok Diyagramı 14
Şekil 4.4 : Keil uVision4 Ekran Görüntüsü 15
Şekil 4.5 : MSP430 LaunchPad Üst Görünüm 16
Şekil 4.6 : Etiket Benzetimi İçin Kullanılan Donanım. 16
Şekil 4.7 : Haberleşme ve Görüntüleme Modülü. 17
Şekil 5.1 : Xbee Explorer. 18
Şekil 5.2 : X-CTU Ekran Görüntüsü. 19
Şekil 5.3 : 31 Byte'lık Verinin Alınması ve Aynen Gönderilmesi İle İlişkili Ekran Görüntüsü. 20
Şekil 5.4 : Terminal Programı İle Gönderilen 16 byte'lık Veri ve Etiket Tarafından Şifreleme Fonksiyonları Sonucu Oluşturulup Geri Gönderilen Verinin Karşılaştırılması. 21
Şekil 5.5 : 31 Byte'lık Verinin Okuyucu Tarafından Gönderilmesi İle İlişkili Ekran Görüntüsü. 22
Şekil 5.6 : Terminal Programı İle Gönderilen 16 byte'lık Veri ve Okuyucu Tarafından Şifreleme Fonksiyonları Sonucu Oluşturulup Geri Gönderilen Verinin Karşılaştırılması. 23
Şekil 5.7 : Sorgulama Çerçevesi 24
Şekil 5.8 : Cevaplama Çerçevesi 24
Şekil 5.9 : Doğrulama Protokolüne İlişkin Veri Akış Diyagramı 25

BİR RFID DOĞRULAMA PROTOKOLÜNÜN GERÇEKLENMESİ

ÖZET

RFID (Radio Frequency Identification, Radyo Frekansı İle Tanımlama) teknolojisi gün geçtikçe kullanımı yaygınlaşan bir teknolojidir. Bu yaygınlaşma doğrudan güvenlik açıklarının delinmesi yönündeki çabaları da beraberinde getirmektedir. RFID'nin kullanım alanları gizlilik seviyesi yüksek uygulamaları da kapsadığından açıkların kötü niyetli amaçlar doğrultusunda suiistimal edilmesi hayati problemlerin ortaya çıkma olasılığını doğurabilmektedir. Bundan ötürü RFID uygulamalarında kriptoloji algoritmalarının koşturulması kaçınılmaması gereken bir çözümdür.

Bu tez içeriğinde bir RFID uygulaması için doğrulama protokolünün gerçeklenmesi anlatılmıştır. Protokolün nasıl gerçeklenmesi gerektiği ile alakalı ön bilgi oluşturulması amacı ile öncelikle RFID tanıtılmış, sonrasında doğrulama protokolünde kullanılacak şifreleme standardı olan AES (Advanced Encryption Standard, Gelişmiş Şifreleme Standardı) çalışma aşamaları ile tanıtılmıştır.

Gerçekleme aşamasında ise öncelikle etiket ve okuyucu benzetimi yapılabilmesi için kartlar hazırlanmıştır. Bu kartlarda kullanılacak mikrodenetleyicilerin hafıza ve işlem kabiliyeti özellikleri önem arz ettiğinden ötürü seçimler bu tip parametreler dikkate alınarak yapılmıştır. Kartların hazırlanması aşamasından sonra ise sırasıyla, kartlar ayrı ayrı olacak biçimde bilgisayar ile haberleştirilmiş, AES standardına uygun şifreleme ve şifre çözme işlemleri gerçekleştirilmiş, sonrasında kartlar birbirleri ile haberleştirilmiş ve son olarak doğrulama protokolünün kartlar arasında gerçeklenmesi hedeflenmiştir.

Sonuç olarak bir RFID doğrulama protokolünün gerçeklenebilmesi için mikrodenetleyicilerin sahip olması gereken minimum işlem kabiliyetleri ve hafıza sınırları belirlenmiştir.

IMPLEMENTATION OF A RFID AUTHENTICATION PROTOCOL

SUMMARY

RFID is an ever prevalent becoming technology. This increase of use brings its disadvantage, efforts to breach its security. Since areas of use contain high security requiring applications, any case of security breach and abuse may lead to significant problems. As a result, generating cryptology algorithms is a solution that should be worked upon.

In this Bsc thesis project an implementation of an RFID authentication protocol is described. In order to compose information about how to implement this authentication protocol, firstly RFID is introduced then the encrypting standard which will be used while authentication protocol implementation AES is described with its working steps.

In the implementation section, in order to simulate tag and reader, suitable electronic cards are equipped. Because of the fact that the microcontrollers' memory sizes and processing capability are very important, the selection of microcontrollers which are capable for this implementation is made with considering these parameters. After the preparing section of tag and reader, these steps are executed consecutively; communication between electronic cards are built individually, AES functions are executed then communication between cards are built. After all, the execution of authentication protocol between tag and reader is aimed.

As a result, the minimum processing capability and memory size limits for an implementation of a RFID authentication protocol are determined.

1. GİRİŞ

Otomatikleşen endüstrinin gereklerinden ötürü tanımlanması gereken ürün veya cisim sayısı gün geçtikçe artmaktadır. Endüstriyel üretimlerde her bir ürünün üreticiden son kullanıcıya ulaşana kadar ya da kullanım sırasında defalarca tanımlanması gerekmektedir [1]. Bu objelerin Radyo Frekansı İle Tanımlama Yöntemi (RFID) ile tanımlanması RFID'ye rakip olabilecek değer alternatif yöntemlere göre çok daha avantajlıdır. İçerdiği mikrodenetleyiciler sayesinde ISO/IEC 18000 standardına uygun bir biçimde iki yönlü iletişimin gerçekleştirilebilmesi avantajlarının başlıcalarındandır [2]. RFID uygulamalarının bu olumlu özelliklerinin yanında herhangi bir doğrulama protokolüne sahip olmaması ise güvenlik açısından bu uygulamanın kullanılabilirliğinin sorgulanmasına sebep olmaktadır.

RFID uygulamalarına doğrulamanın eklenmesinin en iyi yolu kriptografik doğrulamanın uygulanmasıdır. Fakat düşük işlem kabiliyeti, bellek elemanları ve düşük güç tüketimi kısıtlarının bulunmasından ötürü kriptografik algoritmaların uygulanması konusunda problemler mevcuttur [4]. Bu bitirme ödevinde iki yönlü el sıkışma yöntemi ile kurulmuş bir doğrulama protokolü gerçeklenmiş, gerçeklenen protokolün ihtiyaç duyduğu hafıza gereksinimleri belirlenmiştir.

Doğrulama protokolün gerçeklenmesi ile yapılması hedeflenen şey; aktarılmak istenen mesajın aktarılma işleminden önce karşılıklı bir doğrulama işlemidir. Doğrulama işlemi kısaca; okuyucu tarafından rastgele bir sayı üretilmesi, sayının şifrelenmesi, şifrelenmiş sayının karşı tarafa gönderilmesi ve karşı tarafın şifreyi çözüp sayıyı geri göndermesi olarak özetlenebilir. Şifreleme işlemi AES standardına uygun bir biçimde gerçekleştirilmiştir. Okuyucu tarafında bu şifrelenen sayı ile alınan sayının karşılaştırıldığında eşitlik durumunun oluşması iki tarafta da aynı anahtarın bulunduğu, aktarılmak istenen mesajın aktarılması için olması gereken güvenli ortamın sağlandığı anlamına gelir. Böylece aynı anahtara sahip olmayan okuyucu ve etiket ikilisinin birbiri ile haberleşmemesi de sağlanmış olur.

Doğrulama algoritmasının gerçeklenebilmesi için RFID okuyucu ve etiket benzetimini gerçekleştirmek üzere iki adet MCU (Microcontroller Unit,

Mikrodenetleyici Birimi) kullanılmıştır. Etiket tarafında kullanılacak MCU'nun düşük güç ve düşük işlem kabiliyeti olması icap ettiğinden ötürü Texas Instruments firmasının 16 bitlik düşük güçlü RFID uygulamalarına uygun MSP430G2352 ürünü tercih edilmiştir. Okuyucu tarafında ise düşük güç ya da hafıza gibi hedefler gözetilmez; hızlı ve koşturulan algoritmalardan ötürü işlem kabiliyeti yüksek ürünler tercih edilir. Bu sebeplerden ötürü okuyucu tarafında içinde Cortex-M3 ARM çekirdeği olan 32 bitlik ST firmasının STM32F103R6T6 ürünü tercih edilmiştir. Okuyucu ve etiket taraflarının her ikisine de RF haberleşmenin sağlanabilmesi için Digi International firmasının Zigbee tabanlı Xbee XBP24-BCIT-004 ürünü kullanılmış, MCU'lar ve modemler arasındaki haberleşme UART ile sağlanmıştır.

Tezin ikinci bölümünde, bu tezin teorik temellerini oluşturan radyo frekansı ile tanımlama konusundan kısaca bahsedilmiştir. Çalışmada kullanılan şifreleme standardı ise üçüncü bölümde anlatılmıştır. Gerçeklemede kullanılan mikrodenetleyiciler ve geliştirme kartları gibi gereçler dördüncü bölümde tanıtıldıktan sonra beşinci bölümde sistemin nasıl gerçeklendiği açıklanmıştır.

2. RADYO FREKANSI İLE TANIMLAMA

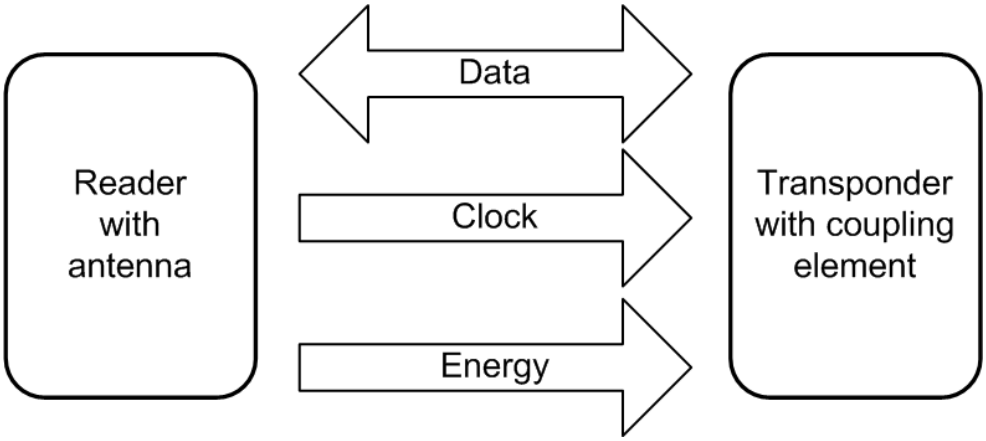
Radyo frekansı ile tanımlama, makineleşen dünyamızda birçok ürün veya objenin tanımlanmasında kullanılan güncel teknolojinin sunduğu, bu konuda en önemli rolü üstlenen tanımlama yöntemidir. Anten teknolojisinin ve MCU tasarım teknolojisinin gün geçtikçe gelişmesi sayesinde RFID daha fazla çeşitli kullanım analında yer almaya başlamıştır. Bunlara otomatik gümrük ödemeleri, hayvan kimliklendirilmesi, ürün takibi, otomatik üretim ve lojistik kontrolü gibi kullanım alanları örnek verilebilir [3].

RFID sistemleri genellikle iki ana öğeden oluşmaktadır. Bu iki öğe okuyucu ve etiket olarak adlandırılırlar. Okuyucu, içinde bulunan anten vasıtası ile etiket ile haberleşir. Etiket ise tanımlanması hedeflenen obje üzerine yerleştirilmiş elektronik cihazdır (bkz. Şekil 2.1).

