T.C.
TOKAT GAZİOSMANPAŞA ÜNİVERSİTESİ
LİSANSÜSTÜ EĞİTİM ENSTİTÜSÜ
MEKATRONİK MÜHENDİSLİĞİ ANABİLİM DALI

ÜRETİM TEZGAHLARINDAN TOPLANAN VERİLERİN STANDARDİZASYONU

YÜKSEK LİSANS TEZİ

Abdullah Arda GÜNAYDIN

Birinci Danışman: Doç. Dr. Mustafa EKER

TOKAT- 2024

ETİK SÖZLEŞME

Tokat Gaziosmanpaşa Üniversitesi Lisansüstü Eğitim Enstitüsü tez yazım kılavuzuna göre, Doç. Dr. Mustafa EKER danışmanlığında hazırlamış olduğum " Üretim Tezgahlarından Toplanan Verilerin Standardizasyonu" adlı Yüksek Lisans tezinin bilimsel etik değerlere ve kurallara uygun, özgün bir çalışma olduğunu, aksinin tespit edilmesi halinde her türlü yasal yaptırımı kabul edeceğimi beyan ederim.

31/01/2024

Abdullah Arda GÜNAYDIN

JÜRİ KABUL VE ONAY

ABDULLAH ARDA GÜNAYDIN tarafından hazırlanan "**Üretim Tezgahlarından Toplanan Verilerin Standardizasyonu**" adlı tez çalışmasının savunma sınavı 04/01/2024 tarihinde yapılmış olup aşağıda verilen Jüri tarafından Oy Birliği / Oy Çokluğu ile Tokat Gaziosmanpaşa Üniversitesi Lisansüstü Eğitim Enstitüsü Mekatronik Mühendisliği Anabilim Dalı'nda Yüksek Lisans olarak kabul edilmiştir.

Jüri Üyeleri (Unvanı, Adı Soyadı).....	**İmzası**
Üye (Başkan) :.....	.....
Üye :	.....
Üye :	.....
Üye :	.....
Üye :	.....

ONAY

……/……/…….

Lisansüstü Eğitim Enstitüsü Müdürü

İTHAF

Söz konusu bu çalışma desteklerini her daim hissettiren aileme, eşim Fatma Nur Günaydın'a ve oğlum Alkım Meriç Günaydın'a ithaf edilmiştir.

TEŞEKKÜR

Bu tez çalışması, Endüstri 4.0'ın temelini oluşturan üretim verilerinin toplanması ve işlenmesi konusunda yapılan araştırma ve geliştirme süreçlerinin bir ürünüdür. Bu süreçte birçok kişinin değerli katkılarını almış olmanın mutluluğunu ve gururunu yaşıyorum.

İlk olarak, bu çalışmanın danışmanı olan Doç. Dr. Mustafa Eker'e teşekkürlerimi sunmak isterim. Değerli yönlendirmeleri, bilgi birikimi ve samimi destekleri sayesinde bu çalışma daha da değerli hale gelmiştir. Kendisinin rehberliğinde bu yola çıkmak, büyük bir şans ve onur olmuştur.

Çalışma arkadaşlarım Gizem Eser Erdek ve Ahmet Can Özdemir'e ve diğer ekip arkadaşlarıma teşekkürlerimi iletmek isterim. Süreç içerisinde sağlamış olduğunuz destekler, farklı bakış açılarından faydalanmak, bu süreci daha verimli kılmıştır. Katkılarınız için teşekkür ederim.

Son olarak eşim Fatma Nur Günaydın'a süreç içerisinde motivasyonumun yüksek kalmasında sağladığı destek, gösterdiği sabırdan ötürü ve her zaman yanımda olduğu, bu çalışmaya şans getirdiği için özellikle teşekkür etmek isterim. Varlığın benim için çok değerli. Ayrıca, bu çalışma sürecinde yanımda olan ve desteklerini esirgemeyen aileme, bana dair inancını her daim ayakta tutan annem Şerife Günaydın'a, her durumda yanımda olduğunu bildiğim babam Orhan Günaydın'a ve yüzümü her daim güldüren kardeşim Gizem Nur Günaydın'a minnettarlığımı sunarım.

Tüm bu destekler, bu çalışmanın ortaya çıkmasına katkı sağlamış ve bu tezi tamamlamamda büyük bir motivasyon kaynağı olmuştur. Her birinize teşekkür ederim.

ÖZET

ÜRETİM TEZGAHLARINDAN TOPLANAN VERİLERİN STANDARDİZASYONU

Günaydın, Abdullah Arda
Yüksek Lisans, Mekatronik Mühendisliği Anabilim Dalı
Tez Danışmanı: Doç. Dr. Mustafa Eker
Ocak 2024, xix + 110 sayfa

Son zamanlarda insanlığın gelişimini birçok açıdan ileriye taşımış olan en büyük gelişmelerden bir tanesi de şüphesiz olarak internetin icadı ve yaygınlaşmasıdır. Bu gelişmelerin en başında ise Dijital Dönüşüm ya da Endüstri 4.0 kavramı gelmektedir. Günümüz üretiminin gerekliliklerinin değişmesi, müşteri taleplerinin değişkenliği, uzak doğu ülkelerinin sektörler arasında rekabeti artıran ucuz ve büyük ölçekli üretim faaliyetleri göstermesi nedeni ile batılı ülkelerin verimliliği, hızı ve kaliteyi artırarak beraberinde üretime esneklik kazandırmak amacı ile geliştirmiş oldukları teknolojiler Endüstri 4.0 kavramının temelini oluşturmaktadır. Dijital Dönüşüm ya da başka bir deyişle Endüstri 4.0 kavramı sanayii devrimlerinin dördüncüsü olarak karşımıza çıkmaktadır. Bu devrimin etkileri insan hayatının birçok alanında görülmektedir. Endüstri 4.0 tek başına bir kavram olmamakla beraber tüm teknolojileri ve beraberinde getirmiş olduğu yenilikler ile, firmaların bu teknoloji ve yenilikleri kullanarak üretim ve diğer faaliyet alanlarını Endüstri 4.0 seviyelerine çıkartarak rekabet güçlerini arttırmayı hedeflemektedir. Bu durum Endüstri 4.0'ın bir hedef olduğunu rahatça göstermektedir. Firmalar, bu alanda geliştirdikleri yeni teknolojiler ile Endüstri 4.0 kavramının genişlemesine katkı sağlarlarken aynı zamanda Endüstri 4.0 seviyelerini yakalamak için hedeflerine daha da yakınlaşmaktadırlar. Endüstri 4.0 teknolojilerinin en önemlisi ise Nesnelerin İnterneti teknolojisi olmuştur. Nesnelerin İnterneti teknolojisi, günümüzde iş hayatlarımızdan günlük hayatlarımıza kadar birçok alanda karşımıza çıkmaktadır. Bu teknolojinin yaygınlaşması, akıllı cihazlarında hayatımızda büyük yer edinmesine neden olmuş ve hayatımızın her bir noktası birer veri kaynağına dönüşmüştür. Akıllı cihazlar verileri kendi içlerinde işleyip depolayabilmelerinin yanı sıra farklı cihazlar ile de iletişim haline geçebilmektedirler. Kullanılan akıllı cihazların endüstriyel boyutta evrilmesi ile farklılaşan protokoller, artan güvenlik gereksinimleri, veri hızları ve büyüklükleri gibi nedenler ile Endüstriyel Nesnelerin İnterneti teknolojisi ortaya çıkmaktadır. Endüstride yer alan birbirinden farklı ekipman çeşitliliğinin fazla olması toplanan verilerin karmaşık ve çok çeşitli olmasına neden olmuştur. Veri yayınlama ve haberleşme farklılıkları bu karmaşıklığın temelini oluşturmaktadır. Kullanılan ekipmanların sahip olduğu farklı tip kontrolcüler veya sensörler destekledikleri veri yayınlama protokolleri marka ve model bilgilerine bağlı olarak değişkenlik gösterebilmektedir. Bu çalışmada, bahsedilen farklılıklara sebep olan Open Platform Communications Unified Architecture (OPC UA), MODBUS, MTConnect gibi protokollerden toplanan farklı çeşitlilikte verilerin ekipman kullanılabilirliklerinin takip edilebilmesi adına standardize edilmesi amaçlanmaktadır.

Sonuç olarak farklı protokol desteği sunan tezgahlardan toplanan veriler anlamlandırılarak standartlaştırılmaktadır. Standartlaştırılan bu veriler, veri tabanına kaydedilmiş verileri için oluşturulan uygun sorgular ve Grafana uygulaması üzerinde oluşturulan izleme ekranları ile tezgâh verileri anlık olarak izlenebilmektedir.

Anahtar Kelimeler: Endüstri 4.0, Dijital Dönüşüm, Nesnelerin İnterneti, OPC UA, MODBUS, MTConnect, Veri.

