T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

FARKLI MİMARİLİ MİKRODENETLEYİCİLERDE HAFİF KRİPTOGRAFİ ALGORİTMALARININ YAZILIM TABANLI ANALİZİ

Rıfkı YARALI

YÜKSEK LİSANS TEZİ

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Elektronik Programı

Danışman

Doç. Dr. Umut Engin AYTEN

Şubat, 2022

T.C.

YILDIZ TEKNİK ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

FARKLI MİMARİLİ MİKRODENETLEYİCİLERDE HAFİF KRİPTOGRAFİ ALGORİTMALARININ YAZILIM TABANLI ANALİZİ

Rıfkı YARALI tarafından hazırlanan tez çalışması 17.02.2022 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Elektronik ve Haberleşme Mühendisliği Anabilim Dalı, Elektronik Tezli Yüksek Lisans Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Doç. Dr. Umut Engin AYTEN
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Doç. Dr. Umut Engin AYTEN, Danışman
Yıldız Teknik Üniversitesi _____

Doç. Dr. Nihan KAHRAMAN, Üye
Yıldız Teknik Üniversitesi _____

Dr. Öğr. Üye. Nurullah ÇALIK, Üye
İstanbul Medeniyet Üniversitesi _____

Danışmanım Doç. Dr. Umut Engin AYTEN sorumluluğunda tarafımca hazırlanan Farklı Mimarili Mikrodenetleyicilerde Hafif Kriptografi Algoritmalarının Yazılım Tabanlı Analizi başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Rıfkı YARALI

İmza

Aileme

ve

biricik eşime

TEŞEKKÜR

Tezimi hazırlık aşamamda ve tüm hayatım boyunca gerek maddi gerek manevi hiçbir desteğini esirgemeyen anneme, babama, kardeşlerime ve biricik eşime teşekkürlerimi sunarım.

Bu çalışmayı hazırlamam aşamasında yardım ve desteklerini esirgemeyen değerli danışmanım Doç. Dr. Umut Engin AYTEN'e teşekkürlerimi bir borç bilirim.

Rıfkı YARALI

İÇİNDEKİLER

SİMGE LİSTESİ

Hz	Hertz
GHz	GigaHertz
Kb	KiloBayt
KHz	KiloHertz
Mb	MegaByte
MHz	MegaHertz
µs	MicroSecond
mA	Miliamper
Ms	Milisaniye
uJ	Mikrojoul
V	Volt

KISALTMA LİSTESİ

AES	Gelişmiş Şifreleme Standartı-Advanced Encryption Standard
BAS	Yapı Otomasyon Sistemleri-Building Automation System
DES	Veri Şifreleme Standartı-Data Encryption Standard
GPRS	Genel Paket Radyo Servisi-General Packet Radio Service
GPS	Küresel Konumlandırma Sistemi-Global Positioning System
GSM	Küresel Mobil İletişim Sistemi
IoT	Nesnelerin İnterneti-Internet of Things
ITS	Akıllı Ulaşım Sistemleri-Intelligent Transportation Systems
KAA	Kablosuz Algılayıcı Ağları- Wireless Sensor Networks
RF	Radyo Frekansı-Radio Frequency
RFID	Radyo Frekansı Tanımlama-Radio Frequency Identification
UMTS	Küresel Mobil Telekomünikasyon Sistemi
Wi-Fi	Kablosuz Bağlantı Alanı-Wireless Fidelity

ŞEKİL LİSTESİ

TABLO LİSTESİ

ÖZET

Farklı Mimarili Mikrodenetleyicilerde Hafif Kriptografi Algoritmalarının Yazılım Tabanlı Analizi

Rıfkı YARALI

Elektronik ve Haberleşme Mühendisliği Anabilim Dalı

Yüksek Lisans Tezi

Danışman: Doç. Dr. Umut Engin AYTEN

Son yıllarda hızla yaygınlaşan Nesnelerin İnterneti teknolojisi ile gelişen kablosuz iletişim cihazları, kablosuz algılayıcı ağları, uzaktan kumanda sistemleri, RFID etiketleri gibi kısıtlı işlem yeteneğine sahip ve düşük enerji tüketmesi gereken uygulamalar gün geçtikçe yaygınlaşmaktadır. Bu uygulamalarda verinin gizliliği, ulaşılabilirliği ve bütünlüğünün sağlanması bilgi güvenliği açısından hayati önem taşımaktadır. Ancak bellek sınırlaması, düşük işlem hacmi ve az miktardaki güç tüketimi ihtiyacı yüzünden geleneksel yöntemlerin uygulanmasını zorlaştırmaktadır. Bu tür uygulamalarda güvenlikten ödün vermeden bilgi güvenliğini sağlanması için hafif yoğunluklu şifreleme yöntemleri önerilmiştir. Yapılan bu çalışmada, yaygın bir kullanım alanına sahip AES-128, Simon, Speck ve Twine simetrik blok şifreleme algoritmaları 8-bit RISC, 8-bit 8051 ve 32-bit ARM Cortex M0 mimarili mikrodenetleyicilerde yazılım tabanlı uygulaması gerçekleştirilmiştir. Oluşturulan donanımlar üzerinden her bir mikrodenetleyici farklı saat frekanslarında çalıştırılarak algoritmanın işlemci hafızasında kapladığı alan, işlem süreleri ve enerji tüketimleri test cihazları ile ölçülerek sunulmuştur. Elde edilen sonuçlar, yapılan benzer çalışmalar ile karşılaştırılmış ve hangi çalışma

şartlarında hangi mikrodenetleyicinin ve şifreleme algoritmasının kullanılabileceği belirtilmiştir.

Anahtar Kelimeler: AES, Simon, Speck, Twine, hafif yoğunluklu şifreleme

ABSTRACT

Software Based Analysis of Lightweight Cryptography Algorithms on Microcontrollers with Different Architectures

Rıfkı YARALI

Department of Electronics and Communication Engineering

Doctor of Philosophy Thesis

Supervisor: Assoc. Prof. Dr. Umut Engin AYTEN

In recent years, applications such as wireless communication devices, wireless sensor networks, remote control systems, RFID tags, which have developed rapidly with the Internet of Things (IoT) technology, are becoming widespread day by day. vital to safety. However, memory limitation, low throughput and low power consumption make it difficult to implement traditional methods. Lightweight encryption methods have been proposed to ensure information security without sacrificing security in such applications. In this study, software-based application of AES-128, Simon, Speck and Twine symmetric lightweight block cipher algorithms, which are widely used, on 8-bit RISC, 8-bit 8051 and 32-bit ARM Cortex M0 microcontrollers. Each microcontroller is run at different clock frequencies on the hardware created, and the area occupied by the algorithm in the processor memory, processing times and energy consumption are measured with test devices. The results obtained were compared with similar studies and it was stated which microcontroller and encryption algorithm could be used under which operating conditions.

Keywords: AES, Simon, Speck, Twine, microcontroller, light density encryption

1
GİRİŞ

1.1 Literatür Özeti

Günümüz dünyasında bir verinin gizliliği, ulaşılabilirliği ve bütünlüğünün sağlanması bilgi güvenliği açısından hayati önem taşımaktadır. Son yıllarda teknolojinin gelişmesi ve internetin mobil cihazlar sayesinde oldukça yaygınlaşmasından sonra özellikle e-ticaret siteleri, mobil bankacılık, kablosuz iletişim cihazları, kimlik tanıma ve uzaktan kumanda sistemleri ve gündelik hayatta kullandığımız her türlü cihazın internete bağlanmasını sağlayan Nesnelerin İnterneti (Internet of Things-IoT) teknolojisi gibi alanlarda bilgi güvenliği öncellikli konu haline gelmiştir.

Her türlü verinin depolanması, dağıtılması ve iletilmesi güvenilir yöntemler ile yapılmalıdır. Bilgisayar ağlarının ve sunucuların yaygınlaşması bilgiye erişimi kolaylaştırırken, bilginin eksiksiz bir şekilde iletimi ve üçüncü şahıslardan korunması önemli bir sorun haline gelmiştir. İki sistem arasında iletilen bilginin güvenli bir şekilde taşındığından emin olunması gerekmektedir. Bunun en güvenilir yolu gönderilen bilginin şifrelenmesidir. Örneğin, birine kısa mesaj ile ileti göndermek istendiğinde bilgi çeşitli donanımlardan geçerek karşı tarafa iletilir. Bu durumda verinin başkaları tarafından okunup okunmadığını ilemeyiz. Bu gibi durumlardan dolayı bilginin şifrelenmesini sağlayan kriptoloji önemli bir bilim dalı haline gelmiştir.

Kriptoloji bilimi, haberleşen taraflar arasında bilgi alışverişinin güvenli olarak gerçekleştirilmesini sağlayan, çoğunlukla temeli matematiksel ifadelere dayanan teknik ve uygulamaların bütünüdür. İletilecek olan bilginin belli bir sistematikte şifrelenerek alıcıya gönderilmesi ve alıcının şifreyi sağlıklı bir şekilde çözerek bilgiye ulaşması kriptolojinin temel amacıdır. Günümüz çağında ise kriptoloji demek güvenlik demektir. İnsanın güvenliği, ailesinin güvenliği, banka hesaplarının güvenliği, e-posta vb. sosyal hesapların güvenliği, milyar dolarlık şirketlerin güvenliği, devletlerin güvenliği demektir.

Son yıllarda kriptoloji biliminden bilgisayarlar başta olmak üzere telefonlar ve telsizler, modemler, RFID cihazlar, medikal ekipmanlar, uzaktan kumandalı sistemler, otomasyon cihazları, Radyo Frekansı (Radio Frequency-RF) kullanan iletişim cihazları gibi birçok teknoloji faydalanmaktadır. Bu cihazlar içerisinde yapı sektöründe kullanılacak ve hayatı kolaylaştıracak birçok ürün geliştirilmiştir. Bu ürünlere evlerde kullanılmaya başlanan panjur sistemleri, iş yerlerinde kepenklerin ve kapının otomatik çalışmasını sağlayan sistemler, site/kurumların girişlerindeki bariyer sistemleri örnek verilebilir. Çevremizdeki çeşitli cihazların birbirleriyle haberleştiği ve bir ağ üzerinden veri göndermelerine olanak sağlayan IoT teknolojisinin yaygınlaşmasıyla birlikte kriptoloji günden güne daha da önem kazanmaktadır. Ayrıca kablosuz cihazlarda son yıllarda hızla yaygınlaşması veri güvenliği alanında daha güvenilir, hızlı ve enerji tüketimi düşük yeni yöntem ve çözümler aranmaktadır.

Bu bağlamda sınırlı flash ve RAM hafızasına sahip, işlem yetenekleri bilgisayar, tablet gibi donanımlara kıyasla oldukça düşük ve enerjilerini genelde bir batarya üzerinden sağlaması gereken bu donanımlarda, verilerin güvenli şekilde iletilmesi ve depolanması için geçtiğimiz on yılda gelişimi hızlanan hafif kriptoloji algoritmaları tasarlanmıştır.

Hafif kriptoloji algoritmaları geleneksel yöntemler ile kıyaslandığında daha basit döngülere ve düşük işlem yoğunluklarına sahiptirler. Hafif blok kriptoloji algoritmaları genellikle 128 bit blok uzunluklarından daha kısa verileri yine 128 bit 'den daha kısa anahtar büyüklükleriyle şifrelerler.

Hafif kriptografi için sahip olduğumuz ana kısıtlamalar tipik olarak güç gereksinimleri, kullanılan kapı miktarı (GE) ve zamanlama ile ilgilidir. Örneğin pasif RFID bir etikete sistemi besleyecek bir güç kaynağı bulunmaz ve enerjisini radyo dalgalarından sağlaması gerekir. Bu nedenle, bir RFID cihazının zamanlama gereksinimleri ve kullanılan kapı sayısının sınırlandırılmasının yanında, herhangi bir şifreleme işleminde güç tüketiminde ciddi şekilde kısıtlanması muhtemeldir. Bir RFID cihazının bir pile sahip (aktif RFID) olsa bile, pili yeniden şarj etmek zor olabilir, bu nedenle güç tüketimi genellikle en aza indirilmelidir [1].

Hafif kriptografi, farklı iletişim teknolojileriyle hem donanım hem de yazılım katmanında uygulanabilmektedir. Hedef cihazların hafıza boyutu, işlem hızları ve

sınırlı enerji kaynakları nedeniyle standart kriptografi algoritmaları uygulamak zordur. Hafif kriptografi, kaynak sınırlı cihazlarda uygulama maliyeti, hızı, güvenliği, performansı ve enerji tüketimini optimum seviyede tutmayı amaçlar. Geleneksel kriptografiye kıyasla daha basit ve daha hızlı olması beklenir. Hafif kartografinin dezavantajı ise düşük anahtar boyutları kullandıkları için daha az güvenilir olmalarıdır.

Yazılım tabanlı uygulamalar için hafif kriptoloji algoritmalarının performansları genelde tükettikleri güç, işlemi gerçekleştirmek için geçen süre, algoritmanın bellekte kapladığı alan (ROM) ve yöntemin çalışması esnasında kullandığı hafıza (RAM) ile belirlenir.

Amerika Bileşik Devletleri Ulusal Standartlar ve Teknoloji Enstitüsünün (National Institute of Standards and Technology-NIST) onayladığı Gelişmiş Şifreleme Standardı (Advanced Encryption Standard-AES) göre performans avantajları elde etmek için bir dizi hafif blok şifreleme önerilmiştir. Bu şifrelerden bazıları geleneksel algoritmaların sadeleştirerek geliştirilen türevleridir. Örnek olarak DESL, Veri Şifreleme Standardının (Data Encrypt System-DES) sadeleştirilerek performansı arttırılmış bir çeşididir. Sekizli döngü fonksiyonu yerine tek bir S-kutusu kullanır. Diğer yandan alternatif olarak birçok blok şifreleme algoritması sıfırdan tasarlanmıştır. PRESENT, SIMON, SPECK, TWINE, CLEFIA ve PICCOLO bu algoritmalardan birkaçıdır [2]. Yaygın kullanılan hafif blok şifreleme algoritmaları bir listesi [3]'de verilmiştir.

Hafif kriptografinin hedef uygulamaları arasında gömülü sistemler önemli bir yer kaplamaktadır. Gömülü sistemlerde genellikle 8-bit, 16-bit ve 32-bit mikrodenetleyiciler kullanılmaktadır. Bu donanımlarda gerçek zamanlı ihtiyaçları karşılamakta geleneksel kriptolojiler zorlanmaktadır. Kaldı ki ilk üretiminden uzun süreler geçse de 4-bit işlemciler hala güçlü bir pazara sahiplerdir. Özellikle sensör ağları ve Radyo Frekansı Tanımlamalı (Radio Frequency Identification-RFID) cihazları sınırlı alan ve yeteneğe sahiptir. Enerji kapasiteleri ve tüketimleri de oldukça kısıtlıdır. Bundan dolayı yapılan çalışmalarda hafif şifreleme algoritmalarının mikrodenetleyicilerde yazılım veya donanım tabanlı analizleri geniş yer tutmaktadır. Analizlerde kullanılan mikrodenetleyici mimarilerinin farklı olması, elde edilen analiz sonuçlarının da farklılaşmasına neden olmaktadır.

Literatürde hafif şifreleme algoritmaları üzerine yapılan birçok araştırma bulunmaktadır. Bu çalışmalardan Hafif Kriptografik Sistemlerin Adil Değerlendirmesi (Fair Evaluation of Lightweight Cryptographic Systems- FELICS) kavramı [4]'daki çalışmada sunulmuştur. FELICS, hafif akış ve blok şifreleme algoritmalarının birbirleriyle kıyaslanması için işlem süresi, RAM kullanımı, verim gibi performanslarını incelemektedir. Yöntem donanım olarak dünyada yaygın olarak kullanılan 8-bit AVR, 16-bit MPS ve 32-bit ARM mikrodenetleyicileri kullanılmaktadır. Bu yöntem sayesinde hafif şifreleme algoritmalarının şeffaf ve güvenilir bir karşılaştırılmasının yapılması için ölçütler belirlenmiştir.

[5]'da sunulan çalışmada birçok hafif şifreleme algoritmasının karşılaştırılması yapılmıştır. Araştırmaya konu olan algoritmalar ATmega 8-bit AVR mikrodenetleyicisi üzerinde uygulanarak şifreleme ve deşifreleme süreleri, kod boyutları, verimleri sunulmuştur. Elde edilen sonuçlar benzer çalışmalar ile kıyaslanmıştır.

[6]'de yapılan araştırmada yüzden fazla hafif kriptoloji algoritması incelenmiştir. Çalışmada ele alınan algoritmalar için yazılım tabanlı FELICS sonuçları paylaşılmıştır. NIST de hafif şifreleme üzerine yaptığı çalışmaları ve hafif şifreleme algoritmalarının standardizasyonuna yönelik planlarını açıkladığı bir rapor yayınlamıştır [2]. Ayrıca [7]'de yapılan çalışmada da şifreleme algoritmaları farklı özelliklere sahip gömülü sistem donanımlarında yazılım tabanlı uygulanmıştır. Performansları işlem süreleri ve bellekte kapladıkları alan açısından analiz edilmiştir.

Hafif şifreleme algoritmalarının hedef cihazları sınırlı kaynaklara sahip, genelde bir batarya ile beslenen gömülü sistemler olduğu için enerji tüketimi de önemli bir konudur. [8]'de yapılan araştırmada verilerin güvenliğini sağlamak için kullanılabilecek PRESENT, CLEFIA, PICCOLO, PRINCE ve LBLOCK hafif şifreleme algoritmalarının Texas Instruments (TI) firması tarafından üretilen 16-bit RISC mimarili MSP430FR5994 mikrodenetleyicisi üzerinde analizleri yapılmıştır. Testler sonucunda algoritmaların enerji ve akım tüketimi bir bilgisayar programı yardımıyla hesaplanmış ve buna dayalı olarak bazı çıkarımlarda bulunulmuştur.

1.2 Tezin Amacı

Bu tezin amacı, düşük işlem yeteneğine sahip mikrodenetleyiciler ile çeşitli hafif kriptoloji algoritmalarının yazılım tabanlı analizini yaparak, oluşturulan donanımlar üzerinde algoritmaların bellek kullanımı, işlem hızları, verimleri ve enerji tüketimlerini ölçümleyerek başarım sonuçlarının karşılaştırılmasıdır. Böylece algoritmalar farklı mimarilere ve çalışma frekanslarına sahip mikrodenetleyicilerde uygulanarak her bir durum için performans kriterleri ölçü aletleri yardımıyla analiz edilecektir. Elde edilen sonuçlar yapılan benzer çalışmalar ile karşılaştırılacaktır. Değişen çalışma koşulları altında hangi hafif kriptoloji algoritmalarının ve hangi mikrodenetleyicinin kullanılabileceği belirlenecektir.

1.3 Hipotez

Hafif şifreleme algoritmaları verilerin şifrelenmesi ve deşifrelenmesi işlemleri için mikrodenetleyici ve bellek (Flash ve RAM) kullanımları farklılık göstermektedir. Bunun yanında aynı uzunluktaki veriyi eşit uzunluktaki anahtarlar ile şifreli metin haline getirmek veya şifreli bir metinden açık metin elde etmek için de farklı sürelere ihtiyaç duyulmaktadır. Bu işlemleri gerçekleştirirken tükettikleri enerji değişik mimarilere sahip mikrodenetleyicilerin bazılarında daha azken bazılarında daha yüksek olmaktadır. Ayrıca farklı çalışma frekansları ile harcanan enerji, işlem süreleri ve verim değişmektedir.

Bu tez çalışmasının hipotezi de sistemden istenen performans kriterlerine uygun olan mikrodenetleyicinin ve çalışma frekansının belirlenebileceği ve karşılaştırmalı olarak verilebileceğidir.

2
GENEL BİLGİLER

Bu bölümde tez çalışmasında kullanılacak temel kavramlar olan bilgi güvenliği, IoT, kriptoloji bilimi ve tarihsel gelişimi, kablosuz iletişim türleri ve gömülü sistem kavramlarından kısaca bahsedilecektir.