2.1. Pasif Etiketler İçin Veri Transferi

Üzerinde güç kaynağı ve saat sinyali üreteci bulunmayan etiketler pasif etiketler olarak isimlendirilirler. Yukarıda da bahsedildiği gibi sistemin iki ana öğesi okuyucu ve etiket arasında Şekil 2.1'de gösterilen 3 hat radyo frekansı aralığındaki belirli bir frekansta transfer edilir. Bu hatlar; iki yönde olmak üzere veri, okuyucudan etikete doğru olmak üzere saat sinyali ve yine okuyucudan etikete doğru ve etiketi beslemek amacıyla olmak üzere enerji olarak tanımlanır. Saat sinyali, etiket tarafından okuyucunun gönderdiği taşıyıcı sinyal üzerinden alınmaktadır. Enerji ise pasif etiket olarak tanımlanan yapılar üzerinde yerel bir besleme kaynağı bulunmamasından ötürü okuyucu tarafından etikete gönderilmelidir [4].

Okuyucudan etikete doğru veri transferi dijital modülasyon ile yapılmaktadır. Bu uygulama için dijital modülasyon metotları arasından ASK (Amplitude Shift Keying, Genlik Kaydırmalı Anahtarlama) basit demodülasyon mekanizması sebebiyle sıklıkla tercih edilen modülasyon türü olmaktadır.



Şekil 2.1 : RFID Siteminin Öğeleri.

Kullanılabilecek ASK seviyeleri arasından iki tanesi RFID uygulamaları için uygundur. %100 seviyeli ASK kullanılması durumunda, verinin gönderilmiş olma durumuna göre taşıyıcı açık – kapalı olacak şekilde anahtarlanır. Bunun yerine %10 seviyeli ASK daha sıklıkla tercih edilir. Bunun sebebi taşıyıcı sinyalin düşük seviyeli fazında etikete gücün iletilememesinden kaynaklanmaktadır.

Etiketten okuyucuya veri transferi ise okuyucudan etikete veri transferinde olduğu gibi okuyucunun gönderdiği taşıyıcı sinyal sayesinde sağlanır. Etiket cevabını gönderebilmek için içinde bulunduğu alandaki enerjiyi kullanır. Bu metot Yük Modülasyonu (Load Modulation) olarak isimlendirilir. Yük Modülasyonu, devredeki yük direncinin gönderilen veriye bağlı olarak açık – kapalı olarak anahtarlandığı bir mekanizmadır. Direnç üzerinde oluşacak fazladan güç sarfiyatı okuyucu tarafından bir ve sıfırlar şeklinde tespit edilir [4].

2.2. ISO/IEC 18000 Standardı

ISO/IEC 18000-3 standardı RFID okuyucu ve etiketler arasında 13.56Mhz'deki haberleşmeyi tarif eder. Modülasyon, çerçeveleme, girişim engelleme metotları, protokol parametreleri ve diğer sistem ile alakalı bilgiler hakkında sınırlar burada bildirilmiştir [2].

Bölüm 2.1'de de belirtildiği üzere etiket ve okuyucu arasındaki iletişim modülasyon ile sağlanmaktadır. Okuyucu ASK modülasyonunu %10 ve %100 indisi ile birlikte kullanmaktadır. Veri şifrelemesi "256da 1" ya da "4te 1" veri şifrelemesi ile bir bayt ya da iki bitin şifrelenmesi durumunda mümkündür. Çerçevelemenin gerçeklenmesi için veri sınır belirticileri Start-of-frame (SOF) ve End-of-frame (EOF) arasında

gönderilir. Bu da sınır belirticilerinin veriden farklı olmasını gerektirir. Etiket yük modülasyonu (Load Modulation) kullanarak yanıtını yollar [2].

Söz konusu iletişim protokolü talimatların ve verinin, etiket ve okuyucu cihazlar arasında çift yönlü olarak nasıl el değiştireceğini tanımlar. Bu protokol "okuyucu önce konuşur" metoduna uygun olarak okuyucu tarafından gönderilmiş ve uygun bir biçimde şifresi çözüşmüş bir talimat geri gelene kadar okuyucunun yayın yapmaya başlamayacağı anlamına gelmektedir. Her bir komut, okuyucu tarafından bir etikete yöneltilmiş talep ve okuyucuya verilmiş bir yanıttan oluşur. Talepler ve yanıtlar SOF ve EOF sınır kaldırıcıları içerisinde bir çerçeve içinde bulunmaktadır. Protokol bit-odaklıdır ve yayınlanan bit sayısı 8'in katıdır. Her bir talep ve yanıt şu alanlardan oluşur:

- Bayraklar: Bir ya da iki yan-taşıyıcı frekansını ve yanıt için hangi veri oranının kullanılacağını gösterir. Ek bilgi uygun görülen etiketleri adreslemek için gösterilir. Etiketin yanıtları, bayrakları iletim hatalarını belirtmek için kullanır.
- Komut Kodu: Hangi talebin gönderildiğini belirten bir baytlık sabit bir sayıdır. 3 ana komut tipi bulunmaktadır. Zorunlu komutlar etiketlerin içeriğinde bulunmalıdır. Seçmeli olanlar etiket tarafından uygulama için gerekli ise etikete dahil edilebilir. Özel komutlar üreticiler tarafından kendi komutlarını protokole eklemek istediklerinde kullanılabilir. Özel komutlar güvenlik katmanının açıklandığı Bölüm 5' te detaylı bir biçimde açıklanmıştır.
- Parametreler ve Veri Alanları: Bunlar sırasıyla talep ve yanıtı işlemek için gerekli bilgileri barındıran komutlara özel verilerdir.
- CRC (Cyclic Redundancy Check, Döngüsel Artıklık Denetimi): İletim hatalarının denetlenmesi için kullanılmaktadır. Bu işlemde SOF'den başlayarak CRC alanı hariç çerçeve içindeki tüm baytlar hesaplanır ve doğrulanır [2].

3. GELİŞMİŞ ŞİFRELEME STANDARDI

Gelişmiş şifreleme standardı (Advanced Encryption Standard - AES) veriyi 128 bitlik bölümler halinde şifreleyen bir şifreleme standardıdır. Şifrelerin 128, 192 ve 256 bit uzunluğunda olmasına göre bu şifreleme standardı AES-128, AES-192 ve AES-256 olarak üç çeşit olarak kullanılmaktadır [5]. Bu bitirme ödevinde; üzerinde gerçeklemenin yapıldığı MCU'nun hafıza sınırları sebebi ile AES-128 şifreleme standardı tercih edilmiştir.

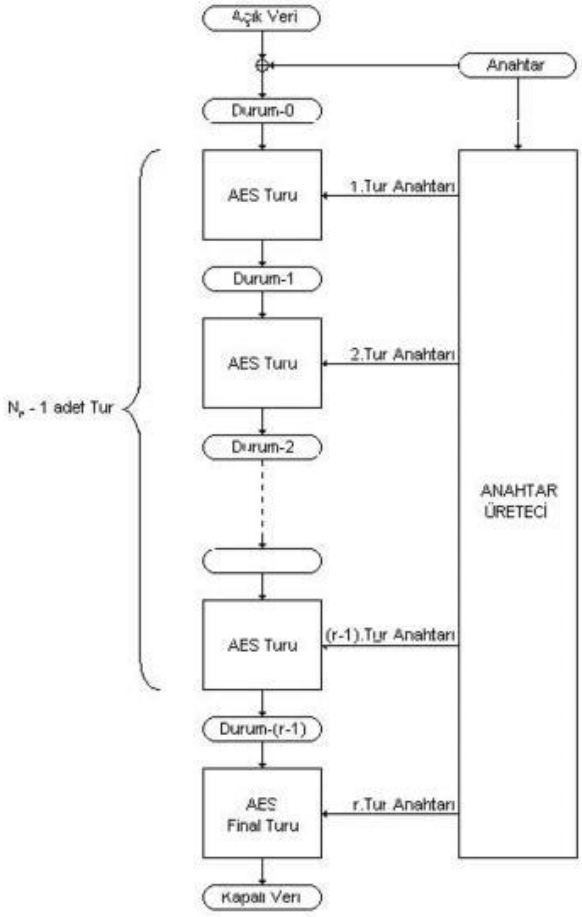
3.1 Gelişmiş Şifreleme Standardının Aşamaları

AES-128, 10 çevrimden oluşmaktadır. Başlangıç olarak 128 bitlik anahtar uygulanacak 10 çevrimde farklı biçimlerde kullanılması için genişletilir [5]. Bu aşamadan sonra Tur Anahtarı Ekleme (Add Round Key) adımı gerçekleştirilir. Bu aşamadan sonra bahsedilen 10 çevrimlik işlemler dizi başlatılır. Gerçekleştirilecek her çevrim birbirini takip edecek şekilde Bayt Değiştirme (Sub Bytes), Satırları Kaydırma (Shift Rows), Sütun Karıştırma (Mix Columns), Tur Anahtarı Ekleme (Add Round Key) işlemlerinden oluşmaktadır. Çevrimlerin sonuncusunda yani onuncu çevrimde Sütunları Karıştırma adımı işletilmez [6]. Söz konusu bu şifreleme algoritmasına ilişkin blok diyagramı Şekil 3.1' de verilmiştir [7]. Şekil 3.1' de AES Turu olarak belirtilmiş aşamaların iç yapısı ile alakalı blok diyagram ise Şekil 3.2'de belirtilmiştir.

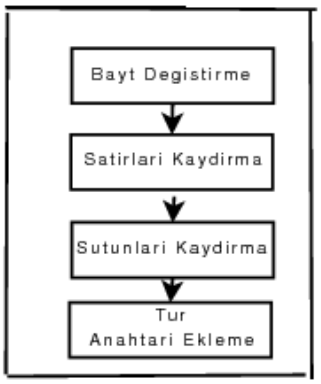
3.1.1. Bayt Değiştirme

Bayt değiştirme işlemi 8 bitlik veriler üzerinde gerçekleştirilen nonlineer bir işlemdir [8]. Bu aşamada ilk olarak 128 bitlik veri 8' er bitlik olmak üzere 16 parçaya bölünür. Elde edilen 16 parça ile Şekil 3.3' de de gösterildiği üzere 4x4 boyutundaki durum matrisi oluşturulur ve bu oluşturulan her parçaya matematiksel bir dönüşüm uygulanır. Söz konusu bu dönüşüm iki aşamadan oluşmaktadır. İlk aşama olarak indirgeme polinomu kullanılarak çarpmaya göre ters alma işlemi gerçekleştirilir. İkinci aşama olarak elde edilen sonuç geçiş matrisi olarak adlandırılan bir matrisle çarpılarak bayt değiştirme adımı sonucu elde edilir. Bu işlemler 8 bitten oluşan 16

elemanlı 4x4lük sütün matrisinin tüm elemanlarına uygulandığında 128 bitlik veri bu adımdan geçirilmiş olur [5].



Şekil 3.1 : AES blok diyagramı [8].



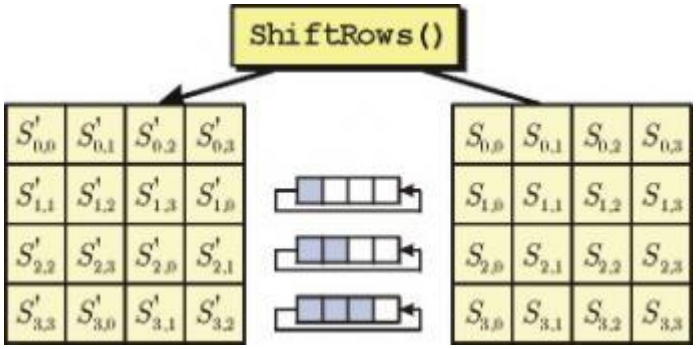
Şekil 3.2 : AES Turu İç Yapısı [8].