ABSTRACT

STANDARDIZATION OF DATA COLLECTED FROM PRODUCTION MACHINES

Günaydın, Abdullah Arda
Master's Thesis, Department of Mechatronic Engineering
Advisor: Assoc. Prof. Dr. Mustafa Eker
January 2024, xix + 110 pages

In recent times, one of the greatest developments that has propelled human progress in various aspects is undoubtedly the invention and widespread adoption of the internet. At the forefront of these advancements is the concept of Digital Transformation or Industry 4.0. The changing requirements of contemporary production, the variability in customer demands, and the competitive cheap and large-scale production activities of Eastern countries have prompted Western nations to develop technologies aimed at enhancing productivity, speed, and quality, thereby introducing flexibility to production processes. These technologies form the foundation of the Industry 4.0 concept. Digital Transformation, or Industry 4.0, emerges as the fourth industrial revolution, with effects observed across various facets of human life. Industry 4.0, not merely a standalone concept, aspires to elevate companies to Industry 4.0 levels by utilizing all technologies and innovations it brings. Companies, through the development of new technologies in this field, contribute to the expansion of the Industry 4.0 concept while getting closer to achieving their goals of reaching Industry 4.0 levels. One of the most significant Industry 4.0 technologies is the Internet of Things (IoT). The IoT technology is prevalent in various aspects of our lives, from business to daily routines. The proliferation of this technology has led smart devices to play a significant role in our lives, turning every aspect of our lives into a source of data. Smart devices not only process and store data internally but also communicate with different devices. The evolution of smart devices at an industrial scale has given rise to Industrial Internet of Things (IIoT) technology due to differences such as evolving protocols, increased security requirements, and varying data speeds and sizes. The diversity in the types of equipment in the industry has led to complex and diverse data collection. Discrepancies in data publishing and communication protocols are at the core of this complexity. Different types of controllers or sensors supported by the equipment may vary based on brand and model information. This study aims to standardize data collected from different protocols such as Open Platform Communications Unified Architecture (OPC UA), MODBUS, MTConnect, which contribute to the mentioned differences, to track the usability of equipment.

As a result, the data collected from machines that offer different protocol support are made meaningful and standardized. Machine data can be monitored instantly with these standardized data, appropriate queries created for the data saved in the database, and monitoring screens created on the Grafana application.

Keywords: Industry 4.0, Digital Transformation, Internet of Things, OPC UA, MODBUS, MTConnect, Data.

İÇİNDEKİLER

Sayfa

ETİK SÖZLEŞME i
JÜRİ KABUL VE ONAY ii
TEŞEKKÜR iv
ÖZET v
ABSTRACT vii
İÇİNDEKİLER viii
TABLOLAR LİSTESİ xi
ŞEKİLLER LİSTESİ xii
KISALTMALAR xvi
1. GİRİŞ 1
2. LİTERATÜR TARAMASI 3
3. GENEL BİLGİLER 21
3.1. Endüstri 4.0 ve Teknolojileri 21
3.1.1. Endüstri devrimlerinin tarihsel gelişimleri ve Endüstri 4.0 21
3.1.2. Endüstri 4.0 beraberinde gelişen teknolojiler 23
3.2. Endüstri 4.0 Teknolojisi Olarak Nesnelerin İnterneti 23
3.2.1. IoT teknolojisi 23
3.2.2. IIoT teknolojisi 24
3.2.3. IoT ve IIoT farkları 25
3.3. Üretimde Kullanılan Tezgahlar, Çeşitleri ve Yapıları 26
3.3.1. Konvansiyonel ekipmanlar 26
3.3.2. CNC/NC ekipmanlar 28
3.3.3. Manuel ekipmanlar 33

3.4. Nesnelerin İnterneti Entegrasyonunda Kullanılan Protokol ve Yazılımlar 34
3.4.1. OPC UA protokolü 34
3.4.2. BFC Gateway yazılımı 35
3.4.3. MTConnect protokolü 37
3.4.4. Fanuc FOCAS protokolü 38
3.4.5. MODBUS protokolü 39
3.4.6. Kepware Kepserver Ex yazılımı 41
3.4.7 Microsoft SQL Server 42
3.4.8. Grafana yazılımı 44
3.5. Tezgahlar ile Nesnelerin İnterneti Teknolojisi Entegrasyonu 46
3.5.1. Endüstriyel nesnelerin interneti altyapısının kurulması ve mimari tasarımı 46
3.5.2. Ekipmanların belirlenmesi 52
3.5.3. Ağ altyapısının kurulması ve desteklenen protokollerin devreye alınması .. 59
3.5.4. Verilerin sunucu ortamına aktarılması 66
3.6. Üretim Verilerinin İncelenmesi ve Anlamlandırılması 72
3.6.1. OPC UA ile toplanan üretim verileri 74
3.6.2. MTConnect ile toplanan üretim verileri 77
3.6.3. FOCAS ile toplanan üretim verileri 79
3.6.4. BFC Gateway ile toplanan üretim verileri 82
3.6.5. MODBUS ile toplanan üretim verileri 82
3.6.6. Enerji analizörü kullanılarak MODBUS ile toplanan üretim verileri 84
4. BULGULAR 87
4.1. Veri Standardizasyon ve Anlamlandırma Uygulaması 87
4.1.1. Veri standardizasyon uygulaması 87
4.2.2. MS SQL veri tabanı kayıtları 102

4.2.3 Grafana ile verilerin görselleştirilmesi 102
5. SONUÇ VE ÖNERİLER 107
KAYNAKLAR 110
ÖZGEÇMİŞ 114

TABLOLAR LİSTESİ

Sayfa

Tablo 3.1. Kepware Kepserver Ex avantaj ve dezavantajları 42
Tablo 3.2. Microsoft SQL Server avantaj ve dezavantajları..... 44
Tablo 3.3. Grafana avantaj ve dezavantajları 45
Tablo 3.4. OPC UA verileri ve anlamları 76
Tablo 3.5. MTConnect verileri ve anlamları 78
Tablo 3.6. FOCAS verileri ve anlamları..... 81
Tablo 3.7. MODBUS verileri ve anlamları..... 83
Tablo 3.8. Enerji analizörü MODBUS verileri ve anlamları..... 85

ŞEKİLLER LİSTESİ

Sayfa

Şekil 3.1. Endüstri devrimlerinin tarihsel gelişimi 21
Şekil 3.2. Endüstri 4.0 teknolojileri 23
Şekil 3.3. Konvansiyonel silindirik taşlama tezgâhı 27
Şekil 3.4. CNC torna tezgâhı 28
Şekil 3.5. Siemens kontrolcülü CNC tezgâh 29
Şekil 3.6. Siemens 840D SL ürün ailesi 30
Şekil 3.7. Mazak kontrolcülü CNC tezgâh 31
Şekil 3.8. Fanuc kontrolcülü CNC tezgâh 32
Şekil 3.9. Yarı otomatik freze tezgâhı 33
Şekil 3.10. OPC UA mimarisi 34
Şekil 3.11. Siemens OPC UA mimari örneği 35
Şekil 3.12. BFC Gateway mimarisi ve desteklediği protokoller 36
Şekil 3.13. BFC Gateway mimarisi 36
Şekil 3.14. MTConnect 37
Şekil 3.15. MTConnect mimarisi ve diyagramı 38
Şekil 3.16. Fanuc FOCAS mimarisi 39
Şekil 3.17. MODBUS mimarisi 40
Şekil 3.18. MODBUS TCP/IP mimarisi 40
Şekil 3.19. IIoT katmanları 46
Şekil 3.20. Tasarlanan IIoT mimarisi 48
Şekil 3.21. Tasarlanan IIoT altyapı mimarisi 49
Şekil 3.22. DMG Mori DMU 65 monoBlock 5 eksen işleme tezgâhı 49
Şekil 3.23. Siemens 840D SL NCU ve operatör paneli 50
Şekil 3.24. Ethernet ve Switch bağlantısı 50
Şekil 3.25. Kepware Kepserver Ex şema örneği 51
Şekil 3.26. Grafana rapor paneli örneği 52
Şekil 3.27. Breton marka beş eksen dikey işleme tezgâhı 53
Şekil 3.28. Mazak beş eksen dikey işleme tezgâhı 54
Şekil 3.29. Mori Seiki marka üç eksen dikey işleme tezgâhı 55

Şekil 3.30. Broetje C-Frame panel montaj tezgâhı 56
Şekil 3.31. Despatch ısıl işlem fırını 57
Şekil 3.32. Avure form pres tezgâhı 58
Şekil 3.33. Sinümerik lisans menüsü ve OPC UA opsiyonu 60
Şekil 3.34. Sinümerik Operate OPC UA konfigürasyon sayfası 61
Şekil 3.35. Fanuc tezgah kontrolcüsü common menüsü 63
Şekil 3.36. Fanuc tezgâh kontrolcüsü FOCAS menüsü 64
Şekil 3.37. BFC Gateway sunucu uygulama sayfası 65
Şekil 3.38. Örnek MODBUS adres tablosu 65
Şekil 3.39 Kepware yeni kanal eklenmesi 67
Şekil 3.40. Kepware OPC UA konfigürasyonu 68
Şekil 3.41. Kepware veri seçimi 69
Şekil 3.42. Kepware üzerinde verilerin anlık görüntülenmesi 69
Şekil 3.43 IoT Gateway ve tezgâh bağlantıları 70
Şekil 3.44. Yeni agent ekleme sayfası 70
Şekil 3.45. IoT Gateway agenta verilerin eklenmesi 71
Şekil 3.46. IoT Gateway ile yayınlanan veriler 72
Şekil 3.47. 001_Breton_OPCUA tezgahından toplanan anlık veriler 75
Şekil 3.48. OPC UA çalışma durumu diyagramı 76
Şekil 3.49. OPC UA duruş durumu diyagramı 77
Şekil 3.50. 002_Mazak_MTConnect tezgahından toplanan anlık veriler 78
Şekil 3.51. MTConnect çalışma durumu diyagramı 79
Şekil 3.52. MTConnect duruş durumu diyagramı 79
Şekil 3.53. 003_MoriSeiki_FOCAS tezgahından toplanan anlık veriler 80
Şekil 3.54. FOCAS çalışma durumu diyagramı 81
Şekil 3.55. FOCAS duruş durumu diyagramı 81
Şekil 3.56. 004_Broetje_BFC tezgahından toplanan anlık veriler 82
Şekil 3.57. 005_Despatch_MODBUS fırınından toplanan anlık veriler 83
Şekil 3.58. MODBUS çalışma durumu diyagramı 84
Şekil 3.59. MODBUS duruş durumu diyagramı 84
Şekil 3.60 006_Avure_MODBUS_EA tezgahından toplanan anlık veriler 85

