



**T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



DOKTORA TEZİ

**ENERJİ VERİMLİ KABLOSUZ ALGILAYICI AĞ
TASARIMI VE GERÇEKLEŞTİRİLMESİ**

Mehmet Erkan YÜKSEL

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Danışman

Prof. Dr. Abdül Halim ZAIM

Temmuz, 2014

İSTANBUL



**T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



DOKTORA TEZİ

**ENERJİ VERİMLİ KABLOSUZ ALGILAYICI AĞ
TASARIMI VE GERÇEKLEŞTİRİLMESİ**

Mehmet Erkan YÜKSEL

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Mühendisliği Programı

Danışman

Prof. Dr. Abdül Halim ZAIM

Temmuz, 2014

İSTANBUL

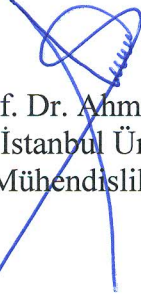
Halim Zaim

Bu çalışma, 23/07/2014 tarihinde aşağıdaki jüri tarafından Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği programında Doktora Tezi olarak kabul edilmiştir.

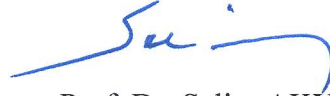
Tez Jürisi:



Prof. Dr. Abdül Halim ZAIM (Danışman)
İstanbul Ticaret Üniversitesi
Mühendislik ve Tasarım Fakültesi



Prof. Dr. Ahmet SERTBAŞ
İstanbul Üniversitesi
Mühendislik Fakültesi



Prof. Dr. Selim AKYOKUŞ
Doğuş Üniversitesi
Mühendislik Fakültesi



Prof. Dr. Rıfat YAZICI
İstanbul Ticaret Üniversitesi
Mühendislik ve Tasarım Fakültesi



Doç. Dr. Hacı Ali MANTAR
Gebze Yüksek Teknoloji Enstitüsü
Mühendislik Fakültesi

ÖNSÖZ

Her şeyden önce; benim bu günlere kadar gelmemi sağlayan, bana maddi ve manevi her konuda destek olup yol gösteren canım annem Günser YÜKSEL’e, canım babam Bekir YÜKSEL’e, çalışmalarımda fikir ve görüşlerinden yararlandığım kendisi de halen bir doktora öğrencisi olan canım kardeşim Asım Sinan YÜKSEL’e sonsuz minnet ve sevgilerimi sunuyorum. Onlarla, kendimi dünyanın en şanslı insanı sayıyorum.

Beni danışmanlığına kabul eden, hoca-öğrenci arasında olabilecek tüm mesafeleri kaldırıp bir dost, bir arkadaş gibi yaklaşan, bana hep sahip çıkan, lisansüstü öğrenimim süresince her konuda değerli zamanlarını ayırarak bilgi ve deneyimlerini esirgemedi paylaşılan, bana her türlü imkanı sağlayan, rehberlik ve yönlendirmeleri ile düşünce ufkumu geliştiren çok değerli danışman hocam Sayın Prof. Dr. Abdül Halim ZAIM’e sonsuz şükranlarımı sunuyorum. Böyle bir danışman hocam olmasından dolayı kendimi ikinci kez dünyanın en şanslı insanı olarak görüyorum.

Tez çalışmalarımın yürütülmesi sürecinde; her türlü ilgi ve desteği gösteren, moral veren, değerli fikirleri, yorumları, önerileri ve katkıları ile çalışmalarımı zenginleştiren tez izleme jüri üyesi hocalarım; Sayın Prof. Dr. Selim AKYOKUŞ’a ve Sayın Doç. Dr. Hacı Ali MANTAR’a en içten teşekkürlerimi arz ediyorum.

Lisansüstü öğrenimim boyunca yardımlarını hiç esirgemeyen İstanbul Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Başkanı Sayın Prof. Dr. Ahmet SERTBAŞ’a, öğretim üyeleri Sayın Yard. Doç. Dr. Muhammed Ali AYDIN’a ve Sayın Yard. Doç. Dr. Selçuk SEVGİN’e desteklerinden dolayı çok teşekkür ederim.

Temmuz, 2014

Mehmet Erkan YÜKSEL

İÇİNDEKİLER

Sayfa No

ÖNSÖZ.....	i
İÇİNDEKİLER	ii
ŞEKİL LİSTESİ.....	v
TABLO LİSTESİ	viii
PROGRAM KODU LİSTESİ.....	x
SİMGE VE KISALTMA LİSTESİ	xi
ÖZET.....	xiii
SUMMARY	xv
1. GİRİŞ.....	1
2. GENEL KISIMLAR	6
2.1. KAA MİMARİSİ VE PROTOKOL YİĞİNİ	7
2.1.1. Fiziksel Katman.....	11
2.1.2. Veri Bağlantı Katmanı.....	11
2.1.2.1. Ortam Erişim Kontrolü (Medium Access Control-MAC)	11
2.1.2.2. Hata Kontrolü	12
2.1.3. Ağ Katmanı	12
2.1.4. Taşıma Katmanı.....	13
2.1.5. Uygulama Katmanı.....	14
2.1.5.1. Kaynak Kodlama (Veri Sıkıştırma)	14
2.1.5.2. Sorgu İşleme.....	15
2.1.5.3. Ağ Yönetimi	18
2.2. STANDARDİZASYON	22
2.2.1. ZigBee	24
2.2.2. WirelessHART	27
2.2.3. 6LoWPAN.....	29
2.3. KAA’LARDA KULLANILAN YAZILIM PLATFORMLARI.....	31
2.4. KAA TASARIMINI ETKİLEYEN FAKTÖRLER	33
2.4.1. Donanım Kısıtlamaları	33

2.4.2. Hata Toleransı.....	36
2.4.3. Ölçeklenebilirlik	38
2.4.4. KAA Topolojisi	38
2.4.4.1. Konuşlandırma Öncesi ve Konuşlandırma Aşamaları.....	38
2.4.4.2. Konuşlandırma Sonrası Aşama.....	38
2.4.4.3. Ek Düğümlerin Yeniden Konuşlandırılması Aşaması	39
2.4.5. İletim Ortamı	39
2.4.6. Güç Tüketimi	41
2.4.6.1. Algılama	43
2.4.6.2. Veri İşleme	44
2.4.6.3. İletişim.....	46
2.4.7. Üretim Maliyetleri	53
2.5. ALGILAYICI DÜĞÜM PLATFORMLARI	54
2.5.1. Alt Düzey Platformlar	55
2.5.2. Üst Düzey Platformlar	57
3. MALZEME VE YÖNTEM	60
3.1. GÜÇ-BİLİNÇLİ KABLOSUZ ALGILAYICI DÜĞÜM TASARIMI.....	61
3.1.1. Kablosuz Algılayıcı Düğüm Devresinin Gerçekleştirilmesi	70
3.1.2. Kablosuz Algılayıcı Düğüm için Geliştirilen Güç-Bilinçli Sistem	73
3.2. KOORDİNATÖR DÜĞÜM İÇİN ENERJİ VERİMLİ SİSTEM TASARIMI	80
3.3. KAA İÇİN HAVA TAHMİNİNE DAYALI, BULANIK MANTIK TABANLI, ENERJİ VERİMLİ SENKRONİZASYON PROTOKOLÜ	86
3.3.1. Senkronizasyon Sebepleri	87
3.3.2. Senkronizasyonda İletişim Gecikmesinin Deterministik Olmaması.....	88
3.3.3. Senkronizasyon Protokolü Tasarımı.....	90
4. BULGULAR	104
4.1. KABLOSUZ ALGILAYICI DÜĞÜM İÇİN GÜÇ BİLİNÇLİ METODOLOJİ.....	107
4.1.1. Sistem Modelleri	108
4.1.1.1. Güç Bilinçli Kablosuz Algılayıcı Düğüm Modeli.....	110
4.1.1.2. Olay Üretim Modeli	112
4.1.2. Algılayıcı Düğümde Uyku Durumu Geçiş İlkesi	114
4.1.3. Kayıp Olaylar	118
4.1.4. Sonuçlar	119
4.2. KOORDİNATÖR DÜĞÜMDE GÜÇ TASARRUFU	122
5. TARTIŞMA VE SONUÇ.....	129

KAYNAKLAR	134
EKLER.....	148
ÖZGEÇMİŞ.....	152

ŞEKİL LİSTESİ

Sayfa No

Şekil 1.1: Bir kablosuz algılayıcı ağ mimarisi.....	1
Şekil 1.2: KAA'da veri iletim yöntemleri: a) Tek sekmeli iletim, b) Çok sekmeli iletim.....	4
Şekil 2.1: Belirli bir bölgeye dağıtılmış kablosuz algılayıcı düğümler.	8
Şekil 2.2: KAA protokol yığını.	9
Şekil 2.3: Sorgu işleme mimarisi.....	17
Şekil 2.4: Merkezi ağ mimarisi.....	21
Şekil 2.5: Dağıtık ağ mimarisi.....	22
Şekil 2.6: Hiyerarşik ağ mimarisi.	22
Şekil 2.7: IEEE 802.15.4 protokol mimarisi.	23
Şekil 2.8: IEEE 802.15.4 frekans bandları.	23
Şekil 2.9: Farklı uygulamalar için geliştirilmiş kablosuz ağ standartları.	24
Şekil 2.10: ZigBee protokol yığını.	25
Şekil 2.11: WirelessHART ağ mimarisi ve bileşenleri.....	28
Şekil 2.12: KAA'ların İnternet (IP ağı) ile entegrasyonu.	29
Şekil 2.13: IEEE 802.15.4'ün IPv6 ile entegrasyonunda 6LoWPAN standardı.....	30
Şekil 2.14: MicaZ algılayıcı düğümün güç tüketimi.	42
Şekil 2.15: Bir algılayıcı sistemi blok diyagramı.	43
Şekil 2.16: Bir RF alıcı-verici devre sisteminin blok diyagramı.	47
Şekil 2.17: Bir kablosuz algılayıcı düğümde bulunan RF alıcı-verici devre modeli.	49
Şekil 2.18: Sadeleştirilmiş enerji modeli blok diyagramı.....	51
Şekil 2.19: Mica algılayıcı düğüm ailesi.	55
Şekil 2.20: Telos/Tmote algılayıcı düğüm ailesi.	56
Şekil 2.21: EYES alt düzey algılayıcı düğüm prototipi.....	56

Şekil 2.22: Stargate tek kartlı bilgisayar.....	58
Şekil 2.23: Imote2 tek kartlı bilgisayar.	59
Şekil 3.1: Kablosuz algılayıcı düğüm mimarisi.....	61
Şekil 3.2: ATmega328P mikrodenetleyici ve Arduino Uno platformu.	64
Şekil 3.3: XBee 868LP RF iletişim modülü.	66
Şekil 3.4: XBee 8686LP RF uygulama geliştirme seti.	67
Şekil 3.5: XBee 8686LP RF uygulama geliştirme seti temel bileşenleri.....	67
Şekil 3.6: Veri toplama ve aktivasyon işlemleri.	68
Şekil 3.7: Kablosuz algılayıcı düğümümüzün blok diyagramı.....	71
Şekil 3.8: Kablosuz algılayıcı düğümümüzün devre şeması.	71
Şekil 3.9: Kablosuz algılayıcı düğümümüzün üst katman baskı devresi.....	72
Şekil 3.10: Kablosuz algılayıcı düğümümüzün alt katman baskı devresi.	72
Şekil 3.11: Kablosuz algılayıcı düğümümüzde DGY/DVÖ alt sistemi blok diyagramı.	74
Şekil 3.12: Algılama, veri işleme ve depolama işlemleri için DGY akış diyagramı.	77
Şekil 3.13: Veri iletişimi için DGY akış diyagramı.	79
Şekil 3.14: Koordinatör: “Raspberry Pi” tek kartlı bilgisayar.	80
Şekil 3.15: “Raspberry Pi” bağlantı arayüzleri.....	81
Şekil 3.16: Koordinatör düğümde DGY alt sistemi blok diyagramı.	82
Şekil 3.17: I ² C yazma protokolü.....	83
Şekil 3.18: I ² C okuma protokolü.	84
Şekil 3.19: Koordinatördeki iletişim modüllerine yönelik DGY/DVÖ akış diyagramı.	85
Şekil 3.20: Senkronizasyon mesajında yaşanan uçtan uca gecikme.....	89
Şekil 3.21: Matlab bulanık çıkarım sistemi.	91
Şekil 3.22: Senkronizasyon protokolü için giriş ve çıkış değişkenleri.	92
Şekil 3.23: Hava durumu değişkeni üyelik fonksiyonları.	94
Şekil 3.24: Pil seviyesi değişkeni üyelik fonksiyonları.	97
Şekil 3.25: Senkronizasyon zamanı üyelik fonksiyonları.....	98
Şekil 3.26: Senkronizasyon zamanına ait kural izleyici.	100

Şekil 3.27: Senkronizasyon zamanına ait yüzey izleyici.	100
Şekil 3.28: Senkronizasyon işlemi için yayın (broadcast ve unicast) akış diyagramı.	103
Şekil 4.1: Kablosuz algılayıcı düğümlerimiz.	105
Şekil 4.2: Tasarladığımız KAA topolojisi.	106
Şekil 4.3: KAA ve kablosuz algılayıcı düğüm mimarisi.	109
Şekil 4.4: Durum geçişleri, gecikme süresi ve harcanan güç.	115
Şekil 4.5: Olasılıksal uyku durumu geçişi.	119
Şekil 4.6: Kablosuz algılayıcı düğümlerimizin enerji tüketimlerini analiz eden simülasyon....	120
Şekil 4.7: Algılayıcı düğümlerimizin olaylara bağlı enerji tüketimleri ($\eta=100$ düğüm).	121
Şekil 4.8: Algılayıcı düğümlerimizin olaylara bağlı enerji tüketimleri ($\eta=250$ düğüm).	121
Şekil 4.9: Algılayıcı düğümlerimizin olaylara bağlı enerji tüketimleri ($\eta=500$ düğüm).	121
Şekil 4.10: Koordinatörün güç tüketimi.	122
Şekil 4.11: İnternete bağlı koordinatörün güç tüketimi.	123
Şekil 4.12: WLAN içerisinde çalışan koordinatörün güç tüketimi.....	123
Şekil 4.13: KAA içerisinde çalışan koordinatörün güç tüketimi.	124
Şekil 4.14: Koordinatörde DGY ve optimizasyon işlemleri sonrası güç tüketimi.	125
Şekil 4.15: İnternete bağlı enerji tasarruflu koordinatörün güç tüketimi.....	125
Şekil 4.16: WLAN içerisinde çalışan enerji tasarruflu koordinatörün güç tüketimi.	126
Şekil 4.17: KAA'daki koordinatörde DGY ve optimizasyon işlemleri sonrası güç tüketimi. ..	127

TABLO LİSTESİ

Sayfa No

Tablo 2.1: Çeşitli kablosuz algılayıcı düğümler ve temel özellikleri.....	54
Tablo 3.1: ATmega328P mikrodenetleyici özellikleri.....	63
Tablo 3.2: XBee 868LP RF iletişim modülü özellikleri ve performansı.	66
Tablo 3.3: TPS62740 özellikleri.	75
Tablo 3.4: Kablosuz algılayıcı düğümümüzde DVÖ konfigürasyonu.	75
Tablo 3.5: “Raspberry Pi” donanım özellikleri.....	81
Tablo 3.6: ISL9112 yazmaç adres içeriği.	83
Tablo 3.7: Koordinatördeki iletişim alt birimleri için DVÖ konfigürasyonu.	83
Tablo 3.8: İstanbul ili için ortalama güneşlenme süreleri.	93
Tablo 3.9: İstanbul ili için Güneş doğuş-batış saatleri arasındaki zaman farkı.....	94
Tablo 3.10: Hava durumları ve ağırlık değerleri.....	95
Tablo 3.11: Çeşitli veri normalizasyon yöntemleri.....	96
Tablo 3.12: Algılayıcı düğümümüzün pil gerilim seviyeleri ve ağırlık değerleri.....	97
Tablo 3.13: Senkronizasyon protokolümüz için kural matrisi.	99
Tablo 4.1: Kablosuz algılayıcı düğümümüz için çalışma ve uyku durumları.....	111
Tablo 4.2: Uyku durumuna ait güç, gecikme süresi ve eşik seviyesi.....	116
Tablo 4.3: Kablosuz algılayıcı düğümümüz için kullanışlı uyku durumları.....	117
Tablo 4.4: Koordinatörün farklı çalışma durumlarına ait güç tüketimleri.	122
Tablo 4.5: İnternete bağlı koordinatörün farklı çalışma durumlarındaki güç tüketimleri.....	123
Tablo 4.6: WLAN’daki koordinatörün farklı çalışma durumları için güç tüketimleri.	124
Tablo 4.7: KAA içerisindeki koordinatörün farklı çalışma durumlarına ait güç tüketimleri....	124
Tablo 4.8: DGY ve optimizasyon işlemleri sonrasında koordinatörün farklı çalışma durumlarına yönelik güç tüketimleri ve enerji tasarrufları.	125

Tablo 4.9: DGY ve optimizasyon işlemleri sonrası İnternete bağlı koordinatörün farklı çalışma durumları için güç tüketimleri ve enerji tasarrufları.....	126
Tablo 4.10: WLAN içerisindeki koordinatörün DGY ve optimizasyon işlemleri sonrası farklı çalışma durumlarına ait güç tüketimleri ve enerji tasarrufları.	126
Tablo 4.11: KAA'daki koordinatörün DGY ve optimizasyon işlemleri sonrası farklı çalışma durumlarına ait güç tüketimleri ve enerji tasarrufları.	127
Tablo 5.1: Geleneksel ağlar ile KAA'ların karşılaştırılması.	132

PROGRAM KODU LİSTESİ

Sayfa No

Program kodu 3.1: Dinamik voltaj ölçeklendirme fonksiyonu.....	76
Program kodu 3.2: Algılama, veri işleme ve depolama işlemleri için DGY fonksiyonu.	76
Program kodu 3.3: Veri iletişimine ait DGY fonksiyonu.....	78
Program kodu 3.4: Koordinatörde DVÖ işlemi yapan fonksiyon.	84
Program kodu 3.5: Koordinatördeki farklı iletişim modülleri için DGY fonksiyonu.	85
Program kodu 3.6: Senkronizasyon protokolümüzde tanımlı bulanık mantık modülü.	101
Program kodu 3.7: Senkronizasyon protokolümüzde tanımlı broadcast modülü.	102
Program kodu 4.1: Uyku durumu geçiş algoritması.	117

SİMGE VE KISALTMA LİSTESİ

Simgeler

Açıklama

α	: ADC’de harcanan gücün örnekleme oranı ve etkin bit sayısı ile ilişkisi
η	: Güç amplifikatörünün güç verimliliği
λ	: Taşıyıcı frekansı dalga boyu
γ	: d mesafesinde istenilen sinyal/gürültü oranı
σ^2	: Güç spektrum enerjisi
σ	: Standart sapma
μ	: Popülasyon ortalaması
f	: Saat frekansı
ω	: Ağırlık değeri

Kısaltmalar

Açıklama

ACK	: Acknowledgement
ACPI	: Advanced Configuration and Power Interface
ADC	: Analog-to-Digital Converter
API	: Application Programming Interface
ARQ	: Automatic Repeat Request
ASIC	: Application Specific Integrated Circuit
BER	: Bit Error Rate
BPSK	: Binary Phase Shift Keying
CDMA	: Code Division Multiple Acces
CMOS	: Complementary Metal-Oxide-Semiconductor
DPM	: Dynamic Power Management
DSP	: Digital Signal Processor
DSSS	: Direct Sequence Spread Spectrum
DVS	: Dynamic Voltage Scaling
FEC	: Forward Error Correction
FIS	: Fuzzy Inference System
FPGA	: Field Programmable Gate Array
GPS	: Global Positioning System
GPIO	: General Purpose Input/Output
GUI	: Graphical User Interface
HART	: Highway Addressable Remote Transducer
I ² C	: Inter-Integrated Circuit
IDE	: Integrated Development Environment
IEEE	: Institute of Electrical and Electronic Engineers
IETF	: Internet Engineering Task Force
IFA	: Intermediate-Frequency Amplifier
IP	: Internet Protocol
IR	: Infrared

ISM	: Industrial, Scientific and Medical
ITU	: International Telecommunication Union
Java ME	: Java Micro Edition
LNA	: Low Noise Amplifier
LOS	: Line-of-Sight
LZW	: Lempel–Ziv–Welch
MAC	: Medium Access Control
MANET	: Mobile Ad Hoc Network
MEMS	: MicroElectroMechanical Systems
MIMO	: Multiple-Input Multiple-Output
MISO	: Master In Slave Out
MQAM	: M-ary Quadrature Amplitude Modulation
MOSI	: Master Out Slave In
MSB	: Most Significant Bit
NRZ	: Non Return to Zero
NTP	: Network Time Protocol
PA	: Power Amplifier
PAN	: Personal Area Network
PCMCIA	: Personal Computer Memory Card International Association
PDA	: Personal Digital Assistant
PLL	: Phase-Locked Loop
QoS	: Quality of Service
QPSK	: Quadrature Phase Shift Keying
RISC	: Reduced Instruction Set Computing
RF	: Radio Frequency
SCL	: Serial Clock Line
SDA	: Serial Data Line
SDIO	: Secure Digital Input Output
SDR	: Software Defined Radio
SNR	: Signal-to-Noise Ratio
SPI	: Serial Peripheral Interface
TCP	: Transmission Control Protocol
TinyOS	: Tiny Operating System
TDMA	: Time Division Multiple Access
UDP	: User Datagram Protocol
UHF	: Ultra High Frequency
USB	: Universal Serial Bus
UTC	: Coordinated Universal Time
VCO	: Voltage-Controlled Oscillator
WLAN	: Wireless Local Area Network
WSN	: Wireless Sensor Network
WSAN	: Wireless Sensor and Actor Network
6LoWPAN	: IPv6 over Low power WPAN

ÖZET

DOKTORA TEZİ

ENERJİ VERİMLİ KABLOSUZ ALGILAYICI AĞ TASARIMI VE GERÇEKLEŞTİRİLMESİ

Mehmet Erkan YÜKSEL

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalı

Danışman : Prof. Dr. Abdül Halim ZAIM

Kablosuz algılayıcı ağlar (KAA), çok sayıda algılayıcı devrenin belirli bir coğrafi bölgeye rastgele dağıtılmasıyla ya da sabit bir şekilde konuşlandırılmasıyla oluşan tasarsız (ad-hoc) ağlardır. Algılayıcı devreler; çevresindeki fiziksel olayları algılayan, algıladığı verileri işleyen, çözümleyen, depolayan, kendi aralarında haberleşerek ve işbirliği yaparak çalışan, düşük güç tüketimli, küçük boyutlu, çok işlevli, kablosuz otonom cihazlardır. KAA'lar, bulundukları ortamla sürekli etkileşim halindedirler. Gerçek zamanlı bilgilere anında ve her yerden kolayca erişilmesine imkan sağlayan özelliktedirler. Konuşlandırıldıkları bölgede herhangi bir altyapıya ihtiyaç duymamaları, organize çalışabilmeleri, herhangi bir bakım, onarım veya düzenleme gerektirmemeleri ve çok çeşitli uygulama alanlarına sahip olmaları nedeniyle, KAA'ların kullanım alanları gün geçtikçe daha da yaygınlaşmaktadır.

Bir KAA'nın yapısı (topolojisi); algılayıcı devrelerin uygulama alanlarına, donanım ve yazılım özelliklerine, konuşlandırıldıkları bölgedeki konumlarına (hareketli/hareketsiz), iletişim mesafelerine, kablosuz bağlantı karakteristiklerine, güç yönetim aktivitelerine, işbirlikçi çalışma yapılarına (kümeleme, yönlendirme), veri aktarım yöntemlerine (tek ya da çok sekmeli iletim) ve çevresel şartlara bağlı olmak üzere dinamik olarak değişir. Dolayısıyla, böyle bir dinamik KAA ortamında, algılayıcı devrelerin enerji tüketimini minimize ederken ağın bağlantısını sürekli sağlamak büyük önem taşır.

Çalışmamız, KAA'ların yapısal özelliklerinde bulunan sınırlamaları ve uygulama alanlarında ortaya çıkan zorlukları, problemleri dikkate alarak; belirli bir bölgeye konuşlandırılmış KAA'da kullanılmak üzere ideal algılama, hatasız veri toplama, işleme ve çözümleme işlemlerini yapan, organize çalışan, yenilenebilir enerji kaynağı güneşten enerji sağlayan, kendi güç yönetim modlarını dinamik bir şekilde düzenleyerek enerjilerini mümkün olduğunca az harcayan, çevresel koşullara göre senkronizasyon yapan, küçük boyutlu, uzun ömürlü, düşük güçlü kablosuz algılayıcı düğümlerin gerçekleştirilmesini; bu algılayıcı düğümler ve mevcut koordinatörler (ağ geçidi, veri toplama merkezi) kullanılarak enerji verimli bir KAA'nın oluşturulmasını; KAA'daki koordinatörler için güç tasarrufu sağlayacak donanım sistemlerinin tasarlanmasını; gerçek zamanlı iletilmesi gereken bilgileri etkin servis kalitesi sunarak KAA içerisindeki veri toplama merkezlerine ve İnternetteki sunucu cihazlara aktaracak ara katman yazılımlarının geliştirilmesini amaçlamaktadır.

Çalışmamızda; hem veri işleyici hem de yönlendirici olarak görev yapabilen, üzerinde güneş paneli bulunan, işbirlikçi çalışmayı destekleyen, dinamik güç yönetim özelliğine sahip, enerji verimli, uzun menzilli, küçük boyutlu, kablosuz algılayıcı düğümler gerçekleştirdik. Algılayıcı düğümlerimizi ve literatür taraması sonucunda seçtiğimiz ağ geçitlerini/koordinatörleri (*Raspberry Pi* marka tek kartlı bilgisayarlar) kullanarak mesh (örgü) ve point-to-multipoint (noktadan-çoklu noktaya) topolojilerini içeren bir KAA oluşturduk. Ağ geçitlerine güneş panelleri entegre ederek ve bunlar için güç yönetim devreleri geliştirerek bu cihazların KAA'da enerji tasarruflu çalışmalarını sağladık. Daha sonra, tasarladığımız KAA için hava koşullarına dayalı, bulanık mantık tabanlı bir senkronizasyon protokolü ve algılayıcı düğümler tarafından toplanan verileri merkezi birimlere gerçek zamanlı aktaran ara katman yazılımları geliştirdik. Ayrıca, algılayıcı düğümlerde ve ağ geçitlerinde güç optimizasyonu, bu cihazlarda çalışan yazılımlarda kod optimizasyonu yaptık. Sonuç olarak; çalışmamızda gerçekleştirdiğimiz kablosuz algılayıcı düğümlerimiz, bu algılayıcı düğümlerimiz için geliştirdiğimiz güç bilinçli metodoloji, donanım yapılarını iyileştirdiğimiz güç tasarruflu ağ geçitleri, kullandığımız senkronizasyon protokolü sayesinde, tasarladığımız KAA'nın enerji verimli ve optimum çalışmasını sağladık.

Temmuz 2014, 172 sayfa.

Anahtar kelimeler: Kablosuz algılayıcı ağlar, güç bilinçli algılayıcı düğüm, dinamik güç yönetimi, zaman senkronizasyonu, enerji tasarrufu.

SUMMARY

Ph.D. THESIS

DESIGN AND IMPLEMENTATION OF ENERGY EFFICIENT WIRELESS SENSOR NETWORK

Mehmet Erkan YÜKSEL

İstanbul University

Graduate School of Science and Engineering

Department of Computer Engineering

Supervisor : Prof. Dr. Abdül Halim ZAIM

Wireless Sensor Networks (WSN) are ad-hoc networks consisting of many sensor nodes that have either fixed position or randomly distributed across a specific geographic area. Sensor nodes are small-sized autonomous devices that have low power consumption; sense physical events around them; process, analyze and store sensed data; communicate with each other and work collaboratively. WSNs continuously interact with the environment they are in. They have ability to work in an organized manner, access real time data instantly, anywhere. WSNs can work smoothly without needing any infrastructure, maintenance or repair, therefore making their application area spread to many fields.

WSN topology changes dynamically in accordance with the environmental conditions, data transmission procedures (single-hop or multi-hop communication), communication range, signal processing techniques, collaborative working behaviors (clustering, routing), location where they are placed (mobile, fixed), hardware and software specifications of wireless sensor nodes. Therefore, it is important to keep the network connection alive while minimizing the energy consumption of sensor nodes in such dynamic WSN environment.

Our study aims to develop sensor nodes that are to be used in the WSNs deployed in a specific region and capable of ideal sensing, correct data collection, data processing and analysis operations; operates in an organized manner, uses the sun as a renewable energy source, consumes energy as low as possible by adjusting its power management, synchronizes itself according to the weather conditions, small-sized, durable, low power

by considering the limitations of WSNs' structural characteristics and the difficulties in the application areas; design a energy efficient WSN by utilizing these sensor nodes and existing coordinators (gateways, data collection center); design hardware systems that will provide power saving for the coordinators in the WSN; develop middleware that will transmit data in real time to the data collection centers and servers in Internet by guaranteeing quality-of-service (QoS).

In our study, we implemented small-sized, energy efficient, long-range wireless sensor nodes that can serve as both data processors and gateways, have sun panels on them, support collaborative working, and have dynamic power management capabilities. We created a WSN that combines the mesh and the point-to-multipoint topologies; utilizes our developed sensor nodes and the network gateways/coordinators (Raspberry Pi computers with single card slots) chosen through literature review. We ensured that network gateways were also energy efficient by developing power management circuits and integrating sun panels on them. Subsequently, we developed a synchronization protocol that is energy efficient, weather forecast and fuzzy-logic based; a middleware that collects data from sensor nodes and transmits them to related central units in real time. Furthermore, we performed power optimization in the sensor nodes, the network gateways and code optimization in the middleware that runs on these devices. As a result, we ensured that the designed WSN was energy efficient and worked optimal by utilizing the sensor nodes that we developed in our study, the low power network gateways of which their hardware was optimized, the synchronization protocol and the power-aware methodology that was developed for sensor nodes.

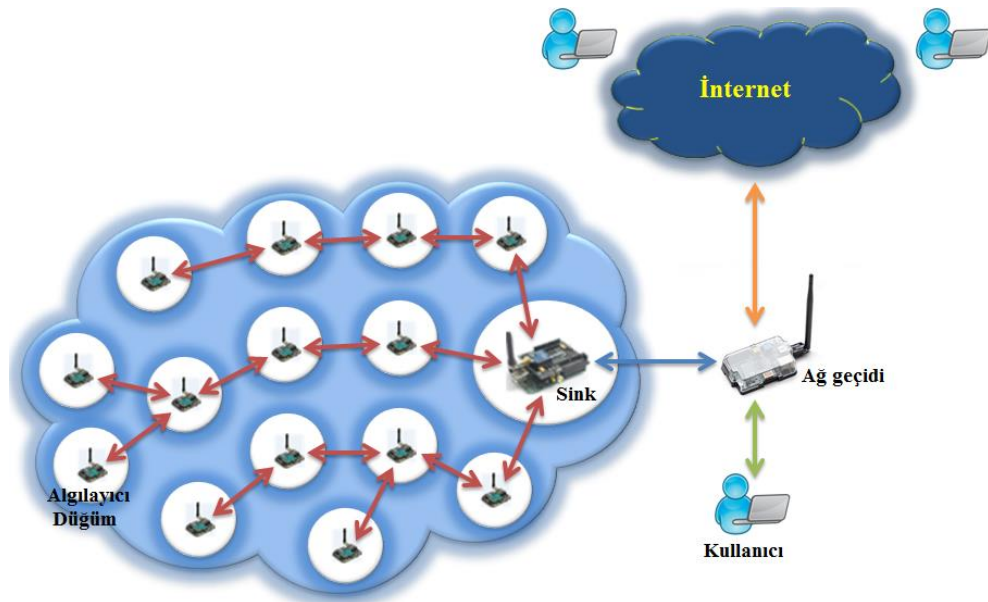
July 2014, 172 pages.

Keywords: Wireless sensor networks, power-aware sensor node, dynamic power management, time synchronization, energy conservation.

1. GİRİŞ

Bilgi ve iletişim teknolojilerinde, entegre devre, donanım, dijital elektronik ve mikro elektro-mekanik sistem (MEMS-MicroElectroMechanical Systems) teknolojilerinde yaşanan sürekli yenilikler; düşük maliyetli, enerji tasarruflu, çok fonksiyonlu, küçük boyutlu kablosuz algılayıcı devrelerin tasarlanmasını, üretilmesini ve geliştirilmesini sağlamıştır [1]. Çevresiyle sürekli etkileşim halinde olma, fiziksel olayları (sıcaklık, nem, basınç, ışık, ses, görüntü, ivme vb.) algılama, veri toplama, işleme, depolama gibi işlemleri gerçekleştirme, gerçek zamanlı bilgileri ilgili merkezi sistemlere iletme, kendi aralarında haberleşme ve işbirlikçi çalışma gibi giderek artan yetenekleri sayesinde bu küçük algılayıcı aygıtlar, KAA'ların oluşturulmasına da imkan sağlamışlardır [1, 2].

KAA'lar, hem veri işleyici hem de ağ aktarıcısı (yönlendirici, koordinatör) olarak görev yapabilen çok sayıda algılayıcı düğümün belirli bir coğrafi bölgeye rastgele dağıtılmasıyla ya da sabit bir şekilde konuşlandırılmasıyla oluşturulan tasarsız (ad-hoc) ağlardır (Şekil 1.1).



Şekil 1.1: Bir kablosuz algılayıcı ağ mimarisi.

Algılayıcı düğümler, bulundukları ortamdaki fiziksel olayları ya da nesneleri gerçek zamanlı gözlemlemek/izlemek üzere algılama işlemi yapan, algıladığı verileri işleyen,

depolayan, kısa ya da uzun mesafelerde aralarında kablosuz haberleşebilen, birbirleri ile işbirliği yaparak çalışabilen, düşük maliyetli, düşük güç tüketimli, çok işlevli otonom cihazlardır. Her bir algılayıcı düğüm, temel olarak; algılayıcı (sensör), algılanan verileri işleyen bir mikroişlemci, verileri depolamada kullanılan sınırlı kapasitede bir bellek, birbirleri ile haberleşebilmek (kablosuz iletişim kurabilmek) ve gözlemlenen olayları ilgili merkezi sistemlere iletme için kablosuz bağlantı sağlayan bir RF (Radyo Frekansı) alıcı-verici birimlerini içerir [1, 2].

Algılayıcı düğümler sınırlı kapasitede pillerle beslenir. İzlenecek/gözlemlenecek ortama yerleştirildikten sonra güç kaynaklarının tekrar şarj edilebilmesi ya da yenileriyle değiştirilebilmesi zordur [3, 4]. Bu durum, KAA'nın işlevlerini sağlıklı bir şekilde yerine getirebileceği belirli bir ağ ömrüne sahip olduğunu gösterir. Hesaplama işlemlerinin ve iletişim becerilerinin optimum şekilde yapılması, gereksiz enerji harcamalarının önüne geçilmesi ve ağ ömrünün uzun olması için enerji tüketiminin minimumda tutulması gerekir [5, 6]. Dolayısıyla, algılayıcı düğümler, pil ömrüne bağlı kısıtlamalar nedeniyle enerjilerini korumak amacıyla ağ içerisinde ya alıcı-verici modüllerini kapatırlar ya da düşük güç tüketimi ile uyku modunda çalışırlar.

Algılayıcı düğümler, algılanacak/gözlemlenecek bölgelere rastgele dağıtılır ya da sabit bir şekilde konuşlandırılırlar. Ağ içerisinde hareketli ya da hareketsiz pozisyonda olup uzaktan kontrol edilebilirler [7, 8]. Kendi aralarında organize çalışmalarından dolayı birbirleriyle sürekli iletişim halindedirler. Bu düğümler, on-board mikroişlemcileri aracılığıyla, sadece ne izlediklerini iletmeekten ziyade karmaşık işlemleri başarmak için de programlanabilir özelliktedirler [9, 10].

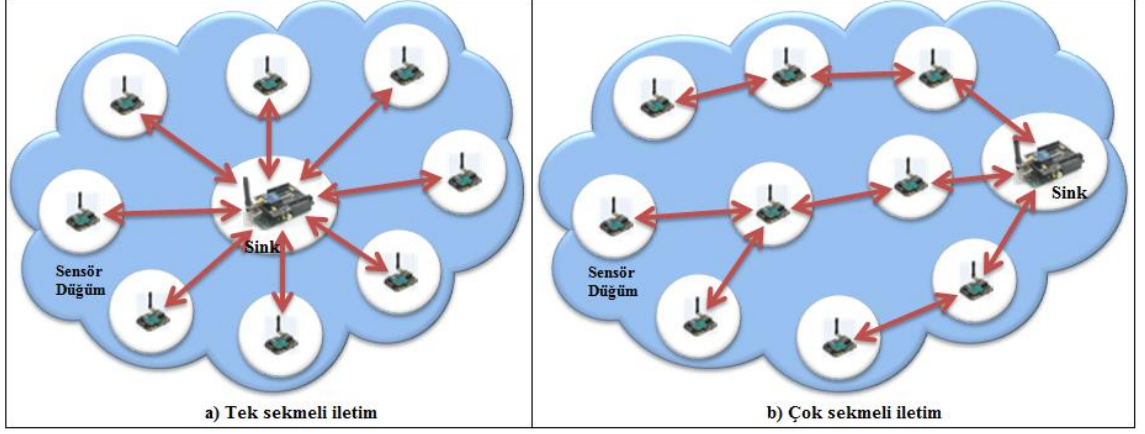
KAA uygulamalarında, bir algılayıcı düğümün arızalanması ya da enerjisinin tükenmesi durumunda güç kaynağının yenisiyle değiştirilememesi (veya tekrar şarj edilememesi), o algılayıcı düğümün ağ içerisinde görevini tam olarak yerine getiremeyeceğini gösterir. Bu tür olumsuz şartlarda, algılayıcı düğüm sorumlu olduğu bölgeden gerekli verileri algılayamayacak ve işleyemeyecektir. Çok sekmeli iletişimin kullanıldığı ağ topolojileri ve işbirliğine dayalı çalışmalar düşünüldüğünde, veri toplama merkezine gönderilmek üzere başka düğümler tarafından kendisine aktarılan bilgilerin hedef birimlere iletilmesi işleminde görev alamayacak ve dolayısıyla kendisiyle organize çalışan diğer düğümlerin de görevlerini tam anlamıyla yapamamasına sebep olacaktır. Nihayetinde,

KAA zamanla işlevlerini sağlıklı bir şekilde yerine getiremeyecektir [11-13]. Algılayıcı düğümlerden gelen bilgilerin merkezi sistemlere etkin bir şekilde aktarılabilmesi için KAA'nın özenle planlanması gerekir. Bununla birlikte; algılayıcı düğümlerin belirli bir coğrafi bölgeye yerleştirildikten sonra sorunsuz çalışabilmeleri, herhangi bir ek bakım veya yeniden düzenleme işlemlerinin yapılmaması, veri toplama, işleme, çözümleme, iletme ve diğer hesaplama işlemleri için muhtemel hatalara (algılayıcı hataları ve çevresel şartlara bağlı olumsuzluklar) karşı sistemi çalışır durumda tutabilmeleri gerekir [1-3]. Bu bağlamda; hata toleranslı haberleşmeyi gerçekleştirecek, bölgesel işbirliğine dayalı çalışmaları destekleyecek, enerji tasarrufu sağlayacak, gelişmiş güvenlik özelliklerine sahip, etkin servis kalitesi sunacak, KAA'ı ve ağ içerisinde kullanılan cihazları dinamik halde tutacak yazılımların ve protokollerin geliştirilmesi de bir o kadar önemlidir.

KAA'larda algılama, veri işleme, iletişim, organize çalışma ve raporlama teknikleri uygulamanın kritik zamanlı olup olmamasına göre belirlenir. Uygulamalar, zaman veya olay bazlı olabilir. Zaman bazlı uygulamalarda, algılayıcı düğümler düzenli (sürekli) veri transferi gerektiren durumlarda periyodik olarak açılır, ortama ait veriler toplanır ve bu veriler ilgili birimlere (Örneğin; koordinatör, ağ geçidi, veri toplama merkezi veya İnternetteki bir sunucu bilgisayar) aktarılır. Olay bazlı uygulamalarda ise sadece belirli bir olay karşısında veriler transfer edilmektedir. Bu tür uygulamalarda algılayıcı düğümler algılama işlemi yaparken daha az enerji tüketecek şekilde ve düşük hassasiyette çalıştırılır. Ortamla ilgili bir değişim algılandığında, algılayıcı düğümün hassasiyeti artırılarak çevreden daha detaylı bilgiler toplanır [14-16].

KAA'larda iletişim tek sekmeli veya çok sekmeli olarak yapılmaktadır (Şekil 1.2). Tek sekmeli iletişimde, bir algılayıcı düğüm bir diğer düğümle ya da koordinatör, ağ geçidi, baz istasyonu gibi cihazlarla doğrudan haberleşerek veri iletimini gerçekleştirir. Çok sekmeli iletişimde ise veriler, ağı oluşturan düğümler üzerinden bir düğümden diğerine aktararak gönderilir. Çok sekmeli iletişim yönteminde; gönderici düğümlerin ve bu düğümlere komşu olan diğer düğümlerin enerjileri, konumları, iletişim mesafeleri gibi parametreler dikkate alınarak ağ içerisinde en uygun yol üzerinden verinin hedefe ulaştırılması amaçlanır. Çok sekmeli iletişim; genellikle kablosuz örgü ağları, KAA'lar,

mobil tasarsız ağlar ve araç ağları gibi verinin kaynaktan hedefe çok sekme üzerinden ulaştırıldığı ağlarda kullanılır [17-19].



Şekil 1.2: KAA'da veri iletim yöntemleri: a) Tek sekmeli iletim, b) Çok sekmeli iletim.

KAA tasarımlarında kullanılan tekniklerin temel amaçları; ağ ömrünü uzatmak, algılayıcı düğümlerin enerji verimli, hatasız ve optimum çalışmasını sağlamak, veri iletiminde ortaya çıkabilecek bağlantı problemlerini önlemektir. Algılayıcı düğümler, algılanacak/gözlemlenecek bölgelere herhangi bir yapısal planlamaya gerek kalmadan (tasarsız şekilde) rastgele yerleştirilirler. Bulundukları ortam hakkında yüksek kalitede bilgiler toplayabilmek, bu bilgileri işleyebilmek ve çok sekmeli iletişim aracılığıyla hedef birimlere aktarabilmek için işbirlikçi/organize çalışırlar. Algılayıcı düğümlerin oluşturduğu ağ, mevcut kablolu/kablosuz haberleşme alt yapısına sahip harici bir ağa (Örneğin; LAN, WLAN) veya İnternete bağlanabilirlik özelliği de taşıyabilir. Böylece, kullanıcılar; ağ üzerinde raporlanan verilere kolayca ve gerçek zamanlı ulaşabilir, verileri analiz ederek izlenecek ortam hakkında tespitlerde bulunabilirler. Bu nedenle, algılayıcı düğümlerin verileri etkin bir şekilde ilgili sistemlere (ağ geçidi, koordinatör, veri toplama merkezi, İnternette bir sunucu, sabit/mobil baz istasyonu) gönderebilmesi için KAA'larda uygun yönlendirme protokolleri (algılayıcı düğümün sahip olduğu veri türü, hesaplama özelliği, konumu, alıcıya/hedef düğüme olan mesafesi, güç seviyesi, ağ yoğunluğu, hizmet kalitesi ve hata toleransı gibi parametreler dikkate alınarak) geliştirilmelidir [20-25]. KAA yapılarında haberleşme hızının oldukça düşük, algılayıcı düğümlerin hesaplama yeteneklerinin ve enerji kapasitelerinin kısıtlı olması nedeniyle bu ağlarda kullanılacak yönlendirme protokollerinin basit (daha az kod, daha düşük hafıza ihtiyacı), verimli ve etkin (sistemi yormayacak performansta) çalışması gerekir.

KAA'lar, oldukça geniş araştırma ve uygulama alanlarına (sağlık, askeri, ulaşım, endüstri, robotik, hassas tarım, meteoroloji, habitat izleme, denizcilik, akıllı evler, otopark sistemleri vb.) sahip olup çok hızlı gelişen bir teknolojidir. Dolayısıyla; herhangi bir sınırlama getirmeksizin konuşlandırma ve kurulum kolaylığı, çok geniş coğrafi alanlarda çevrelerindeki fiziksel olayları algılayabilmeleri, aralarında organize olabilmeleri ve işbirlikçi çalışabilmeleri, bulundukları ortam hakkında gerçek zamanlı ve dağıtık veriler toplayabilmeleri, nesneleri tanımlayabilmeleri ve izleyebilmeleri, hızlı, etkin, esnek, güvenli, enerji tasarruflu, yüksek kapasiteli ve düşük maliyetli dağıtık sistemler olması, KAA'ları günlük hayatımızın tartışılmayacak derecede önemli bir bileşeni yapmaktadır [20-25].

2. GENEL KISIMLAR

Kablosuz iletişim, sayısal elektronik, mikro elektro-mekanik sistem teknolojilerinde yaşanan son gelişmeler ile kısa mesafede bağlanmadan iletişim sağlayan, çok işlevli, küçük boyutlu, düşük maliyetli, az güç harcayan algılayıcı devrelerin tasarımı ve gerçekleştirilmesi, KAA'ların birçok alanda uygulanabilir hale gelmesini sağlamıştır. Bu küçük algılayıcı devrelerin sürekli artmakta olan becerileri arasında algılama, veri işleme, iletişim kurma, kendi aralarında organize çalışarak bir KAA oluşturma gibi işlevler bulunmaktadır [1, 20].

KAA'ların çok geniş uygulama alanları vardır. Bu tarz ağlar, yavaş yavaş hayatımızın ayrılmaz bir parçası haline gelmektedir. Son zamanlarda yapılan araştırmalar; belirli algılama, gözetim ve izleme uygulamalarına özel imal edilen algılayıcı ağların gerçek anlamda uygulanmasını ve yaygınlaşmasını mümkün kılmıştır.

KAA'lara yönelik mevcut ve potansiyel uygulamaları hayata geçirebilmek için çok gelişmiş ve son derece etkin iletişim protokolleri gereklidir. KAA'lar büyük miktarda algılayıcı düğümlerden oluşur. Algılayıcı düğümler, fiziksel bir olay içine veya bunun çok yakınına yoğun bir şekilde konuşlandırılır [23, 26, 27]. Güvenilir ve verimli gözlem yapabilmek, doğru eylemleri uygulamak için; olayın fiziksel özellikleri, algılayıcı düğümlerin sağladığı kolektif bilgiden güvenilir bir şekilde tespit edilebilmelidir. Algılayıcı düğümler, füzyondan sorumlu düğümlere ham veri göndermek yerine, işlem becerileriyle basit hesaplamaları yerel olarak gerçekleştirir, sadece gerekli ve kısmen işlenmiş veriyi iletirler. Dolayısıyla, KAA'ların bu özellikleri, iletişim protokollerinin geliştirilmesi için kendine özgü ciddi zorluklar ortaya çıkarmaktadır [20, 21].

Kablosuz algılayıcı düğümlerin kendilerine has iç-yapı özellikleri, enerji tüketimi bakımından iletişim protokolleri için ek sorunlara neden olur. KAA uygulamaları ve iletişim protokolleri, çoğunlukla yüksek enerji verimliliği sağlamak amacıyla geliştirilir [11-13]. Algılayıcı düğümler sınırlı kapasitede güç kaynaklarına sahiptir. Bu nedenle, geleneksel ağlar; band genişliği (kanal kapasitesi, saniyedeki paket sayısı), verimlilik (paket kaybı, iletim hataları, tıkanıklık), gecikme ve jitter gibi performans ölçütlerini

artırmaya yönelik tasarlanırken, KAA protokolleri öncelikle güç tasarrufuna odaklanır [28-32]. Ayrıca, KAA'ların konuşlandırılması, KAA protokollerinin geliştirilmesinde hesaba katılan bir başka faktördür. Algılayıcı düğüm pozisyonlarının tasarlanmış ya da önceden belirlenmiş olması gerekmez [33-35]. Bu durum, ulaşılamayan arazilerde veya afet yardım operasyonlarında algılayıcı düğümlerin rastgele dağıtımının yapılabilmesini sağlar [36]. Öte yandan, bu rastgele dağıtım, iletişim protokol yığını için kendi kendini düzenleyen protokollerin geliştirilmesini gerektirir. KAA protokollerinde düğümlerin yerleştirilmesine ek olarak, ağ yoğunluğundan da yararlanır. Kısa iletim menzilleri nedeniyle, çok fazla sayıda algılayıcı düğüm yoğun bir şekilde dağıtılabilir ve düğümler birbirine çok yakın olabilir [37-39]. Bu yüzden, düğümler arası iletişimde geleneksel tek sekmeli iletişime göre daha az güç tüketimi sağlayan çok sekmeli iletişimden yararlanır. Ayrıca, algılanan olayın fiziksel özellikleri ile bir araya gelen yoğun dağıtım, mekânsal ve zamansal etki alanları arasındaki korelasyonu ortaya çıkarır. Sonuç olarak, uzaysal-zamansal korelasyon tabanlı protokoller, KAA'ları kurarken gelişmiş verimlilik sağlamak için ortaya çıkmıştır [20, 40, 41].

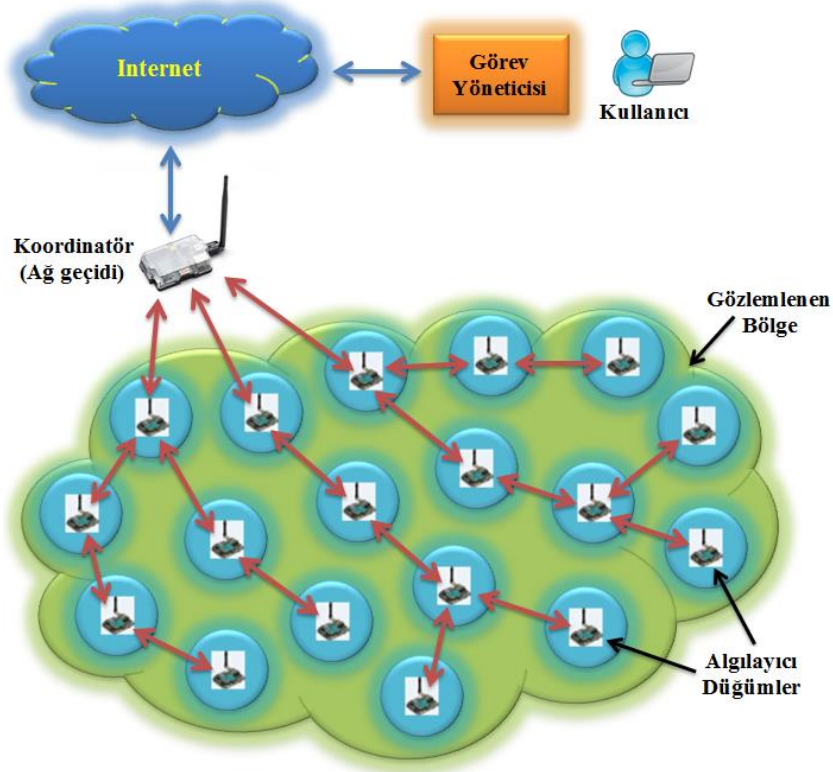
2.1. KAA MİMARİSİ VE PROTOKOL YIĞINI

Bir KAA'daki algılayıcı düğümler, Şekil 2.1'de gösterildiği gibi genellikle algılama yapılacak alana rastgele dağıtılırlar. Bu dağıtılmış algılayıcı düğümlerin her biri; veri toplama, verileri ağ geçidine, İnternette bir sunucuya (veri toplama merkezi) ve son kullanıcıya yönlendirme becerisine sahiptir. Veriler, çok sekmeli altyapısız mimari yoluyla ağ geçidi üzerinden son kullanıcıya yönlendirilir. Ağ geçidi; İnternet, uydu veya herhangi bir kablosuz ağ türü (WiFi, örgü ağları, hücreli sistemler, WiMAX vb.) aracılığıyla ya da bu ağlardan herhangi biri olmaksızın doğrudan görev yöneticisi/son kullanıcı ile iletişim kurabilir [20, 42, 43]. Şekil 2.1'de sunulan mimaride birden fazla ağ geçidinin ve son kullanıcının olabileceği unutulmamalıdır.

KAA uygulamalarında kullanılan algılayıcı düğümler, hem veri işleyici hem de ağ aktarıcısı (koordinatör, yönlendirici) olmak üzere iki özelliğe sahiptirler. Bu yüzden, iletişim iki nedenden dolayı yapılır [20, 24]:

- *Kaynak fonksiyonu:* Olay bilgisine sahip kaynak düğümler, verilerini alıcı birime göndermek için iletişim işlevsellikleri gerçekleştirirler.

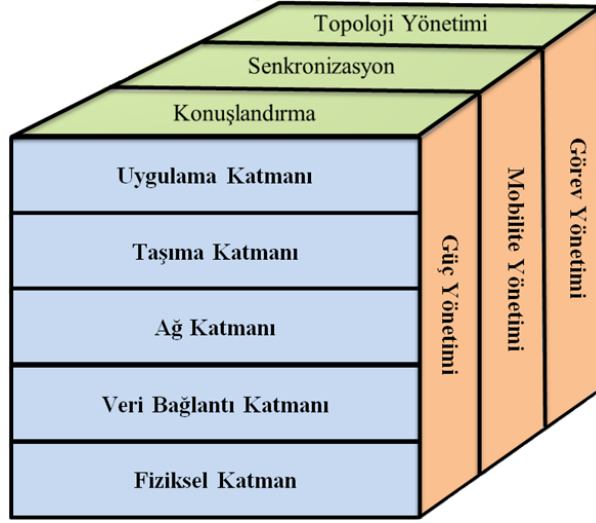
- *Yönlendirici fonksiyonu:* Algılayıcı düğümler, diğer düğümlerden alınan verileri alıcıya doğru giden çok sekmeli yol içerisinde bir sonraki hedefe iletmekte görev alırlar.



Şekil 2.1: Belirli bir bölgeye dağıtılmış kablosuz algılayıcı düğümler.

Alıcı ve tüm algılayıcı düğümler tarafından kullanılan protokol yığını Şekil 2.2'de gösterilmiştir. Bu protokol yığını; güç ve yönlendirme bilincini birleştirir, verileri ağ protokolleri ile entegre eder, kablosuz ortam aracılığıyla güç verimli iletişim kurar ve algılayıcı düğümlerin işbirlikçi çalışmalarını destekler [20-23]. Protokol yığını; fiziksel katman, veri bağlantı katmanı, ağ katmanı, taşıma katmanı, uygulama katmanı ve ayrıca lokalizasyon düzlemi, senkronizasyon düzlemi, topoloji yönetim düzlemi, güç yönetimi düzlemi, mobilite yönetimi düzlemi ve görev yönetimi düzleminde oluşur. Fiziksel katman; basit ama sağlam modülasyon, veri iletim ve alma tekniklerine yönelik ihtiyaçları giderir. Çevre gürültülü ve algılayıcı düğümler mobil olabileceğinden, bağlantı katmanı, hata kontrol teknikleri ile güvenilir iletişim sağlamakla ve komşu düğümlerin yayınları ile çakışmayı en aza indirmek için MAC üzerinden kanal erişimini yönetmekle sorumludur. Algılama görevlerine bağlı olarak, uygulama katmanında farklı türde uygulama yazılımları yapılandırılabilir ve kullanılabilir. Ağ katmanı, taşıma

katmanı tarafından sağlanan verileri yönlendirme ile ilgilenir. Taşıma katmanı, KAA eğer algılayıcı ağ uygulamasını gerektiriyorsa veri akışının sağlanmasına yardımcı olur. Bunlara ek olarak; güç, mobilite, görev yönetimi düzlemleri, algılayıcı düğümler arasındaki güç, hareket ve görev dağılımını izler [24]. Bu düzlemler, kablosuz algılayıcı düğümlerin algılama görevlerini koordine etmelerine ve genel güç tüketimini düşürmeye yardımcı olurlar.



Şekil 2.2: KAA protokol yığını.

Güç yönetim düzlemi, bir algılayıcı düğümün gücünü nasıl kullandığını yönetir. Örneğin: Algılayıcı düğüm, komşularından birinden bir mesaj aldıktan sonra alıcısını kapatabilir. Bu, çoğaltılan mesajları almaktan kaçınmak içindir. Ayrıca, algılayıcı düğümün güç seviyesi düşük olduğunda, algılayıcı düğüm komşularına gücünün azaldığını ve yönlendirme mesajlarına katılamayacağını yayınlar. Kalan güç, algılama için ayrılır [44-46]. Mobilite yönetim düzlemi, algılayıcı düğümlerin hareketini algılar ve kaydeder; bu sayede kullanıcıya bir geri dönüş yolu her zaman sağlanır ve algılayıcı düğümler komşularını takip edebilirler. Algılayıcı düğümler, güçlerini ve görev kullanımlarını bu komşu algılayıcı düğümleri bilerek dengeleyebilirler [47-49]. Görev yönetimi düzlemi, belirli bir bölgeye verilen algılama görevlerini dengeler ve programlar. Bu bölgedeki tüm algılayıcı düğümlerin algılama görevini aynı anda yerine getirmesi gerekli değildir. Sonuç olarak, bazı algılayıcı düğümler, güç düzeylerine bağlı olarak diğerlerinden daha fazla görev yerine getirir [50, 51]. Bu yönetim düzlemleri, algılayıcı düğümlerin verimli (güç tasarruflu) bir şekilde birlikte çalışabilmeleri, mobil algılayıcı ağda veri yönlendirebilmeleri ve aralarında kaynak paylaşımı yapabilmeleri

için gereklidir. Bunlar olmadan, her bir algılayıcı düğüm sadece bireysel olarak çalışır. Tüm algılayıcı ağ açısından bakıldığında, algılayıcı düğümlerin birbirleriyle işbirliği içinde çalışmaları daha verimli olur ve böylece KAA'ların çalışma ömrü uzatılabilir.

Katmanlı mimari yığını, başlangıçta İnternetle yaşadığı başarısı nedeniyle KAA'ların gelişiminde de benimsenmiştir. Ancak, KAA uygulamalarının geniş çaplı gerçekleştirilmesi, kablosuz kanalın daha yüksek katman protokolleri üzerinde önemli bir etkiye sahip olduğunu ortaya koymaktadır. Ayrıca, kaynak kısıtlamaları ve KAA'ların uygulamalara özgü doğası, katmanlı protokol yığınına entegre eden katmanlar arası çözümlere yol açar. Katmanlar ve ilişkili arabirimler arasındaki sınırları kaldırarak, kod alanı ve çalışma yükü açısından artan verimlilik elde edilebilir.

Katmanlı yığındaki iletişim işlevselliklerine ek olarak, KAA'lar, önerilen çözümlerin uygulanmasına katkı sağlayan çeşitli fonksiyonelliklerle donatılmışlardır. Bir KAA'da görev yapan her bir algılayıcı düğüm, iç işlemler için kendi yerel saatine sahiptir. Algılama, veri işleme ve iletişim dâhil olmak üzere algılayıcı düğümün operasyonu ile ilgili her olay, yerel saat aracılığıyla kontrol edilen zaman bilgisi ile ilişkilidir [52-54]. Kullanıcılar, çoklu algılayıcıların işbirlikçi çalışmaları sonucunda topladıkları bilgilerle ilgilendiklerinden, her bir algılayıcı düğümdeki veri ile ilişkilendirilmiş zaman bilgisinin tutarlı olması gerekir. Ayrıca, KAA bulunduğu fiziksel ortamı tam olarak modelleyebilmek için dağıtık algılayıcı düğümler tarafından algılanan olayları doğru bir şekilde düzenleyebilmelidir. Dolayısıyla, zaman bilgisi gereksinimi, KAA'larda zaman senkronizasyonu protokollerinin geliştirilmesine yol açmıştır [55-57].

KAA'nın bir bölgedeki fiziksel olaylarla etkileşim içinde olması, zaman bilgisine ek olarak konum bilgisi de gerektirir. Gözlemlenen alanın doğru bir şekilde incelenmesini sağlamak için toplanan bilgilerin algılayıcı düğümlerin konumları ile ilişkilendirilmesi gerekir [58-60]. Ayrıca, KAA'lar, takip algoritmalarına eklenmiş konum bilgilerine ihtiyaç duyan gözetim uygulamalarında (örneğin; hareketli nesnelerin izlenmesi) kullanılabilir. İlaveten, konum tabanlı hizmetler ve iletişim protokolleri pozisyon bilgisi de gerektirebilir. Bu yüzden, lokalizasyon (konuşlandırma) protokolleri iletişim yığını içerisine dâhil edilmiştir [61-63].

KAA'nın bağlanabilirliğini ve kapsama alanını muhafaza edebilmek için çeşitli topoloji yönetimi çözümleri gereklidir. Topoloji yönetimi algoritmaları, ağ dağıtımına yönelik daha uzun kullanım ömrü ve verimli bilgi içeriği ile sonuçlanan etkili yöntemler sağlar. Ayrıca, topoloji kontrol protokolleri, ağ bağlantısının sürekli sağlanması sırasında enerji tüketimini en aza indirmek için algılayıcı düğümlerin iletim güç seviyelerini ve aktivite sürelerini belirlemeye destek olur. Son olarak, kümeleme protokolleri, ölçeklenebilirliği geliştirmek ve ağ ömrünü uzatmak amacıyla ağ kümeler halinde organize etmede kullanılır [64-70].

2.1.1. Fiziksel Katman

Fiziksel katman; frekans seçimi, taşıyıcı frekans üretimi, sinyal algılama, sinyal yayılma etkileri, enerji verimliliği (güç tasarrufu), modülasyon/demodülasyon ve veri şifreleme işlemlerini kapsar. Frekans üretimi ve sinyal algılama, daha çok temel donanım ve alıcı-verici tasarımı ile ilgilidir.

2.1.2. Veri Bağlantı Katmanı

Veri bağlantı katmanı; veri akışlarının çoğullanması, veri çerçevesi tespiti, ortam erişimi ve hata kontrolü işlemlerinden sorumludur. Bu, bir iletişim ağında noktadan noktaya ve noktadan çoklu noktaya güvenilir bağlantılar sağlar.

2.1.2.1. Ortam Erişim Kontrolü (Medium Access Control-MAC)

Kablosuz, çok sekmeli ve kendi kendini düzenleyen bir algılayıcı ağda, MAC protokolünün gerçekleştirmesi gereken iki hedef vardır. Birincisi, ağ altyapısının oluşturulmasıdır. Binlerce algılayıcı düğüm bir algılayıcı alana yoğun bir şekilde dağıtılmış olabileceğinden, MAC düzeneğinin veri aktarımı için iletişim bağlantıları kurması gerekir. Bu, sekmeden-sekmeye (hop-by-hop) kablosuz iletişim için gereken temel altyapıyı oluşturur ve ağa kendi kendini organize edebilme becerisi sağlar. İkinci hedef ise, algılayıcı düğümler arasında iletişim kaynaklarını eşit ve verimli bir şekilde paylaşmaktır. Kaynaklar zaman, enerji ve frekans içerir. Bu gereksinimleri gidermede KAA'lar için çeşitli MAC protokolleri geliştirilmiştir [20, 23, 24].

Ortam erişim düzeni/planı nasıl olursa olsun, bir KAA'da enerji verimliliği son derece büyük önem taşımaktadır. MAC protokolünün algılayıcı düğümde güç tasarruf modları çalıştırmayı kesinlikle desteklemesi gerekir [71]. Güç tasarrufunun en belirgin yöntemi,

ihtiyaç duyulmadığı zaman alıcıyı-verici modülü kapatmaktır. Bu güç tasarruf yöntemi önemli enerji kazançları sunuyor gibi görünse de, ağın bağlanabilirliğini zorlaştırabilir. Alıcı-verici kapatıldıktan sonra, algılayıcı düğüm, ağ bağlantısı kesileceğinden komşularından herhangi bir paket alamaz. Ayrıca, alıcı-vericiyi açmak ve kapatmak, hem donanım hem de yazılım için gerekli başlatma ve kapatma işlemleri nedeniyle oluşacak enerji tüketimi açısından ek yüke sahiptir. Aslında, alıcı-verici her boşta kalma sırasında (boş zaman diliminde) rastgele/bilinmeden kapatılırsa, bir süre sonra algılayıcı düğüm alıcı-vericinin açık bırakılması halinde harcayacağı enerjiden daha fazla enerji harcayacak duruma gelebilir [20, 23, 24].