$s_{0,0}$	$s_{0,1}$	$s_{0,2}$	$s_{0,3}$
$s_{1,0}$	$s_{1,1}$	$s_{1,2}$	$s_{1,3}$
$s_{2,0}$	$s_{2,1}$	$s_{2,2}$	$s_{2,3}$
$s_{3,0}$	$s_{3,1}$	$s_{3,2}$	$s_{3,3}$

Şekil 3.3 : Durum Matrisi [8].

3.1.2. Satırları Kaydırma

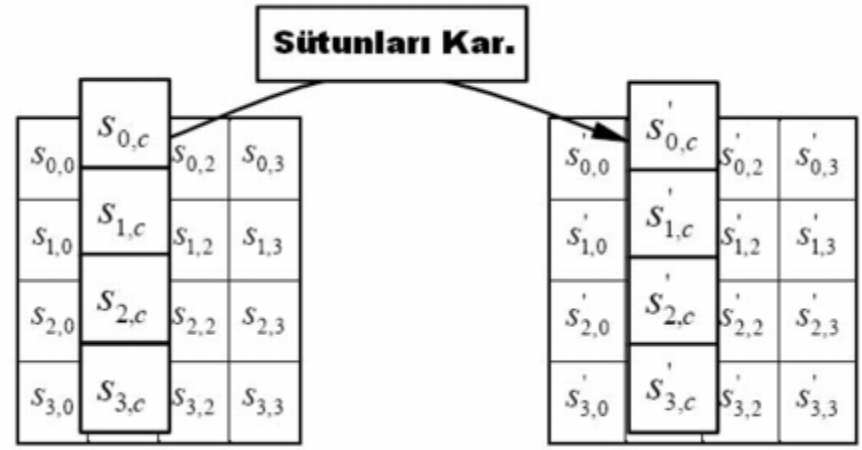
Satır Kaydırma adımında, Bayt Değiştirme işlemi ile elde edilmiş veri Bayt Değiştirme işleminde yapıldığı gibi yine 8'er bitlik 16 parçaya 4x4'lük boyutunda bir matris oluşturulacak şekilde bölünür. Oluşturulan matrisin ilk satırı sabit bırakılır. Ardından ikinci satır bir, üçüncü satır iki, dördüncü satır ise üç kere sola kaydırılır. 4x4lük matrisin elde edilen son hali Satır Kaydırma işleminin sonucunda edilmiş matristir. Bu elemanlar birleştirilerek 128 bitlik veri elde edilir. Söz konusu işlemlerin blok diyagramı Şekil 3.4'de gösterilmiştir [5].



Şekil 3.4 : Satırları Kaydırma İşlemi Blok Diyagramı

3.1.3. Sütun Karıştırma

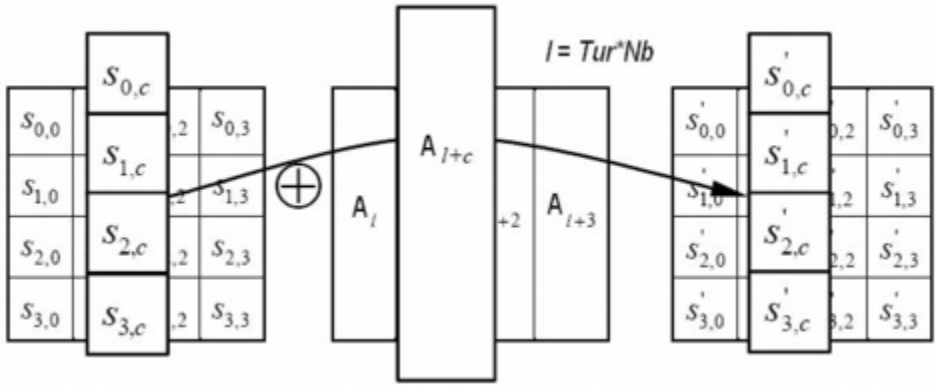
Bu adımda Satırları Kaydırma işlemi çıkışından alınan 4x4lük durum matrisinin her bir sütunu üzerinde bağımsız olarak gerçekleştirilir. Durum matrisindeki her bir sütun bir polinom olarak ele alınır. $A(x) = (03)x^3 + (01)x^2 + (02)$ polinomu ile modülo $x^4 + 1$'de çarpma işlemi gerçekleştirilir [8]. İşlemin blok diyagramı Şekil 3.5'te verilmiştir.



Şekil 3.5 : Sütun Kaydırma İşlemi Blok Diyagramı

3.1.4. Tur Anahtarı Ekleme

Bu aşamada Sütun Karıştırma işlemi ile elde edilmiş durum matrisi ile genişletilen anahtarın o çevrim ile alakalı bölümü olan 128 bitlik anahtar dizisi ile Özel Veya (XOR) işlemine sokulur. Bu aşama Şekil 3.6'da verilmiştir.



Şekil 3.6 : Tur Anahtarını Ekleme İşlemi Blok Diyagramı [8].

3.1.5. Gelişmiş Şifreleme Standardının Gerçeklenmesi

Bu bitirme çalışmasında AES gerçeklenmesi C dili kullanılarak yapılmıştır. Gerçekleme ortamı olarak Texas Instruments firmasının MS430G2352 mikrodenetleyicisi tercih edildiğinden ötürü üretici firmanın örnek olarak sunduğu AES uygulama örneğinden yararlanılmıştır. Yazılımın doğruluğu yapılan farklı anahtar ve giriş dizileri ile denenerek sorgulanmıştır. Gerçeklemenin kaynak kodu ektedir. Gerçekleme ile alakalı detaylı bilgi Bölüm 4'te verilmiştir.

4. DENEY DÜZENEĞİ

Bu çalışmada okuyucu ve etiket olmak üzere iki birim gerçeklenmiş ve bunların birbiri ile haberleşmesi sağlanmıştır. Bu bölümde okuyucu ve etiketin gerçeklenmesinde kullanılan düzenekler hakkında bilgi verilecektir. Okuyucu ve çekirdek için mikrodenetleyiciler, ve geliştirme kartlarıdır.

4.1. Kullanılan Mikrodenetleyiciler

Bu bitirme ödevinde gerçeklenmesi hedeflenen RFID doğrulama protokolü için okuyucu ve etiketi temsil edecek iki farklı mikrodenetleyici seçilmiştir. Sistemin kurulmasında, pasif etiket model olarak alındığından ötürü mikrodenetleyicinin seçilmesinde özellikle pasif etiketin özellikleri göz önünde bulundurulmuştur. Pasif etiketler oldukça düşük bir güç ile beslendiklerinden ötürü iletebileceği veri çok sınırlıdır. Düşük güçlü olması haberleşme menzilinin çok düşük olmasını da beraberinde getirir. Bu özelliğinden ötürü pasif etiketlerin içinde mikrodenetleyici gibi çok güç harcayan birimleri ancak bir kısıt çerçevesinden barındırması mümkündür. Bundan dolayı da pasif etiketler büyük miktarlarda veri toplamaya ve işletmeye yetecek kadar yüksek kapasiteli olamazlar [9]. Bu bahsedilen kısıtlardan ötürü etiketin benzetimi için 16 bitlik, düşük hafızalı ve yüksek seviyede gelişkin işlem kabiliyetleri bulunmayan Texas Instruments üreticisinin MSP430G2352 mikrodenetleyicisi tercih edilmiştir. Söz konusu mikrodenetleyicinin özelliklerine ayrıca Bölüm 4.1.1'de değinilecektir.

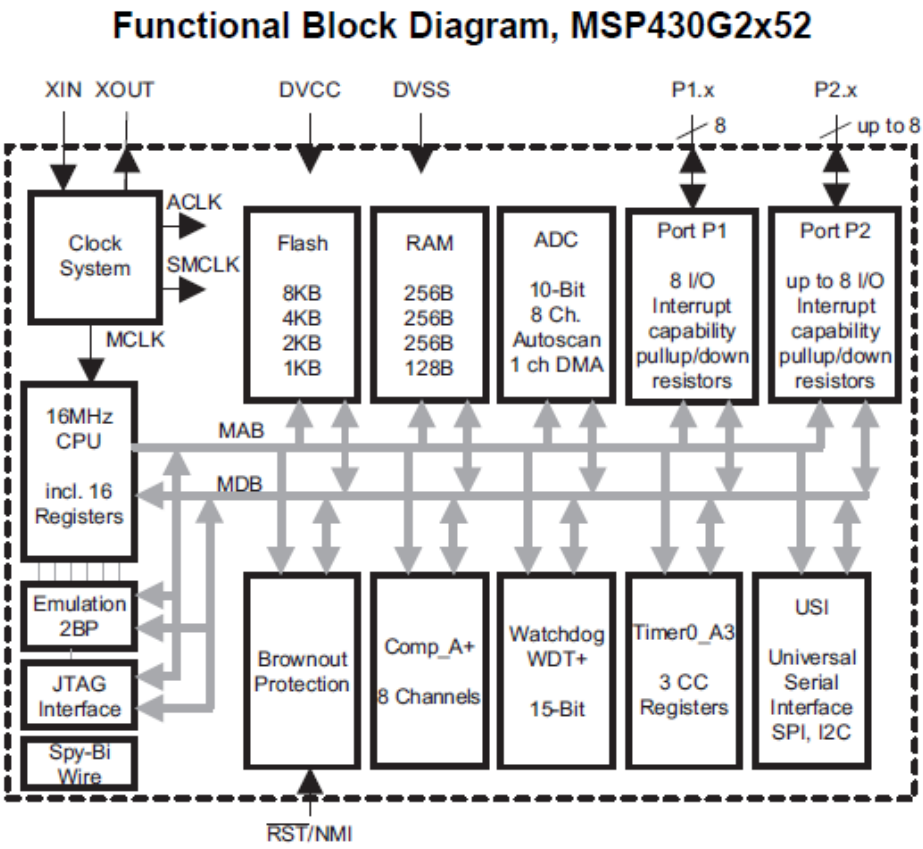
Okuyucu tarafında ise etiket tarafında olduğu gibi kısıtlar bulunmamaktadır, RFID uygulamalarının çoğunda okuyucu tarafına ID'lerin kayıt edildiği bir de sunucu bağlıdır. Bundan ötürü okuyucunun hafızasının yüksek olması istenir. Bunun yanısıra okuyucu yapısı aynı zamanda RFID uygulamasının kullanıldığı alanına göre bazı fonksiyonları tetiklemek ile yükümlüdür. Buna örnek olarak bir takım bilgileri gösterge paneline yansıtmak, ID uygunluğun kontrolünden sonra kapı kilidinin açılması ya da eşleşmenin sağlanamadığı bir durumda alarm sisteminin tetiklenmesi gösterilebilir. Bundan ötürü erişilebilen en gelişkin mikrodenetleyicilerden, ARM

Cortex-M3 çekirdekli STM32F103R6T6 mikrodenetleyicisi bu uygulama için tercih edilmiştir.