Şekil 3.61. Enerji analizörü MODBUS çalışma durumu diyagramı 86
Şekil 3.62. Enerji analizörü MODBUS duruş durumu diyagramı 86
Şekil 4.1. Uygulama sensör change fonksiyonları 88
Şekil 4.2. Uygulama anlamlandırma fonksiyonları 88
Şekil 4.3. Uygulama kaydetme fonksiyonları 88
Şekil 4.4. Veri tabanı fonksiyonları 89
Şekil 4.5. Tezgâh durumlarının tutulduğu değişken fonksiyonları 89
Şekil 4.6. Sensor Change sınıf fonksiyonları 90
Şekil 4.7. runForOpcUa fonksiyonu 90
Şekil 4.8. setValueForOPCUa fonksiyonu 91
Şekil 4.9. tezgahDurumKaydetOPC fonksiyonu 92
Şekil 4.10. runForMTConnect fonksiyonu 93
Şekil 4.11. runForFocas fonksiyonu 93
Şekil 4.12. runForMODBUS fonksiyonu 93
Şekil 4.13. runForEAMODBUS fonksiyonu 94
Şekil 4.14. setValueForDBMTConnect fonksiyonu 94
Şekil 4.15. setValueForDBFocas fonksiyonu 95
Şekil 4.16. setValueForDBMODBUS fonksiyonu 95
Şekil 4.17. setValueForDBEAMODBUS fonksiyonu 95
Şekil 4.18. tezgahDurumKaydetMTConnect fonksiyonu 96
Şekil 4.19. tezgahDurumKaydetFocas fonksiyonu 97
Şekil 4.20. tezgahDurumKaydetMODBUS fonksiyonu 98
Şekil 4.21. tezgahDurumKaydetEAMODBUS fonksiyonu 99
Şekil 4.22. Veri tabanı bağlantı fonksiyonu 100
Şekil 4.23. getAlarmAciklama fonksiyonu 101
Şekil 4.24. Veri tabanı Kayıt Fonksiyonu 101
Şekil 4.25. TEZGAH_VERILERI_STD tablosu örnek sorgu 102
Şekil 4.26. Anlık tezgâh verileri Grafana ekranı 103
Şekil 4.27. 001_Breton_OPC tezgâh durumu görüntüleme ekranı 104
Şekil 4.28 002_Mazak_MTConnect tezgâh durumu görüntüleme ekranı 104
Şekil 4.29 003_MoriSeiki_Focas tezgâh durumu görüntüleme ekranı 105

Şekil 4.30. 004_Broetje_BFC tezgâh durumu görüntüleme ekranı 105
Şekil 4.31. 005_Despatch_MODBUS tezgâh durumu görüntüleme ekranı 106
Şekil 4.32. 006_Avure_MODBUS_EA tezgâh durumu görüntüleme ekranı..... 106

KISALTMALAR

AMQP:	Advanced Message Queuing Protocol
API:	Uygulama Programlama Arayüzü
Ar-Ge:	Araştırma ve Geliştirme
AR:	Artırılmış Gerçeklik
ASCII:	American Standard Code for Information Interchange
AUTO:	Otomatik
BFC Gateway:	Brownfield Connectivity Gateway
BFC:	Brownfield Connectivity
CC3200:	Connectivity Controller 3200
CFG:	Configuration
CNC:	Bilgisayar Sayısal Kontrol
CoAP:	Constrained Application Protocol
DDS:	Data Distribution Service
DHT 11:	Digital Humidity and Temperature 11
EA:	Enerji Analizörü
ERP:	Kurumsal Kaynak Planlaması
FOCAS:	Fanuc Open CNC API Specification
GEE:	Genel Ekipman Etkinliği
HTTP:	Hypertext Transfer Protocol
IIoT:	Endüstriyel Nesnelerin İnterneti
IP:	Internet Protocol
IPv6:	Internet Protocol version 6
IoT:	Nesnelerin İnterneti
IT:	Information Technology
IV. Sanayi Devrimi:	Dördüncü Sanayi Devrimi
JOG:	Adım Kontrol
JSON:	JavaScript Object Notation
M2M:	Makine-Makine İletişimi
mA:	miliamper

ms:	milisaniye
MCU:	Micro Controller Unit
MDI:	Manuel Veri Girişi
MES:	Manufacturing Execution System
MODBUS:	Modicon Bus
MQTT:	Message Queuing Telemetry Transport
MTConnect:	Makine Takım Bağlantısı
NC:	Sayısal Kontrol
NCU:	Sayısal Kontrol Ünitesi
OPC UA:	Open Platform Communications Unified Architecture
OT:	Operational Technology
PL:	Power Line
PLC:	Programmable Logic Controller
PLM:	Product Lifecycle Management
REST:	Representational State Transfer
RS-232 ve RS-485:	Seri haberleşme bağlantılarında kullanılan standartlardır.
RTU:	Remote Terminal Unit
SL:	Solution Line
SP:	Service Pack
TCP/IP:	Transmission Control Protocol/Internet Protocol
TÜBİTAK:	Türkiye Bilimsel ve Teknolojik Araştırma Kurumu
UDP:	User Datagram Protocol
URL:	Uniform Resource Locator
WORD:	16 bitlik bir tam sayı türü
XML:	eXtensible Markup Language
XMPP:	Extensible Messaging and Presence Protocol

1. GİRİŞ

Son yıllarda endüstriyel alanlarda meydana gelen dördüncü sanayi devrimi, üretimde rekabetin artmasını, hızın ve kalitenin öncelikli hale gelmesini beraberinde getirmiştir. Bu dönüşümün temelinde, üretim süreçlerinin izlenebilir ve ölçeklenebilir olması büyük önem taşımaktadır (Dengiz, 2017:41). Endüstri 4.0, internetin gelişimi ve yaygınlaşmasıyla ortaya çıkmış, sanal ve fiziksel dünyanın birleşmesine olanak tanıyarak siber-fiziksel sistemlerin üretimde yeni teknolojilere yol açmasına neden olmuştur (Yıldız, 2018:549). Bu sistemler, üretim süreçlerini takip etme, analiz etme ve yönlendirme yeteneği sağlayarak "Akıllı Fabrika" konseptini ortaya çıkarmıştır. Ayrıca, bu sistemler sayesinde makinelerden elde edilen verilerin işlenerek arızaların önceden tespit edilmesi ve önleyici eylemler önerilmesi mümkün hale gelmiştir (Gürsoy ve ark., 2019:57).

Bu bağlamda, fabrikaların işleyişini anlamak ve Endüstri 4.0'ın nasıl işlediğini gözlemlemek önemlidir. Yatırım kararlarının doğru alınabilmesi için mevcut ekipmanların verimliliğinin takip edilmesi gereklidir. Gelişen sistemler, ekipmanlardan elde edilen verilerin Genel Ekipman Etkinliği (GEE) parametrelerine dönüşmesini sağlayarak fabrikaların daha doğru yatırım yapmasına yardımcı olmaktadır. Tüm bu faktörler, firmalar için hız ve teknoloji kullanımının önemini artırmıştır.

Endüstri 4.0 kavramının bileşenlerinden biri olan ve önem taşıyan teknolojilerden bir tanesi de Nesnelerin İnterneti (IoT) teknolojisidir. Bu teknoloji temelinde fiziksel sinyallerin üretmiş olduğu verilerin, siber sistemlere aktarılmasını amaçlamaktadır. Endüstri 4.0 kavramı çerçevesinde yol haritası oluşturan firmalar, fiziksel sistemlerin ürettiği verileri dijitalleştirebilmek için IoT teknolojisini kullanmaktadır. IoT teknolojisi günlük yaşantımızda sıkça karşımıza çıkmaktadır. Endüstriyel alanda IoT teknolojisi ise kullanılan iletişim protokolleri, veri çeşitliliği ve büyüklüğü, güvenlik protokolleri gibi farklılıklar nedeni ile Endüstriyel Nesnelerin İnterneti (IIoT) olarak adlandırılmaktadır. IIoT teknolojisi kullanılarak, firmaların üretim alanlarında bulunan fiziksel sistemlerinden verilerin toplanması sağlanmaktadır. Üretim alanlarından toplanan bu veriler genellikle karmaşık bir yapıda olmaktadır (Liu ve ark., 2018:16).