Sonuçta; güç tasarrufu modunda çalışma, sadece bu modda harcanan zaman belirli bir eşik değerinden daha fazla ise güç tüketimi açısından enerji tasarrufludur. Mikroişlemci, bellek, algılayıcı, ADC ve alıcı-verici durumlarının sayısına bağlı olarak kablosuz algılayıcı düğüm için yararlı birçok çalışma modu mevcut olabilir. Çalışma modların her biri; kendi güç tüketimleri ve gecikme yükleri (çalışma moduna giden ve bu moddan gelen geçiş gücü) ile karakterize edilebilir.

2.1.2.2.Hata Kontrolü

Veri bağlantı katmanının bir diğer önemli fonksiyonu, iletilen verinin hata kontrolüdür. İletişim ağlarında hata kontrolünün 3 önemli modu; *ileri hata düzeltme (FEC)*, *otomatik tekrar isteği (ARQ)* ve *hibrid ARQ*'dur. KAA uygulamalarında ARQ'nun yararlılığı, fazladan yeniden iletim maliyeti ve ek yükü ile sınırlıdır. Diğer yandan, kod çözme karmaşıklığı FEC'de daha fazladır çünkü hata düzeltme becerilerinin yapılandırılması gerekir. Sonuç olarak, düşük karmaşıklık kodlamaya ve kod çözmeye sahip basit hata kontrol kodları, KAA'lar için en iyi çözümleri sunabilir [72-74]. Böyle bir düzenek tasarımı, kanal özellikleri ve uygulama teknikleri hakkında çok kapsamlı bilgilere sahip olmak gerekir.

2.1.3.Ağ Katmanı

Şekil 2.1'de gösterildiği gibi, algılayıcı düğümler belirli bir bölgeye yoğun olarak dağıtılmıştır. Fiziksel olay hakkında toplanan bilgiler, algılama alanından uzakta bulunan birimlere (koordinatör, ağ geçidi, İnternette bir sunucu ya da son kullanıcı) iletilmelidir. Ancak, algılayıcı düğümlerin sınırlı iletişim menzilleri, her bir algılayıcı düğüm ile alıcı birimler arasında doğrudan iletişim kurulmasını önler. Bu

işlem, röle görevi yapacak ara algılayıcı düğümlerin bulunmasını, algılayıcı düğümler ve alıcı birimler arasında etkin çok sekmeli kablosuz yönlendirme protokollerinin kullanılmasını gerektirir [20, 24]. Dolayısıyla, kablosuz tasarsız ağlar için geliştirilmiş mevcut yönlendirme teknikleri KAA'ların gereksinimlerine uygun değildir. KAA'ların ağ katmanını genellikle aşağıdaki prensiplere göre tasarlanmıştır:

- Algılayıcı ağlar veri merkezlidir.
- Enerji verimliliği (güç tasarrufu) her zaman önemli bir husustur.
- Yönlendirmeye ilaveten, röle düğümler yerel işlem yoluyla birden fazla komşudan veri toplar.
- KAA'larda çok sayıda algılayıcı düğümün kullanılması nedeniyle, her düğüme özgün kimlikler atanamayabilir. Düğümlerin kendi verilerine veya bulundukları konumlara göre adreslenmeleri gerekebilir.

KAA'larda yönlendirme işlemi için önemli bir konu da yönlendirmenin veri merkezli sorgulara dayalı olmasıdır [20]. Kullanıcı tarafından istenen bilgilere göre, yönlendirme protokolü, gerekli bilgiyi sağlayacak farklı düğümleri de adreslemelidir. Daha belirgin olarak, kullanıcılar tek bir düğümü sorgulamaktan ziyade olayın bir özelliğini sorgulamakla daha çok ilgilenir. Örneğin: “Sıcaklığın 21°C'den daha yüksek olduğu alanlar” sorgusu, “düğüm 56 tarafından okunan sıcaklık” sorgusundan daha yaygındır.

Ağ katmanının bir diğer önemli fonksiyonu; diğer algılayıcı ağlar, komut ve kontrol sistemleri ve İnternet gibi dış ağlar ile ağlar arası iletişimi sağlamaktır. Örneğin; alıcı düğümler, diğer ağlara erişim için ağ geçitleri gibi kullanılabilir. Ya da, alıcı düğümleri birbirine bağlayarak bir omurga oluşturulabilir ve bu omurganın diğer ağlara bir ağ geçidi ile erişimi sağlanabilir [20, 23, 24].

2.1.4. Taşıma Katmanı

Taşıma katmanı, özellikle İnternet veya diğer dış ağlar üzerinden KAA'a erişilmesi planlanıyorsa gereklidir. Mevcut iletim penceresi ve akış-kontrol mekanizmaları içeren TCP, KAA uygulamalarının ortaya koyduğu özel zorlukları gidermez. TCP gibi protokollerden farklı olarak, algılayıcı ağlardaki uçtan uca iletişim düzenekleri global adresleme tabanlı değildir. Bu şemalar, veriye veya konuma göre yapılan adreslemelerin veri paketlerinin hedef konumlarını göstermek için kullanıldıklarını göz önünde

bulundurmalıdır [23]. Güç tüketimi, ölçeklenebilirlik, veri merkezli yönlendirme gibi etmenler; algılayıcı ağlarda taşıma katmanının farklı ele alınması gerektirir. Dolayısıyla, bu gereksinimler taşıma katmanı protokollerinde yeni yöntemlerin geliştirilmesine duyulan ihtiyaçları vurgulamaktadır [75-78].

Algılayıcı düğümler, sınırlı kapasitede güç kaynağı ve bellek gibi donanım kısıtlamalarından etkilendiği için taşıma katmanı protokollerinin geliştirilmesi zor bir iştir. Ayrıca, her bir algılayıcı düğüm İnternetteki bir sunucu gibi büyük miktarda veri depolayamaz ve alındı bildirimleri algılayıcı ağlar için çok masraflıdır. Bu nedenle, algılayıcı ağlarda UDP tipi protokollerin kullanıldığı durumlarda, uçtan uca iletişimi muhtemelen alıcı birimlerde ayıran yeni düzenekler gerekli olabilir [20, 23].

KAA’larda kullanılan taşıma katmanı protokolleri, iki ana işlev için gereklidir. Bunlar; tıkanıklık denetimi ve güvenilirliktir. Sınırlı kaynaklar ve yüksek enerji maliyetleri, uçtan uca güvenilirlik sunan mekanizmaların KAA’larda kullanılmasını önlemektedir. Bunun yerine, KAA’larda yerel güvenilirlik sağlayan mekanizmalara ihtiyaç duyulur. Ayrıca, olaylar sırasında yüksek trafik nedeniyle meydana gelebilecek tıkanıklık, taşıma katmanı protokolleri ile hafifletilmelidir [79]. Taşıma katmanı protokolleri, algılayıcı düğümlerin veri işleme, depolama ve enerji tüketimindeki kısıtlamalar nedeniyle ortak çalışabilme yeteneklerini sonuna kadar kullanmalarını ve bilgiyi algılayıcı düğümlerden ziyade alıcı birimlere kaydırmayı amaçlar [20, 23].

2.1.5. Uygulama Katmanı

Uygulama katmanı, KAA topolojisini uygulamalara göre soyutlar. Kullanıcılara KAA aracılığıyla fiziksel dünya ile etkileşimde bulunmak için gerekli ara-yüzler sağlar. Ana uygulamaların yanı sıra çok sayıda yönetim işlevleri de içerir. Uygulama katmanına yönelik çözümler üç kategoride tanımlanabilir. Bunlar: *kaynak kodlama* (her bir uygulamaya özel olan uygulama kodu), *sorgu işleme* ve *ağ yönetimidir* [20].

2.1.5.1. Kaynak Kodlama (Veri Sıkıştırma)

Kaynak kodlama, kablosuz iletişim yoluyla bilgi göndermede ilk adımdır. Bir algılayıcı düğümün iletecek verisi olacağı zaman, bilgi kaynağı ilk önce kaynak kodlayıcı ile kodlanır. Kaynak kodlama, kaynağı daha az bit ile temsil etmek için bilgi istatistiklerinden yararlanır (örneğin, kaynak kod sözcüğü) [20]. Kaynak kodlama aynı

zamanda veri sıkıştırma olarak da adlandırılır. Bu yüzden, bilgi içeriğinin tümünü ya da bir kısmını korurken artık bilgiler veri hacmini azaltmak amacıyla sıkıştırılır.

Temel sıkıştırma çözümleri, kodlama sonrası bilgilerin korunmasına bağlı olarak iki şekilde sınıflandırılır: kayıpsız sıkıştırma ve kayıplı sıkıştırma. Kayıpsız sıkıştırma, bilgi bütünlüğünü engellemeden veri içeriğini azaltır (örneğin, paket boyutunu düşürmek). Çok daha iyi sıkıştırma verimliliği, verilerde daha fazla azaltma sağlayan kayıplı sıkıştırma ile başarılabilir [20]. Kayıplı sıkıştırmanın yüksek karmaşıklığı nedeniyle, KAA’larda genellikle kayıpsız sıkıştırma teknikleri uygulanır.

Çeşitli kayıpsız sıkıştırma algoritmaları, hem dijital iletişim hem de hesaplama için geliştirilmiştir. Ancak, bu algoritmaların KAA’lara uygulanabilirliği sınırlıdır. Birçok algılayıcı düğümün programlama bellekleri sınırlı olduğundan, *LZW*, *S-LZW*, *gzip* ya da *bzip2* gibi daha yüksek karmaşıklığa sahip algoritmalar doğrudan uygulanamaz [80-83]. Ayrıca, bazı sıkıştırma algoritmaları sıkıştırma/açma için depolanacak statik bir sözlük gerektirir ve sık bellek erişimine ihtiyaç duyar. Okuma/yazma operasyonları kablosuz algılayıcı düğümlerde yüksek enerji tüketimi ile sonuçlandığından, bu çözümler enerji verimli değildir [84-86].

Sıkıştırma algoritmalarının enerji tüketimi bakımından verimsizliğine ilaveten, her bir algılayıcı düğümde toplanan bilgi artıklığı da her kaynakta etkili sıkıştırmaya izin vermeyebilir. KAA’ların yüksek yoğunlukta konuşlandırılması; çok sayıda ve birbirine yakın yerleştirilen düğümler tarafından toplanmış ilişkili bilgilerin ortaya çıkmasını sağlar. Dolayısıyla, bireysel kaynak düğümde sıkıştırma yerine, bir grup algılayıcı düğüm tarafından toplanan artık bilgiler sıkıştırılabilir. Bu, alıcıya iletilecek verilerde ciddi azalmalara yol açar ve güç tüketimini düşürür. Sonuç olarak, sıkıştırma işlemi için farklı kaynaklardaki korelasyondan faydalanan dağıtık sıkıştırma teknikleri KAA uygulamalarında kullanılabilir [80-86]. KAA’larda sıkıştırma işlemine yönelik iki ana yaklaşım vardır: *düğüm merkezli kaynak kodlama* (örneğin, Sensor LZW) ve sıkıştırma işlemini çoklu algılayıcı düğümlere dağıtma (*dağıtık kaynak kodlama*) [20].

2.1.5.2.Sorgu İşleme

KAA’lar, uygulama gereksinimlerine göre fiziksel olayı izleyen pek çok algılayıcı düğümden oluşur. Alıcı, algılanan ortam hakkında bilgiler içeren düğümlere sorgular

göndererek ilgili verilerin teslimini garantiye alır. Sorgu yanıtları, istenen ham veriyi anında alıcıya göndererek basitçe yapılabilir. Bununla birlikte, algılayıcı düğümlerin işlem kabiliyetleri, önemli ölçüde enerji tasarrufu sağlayacak şekilde ağ içerisinde bu sorguları işlemenin alternatif yollarını sunar.

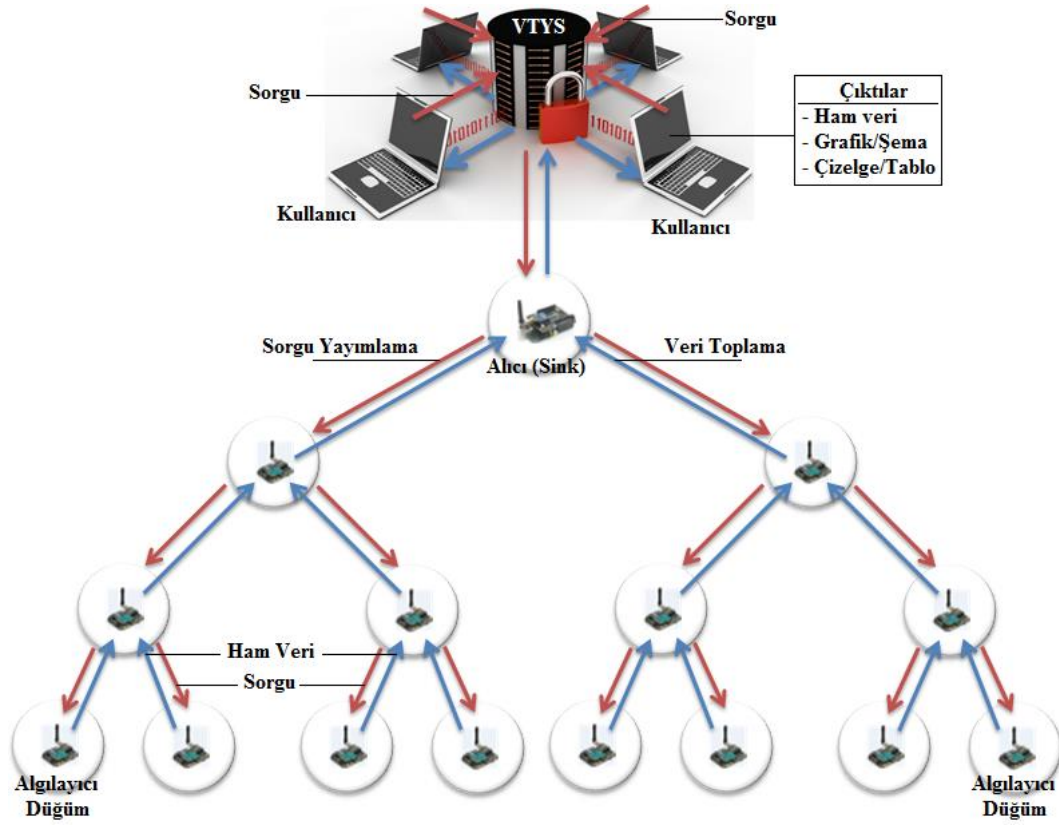
KAA, algılayıcı düğümlerin alıcıya sürekli olarak veri akışları yaptığı bir dağıtık veritabanı yönetim sistemi gibi düşünülebilir. Geleneksel dağıtık veriabanları için geliştirilmiş pek çok veritabanı yönetim sistemi olmasına rağmen, KAA'lara özgü karakteristikler bu çözümleri uygulanamaz kılar. KAA'lara özgü karakteristiklerden bazıları aşağıda sunulmuştur [20]:

- *Duraksız veriler:* Algılayıcı düğümler; her hangi bir sorguya uğramadan (veri talep edilmeksizin), genellikle iyi tanımlanmış zaman aralıklarında sürekli veri üretirler.
- *Gerçek zamanlı işleme:* Algılayıcı verileri genellikle gerçek zamanlı olayları temsil eder. Ayrıca, alıcıdaki diske ham algılayıcı veri akışlarını kaydetmek pahalıdır. Bu nedenle, veri akışları üzerinden yapılan sorguların gerçek zamanlı işlenmesi gerekir.
- *İletişim hataları:* Algılayıcı düğümler, verileri çok sekmeli kablosuz iletişim yoluyla iletirler. Kablosuz iletişim hataları, alıcıya ulaşan dağıtık bilgilerin güvenilirliğini ve gecikmesini etkiler.
- *Belirsizlik:* Algılayıcılar tarafından toplanan bilgiler çevreden gürültü içerir. Bununla birlikte, algılayıcı arızası ve algılayıcı düğüm yerleştirme gibi faktörler de algılayıcı düğümlerin ortamla ilgili okumalarını yanıltabilir.
- *Sınırlı disk alanı:* Algılayıcı düğümlerin sınırlı kapasitede disk alanları vardır. Dolayısıyla, algılayıcı düğümler tarafından gönderilen bilgiler daha sonraki zamanlarda sorgulanamayabilir.
- *İletişime karşı veri işleme:* KAA'larda veri işlemede harcanan enerji, veri iletişimde harcanan enerjiden çok daha azdır. Bu nedenle, algılayıcı düğümlerin veri işleme yetenekleri sorgu işleminde kullanılmalıdır.

Algılayıcı düğümlerdeki mevcut işlem gücü, KAA sorgu işlemede karşılaşılan zorluklara potansiyel çözümler sağlar. Alıcı tarafından gönderilen sorgular, ham algılayıcı gözlemlerini alıcıya göndererek cevaplanabilir. Bu yaklaşıma *veri depolama* (*data warehousing*) denir. Veri depolamada, algılayıcı düğüm sorgularını işleme ve

KAA'ya erişim ayrılır [20]. Sonuç olarak; bir merkezi veritabanı yönetim sistemi, klasik veritabanı yönetim teknikleri ile toplanan verilere erişim sağlamak için geliştirilebilir.

Depolama yaklaşımı, KAA'daki iletişim kaynaklarının aşırı kullanılmasına ve alıcıda yüksek oranda artık bilgi birikmesine yol açar. Örneğin; eğer bir uygulama belirli bir konumda sadece belirli bilgilerin ortalama değeri ile ilgileniyorsa, ortalamayı yerel olarak hesaplamak ve tüm bireysel düğümlerdeki bilgileri göndermektense bilgiyi alıcıya tek bir paket şeklinde göndermek, o konumdaki düğümler için daha verimli olur. Dolayısıyla, KAA'larda sorgu işlemeye yönelik mevcut teknikleri kullanmak verimsiz çözümlere sebep olabilir. Bunun yerine, KAA'lara uygun yeni yaklaşımlar gereklidir.



Şekil 2.3: Sorgu işleme mimarisi.

Kullanıcının KAA ile etkileşimde bulunduğu sorgu işleme yapısının genel mimarisi Şekil 2.3'de gösterilmiştir. Sorgu işleme çözümleri, kullanıcı ile algılayıcı düğümlerin dağıtık kümesi arasında bu etkileşimi gerçekleştirmek için gerekli araçları sağlar. Sorgu işleme çözümleri tarafından sağlanan servisler iki şekilde incelenebilir: *Sunucu tarafı servisler* ve *KAA tarafı servisler*. Her bir sorgu işleme yaklaşımında, kullanıcı ilgileri, ağın geri kalanına kolayca sorgulara dönüştürülebilen iyi kurulmuş sözdizimi (sintaks)

aracılığıyla temsil edilir [20]. Bu, sorgu işleme çözümlerinin sunucu-tarafı bölümünü oluşturur. KAA tarafı servisler ise, sorgu yayımlama ve veri toplama olmak üzere iki adıma ayrılabilir. Sorgu yayımlama, kullanıcı ilgilerine göre üretilmiş sorguları bir ağdaki tüm düğümlere ya da bunların alt gruplarına ileterek gerçekleştirilir. Veri toplamada, algılayıcı düğümler, eğer düğümlerin yerel olarak gözlenmiş algılayıcı verileri sorgularla eşleşiyorsa bu sorgulara yanıt verirler [20]. Sorguların enerji verimli temsili, yayımlanması ve ilgili verilerin toplanması sorgu işleme algoritmalarının tasarımında önemli bir sorundur.

2.1.5.3. Ağ Yönetimi

KAA'ların dinamik yapısı, bileşenleri izlemek ve yönetmek için etkin şemalar gerektirir. Herhangi bir ağ mimarisine benzer şekilde KAA'lar da, ağ yöneticileri veya sistem kullanıcılarının ağdaki sensör düğümleri ile kolayca etkileşime girebilmeleri için verimli yönetici araçları gerektirir. Ancak, KAA'larda bilgi iletimi, düşük güç kablosuz bağlantılar sebebiyle pahalı ve güvenilmezdir. Bu nedenle, yüksek enerji verimliliği ile güvenilir yönetim önemli bir zorluk teşkil eder. Benzer şekilde, sensör düğümleri veri işleme ve hafıza açısından sınırlıdır. Dolayısıyla, sensör düğümlerindeki ağ yönetim yükü mümkün olduğu kadar küçük olmalıdır.

Ağ yönetimi görevleri genellikle iki adımda gerçekleştirilir: *ağ izleme* ve *yönetim kontrolü*. KAA'lardan alınan bilgiler, ağ izlemenin bir parçası olarak ağın güncel durumunu değerlendirmede verimli bir şekilde toplanmalıdır. Toplanan bilgiler; bağlantırlık, kapsama (yayın alanı), topoloji, mobilite gibi ağ çapında verilere ilaveten düğüm durumu, kalan enerji, bellek kullanımı, verici gücü gibi düğüm tabanlı verileri içerir. Kablosuz iletişimin etkileri dikkate alındığında, bu dağıtık bilgi kümelerini toplamak oldukça güçtür. Ayrıca, KAA özelliklerindeki dinamik değişiklikler, doğru durum temsili için sürekli ağ izlemeyi gerektirir [20, 23, 24].

Çeşitli yönetim kontrolü görevleri, KAA'nın gözetimi sonucunda alınan bilgilere bağlı olarak operasyonun istenen durumunu korumak amacıyla uygulanabilir. Bu görevler arasında; rota yönetimi, protokol güncelleme, trafik yönetimi gibi ağ çapında işlemlerin yanı sıra algılayıcı düğümleri hareket ettirme, algılayıcı düğüm bileşenlerini kapatıp açma, verici gücü yönetimi, örnekleme oranı kontrolü gibi düğüm tabanlı işlemler bulunur. Ayrıca, KAA'nın istenen duruma ulaşıldığını garantiye almak için ağın daha

fazla izlenmesi gerekir. Bu nedenle, ağ izleme ve yönetim kontrolü görevleri, ağ yönetimi çözümleri için uyum içinde çalışmalıdır [20, 23, 24].

Ağ yönetimi, KAA'lara özgü özelliklerden dolayı çeşitli zorluklarla karşılaşır. Bu önemli zorluklar şunlardır [20, 23, 25]:

- *Veri aşırı yükleme:* KAA'ların yüksek yoğunlukta olması (çok sayıda algılayıcı düğüm içermesi), ağdan toplanan veri miktarını artırır. Ağ yönetimi çözümleri tarafından istenen farklı veri türleri, her bir algılayıcı düğümden gelen bilgileri önemli ölçüde artırır.
- *Güvenilir olmayan iletişim:* Çevresel koşullar, enerji kısıtlamaları, kablosuz kanal hataları ve ağ tıkanıklığı algılayıcı düğümlerin doğru izlenmesini zorlaştırır. Bu faktörler nedeniyle oluşabilecek çok sayıda paket kaybı, ağ yönetim kararlarında dikkate alınmalıdır.
- *Bilgi görselleştirme:* Çok sayıda algılayıcı düğümden sürekli olarak alınan büyük miktardaki verilerin görselleştirilmesi birçok zorluğu da beraberinde getirir.

Ağ yönetim çözümleri, KAA'lara özgü kullanılan birkaç ortak karakteristiği paylaşırlar. Algılayıcı düğümler üzerinde çalışan ağ yönetim bileşenleri, algılayıcı düğümlerin kaynak kısıtlamalarından dolayı enerji, veri işleme ve bellek ek yüklerini en aza indirmede basit işlemleri yürütür. Çoğu çözüm olay tabanlı yaklaşımlara dayanır ve algılayıcı düğümler kendi durumlarındaki değişikliklere göre otonom bir şekilde hareket eder. Böylece, değerli kaynaklar uçtan uca tarama/sorgulama için tüketilmez [20-25].

Ağ yönetim çözümleri, kablosuz kanalın olumsuz/istenmeyen etkileri ile mücadele etmek amacıyla dayanıklı/sağlam ve hata toleranslı olacak biçimde geliştirilmiştir. Bu nedenle, ağ izleme ve yönetim kontrol görevleri, belirli düğümlere dayalı olmayıp daha ziyade dağıtık operasyonları amaçlarlar. Kanal hataları, çevresel faktörler, düğüm hareketleri sebebiyle meydana gelen dinamik değişiklikler ağ operasyonlarını dinamik olarak değiştiren uyarlanabilir çözümler gerektirir. Ayrıca, ağ yönetim çözümleri KAA mimarisi üzerine yerleşir ve diğer ağ çözümlerinde olduğu gibi ağ yönetim protokolleri ölçeklenebilir olacak şekilde tasarlanır. Dolayısıyla, KAA'ları İnternet, hücresel ağlar gibi diğer iletişim mimarileriyle entegre etmek için uygun arayüzler gerekir [20-25].

Ağ yönetim çözümleri, *yönetim türü* ve *ağ mimarisi* olmak üzere iki farklı kritere göre sınıflandırılır. Yönetim türünde üç farklı yaklaşım vardır [20]:

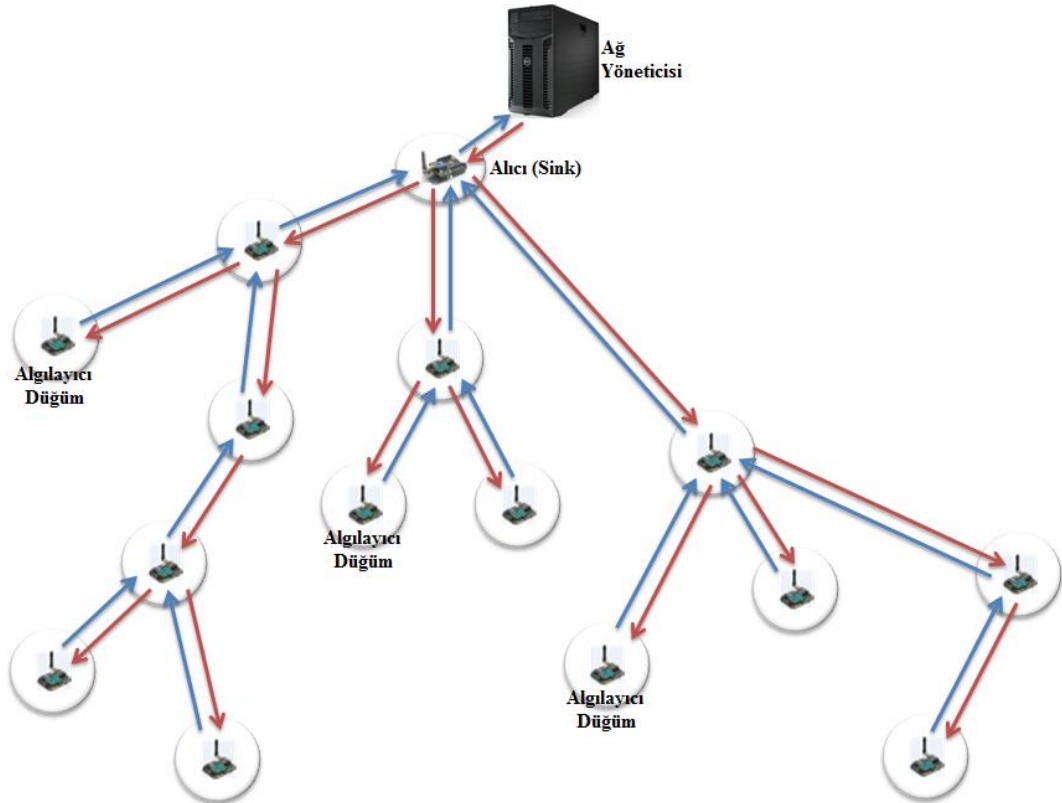
- *Pasif*: Ağ izleme sürekli yapılır. Alınan veriler alıcıda işlenir. Bu tür bir yönetim şeması, herhangi bir durum değişikliğini algılamada ileri-işlemeye dayanır. Durum değişiklikleri halinde, yönetim kontrolü görevleri dağıtılır.
- *Reaktif (olay tetiklemeli)*: Reaktif çözümler, ağ yönetimi için algılayıcı düğümlerin on-board işlem becerilerine bağlıdır. Düğümün çeşitli durumları ile ilişkili belirli olaylar, olay tabanlı sorgulamalara benzer şekilde tanımlanır. İlgilenilen bir olay meydana geldiğinde ağ izleme başlatılır. Yönetim kontrolü görevleri, olaylara bir yanıt olarak gönderilir. Ağ yönetimi çözümleri, toplanan verilerin gerçek zamanlı işlenmesine dayanır.
- *Proaktif*: Ağ izleme aktif olarak yapılır. Toplanan veriler gerçek zamanlı işlenir. Ağ yönetimi protokolü, ileride meydana gelebilecek olayları tahmin etmek ve denetlemek için her değişikliğe bir tepki verir.

Ağ mimarisi, yönetim operasyonlarının ağ içerisinde nerede bulunduklarına bağlı olarak üç farklı yapıdadır [20]:

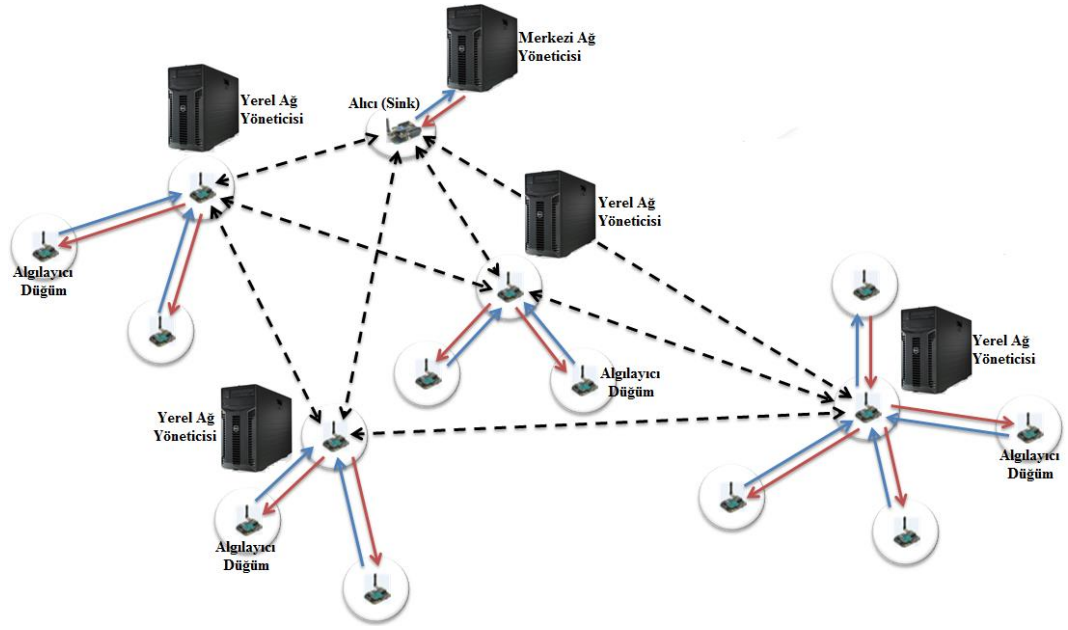
- *Merkezi ağ mimarisi*: Ağ yönetim işlevsellikleri alıcıda bulunur (Şekil 2.4). KAA, hem ağ gözetimi hem de yönetim kontrolünde veri iletim amaçları için kullanılır. Alıcı, ağ hakkında global bilgi toplar ve bu bilgilere dayalı kompleks yönetim kararları uygular. Merkezi mimari, kaynak kısıtlı algılayıcı düğümlerde işlem yükünü azaltır. Ancak, veri izleme işleminin ağdaki her bir düğümden alıcıya iletilmesi gerekir. Bu da ağdaki iletişim yükünü artırır. Bu durumda, bireysel düğüm bilgileri önemli olduğundan birleştirme/kümeleme çözümleri de KAA'da uygulanamayabilir.
- *Dağıtık ağ mimarisi*: Alıcı gibi tek bir ağ yöneticisine bağlı kalmak yerine, dağıtık çözümler, ağ içerisinde dağıtılmış birkaç ağ yöneticisi çalıştırır (Şekil 2.5). Belirli algılayıcı düğümler veya özel cihazlar yönetim görevleri dağıtan ağ yöneticileri biçiminde kullanılır. Her bir ağ yöneticisi, algılayıcı düğümlerin bir kümesi ile ilişkilidir. Bu durum; düğümler tek bir merkezi yönetici yerine yerel ağ yöneticileri tarafından kontrol edildiğinden, çözümlerin dayanıklılığını artırır ve iletişim

maliyetlerini azaltır. Ancak, dağıtık ağ yöneticileri arasındaki koordinasyon ve ilgili iletişim önemli bir sorundur. Ayrıca, işlem yükünün ağ yöneticilerinde fazla olması ya özel donanım birimleri gerektirir ya da sınırlı ağ yönetim işlevsellikleri ile sonuçlanır.

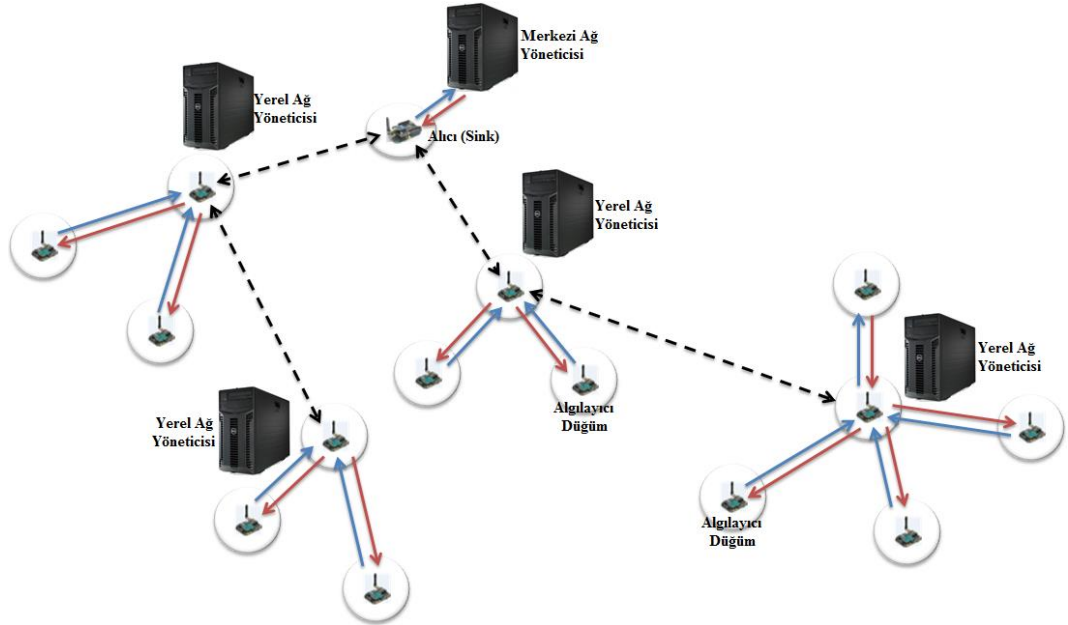
- *Hiyerarşik ağ mimarisi:* Dağıtık ağ yönetim çözümleri, yöneticiler ağ içerisinde dağıtıldığından, ağın global görünümünü toplama bakımından çeşitli problemlerle karşı karşıya kalırlar. Bunun yerine, hiyerarşik ağ yönetim çözümleri, ağdaki ağ yöneticilerini hiyerarşik biçimde çalıştırır (Şekil 2.6). Ağ yönetim görevleri, ağ yöneticileri arasında dağıtılmış olsa da -alıcı en üst seviye kabul edilirse- her yönetici daha üst düzey yöneticiye rapor verir. Dolayısıyla, yerel yönetim görevleri, ağdaki iletişim maliyetini artırmadan daha alt düzey yöneticilerde yürütülebilir. Ayrıca, ağın global görünümü, yüksek düzey yöneticilere ve alıcıya gerekli bilgileri raporlayarak korunabilir. Bu durum, ağ çapında yönetim kontrolü kararlarının alıcıda alınmasını sağlar.



Şekil 2.4: Merkezi ağ mimarisi.



Şekil 2.5: Dağıtık ağ mimarisi.

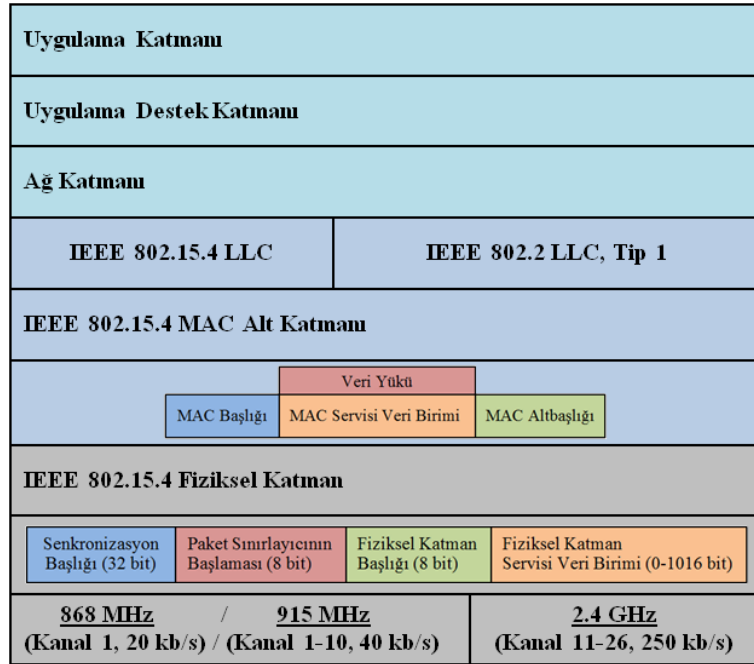


Şekil 2.6: Hiyerarşik ağ mimarisi.

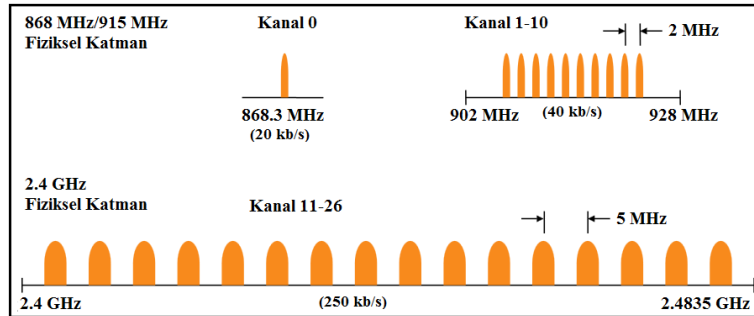
2.2. STANDARDİZASYON

Algılayıcı düğümlerdeki heterojenlik, KAA uygulamalarının gerçekleştirilmesinde uyumluluk sorunlarına sebep olmaktadır. Dolayısıyla, iletişimin belirli yönlerinin standardize edilmesi gereklidir. Şekil 2.7’de protokol mimarisi verilmiş IEEE 802.15.4 standardı [87], uzun pil ömrü ve basitlik hedeflenerek düşük veri hızlı kablosuz alıcı-verici teknolojinin özelliklerini belirlemek amacıyla oluşturulmuştur. İletişim için üç

farklı frekans bandı kullanılmaktadır. Şekil 2.8’de gösterilen bu frekans bandları; 868 MHz (Avrupa), 915 MHz (Amerika) ve 2.4 GHz (Global) olarak belirlenmiştir. Fiziksel katman; 868/915 MHz bandlarında ikili faz kaydırmalı anahtarlama (BPSK), 2.4 GHz bandında ise ofset karesel faz kaydırmalı anahtarlama (O-QPSK) kullanır. MAC (Medium Access Control-Ortam Erişim Denetimi) katmanı; yıldız, örgü, küme ve ağaç tabanlı topolojilere yönelik iletişim sağlar. Düğümlerin iletim aralığı 20-250 kbps veri hızları ile 10-100m mesafe olarak kabul edilir [87]. KAA araştırmalarında geliştirilen yeni platformların çoğu IEEE 802.15.4 standardı ile uyumludur. IEEE 802.15.4 standardı, geniş bir kitle tarafından kabul görmüş olup düşük güç iletişimde fiziksel ve MAC katmanları için fiili standart haline gelmiştir. Bu durum, aynı ağ içerisinde farklı yeteneklere sahip platformların entegrasyonunu sağlar.

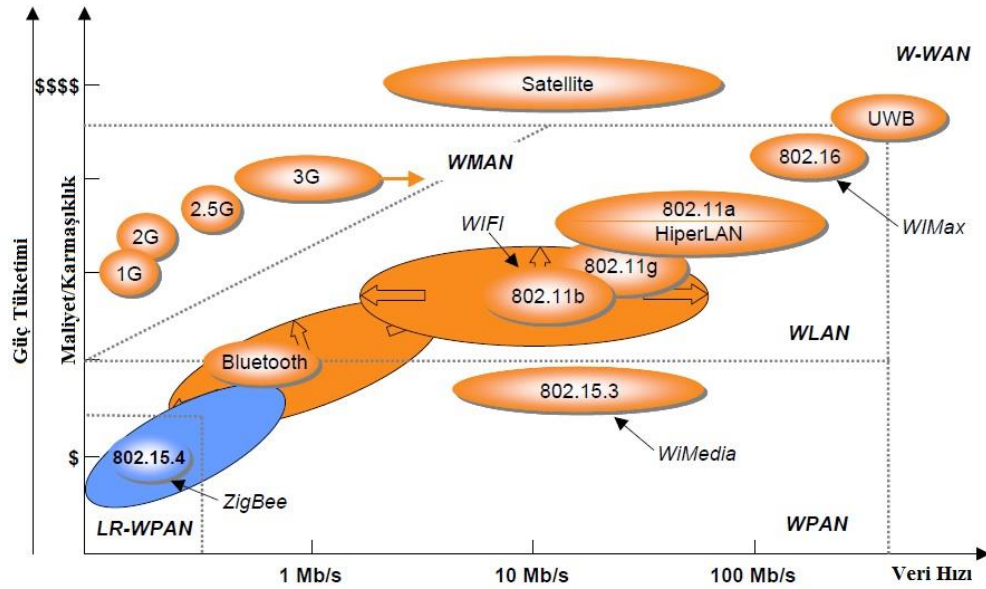


Şekil 2.7: IEEE 802.15.4 protokol mimarisi.



Şekil 2.8: IEEE 802.15.4 frekans bandları.

IEEE 802.15.4 standardına ilaveten, çeşitli alanlarda düşük güç tüketimli kablosuz ağların kullanılmasına yönelik pek çok standart geliştirilmiştir (Şekil 2.9). Bluetooth [88] ve WLAN [89] gibi standartların düşük güçlü KAA uygulamaları için genellikle çok uygun olmadığı kabul edilir. Öte yandan, özellikle kablosuz kontrol, gözetim ve izleme uygulamalarının tipik ihtiyaçlarına hitap eden ZigBee [90], WirelessHART [91], Wina [92], SP100.11a [93] ve KAA'lar ile İnternet (veya diğer ağlar) arasında uyumluluk sağlayan 6LoWPAN [94] gibi standartlar, KAA'ların hızlı bir şekilde gelişmesine katkı sağlamıştır. KAA'ların geliştirilmesinde kullanılan üç ana standart: *ZigBee*, *WirelessHART* ve *6LoWPAN* sonraki bölümlerde incelenmiştir.



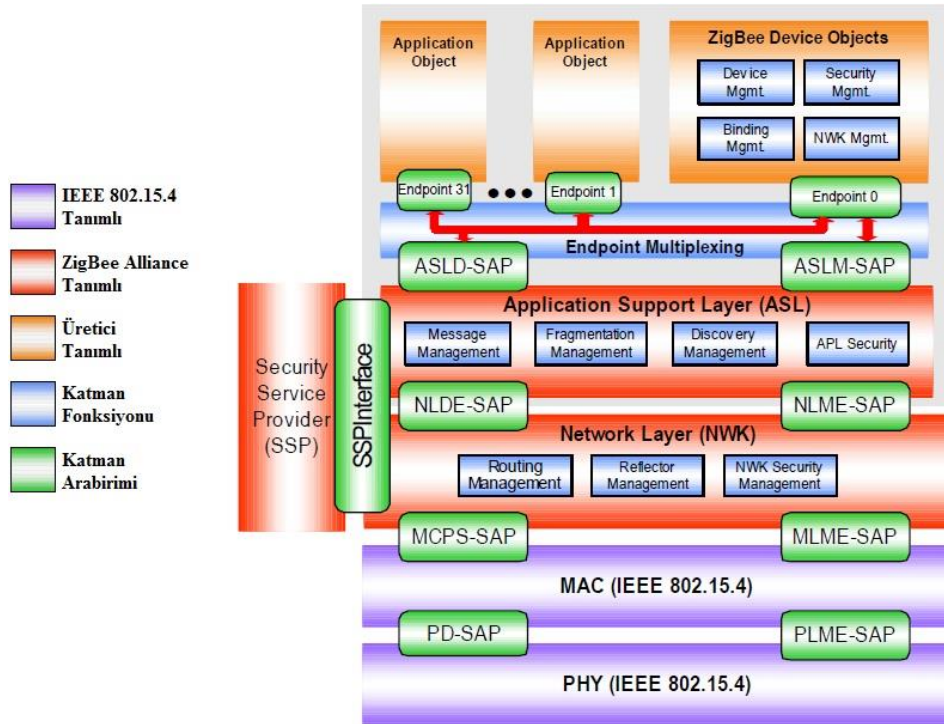
Şekil 2.9: Farklı uygulamalar için geliştirilmiş kablosuz ağ standartları.

2.2.1. ZigBee

ZigBee [90] standardı; düşük veri hızları, düşük güç tüketimi, güvenlik ve kişisel alan ağlarında (WPAN) güvenilirlik gibi unsurları destekleyen uygun maliyetli, standart bazlı kablosuz ağ çözümleri için piyasa ihtiyacına yönelik oluşturulmuştur. ZigBee, IEEE 802.15.4 altyapısında geliştirilmiş ve standart sarmal ağlar ile uygulama profilleri kullanılarak kurulan kısa mesafe kablosuz ağ standardıdır. Yıldız bağlantı ve noktadan noktaya bağlantı şekillerini içeren çoklu ağ topolojilerine sahiptir. Güvenirlik, düşük maliyet ve enerji tasarrufu gibi avantajları göz önüne alındığında ZigBee; pc girdi aygıtları, algılayıcı ve ağ yönetim cihazlarının kablosuz bağlantılarında kullanılması uygun olan bir standarttır [95]. ZigBee, kablosuz iletişim kanallarının otomatik olarak aranmasına ve çok sayıda kablosuz ağın bir arada var olmasına imkân tanır. KAA'larda,

uzaktan görüntüleme uygulamalarında, otomasyon ve kontrol sistemlerinde, bina içi ortamlarda kablosuz iletişim sağlamak için yaygın biçimde kullanılmaktadır.

ZigBee cihazlar, tüm dünya genelinde lisans-ücretsiz kullanıma açık olan 2.4 GHz ISM frekans bandında çalışır. Bu cihazlar; 2.4 GHz frekans bandında 10 kanal ile 250 kbps, 915 MHz frekans bandında 6 kanal ile 40 kbps ve 868 MHz frekans bandında 1 kanal ile 20 kbps hızlara erişilebilmektedir. ZigBee cihazların erişim mesafesi, iletim gücüne ve çevresel etkenlere göre 10 ile 100 metre arasında değişmektedir. ZigBee aygıtları, veri akışına bağlı olarak derin uyku modlarına sahiptir ve enerji tasarrufu sağlarlar. Uyku devrelerinin gelişmiş özellikleri, düşük güç harcamaları ve ideal bağlantı teknikleri sayesinde ZigBee cihazların batarya kullanım süresi çok uzundur [96].



Şekil 2.10: ZigBee protokol yığını.

ZigBee standardı, özellikle IEEE 802.15.4 standardıyla beraber tanımlanmaktadır. Bu yüzden, her ikisi de genellikle karıştırılır. Bununla birlikte, Şekil 2.10'da gösterildiği gibi her bir standart, protokol yığınının belirli katmanlarını açıklar. Fiziksel ve MAC katmanları IEEE 802.15.4 standardı, ağ ve uygulama katmanları ZigBee standardı, uygulama nesneleri ise kullanıcı tarafından tayin edilir [97]. Çeşitli uygulamalara uyum sağlamak amacıyla üç trafik tipi belirlenmiştir. Birincisi; *periyodik veri trafiği*, algılayıcı düğümlerin fiziksel olaylar hakkında sürekli bilgi sağladıkları gözetim

uygulamalarında gereklidir. Veri alışverişi, ağ denetleyicisi veya yönlendirici ile kontrol edilir. İkincisi; *aralıklı veri trafiği*, çoğu etkinlik/olay tabanlı uygulama için geçerli olup ya uygulama ya da dış bir etken ile tetiklenir. Bu tür trafik, ağdaki her bir yönlendirici düğüm aracılığıyla yönetilir. Cihazlar, enerji tasarrufu sağlamak amacıyla çoğu zaman uyku modunda oldukları bağlantı kesik modda çalışırlar. Bilgi iletilmesi gereken zamanlarda alıcı-verici etkinleştirilir ve cihaz ağ ile bağlantıya geçer. Son olarak, *tekrarlı düşük gecikmeli veri trafiği*, belirli bir süre içerisinde tamamlanması gereken belirli iletişim çeşitleri için kullanılır. Bu tür trafik, IEEE 802.15.4 standardı tarafından tanımlanan sorgulama tabanlı çerçeve yapısı ile sağlanır [20, 23].

ZigBee ağ katmanı, ağ işlemleri ile ilgili yönetim işlevsellikleri sağlar. Yeni bir ağ kurulması, cihazların ağa katılması ya da ağ üyeliğini bırakması gibi prosedürleri tanımlar. Ağ işletimine bağlı olarak her cihazın iletişim yığını yapılandırır. ZigBee cihazları, kullanım ömürleri boyunca farklı ağların parçaları olabildiklerinden standart, aynı zamanda esnek adresleme mekanizması sunar. Buna göre, ağ koordinatörü, cihazlar ağa katıldıkları anda bir adres atar. Sonuç olarak, her cihazın özgün kimliği iletişim için kullanılmaz; fakat, iletişim sırasında verimliliği artırmak için daha kısa bir adres tayin edilir. Bir ağaç mimarisinde, bir cihazın adresi aynı zamanda yönlendirme amacıyla kullanılan üst ögesini (kaynak düğümü) belirtir. Ağ katmanı, cihazlar ve ağ denetleyicileri arasında senkronizasyon sağlar. Çok sekmeli yollar, tanımlı protokollere göre ağ katmanı tarafından oluşturulur [98].

Şekil 2.10'da gösterildiği üzere, ZigBee standardı, uygulama katmanındaki belirli bileşenleri tanımlar. Bu katman; uygulama destek alt katmanı, ZigBee aygıt nesnesi ve üretici tarafından tanımlanan uygulama nesnelerinden oluşur. Uygulamalar, üretici tanımlı uygulama nesneleri aracılığıyla yürütülür ve standartla belirlenen gereksinimlere dayalı gerçekleştirilir. ZigBee aygıt nesnesi, ağ işlemi için cihaz tarafından sağlanan fonksiyonları, ağ koordinatörü veya yönlendirici gibi cihazların rollerini tanımlar [99]. Ayrıca, bir cihazın ağ ile ilişkilendirilmesi gerektiği zaman, bağlanma istekleri ZigBee aygıt nesnesi üzerinden yönetilir. Son olarak, uygulama destek alt katmanı, cihazlara keşif kabiliyeti sağlar. Böylece bir cihazın komşuları ve bu komşular tarafından sağlanan işlevler depolanabilir. Bu bilgiler, aynı zamanda özel fonksiyonlara sahip olan komşuların bağlanma isteklerini eşleştirmek için kullanılır.

2.2.2. WirelessHART

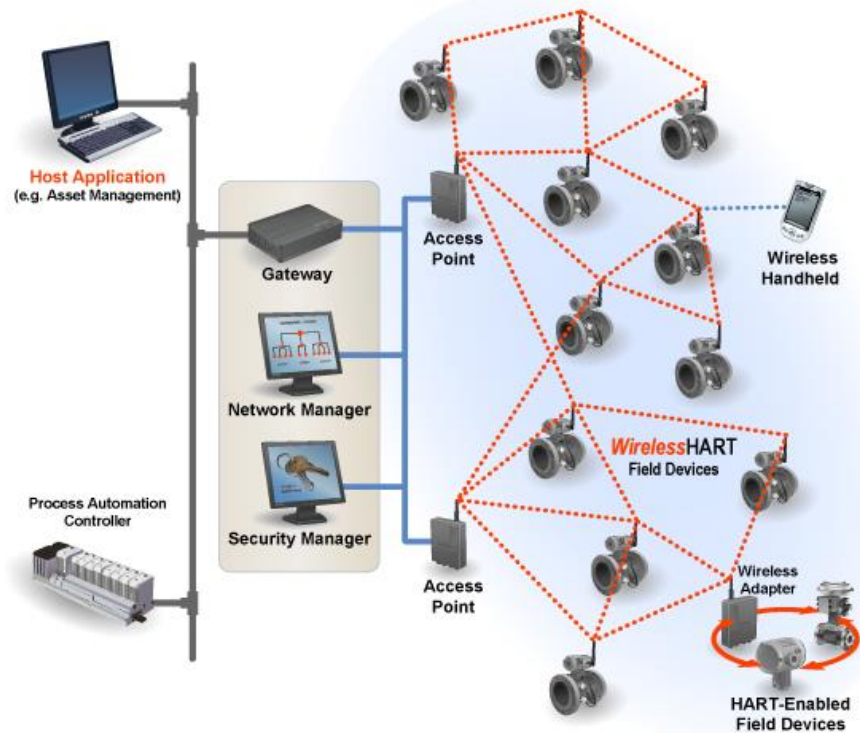
WirelessHART [91], bir endüstri standardı olan HART (Highway Addressable Remote Transducer-Karayolu Adreslenebilir Uzaktan Dönüştürücü) protokolü için kablosuz uzantı sağlamaya yönelik geliştirilmiştir. HART, bir cihaz vasıtasıyla gerçek zamanlı destek gerektiren otomasyon ve endüstriyel uygulamalarında en çok kullanılan iletişim protokolüdür [100]. Bu protokol, farklı bileşenler arasında 4-20mA analog akım döngüsü üstüne bir dijital FSK modülasyonlu sinyal bindirmeye dayanmaktadır. HART, bir ağda iki master cihazın barındırıldığı bir master/slave iletişim şeması sağlar. Buna göre; sisteme bağlı cihazlar, izleme ve kontrol amaçları için sabit bir sistem üzerinden ve el cihazları ile kontrol edilebilir [101].

WirelessHART standardı, özellikle proses ölçümüne ve kontrol uygulamalarına yönelik tasarlanmış ilk açık kablosuz iletişim standardı şeklinde ve HART-7 standardı spesifikasyonun bir parçası olarak piyasaya sürülmüştür [91]. WirelessHART, 2.4 GHz bandı için IEEE 802.15.4 fiziksel katman standardına dayanır. Ayrıca, TDMA tabanlı MAC protokolü, çeşitli mesajlaşma modlarını sağlamak amacıyla tanımlanmıştır [54]. Bunlar: proses ve kontrol değerlerinin tek yönlü yayını, istisnai spontane bildirim, tasarsız istek ve yanıt, büyük veri setlerinin otomatik segmentli blok transferleridir [20].

WirelessHART standardının ağ mimarisi Şekil 2.11'de gösterilmektedir. Bu mimariye uygun olarak beş tip bileşen tanımlanmıştır [91]: WirelessHART alan cihazları, proses veya tesis ekipmanlarına bağlı olan algılayıcı ve kontrol elemanlarıdır. Ağ geçitleri, kablolu ve kablosuz ağ altyapısına uygun arayüzler içerir. Dolayısıyla, denetleyici ve sunucu üzerinde çalışan uygulamalar WirelessHART alan cihazları ile etkileşimde bulunabilir. Ağ yöneticisi, cihazlar için iletişim slotları programlayarak, yönlendirme tabloları belirleyerek ve ağ sağlığını izleyerek ağ işlemlerini yürütür. WirelessHART bağdaştırıcıları, mevcut HART alan cihazlarını kablosuz ağ ile entegre ederek geriye dönük uyumluluk sağlar. On-board alıcı-vericilere sahip el terminalleri, WirelessHART alan cihazlarına bağlanmak ve kablosuz ağa erişmek amacıyla kullanılır [20, 100].

Yukarıda açıklanmış bileşenler temel alınarak, WirelessHART standardı için tam bir protokol yığını tanımlanmıştır [20]: Fiziksel katman, IEEE 802.15.4 standardını içerir. Veri bağlantı katmanında TDMA tabanlı MAC protokolü çalışır. Ağ topolojisi örgü ağı şeklinde tasarlanır ve her bir cihaz ağ içerisinde kaynak düğüm ya da yönlendirici gibi

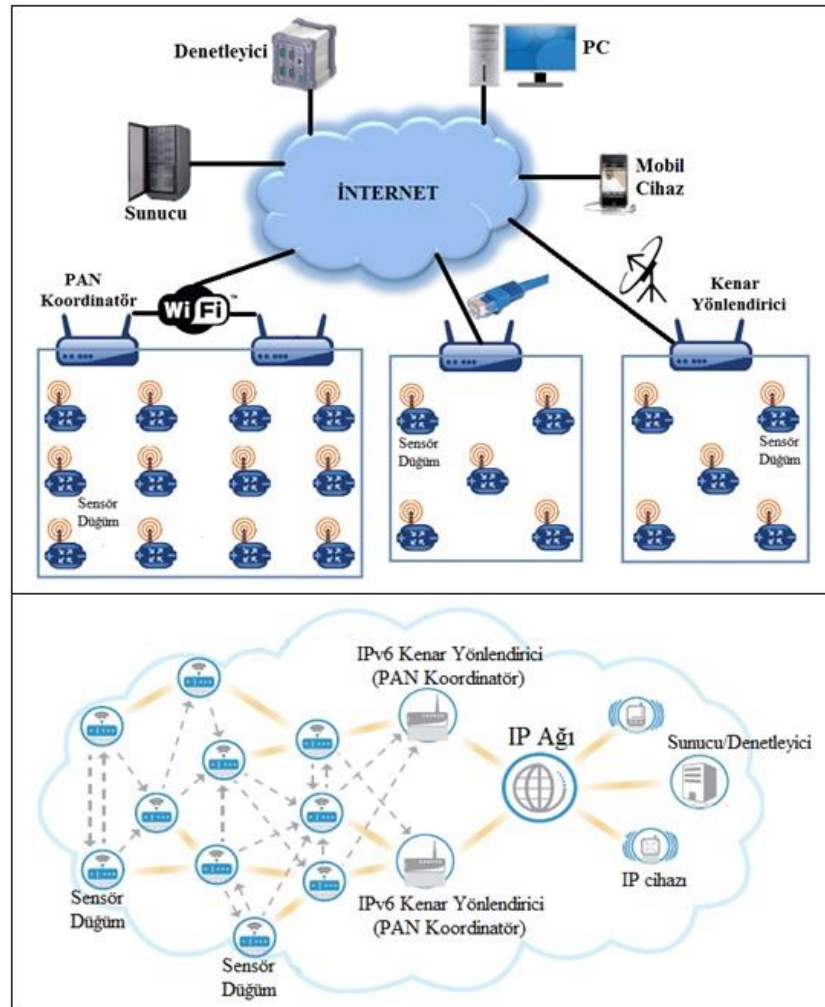
hareket edebilir. Ağ topolojisi, KAA’larda kullanılan yapılarla büyük benzerlik gösterir. Ağ katmanında tablo tabanlı yönlendirme kullanılır. Çoklu gereksiz yollar, ağ oluşumu sırasında saptanır ve sürekli doğrulanır. Dolayısıyla, WirelessHART alan cihazları ve ağ geçidi arasındaki iletişim yolu bozulsa bile alternatif yollar, ağ güvenilirliği sağlamak amacıyla hizmet verirler. Saptanmış yollara ilaveten, kaynak yönlendirme teknikleri, tasarsız iletişim yolları belirlemek için kullanılır. Ağ katmanı, belirli cihazlara ayrılmış band genişliği tahsis ederek dinamik band genişliği yönetimini destekler. Bu durum, protokol yığınının temelinde bulunan TDMA yapısı tarafından da desteklenir (örneğin, cihazlara uygun sayıda slot atayarak). Band genişliği, talebe bağlı ayrılır ve bir aygıt ağa katıldığı zaman yapılandırılabilir. WirelessHART standardının taşıma katmanı, uçtan uca yol üzerinde güvenilirlik sağlar ve büyük veri kümelerinin TCP-benzeri güvenilir blok transferlerini destekler. Ağın uçtan uca izlenmesine ve denetlenmesine imkan sunar. Dolayısıyla, WirelessHART alan cihazları, ağdaki iletişim başarılarına ve komşularına ilişkin istatistikleri sürekli olarak yayınlarlar. İstatistiki veriler, gereksiz yolları tespit etmek ve enerji verimliliğini arttırmak için ağ yöneticisi tarafından incelenir. Uygulama katmanı, mevcut çözümlerin sorunsuz bir şekilde uygulanabildiği standart HART uygulama katmanını kapsar [20, 100, 101].



Şekil 2.11: WirelessHART ağ mimarisi ve bileşenleri.

2.2.3.6LoWPAN

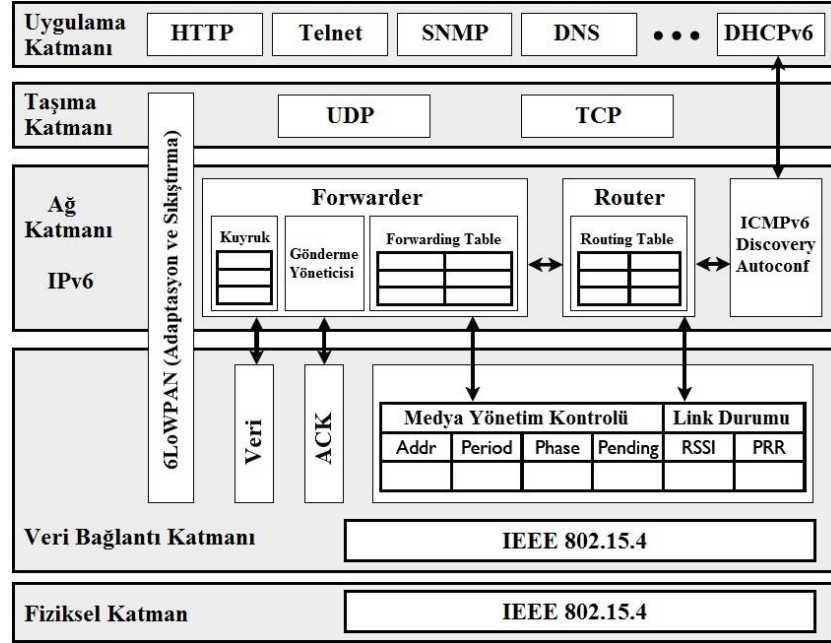
Mevcut standartlar, KAA'lara yönelik geliştirilmiş uygulamalara özel çözümler sunar. Bağımsız KAA'lar belirli uygulamalar için kullanılabilir. Ancak, IEEE 802.15.4 tabanlı protokoller IP protokolü ile uyumlu olmadığından, KAA'lar İnternete entegre edilemez. Dolayısıyla, kablosuz algılayıcı düğümler, web tabanlı cihazlar veya sunucular ile kolayca iletişim kuramazlar. KAA'lardan gelen bilgileri toplayabilmek ve İnternet ile iletişim kurabilmek için ağ geçitleri (PAN koordinatör) gereklidir (Şekil 2.12). Bu durum, ağ geçitlerinde tek arıza noktası sorunlarını yaratır ve ağ geçidinin komşularına baskı yapar [102-104].