2.1 Bilgi Güvenliği

Modern dünyamızda bilgiye sürekli erişim sağlanması, verinin göndericiden alıcıya kadar gizlilik içinde, değişime ve bozulmaya uğramadan, başkaları tarafından ele geçirilmeden bütünlük içerisinde güvenilir bir şekilde alıcıya iletimi bilgi güvenliği olarak tanımlanabilir. Bilgi güvenliği genel olarak üç bileşenden oluşur [9]:

- Gizlilik: yalnızca yetki tanınan kişilerce verilere erişim sağlanmasını ve yetkisiz erişime karşı korunmasını ifade eder. Örneğin banka kayıtları ele alındığında, yetkisi olmayan başka hiç kimsenin erişmemesi gerekir. Bilgilere erişim iznine sahip olmayan bir kişinin bilerek veya kaza yoluyla bilgileri edinmesi gizliliğin sağlanamadığı anlamına gelir. Gizliliğin etkin olarak sağlanmasında birçok şifreleme algoritmaları kullanılır.
- Bütünlük: bir bilginin iletimi boyunca yetkisiz kişiler tarafından değiştirilmemesi ve kaynağından başlayarak güvence altına alınmasıdır.
- Kullanılabilirlik: bilgilere yetkili kullanıcılar tarafından erişilebileceği anlamına gelir.

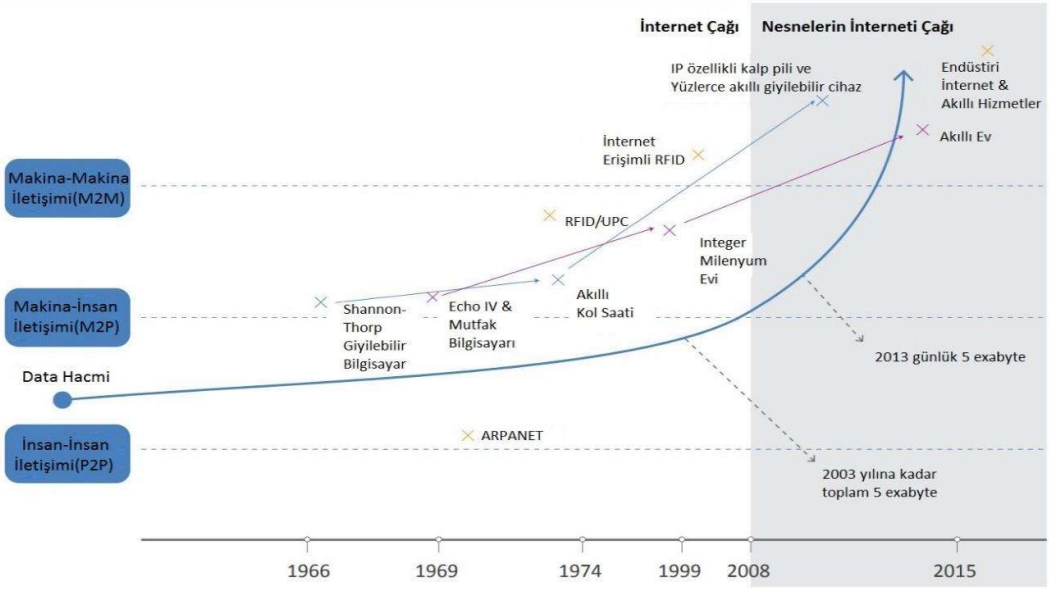
2.2 Nesnelerin İnterneti Teknolojisi (IoT)

IoT, her bir cihazın başka bir cihaz ile belirli bir amaç için iletişim kurduğu bir bilgi işlem konseptidir. İnternet üzerinden bağlanıldığında veri alışverişi yapabilen her bir nesneye IoT cihazı denir. İletişim için internete bağlanamayan herhangi bir nesne, IoT ağının bir parçası değildir ve ayrıca IoT cihazı olarak sınıflandırılmaz.

IoT, dünyanın herhangi bir yerinden uzak bir yerden İnternet kullanımı yoluyla fiziksel cihazlar arasında etkileşim ve tam erişim kontrolü sağlamak için kullanılan terimdir [10]. IoT, katılımcı kuruluşların cihaza özel yerleşik yazılım, algılayıcılar

ve ağ destek bileşenleri ile donatılmasını gerekmektedir.

Cihaza özel görevler gömülü yazılım tarafından sağlanmaktadır. Fiziksel cihaz algılayıcıları, çevredeki diğer fiziksel varlıkları algılar ve gerekli bilgileri toplar.



Şekil 2.1 Son yıllarda IoT cihazlardaki gelişim eğrisi [11]

İnternet, farklı fiziksel cihaz türleri arasında bir iletişim aracı görevi görerek bu cihazların uzaktan yönetimini ve erişim kontrolünü sağlamaktadır [10]. Şekil 2.1'de IoT teknolojisinin insan-insan iletişiminden makina-makina iletişimine evirilme eğrisi verilmiştir [11].

IoT teknolojisinin genel kullanım alanları akıllı ev teknolojileri, alışveriş sektörü, endüstride kullanılan sistemler, giyilebilir teknolojiler, şehir içi konum belirleme uygulamaları, medikal uygulamalar ve otomobil sistemleri olarak sıralanmaktadır [12]. Şekil 2.2'de genel uygulama alanları gösterilmiştir.

Şekil 2.2 IoT teknolojisinin genel kullanım alanları [13]

Akıllı binalar, yapı otomasyon sistemlerini (Building Automation Systems-BAS) internete bağlar. BAS, aydınlatma ve gölgeleme, güvenlik, eğlence vb. gibi algılayıcıları ve aktüatörleri kullanarak farklı bina cihazlarının kontrol edilmesine ve yönetilmesini sağlar. Ayrıca BAS, binaların enerji tüketimini, azaltmaya yardımcı olabilir. Örneğin, bir bulaşık makinesi veya soğutma / ısıtma sistemi gibi sistemler bilgi ekran aracılığıyla takip edilebilir. Böylece arıza veya bakım talepleri, herhangi bir insan müdahalesi olmadan sözleşmeli bir şirkete gönderilebilir.

Akıllı ev IoT hizmetleri, ev aletlerini ve sistemlerini (örn. klima, ısıtma sistemleri, enerji tüketim sayaçları, vb.) uzaktan izlemeyi ve çalıştırmayı daha kolay hale getirerek kişisel yaşam konforunun artırılmasına katkıda bulunur. Örneğin, akıllı bir ev hava tahminlerine göre pencereleri otomatik olarak kapatabilir veya soğuk havalarda evin ısınmasını sağlayabilir.

Akıllı ulaşım sistemleri (Intelligent Transportation Systems-ITS) ulaşım ağını izlemek ve kontrol etmek için kullanılır. ITS, ulaşım altyapısında daha iyi güvenilirlik, verimlilik, kullanılabilirlik ve güvenlik sağlamayı amaçlamaktadır. ITS dört ana bileşenden oluşur: araç alt sistemi (GPS, RFID okuyucu, OBU ve iletişimden oluşur),

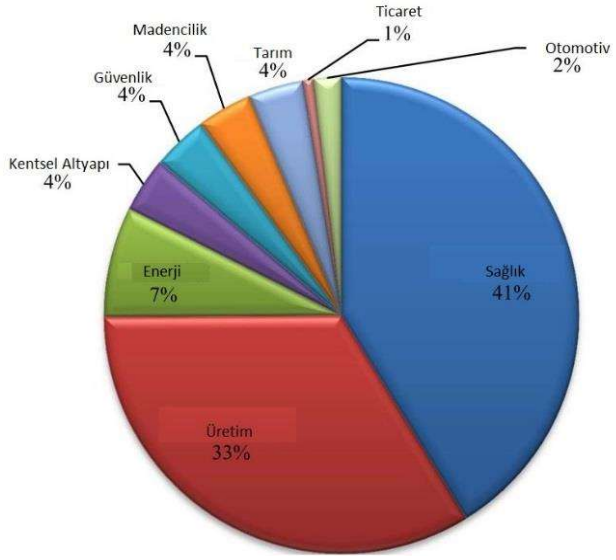
istasyon alt sistemi (yol kenarı donanımı), ITS izleme merkezi ve güvenlik alt sistemi. Ayrıca bu sistemler, sürüşü daha güvenilir, keyifli ve verimli hale getirmek amacıyla önem kazanmaktadır.

Endüstriyel otomasyon üretim süreçlerini minimum insan katılımıyla tamamlamak için robotik cihazları bilgisayarlaştırmaktadır. IoT, endüstriyel otomasyonda üretim makinelerinin işlemlerini, işlevlerini ve üretkenlik oranını İnternet üzerinden kontrol etmek ve izlemek için kullanılır.

Akıllı sağlık hizmetleri, hastaların sağlık durumlarının algılayıcılar aracılığıyla izlenebilmesi ve takibinin yapılabilmesine olanak sağlamaktadır. IoT, hastaların verilerini toplamak ve analiz edilen verileri uygun aksiyonu almak için işlem merkezlerine uzaktan gönderip hastaların fizyolojik durumlarını izlemek için kullanılır.

Akıllı şebekeler, IoT'yi evlerin ve binaların enerji tüketimini takip etmek ve enerji tasarruflarını arttırmak için kullanmaktadır. IoT'yi akıllı şebekelerde kullanmak, elektrik dağıtım şirketlerinin nüfus artışıyla orantılı olarak güç sağlamak için kaynakları kontrol etmesine ve yönetmesine yardımcı olur [13].

Hava ve su kalitesi, atmosfer ve toprak ölçümleri, afet ve acil durum uyarı gibi çevresel kalite sistemlerinde kullanılır [12].
IoT uygulamaların 2025 yılına kadar öngörülen pazar payı Şekil 2.3'de verilmiştir.



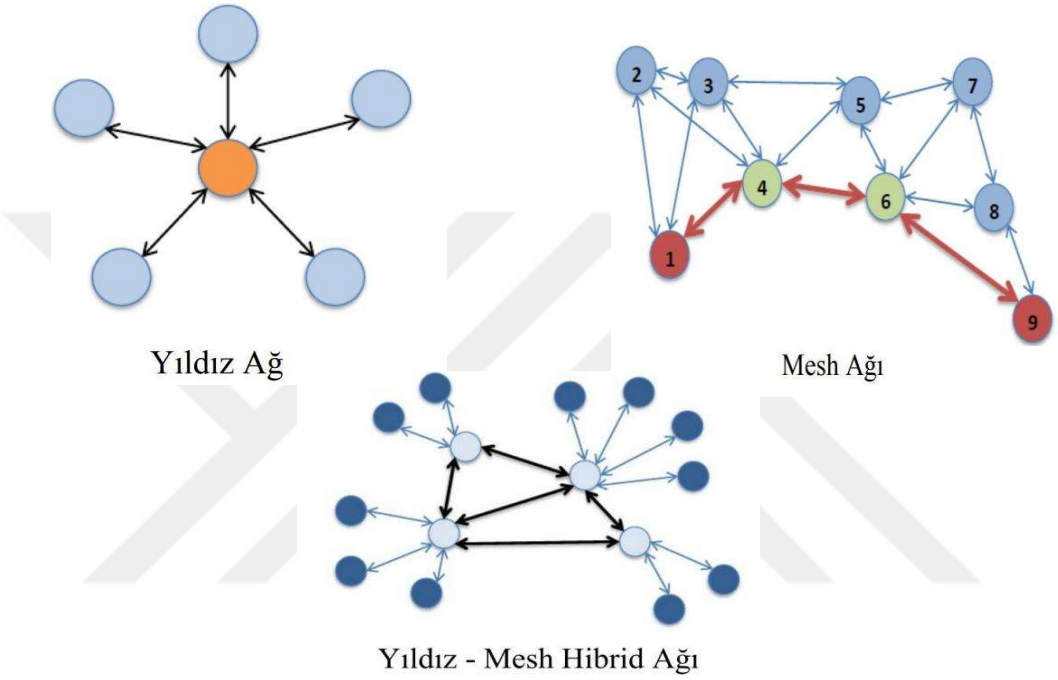
Şekil 2.3 IoT uygulamalarının 2025 yılına kadar öngörülen pazar payı [13]

2.2.1 Kablosuz Algılama Ağları

Uygun fiyatlı mikro elektronik cihazların, daha ucuz depolama sistemlerinin ve verimli İnternet iletişim teknolojilerinin geliştirilmesi, bilginin üretilmesi, işlenmesi ve iletilmesi şeklini değiştirmiştir. Bu gelişmeler ışığında son yıllarda IoT yaygınlaşmıştır. IoT dünyası farklı birçok unsurdan oluşmaktadır. Ancak çevre ile etkileşimde bulunan ve verileri toplayan algılayıcıların oluşturduğu Kablosuz Algılayıcı Ağları (Wireless Sensor Networks-KAA), IoT sistemlerin ana bileşenini oluşturmaktadır [15,16].

KAA, ortamın sıcaklık, ses, basınç vb. fiziksel koşullarını izlemek, kaydetmek ve toplanan verileri merkezi bir yerde düzenlemek için planlı olarak konumlandırılmış otonom algılama ağlarıdır. Modern ağlar çift yönlü etkileşime girerek algılayıcı etkinliğinin kontrolünü sağlayabilirler. KAA, bir veya birkaç düğüm(nodes) ile yüzlerce hatta binlerce algılama ağı düğümlerinden oluşabilir. Böylelikle bilgiye her an, her yerden kolayca ulaşılması sağlanır. Tipik olarak bir algılayıcı ağı düğümü; dahili veya harici RF alıcı/verici anten, bir mikro denetleyici, algılayıcı(sensör) ile arabirim oluşturmak için bir elektronik devre ve sistemi besleyecek bir enerji kaynağından meydana gelmektedir. Ayakkabı kutusu boyutundan bir toz tanesi boyutuna değişen ölçülerde algılama düğümü imal edilebilmektedir. Bir düğümün boyut ve maliyeti, o düğümün enerji tüketimi,

hafıza boyutu, hesaplama hızı, algılama hassasiyeti ve iletişim bant genişliği gibi performans kriterlerine bağlı olarak değişmektedir. KAA'ların ağ yapıları Şekil 2.4'te gösterildiği gibi genellikle; yıldız, yıldız-mesh ve mesh mimarileri kullanılmaktadır [17, 18].



Şekil 2.4 Kablosuz algılama ağ yapıları [18]

KAA oldukça geniş kullanım alanına sahiptir ve gün geçtikçe artmaktadır. Bilimsel tahminlere göre, geliştirilip piyasaya sürülen toplam kablosuz algılayıcı sayısının 2022 yılı sonunda 60 trilyona ulaşması, yani dünyadaki her insan için 10 bin algılayıcı olması bekleniyor [19]. Uygulama alanları genel olarak askeri, sağlık, çevre izleme, habitat izleme, tarım, endüstri, yapı, trafik ve yol, lojistik ve taşıma, web, eğitim, enerji, denizcilik, su altı ve diğer uygulamalar olarak sıralanabilir [19, 18].

KAA'da verinin doğru ve eksiksiz bir şekilde ana düğüme aktarılması, güvenli veri toplama olarak bilinir. Bu sistemler hassas veriler taşıdığından verinin iletiminde; gizlilik, veri bütünlüğü ve kullanılabilirliği ile ilgili güvenlik sorunlarını giderebilecek güvenlik altyapısının oluşturulması mutlak bir gerekliliktir. Çeşitli

şifreleme algoritmaları sayesinde bilgi şifrelenerek ana düğüme güvenli bir şekilde aktarılır. Böylece kritik öneme sahip bilgiler korunmuş olur [20].

2.2.2 Uzaktan Kontrol Sistemleri

Kepenk sistemleri (garaj, depo, market, vitrinler vs. kullanılan), ev güvenlik sistemleri, aydınlatma sistemleri gibi elektronik olarak çalışan uzaktan kontrol sistemleri giderek yaygınlaşmaktadır. Bu tür elektronik uzaktan kumanda sistemleri tipik olarak, modüle edilmiş ve kodlanmış bilgiyi alıcıya iletmek için belirli RF sinyalini kullanırlar [21]. Verici RF devreleri pille çalışan devrelerdir. Örneğin, garaj kapılarının kontrolünde bir RF alıcı kart garaj kapısının kontrolü için konumlandırılmıştır. Alıcı kart aynı frekans bandında yayın yapan bir kablosuz verici ile eşleştirilerek modüle edilmiş ve önceden belirlenmiş kod ile kontrol edilir. Böylelikle garaj kapılarının uzaktan kontrol ile açılıp kapanması sağlanabilmektedir. Bu bağlamda, kollu ve mantar bariyerler, hareketli kapılar, otomatik panjur sistemleri, aydınlatma ve iklimlendirme sistemleri, makine ve alarm sistemleri, araç kapıları gibi daha pek çok alanda kablosuz uzaktan kontrol sağlanmaktadır.

Uzaktan kumandalı cihazların güvenliğini artırmak için kumandaya her basıldığında verici, atlamalı(hops) kod olarak da bilinen yuvarlama(rolling) kodları üretir. Oluşturulan kontrol sinyali her defasında değişmektedir. Hangi kodun iletileceğini ve alınacağını takip etmek için sonuncu kod bilgisi alıcı ve verici hafızalarında belirli seri numaralar ile saklanır. Böylelikle verici her defasında hafızasındaki kayıtlı veriyi okuyarak farklı komut sinyalleri üretirken alıcı ise hafızasında kayıtlı veri sayesinde gelecek bir sonraki sinyale odaklanır.

Verici tarafından üretilen değişen kontrol sinyalleri tahmin edilemez ve kopyalanamaz olmalıdır. Bunu sağlamak için de çeşitli şifreleme algoritmaları kullanılmaktadır. Farklı matematiksel algoritmalara ve farklı tekniklerin uygulanmasına bağlı olarak, kriptografi algoritmaları genellikle işlem süreleri, bellekte kapladıkları alan ve saldırılara karşı direnç açısından farklı performanslara sahiptirler. Kullanım alanına göre en uygun algoritmanın seçimi yapılmalıdır.

2.3 Kablosuz İletişim Türleri

Günümüzde IoT sistemlerde kullanılan birçok kablosuz iletişim türü bulunmaktadır. Bu başlık altında bu iletişim yöntemleri hakkında genel bilgiler verilecektir.

2.3.1 Wi-Fi

İngilizcede Wireless Fidelity, kelimelerinin kısaltmasıdır ve kablo olmadan radyo dalgalarıyla veri transferi sağlayan bir dizi haberleşme standardına verilen addır. Elektrik ve Elektronik Mühendisleri Enstitüsü (IEEE) kablosuz iletişim standardı 802.11'e dayanır. Bu standartlara uyumlu cihazlar (bilgisayar, cep telefonu, PDA) geniş bant hızında internete kablosuz olarak bağlanabilir ve birbirleri ile 2.4 GHz ve 5 GHz frekans bantlarını kullanarak haberleşebilir. Kablosuz ağlar telsiz telefon, televizyon ve radyoların yaptığı gibi radyo sinyalleri ile haberleşir. Wi-Fi standartları ve sertifikaların düzenlenmesi Wi-Fi Alliance tarafından sağlanmaktadır. Kablosuz ağda çift yönlü radyo haberleşmesi kullanılır (23).

2.3.2 Bluetooth

Bluetooth, kısa mesafe veri iletimi için kullanılan özelleştirilmiş radyo frekansıdır. Bluetooth standartları, ilk olarak 1994 yılında Ericsson şirketi tarafından ortaya atılmıştır. Amaçları ürettikleri telefonlar ile donanımlarının kablosuz, enerji tüketimi az ve ucuz teknoloji ile birbirlerine bağlanmasıdır. Sonraları bu teknolojinin daha da gelişeceği düşünülerek kısa mesafeli ses veri iletimini de kapsayan bir protokol haline dönüştürülmüştür. Ayrı zamanda Bluetooth'un diğer veri iletim türü olan kızılötesine göre, alıcı ve vericinin birbirlerini görmesini gerektirmemesi, aynı anda birkaç cihazın birden kontrolünün sağlanması gibi üstün avantajları bulunmaktadır.