Fiziksel sistemler üzerinde bulunan verileri dijitalleştirmek isteyen firmalar, IIoT altyapısını kurmaktadırlar. Bu altyapı, verilerin toplanacağı ekipmanların belirlenmesinden, ekipmanlardan hangi protokoller ile veri toplanacağından, verilerin sunucuya nasıl aktarılacağından ve verilerin nasıl işlenip depolanacağından oluşmaktadır. Üretim alanlarında bulunan ekipman çeşitliliği bu altyapının kurulumu sırasında birden fazla protokolün kullanılmasına neden olmaktadır. Bu protokoller Open Platform Communications Unified Architecture (OPC UA), Machine Tool Connect (MTConnect), Modicon Bus (MODBUS), Fanuc Open Computer Numerical Control Application Programming Interface Specification (FOCAS) gibi endüstriyel protokoller olabilmektedir. Bu protokollerin yanı sıra ekipmanlardan veri toplanabilmesi için geliştirilen üçüncü parti yazılımlarda benzer protokoller ile eş değer çalışmaktadır. Bu yazılımlardan biri ise Brownfield Connectivity Gateway (BFC Gateway) yazılımıdır. Bu katmanda yer alan karmaşıklık ve farklılıklar, üretimden toplanan verilerin çeşitliliğine neden olmaktadır. Bu çeşitlilik, genellikle ekipman ve veri tiplerinde yaşanan farklılıklar kaynaklıdır.

Bu çalışmanın odak noktası üretim alanlarında bulunan ve üretimin en değerli ekipmanları olan tezgahlardan, IIoT teknolojisi kullanılarak üretim verilerinin toplanması, standartlaştırılması, anlamlandırılması ve görselleştirilmesidir. Bu kapsamda GEE hesaplamalarında kullanılan kullanılabilirlik parametresinin tespit edilmesi için gereken çalışma verilerinin anlamlandırılması ve elde edilen bu verilerin görselleştirilerek, tezgahların izlenebilirliğinin artırılması hedeflenmektedir. Bu doğrultuda, endüstriyel firmalarda IIoT altyapısının nasıl olması gerektiği, kullanılacak mimari, protokoller ve yöntemler, toplanan verilerin karmaşıklığı ve bu verilerin standardizasyonu üzerine Araştırma ve Geliştirme (Ar-Ge) çalışmaları gerçekleştirilmektedir. Bu çalışma önümüzdeki yıllarda ülkemiz sanayisinde dijitalleşme vizyonu ile yola çıkan firmalara bir rehber niteliği sağlamak ile, araştırmacıların ileride gerçekleştirebileceği çalışmalara da katkı sağlamaktadır. Çalışma, Endüstri 4.0 teknolojilerinin temelini oluşturan 'veri' ile ilgili olup, bu bağlamda önemli bir değer katkısı sağlamaktadır.

2. LİTERATÜR TARAMASI

Bu bölümde çalışmanın amacına uygun olarak gösterilebilecek ve konu ile ilgili doğrudan ya da dolaylı olarak ilgili olan çalışmalar incelenmektedir. Çalışmayı kapsamakta olan Endüstri 4.0 kavramının tanımlanması, disiplinler arasındaki etkileri, sosyal etkileri ve öngörülen sonuçları, Türkiye'deki yeri ve Endüstri 4.0 kavramı ile var olan teknolojileri ve bu teknolojilerden bir tanesi olan IoT kavramını konu edinen çalışmalar taranmaktadır.

Yang, Zhang ve Vasilakos (2014) tarafından gerçekleştirilen "*Nesnelerin İnterneti için Güven Yönetimi Üzerine Bir Anket*" adlı çalışmada, IoT üzerinde güven duygusunun geliştirilmesine yönelik bir anket çalışması sunulmuştur. Çalışmanın temel amacının IoT güven yönetimi üzerine kapsamlı bir araştırma eksikliği olduğu düşüncesi ile gerçekleştirdiğini belirtmişlerdir. Çalışmada, insanların belirsizlik ve risk algılarını ele alarak güven yönetiminin önemine vurgu yapılmıştır. Çalışma, IoT'nin güven yönetimi alanında daha fazla araştırmaya ihtiyacı olduğunu savunarak, mevcut literatüre bu kapsamda geniş bir katkı sağlamayı hedeflemiştir. Sensörler ve mobil cihazlar gibi insan sosyal hayatı ile ilgili bilgilerin toplanmasına yardımcı olan cihazlar aracılığı ile toplanan verilerin akıllı hizmetlerin geliştirilmesinde kullanılabileceği vurgulanmıştır. Sonuç olarak, güven öğeleri ve IoT sistem modeli üzerine güven değerlendirmesinin, güven oluşturma özelliklerinin dijital verinin işlenmesi amacı ile teknik bir yaklaşım oluşturduğu açıklanmaktadır.

Ercan ve Kutay (2016) tarafından hazırlanan "*Endüstride Nesnelerin İnterneti (IoT) Uygulamaları*" adlı çalışmada Endüstri 4.0 teknolojilerinden biri olan IoT teknolojisinin ve uygulamalarının incelenmesi ve literatüre katkı sağlanması hedeflenmiştir. Bu çalışma, IoT teknolojisini, internet kullanımının endüstriyel süreçlerde yaygın kullanılmasıyla, akıllı cihazların birbirleri ile iletişim kurabilen nesnelere dönüşmesini sağlayan ve bu entegrasyonun yapı taşı olan bir teknoloji olarak ele almıştır. Bu teknolojinin, çok sayıda küçük boyutlu ve kablosuz haberleşme sağlayabilen sensörler kullanılarak çevremizde gerçekleşen durumları izleme ve bilgi toplama kabiliyeti sunduğu vurgulanmaktadır. Bu teknoloji beraberinde var olan bir diğer kavramın ise IIoT olduğu ele alınmıştır ve bu teknolojinin endüstriyel süreçlere uygulanması olarak açıklanmıştır. Çalışmada bu teknolojinin, algılayıcılar ve üretim terminallerinden

toplanan verilerin, bulut bilişim servislerinde depolanıp bilgi sistemleri tarafından günlük ve iş yaşamını, endüstriyel üretim süreçlerini olumlu etkileyecek sistemlerin geliştirilmesine imkân sağladığı görülmüştür. Çalışma, endüstriyel süreçlerde IoT'nin olumlu etkilerini ele alırken, otomatik depolama, önleyici bakım, madencilik ve akıllı çevre düzenlemeleri gibi örnekler üzerinden bu teknolojinin sunduğu avantajları tartışmaktadır. Ayrıca, IoT'nin uygulama alanlarında gelişmeye açık alanların olduğu vurgulanmaktadır. Sonuç olarak, IoT teknolojisi endüstriyel ve günlük süreçlere entegrasyonu ile insan hayatını ve endüstriyel süreçlerin olumlu yönde etkileyecek çeşitli uygulamalara yol açtığı görülmüştür. Bu teknoloji ile varlığını ortaya koyan akıllı cihazların kullanımı, insan hatalarının en aza indirilmesini ve gerçek zamanlı verinin değerlendirilmesi konularında önemli avantajlar sağlayabildiği ancak, tam olarak fayda sağlanabilmesi için farklı cihazların birlikte çalışabilmesi için bilgi işleme zorluğu, siber güvenlik ve kullanıcı endişeleri gibi zorlukların çözülmesi ile bu teknolojinin potansiyelini tam olarak ortaya koyabileceği sonucuna varılmaktadır.

Ives, Rodriguez ve Palese (2016) tarafından gerçekleştirilen "*Iot ile Müşteri Hizmetlerinin Geliştirilmesi*" başlıklı çalışma, kuruluşların IoT tarafından yönlendirilen yeni bir döneme girildiğini ve bu dönemin önceki bilişim teknolojileri dönemlerinden daha derin değişikliklere neden olabileceğini vurgulamıştır. Çalışma, genellikle düşük maliyetli, küçük elektronik cihazlara sahip olan kuruluşların algılama, iletişim ve bilgi işleme yetenekleri ile bu değişikliklere nasıl uyum sağlayabileceğini ele almıştır. Özellikle IoT cihazlarının yaygın olarak benimsenmesiyle ortaya çıkacak dijital veri akışlarının iş ortamlarını dönüştürme potansiyelleri incelenmiştir. Bu veri akışlarının, kuruluşlara çeşitli fırsatlar sunarak müşteri deneyimlerini artırmak amacı ile kullanılabileceği varsayımı yapılmıştır. IoT teknolojisinin potansiyelinin büyük olduğu belirtilmiş, ancak yeni teknolojilere adaptasyon sürecinde yaşanan zorluklar ve kontrol edilmesi gereken karmaşıklıklar nedeni ile kuruluşların maksimum yararı sağlamak için gerekli uzmanlığı geliştirmekte zorlandıkları vurgulanmıştır. Sonuç olarak çalışma, IoT'nin iş dünyasını nasıl dönüştürebileceği, müşteri deneyimini nasıl iyileştirebileceği ve potansiyeli değerlendirmek için kuruluşların hangi stratejileri benimseyebileceği alanlarında kapsamlı bir inceleme sunmaktadır.