Şekil 2.12: KAA'ların İnternet (IP ağı) ile entegrasyonu.

IETF (Internet Engineering Task Force-İnternet Mühendisliği Görev Grubu) [105], Şekil 2.12'de gösterildiği gibi KAA'ların İnternet ile entegrasyonu sağlamak için 6LoWPAN (IPv6 over Low-power Wireless Personal Area Network-Düşük güç

Kablosuz Kişisel Alan Ağı üzerinden IPv6) standardını geliştirmiştir. Bu standart, herhangi bir ağıta İnternet üzerinden erişilebilmesini sağlamak amacıyla, IEEE 802.15.4 standardı üzerine IPv6 yığınının uygulanmasını tanımlar [106] (Şekil 2.13).



Şekil 2.13: IEEE 802.15.4'ün IPv6 ile entegrasyonunda 6LoWPAN standardı.

IPv6 ve KAA'ların entegrasyonundaki temel sorun, 40 baytlık bir başlık ve adres bilgisi alanı tanımlayan IPv6'nın adresleme yapısıdır. IEEE 802.15.4 standardı, başlık ve yük bilgileri dahil olmak üzere tüm paket için 127 bayta kadar izin verir. Bu yüzden, her iki standardın basit entegrasyonu verimli değildir [107]. Bunun yerine, 6LoWPAN, IEEE 802.15.4 standardına radyo yığını ve IPv6 iletişimlerinin birlikte faaliyet göstermesini sağlayan bir adaptasyon katmanı ekler. Burada; tek parçalı başlık yerine, gönderilen paket türüne göre değerlendirilen dört tip başlığın kullanıldığı bir yığılı başlık yapısı, 6LoWPAN standardı için önerilmiştir. Ayrıca, durumsuz sıkıştırma teknikleri, başlığın boyutunu KAA'lara uygun olacak şekilde 40 bayttan yaklaşık 4 bayta indirmek için kullanılır [107, 108]. Dört başlık türü aşağıda açıklanmıştır [103]:

- **Gönderme başlığı:** Bu başlık türü, kendisini takip eden başlık türünü belirler. İlk 2 bit, gönderme başlığı için "01" olarak ayarlanır. Kalan 6 bit, kendisini takip eden başlık türünü tanımlar (sıkıştırılmamış IPv6 başlığı veya başlık sıkıştırma başlığı).
- **Örgü başlığı:** Bu başlık, ilk 2 bitte "10" ile tanımlanır. Örgü topolojilerinde yönlendirme amacıyla kullanılır. İlk 2 biti takip eden ekstra 2 bit, kaynak ve hedef

adreslerin 16-bit kısa ya da 64-bit uzun adresler olup olmadığını belirtir. Sonraki 4 bitlik sekme ayrılmış alan, kalan sekme sayısını gösterir ve başlangıçta 15 sekme destekler. Ancak, ekstra bir bayt, 255 sekme desteklemek için kullanılabilir. Son olarak, kalan alanlar, paketin kaynak ve hedef adreslerini tanımlar. Bu bilgiler, yönlendirme protokolleri tarafından bir sonraki sekmeyi bulmak için kullanılır.

- *Parçalama başlığı:* IPv6 protokolü 1280 bayta kadar veri yükü destekleyebilir. Oysaki IEEE 802.15.4 standardı 102 bayta kadar veri yükü destekler. Dolayısıyla, parçalama başlığı, daha büyük veri yüklerini birkaç pakete parçalamak ve bu paketleri tekrar birleştirmek için kullanılır. İlk parça, ilk 2 biti “11” ve sonraki 3 biti “000” ile belirtilen 4 baytlık bir başlığa sahiptir. Bunları, datagram boyutu ve datagram etiket alanları izler. Sonraki parça başlığı, ilk 5 bitinde “11100” içerir ve bu ilk 5 biti datagram boyutu, etiket ve datagram ofseti takip eder.
- *Başlık sıkıştırma başlığı:* 40 bayt IPv6 başlığı, başlık sıkıştırma başlığı içeren 2 bayta sıkıştırılır. IEEE 802.15.4 paket başlıkları, bu sıkıştırmada, kaynak ve hedef çiftlerin MAC adreslerini halihazırda içerir. MAC adresleri, IPv6 adresinin en düşük 64 bitine eşlenebilir. Nihayetinde, kaynak ve hedef adresleri IPv6 başlığından tamamen elenir. Benzer teknikler, her iletişim için gereksiz alanları ortadan kaldırmak ve bir paket İnternete erişen bir ağ geçidine ulaştığında bu alanların eklenmesine izin vermek için kullanılır.

2.3. KAA’LARDA KULLANILAN YAZILIM PLATFORMLARI

Donanım platformlarına ve standartlara ilaveten, çeşitli yazılım platformları da KAA’lara özel olarak geliştirilmiştir. Bunlar arasında, kablosuz gömülü algılayıcı ağlar için tasarlanmış açık kaynak kodlu bir işletim sistemi olan TinyOS [109], en çok kabul görmüş platformdur. TinyOS, kod boyutunu en aza indiren ve yeni iletişim protokollerinin uygulanabilmesi için esnek bir platform sağlayan bileşen tabanlı bir mimari içerir. TinyOS’un bileşen kütüphanesinde; özel uygulama gereksinimlerine bağlı olarak değiştirilebilir veya geliştirilebilir ağ protokolleri, dağıtık servisler, algılayıcı sürücüler ve veri toplama araçları yer alır. TinyOS, ince ayarlı güç yönetim stratejileri sağlayan bir olay güdümlü yürütme modeline dayanır [23, 24].

Günümüzde, iletişim protokollerine yönelik mevcut yazılım kodlarının çoğu TinyOS platformuna uygun yazılmıştır. TinyOS’la birlikte gelen ve bir TinyOS algılayıcı düğüm simülatörü olan TOSSIM [110], KAA protokolleri ve uygulamalarının geliştirilmesini kolaylaştırmak amacıyla lanse edilmiştir. TOSSIM, bir ölçeklenebilir simülasyon ortamı sağlar ve doğrudan TinyOS kodundan derleme yapar. Üst düzey uygulama sistemlerine ilaveten alt düzey protokollerle de deney yapılmasına izin vererek, TinyOS ağ yığını bit seviyesinde simule eder. Ayrıca, çalışan simülasyonlarla etkileşimde bulunmak ve bunları görselleştirmek için bir grafik kullanıcı arabirimi aracı TinyViz’i içerir [24].

TinyOS’a ek olarak, günümüzde çeşitli yazılım platformları ve işletim sistemleri de mevcuttur. LiteOS [24, 111, 112], Unix benzeri soyutlamalar sağlayan çoklu iş parçacığı işletim sistemidir. TinyOS ile karşılaştırıldığında, LiteOS; çoklu iş parçacıklı çalışma, dinamik bellek yönetimi ve komut satırı kabuk desteği (LiteShell) özellikleri sunar. LiteShell, kullanıcı tarafında bir komut satırı arabirimi bulundurur. Örneğin; programlanacak algılayıcı düğümle etkileşimde olmayı sağlamak için bir bilgisayar. Contiki [24, 113, 114]; Telos, Tmote ve Mika algılayıcı düğüm ailelerinde kullanılan TI MSP430 ve Atmel AVR gibi mikroişlemcileri içeren çeşitli platformlarda kullanılmak üzere geliştirilmiş açık kaynak, çoklu görev işletim sistemidir. Contiki, olay güdümlü çekirdek etrafına inşa edilmiştir. Bireysel programların ve servislerin dinamik yükleme ve değiştirilmesinin yanısıra belirli programlar için önleyici çoklu kullanım çalıştırmak mümkündür. Derleme zamanında statik olarak bağlantılı TinyOS ile karşılaştırıldığında, Contiki, program ve sürücülerin çalışma süresince ve yeniden bağlama olmaksızın değiştirilebilmesine imkan verir. Ayrıca, μ IP yığını aracılığıyla TCP/IP desteği sağlar. Sun SPOT [115] platformu bir işletim sistemi kullanmaz; fakat, tam kapasiteli Java ME uygulaması olan Squawk java sanal makinesini makine düzeyinde çalıştırır. Sanal makine, işlemleri doğrudan flaş bellek üzerinden yürütür.

Ek özelliklere sahip çeşitli işletim sistemleri KAA araştırmalarında kullanılmasına rağmen TinyOS hâlâ en yaygın kullanılan yazılım platformudur. Bu popülerliğin başlıca nedenlerinden biri, KAA çözümlerinin geliştirilmesi süresince yapılandırılmış büyük kod alanıdır. Mevcut KAA uygulamalarını ve iletişim protokollerini bu yeni işletim sistemlerine bağlamak zordur. Bu durum, var olan kod alanı için birlikte çalışabilirliği destekleyen esnek platformlar aranmasına sebep olmuştur [20, 23, 24].

2.4. KAA TASARIMINI ETKİLEYEN FAKTÖRLER

KAA'ların tasarımı; kablosuz iletişim, ağ oluşturma, gömülü sistemler, dijital sinyal işleme ve yazılım mühendisliği gibi çeşitli araştırma alanlarında çok geniş bilgi gerektirir. Araştırmalar, kablosuz algılayıcı cihazların donanım ve yazılım öğelerinin yanı sıra bu cihazların oluşturduğu bir ağın dağıtık operasyonu ile ilişkilidir. Dolayısıyla, KAA'ların tasarımını önemli ölçüde etkileyen çeşitli faktörler bulunmaktadır. Etken ana faktörler arasında; donanım kısıtlamaları, üretim maliyetleri, algılayıcı ağ topolojisi, hata toleransı, ölçeklenebilirlik, iletim ortamı ve güç tüketimi vardır.

2.4.1. Donanım Kısıtlamaları

Bir kablosuz algılayıcı cihaz genellikle dört temel bileşenden oluşur: *algılama birimi*, *işlem birimi*, *alıcı-verici birim* ve *güç birimidir*. Ayrıca, *konum bulma sistemi (GPS)*, *güç üretici (güneş paneli)* ve *harekete geçirici* gibi ek bileşenler de uygulamaya bağlı olarak algılayıcı cihaza entegre edilebilir.

- *Algılama birimi*: İletişim becerilerine sahip olması nedeniyle kendisini diğer gömülü sistemlerden farklı kılan bir kablosuz algılayıcı cihazın ana bileşenidir. Genellikle, fiziksel dünyadan bilgi toplama yetisi sağlayan çeşitli algılayıcıları içerir. Her bir algılayıcı; sıcaklık, nem, basınç, ışık gibi belirli türde veri toplama aktivitesinden sorumlu olup genellikle iki alt birimden oluşmaktadır. Bunlar, algılayıcı ve analog-dijital dönüştürücüdür (ADC). Gözlemlenen olaya bağlı olarak algılayıcı tarafından üretilen analog sinyaller, ADC aracılığıyla dijital sinyallere dönüştürülür ve daha sonra işlem birimine aktarılır.
- *İşlem birimi*: Kablosuz algılayıcı cihazın ana denetleyicisidir. Algılayıcı cihazı meydana getiren tüm bileşenlerin yönetilmesini sağlar. İşlem birimi, bir on-board bellekten oluşabilir veya gömülü karta entegre küçük bir depolama ünitesi ile ilişkilendirilmiş olabilir. Algılayıcı cihaza; algılama işlemlerini gerçekleştirmesi, algılanan verilere dayalı basit hesaplamaları yapması, ilgili algoritmaları ve protokolleri çalıştırması, kablosuz iletişim yoluyla diğer algılayıcı düğümlerle işbirliği yapması gibi yetenekler sağlayan prosedürleri yönetir.
- *Alıcı-verici birim*: Herhangi iki kablosuz algılayıcı cihaz arasındaki iletişim alıcı-verici birimler ile gerçekleştirilir. Bir alıcı-verici birim, bitleri radyo frekansı

dalgaları içinde iletilecek duruma dönüştürmek ve diğer uçta onları eski haline çevirmek için gerekli işlemleri uygular. KAA, bu birim aracılığıyla ağa bağlanır.

- *Güç birimi:* Kablosuz algılayıcı cihazın en önemli bileşenlerinden biri de güç birimidir. Genellikle pil kullanılır; ancak, farklı enerji kaynaklarını kullanmak da mümkündür. Algılayıcı cihazı oluşturan her bir bileşen güç biriminden enerji sağlar. Güç birimi sınırlı kapasiteye sahip olduğundan, algılayıcı cihazın bileşenleri tarafından gerçekleştirilen görevler için enerji tasarruflu işlemlere gerek duyulur.
- *Konum bulma sistemi:* KAA uygulamaları, algılama işlemleri ve yönlendirme tekniklerinin çoğu, algılayıcı düğümün fiziksel konumunu içeren bilgiye ihtiyaç duyar. Bu nedenle, bir algılayıcı düğümün konum bulma sistemi ile donatılması yaygındır. Konum bulma sistemi, gelişmiş özelliklere sahip üst düzey bir algılayıcı düğüm için GPS modülünden oluşabilir veya dağıtık hesaplamalar yoluyla konum bilgisi sağlayan lokalizasyon algoritmalarını gerçekleştiren bir yazılım olabilir.
- *Harekete geçirici:* Tahsis edilmiş işlemleri gerçekleştirmek gerektiğinde algılayıcı düğümleri hareket ettirmesi için bir harekete geçiriciye ihtiyaç duyulabilir. Mobilite desteği büyük enerji kaynakları gerektirir ve verimli bir şekilde sağlanmalıdır. Harekete geçirici, algılayıcı düğümün hareketlerini kontrol etmek için algılama ünitesi ve işlemci ile yakın etkileşim içinde çalışır.
- *Güç Üretici:* Bir algılayıcı cihaz, enerjisini genellikle pilden sağlar. İlave bir güç üretici, daha uzun ağ ömrünün gerekli olduğu uygulamalarda tercih edilebilir. Dış mekân uygulamalarında, güneş panelleri güç üretmek amacıyla kullanılır. Benzer şekilde; termal, kinetik ve titreşim enerjilerine yönelik enerji elde etme teknikleri de kullanılabilir [116].

Yukarıda tanımlanmış bileşenler, aynı zamanda bir kibrit kutusu boyutundaki gömülü sisteme sığmalıdır [117]. Belirli uygulamalara özgü geliştirilmiş algılayıcı düğümler, bir santimetre küpten bile daha küçük boyutlarda ve havada asılı kalacak kadar hafif ağırlıklarda olabilirler [118]. Bu gelişmiş özellikler ve boyut gereksinimleri, kablosuz algılayıcı düğüm tasarımlarına ek zorluklar getirmektedir. Boyuta ilaveten, algılayıcı düğümler için başka kısıtlamalar da vardır. Sonuç olarak, bu cihazlar [119]: son derece düşük güç tüketmeli; yüksek yoğunlukta, otonom ve gözetimsiz çalışabilmeli; çevreye karşı uyartılabilmeli; düşük üretim maliyetine sahip ve vazgeçilebilir olmalıdır.

KAA işlemlerinde temel kaygı enerji tüketimidir. Uygulamaların çoğunda, algılayıcı düğümlerin pillerini yenilemek ya da tekrar şarj etmek zordur. Pil gücü ile birlikte, ağın işlevsel durumda kaldığı maksimum süre olan ağ ömrü de sınırlıdır. Güç, algılayıcı düğümlerin boyut ve maliyet kısıtlamaları nedeniyle KAA'larda kıt bir kaynaktır. Örneğin: Smart Dust [120] algılayıcı düğümün kapasitesi, tek beş-boyutlu işitme cihazı pilinden güç sağladığından, 33mAh değerindedir [118]. İki adet AA sınıfı kalem pil bulunduran MicaZ ve Mica2 algılayıcı düğümlerin kapasitesi 1.400-3.400mAh arasında sınırlıdır [121]. Benzer şekilde, Sun SPOT platformu 750mAh lityum iyon pil içerir. Çevreden enerji çıkarma (energy scavenging) teknikleri kullanılarak KAA'ların ömrünü uzatmak mümkündür [116, 122]. Güneş panelleri, yenilenebilir enerji kaynağı güneşten enerji elde etmek için kullanılan, pratik ve düşük maliyetli bir örnektir. Ancak, KAA'lar için sağlanabilecek enerji miktarı hâlâ kısıtlıdır. Bu nedenle, bir KAA'ın aylar hatta yıllar boyunca çalışması gerekebileceğinden, enerji tasarruflu operasyon, KAA tasarımıyla çok en önemli bir faktördür.

Yukarıda açıklanan bileşenler arasında, bir kablosuz algılayıcı düğümün en önemli parçası alıcı-verici birimdir. Çünkü, alıcı-verici modül, hem çok enerji tüketir hem de ağdaki diğer cihazlara bağlantı sağlar. Alıcı-verici; Smart Dust düğümlerde olduğu gibi aktif veya pasif bir optik aygıt, ya da Mica ailesi, TelosB [121] ve Sun SPOT içinde bulunan bir RF cihazı olabilir. RF iletişimleri; modülasyon, band geçiren filtreleme, demodülasyon ve bunları enerji tüketiminin yanısıra daha karmaşık ve pahalı hale getiren çoğullama devresi gerektirir [20, 21]. Ayrıca, RF dalgaları, iki düğüm arasındaki mesafenin dördüncü dereceden üssüyle orantılı yüksek yol kaybına maruz kalır [118, 123]. Dolayısıyla, düşük güç RF alıcı-vericinin iletişim menzili on-yüz metrelerle sınırlıdır. Bunlara rağmen, RF iletişimi, çoğu algılayıcı düğüm prototipinde fiili standart olarak tercih edilir. Bu; küçük paketlerin düşük veri hızlı RF sinyalleri aracılığıyla verimli bir şekilde gönderilebilmesi ve frekansı yeniden kullanmak için sınırlı iletim menzillerinin göz ardı edilebilmesi gerçeğine dayanır. Ayrıca, mevcut algılayıcı düğüm alıcı-vericilerinin düşük başlatma ve kapatma gecikmeleri, düşük hizmet çevrimine sahip işlemlerin uygulanabildiği iletişim protokollerine kapsamlı yetenekler sağlar [21]. Diğer bir ifadeyle, bir algılayıcı düğümün alıcı-verici modülü çoğu zaman kapalı tutulabilir ve sadece gönderilecek ya da alınacak bir paket olduğu zaman açılabilir.

Alıcı-vericiler için en önemli faktörlerden biri de teknik olarak hâlâ zor olan düşük maliyetli, enerji tasarruflu ve düşük hizmet çevrimli radyo devrelerinin tasarımıdır.

Alıcı-verici birime ek olarak, algılayıcı düğüm işlem ve bellek açısından da sınırlıdır. Yüksek hesaplama gücü, daha küçük işlemcilerle düşük maliyete sağlanmasına rağmen, mevcut algılayıcı düğümlerin işlem güçleri, maliyet ve boyut sınırlamaları nedeniyle diğer pek çok gömülü sistemden önemli ölçüde düşüktür. Örneğin: Smart Dust düğüm prototipi gibi daha eski platformlar; 8kB flaş komut belleği, 512 bayt SRAM veri belleği, 512 bayt EEPROM veri belleği ve 4MHz Atmel-AVR-8535 mikroişlemciye sahiptir [124]. Bu özellikler, Sun SPOT ve Imote2 [125] platformları ile artırılmıştır. Sun SPOT; 512kB RAM, 4MB flaş ve 180MHz 32-bit ARM920T çekirdek işlemciye sahipken, daha üst düzey bir platform olan Imote2; 256kB SRAM, 32MB flaş, 32MB SDRAM ve 13-416MHz Marvell-PXA271-XScale mikrodenetleyici ile donatılmıştır. Algılayıcı düğümlerin donanım özellikleri artsa da, bu değerler, PDA ya da cep telefonu gibi gömülü cihazların mevcut yeteneklerinin hala çok altındadır [20, 21]. Sonuç olarak; KAA'lara yönelik geliştirilen yazılımlar sistemi yormamalı ve algoritmaların hesaplama gereksinimleri KAA'lardaki etkin operasyonlar için düşük olmalıdır.

KAA'lar, çeşitli fiziksel olaylar hakkında veri toplamak için çevreleri ile yakın etkileşim içindedirler. Çoğu algılama işlemi, pozisyon bilgisine gerek duyar. Algılayıcı düğümler genellikle rastgele dağıtıldıklarından ve gözetimsiz çalıştıklarından dolayı, bir konum bulma sistemi gereklidir. Ayrıca, KAA'lar için geliştirilen birçok yönlendirme protokolleri konum bulma sistemlerine ihtiyaç duyar [25]. Konum bilgileri, KAA'lar için geliştirilen GPS üniteleri aracılığıyla 10 metreye kadar doğruluk sağlayan GPS tarafından kolayca sağlanabilir. Ancak, bu birimlerin maliyetleri tek bir algılayıcı düğümden daha yüksektir ve dolayısıyla, tüm algılayıcı düğümleri GPS ile donatmak KAA'larda uygulanabilir değildir. Bunun yerine, konumlarını tanımlamada GPS veya başka teknikler kullanan az sayıda algılayıcı düğüm, diğer düğümlerin konumlarını belirlemelerine yardım etmek amacıyla KAA içerisinde görev yapabilir [24, 25].

2.4.2. Hata Toleransı

Donanım kısıtlamaları, algılayıcı düğümlerin sık sık bozulmasına ya da bir süreliğine bloke olmasına sebebiyet verebilir. Algılayıcı düğüm arızaları; güç yetersizliği, fiziksel hasar, çevresel müdahaleler, parazitlenme, yazılım sorunları gibi nedenlerden meydana

gelebilir. KAA'daki bir düğümün bozulması, onun ağ ile bağlantısının kesilmesine yol açar. KAA, tek bir algılayıcı düğümden gelen veriden ziyade fiziksel olaya ait bilgi ile ilgilendiğinden, tek bir düğümün arızalanması KAA'nın çalışmasını etkilememelidir. Bir ağın işlevlerini yeterli düzeyde devam ettirebilmesi için ağ tarafından izin verilen arıza seviyesi hata toleransını tanımlar. Hata toleransı, algılayıcı düğüm arızalarına bağlı herhangi bir kesinti dışında KAA'nın fonksiyonlarını sürdürebilme becerisidir [126-128]. Bir algılayıcı düğümün donanım ve yazılım bileşenleri arıza oranını ilk önce etkiler. Algılayıcı düğümler düşük maliyetli cihazlarla tümleşik olduğundan, arızaların çoğunluğu donanım problemlerinden kaynaklanır. Bundan başka, algılayıcı düğüm üzerinde çalışan yazılım, düğümün sınırlı bellek alanı ve işlem yetenekleri sebebiyle durabilir. Dahili sorunlara ek olarak, konuşlandırma ortamı da bir algılayıcı düğümün nasıl çalıştığını etkileyebilir. İç mekân uygulamaları, algılayıcı düğümlerin daha az parazitlenmesine neden olur ve arıza oranını arttırmayabilir. Öte yandan, algılayıcı düğümlerin açık alanlarda konuşlandırıldığı uygulamalarda düğüm arızaları, çevresel müdahaleler sebebiyle meydana gelebilir [20, 21, 23].

KAA'lar için tasarlanmış algoritmalar ve protokoller, sık görülen algılayıcı düğüm arızalarını artıklık yoluyla çözmeyi hedefler. Bir ağın hata toleransı; bir düğümün yayın aralığı içindeki tek bir düğümden daha fazlasına dayanarak iyileştirilebilir. Sonuçta, bir algılayıcı düğüm bozulsa bile, yayın aralığındaki diğer düğümler ağa bağlanırlık sağlamak amacıyla kullanılabilirler.

Bir ağın hata toleransı, uygulamanın ne amaçla yapıldığına da bağlıdır. Algılayıcı düğümlerin konuşlandırıldığı ortamda az parazit varsa, protokoller daha esnek olabilir. Örneğin: Eğer algılayıcı düğümler sıcaklık ve nem seviyelerini izlemek amacıyla iç mekanlarda konuşlandırılmışsa, bu tür bir algılayıcı ağ kolayca zarar görmeyeceğinden veya çevresel gürültüden/parazitten etkilenmeyeceğinden hata toleransı gereksinimi düşüktür. Öte yandan, algılayıcı düğümler gözetim, denetleme, tarama, izleme ve tespit gibi işlemler için açık arazide konuşlandırılmışsa, algılanan veriler kritik öneme sahip olacağından ve algılayıcı düğümler çevredeki hareketli nesne eylemleri ile tahrip/imha edilebileceğinden hata toleransı gereksinimi yüksektir [23]. Sonuç olarak, hata toleransı seviyesi KAA uygulamalarına bağlı olup, protokoller ya da algoritmalar uygulamalara özgü geliştirilmelidir.

2.4.3. Ölçeklenebilirlik

Bir KAA'daki algılayıcı düğümlerin yüksek yoğunlukta konuşlandırılmaları artıklık sağlar ve ağına hata toleransını artırır iken, aynı zamanda ölçeklenebilirlik zorluklarını da yaratır. Fiziksel olayı algılamak amacıyla konuşlandırılmış algılayıcı düğümler yüzlerce veya binlerce sayıda olabilir. Bu nedenle, KAA'lar için geliştirilen ağ protokolleri, çok sayıda düğümü verimli bir şekilde yönetebilmelidir [20]. Yoğunluk, çapı 10 metreden az olan bir bölgede bulunan algılayıcı düğüm sayısını ifade eder [129] ve algılayıcı düğümlerin konuşlandırıldığı uygulamaya bağlıdır.

2.4.4. KAA Topolojisi

Sık sık arıza yapmaya meyilli, erişilmez bölgelerde bulunan ve gözetimsiz çalışan çok sayıda algılayıcı düğüm, topoloji bakımını zora sokar. İlgilenilen olayın verimli bir şekilde izlenebilmesi için algılayıcı düğümlerin alana konuşlandırılması önemli bir sorundur. Bu durum, konuşlandırma öncesi ve konuşlandırma aşamalarını teşkil eder. Topoloji bakımı, algılayıcı düğümlerin ilk konuşlandırılmalarından sonra da önemlidir. Örneğin; protokol parametrelerinin ve işlemlerin ağ topolojisine göre uygulanabileceği konuşlandırma sonrası aşamalar [20]. Birkaç düğümün arızalanması veya enerjilerinin tükenmesi durumunda ağ ömrünü uzatmak amacıyla yeniden konuşlandırma gerekebilir. Sonuç olarak, çok sayıda algılayıcı düğümün yoğun bir şekilde konuşlandırılması topoloji bakımının dikkatli yapılmasını gerektirir [130, 131].

2.4.4.1. Konuşlandırma Öncesi ve Konuşlandırma Aşamaları

Algılayıcı düğümler, izlenecek alana topluca atılabilir veya teker teker yerleştirilebilir. Örneğin: bir uçaktan bırakılarak; roket veya füzeyle fırlatılarak; insan ya da robot tarafından tek tek yerleştirilerek. Çok sayıda algılayıcı düğüm ve bunların gözetimsiz konuşlandırılması, topoloji tasarımlarında zorluklara yol açar. Bu nedenle, ilk/başlangıç konuşlandırma planları: kurulum maliyetini azaltmalı; herhangi bir ön-organizasyon ve ön-planlama için gerekli ihtiyacı ortadan kaldırmalı; düzenleme esnekliğini arttırmalı; özörgütlenme (kendi kendini organize etme) ve hata toleransını desteklemelidir [132].

2.4.4.2. Konuşlandırma Sonrası Aşama

KAA topolojisi, algılayıcı düğümleri konuşlandırma aşamasından sonra düğümlerin durumlarında meydana gelebilecek değişiklikler nedeniyle değişebilir [117, 133]. Mobil

KAA’larda, algılayıcı düğümlerin hareketleri ağ topolojisini etkiler (topolojide uzun süre önemli değişiklikler görülebilir). Ayrıca, düğümlerin bağlanabilirliği; jamming (yayın bozma), parazit, gürültü, müdahale veya hareketli engeller sebebiyle değişebilir. Bu faktörler de ağ topolojisini kısa süreliğine etkiler. Konuşlandırma işleminden sonraki topoloji değişikliklerinin bir başka nedeni, kalıcı değişikliklere yol açan düğüm arızalarıdır. Bunlara ilaveten, KAA içerisindeki bazı düğümlerin belirli bir süre kapalı tutulması gerektiğinde, ağ topolojisi, uygulamaya ve algılama görevlerine göre periyodik olarak değişebilir [134-136].

Yukarıda bahsedilen değişiklikler, ağın ilk konuşlandırılmasından farklı bir operasyonla sonuçlanır. Dolayısıyla, ağ protokolleri, KAA topolojilerinde ortaya çıkabilecek bu periyodik, kısa ve uzun vadeli değişimlere uyum sağlayabilmelidir [132].

2.4.4.3. Ek Düğümlerin Yeniden Konuşlandırılması Aşaması

Konuşlandırma sonrası aşamada yapılan değişiklikler, eğer ağın bağlanabilirliği ve hata toleransı topolojideki değişikliklerden ciddi bir şekilde etkileniyorsa, ilave düğümlerin konuşlandırılmasını gerektirebilir. Bu nedenle, ek düğümler, arızalı/bozuk düğümlerin yerlerini almaları amacıyla veya görev dinamiklerindeki değişiklikler nedeniyle herhangi bir zamanda yeniden konuşlandırılabilir. Yeni düğümlerin eklenmesi, KAA’ı yeniden düzenleme (organize etme) ihtiyacı doğurur. Çok büyük sayıda düğüme ve çok sıkı güç tüketimi kısıtlamalarına sahip bir tasarsız ağda, sık tekrarlayan topoloji değişiklikleriyle başa çıkabilmek için özel yönlendirme protokolleri gereklidir [20,132].

2.4.5. İletim Ortamı

Bir KAA’nın başarılı operasyonu, ağdaki düğümler arasında güvenilir bir iletişime dayanır. Çok sekmeli bir KAA’da, algılayıcı düğümler, kablolu bir modül aracılığıyla birbirleri arasında bağlantılar oluşturarak haberleşir. İletişim; *radio frekansı*, *kızılötesi*, *optik*, *akustik* veya *manyetik-endüktif* bağlantılar ile kurulabilir. Seçilen iletim ortamı, KAA’ların global operasyonunu ve birlikte çalışabilirliğini sağlamak için dünya çapında mevcut olmalıdır [20, 132].

RF bağlantılar için popüler seçenek, pek çok ülkede lisans-ücretsiz iletişim sunan ISM bandlarının kullanılmasıdır. ITU (International Telecommunication Union-Uluslararası Telekomünikasyon Birliği) telsiz yönetmeliği S5 maddesinde yer alan Uluslararası

Frekans Tahsisi Tablosu, ISM uygulamaları için kullanıma sunulan frekans bandlarını belirler [137]. Bu frekans bandlarının bazıları, kablosuz telefon sistemlerinde ve kablosuz yerel alan ağlarında haberleşme için halihazırda kullanılmaktadır. KAA'lar için küçük boyutlu, düşük maliyetli, ultra-düşük güçte alıcı-verici birimler gereklidir. Donanım kısıtlamaları, anten verimliliği ile güç tüketimi arasındaki ödünleşme, bu tür alıcı-vericilerden ultra yüksek frekansa (UHF) olan taşıyıcı frekans seçimini sınırlar [138]. Neticede, eski algılayıcı düğümler Avrupa'da 433MHz ve Kuzey Amerika'da 915MHz ISM frekans bandlarını desteklerken, yeni çıkan algılayıcı düğümler, IEEE 802.15.4 standardı tarafından da desteklenen 2.4GHz ISM frekans bandını kullanır.

ISM frekans bandlarını kullanmanın ana avantajları; ücretsiz radyo, büyük spektrum tahsisi ve global geçerliliğidir [20]. ISM bandında iletişim, KAA'larda enerji tasarruflu protokollerin uygulanmasına yönelik daha fazla serbestliğin sağlanması nedeniyle belirli bir standarda bağlı değildir. Diğer yandan, mevcut uygulamalardan ötürü güç sınırlamaları ve zararlı parazitler gibi çeşitli kurallar ve kısıtlamalar da vardır. Ayrıca, ISM frekans bandları belirli tipteki kullanıcılara tahsis edilmiş ya da ayarlanmış olmadıklarından, herhangi bir kablosuz ağ tarafından kullanılabilirler. Bu durum, bu spektrum bandında genellikle düşük güç iletişim teknikleri kullanan KAA'lara karşı parazit olasılığını artırır [132].

Algılayıcı düğümler için geliştirilmiş güncel donanımların çoğu, RF devre tasarımına dayanmaktadır. Örneğin: daha eski bir düğüm olan μ AMPS kablosuz algılayıcı düğüm; entegre frekans sentezleyicisine sahip, Bluetooth uyumlu 2.4GHz alıcı-verici içerir [139]. MicaZ, TelosB, Shimmer [140], Imote2, XYZ [141], Sun SPOT gibi algılayıcı düğümlerin çoğu; IEEE 802.15.4 ve ZigBee standartlarına uyumlu, DSSS (Direct Sequence Spread Spectrum-Doğrudan Sıralı Yaygın Spektrum) tekniğini destekleyen, 2.4GHz frekans bandında çalışan TI-CC2420 alıcı-verici çipini kullanır. IEEE 802.15.4 standardının dünya çapında kabulü, KAA'lar tarafından kullanılan iletim ortamının da standart hale geldiğini göstermektedir. Bu durum, algılayıcı düğüm tasarımına, birlikte çalışabilirlik ve geliştirme kolaylığı açısından kapsamlı yetenekler sağlar [20].

KAA'larda, düğümler arası iletişimin bir başka muhtemel yöntemi de kızılötesidir. Kızılötesi iletişim, lisans ücretsizdir ve elektrikli cihazlar nedeniyle oluşan parazitlere dayanıklıdır. Ayrıca, kızılötesi tabanlı alıcı-vericilerin üretimleri daha ucuz ve kolaydır.

Günümüzde çok sayıda laptop, PDA, cep telefonu gibi cihazlar kızıllötesi arabirimi sunmaktadır. Öte yandan, kızıllötesinin en büyük dezavantajı, gönderici ve alıcı arasında bir görüş hattı gereksinimidir. Bu durum, kızıllötesini KAA uygulamalarında iletim ortamı için tercih edilmeyen bir seçenek haline getirir. Buna rağmen, kızıllötesi, RF sinyallerinin çok yüksek oranda zayıfladığı sert ortamlarda kullanılabilir (örneğin: sualtı KAA’larda algılayıcı düğüm bağlantıları için) [20].

Algılayıcı ağların olağandışı uygulama gereksinimleri, iletim ortamı seçimini daha zor hale getirmektedir. Örneğin; denizcilik uygulamaları sulu iletim ortamı kullanılmasını gerektirebilir [142]. Burada, su yüzeyine nüfuz edebilen uzun dalga boyu radyasyonu kullanmak akıllıca olur. Sualtı algılayıcı ağ uygulamaları için, bu ortamlarda yüksek oranda zayıflamaya maruz kalan RF dalgaları yerine akustik iletişim teknikleri kullanılır [143]. Yaşanması güç arazilere veya muharebe meydanlarına yönelik uygulamalar, hata eğilimli kanallarla ve daha büyük parazitlerle karşılaşabilir. Ayrıca, bir algılayıcı düğümün anteni, diğer kablosuz cihazlarda bulunan antenlerdeki irtifa ve radyasyon gücüne sahip olmayabilir. Dolayısıyla, iletim ortamının seçimi, sağlam kodlama ve çok farklı kanal karakteristiklerini verimli bir şekilde modelleyen modülasyon şemaları tarafından desteklenmelidir [20].

2.4.6. Güç Tüketimi

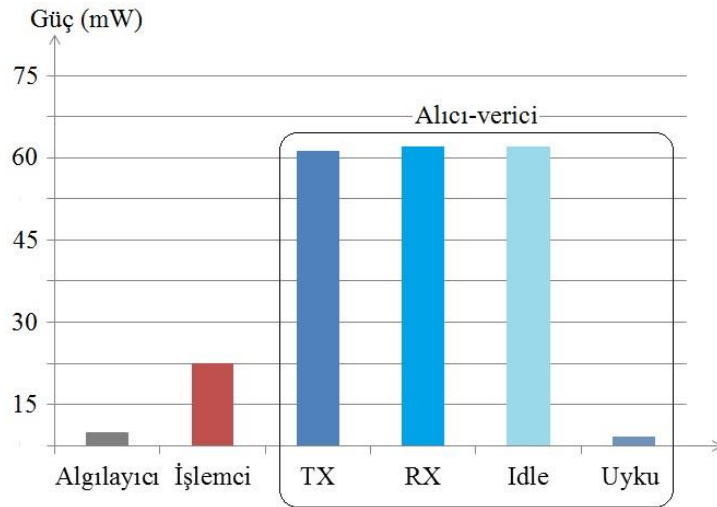
Bir kablosuz algılayıcı düğüm, çeşitli donanım kısıtlamaları nedeniyle sınırlı kapasitede bir güç kaynağı ile donatılır. Pek çok uygulama için, düğümlerin güç kaynaklarının değiştirilebilmesi ya da tekrar şarj edilebilmesi imkansızdır. Bu nedenle, KAA ömrü pil ömrüne büyük oranda bağımlılık gösterir. Sonuç olarak, her bir düğümün çalışması esnasında enerji tüketen kaynaklar iyi analiz edilmeli ve verimli biçimde işletilmelidir.

Çok sekmeli tasarsız algılayıcı ağda görev yapan algılayıcı düğümler, farklı ve birbirini tamamlayıcı iki role sahiptir [20]:

- *Veri yaratıcı:* Her bir algılayıcı düğümün ana rolü, çeşitli algılayıcılar aracılığıyla ortamdan veri toplamaktır. Çevreyi algılamaktan elde edilen verilerin işlenmesi ve çok-sekmeli iletişim yoluyla alıcıya teslim edilebilmesi için, bu verilerin yakındaki algılayıcı düğümlere iletilmesi gerekir.

- *Veri yönlendirici*: Veri oluşturmaya ek olarak, her bir algılayıcı düğüm, komşuları tarafından kendisine gönderilen bilgileri aktarmakla sorumludur. KAA'lardaki düşük güç iletişim teknikleri bir düğümün haberleşme mesafesini kısıtlar. Büyük bir ağda, düğümlerin bilgileri alıcıya (veri toplama merkezi) gönderebilmeleri için çok sekmeli iletişim gereklidir. Bu nedenle, algılayıcı düğüm, komşuları tarafından gönderilen verileri alma ve bu verileri yönlendirme kararlarına göre komşularından birine iletmekle sorumludur.

İzlenecek/gözlemlenecek ortama konuşlandırılmış bir algılayıcı düğümün ana görevleri; fiziksel olayları algılamak, yerel veri işlemeyi gerçekleştirmek ve sonra bu veriyi iletmektir. Dolayısıyla, bir algılayıcı düğümde güç tüketimi üç etki alanına ayrılabilir: *algılama*, *veri işleme* ve *iletişim*. Bu işlemler de sırasıyla algılayıcılar, mikroişlemci ve RF alıcı-verici tarafından gerçekleştirilir. Örneğin; bir MicaZ algılayıcı düğümün güç tüketim analizi Şekil 2.14'de gösterilmiştir. Üç etki alanı arasında, algılayıcı düğümün en fazla güç harcadığı durum veri iletişimidir [20].



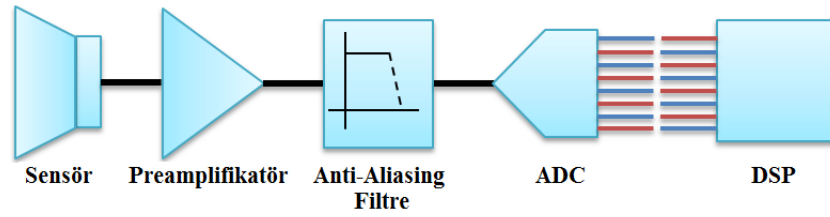
Şekil 2.14: MicaZ algılayıcı düğümün güç tüketimi.

Her bir rol ile ilgili işlemler, bir algılayıcı düğümde enerjinin nasıl tüketildiğini etkiler. Ayrıca, birkaç düğümün arızalanması; önemli topolojik değişikliklere neden olur, paketlerin tekrar yönlendirilmesini ve ağın yeniden düzenlenmesini gerektirir. Bu nedenle, güç tasarrufu ve güç yönetimi, KAA'larda kullanılan herhangi bir iletişim protokolünün ayrılmaz iki parçasıdır. Sonuç olarak, güç duyarlı algoritmaların ve protokollerin tasarımı KAA'lar için büyük önem taşır [132].

2.4.6.1. Algılama

Algılamada harcanan güç, uygulamaların yapısına ve kullanılan özel algılayıcılara göre değişir. Aralıklı algılama, sürekli olay izlemeye göre daha az güç tüketebilir. Olayları algılamanın karmaşıklığı da enerji tüketiminin belirlenmesinde önemli bir rol oynar. Daha yüksek ortam gürültü seviyeleri, ciddi bozulmalara sebep olabilir ve algılama karmaşıklığını arttırabilir.

Algılama işlemindeki enerji tüketimi, kullanılan algılayıcı türüne göre önemli derecede değişiklik gösterir. Algılayıcı sistem, genellikle bir ADC alt sistemi ile ilişkilendirilir. Şekil 2.15’de blok diyagramı verilen bu ADC alt sistemi; algılayıcı (sensör), düşük gürültülü preamplifikatör, anti-aliasing filtre, ADC ve DSP bileşenlerini içerir. ADC alt sisteminin enerji tüketimi, bu bileşenlerin her birinin performansına bağlıdır [144].



Şekil 2.15: Bir algılayıcı sistemi blok diyagramı.

Bir ADC tarafından harcanan güç, örnekleme oranı ve etkin bit sayısı olmak üzere iki ana faktöre bağlıdır (Örneğin; algılayıcı çözünürlüğü) [145]:

$$GÜÇ \propto O_0 \cdot 2^{EBS} \quad (2.1)$$

Denklem 2.1’de, O_0 örnekleme oranını ve EBS etkin bit sayısını ifade eder. Algılanan verilerin hassasiyeti, uygulama ihtiyaçlarına bağlı olarak örnekleme oranını veya çözünürlüğü ayarlamakla kontrol edilebilir. Artan örnekleme oranı, algılanan verilerin daha iyi geçici çözünürlüğünü sağlar. Buna rağmen, algılanan olayın fiziksel özellikleri yüksek örnekleme oranlarına bazen gerek duymayabilir. Örneğin; bir sıcaklık algılama uygulaması için 1ms’lik algılama oranına ihtiyaç duyulmaz. Çünkü, sıcaklık genellikle onlarca dakika hatta saatler sonra değişir [20]. Sonuç olarak, algılama işlemindeki enerji tüketimi, algılanan olayın ve uygulamanın gereksinimlerine göre örnekleme oranını ayarlayarak azaltılabilir. Diğer yandan, çözünürlük, tek bir algılama yoluyla toplanan bilginin düzeyini tanımlar. Çözünürlüğü 8-bitten 10-bit ADC’ye çıkartmak daha doğru

sonuçlar sağlar. Bu artış, daha gelişmiş ADC devresi ve tek bir algılama için artan operasyon sayısı ile mümkündür. Bu yüzden, enerji tüketimi yaklaşık 4 kat artabilir [144]. Örnekleme frekansını ve çözünürlüğü ayarlanmanın yanısıra, algılayıcılardaki enerji yönetimi uyku durumları da içermelidir. Bir algılayıcı, belirli bir süre zarfında gerekli olmadığında, sadece kaçak akım oranında güç tüketen bir uyku durumuna geçirilmelidir. Ancak, algılayıcıyı açmadaki ek yük ve buna bağlı enerji tüketimi, algılayıcı ile ilgili sık kapatıp açmaları önlemek için dikkate alınmalıdır [20, 24, 132].

2.4.6.2. Veri İşleme

Veri işlemede harcanan enerji algılamada harcanan enerjiye benzerdir. Bununla birlikte, hesaplama işlemi veri iletişimine kıyasla çok daha az enerjiye ihtiyaç duyar (Şekil 2.14). Örneğin; 100m mesafeden 1Kb paket iletmenin enerji maliyeti, tipik bir mikroişlemci tarafından üç milyon komut yürütülmesindeki enerji maliyetine denk olabilir [118]. İletişim ve hesaplama enerjileri arasındaki bu ciddi fark, çok sekmeli bir algılayıcı ağda güç tüketiminin en aza indirilmesinde yerel veri işlemenin önemini gösterir.

Bir algılayıcı düğüm, yerleşik hesaplama becerilerine sahiptir ve üzerindeki alıcı-verici aracılığıyla çevresindeki düğümlerle etkileşime girer. Hesaplama yetenekleri, maliyet ve boyut sınırlamaları nedeniyle CMOS teknolojisine dayalı bir mikroişlemci tarafından sağlanır. CMOS teknolojisi, düşük maliyetli işlemcilerin algılayıcı düğüm devrelerinin içerisine gömülmesine imkan tanımaktadır. Fakat, CMOS teknolojisi enerji verimliliği bakımından doğal kısıtlamalara sahiptir. Bir CMOS transistörü, her açılıp kapatıldığında güç harcar. Açma-kapama gücü; saat frekansı, cihaz kapasitansı ve gerilim salınımının karesi ile orantılıdır. Ayrıca, işlemci, güç ve toprak arasındaki eşik altı kaçak akımlar nedeniyle enerji (kaçak akım enerjisi) kaybeder. Sonuç olarak, tek bir göreve yönelik veri işleme için enerji tüketimi ($E_{T(VI)}$), iki bileşenin toplamı şeklinde aşağıdaki denklem ile ifade edilebilir [146]:

$$E_{T(VI)} = N \cdot C \cdot V_{dd}^2 + V_{dd} \cdot \left(I_0 \cdot e^{\frac{V_{dd}}{n \cdot V_T}} \right) \cdot \left(\frac{N}{f} \right) \quad (2.2)$$

Denklem 2.2'nin ilk terimi, transistördeki değişimler (anahtarlama) sonucu gerçekleşen aktif enerji kaybını tanımlar. N , görev başına saat çevrimleri sayısı; C , çevrim başına

anahtarlanmış ortalama/toplam kapasitans; V_{dd} , besleme gerilimidir. Denklemdaki ikinci terim ise kaçak akımlar nedeniyle oluşan güç kaybını (kaçak akım enerjisini) ifade eder. I_0 , kaçak akım; n , mikroişlemci donanımı ile ilgili sabit sayı; V_T , eşik gerilimi ve f saat frekansındır.

Denklem 2.2'de gösterildiği gibi, veri işleme operasyonundaki enerji tüketimi; besleme gerilimine (V_{dd}), kontrol edilebilen saat frekansına (f) ve mikroişlemci mimarisine özgü diğer parametrelere bağlıdır. Besleme gerilimi azaldıkça enerji tüketimi kuadratik olarak azalır. Öte yandan, kapı gecikmesi de besleme gerilimine bağlıdır ve aşağıdaki denklem ile tanımlanabilir [146, 147]:

$$T_g = \frac{V_{dd}}{K \cdot (V_{dd} - V_{th})^a} \quad (2.3)$$

Denklem 2.3'deki a , K ve V_{th} ifadeleri mikroişlemci bağımlı değişkenlerdir. Besleme gerilimini (V_{dd}) azaltmak, mikroişlemcinin boşa kalma süresini düşürmek için kontrol edilebilen kapı gecikmesini artırır. Eğer mikroişlemci saat frekansında (f) çalışıyorsa, bu durum, her $T_0 = 1/f$ saniye (bu süre boyunca mikroişlemci tek bir görevi işletir) için bir kapı anahtarlamasına karşılık gelir. Eğer kapı gecikmesi (T_g) kapı anahtarlama süresinden (T_0) çok daha az olursa, işlemci bir görevin bitiminden bir sonraki görevin atanmasına kadar boşa tutulur. Bu nedenle; $T_g \leq T_0$, $T_0 \leq 1/f$ ve Denklem 2.3 dikkate alındığında, kapı gecikmesi besleme gerilimini düşürerek artırılabilir [146].

$$\left. \begin{array}{l} T_g \leq T_0 \\ \text{veya} \\ T_0 \leq 1/f \end{array} \right\} \Rightarrow T_g \leq 1/f \Rightarrow f \leq \frac{K \cdot (V_{dd} - V_{th})^a}{V_{dd}}, \quad (a \sim 2) \quad (2.4)$$

Denklem 2.4'de gösterildiği üzere, her saat frekansı değeri için bir minimum besleme gerilimi seviyesi vardır. Besleme gerilimini bu seviyeye indirmek, performansı engellemeden güç tüketimini azaltmanın etkili bir aracıdır. Bu yöntem, iş yüklerini eşleştirmek için işlemci güç kaynağı (V_{dd}) ve saat frekansının (f) uyarlandığı dinamik voltaj ölçeklendirme (DVÖ) olarak adlandırılır [148-154].

Bir mikroişlemci zamanla değişen hesaplama yükü kullandığında, azalan aktivite periyodları süresince saat frekansını basitçe düşürmek, Denklem 2.2’de belirtildiği gibi sadece kaçak akımlara bağlı güç kaybını etkiler. Ayrıca, besleme gerilimi saat frekansıyla bağlantılı olarak düşürülebilir. Bu durum, enerji tüketiminde kuadratik tasarruflara yol açar ve kaçak akımı azaltır. DVÖ tekniği hesaplama enerji tasarrufları sağlamasına rağmen, besleme gerilimini düşürmek işlemcinin pik performansından ödün verilmesine neden olur. Pik performansın her zaman istenmediğini kabul ederek önemli enerji kazançları sağlanabilir [148-154]. Dolayısıyla, işlemcinin çalışma gerilimi ve frekansı, anlık işlem gereksinimlerine dinamik olarak uyarlanabilir. Şekil 2.14'den anlaşılacağı üzere, veri işlemedeki güç tüketimi iletişimdekine göre çok daha azdır. Bu, enerji tasarruflu iletişim protokollerinin önemini belirtir. Her bir iletişim görevi bir hesaplama bileşeniyle ilişkilidir. Dolayısıyla, enerji tasarruflu iletişim tekniklerini DVÖ gibi adaptif hesaplamayla birleştirerek daha fazla enerji tasarrufu sağlamak mümkündür.

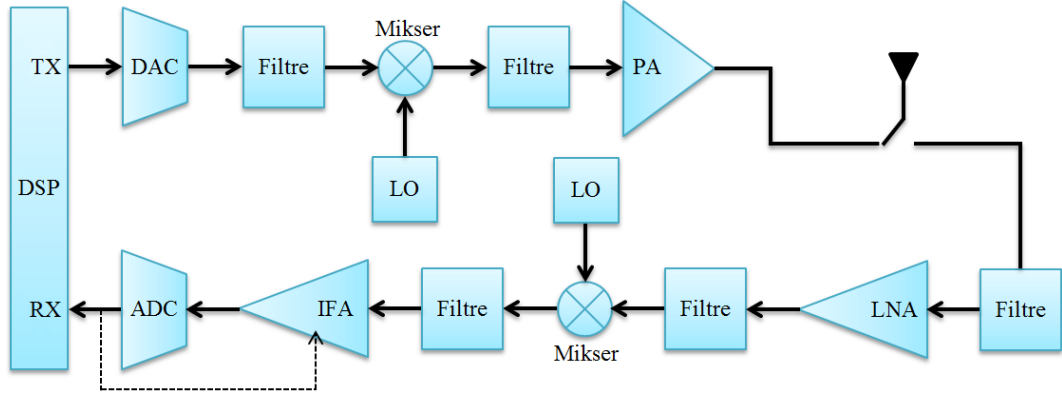
2.4.6.3. İletişim

Bir algılayıcı düğüm en fazla enerjiyi veri iletişiminde harcar. İletişim, verilerin hem alınması hem de gönderilmesi süresince alıcı-verici devre tarafından gerçekleştirilir. Şekil 2.14’de gösterilen algılayıcı düğümün güç tüketim analizi, bir alıcı-vericinin boşa kalma (Idle) zamanındaki enerji tüketimi ile veri gönderme (TX) ve alma (RX) işlemlerindeki enerji tüketimlerinin aynı miktarda olduğunu ortaya koymaktadır [20]. Ayrıca, algılayıcı düğümün herhangi bir veriyi iletmesi ya da alması gerekmediğinde alıcı verici kapatılıp uyku konumuna getirilerek önemli miktarda enerji tasarrufu sağlanabilir. Örneğin; XBee 868LP radyo modülü için yüksek veri hızında gönderme akımı 48mA, alma akımı 27mA, boşa kalma akımı 27mA ve uyku akımı 1.7µA’dır.

Bir alıcı-verici devresi (Şekil 2.16 ve 2.17); her biri ciddi güç tüketen ADC, DAC, filtre, mikser, güç amplifikatörü (PA), ara frekans amplifikatörü (IFA), düşük gürültülü amplifikatör (LNA), frekans sentezleyici, voltaj kontrollü osilatör (VCO), faz kilitlemeli döngü (PLL), modülatör ve demodülatör gibi çeşitli bileşenlerden oluşur. Alıcı-verici çifti için veri iletişimine yönelik güç tüketimi (G_{VI}), bu bileşenler tarafından tüketilen güçlerin toplamıdır ve basitçe aşağıda gösterildiği şekilde tanımlanabilir:

$$G_{VI} = G_C + G_{TX} + G_{RX} \quad (2.5)$$

G_C , verici gücü çıkışını belirtir. G_{TX} ve G_{RX} , sırasıyla gönderici ve alıcı elektroniklerde tüketilen güç miktarlarını ifade eder. Denklem 2.5'deki ilk iki terim gönderme tarafının güç tüketimini modellerken son terim ise alma tarafının güç tüketimini modeller.



Şekil 2.16: Bir RF alıcı-verici devre sisteminin blok diyagramı.

Veri gönderme ve alma işlemleri arasındaki en önemli fark, veri iletiminde güç amplifikatörlerinin (PA) kullanılmasıdır (Örneğin; G_C). Buna rağmen, düşük radyasyon gücü ile (~ 0 dBm) kısa mesafe iletişim için gönderme ve alma enerji maliyetleri hemen hemen aynıdır. Ayrıca, daha karmaşık modülasyon şemalarının geliştirilmesiyle birlikte bazı algılayıcı düğümlerde kod çözme enerjisi kodlama enerjisine göre baskındır ve alıcı gücü verici gücünden daha yüksektir (Örnek: MicaZ aygıtı). İletim güç tüketimi, algılayıcı düğüm yazılımı tarafından ayarlanabilen verici gücü seviyesine göre (0dBm ila -25dBm aralığında olabilir) değişebilir. Dolayısıyla, eğer verici gücü seviyesi çok düşük değerlere ayarlanırsa, alma gücü KAA'lardaki veri iletişimini domine eder [20].

Denklem 2.5'de gösterilen güç tüketim modeli enerji tüketimine yönelik temel faktörleri içerse de, iletişimdeki enerji tüketimini doğru bir şekilde analiz etmek için daha gelişmiş modeller gerekir. Gönderme ve alma modlarına ek olarak, bir alıcı-verici, aktif olmadığı periyotlarda enerji tasarrufu sağlamak amacıyla uyku moduna geçirilebilir. Ancak, alıcı-vericinin farklı modları arasındaki geçiş anlık değildir ve fazladan enerji tüketir. Uyku modu ile aktif (gönderme veya alma) modlar arasındaki geçişe bağlı enerji tüketimine başlangıç enerji tüketimi denir. Başlangıç enerji tüketimi, alıcı-vericide PLL kilit zamanıyla birlikte frekans sentezleyicinin ve voltaj kontrollü osilatörün başlangıç gecikmelerine dayanır. Başlangıç zamanı (yaklaşık yüzlerce mikro-saniye), başlangıç enerji tüketimini göz ardı edilemez kılar. Paket boyutu azaldıkça başlangıç güç tüketimi

aktif güç tüketimine hakim olmaya başlar. Sonuç olarak, alıcı-vericiyi sık sık kapatmak ve açmak verimsizdir. Çünkü, alıcı-vericiyi her zaman kapatıp açmada harcanan güç miktarı oldukça fazladır.

Başlangıç enerji tüketimi, (E_{BAS}), aşağıda belirtildiği biçimde modellenenebilir [155]:

$$E_{BAS} = P_{SO} \cdot t_{BAS}, (P_{SO} = P_{FS} + P_{VCO}) \quad (2.6)$$

Denklem 2.6'daki P_{SO} , frekans sentezleyici (FS) ve voltaj kontrollü osilatör (VCO) içeren devrenin güç tüketimidir. t_{BAS} , tüm alıcı-verici bileşenlerini başlatmak için gereken zamanı (ayarlama süresi) belirtir.

Uyku ile aktif modlar arasındaki geçişte harcanan enerjiye ek olarak, alıcı-vericinin gönderme modundan alma moduna anahtarlandığı zaman da enerji harcanır [155]. Bu anahtarlama zamanındaki ($t_{A(TX \rightarrow RX)}$) enerji tüketimi, ($E_{A(TX \rightarrow RX)}$), şu şekilde tanımlanır:

$$E_{A(TX \rightarrow RX)} = P_{SO} \cdot t_{A(TX \rightarrow RX)} \quad (2.7)$$

Şekil 2.17, bir kablosuz algılayıcı düğümde bulunan RF alıcı-verici devresine ait blok diyagramı göstermektedir. Alma ve gönderme modlarına yönelik enerji tüketimleri, Denklem 2.8 ve 2.9 ile temsil edilmektedir. Alıcı-verici; düşük gürültülü amplifikatör, mikser, frekans sentezleyici, voltaj kontrollü osilatör, ara frekans amplifikatörü ve demodülatör aktif bileşenlerini alma modunda iken kullanır. Frekans sentezleyici ve voltaj kontrollü osilatörün güç tüketimi P_{SO} olarak ifade edildiğinden, alma esnasındaki enerji tüketimi (alıcı enerji tüketimi) aşağıdaki gibi modellenenebilir [155]:

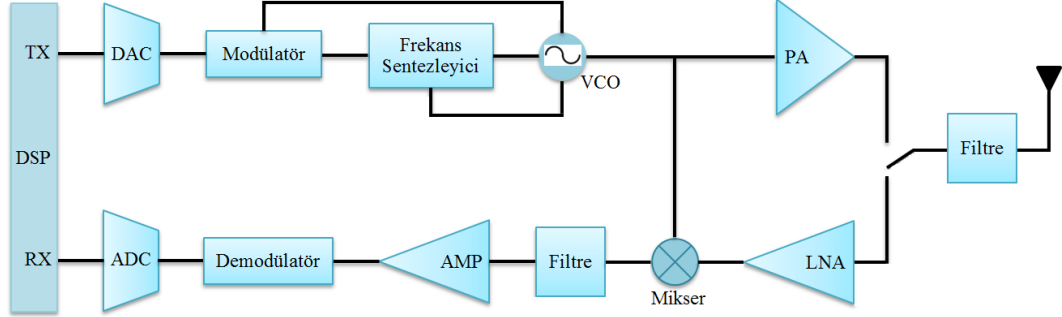
$$E_{RX} = (P_{SO} + P_{RX}) \cdot t_{RX} \quad (2.8)$$

Denklem 2.8'deki P_{RX} , kalan aktif bileşenlerin (düşük gürültülü amplifikatör, mikser, ara frekans amplifikatörü ve demodülatör) güç tüketimidir ve KAA'larda kullanılan veri hızı değerleri için sabittir. t_{RX} , bir paket almak için geçen süredir.

Şekil 2.17'de devre modeli gösterilen alıcı-verici; modülatör, frekans sentezleyici, voltaj kontrollü osilatör (alıcı ile paylaşımlı) ve güç amplifikatörü aktif bileşenlerini gönderme

modunda iken kullanır. Modülâtörün enerji tüketimi ihmal edilebilir olduğundan, veri iletimine yönelik enerji tüketimi şu şekilde ifade edilebilir [155]:

$$E_{TX} = (P_{SO} + P_{PA}) \cdot t_{TX} \quad (2.9)$$



Şekil 2.17: Bir kablosuz algılayıcı düğümde bulunan RF alıcı-verici devre modeli.

Denklem 2.9'daki P_{PA} , güç amplifikatörünün güç tüketimidir. Sabit olan alıcı modu enerji tüketiminin aksine, veri göndermedeki enerji tüketimi verici gücüne bağlı olarak (RF çıkış gücü) değişir. İstenen RF çıkış gücü seviyesi sistem yazılımı ile artırıldıkça güç amplifikatörü daha fazla enerji harcar. Bu da enerji tüketimini artırır. Dolayısıyla,

$$P_{PA} = \frac{1}{\eta} P_{out} \quad (2.10)$$

Denklem 2.10'daki η , güç amplifikatörünün güç verimliliğidir. P_{out} , istenen RF çıkış gücü seviyesidir. Örnek: MicaZ algılayıcı düğümdeki alıcı-verici, 0dBm ve -10dBm RF çıkış güçleri için sırasıyla 52.2mW ve 33mW tüketim gerçekleştirir [20]. RF çıkış gücü seviyesi, verilen d mesafesinde kabul edilebilir bir iletişim başarısı sağlamak için genellikle MAC ya da yönlendirme protokolleri gibi daha üst katman protokoller tarafından kontrol edilir. Dolayısıyla, güç amplifikatörünün güç tüketimi, d mesafesi cinsinden aşağıdaki gibi tanımlanabilir [155]:

$$P_{PA} = \frac{1}{\eta} \cdot \gamma_{PA} \cdot r \cdot d^n \quad (2.11)$$

$$\gamma_{PA} = \frac{(4\pi)^2}{G_{TX} \cdot G_{RX}} \cdot \lambda^{-2} \cdot N_o \cdot \left(\frac{E_b}{N_o} \right) \cdot F \cdot L_{mar} \quad (2.12)$$

Denklem 2.11’de; r veri hızı; n , kanalın yol kaybı üssü; η , güç amplifikatörü verimliliği; d , iletim mesafesidir. γPA ; d mesafesinde istenen sinyal/gürültü oranı ($SNR: E_b/N_o$), gönderme/alma anten kazançları (G_{TX}, G_{RX}), taşıyıcı frekansı dalga boyu (λ), termal gürültü gücü spektral yoğunluğu (F), bağlantı marjı (L_{mar}) ögelerine bağlı bir faktördür [155]. Güç amplifikatörünün güç tüketimi (P_{PA}), d mesafesinin bir fonksiyonu olduğundan, enerji tüketimi için mesafe-bağımlı bileşen olarak adlandırılır. Gönderme, alma, başlangıç ve anahtarlama modlarındaki enerji tüketimlerine yönelik diğer bileşenlere ise mesafeden-bağımsız bileşenler denir [20].

Yukardaki tanımlamalara göre, iletişimde tüketilen enerji için ayrıntılı bir model çıkartılabilir [20]. Bir algılayıcı düğümün komşu düğüme bir paket gönderdiği ve bunun sonucunda geriye bir cevap aldığı iletişim döngüsü; alıcı-vericinin başlatılması, paket iletimi, gönderme modundan alma moduna geçiş ve paket alma işlemlerini içerir. Sonuç olarak, iletişimin 1 çevrimindeki toplam enerji tüketimi şu şekilde modellenebilir [155]:

$$E_{TOPLAM} = E_{BAŞ} + E_{TX} + E_{A(TX \rightarrow RX)} + E_{RX} \quad (2.13)$$

$$E_{TOPLAM} = P_{SO} \cdot t_{st} + (P_{SO} + P_{PA}) \cdot t_{TX} + P_{SO} \cdot t_{A(TX \rightarrow RX)} + (P_{SO} + P_{RX}) \cdot t_{RX} \quad (2.14)$$

L_{PKT} uzunluğundaki bir paketin r veri hızında gönderildiği ya da alındığı varsayılırsa, gönderme ve alma süreleri $t_{TX} = t_{RX} = L_{PKT}/r$ olur. Buradan hareketle, toplam enerji tüketimi, Denklem 2.11 kullanılarak aşağıda belirtildiği biçimde modellenebilir [20]:

$$E_{TOPLAM} = P_{SO} \cdot (t_{st} + t_{A(TX \rightarrow RX)}) + (2P_{SO} + P_{RX}) \cdot \frac{L_{PKT}}{r} + \frac{1}{\eta} \cdot \lambda PA \cdot d^n t_{TX} \cdot L_{PKT} \quad (2.15)$$

Denklem 2.15’den görülebileceği üzere, iletişim nedeniyle ortaya çıkan enerji tüketiminin üç ana bileşeni vardır. İlk bileşen sabittir ve alıcı-verici devre sistemine bağlıdır. İkinci bileşen, d iletişim mesafesinden bağımsızdır; fakat, paket boyutu ya da iletim hızı yoluyla kontrol edilebilir. Denklem 2.15’deki ilk iki terim, enerji tüketiminin mesafeden-bağımsız bileşenlerini oluşturmaktadır. Son bileşen ise, iletişim mesafesine ve paket uzunluğuna bağlı olup MAC ve yönlendirme protokolleri gibi üst katman protokoller aracılığıyla kontrol edilebilir [20].