4.1.1. MSP430G2352 Mikrodenetleyicisi

MSP430G2352 mikrodenetleyicisi, Texas Instruments firmasının ürettiği ultra düşük güçlü, içerisinde sayısızca uygulama alanına hitap edebilmesine olanak sağlayan çevreselleri barındıran bir mikrodenetleyicidir. Bu mikrodenetleyici, beş farklı düşük güç ile çalışma modunu destekler. Mikrodenetleyicinin güç ihtiyacının düşük olması, sistemi beslemek için kullanılan kaynağın ömrünün uzun olmasını sağlar. İçerisinde 16 bitlik RISC (reduced instruction set computing, indirgenmiş komut takımı bilgisayarı) mimarisi ile çalışan bir işlemci bulunmaktadır. 16 bit olması aynı zamanda yazılım optimizasyonu sağlanmasına da imkân sağlamaktadır. Aynı zamanda içerdiği DCO (digitally-controlled oscillator, dijital kontrollü osilatör) sayesinde düşük güçlü uygulamalarda sıklıkla kullanılan uyku modundan çalışma moduna geçilmesi işlemi 1 μs'nin altında gerçekleşmektedir [10].

Bu mikrodenetleyicinin içinde başlıca çevresel anlamında 16 bitlik zamanlayıcı, 16 adet giriş/çıkış bacağı ve 10 bitlik analog/dijital çevirici bulunmaktadır. Hafıza elemanları olarak ise, içerisinde 4 KB'lık Flash ve 256B'lık RAM (Random – Access Memory) bulundurmaktadır. Söz konusu bu çevreseller ile alakalı fonksiyon blok diyagramı Şekil 4.1'de verilmiştir.

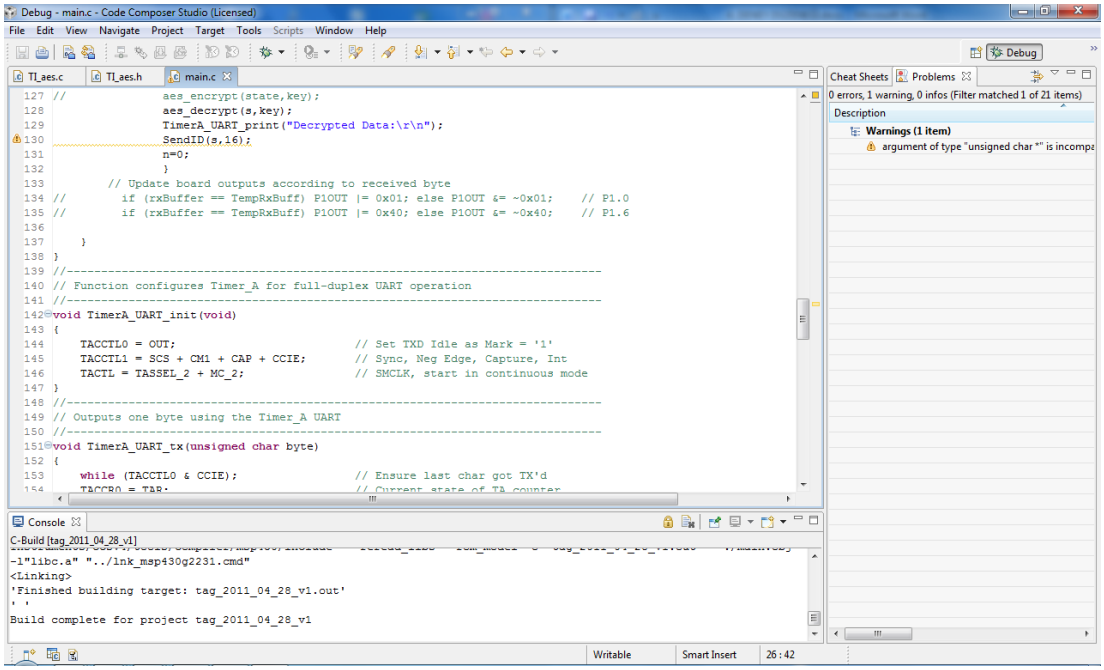


Şekil 4.1 : MSP430G2352 Fonksiyon Blok Diyagramı.

Bu bitirme tezinde etiket tarafının gerçeklenebilmesi için kullanılacak mikrodenetleyicinin barındırması gereken özellikler zamanlayıcı (TIMER) ve UART (Universal Asynchronous Receiver/Transmitter, Evrensel Eşzamansız Alıcı/Verici) modüllerinden ibarettir. Donanım yükünün hafifletilmesi istenen uygulamalarda UART modülü içermeyen mikrodenetleyicilerin tercih edilip bu işlevin zamanlayıcı (TIMER) modülü kullanılarak yazılım ile gerçeklenmesi tercih edilmektedir. Etiket benzetiminde de bu tercihe uyulmuştur. Sonuç olarak söz konusu mikrodenetleyicinin çevresel anlamında sadece zamanlayıcı modülünden faydalanılmıştır.

MSP430G2352 mikrodenetleyicisinin içerisine hazırlanan yazılımın gömülebilmesi için ise üretici firmanın önerdiği iki IDE (Integrated Development Enviroment) arasından kullanım kolaylığı göz önünde bulundurularak “Code Composer Studio v4 Core Edition” isimli tümleşik geliştirme ortamı tercih edilmiştir. Kullanılan bu IDE ile hem yazılım geliştirilmesine hem de mikrodenetleyicinin çalışması için içerisine gömülmesi gereken “.hex” dosyası oluşturulmuştur. Ek olarak yazılım geliştirilmesi sırasında ortaya çıkan hataların temizlenebilmesi de kullanılan bu IDE’nin debug

özelliğinin kullanıcı dostu olması projenin sürdürülmesinde kolaylık sağlamıştır. Söz konusu IDE ile alakalı ekran görüntüsü Şekil 4.2'de verilmiştir.



Şekil 4.2 : Code Composer Studio Ekran Görüntüsü.

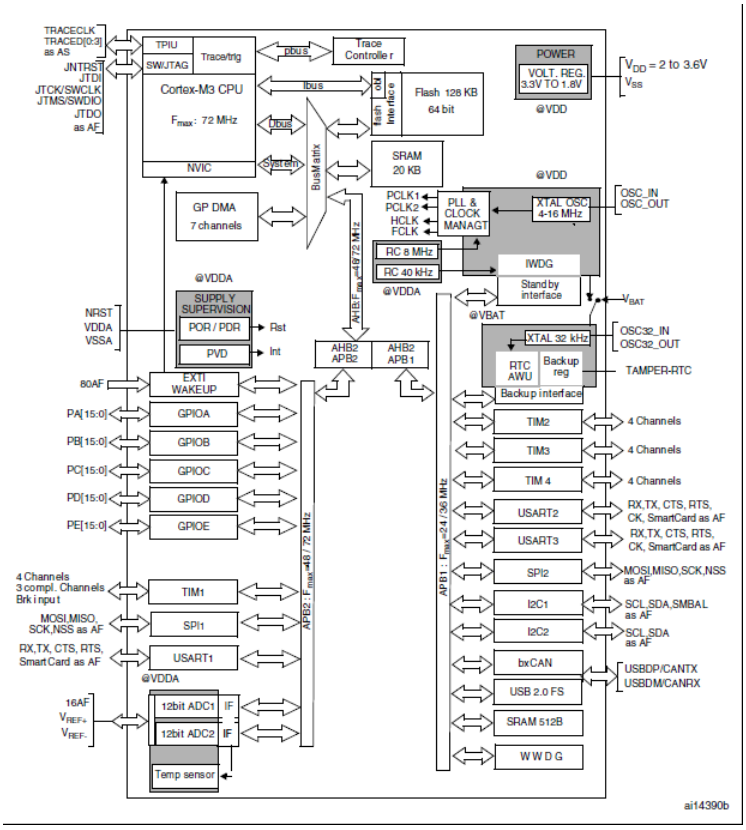
4.1.2. STM32F103R6T6 Mikrodenetleyicisi

RFID doğrulama protokolünün gerçeklenmesinde diğer önemli rolü üstlenecek okuyucu tarafının benzetiminin yapılabilmesi için ST firmasının STM32F103R6T6 kodlu mikrodenetleyici seçilmiştir. Bu mikrodenetleyici 32-bitlik RISC çekirdeğine sahiptir. 72Mhz'e kadar çalışma sağlayabilen bu MCU yüksek hızlı gömülü hafıza ve uygulamanın gerektirdiğin çok üzerinde bir sayıda giriş çıkış bacağını kullanıma sunmaktadır.

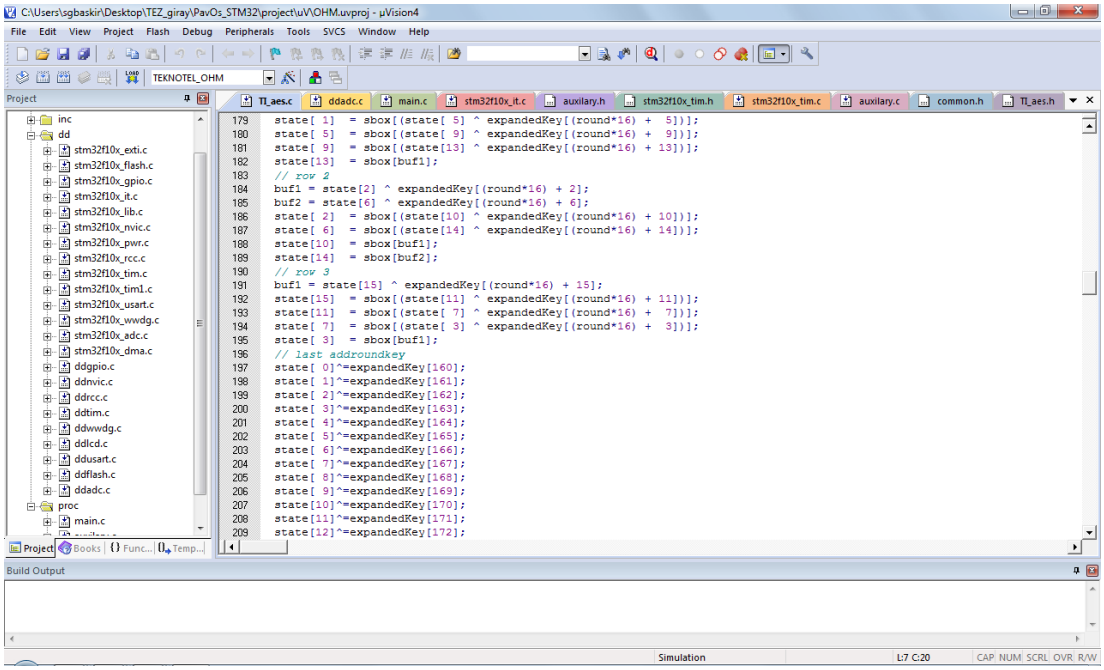
Sahip olduğu çevresel birimler anlamında ise bu MCU 16 bitlik zamanlayıcı modülü, 12 bitlik analog/dijital çevirici, PWM (Pulse Width Modulation) üreteci ve I^2C (Inter-Integrated Circuit), SPI (Serial Peripheral Interface), üç ayrı UART, USB (Universal Serial Bus) ve CAN (Controller Area Network) gibi günümüzde popüler olarak kullanılmakta olan haberleşme protokollerini de kullanıma sunmaktadır. Bunların yanında 64KB'lık Flash, 20KB'lık RAM'i ile de bir RFID uygulamasının gereksinimlerini fazlasıyla sağlayabilecek hafızaya sahiptir. Söz konusu bu çevreseller ile alakalı fonksiyon blok diyagramı Şekil 4.3'de verilmiştir.

Elektriksel olarak ise 2.0V ile 3.6V arasında sağlıklı çalışma sergileyebilen STM32F103 ailesinin bu üyesi -40 ile 85^{o}C gibi geniş bir çalışma sıcaklık aralığına sahiptir [11].