Her iletişim türünde olduğu gibi Bluetooth'ta da veri iletiminde belirli bir protokol bulunmaktadır. Bu protokol fikrini ilk dile getiren Ericsson şirketi, Bluetooth üzerinde yaptığı başarılı denemelerin ardından 1998 yılında Ericsson, Nokia, IBM, Toshiba ve İntel şirketleri bir araya gelerek Bluetooth SIG'i kurdular. Bluetooth SIG (Special Interest Group) Bluetooth standartlarını belirleyen, günümüzde ise 1500'ün üzerinde şirketi barındıran, kâr amacı gütmeyen kuruluştur. Bluetooth standartları ücretsiz olarak herkesin kullanımına açıktır [24]. Bluetooth ilk sürümü

1999 yılında tanıtılmıştır. Zaman içerinde birçok değişiklik ve güncelleme yapılmıştır. Bu güncellemeler içerisinde 2010 yılında tanıttığı 4.0 sürümü düşük enerji çözümleri sunmaktadır. Blutooth 4.0 ve üstü sürümler için Bluetooth Low Energy (BLE) ifadesi kullanılmaktadır. Bluetooth sürekli geliştirilerek yeni sürümleri yayımlanmaktadır [24, 25].

2.3.3 ZigBee

Zigbee, IEEE 802.15.4 standardını temel alır ve kablosuz haberleşme teknolojilerinde düşük hız kablosuz kişisel yerel ağ (LR-WPAN, Low-Rate Wireless Personal Area Network) haberleşmesi olarak bilinir. Arıların kovanlarına gitmek için takip ettikleri zikzak yoldan esinlenerek isimlendirilmiştir.

Zigbee teknolojisi, küçük boyutta veri alışverişi ile gerçekleştirilmesi mümkün uygulamalarda düşük maliyetli olması, minimum güç tüketme prensibine dayanması, kurulumunun kolay ve esnek yapıda olması açısında büyük oranda tercih edilmektedir. Bu teknoloji sayesinde karmaşık ağ yapıları kurmak, bunları genişletmek ve bu yapıların diğer teknolojilerle haberleşmesini sağlamak mümkündür.

Zigbee Alliance, Zigbee teknolojisinin standartlarından sorumlu dünya çapında bir birim olup; güvenilir, düşük maliyetli ve güç tüketimi az ürünler ortaya çıkarmak için birçok firmanın bir araya gelip görüntüleme ve kontrol amaçlı ürünlerin standartları üzerinde çalıştığı bir topluluktur [23, 26].

2.3.4 Küresel Mobil İletişim Sistemi

Küresel Mobil İletişim Sistemi (Global System for Mobile Communications-GSM) hücresel ağlardan oluşmaktadır. Birkaç radyo hücresi kullanılarak oluşturulmuş olan kablosuz ağ türüne hücresel ağ denilmektedir. Her hücreye hizmet bir baz istasyonu tarafından verilir. Hücresel ağlara verilebilecek en önemli örnek olan GSM mobil iletişimde bir dönüm noktası olmuştur. Teknolojide meydana gelen gelişmelerle Genel Paket Radyo Servisi (General Packet Radio Service-GPRS) ve Küresel Mobil Telekomünikasyon Sistemi (The Universal Mobile Telecommunications System-UMTS) gibi yeni nesil uygulamaların iletim hızlarının arttığı ve yeni servisleri destekledikleri görülmektedir. GSM ve GPRS haberleşme alanını belli büyüklükteki hücrelere bölmektedir. Hücre büyüklükleri o bölgenin ihtiyacı ve

iletişim yüküne bağlıdır. Frekansın yeniden kullanımına dayalı bu teknolojide hem ses hem de veri iletişimine olanak sağlanmaktadır [27].

Nesnelerin internetinde uzun mesafeli iletişim için GSM/GPRS teknolojisinden faydalanılmaktadır. Birçok uygulamanın uzaktan izlenmesi, kontrol edilmesi ve otomatik olarak işlemlerin yapılmasında bu teknoloji kullanılmaktadır. Özellikle düşük maliyetli sistemlerde daha çok tercih edilmektedir. Uygulamanın veri iletim hızı 2G, 3G ve 4G hücresel iletişim seçeneklerinde farklı olmaktadır. Genelde bu sistemler için uzun ömürlü, fiziksel koşullara dayanıklı ve gömülü olarak kullanılan Abone Kimlik Modülü (Subscriber Identity Module-SIM) kartlar tercih edilmektedir [23].

2.3.5 Kızılötesi

Kızılötesi, dalga boyu 750nm ile 1μm arasında değişiklik gösteren bir elektromanyetik dalgadır. Kısa dalgalar düşük enerji kaybından dolayı genellikle fiber optik kablolarda veri iletiminde ve gece görüş donanımlarında, orta dalgalar genellikle askeri sanayide, uzun dalgalar ise termal görüntüleme cihazlarında kullanılmaktadır.

Kızılötesi kısa mesafeli veri iletiminde kullanılır. Kızılötesi ışıklar duvarları geçemezler. Bundan dolayı genellikle aynı alan içerisindeki uygulamalarda kullanılır. Günümüzde uzaktan kumandalarda en sık tercih edilen iletişim türüdür [24].

2.3.6 1 GHz Altı Haberleşme

1 GHz altında lisansız Endüstriyel, Bilimsel ve Tıbbi (Industrial Scientific Medical band-ISM) spektrum bantlarında çalışan RF sistemleridir. ISM bantları genellikle 315 – 468MHz, 769 – 935MHz aralığındaki frekans değerlerini kapsar.

1 GHz altı haberleşme IoT uygulamalarında sıklıkla kullanılmaktadır. Bunun nedeni uzun menzillere sahip modellerinin bulunması, düşük güç tüketimini ve parazite karşı dayanıklılığının olmasıdır.

Akıllı ev sistemlerinde, yapı otomasyonunda kullanılan kumandalarda, aydınlatma sektöründe, KAA'da ve birçok alanda yoğun olarak kullanılmaktadırlar [28].

Günümüzde yaygın olarak kullanılan haberleşme türleri bunlardır. Burada zikretmediğimiz LoraWan, Z-Wave gibi iletişim türleri de bulunmaktadır. Bunlar çok yaygın olmadıkları veya daha yerel alanlarda kaldıkları için burada açıklanmamıştır.

2.4 Gömülü Sistemler

Farklı birçok tanım yapılsa da gömülü sistem terimi yaygın tanımıyla; özel bir görevi gerçekleştirmek için, yazılım ve donanımdan oluşan, genelde daha büyük sistemlerin alt bileşeni olarak görev yapan işlem birimleridir. İşlem gücü ve hafıza yetenekleri sınırlı olduğundan kullanımları bilgisayarlardan farklı olarak tasarlandıkları amaca yöneliktir.

Bir gömülü sistem tasarımları genellikle amaca uygun olarak geliştirilirler. Projedeki gerekli işlem gücü ihtiyaçlarını karşılayacak şekilde tasarım yapıldığı için maliyet açısından oldukça önemli avantajlar sağlarlar. Genellikle gereksinim duyulan donanımlar kullanılarak belirlenen görev süresi içerisinde gerçek zamanlı işlemlerini tamamlarlar.

Bir gömülü sistem; merkezi işlem birimi (CPU), flash hafıza, giriş-çıkış (IO) pinleri, analog dijital çeviriciler (ADC, DAC vs.), sayıcılar, zamanlayıcılar ve arayüz gibi çevresel elemanlardan oluşmaktadır. Kullanım yerine göre sensörler, harici depolama birimi (EEPROM) ve aktüatörler de kullanılmaktadır. Yazılım ile verilen komutlar CPU tarafından işlenir ve çevresel birimlerin görevlerini, hangi sırayla çalışacaklarını belirler. Dâhili ve harici hafızadan program ve kayıtlı verileri okuma-yazma işlemlerini gerçekleştiren CPU donanımın beyni konumundadır. Eğer merkezi işlem birimi CPU giriş/çıkış portları, zamanlayıcı, sayıcı, bellek gibi çevresel birimler ile aynı paket içerisinde ise buna mikrodenetleyici (MCU), eğer ayrı tek bir pakette ise mikroişlemci (MPU) olarak isimlendirilmektedir.

Mikrodenetleyiciler CPU yanında çevresel birimleri de içerisinde barındırarak mikro bir bilgisayar işlevi görmektedir. Günümüzde gömülü sistem uygulamalarında genellikle mikrodenetleyiciler kullanım kolaylığından dolayı tercih edilmektedir. Mikroişlemciler çevresel birimler ile iletişimde ek bir ara yüze ihtiyaç duyduğu için daha karmaşık tasarımlarda kullanılmaktadır. Bir uygulamada kullanılacak mikroişlemci ve mikrodenetleyiciler çalışma frekansları ve veri yolu

genişliğine göre belirlenir. Bir saniyede işlenen komut sayısı çalışma frekansı olarak tanımlanırken, bir komutta kaç bit veri işleneceği ise veri yolu genişliği olarak tanımlanır. Tasarımda kullanılacak mikrodenetleyici belirlenirken dikkate alınması gereken diğer özellikler ise hafıza boyutu, giriş/çıkış pin sayısı, ADC ve DAC sayıları, çevre birimleri ile haberleşme protokollerinin uyumu olarak söylenebilir.

Gömülü sistem tasarımlarında yaygın olarak kullanılan MCU ve MPC'ler 8, 16 ve 32 bit veri yolu genişliğine sahip çeşitleri bulunmaktadır. Son zamanlarda gelişen 32 bit sistemler hem işlem güçlerinin yüksek olması hem de fiyatlarının da hızla düşmesi ile tasarımlarda kullanımları artmaktadır.

Mikrodenetleyiciler Von Neumann veya Harvard mimarisi temel alınarak üretilmektedirler. Günümüzde oldukça yaygın olan 8051 Mikrodenetleyiciler Von Neumann mimarisi kullanılarak tasarlanmaktadırlar. Ayrıca 8051 mikrodenetleyiciler komut işleme tekniği olarak da CISC (Complex Instruction Set Computers) mimarisini kullanmaktadır. 8051 MCU'lar 8-bit veri yolu genişliğine sahiptirler.

Diğer yaygın kullanım alanına sahip mikrodenetleyiciler ise Microchip firmasına ait PIC serisidir. Bu MCU lar Harvard mimarisi temel alınarak tasarlanmaktadırlar. Komut işleme yapısı olarak RISC (Reduced Instruction Set Computers) mimarisinin özelliklerini kullanırlar. Ayrıca 8-bit, 16-bit ve 32-bit veri yolu genişliğine sahip PIC mikrodenetleyicileri üretilmektedir.

Bir diğer geniş uygulama alanına sahip ARM (Advanced RİSC Machine) mimari ailesi 8 ve 16 bit veri yolu genişliğine sahip mikrodenetleyicilere rakip olarak Cortex M serisini geliştirmiştir. ARM temel olarak Harvard mimarisini kullanırken, komut işleme yapısı olarak RISC mimarisinin geliştirilmiş bir türevidir. Veri yolu genişliği olarak 32-bit yaygın olarak kullanılsa da 64-bit versiyonları da üretilmektedir.

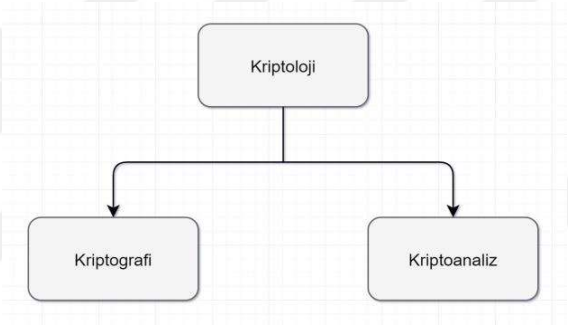
Kullanım alanları oldukça geniştir ve gündelik hayatımızda kullandığımız TV kumandası, çamaşır makinası, telefon, tıraş makinası, süpürge gibi hemen hemen tüm elektronik cihazda bu sistemlere rastlamak mümkündür. Ayrıca yüksek kalite ve güvenilirliğe sahip olmalarıyla kritik önemdeki tıbbi cihazlarda, hava savunma

sistemlerinde, otonom araçlarda, uçuş kontrol sistemleri gibi geniş bir kullanım alanları vardır. Güvenilirliğini kanıtlamış olan bu sistemler oldukça düşük hata paylarına sahiptir [29, 30].

Mikrodenetleyicilerin kullanım kolaylığı ve çoğu uygulama için isterleri karşılamalarından dolayı endüstride daha yaygın olarak kullanılmaktadır. Birçok mikrodenetleyici üreticisi bulunmaktadır. Bunlardan bazıları: NXP, Microchip, ATMEL, STMicroelectronic, Renesans Electronics, Texas Instruments, Nuvoton, Cypress Semiconductors olarak sıralanabilir.

3
KRİPTOLOJİ BİLİMİ

Matematiksel şifre bilimine kriptoloji denmektedir. Kelime anlamı olarak yunanca saklanmış/gizli anlamındaki kriptos ve bilgi, söz, bilim gibi anlamlar gelen logos kelimelerinden türetilmiştir [14]. Kriptoloji bilgilerin şifrelenmesi ve şifrelenmiş bilgilerin çözümlenmesi için yöntemler geliştirir. Kriptoloji bilimi Şekil 3.1'de gösterildiği gibi iki alt bilime ayrılmaktadır.



Şekil 3.1 Kriptoloji alt bilim dalları

Bilim dallarından kriptografi, gönderici ve alıcı haricinde üçüncü şahısların mesajlara erişimini engellemek için açık olan bilgiyi şifreleme bilimidir. Kriptoanaliz ise şifrelenmiş bilgilerin anlamını elde etmek için yöntemlerin araştırılması yani şifreleme algoritmalarının nasıl kırılacağı üzerine yapılan çalışmadır [15].

3.1 Kriptografi

Alıcıya iletilmek istenen bilginin belirli matematiksel işlemler sistematiği ile şifrelenerek güvenilir bir şekilde alıcıca teslim edilmesi ve alıcı tarafta birtakım tersi işlemler uygulanarak şifrenin çözülüp bilginin elde edilmesini kapsar. Kriptografi kelime anlamı olarak Yunancada gizli anlamına gelen "kript" ve yazı anlamına gelen "graf" kelimelerinden oluşmaktadır. Günümüzde en yoğun kullanım alanları; dijital imzalar, gizli mesajlaşma, onaylama, internet hizmetleri ve dijital bankacılık işlemleridir [12]. Performans olarak farklı özelliklere sahip

kriptografi algoritmaları geliştirilmiştir. Uygulamalara göre en uygun algoritmanın seçimini yapılmalıdır.

Kriptografinin bazı temel prensipleri ise şöyle sıralanabilir [31]:

- Şifreleme (Encryption): Şifreleme, kriptografinin önemli ilkelerinden biridir. Bu ilke, bir iletinin veya bilginin okunamaz hale gelebilmesi için şifrelenmesi gerektiğini gösterir, böylece bireylerin gizliliği korunur. Bu ilke, alıcısının özel bir dijital anahtar kullanarak alınan bilgilerin şifresini çözmesi gerektiğini de gösterir.
- Kimlik Doğrulama (Authentication): Kriptografinin önemli ilkelerinden biri bilginin kaynağını belirlemektir. Bilgi kaynağı belirlendiğinde güvenli iletişim kurmak kolaydır. Kimlik doğrulaması, yalnızca gönderen tarafından kimliğini kanıtlamak için kullanılacak özel bir anahtar değişimi sağlamakla mümkündür.
- Bütünlük (Integrity): Alıcıya gönderilen bilgilerin bütünlüğü çok önemlidir. Bu ilke, kriptografinin, aldığımızın gerçek ve hedeflenen kişiden olduğundan emin olmak için kodlar ve dijital anahtarlar sağlayarak verilerin bütünlüğünün temin edildiğini gösterir. Alıcı, alınan bilgilerin iletim işlemi sırasında değiştirilmediğinden veya tehlikeye atılmadığından emin olur.
- İnkâr Edememe (Non-repudiation): Bu ilke, bilgiyi gönderenin hiçbir zaman önceki taahhüdünü veya eylemini inkâr etmemesini sağlar. Bu ilke, gönderenin verilerin kaynağını reddetmesini önlemek için dijital imzalar kullanır.

3.2 Kriptoanaliz

Kriptoanaliz, şifreli metin ve kripto sistemlerin nasıl çalıştıklarını anlamayı, açık metin ya da anahtarları elde etme amacıyla teknikler geliştirmeyi amaçlar. Kriptoanalizin amacı, kripto sistemlerindeki güçlü ve zayıf noktaları ortaya çıkarmaktır. Böylelikle hatalı ve zayıf algoritmaların iyileştirmesi ve güçlendirilmesi sağlanabilir. Hem şifrelenmiş metinlerin deşifresine odaklanan kriptoanaliz diğer yandan da kodların ve şifrelerin ilgili algoritmalarının matematiksel çalışmalarını ortaya koyarak yeni algoritmaların geliştirilmesine yardımcı olmaktadır. Diğer taraftan açık metin ya da anahtarı elde etme yoluyla kötü amaçlar için de kullanılabilmektedir. Tarih boyunca kriptoanalistlerin

başarısı kriptografları yeni ve daha güvenli sistemler tasarlamaya zorlamıştır [14].

Analistin analiz edilen şifreleme metni hakkında ne kadar bilgiye sahip olduğuna bağlı olarak değişen birçok farklı kriptoanaliz modeli ve tekniği vardır. Sadece şifreli metin saldırısı (ciphertext only attack), bilinen açık metin saldırısı (known plaintext attack), seçilen açık metin saldırısı (chosen ciphertext attack), seçilmiş şifreli metin saldırısı (chosen ciphertext attack), seçilmiş anahtar saldırısı (related key attack) bazı kriptoanalitik yöntemlerindendir [32, 12].

3.3 Güvenlik Endişeleri ve Kriptolojinin Önemi

Teknolojinin hızla gelişmeye devam ettiği son yıllarda birçok teknoloji hem kamu ve şirketler hem de kişisel olarak gündelik hayatta hızla yaygınlaşmaktadır. İnternet ve bilgisayar gibi teknolojiler hayatımızın vazgeçilmez parçaları haline gelmiştir. Alışveriş, bankacılık işlemleri, kamusal hizmetler, rezervasyon işlemleri, sağlık hizmetleri, faturalar, şehir şebeke hatları, e-imza gibi daha birçok alanda internet üzerinden işlemler yapılmaktadır. Ayrıca gizliliği stratejik öneme sahip askeri, şahsi, ticari ve politik birçok belge ve bilgi elektronik ortama taşınmıştır. Bu durum bilginin üçüncü şahıslardan korunması, saklanması ve güvenli bir şekilde iletilmesi açısından yeni yöntemlere olan ihtiyacı artırmaktadır. Sanal ağlar bilgiye erişimi kolaylaştırmasına karşın bir o kadar da güvenlik risklerini artırmaktadır [14, 33, 34].

IoT ağlarındaki cihazlar, kablolu ve kablosuz şebekeler aracılığıyla internete dahil olmasıyla birlikte, internet sadece insanlar arasında değil sistemdeki nesneler ve makinalar arasında da iletişim sağlayan bir platform haline gelmiştir. IoT cihazların sahip olduğu sınırlı işlem yeteneği, bant genişliği ve hafıza yetileri, internet üzerinde her geçen gün artan güvenlik tehditleri karşısında onları savunmasız bırakmaktadır [12, 15, 35].