Sadeleştirilmiş Enerji Modeli

Enerji tüketim modeli, gönderici ile alıcı arasında d mesafesi kadar uzaklık bulunan bir gönderici-alıcı çifti için aşağıda gösterildiği biçimde sadeleştirilebilir [20]:

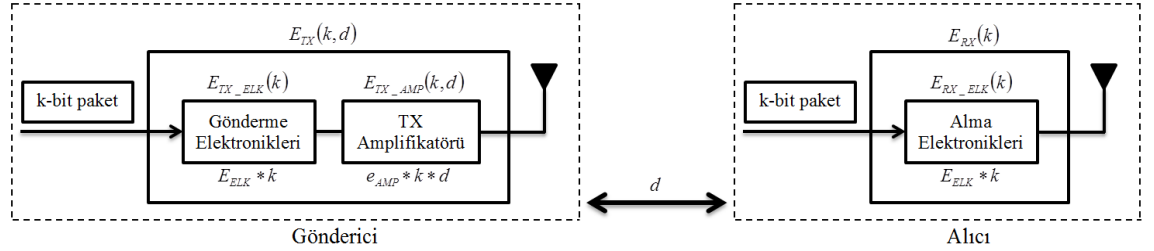
$$E_{TOPLAM} = E_{TX}(k, d) + E_{RX}(k) \quad (2.16)$$

$E_{TX}(k, d)$ ve $E_{RX}(k)$, sırasıyla gönderici ve alıcı enerji tüketimidir. Sadeleştirilmiş enerji modeli, gönderici-alıcı çifti için Şekil 2.18'de örneklendirilmiştir. Buna göre, göndericideki enerji tüketimi, gönderme elektronikleri ve gönderici amplifikatöründeki enerji tüketimlerini kapsar. Alıcı enerji tüketimi, sadece alıcı elektroniklerine bağlıdır. Dolayısıyla, gönderici ve alıcı enerji tüketimleri şu şekilde modellenebilir [20]:

$$E_{TX}(k, d) = E_{TX_ELK} \cdot k + e_{AMP} \cdot k \cdot d^n \quad (2.17)$$

$$E_{RX}(k) = E_{RX_ELK} \cdot k \quad (2.18)$$

E_{TX_ELK} ve E_{RX_ELK} , sırasıyla gönderici ve alıcı devre sisteminde bit başına enerji tüketimidir. e_{AMP} , güç amplifikatörü için mesafeye göre bit başına enerji tüketimidir.



Şekil 2.18: Sadeleştirilmiş enerji modeli blok diyagramı.

Ayrıntılı Enerji Modeli

İletişimdeki enerji tüketimi; donanım profilleri, verici gücü seviyesi, paket boyutu ve mesafe gibi çeşitli faktörlere bağlıdır. Enerji verimliliği KAA'larda çok önemli bir parametre olduğundan, bu faktörlerin dikkatle seçilmesi gerekir. Denklem 2.15'deki toplam enerji tüketiminden görülebileceği üzere, enerji tüketimi, alıcı-vericinin aktif olduğundaki süreye doğrudan bağlıdır. İletişimin enerji verimliliğini artırmanın bir yolu, alıcı-vericinin gönderme zamanını azaltmaktır [20]. Bu durum, sembol başına çoklu bit göndererek başarılabilir (Örneğin; M-ary modülasyon kullanarak) [156]. Böylece; etkin paket boyutu (L_{PKT}) azalır, L_{PKT} 'nin azalmasıyla gönderme ve alma zamanları da düşer.

L_{PKT} bitlik bir paket göndermek için, bir ikili modülasyon şeması L_{PKT} kadar sembol kullanır. Sembol süresi $T_s = 1/r$ olarak alınırsa, ikili modülasyon için bir paketin iletim zamanı $L \cdot T_s$ olur [20]. Bu süre, sembol başına b ($b = \log_2^M$) bit kullanan bir M-ary modülasyon şeması aracılığıyla kısaltılabilir. Etkin paket uzunluğu $L_s = L_{PKT}/b$ olur ve iletim zamanı b 'nin faktörüyle azaltılabilir [156]. M-ary modülasyon gönderme süresini azaltmasına rağmen, alıcı-vericinin devre karmaşıklığı ve güç tüketimi artar. Ayrıca, güç amplifikatörünün verimliliği de düşer. Bu, tek sembol için uygun verici çıkış gücü seviyeleri sağlamak amacıyla daha fazla güce ihtiyaç duyulduğunu gösterir. Dolayısıyla, M-ary modülasyon kullanarak alıcı-verici devre sisteminin enerji tüketimini artırma bedeline karşılık güç amplifikatörünün enerji tüketimi düşürülebilir [20]. Sonuç olarak, belirli bir KAA için devre enerjisinden ödün verilmeksizin faaliyet süresinin azaldığı bir optimal modülasyon seviyesi vardır [24, 156].

Çok seviyeli dört evreli genlik modülasyonu (MQAM) şeması kullanıldığını varsayarak, bir ikili iletişimin toplam enerji tüketimi şu şekilde ifade edilebilir [156]:

$$E_{TOPLAM} = (1 + \alpha) \frac{4}{3} (N_f) \cdot \sigma^2 \cdot \frac{2^b - 1}{b} \cdot \ln \frac{4(1 - 2^{-b/2})}{b \cdot P_b} G_d + \frac{P_C \cdot T_{on} + 2P_{syn} \cdot T_{tr}}{L_{PKT}} \quad (2.19)$$

Burada;

$$\alpha = \frac{\xi}{\eta} - 1, \quad \xi = 3 \frac{\sqrt{M} - 1}{\sqrt{M} + 1}, \quad M = 2^{L/(B \cdot T_{on})} \quad (2.20)$$

L , paket uzunluğu; B , kanal band genişliği; N_f , alıcı gürültü sayısı; σ^2 , güç spektrum enerjisi; P_b , bit hata olasılığı; G_d , güç kazancı faktörü; P_C , devre güç tüketimi; P_{syn} , frekans sentezleyici güç tüketimi; T_{tr} , frekans sentezleyici ayarlama zamanı (kurulum zamanı); T_{on} , zamanında alıcı-verici; M , ise modülasyon parametresidir [156].

Denklem 2.19'daki ayrıntılı enerji modeli, iletişim için enerji tüketimini etkileyen çeşitli faktörleri içerir. Denklem 2.16'daki sadeleştirilmiş enerji modeliyle karşılaştırıldığında, bu model, enerji tüketimini analiz etmek amacıyla daha fazla detay ve esneklik sağlar. Fakat, bu seviyede bir ayrıntı belirli uygulamalar için gerekli değildir. Dolayısıyla, en uygun enerji modelinin seçimi analiz odağına bağlıdır [20].

Genel Enerji Tüketimi

Denklem 2.15'deki iletişimin tek çevriminde tüketilen enerjiye, Denklem 2.16'daki sadeleştirilmiş enerji modeline ve Denklem 2.19'daki ayrıntılı enerji modeline ek olarak, iletişim enerjisi alıcı-vericinin kullanıldığı orana bağlıdır. Bu oran; uygulamanın türü, taşıma katmanı protokolü, yönlendirme protokolü ve MAC katmanı gibi birçok faktöre bağlıdır [20]. Sadece belirli koşullar yerine geldiğinde verilerin oluşturulduğu ve diğer durumlarda ağın boşa kaldığı bir olay temelli uygulamaya kıyasla, alıcı-verici, sürekli bilgi erişimi gerektiren bir gözetim uygulamasında daha fazla sıklıkta kullanılır. Benzer şekilde, taşıma katmanı protokolü, alıcı-vericinin ne kadar sıklıkta açılması gerektiğini belirleyen oranı (ağa aktarılacak veri miktarı) ayarlar [20]. Öte yandan, yönlendirme protokolleri, komşuluk bilgisi toplamak için komşular arasındaki kontrol mesajlarını algılayarak haberleşme hızını artırabilir. Kötü kanal durumlarına sahip bir algılayıcı düğümün seçilmesinden dolayı çoklu yeniden iletimler gerekebileceğinden, yönlendirme protokolü tarafından seçilen bir sonraki sekme iletişim hızını da belirler. MAC katmanı, yayın kanalına erişim sağlayan algılayıcı düğüm aracılığıyla kanal erişim hızını kontrol eder. Bu hız, MAC türüne bağlı olarak değişebilir [20-25].

Yukarıdaki açıklamalar dikkate alındığında, iletişimin genel/tüm enerji tüketimi aşağıda gösterildiği şekilde modellenebilir [139]:

$$E_{TOPLAM} = N_{TX} \cdot [P_{SO} \cdot (t_{TX} + t_{BAS}) + P_{PA} \cdot (t_{TX})] + N_{RX} \cdot [P_R \cdot (t_{RX} + t_{BAS})] \quad (2.21)$$

Denklem 2.21'de; P_{SO} , frekans sentezleyici (senkronizer) ve voltaj kontrollü osilatör (VCO) tarafından tüketilen güç; P_{PA} , verici çıkış gücü (güç amplifikatörünün harcadığı güç); $P_R = (P_{SO} + P_{RX})$, alıcı tarafından tüketilen güç; t_{TX} ve t_{RX} , sırasıyla gönderici ve alıcı zamanı; t_{BAS} , gönderici veya alıcı başlangıç zamanıdır. N_{TX} ve N_{RX} , sırasıyla gönderici ve alıcının birim zaman başına çalıştırılma sayısıdır (kaç kez anahtarlandığı) ve bu sayı; uygulama, taşıma, ağ ve MAC katmanlarına bağlıdır [139].

2.4.7. Üretim Maliyetleri

KAA'lar çok sayıda algılayıcı düğümden oluştuğundan, tek bir düğümün maliyeti ağın toplam maliyeti açısından büyük önem taşır. Ağın maliyeti geleneksel algılayıcı düğümlerin konuşlandırılmasından daha pahalı ise, algılayıcı ağ maliyet açısından

ekonomik olmaz. Dolayısıyla, KAA'ların pratik bir şekilde uygulanabilmesi için her bir algılayıcı düğümün maliyeti düşük tutulmalıdır. Ayrıca, algılayıcı düğüm bazı ek donanımlar içerebilir. Örneğin; algılayıcı düğüm, KAA uygulamalarına bağlı olarak bir konum bulma sistemi, harekete geçirici veya güç üretici ile donatılabilir. Bu birimlerin tümü algılayıcı cihazlarının maliyetine ek külfet getirir. Nihayetinde, bir algılayıcı düğümün maliyeti, işlevsellik sayıları göz önüne alındığında oldukça masraflı bir durumu ortaya koymaktadır.

2.5. ALGILAYICI DÜĞÜM PLATFORMLARI

Algılayıcı düğümlerin donanım özellikleri uygulamalara bağlı olarak farklılık gösterir. Bununla birlikte, sınırlı sayıda kablosuz iletişim modülü KAA'lar için kullanılmaktadır. Tablo 2.1; işlemci hızı, programlanabilir ve depolama bellek boyutu, çalışma frekansı, iletim hızı bakımından son yıllarda tasarlanmış popüler algılayıcı düğüm platformları ile çalışmamızda geliştirdiğimiz kablosuz algılayıcı düğümümüzün ve koordinatör cihazın temel özelliklerini ortaya koymaktadır. Tablodan anlaşılacağı üzere, algılayıcı düğüm platformlarının uygulamaya özgü yetenekleri önemli ölçüde çeşitlilik göstermektedir. Buradan hareketle, kablosuz algılayıcı düğümler, işlem becerileri ve KAA'lardaki kullanım amaçlarına göre alt-düzey ve üst-düzey platformlar olmak üzere iki grupta sınıflandırılabilir.

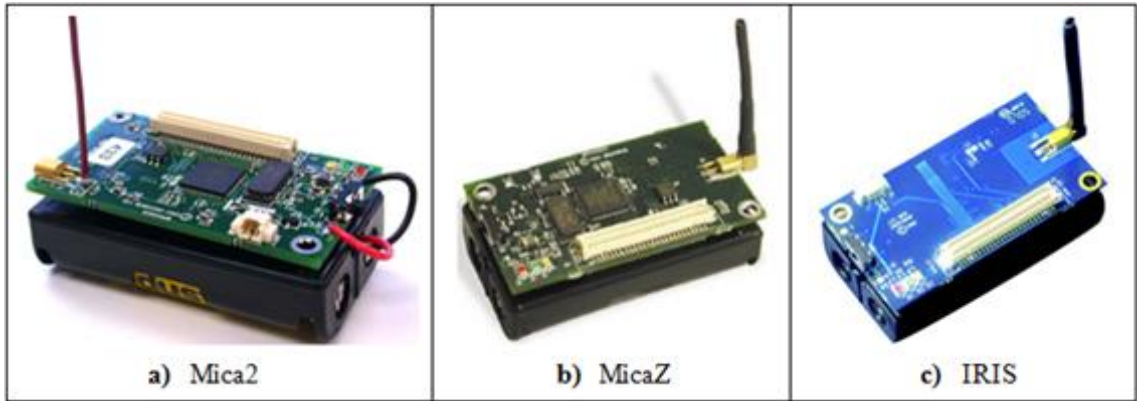
Tablo 2.1: Çeşitli kablosuz algılayıcı düğümler ve temel özellikleri.

Algılayıcı Düğüm	İşlemci	CPU Hızı	Alıcı-Verici Modül	Program Hafızası	RAM	Harici Hafıza	Radyo Frekansı	İletim hızı
BTnode	Atmel ATmega128L	8 MHz	TI CC1000, Bluetooth	64Kbyte	180 Kbyte	128 Kbyte Flash 4 Kbyte EEPROM	433-915 MHz, 2.4 GHz	değişken
Cricket	Atmel ATmega128L	8-16 MHz	RF ve ultrason alıcı-verici	128 Kbyte	4 Kbyte	128 Kbyte Flash	433 MHz	değişken
EPIC mote	TI MSP430	8 MHz	TI CC2420	----	10 Kbyte	48 Kbyte Flash	2.4 GHz	250 kbps
EYES	TI MSP430F149	8 MHz	RFM TR1001	----	----	8 Mbit	868 MHz	115.2 kbps
EyesIFX v2	TI MSP430F1611	8 MHz	TDA5250 FSK	60 Kbyte	2 Kbyte	8 Mbit	868 MHz	115.2 kbps
Imote	ARM core	12 MHz	Bluetooth	----	64 Kbyte	512 Kbyte Flash	2.4 GHz	100 kbps
Imote2	Marvel PXA271	13-416 MHz	TI CC2420	32 MB SRAM	256 Kbyte	32 MB Flash	2.4 GHz	250 kbps
IRIS	Atmel ATmega1281	8 MHz	Atmel T86RF230	128 Kbyte	8 Kbyte	4 Kbyte EEPROM	2.4 GHz	250 kbps
Mica	Atmel ATmega103	4 MHz	RFM TR1000	128 Kbyte	4 Kbyte	512 Kbyte Flash	868 MHz	10/40/50 kbps
Mica2	Atmel ATmega128L	8-16 MHz	TI CC1000	128 Kbyte	4 Kbyte	128 Kbyte Flash	433-916 MHz	38.4 kbps
MicaZ	Atmel ATmega128L	8-16 MHz	TI CC2420	128 Kbyte	4 Kbyte	128 Kbyte Flash	2.4 GHz	250 kbps
Shimmer	TI MSP430F1611	8 MHz	TI CC2420	48 Kbyte	10 Kbyte	2 GB microSD	2.4 GHz	250 kbps
TelosB	TI MSP430	8 MHz	TI CC2420	----	10 Kbyte	48 Kbyte Flash	2.4 GHz	250 kbps
Tmote SKY	TI MSP430	8 MHz	TI CC2420	----	10 Kbyte	48 Kbyte Flash	2.4 GHz	250 kbps
XYZ	OKI ML67Q500x ARM THUMB	57.6 MHz	TI CC2420	256 Kbyte	32 Kbyte	2 Mbit SRAM	2.4 GHz	250 kbps
Algılayıcı Düğümümüz	Atmel ATmega328P	8 MHz	XBee 868LP	32 Kbyte	2 Kbyte	1Kbyte EEPROM 2Kbyte SRAM	868 MHz	80 kbps
Sun SPOT	ARM AT91SAM9G20	400 MHz	CC2420	----	1 MB	8 MB Flash	2.4 GHz	250 kbps
V-Link	----	----	----	----	----	----	2.4 GHz	250 kbps
TEHU-1121	----	----	IEEE 802.15.4	----	----	----	2.4 GHz	250 kbps
NI WSN-3202	----	----	IEEE 802.15.4	----	----	----	2.4 GHz	250 kbps
Stargate	Intel PXA255	400 MHz	IEEE 802.11	----	64 MB	32 MB Flash	2.4 GHz	değişken
Netbridge NB-100	Intel IXP420	266 MHz	Ethernet	8 MB	32 MB	2 GB USB 2.0	Değişken	değişken
Koordinatör Düğümümüz (Raspberry Pi)	Broadcom BCM2835	700 MHz	XBee, Ethernet, WiFi, 3G	----	256 MB 512 MB	8 GB SD kart	Değişken	değişken

2.5.1. Alt Düzey Platformlar

Alt düzey platformlar; veri işleme (işlemci), depolama (bellek), güç kaynağı ve iletişim anlamında sınırlı kapasitelere sahiptir. Genellikle algılama görevlerini yerine getirmek, veriler üzerinde basit hesaplamalar yapmak ve bağlantı altyapısı sağlamak için bir KAA'da çok sayıda konuşlandırılacak şekilde tasarlanmaktadır. Son zamanlarda en çok kullanılan alt düzey platformlardan bazıları aşağıda açıklanmıştır:

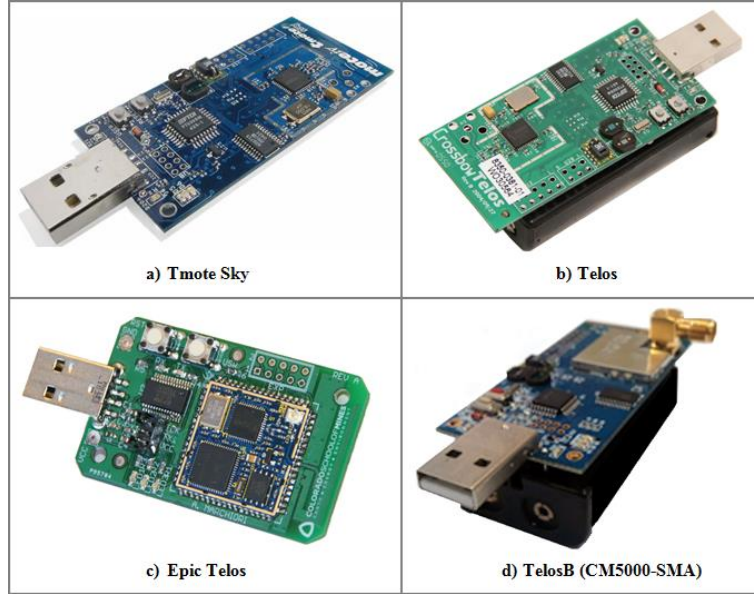
Mica Ailesi [121]: Crossbow firması tarafından üretilmektedir. Mica, Mica2, MicaZ ve IRIS kablosuz algılayıcı düğümlerini kapsar (Şekil 2.19). Her düğüm, 4-16MHz hıza ve 128-256kB program flaş hafızaya sahip 8-bit Atmel-AVR mikrodenetleyicisini kullanır. Mikrodenetleyicileri benzer olmakla birlikte, Mica kablosuz algılayıcı düğüm ailesinde farklı tipte RF alıcı-verici modüller vardır. Mica düğümü; 40kbps, 433/916MHz alıcı-verici içerir. Mica2 düğümü; 40kbps, 433/868/916MHz alıcı-vericiye sahiptir. MicaZ ve IRIS düğümleri ise; 250kbps veri hızı ile 2.4GHz hızında çalışabilen IEEE 802.15.4 uyumlu alıcı-vericiler ile donatılmıştır. Mica ailesine ait her bir algılayıcı düğüm, RAM (4-8kB) ve program belleği (128kB) açısından sınırlı kapasiteye sahiptir. Ayrıca, ilave algılayıcı ve programlama kartları bağlamak için düğüm platformu üzerinde 51-pinli bir konektör bulunur.



Şekil 2.19: Mica algılayıcı düğüm ailesi.

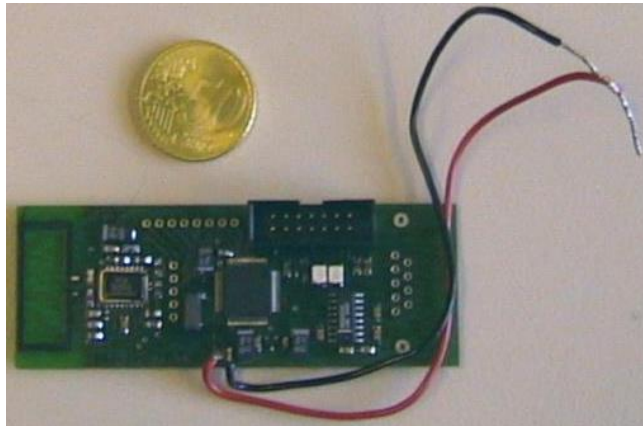
Telos/Tmote [121]: MicaZ kablosuz algılayıcı düğüm platformuna benzer bir mimari, Crossbow'un Telos ve Sentilla'nın Tmote Sky [157] algılayıcı düğüm platformları için uyarlanmıştır (Şekil 2.20). Telos/Tmote, MicaZ'deki ile aynı radyo modülünü (250kbps, 2.4GHz IEEE 802.15.4/ZigBee uyumlu Chipcon CC2420 RF alıcı-verici) kullanmasına rağmen daha fazla RAM (10kB) kapasitesine sahip 8MHz TI-MSP430 mikrodenetleyici

bulundurur. Telos/Tmote platformları; ışık, IR(Infrared-Kızılötesi), nem ve sıcaklık gibi çeşitli algılayıcılarla tümleşik vaziyette olup ilave algılayıcı ya da programlama kartları ihtiyacını ortadan kaldıran bir USB konektöre sahiptir. Ayrıca, bu platformlar üzerinde ek algılayıcılar için 6 ve 10 pinli bir konektör de mevcuttur.



Şekil 2.20: Telos/Tmote algılayıcı düğüm ailesi.

EYES [158]: TelosB/Tmote algılayıcı düğüm mimarilerine benzerdir. 60kB program belleği ve 2kB veri belleği olan 16-bit mikrodenetleyici kullanır. Pusula, ışık, sıcaklık, basınç, ivmeölçer algılayıcılarını bulundurur. EYES (Şekil 2.21); maksimum 115.2kbps iletim hızını destekleyen ve alma, gönderme, uyku modlarında sırasıyla 14.4mW, 16.0mW, 15.0µW güç tüketimlerine sahip RFM-TR1001 RF alıcı-verici modül kullanır. Bu platform, programlama için RS232 seri haberleşme arabirimi de içerir.



Şekil 2.21: EYES alt düzey algılayıcı düğüm prototipi.

Yukarıda açıklanmış olan kablosuz algılayıcı düğüm platformlarıyla benzer özelliklere ve yeteneklere sahip çok sayıda alt düzey platform bulunmaktadır. Bunlardan bazıları, Tablo 2.1'de listelenmiştir. Son yıllarda; LORD MicroStrain Sensing Systems (V-Link-LXRS), Wireless Sensors (SensiNet TEHU-21XX, SensiNet TEMP-10XX/11XX) ve National Instruments (NI WSN-32XX) gibi patentli endüstriyel kablosuz algılayıcı düğüm platformları da geliştirilmiştir.

Alt düzey platformlar; KAA uygulamalarında algılama, izleme, gözetim işlemleri için kullanılırlar ve çok sekmeli ağ oluşturulurken bir bağlantırlık altyapısı sağlarlar. Maliyeti ve enerji tüketimini azaltmak amacıyla genellikle düşük güçlü mikrodenetleyiciler ve RF alıcı-verici modüller ile donatılırlar. Dolayısıyla, KAA'ların algılanacak ortama konuşlandırılmasında çok fazla sayıda bulunurlar. Alt düzey platformlar, birçok ülkede lisans ücretsiz iletişim sunan ISM frekans bandlarında çalışırlar. Daha yeni platformlar; 2.4GHz frekans bandında çalışan, IEEE 802.15.4 ve ZigBee standartları ile uyumlu olan CC2420 radyo modülü içerir. Bu standardizasyon, bir ağda farklı özellikte platformların kullanılmasını ve KAA'ların heterojen bir şekilde yerleştirilmesini sağlar. KAA iletişim protokollerinin çoğu alt düzey platformlar kullanılarak geliştirilmiştir.

2.5.2. Üst Düzey Platformlar

Algılama, lokal veri işleme ve çok sekmeli iletişim işlemlerine ilaveten KAA'lar, alt düzey platformların verimli bir şekilde yerine getiremeyeceği kompleks özellikler de içerir. Ağ yönetimi gibi yüksek seviye görevler, alt düzey platformların kapasiteleri dikkate alındığında daha yüksek işlem gücü ve bellek gereksinimlerine ihtiyaç duyar. Ayrıca, KAA'ların mevcut ağ altyapısı ile entegrasyonu, ağ geçidi modülleri aracılığıyla entegre edilmiş çoklu iletişim tekniklerini gerektirir. Bunun yanında, ağ cihazlarının algılayıcı düğümler ile birlikte kullanıldığı ağlarda yüksek kapasiteli düğümlere ihtiyaç duyulur. Sonuç olarak, KAA'larda bu tür gereksinimleri karşılamak için üst düzey platformlar geliştirilmiştir. KAA uygulamalarında kullanılan üst düzey platformlardan bazıları aşağıda açıklanmıştır:

Stargate [159]: Algılama, sinyal işleme, kontrol ve algılayıcı ağ yönetimi amacıyla tasarlanmış düşük güçlü, yüksek performanslı işlem platformudur (Şekil 2.22). Compaq IPAQ ve Dell Axim gibi birçok el bilgisayarında bulunan 32-bit 400MHz Intel-PXA255 XScale işlemci kullanır. 32MB flaş bellek, 64MB SDRAM, PCMCIA Bluetooth, IEEE

802.11 kartlarının yanı sıra Mica ailesi algılayıcı düğümleri bağlamak için bir on-board konektöre sahiptir. Stargate, ağ-içi işlem algoritmalarını gerçekleştirmede kablosuz ağ geçidi ve hesaplama merkezi olarak çalışabilir. Bir web kamerası veya farklı bir görüntüleme cihazı bağlandığı zaman orta çözünürlüklü multimedya algılayıcı düğüm gibi işlev görebilir. Bu özellikleri nedeniyle enerji tüketimi yüksektir [160].



Şekil 2.22: Stargate tek kartlı bilgisayar.

Stargate NetBridge [161], Stargate'in halefi olarak geliştirilmiştir. 266MHz'de çalışan Intel-IXP420-XScale işlemcisine dayalı tasarlanmıştır. Donanım özellikleri bakımından bir Ethernet, iki USB 2.0 portu bulunur. 8MB program flaş, 32MB RAM, 2GB USB 2.0 sistem diskisi ile donatılmış cihaz Linux işletim sistemi çalıştırır. Bir algılayıcı düğüm, USB portları kullanılarak ağ geçidi işlevsellikleri için bağlanabilir.

Imote/Imote2: Intel; Imote ve Imote2 olarak bilinen, yüksek performanslı algılama ve ağ geçidi uygulamalarında kullanılmak üzere iki yeni nesil üst düzey kablosuz algılayıcı düğüm prototipi geliştirmiştir (Şekil 2.23) [125]. Imote; 8-bit 12-48MHz ARM-7TDMI işlemci, 512kB flaş bellek, 64kB SRAM, bir Bluetooth alıcı-verici ve bunların yanı sıra çeşitli giriş/çıkış konektörlerinden oluşan entegre kablosuz mikrodenetleyici üzerine inşa edilmiştir. Yazılım mimarisi TinyOS ARM portuna dayanmaktadır.

Intel algılayıcı düğümlerinin ikinci nesil ögesi olan Imote2; depolama veya sıkıştırma için DSP işlemleri sağlayan, düşük güçlü, yüksek performanslı 32-bit 13-416MHz PXA271-XScale işlemci ve IEEE 802.15.4 TI-ChipCon-CC2420 RF alıcı-verici modül

kullanılarak geliştirilmiştir. 256kB SRAM, 32MB SDRAM ve 32MB flaş belleklere, alternatif alıcı-verici modüller için ek desteğe, dijital algılayıcı veya kamera bağlamak amacıyla yüksek hızlı giriş/çıkış konektörlerine sahiptir. Linux işletim sistemi ve Java uygulamalarını çalıştırabilir.



Şekil 2.23: Imote2 tek kartlı bilgisayar.

3. MALZEME VE YÖNTEM

KAA'lar, çeşitli algılayıcılar aracılığıyla çevreleri ile etkileşime girebilme, lokal olarak bilgi işleyebilme ve bu bilgiyi kablosuz iletişim kullanarak komşularına aktarabilme yeteneklerine sahip, amaca özel geliştirilmiş algılayıcı düğümlerden oluşmaktadır. Bir algılayıcı düğüm; tek bir devre şeklinde, özel bir devre kartı halinde ya da bir gömülü sistem içinde olmak üzere üç farklı yapıda tasarlanabilir:

Algılayıcı devre: Sahip olduğu iletişim becerisi ve uygulama kodunun bulunduğu programlanabilir belleği ile KAA'nın en önemli bileşenidir. Algılayıcı devre, genellikle; *algılayıcılar, mikroişlemci, bellek, alıcı-verici ve güç kaynağı* birimlerini içerir. Örnek: Sun SPOT, Mica2, MicaZ, IRIS, Cricket, TelosB, Lotus, Imote, Shimmer ve XYZ.

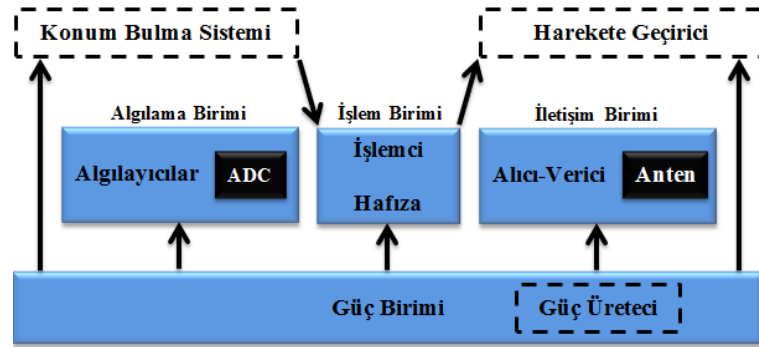
Algılayıcı kart: Farklı türde algılayıcıların gömülü halde bulunduğu modüler yapıya sahip bir algılayıcı kartın, algılayıcı düğüm üzerine monte edilmesiyle oluşturulur. Algılayıcı kart üzerinde, uygulamaya özel üretilmiş algılayıcıları bağlamak için kullanılan ek prototip alanlar bulunur. Örneğin; Mica ve IRIS kablosuz algılayıcı düğümlerde kullanılan MTS400/420 ve MDA100/300 algılayıcı kartlar [121]. Alternatif olarak algılayıcılar, Telos veya Sun SPOT platformlarındaki gibi kablosuz iletişim modülünün içerisine de entegre edilebilir.

Ağ Geçidi (Koordinatör): Çeşitli özelliklere sahip kablosuz algılayıcı düğümleri; lokal bir bilgisayara/sunucuya, endüstriyel/işletme ağına, kablolu/kablosuz yerel alan ağına veya İnternete bağlamak amacıyla Ethernet, WiFi, USB ya da seri port içeren çoklu iletişim arayüzleri sağlar. Programlanabilir/ayarlanabilir özelliktedir (radyo aracılığıyla programlanabilir). Algılayıcı düğümleri programlamak, yönetmek ve onlardan veri toplamak için kullanılır. Ağ geçidine verilebilecek örnekler, MIB510/520/600 [121] ve AS-SG1000 [162] cihazlarıdır. Uygulamayı programlanabilir belleğe yüklemek için özel platformların ağ geçidine bağlı olması gerekir.

3.1. GÜÇ-BİLİNÇLİ KABLOSUZ ALGILAYICI DÜĞÜM TASARIMI

Bir kablosuz algılayıcı düğüm KAA için merkezi ögedir. Konuşlandırıldığı bölgede algılama işlemini gerçekleştirir. Veri işleme algoritmalarını, iletişim protokollerini depolar ve çalıştırır. KAA'dan çıkartılan algılanmış verilerin kalitesi, boyutu ve frekansı algılayıcı düğümüne sağlanan fiziksel kaynaklardan etkilenir. Bu nedenle, kablosuz algılayıcı cihazın tasarımı ve gerçekleştirilmesi kritik bir adımdır.

Bir kablosuz algılayıcı düğümün genel mimarisi Şekil 3.1'de gösterilmektedir. Buna göre, algılayıcı düğüm genellikle dört ana bileşenden oluşur: *algılama birimi*, *işlem birimi*, *iletişim birimi* ve *güç birimidir*. Ayrıca, *konum bulma sistemi*, *güç üretici*, *harekete geçirici* ve *aktüatör* gibi özel birimler de uygulamaya bağlı olarak algılayıcı düğümüne entegre edilebilir.



Şekil 3.1: Kablosuz algılayıcı düğüm mimarisi.

Algılama birimi: Fiziksel dünyadan bilgi toplama yeteneğine sahip çeşitli algılayıcıları içerir. Sıcaklık, nem, basınç, ışık şiddeti, ivme gibi belirli olayları algılama işleminden sorumlu olup genellikle 2 alt birimden oluşmaktadır. Bunlar, algılayıcı ve analog-dijital-dönüştürücüdür (ADC). Gözlemlenen olaya bağlı olarak algılayıcı tarafından üretilen analog sinyaller, ADC tarafından dijital sinyallere dönüştürülür ve daha sonra işlem birimine aktarılır. Algılayıcı birimde güç tüketimi; algılama, sinyal örnekleme, sinyal iyileştirme ve analog sinyalleri dijital sinyallere çevirme işlemlerinden kaynaklanır. Çalışmamızda, algılama biriminde kullandığımız algılayıcılar aşağıda belirtilmiştir:

- *İklim sensörleri:* Başınç sensörü (Bosch-BMP085), ışık sensörü (AMS-TSL2771X, AMS-TSL4531X), sıcaklık ve nem sensörü (SENSIRON-SHT2X).
- *Toprak sensörü:* Toprak sıcaklık/nem sensörü (Spectrum Technologies-WaterScout SMEC 300 SMEC/T).

İşlem birimi: Kablosuz algılayıcı düğümün temel denetleyicisidir. Algılama, veri işleme, hesaplama ve iletişim ile ilgili komutları yürütür. Algılayıcı düğümdeki diğer bileşenlerin yönetilmesini sağlar. İşlem birimi; gerekli hesaplamaları yapmak için bir mikroişlemci, program komutlarını depolamak için bir kalıcı bellek (flaş program hafızası), algılanan verileri geçici olarak saklamak için bir aktif hafıza (RAM), yüksek hızlı veri yolları, iletişim arayüzleri ve dahili saat içerir. Algılama işlemlerini denetleme, algılanan verilere dayalı hesaplamalar yapma, ilgili algoritmaları/protokolleri çalıştırma ve kablosuz iletişim yoluyla diğer düğümlerle organize çalışma gibi algılayıcı düğüme önemli özellikler/yetenekler sağlayan prosedürleri işler.

Kablosuz algılayıcı düğüm tasarımlarında çeşitli özellikte işlemciler kullanılmaktadır. İşlem birimi algılayıcı düğümün maliyetini, esnekliğini, performansını ve enerji tüketimini etkilediğinden işlemci seçimi dikkatli yapılmalıdır. Algılama görevi baştan iyi tanımlanır ve zamanla değişmezse, FPGA (Field Programmable Gate Array-Alanda Programlanabilir Kapı Dizisi) ya da DSP (Digital Signal Processor-Dijital Sinyal İşlemci) kullanılabilir. Bu işlemciler; enerji tüketimi açısından çok verimli, en basit algılama görevleri ve hesaplamalar için oldukça yeterlidir. Ancak, genel amaçlı işlemciler olmadıklarından tasarım ve uygulama süreci karmaşık ve masraflıdır. Bununla birlikte, birçok pratik durumda algılama amacı değişir veya modifikasyon gerekli olabilir. Ayrıca, kablosuz algılayıcı düğüm üzerinde çalışan yazılım zaman zaman güncellemeler veya uzaktan hata ayıklama gerektirebilir. Bu tür görevler çalışma süresinde ciddi miktarda hesaplama ve işlem alanı gerektirir. Bu durumda; özel amaçlı, enerji tasarruflu işlemciler kablosuz algılayıcı düğüm tasarımı için uygun değildir.

DSP; pahalı, büyük boyutlu ve daha az esnektir. Buna rağmen, spesifik algoritmalar kullanılarak sinyal işleme konusunda en iyi işlemcidir. FPGA, hem DSP'den hem de mikrodenetleyiciden daha hızlıdır. Ayrıca, paralel hesaplamayı destekler. KAA'larda algılama, veri işleme, hesaplama, iletişim işlemlerinin birbirine bağımlı gerçekleşmesi gerektiğinden FPGA'in kullanılması çok yararlıdır. Ancak, yüksek üretim maliyeti ve programlama zorluğu FPGA'i daha az cazip hale getirir. Hem basit algılama görevleri hem de büyük ölçekli konuşlandırma gerektiren uygulamalar için ASIC (Application Specific Integrated Circuit-Uygulamaya Özel Entegre Devre) gibi mimarisi basit ama enerji tasarruflu, düşük maliyetli işlemciler kullanmak algılayıcı düğüm tasarımında

tercih edilebilir. Performans, birkaç uygulamanın paralel olarak çalıştırılabileceği çok çekirdekli sistemlerin uygulanmasıyla geliştirilebilir. Bu, ASIC'in diğer alt-sistemlere entegrasyonunu sağlar. Böylece, ana işlemci alt-sistemi boşta olduğunda ve bu yüzden kapatılması gerektiğinde, temel ve basit görevler daha verimli olan ASIC'ler tarafından gerçekleştirilebilir.

Kablosuz algılayıcı düğüm tasarımında kullandığımız işlem birimi *Atmel ATmega328P* mikrodenetleyicidir. ATmega328P, gelişmiş RISC mimarili AVR işlemcisine dayalı düşük güç CMOS 8-Bit mikrodenetleyicidir. Tek saat çevriminde güçlü komutları yürüterek ve sistem tasarımcılarına işlem hızına karşı güç tüketimini optimize etmeye izin vererek, MHz başına 1MIPS'e yaklaşan işi başarır. ATmega328P, algılayıcı düğüm devremizde kullandığımız denetleyici (saat) birime bağlı olarak algılayıcılar tarafından algılanan verileri belirli periyotlarla okur, işler/hesaplar ve EEPROM'a yazar. Algılayıcı düğümler arasında veri iletişimi için senkronizasyon başlatıldığında, EEPROM'daki verileri KAA'a aktarmak üzere XBee 868LP RF iletişim modülüne gönderir. Boşta kaldığı durumlarda en derin uyku moduna geçerek enerji tasarrufu sağlar. ATmega328P mikrodenetleyicisine ait bazı özellikler Tablo 3.1'de sunulmuştur.

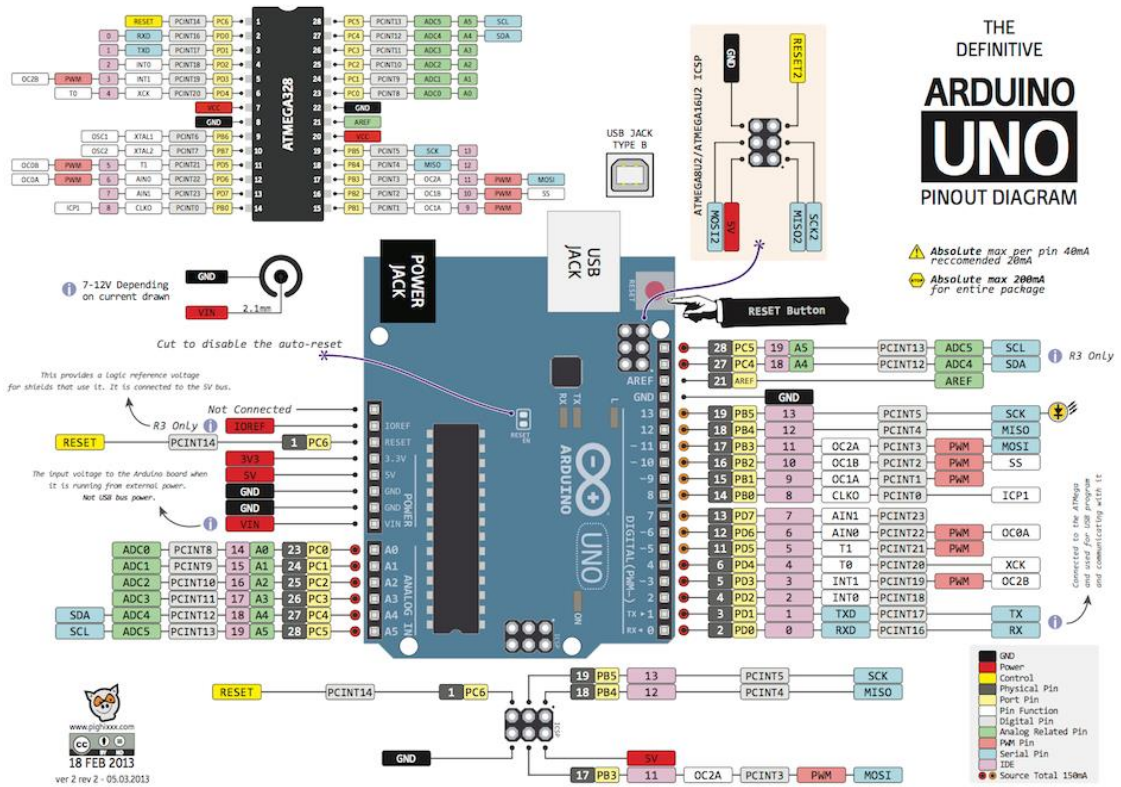
Tablo 3.1: ATmega328P mikrodenetleyici özellikleri.

İşlemci	Yüksek performans, düşük güç AVR 8 Bit mikrodenetleyici	
Hız	8 MHZ ($1.8V < V_{CC} < 2.7V$)	
Program Flaş Hafıza/ EEPROM/SRAM	32 KB/1KB/2KB	
Analog-Dijital-Dönüştürücü	10 Bit ADC (8 kanal)	
Seri İletişim	Programlanabilir seri USART	
Giriş/Çıkış Birimleri	23 programlanabilir GPIO	
Diğer İletişim Arayüzleri	Dijital I/O, I2C, master/slave SPI, byte-oriented 2-wire Serial Interface	
Elektriksel Karakteristikleri	$1.8V < V_{CC} < 2.7V$	
Çalışma Sıcaklığı Aralığı	$-40^{\circ}C - +85^{\circ}C$	
Güç Tüketimi	Aktif mod : 0.2 mA, Uyku modu : 0.1 μA , Güç koruma modu : 0.75 μA	
Diğer Özellikler	6 farklı uyku modu	
	- Boşta kalma	- Güç kapalı
	- ADC gürültü azaltma	- Bekleme
	- Enerji kazanç (güç koruma)	- Uzatılmış bekleme

Mevcut birçok kablosuz algılayıcı düğümde mikrodenetleyici kullanılır. Programlama esnekliği (düşük ya da yüksek seviye programlama dilleri ile programlanabilmesi), kompakt yapısı, küçük boyutu, düşük güç tüketimi, düşük maliyeti, daha az hesaplama

yoğunluğu gibi özelliklerinden dolayı algılayıcı düğüm tasarımlarında mikrodenetleyici en iyi seçenektir. Ayrıca, KAA'larda enerji verimli iletişim protokolleri ve sinyal işleme algoritmaları geliştirmek, dinamik kod kurulumu ve güncelleme gerektirdiğinden mikrodenetleyici kullanmak daha uygundur.

Algılayıcı düğüm devremize ait işlem birimi üzerinde enerji tasarrufuna yönelik uygulamalar geliştirmek için mikrodenetleyici geliştirme kartı olarak *Arduino Uno* kullandık (Şekil 3.2).



Şekil 3.2: ATmega328P mikrodenetleyici ve Arduino Uno platformu.

Arduino Uno [163]; kolay kullanılabilir, platform bağımsız, açık kaynaklı, esnek donanım ve yazılım mimarisine sahip, ATmega328P mikrodenetleyicisine dayalı elektronik devre geliştirme kartıdır. Platform bağımsız (Linux, Mac, Windows işletim sistemleri üzerinde çalışabilir), alt seviyede *gcc* derleyicisi kullanan, Java ile geliştirilen açık kaynaklı bir tümleşik geliştirme ortamına (IDE) sahiptir. *Wiring* tabanlı C/C++ programlama diline çok yakın bir dil ile programlanır. Arduino IDE ortamı ise, *Processing* isimli açık kaynak programlama dili temel alınarak geliştirilir. Arduino kaynak kod dosyalarına (*.ino) *Sketch* (taslak) adı verilir. Arduino Uno platformunu oluşturan temel bileşenler: Arduino geliştirme ortamı; Arduino bootloader (Optiboot),

Arduino kütüphaneleri, AVR Dude (Arduino üzerindeki mikrodenetleyici programlayan yazılım) ve derleyicidir (AVR-GCC). Arduino bütün bu bileşenleri bir araya getirerek tasarımcılara ve programcılara kolay kullanım imkanı sunar. Ayrıntılarla uğraşmadan doğrudan mikrodenetleyicinin programlanmasını sağlar.

İletişim birimi: KAA'daki düğümler arasında veri iletişiminin gerçekleşmesini sağlar. Bitleri RF dalgaları içinde iletilecek duruma dönüştürmek ve diğer uçta onları eski haline çevirmek için gerekli işlemleri uygulayan RF alıcı-verici modül içerir. Haberleşmede, RF alıcı-verici modül 4 farklı işlem modunda çalışır. Bunlar; *gönderme*, *alma*, *boşta kalma* ve *uykudur*. Boşta kalma modunda çalışan alıcı-vericinin güç tüketimi neredeyse alma modundakine eşittir. Bu yüzden, alma veya gönderme işlemi yapmayan alıcı-vericiyi boşta kalma moduna geçirmek yerine kapatmak en iyi çözümdür. Ayrıca, paket iletimi için uyku modundan gönderme moduna geçerken önemli miktarda enerji tüketimi olmaktadır.

Kablosuz algılayıcı düğümlerin birbirleri ile haberleşebilmesinde kullanılabilecek farklı iletişim yöntemleri vardır. Bunlar: *optik iletişim (lazer)*, *kızılötesi (IR)* ve *radio frekansıdır (RF)*. Lazer daha az enerji harcar; ancak görüş alanı gereksinimi, atmosfer koşullarından etkilenme ve tek yönlü olması nedeniyle tercih edilen bir yöntem değildir. Lazerle benzerlik gösteren kızılötesi iletişim de tek yönlü olması ve kısa erişimi (yayım kapasitesi) sebebiyle tercih edilmez. RF iletişim, elektromanyetik dalgalar şeklinde yapılan haberleşme tekniğidir. En önemli problemi anten gereksinimidir. Etkin veri gönderme-alma işlemi için minimum anten uzunluğu $\lambda/4$ (λ , taşıma frekansı dalga boyu) olmalıdır. RF iletişimin avantajları; bütünlük, kullanım kolaylığı ve ticari olarak yaygın olmasıdır. Dikkat edilmesi gereken bir başka unsur, güç tüketimini azaltmak amacıyla modülasyon, filtreleme, demodülasyon gibi işlemlerin yapılması gerekliliğidir.

Kablosuz algılayıcı düğüm tasarımında kullandığımız iletişim birimi, “Kısa Mesafe Erişimli Telsiz Cihazları Hakkında Yönetmelik” [164] dikkate alınarak seçilmiştir. İletişim birimi; Digi firması tarafından üretilen, 868MHz ISM frekans bandında çalışan, yüksek hızda veri alış-verişi yapan, düşük güç tüketimli *XBee 868LP* RF alıcı-vericidir [165]. Bu radyo modülü; KAA uygulamasına bağlı olarak *DigiMesh*, *point-to-point*, *point-to-multipoint*, *peer-to-peer* ağ topolojilerini desteklemekte ve *tekrarlayıcı* gibi çalışabilmektedir (Şekil 3.3). XBee 868LP radyo özellikleri Tablo 3.2’de verilmiştir.

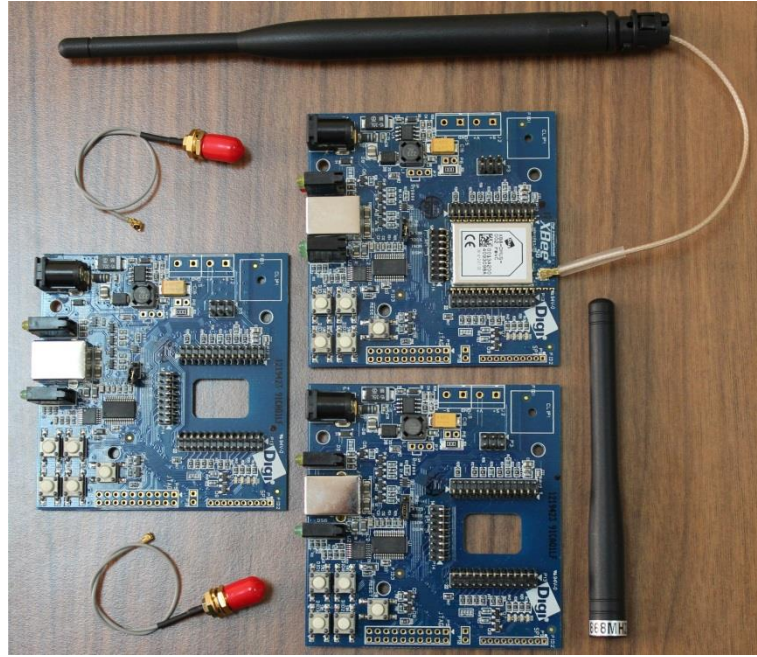


Şekil 3.3: XBee 868LP RF iletişim modülü.

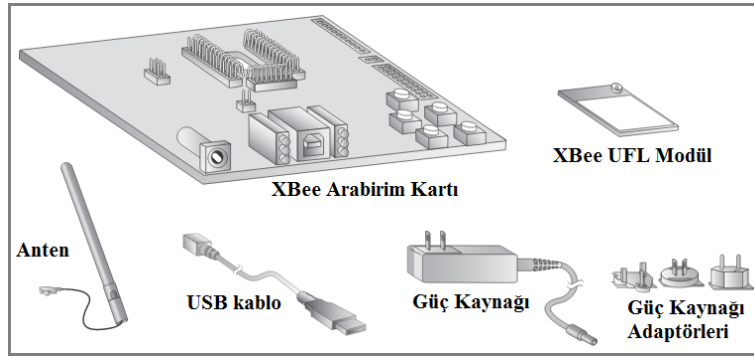
Tablo 3.2: XBee 868LP RF iletişim modülü özellikleri ve performansı.

İşlemci	ADF7023 alıcı-verici, 32 MHz Cortex™-M3 EFM32G230
Frekans Bandı	863 MHz -870 MHz
Anten Seçenekleri	U.FL, RF pad, PCB (PCB anten sadece 10 Kbps veri hızı ile onaylı)
RF Veri Hızı	Düşük: 10 Kbps, Yüksek: 80 Kbps
Kapalı Alan Menzili	2.1 dBi anten ile 112 m'ye kadar, PCB gömülü anten ile 14 m'ye kadar
Açık Alan Menzili	2.1 dBi anten ile 8.4 km'ye kadar, PCB gömülü anten ile 0.64 km'ye kadar
Verici Gücü	12dBm'e kadar (16mW), yazılımla seçilebilir
Alıcı Hassasiyeti	80 Kbps'de -101 dBm, 10Kbps'de -106 dBm
Giriş/Çıkış Birimleri	13 I/O hattı
Analog girişler	10 Bit ADC (4 kanal)
Çalışma Sıcaklığı	-40 °C - +85 °C
Besleme Gerilimi	2.7V-3.6V
Gönderme (TX) akımı	48mA
İdle/Alma (RX) akımı	27mA
Uyku akımı	1.7µA
UART Arabirimi	CMOS serisi UART
UART Veri Hızı	9600-230400 band, yazılımla seçilebilir
SPI Saat Hızı	3.5 MHz'e kadar
Adresleme Seçenekleri	PAN ID ve 64 Bit adres
Şifreleme	128 Bit AES
Ağ Topolojisi	DigiMesh, repeater, point-to-point, point-to-multipoint, peer-to-peer

Çalışmamızda; tasarladığımız KAA için enerji verimliliği, iletişim ve senkronizasyon konularında uygulamalar geliştirmek, performans testleri yapmak, bu iletişim modülünü içeren algılayıcı düğümlerimizin KAA topolojisinde koordinatör, yönlendirici veya uç düğüm gibi çalışmalarını sağlamak amacıyla XBEE 868LP uygulama geliştirme seti kullandık (Şekil 3.4 ve 3.5) [95, 166].



Şekil 3.4: XBee 8686LP RF uygulama geliştirme seti.



Şekil 3.5: XBee 8686LP RF uygulama geliştirme seti temel bileşenleri.

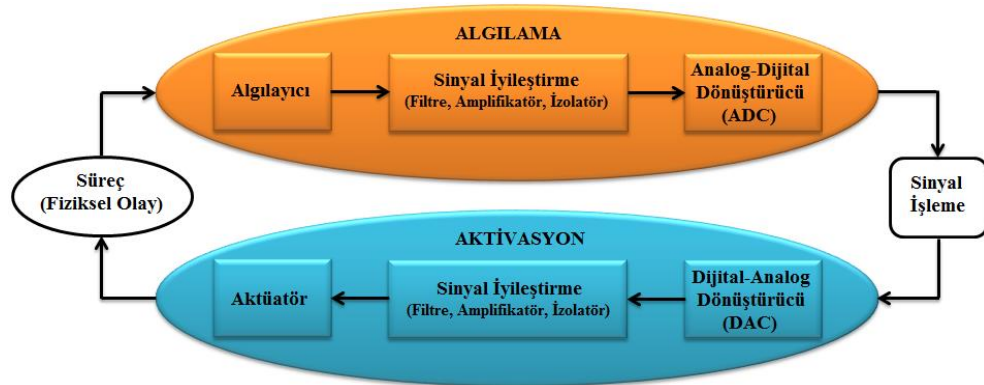
Güç birimi: Kablosuz algılayıcı düğümün en önemli bileşenlerinden biridir. Güç biriminde genellikle pil tercih edilmektedir; ancak, diğer enerji kaynaklarını kullanmak da mümkündür. Pil, şarj edilemeyen ve şarj edilebilen olmak üzere iki tiptir. İçerisinde kullanılan elektromekanik malzemeye göre sınıflandırılır. Örneğin: NiZn (Nikel Çinko), NiCd (Nikel Kadmiyum), NiMh (Nikel Metal Hidrid), LiFePO₄ (Lityum Demir Fosfat), Li-ion (Lityum İyon), Li-PO (Lityum Polimer).

Kablosuz algılayıcı düğümde enerji tüketimi; algılama, veri işleme ve iletişim işlemleri nedeniyle olmaktadır. İletişimde harcanan enerji, algılama ve veri işlemede harcanan enerjiden çok daha fazladır. Dolayısıyla, pil sınırlı kapasitede güce sahip olduğundan, algılayıcı düğümüne ait her bir bileşen tarafından gerçekleştirilen operasyonlar için enerji tasarruflu çalışma modlarına gerek duyulur. Buradan hareketle, tasarladığımız kablosuz

algılayıcı düğümümüzde güç birimi bileşenleri olarak; *şarj edilebilir LiFePO₄ pil (3.7V, 180mAH)*, *pil seviye denetleyici (Maxim Integrated-MAX1704X)* ve *programlanabilir voltaj regülatörü (Texas Instruments-TPS62740, Intersil-ISL9112)* kullandık.

Güç üretici: Pil, kablosuz algılayıcı düğümlerde yaygın olarak tercih edilir. İlave bir jeneratör, daha uzun ağ ömrünün gerekli olduğu uygulamalarda kullanılabilir. Örneğin; dış mekân uygulamalarında güneş panelleri kullanmak iyi bir seçimdir. Benzer şekilde; termal, kinetik ve titreşim enerjileri için güç üretme teknikleri de bulunmaktadır. Geliştirdiğimiz kablosuz algılayıcı düğüm devremizde güç üretici olarak küçük boyutlu bir *güneş paneli (4.2V, 100mAH)* kullandık.

Harekete geçirici: Verilen görevleri gerçekleştirmek gerektiğinde algılayıcı düğümleri hareket ettirmek amacıyla bazen bir harekete geçiriciye ihtiyaç duyulabilir. Mobilite desteği büyük enerji kaynakları gerektirir ve etkin bir şekilde sağlanmalıdır. Harekete geçirici, algılayıcı düğümün hareketlerini kontrol etmek için algılama ünitesi ve işlemci ile yakın etkileşimde çalışır. Birçok KAA, fiziksel dünyayı doğrudan kontrol etmeye yönelik aktüatörlere de sahiptir. Örneğin: bir aktüatör; sıvı ya da gaz akışını denetleyen bir valf, kapı veya pencereyi açıp kapayan bir motor, motor içerisine basılacak yakıt miktarını kontrol eden bir pompa olabilir. Böyle bir kablosuz algılayıcı ve aktüatör ağı (WSAN: Wireless Sensor and Actuator Networks), işlem cihazından (denetim birimi) komutları alır ve bu komutları aktüatör için giriş sinyallerine dönüştürür. Aktüatör daha sonra fiziksel olayla (proses) etkileşime girer. Dolayısıyla, bir kapalı kontrol döngüsü oluşturulur (Şekil 3.6).



Şekil 3.6: Veri toplama ve aktivasyon işlemleri.

Konum bulma sistemi: KAA uygulamalarının, algılama işlemleri ve yönlendirme tekniklerinin çoğu, bir kablosuz algılayıcı düğümün fiziksel konumunu içeren bilgiye sahip olma ihtiyacı duyar. Dolayısıyla, bir algılayıcı düğümün konum bulma sistemi ile donatılması yaygındır. Bu sistem, üst düzey bir algılayıcı düğüm için GPS modülünden oluşabilir veya dağıtık hesaplamalar yoluyla konum bilgisi sağlayan lokalizasyon algoritmalarını gerçekleştiren bir yazılım da olabilir.

İletişim Arayüzleri: Doğru mikroişlemci türünün seçimi kablosuz algılayıcı düğümün performansı ve enerji tüketimi açısından büyük öneme sahip olduğu gibi, düğüm bileşenlerinin mikroişlemci alt-sistemi ile ara bağlantısı da hayati önem taşır. Kablosuz algılayıcı düğümün alt bileşenleri arasında hızlı, etkin ve enerji tasarruflu veri aktarımı, KAA'nın genel verimliliği için çok önemlidir. Ancak, algılayıcı düğüm boyutu, sistem veri yolları üzerine bazı kısıtlamalar getirir. Paralel veri yolu üzerinden iletişim seri veri yoluna göre daha hızlı olduğundan, paralel veri yolu daha fazla yer/alan ihtiyacı duyar. Ayrıca, seri veri yolu tek bir veri hattı kullanırken, paralel veri yolu aynı anda iletilmesi gereken her bit için özel bir hat gerektirir. Algılayıcı düğümün boyutu nedeniyle, paralel veri yolları düğüm tasarımlarında desteklenmez. Dolayısıyla, kablosuz algılayıcı düğüm tasarımlarında seçim; *Seri Çevresel Arabirim (SPI : Serial Peripheral Interface)*, *Genel Amaçlı Girdi/Çıktı (GPIO : General Purpose Input/Output)*, *Güvenli Dijital Girdi/Çıktı (SDIO: Secure Digital Input Output)*, *Ara-Entegre Devre (I²C : Inter-Integrated Circuit)* ve *Evrensel Seri Veri Yolu (USB: Universal Serial Bus)* gibi seri arayüzler arasından yapılır. Düğüm tasarımlarında en sık kullanılan veri yolları; *SPI* ve *I²C* 'dir.

SPI ; yüksek hızlı, tam dubleks senkron seri veri yoludur. Senkron iletişim protokolünü destekler. Resmi bir standardı yoktur. *SPI* veri yolu dört pin tanımlar: *MOSI (Master-Out/Slave-In)*, *MISO (Master-In/Slave-Out)*, *SCLK (Serial Clock)* ve *CS/SS (Chip Select/Slave Select)*. *MOSI* portu, bir aygıt master biçiminde yapılandırıldığında, master'dan slave'e veri iletimi amacıyla kullanılır. Aygıtın slave olarak yapılandırılması durumunda, karşılık gelen master'dan veri almak için *MISO* portu kullanılır. *SCLK*, master tarafından iletimi senkronize etmek için gerekli olan saat sinyalinin göndermede ve slave tarafından okumada kullanılır. Her iletişim master tarafından başlatılır. Master bir aygıt, iletişim kurmak istediği slave'e *CS* portu üzerinden sinyal gönderir. Sonuç olarak, master ve slave zamanlama konusunda anlaşmalıdır. Bunu yapmak için, master,

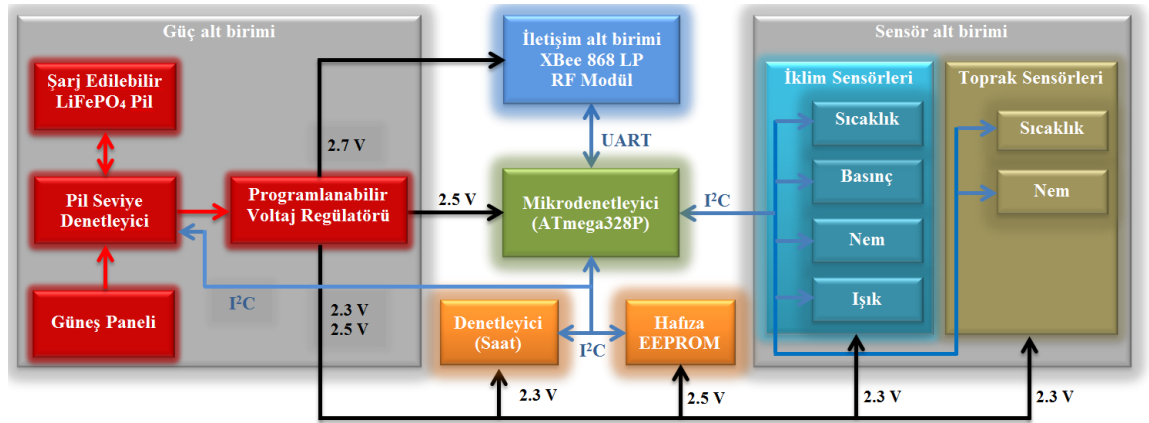
slave'in maksimum saat hızına göre saati ayarlar. Master'ın baud üretici, slave'nin saatini okur ve master'ın saatini okuma hızını içsel olarak tanımlanmış değere bölerek hesaplar. *SPI* tek bir ana veri yolu olduğundan, algılayıcı düğümdeki mikrodenetleyici varsayılan olarak master görevini üstlenir. Bu nedenle, bileşenler birbirleriyle doğrudan haberleşemez ve sadece mikrodenetleyici üzerinden iletişim kurabilirler. Örneğin; böyle bir yapılandırma ile bir ADC, örneklenmiş veriyi doğrudan RAM'e gönderemez.