Yazılım geliştirilebilmesi ve hazırlanan yazılımın MCU içerisine gömülebilmesi için gerekli olan hex kodunun oluşturulabilmesi için ise "Keil uVision4" IDE'si kullanılmıştır. Bu IDE'nin seçilme sebebi ARM çekirdekli MCU'lar için yazılım geliştirilmesi amacı ile piyasada sıklıkla kullanılması ve lisans programı dahilinde yapılmış olan stajlarda edinilen tecrübenin bu IDE'nin tercih edilmesi durumunda yazılım geliştirilmesi sürecinde hız kazanılacağının düşünülmesidir. Söz konusu bu IDE'ye ilişkin ekran görüntüsü Şekil 4.4'te verilmiştir.



Şekil 4.3 : STM32F103xx Fonksiyon Blok Diyagramı



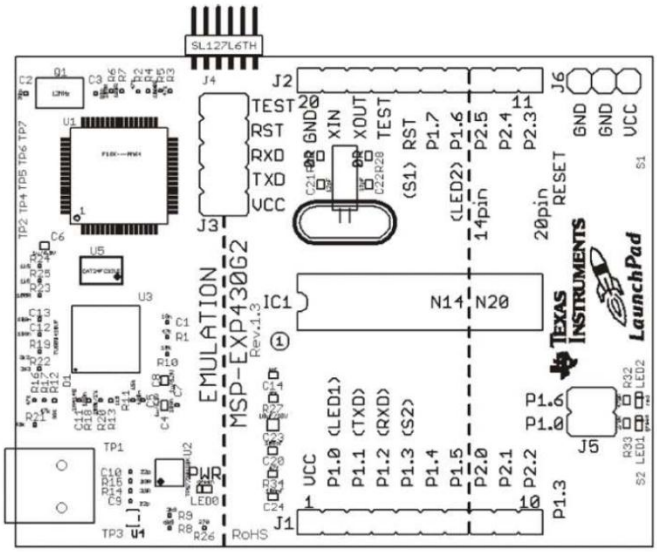
Şekil 4.4 : Keil uVision4 Ekran Görüntüsü

4.2. Kullanılan Kartlar

Bu çalışmada MSP430 Deneme Kiti ve STM32 Haberleşme ve Görüntüleme Modülü olmak üzere iki adet kart kullanılmıştır.

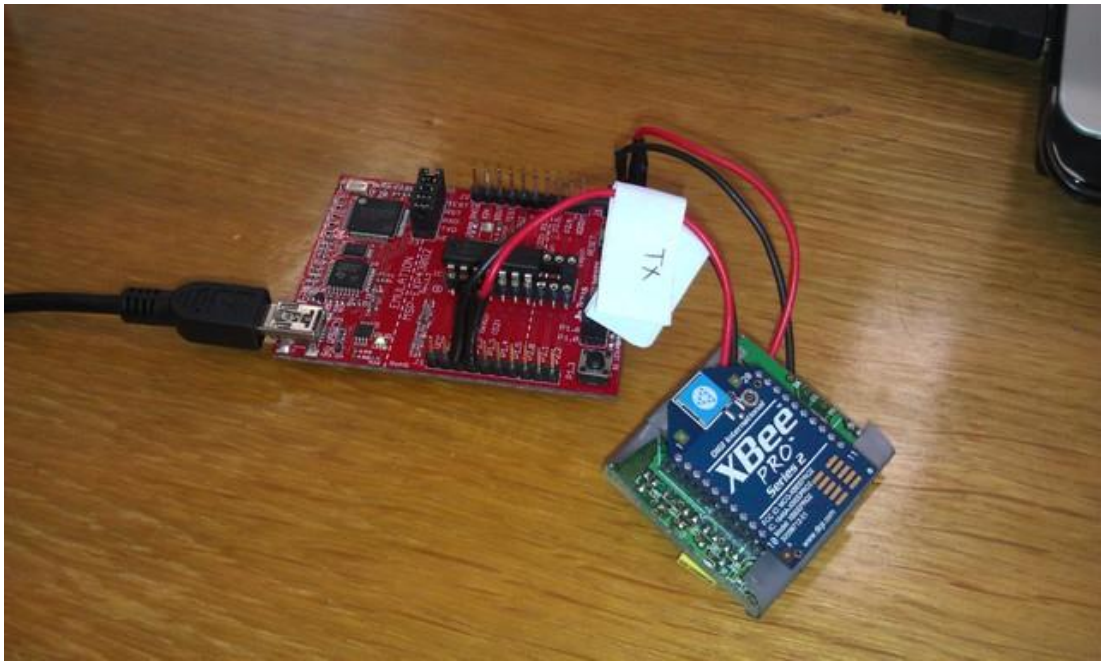
4.2.1. MSP430 Deneme Kiti

MSP430 LaunchPad olarak da adlandırılan bu kit hem MCU fonksiyonlarının rahatlıkla denenebilmesine imkân sağladığından hem program atma işinin yazılım geliştirilen bilgisayara bağlı bir USB üzerinden kolaylıkla yapılabilmesinden hem de ücretinin 5$ civarında olmasından ötürü tercih edilmiştir. Aynı zamanda söz konusu bu kit çok geniş yelpazedeki MSP430 MCU'larını problemsiz şekilde programlayabilip hata ayıklama özelliğine de sahiptir. Deneme kiti üzerine takılacak MCU'nun çevreselleri ile alakalı işlemlerin rahatlıkla gerçeklenebilmesi ve denenebilmesi için MCU'nın tüm bacaklarını birer pin ile dışarıdan erişilebilir kılmıştır. Etiket benzetimi uygulamasında da bu pinlerden faydalanılmıştır. Kitin üst görünümü Şekil 4.5'de verilmiştir.



Şekil 4.5 : MSP430 LaunchPad Üst Görünüm

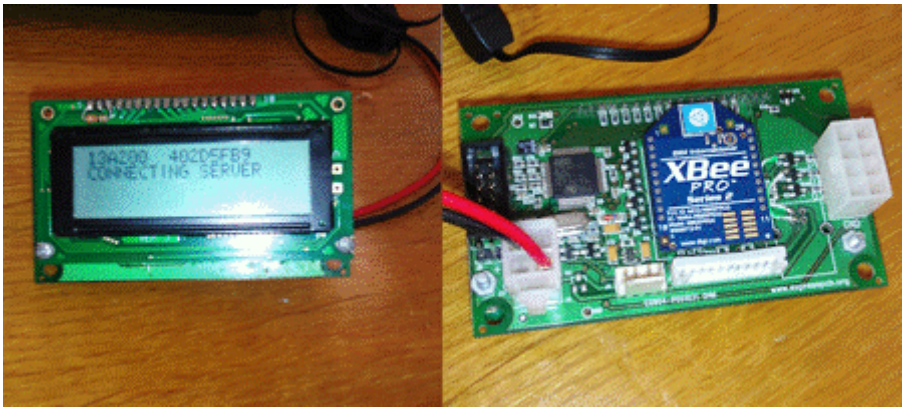
MSP430G2352 mikrodenetleyicisinin üzerine takıldığı bu kit önceki konularda da bahsedildiği gibi etiket benzetimi için kullanılmıştır. Bundan ötürü deneme kitinin dış dünyadan yalnızca UART RX, UART TX ve besleme bacaklarının erişilebilir olması yeterlidir. Bu bacaklara RF haberleşmenin kurulacağı Zigbee tabanlı Digi International firmasının Xbee XBP24-BCIT-004 ürünü takılmıştır. Kurulan yapının görüntüsü Şekil 4.6'de verildiği gibidir.



Şekil 4.6 : Etiket Benzetimi İçin Kullanılan Donanım.

4.2.2. STM32 Haberleşme ve Görüntüleme Modülü

STM32 Haberleşme ve Görüntüleme Modülü, Teknotel Ltd. Şti. firmasının oyun makinalarının birbiri ile haberleşmesi ve kullanıcı bilgilerinin görüntülenebilmesi için geliştirip ürettiği bir modüldür. Üzerinde hem okuyucu benzetimi için seçilmiş olan mikrodenetleyicinin bulunması, hem kullanılmak istenen RF modemin bulunması hem de görüntüleme ünitesinin bulunması sebebiyle okuyucu benzetimi için kullanılmasının uygun olacağı düşünülmüştür. Modül'ün ön tarafında görüntüleme ünitesi (Grafik LCD) arka tarafında ise MCU, RF modem ve diğer elemanlar bulunmaktadır. Devreyle alakalı resim Şekil 4.7'de verilmiştir.



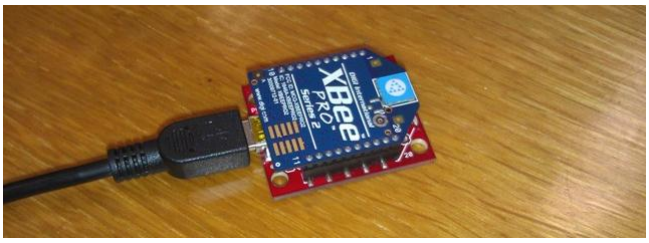
Şekil 4.7 : Haberleşme ve Görüntüleme Modülü.

5. DOĞRULAMA PROTOKOLÜNÜN GERÇEKLENMESİ

Doğrulama protokolünün gerçeklenebilmesi için öncelikle Bölüm 4'de de belirtildiği gibi benzetimi yapılacak etiket ve okuyucunun ayrı ayrı olacak biçimde devreleri hazırlanmıştır. Sonrasında benzetimi yapılacak uygulama için yazılım geliştirme aşaması gerçekleştirilmiştir. Aşağıda etiket ve okuyucu benzetimlerinin nasıl ve hangi aşamalar gerçekleştirilerek yapıldığı açıklanmıştır.