Kablosuz algılama ağlarının çalışma özellikleri güvenlik tehditleri açısından yeni endişeler yaratmıştır. Algılayıcılar tarafından toplanan veriler iletim sırasında çalınabilir veya değiştirilebilir. Hatta yetkisiz kötü amaçlı algılayıcılar bile ağa yanlış bilgi gönderebilir. Genel olarak, kablosuz algılama ağları birçok güvenlik tehdidiyle karşı karşıya kalabilir [16]. Örneğin, bir askeri uygulamada sahadan toplanan verilerin çalınması veya değiştirilmesi sonucu yanlış taktikler

geliştirilebilir. Bunun sonucunda teçhizat hatta dost güçler için can kaybına neden olabilir. Bir diğer örnekte sağlık sektörü için verilebilir. Yakın tarihlerde yapılan çalışmalar mevcut durumda kullanılan sistemlerin çoğunun hasta mahremiyetini koruyabilecek güvenlik sistemlerine sahip olmadıklarını ortaya koymuştur [36]. Gizli hasta bilgilerinin sistem arızaları veya güvenli veri iletiminin sağlanamamasından dolayı yetkisiz kişilerin eline geçmesi büyük bir güvenlik açığıdır [19]. Bu ağlardaki güvenlik sorunlarını gidermek için birçok çalışma yapılmıştır [16].

Uzaktan kumandalı sistemler için de güvenli bilgi iletimi önemli bir konudur. Örneğin, araç kapıları uzaktan kumanda ile açılıp kapanabilmektedir. Kullanıcı kumandanın her tuşuna bastığında verici havaya gerekli komutu gönderir ve komut RF sinyalleri aracılığıyla kablosuz iletilir. Yeterince güvenli iletişim sağlanamadığında kumanda sinyallerini taklit eden kopya kumandalar yapılmakta ve bu durum büyük güvenlik açıkları oluşturmaktadır. Benzer bir şekilde market, iş yeri kepenk sistemleri veya korunaklı bir bölge girişinde bariyer sistemleri için de aynı güvenlik endişeleri mevcuttur. Nihayetinde bu durum terörist saldırılar, mal ve bilgi hırsızlığı ile sonuçlanabilir. Bundan dolayı verici tarafından değişken kodlu sinyaller üretilmesi ve bunların çeşitli şifreleme algoritmaları ile şifrelenerek iletilmesi gerekmektedir [37].

Sonuç olarak bilginin güvenli bir şekilde iletilmesi kritik öneme sahiptir. Bilginin çeşitli şifreleme yöntemleri ile gizlenerek iletilmesi ve elde edilmesi yetkisiz olan üçüncü şahısların bilgiye erişim riskini azaltacak yöntemlerden biridir. Kriptoloji alt bilim dalı olan kriptografi bilginin gizliliğini ve güvenilirliğini korumak için kullanılan yöntemlerden biridir. Bundan dolayı, kişisel ve ulusal bilgi güvenliğini sağlamak, olası saldırılardan korunmak için kriptoloji alanında yapılan çalışmalar çok önemlidir [14, 34].

3.4 Kriptoloji Tarihçesi

Tarihte ilk kez bilginin arşivlenmesi veya haberleşme için kâğıda yazılma ihtiyacının oluşmasıyla birlikte özellikle önemli görülen bilgilerin ortaya çıkma ve istenmeyen kişiler tarafından ele geçirilme endişesi doğmuştur. Böylelikle insanlar ya bilginin şifrelenmesinin yollarını bulmaya ya da başkalarının şifreli bilgilerini

anlamaya çalışarak kriptoloji biliminin gelişmesini sağlamışlardır [32].

Tarihte bilinen en eski şifrele metni bunda 4000 yıl önce antik mısır kasabası MENET KHUFU'da soylu KHNUMHOTEP II mezarındaki kitabenin olduğu düşünülmektedir. Mısırlı kâtip bu yazıtta sıra dışı hiyeroglif işaretler kullandığı tespit edilmiştir [33, 38].

Kriptoloji geçmiş dönemlerde daha çok, günümüzde hala en önemli alanlarından olan, askeri ve diplomatik haberleşmede bilgi güvenliğinin sağlanması için kullanıldığı bilinmektedir [14]. M.Ö. 5. yüzyılda kriptolojiyi askeri iletişimde ilk spartalılar kullandı. SCYTALE adı verilen silindir cihazın etrafına bir parşömen veya deri şerit sarılır ve mesaj üzerine yazılırdı. Alıcı şeritleri aynı çapta silindire sarıp çözmesi gereklidir. Cihazın çapı kripto anahtarına denk gelmektedir [14, 33].

M.Ö. 60-50 yıllarında Yunan yazar Polyibus'un tanımladığı teknik ilk olarak Roma İmparatoru Julius Caesar tarafından generallerine mesaj iletmek için kullanılmıştır. Bu şifreleme yönteminde alfabedeki harfler belli miktar sağa ötelenerek gizli metin elde ediliyordu [38].

Kriptoloji bilimi iletilecek bilginin şifrelenmesinin yanında şifreli bir metnin açık hale getirilmesi için de metotlar geliştirir. Gizli bilginin anahtar olmadan deşifre etmeye kriptoanaliz denmektedir. Şifrelerin çözümünde istatistik, matematik ve dilbilim alanlarından yararlanılır. Bu bağlamda ilk kriptoanaliz çalışması Al-Kindi tarafından 9. yüzyılda yapılmıştır. Al-Kindi çalışmalarında, hangi dilde yazıldığı bilinen bir şifreli mesajın yine o dilde yazılmış yeteri uzunluktaki bir metnin analizi sonucunda harflerin kullanım sıklıklarının belirlenmesiyle kolaylıkla çözülebildiğini ortaya koymuştur. Bu nedenle daha sonra geliştirilen kriptolarda birden fazla alfabe kullanılmış, böylelikle çok alfabeli şifreleme algoritmaları geliştirilmiştir [14].

Leon Alberti 1466 yılında yazdığı makalede, ilk kez çoklu alfabe kullanarak Alberti şifreleme diskinin yapımını anlattı. Temel olarak Sezar şifrelemede kullanılan harf öteleme yöntemini kullanmakla beraber harflerin öteleme miktarı sabit değildir ve bunu kullanıcı belirlemektedir. Son beş yüz içinde kriptografideki en önemli ilerleme olarak kabul edilmektedir [14, 38].

Giovan Battista Bellaso 1553 yılında 1863 yılına kadar yani üç yüz yıl kırılamayan

çok alfabeli kriptografi tekniğini geliştirdi. Bu şifreleme, Fransız Blaise de Vigenère tarafından biraz daha geliştirilerek Vigenère Şifresi adını almıştır. 1863 yılında Friedrich Kasiski vigenère şifrelerini deşifre etmek için genel bir yöntem yayınlayan ilk kişidir [14, 38].

1790'da Thomas Jefferson, Jefferson diski adıyla her birinin üzerinde İngiliz alfabesindeki 26 harfin rasgele dizildiği toplamda 36 diskten oluşan şifreleme sistemini geliştirmiştir. Ortaları delik olan bu her disk benzersiz bir numaraya sahiptir ve istenilen sıraya göre dizilirler. Disklerin sırası şifre anahtarıdır ve hem mesajı gönderici hem de alıcı diskleri önceden tanımlanmış aynı sırada düzenlemelidir. Alıcı şifreli metin ile diskte sırayı oluşturup, diğer sıralardaki açık metne ulaşmış olur. Bu sistemi temel alarak oluşturulan M-94, 1923'ten 1942 yılları arasında Birleşik Devletler Ordusu M-94 olarak kullanıldı [14, 33].

Frank Miller tarafından 1882 yılında tanımlanmış olduğu şifreleme çalışması, 1917 yılında Gilbert Vernam tarafından geliştirilip bir yazı makinası için şifreleme çözümü icat edildi. Joseph O. Mauborgne anahtar kasetindeki karakterlerin tamamen rasgele olabileceğini keşfetti. Birlikte tasarladıkları ilk One-Time Pad Şifreleme sisteminin patentini 22 Temmuz 1919 yılında almışlardır [33, 39].

1883'te Auguste Kerckhoffs tarafından yazılan La Cryptographie Militaire adlı makalede şifreleme sistemlerinin tasarım prensiplerini anlatmıştır. Bu prensipler sonraki dönemlerde geliştirilen sistemlerde önemli bir yol gösterici olmuştur. Kerckhoff Prensipleri [14]:

- Sistem pratik ve matematiksel bir gerçekliğe dayanmalıdır.
- Sistem gizliliğe dayanmamalıdır. Yani sistem hakkındaki her şey herkes tarafından bilinmelidir.
- Sistemde kullanılan anahtarlar taraflar arasında kolayca, üçüncü kişinin değiştirmesine izin verilmeden değiştirilebilmelidir.
- Sistemin kullanılabilmesi için fazla sayıda insana ihtiyaç duyulmamalıdır.

Sistemin uygulaması ve anlaşılması kolay olmalı ve şifreleme sisteminin güvenliği, şifreleme algoritmasını gizli tutmaya dayanmamalıdır. Güvenlik; yalnızca anahtarı gizli tutmaya dayanmalıdır.

19. yüzyılın sonlarına doğru İtalyan fizikçi Markoni'nin telsizi icat etmesiyle yeni güvenlik kaygıları ortaya çıkmış ve güvenli iletişime duyulan ihtiyaç daha da artmıştır. Özellikle telsizlerin savaş alanlarında kullanımının yaygınlaşmış olması ve kablosuz iletişimden dolayı düşmanın mesajlara kolaylıkla erişebiliyor olması, telsiz iletişimde şifreleme tekniklerinin araştırmaları yoğunlaştırmıştır.

I. Dünya savaşında kullanılan şifreleme yöntemlerinden biri Almanların geliştirmiş oldukları ADFGVX sistemleri ve "kod kitabı" yöntemidir. Bu yöntemde iletilecek mesajlardaki her kelime için sayı grupları kullanılıyordu. Savaş sırasında Almanların Meksika başbakanına gönderdiği bilgiler İngiliz işaret toplama birimleri tarafından çözülmüş ve elde edilen bilgiler ABD'nin savaşa girmesine ve savaşın seyrinin değişmesine neden olmuştur [14, 40].

Almanya 1923 başlarında Enigma adlı kriptoloji cihazını geliştirdi ve 1930 yıllarının ortalarına gelindiğinde artık ordunun onaylı şifreleme cihazı oldu. Bu Makine II. Dünya savaşında alman ordusu tarafından aktif olarak kullanılmıştır. Elon Tureng öncülüğündeki İngiliz kriptoanaliz ekibi geliştirdikleri Colossus makinası sayesinde Enigma şifresi çözülmüş ve savaşın seyri değişmiştir [14, 40].

Teknolojinin ilerlemesi harflerin yerini 1 ve 0'ların almasıyla şifreleme işlemleri bitler ile yapılmaya başlanmıştır. 1970 yılında IBM tarafından, DES algoritmalarının da temelini oluşturan, Lucifer adı verilen şifreleme sistemi tanıttı ve 1975 yılında Birleşik Bilgi İşleme Standardı olarak kabul edildi [14, 33].

Elektronik haberleşmenin yaygınlaşması sonucu herkesin kullanabileceği ve hassas elektronik devlet bilgilerinin korunabileceği şifreleme algoritma ihtiyacı doğmuştur. Bu ihtiyacı karşılamak için Lucifer algoritmasında birtakım değişiklikler yapılarak DES geliştirilmiştir. DES ikilik tabanda olan düz metni 56 bitlik anahtar kullanarak 64 bitlik veri bloklarını şifreler. Şifreleme anahtarı rasgele ve gizli seçilmektedir [12, 14].

1976 yılına gelindiğinde gönderici ve alıcının şifreleme anahtarı için bir araya gelmelerinin zorunlu olmadığı, birbirini hiç tanımayan kişiler arasında bile güveli iletişime olanak sağlayan Diffie-Hellman Anahtar Değişim Algoritması yayınlandı. Bu algoritma geliştirilene kadar haberleşen her iki taraf da önceden belirlenmiş anahtarları bilmesi gereken tasarlanan bu sistemde her birey kendi özel anahtarına sahiptir. Böylelikle bu çalışmayı temel alan, 1977'de Rivert, Shamir ve

Adleman tarafından adlarının baş harflerinden oluşan, asimetrik anahtar paylaşımının güçlüklerini ortadan kaldıran RSA algoritmasının tanıtılması kriptolojide çığır açmıştır [14, 41].

Neal Koblitz ve Victor S. Miller 1985 yılında birbirlerinden bağımsız olarak eliptik eğrilerin kriptolojide kullanımını önermişlerdir. Eliptik Eğri Şifreleme (ECC) aynı güvenlik seviyesini sağlamak için, diğer açık anahtarlı algoritmalara göre, daha küçük anahtar boyutuna ihtiyaç duyar [33, 42].

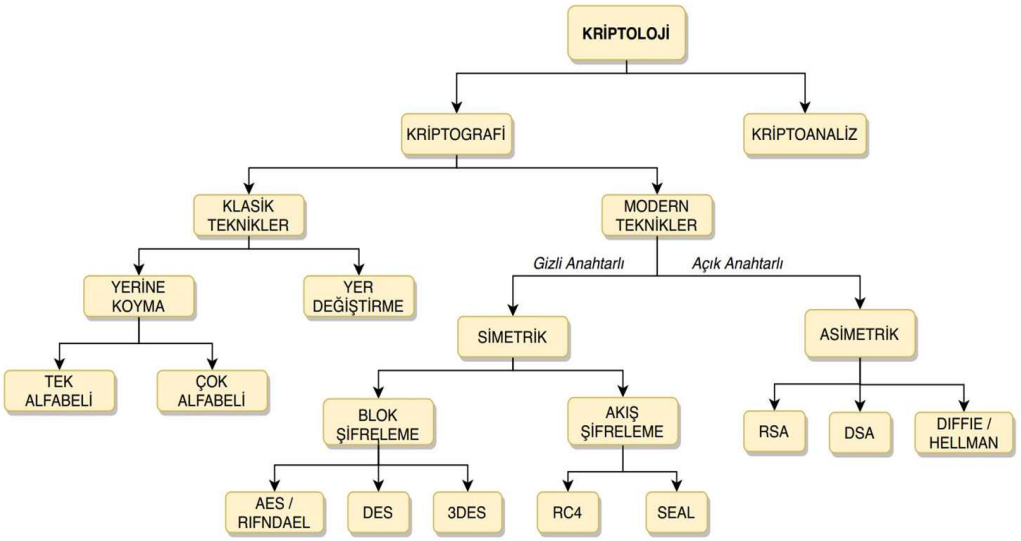
Bilgisayar işlem gücünün gelişmesiyle DES algoritmasının güvenilirliğinin azalması yeni bir şifreleme algoritması ihtiyacı doğurdu. NIST 1997 yılında bir yarışma başlatmış, Joan Daemen ve Vincent Rijmen tarafından geliştirilen Rijndael Algoritması DES'in yerine 2001 yılında AES standartlaştırılmıştır. 128, 192, 256 bitlik anahtar uzunluğu seçeneklerine sahip olan AES günümüzde hala güvenilirliğini korumaktadır [33, 12].

2009 yılında ilk Kriptografi olimpiyatları yeni güvenilir algoritmaların geliştirilmesi amacıyla Belçika Katholieke Üniversitesinde düzenlenmiştir [14].

Dünyada ve Türkiye'de araştırma ve geliştirme faaliyetleri hızla ilerlemektedir. Bu alandaki çalışmalara ülkemizde TÜBİTAK Bilgem bünyesindeki Ulusal elektronik ve Kriptoloji Araştırma Enstitüsü (UEKAE) öncülük etmektedir [14].

3.5 Kriptolojinin Sınıflandırılması

Kriptolojinin tarihsel gelişiminde görüleceği gibi teknolojinin gelişmesi, yeni kriptoanaliz yöntemlerinin bulunması ve faklı ihtiyaçların oluşması, şifreleme yöntemlerinde farklılıklara neden olmuştur. Kriptoloji Şekil 3.2'de ki gibi sınıflandırılabilir:



Şekil 3.2 Kriptolojinin sınıflandırılması [14]

3.5.1 Klasik Şifreleme Algoritmaları

Klasik şifreleme algoritmalarında iletilecek olan mesaj, harf veya harf gruplarının sistematik olarak değiştirilmesiyle elde edilmektedir [14]. Elektronik bilgisayarlardan önce kalem-kâğıt kullanarak oluşturulabilen algoritmalardır. Modern kriptoloji algoritmaları ile karşılaştırıldığında, kriptoanalizlerinin yapılmış olması ve açık anahtar dağıtımındaki problemler en önemli dezavantajlarıdır. Ayrıca algoritmalarının gizli tutulması gereken şifreleme teknikleri oldukları için güvenilirlikleri düşüktür. Klasik kriptoloji teknikleri sadece askeri ve ileri akademik kurumlarda kullanılmıştır [14, 43]. Sezar(öteleme), Affine, Vigenere, Hill gibi şifreleme algoritmaları klasik kriptolojiye örnek olarak verilebilir.

3.5.2 Modern Şifreleme Algoritmaları

Elle yapılan işlemlerin yerini hızlı işlem kapasitesine sahip bilgisayarların almasıyla modern şifreleme bilimi gelişmiştir. Modern şifreleme algoritmaları simetrik (gizli anahtarlı) ve asimetrik (açık anahtarlı) olmak üzere iki gruba ayrılırlar.

Bilgilerin hem şifrelenmesinde hem de şifreli metinlerin çözülmesinde sadece iki tarafın bildiği tek bir anahtarın kullanıldığı şifreleme türüne simetrik şifreleme denir. Simetrik şifreleme tekniğini kullanan taraflar daha önceden anahtarı

değişimi yapmalıdırlar. Bu şifreleme yöntemi, mesajları şifrelemek ve şifresini çözmek için biri genel diğeri özel olmak üzere iki farklı anahtarın kullanıldığı asimetrik şifrelemeden farklıdır.

Simetrik şifreleme algoritmaları kullanılarak veriler, şifreli metni çözmek için gizli anahtara sahip olmayan taraflarca çözülemeyecek forma dönüştürülürler. Anahtara sahip alıcı şifreli mesajı aldığında, algoritma işlemlerini tersine çevirerek mesajın orijinal ve anlaşılır biçimine ulaşır. Hem göndericinin hem de alıcının kullandığı gizli anahtar, birbirini tanımayan taraflar arasında asimetrik (açık anahtarlı) algoritmalar kullanılarak taraflar arasında paylaşılır.

Simetrik algoritmalar blok ve akış algoritmaları olarak ikiye ayrılmaktadır. Blok şifrelemede belirli uzunluktaki veri blokları anahtar yardımıyla şifrelenir. Veriler şifrelenirken, sistem blokların tamamlanmasını bekler ve bu süre içerisinde verileri belleğinde tutar. Akış şifreleme de ise veriler bellekte tutulmak yerine gelen veri şifrelenir.

Simetrik şifreleme daha eski bir şifreleme yöntemi olsa da veri boyutu ve yoğun CPU kullanımıyla ilgili performans sorunları nedeniyle asimetrik şifrelemeden daha hızlı ve verimlidir çalışmaktadır. Simetrik şifrelemenin (asimetrik ile karşılaştırıldığında) daha iyi performansı göstermesi ve daha hızlı olması nedeniyle, simetrik şifreleme tipik olarak büyük miktarda veriyi şifrelemek için sıklıkla kullanılır. Tez konusu kapsamında analiz edilen hafif kriptoloji algoritmaları bu sınıfa girmektedir.

3.6 Hafif (Lightweight) Kriptoloji

Nesnelerin İnterneti yaygınlaştıkça kullanım alanlarına göre bilgi güvenliği kritik hal almaya başladı. IoT cihazların bellek, işlem gücü, batarya gibi sınırlı yeteneklere sahip olması kullanılan klasik şifreleme tekniklerinden farklı olarak daha az işlem gerektiren, hafızada az yer kaplayan, düşük enerji tüketen yeni tekniklerin oluşturulmasını zorunlu hale getirdi. Bu kapsamda hafif kriptoloji algoritmaları geliştirilmeye başlandı. Özellikle geçtiğimiz son on yılda birçok algoritma geliştirilerek düşük kapasiteli bu cihazlarda kullanılmıştır. Hem ulusal kuruluşlar (National Institute of Standards and Technology) hem de uluslararası kuruluşlar (International Organization for Standardization-ISO/International

Electrotechnical Commission-IEC) hafif kriptoloji için IoT ve RFID cihazlarda kullanılabilecek bir dizi yöntem önermişlerdir. Uygulama alanlarına göre yapılan sınıflandırma;

- Geleneksel Kriptoloji: Bilgisayarlar, Tabletler, Akıllı telefonlar, Sunucular
- Hafif Kriptoloji: Gömülü sistemler, RFID, Sensör ağları

Tez kapsamında analizi yapılacak olan hafif şifreleme algoritmalarından TWINE, kısıtlı donanım ortamları için önerilen hafif blok şifreleme tasarımlarından biridir. SIMON ve SPECK basit, esnek ve donanım veya yazılım tabanlı iyi performans gösterecek şekilde tasarlanmış hafif blok şifreleme aileleridir. Ayrıca AES blok şifreleme ailesinin AES-128 varyantı yine [2]'de sunulan çalışmada hafif kriptoloji olarak kullanılabileceği ortaya konmuştur.