I^2C ; çoklu-master yarı-dubleks senkron seri veri yoludur. İki çift yönlü hat ve 10-bit adresleme kullanmaktadır. I^2C 'nin amacı, düşük aktarım hızları sağlayarak bir sistemdeki aygıtları bağlamak için maliyetleri en aza indirmektir. I^2C iki hat içerir. Bunlar, seri saat (*SCL-Serial Clock*) ve seri veri analizcidir (*SDA-Serial Data Analyzer*). Her master kendi saat sinyalini ürettiğinden, iletişim aygıtları saat hızlarını senkronize etmelidir. Bunu yapmazlarsa, daha hızlı bir master aygıt üçüncü bir aygıtı veri gönderirken yavaş bir slave aygıt *SDA* hattındaki adresini yanlış tespit edebilir. I^2C , iki hız modu tanımlar: 400kbps'ye kadar bit hızı ile bir hızlı-mod ve 3.4Mbps'ye kadar iletim hızı destekleyen yüksek-hız modu. Standart bir CS ya da SS portu belirtilmediğinden, I^2C kullanan her aygıt türünün bir aygıtlarla iletişim kurabilecek benzersiz bir adresi olmalıdır. I^2C veri iletişimde, master aygıt başlangıç durumunu (veya durumlarını) işaretler ve slave'in adresini iletir. Sonrasında, master okuma veya yazma ilgisini ifade eder. Bu esnada, slave bir alındı bilgisi (*ACK-Acknowledge*) gönderir. Ardından veri ileticisi, daha sonra alıcı verici tarafından kabul edilen 1 baytlık (8 bit) veri gönderir. Hala gönderilecek veri varsa, master göndermeye slave ise alındı bilgisi vermeye devam eder. Son olarak, master iletişimi sonlandırmayı belirtmek için durdurma işaretini/sinyalini gönderir.

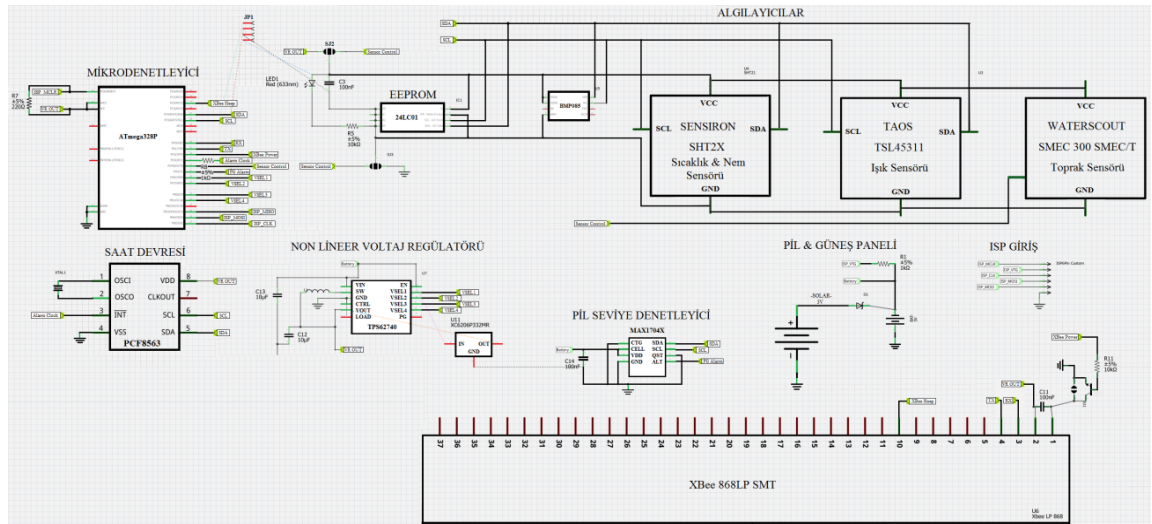
3.1.1. Kablosuz Algılayıcı Düğüm Devresinin Gerçekleştirilmesi

Bir KAA'da kullanılan kablosuz algılayıcı düğümler; son derece düşük güç tüketmeli, yüksek yoğunlukta konuşlandırılmalı, gözetimsiz ve aralarında organize çalışabilmeli, üretim maliyeti düşük, çok hafif, küçük boyutlu, vazgeçilebilir, otonom ve çevreye uyurlanabilir olmalıdırlar. Dolayısıyla, yukarıda tanımlanmış olan kablosuz algılayıcı düğüm mimarisine ait temel bileşenler ve sofistike özellikler, düğüm tasarımlarına ek zorluklar ve kısıtlamalar getirmektedir.

Şekil 3.7’de blok diyagramı ve Şekil 3.8’te devre şeması gösterilen kablosuz algılayıcı düğümümüz; sabit yapıda olup konuşlandırıldığı bölgeye ait iklim koşullarını (sıcaklık, nem, basınç, çiğ, ışık şiddeti) ve toprak durumunu (sıcaklık, nem) otonom algılama, algılanan veriler üzerinde gerekli hesaplamaları yapma, ağdaki diğer düğümlerle iletişime geçme, organize çalışma ve hava koşullarına bağlı senkronizasyon işlemini dinamik bir biçimde gerçekleştirme gibi yeteneklere sahiptir.



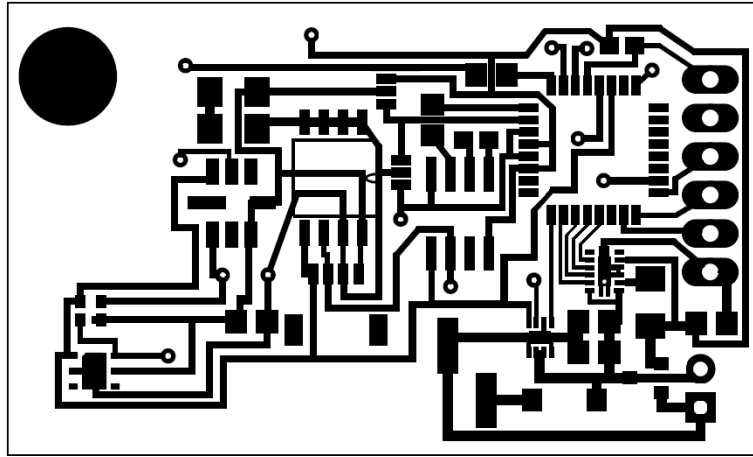
Şekil 3.7: Kablosuz algılayıcı düğümümüzün blok diyagramı.



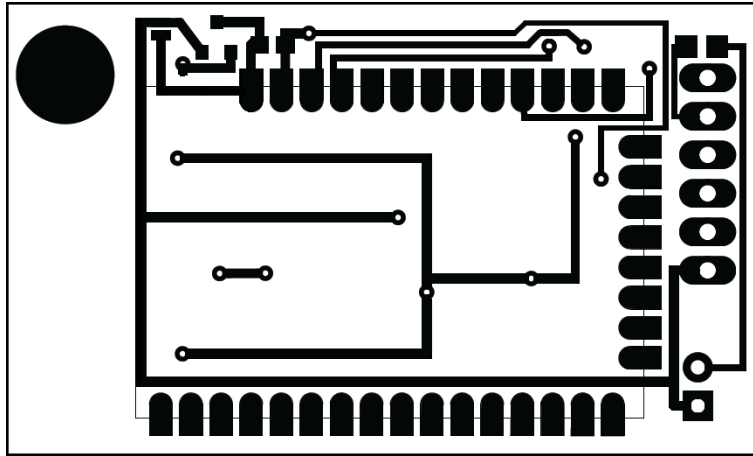
Şekil 3.8: Kablosuz algılayıcı düğümümüzün devre şeması.

Şekil 3.9 ve Şekil 3.10, kablosuz algılayıcı düğümümüze ait baskı devre şemalarını göstermektedir. Algılayıcı düğümümüzün bazı özellikleri aşağıda belirtilmiştir:

- ✓ Küçük boyutlu (3x5cm) ve düşük güç tüketimli olması,
- ✓ Entegre güneş paneli bulundurması (güneşten maksimum oranda enerji sağlama),
- ✓ Uç düğüm, yönlendirici ve koordinatör olarak çalışabilmesi,
- ✓ Hava koşullarına dayalı, enerji verimli senkronizasyon yapması,
- ✓ Üzerinde çalışan yerel dinamik güç yönetim protokolü sayesinde; algılama, hesaplama, depolama, iletişim işlemlerini etkin ve enerji tasarruflu yürütmesi,
- ✓ Kullanılan anten özelliklerine bağlı olarak kısa ya da uzun mesafe veri iletişimi gerçekleştirmesi.



Şekil 3.9: Kablosuz algılayıcı düğümümüzün üst katman baskı devresi.



Şekil 3.10: Kablosuz algılayıcı düğümümüzün alt katman baskı devresi.

3.1.2. Kablosuz Algılayıcı D ğ m i in Geliştirilen G  -Bilin li Sistem

Bir KAA'ın konuşlandırıldığı ortamda enerji tasarruflu  alışması, algılayıcı d ğ mlerin kısıtlı kapasitede g   kaynaklarına sahip olmaları nedeniyle hayati  nem taşıyan bir husustur. D ş k enerji t ketimi t m kablosuz cihazlara  zg  problem iken, KAA'larda sorunun daha da  nem kazanması aŗağıdaki nedenlerle ilgilidir:

1. Y r tt kleri g revlerin (algılama, veri iŗleme,  z-y netim, iletiŗim) karmaŗıklığı ile karŗılaŗtırılırsa, kablosuz algılayıcı d ğ mlerin boyutu y ksek kapasiteli g   kaynaklarına uygunluk a ısından  ok k  kt r.
2. Bir KAA  ok sayıda algılayıcı d ğ mden oluşur. Bu durum, algılayıcı d ğ mlerin pillerini yenisiyle deėiŗtirmek veya tekrar řarj etmek gibi iŗlemleri manuel olarak yapmayı neredeyse imkansız hale getirir.
3. Kablosuz algılayıcı d ğ mlerde meydana gelebilecek arıza, t m aėın erken par alanmasına neden olabilir.
4. Araŗtırma camiası, yenilenebilir enerji kaynakları ve kendi kendine řarj olma mekanizmaları  zerine araŗtırmalar yapmasına raėmen, algılayıcı d ğ m boyutu hala kısıtlayıcı bir fakt rd r ve bu sistemler hen z tam olarak KAA'lar i in kullanılamamaktadır.

G   t ketimi sorununu   zmeye iki a ıdan yaklaŗılabilir. Biri, KAA'ların  zelliklerini hesaba katan enerji verimli iletiŗim protokolleri (kendi kendini organize etme, iŗbirlik i  alışma, k meleme, ortam eriŗimi, y nlendirme) geliŗtirmektir. Diėeri, KAA'larda ortaya  ıkan m srif, gereksiz aktiviteleri tespit etmek ve bunların etkilerini azaltmaktır. M srif ve gereksiz aktiviteler, yerel (algılayıcı d ğ mle sınırlı) ya da global (t m aėı kapsayan) olarak tanımlanabilir. Her iki durumda da bu aktiviteler; tesad fi yan etkiler,  evresel koŗullar, optimal olmayan yazılım ve donanım uygulamaları, hatalı konfig rasyonlar sonucu meydana gelebilir. Bu durumlarla ilgili birka   rnek verecek olursak: Bazı d ğ mler, eriŗemeyecekleri bir aė ile ama sızca baėlantı kurma  abası i erisine girerek pillerini erkenden t ketirler.  nemli miktarda enerji, boŗta kalma durumundaki iŗlem veya iletiŗim alt sistemleri tarafından israf edilir. Ama sızca ortamı algılayan ya da komŗu d ğ mler birbirleri ile haberleŗirken gereksiz dinleme yapan bir d ğ m ciddi miktarda g   t ketir.

VSEL(Voltage Select-Voltaj Seçim) pinleri aracılığıyla çıkışında farklı gerilim değerleri üreten TPS62740 programlanabilir voltaj regülatörü kullandık. TPS62740 entegresinin özellikleri Tablo 3.3’de sunulmuştur.

Tablo 3.3: TPS62740 özellikleri.

Giriş voltajı	2.2V-5.5V arasında (USB portundan ya da ince film güneş modüllerinden çalışabilir)
Çıkış voltajı	1.8V-3.3V arasında 100 <i>mV</i> adımlarda 16 seçilebilir çıkış voltajı
Çıkış akımı	300mA’e kadar
Verimlilik	10 μA çıkış akımında %90’a kadar
Topoloji	RF uyumlu DCS-Control (Direct Control with Seamless Transition into Power Save Mode-Güç Koruma Moduna Hatasız Geçiş ile Doğrudan Kontrol)
Uyumluluk	Şarj edilebilir Li-on piller, li-on kimyali piller, 2 ya da 3 hücreli alkalın piller, USB portu, ince film güneş modülleri ile çalışabilir.
Uygulama alanları	Düşük güç bluetooth, ZigBee, RF4CE, endüstriyel ölçümler, enerji toplama

TPS62740 entegresi; çıkış voltajını ölçeklendirmek, işletim modunu ve çıkış voltajı azami değişim hızını programlamak için kullanılan 4 adet voltaj kontrol pinine (VSEL) sahiptir. Bu pinler, kullanıcı tarafından seçilebilir özellikte olup TPS62740’ın aktif olduğu herhangi bir zamanda okunabilir ve yazılabilir. Tablo 3.4; kablosuz algılayıcı düğümümüzde DGY/DVÖ işlemleri için kullanılan VSEL pin konfigürasyonlarını ve düğümü oluşturan alt birimlerin çalışma gerilimlerini göstermektedir.

Tablo 3.4: Kablosuz algılayıcı düğümümüzde DVÖ konfigürasyonu.

Çıkış Voltajı	VSEL4	VSEL3	VSEL2	VSEL1	Düğüm Alt Birimi
2.0V	0	0	1	0	----
2.1V	0	0	1	1	----
2.2V	0	1	0	0	----
2.3V	0	1	0	1	Algılayıcılar, saat
2.4V	0	1	1	0	----
2.5V	0	1	1	1	EEPROM, işlemci
2.6V	1	0	0	0	----
2.7V	1	0	0	1	XBee 868LP

Program kodu 3.1; kablosuz algılayıcı düğümümüzde DGY işlemini gerçekleştiren yazılımımızda tanımlı AdjustVoltage() isimli DVÖ fonksiyonunu belirtmektedir. Bu fonksiyon, programlanabilir voltaj regülatörü aracılığıyla istenilen işleme (algılama, veri işleme, depolama, iletişim) ait gerekli voltajı dinamik bir biçimde ölçeklendirmektedir.

```

void AdjustVoltage (unsigned new_voltage)
{
    if (new_voltage >= minvolt || new_voltage <= maxvolt)
    {
        unsigned voltage_value = (new_voltage-minvolt)/100;
        digitalWrite (VSEL1, voltage_value&0x01);
        digitalWrite (VSEL2, voltage_value&0x02);
        digitalWrite (VSEL3, voltage_value&0x04);
        digitalWrite (VSEL4, voltage_value&0x08);
    }
}

```

Program kodu 3.1: Dinamik voltaj ölçeklendirme fonksiyonu.

Program kodu 3.2’de fonksiyon tanımlaması ve Şekil 3.12’de akış diyagramı verilen SensorProcess() fonksiyonu; Program kodu 3.1’deki AdjustVoltage() fonksiyonunu kullanarak kablosuz algılayıcı düğümümüzde yürütülen algılama, veri işleme, depolama görevleri için gerekli gücü dinamik olarak kontrol etmektedir.

```

void SensorProcess()
{
    AdjustVoltage(Sensing_Voltage);
    Sensor stSensor;

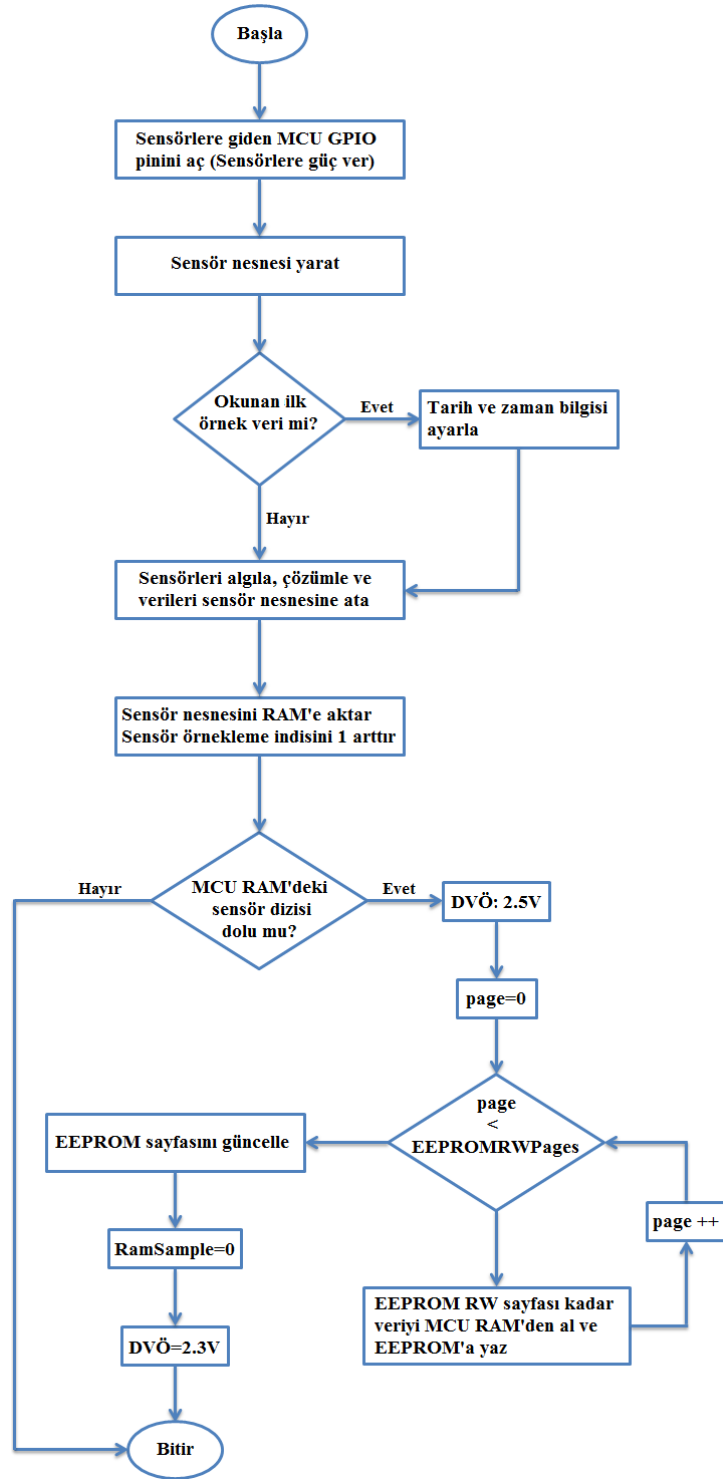
    if (RAMSample == 0 && EEPROMPage == 0)
    {
        rtc.getDate();
        rtc.getTime();
        FirstReadTime.second = rtc.getSecond();
        FirstReadTime.minute = rtc.getMinute();
        FirstReadTime.hour = rtc.getHour();
        FirstReadTime.day = rtc.getDay();
        FirstReadTime.month = rtc.getMonth();
        FirstReadTime.year = rtc.getYear();
    }

    ReadSensors(stSensor);
    memcpy(RAMBuffer + SampleSize*RAMSample, &Sensor, 16);
    RAMSample ++;

    if (RAMSample >= MaxRAMSample)
    {
        AdjustVoltage(EEPROM_Voltage);
        for (uint16_t page = 0; page < EEPROMRWPAGES; page ++ )
            WireEEPROMWritePage(EEPROMPage+page, RAMBuffer+128*page);
        EEPROMPage += EEPROMRWPAGES;
        RAMSample = 0;
        AdjustVoltage(Sensing_Voltage);
    }
}

```

Program kodu 3.2: Algılama, veri işleme ve depolama işlemleri için DGY fonksiyonu.



Şekil 3.12: Algılama, veri işleme ve depolama işlemleri için DGY akış diyagramı.

Program kodu 3.3'te fonksiyon tanımlaması ve Şekil 3.13'de akış diyagramı gösterilen NodeCommunication() isimli fonksiyon; Program kodu 3.1'de sunulan AdjustVoltage() fonksiyonunu kullanarak, kablosuz algılayıcı düğümümüzde veri iletişimi için gerekli voltajı senkronizasyon zamanına bağlı olarak dinamik bir şekilde ayarlamaktadır.

```

void NodeCommunication()
{
    Narcoleptic.delay(SyncDelay);
    AdjustVoltage(Communication_Voltage);
    XBeeOn(true);
    Serial.print("SensorNodeID:");
    Serial.print(SensorNodeID);
    Serial.flush();

    SensorPowerOn(true);
    uint16_t PilVolt = batMon.getmVCell();
    float PilDurumu = batMon.getSoC();
    if (EEPROMPage >= 0)
        SensorPowerOn(false);

    Serial.write((uint8_t*) & PilVolt, 2);
    Serial.write((uint8_t*) & PilDurumu, 4);
    Serial.write((uint8_t*) & FirstReadTime, 6);
    Serial.write((uint8_t*) & SensorReadPeriod, 2);
    uint16_t SampleCount = RAMSample + ((EEPROMPage*128)/SampleSize);
    Serial.write((uint8_t*) & SampleCount, 2);

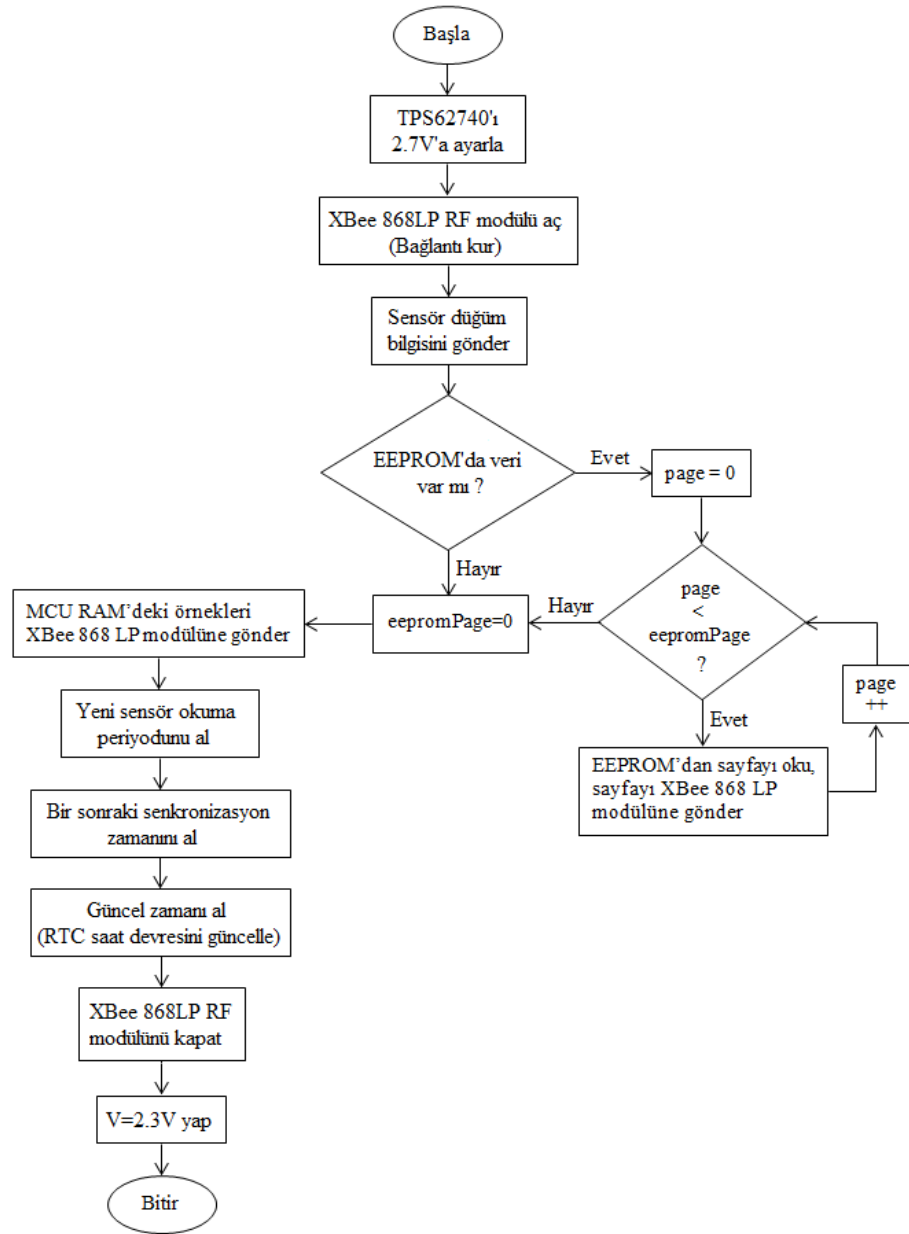
    if (EEPROMPage >= 1){
        SensorPowerOn(true);
        AdjustVoltage(EEPROM_Voltage);
        for(uint16_t page = 0; page < EEPROMPage; page ++){
            for (int i = 0; i < 128; i ++){
                EEPROMPageBuffer[i] = WireEEPROMReadPage(i + page*128);
                Serial.write(EEPROMPageBuffer, EEPROMPageByte);
            }
        }
        SensorPowerOn(false);
    }
    EEPROMPage=0;
    Serial.write(RAMBuffer, SampleSize*RAMSample);
    Serial.flush();
    RAMSample = 0;

    while(Serial.available() < 14){
        delay(500);
        if (alarm_flag) { alarm_flag = 0; SensorTask(); rtc.resetTimer(); }
    }
    Serial.readBytes ((char*) & SensorReadPeriod, 2);
    Serial.readBytes ((char*) & synctime, 6);
    Serial.readBytes ((char*) & datetime, 6 );

    SensorPowerOn(true);
    rtc.setDate(datetime.day , 0 , datetime.month, 0, datetime.year);
    rtc.setTime(datetime.hour, datetime.minute, datetime.second );
    SetSensorReadTime (SensorReadPeriod);
    rtc.setAlarm(synctime.minute, synctime.hour, synctime.day, 99);
    SensorPowerOn(false);
    Narcoleptic.delay(1000);
    XBeeOn(false);
    AdjustVoltage(Sensing_Voltage);
}

```

Program kodu 3.3: Veri iletişimine ait DGY fonksiyonu.



Şekil 3.13: Veri iletişimi için DGY akış diyagramı.

Tasarladığımız KAA'da, algılayıcı düğümler ile alıcı (ağ geçidi, veri toplama merkezi) arasında iletişim gerçekleşeceği zaman, hava koşullarına ve düğümlerin pil seviyelerine göre değişen bir senkronizasyon yapılır. Bu nedenle, değişken senkronizasyon zamanına bağlı olarak iletişim birimini aktif, uyku ya da kapalı modlara getirmek amacıyla algılayıcı düğümümüzdeki alıcı-verici modül için bir anahtar yapısı geliştirdik. Güç-bilinçli yazılımımız; kısa senkronizasyon zamanlarında algılayıcı düğümlerle alıcı arasında iletişim bittiğinde alıcı-vericiyi uyku moduna almakta, uzun senkronizasyon zamanlarında ise anahtarı kapatarak alı-vericinin gücünü kesmektedir. Bu yöntemle, algılayıcı düğümümüzün daha enerji tasarruflu çalışmasına katkı sağladık.

3.2. KOORDİNATÖR DÜĞÜM İÇİN ENERJİ VERİMLİ SİSTEM TASARIMI

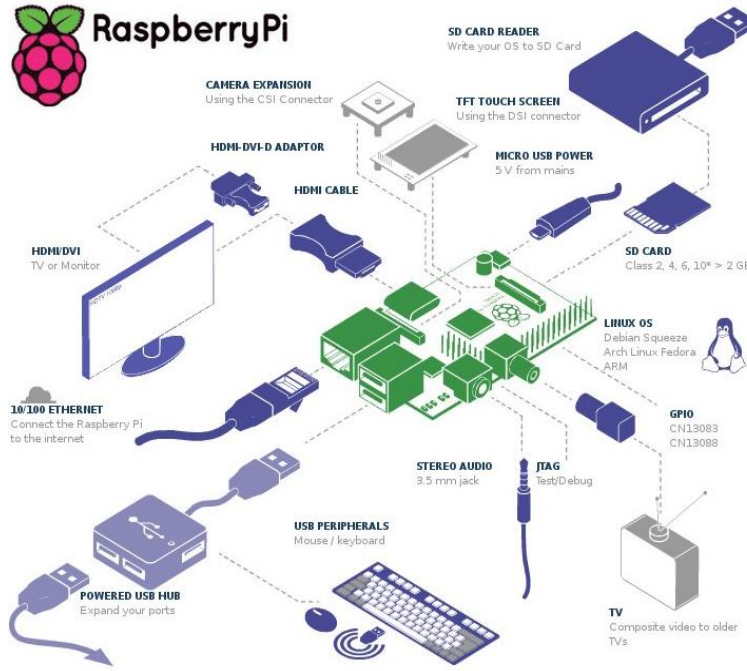
Tasarladığımız KAA'da koordinatör (ağ geçidi) olarak *Raspberry Pi* [167-170] marka tek kartlı bilgisayar kullandık (Şekil 3.14). Koordinatör düğümümüz; küçük boyutlu (kredi kartı büyüklüğünde), düşük maliyetli (\$25), çok fonksiyonlu bir üst düzey platformdur. *Raspberry Pi* üzerinde; Arch Linux işletim sistemi, MariaDB ilişkisel veri tabanı yönetim sistemi ve Apache HTTP sunucu çalışmaktadır.



Şekil 3.14: Koordinatör: “Raspberry Pi” tek kartlı bilgisayar.

Koordinatör; KAA topolojisine ve ağ içerisinde gerçekleştirdiği işlemlere bağlı olarak Bluetooth, XBee, WiFi, 3G iletişim modülleri ve çeşitli bağlantı arayüzleri içermektedir (Şekil 3.15). Algılayıcı düğümler ile koordinatör arasındaki iletişim (senkronizasyon) XBee 868LP radyo aracılığıyla gerçekleşmektedir. Koordinatör, üzerinde bulunan USB WiFi adaptör sayesinde bir kablosuz yerel alan ağına bağlanabilmektedir. İnternet üzerinden koordinatöre erişim sağlamak için koordinatörde 3G modem kullanılmıştır. Kablosuz algılayıcı düğümlerden gelen veriler, koordinatör üzerinde çalışan MariaDB veri tabanı yönetim sisteminde tutulmaktadır. Koordinatör, verileri çok sekmeli iletim yoluyla KAA'daki diğer koordinatörlere veya doğrudan İnternetteki ana veri toplama

merkezine aktarabilmektedir. Koordinatörde yüklü Apache HTTP sunucu, kullanıcıların İnternete bağlı herhangi bir cihaz (Örneğin, cep telefonu, PDA, Laptop vb.) üzerinden koordinatördeki kayıtlı bilgilere gerçek zamanlı erişebilmelerini sağlamaktadır. Özetle: *Raspberry Pi* bilgisayar, tasarladığımız KAA içerisinde; koordinatör düğüm, ağ geçidi, yönlendirici, yerel veri toplama merkezi ve HTTP sunucu olarak görev yapabilmektedir. *Raspberry Pi* tek kartlı bilgisayarın donanım özellikleri Tablo 3.5’de sunulmuştur.

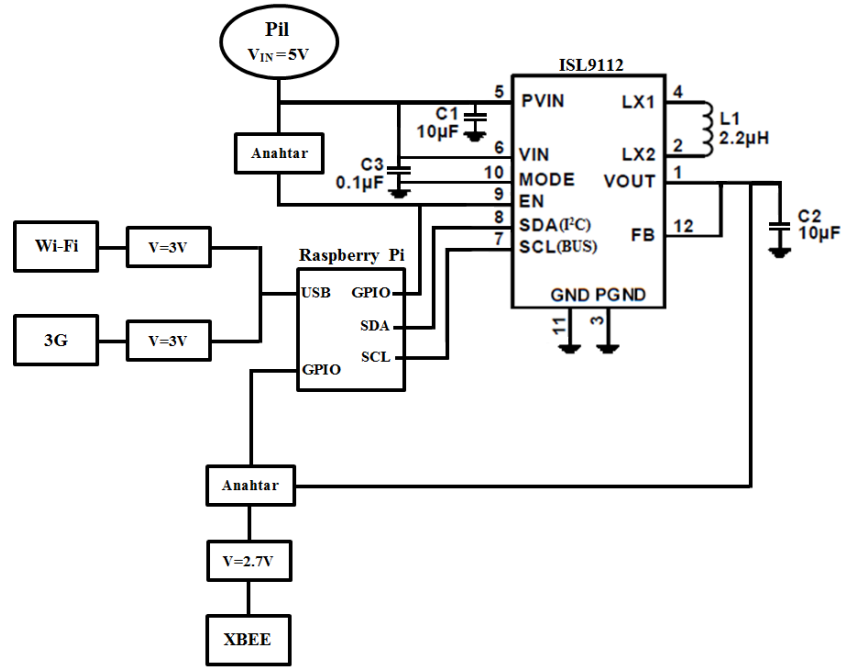


Şekil 3.15: “Raspberry Pi” bağlantı arayüzleri.

Tablo 3.5: “Raspberry Pi” donanım özellikleri.

	Model A	Model B
SoC	Broadcom BCM2835 (CPU, GPU, DSP, SDRAM, USB portu) multimedya işlemci	
CPU	700 MHz ARM1176JZF-S çekirdek (ARM11 ailesi, ARMv6 komut seti)	
GPU	Broadcom VideoCoreIV @ 250 MHz, - OpenGL ES 2.0 (24 GFLOPS) MPEG-2 ve VC-, 1080p30 h.264/MPEG-4 üst düzey kod çözümü ve kodlayıcı	
Hafıza	256 MB (GPU ile paylaşımlı)	512 MB (GPU ile paylaşımlı)
USB 2.0 port	1 (BCM2835 chipinde)	2 (yerleşik entegreli 3-port USB hub ile)
Video Girişi	RPF tasarımı kamera modülü bağlantısına izin veren bir CSI giriş konektörü	
Video Çıkışı	Kompozit RCA (çeşitli PAL ve NTSC standartları), - HDMI, DSI ile raw LCD Panelleri, 640x350 ile 1920x1200 arasında 14 HDMI çözünürlük	
Ses Çıkışı	3.5 mm jak, - HDMI, I ² S ses (imkan dahilinde ses girişi içinde kullanılabilir.)	
Onboard Depolama	SD/MMC/SDIO kart slotu (3.3V kart güç desteği)	
Onboard Ağ	-----	10/100 Ethernet, USB adaptör
Alt-düzye çevre birimleri	8xGPIO, UART, I ² C veri yolu, SPI veri yolu, I ² S audio, +3.3V, +5V, toprak	
Güç değerleri	300mA (1.5W)	700mA (3.5W)
Güç Kaynağı	MicroUSB ya da GPIO başlığı ile 5V	
İşletim Sistemleri	Arch Linux ARM, Debian GNU/Linux, Fedora, Ubuntu, FreeBSD, NetBSD, Raspbian OS, Plan 9, RISC OS, Slackware Linux	

Koordinatör, bir KAA içerisinde en fazla güç harcayan cihazdır. Örneğin; *Raspberry Pi* tek kartlı bilgisayarın güç tüketimi Tablo 3.5’de 1.5W(300mA) olarak görülmektedir. Buradan hareketle, çalışmamızda, kablosuz algılayıcı düğümümüzde enerji tasarrufu sağlamak amacıyla tasarladığımız DGY sistemine benzer bir tekniği koordinatör için de gerçekleştirdik (Şekil 3.16). Sonuç olarak, geliştirdiğimiz yöntem sayesinde koordinatör düğümün de daha az enerji harcamasını sağladık.



Şekil 3.16: Koordinatör düğümde DGY alt sistemi blok diyagramı.

Koordinatör düğümde DVÖ tekniği kullanarak enerji tasarrufu sağlamak amacıyla; I^2C veri yolu arabirimi ile programlanabilen, çıkışında farklı voltaj değerleri üretebilen, Intersil firmasına ait ISL9112 programlanabilir voltaj regülatörü kullandık. ISL9112; çıkış voltajını, işletim modunu ve çıkış gerilimi azami değişim hızını ayarlamakta kullanılan iki adet I^2C erişilebilir kontrol yazmacına sahiptir (Tablo 3.6). Bu yazmaçlar, ISL9112’nin aktif olduğu herhangi bir zamanda okunabilir ve yazılabilir. ISL9112 kapalı olduğu durumda ise I^2C arabirimi üzerinden ISL9112 ile iletişim sağlanamaz. ISL9112’nin I^2C komutlarını geçerli olarak algılayabilmesi için kontrol yazmacının önem derecesi en yüksek biti (MSB), “I2CEN”, 1’e ayarlanmıştır. Eğer bu bit “0” değeri alırsa I^2C komutları kabul edilemez. Bu durumda ISL9112 çıkış voltajı ve işletim modu, fabrika programı tanımlı ayarlara geçer.

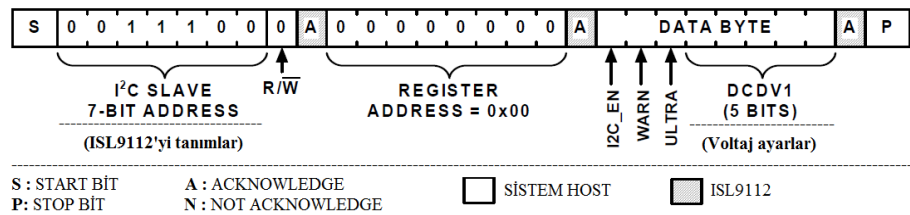
Tablo 3.6: ISL9112 yazmaç adres içeriği.

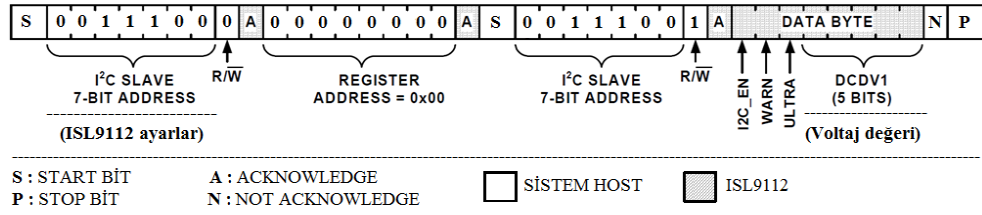
Bit	Adı	Tipi	Reset	Tanım
4:0	DCOUT	R/W	00000	Çıkış voltajı programlama (Tablo 3.7)
5	ULTRA	R/W	0	Ultrasonik mod seçimi. Zorlamalı PWM modda uygulanabilir değil. 0: Ultrasonik özelliği devre dışı. 1: Ultrasonik özelliği etkin.
6	Reserved	R/W	0	----
7	I2CEN	R/W	0	I ² C programlama etkin biti. 0: Cihaz I ² C komutunu yok sayar. En son programlanmış DCOUT ve ULTRA ayarlarını kullanır. Eğer I ² C iletişimi POR modundan itibaren gerçekleşmemişse fabrika programlı geçerli DCOUT ve ULTRA ayarları kullanılır. 1: Cihaz I ² C programlı DCOUT ve ULTRA ayarlarını kullanır.

Tablo 3.7: Koordinatördeki iletişim alt birimleri için DVÖ konfigürasyonu.

DCDOUT [4:0]	Çıkış Voltajı (V)	Koordinatör Düğüm İletişim Alt Birimleri
0b00100	2.3	----
0b00101	2.4	----
0b00110	2.5	----
0b00111	2.6	----
0b01000	2.7	XBee 868LP
0b01001	2.8	----
0b01010	2.9	----
0b01011	3.0	WiFi, 3G

ISL9112, çift yönlü veri yoluna dayalı protokol destekler. Bu protokol, ana veri yoluna veri gönderen herhangi bir cihazı gönderici, ana veri yolundan veri alanı da alıcı olarak tanımlar (Şekil 3.17 ve 3.18). Veri transferini kontrol eden cihaz master (Raspberry Pi) ve kontrol edilen cihazlar (iletişim modülleri) da slave'dir. Master her zaman veri transferini kontrol eder ve hem gönderme hem de alma operasyonları için saat sağlar. Bu nedenle, ISL9112 bütün uygulamalarda slave aygıt olarak çalışır. I²C arabirimi üzerinden gerçekleşen bütün iletişim, ilk önce verinin her bir byte'ın MSB'sini (Most Significant Bit-En Önemli Bit) göndermekle sağlanır.

**Şekil 3.17:** I²C yazma protokolü.



Şekil 3.18: I²C okuma protokolü.

ISL9112'nin çıkış voltajı aralığı 1.9V ile 5.0V arasında değişmektedir. Denklem 3.1, Tablo 3.6 ve 3.7 kullanarak geliştirdiğimiz DGY yazılımı ile koordinatörde bulunan iletişim birimleri (XBee, WiFi, 3G) için gerekli voltajları dinamik olarak kontrol ettik.

$$V_{OUT} = \min_V + (n \cdot 0.1V), \quad 0 \leq n \leq 31 \quad (3.1)$$

Program kodu 3.4, koordinatörde DVÖ işlemini gerçekleştiren fonksiyon tanımlamasını göstermektedir. Bu fonksiyon, I²C arabirimini kullanarak iletişim modüllerine yönelik ilgili voltajları ölçeklendirmektedir.

```
void AdjustComVoltage(unsigned new_voltage)
{
    if(new_voltage >= minvolt || new_voltage <= maxvolt)
    {
        Wire.beginTransmission(ISL9112_ADDRESS);
        int voltage_value = (new_voltage - minvolt)/100;
        Wire.write(0x00);
        Wire.write(_BV(I2CEN) | voltage_value);
        Wire.endTransmission();
    }
}
```

Program kodu 3.4: Koordinatörde DVÖ işlemi yapan fonksiyon.

Program kodu 3.5'de fonksiyon tanımlaması ve Şekil 3.19'da akış diyagramı verilmiş XBeeWiFi3G() fonksiyonu; Program kodu 3.4'deki AdjustComVoltage() fonksiyonunu kullanarak koordinatör düğümde bulunan iletişim modülleri için gerekli gücü dinamik olarak yönetmektedir. Kaynak düğümle hedef sistem (KAA, WLAN, İnternet) arasında iletişim (algılayıcı düğümlerle senkronizasyon, WLAN'a bağlanma, İnternete erişim) gerçekleşeceği zaman alıcı-verici modüller aktif hale getirilir. İletişim sona erdiğinde ise, alıcı-verici birimler kapatılır ve koordinatör minimum güç tüketim modunda çalışır. Böylece, geliştirdiğimiz sistem ile koordinatörün de enerji verimli çalışmasını sağladık.

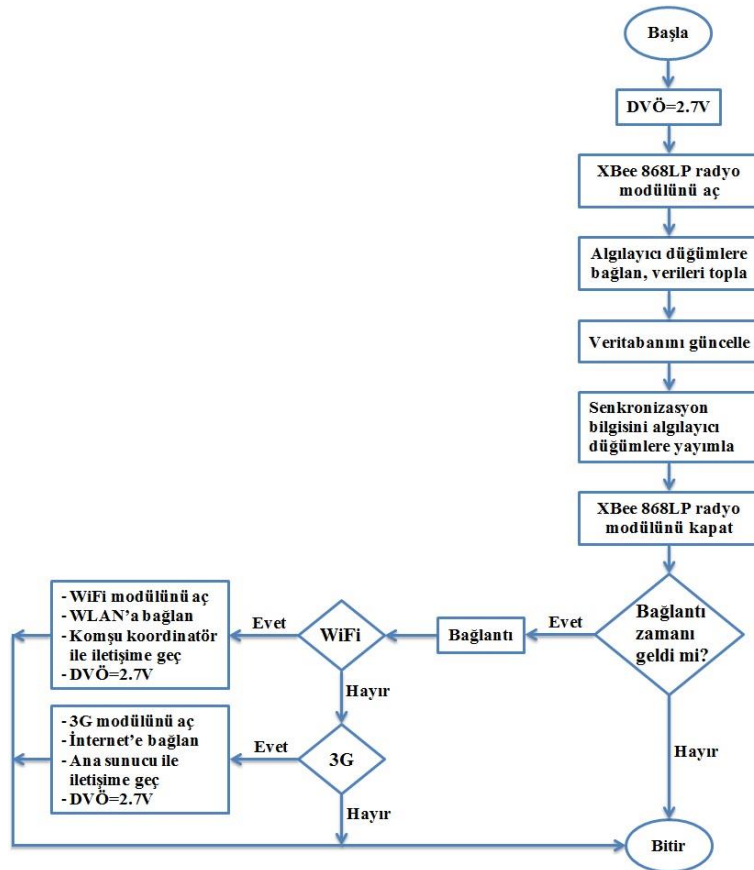
```

void XBeeWiFi3G()
{
    AdjustComVoltage(XBee_Voltage);
    XbeeOn(True);
    ReadClients();
    UpdateDB();
    BroadcastSyncData();
    XBeeOn(False);

    if(isConectionTime()){
        switch(connection){
            case "WiFi":{
                TurnWiFiOn();
                ConnectToWLAN();
                CommunicateWithNeighborCoordinator();
                AdjustComVoltage(XBee_voltage);
                break; }
            case "3G":{
                Turn3GOn();
                ConnectToInternet();
                CommunicateWithMainServer();
                AdjustComVoltage(XBee_voltage);
                break; }
            default: break;}}
}

```

Program kodu 3.5: Koordinatördeki farklı iletişim modülleri için DGY fonksiyonu.



Şekil 3.19: Koordinatördeki iletişim modüllerine yönelik DGY/DVÖ akış diyagramı.

3.3. KAA İÇİN HAVA TAHMİNİNE DAYALI, BULANIK MANTIK TABANLI, ENERJİ VERİMLİ SENKRONİZASYON PROTOKOLÜ

Dağıtık sistemlerde her kablosuz algılayıcı düğüm kendi saatine ve zaman kavramına sahiptir. Düğümler arasında ortak bir zaman ölçeği; fiziksel dünyaya ait olaylar arasındaki nedensel ilişkileri belirlemek, gereksiz algılayıcı verilerinin elenmesini desteklemek, KAA operasyonunu kolaylaştırmak için önemlidir. KAA'daki her düğüm bağımsız çalıştığından ve kendi saatine dayandığından, farklı algılayıcı düğümlerin saat okumaları da farklı olmaktadır [24]. Bu rastgele farklılıklara (örneğin; faz kaymaları) ek olarak, farklı algılayıcıların saatleri arasındaki boşluk/açıklık osilatörlerin değişken sapma hızları sebebiyle daha fazla artmaktadır. Dolayısıyla, zaman senkronizasyonu, algılama zamanlarının anlamlı bir şekilde mukayese edilebilir olmasını sağlamak adına gereklidir. Araştırmalarda kablolu ağlara yönelik zaman senkronizasyonu teknikleri önemli miktarda ilgi çekmiş olsa da algılama ortamlarının yarattığı benzersiz zorluklar nedeniyle KAA'lar için uygun değildir. Bu zorluklar arasında; büyük ölçekli KAA'lar, kendi kendine konfigürasyon, sağlamlık/dayanıklılık gerekliliği, algılayıcı hareketlilik potansiyeli ve enerji tasarrufu ihtiyacı bulunmaktadır [171-173].

Donanım osilatörlerine dayalı bilgisayar saatleri tüm hesaplama cihazlarının temel bileşenleridir. Tipik bir saat, kuvars-stabilizeli bir osilatörden ve kuvars kristalinin her salınımı ile azaltılan bir sayaçtan oluşur. Sayaç değeri ne zaman sıfıra gelse asıl değerine yeniden ayarlanır ve bir kesme üretilir. Her kesme veya saat tıklaması, uygun bir API(Application Programming Interface-Uygulama Programlama Arayüzü) kullanan uygulamalar tarafından okunabilen ve kullanılabilen bir yazılım saatini (başka bir sayaç) artırır. Bu nedenle, yazılım saati algılayıcı düğüm için bir yerel saat sağlar [24, 174].

Harici ve dahili olmak üzere iki çeşit senkronizasyondan bahsedebiliriz [174]. Harici senkronizasyon, tüm düğüm saatlerinin harici zaman kaynağı (veya referans saat) kullanılarak senkronize edilmesidir. Harici referans saati, UTC (Coordinated Universal Time-Koordineli Evrensel Zaman) gibi hassas gerçek zaman standardıdır. Dahili senkronizasyon ise, tüm düğüm saatlerinin harici referans saati desteği olmadan birbirleriyle senkronize edilmeleri anlamına gelir [174]. Dahili senkronizasyonun amacı, herhangi bir harici referans zamanından farklı olsa bile ağdaki tüm düğümler üzerinde tutarlı bir zaman görünümü sağlamaktır. Harici senkronizasyon, hem harici kaynak ile

senkronizasyon hem de ağdaki tüm düğüm saatleri arasında tutarlılık sağlar. Düğümler harici referans saatine senkronize edildiğinde, bir saatin doğruluğu, referans saate göre bir saatin maksimum ofsetini açıklar. Bir ağdaki düğümler dahili olarak senkronize edildiğinde, hassasiyet, ağdaki herhangi 2 saat arasında maksimum ofseti gösterir [174].

Zaman senkronizasyonu, dağıtık sistemlerde birçok uygulama ve hizmet için gerekli bir servistir. Hem kablolu hem de kablosuz sistemlerde zaman senkronizasyonuna yönelik çok sayıda protokol önerilmiştir. Örneğin, NTP (Network Time Protocol-Ağ Zaman Protokolü) [175]; yaygın olarak konuşlandırılan, ölçeklenebilir, sağlam ve kendinden ayarlanabilen bir senkronizasyon yaklaşımıdır. Özellikle GPS (Global Positioning System-Küresel Konumlandırma Sistemi) ile birlikte kullanılarak birkaç mikrosaniye içinde doğruluk sağladığını ortaya koymuştur. Ancak, NTP gibi yaklaşımlar KAA'ların kendine has özellikleri ve kısıtlamaları nedeniyle bu ağlara uygun değildir [176].

3.3.1. Senkronizasyon Sebepleri

Bir KAA'daki algılayıcı düğümler; fiziksel dünyadaki nesneleri, aktiviteleri, olayları izler ve bunlara ait verileri ilgili birimlere rapor eder. Bu nedenle, olaylar hakkındaki zamansal bağıntıları/ilişkileri ve doğru mantıksal düzeni saptamak çok önemlidir.

Aynı ya da ilişkili olayları izleyen çoklu algılayıcı düğümlerden gelen verilerin toplanması, aynı ya da benzer algılayıcı verilerinin önlenmesi, kritik olaylara daha kısa tepki süreleri ve kaynak gereksinimlerinin azaltılması (örneğin, enerji tüketimi) veri füzyonunun amaçları arasındadır. Dolayısıyla, senkronizasyon, KAA'larda veri füzyonu için önemli bir gereksinimdir. Ayrıca; haberleşme protokolleri, güvenlik, veri tutarlılığı, koşut zamanlılığı denetimi gibi yapıları içeren dağıtık sistemlere ait çeşitli uygulama ve algoritmalarda senkronizasyon gereklidir [24].

TDMA (Time Division Multiple Access-Zaman Bölmeli Çoklu Erişim) gibi ortam-erişim katmanı protokolleri, çoklu cihazların ortak bir iletişim kanalına erişimi paylaşımlarına izin verir. Zaman kablosuz cihazlara tahsis edilmiş slotlara bölünür ve her bir slot yalnızca tek bir kablosuz ağıta aittir [177]. TDMA tabanlı yaklaşımların avantajları; ortam erişiminin öngörülebilirliği (her düğümün bir ya da daha fazla aralıklı olarak yinelenen zaman dilimi süresince veri iletmesine izin verilir) ve bu algoritma ile sağlanan enerji verimliliğidir (bir düğüm, bir slot süresince veri gönderici ya da alıcı

olmadığı takdirde güç tasarruflu uyku moduna girer) [178]. Ancak, TDMA uygulamak için düğümler zamanın ortak bir görünümünü paylaşmalıdır, yani düğümlerin her bir slotun başlangıç ve bitişinin tam olarak farkında olması gerekir.

Enerji tasarrufu hususunda birçok KAA, algılayıcı düğümleri seçime bağlı olarak kapatmak ya da onları düşük güçlü uyku modlarına sokmak için ağa izin veren uyku/uyanma protokollerine dayanmaktadır. Burada, komşu düğümlerin aralarında iletişim sağlamak amacıyla uyanma periyodlarında örtüşmekten emin olmak için ne zaman uyku moduna gireceklerini ve ne zaman uyanacaklarını düğümlerin bilmesi açısından algılayıcılar arasındaki zamana bağlı koordinasyon önemlidir [24].

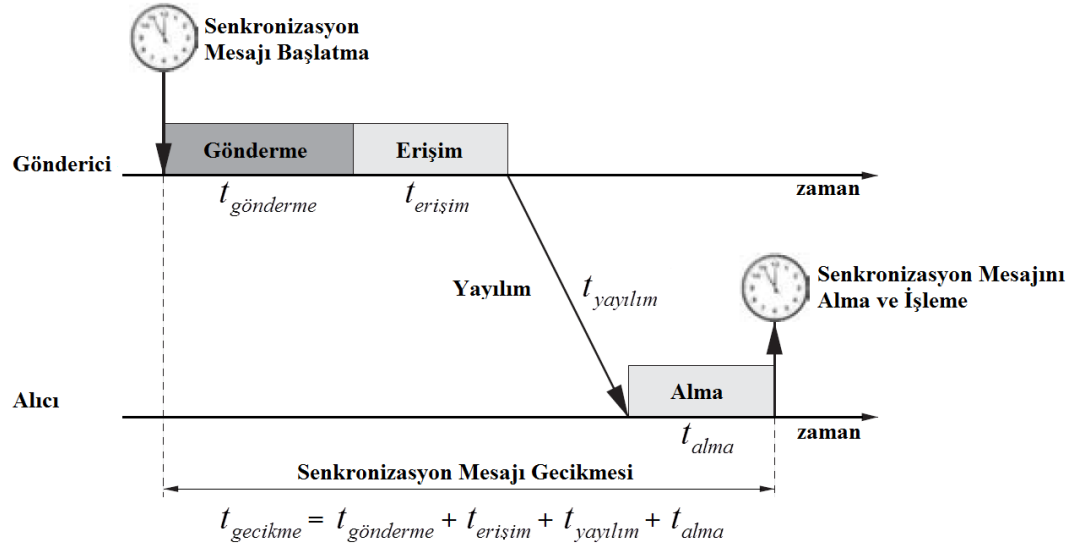
Son olarak, KAA'larda lokalizasyon, kablosuz algılayıcı düğümleri ya da bu düğümlerin izledikleri nesneleri doğru konumlandırmak için gereklidir. Birçok lokalizasyon tekniği, düğümler arasındaki mesafeleri tahmin etmede mesafe tayini teknolojilerine dayanır. Dolayısıyla, radyo veya akustik sinyallerin uçuş-zamanı ölçümlerinde senkronizasyon gerekir.

3.3.2. Senkronizasyonda İletişim Gecikmesinin Deterministik Olmaması

Senkronizasyon, KAA uygulamaları için hayati bir öneme sahiptir. KAA'ların dağıtık yapısının, algılayıcı düğümlerin donanım ve yazılım kısıtlamalarının yarattığı benzersiz zorluklar KAA'lara özel senkronizasyon çözümleri gerektirir. Önemli zorluklardan biri, kablosuz yayın kanalının etkisidir. Senkronizasyon sadece düğümler arasındaki iletişim yoluyla gerçekleştiğinden, kablosuz iletişim kanalının etkileri senkronizasyon protokolü tasarımı dikkate alınmalıdır. Kablosuz iletişim iki düğüm arasındaki gecikmeye rastgelelik getirir. İletişim gecikmesinin deterministik olmaması erişilecek hassasiyete ciddi katkıda bulunur. Şekil 3.20, senkronizasyon mesajlarında karşılaşılan gecikmeyi göstermektedir. Buna göre, iki düğüm arasındaki gecikmenin dört temel bileşeni vardır:

1. *Gönderme gecikmesi*: Göndericinin, senkronizasyon mesajını oluşturmak ve mesajı ağ arabirimine aktarmak için harcadığı zamandır. Bu gecikme; işletim sistemi davranışı (sistem çağrı arayüzü, içerik anahtarları), ağ protokol yığını ve ağ aygıt sürücüsü tarafından kaynaklanan gecikmeleri kapsamaktadır. Gönderme gecikmesi, gömülü sistemdeki her donanım ve yazılım bileşeni arasında farklılık gösteren etkileşimler sebebiyle deterministik değildir.

2. *Erişim gecikmesi*: Kablosuz kanalın yayın yapan yapısı gereği, paket hazırlandıktan ve alıcı-vericiye aktarıldıktan sonra da ek bir gecikme meydana gelmektedir. Erişim gecikmesi, fiziksel kanala erişmek için gönderici tarafından harcanan zamandır ve çoğunlukla kullanılan ortam erişim kontrolü (MAC) protokolü ile belirlenir.
3. *Yayılım gecikmesi*: Bir düğüm kanala eriştikten sonra, senkronizasyon paketi iletilir ve paketin amaçlanan alıcıya ulaşması bir miktar zaman alır. Mesajın göndericiden alıcıya ulaşması için gereken bu gerçek süreye yayılım gecikmesi denir. Düğümler aynı fiziksel ortamı paylaştığı zaman gecikme süreleri çok küçük olur ve kritik yol analizinde genellikle göz ardı edilebilir.
4. *Alma gecikmesi*: Alıcı aygıtın; ortamdan mesajı almak, mesajı işlemek (ADC işlemini gerçekleştirmek, işletim sistemini haberdar etmek) ve mesajın alındığını göndericiye bildirmek için harcadığı süredir. Gönderici bildirimi genellikle kesmeler yoluyla olur ve bu sayede yerel saat (mesaj varış saati) okunabilir. Sonuçta, alma zamanı gönderme zamanından çok daha kısa olma eğilimi gösterir.



Şekil 3.20: Senkronizasyon mesajında yaşanan uçtan uca gecikme.

İletişim gecikmesine katkı yapan bu dört ana bileşen, senkronizasyonda kritik yol olarak adlandırılır. Kritik yol, doğada deterministik olmayan bir yapıdadır ve dolayısıyla NTP tarafından kullanılan geleneksel ofset tahmin yöntemlerini kullanmada önemli bir sorun oluşturur. Literatürde, KAA'lar için geliştirilen birçok senkronizasyon protokolü (Reference-Broadcast Synchronization [179], Tiny-sync and Mini-sync Protocols [180],

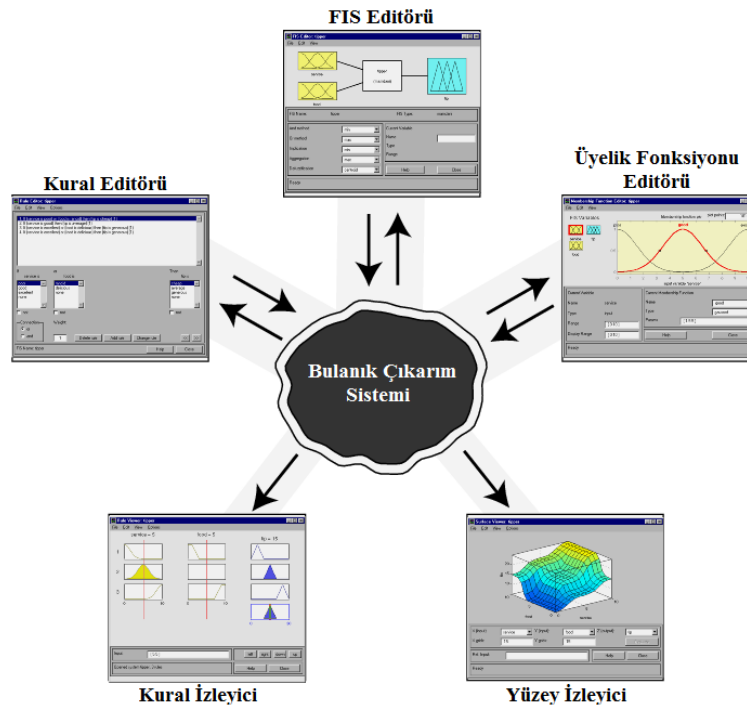
Lightweight Tree-Based Time Synchronization [181], Asymptotically Optimal Time Synchronization [182], Flooding Time Synchronization Protocol [183], Adaptive Clock Synchronization [184], TSync: Time Synchronization Service [185], Time-Diffusion Synchronization Protocol [186], Rate-Based Diffusion Protocol [187], Timing-Sync Protocol for Sensor Networks [172]) bu rastgele gecikmenin etkilerini en aza indirmeyi hedefleyen alt düzey teknikler uygulamaktadır. Örneğin; MAC katmanı zaman damgası, gönderme ve alma gecikmelerini sırasıyla gönderici ve alıcı için azaltabilir.

KAA'lar, her bir algılayıcı düğümün dahili operasyonları için kendi yerel saati ile donatıldığı dağıtık sistemlerdir. Algılama, veri işleme ve iletişim de dahil olmak üzere düğümlerin operasyonu ile ilgili her olay, yerel saat tarafından kontrol edilen zamanlama bilgisi ile ilişkilidir. Kullanıcılar algılayıcı düğümlerden gelen ortak bilgilerle ilgilendiklerinden, her bir düğümdeki veri ile ilişkili zamanlama bilgisinin tutarlı olması gerekir. Ayrıca, KAA, algılayıcı düğümleri koordine etmeli ve fiziksel çevreyi doğru bir şekilde modelleyebilmek için dağıtık düğümler tarafından algılanan olayları düzenleyemebilmelidir. Bu nedenle, senkronizasyon KAA'lardaki iletişimin önemli bir parçasıdır.

3.3.3. Senkronizasyon Protokolü Tasarımı

Tasarladığımız KAA'daki algılayıcı ve koordinatör düğümlerimiz, üzerlerine monte edilmiş güneş panelleri aracılığıyla güneşten enerji sağlayarak şarj olan $LiFePO_4$ pillerden güç almaktadır. Dolayısıyla, KAA'nın konuşlandırıldığı bölgedeki hava durumu, düğümlerin etkin/verimli çalışabilmelerinde çok önemli bir rol oynamaktadır. Buradan hareketle, çalışmamızda; tasarladığımız KAA için hava tahmine dayalı, enerji verimli, bulanık mantık tabanlı bir senkronizasyon protokolü geliştirdik. Protokolümüz; kablolu algılayıcı düğümlerimizin pil seviyelerine, KAA'nın konuşlandırıldığı bölgedeki hava koşullarına, ileriki günlere ait hava tahminlerine, bulut yoğunluk oranlarına (bulutluluk miktarı) ve günlük ortalama güneşlenme sürelerine bağlı olarak KAA'daki her bir düğümü disipline etmektedir. Ayrıca, algılayıcı düğümlerimizin kendi uyku modlarını düzenleyerek enerji verimli çalışmalarına, KAA'da yerel ve global güç yönetimine katkı sağlamaktadır. Sonuçta, gerçekleştirdiğimiz senkronizasyon protokolü, kendine özgü yapısal özellikleri dikkate alındığında, literatürde bulunan KAA'lar için geliştirilmiş diğer senkronizasyon protokollerinden farklılık göstermektedir.