Çavdar ve Öztürk (2017) tarafından hazırlanan "*Nesnelerin İnterneti için Yeni Bir Mimari Tasarımı*" adlı çalışma, internetin hayatımızdaki temel bir bileşen haline gelmesi ile IoT kavramının giderek önem kazanması ile, mevcut mimari modellerin yetersizlikleri ele alınarak bu alandaki literatür incelenmiştir. IoT'nin hızla gelişen yapısında, veri iletişimi ve nesneleri tanıma sorunlarını barındıran zorlukların ortaya çıktığı belirtilmiştir. Bu çalışmada, mevcut literatür incelendiğinde temel bir referans noktasının olmaması ve bu eksikliğin giderilebilmesi adına katmanlı bir mimari model önerilmiştir. Önerilen model, özellikle dönüşüm ve karar alt katmanlarına vurgu yaparak, veri trafiğindeki yoğunluğu azaltmayı, sistem güvenilirliğini artırmayı ve bütünlüğü korumayı amaçlamaktadır. Bu bağlamda, farklı literatürde öne çıkan mimari modellerin eksiklikleri ve sınırlamaları detaylı bir şekilde incelenmiş ve ilgili literatür bu çalışmanın teorik temelini oluşturmuştur. Önerilen model, benzetim ve matematiksel karşılaştırmalar ile desteklenerek, algılama ve karar katmanlarındaki etkileşimlerin açıkça anlatılması ile diğer çalışmalardan ayrılmaktadır. Matlab ortamında gerçekleştirilen bir çalışma ile önerilen mimari modelin algılama ve karar alt katmanlarının kullanılmasının veri trafiğinde belirgin bir azalmaya yol açtığı gözlemlenmiştir. Bu durum, algılayıcılardan gelen verilerin internete açılmadan doğrudan harekete geçiricilere yönlendirilerek, internete açılan paket sayısında önemli bir azalma sağlanmasıyla açıklanmaktadır. Ayrıca önerilen modelin, katmanlar arasında veri iletimini hızlı ve etkili hale getirebilmek amacıyla alt katmanlarda verilerin uygun bir şekilde yönlendirilip dönüştürülmesi konusunda benzersiz bir yaklaşım sunduğu belirtilmiştir. Sonuç olarak, gelecekte önerilen mimari modelin veri trafiğini dahada azaltmak, farklı teknolojilere uyum sağlamak ve özel uygulama alanlarını kapsamak üzere yeni protokollerin yazılması, mevcut protokollerin güncellenmesi ve sistem bütününde çalışmalar yapılması gerektiği ön görülmektedir.

Gabaçlı ve Uzunöz (2017) tarafından gerçekleştirilen "*IV. Sanayi Devrimi: Endüstri 4.0 ve Otomotiv Sektörü*" adlı çalışmada sanayide yaşanan devrimsel gelişimler, Endüstri 4.0 kavramının ortaya çıkmasında rol oynayan nedenler ve Endüstri 4.0'ın otomotiv sektörü üzerindeki etkileri incelenmiştir. Yıllar içerisinde devrimlerin ve farklılaşmaların meydana gelmesindeki asıl nedenler ele alınırken bu durum azalan kaynaklar ve artan üretim hacminin bir sonucu olarak değerlendirilmiştir. Bu çerçevede incelendiğinde yıllar

içerisinde endüstri devrimlerinin ortaya çıktığı ve son olarak Endüstri 4.0 devriminin günümüzde gelişmesinin nedenlerinin başlıcaları artan maliyetler, sürekli yenilik beklentileri ve şiddetlenen küresel rekabet olarak gösterilmiştir. Bu nedenler ele alındığında bu gelişimden en çok etkilenecek olan sektörün otomotiv sektörü olacağı değerlendirmesi neticesinde, Endüstri 4.0 kavramının sektör özelinde oluşturabileceği avantaj ve dezavantajları değerlendirilmiştir. Neticede çalışmanın varmak istediği nokta değerlendirildiğinde Endüstri 4.0 kavramının gelişen ve gelişmekte olan ülkeler için bir fırsat kapısı olduğu çıkarımı yapılabilmektedir. Bahsi geçen ülkelerden bir tanesi olan Türkiye tarafından incelendiğinde otomotiv sektörü ülkenin lokomotif sektörlerinden bir tanesi olarak değerlendirilmiştir. Bu sektörde rekabet, maliyet, kalite ve verim gibi Endüstri 4.0'ın ortaya çıkmasına neden olan değerler fazlasıyla önem arz ettiğinden ötürü Endüstri 4.0 devriminin getirmiş olduğu değişimlerden en çok etkilenmesi beklenen sektörün otomotiv sektörü olduğu yönünde değerlendirmeler yapılmıştır. Bu nedenle Endüstri 4.0 değişimlerine Dünya ile aynı anda uyum sağlamak beraberinde birçok avantajı taşıyabilecekken, bu adımlarda geri kalınması durumunda fırsat olarak görülebilecek değerler dezavantaj olarak geri dönebileceği öngörüsüne ulaşılabilmektedir. Çalışma bu sonuç çevresinde öneriler ile alınabilecek aksiyonları da ortaya koymaktadır.

Gündüz ve Daş (2017) tarafından sunulan "*Nesnelerin İnterneti: Gelişimi, Bileşenleri ve Uygulama Alanları*" adlı çalışma IoT teknolojisini incelemiş ve bu teknolojinin gelişimini destekleyen ana kavramın farklı teknolojilerin ve ağ teknolojileri ile iletişim çözümlerinin entegrasyonu olduğunu ele almıştır. IoT uygulamalarının temelini, izleme ve takip teknolojileri, kablolu ve kablosuz sensör ağları, geliştirilmiş iletişim protokolleri gibi bileşenlerin oluşturduğu değerlendirilmiştir. Çalışma, IoT'nin gelişmesinde sosyal bilimler, elektronik ve telekomünikasyon gibi farklı alanlarda yürütülen aktivitelerin ciddi katkılar sağladığını belirtir. İnternetin gelişimi ve evrimi ele alınırken; 1969 yılında başlayan ağ sisteminin gelişerek günümüzde 50 milyar nesnenin internete bağlı olacağı öngörüsüyle genişlediği ifade edilmiştir. Bu bağlamda, çalışma IoT'nin temel kavramları, bileşenleri ve gelişimi için temel uygulamalar ve alınabilecek siber güvenlik önlemleri üzerine incelemelerde bulunmuştur. Sonuç olarak çalışmada, internetin insanlık üzerinde etkileri en hızlı görülen ve gelişen teknolojik gelişme olduğu vurgulanırken, IoT

teknolojisinin insan hayatını daha kapsamlı etkileyerek bu büyümeyi dahada hızlandırarak toplumların yaşam tarzını etkileyeceği öngörüsünde bulunulmuştur. Çalışma, IoT'nin her yerden, herkesle, her nesneyle ve her zaman bağlantı ilkesi ile geliştiğini, günümüzde gelişen teknolojilerin IoT kavramını uygulanabilir hale getirdiğini ancak verimlilik ve ölçeklenebilirlik konularında henüz tam anlamıyla performans veremediğini ifade etmiştir. IPv6 (Internet Protocol Version 6) gibi yeni ağ teknolojilerinin ve protokollerinin gelişmesinin, IoT'nin daha geniş kapsamlı kullanılmasına olanak sağladığı belirtilmiştir. Son olarak çalışmada, IoT'nin etkili bir şekilde kullanılabilmesi için protokol ve standart geliştirme çalışmalarının akademik ve ticari düzeyde yapıldığını ve bu özelliklerin IoT mimarisinin oluşumunda ve ağ altyapısının tasarımında önemli rol oynadığı vurgulanmaktadır.

Oral ve Çakır (2017) tarafından yayımlanan "*Nesnelerin İnterneti Kavramı ve Örnek Bir Prototipin Oluşturulması*" adlı çalışmada Endüstri 3.0 ve otomasyon çağı ile manuel sistemlerin yerini otomatik sistemlerin alması ile, özellikle IoT ve Makine-Makine İletişimi (M2M) teknolojilerinin gelişmesi sayesinde, insanların yaşamlarının kolaylaştığı ve birçok alanda değiştiği vurgulanmıştır. Çalışmada IoT, cihazlara uzaktan bağlanılmasını ve kontrol edilebilmesini sağlayan bir platform olarak tanımlanmış, algılayıcı teknolojilerindeki etkisi nedeniyle küçük ve düşük maliyetli algılayıcı üretiminin arttığı belirtilmiştir. Sıcaklık, basınç, titreşim, ses, ışık, koku gibi çeşitli sensörlerin kullanılabildiği ve bu sensörlerin IoT teknolojisinin gücünü arttıran etkenlerden olduğuna değinilmiştir. Çalışma, IoT hakkında bilgi verirken aynı zamanda mevcut IoT ve M2M platformları kullanılarak gerçekleştirilen prototip bir uygulamaya da yer verilmiştir. Gerçekleştirilen prototip çalışmada, MQTT (Message Queuing Telemetry Transport) protokolü ve Node MCU (Micro Controller Unit) kullanılarak bağlanan DHT 11 (Digital Humidity and Temperature 11) sensörü ile toplanan sıcaklık ve nem verisinin kablosuz olarak takip edilmesini sağlamıştır. Ayrıca farklı tipte sensörler entegre edilerek kullanıcıların eş zamanlı olarak ilgili verileri takip edebilmesine olanak sağlamıştır. Bu sonuçlar neticesinde, algılayıcı teknolojileri ve kablosuz iletişim protokollerinin entegrasyonu ile elde edilen IoT çalışmalarının pratik uygulama potansiyeli ortaya koyulmaktadır.