3.6.1 AES (Advanced Encryption Standard) Algoritması

DES algoritmasına karşı yeni kriptoanaliz tekniklerinin geliştirilmesi ve algoritmaya olan güvenin 90'lı yılların sonuna doğru azalmasıyla birlikte yeni bir blok şifreleme algoritması ihtiyacı ortaya çıkmıştır. Bunu için 1997 yılında NIST tarafından bir yarışma düzenlenmiş ve yarışmaya 15 farklı algoritma katılmıştır. Değerlendirmeler sonucunda yarışmaya katılan Rijndael algoritması, AES olarak seçilmiş ve standartlaştırma işlemleri 26 Mayıs 2002'de tamamlanarak resmi olarak onaylanmıştır.

Günümüzde yaygın olarak kullanılan AES algoritması NSA (National Security Agency) tarafından yüksek gizlilik gerektiren verilerin şifrelenmesi için onaylanmış kamuya açık ilk şifreleme algoritmasıdır.

Temelde karıştırma-kaydırma (Permutation-Substutition) işlemlerine dayanan AES algoritması, 128, 192 ve 256 bitlik anahtar uzunluğunda üç farklı seçeneğe sahip algoritma ile blok şifreleme yapabilir. Açık metnin şifreli metin elde edilmesi işlemlerinde, 128 bitlik anahtar için 10, 192 bitlik anahtar için 12, 256 bitlik anahtar için 14 çevrim uygulanmaktadır.

İlk olarak açık metin anahtar uzunluğuna göre parçalanır ve Tablo 3.2'de verilen durum matrisi elde edilir. 128 bitlik veri 128 bitlik anahtar değeri ile şifrelenecek ise 4x4 anahtar matrisi Tablo 3.1'de sunulan biçimde yazılarak 4x4 anahtar matrisi oluşturulur. Kullanılan anahtar uzunluğu 192-bit ise 6x4 boyutunda, 256-bit ise

6x8 boyutunda anahtar ve durum matrisi oluşturulur. Şifreleme işlemlerini üç bölümde ele alınabilir.

1. Döngü

- Tur Anahtarı Eklenir (AddRoundKey)

2. Döngü

- Bayt Değiştirme (SubBaytes)
- Satır Kaydırma (ShiftRows)
- Sütun Karıştırma (MixColumn)
- Tur Anahtarı Eklenir (AddRoundKey)

3. Döngü

- Bayt Değiştirme (SubBaytes)
- Satır Kaydırma (ShiftRows)
- Tur Anahtarı Eklenir (AddRoundKey)

Tablo 3.1 Anahtar matrisi

k_1	k_5	k_9	k_{13}
k_2	k_6	k_{10}	k_{14}
k_3	k_7	k_{11}	k_{15}
k_4	k_8	k_{12}	k_{16}

Tablo 3.2 Açık metin durum matrisi

d_1	d_5	d_9	d_{13}
d_2	d_6	d_{10}	d_{14}
d_3	d_7	d_{11}	d_{15}
d_4	d_8	d_{12}	d_{16}

Elde edilen anahtar matrisi ve açık metin durum matrisi Tablo 3.3'de verildiği gibi XOR işlemine tabi tutulur.

Tablo 3.3 Anahtar matrisi ile durum matrisi XOR işlemi

k_1	k_5	k_9	k_{13}	XOR	d_1	d_5	d_9	d_{13}	=	S_1	S_5	S_9	S_{13}
k_2	k_6	k_{10}	k_{14}		d_2	d_6	d_{10}	d_{14}		S_2	S_6	S_{10}	S_{14}
k_3	k_7	k_{11}	k_{15}		d_3	d_7	d_{11}	d_{15}		S_3	S_7	S_{11}	S_{15}
k_4	k_8	k_{12}	k_{16}		d_4	d_8	d_{12}	d_{16}		S_4	S_8	S_{12}	S_{16}

Bayt değiştirme işlemi; durum matrisi 4x4 boyutunda yazılır. Bayt karıştırma işlemleri için bir "S" kutusuna ihtiyaç vardır. Bu kutular 16x16 boyutunda ve onaltılık tabanda tüm elemanları içermektedir. Karıştırma işlemi için kullanılan S kutusu Tablo 3.4'de verilmiştir.

Tablo 3.4 S kutusu

		Y															
		0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
X	0	63	7C	77	7B	F2	6B	6F	C5	30	01	67	2B	FE	D7	AB	76
	1	CA	82	C9	7D	FA	59	47	F0	AD	D4	A2	AF	9C	A4	72	C0
	2	B7	FD	93	26	36	3F	F7	CC	34	A5	E5	F1	71	D8	31	15
	3	04	C7	23	C3	18	96	05	9A	07	12	80	E2	EB	27	B2	75
	4	09	83	2C	1A	1B	6E	5A	A0	52	3B	D6	B3	29	E3	2F	84
	5	53	D1	00	ED	20	FC	B1	5B	6A	CB	BE	39	4A	4C	58	CF
	6	D0	EF	AA	FB	43	4D	33	85	45	F9	02	7F	50	3C	9F	A8
	7	51	A3	40	8F	92	9D	38	F5	BC	B6	DA	21	10	FF	F3	D2
	8	CD	0C	13	EC	5F	97	44	17	C4	A7	7E	3D	64	5D	19	73
	9	60	81	4F	DC	22	2A	90	88	46	EE	B8	14	DE	5E	0B	DB
	a	E0	32	3A	0A	49	06	24	5C	C2	D3	AC	62	91	95	E4	79
	b	E7	C8	37	6D	8D	D5	4E	A9	6C	56	F4	EA	65	7A	AE	08
	c	BA	78	25	2E	1C	A6	B4	C6	E8	DD	74	1F	4B	BD	8B	8A
	d	70	3E	B5	66	48	03	F6	0E	61	35	57	B9	86	C1	1D	9E
	e	E1	F8	98	11	69	D9	8E	94	9B	1E	87	E9	CE	55	28	DF
	f	8C	A1	89	0D	BF	E6	42	68	41	99	2D	0F	B0	54	BB	16

Durum matrisinin bayt değişim işleminde her bir durum matrisine karşılık gelen "S" kutusu elemanı yazılarak yeni durum matrisi elde edilir. Örneğin durum matrisinin ile elemanı 13 olsun. Yerini alacak eleman 1. Satır 3. sütun şeklinde hesaplanır ve 7D değeri tablodan bulunur.

Satır kaydırma işlemi; bir önceki adımda elde ettiğimiz yeni durum matrisi kullanılarak Tablo 3.5'de gösterildiği gibi yapılır. Satırlarda sırasıyla oransal olarak sola kaydırma işlemi uygulanır. 1. satır 0(yani kaydırma yok), 2. satır 1, 3. satır 2 şeklinde n. satır $n-1$ kadar sola kaydırılır.

Tablo 3.5 Satır kaydırma işlemi

S_1	S_5	S_9	S_{13}
S_2	S_6	S_{10}	S_{14}
S_3	S_7	S_{11}	S_{15}
S_4	S_8	S_{12}	S_{16}

$\rightarrow$

S_1	S_5	S_9	S_{13}
S_6	S_{10}	S_{14}	S_2
S_{11}	S_{15}	S_3	S_7
S_{16}	S_4	S_8	S_{12}

Sütun karıştırma işlemi; güncel durum matrisi Tablo 3.6'da verilen sütün karıştırma matrisi ile Tablo 3.7'de verilen çarpma işlemine tabi tutulur.

Tablo 3.6 Sütun karıştırma matrisi

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix}$$

Tablo 3.7 Durum matrisi ile sütun karıştırma matrisinin çarpımı

$$\begin{bmatrix} 02 & 03 & 01 & 01 \\ 01 & 02 & 03 & 01 \\ 01 & 01 & 02 & 03 \\ 03 & 01 & 01 & 02 \end{bmatrix} \begin{bmatrix} S_1 & S_5 & S_9 & S_{13} \\ S_2 & S_6 & S_{10} & S_{14} \\ S_3 & S_7 & S_{11} & S_{15} \\ S_4 & S_8 & S_{12} & S_{16} \end{bmatrix} = \begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} \\ S_{21} & S_{22} & S_{23} & S_{24} \\ S_{31} & S_{32} & S_{33} & S_{34} \\ S_{41} & S_{42} & S_{43} & S_{44} \end{bmatrix}$$

Bu işlem sonucunda güncel durum matrisi elde edilmiş olunur.

Döngü anahtarı elde etme: Anahtar uzunluklarına göre çevrim sayılarının değiştiğini söylemiştik. 128 bitlik bir anahtar değeri için 10 çevrim uygulanır ve her bir çevrimde kullanılan anahtar bir sonraki çevrimde kullanılacak anahtarı türetmek için kullanılır. Sırasıyla anahtar değerlerini elde etmek için anahtar

değerini sola kaydırma, S kutusundan geçirme ve R_c döngü sabiti vektörü ile toplama adımlarının uygulanması gerekir.

Tablo 3.8 Döngü sabiti vektörü

Döngü Sayısı	Rc Değeri
1	01 00 00 00
2	02 00 00 00
3	04 00 00 00
4	08 00 00 00
5	10 00 00 00
6	20 00 00 00
7	40 00 00 00
8	80 00 00 00
9	1B 00 00 00
10	36 00 00 00

Döngü1 anahtarını bulmak için döngü0'ın anahtar matris değeri sütun bazlı olarak ayırılır. $p[0] = "k_1, k_2, k_3, k_4"$ $p[1] = "k_5, k_6, k_7, k_8"$ $p[2] = "k_9, k_{10}, k_{11}, k_{12}"$ $p[3] = "k_{13}, k_{14}, k_{15}, k_{16}"$ olacak şekilde ayrılır. Ardından $p[3]$ değeri üzerinden sola kaydırma, S kutusundan geçirme ve Tablo 3.8'de verilen R_c vektörü ile toplama işlemi uygulanarak $g(S(p[3]))$ elde edilir.

Döngü1 anahtarının ilk sütunu, döngü0 anahtarının ilk sütun değeri ile $g(S(p[3]))$ değerinin XOR işlemine tabi tutulmuş halidir. Sırasıyla (3.1), (3.2), (3.3) ve (3.4) denklemleri kullanılarak döngü anahtarı elde edilir.

$$p[4] = p[0] \oplus g(S(p[3])) \quad (3.1)$$

$$p[5] = p[4] \oplus p[1] \quad (3.2)$$

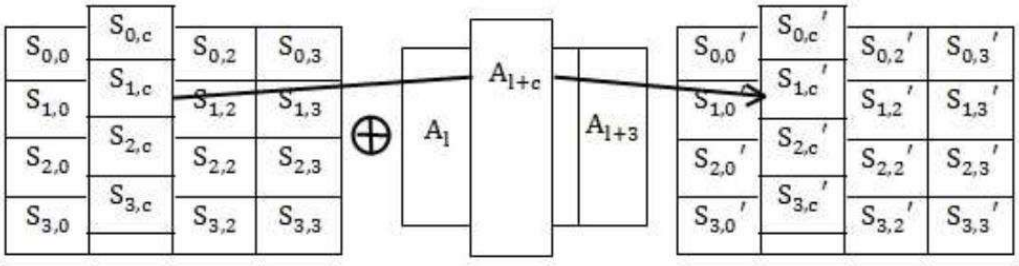
$$p[6] = p[5] \oplus p[2] \quad (3.3)$$

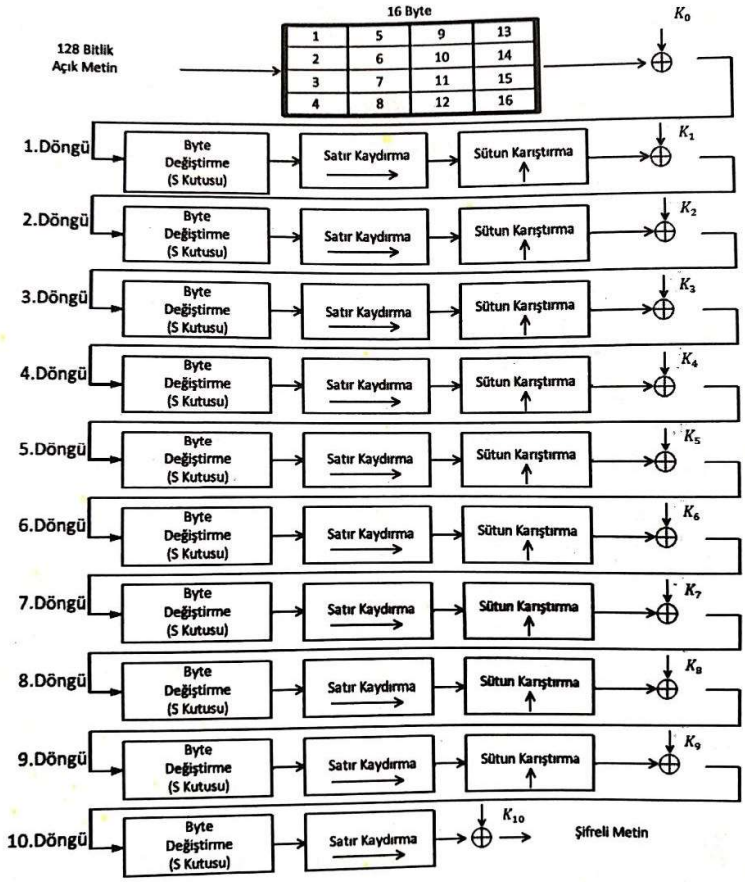
$$p[7] = p[6] \oplus p[3] \quad (3.4)$$

Döngü1 anahtarı: $p[4], p[5], p[6], p[7]$ elde edilmiş olunur. Benzer şekilde Döngü2 anahtarını bulmak için de Döngü1 anahtarı kullanılır ve işlem tekrarlanarak Döngü10 anahtar değerine kadar tümüm anahtar değerleri elde edilmiş olur.

Döngü anahtarı ekleme işlemi; AES şifreleme algoritmasında her bir döngü sonunda güncellenmiş durum matrisi ile elde edilen döngü anahtarları özel veya (XOR) işlemine tabi tutulur. İşleme ait genel şema Tablo 3.9'da verilmiştir. AES şifreleme işleminin genel akış şeması Şekil 3.3'de sunulmuştur.

Tablo 3.9 Döngü anahtar matrisi ekleme işlemi





Şekil 3.3 AES algoritması blok şeması [44]

3.6.2 Twine Algoritması

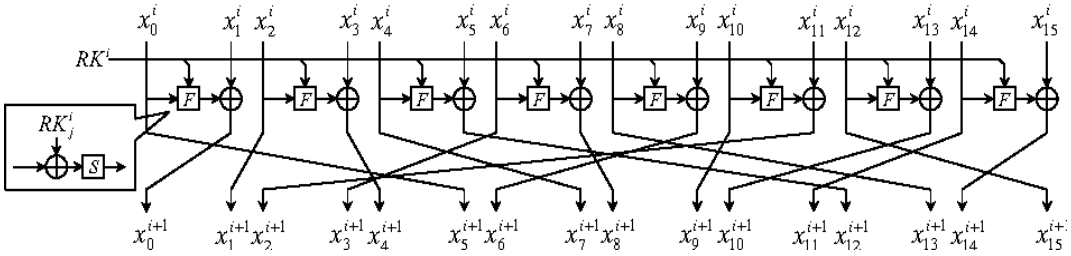
Twine, 2011 yılında Belçika'da düzenlenen Workshop on Lightweight Cryptography etkinliğinde önerilmiş bir hafif blok şifreleme algoritmasıdır. Algoritma 36 turdan oluşur ve sırasıyla 80 ve 128 bitlik anahtar uzunluklarını destekleyen Twine -80 ve Twine -128 olmak üzere iki versiyona sahiptir. 64 bitlik

blok şifreleme algoritması olan Twine bu çalışmada 128 bitlik veriyi şifreleme ve deşifreleme için kullanılacaktır.

Yöntemi açıklarken kullandığımız temel matematiksel işlemler aşağıdaki gibidir;

- $\oplus$: Bit düzeyinde özel VEYA'yı belirtir.
- D : Elemanları byte lardan oluşan dizi
- D(i) : D'nin i'ninci elemanı. En soldaki byte D (0)'dır.
- D(i − j) : i ≤ j koşulunda D'nin elemanlarının sıralanması i, (i + 1), ..., j
- D <<< i : D i bit sola kayması.
- x‖y : x ve y nin birleşimi
- RKi : 1 ≤ i ≤ 36 koşuluyla i. turda kullanılan 32 bitlik döngü anahtarı.
- K^i : k uzunluğundaki anahtarın k bit değeri
- X_j : X0'ın açık metin X36'nın şifreli metin olduğu i. turun çıktısı.

Şifreleme işlemi; Twine algoritmasının genel yapısı, 16 adet 4 bitlik alt bloğa sahip genelleştirilmiş Feistel yapısının bir çeşididir. Algoritmanın iki versiyonu da Şekil 3.4'de gösterilen döngü işlevine sahiptir ve toplam 36 döngüden oluşur.



Şekil 3.4 Twine tek tur şeması [45]

Şifreleme ve deşifreleme işlemlerinde Tablo 3.10'da verilen S-kutusu, Tablo 3.11'de verilen Pi ve evrik Pi tabloları kullanılmaktadır.