5.1. MSP430 LaunchPad ile Etiket Benzetimi

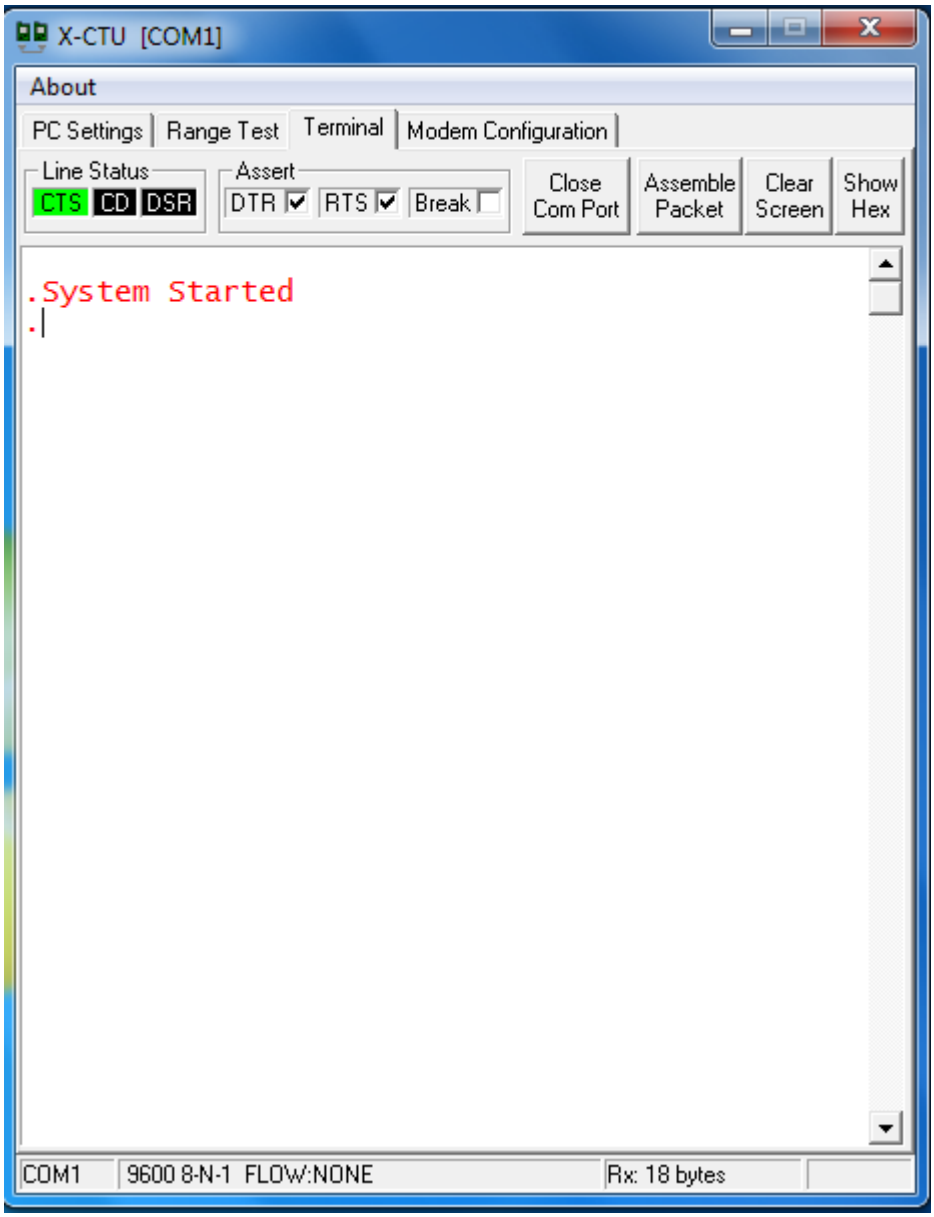
Bölüm 4.2.1'de anlatılan donanım ile etiket benzetiminin yapılabilmesi için öncelikle kullanılan MCU'nun istenen biçimde UART ile haberleştirilmesi sağlanmıştır. Bu haberleşme 9600 baudrate ile yapılmış ve haberleşme sırasında protokolün gerçeklenmesinde de olacağı gibi 16 byte'lık dizilerin gönderilmesi ve alınması gerçekleştirilmiştir. Bu haberleşme kullanılan kit ile bilgisayar arasında yapılmıştır. Bilgisayardan alınan ve gönderilen bilgilerin takip edilebilmesi için kit üzerinde kullanılan modem ve bilgisayar ara bağlantı ikilisi ve X-CTU isimli modem üreticisinin yayınladığı yazılım kullanılmıştır. "Xbee Exporer" ara bağlantı yapısı aslında bir UART TTL (Transistor Transistor Logic) USB çeviriciden ibarettir. Modemin ürettiği TTL çıkışları önce CMOS (Complementary Metal Oxide Semiconductor; Bütünleyici Metal Oksit Yarıiletken) seviyesindeki sinyale dönüştürür, sonrasında ise UART – USB dönüşünümü gerçekleştirir. Kullanılan cihazın görüntüsü Şekil 5.1'de verilmiştir.



Şekil 5.1 : Xbee Explorer.

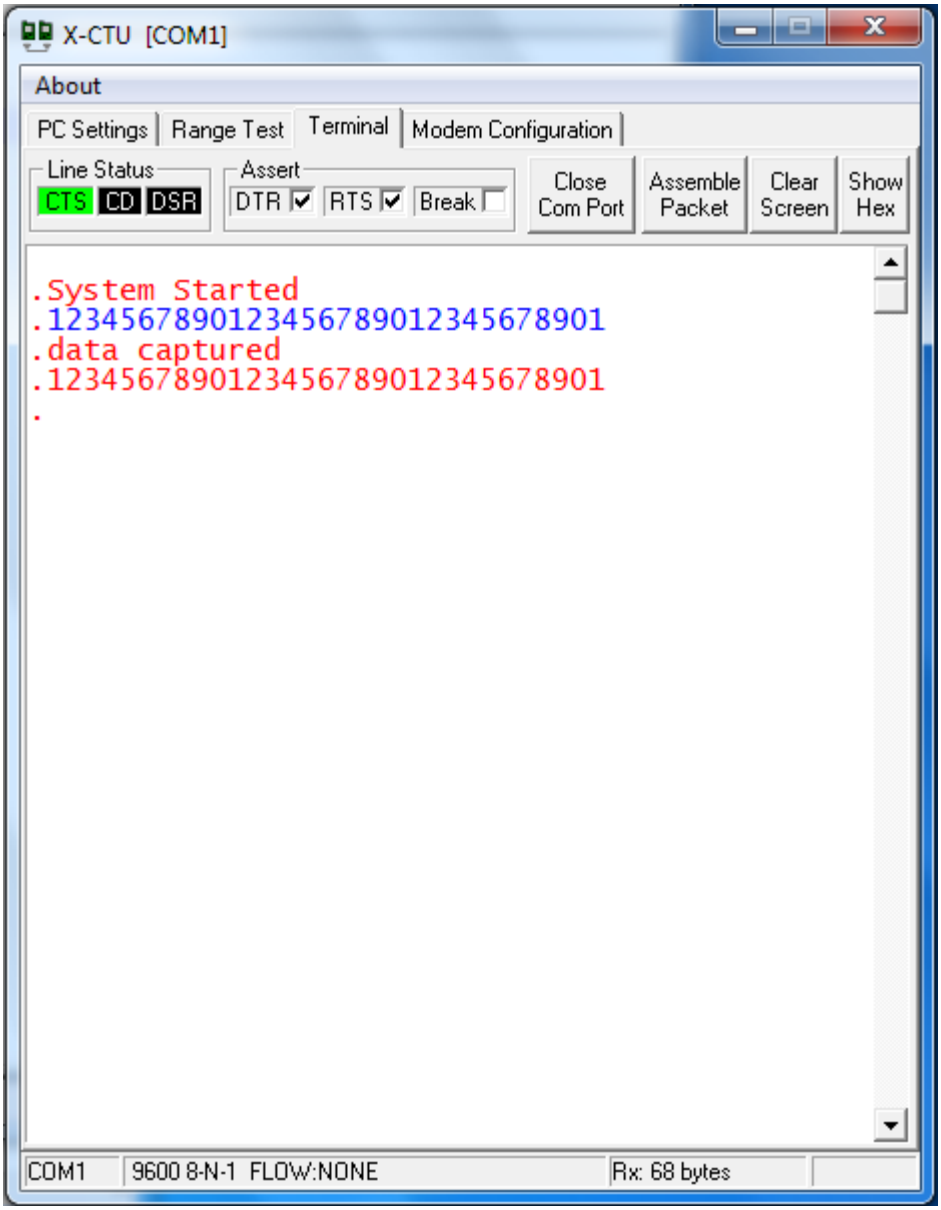
Alınan ve gönderilen bilgilerin takip edilmesi için kullanılan yazılım ise aslında bir Seri Port terminal programından farksızdır. Seri Port'a gelen ve seri porta gönderilen verilerin ham bir biçimde ekranda gösterilmesini sağlar. Bu program sayesinde

protokolün gerçeklenmesinden önceki aşamalarda hem etiket devresinden doğru nitelikte veri üretebilmesi aşamasında sayısız deneme yapılabilme imkanı oluşturulmuştur. Söz konusu yazılıma ait ekran görüntüsü Şekil 5.2'deki gibidir.



Şekil 5.2 : X-CTU Ekran Görüntüsü.

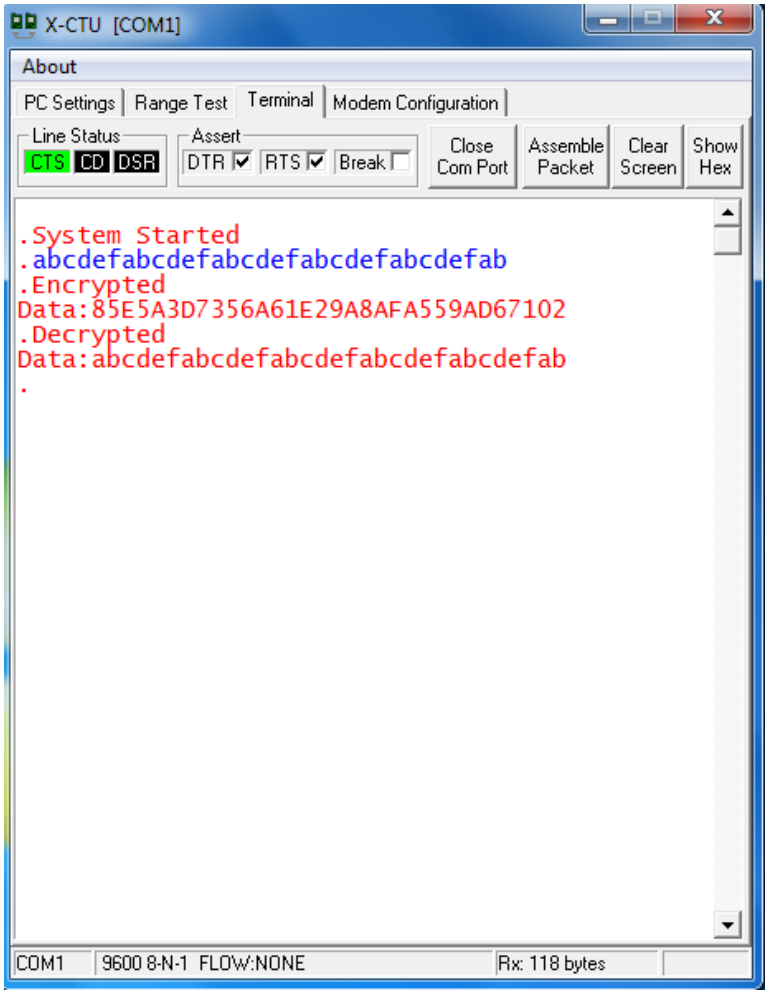
Donanım kurulduktan ve sağlıklı UART haberleşmesi sağlandıktan sonra ilk yapılan işlem protokolün gerçeklenmesi için yapılacak veri genişliğinde rastgele bir verinin etiket tarafından alınması yani bilgisayar tarafından gönderilmesi ve sonrasında da alınan verinin aynen geri gönderilmesi işlemi gerçekleştirilmiştir. Söz konusu veri genişliği 31 byte'tır. Gerçek uygulamada alınması ve gönderilmesi gereken verilerin içerikleri Bölüm 5.1.3'de belirtilecektir. Kurulan haberleşme sonrasında terminal programından alınan ekran görüntüsü Şekil 5.3'de verilmiştir.



Şekil 5.3 : 31 Byte'lık Verinin Alınması ve Aynen Gönderilmesi İle İlişkili Ekran Görüntüsü.