Akben ve Avşar (2018) tarafından gerçekleştirilen "*Endüstri 4.0 ve Karanlık Üretim: Genel Bir Bakış*" adlı çalışmada Endüstri 4.0 ve teknolojileri kullanılarak çalışma hayatlarında farklı pozisyonlarda görev almalarını sağlayabilecek insanlar ve akıllı cihazlar ile iletişim kurabilen ve bu teknolojiler doğrultusunda çeşitli yeteneklere sahip robotlar ile karanlık üretim kavramları incelenmiştir. Çalışmada, Endüstri 4.0 kavramı ve teknolojilerinin çalışma hayatına etkileri incelenmiş, özellikle insanlar, akıllı cihazlar ve robotlar arasındaki iletişim, karanlık üretim kavramı üzerinden değerlendirilmiştir. Yazarlar, Endüstri 4.0 kavramının çalışma hayatındaki çeşitli pozisyonlarda görev alan insanları ve akıllı cihazlar ile iletişim kurabilen, çeşitli yeteneklere sahip robotları barındıran karanlık üretim kavramını ele almışlardır. Sonuç olarak bu çalışmada, Endüstri 4.0 ve teknolojileri anlatılmış, bu teknolojiler kullanılarak varılabilecek nihai sonucun karanlık fabrikalar olması üzerinde değerlendirmelerde bulunulmuştur. Karanlık fabrikaların uzun vadede Akıllı Fabrikaların bir sonucu olarak ortaya çıkabileceği ön görülmüş ve bu kavramın avantaj ve dezavantajları irdelenmiştir. Bu kapsamda, Endüstri 4.0 kavramının getirdiği teknolojik gelişmelerin çalışma hayatındaki etkileri, insan-robot etkileşimi ve karanlık üretim kavramının gelecekteki potansiyel sonuçları ele alınmıştır.

Altuntaş (2018) tarafından gerçekleştirilen "*Dijital Dönüşüm Uygulamalarının Kurumların Marka Değeri Üzerindeki Etkisi*" adlı çalışmada Endüstri 4.0, diğer adıyla Dijital Dönüşüm kavramının beraberinde getirmiş olduğu uygulamaların, firmaların marka değeri üzerindeki etkilerinin ortaya koyulabilmesi hedeflenmektedir. Dijital Dönüşüm' ün iş dünyasında rekabeti ve müşteri deneyimini artırmak amacıyla, firmaların gelecek stratejilerinin planlanmasında büyük öneme sahip olduğu belirtilmiştir. Dijital Dönüşüm beraberinde gelişen teknolojik yeniliklerin organizasyonel süreçlere etkisi ve entegrasyonu sonucunda, müşteri memnuniyetini ve marka değerini olumlu yönde etkilediği görülmüştür. Çalışmada kullanılan yöntemler, kurumsal marka değerlerinin Dijital Dönüşüm uygulamaları kullanılarak nasıl arttırılabileceği üzerine odaklanılmıştır. Sonuçlar değerlendirildiğinde, Dijital Dönüşüm uygulamalarının müşteri deneyimini olumlu yönde etkilediği ve kurumların rekabet avantajını sağlayabildiği açıkça görülmüştür. Teknolojik gelişmelerin, müşterilerin deneyimini arttırmak amacıyla toplanan verilerin işlendiği algoritmalar sayesinde, müşterilerin bağlılığını arttırmanın yanı sıra, firmaların sektörlerindeki konumlarını güçlendirdiğini ve bu durumun marka

değeri üzerinde olumlu etki sağladığı görülmüştür. Bu çalışma, firmaların veya kurumların Dijital Dönüşüm stratejilerini, marka yönetimi odağına alarak değerlendirmesi gerektiğini de ortaya koymuştur. Dijital Dönüşüm stratejilerine odaklanarak firmaların marka değeri oluşturmak veya bu değeri arttırmak için süreçlerine katkı sağladığı görülmektedir.

Calp, Bahçekapılı ve Berigel (2018) tarafından gerçekleştirilen "*Endüstri 4.0 Kapsamında Akıllı Fabrikaların İncelenmesi*" adlı çalışmada Endüstri 4.0 ve alt bileşenleri açıklanırken Akıllı Fabrika kavramı incelenmiştir. Akıllı Fabrikaların evrim süreci, mimarisi, özellikleri, yaşam döngüsü, görevleri, kullanım alanları ve gerekli teknolojiler üzerine detaylandırılan çalışmada uygulama aşamasında teknik sorunlar yaşanmasına rağmen makineler arasında iletişim sağlanarak etkin üretim süreçlerinin gerçekleştirilebildiği ortaya koyulmuştur. Bunun beraberinde küresel rekabet ortamı ile başa çıkabilmek için Akıllı Fabrikaların etkilerinin önemli derecede olduğu tespit edilmiştir. Sonuç olarak fabrikalarda var olan nesnelerin akıllandırılarak büyük verilerin toplandığı ve bu verilerin sistemi optimize etmek için buluta aktarılarak ihtiyaç halinde kullanılabildiği, bu yetenekler sayesinde üretimden geribildirimlerin rahatlıkla ulaşılabildiği, kolay yönetim ve koordinasyon sağlanabildiği tespit edilmiştir. Bu durumun daha hızlı, verimli, kolay ve daha az maliyetler ile üretimin gerçekleştirilebileceği açıklanmıştır. Beraberinde Akıllı Fabrika kavramı ile enerji verilerine erişimin sağlanabileceğine ve bu verilerin işlenerek karar destek mekanizmalarına yardım sağlayabileceği ortaya koyulmuştur. Son olarak çalışma büyük küçük ölçekli firmaların bu uygulamaları kullanabilmeleri için eğitimin önemine değinerek, Akıllı Fabrika teknolojilerinin tüm endüstrilere entegre edilebilerek üretim süreçlerine katkı sağlayabileceği sonucuna ulaşılmaktadır.

Soylu (2018) tarafından gerçekleştirilen "*Endüstri 4.0 ve Girişimcilikte Yeni Yaklaşımlar*" adlı çalışma, Endüstri 4.0'ın girişimcilik kavramına yapmış olduğu etkileri incelemeyi amaçlamıştır. Çalışmada Endüstri 4.0 teknolojik gelişmelerin üretim süreçlerine yapmış olduğu yoğun etkilerin ele alındığı bir ortam olarak değerlendirmiştir. Bu durumun çıktıları olarak ele alınabilecek verimlilik, maliyet avantajı, karlılık gibi firmaları etkileyen etkilerin yanında istihdam, eğitim, girişimcilik, yatırım, büyüme, insan kaynakları gibi makro seviye de etkilerini de ele almıştır. Çalışma Endüstri 4.0

kavramının yeni olması ve beraberinde gelişen teknolojilerinde gelişime açık olması sebebi ile girişimcilik alanında büyük atılımlara neden olabileceğini öngörmüş ve bu girişimler sırasında yeni iş modellerinin ve farklı iş yapma uygulamalarının da ortaya çıkabilme ihtimalini değerlendirmiştir. Sonuç olarak bu çalışma, gerçekleşen teknolojik atılımlar çerçevesinde rekabeti artıran meydan okumaları Endüstri 4.0'ın temeli olarak değerlendirmiştir. Bunun beraberinde Endüstri 4.0 teknolojileri ele alınarak incelendiğinde bu devrimin kapsam ve hızı ile ilgili öngörülere ulaşılamamıştır. Bu kapsamda büyük bir devrimin beraberinde birçok teknolojik fırsatı ve girişimi de barındıracağı açıkça görülmektedir. Bu aşamada çalışmanın bakış açısı yapılan girişimlerin ve teknolojik hareketlerin birbirleri ile iç içe girişen çalışmalar yönünde olacağı yönündedir. Bu nedenle ileride gelişecek olan ve büyük atılım gerçekleştiren girişimlerin, beraberinde birçok girişiminde atılım yapmasını sağlayabileceği ön görülmektedir. Ayrıca çalışma Endüstri 4.0 beraberinde ortaya çıkması muhtemel yeni iş modelleri, farklı iş alanları ve mesleklerin girişimciler için yeni fırsatları barındırabileceği tartışılmaktadır. Girişimcilik kavramı tüm bunlar etrafında değerlendirilmektedir. Girişimcilik ile ilgili ortaya çıkan yeni konseptlerin dijital bağlılık temelli platform yapısında, satın alma yerine kiralama gibi değişen politikalar çevresinde şekillendiği çıkarımında bulunulmaktadır. Neticede bu çalışma, Endüstri 4.0 teknolojileri üzerine yapılacak olan girişimlerin büyük kazanımlar elde edebileceği öngörüsü üzerine bir sonuca varmaktadır.