Tablo 3.10 Twine S-kutusu

x	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$S(x)$	C	0	F	A	2	B	9	5	8	3	D	7	1	E	6	4

Tablo 3.11 Pi ve Pi^{-1}kutuları

j	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
$P_i[j]$	5	0	1	4	7	C	3	8	D	6	9	2	F	A	B	E
$P_i^{-1}[j]$	1	2	B	6	3	0	9	4	7	A	D	E	5	8	F	12

Şifreleme işlemine ait sözde kod aşağıda verilmiştir [45].

$$
\begin{array}{l}
X^1_{(64)} \leftarrow P \\
RK^1_{(32)} \| \ldots \| RK^{35}_{(32)} \leftarrow RK_{(32\times 36)} \\
\textbf{for } i \leftarrow 1 \textbf{ to } 35 \\
\quad \textbf{do} \begin{cases}
X^i_{0(4)} \| X^i_{1(4)} \| \ldots \| X^i_{14(4)} \| X^i_{15(4)} \leftarrow X^i_{(64)} \\
RK^i_{0(4)} \| RK^i_{1(4)} \| \ldots \| RK^i_{6(4)} \| RK^i_{7(4)} \leftarrow RK^i_{(32)} \\
\textbf{for } j \leftarrow 0 \textbf{ to } 7 \\
\quad \textbf{do } X^i_{2j+1} \leftarrow S(X^i_{2j} \oplus RK^i_j) \oplus X^i_{2j+1} \\
\textbf{for } h \leftarrow 0 \textbf{ to } 15 \\
\quad \textbf{do } X^{i+1}_{\pi[h]} \leftarrow X^i_h \\
X^{i+1} \leftarrow X^{i+1}_0 \| X^{i+1}_1 \| \ldots \| X^{i+1}_{14} \| X^{i+1}_{15}
\end{cases} \\
\textbf{for } j \leftarrow 0 \textbf{ to } 7 \\
\quad \textbf{do } X^{36}_{2j+1} \leftarrow S(X^{36}_{2j} \oplus RK^{36}_j) \oplus X^{36}_{2j+1} \\
C \leftarrow X^{36}
\end{array}
$$

Deşifreleme işlemi; TWINE algoritmasında şifre çözme işlemi şifreleme işleminde olduğu gibi benzer adımlardan oluşur. Ters blok karıştırma ile şifrelemede kullanılan S-kutusu ve Pi dizileri şifre çözme işleminde de kullanılmaktadır. Aşağıda şifre çözme işleminin sözde kodu verilmiştir [45].

$$
\begin{array}{l}
X^{36}_{(64)} \leftarrow C \\
\mathrm{RK}^0_{(32)} \| \ldots \| \mathrm{RK}^{35}_{(32)} \leftarrow RK_{(32\times 36)} \\
\textbf{for } i \leftarrow 36 \textbf{ to } 2 \\
\quad \textbf{do} \begin{cases}
X^i_{0(4)} \| X^i_{1(4)} \| \ldots \| X^i_{14(4)} \| X^i_{15(4)} \leftarrow X^i_{(64)} \\
\mathrm{RK}^i_{0(4)} \| \mathrm{RK}^i_{1(4)} \| \ldots \| \mathrm{RK}^i_{6(4)} \| \mathrm{RK}^i_{7(4)} \leftarrow \mathrm{RK}^i_{(32)} \\
\textbf{for } j \leftarrow 0 \textbf{ to } 7 \\
\quad \textbf{do } X^i_{2j+1} \leftarrow S(X^i_{2j} \oplus \mathrm{RK}^i_j) \oplus X^i_{2j+1} \\
\textbf{for } h \leftarrow 0 \textbf{ to } 15 \\
\quad \textbf{do } X^{i-1}_{\pi^{-1}[h]} \leftarrow X^i_h \\
X^{i-1} \leftarrow X^{i-1}_0 \| X^{i-1}_1 \| \ldots \| X^{i-1}_{14} \| X^{i-1}_{15}
\end{cases} \\
\textbf{for } j \leftarrow 0 \textbf{ to } 7 \\
\quad \textbf{do } X^1_{2j+1} \leftarrow S(X^1_{2j} \oplus \mathrm{RK}^1_j) \oplus X^1_{2j+1} \\
P \leftarrow X^1
\end{array}
$$

3.6.3 Simon Algoritması

Simon algoritması kısıtlı kaynaklara sahip cihazlar için NSA tarafından tasarlanarak Haziran 2013 tarihinde halka açık yayınlanmış bir hafif blok kriptoloji algoritmasıdır. Algoritma ile yüksek performansta yüksek güvenlik sağlamak amaçlanmıştır. SIMON ailesi (SPECK ailesiyle birlikte), artan esnek, hafif kriptografi ihtiyacını karşılamak için önerilmiştir.

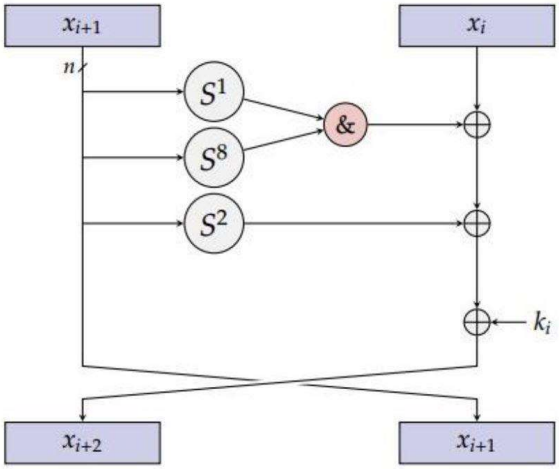
Tablo 3.12 Simon şifreleme ailesi

Blok Boyutu $2n$	Anahtar Boyutu mn	n Boyutu	m Anahtar Kelimesi	Sabit Dizi	Döngü
32	64	16	4	z_0	32
48	72	24	3	z_0	36
	96		4	z_1	36
64	96	32	3	z_2	42
	128		4	z_3	44
96	96	48	2	z_2	52
	144		3	z_3	54
128	128	64	2	z_2	68
	192		3	z_3	69
	256		4	z_4	72

SIMON şifreleme ailesi Tablo 3.12 de görüldüğü gibi farklı blok ve anahtar boyutuna sahip versiyonlar içermektedir. Bu özellikleri belirleyen bir dizi parametre vardır. Şifrelenecek blok uzunluğu 2n olarak alınırsa anahtar büyüklüğü 2, 3, ve 4 ün katı olacaktır. Böylelikle Simon şifreleme yapısı Simon2n/nm şeklinde de tanımlanabilir. 128 bitlik blok uzunluğuna sahip bir mesaj 128 bitlik bir anahtar ile şifrelenecek ise n = 64 ve m = 2 olarak bulunur.

Döngü fonksiyonu; Simon blok şifreleme algoritması dengeli Feistel şifresidir ve aşağıdaki işlemleri kullanır.

- $x \oplus y$: Bit düzeyinde XOR
- $x \,\&\, y$: Bit düzeyinde AND
- $S^y(x)$: Bit düzeyinde y bit sola dairesel kaydırma ROL
- $S^{-y}(x)$: Bit düzeyinde y bit sağa dairesel kaydırma ROR



Şekil 3.5 Simon döngü fonksiyonu [46]

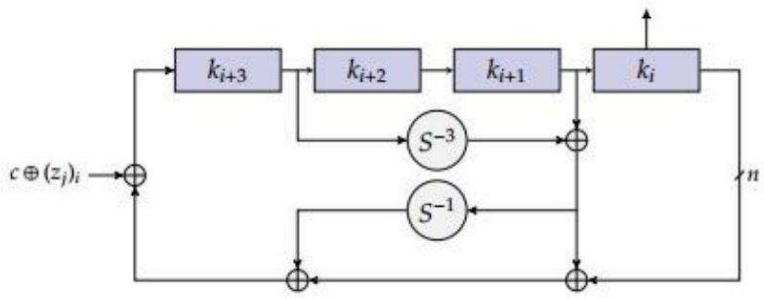
SIMON algoritması döngü fonksiyonu şeması Şekil 3.5'de sunulmuştur ve şifreleme işlemi için (3.5) eşitliği kullanılmaktadır.

$$R(l,r,k) = ((S^1(l) \,\&\, S^8(l)) \oplus S^2(l) \oplus \text{r} \oplus \text{l} \oplus \text{k}) \tag{3.5}$$

Deşifreleme işleminde kullanılan ters döngü fonksiyonu için (3.6) denklemi kullanılır.

$$R^{-1}(l,r,k) = (r, (S^1(r) \,\&\, S^8(r)) \oplus S^2(r) \oplus l \oplus k) \tag{3.6}$$

SIMON anahtar çizelgesi, ana anahtardan tüm alt grup anahtarları üreterek anahtar genişletme yetenekleri sağlar. Seçtiğimiz SIMON128/128 konfigürasyonunda, anahtar programı, ilk 128-bit ana anahtardan 44 adet 32-bit boyutlu alt anahtar üretir. Anahtar genişletme işlemi Şekil 3.6'daki gibidir.



Şekil 3.6 Simon anahtar üretimi [46]

Anahtar üretimi için (3.7) ve (3.8) denklemleri kullanılmaktadır.

$$K_j(k, c, z_j) = F(k_{i+3}, k_{i+1}) \oplus S^{-1}(F(k_{i+3}, k_{j+1})) \oplus k_j \oplus c \oplus (z_j)i \quad (3.7)$$

$$F(x, y) = S^{-3}(x) \oplus y \quad (3.8)$$

SIMON128/128 için: $c = 2^n - 4 \Rightarrow c = 2^{64} - 4$ olarak verilir.

Her genişletme işleminden sonra, önbelleğe alınmış alt grup anahtarlar, ilki atılarak ve sonuncusu yeni oluşturulan alt grup anahtarla değiştirilerek sağa döndürülür. Anahtar genişlemesinin son adımında kullanılacak SIMON boyutuna göre z_x sabit dizisi her turda anahtar kelimenin en düşük bitine uygulanmaktadır. Bizim kullandığımız yöntem SIMON128/128 olduğu için z_2 olacaktır.

z_2=10101111011100000011010010011000101000010001111110010110110011

Şifreleme işlemi; 128 bitlik bir açık metin bloğunun şifrelenmesi, anahtar fonksiyonu tarafından üretilen ilgili alt fonksiyonun 68 kez uygulamasından oluşur.

Deşifreleme işlemi; 128 bitlik şifreli metnin deşifreleme işlemi için önce sol ve en sağdaki 32 bitlik sözcüklerin değiştirilmesi gerekir. Ardından ters sıra ile döngü fonksiyonları 68 kez uygulanarak açık metin elde edilir [46, 47].

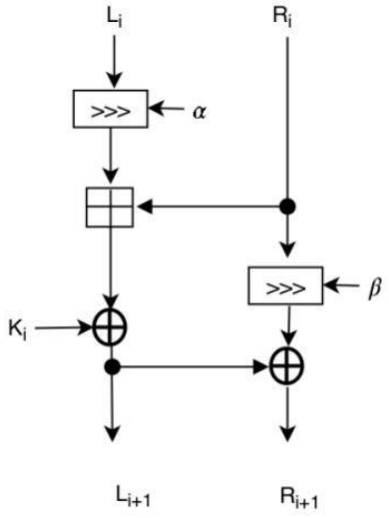
3.6.4 Speck Algoritması

Speck, NSA tarafından Haziran 2013'te halka açık olarak yayınlanan bir hafif blok şifreleme ailesidir. Speck, yazılım uygulamalarında performans için optimize edilmiştir, kardeş algoritması Simon ise donanım uygulamaları için optimize edilmiştir. Speck şifreleme ailesi Şekil 3.13'de verilmiştir.

Tablo 3.13 Speck şifreleme ailesi

Blok Boyutu	Anahtar Boyutu	n Boyutu	a	B	Döngü
32	64	16	7	2	22
48	72	24	8	3	22
	96		8	3	23
64	96	32	8	3	26
	128		8	3	27
96	96	48	8	3	28
	144		8	3	29
128	128	64	8	3	32
	192		8	3	33
	256		8	3	34

Döngü fonksiyonu toplama, döndürme ve XOR işlemlerinden oluşur. Şekil 3.7'de genel bir döngü şeması verilmiştir.



Şekil 3.7 Genel bir Speck şifreleme döngü şeması [48]

L_i ve R_i, i. kaydırma için girdinin sırasıyla sol ve sağ ara değerleridir. K_i , i. turda kullanılan n bit anahtarıdır. >>> α ve <<<β, α veya β bitleriyle dairesel sağa ve sola kaydırmayı ifade eder. $\oplus$ XOR işlemini, + ekleme işlemini temsil etmektedir. i. tur

için çıktılar L_{i+1} ve R_{i+1} elde edilmiş olur. Döngü işlemi (3.9) denklemi ile tanımlanabilir;

$$L_i = ((L_i >>> a) + R)) \oplus k_i \; ve \; R_{i+1} = (R_i <<< \beta) \oplus L_{i+1} \quad (3.9)$$

Anahtar üretimi için ilk m uzunluğundaki ana anahtarını $(l_{m-2}, \ldots\ldots\ldots, l_0, k_0)$ tur sayısı miktarınca $(k_0, k_1, \ldots\ldots\ldots, k_{ith})$ genişletir, ardından k_i ve l_i değerlerinden (3.10) denklemi kullanılarak iki dizi oluşturulur.

$$l_{i+m-1} = ((k_i + (l_i >>> \alpha)) \oplus i \; ve \; k_{i+1} = (k_i <<< \beta) \oplus l_{i+m-1} \quad (3.10)$$

Şifreleme işlemi; Uygulamamızda Speck128/128 şifreleme varyantını kullandığımız için 128 bitlik açık metin bloğunun şifrelenmesi, yukarıda verilen anahtar üretim fonksiyonun 32 kez uygulamasıyla oluşur.

Deşifreleme işlemi; 128 bitlik şifreli metnin deşifreleme işlemi için ters sıra ile döngü fonksiyonları 32 kez uygulanarak açık metin elde edilir [48].

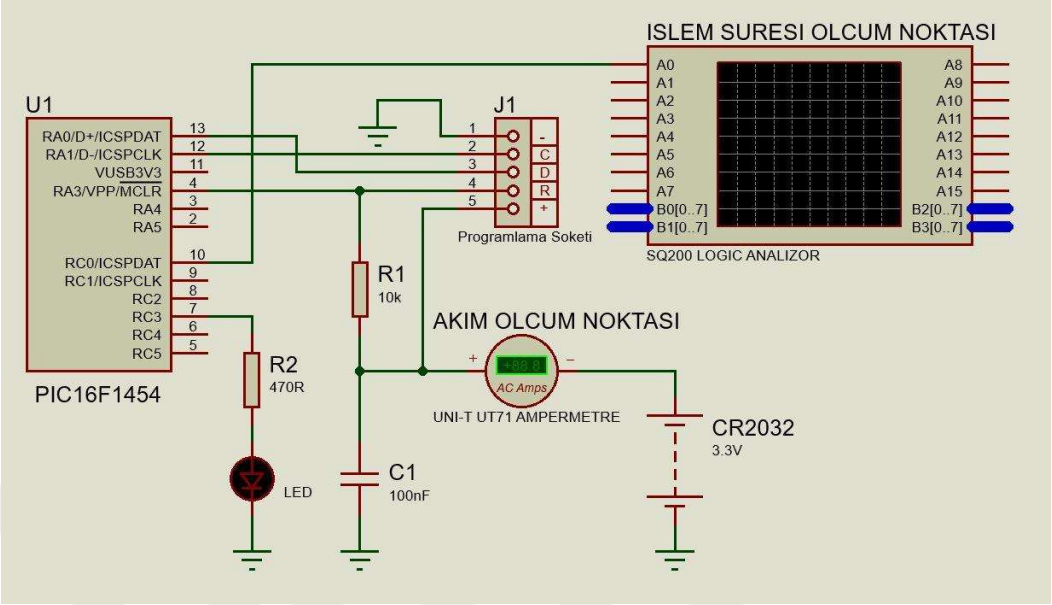
4
TEST DONANIMLARININ TASARIMI

Bu bölümde kullanılan materyaller sunulmuştur. Hafif şifreleme algoritmalarının yazılım uygulamaları gerçekleştirilerek sonuçları verilmiştir.

4.1 Kullanılan Materyaller

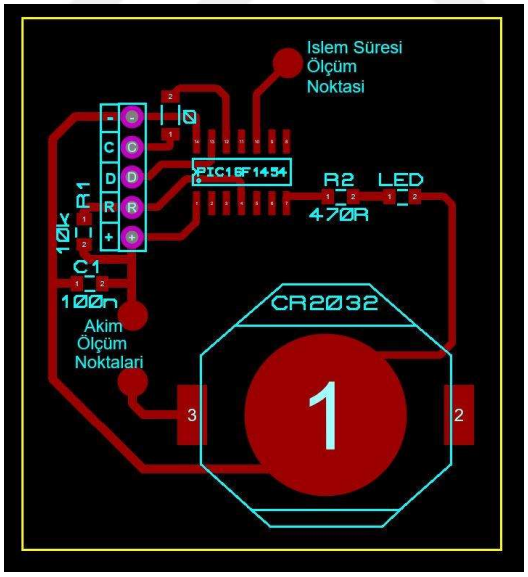
Önceki bölümde de bahsedildiği gibi geleneksel kriptografi yöntemleri yüksek işlem yeteneğine sahip, bellek ve enerji kaynak sınırlaması olmayan serverlar, bilgisayarlar, tabletler ve akıllı telefonlar donanımlarda kullanılmaktadır. Bellek, işlem gücü ve enerji kaynağı açısından kaynakların sınırlı olduğu RFID etiketler, sensör ağları, uzaktan kumanda sistemleri gibi kablosuz haberleşen IoT uygulamalar için hafif şifreleme yöntemleri geliştirilmiştir. Bu uygulamalarda genellikle işlem gücü ve hafıza yetenekleri sınırlı olan özel bir görevi gerçekleştirmek için tasarlanmış mikrodenetleyiciler kullanılmaktadır. Bu çalışmada hafif şifreleme yöntemleri düşük işlem gücüne sahip 8-bit 8051, 8-bit RISC ve 32-bit ARM Cortex M0 mimarili mikrodenetleyiciler üzerinde yazılım tabanlı uygulaması yapılmıştır. Mikrodenetleyiciler 4 MHz, 8 MHz ve 16 MHz saat frekanslarında çalıştırılmıştır.

Testlerde kullanılan RISC mimarisine sahip 8-bit PIC16f1454 mikrodenetleyicisi 14 bacaklı ve 48 MHz dahili hassas osilatör ile çalışmaktadır. 14 Kb Flash hafıza kapasitesi bulunmaktadır. Kullanıcılara 2.3V-5.5V arasında çalışma gerilim aralığı sunmaktadır. AES-128 algoritması Microchip firması tarafından kendi mikrodenetleyicilerini programlamak için geliştirdiği MPLAB X programı üzerinden yazılıp derlenmiştir [49]. Test kartının Proteus 8 programı kullanılarak oluşturulan devre şematiği Şekil 4.1'de verilmiştir.



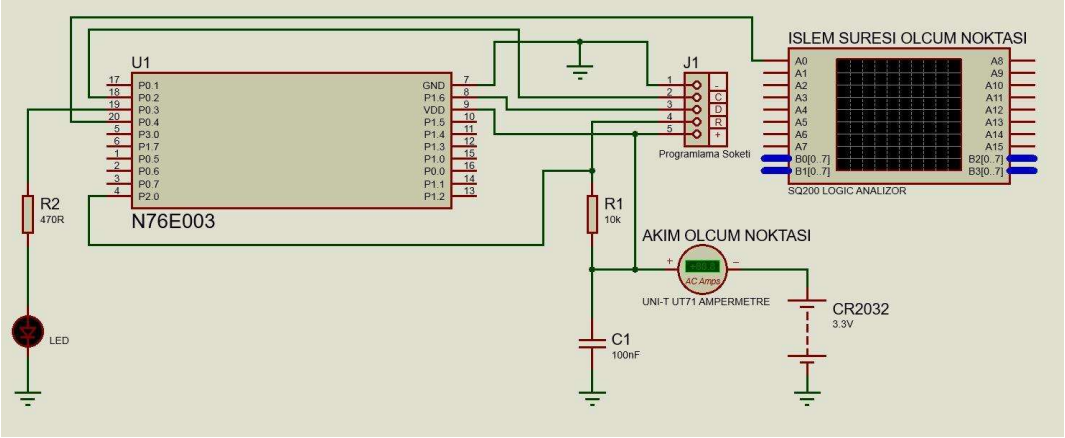
Şekil 4.1 PIC16f1454 mikrodenetleyicisinin test devresi şematiği

Oluşturulan şematik üzerinden devrenin çizimi Proteus 8 programı kullanılarak Şekil 4.2'deki gibi oluşturulmuştur.



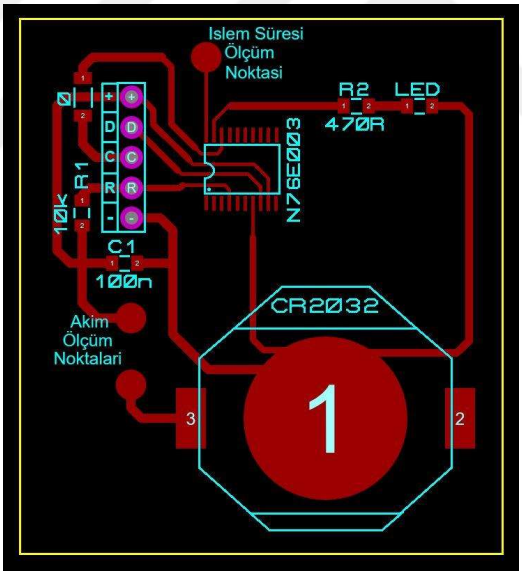
Şekil 4.2 PIC16f1454 test devresi baskı devre çizimi

Analizde 8051 mikrodenetleyici olarak Nuvoton firması tarafından üretilen, 18Kb Flash hafızaya, 16 MHz dahili hassas osilatöre sahip N76E003 kullanılmıştır. 2.4V-5.5V aralığında besleme gerilimi ile çalışan entegre 20 bacaklı 8-bit veri yoluna sahiptir. Şifreleme algoritması Keil uVision 5 programı üzerinde C dilinde yazılarak derlenmiştir [50]. Oluşturulan test devresinin şematiği Şekil 4.3'de verilmiştir.