Gerçekleştirilen bu aşamadan sonrasında doğrulama protokolü gereğince etiket tarafında AES standardına uygun bir biçimde şifreleme ve şifre çözme işlemleri gerçekleştirilmiştir. Bu aşamanın denenebilmesi için öncelikle bilgisayarda çalışan terminal programı kullanılarak 32 byte'lık veri etiket'e gönderilmiş, etiket sırasıyla şifreleme ve şifre çözme işlemlerini gerçekleştirilmiş ve çözülmüş veriyi bilgisayara geri göndermiştir. Terminal programında gönderilen ve alınan verinin karşılaştırılması ve eş olduğunun gözlemlenmesi ile verinin etikete doğru formda ulaşmış olduğu, doğru bir şifreleme işleminin gerçekleştirildiği ve doğru bir şifre çözme işleminin gerçekleştirildiği gözlemlenmiş ve test edilmiş olmuştur. Söz konusu haberleşme ve şifreleme işlemi 100 defa olmak üzere çeşitli mesafeler uygulanarak tekrarlanmış ve

sistemin sağlıklı çalışma sergilediği gözlemlenmiştir. Gönderilen ve alınan 16 byte'lık veri ile ilişkili terminal programının ekran görüntüsü Şekil 5.4'de verilmiştir. Veri dizisi içerisinde bulunan her ikili bir hexadesimal sayının iki basamağına karşılık düşmektedir

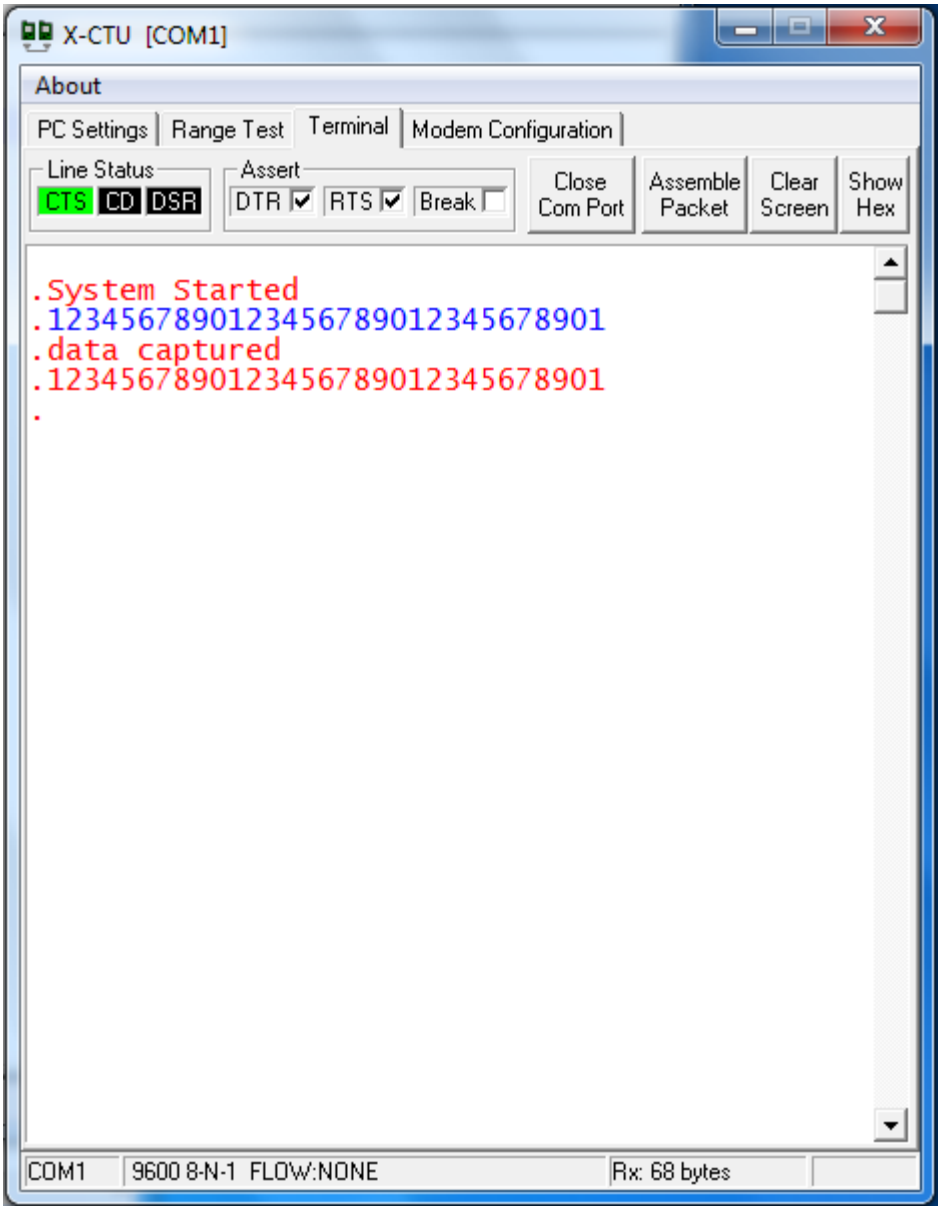


Şekil 5.4 : Terminal Programı İle Gönderilen 16 byte'lık Veri ve Etiket Tarafından Şifreleme Fonksiyonları Sonucu Oluşturulup Geri Gönderilen Verinin Karşılaştırılması.

5.2. STM32 Haberleşme ve Görüntüleme Modülü İle Okuyucu Benzetimi

Bölüm 4.1.3'te anlatılan donanım ile okuyucunun benzetiminin yapılabilmesi için etiket tarafında da olduğu gibi öncelikle sağlıklı UART haberleşmesinin sağlanabilmesi gerçekleştirilmiştir. Bu haberleşmenin gelecek aşamalarda okuyucu tarafı ile eşzamanlı çalışabilmesi için bu benzetimde de baudrate 9600 seçilmiştir. Bu haberleşme de öncelikle Bölüm 5.1.1'de bahsedilen bilgisayar ara bağlantı aygıtı ve modem üreticisinin sunduğu yazılım ile gerçekleştirilmiştir.

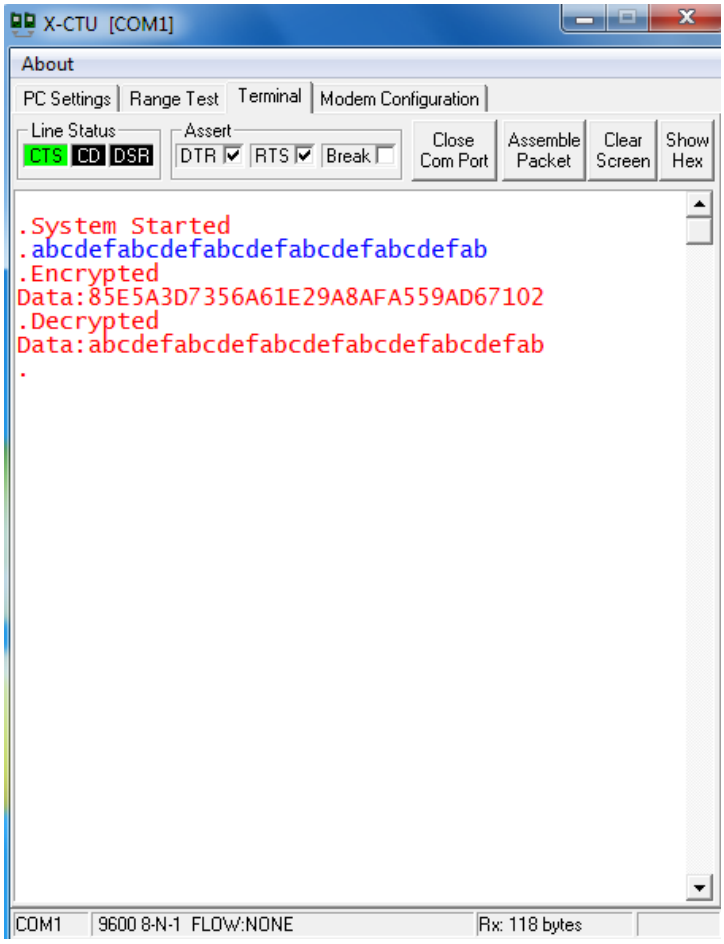
Donanım kurulduktan ve sağlıklı haberleşme sağlandıktan sonra yapılan işlem etiket tarafında da olduğu gibi 31byte'lık verinin gönderilmesi ve sonrasında da alınan verinin aynen geri gönderilmesi işlemi gerçekleştirilmiştir. Alınan çıktı etiket tarafından alınan çıktı ile eştir. Yapılan işleme dair ekran görüntüsü Şekil 5.5'te verilmiştir.



Şekil 5.5 : 31 Byte'lık Verinin Okuyucu Tarafından Gönderilmesi İle İlişkili Ekran Görüntüsü.

Gerçekleştirilen bu işlemden sonra aynı etiket tarafında da yapıldığı gibi AES standardına uygun bir biçimde şifreleme ve şifre çözme işlemlerinin doğru bir biçimde yapılması gerçekleştirilmiştir. Bu aşamada gerçekleştirilmesi istenen işlem AES fonksiyonlarının doğru bir biçimde gerçekleştirilmesi olduğundan ötürü veri

alma, şifreleme, şifre çözme, veri gönderme işlemleri etiket tarafında yapılan aynı sıralama ile yapılmıştır. Gönderilen veri ile alınan AES fonksiyon çıktılarının karşılaştırılması ve mesafe deneylerinde etiket tarafından yapılan deneyler taklit edilmiştir. Gönderilen ve alınan 16 byte'lık veri ile ilişkili terminal programının ekran görüntüsü Şekil 5.6'de verilmiştir.



Şekil 5.6 : Terminal Programı İle Gönderilen 16 byte'lık Veri ve Okuyucu Tarafından Şifreleme Fonksiyonları Sonucu Oluşturulup Geri Gönderilen Verinin Karşılaştırılması.

5.3. Cihazların Birbirleri İle Haberleştirilmesi

Etiket ve Okuyucu yapılarının ayrı ayrı bilgisayar ile haberleştirilmesinden sonra cihazların birbirleri ile haberleştirilmesi işleminin gerçekleştirilmiştir. Bu aşamada doğrulama protokolü olmaksızın AES standardına uygun şifreleme ve şifre çözme işlemleri uygulanmıştır. Yapılan deneylerde öncelikle okuyucu tarafında; iki tarafta da önceden birbirinin eşi olarak tanımlanmış 16 byte'lık anahtar ile 16 byte'lık rastgele bir mesaj şifrelenmesi ve RF modem ile havaya basılması hedeflenmektedir. Sonrasında etiket UART kesmesine girmeli ve veriyi bir değişkene yüklemelidir.

Değişkene alınan şifrelenmiş veri aynı anahtar ile çözülmeli ve RF modem ile tekrardan havaya basılmalıdır. Okuyucu bu veriyi yakalamalı ve şifreleme öncesinde oluşturduğu 16 byte'lık veri ile karşılaştırmalıdır. Karşılaştırma sonucunun eşit olup olmadığı incelenir. Karşılaştırma sonuncunun eşit olması durumunda okuyucu üzerindeki alarm yapısı aktif edilerek okuyucu ve etiket ikilisinde aynı anahtarın bulunduğunu birdirilir. Yapılan deneylerde karşılıklı haberleşmenin ve anahtar doğrulama işlemlerinin gerçekleştirildiği gözlemlenmiştir.

5.4. Doğrulama Protokolünün Gerçeklenmesi

Cihazların birbirleri ile haberleştirilmesi aşamasının gerçeklenmesinden sonra doğrulama protokolü sisteme aşağıda açıklandığı eklenmiştir.

Bu bitirme tezinde iki yönlü temel bir parola sorma – cevaplama (Challenge - Response) algoritması seçilmiştir. Bu protokolü ISO/IEC standardı içerisine oturtmak için Şekil 5.7 ve Şekil 5.8'de gözüken çerçeve formatları kullanılmıştır [4].