Taş (2018) tarafından gerçekleştirilen "*Dördüncü Sanayii Devrimi'nin (Endüstri 4.0) Çalışma Hayatına ve istihdama Muhtemel Etkileri*" adlı çalışmada gelişen ve hayatımızın bir parçası haline gelen teknolojik gelişmelerden olan özellikle üç boyutlu yazıcılar, yazılım, robotlar ve yapay zeka gibi teknolojilerin Endüstri 4.0 beraberinde öne çıkmasını ve Endüstri 4.0 çerçevesinde üretim süreçlerinin geçmişten günümüze gelişimini inceleyerek günümüzde bu süreçlerin istihdama etkilerini inceleyebilmeyi amaçlamıştır. Teknolojik gelişmeler geçmişten günümüze birçok farklı alanda tepki gösteren meslek grupları olduğuna değinilmiştir. Bu tepkiler genellikle işsizlik ve toplum içerisinde statü kaybı korkusuyla ortaya çıkmaktadır. Çalışmada bu olumsuz etkilerin her zaman olabileceği ancak Endüstri 4.0 süreçlerinde toplumların bu süreçlere doğru hazırlanabilmesi ile bu olumsuz etkilerin azalarak yerini farklı alanların değerlendirilmesi

sonucunda olumlu etkilere bırakılabileceği tartışılmıştır. Bu süreçte oluşması öngörülen olumsuz etkilerden etkilenebilecek kesimlere yönelik geliştirilebilecek stratejilerin ve adımların neler olabileceği ele alınmıştır. Çalışma teknolojik gelişmelerin toplumsal etkilerini ve bu değişime adaptasyon sürecinde alınabilecek tedbirleri incelemektedir. Sonuç olarak, Endüstri 4.0'ın, azalacak olan toplam işgücü ihtiyacının yanı sıra oluşması muhtemel yeni iş alanlarının ve bu doğrultuda nitelikli işgücüne geçişin gerektireceği adımların neler olabileceği incelenmiştir. Bu kapsamda potansiyel iş gücünün niteliklerini geliştirebilmek için hızlı ve etkili eğitim politikalarının geliştirilmesinin önemli olduğu vurgulanmıştır. Endüstri 4.0 beraberinde gelecek olan teknolojilerin iş yapma biçimlerinin ve iş tanımlarının değişebileceğine ancak insanların işlerini kaybetmelerine sebep olmayacağını belirterek, toplumların psikolojik olarak etkilenmemeleri için gerekli eğitim politikaları ile desteklenmelerinin önemi ortaya koyulmaktadır. Bu gelişmelerin sonucunda yeni iş alanlarının açılacak olması ve verimli üretimin artması ile istihdamın artabileceği sonucuna varılmıştır. Özet olarak çalışma, istihdama yönelik bu avantajlardan faydalanabilmek için Endüstri 4.0'ın doğru analiz edilerek etkili bir şekilde kullanılmasının önemli olduğunu vurgulamıştır.

Toker (2018) tarafından gerçekleştirilen "*Endüstri 4.0 ve Sürdürülebilirliğe Etkileri*" adlı çalışmada Endüstri 4.0 gelişmelerinin sürdürülebilirlik üzerindeki etkileri incelenmiştir. Endüstri 4.0 gelişmelerinin temel amacını daha hızlı, esnek ve düşük maliyetli üretimi mümkün kılarak, Batılı ülke ekonomilerinin, Uzak Doğu ülkelerine karşılık olarak üretim ve rekabet gücünü arttırmak olarak ele almıştır. Bu sürecin beraberinde getirdiği gelişmelerin, yenilenebilir enerji kaynaklarının kullanımı, kaynakların daha verimli kullanılması, robotların geleneksel iş süreçlerinde aktif rol alması gibi konular eşliğinde sürdürülebilirlik etkileri incelenmiştir. Bu çalışmada Endüstri 4.0'ın sürdürülebilirlik üzerindeki etkileri literatür ve iş dünyasındaki gelişmeler doğrultusunda incelenmiştir. Sonuç olarak Endüstri 4.0'ın sürdürülebilirlik üzerindeki etkilerinin ortaya çıkarılabilmesi için çevre, ekonomi ve toplumun farklı boyutlarını ele almanın önemli olduğundan ve bu etkilerin anlaşılabilmesi için bu kavramlarında incelenmesi gerektiğini vurgulamıştır. Ekonomik boyutta incelendiğinde işletmelerin Endüstri 4.0 alanında yapmaları planlanan dijital yatırımların teşvik edilerek küresel ekonomide hızlı bire büyüme sağlamaları beklenmektedir. Bu durumun neticesi olarak dijitalleşen ürün ve

hizmetlerin, işletmeler ve ülkelerin sürdürülebilirlik performanslarını etkileyen temel değerler arasında yer alması çıkarımına varılmıştır. Toplum açısından incelendiğinde ise, işgücü piyasasına önemli etkiler doğuracağı ve bu nedenle çalışanların iş tanımlarında değişikler olabileceğini öne çıkarmıştır. Bu durumun neticesinde eğitim ve teknolojik okuryazarlık düzeylerinde süreklilik sağlanmasının önemi vurgulanmıştır. Sonuç olarak, Endüstri 4.0'ın henüz yeni olması ve etkilerinin henüz tam olarak görülememiş olması nedeni ile sürdürülebilirlik kapsamında çevre, ekonomi ve toplum üzerindeki etkileri henüz tam olarak belirlenememiştir ancak yapılan tahminler neticesinde çeşitli boyutlarda değişiklikler olabileceğine işaret edilmektedir.

Yıldız (2018) tarafından gerçekleştirilen "*Endüstri 4.0 ve Akıllı Fabrikalar*" adlı çalışmada Endüstri 4.0 ve teknolojileri incelenmiştir. Bu çalışmada modern sanayi gelişiminin tarihçesinden ve günümüze kadar gerçekleşmiş olan sanayii devrimlerinden bahsedilirken Endüstri 4.0 olarak adlandırılan Dördüncü Sanayii devriminin öneminden de bahsedilmektedir. Bu neticede şekillenmiş olan çalışmada Endüstri 4.0 ve Endüstri 4.0'ın yapı taşı teknolojiler anlatılmış, Endüstri 4.0 ve teknolojileri kullanılarak nihai hedef olarak gösterilebilecek olan akıllı fabrikalar hakkında bilgi verilerek Endüstri 4.0 veya Dördüncü Sanayii devriminin tarihi, gelişimi ve amaçları ile ilgili genel bir değerlendirme yapılmıştır. Bu değerlendirme sonucunda Endüstri 4.0 kavramının önemi net bir şekilde ortaya koyulmaktadır. Gerek sanayileşme olarak değerlendirildiğinde Pazar hacmi yüksek olan ülkelerin stratejik planlarında, gerekse sektörlerinde yer edinmiş Mercedes, BMW, Siemens gibi firmaların ülke gelirlerine ve istihdamına sağladıkları katkılar ve bu doğrultuda uyguladıkları stratejileri referans alarak Endüstri 4.0 kavramının önemini daha anlaşılır kılmaktadır. Tüm bu detayların neticesinde Türkiye Bilimsel ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından 2016 yılında ülkemizin endüstri devrimleri arasındaki konumunu tespit etmek amacı ile yapmış olduğu dijital olgunluk seviyesi ölçümünü baz almaktadır. Bu ölçüm neticesinde ülkemiz sanayisinin Endüstri 2.0 ile Endüstri 3.0 arasında olduğunu vurgulamaktadır. Sonuç olarak bu çalışmada, ülkemizin gelişen teknolojilerden ve rekabetçi üretimden uzak kalmaması için yapılması gereken adımlar öne çıkarılmaktadır.

Bayuk ve Demir (2019) tarafından hazırlanan "*Endüstri 4.0 Kapsamında Yapay Zekâ ve Pazarlamanın Geleceği*" adlı çalışmada, sanayi devrimlerinden biri olarak Endüstri 4.0 kavramı ele alınarak, bu kavramın teknolojik gelişmelere etkileri incelenmiştir. Özellikle, Endüstri 4.0 beraberinde gelişen teknolojiler arasında önemli bir rol oynayan yapay zekâ teknolojileri incelenmiştir. Yapay zekâ teknolojisinin günümüzdeki rolü ve pazarlama sektöründeki uygulamaları üzerine odaklanan çalışmada, yapay zekanın pazarlama alanında kullanılan uygulamalara entegrasyon süreci ve ilerleyen yıllarda nasıl kullanılabileceği üzerine incelemeler yer almaktadır. Çalışma, yapay zekâ teknolojisi ve uygulamalarının pazarlama sektöründeki etkilerine odaklanarak işletmelere ve müşterilere sağladığı katkıları incelemeyi amaçlamıştır. Bu nedenle, Endüstri 4.0'ın beraberinde getirmiş olduğu teknolojik değişimlerin pazarlama stratejileri üzerindeki etkileri ve yapay zekanın bu stratejilere ne yönlü katkı sağladığı ele alınmıştır. Bu çalışma neticesinde, pazarlama alanında Endüstri 4.0 beraberinde gelen teknolojilerin önemli bir değişime neden oldukları ve özellikle yapay zekâ teknolojisi ile geliştirilen uygulamaların bu değişimlerin mimarisi olduğu görülmektedir. Yapay zekâ uygulamalarının pazarlama aşamalarında; müşteri etkileşimi, karar verme süreçlerinin desteklenmesi ve optimizasyonu, müşteri ilişkilerinin iyileştirilmesi ve fiyatlandırılmaya yansıtılması, tanıtım ve marka operasyonlarının iyileştirilmesinde yaygın bir şekilde kullanıldığı ele alınmıştır. Bu çalışmanın çıktısı olarak, yapay zekânın getirmiş olduğu avantajları gözler önüne sererek işletmelerin bu avantajları değerlendirerek, kendi sektörleri ve çalışma alanları için kullanarak önemli bir fırsata çevirmeli ve etkili bir şekilde kullanmaları gerektiği sonucu gösterilebilmektedir.