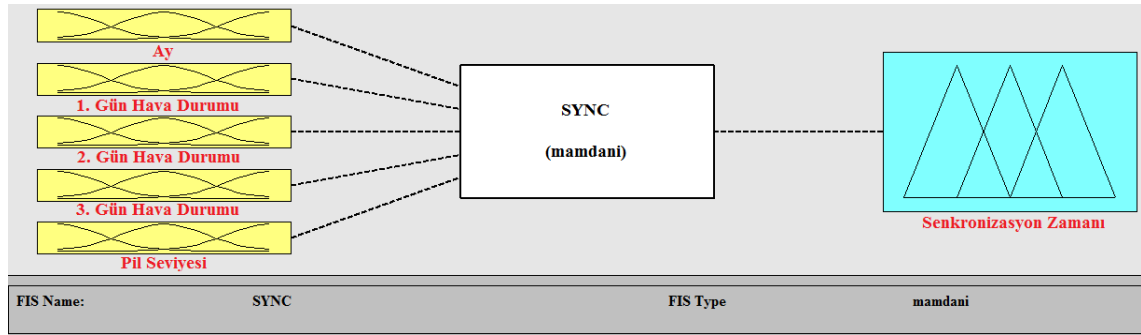
Senkronizasyon protokolünün gerçekleştirilmesinde; *Matlab programlama dili*, *Matlab bulanık mantık araç kutusu*, *Matlab FIS (Fuzzy Inference System-Bulanık Çıkarım Sistemi) editörü*, *Matlab GUI (Graphical User Interface-Grafik Kullanıcı Arabirimi)*, *Arch Linux işletim sistemi*, *Python programlama dili*, *MariaDB ilişkisel veri tabanı yönetim sistemi* kullandık. Matlab bulanık mantık araç kutusunda bir sistemi kurmak, kontrol etmek ve gözlemek için beş temel GUI aracı vardır. Bunlar: FIS, üyelik fonksiyonu editörü, kural editörü, kural izleyici ve yüzey izleyicidir (Şekil 3.21). Bu GUI'ler dinamik olarak birbirine bağlıdır.



Şekil 3.21: Matlab bulanık çıkarım sistemi.

FIS editörü, sistemin en üst seviyesindeki işlemleri yapar. Giriş-çıkış değişkenleri ve isimleri burada tanımlanır. Üyelik fonksiyonu editörü, her bir değişkenle ilgili üyelik fonksiyonlarının oluşturulmasını sağlar. Kural editörü; sistem davranışını belirleyen kuralları tanımlamak ve bunlar üzerinde düzenlemeler yapmak için kullanılır. Kural ve yüzey izleyici, FIS'i gözlemler. Kural izleyici; hangi kuralların aktif olduğunu, her bir üyelik fonksiyonunun sonucu nasıl değiştirdiğini gösterir. Yüzey izleyici; çıkışın herhangi bir veya iki çıkış değerine bağlılığını gösterir, yani sistemin çıkış yüzeyini oluşturur ve haritasını çizer.

Senkronizasyon protokolü tasarımı, sisteme verilen her giriş değişkeni için üyelik fonksiyonu oluşturulmuştur. Kullanılan üyelik fonksiyonları gauss fonksiyonlarıdır. FIS ortamının kullanıma sunduğu başka üyelik fonksiyonları da mevcuttur. Senkronizasyon protokolümüze ait giriş (*ay*, *1. gün hava durumu*, *2. gün hava durumu*, *3. gün hava durumu*, *pil seviyesi*) ve çıkış (*senkronizasyon zamanı*) değişkenleri, Şekil 3.22’de görüldüğü gibi belirlenip adlandırılmıştır. FIS ortamında, her bir ay için günlük hava durumu ve pil seviyesi giriş değişkenleri tanımlanmıştır. “Ay” giriş değişkenine ait üyelik fonksiyonları oluşturulurken Tablo 3.8 ve 3.9’daki günlük ortalama güneşlenme süreleri ve ağırlık değerleri göz önüne alınmıştır.



Şekil 3.22: Senkronizasyon protokolü için giriş ve çıkış değişkenleri.

Tablo 3.8, T.C. Orman ve Su İşleri Bakanlığı Meteoroloji Genel Müdürlüğü web sitesinde bulunan İstanbul ilinin aylara göre günlük ortalama güneşlenme sürelerini göstermektedir. Güneşlenme süresi ile güneş panellerinin güneşten sağladığı enerji arasında doğru orantı vardır. Güneşlenme süresinin artmasıyla algılayıcı ve koordinatör düğümler üzerinde bulunan güneş panellerinin güneşten sağladığı enerji de güneşlenme süresine bağlı olarak artacaktır. Tablo 3.8’de; İstanbul ili için güneşlenme süresinin en az olduğu aylar Aralık (2.2 saat) ve Ocak (2.3 saat), güneşlenme süresinin en fazla olduğu aylar ise Haziran (10 saat) ve Temmuz’dur (10.4 saat). Güneş panellerinin güneşten sağlayacağı enerji dikkate alındığında, minimum güneşlenme süresi Aralık ayında olmaktadır. Dolayısıyla, Aralık ayına ait ortalama güneşlenme süresine ağırlık değeri olarak 1 verilmiştir. Aralık ayı ortalama güneşlenme süresinin diğer ayların ortalama güneşlenme sürelerine bölünmesiyle her bir ayın ortalama güneşlenme süresinin ağırlık değeri hesaplanmıştır. Böylece, aylara ait günlük ortalama güneşlenme süreleri [0-1] aralığına ölçeklendirilmiştir.

Tablo 3.8: İstanbul ili için ortalama güneşlenme süreleri.

ID	Aylar	Ortalama Güneşlenme Süresi	Ağırlık (ω)	ID	Aylar	Ortalama Güneşlenme Süresi	Ağırlık (ω)
1	Ocak	2.3 saat	0.956	7	Temmuz	10.4 saat	0.211
2	Şubat	3.1 saat	0.709	8	Ağustos	9.5 saat	0.231
3	Mart	4.3 saat	0.511	9	Eylül	8.1 saat	0.271
4	Nisan	6.0 saat	0.366	10	Ekim	5.3 saat	0.415
5	Mayıs	8.1 saat	0.271	11	Kasım	3.4 saat	0.647
6	Haziran	10.0 saat	0.220	12	Aralık	2.2 saat	1.000

Örnek ağırlık hesaplama: İlk 5 ay için günlük ortalama güneşlenme sürelerine yönelik ağırlık değerleri aşağıda hesaplanmıştır:

$$\omega(\text{Ocak}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Ocak})} = \frac{2.2}{2.3} = 0.956$$

$$\omega(\text{Şubat}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Şubat})} = \frac{2.2}{3.1} = 0.709$$

$$\omega(\text{Mart}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Mart})} = \frac{2.2}{4.3} = 0.511$$

$$\omega(\text{Nisan}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Nisan})} = \frac{2.2}{6.0} = 0.366$$

$$\omega(\text{Mayıs}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Mayıs})} = \frac{2.2}{8.1} = 0.271$$

Tablo 3.9; Boğaziçi Üniversitesi Kandilli Rasathanesi ve Deprem Araştırma Enstitüsü Astronomi Laboratuvarı web sitesinde yer alan, İstanbul ili için yıllık Güneş doğuş-batış zamanlarına ait bilgiler kullanılarak hazırlanmıştır. Güneş'in doğuş ve batış saatleri arasındaki zaman farkı dakika cinsinden hesaplanarak her ayın ortalama güneşlenme süresi bulunmuştur. Tablo 3.9'dan anlaşılabacağı üzere, minimum ortalama güneşlenme süresi Aralık (565 dk.) ayında olmaktadır. Aralık ayı ortalama güneşlenme süresine ağırlık değeri olarak 1 verilmiştir. Aralık ayı ortalama güneşlenme süresinin her bir ayın ortalama güneşlenme süresine bölünmesiyle, diğer ayların ortalama güneşlenme sürelerinin ağırlık değerleri hesaplanmıştır. Dolayısıyla, aylara ait ortalama güneşlenme süreleri [0-1] aralığına ölçeklendirilmiştir.

Tablo 3.9: İstanbul ili için Güneş doğuş-batış saatleri arasındaki zaman farkı.

ID	Aylar	Ortalama Güneşlenme Süresi	Ağırlık (w)	ID	Aylar	Ortalama Güneşlenme Süresi	Ağırlık (w)
1	Ocak	584.064 dk.	0.967	7	Temmuz	894.516 dk.	0.631
2	Şubat	646.620 dk.	0.873	8	Ağustos	832.870 dk.	0.678
3	Mart	726.161 dk.	0.778	9	Eylül	754.933 dk.	0.748
4	Nisan	808.133 dk.	0.699	10	Ekim	673.225 dk.	0.839
5	Mayıs	877.612 dk.	0.643	11	Kasım	601.866 dk.	0.938
6	Haziran	912.533 dk.	0.619	12	Aralık	565 dk.	1.000

Örnek ağırlık hesaplama: İlk 5 aya ait günlük ortalama güneşlenme süreleri için ağırlık değerleri aşağıda hesaplanmıştır:

$$\omega(\text{Ocak}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Ocak})} = \frac{565}{584.064} = 0.967$$

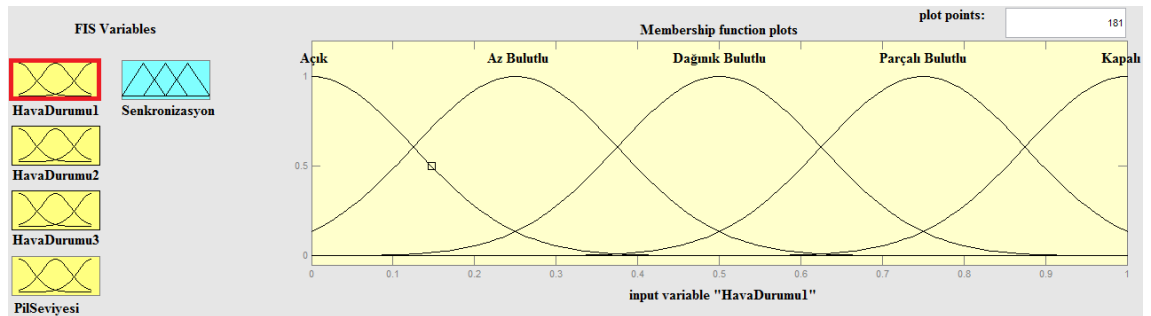
$$\omega(\text{Şubat}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Şubat})} = \frac{565}{646.620} = 0.873$$

$$\omega(\text{Mart}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Mart})} = \frac{565}{726.161} = 0.778$$

$$\omega(\text{Nisan}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Nisan})} = \frac{565}{808.133} = 0.699$$

$$\omega(\text{Mayıs}) = \frac{\text{Gün_Süresi}(\text{Aralık})}{\text{Gün_Süresi}(\text{Mayıs})} = \frac{565}{877.612} = 0.643$$

FIS ortamında “hava durumu” değişkenine yönelik üyelik fonksiyonu Şekil 3.23’de gösterilmektedir. Bu üyelik fonksiyonu oluşturulurken, Tablo 3.10’daki hava durumu bilgileri ve ağırlık değerleri göz önüne alınmıştır.

**Şekil 3.23:** Hava durumu değişkeni üyelik fonksiyonları.

Tablo 3.10; T.C. Orman ve Su İşleri Bakanlığı Meteoroloji Genel Müdürlüğü web sitesinde yer alan günlük hava tahmin raporları ve “openweathermap.org” web sitesinde bulunan ayrıntılı hava durumu bilgileri referans alınarak oluşturulmuştur. Tablo 3.10; beş farklı hava durumunu ve bu hava durumları için havadaki bulutluluk miktarına bağlı olarak [0-1] aralığına ölçeklendirilmiş ağırlık değerlerini göstermektedir. Bulutluluk miktarına ait ağırlıklar hesaplanırken, Tablo 3.11’deki *z-score*, *min-max*, *ondalık ölçeklendirme* normalizasyon yöntemlerinden faydalanılmıştır. Bulutluluk miktarı, 8 eşit parçaya bölündüğü varsayılan gökyüzünün ne kadarının bulutlar tarafından kaplandığını belirtmektedir. Dikey görüşün ifade edilmesi ve güneş panellerinin güneşten maksimum oranda enerji sağlamaları açısından çok önemli bir tanımlamadır.

Tablo 3.10: Hava durumları ve ağırlık değerleri.

ID	Hava Durumu	Bulutluluk Miktarı	Ortalama	(ω)	1.Gün (ω)	2. Gün (ω)	3. Gün (ω)
1	Kapalı	% (87.5-100)	% 93.75	1.0	0.95	0.93	0.9
2	Parçalı Bulutlu	% (62.5-87.5)	% 75	0.88	0.84	0.82	0.8
3	Dağınık Bulutlu	% (37.5-62.5)	% 50	0.63	0.62	0.61	0.6
4	Az Bulutlu	% (12.5-37.5)	% 25	0.38	0.39	0.40	0.4
5	Açık	% (0-12.5)	% 6.25	0.13	0.17	0.19	0.2

Yenilenebilir enerji kaynağı olan güneşten maksimum oranda güç sağlamak amacıyla algılayıcı ve koordinatör düğümlerde güneş panelleri kullandığımızı önceki bölümlerde açıklamıştık. Güneş panellerinin güneşten ne kadar enerji sağlayacakları; KAA’nın konuşlandırıldığı bölgedeki hava koşullarına (kapalı, parçalı bulutlu, yağmurlu, karlı, sisli, tozlu, güneşli) bağlı olarak değişmektedir. Bu değişkenlik, algılayıcı düğümlerdeki pillerin şarj sürelerini ve KAA’daki senkronizasyon zamanını önemli derecede etkiler. Örneğin; hava durumunun çok bulutlu, aşırı sisli, kuvvetli yağmurlu, yoğun kar yağışlı olduğu günlerde ortalama güneşlenme süresi azaldığından, pillerin güneş panellerinden sağlayacakları güç düşmekte ve şarj olmaları uzun sürmektedir. Dolayısıyla, senkronizasyon protokolümüz, kötü hava koşullarında senkronizasyon zamanını uzun tutmaktadır. Ayrıca, hava durumunun açık, güneşli, az bulutlu olduğu günlerde ortalama güneşlenme süresi arttığından, pillerin güneş panellerinden sağlayacakları güç artmakta ve şarj olmaları kısa sürmektedir. Bu tür durumlarda da senkronizasyon protokolümüz senkronizasyon zamanını kısa tutmaktadır.

Senkronizasyon protokolümüz, üç günlük hava tahmin bilgisi kullanmaktadır. Dolayısıyla, bulutluluk miktarına ait ağırlıklar hesaplanırken hava tahmin tutarlılığı da dikkate alınmıştır. Meteoroloji Genel Müdürlüğü web sitesinde yer alan bilgiler; hava tahmin tutarlılığının 1, 2 ve 3. günler için %80-90, 4 ve 5. günler için %65-80 civarında olduğunu belirtmektedir. Bu bilgiler doğrultusunda; 1. gün hava tahmin tutarlılığı 0.9, 2. gün hava tahmin tutarlılığı 0.85 ve 3. gün hava tahmin tutarlılığı 0.80 olarak belirlenmiştir.

Tablo 3.11: Çeşitli veri normalizasyon yöntemleri.

Z-Score	Minimum-Maksimum	Ondalık Ölçeklendirme
μ = popülasyon ortalaması σ = popülasyonun standart sapması x = ham veri z = ham veriye ait z skoru $z = \frac{x - \mu}{\sigma}$	v' = normalize edilmiş yeni veri, v = ham veri $Max(A)$ = A popülasyonunun maksimum değeri $Min(A)$ = A popülasyonunun minimum değeri $max(a)$ = [0, 1] aralığındaki maksimum değer $min(a)$ = [0, 1] aralığındaki minimum değer $v' = \frac{v - Min(A)}{Max(A) - Min(A)} * (max\ a - min\ a) + min\ a$	v' = yeni değer, v = ham değer $i = \max v < 1$ ifadesini sağlayacak en küçük i tamsayı değeri $v' = \frac{v}{10^i}$

Örnek ağırlık hesaplama: z-score normalizasyon yöntemi kullanılarak “1. gün kapalı”, “2. gün parçalı bulutlu”, “3. gün dağınık bulutlu” hava durumlarına ait ağırlıklar aşağıda hesaplanmıştır. Bulutluluk miktarı (BM) = [0-100] arası, $\mu = 50$ ve $\sigma = 29.012$ için:

1. gün hava kapalı (BM=%100) ise:

$$z = \frac{x - \mu}{\sigma} = \frac{100 - 50}{29.012} = 1.723 \Rightarrow \omega(z) = 0.958 \approx 0.96$$

$$\omega(1.gün) = 0.96 * 0.9 + 0.05 = 0.91$$

2. gün hava parçalı bulutlu (BM=%87.5) ise:

$$z = \frac{87.5 - 50}{29.012} = \frac{27.5}{29.012} = 1.293 \Rightarrow \omega(z) = 0.902 \approx 0.9$$

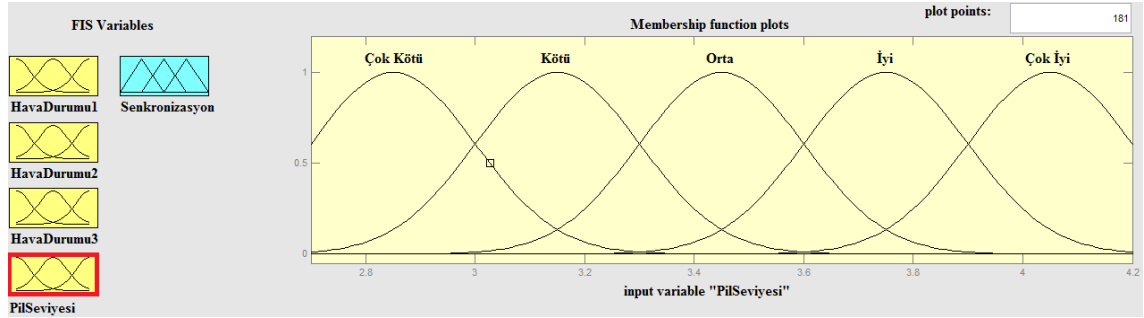
$$\omega(2.gün) = 0.9 * 0.85 + 0.075 = 0.84$$

3. gün hava dağınık bulutlu (BM=%55) ise:

$$z = \frac{55 - 50}{29.012} = \frac{5}{29.012} = 0.172 \Rightarrow \omega(z) = 0.568 \approx 0.57$$

$$\omega(3.gün) = 0.57 * 0.8 + 0.1 = 0.56$$

FIS editöründe tanımlanmış “pil seviyesi” değişkenine ait üyelik fonksiyonları, Şekil 3.24’te gösterilmektedir. Üyelik fonksiyonları oluşturulurken, Tablo 3.12’deki kablosuz algılayıcı düğümümüzün pil gerilim seviyeleri ve ağırlık değerleri göz önüne alınmıştır.



Şekil 3.24: Pil seviyesi değişkeni üyelik fonksiyonları.

Tablo 3.12; tasarladığımız KAA’da kullanılan kablosuz algılayıcı düğümümüzün pil gerilim seviyelerini (pil seviye denetleyiciden mikrodenetleyiciye gönderilen gerilim % oranları), pil gerilim seviyeleri aralıklarına denk gelen durumları ve ağırlıkları göstermektedir. Minimum pil gerilim değeri olarak, algılayıcı düğümümüzün iletişim alt biriminin (XBee 868LP) minimum çalışma gerilimi (2.7V) dikkate alınmıştır. Bunun sebebi; pil seviyesinin 2.7V altında olması durumunda iletişim modülünün çalışmaması ve algılayıcı düğümün KAA’da veri iletişimini gerçekleştirememesidir. Maksimum pil gerilim değeri olarak da, LiFePO₄ pilin maksimum çıkış gerilimi (3.7V) referans alınmıştır. Pil gerilim seviyelerine yönelik ağırlık değerleri; z-score, min-max, ondalık ölçeklendirme normalizasyon yöntemlerinden sistemimiz için uygun olanı kullanılarak hesaplanmış ve [0-1] aralığına ölçeklendirilmiştir. Her bir durumun ağırlığı, o durum ile eşleşen pil gerilim seviyesi aralığındaki minimum gerilime ait ağırlık değeri ölçüt alınarak belirlenmiştir.

Tablo 3.12: Algılayıcı düğümümüzün pil gerilim seviyeleri ve ağırlık değerleri.

ID	Pil Gerilim Seviyesi	Durum	(ω)
1	% (0-20)	Çok Kötü	1.0
2	% (21-40)	Kötü	0.84
3	% (41-60)	Orta	0.62
4	% (61-80)	İyi	0.35
5	% (81-100)	Çok İyi	0.14

Örnek ağırlık hesaplama: “Kötü: %25, “Orta: %50”, “İyi: %75”, “Çok İyi: %90” pil seviyesi durumlarına ait ağırlıklar aşağıda hesaplanmıştır. Pil gerilim seviyesi aralığı [%0-%100] arası, $\mu = 50$, $\sigma = 29.012$ için:

- Pil seviyesi durumu kötü (%25) ise:

$$z = \frac{25 - 50}{29.012} = \frac{-25}{29.012} = -0.862 \Rightarrow \omega(z) = 0.194, \omega(pil) = 1 - 0.194 = 0.806$$

- Pil seviyesi durumu orta (%50) ise:

$$z = \frac{50 - 50}{29.012} = \frac{0}{29.012} = 0 \Rightarrow \omega(z) = 0.5, \omega(pil) = 1 - 0.5 = 0.5$$

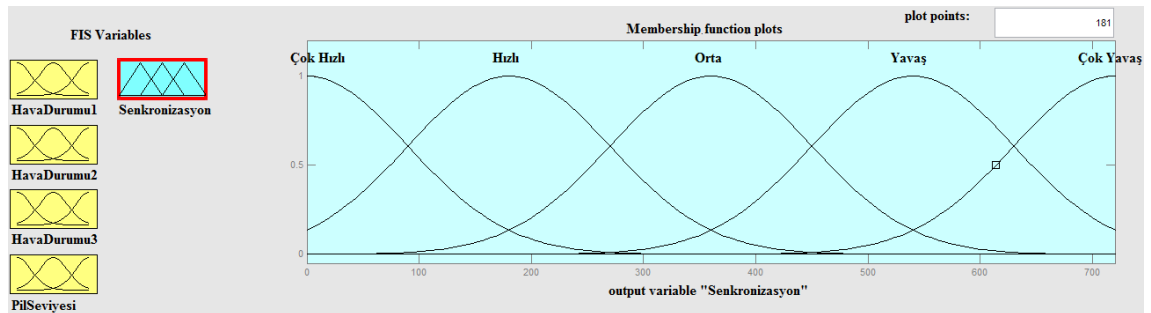
- Pil seviyesi durumu iyi (%75) ise:

$$z = \frac{75 - 50}{29.012} = \frac{25}{29.012} = 0.862 \Rightarrow \omega(z) = 0.806, \omega(pil) = 1 - 0.806 = 0.194$$

- Pil seviyesi durumu çok iyi (%90) ise:

$$z = \frac{90 - 50}{29.012} = \frac{40}{29.012} = 1.379 \Rightarrow \omega(z) = 0.916, \omega(pil) = 1 - 0.916 = 0.084$$

FIS ortamında hazırlanan “senkronizasyon” değişkenine ait üyelik fonksiyonları Şekil 3.25’de belirtilmiştir. Senkronizasyon zamanı; “çok hızlı”, “hızlı”, “normal”, “yavaş” ve “çok yavaş” durumlarını kapsamaktadır.



Şekil 3.25: Senkronizasyon zamanı üyelik fonksiyonları.

Giriş parametrelerinin istediğimiz çıkış değerleriyle ilişkisi, tanımladığımız kurallarla olmaktadır. Dolayısıyla, KAA’da senkronizasyon zamanını belirleyebilmek için bulanık

mantık kurallarının hazırlanması gerekir. Senkronizasyon protokolümüz için tanımlanan kural matrisi Tablo 3.13’de verilmiştir.

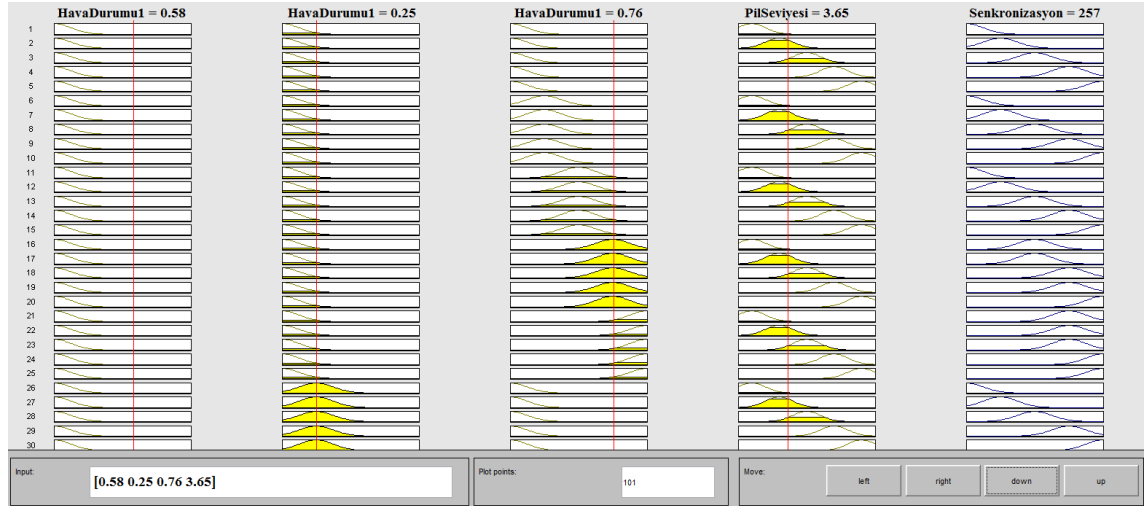
Tablo 3.13: Senkronizasyon protokolümüz için kural matrisi.

“AND” İşlemi		Günlük Hava Durumu				
		Açık	Az Bulutlu	Dağınık Bulutlu	Parçalı Bulutlu	Kapalı
Pil Seviyesi	Çok İyi	Çok Hızlı	Çok Hızlı	Çok Hızlı	Hızlı	Hızlı
	İyi	Çok Hızlı	Çok Hızlı	Hızlı	Hızlı	Orta
	Orta	Hızlı	Hızlı	Orta	Orta	Yavaş
	Kötü	Yavaş	Yavaş	Yavaş	Çok Yavaş	Çok Yavaş
	Çok Kötü	Çok Yavaş	Çok Yavaş	Çok Yavaş	Çok Yavaş	Çok Yavaş

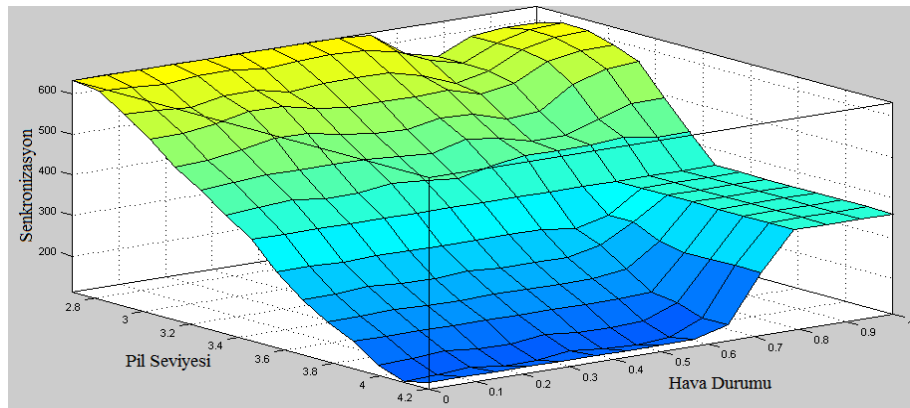
FIS ortamında senkronizasyon protokolümüz için hazırlanan kurallardan bazıları örnek olarak aşağıda sunulmuştur:

1. (HavaDurumu1==Açık)&(HavaDurumu2==Açık)&(HavaDurumu3==Açık)&(Pil Seviyesi==Çokİyi) => (Senkronizasyon=ÇokHızlı) (1)
2. (HavaDurumu1==Açık)&(HavaDurumu2==Kapalı)&(HavaDurumu3==AzBulutlu)&(PilSeviyesi==Orta) => (Senkronizasyon=Orta) (1)
3. (HavaDurumu1==Açık)&(HavaDurumu2==AzBulutlu)&(HavaDurumu3==Dağınık Bulutlu) & (PilSeviyesi==İyi) => (Senkronizasyon=Hızlı) (1)
4. (HavaDurumu1==AzBulutlu)&(HavaDurumu2==Açık)&(HavaDurumu3==Kapalı)&(PilSeviyesi==İyi) => (Senkronizasyon=Hızlı) (1)
5. (HavaDurumu1==AzBulutlu)&(HavaDurumu2==Açık)&(HavaDurumu3==Kapalı)&(PilSeviyesi==Orta) => (Senkronizasyon=Orta) (1)
6. (HavaDurumu1==AzBulutlu)&(HavaDurumu2==Kapalı)&(HavaDurumu3==Kapalı)&(PilSeviyesi==Kötü) => (Senkronizasyon=ÇokYavaş) (1)
7. (HavaDurumu1==AzBulutlu)&(HavaDurumu2==DağınıkBulutlu)&(HavaDurumu3==ParçalıBulutlu) & (PilSeviyesi==İyi) => (Senkronizasyon=Orta) (1)
8. (HavaDurumu1==AzBulutlu)&(HavaDurumu2==DağınıkBulutlu)&(HavaDurumu3==Kapalı) & (PilSeviyesi==Orta) => (Senkronizasyon=Yavaş) (1)

FIS ortamında kurallar yazılırken AND (VE) bulanık operatörü kullanılmıştır. Bu operatörün kullanılması sonucunda kuralların yorumlanması, kural izleyici (Şekil 3.26) ve yüzey izleyici (Şekil 3.27) yardımıyla aşağıda gösterilmektedir.



Şekil 3.26: Senkronizasyon zamanına ait kural izleyici.



Şekil 3.27: Senkronizasyon zamanına ait yüzey izleyici.

KAA'daki *Raspberry Pi* koordinatör düğümümüz (yerel veri toplama merkezi, HTTP sunucu, ağ geçidi), sürekli İnternet bağlantısına sahiptir ve sistem saatini NTP protokolü ile İnternet üzerinden ayarlamaktadır. KAA'nın konuşlandırıldığı bölgeye ait üç günlük meteoroloji bilgisini(bulutluluk miktarı, günlük ortalama güneşlenme süresi) İnternet üzerinden alarak MariaDB veri tabanı yönetim sistemine aktarır. Koordinatörümüz; veri tabanı yönetim sisteminde tutulan hava durumu bilgilerini ve algılayıcı düğümlerin pil seviyelerini inceler. Üzerinde çalışan bulanık mantık tabanlı yazılım (Program kodu 3.6) sayesinde senkronizasyon işleminin ne zaman gerçekleşmesi gerektiğine karar verir ve sonra senkronizasyon bilgisini algılayıcı düğümlerimize yayınlar (Program kodu 3.7). Senkronizasyon işlemine ait yayın akış diyagramı Şekil 3.28'de verilmiştir.

```

from xml.dom import minidom
import urllib
import time
url_str = 'http://api.openweathermap.org/data/2.5/forecast/daily?q=Istanbul&mode=xml&units=
metric&cnt=14'
import MySQLdb

def ImportTable(tableFile):
    x = open(tableFile, 'r').read()
    a = []
    for i in x.split("\n") [1:-1]:
        a.append(i.split(','))
    try:
        db = MySQLdb.connect(db = 'SensorDB', user = 'root', passwd = '1234')
        c = db.cursor()
        c.execute("truncate FuzzyLogic")
        for i in a:
            c.execute("INSERT INTO FuzzyLogic VALUES (%s, %s, %s, %s, %s, %s);",
                (i[0], i[1], i[2], i[3], i[4], i[6]))
        db.commit()
    except:
        db.rollback()
    finally:
        db.close()

def FuzzyCheck(Month, Weather, Battery):
    print "Ay:", Month, "Hava:", Weather, " Pil:", Battery
    try:
        db = MySQLdb.connect(db = 'SensorDB', user = 'root', passwd = '1234')
        c = db.cursor()
        c.execute("SELECT SyncTime FROM FuzzyLogic WHERE MONTH = %s AND
Forecast1 = %s AND Forecast2 = %s AND Forecasat3 = %s AND MinBattery = %s",
            (Month, Weather[0], Weather[1], Weather[2], Battery));
        return c.fetchone()[0]
    except:
        db.rollback()
    finally:
        db.close()

def getClouds(num_days, url_str):
    try:
        xml_str = urllib.urlopen(url_str).read()
    except urllib2.HTTPError, err:
        print "Error: Unable to get weather information!", err.code()
    xmldoc = minidom.parseString(xml_str)
    q = xmldoc.getElementsByTagName('time')
    return [int(a.getElementsByTagName('clouds')[0].getAttribute('all')) for a in q[0:num_days]]

def AFC (BatteryStatus):
    Battery = min(5, int (BatteryStatus)/20 + 1);
    return FuzzyCheck(time.gmtime().tm_mon, [6-min(5, (i/20+1)) for i in getClouds(3, url_str)],
        Battery)

def mVoltToPid(BatteryMVolt):
    return((BatteryMVolt-2700)/300) + 1

```

Program kodu 3.6: Senkronizasyon protokolümüzde tanımlı bulanık mantık modülü.

```

def Broadcast(BatteryStatus):
    NewSensorReadPeriod = 60
    print "Yeni Sensor Okuma Periyodu:", NewSensorReadPeriod
    ser.write(struct.pack('<H', NewSensorReadPeriod))

    try:
        NewSyncTime = max(2, AFC(BatteryStatus))
        SyncTime = NewSyncTime
        print "Senkronizasyon Süresi (Fuzy Logic):", SyncTime
    except:
        SyncTime = 2
        print "Senkronizasyon Süresi (FuzzyLogic Fail ):", SyncTime

    if 1:
        SyncTime = 30; #test amaçlı
        print "Senkronizasyon Süresi (Test):", SyncTime

    tmx = time.time()
    if tmx%60 > 30: tmx += 60-tmx%60
    else: tmx -= tmx%60

    NextSyncTime = tmx + SenkronZaman*60 #Senkronizasyon zamani dakika (saniye)
    tm = time.gmtime(NextSyncTime)

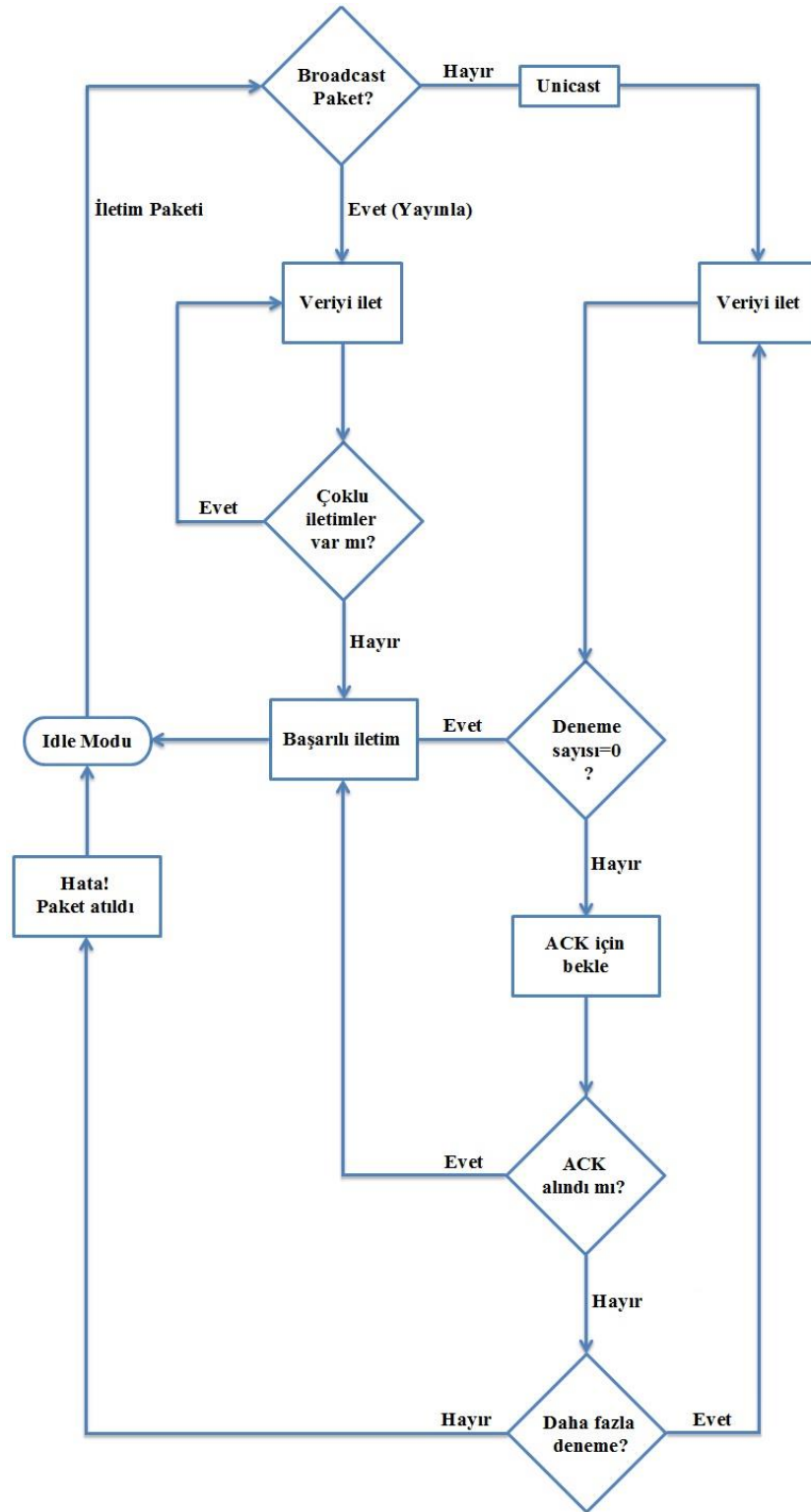
    print "Senkronizasyon Zamanı:", time.strftime("%Y-%m-%d %H:%M:00", tm)
    round_sec = 0
    if tm[4]>30: round_sec = 1
    ser.write(chr(tm[2]) + chr(tm[1]) + chr(tm[0]-2000) + chr(tm[3]) + chr(tm[4]) + chr(0))

    #Time Synchronization
    tm = time.gmtime(time.time())
    print "Saat Senkronizasyonu:", time.strftime( "%Y-%m-%d %H:%M:%S", tm)
    ser.write(chr(tm[2]) + chr(tm[1]) + chr(tm[0]-2000) + chr(tm[3]) + chr(tm[4]) + chr(tm[5]))

    return NextSyncTime

```

Program kodu 3.7: Senkronizasyon protokolümüzde tanımlı broadcast modülü.



Şekil 3.28: Senkronizasyon işlemi için yayın (broadcast ve unicast) akış diyagramı.

4. BULGULAR

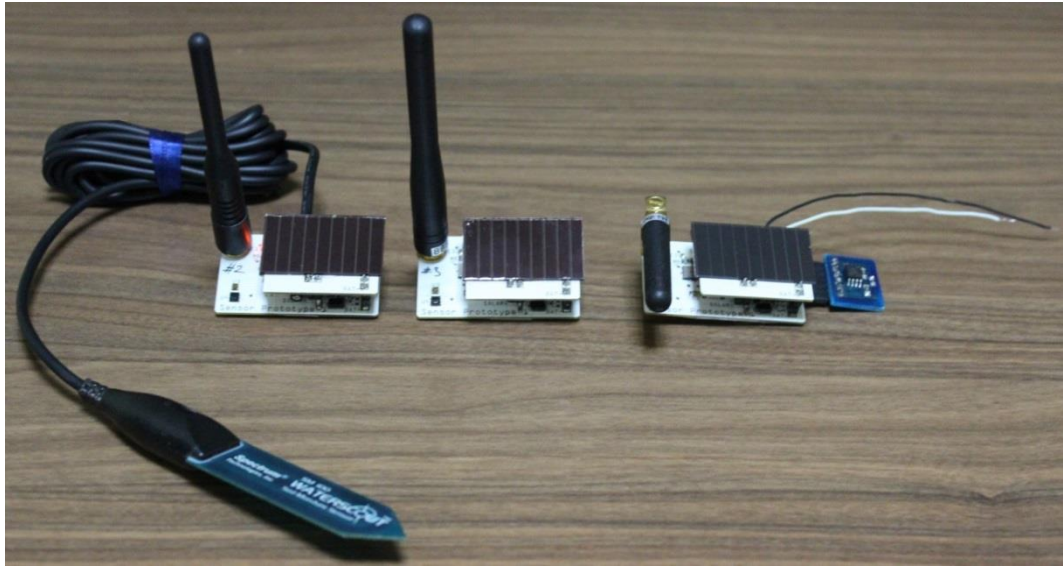
KAA'lar, işbirlikçi çalışabilme yeteneğine sahip çok sayıda kablosuz algılayıcı cihazın belirli bir bölgeye tasarsız şekilde konuşlandırılmasıyla oluşturulan dağıtık sistemlerdir. Algılayıcı düğümler, bulundukları ortamda gerçekleşen fiziksel olaylar hakkında kendilerine has topladıkları verileri çok sekmeli iletişim aracılığıyla alıcı birimlere gönderirler. Ayrıca, birçok dağıtık uygulama ve protokol; düğümler arasındaki eylemleri eşgüdümlemek ya da meydana gelen olaylar arasında ilişkiler kurabilmek amacıyla ortak bir zaman kavramına gereksinim duyarlar. Dolayısıyla, algılayıcı düğümlere özgü işlemler dikkate alındığında KAA uygulamalarında çok sık rastlanan en önemli iki problem karşımıza çıkmaktadır. Bunlar;

- *Algılayıcı düğümlerin arızalanması:* Kötü arazi ya da zor iklim şartlarında arızalı düğümlerin yerine yenilerinin konulamaması.
- *Algılayıcı düğümlerin kısıtlı kapasitede güç kaynaklarına sahip olmaları:* Bir algılayıcı düğümün enerjisinin bitmesi durumunda üzerindeki pilin şarj edilememesi ya da yenisi ile değiştirilememesi.

Uygulamanın amacına ve yerine göre, KAA'nın fonksiyonellik ve etkililik açısından uzun ömürlü olması beklenir. Enerjisi tükenen düğümlerin güç kaynaklarının şarj edilememesi ya da yenisiyle değiştirilememesi, arızalı/bozuk düğümlerin yerine yenilerinin konuşlandırılmaması KAA'ların verimli çalışabilmesinde ciddi sorunları da beraberinde getirir. Arızalı ya da enerjisi tükenmiş düğümün bulunduğu bölgeden fiziksel olaylara ait veriler toplanamaz. Bu durum, ağ içerisinde yapısal değişikliklerin meydana gelmesine, ağın işlevselliği bakımından merkezi sistemlerde verimli ve doğru sonuçların elde edilememesine sebep olur.

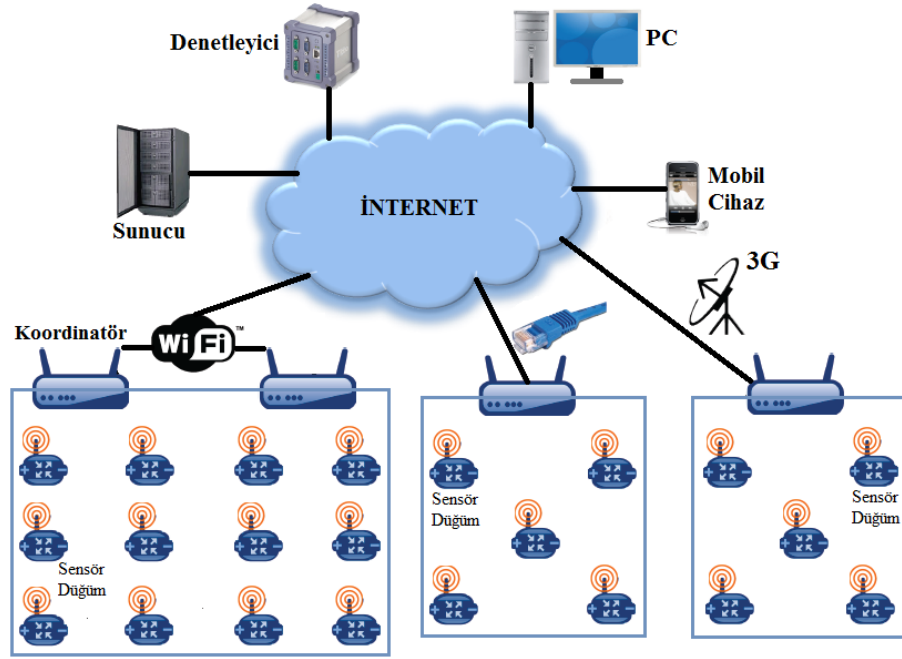
Çalışmamızda; Bluetooth ve IEEE 802.15.4 standartlarına uyumlu, kısa ya da uzun mesafelerde haberleşme yapabilen, organize çalışabilme yeteneğine sahip, düşük güçlü, küçük boyutlu kablosuz algılayıcı düğümler gerçekleştirdik (Şekil 4.1). Bu algılayıcı düğümlerimizin güç birimlerinde iki farklı enerji kaynağı kullandık. Bunlardan ilki,

çevresel faktörlerden fazla etkilenmeyen, şarj edilebilir bir LiFePO_4 pildir. LiFePO_4 piller, mevcut kablosuz algılayıcı düğümlerde kullanılan diğer pillere göre pek çok üstün özelliklere sahiptir. Çevreye zararları daha azdır ve uzun ömürlüdürler. Termal ve kimyasal kararlılıkları bakımından lityum bazlı pillerden daha avantajlıdır. Korunaklı dış kılıfları ve kimyasal özellikleri bakımından daha güvenlidirler. Enerji yoğunluğu diğer lityum pillerden daha düşüktür. Çok sabit deşarj gerilimleri vardır ve hızlı şarj imkanı sunarlar. Daha yüksek akım ya da maksimum güç değerlerine sahiptirler. Literatür araştırmalarımız sırasında, KAA uygulamaları için geliştirilmiş mevcut kablosuz algılayıcı düğümlerde güneş panellerinin kullanılmadığını tespit ettik. Çok nadiren, sabit ağ geçitlerinde güneş panellerinin bulunduğunu gördük. Bu durumdan hareketle, literatürde adı geçen mevcut algılayıcı düğümlerden (EYES, IRIS, Mica2, MicaZ, TelosB, Shimmer, XYZ) donanımsal özellikleri bakımından birçok farklılık gösteren kablosuz algılayıcı düğümlerimizde, ikinci bir enerji kaynağı (güç üretici) olarak güneş panelleri kullandık.



Şekil 4.1: Kablosuz algılayıcı düğümlerimiz.

KAA mimarisini, karakteristiklerini, standartlarını, protokollerini, uygulama alanlarını daha iyi kavrayabilmek ve bunun sonucunda KAA'lar ile ilgili enerji problemlerini çözebilmek amacıyla, ağ geçitleri (koordinatör, veri toplama merkezi) ve tasarladığımız kablosuz algılayıcı düğümlerimizi kullanarak bir KAA oluşturduk (Şekil 4.2). Algılayıcı düğümlerimiz ve ağ geçitleri için geliştirdiğimiz dinamik güç yönetim sistemleri ve güç bilinçli yazılımımız sayesinde KAA'nın enerji verimli çalışmasını sağladık.



Şekil 4.2: Tasarladığımız KAA topolojisi.

Bir kablosuz algılayıcı düğümde kullanılan RF alıcı-verici modül, enerji tüketimini en çok etkileyen birimdir. KAA alanında yapılan çalışmalar, bir algılayıcı düğümde veri iletişimi için harcanan enerjinin, mikroişlemci ve diğer birimlerde harcanan enerjiden çok daha fazla olduğunu göstermektedir. Örneğin; Rayleigh sönümlemesini ve dördüncü güç mesafesi kaybını varsayarak, 100m mesafeden 1kB paket göndermedeki enerji maliyetinin, tipik bir mikroişlemci tarafından üç milyon komut yürütülmesindeki enerji maliyetine yaklaşık olarak eşit olduğu belirlenmiştir [118]. Çalışmamızda, KAA uygulamaları için geliştirilmiş mevcut algılayıcı düğümleri, ağ geçitlerini (koordinatör) iletişim protokollerini, standartları, haberleşme donanımlarını ayrıntılı bir şekilde inceledik. Literatürde, KAA uygulamalarında kullanılan cihazların veri iletişimi sırasındaki enerji tüketimleri ile ilgili problemleri çözmeye yönelik çok çeşitli çalışmalar olduğunu gördük. Sonuç olarak, uygulamalara özgü ağ topolojileri ve tasarımları dikkate alındığında, KAA'ların enerji tüketimlerini azaltma ve uzun ömürlü olmalarını sağlamadaki en büyük katkının, düşük güçlü algılayıcı ve koordinatör düğüm devreleri tasarlamakla ve bu düğümler üzerinde çalışacak etkin, enerji verimli sistemler, protokoller ya da ara katman yazılımları geliştirmekle olabileceğini tespit ettik. Buradan hareketle, çalışmamızda; bir programlanabilir voltaj regülatörü aracılığıyla dinamik voltaj ölçeklendirme yaparak algılayıcı düğüm devremizdeki algılama, veri işleme, depolama, iletişim işlemlerini gerçekleştiren birimlerin ihtiyaç duyduğu çalışma

gerilimlerini uyarlayan ve enerjilerini kontrol eden güç bilinçli bir sistem geliştirdik. Sistemimiz; senkronizasyon zamanına bağlı olarak XBee 868LP radyo modülünün gücünü ayarlamakta, veri gönderme/alma işleminde radyoyu aktif hale getirmekte, veri gönderme/alma işlemi bittikten sonra radyo modülünü uyku moduna geçirmekte ya da doğrudan kapatmaktadır. Algılama, veri işleme, depolama birimleri için dinamik voltaj ölçeklendirme yapmakta ve mikrodenetleyiciyi boşa kaldığı durumlarda en derin uyku moduna almaktadır. Bu sayede, kablosuz algılayıcı düğümlerimizin enerji tüketimlerini önemli ölçüde azaltmış ve uzun ömürlü çalışmalarını sağlamış olduk.

4.1. KABLOSUZ ALGILAYICI DÜĞÜM İÇİN GÜÇ BİLİNÇLİ METODOLOJİ

Bir kablosuz algılayıcı düğüm genellikle pille çalışır ve bu nedenle enerji kısıtlıdır. Algılayıcı düğümü konuşlandırdıktan sonra düğümün yaşam süresini uzatmak için devreleri, mimariyi, algoritmaları ve protokolleri kapsayan diğer hususlar enerji verimli olmak zorundadır. Bir kez sistem tasarlandığında, ilave enerji tasarrufları dinamik güç yönetimi kullanarak kazanılır. Dinamik güç yönetimi, hiçbir olayın meydana gelmediği durumlarda algılayıcı düğümün kapatılmasına dayanan bir tekniktir. Böyle olay güdümlü güç tüketimi, maksimum pil ömrü için son derece önemlidir. Ayrıca, algılayıcı düğüm, uygulamaya bağlı olarak görev süresi uzatılabileceğinden algılama hassasiyeti bedelinde enerji ölçeklenebilirliğine sahip olmalıdır. Enerji ölçeklendirme algoritmaları ve protokolleri bu enerji kısıtlı durumlar için önerilmiştir [146-154].

Algılama uygulamaları; veri hızı, hesaplama, ortalama iletim mesafesi bakımından çok kapsamlı gereksinimler sunmaktadır. Elektronik devreler ve sistemler, protokoller, algoritmalar uygulamalara özel geliştirilmelidir ve hassas ayarlanmalıdır. Dolayısıyla, programlanabilirlik vazgeçilmez bir ihtiyaç olduğundan gömülü işletim sistemleri ve yazılımlar KAA'lar için son derece önemlidir.

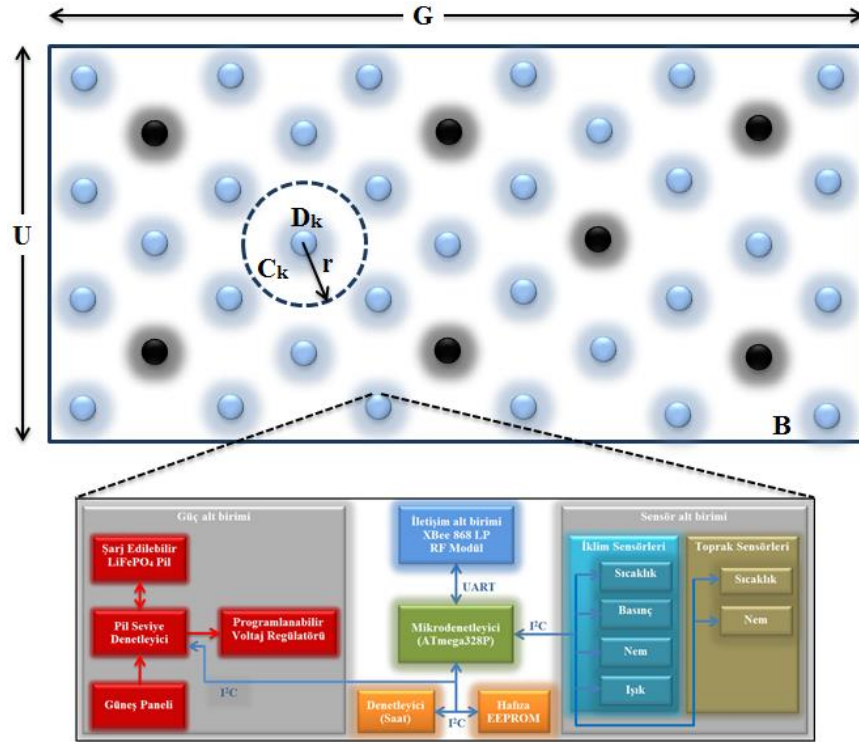
Çalışmamızda, tasarladığımız kablosuz algılayıcı düğümümüzün enerji verimliliğini artırmak için donanım ve yazılım tabanlı bir dinamik güç yönetim (DGY) tekniği önerdik. DGY, performansı düşürmeksizin sistem güç tüketimini azaltmada etkili bir araçtır. Temel fikrimiz; kablosuz algılayıcı düğümlerimizi ihtiyaç duyulmadığında uyutmak (ya da kapatmak) ve gerekli olduğunda onları tekrar uyandırmaktır (çalışır hale getirmek). DGY'nin uygulanması ciddi bir problemdir. Eğer uyku durumu geçişindeki

enerji ve performans ek yükleri önemsiz ise (ihmal edilebilirse), o zaman, sistem boşta çalışırken sistemi en derin uyku durumuna sokan bir greedy algoritması kullanmak iyi bir çözümdür [147]. Fakat, gerçekte, uyku durumu geçişi, gücü kapatmanın ve işlemci durumunu depolamanın ek yüküne sahiptir. Ayrıca, uyanma ölçülebilir miktarda zaman alır. Bu nedenle, uyku durumu geçişine yönelik doğru ilkeyi uygulama DGY başarımında çok önemlidir.

Kapatma ya da uyku teknikleri, boşta olan sistem durumlarında önemli enerji tasarrufları sağlar. Bununla beraber, aktif durumda algılayıcı düğüm performansını optimize ederek ek enerji tasarrufları sağlamak da mümkündür. Dinamik voltaj ölçeklendirme (DVÖ), işlemcinin enerjisini azaltmada etkili bir yöntemdir. Çoğu mikroişlemci sistemi, zamanla değişen hesaplama yükü ile karakterize edilir. Azaltılmış aktivite periyodları boyunca çalışma frekansını düşürmek, güç tüketiminde lineer azalmalara yol açar; fakat, görev başına tüketilen toplam enerjiyi etkilemez. Çalışma gerilimini düşürmek ise, daha büyük kritik yol gecikmelerini beraberinde getirir ve dolayısıyla pik performanstan ödün verilmesine neden olur. Pik performansın her zaman gerekmediği durumlarda önemli enerji kazançları elde edilebilir. Bu nedenle, işlemcinin çalışma gerilimi ve frekansı anlık işlem gereksinimlerine bağlı olarak dinamik bir biçimde uyarlanabilir. DVÖ'nün amacı, iş yükünü eşleştirmek için güç kaynağını ve çalışma frekansını uyarlamaktır. Böylelikle görünür performans kaybı gözardı edilebilir. Burada en önemli sorun, gelecek iş yüklerinin genellikle deterministik olmamasıdır. DVÖ yapıldığındaki hızın da performans ve enerji üzerine önemli etkisi vardır. Düşük güncelleme hızı, daha düşük enerji ile sonuçlanan daha büyük iş yükü ortalaması anlamına gelir. Güncelleme, enerji ve performans maliyeti daha uzun zaman dilimi üzerinden amorti edilir. Öte yandan, düşük güncelleme hızı, sistem iş yükündeki ani artışa cevap vermeyeceğinden daha büyük performans düzeyini belirtir [146-154].

4.1.1. Sistem Modelleri

Şekil 4.3, η adet kablosuz algılayıcı düğümün belirli bir bölgeye rasgele konuşlandırılmasıyla oluşturulan örnek bir KAA uygulama modelini göstermektedir. Bu modelde kullanılan algılayıcı düğümler, tasarladığımız kablosuz algılayıcı düğümleri belirtmektedir.



Şekil 4.3: KAA ve kablosuz algılayıcı düğüm mimarisi.

Dağıtık KAA uygulamalarında temel amaç, algılayıcı düğümlerin öz-yapılandırılmalı ve adaptif olması için her bir düğümün yeterli işlem gücünü bünyesinde bulundurmasıdır. Şekil 4.3, aynı zamanda, tasarladığımız kablosuz algılayıcı düğümlerimizin temel mimarisini de göstermektedir. Algılama bölgesindeki her bir düğümümüz; algılayıcı, mikrodnetleyici, radyo, hafıza, harici saat ve güç alt birimlerinden oluşmaktadır. Kablosuz algılayıcı düğümlerimizdeki her bir bileşen, mikrodnetleyici üzerinde çalışan ve mikrocihaz sürücüleri içeren güç bilinçli yazılımımız tarafından kontrol edilmektedir. Geliştirdiğimiz yazılımın en önemli fonksiyonu, dinamik güç yönetimine imkan sağlamasıdır. Yazılımımız, olay istatistiklerine bağlı olarak algılayıcı düğümde hangi elamanların kapatılması ve açılması gerektiğine karar vermektedir. Tasarladığımız KAA'nın, GxU boyutlarında B dikdörtgensel bölgesi üzerine dağıtılmış η homojen kablosuz algılayıcı düğümden oluştuğunu varsaydık. Her bir düğüm, r yarıçaplı görüş mesafesine sahiptir. Böyle bir KAA için üç farklı iletişim modeli kullanabiliriz:

- *Doğrudan iletişim:* Her bir düğüm, verileri doğrudan ağ geçidine (ya da baz istasyonuna) gönderir.
- *Çok sekmeli iletişim:* Veriler, bireysel düğümler aracılığıyla (bir düğümden diğer düğüme aktarılacak) ağ geçidine doğru yönlendirilir.

- **Kümeleme:** Bir grup algılayıcı düğüm küme oluşturur. Küme üyeleri arasındaki yerel etkileşimler koordinatör düğüm (küme başları) aracılığıyla kontrol edilir. Küme üyeleri genellikle koordinatör ile iletişim kurar. Veriler, enerji tasarrufu için koordinatör tarafından toplanır/birleştirilir ve alıcıya (ağ geçidi, baz istasyonu) ya da komşu küme başına yönlendirilir.

Komşu algılayıcı düğümler arasındaki mesafenin alıcı ile algılayıcı düğümler arasındaki ortalama mesafeden daha az olması durumunda, iletim gücü, düğümlerin yerel olarak birlikte çalışmasıyla korunabilir. Dahası, yerel kümelerdeki algılayıcı düğümler çok ilişkili verileri paylaşırlar. Algılayıcı düğümlerin bazıları kendilerini küme başı (koordinatör) olarak atarlar ve kalan düğümler minimum iletim gücü kriterine göre kümelerden birine katılırlar. Küme başı, küme içerisindeki düğümlerden verileri toplar ve ilgili birimlere iletir. KAA’larda bu işlemleri gerçekleştiren çok sayıda uygulamaya özel ağ protokolleri geliştirilmiştir. Bu protokoller, kümeleme şemasının doğrudan iletim şemasına kıyasla daha enerji verimli olduğunu kanıtlarlar.

4.1.1.1. Güç Bilinçli Kablosuz Algılayıcı Düğüm Modeli

Güç bilinçli kablosuz algılayıcı düğüm modeli, düğüm uyku durumunun farklı seviyelerindeki güç tüketimlerini tanımlar. Algılayıcı düğümü oluşturan her bir bileşen farklı güç modlarına sahiptir. Örneğin; ATmega328P mikrodenetleyici; aktif, boştaki çalışma ya da uyku modunda (6 farklı uyku modundan biri) olabilir. Radyo modülü (XBee 868LP); gönderme, alma, bekleme ya da kapalı modlarını içerir. Her bir algılayıcı düğümün uyku durumu, algılayıcı düğüm bileşenlerine ait güç modlarının belirli bir kombinasyonuna karşılık gelir. Eğer algılayıcı düğümde her biri k_i uyku durumlu N

adet bileşen varsa, toplam düğüm uyku durumları sayısı: $\prod_{i=1}^N k_i$ olur. Her bileşen güç

modu, o moda geçişle ilişkili gecikme süresi ek yüküne sahiptir. Dolayısıyla, her bir düğümün uyku durumu, güç tüketimi ve gecikme süresi ek yükü ile nitelendirilir. Ancak, tüm uyku durumları pratikte kullanışlı değildir. Tablo 4.1, tasarladığımız kablosuz algılayıcı düğümümüzde 20 farklı kullanışlı duruma karşılık gelen bileşen güç modlarını belirtmektedir. Algılayıcı düğümümüze ait uyku durumlarının her biri, artan derin uyku durumuna göre sıralanmıştır. Sonuç olarak, bu uyku durumlarının her biri, artan gecikme süresi ve azalan güç tüketimi ile karakterize edilir.

Tablo 4.1: Kablosuz algılayıcı düğümümüz için çalışma ve uyku durumları.

Uyku Durumları	ATmega328P	Hafıza	Algılayıcılar	Saat	Pil Seviye Denetleyici	Radyo
S0	Aktif	Aktif (O)	Aktif	Aktif	Hibernate	TX
S1	Aktif	Aktif (O)	Bekleme	Aktif	Hibernate	TX
S2	Aktif	Bekleme	Bekleme	Aktif	Hibernate	TX
S3	Aktif	Kapalı	Kapalı	Aktif	Hibernate	TX
S4	Aktif	Bekleme	Bekleme	Aktif	Aktif	RX/Idle
S5	Aktif	Bekleme	Bekleme	Aktif	Hibernate	RX/Idle
S6	Aktif	Kapalı	Kapalı	Aktif	Hibernate	RX/Idle
S7	Uyku (PDWNE)	Kapalı	Kapalı	Aktif	Hibernate	RX/Idle
S8	Aktif	Aktif (Y)	Aktif	Aktif	Hibernate	Uyku
S9	Aktif	Aktif (Y)	Bekleme	Aktif	Hibernate	Uyku
S10	Aktif	Bekleme	Aktif	Aktif	Hibernate	Uyku
S11	Aktif	Bekleme	Bekleme	Aktif	Aktif	Uyku
S12	Aktif	Kapalı	Kapalı	Aktif	Aktif	Uyku
S13	Aktif	Bekleme	Bekleme	Aktif	Hibernate	Uyku
S14	Aktif	Kapalı	Kapalı	Aktif	Hibernate	Uyku
S15	Uyku (PDWNE)	Bekleme	Aktif	Aktif	Hibernate	Uyku
S16	Bekleme	Kapalı	Kapalı	Aktif	Hibernate	Uyku
S17	Uyku (PDWNE)	Kapalı	Kapalı	Aktif	Aktif	Uyku
S18	Uyku (PDWNE)	Kapalı	Kapalı	Aktif	Hibernate	Uyku
S19	Uyku (PDWNE)	Kapalı	Kapalı	Aktif	Hibernate	Uyku

Uyku durumları, algılayıcı düğümümüzün mevcut çalışma koşullarına dayalı seçilir. Örneğin; hafıza ile saatin aktif ve diğer bileşenlerin tamamen kapalı olması mantıklı bir durum değildir. Tasarım problemi; uyku durumları arasındaki geçişe yönelik bir kuralı, enerji verimliliğini artırmak amacıyla gözlemlenen olaya bağlı olarak formüle etmektir. Güç bilinçli algılayıcı düğüm modeli, ACPI (Advanced Configuration and Power Interface- İleri Konfigürasyon ve Güç Arayüzü) standardındaki sistem güç modeline benzerdir [147]. ACPI uyumlu sistem beş global durumu kapsar:

- *SystemStateS0*: çalışma durumuna karşılık gelir.
- *SystemStateS1* ile *SystemStateS4* arası: 4 farklı uyku durumu seviyesine denk düşer.