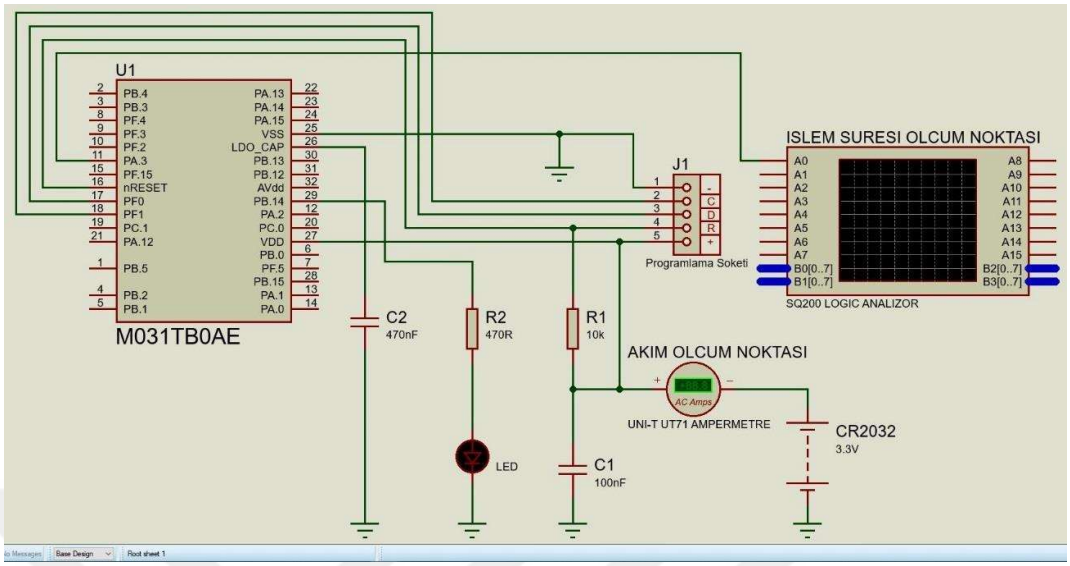
Şekil 4.3 N76E003 mikrodenetleyicisinin test devresi şematiği

Devrenin baskı devre çizimi Proteus programı yardımıyla Şekil 4.4'deki gibi gerçekleştirilmiştir.



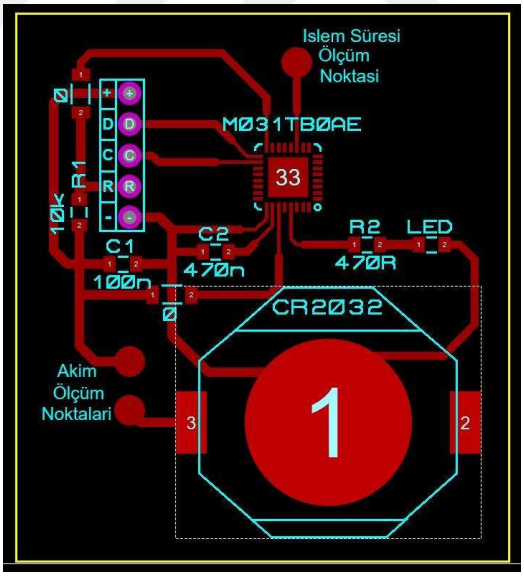
Şekil 4.4 N76E003 test devresi baskı devre çizimi

Son olarak testlerde 32-bit ARM Cortex M0 tabanlı Nuvoton firması tarafından geliştirilen M031TB0AE entegresi kullanılmıştır. 16 Kb flash hafızası bulunan mikrodenetleyici 1.8V-3.6V çalışma gerilimine sahiptir. Dahili olarak maksimum 48 Mhz çalışma frekansı ile komutları işlemektedir. Program geliştirilmesi Keil uVision 5 derleyicisi üzerinden yapılmıştır. Test kartlarının devre şeması içimi Şekil 4.5'de sunulduğu gibi Proteus 8 programı kullanılarak oluşturulmuştur.



Şekil 4.5 M031TB0AE mikrodenetleyicisinin test devresi şematiği

Test devresi için baskı devre çizimi Proteus programı yardımıyla Şekil 4.6'deki gibi tasarlanmıştır.



Şekil 4.6 M031TB0AE test devresi baskı devre çizimi

Testlerde kullanılan mikrodenetleyicilerin genel özellikleri karşılaştırmalı olarak Tablo 4.1'de verilmiştir.

Tablo 4.1 Analiz edilen mikrodenetleyici özellikleri

	PIC 16F1454	N76E003	M031TB0AE
CPU	8-bit RISC	8-bit 8051 (CISC)	32-bit RISC ARM Cortex M0
Frekans (MHz)	48	16	48
Mimari	Harvard	Von Neumann	Harvard
Flash Hafıza (KB)	14	18	16
SRAM(Byte)	256	1024	2048
EEPROM(KB)	-	-	-
Çalışma Voltajı(V)	2.3-5.5	2.4-5.5	1.8-3.6
I/O Pin Sayısı	11	18	27
Programlama Platformu	MPLAB X	Keil uVision5	Keil uVision5

Akım ölçümü için masaüstü uygulaması da bulunan UNI-T UT71D ölçü aleti kullanılmıştır [51]. İşlem süreleri baskı devrelerin üzerine konulan ölçüm pedleri aracılığıyla ikalogic sq200 logic analizörü kullanılarak elde edilmiştir [52].

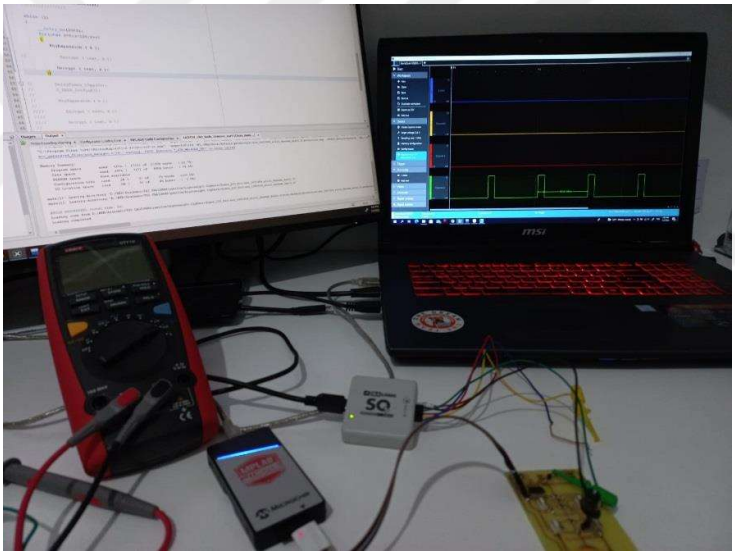
Bu tez kapsamında test edilecek hafif blok şifreleme algoritmaları ve özellikleri Tablo 4.2' de verilmiştir. Her algoritma 128 anahtar uzunluğu ve 128 blok boyutuna sahiptir. Twine algoritması blok boyutu 64-bit olmasına karşın sonuçların adil kıyaslanması için 128-bit blok boyutu kullanılmıştır [3].

Tablo 4.2 AES, Simon, Speck ve Twine özellikleri

ALGORİTMA	GELİŞTİRİCİ	SUNULMA YERİ-TARİHİ	BLOK BOYUTU(Bit)	ANAHTAR BOYUTU(Bit)	MİMARİ	TUR SAYISI
AES	Rijmen et al.	AES konferansı 1998	128	128	SPN	10
SIMON	Beaulieu et al.	eprint.iacr 2013	128	128	Feistel	68
SPECK	Beaulieu et al.	eprint.iacr 2013	128	128	ARX	32
TWINE	Suzaki et al.	Workshop on LC 2011	64*	128	GFN	36

4.2 Yöntem ve Test

AES-128 algoritması yukarıda da belirtilen derleyici programlar kullanılarak C dilinde derlenmiş ve her bir mikrodenetleyiciye yüklenmiştir. Öncelikle algoritmanın şifreleme ve deşifreleme kodlarının her bir mikrodenetleyicide kullandıkları Flash ve RAM boyutları hesaplanmıştır. Ardından 8-bit RISC mimarili PIC16F1454, 8-bit 8051 mimarili N76E003 ve M031TB0AE mikrodenetleyicileri sırasıyla 4Mhz, 8Mhz ve 16Mhz saat frekanslarında çalıştırılmıştır. Her bir saat frekansı için şifreleme ve deşifreleme süreleri deneme kartları üzerindeki test noktaları aracılığıyla ikalogic sq200 logic analizörü kullanılarak ölçülmüştür. Test ortamı Şekil 4.7'de verilmiştir.



Şekil 4.7 Test ortamı

Harcanan enerjiyi hesaplamak için devreler 3V gerilim değerine sahip CR2032 pili kullanılarak enerjileri sağlanmıştır [53]. İlk olarak mikrodenetleyicilerin yukarıda belirtilen her bir çalışma frekansında boşta çalışma akımları UNI-T UT71 ölçü aleti kullanılarak ölçülmüştür. Daha sonra şifreleme ve deşifreleme işlemi gerçekleştirilerek işlem boyunca çekilen akımlar ölçülmüştür. İşlem süreleri daha önceden hesaplandığı için (4.1) denklemi kullanılarak harcanan enerjiler elde edilmiştir.

$$\text{Harcanan enerji(uJ)} = \text{Çalışma gerilimi(V)*Çekilen akım(uA)*İşlem süresi(ms)/1000} \quad (4.1)$$

$$\text{Veri işleme hacmi (Kbit/sn)} = \text{Blok boyutu (128)} / \text{İşlem süresi} \quad (4.2)$$

$$\text{Data işleme verimi (Kbit/sn.A)} = \frac{\text{Veri işleme hacmi/İşlem süresince harcanan akım(A)}}{} \quad (4.3)$$

Her bir şifreleme algoritması ile 128-bit veri blokları şifrelenip daha sonra deşifreleme işlemi yapılmıştır. Veri işleme hacmini (4.2) denklemi kullanılarak hesaplanmaktadır. Ayrıca (4.3) eşitliliği kullanılarak mikrodenetleyicilerin veri işleme hacimleri, o işlemi yaparken harcadıkları akıma oranlayarak işlemcilerin Data işleme verimleri hesaplanmıştır. Elde edilen data işleme verileri daha anlaşılır hale getirilmesi amacıyla 100'e bölünerek sadeleştirilmiştir.

5
TEST SONUÇLARI

Bu kısımda tasarlanan donanımlar ile AES128/128, Simon128/128, Speck128/128 ve Twine128/128 algoritmaları koşturulmuştur. Şifreleme ve deşifreleme işlemleri için harcanan süreler, Flash ve RAM kullanımları, mikrodenetleyicilerin boşta çalışma akımları ve algoritmaların mikrodenetleyiciler üzerinde koşturulması esnasında harcanan enerjiler fiziksel olarak ölçümlenerek test sonuçları elde edilmiştir.

5.1 Bellek Kullanımı

Yazılım uygulamaları için Flash ve RAM hafızalarının kullanımı, hedeflediğimiz sınırlı kaynakları olan cihazlar için önemlidir. Flash hafıza programlama kodunu depolamak için kullanılır. Algoritmada kullanılan S-kutuları, anahtarlar gibi sabit değerler yine bu hafızada tutulur. RAM ise algoritma çalışırken hesaplamalarda ortaya çıkan ara değerleri depolamak için kullanılır. Bu bölümde algoritmaların şifreleme ve deşifreleme işlemi için kapladıkları Flash hafıza ve RAM kullanım boyutları ayrı ayrı hesaplanarak sunulmuştur.

Yazılım tabanlı olarak C dilinde derlenen AES128/128, Simon128/128, Speck128/128 ve Twine128/128 algoritmaları her bir işlemcide test edilmiştir. Algoritmaların şifreleme bölümlerinin Flash hafızada kapladıkları alan Tablo 5.1'de sunulmaktadır.

Tablo 5.1 Şifreleme algoritmalarının bellek kullanımı

	8-Bit RISC (byte)		**8-Bit 8051 (byte)**		**32-Bit ARM (byte)**	
	FLASH	RAM	FLASH	RAM	FLASH	RAM
AES	1.085	85	1.698	583	1.072	48
SIMON	857	691	1.630	346	532	624
SPECK	454	570	1.500	264	252	528
TWINE	888	388	1.653	518	588	368

Algoritmaların deşifreleme bölümlerinin Flash hafızada kapladıkları alan Tablo 5.2'de sunulmaktadır.

Tablo 5.2 Deşifreleme algoritmalarının bellek kullanımı

	8-Bit RISC (byte)		**8-Bit 8051 (byte)**		**32-Bit ARM (byte)**	
	FLASH	RAM	FLASH	RAM	FLASH	RAM
AES	1.469	257	1.896	746	1.396	208
SIMON	874	691	1.739	346	534	624
SPECK	562	570	1.586	264	264	528
TWINE	880	388	1.622	518	580	368

5.2 İşlem Süreleri

Algoritmalar veriyi şifrelerken ve deşifrelerken işlemleri belirli bir sürede gerçekleştirmektedir. İşlem süresini algoritmanın karmaşıklığı, şifrelenecek verinin boyutu, mikroişlemcinin mimarisi ve çalışma frekansı doğrudan etkilemektedir. Şifreleme ve deşifreleme sürelerine kadar artarsa tüketilen enerji artmaktadır. Ayrıca işlem süreleri veri işleme hacmini doğrudan etkilemektedir ve gecikme/tepki sürelerini de artırmaktadır. Çalışmamızda her bir algoritmanın farklı mimarideki işlemcilerde 128-bit veriyi şifreleme ve deşifrelemesi için geçen süreler sunulmuştur.

Algoritmaların şifreleme işlemi için harcadıkları süreler her bir mikrodenetleyici için Tablo 5.3'de sunulmaktadır.

Tablo 5.3 Şifreleme işlem süreleri

	8-Bit RISC (ms)			**8-Bit 8051 (ms)**			**32-bit ARM (ms)**		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	54	27	14	23	11	6	4,5	2,3	1,1
SIMON	304	151	76	125	62	31	2,1	1,1	0,5
SPECK	76	38	19	49	24	12	0,9	0,4	0,2
TWINE	89	44	21	61	30	15	8	4	2

Algoritmaların deşifreleme işlemi için harcadıkları süreler her bir mikrodenetleyici için Tablo 5.4'de sunulmaktadır.

Tablo 5.4 Deşifreleme işlem süreleri

	8-Bit RISC (ms)			8-Bit 8051 (ms)			32-Bit ARM (ms)		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	597	299	149	81	41	20	34	17	8
SIMON	304	152	76	126	62	31	2,2	1,1	0,6
SPECK	97	49	24	52	26	13	0,9	0,5	0,2
TWINE	88	44	21	61	30	15	8	4	2

Veri işleme hacmi(throughtput) şifreleme ve deşifreleme işleminde şifreli metnin veya açık metnin üretilme hızını göstermektedir. Geleneksel kriptografi yöntemlerinin aksine hafif kriptografi tasarım hedeflerinden biri olmayabilir. Çünkü hafif şifrelemenin hedeflediği uygulamalar yüksek veri alışverişi gereken uygulamalar değildir. Bununla birlikte, çoğu uygulamada orta düzeyde veri işleme hacmi gerekmektedir [2].

Yapılan çalışmada veri işleme hacimleri elde edilen işlem süreleri kullanılarak hesaplanmıştır. Her bir algoritmanın şifreleme yapısı farklı çalışma frekansı ve işlemci mimarilerinde denenmiş ve Tablo 5.5'daki sonuçlar elde edilmiştir.

Tablo 5.5 Şifreleme veri işleme hacmi

	8-Bit RISC (Kbit/sn)			8-Bit 8051 (Kbit/sn)			32-Bit ARM (Kbit/sn)		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	2,4	4,7	9,4	5,6	11,2	22,5	28,2	56,4	113,3
SIMON	0,4	0,8	1,7	1,0	2,1	4,1	60,4	120,8	241,5
SPECK	1,7	3,4	6,7	2,6	5,3	10,7	150,6	304,8	609,5
TWINE	1,4	2,9	6,1	2,1	4,3	8,5	16,0	32,0	64,0

Her bir algoritmanın deşifreleme yapısı farklı çalışma frekansı ve işlemci mimarilerinde denenmiş ve Tablo 5.6'daki sonuçlar elde edilmiştir.

Tablo 5.6 Deşifreleme veri işleme hacmi

	8-Bit RISC (Kbit/sn)			8-Bit 8051(Kbit/sn)			32-Bit ARM (Kbit/sn)		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	0,2	0,4	0,9	1,6	3,2	6,4	3,8	7,6	15,3
SIMON	0,4	0,8	1,7	1,0	2,1	4,1	58,2	116,4	232,7
SPECK	1,3	2,6	5,3	2,5	4,9	9,8	147,1	284,4	581,8
TWINE	1,5	2,9	6,1	2,1	4,3	8,5	16,0	32,0	64,0

5.3 Harcanan Enerji

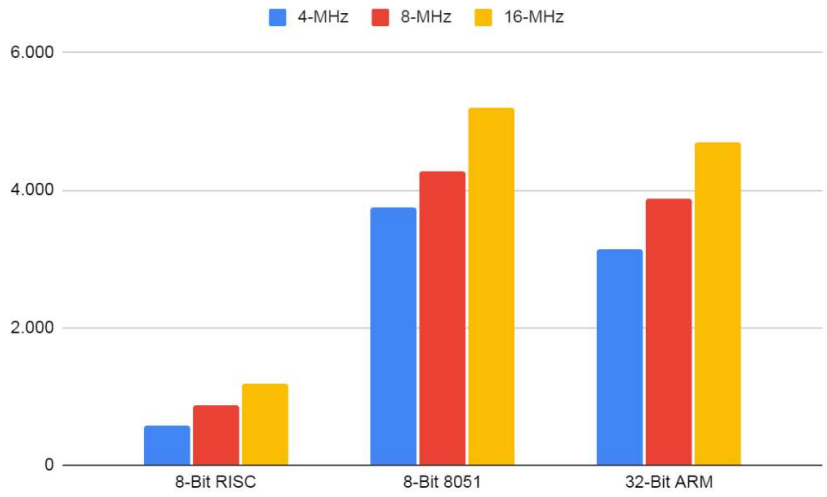
Hafif kriptolojinin kullanımının hedeflendiği RFID, Sensör Ağları, Uzaktan kumanda sistemleri gibi kablosuz haberleşen IoT uygulamaları enerji ihtiyaçlarını batarya üzerinden sağlamaktadırlar. Kablosuz haberleşen bu sistemlerin uzun süreler şarj olmadan bataryalarının dayanması beklenir. Bazı cihazlarda ise tekrar şarjın zor veya imkânsız olduğu durumlar olmaktadır. Bundan dolayı uygulamanın düşük enerji harcamaları gerekmektedir.

Enerji tüketimi birçok faktöre bağlı olarak değişmektedir. Besleme gerilimi, çalışma frekansı, işlemci mimarisi gibi birçok unsurun yanında algoritmanın gerçekleştirilmesi için gereken işlem yoğunluğu da enerji tüketimini etkileyen önemli bir etkendir.

Bu çalışmada öncelikle kullanılan mikrodenetleyicilerin 4 MHz, 8 MHz ve 16 MHz de hiçbir işlem yapmadan boşta çalışma akımları UNI-T UT71 cihazı kullanılarak Tablo 5.7 ve Şekil 5.1'de verilmiştir. Mikrodenetleyicilerin derin uyku veya uyku modları aktif değilken ölçümler yapılmıştır.