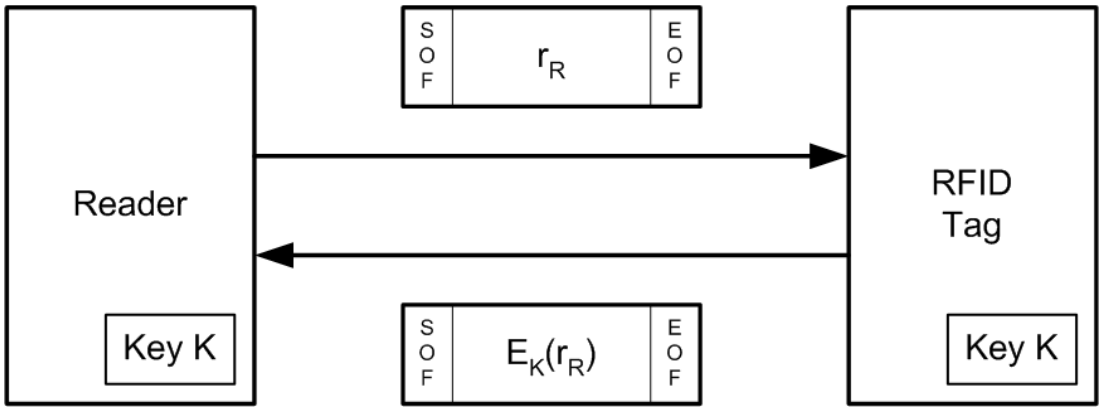
SOF	Flags	0xA0	IC Mfg code	UID	Random number r_R	CRC	EOF
	8 bit	8 bit	8 bit	64 bit	128 bit	16 bit	

Şekil 5.7 : Sorgulama Çerçevesi

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

Şekil 5.8 : Cevaplama Çerçevesi

Protokolün gerçeklenmesi sırasında öncelikle okuyucu tarafında 16 byte'lık rastgele bir sayı oluşturulmalıdır. Şekil 5.7'de verilen çerçeve formatı ile bu sayı etikete gönderilir. Etiket bu sayıyı AES standardına uygun bir biçimde şifreleyip Şekil 5.8'de verilen çerçeve formatı ile okuyucuya geri gönderir. Okuyucu şifrelenmiş veriyi çözer ve protokolün ilk aşamasında oluşturduğu sayı ile karşılaştırır. Eşitliğin sağlanması durumunda okuyucu için etiketin doğruluğu güvenilirdir. Söz konusu veri alışverişi ile alakalı akış diyagram Şekil 5.9'da verilmiştir.



Şekil 5.9 : Doğrulama Protokolüne İlişkin Veri Akış Diyagramı

Doğrulama protokolünün gerçeklenmesinde etiket tarafında oluşan hafıza yetersizliğinden ötürü bu aşamanın gerçeklenmesi MSP430G2352 mikrodenetleyicisi ile sağlanamamıştır. Bundan ötürü 8 KB Flash hafızalı yine aynı üreticinin MSP430G2452 mikrodenetleyicisine geçilmiştir. Bu durumda oluşan hafıza yetersizliğinin önüne geçilmiş ve protokolün etiket tarafında gerektirdiği fonksiyonların koşturulması sağlanmıştır.

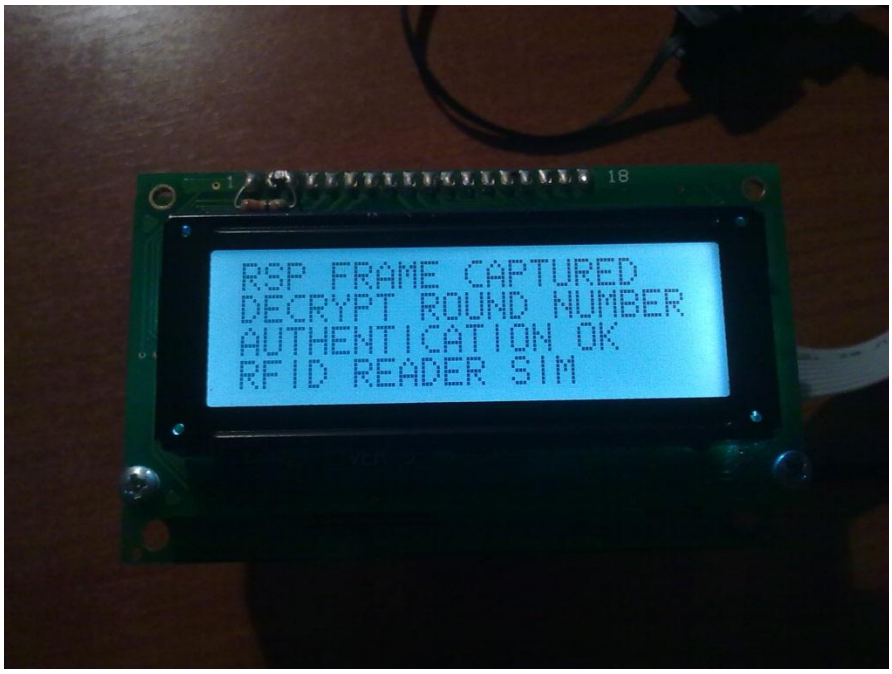
$$A \leftarrow B : r_B \tag{5.1a}$$

$$A \rightarrow B : E_K(r_B) \tag{5.1b}$$

Protokolün gerçeklenmesi denklem 5.1a ve 5.1b de verildiği üzere aşağıdaki adımlar ile yapılmıştır. Denklemde belirtilen A etiket, B ise okuyucu aygıtına işaret eder.

Öncelikle okuyucu, içerisindeki zamanlayıcı modülü vasıtası ile saydırdığı sayı ile rastgele bir sayı oluşturmuştur, sonrasında sorgulama penceresi yapısına uygun bir formatta etikete rastgele sayı içerikli çerçeve bütününü göndermiştir. Sonrasında ise etiket, okuyucu tarafından gönderilen sorgulama penceresi içerisinden 128bitlik rastgele sayıyı çekmiş ve önceden iki tarafa da yüklenmiş anahtar ile sayıyı şifrelemiştir. Bu aşamadan sonra etiket Şekil 5.8'de verilen çerçeve yapısı ile geri şifrelediği sayıyı okuyucuya geri göndermiştir. Okuyucu aldığı bu çerçeve içerisinden şifrelenmiş rastgele sayıyı çekmiş, şifre çözme işlemini gerçekleştirmiştir. Sonrasında kendisinin sorgulama penceresini oluştururken ürettiği rastgele sayı ile bu rastgele sayıyı karşılaştırmıştır. Karşılaştırmanın sonucunda okuyucu ekranında Şekil 5.10'da gözüken doğrulama tamamladı ifadesinin çıktığı durumlar okuyucu ve etiket ikilisinin aynı anahtara sahip olduğun sonucunun çıkartılmasını sağlar. Eğer okuyucu ekranında

bu ifade belirtilmez ise okuyucudan gönderilen ve geri alınan rastgele sayı birbirine eşit değildir bu da söz konusu ikilin aynı anahtara sahip olmadığı, doğrulamanın sağlanmadığı anlamına gelmektedir.



Şekil 5.10 : Doğrulama Protokolünün Sağlanması Durumunda Okuyucu Ekranı Görüntüsü

Protokolün sağlıklı bir biçimde çalıştırılmasından sonra AES fonksiyonları ve protokolün gerçeklenmesi için oluşturulan tablo ve dizilerden ötürü hazırlanan yazılımın boyutu 7311 byte olarak ölçülmüştür. Bu işlem için son aşamada seçildiği gibi 8KB'lık Flash hafızasına sahip olan bir mikrodenetleyicinin seçilmesi uygundur. Ancak gerçek uygulamada kullanılması hedeflenecek mikrodenetleyicinin içerebileceği hafıza miktarı çok daha sınırlı olması gerektiğinden yazılım optimizasyonuna gidilmesi gerekmektedir. Bu optimizisyon; farklı işlemler için aynı dizilerin birkaç defa kullanılması ile veyahut AES fonksiyonlarının koşturulması için gereken s-box isimli tablonun elemanlarının tablodan okunması ile her gerektiğinde üretilmesi ile gerçekleştirilebilir.

6. SONUÇLAR VE TARTIŞMA

Bu çalışmada bir RFID doğrulama protokolünün gerçeklenmesi hedeflenmiştir. Bunun için C programlama dilinde okuyucu ve etiket benzetimi için iki farklı yazılım geliştirilmiş, bu yazılım sırasıyla STM32F103R6T6 ve MSP430G2352 mikrodenetleyicileri üzerinde gerçeklenmiştir. Sonrasında bu iki mikrodenetleyicilerin üzerinde bulundukları donanımlar birbirleri ile haberleştirilmiş, AES fonksiyonları çalıştırılarak karşılıklı şifreleme ve şifre çözme işlemleri gerçekleştirilmiş, söz konusu AES fonksiyonlarının içersinde koşturulduğu rastgele bir sayı üzerine kurulmuş bir doğrulama protokolünün gerçeklenmesi yapılmıştır.

Çalışma neticesinde AES değişkenlerinin tablodan okuma yöntemi ile alınmasının ve protokolün gerçeklenmesi sırasında yapılması gereken çerçeve öğelerinin değerlendirilmesi işlemlerinin kayda değer büyüklükte hafıza elemanlarına ihtiyaç duyduğu tespit edilmiştir. Yüksek hafızalı mikrodenetleyicilerin pasif etiketler söz konusu olduğunda kullanılmasının fiziksel kısıtlardan ötürü kullanılmasının mümkün olmamasından ötürü oluşturulan prototipin gerçek uygulamaya dönüştürülebilmesi için etiket yazılımı üzerinde kayda değer bir hafıza tasarrufu sağlaması hedefiyle yazılım optimizasyonu yapılması gerektiği tespit edilmiştir.

KAYNAKLAR

[1] Finkenzeller, K., 2002. RFID-Handbuch., Carl Hanser Verlag München, third edition.

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

[3] Jeon, **J. ve diğ.,** 2007, "Digital Codec Design for RFID Tag Based on Cryptographic Authentication Protocol," *Future Generation Communication and Networking* (FGCN 2007) **(1)**, pp.119-124.

[4] 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**, pp. 759- 762.

[5] **FIPS 197,** 2001. Advanced Encryption Standard, *National Institute of Standarts and Technology*.

[6] Ferguson, N., Schneier B., 2003. Practical Cryptography, Wiley Pulishing, Inc., Indianapolis, Indiana.

[7] Topaloğlu, N., 2008, X86 Tabanlı Mikroişlemci Mimarisi ve Assembly Dili. 3. Baskı, Seçkin Yayıncılık, Türkiye.

[8] Doğan, A. H., 2006. AES Algoritmasının FPGA Üzerinde Düşük Güçlü Tasarımı, İstanbul Teknik Üniversitesi, *Yüksek Lisans Tezi*, İ.T.Ü Fen Bilimleri Enstitüsü, İstanbul.

[9] Kaan Bulut, 2008, Guvenli Radyo Frekansi ile Dogrulama (Radio Frequency Identification -RFID) sistemlerinin incelenmesi ve mikroislemci üzerinde güvenli olacak şekilde gerçeklenmesi, *Lisans Tezi*, İ.T.Ü Elektrik Elektronik Fakültesi, İstanbul.

[10] Msp430G2x52 veri yaprağı, 2008, Texas Instruments.

[11] STM32F103xx veri yaprağı, 2006, ST Microelectronics.

[12] MSP-EXP430G2 işlemci geliştirme kartı veri yaprağı, 2009, Texas Instruments.

ÖZGEÇMİŞ

Ad Soyad: Subutay Giray BAŞKIR

Doğum Yeri ve Tarihi: İstanbul, 1989

Adres: Eğitim Mah. Abdibey Sok. Kent Apt. No: 10/25 Ziverbey/İSTANBUL

Lisans: İstanbul Teknik Üniversitesi, Elektronik Mühendisliği - 2006