Uyku durumları; tüketilen güç, uyku ve uyanma zamanlarına gitmede gereken ek yük ile farklılaştırılır. Genelde, daha derin uyku durumu daha az güç harcar ve daha uzun uyanma zamanına sahiptir. ACPI'daki bir diğer benzer durum, güç yöneticisinin bir işletim sistemi modülü (yazılım) olmasıdır [147].

4.1.1.2. Olay Üretim Modeli

Bir olayı algılama, algılayıcı düğüm önceden belirlenmiş bir eşik seviyesi üzerinde güç ile sinyal aldığı zaman gerçekleşir. Şekil 4.3’de gösterilen örnek KAA uygulama modelimizde herbir algılayıcı düğümümüzün uniform r yarıçaplı görüş mesafesine sahip olduğunu varsaydık. Gerçek uygulamalarda, algılayıcı düğümlerin konuşlandırıldığı arazi görülebilir yarıçapı etkiler. Ayrıca, olaylar statik (bir çevre gözetim uygulamasında sıcaklık, basınç, nem durumlarındaki yerel değişiklikler) ya da çoğalan/yayılan (bir izleme uygulamasında hareketli nesne tarafından üretilen sinyaller) tipte olabilir. Genelde, olaylar uzayda ve zamanda nitelendirilebilir (muhtemelen durağan olmayan) dağılıma sahiptir. Tasarladığımız KAA sistemi için [147]’de önerilen yöntemi kullanarak uyarladığımız olay üretim modelinde:

- ✓ Tüm algılama bölgesi (B) üzerindeki zamana bağlı olay davranışının, λ_{Toplam} ile verilen bir ortalama olay oranlı Poisson süreci olduğunu;
- ✓ Olayların uzaysal dağılımının, $\rho_{XY}(x, y)$ ile verilen bağımsız olasılık dağılımı ile ifade edildiğini;

varsayalım. ρ_{O,D_k} ; B bölgesinde meydana gelen bir olayın D_k algılayıcı düğüm tarafından algılanma olasılığını belirtir ve şu şekilde modellenenabilir:

$$\rho_{O,D_k} = \frac{\int_{C_k} \rho_{XY}(x, y) dx dy}{\int_B P_{XY}(x, y) dx dy} \quad (4.1)$$

Denklem 4.1’de;

- $\rho_{XY}(x, y) dx dy$: C_k birim alanında, D_k algılayıcı düğüm tarafından bir olayın algılanma olasılığıdır (Bir olayın C_k birim alanında meydana gelme olasılığı).
- $P_{XY}(x, y) dx dy$: “B” algılama bölgesinde, algılayıcı düğümler tarafından bir olayın algılanma olasılığıdır (Bir olayın “B” birim alanında meydana gelme olasılığı).
- $\int$: Olaylar (sıcaklık, nem, basınç vb.) sürekli rasgele değişken özelliktedir. Dolayısıyla, olasılık hesabında integral işlemi kullanılmıştır. Eğer olaylar aralıklı özellikte (sonlu sayıda) olsaydı $\sum$ işlemi seçilirdi.

$\rho_{D_k}(t, n)$; D_k algılayıcı düğümde, t zamanı içerisinde n olayın meydana gelme olasılığını belirtmektedir. Bu ifadeye göre; T_{th} eşik seviyesi aralığı üzerinde, C_k birim alanı içerisinde hiç bir olayın meydana gelmeme olasılığı (D_k algılayıcı düğümün hiç bir olayı algılamama olasılığı) aşağıda gösterildiği biçimde modellenebilir:

$$P_{D_k}(T_{th}, 0) = \sum_{i=0}^{\infty} \frac{e^{-\lambda_{t0t} \cdot T_{th}} \cdot (\lambda_{t0t} \cdot T_{th})^i}{i!} \cdot (1 - P_{O, D_k})^i = e^{-P_{O, D_k} \cdot \lambda_{t0t} \cdot T_{th}} \quad (4.2)$$

Denklem 4.2’de;

- $f(i, \lambda_{t0t} \cdot T_{th}) = \sum_{i=0}^{\infty} \frac{e^{-\lambda_{t0t} \cdot T_{th}} \cdot (\lambda_{t0t} \cdot T_{th})^i}{i!}$: Poisson dağılımına ait olasılık kütle fonksiyonudur. Bu fonksiyon; C_k birim alanında, olayın i sayıda meydana gelme olasılığını tanımlar. i : T_{th} eşik seviyesi aralığında, olayın kaç kere ortaya çıktığını belirtir. $\lambda_{t0t} \cdot T_{th}$: T_{th} eşik seviyesi aralığında ortaya çıkan i sayıda olay için ortalama değerdir (T_{th} eşik seviyesi aralığında, olayın ortalama olma sayısıdır).
- $(1 - P_{O, D_k})^i$: Beklenen olay meydana geldiğinde, algılayıcı düğümün o olayı algılamama olasılığını ifade eder.

$P_{th, D_k}(t)$; C_k birim alanında, D_k algılayıcı düğümde t zamanı içerisinde en az 1 olayın meydana gelme olasılığı ise:

$$P_{th, D_k}(T_{th}) = 1 - P_{D_k}(T_{th}, 0) = 1 - e^{-P_{O, D_k} \cdot \lambda_{t0t} \cdot T_{th}} \quad (4.3)$$

En az bir olayın meydana gelme olasılığı, uzaysal olarak ağırlıklandırılmış olay ortaya çıkış oranı ($\lambda_{D_k} = \lambda_{Toplam} + \rho_{O, D_k}$) ile karakterize edilen bir üstel dağılımdır. Üstel dağılım için olasılık yoğunluk fonksiyonu aşağıda gösterilmiştir:

$$f(x; \lambda_{D_k}) = \begin{cases} \lambda_{D_k} \cdot e^{-\lambda_{D_k} \cdot x} & , \quad x \geq 0 \\ 0 & , \quad x < 0 \end{cases} \quad (4.4)$$

Ayrıca, bir olayın uzayda çoğalabilme/yayılabilme olasılığını bulmak amacıyla herbir olayı pozisyon vektörü ile şu biçimde ifade edebiliriz:

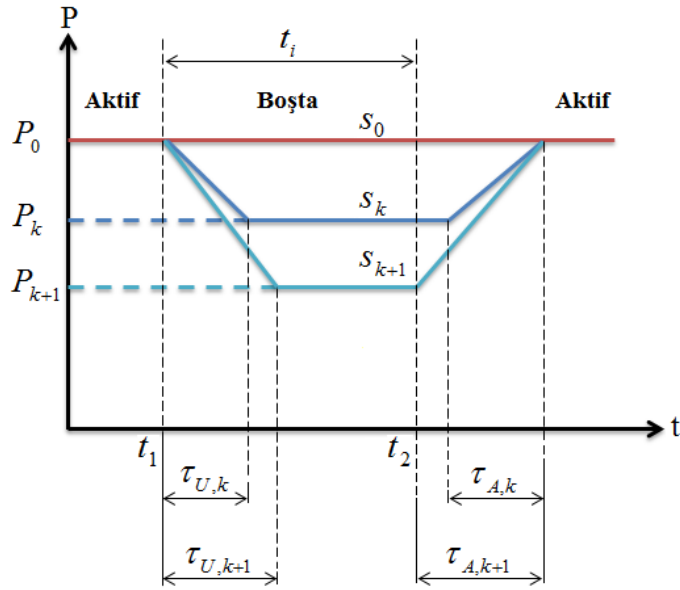
$$\bar{\rho} = \bar{\rho}_0 + \int \bar{v}(t) dt \quad (4.5)$$

Denklem 4.5'te; $\bar{\rho}_0$, olayın çıkış(başlangıç) noktasının koordinatlarıdır. $\bar{v}(t)$, olayın yayılım (çoğalma) hızını belirtmektedir. Başlangıç noktası; Denklem 4.1, 4.2, 4.3 ile tanımlanmış bir uzaysal ve zamansal dağılıma sahiptir. Sonuçta, olay üretim modelini kullanarak üç bağımsız olay sınıfını inceleyebiliriz:

- ✓ $\bar{v}(t) = 0$: olaylar sabit/hareketsiz noktalar şeklinde meydana gelir.
- ✓ $\bar{v}(t) = \text{sabit sayı}$: olay sabit bir hızla çoğalır/yayılır (örneğin: hareketli nesne).
- ✓ $|\bar{v}(t)| = \text{sabit sayı}$: olay sabit bir hızla fakat rastgele yönde çoğalır/yayılır (örneğin: rasgele hareket etme).

4.1.2. Algılayıcı Düğümde Uyku Durumu Geçiş İlkesi

KAA'nın konuşlandırıldığı B algılama bölgesinde meydana gelen bir olayın, belirli bir zamanda D_k algılayıcı düğümümüz tarafından algılandığını varsayalım (Şekil 4.3). Algılayıcı düğüm, t_1 süresinde bu olayı işlemeyi bitirir. Bir sonraki olay, $t_2 = t_1 + t_i$ zamanında meydana gelir. D_k algılayıcı düğüm, t_1 zamanında s_0 aktif durumundan s_k uyku durumuna geçiş için karar verir. Her bir s_k durumu, P_k güç tüketimine sahiptir. Aktif durumdan her bir uyku durumuna geçiş zamanı $\tau_{d,k}$, her bir uyku durumundan tekrar aktif duruma geçiş zamanı ise $\tau_{u,k}$ olarak belirlenir. Şekil 4.4, açıklanan durumları ve geçişleri örneklendirmektedir. Algılayıcı düğümümüzün uyku durumlarına yönelik tanımlamamız; herhangi $i > j$ için $P_j > P_i$, $\tau_{d,i} > \tau_{d,j}$, $\tau_{u,i} > \tau_{u,j}$ biçimindedir.



Şekil 4.4: Durum geçişleri, gecikme süresi ve harcanan güç.

$0 \leq k \leq N$ olmak üzere, N uyku durumu için, $\{s_k\}$ uyku durumlarına karşılık gelen $\{T_{th,k}\}$ uyku zamanı eşik seviyelerinin bir kümesini türetebiliriz. Eğer boşta kalma zamanı uyku zamanı eşik seviyesinden küçük olursa ($t_i < T_{th,k}$), s_0 durumundan s_k uyku durumuna geçiş, geçiş enerjisi ek yükü nedeniyle net enerji kaybına yol açar. Bu, geçiş periyodunda hiçbir verimli işin yapılamayacağı anlamını taşır. Örneğin; bir işlemci uyandığı zaman kilitlemek için PLL (faz kilitlemeli döngüler), stabilize olmada (dengede tutmak) saat, yeniden yüklenmesinde (eski haline getirme) işlemci içeriği bekleyen geçiş zamanı harcar [147]. Bir durumdan uyku durumuna geçişteki enerji tasarrufu aşağıda gösterildiği şekilde modellenenebilir:

$$E_{Tasarruf,k} = P_0 \cdot t_i - \left(\frac{P_0 + P_k}{2} \right) \cdot (\tau_{U,k} + \tau_{A,k}) - P_k (t_i - \tau_{U,k}) \quad (4.6)$$

$$E_{Tasarruf,k} = (P_0 - P_k) \cdot t_i - \left(\frac{P_0 - P_k}{2} \right) \cdot \tau_{U,k} - \left(\frac{P_0 - P_k}{2} \right) \cdot \tau_{A,k} \quad (4.7)$$

Böyle bir geçiş, sadece $E_{Tasarruf,k} > 0$ olduğu zaman geçerlidir. Bu durumdaki uyku zamanı eşik seviyesi ise;

$$T_{th,k} = \frac{1}{2} \left[\tau_{U,k} + \left(\frac{P_0 + P_k}{P_0 - P_k} \right) \cdot \tau_{A,k} \right] \quad (4.8)$$

Denklem 4.8 dikkatle incelendiğinde iki önemli sonuç ortaya çıkmaktadır:

- ✓ $s_0 \rightarrow s_k$ geçişinin gecikme ek yükü daha uzun olursa, geçiş zamanı eşik seviyesi de daha uzun olur ve enerji kazancı eşik seviyesi de buna bağlı olarak daha yüksek olur.
- ✓ P_0 ile P_k arasındaki farkın daha fazla olması, eşik seviyesinin de daha küçük olması anlamına gelir.

Yukarıda açıklanan sonuçlar doğrultusunda, algılayıcı düğümde T_{th} zamanında en az bir olayın olma olasılığını Denklem 4.3'deki Poisson teorisini kullanarak bulabiliriz:

$$P_{th,k}(T_{th}) = 1 - e^{-\lambda_k \cdot T_{th}} \quad (4.9)$$

Tablo 4.2, Şekil 4.3'de gösterilen D_k kablosuz algılayıcı düğümümüzün farklı uyku durumlarındaki güç tüketimlerini listelemektedir. Algılayıcı düğüm, hazır bileşenlerden oluştuğundan güç tüketimi için optimize edilmez. Ancak, temel fikri örneklendirmek amacıyla Tablo 4.2'de detaylandırılmış eşik seviyesi ve güç tüketim değerlerini referans olarak kullanabiliriz.

Tablo 4.2: Uyku durumuna ait güç, gecikme süresi ve eşik seviyesi.

Uyku Durumu	$P_k (mW)$	$t_k (ms)$	$T_{th,k}$	Uyku Durumu	$P_k (mW)$	$t_k (ms)$	$T_{th,k}$
S0	138.582	X	X	S10	5.587	180	188
S1	137.448	X	X	S11	4.666	X	X
S2	136.371	X	X	S12	4.658	X	X
S3	136.362	X	X	S13	4.622	5	48
S4	79.723	X	X	S14	4.614	500	517
S5	79.671	X	X	S15	1.04	90	91
S6	79.662	10	24	S16	0.934	2000	2015
S7	72.923	100	211	S17	0.068	X	X
S8	19.195	180	209	S18	0.023	100	100
S9	18.146	120	138	S19	0.015	5000	5002
X: Uygulanamaz							

Tablo 4.3, Şekil 4.3'de sunulan D_k kablosuz algılayıcı düğümümüze yönelik kullanışlı uyku durumlarını ve bu durumlardaki güç tüketimlerini belirtmektedir.

Tablo 4.3: Kablosuz algılayıcı düğümümüz için kullanışlı uyku durumları.

Uyku Durumu	$P_k (mW)$	$t_k (ms)$	$T_{th,k}$
S0	138.582	X	X
S1	79.662	10	24
S2	72.923	100	211
S3	18.146	120	138
S4	5.587	180	188
S5	4.614	500	517
S6	0.015	5000	5002

Kablosuz algılayıcı düğüm için geliştirdiğimiz uyku durumu geçiş algoritması Program kodu 4.1’de verilmiştir:

```

void OlayIsle(){
    OlaySayisi++;
    long saniye = GecenSureyiBul();
    if (saniye == 0){
        setLambdaK(0);
    } else {
        setLambdaK((double) OlaySayisi/saniye);
    }
    for (int k = 0; k < DurumIslemleri.durumlar.size(); k++)
    {
        if (PthHesapla(TthkHesapla(k))) {
            uykuDurumunaGec(k);
        }
    }
}

```

Program kodu 4.1: Uyku durumu geçiş algoritması.

Kablosuz algılayıcı düğümümüz (D_k), bir olayı algıladığı zaman uyanır (mikroişlemci, uyku durumundayken algılama işleminin yapılması amacıyla öncelikle algılayıcılara güç verir. Algılayıcılar olayı algılamak için aktif hale geçer) ve olayı işler (algılayıcıları okuma, algılanmış verileri hesaplama, sınıflandırma, depolama, iletme vb. durumları içerir). Sonra, D_k düğümümüz tarafından kaydedilen toplam olay sayısını depolayan global sayaç (OlaySayisi) güncellenir. Daha sonra, D_k düğümü için olay ortalama ortaya çıkış oranı λ_k güncellenir. Bu, algılayıcı düğümün açıldığı andan itibaren geçen zamanı geri döndüren zamanlayıcı fonksiyonu kullanılarak yapılır. Yazılımımız; sistem tanımlı sabit $\rho_{th,0}$ ’a karşı ilgili $T_{th,k}$ uyku zamanı eşik seviyesinde meydana gelen bir olayın olasılığını test ederek algılayıcı düğümümüzü s_k (s_6 en derin uyku durumundan başlayarak s_1 ’e doğru) uyku durumuna sokmayı dener.

4.1.3. Kayıp Olaylar

Tablo 4.1 dikkatle incelenirse, $\{s_3, s_6, s_7, s_{12}, s_{14}, s_{16}, s_{17}, s_{18}, s_{19}\}$ durumları dışındaki bütün uyku durumlarının algılayıcılarla ilişkisi vardır. Dolayısıyla, eğer bir olay algılanırsa (eşik seviyesi düzeyinin üzerinde bir sinyal gücü oluşursa) algılayıcı düğüm s_0 durumuna geçiş yapar ve olayı işler. Bu durumda, sadece gecikme süresi ek yüke yol açar. Ancak, algılayıcı düğüm, s_6 durumunda (en derin uyku) neredeyse kapalı konumdadır ve mevcut saate bağlı olarak uyanacağı zamana kendisi karar verir. Seyrek olay algılama sisteminde (örneğin; nesne izleme, sismik algılama vb.), olaylar için interarrival (olaylar arasındaki) zamanı uyku zamanı eşik seviyesinden ($T_{th,k}$) çok daha büyüktür [147]. Bu nedenle, algılayıcı düğüm her zaman s_6 en derin uyku durumuna girer. Burada mikrodenetleyici, önceden programlanmış uyanma sinyallerini gözlemler. İşlemci, uyku durumuna giriş öncesi bu sinyal koşullarını programlar. Algılayıcı düğüm, kendiliğinden uyanması için bir sonraki olayın gelişini tahmin edebilmelidir. Basit bir yöntem, düğümün gereksiz yere uyanmasına yol açar. Kötü bir yöntem ise, bazı olayların kaçırılmasına (kaybedilmesi) sebep olur.

s_6 durumunda bulunma, algılayıcı düğüm uyarılmadıkça kaçırılmış olaylarla sonuçlanır. Geliştirilecek teknikler, algılama görevinin önem derecesine bağlı olarak tasarlanmalı ve kullanılmalıdır. s_6 durumunu göz önünde bulundurarak algılayıcı düğümümüz için iki muhtemel çözüm önerebiliriz:

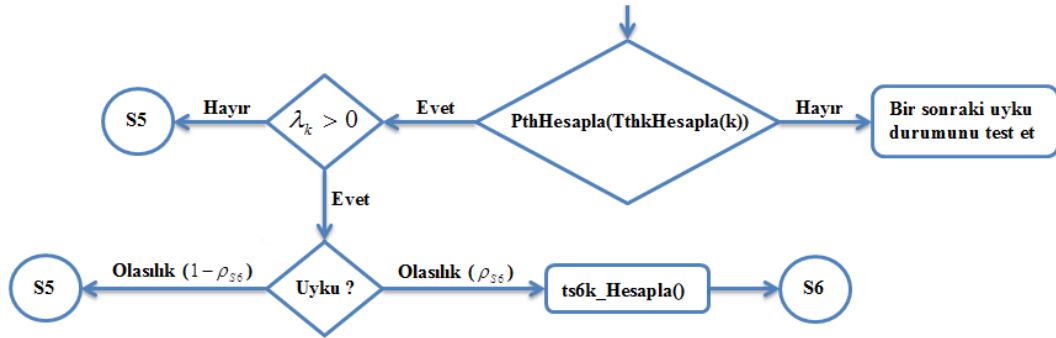
- ✓ s_6 durumunu tamamen iptal etmek: Eğer algılama görevi çok kritik ve olaylar kaçırılmaz ise bu duruma geçişe izin verilmez.
- ✓ s_6 durumunu seçime bağlı devre dışı bırakmak: Eğer olaylar uzaysal dağıtılmış ve tümüyle kritik değilse, s_6 durumu işletilebilir. Bu durumda, hem rasgele hem de deterministik yaklaşımlar kullanılabilir. Örneğin; bir kümeleme protokolünde normal düğümler s_6 durumuna geçebilirken, küme başları (koordinatör düğümler) s_6 durumunu iptal edebilir.

Çalışmamızda; olaylar tümüyle kritik değildir ve uzaysal dağıtılmıştır. Dolayısıyla; tasarladığımız kablosuz algılayıcı düğümümüz, pil seviye denetleyiciden gelen bilgileri

inceleyerek (örneğin; pil gerilimin 3.7 volttan küçük ya da büyük olması koşullarında) s_{14} , s_{15} ve s_{16} durumlarını seçime bağlı kullanmaktadır. Alternatif olarak, [147]'de önerilen şema da algılayıcı düğümümüze uyarlanabilir. Şekil 4.3'de gösterilen örnek KAA uygulama modelindeki her bir D_k algılayıcı düğüm; eğer s_6 durumuna yönelik uyku eşik seviyesi koşulunu sağlarsa, aşağıda verilen zaman süresi için bir sistem tanımlı olasılık (p_{s6}) ile uykuya girebilir [147].

$$t_{s6,k} = -\frac{1}{\lambda_k} \ln(p_{s6}) \quad (4.10)$$

Denklem 4.10, D_k algılayıcı düğümümüzün kararlı durum davranışını tanımlar. $t_{s6,k}$ 'da hiç bir olayın meydana gelmeme ihtimalinin $p_k(t_{s6,k}, 0) = p_{s6}$ olması için uyku zamanı hesaplanır. Ancak; gerçekte, KAA çalıştırıldığı zaman algılayıcı düğümlerde belirli bir süre hiçbir olay meydana gelmez ve bu durumda $\lambda_k = 0$ olur. KAA'nın açılmasıyla birlikte, düğümlerde en az bir olay algılanana kadar s_6 durumu devre dışı bırakılır. Dolayısıyla, başlangıçta sıfır olan ve olaylar algılandıkça artan bir adaptif geçiş olasılığı (p_{s6}), KAA uygulamasında bu hesaba katılabilir. Olasılıksal durum geçişi Şekil 4.5'de tanımlanmıştır.



Şekil 4.5: Olasılıksal uyku durumu geçişi.

4.1.4. Sonuçlar

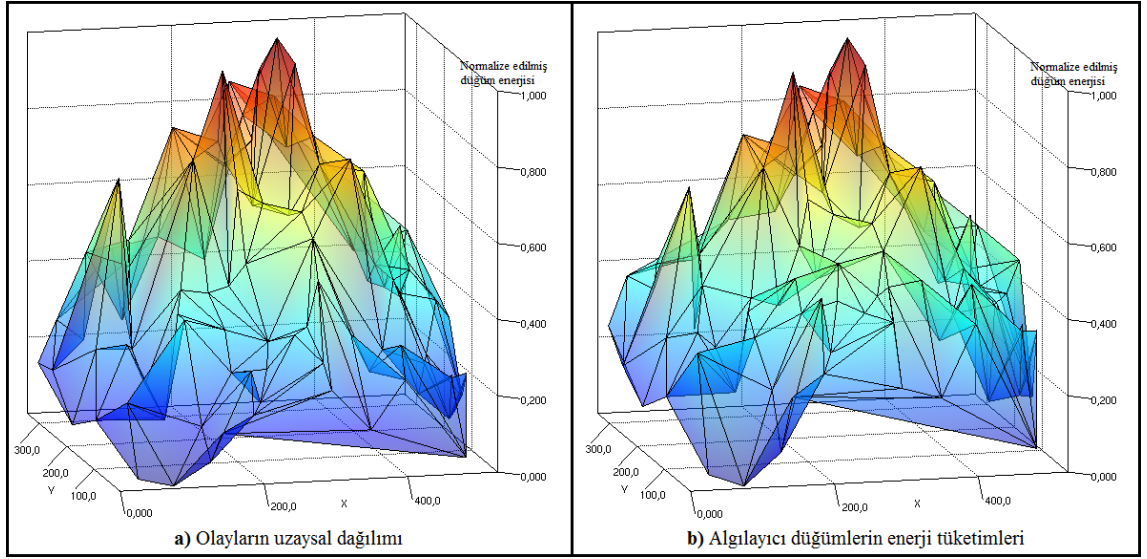
Kablosuz algılayıcı düğümümüz için önerdiğimiz yöntemin avantajı; olayları algılama olasılığı kullanılarak, algılayıcı düğümde enerji dengeleme işlemin verimli bir şekilde yapılabilmesidir. Bu yöntemde, p_{s6} 'yı artırarak sistem enerji tüketimi düşürülebilir; ancak, kaçırılan olayların olasılığı artar. Ayrıca, tam tersi durum da söz konusudur.

Yani; p_{s6} 'nın azalmasıyla sistem enerji tüketimi artarken, kaçırılan olayları algılama olasılığı da azalır. Bu nedenle, genel kapatma ilkesi, iki uygulamaya özel olasılık parametresi ($p_{th,0}$ ve p_{s6}) ile yönetilmiştir.

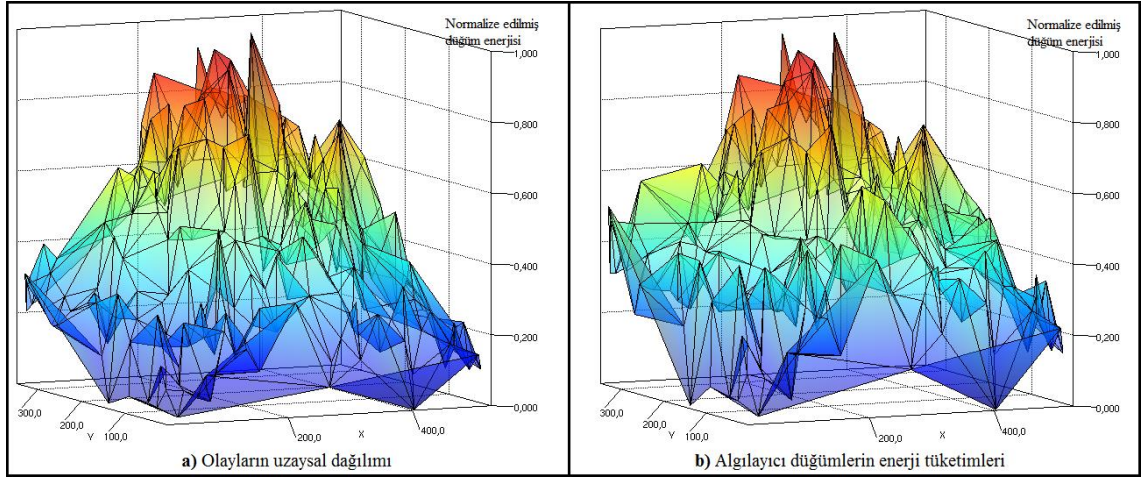
Çalışmamızda geliştirdiğimiz güç bilinçli sistem; uyku durumlarından ve aktif güç yönetiminden faydalanarak, kablosuz algılayıcı düğümümüzün enerji tüketimini azaltan bir yazılım kullanmaktadır. Sistem modelleri tasarlanırken, [147]'de sunulan çalışmayı referans aldık. [147]'de önerilen yöntemi, kablosuz algılayıcı düğüm için geliştirdiğimiz güç bilinçli sisteme uyarladık. Daha sonra, algılayıcı düğümümüzün enerji tüketimine yönelik davranışlarını incelemek üzere java programlama dili kullanarak bir simülasyon gerçekleştirdik. Simülasyonumuz Şekil 4.6'da gösterilmektedir. 400x400m alan üzerine homojen bir biçimde ve rasgele dağıtılmış $\eta=100/250/500$ adet algılayıcı düğümden oluşan bir KAA sistemini simule ettik. Her bir düğümün görünebilirlik yarıçapını $r=3m$ olarak varsaydık. Uyku durumu eşik seviyeleri ve güç tüketimlerine ait parametreler simülasyon içerisinde mevcuttur. Şekil 4.7, 4.8, 4.9; (100, 300) koordinatları çevresinde dönen bir Gaussian uzaysal dağılımlı olay için toplam uzaysal düğüm enerji tüketimini göstermektedir. 2 olay arasındaki interarrival süresi, saniyede ortalama 500 ($\lambda_{tot} = 500$) olan Poisson dağılımını izlemektedir. Şekil 4.7, 4.8 ve 4.9'dan görüleceği üzere; düğüm enerji tüketimi olay olasılığını izlemektedir. Güç yönetimi olmayan bir senaryoda, bütün düğümlerde uniform enerji tüketimi vardır.



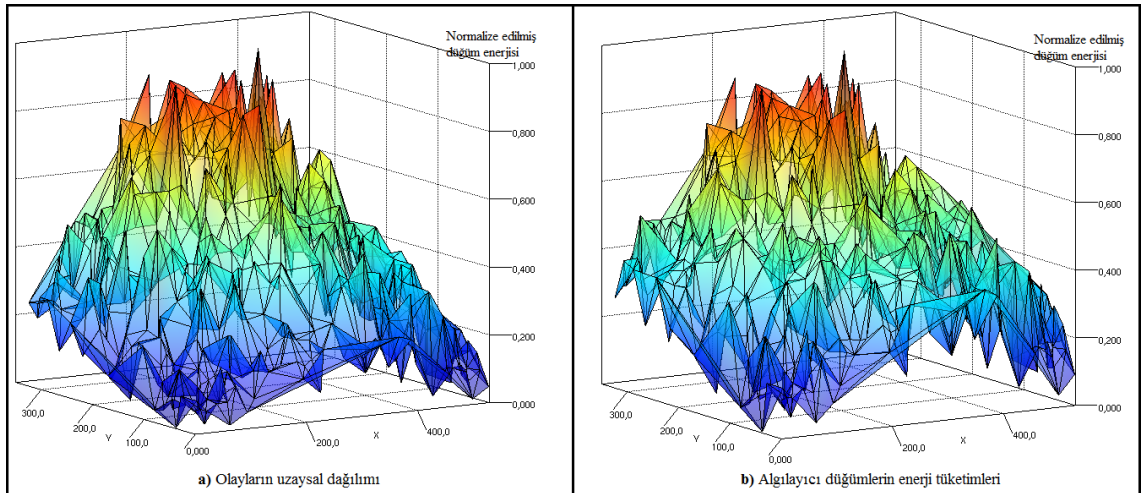
Şekil 4.6: Kablosuz algılayıcı düğümümüzün enerji tüketimlerini analiz eden simülasyon.



Şekil 4.7: Algılayıcı düğümlerimizin olaylara bağlı enerji tüketimleri ($\eta=100$ düğüm).



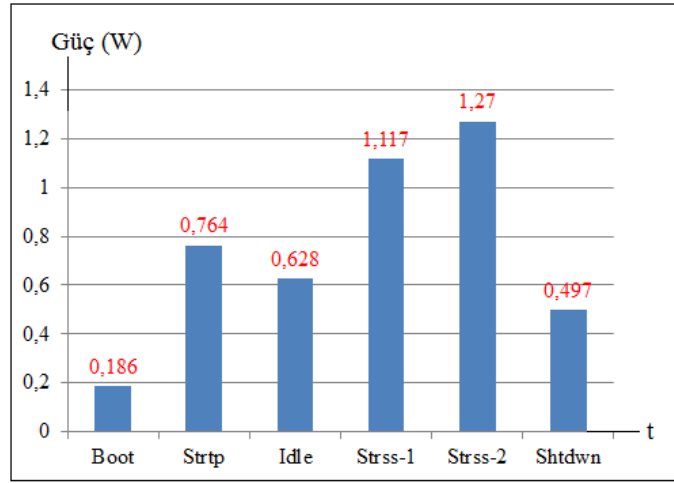
Şekil 4.8: Algılayıcı düğümlerimizin olaylara bağlı enerji tüketimleri ($\eta=250$ düğüm).



Şekil 4.9: Algılayıcı düğümlerimizin olaylara bağlı enerji tüketimleri ($\eta=500$ düğüm).

4.2. KOORDİNATÖR DÜĞÜMDE GÜÇ TASARRUFU

Çalışmamızda, KAA’da kullandığımız mevcut koordinatörün güç tasarruflu çalışmasını sağlamak amacıyla bir DGY sistemi tasarladık. Geliştirdiğimiz yazılım, kullandığımız şarj edilebilir Li-ion pil ve güneş paneli sayesinde koordinatörün de enerji verimli görev yapmasını sağladık. Şekil 4.10; üzerinde hiçbir iletişim modülü (WiFi, 3G, XBee) bulunmayan, DGY sistemi uygulanmamış, güç optimizasyonu ve kod optimizasyonu yapılmamış koordinatörün çalışma sırasında harcadığı güç miktarlarını göstermektedir.



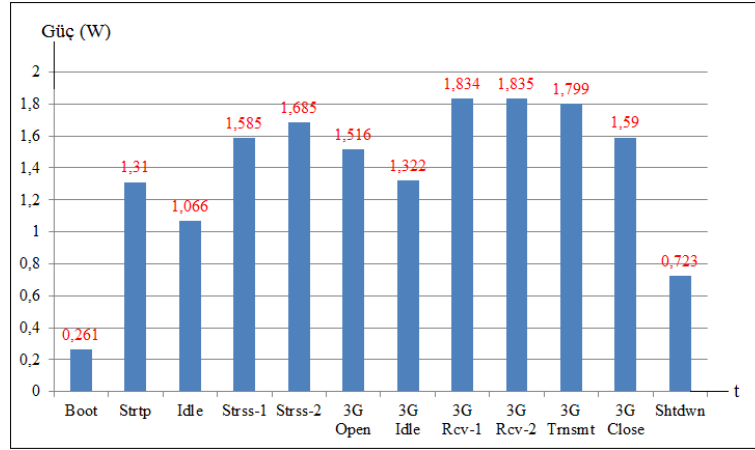
Şekil 4.10: Koordinatörün güç tüketimi.

Tablo 4.4, herhangi bir ağa (KAA, WLAN, İnternet) bağlı olmayan koordinatörün farklı çalışma durumlarında meydana gelen güç tüketimlerini belirtmektedir.

Tablo 4.4: Koordinatörün farklı çalışma durumlarına ait güç tüketimleri.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)
S1	Pasif	Pasif	Pasif	----	----	----	0.628
S2	Aktif	Aktif	Pasif	----	----	----	1.17
S3	Aktif	Aktif	Aktif	----	----	----	1.27

Şekil 4.11; üzerinde çeşitli iletişim modülleri (WiFi, 3G, XBee) bulunan, DGY sistemi uygulanmamış, güç optimizasyonu ve kod optimizasyonu yapılmamış koordinatörün İnternetle etkileşimi sırasında harcadığı güç miktarlarını göstermektedir. Bu durumda koordinatör; kablosuz algılayıcı düğümlerden gelen verileri işlemekte, İnternetteki ana veri toplama merkezine algılayıcı verilerini aktarmakta ve KAA’da senkronizasyon için sunucudan hava durumu bilgilerini almaktadır.



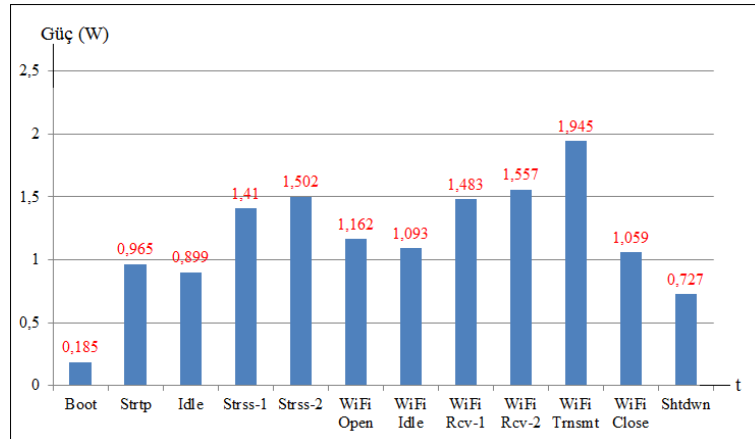
Şekil 4.11: İnternete bağlı koordinatörün güç tüketimi.

İnternete bağlı olan koordinatörün farklı çalışma durumlarında harcadığı güç miktarları Tablo 4.5'te sunulmaktadır.

Tablo 4.5: İnternete bağlı koordinatörün farklı çalışma durumlarındaki güç tüketimleri.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)
S1	Pasif	Pasif	Pasif	Kapalı	Kapalı	Kapalı	1.066
S2	Aktif	Aktif	Pasif	Kapalı	Kapalı	Kapalı	1.585
S3	Aktif	Aktif	Aktif	Kapalı	Kapalı	Kapalı	1.685
S4	Pasif	Pasif	Pasif	Kapalı	Pasif	Kapalı	1.322
S5	Aktif	Aktif	Aktif	Kapalı	Alma	Kapalı	1.835
S6	Aktif	Aktif	Aktif	Kapalı	Gönderme	Kapalı	1.799

Şekil 4.12; üzerinde WiFi, 3G, XBee iletişim birimlerini bulunduran, DGY sistemi uygulanmamış, güç optimizasyonu ve kod optimizasyonu yapılmamış koordinatörün WLAN içerisinde çalışırken harcadığı güç değerlerini göstermektedir.



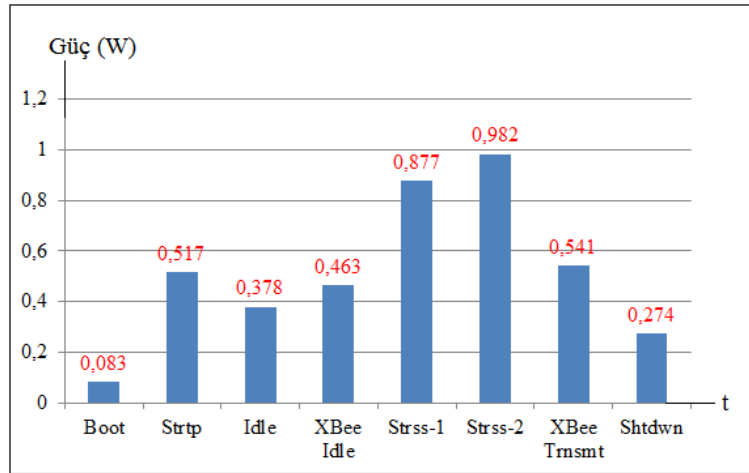
Şekil 4.12: WLAN içerisinde çalışan koordinatörün güç tüketimi.

Tablo 4.6, WLAN içerisindeki koordinatörün farklı çalışma durumlarında harcadığı güç miktarlarını tanımlamaktadır.

Tablo 4.6: WLAN'daki koordinatörün farklı çalışma durumları için güç tüketimleri.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)
S1	Pasif	Pasif	Pasif	Kapalı	Kapalı	Kapalı	0.899
S2	Aktif	Aktif	Pasif	Kapalı	Kapalı	Kapalı	1.41
S3	Aktif	Aktif	Aktif	Kapalı	Kapalı	Kapalı	1.502
S4	Pasif	Pasif	Pasif	Pasif	Kapalı	Kapalı	1.093
S5	Aktif	Aktif	Aktif	Alma	Kapalı	Kapalı	1.520
S6	Aktif	Aktif	Aktif	Gönderme	Kapalı	Kapalı	1.945

Şekil 4.13 ve Tablo 4.7; üzerinde WiFi, 3G, XBee iletişim modülleri olan, DGY sistemi uygulanmamış, güç optimizasyonu ve kod optimizasyonu yapılmamış koordinatörün KAA içerisinde çalışırken farklı durumlarda harcadığı güç değerlerini belirtmektedir. Bu durumlarda koordinatör; algılayıcı düğümlerden verileri toplamakta, işlemekte ve düğümlere senkronizasyon bilgisini yayımlamaktadır.

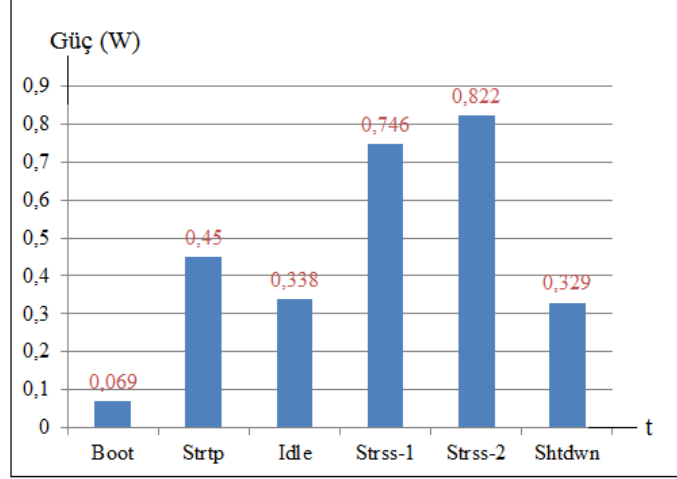


Şekil 4.13: KAA içerisinde çalışan koordinatörün güç tüketimi.

Tablo 4.7: KAA içerisindeki koordinatörün farklı çalışma durumlarına ait güç tüketimleri.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)
S1	Pasif	Pasif	Pasif	Kapalı	Kapalı	Uyku	0.378
S2	Aktif	Aktif	Pasif	Kapalı	Kapalı	Alma	0.877
S3	Aktif	Aktif	Aktif	Kapalı	Kapalı	Alma	0.982
S4	Pasif	Pasif	Pasif	Kapalı	Kapalı	Pasif	0.463
S5	Aktif	Aktif	Aktif	Kapalı	Kapalı	Gönderme	0.541

Şekil 4.14 ve Tablo 4.8; üzerinde hiçbir iletişim birimi (WiFi, 3G, XBee) bulunmayan, DGY uygulanmış, güç ve kod optimizasyonları yapılmış koordinatörün enerji tasarrufu sağlandıktan sonraki farklı durumlarda harcadığı güç miktarlarını göstermektedir.

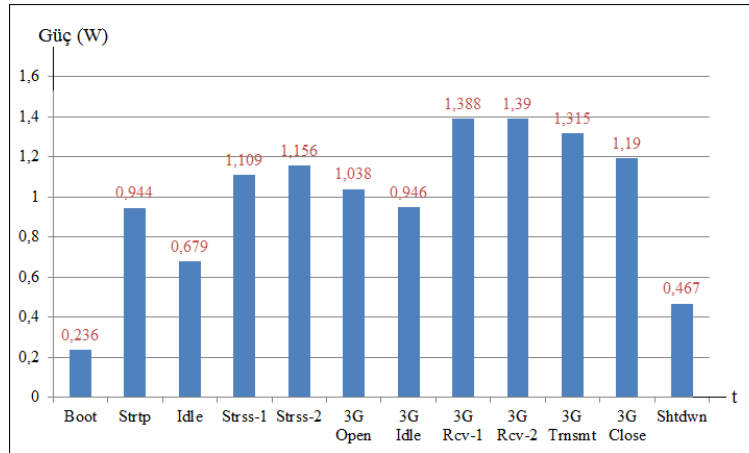


Şekil 4.14: Koordinatörde DGY ve optimizasyon işlemleri sonrası güç tüketimi.

Tablo 4.8: DGY ve optimizasyon işlemleri sonrasında koordinatörün farklı çalışma durumlarına yönelik güç tüketimleri ve enerji tasarrufları.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)	Enerji Tasarrufu
S1	Pasif	Pasif	Pasif	----	----	----	0.338	%46
S2	Aktif	Aktif	Pasif	----	----	----	0.746	%36
S3	Aktif	Aktif	Aktif	----	----	----	0.822	%35

Şekil 4.15; üzerinde WiFi, 3G, XBee iletişim modülleri bulunan, DGY uygulanmış, güç ve kod optimizasyonları yapılmış koordinatörün enerji tasarrufu sağlanmasıyla birlikte İnternetle etkileşimi sırasında harcadığı güç miktarlarını belirtmektedir.



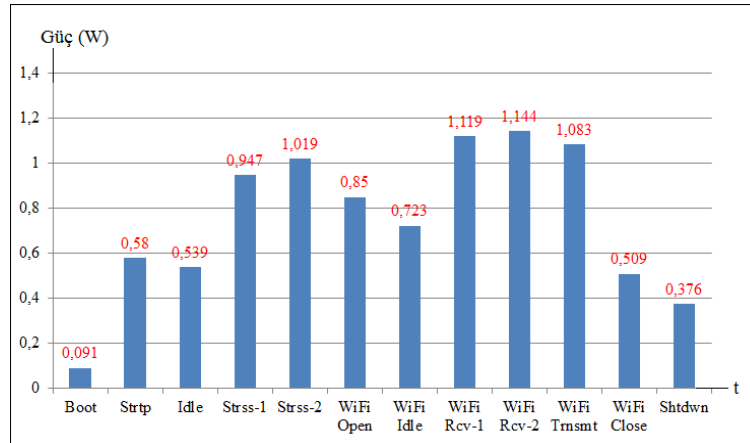
Şekil 4.15: İnternete bağlı enerji tasarruflu koordinatörün güç tüketimi.

DGY işlemi, güç ve kod optimizasyonları yapıldıktan sonra İnternete erişimi sağlanmış koordinatördeki güç tüketimleri ve enerji tasarrufları, Tablo 4.9’da sunulmaktadır.

Tablo 4.9: DGY ve optimizasyon işlemleri sonrası İnternete bağlı koordinatörün farklı çalışma durumları için güç tüketimleri ve enerji tasarrufları.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)	Enerji Tasarrufu
S1	Pasif	Pasif	Pasif	Kapalı	Kapalı	Kapalı	0.679	%36
S2	Aktif	Aktif	Pasif	Kapalı	Kapalı	Kapalı	1.109	%30
S3	Aktif	Aktif	Aktif	Kapalı	Kapalı	Kapalı	1.156	%31
S4	Pasif	Pasif	Pasif	Kapalı	Pasif	Kapalı	0.946	%28
S5	Aktif	Aktif	Aktif	Kapalı	Alma	Kapalı	1.39	%24
S6	Aktif	Aktif	Aktif	Kapalı	Gönderme	Kapalı	1.315	%27

Şekil 4.16 ve Tablo 4.10; üzerinde WiFi, 3G, XBee iletişim modülleri olan, DGY sistemi uygulanmış, güç ve kod optimizasyonları yapılmış koordinatörün WLAN içerisinde çalışırken harcadığı güç miktarlarını ve enerji tasarruflarını göstermektedir.

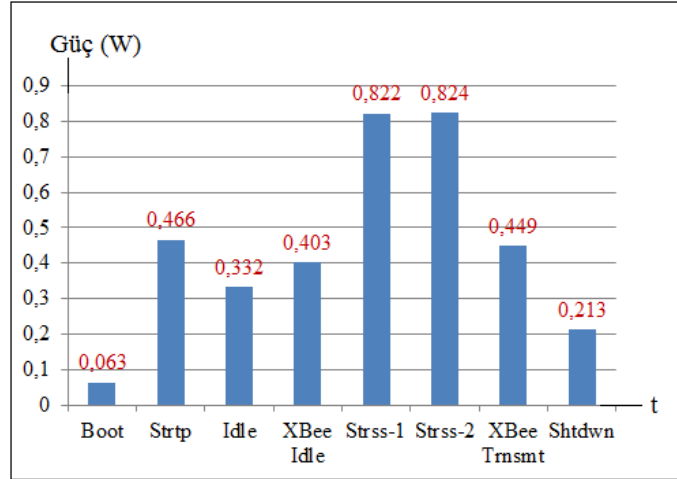


Şekil 4.16: WLAN içerisinde çalışan enerji tasarruflu koordinatörün güç tüketimi.

Tablo 4.10: WLAN içerisindeki koordinatörün DGY ve optimizasyon işlemleri sonrası farklı çalışma durumlarına ait güç tüketimleri ve enerji tasarrufları.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)	Enerji Tasarrufu
S1	Pasif	Pasif	Pasif	Kapalı	Kapalı	Kapalı	0.539	%40
S2	Aktif	Aktif	Pasif	Kapalı	Kapalı	Kapalı	0.947	%33
S3	Aktif	Aktif	Aktif	Kapalı	Kapalı	Kapalı	1.019	%32
S4	Pasif	Pasif	Pasif	Pasif	Kapalı	Kapalı	0.723	%34
S5	Aktif	Aktif	Aktif	Alma	Kapalı	Kapalı	1.132	%26
S6	Aktif	Aktif	Aktif	Gönderme	Kapalı	Kapalı	1.083	%44

Şekil 4.17 ve Tablo 4.11; üzerinde WiFi, 3G, XBee iletişim modülleri bulunan, DGY işlemi uygulanmış, güç ve kod optimizasyonları yapılmış koordinatörün KAA içerisinde harcadığı güç değerlerini ve sağlanan enerji tasarruflarını belirtmektedir.



Şekil 4.17: KAA'daki koordinatörde DGY ve optimizasyon işlemleri sonrası güç tüketimi.

Tablo 4.11: KAA'daki koordinatörün DGY ve optimizasyon işlemleri sonrası farklı çalışma durumlarına ait güç tüketimleri ve enerji tasarrufları.

Durumlar	CPU	RAM	HDD	WiFi	3G	XBee	Güç Tüketimi (W)	Enerji Tasarrufu
S1	Pasif	Pasif	Pasif	Kapalı	Kapalı	Uyku	0.332	%12
S2	Aktif	Aktif	Pasif	Kapalı	Kapalı	Alma	0.822	%6
S3	Aktif	Aktif	Aktif	Kapalı	Kapalı	Alma	0.824	%16
S4	Pasif	Pasif	Pasif	Kapalı	Kapalı	Pasif	0.403	%13
S5	Aktif	Aktif	Aktif	Kapalı	Kapalı	Gönderme	0.449	%17

KAA'larda kullanılan kablosuz algılayıcı düğümlerin iletişim mesafeleri sınırlıdır ve bulundukları ortama dağıtımları yoğun bir şekilde yapılır. Algılayıcı düğümlerin yoğun olarak konuşlandırıldığı ağ topolojilerinde, bir algılayıcı düğümdeki veri/paket, tek ya da çok sekmeli iletim yöntemiyle bir düğümden diğer düğüme aktarılarak hedef birime iletilir. Ancak, kaynak düğümden çıkan verilerin sürekli aynı düğümler üzerinden gönderilmesi, bu düğümlerin diğerlerine göre daha fazla enerji harcamasına sebep olur. Aynı düğümleri kullanmak yerine, komşu düğümlerin enerji seviyelerini dikkate alarak en uygun bir sonraki veri aktarıcı düğümü seçmek ve bu düğüm üzerinden veri göndermek, KAA'daki düğümler arasında daha adil bir yük dağılımının olmasını sağlar. Bu durumda, tek bir veri toplama merkezinin (sink, koordinatör, ağ geçidi) kullanıldığı KAA'larda, veri toplama merkezine yakın bölgede konuşlandırılmış düğümler, ağdaki

diğer düğümlerden gelen paketleri veri toplama merkezine iletebilecek tek elemandır. Diğer bir ifadeyle, KAA'daki düğümlerden gelen paketlerin veri toplama merkezine ulaşmadan önce uğrayacakları son nokta, veri toplama merkezi yakınındaki alanda (aktif bölgede) bulunan düğümlerdir. Aktif bölgede yer alan düğümlerin enerjilerinin ağdaki diğer bütün düğümlerin enerjilerinden önce tükenmesi, bu düğümlerin görevlerini tam anlamıyla yapamaması ve dolayısıyla KAA ömrünün sona ermesi anlamına gelir. Buradan hareketle, çalışmamızda, veri toplama merkezi olarak Linux tabanlı *Raspberry Pi* tek kartlı bilgisayardan birden fazla kullanarak aktif bölge alanını genişlettik. Aktif bölgede daha fazla düğümün görev yapmasını ve KAA'daki düğümler arasında eşit yük dağılımının gerçekleşmesini sağladık. Ayrıca, *Raspberry Pi* üzerinde çalışan, çevresel koşulları (örneğin; hava durumu, ortalama güneşlenme süresi, bulut yoğunluğu), algılayıcı düğümlerin sinyal seviyelerini, pil gerilimlerini, güç tüketimlerini inceleyerek senkronizasyon zamanını düzenleyen; KAA'da yerel ve global güç yönetimine katkı sağlayan, hava tahminine dayalı, bulanık mantık tabanlı, enerji verimli bir senkronizasyon protokolü gerçekleştirdik. Böylece, tasarladığımız KAA'nın enerji verimli çalışmasına katkı sağladık.

KAA'larda, enerji tüketimini azaltmak ve ağ ömrünü uzatmak için geliştirilmiş yöntemlerden biri de ağ topolojisini kümelerle ayırmaktır. Kümelemede, ağ içerisinde konuşlandırılmış bütün düğümler verilerini merkeze iletmez. Ağ yapısı kümelerle bölünür, her kümeye bir koordinatör atanır ve küme içerisindeki algılayıcı düğümler kendilerine has topladıkları verileri bu koordinatöre gönderirler. Koordinatör, sorumlu olduğu küme içerisindeki algılayıcı düğümlerden gelen verileri filtreleme, çözümleme ya da özetini çıkarma gibi işlemlere tabi tutarak sadeleştirir. Son olarak, koordinatör, bu sadeleşmiş verileri ya doğrudan veri toplama merkezine gönderir ya da kendisine en uygun komşu kümenin koordinatörüne iletir. Literatürde, KAA'lara yönelik geliştirilen kümeleme teknikleri üzerine birçok çalışma mevcuttur. Araştırmacıların çoğu genellikle küme-içi (intracluster) iletişim üzerine odaklanmıştır. Fakat, kümeler-arası (inter-cluster) iletişimde daha fazla enerji harcanmaktadır. Gelecek çalışmamızda; KAA'daki algılayıcı ve koordinatör düğümlerin sinyal seviyelerine ve güç tüketimlerine bağlı olarak dinamik kümeleme işlemi yapan, enerji verimli bir kümeleme protokolü geliştirmeyi hedeflemekteyiz. Böylece, hem küme içi hem de kümeler arası iletişimde enerji tasarruflu çalışan ve uzun ağ ömrüne sahip bir KAA gerçekleştirmiş olacağız.

5. TARTIŞMA VE SONUÇ

Algılama, fiziksel bir nesne veya süreç/olay hakkında bilginin alınması (toplanması), yorumlanması, seçilmesi ve düzenlenmesi için kullanılan bir tekniktir. Böyle bir algılama işlemi yerine getiren cihaza algılayıcı (sensör) adı verilir. Diğer bir ifadeyle algılayıcı; gerçek dünyaya ait olayları (sıcaklık, nem, basınç, ışık, ses, görüntü, kuvvet, elektrik, uzaklık, ivme, pH vb.) algılayan ve bunları işlenebilen, saklanabilen, uygulanabilen bir forma (ölçülebilir ve analiz edilebilir elektrik sinyalleri) dönüştürerek dijital dünya ile fiziksel dünyayı birbirine bağlayan küçük boyutlu, düşük maliyetli, otonom bir aygıttır. Algılayıcı, sınırlı kapasitede enerji kaynağına sahiptir ve genellikle pilden güç sağlar. Pilin bitmesi durumunda değiştirilmesi ya da yeniden şarj edilmesi gerekir. Pilin şarj edilebilip edilememesi, algılayıcının enerji tüketim stratejisini ve çalışma süresini önemli derecede etkiler.

KAA uygulamalarının çoğu, uzak ve ulaşılamaz bölgelerde konuşlandırılmış çok sayıda algılayıcı düğüm gerektirir. Bu nedenle, bir algılayıcı düğüm sadece algılama değil, aynı zamanda veri işleme, çözümleme, depolama, çevresinden topladığı verileri kablolu ya da kablosuz iletişim aracılığıyla merkezi birimlere (örneğin; denetleyici, sunucu, iş istasyonu, veri toplama merkezi) aktarabilme, işbirlikçi çalışabilme becerilerine sahiptir. KAA'daki algılayıcı düğümlerin yetenekleri yaygın olarak değişiklik gösterebilir. Örneğin; basit yapıdaki algılayıcı düğümler gözlemlenen ortam hakkında tek bir fiziksel olayı izleyebilirken, kompleks özellikteki algılayıcılar (işlem, enerji ve depolama kapasitesi yüksek olan aygıtlar) daha kapsamlı işlemleri yerine getirebilir.

KAA'larda, algılayıcı düğümler ile alıcı (koordinatör, ağ geçidi, baz istasyonu) arasında artan uzaklık gerekli iletim gücünü ani bir şekilde artırır. Enerji verimliliği bakımından geniş mesafeyi birden fazla kısa mesafelere bölmek daha avantajlıdır; ancak, çok sekmeli iletişim ve yönlendirmeyi destekleme zorluğuna yol açar. Çok sekmeli iletişim, bir ağdaki düğümlerin, etkin rotaları tespit etme ve röle olarak hizmet vermede organize çalışmasını gerektirir. Bu zorluk, enerjiyi korumak için hizmet çevrimleri uygulayan ağlarda daha da artar. Çünkü, birçok algılayıcı düğüm, boşta kaldığı

durumlarda radyoların kapatıldığı bir güç koruma politikası işler. Kullanım dışı zamanlarda algılayıcı düğüm, ne komşularından mesaj alabilir ne de diğer düğümlere röle hizmeti verebilir. Bazı ağlar, gerektiğinde düğümlerin uyandırılmasını sağlamak için isteğe bağlı uyanma stratejilerine sahiptir. Bu özellik genellikle iki radyo bulunduran aygıtları içerir. Düşük güç radyo, uyandırma çağrılarını almada kullanılırken yüksek güç radyo da uyandırma çağrısına cevap olarak etkinleştirilir. Başka bir strateji de adaptif hizmet çevrimidir. Burada tüm düğümlerin aynı zamanda uyku konumuna geçmesine izin verilmez. Bunun yerine, ağdaki düğümlerin bir alt kümesi, ağ omurgası oluşturmak üzere aktif kalır.

KAA’larda geniş ölçekli konuşlandırılmalar ve enerji kısıtlamaları, topoloji yönetimi ya da yönlendirme gibi ağ yönetim çözümleri uygulamada merkezi algoritmaların yürütülmesini olanaksız kılar. Bunun yerine, algılayıcı düğümler, global bilgi gerekmeksizin lokal kararlar vermede komşuları ile işbirliği yapmalıdır. Dağıtık algoritmalar, optimal sonuçlar vermeyebilir ama merkezi algoritmalara göre daha enerji verimli çözümler sunabilirler. Merkezi ve dağıtık algoritmalara yönlendirmeyi örnek verebiliriz. Bir baz istasyonu, tüm algılayıcı düğümlerden bilgi toplayabilir, optimal rotalar belirleyebilir (enerji açısından) ve rotasındaki her düğümü bilgilendirebilir. Ancak, özellikle ağ topolojisi sık sık değişiyorsa ciddi miktarda ek yük oluşabilir. Dağıtık mimari yaklaşımları, her düğümün yönlendirme işlemini sınırlı yerel bilgiye (örneğin; bir düğümün komşularının listesi, düğümlerin baz istasyonuna mesafeleri) dayalı gerçekleştirmesini sağlarlar. Optimal olmayan rotalara yol açsalar da, yönetimden kaynaklanan ek yükleri önemli ölçüde azaltabilirler.

KAA’lar, doğaları gereği uzak yerlerde ve zorlu ortamlarda işlem yapar. Altyapı desteği, bakım ve onarım imkanları yoktur. Ayrıca, güvenilir KAA’lar oluşturmak için en önemli tasarım zorluğu enerji verimliliğidir. Enerji tüketimi, ağ topolojisine her yönüyle nüfuz eder. Bu nedenle, KAA’lar; kendi kendini yapılandırma (çeşitli kurulum ve konfigürasyon adımlarını otonom gerçekleştirme), işbirliği içinde çalışma, konumlarını belirleme, ortam değişikliklerine ve çevresel uyaranlara insan müdahalesi olmadan uyum sağlama yeteneklerine sahip olmalıdır. Kendi sistem kaynaklarının kullanımını izlemeli ve optimize etmelidir. Dışardan gelebilecek müdahale ve saldırıları tanımalı, kendilerini bu durumlara karşı korumalıdır. Ağ aksaklıklarını keşfetmeli,

arızalara/hatalara karşı tepki vererek kendilerini iyileştirmelidirler. Sonuç olarak, kısıtlı enerji kaynaklarına sahip KAA'larda tüm bu öz-yönetim özellikleri, aşırı enerji tüketimleri yaratmayacak şekilde tasarlanmalı ve uygulanmalıdır.

KAA tasarımlarında amaç; daha küçük, daha ucuz ve daha verimli algılayıcı cihazlar gerçekleştirmektir. Özel uygulamaları “az enerji harcayarak çalıştırma” gereksinimi ile işleyen algılayıcı düğümler, düşük işlem hızlarına ve depolama kapasitelerine sahiptir. Küçük form faktörü ve düşük enerji tüketimi, GPS alıcıları gibi birçok arzu edilen bileşenin kablosuz algılayıcı cihazlara entegrasyonunu önlemektedir. Bu sınırlamalar ve gereksinimler, çeşitli seviyelerde yazılım tasarımlarını etkilemektedir. Örneğin; işletim sistemleri küçük bellek alanlarına sahip olmalı, kaynak yönetimi görevlerini verimli bir biçimde gerçekleştirmelidir. Gelişmiş donanım özelliklerinin eksikliği (örneğin, paralel çalışmalara destek) küçük ve etkin işletim sistemlerinin tasarımını kolaylaştırmaktadır. Bir algılayıcı cihazın donanım kısıtlamaları, KAA'da yürütülen birçok protokol ve algoritmanın tasarımını da etkilemektedir. Örneğin, bir ağ içindeki her bir potansiyel hedefin girdilerini içeren yönlendirme tabloları, algılayıcı cihazın belleğine sığmayacak kadar büyük olabilir. Bunun yerine, sadece az miktarda veri (örneğin, komşuların bir listesi gibi) algılayıcı düğümün hafızasında saklanabilir. Ağ-içi işlemler gereksiz/artık bilgileri ortadan kaldırmada kullanılabilirken, bazı birleştirme/kümeleme algoritmaları düşük maliyetli algılayıcı düğümler ile sağlanabilenden daha fazla hesaplama gücü ve depolama kapasitesi gerektirebilir. Bu nedenle, birçok yazılım mimarisi ve çözümü (işletim sistemleri, ara katman yazılımları, ağ protokolleri) kaynak kısıtlı donanım üzerinde verimli bir şekilde çalışmak için tasarlanmış olmalıdır.

KAA'daki tasarım seçeneklerinin birçoğu, diğer sistemlerin ve ağların tasarım seçeneklerinden farklıdır. Tablo 5.1, geleneksel ağlar ile KAA'lar arasındaki bazı önemli farkları özetlemektedir. Çeşitli ek zorluklar, algılayıcı düğümlerin ve KAA'ların tasarımını etkileyebilir. Örneğin; bazı algılayıcılar, araç ya da robot gibi hareketli nesnelere monte edilebilirler. Bu da; yönlendirme (komşu listelerini yenileme), ortam erişim denetimi (yoğunluk değiştirme), veri toplama (üst üste binen algılayıcı bölgelerini değiştirme) gibi bir sistemin çoklu katmanlarında sık adaptasyonlar gerektiren ağ topolojilerini sürekli değiştirmeye sebebiyet verir. Heterojen bir KAA, değişken donanım becerilerine sahip aygıtlardan oluşur. Örneğin; eğer algılayıcı

görevleri daha fazla hesaplama ve depolama gerektiriyorsa ya da ağdaki diğer algılayıcılardan veri toplamak ve bunları işleme almaktan sorumlu iseler, algılayıcı düğümler daha fazla donanım kaynağı içerirler. Ayrıca, bazı algılayıcı uygulamalarının spesifik performans ve kalite gereksinimleri olabilir. Mesela, kritik algılayıcı olaylarında düşük gecikme süreleri veya video algılayıcıları tarafından toplanan veriler için yüksek verimlilik gibi. Hem heterojenlik hem de performans gereksinimleri, kablolu algılayıcı cihazların tasarımlarını ve bu cihazlara yönelik protokollerin geliştirilmesini etkiler. Geleneksel bilgisayar ağları oturmuş standartlara dayalı olsa da, KAA'lardaki birçok protokol ve mekanizma özel çözümlere ihtiyaç duyar. Standartlar, birlikte çalışabilirlik için çok önemlidir ve KAA uygulamalarının tasarım ve konuşlandırılmasını kolaylaştırırlar. Bu nedenle, KAA tasarımında önemli bir zorluk, umut verici çözümlerin standardizasyonu ve rekabet içindeki standartların birbirleriyle uyumlaştırılmasıdır.