Tablo 5.7 Mikrodenetleyicilerin boşta çalışma akımları

	4-MHz (µA)	8-MHz (µA)	16-MHz (µA)
8-Bit RISC	577	872	1.179
8-Bit 8051	3.750	4.265	5.210
32-Bit ARM	3.130	3.880	4.700



Şekil 5.1 Çalıma frekanslarına göre boşta çalışma akımları(µA)

Algoritmaların şifreleme işlemlerine ait kodlar C dilinde yazılıp MpLab X ve Keil uVision 5 programlarında derlenerek Proteus ile çizilen deneme kartlarına yüklenmiştir. Çalışmaları esnasında çektikleri akımlar UNI-T UT71 ile ölçülmüştür. Elde edilen değer ile (4.1) deki formülde yerine konarak her bir algoritmanın şifreleme işlemi için harcadıkları enerji Tablo 5.8'deki gibi elde edilmiştir.

Tablo 5.8 Algoritmaların şifreleme işleminde harcadıkları enerji

	8-Bit RISC (µJ)			**8-Bit 8051 (µJ)**			**32-Bit ARM (µJ)**		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	106	80	55	302	171	104	51	34	22
SIMON	607	469	327	1.563	904	558	37	25	17
SPECK	150	115	80	611	342	311	10	6	4
TWINE	173	128	84	762	434	268	90	59	39

Algoritmaların deşifreleme işlemlerine ait kodlar yine C dilinde yazılıp MpLab X ve Keil uVision 5 programlarında derlenerek deneme kartlarına yüklenmiştir. Çalışmaları esnasında çektikleri akımlar UNI-T UT71 ile ölçülmüştür. Elde edilen değer ile (4.1) deki formülde yerine konarak her bir algoritmanın deşifreleme işlemi için harcadığı enerji Tablo 5.9'deki gibi elde edilmiştir.

Tablo 5.9 Algoritmaların deşifreleme işleminde harcadıkları enerji

	8-Bit RISC (µJ)			8-Bit 8051 (µJ)			32-Bit ARM (µJ)		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	1.168	879	637	1.037	598	357	383	250	156
SIMON	607	473	327	1.572	904	558	39	23	17
SPECK	190	148	101	648	370	229	10	7	4
TWINE	170	127	58	762	434	268	90	59	39

Kullanılan her mikrodenetleyici çalıştırıldıkları frekanslardaki veri işleme hacimlerini sunmuştuk. Bu işlemleri yaparken belirli bir akım tüketirler. Yapılan çalışmada işlemcilerin veri işleme hacimleri, işlem sırasında çekilen akıma oranlanarak mikrodenetleyicilerin çalıştırıldıkları frekanstaki data işleme verimleri elde edilmiştir. Açık metnin şifreleme işlemleri için mikrodenetleyicilerin data işleme verimleri Tablo 5.10'de sunulmuştur.

Tablo 5.10 Mikrodenetleyici data işleme verimi-şifreleme

	8-Bit RISC (Kbit/sn.A)			8-Bit 8051 (Kbit/sn.A)			32-Bit ARM (Kbit/sn.A)		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	40	53	77	14	25	41	83	125	195
SIMON	7	9	13	3	5	8	171	259	385
SPECK	28	37	53	7	12	20	443	674	1.009
TWINE	24	33	50	6	10	16	47	72	109

Açık metnin elde edildiği şifre çözme işlemlerinde mikrodenetleyicilerin data işleme verimleri de Tablo 5.11'de sunulmuştur.

Tablo 5.11 Mikrodenetleyici data işleme verimi-deşifreleme

	8-Bit RISC (Kbit/sn.A)			8-Bit 8051 (Kbit/sn.A)			32-Bit ARM (Kbit/sn.A)		
	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz	4-MHz	8-MHz	16-MHz
AES	4	5	7	4	7	12	11	17	27
SIMON	7	9	13	3	5	8	164	277	371
SPECK	22	29	42	7	11	18	425	570	949
TWINE	25	33	50	6	10	16	47	72	108

6
SONUÇ VE ÖNERİLER

Bu tezde, 128-bit anahtar uzunluğuna sahip AES, SIMON, SPECK ve TWINE hafif şifreleme algoritmaları açıklanmıştır. Farklı mimarili mikrodenetleyicilerde algoritmaların bellek kullanımı, işlem süreleri enerji tüketimleri açısından performansları analiz edilmiştir.

[4]'de yapılan çalışmada yazılım tabanlı hafif blok ve akış şifreleme algoritmalarının işlem süreleri, Flash hafıza ve RAM kullanımı açısından performanslarının ölçüldüğü FELICS önerilmiştir. FELICS de kullanılan hedef cihazlardan farklı olarak bu çalışmada 16-bit RISC mimarisi yerine 8-bit CISC mimarili mikrodenetleyici kullanılmıştır.

Bellek kullanımı açısından yapılan analiz sonucunda flash hafıza kullanımı açısından 32-bit işlemci diğerlerine göre daha iyi sonuç vermiştir. 8-bit 8051 işlemcisi ise bellek organizasyonu açısından diğerlerinin gerisinde kalmaktadır. [1] de sunulan çalışmada olduğu gibi, tez kapsamında yapılan analiz sonuçlarında Speck algoritması hem şifreleme hem de deşifreleme işlemleri için diğer algoritmalara göre flash hafızayı en az kullanan yöntemdir. AES algoritması ise diğerlerine göre daha fazla flash hafızada yer kaplamaktadır. Ayrıca AES algoritmasının deşifreleme işlemi için kullandığı hafıza şifreleme işlemi için kullandığından daha fazlayken, diğer algoritmalar şifreleme ve deşifreleme işlemleri için neredeyse aynı miktarda flash bellek kullanmaktadır. [1]'de yapılan çalışmada AES-128 algoritması 8-bit kod boyutu olarak 1246 bayt bellekte yer kaplarken, tez kapsamında 1085 bayt elde edilmiştir. RAM boyutu olarak [1]'de elde edilen sonuç 81 bayt iken tez çalışmasında 85 olarak gerçekleşmiştir.

Tez kapsamında odaklanılan bir diğer konu ise algoritmaların işlem süreleridir. Algoritmalar farklı mikrodenetleyici ve frekanslarda çalıştırılarak sonuçlar analiz edilmiştir. Elde edilen sonuçlar, 32-bit işlemcilerin aynı frekanslarda bile çok daha hızlı işlem yaptıklarını göstermektedir. 8-bit 8051 mimarili işlemci, 8-bit RISC mimarili işlemciye göre daha hızlı işlem yapmaktadır. Şifreleme işleminde AES

algoritması, Speck algoritmasından daha hızlı işlense de deşifreleme işleminde Speck algoritması daha hızlıdır. Ayrıca veri işleme hacmi bakımından Speck algoritması daha iyi sonuç vermektedir. [7]'de yapılan çalışmada AES algoritması 8-bit RISC ve 8-bit CISC mikrodenetleyicisi ile test edilmiştir. Tez kapsamında elde edilen sonuçlar [7]'de sunulan sonuçlardan daha düşüktür.

İşlem süresince harcanan enerjiye oranla işlenen veri miktarını gösteren mikrodenetleyici data işleme verimleri çalışmada sunulmuştur. Tez kapsamında elde edilen sonuçlar şifreleme ve deşifreleme işlemlerinde harcanan enerjiye karşılık en yüksek verimi sağlayan işlemcinin 16 MHz çalışma frekansında 32-bit ARM olduğunu göstermektedir. 8-bit 8051 mimarisi, 8-bit RISC mimarisine göre daha hızlı işlem yapsa da verim olarak 8-bit RISC işlemcisinin gerisinde kalmaktadır.

Son olarak algoritmaların çalıştırılması esnasında mikrodenetleyicilerin harcadıkları enerji her bir çalışma frekansı için sunulmuştur. Enerji hesaplamadan önce her bir işlemcinin boşta çalışma akımları hesaplanmıştır. Verilere göre en yüksek akımı 8-bit 8051 mimarisi, daha sonra 32-bit ARM ve en düşük de 8-bit RISC tüketmektedir. Algoritmalar şifreleme ve deşifreleme işlemleri için harcadıkları enerjiler hesaplanarak tez kapsamında sunulmuştur. Şifreleme ve deşifreleme işlemleri için en düşük enerji 16 MHz de çalıştırılan 32-bit ARM mikrodenetleyicisi harcamaktadır. Algoritmalar arasında ise şifreleme işlemi için en düşük enerji harcayan AES şifreleme algoritmasıdır. Deşifreleme işleminde ise en düşük enerji tüketimi Speck hafif şifreleme algoritmasına aittir. [8]'de yapılan çalışmada 16 MHz çalışma frekansında, 16-bit mikrodenetleyici üzerinden farklı hafif şifreleme algoritmaları çalıştırılmış ve enerji analizleri yapılmıştır. Tez kapsamında yaptığımızdan farklı olarak harcanan enerji Energy Trace++ uygulaması kullanılarak hesaplanmıştır.

Yapılacak tasarımlarda bellek kullanımı önemli bir faktördür. Kullanılan işlemci hafızasına göre algoritma seçimi yapılmalıdır. Düşük güçlü IoT uygulamalarında genelde orta düzeyli veri işleme hacmi yeterli olsa da alarm sistemleri gibi gecikmenin istenmediği uygulamalarda hızlı olan algoritma seçilerek daha yüksek çalışma frekansları ile çalışılabilir. Pilli uygulamalar için ise gönderilen veri

sıklığına göre tasarım yapılmalıdır. Yüksek sıklıkta veri alışverişi yapan uygulamalarda hızlı işlemciler pil ömrü açısından daha yararlı olurken, düşük aralıklarla veri alışverişi yapılan uygulamalarda boşta çalışma akımı düşük olan mikrodenetleyiciler tercih edilebilir.

KAYNAKÇA

[1] Buchanan, William J., Shancang Li, and Rameez Asif. "Lightweight cryptography methods." Journal of Cyber Security Technology 1.3-4 (2017): 187-201.

[2] K. A. McKay, L. Bassham, M. Sönmez Turan, and N. Mouha, Report on lightweight cryptography, 2017, NIST Internal Report (IR) 8114

[3] Université du Luxembourg, Lightweight Block Ciphers, https://www.cryptolux.org/index.php/Lightweight_Block_Ciphers, [Son erişim tarihi; January 10, 2022]

[4] Dinu D, Biryukov A, Großschädl J, et al. FELICS – fair evaluation of lightweight cryptographic systems. In: NIST Workshop on Lightweight Cryptography 2015, National Institute of Standards and Technology (NIST) (2015)

[5] Eisenbarth, T., Kumar, S., Paar, C., Poschmann, A., & Uhsadel, L. (2007). A survey of lightweight-cryptography implementations. IEEE Design & Test of Computers, 24(6), 522-533.

[6] Biryukov, A., & Perrin, L. P. (2017). State of the art in lightweight symmetric cryptography (2017).

[7] Hyncica, Ondrej, et al. "Performance evaluation of symmetric cryptography in embedded systems." Proceedings of the 6th IEEE international conference on intelligent data acquisition and advanced computing systems. Vol. 1. IEEE, 2011.

[8] Aslan, Bora, Füsun Yavuzer Aslan, and M. Tolga Sakallı. "Energy Consumption Analysis of Lightweight Cryptographic Algorithms That Can Be Used in the Security of Internet of Things Applications." Security and Communication Networks 2020

[9] Pfleeger, C.P. 1997. The fundamentals of information security. Software, IEEE ,14 (1,14)

[10] Reetu Gupta and Rahul Gupta, 'ABC of Internet of Things: Advancements, Benefits, Challenges, Enablers and Facilities of IoT', IEEE 2016, 2016 Symposium on Colossal Data Analysis and Networking (CDAN), 978-1-5090-0669-4/16.

[11] ITU Measuring the Information Society Report 2015

[12] Samet Akkuş, ''Gömülü Sistem Tabanlı, Kriptolu TCP/IP Veri Haberleşmesi Uygulaması'' Yüksek Lisans Tezi, 2015

[13] Al-Fuqaha, Ala et al., Internet of Things: A Survey on Enabling Technologies, Protocols and Applications. Communications Surveys & Tutorials, IEEE. 2015. Volume: PP, Issue: 99.

[14] Ülker, Ülkü. "Klasik Teknikler Kullanılarak Bir Kriptografi Algoritması Geliştirilmesi ve Des Algoritması ile." 2014

[15] Saadin Oyucu, Hüseyin Polat, Bir Bütün Olarak M2M ve IoT Güvenliği, Conferece ISCTURKEY 2017

[16] Winkler, T.; Rinner, B. Security and Privacy Protection in Visual Sensor Networks: A Survey. ACM Comput. Surv. 2014, 47, 97–116

[17] Dargie, W. and Poellabauer, C. (2010). Fundamentals of wireless sensor networks: theory and practice. John Wiley and Sons. pp.168–183, 191–192. ISBN 978-0-470-99765-9.

[18] Tuncay Soylu, "Kablosuz Algılayıcı Ağların Uygulama Alanları ve Bir Algılayıcı Düğüm Tasarımı" Yüksek Lisans Tezi, Temmuz, 2012

[19] I. Butun, P. Österberg, H. Song "Security of the Internet of Things: Vulnerabilities, Attacks, and Countermeasures" IEEE Communications Surveys & Tutorials, vol.22, Issue: 1, pag.616- 644 2020

[20] Ivana Tomi; Julie A. McCann, "A Survey of Potential Security Issues in Existing Wireless Sensor Network Protocols" IEEE Internet of Things Journal (Volume: 4, Issue: 6, Dec. 2017)

[21] Bilgi Teknolojileri ve İletişim Kurumu, "Milli Frekans Planı" 2019

[22] S. R. Geerlings, L.D. Vredevoogd, D. A. Bleker, J. D. Spencer, (Mayıs 2012) "System and Method for Training a transmitter to Control a Remote Control System" Patent no: US 8,174,357 B2

[23] E.Söğüt, O.Ayhan Erdem, "Günümüzün Vazgeçilmez Sistemleri: Nesnelerin Haberleşmesi ve Kullanılan Teknolojiler" AB 2017 Akademik Bilişim Konferansları, 2017

[24] Mutlu A. K., Sürmeli C. "Mikrodenetleyiciler ile Seri İletişim" İstanbul: Kodlab Yayınları 2015

[25] The Bluetooth Special Interest Group (SIG), "The history of the Bluetooth SIG" https://www.bluetooth.com/about-us/our-history/ Son Erişim Tarihi: Nisan 2020

[26] Arslan, O. (2009). ZIGBEE ile Bina İçi Güvenlik Otomasyon Sistemi. (Yayınlanmamış Lisans Tezi). İstanbul: İstanbul Teknik Üniversitesi

[27] M. Serkan Öcal, "Android Kontrollü Wifi Keşif Robotu", Yüksek Lisans Tezi 2017

[28] Stefan Aust, R. Venkatesha Prasad, Ignas G.M.M. Niemegeers, "Performance evaluation of Sub 1 GHz wireless sensor networks for the smart grid" 37th Annual IEEE Conference on Local Computer Networks, IEEE, February 2013

[29] Ali Batuhan KINDAN, Arm Cortex M0 Serisi Mikrodenetleyicilerden Oluşan Ayrık Gömülü Sistemler için Can Bus Tabanlı Yazılım Güncelleme Sistemi Tasarımı, 2019

[30] Robert Oshana and Mark Kraeling (Eds.) - Software Engineering for Embedded Systems. Methods, Practical Techniques, and Applications-Newnes, 2013

[31] Muharrem Tuncay Gençoğlu, Importance of Cryptography in Information Security IOSR Journal of Computer Engineering (IOSR-JCE) Volume 21, Issue 1, Ser. II (Jan- Feb 2019), PP 65-68

[32] Christopher Swenson "Modern Cryptanalysis Techniques For Advanced Code Breaking" Chapter 5, General Cryptanalytic Methods

[33] Ümit Günden, Şifreleme Algoritmalarının Performans Analizi, Yüksek Lisans Tezi 2010

[34] Kenneth W. Dam and Herbert S. Lin Cryptography's Role In Securıng The Informatıon Socıety, Part I, Say. 51-60

[35] Sedat Görmüş, Hakan Aydın, "Güzin Ulutaş Nesnelerin İnterneti Teknolojisi İçin Güvenlik: Var Olan Mekanizmalar, Protokoller ve Yaşanılan Zorlukların Araştırılması" Journal of the Faculty of Engineering and Architecture of Gazi University

[36] P. Gope, T. Hwang, "BSN-Care: A secure IoT-based modern healthcare system using body sensor network", IEEE Sensors J., vol. 16, no. 5, pp. 1368-1376, Mar. 2016.

[37] S. R. Geerlings, L.D. Vredevoogd, D. A. Bleker, J. D. Spencer, (Mayıs 2012) "System and Method for Training a transmitter to Control a Remote Control System" Patent no: US 8,174,357 B2

[38] "A Brief History of Cryptography", Cypher Research Laboratories, 24 Şubat 2006, Son Erişim Nisan 2021.

[39] Steven M. Bellovin, "Vernam, Mauborgne, and Friedman: The One-Time Pad and the Index of Coincidence" 2014.

[40] M.Ü. Çeşmeci, Elektronik Çağ Öncesi Dönem Kriptoloji Tarihi, Tübitak UEKAE Dergisi, Sayı 01, Eylül/Aralık 2009.

[41] U.K. Boyacı, Günümüzde Kriptoloji, Tübitak UEKAE Dergisi, Sayı 01, Eylül/Aralık 2009.

[42] "The Case for Elliptic Curve Cryptography".NSA. Arşiv, Son erişim Nisan 2021

[43] Bayar, E., "Modern Kriptosistemlerle Şifrelemenin Modellenmesi ile Veri Güvenliğinin Sağlanması", Yüksek Lisans Tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, 1-32 (2012)

[44] Bodur, H. (2016). Java Diliyle Kriptoloji Uygulamaları. İstanbul: ABAKÜS

[45] Tomoyasu Suzaki, Kazuhiko Minematsu, Sumio Morioka, and Eita Kobayashi. Twine: A lightweight, versatile block cipher. In Proceedings of ECRYPT Workshop on Lightweight Cryptography, 2011. http://www.uclouvain.be/.

[46] Wetzels, Jos, and Wouter Bokslag. "Simple SIMON: FPGA implementations of the SIMON 64/128 Block Cipher." arXiv preprint arXiv:1507.06368 (2015).

[47] Beaulieu, Ray, et al. "SIMON and SPECK: Block Ciphers for the Internet of Things." IACR Cryptol. ePrint Arch. 2015 (2015): 585.

[48] Sleem, Lama, and Raphaël Couturier. "Speck-R: An ultra light-weight cryptographic scheme for Internet of Things." *Multimedia Tools and Applications* 80.11 (2021): 17067-17102.

[49] Microchip (2021), PIC 16F1454, 9 Aralık 2021 Tarihinde http://ww1. microchip.com/downloads/en/devicedoc/40001639b.pdf

[50] Nuvoton 'N76E003' (2021), DS_N76E003_EN_ Rev1.09, 9 Aralık 2021 Tarihinde:https://www.nuvoton.com/export/resourcefiles/DS_N76E003_EN_Rev1.09.pdf

[51] UNI-T Model UT71A/B/C/D/E OPARATING MANUAL, Veri sayfası, Son Erişim Tarihi: 9 Ocak 2022, https://www.uni-trend.com/ uploadfile/cloud/ English%20manual/General%20Meters/UT71%20English%20Manual.pdf

[52] Ikalogic SQ Series User Manuel, Son Erişim Tarihi: 9 Ocak 2022, https://cdn.ikalogic.com/docs/ds/sq-series-manual-EN.pdf

[53] Maxwell CR2032, Veri sayfası, Son Erişim Tarihi: 9 Ocak 2022, https://cdn-shop.adafruit.com/datasheets/maxell_cr2032_datasheet.pdf

TEZDEN ÜRETİLMİŞ YAYINLAR

Konferans Bildirileri

1. Umut Engin Ayten,Rıfkı Yaralı, "Farklı Mimarili Mikrodenetleyicilerde Aes-128 Şifreleme Algoritmasının Yazılım Tabanlı Analizi" International Marmara Sciences Congress (Imascon Autumn) 2021 Proceedings Book (Sayfa: 338)