Tablo 5.1: Geleneksel ağlar ile KAA'ların karşılaştırılması.

Geleneksel Ağlar	KAA
- Genel amaçlı tasarım. - Birçok uygulamaya hizmet verir.	- Özel amaçlı tasarım. - Belirli bir uygulamaya hizmet verir.
- Tasarım kaygıları, ağ performansı ve gecikmelerdir. - Enerji, temel sorun değildir.	- Tüm düğüm ve ağ bileşenlerinin tasarımlarında enerji (güç tüketimi) ana problemdir.
- Ağlar planlı tasarlanır ve oluşturulur.	- Dağıtım, konuşlandırma, ağ yapısı ve kaynak kullanımı genelde tasarsızdır.
- Ağlar ve ağ cihazları genellikle denetlenen ve hafif koşullara sahip ortamlarda çalıştırılır.	- Genellikle sert koşullara sahip ortamlarda çalıştırılır.
- Bakım ve onarım birlikte yapılır. - Ağlara erişim kolaydır.	- Algılayıcı düğümlere fiziksel olarak erişim genellikle zordur, hatta imkansızdır.
- Bileşen arızası, bakım ve onarım ile ele alınır.	- Bileşen arızası ağ tasarımında beklenir ve ele alınır.
- Global ağ bilgisi almak ve merkezi yönetim mümkündür.	- Çoğu karar, merkezi yöneticinin desteği olmadan lokalize verilir.

Birçok KAA, konuşlandırıldıkları bölgeden hassas bilgiler toplar. Algılayıcı düğümlerin uzaktan ve gözetimsiz yürütülen işlemleri, kötü niyetli müdahale ve saldırılara maruz kalma riskini artırır. Kablosuz iletişim, bir düşmanın algılayıcı düğümün yayınlarını gizlice dinlemesini kolaylaştırır. Örneğin, amacı KAA operasyonunu bozmak olan hizmet-engelleme-saldırısı, en zorlu güvenlik tehditlerden biridir. Bu, başarılı algılayıcı

düğüm iletişimlerini önlemede yüksek güçlü kablosuz sinyallerin kullanıldığı çeşitli saldırılarla (sinyal/yayın bozma, parazit yapma, frekans karıştırma) gerçekleştirilebilir. Sonuçlar, KAA uygulamasının türüne göre değişir ve kötü olabilir. Dağıtık sistemler için saldırıları önleyen veya bu tür saldırıların boyutunu, zararını azaltan çok sayıda yöntem ve çözüm bulunsa da, bunların çoğu, kaynak kısıtlı algılayıcı düğümler tarafından genellikle karşılanamayan ciddi hesaplama, iletişim ve depolama ihtiyaçları yaratır. Bu nedenle, KAA'lar; anahtar değişimi ve dağıtımı, düğüm kimlik doğrulaması, gizlilik gibi konular için yeni çözümler gerektirir.

KAA'larda, çevreye ait fiziksel olayları algılama, algılanan verileri işleme ve hedef birimlere iletme gibi işlevleri gerçekleştirmek için enerji tasarrufu sağlamakla birlikte, bilgilerin dağıtık hesaplama teknikleri kullanılarak gerçek zamanlı ve etkin servis kalitesi sağlayacak şekilde ilgili merkezi birimlere gönderilmesi gerekir. Ayrıca, dağıtık ya da bir arada bulunan, paylaşılabilen ve ayarlanabilen kaynaklara (ağ cihazları, bilgisayarlar, sunucular, iş istasyonları, depolama birimleri, web servisleri ve diğer uygulamalar) kablolu/kablosuz ağ ya da İnternet üzerinden kullanıcıların istekleri doğrultusunda erişim sağlanması gerekebilir. Bu nedenle, gelecek çalışmamız olarak; KAA'lar ile web servisleri, sosyal ağlar, bulut bilişim, mobil iletişim gibi teknolojileri birleştirecek sistem tasarımlarının gerçekleştirilmesi hedeflenmektedir.

Sonuç olarak: KAA'lar üzerine yapılan araştırmalar ve geliştirilen teknolojiler; küçük boyutlu, düşük maliyetli, enerji tasarruflu (düşük güçlü), organize çalışan, kompleks özellikte algılayıcı düğümlerle birlikte aktüatör ve denetleyici sistemlerin gelişmesine, buna bağlı olarak ortam-bilinçli, akıllı sistemlerin kullanıldığı yeni uygulama alanlarının ortaya çıkmasına ve yaygınlaşmasına, dolayısıyla çok önemli toplumsal faydaların sağlanmasına neden olmuştur.

KAYNAKLAR

- [1]. Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., Cayirci, E., 2002, Wireless sensor networks: a survey, *Computer Networks*, 38 (4), 393-422.
- [2]. Lewis, F. L., 2005, *Wireless sensor networks*, Smart environments: technologies, protocols, and applications, In: Cook, D. J. (ed.), Das, S. K. (ed.), Chapter 2, John Wiley & Sons, Hoboken, NJ, USA, ISBN: 978-0-471-54448-7, 13-46.
- [3]. Raghunathan, V., Schurgers, C., Park, S., Srivastava, M. B., 2004, *Energy efficient design of wireless sensor nodes*, Wireless sensor networks, In: Raghavendra, C. S. (ed.), Sivalingam, K. M. (ed.), Znati, T. (ed.), Chapter 3, Springer, NY, USA, ISBN: 1-4020-7883-8, 51-69.
- [4]. Rakhmatov, D., Vruthula, S., 2003, Energy management for battery-powered embedded systems, *ACM Transactions on Embedded Computing Systems*, 2 (3), 277-324.
- [5]. Raghunathan, V., Ganeriwal, S., Srivastava, M. B., Schurgers, C., 2004, Energy efficient wireless packet scheduling and fair queuing, *ACM Transactions on Embedded Computing Systems*, 3 (1), 3-23.
- [6]. Schurgers, C., Raghunathan, V., Srivastava, M. B., 2003, Power management for energy-aware communication systems, *ACM Transactions on Embedded Computing Systems*, 2 (3), 431-447.
- [7]. Patwari, N., Ash, J. N., Kyperountas, S., Hero, A. O., Moses, R. L., Correal, N. S., 2005, Locating the nodes: cooperative localization in wireless sensor networks, *IEEE Signal Processing Magazine*, 22 (4), 54-69.
- [8]. Langendoen, K., Reijers, N., 2003, Distributed localization in wireless sensor networks: a quantitative comprasion, *Computer Networks*, 43 (4), 499-518.
- [9]. Mao, G., Fidan, B., Anderson, B. D. O., 2007, Wireless sensor network localization techniques, *Computer Networks*, 51 (10), 2529-2553.
- [10]. Ramadurai, V., Sichitiu, M. L., 2003, Localization in wireless sensor networks: a probabilistic approach, *The International Conference on Wireless Networks*, 23-26 June 2003, Las Vegas, Nevada, USA, 275-281.
- [11]. Jiang, X., Taneja, J., Ortiz, J., Tavakoli, A., Dutta, P., Jeong, J., Culler, D., Levis, P., Shenker, S., 2007, An architecture for energy management in wireless sensor networks, *Special Issue on the International Workshop on Wireless Sensor Network Architecture*, *ACM SIGBED Review*, 4 (3), 31-36.

- [12]. Alippi, C., Anastasi, G., Francesco, M. D., Roveri, M., 2009, Energy management in wireless sensor networks with energy-hungry sensors, *IEEE Instrumentation and Measurement Magazine*, 12 (2), 16-23.
- [13]. Lin, T.-H., Kaiser, W. J., Pottie, G. J., 2004, Integrated low-power communication system design for wireless sensor networks, *IEEE Communications Magazine*, 42 (12), 142-150.
- [14]. Akan, Ö. B., Akyildiz, I. F., 2005, Event-to-sink reliable transport in wireless sensor networks, *IEEE/ACM Transactions on Networking*, 13 (5), 1003-1016.
- [15]. Jamieson, K., Balakrishnan, H., Tay, Y. C., 2006, Sift: a MAC protocol for event-driven wireless sensor networks, *The 3rd European Conference on Wireless Sensor Networks*, 13-15 February 2006, Zurich, Switzerland, 260-275.
- [16]. Levin, L., Efrat, A., Segal, M., 2014, Collection data in ad-hoc networks with reduced uncertainty, *Ad Hoc Networks*, 17, 71-81.
- [17]. Akkaya, K., Younis, M., 2005, A survey on routing protocols for wireless sensor networks, *Ad Hoc Networks*, 3 (3), 325-349.
- [18]. Rao, J., Biswas, S., 2012, Analyzing multi-hop routing feasibility for sensor data harvesting using mobile sinks, *Journal of Parallel and Distributed Computing*, 72 (6), 764-777.
- [19]. Biswas, S., Morris, R., 2005, ExOR: opportunistic multi-hop routing for wireless sensor networks, *ACM SIGCOMM Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, 22-26 August 2005, Philadelphia, PA, USA, 133-144.
- [20]. Akyildiz, I. F., Vuran, M. C., 2010, *Wireless sensor networks*, John Wiley & Sons, West Sussex, UK, ISBN: 978-0-470-03601-3.
- [21]. Callaway Jr., E. H., 2003, *Wireless sensor networks architecture and protocols*, Auerbach Publications, CRC Press, FL, USA, ISBN: 978-0-8493-1823-8.
- [22]. El Emary, I. M. M., Ramakrishnan, S., 2013, *Wireless sensor networks: from theory to applications*, CRC Press, FL, USA, ISBN: 978- 1-4665-1810-0, 2013.
- [23]. Sohraby, K., Minoli, D., Znati, T., 2007, *Wireless sensor networks: technology, protocols, and applications*, John Wiley & Sons, NJ, USA, ISBN: 978-0-471-74300-2.
- [24]. Dargie, W., Poellabauer, C., 2010, *Fundamentals of wireless sensor networks: theory and practice*, John Wiley & Sons, West Sussex, UK, ISBN: 978-0-470-99765-9.
- [25]. Raghavendra, C. S., Sivalingam, K. M., Znati, T., 2004, *Wireless sensor networks: technology, protocols, and applications*, Springer, NY, USA, ISBN: 978-1-40207883-5.

- [26]. Whitehouse, K., 2006, *Understanding the prediction gap in multi-hop localization*, Ph.D. dissertation, University of California at Berkeley.
- [27]. Whitehouse, K., Jiang, F., Woo, A., Karlof, C., Culler, D., 2004, Sensor field localization: a deployment and empirical analysis, *UC Berkeley Technical Report UCB/CSD-04-1349*.
- [28]. Min, R., Bhardwaj, M., Cho, S.-H., Shih, E., Sinha, A., Wang, A., Chandrakasan, A., 2001, Low-power wireless sensor networks, *14th International Conference on VLSI Design*, 03-07 January 2001, Bangalore, India, 205-210.
- [29]. Ye, W., Heidemann, J., Estrin, D., 2002, An energy efficient MAC protocol for wireless sensor networks, *21st Annual Joint Conference of the IEEE Computer and Communications Societies*, 23-27 June 2002, NY, USA, 3, 1567-1576.
- [30]. Hahm, O., Adler, S., 2012, Profiling energy consumption of wireless sensor nodes with almost zero effort, *IEEE International Conference on Communications*, 10-15 June 2012, Ottawa, Canada, 6273-6277.
- [31]. Thandapani, S., Kailas, A., 2012, An accurate energy consumption model for physical layer in a wireless mote, *IEEE International Conference on Communications*, 10-15 June 2012, Ottawa, Canada, 6283-6287.
- [32]. Ilow, J., Rangappa, S., Aslam, N., 2012, Energy efficient broadcasting in WSNs with cocasting and power control, *IEEE International Conference on Communications*, 10-15 June 2012, Ottawa, Canada, 6288-6292.
- [33]. Aspnes, J., Whiteley, W., Yang, Y. R., 2006, A theory of network localization, *IEEE Transactions on Mobile Computing*, 5 (12), 1663-1678.
- [34]. Huseth, S., Kolavennu, S., 2011, *Localization in wireless sensor networks*, Wireless Networking Based Control, In: Mazumder, S. K. (ed.), Chapter 7, Springer, New York, USA, ISBN: 978-1-4419-7393-1, 153-174.
- [35]. Gentile, C., Alsindi, N., Raulefs, R., Teolis, C., 2013, *Geolocation techniques: principles and applications*, Springer, New York, USA, ISBN: 978-1-4614-1835-1.
- [36]. Beutel, J., Römer, K., Ringwald, M., Woehrle, M., 2009, *Deployment techniques for sensor networks*, Sensor Networks: Where Theory Meets Practice, Series: Signals and Communication Technology, In: Gianluigi, F. (ed.), Chapter 3, Springer, Berlin, Germany, ISBN: 978- 3-642-01341-6, 219-248.
- [37]. Deif, D. S., Gadallah, Y., 2013, Classification of wireless sensor networks deployment techniques, *IEEE Communications Surveys&Tutorials*, PP, 1-22.
- [38]. Priyantha, N. B., Balakrishnan, H., Demaine, E. D., Teller, S., 2005, Mobile-assisted localization in wireless sensor networks, *The 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, 13-17 March 2005, Miami, FL, USA, 1, 172-183.

- [39]. He, T., Huang, C., Blum, B., Stankovic, J., Abdelzaher, T., 2003, Range-free localization schemes for large scale sensor networks, *The 9th Annual International Conference on Mobile Computing and Networking*, 14-19 September 2003, San Diego, CA, USA, 81-95.
- [40]. Vuran, M. C., Akan, Ö. B., Akyildiz, I. F., 2004, Spatio-temporal correlation: theory and applicaitons for wireless sensor networks, *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 45 (3), 245-259.
- [41]. Vuran, M. C., Akyildiz, I. F., 2006, Spatial correlation-based collaborative Medium Access Control in wireless sensor networks, *The IEEE/ACM Transactions on Networking*, 14 (2), 316-329.
- [42]. Akyildiz, I. F., Wang, X., Wang, W. 2005, Wireless mesh networks: a survey, *Computer Networks*, 47 (4), 445-487.
- [43]. Guo, W., Zhang, W., 2014, A survey on intelligent routing protocols in wireless sensor networks, *Journal of Network and Computer Applications*, 38, 185-201.
- [44]. Kansal, A., Hsu, J., Zahedi, S., Srivastava, M. B., 2007, Power management in energy harvesting sensor networks, *ACM Transactions on Embedded Computing Systems*, 6 (4), 1-35.
- [45]. Moser, C., 2009, *Power Management in Energy Harvesting Embedded Systems*, Ph.D. dissertation, Swiss Federal Institute of Technology.
- [46]. Luo, R.C., Chen, O., 2012, Mobile sensor node deployment and asynchronous power management for wireless sensor networks, *IEEE Transactions on Industrial Electronics*, 59 (5), 2377-2385.
- [47]. Dong, Q., Dargie, W., 2013, A survey on mobility and mobility-aware MAC protocols in wireless sensor networks, *IEEE Communications Surveys & Tutorials*, 15 (1), 88-100.
- [48]. Wang, Y.-C., Wu, F.-J., Tseng, Y.-C., 2010, Mobility management algorithms and applications for mobile sensor networks, *Wireless Communications and Mobile Computing*, 12 (1), 7-21.
- [49]. Zinonos, Z., Chrysostomou, C., Vassiliou, V., 2014, Wireless sensor networks mobility management using fuzzy logic, *Ad Hoc Networks*, 16, 70-87.
- [50]. AbdelSalam, H. S., Olariu, S., 2014, *Energy-efficient task management*, The Art of Wireless Sensor Networks, Volume 1: Fundamentals, Series: Signals and Communication Technology, In: Ammari, H. M. (ed.), Chapter 12, Springer, Berlin, Germany, ISBN: 978-3-642-40009-4, 385-425,
- [51]. AbdelSalam, H.S., Olariu, S., 2011, Toward efficient task management in wireless sensor networks, *IEEE Transactions on Computers*, 60 (11), 1638-1651.

- [52]. Noh, K.-L., Serpedin, E., Qaraqe, K., 2008, A new approach for time synchronization in wireless sensor networks: pairwise broadcast synchronization, *IEEE Transactions on Wireless Communications*, 7 (9), 3318-3322.
- [53]. Wu, Y.-C., Chaudhari, Q., Serpedin, E., 2011, Clock synchronization of wireless sensor networks, *IEEE Signal Processing Magazine*, 28 (1), 124-138.
- [54]. Schenato, L., Fiorentin, F., 2011, Average TimeSync: A consensus-based protocol for clock synchronization in wireless sensor networks, *Automatica*, 47 (9), 1878-1886.
- [55]. Leng, M., Wu, Y.-C., 2011, Distributed clock synchronization for wireless sensor networks using belief propagation, *IEEE Transactions on Signal Processing*, 59 (11), 5404-5414.
- [56]. Wu, J., Jiao, L., Ding, R., 2012, Average time synchronization in wireless sensor networks by pairwise messages, *Computer Communications*, 35 (2), 221-233.
- [57]. Huang, G., Zomaya, A. Y., Delicato, F. C., Pires, P. F., 2014, Long term and large scale time synchronization in wireless sensor networks, *Computer Communications*, 37, 77-91.
- [58]. Lazos, L., Poovendran, R., 2005, SeRLoc: Robust localization for wireless sensor networks, *ACM Transactions on Sensor Networks*, 1 (1), 73 – 100.
- [59]. Vecchio, M., López-Valcarce, R., Marcelloni, F., 2012, A two-objective evolutionary approach based on topological constraints for node localization in wireless sensor networks, *Applied Soft Computing*, 12 (7), 1891–1901.
- [60]. Stoleru, R., Tian, H., Mathiharan, S. S., George, S. M., Stankovic, J. A., 2012, Asymmetric Event-Driven Node Localization in Wireless Sensor Networks, *IEEE Transactions on Parallel and Distributed Systems*, 23 (4), 634 – 642.
- [61]. Wu, G., Wang, S., Wang, B., Dong, Y., Yan, S., 2012, A novel range-free localization based on regulated neighborhood distance for wireless ad hoc and sensor networks, *Computer Networks*, 56 (16), 3581–3593.
- [62]. Chen, H., Wang, G., Wang, Z., So, H. C., Poor, H. V., 2012, Non-Line-of-Sight Node Localization Based on Semi-Definite Programming in Wireless Sensor Networks, *IEEE Transactions on Wireless Communications*, 11 (1), 108 – 116.
- [63]. Cheng, W., Zhang, N., Cheng, X., Song, M., Chen, D., 2013, Time-bounded essential localization for wireless sensor networks, *IEEE/ACM Transactions on Networking*, 21 (2), 400-412.
- [64]. Chamam, A., Pierre, S., 2010, A distributed energy-efficient clustering protocol for wireless sensor networks, *Computers & Electrical Engineering*, 36 (2), 303-312.

- [65]. Dimokas, N., Katsaros, D., Manolopoulos, Y., 2010, Energy-efficient distributed clustering in wireless sensor networks, *Journal of Parallel and Distributed Computing*, 70 (4), 371-383.
- [66]. Bajaber, F., Awan, I., 2011, Adaptive decentralized re-clustering protocol for wireless sensor networks, *Journal of Computer and System Sciences*, 77 (2), 282-292.
- [67]. Forero, P. A., Cano, A., Giannakis, G. B., 2011, Distributed Clustering Using Wireless Sensor Networks, *IEEE Journal of Selected Topics in Signal Processing*, 5(4), 707-724.
- [68]. Schaffer, P., Farkas, K., Horváth, A., Holczer, T., Buttyán, L., 2012, Secure and reliable clustering in wireless sensor networks: A critical survey, *Computer Networks*, 56 (11), 2726-2741.
- [69]. Wei, D., Jin, Y., Vural, S., Moessner, K., Tafazolli, R., 2011, An Energy-Efficient Clustering Solution for Wireless Sensor Networks, *IEEE Transactions on Wireless Communications*, 10 (11), 3973-3983.
- [70]. Liu, Z., Zheng, Q., Xue, L., Guan, X., 2012, A distributed energy-efficient clustering algorithm with improved coverage in wireless sensor networks, *Future Generation Computer Systems*, 28 (5), 780-790.
- [71]. Bachir, A., Dohler, M., Watteyne, T., Leung, K. K., 2010, MAC Essentials for Wireless Sensor Networks, *IEEE Communications Surveys & Tutorials*, 12 (2), 222-248.
- [72]. Vuran, M. C., Akyildiz, I. F., 2009, Error Control in Wireless Sensor Networks: A Cross Layer Analysis, *IEEE/ACM Transactions on Networking*, 17 (4), 1186-1199.
- [73]. Howard, S. L., Schlegel, C., Iniewski, K., 2006, Error control coding in low-power wireless sensor networks: when is ECC energy-efficient?, *EURASIP Journal on Wireless Communications and Networking*, 2006 (2), 1-14.
- [74]. Mukhopadhyay, S., Schurgers, C., Panigrahi, D., Dey, S., 2009, Model-Based Techniques for Data Reliability in Wireless Sensor Networks, *IEEE Transactions on Mobile Computing*, 8 (4), 528-543.
- [75]. Stann, F. , Heidemann, J., 2003, RMST: reliable data transport in sensor networks, *The 1st IEEE International Workshop on Sensor Network Protocols and Applications*, 11 May 2003, Anchorage, AK, USA, 102-112.
- [76]. Iyer, Y.G., Gandham, S., Venkatesan, S., 2005, STCP: a generic transport layer protocol for wireless sensor networks, *The 14th International Conference on Computer Communications and Networks*, 17-19 October 2005, San Diego, CA, USA, 449-454.

- [77]. Wan, C.-Y., Campbell, A. T., Krishnamurthy, L., 2005, Pump-Slowly, Fetch-Quickly (PSFQ): A Reliable Transport Protocol for Sensor Networks, *IEEE Journal on Selected Areas in Communicaitons*, 23 (4), 862-872.
- [78]. Park, S.-J., Vedantham, R., Sivakumar, R., Akyildiz, I. F., 2008, GARUDA: Achieving Effective Reliability for Downstream Communication in Wireless Sensor Networks, *IEEE Transactions on Mobile Computing*, 7 (2), 214-230.
- [79]. Wang, C., Sohraby, K., Bo L., Daneshmand, M., Yueming, H., 2006, A Survey of Transport Protocols for Wireless Sensor Networks, *IEEE Network*, 20(3), 34-40.
- [80]. Marcelloni, F., Vecchio, M., 2008, A Simple Algorithm for Data Compression in Wireless Sensor Networks, *IEEE Communications Letters*, 12 (6), 411-413.
- [81]. Marcelloni, F., Vecchio, M., 2009, An Efficient Lossless Compression Algorithm for Tiny Nodes of Monitoring Wireless Sensor Networks, *The Computer Journal*, 52 (8), 969-987.
- [82]. Razzaque, M. A., Bleakley, C., Dobson, S., 2013, Compression in wireless sensor networks: A survey and comparative evaluation, *ACM Transactions on Sensor Networks*, 10 (1), 1-44.
- [83]. Sadler, C. M., Martonosi, M., 2006, Data Compression Algorithms for Energy-Constrained Devices in Delay Tolerant Networks, *The 4th ACM International conference on Embedded networked sensor systems*, October 31 - November 3 2006, Boulder, Colorado, USA, 265-278.
- [84]. Marcelloni, F., Vecchio, M., 2010, Enabling energy-efficient and lossy-aware data compression in wireless sensor networks by multi-objective evolutionary optimization, *Information Sciences*, 180 (10), 1924-1941.
- [85]. Chen, F., Chandrakasan, A. P., Stojanovic, V.M., 2012, Design and Analysis of a Hardware-Efficient Compressed Sensing Architecture for Data Compression in Wireless Sensors, *IEEE Journal of Solid-State Circuits*, 47 (3), 744-756.
- [86]. Srisooksai, T., Keamarungsi, K., Lamsrichan, P., Araki, K., 2012, Practical data compression in wireless sensor networks: A survey, *Journal of Network and Computer Applications*, 35 (1), 37-59.
- [87]. IEEE standart for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirement part 15.4: wireless MAC and PHY layer specifications for low-rate wireless personal area networks. IEEE Std. 802.15.4a2007, 2007, 1-203.
- [88]. Bluetooth, www.bluetooth.com, [Ziyaret tarihi: 02 Temmuz 2014].
- [89]. IEEE standart for information technology-telecommunications and information exchange between systems-local and metropolitan area networks-specific requirements part 11: WLAN MAC and PHY layer specifications, 2012, 1-2793.
- [90]. ZigBee, <https://www.zigbee.org/>, [Ziyaret tarihi: 02 Temmuz 2014].

- [91]. WirelessHART, <https://en.hartcomm.org/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [92]. WINA, <https://www.wina.org/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [93]. ISA SP100.11a Working Group, www.isa.org/, [Ziyaret tarihi: 02 Temmuz 2014].
- [94]. 6LoWPAN, <https://tools.ietf.org/wg/6lowpan/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [95]. Faludi, R., 2011, *Building wireless sensor networks*, O'Reilly Media, CA, USA, ISBN: 978-0-596-80773-3.
- [96]. Farahani, S., 2008, *ZigBee Wireless Networks and Transceivers*, Newnes, Elsevier, Oxford, UK, ISBN: 978-0-7506-8393-7.
- [97]. Baronti, P., Pillai, P., Chook, V. W. C., Chessa, S., Gotta, A., Hu, Y. F., 2007, Wireless sensor networks: a survey on the state of the art and the IEEE 802.15.4 and ZigBee standarts, *Computer Communications, The International Journal for the Computer and Telecommunications Industry*, 30 (7), 1655-1695.
- [98]. Gislason, D., 2004, *ZigBee wireless networking*, Newnes, Elsevier, Oxford, UK, ISBN: 978-0-7506-8597-9.
- [99]. Wang, C., Jiang, T., Zhang, Q., 2014, *ZigBee network protocols and applications*, CRC Press, Taylor & Francis Group, FL, USA, ISBN 978-1-4398-1601-1.
- [100]. Chen, D., Nixon, M., Mok, A., 2010, *WirelessHART: real-time mesh network for industrial automation*, Springer, NY, USA, ISBN 978-1-4419-6046-7.
- [101]. Güngör, V. Ç., Hancke, G. P., 2013, *Industrial wireless sensor networks: applications, protocols, and standards*, CRC Press, Taylor & Francis Group, FL, USA, ISBN 978-1-4665-0051-8.
- [102]. Kushalnagar, N., Montenegro, G., Schumacher, C., 2007, IPv6 over low-power wireless personal area networks (6LoWPANs): overview, assumptions, problem statement, and goals, *RFC: 4919, IETF Network Working Group*, <https://tools.ietf.org/html/rfc4919>, [Ziyaret tarihi: 02 Temmuz 2014].
- [103]. Montenegro, G., Kushalnagar, N., Hui, J., Culler, D., 2007, Transmission of IPv6 packets over IEEE 802.15.4 networks, *RFC: 4944, IETF Network Working Group*, <https://tools.ietf.org/html/rfc4944>, [Ziyaret tarihi: 02 Temmuz 2014].
- [104]. Ee, G. K., Ng, C. K., Noordin, N. K., Mohd. Ali, B., 2010, A review of 6LoWPAN routing protocols, *In Proceedings of the Asia-Pacific Advanced Network (APAN)*, 30, 71-81.
- [105]. IETF, <https://www.ietf.org/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [106]. Mulligan G., 6LoWPAN Working Group, 2007, The 6LoWPAN architecture, *The 4th Workshop on Embedded Networked Sensors*, 25-26 June 2007, Cork, Ireland, 78-82.

- [107]. Hui, J. W., 2008, An extended Internet architecture for low-power wireless networks: design and implementation, *UC Berkeley Technical Report, UCB/EECS-2008-116*.
- [108]. Shelby, Z., Bormann, C., 2009, *6LoWPAN: The wireless embedded Internet (Wiley Series in Communications Networking & Distributed Systems)*, John Wiley & Sons, West Sussex, UK, ISBN 978-0-470-74799-5.
- [109]. TinyOS, <http://www.tinyos.net/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [110]. Levis, P., Lee, N., Welsh, M., Culler, D., 2003, TOSSIM: accurate and scalable simulation of entire TinyOS applications, *The ACM Conference on Embedded Networked Sensor Systems*, 5-7 November 2003, Los Angeles, USA, 126-137.
- [111]. LiteOS, <http://lanterns.eecs.utk.edu/software/liteos/liteos.html>, [Ziyaret tarihi: 02 Temmuz 2014].
- [112]. Cao, Q., Abdelzaher, T., Stankovic, J., He, T., 2008, The LiteOS operating system: towards unix-like abstractions for wireless sensor networks, *The 7th International Conference on Information Processing in Sensor Networks*, 22-24 April 2008, St. Louis, Missouri, USA, 233-244.
- [113]. Contiki, <http://www.contiki-os.org/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [114]. Dunkels, A., Grönvall, B., Voigt, T., 2004, Contiki- a lightweight and flexible operating system for tiny networked sensors, *The 1st IEEE Workshop on Embedded Networked Sensors*, 16-18 November 2004, Florida, USA.
- [115]. Sun SPOT, <http://www.sunspotworld.com/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [116]. Roundy, S., Wright, P. K., Rabaey, J. M., 2003, *Energy scavenging for wireless sensor networks: with special focus on vibrations*, Kluwer Academic Publishers, Springer, Massachusetts, USA, ISBN: 978-1-4615-0485-6.
- [117]. Intanagonwiwat, C., Govindan, R., Estrin, D., 2000, Directed diffusion: a scalable and robust communication paradigm for sensor networks, *6th Annual International Conference on Mobile Computing and Networking*, 6-11 August 2000, Boston, MA, USA, 56-67.
- [118]. Pottie, G. J., Kaiser, W. J., 2000, Wireless integrated network sensors, *Communications of the ACM*, 43 (5), 51-58.
- [119]. Kahn, J. M., Katz, R. H., Pister, K. S. J., 1999, Next century challenges: mobile networking for "Smart Dust", *The 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking*, 15-19 August 1999, Seattle, WA, USA, 271-278.
- [120]. Warneke, B., Last, M., Leibowitz, B., Pister, K. S. J., 2001, Smart Dust: Communicating with a Cubic-Millimeter Computer, *Computer*, 34 (1), pp. 44-51, <http://robotics.eecs.berkeley.edu/~pister/>, [Ziyaret tarihi: 02 Temmuz 2014].

- [121]. Mica, <http://www.memsic.com.php5-12.dfw1-1.websitetestlink.com/products/wireless-sensor-networks/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [122]. Rabaey, J. M., Ammer, M. J., da Silva Jr., J. L., Patel, D. Roundy, S., 2000, Pico radio supports ad hoc ultra-low power wireless networking, *IEEE Computer Magazine*, 33 (7), 42-48.
- [123]. Zuniga, M., Krishnamachari, B., 2004, Analyzing the transitional region in low power wireless links, *The 1st Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, 4-7 October 2004, Santa Clara, CA, USA, 517-526.
- [124]. Perrig, A., Szewczyk, R., Tygar, J. D., Wen, V., Culler, D. E., 2002, Spins: security protocols for sensor networks, *Wireless Networks, The Journal of Mobile Communication, Computation and Information*, 8 (5), 521-534.
- [125]. Imote2, http://wsn.cse.wustl.edu/images/c/cb/Imote2-ds-rev2_2.pdf, [Ziyaret tarihi: 02 Temmuz 2014].
- [126]. Koushanfar, F., Potkonjak, M., Vincentelli, A. S., 2002, Fault tolerance techniques for wireless ad hoc sensor networks, *The 1st IEEE International Conference on Sensors*, 12-14 June 2002, Orlando, FL, USA, 2, 1491-1496.
- [127]. Hoblos, G., Staroswiecki, M., Aitouche, A., 2000, Optimal design of fault tolerant sensor networks, *IEEE International Conference on Control Applications*, 27 September 2000, Anchorage, AK, USA, 467-472.
- [128]. Han, X., Cao, X., Lloyd, E. L., Shen, C-C., 2010, Fault-Tolerant Relay Node Placement in Heterogeneous Wireless Sensor Networks, *IEEE Transactions on Mobile Computing*, 9(5), 643-656.
- [129]. Cho, S., Chandrakasan, A. P., 2001, Energy efficient protocols for low duty cycle wireless microsensor networks, *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 7-11 May 2001, Salt Lake City, UT, USA, 4, 2041-2044.
- [130]. Younis, M., Senturk, I. F., Akkaya, K., Lee, S., Senel, F., 2014, Topology management techniques for tolerating node failures in wireless sensor networks: a survey, *Computer Networks*, 58, 254-283.
- [131]. Wang, Y., 2008, *Topology Control for Wireless Sensor Networks*, Wireless Sensor Networks and Applications, Series: Signals and Communication Technology, In: Li, Y. (ed.), Thai, M. T. (ed.), Wu, W. (ed.), Chapter 5, Springer, NY, USA, ISBN: 978-0-387-49592-7, 113-147.
- [132]. Karl, H., Willig, A., 2007, *Protocols and Architectures for Wireless Sensor Networks*, John Wiley & Sons, West Sussex, UK, ISBN: 978-0-470-09510-2.

- [133]. Meguerdichian, S., Koushanfar, F., Potkonjak, M., Srivastava, M. B., 2001, Coverage problems in wireless ad-hoc sensor networks, *The 20th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM)*, 22-26 April 2001, Anchorage, AK, USA, 3, 1380-1387.
- [134]. Schurgers, C., Tsiatsis, V., Ganeriwal, S., Srivastava, M., 2002, Topology management for sensor networks: exploiting latency and density, *The 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, 9-11 June 2002, Lausanne, Switzerland, 135-145.
- [135]. Shahid, A., Qureshi, H. K., 2013, A survey on topology maintenance techniques to extend the lifetime of wireless sensor networks, *Wireless Sensor Networks for Developing Countries, The 1st International Conference on WSN4C, Communications in Computer and Information Science*, Springer, 366, 96-107.
- [136]. Santi, P., 2005, Topology control in wireless ad hoc and sensor networks, *ACM Computing Surveys (CSUR)*, 37 (2), 164-194.
- [137]. ITU, <https://www.itu.int/> [Ziyaret tarihi: 02 Temmuz 2014].
- [138]. Porret, A.-S., Melly, T., Enz, C. C., Vittoz, E. A., 2000, A low-power low-voltage transceiver architecture suitable for wireless distributed sensors network, *IEEE International Symposium on Circuit and Systems*, 28-31 May 2000, Geneva, Switzerland, 1, 56-59.
- [139]. Shih, E., Cho, S. H., Ickes, N., Min, R., Sinha, A., Wang, A., Chandrakasan, A., 2001, Physical layer driven protocol and algorithm design for energy-efficient wireless sensor networks, *The 7th Annual International Conference on Mobile Computing and Networking*, 16-21 July 2001, Rome, Italy, 272-287. <http://www-mtl.mit.edu/researchgroups/icsystems/uamps/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [140]. Shimmer, <http://www.shimmersensing.com/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [141]. XYZ, <http://www.eng.yale.edu/enalab/XYZ/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [142]. Underwater Acoustic Sensor Networks (UW-ASN), <http://www.ece.gatech.edu/research/labs/bwn/UWASN/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [143]. Cui, J. H., Kong, J., Gerla, M., Zhou, S., 2005, *Challenges: building scalable and distributed underwater wireless sensor networks (UWSNs) for aquatic applications*, UCONN CSE Technical Report: UbiNet-TR05-02.
- [144]. Calhoun, B. H., Daly, D. C., Verma, N., Finchelstein, D. F., Wentzloff, D. D., Wang, A., Cho, S. H., Chandrakasan, A. P., 2005, Design considerations for ultra-low energy wireless microsensor nodes, *IEEE Transactions on Computers*, 56 (6), 727-740.
- [145]. Walden, R. H., 1999, Analog-to-Digital converter survey and analysis, *IEEE Journal of Selected Areas in Communication*, 17 (4), 539-550.

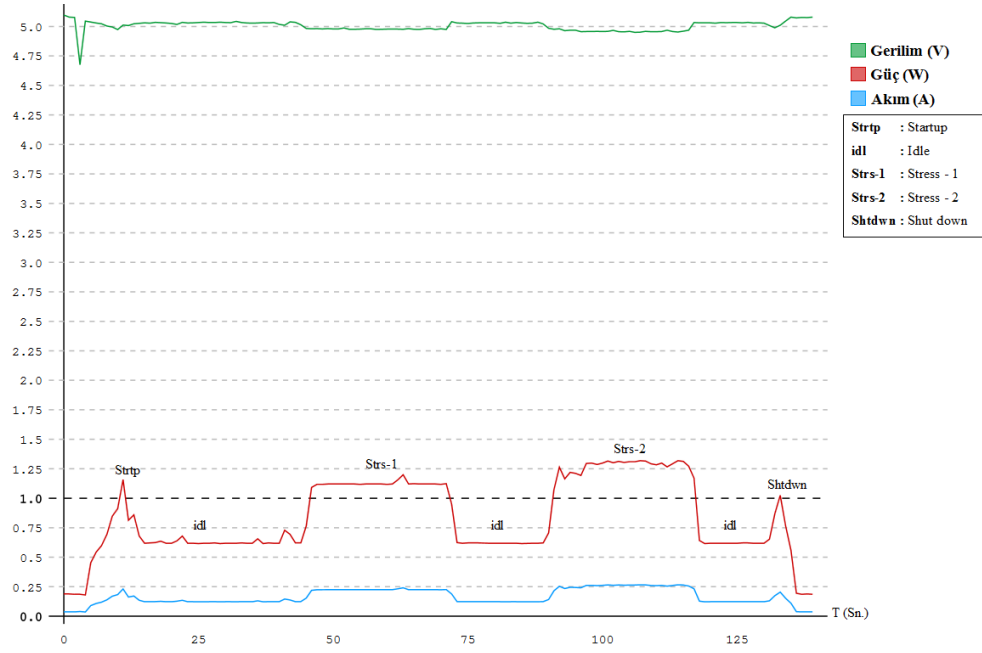
- [146]. Wang, A., Chandrakasan, A., 2002, Energy-efficient DSPs for wireless sensor networks, *IEEE Signal Processing magazine*, 19 (4), 68-78.
- [147]. Sinha, A., Chandrakasan, A., 2001, Dynamic power management in wireless sensor networks, *IEEE Design & Test of Computers*, 18 (2), 62-74.
- [148]. Min, R., Furrer, T., Chandrakasan, A., 2000, Dynamic voltage scaling techniques for distributed microsensor networks, *IEEE Computer Society Workshop on VLSI*, 27-28 April 2000, Orlando, FL, USA, pp. 43-46.
- [149]. Pering, T., Burd, T., Brodersen, R., 1998, The simulation and evaluation of dynamic voltage scaling algorithms, *International Symposium on Low Power Electronics and Design*, 10-12 August 1998, Monterey, CA, USA, 76-81.
- [150]. Burd, T., Pering, T., Stratakos, A., Brodersen, R., 2000, A dynamic voltage scaled microprocessor system, *IEEE Journal of Solid-State Circuits*, 35 (11), 1571-1580.
- [151]. Pillai, P., Shin, K. G., 2001, Real-time dynamic voltage scaling for low-power embedded operating systems, *The 18th ACM symposium on Operating systems principles*, 21-24 October 2001, Chateau Lake Louise, Banff, Canada, 89-102.
- [152]. Qadi, A., Goddard, S., Farritor, S., 2003, A dynamic voltage scaling algorithm for sporadic tasks, *The 24th IEEE Real-Time Systems Symposium*, 3-5 December 2003, Cancun, Mexico, 52-62.
- [153]. Yu, Y., Prasanna, V. K., 2005, Energy-balanced task allocation for collaborative processing in wireless sensor networks, *Mobile Networks and Applications*, 10 (1-2), 115-131.
- [154]. Zhuo, J., Chakrabarti, C., 2008, Energy-efficient dynamic task scheduling algorithms for DVS systems, *ACM Transactions on Embedded Computing Systems*, 7 (2), 1-22.
- [155]. Wang, A. Y., Sodini, C. G., 2004, A simple energy model for wireless microsensor transceivers, *The IEEE Global Telecommunications Conference (Globecom)*, 29 November-3 December 2004, Dallas, TX, USA, 5, 3205-3209.
- [156]. Cui, S., Goldsmith, A. J., Bahai, A., 2005, Energy-constrained modulation optimization, *IEEE Transactions on Wireless Communications*, 4 (5), 2349-2360.
- [157]. Tmote Sky, <http://www.eecs.harvard.edu/~konrad/projects/shimmer/references/tmote-sky-datasheet.pdf>, [Ziyaret tarihi: 02 Temmuz 2014].
- [158]. EYES, <http://www2.tkn.tu-berlin.de/research/eyes/>, <http://doc.utwente.nl/46715/>, [Ziyaret tarihi: 02 Temmuz 2014].

- [159]. Stargate, http://www.cse.nd.edu/~cpoellab/teaching/cse40815/Stargate_Manual_7430-0317-13_B.pdf, [Ziyaret tarihi: 02 Temmuz 2014].
- [160]. Margi, C. B., Petkov, V., Obraczka, K., Manduchi, R., 2006, Characterizing energy consumption in a visual sensor network testbed, *The 2nd International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities*, 1-3 March 2006, Barcelona, Spain, 1-8.
- [161]. Stargate NetBridge, http://www.openautomation.net/uploadsproductos/stargate_netbridge_users_manual.pdf, [Ziyaret tarihi: 02 Temmuz 2014].
- [162]. AS-SG1000, <http://www.advanticsys.com/shop/assg1000-p-28.html>, [Ziyaret tarihi: 02 Temmuz 2014].
- [163]. Arduino Uno, <http://arduino.cc/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [164]. KET cihazları yönetmeliği, <http://tk.gov.tr/mevzuat/yonetmelikler/dosyalar/KETCihazlariHkYonetmelik.pdf>, [Ziyaret tarihi: 02 Temmuz 2014].
- [165]. XBee 868LP, <http://www.digi.com/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [166]. XBEE 868LP uygulama geliştirme kiti, <http://www.digi.com/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [167]. Raspberry Pi, <http://www.raspberrypi.org/>, [Ziyaret tarihi: 02 Temmuz 2014].
- [168]. Halfacree, G., Upton, E., 2012, *Raspberry Pi user guide*, John Wiley & Sons, West Sussex, UK, ISBN: 978-1-118-46446-5.
- [169]. Monk, S., 2012, *Programming the Raspberry Pi: getting started with Python*, Mc Graw Hill, London, UK, ISBN: 978-0-07-180783-8.
- [170]. Richardson, M., Wallace, S., 2012, *Getting started with Raspberry Pi (Make: Projects)*, Maker Media, USA, ISBN: 978-1-449-34421-4.
- [171]. Elson, J., Estrin, D., 2001, Time synchronization for wireless sensor networks, *The 15th International Parallel and Distributed Processing Symposium, Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing*, 23-27 April 2001, San Francisco, CA, USA, 1-6.
- [172]. Ganeriwal, S., Kumar, R., Srivastava, M. B., 2003, Timing-sync protocol for sensor networks, *The 1st International Conference on Embedded Networked Sensor Systems*, 5-7 November 2003, Los Angeles, CA, USA, 138-149.
- [173]. Sundararaman, B., Buy, U., Kshemkalyani, A. D., 2005, Clock synchronization for wireless sensor networks: a survey, *Ad Hoc Networks, Elsevier*, 3, 281-323.
- [174]. Kopetz, H., 2011, *Real-Time Systems: Design Principles for Distributed Embedded Applications*, Springer, NY, USA, ISBN: 978-1-4419-8237-7.
- [175]. Mills, D. L., 1991, Internet time synchronization: The network time protocol, *IEEE Transactions on Communications*, 39 (10), 1482-1493.

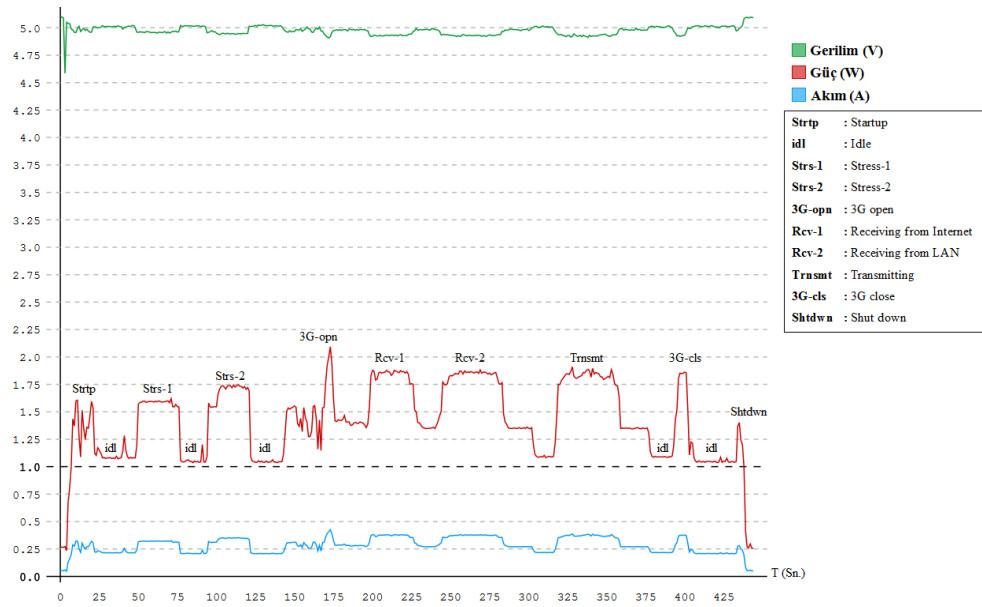
- [176]. Faizulkhakov, Y. R., 2007, Time synchronization methods for wireless sensor networks: a survey, *Programming and Computing Software*, Springer, 33 (4), 214-226.
- [177]. Song, W.-Z., Huang, R., Shirazi, B., LaHusen, R., 2009, TreeMAC: Localized TDMA MAC protocol for real-time high-data-rate sensor networks, *Pervasive and Mobile Computing*, 5 (6), 750-765.
- [178]. Arisha, K., Youssef, M., Younis, M., 2002, *Energy-Aware TDMA-Based MAC for sensor networks*, System-Level Power Optimization for Wireless Multimedia Communication, Power Aware Computing, In: Karri, R. (ed.), Goodman, D. (ed.), Chapter 2, Springer, NY, USA, ISBN: 978-1-4020-7204-8, 21-40.
- [179]. Elson, J., Girod, L., Estrin, D., 2002, Fine-grained network time synchronization using reference broadcasts, *The 5th Symposium on Operating Systems Design and Implementation*, 9-11 December 2002, Boston, MA, USA, 36 (SI), 147-163.
- [180]. Sichitiu, M. L., Veerarittiphan, C., 2003, Simple, accurate time synchronizaiton for wireless sensor networks, *IEEE Wireless Communications and Networking*, 16-20 March, New Orleans, LA, USA, 2, 1266-1273.
- [181]. Greunen, J. V., Rabaey, J., 2003, Lightweight time synchronization for sensor networks, *The 2nd ACM International Conference on Wireless Sensor Networks and Applications*, 19 September 2003, San Diego, CA, USA, 11-19.
- [182]. Hu, A-S., Servetto, S. D., 2003, Asymptotically optimal time synchronization in dense sensor networks, *The 2nd ACM International Conference on Wireless Sensor Networks and Applications*, 19 September 2003, San Diego, USA, 1-10.
- [183]. Maroti, M., Kusy, B., Simon, G., Ledeczi, A., 2004, The flooding time synchronization protocol, *The 2nd ACM Conference on Embedded Networked Sensor Systems*, 3-5 November 2004, Baltimore, Maryland, USA, 39-49.
- [184]. PalChaudhuri, S., Saha, A. K., Johnson, D., 2004, Adaptive clock synchronization in sensor networks, *The 3rd International Symposium on Information Processing in Sensor Networks*, 26-27 April 2004, Berkeley, CA, USA, 340-348.
- [185]. Dai, H., Han, R., 2004, TSynC: a lightweight bidirectional time synchronization service for wireless sensor networks, *ACM SIGMOBILE Mobile Computing and Communications Review, Special Issue on Wireless PAN & Sensor Networks*, 8 (1), 125-139.
- [186]. Su, W., Akyildiz, I. F., 2005, Time-diffusion synchronization protocol for wireless sensor networks, *IEEE/ACM Transactions on Networking*, 13 (2), 384-397.
- [187]. Li, Q., Rus, D., 2006, Global clock synchronization in sensor networks, *IEEE Transactions on Computers*, 55 (2), 214-226.

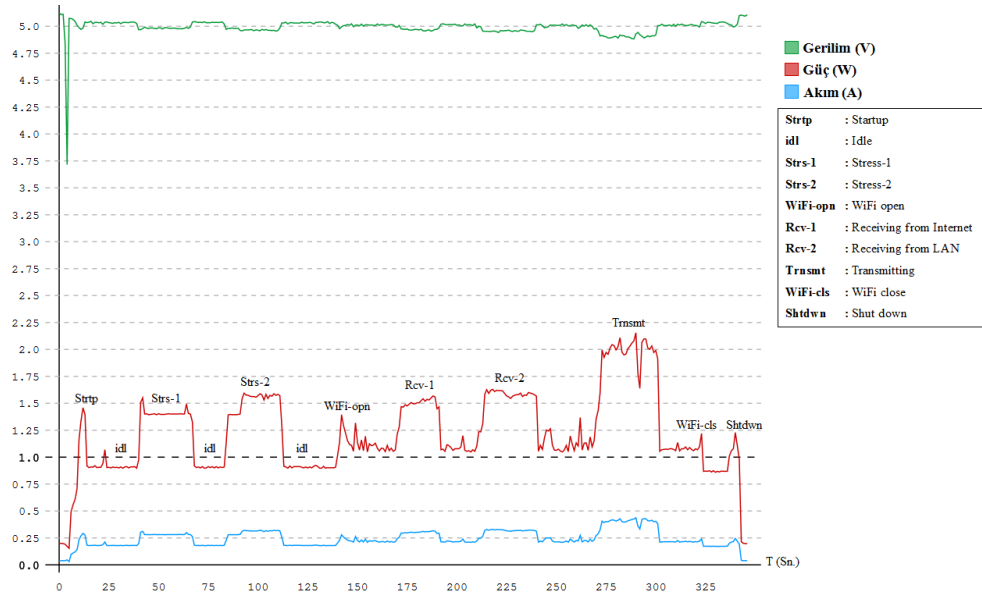
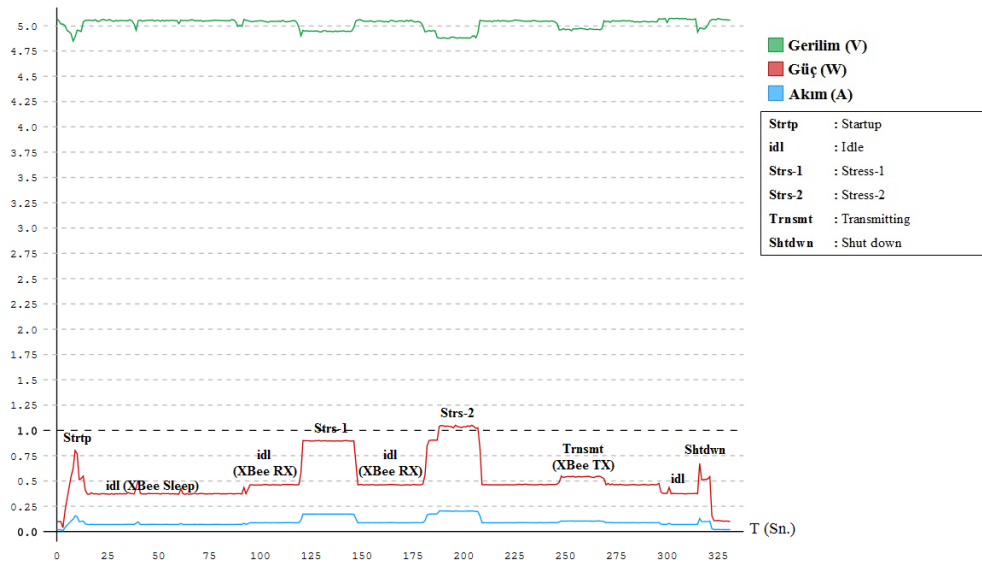
EKLER

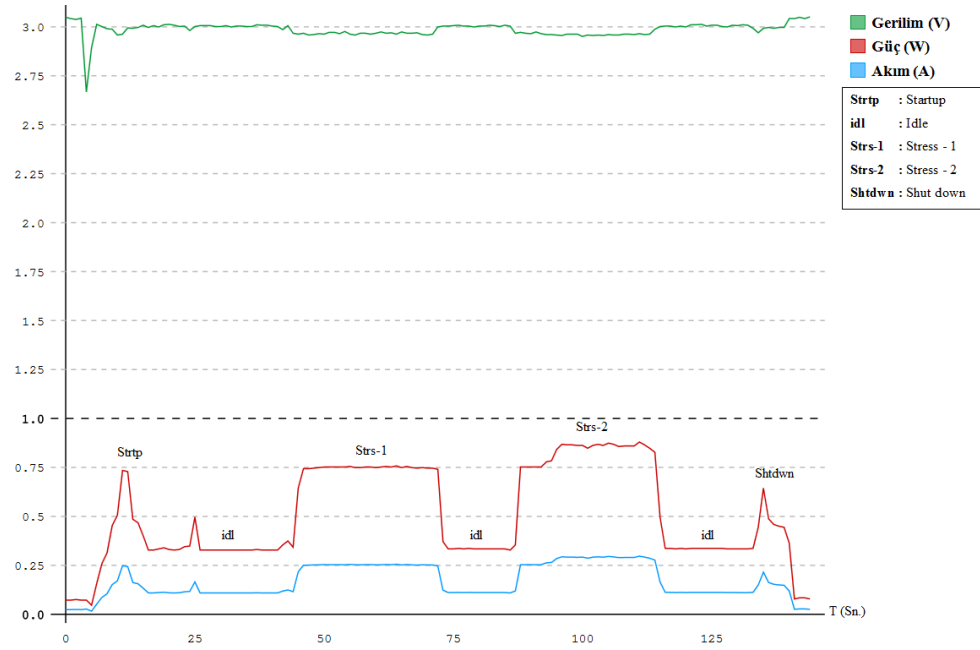
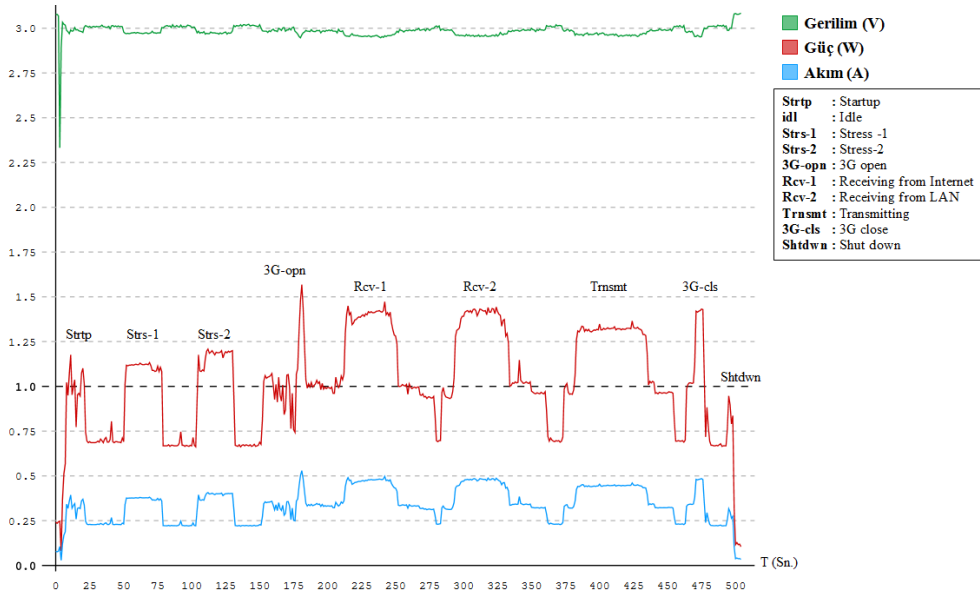
EK 1. Koordinatör düğümün güç tüketim grafiği (Orjinal).



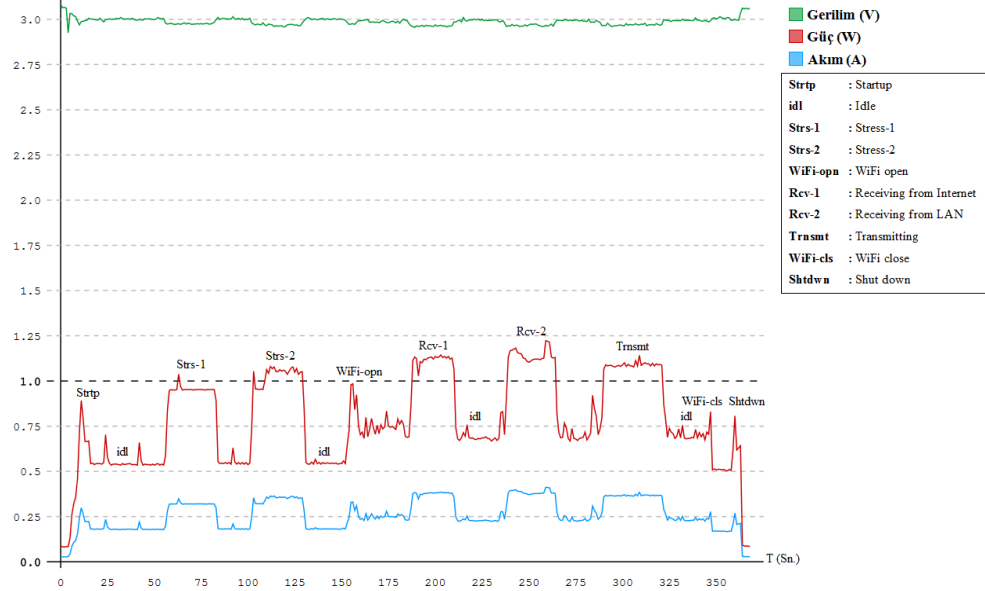
EK 2. İnternete bağlı koordinatör düğümün güç tüketim grafiği (Orjinal).



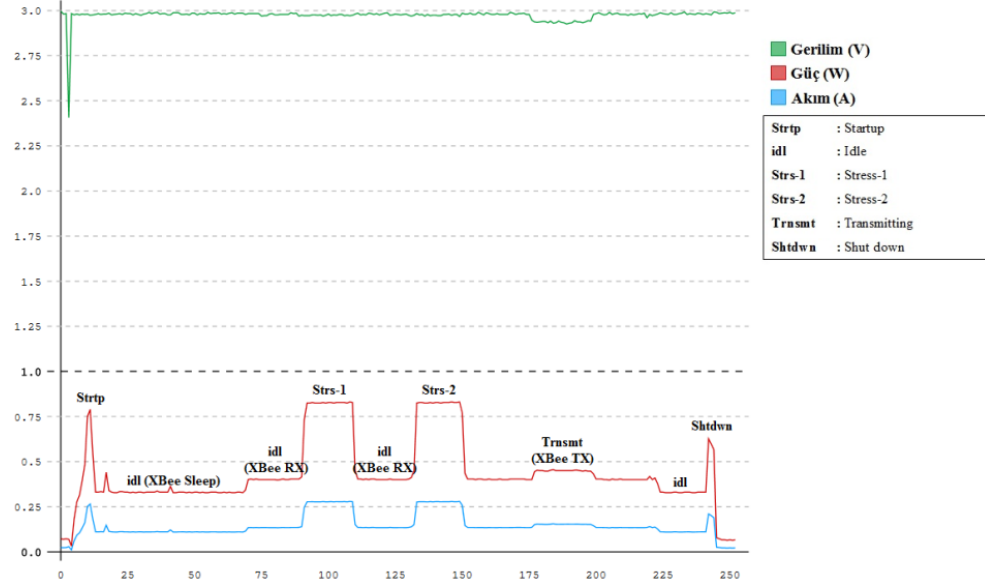
EK 3. WLAN’da çalışan koordinatör düğümün güç tüketim grafiği (Orjinal).**EK 4.** KAA içerisinde çalışan koordinatör düğümün güç tüketimi (Orjinal).

EK 5. DGY ve optimizasyonu yapılmış koordinatörün güç tüketim grafiği.**EK 6.** DGY ve optimizasyonu yapılmış, İnternete bağlı koordinatörün tüketim grafiği.

EK 7. DGY ve optimizasyonu yapılmış, WLAN'daki koordinatörün güç tüketimi.



EK 8. DGY ve optimizasyonu yapılmış, KAA'daki koordinatörün güç tüketim grafiği.



ÖZGEÇMİŞ

Fotoğraf

Kişisel Bilgiler

Adı Soyadı	Mehmet Erkan YÜKSEL
Uyruğu	T.C.
Doğum tarihi, Yeri	1980, Zile
Telefon	
E-mail	eyuksel@istanbul.edu.tr, m.erkanyuksel@gmail.com
Web adres	

Eğitim

Derece	Kurum/Anabilim Dalı/Programı	Yılı
Doktora	İ.Ü. Fen Bilimleri Enstitüsü/ Bilgisayar Mühendisliği Anabilim Dalı/ Bilgisayar Mühendisliği Programı	2014
Yüksek Lisans	İ.Ü. Fen Bilimleri Enstitüsü/ Bilgisayar Mühendisliği Anabilim Dalı/ Bilgisayar Mühendisliği Programı	2010
Lisans	Fırat Üniversitesi/ Mühendislik Fakültesi/ Bilgisayar Mühendisliği Bölümü	2005
Lise	Zile Dinçerler Anadolu Lisesi	1998

Makaleler / Bildiriler

- [1]. Yüksel, M. E., Yüksel A. S., Zaim, A. H., 2013, A reputation-based privacy management system for social networking sites, *Turk. J. Elec. Eng. & Comp. Sci.*, 21 (3), 766-784.
- [2]. Yüksel, M. E., Yüksel A. S., 2011, RFID Technology in Business Systems and Supply Chain Management, *Journal of Economic and Social Studies*, 1 (1), 53-71, 2011, ISSN:1986-8502. (Selected Paper Award)
- [3]. Yüksel M. E., Yüksel A. S., 2011, A System Design for The Integration of RFID Systems with Wireless Network Technologies, *The 7th International Conference on Wireless and Mobile Communications (ICWMC)*, June 2011, Luxembourg.
- [4]. Sevgen, S., Yüksel, M. E., 2011, Image Processing Applications on ACE16k Using Iterative Annealing Method, *The 6th International Advanced Technologies Symposium (IATS)*, May 2011, Elazığ, Turkey.
- [5]. Ensari, T., Yüksel M. E., 2011, Convergence Analysis of Hopfield Neural Networks, *The 6th International Advanced Technologies Symposium (IATS)*, May 2011, Elazığ, Turkey.
- [6]. Yüksel M. E., Yüksel, A. S., Memiş, Z. T., Zaim, A. H., 2011, The Interaction of RFID with Wireless Communication Technologies, *The 1st International Conference on Networking and Future İnternet (ICNFI)*, April 2011, Paris, France.
- [7]. Yüksel A. S., Yüksel, M. E., Zaim, A. H., 2010, An Approach for Protecting Privacy on Social Networks, *The 5th International Conference on Systems and Networks Communications (ICSNC)*, August 2010, Nice, France. (Best Paper Award)
- [8]. Yüksel, M. E., Yüksel, A. S., 2010, An Application for Protecting Personal Information on Social Networking Websites, *The 4th International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM)*, October 2010, Florence, Italy.
- [9]. Yüksel, M. E., Zaim, A. H., Yüksel, A. S., 2010, Impacts of RFID in Business Systems and Supply Chain Management, *The 2nd International Symposium on Sustainable Development*, June 2010, Sarajevo, Bosnia and Herzegovina. (Selected Paper)
- [10]. Yüksel M. E., Zaim, A. H., 2009, RFID As A New Technology, RFID System Structures and An RFID System Design Approach, *The 5th International Advanced Technologies Symposium (IATS'09)*, Karabük, Turkey.
- [11]. Yüksel M. E., Zaim, A. H., 2009, The Advantages of RFID on Object Identification and Tracking, Data Management and Analysis Systems, *The 5th International Advanced Technologies Symposium (IATS'09)*, Karabük, Turkey.