Akram ve Dağdeviren (2020) tarafından hazırlanan "*Nesnelerin İnterneti için Gerçek Zamanlı Tasarsız Veri Toplama Platformu*" adlı çalışma, IoT cihazlarının çeşitliliğinin hızla artması ve farklı arayüzlere ihtiyaç duyulması sonucunda ortaya çıkan zorlukların önüne geçilmesi amacıyla gerçekleştirilen gerçek zamanlı tasarsız veri toplama, raporlama ve analiz platformunu ele almaktadır. Çalışmada önerilen platform, sensörlerden veri toplama, yerel veri işleme, bulut tabanlı sunucu ve ağ tabanlı arayüzden oluşan dört ana bileşenle tasarlanmıştır. Platform, sensörler arasında merkezi bir iletişim altyapısına ihtiyaç olmadan ağaç yapısında veri iletimi sağlayarak, genişletilebilir ve kullanıcı dostu bir çözüm sunmuştur. Çalışma deneysel testler yapılarak denenmiş ve bu

süreçte istikrarlı bir şekilde çalıştığı, anlık raporlar üretebildiği gözlemlenmiştir. Çalışmanın sonuçları incelendiğinde, IoT alanında geliştirilen veri toplama, raporlama ve analiz platformu tasarımının başarılı olduğu ve başarılı bir şekilde uygulanabildiği görülmüştür. Platform, donanımların üzerine eklenen ve gömülü olarak çalışan uygulama sayesinde verileri yerel bir bilgisayara iletmektedir. HTTP (Hypertext Transfer Protocol) protokolü yardımıyla bulut tabanlı sunucuya verilerin aktarılması gerçekleştirilmektedir. Kullanıcılar, web tabanlı arayüz sayesinde verileri inceleyebilmektedirler. Sonuç olarak deneysel çalışmalar, 96 saat boyunca gerçekleştirilen ölçümler ile ortamın ışık, sıcaklık, sinyal gücü gibi parametrik verilerin ölçümünü gerçekleştirerek çeşitli grafiksel raporları ortaya koymaktadır. Çalışma uygulamanın, platform bileşenlerinin bağımsız olması sayesinde farklı uygulamalarda ve alanlarda kullanılabileceğini ele almaktadır. Uygulamanın gelecek çalışmalarda Arduino, Node MCU gibi çeşitli platformların desteklenmesi ve yapay zekâ teknikleri ile verilerin daha derin analizlerinin yapılması gibi noktalarda geliştirilmesi hedeflenmektedir.

Annaç Göv ve Erdoğan (2020) tarafından gerçekleştirilen "*Dördüncü Endüstri Devriminin (Endüstri 4.0) Neresindeyiz?*" adlı çalışma 2011 yılından itibaren gündemde olan Endüstri 4.0 kavramının devrim niteliğindeki uygulamalarına odaklanmaktadır. Bu çalışmanın amacı, Gaziantep Teknopark 'ta faaliyet gösteren on şirket yöneticisinin Endüstri 4.0 uygulamalarına yönelik düşüncelerini incelemektir. İncelemelerin beraberinde Endüstri 4.0 uygulamalarından bulut bilişim, sistem entegrasyonu, IoT ve büyük veri teknolojilerinin yaygın olarak kullanıldığı görülmüştür. Ancak, siber-fiziksel sistemler, artırılmış gerçeklik ve simülasyon gibi teknolojilerin daha az kullanıldığı dikkat çekmiştir. Firmaların Endüstri 4.0 farkındalığının değişken seviyelerde olduğu, bazı firmaların bu uygulamaları kullanmalarına rağmen Endüstri 4.0'ı inceleme ve değerlendirme konusunda eksik kaldıkları belirtilmiştir. Sonuç olarak bu çalışmada Endüstri 4.0 farkındalığını artırmak için öneriler sunulmaktadır. Bu öneriler eğitim ve bilinç seviyelerinin artırılması, üniversitelerin konu ile ilgili paneller ve seminerler düzenlemesi, fuarların organize edilmesi, işçilerin konu ile ilgili daha nitelikli hale getirilmesi ve tüm aşamaların devlet tarafından desteklenmesi yer almaktadır. Çalışmada yer alan katılımcıların Endüstri 4.0 tanımları içerisinde, otomasyon, Dijital Dönüşüm, makine teknolojisi, yazılım entegrasyonu gibi unsurlar yer almaktadır. Bu araştırma,

Endüstri 4.0 devriminin Türkiye'de henüz yeni bir kavram olduğunu ve ülkede Endüstri 4.0 kavramına olan farkındalığın artırılması gerektiğini gözler önüne sermektedir. Elde edilen sonuçlar ve bulgular neticesinde bazı durumlarda Endüstri 4.0 uygulamalarının kullanılmasının dezavantajlı olabileceği öne çıkmaktadır.

Çark (2020) tarafından "*İşletmelerin Dijital Dönüşüm Sürecinde 'Nesnelerin İnterneti' Teknolojisinin Etkisi*" adlı çalışmada Endüstri 4.0 teknolojilerinden bir tanesi olan IoT teknolojisinin işletmeler üzerindeki etkisi incelenmiştir. Endüstri 4.0 ya da diğer adıyla Dijital Dönüşüm, günümüzde iş dünyasında daha sürdürülebilir rekabetin kurulması ve bu rekabeti avantaj haline dönüştürmek ve operasyonel süreçlerin verimliliğini artırmayı amaçlayan bir devrim olarak bu çalışmada ele alınmıştır. Bu kapsamda değerlendirildiğinde, IoT teknolojisinin, işletmelerin operasyonel süreçlerini optimize etmesinde, verimliliği artırmasında ve müşteri deneyimlerini iyileştirmesinde rol aldığı görülmüştür. IoT, fiziksel nesne ve ekipmanların sensörler ve internet teknolojisinin beraberinde getirmiş olduğu ağ teknolojileri kullanılarak, birbirleri ve diğer sistemler ile iletişim kurmalarını sağlayan bir oluşum sunmaktadır. Bu sayede işletmelerin, gerçek zamanlı veri toplama, analiz etme ve anlamlı öngörüler elde etme şansına sahip oldukları ortaya çıkmaktadır. Bu çalışma doğrultusunda bu teknolojinin rekabet avantajlarının yanı sıra, Pazar taleplerine daha hızlı ve esnek dolayısı ile daha verimli yanıt verilebildiğini ortaya koymuştur. Ancak, hem oluşabilecek siber tehlikeler nedeni ile hem de veri entegrasyonunda yaşanan zorluklar ve maliyet gibi faktörler nedeni ile çeşitli zorluklarında göze alınması gerektiği vurgulanmıştır. Sonuç olarak, işletmelerin IoT teknolojisi altyapılarını, kapsamlı olarak oluşturmuş oldukları stratejileri doğrultusunda, teknolojinin gerekliliklerini ve tüm süreçlerini benimseyerek ve dikkatle yöneterek kurmaları gerektiği öne çıkmıştır. Bu sayede, işletmelerin Endüstri 4.0 süreçlerinde sürdürülebilir bir başarı elde edebildikleri sonucu ortaya çıkmıştır.

Mısır ve Gökrem (2020) tarafından "*Nesnelerin İnterneti için MQTT ile Hiyerarşik Haberleşme*" adlı çalışma, IoT kavramını ele alarak, cihazların internete bağlanması ve birbirleri arasında iletişim kurması gerekliliği kapsamında haberleşme ağlarının gelişimleri değerlendirilmiştir. Çalışmada özellikle TCP/IP (Transmission Control Protocol/Internet Protocol) alt yapısı üzerine gelişen MQTT protokolü incelenmiştir.

MQTT'nin hafif, esnek ve hızlı bir protokol olmasının yanı sıra, yayımlama/abone prensibini kullanarak verileri konu başlıkları altında toplar ve bu konulara abone olan kullanıcılara bilgi erişimi sağladığı vurgulanmaktadır. Bu protokolün, enerji verimliliği sağlaması ve internet üzerinden doğrudan erişilebilir olması gibi yetenekleri sayesinde IoT uygulamalarında önemli bir rol oynadığı çıkarımında bulunulmaktadır. Çalışmanın sonuçları incelendiğinde, MQTT'nin konu tabanlı haberleşme özelliğinin, hiyerarşik sınıflandırma sayesinde farklı algılayıcıların amacına uygun olarak veri iletimini kolaylaştırdığı öne çıkmaktadır. Çeşitli hiyerarşik sınıfların kullanılmasının, nesneler arası iletişimde veri filtreleme ve enerji verimliliği gibi avantajlar sunduğu vurgulanmaktadır. Ayrıca, MQTT'nin internet üzerinden kullanılabilme ve çok sayıda istemciyi destekleme yetenekleri IoT kullanımında öne çıkmasına neden olan özelliklerin başında geldiği sonucunun çıkarılmasına neden olmuştur. CC3200 (Connectivity Controller) ve Node MCU geliştirme kartları kullanılarak gerçekleştirilen uygulamalar neticesinde, MQTT protokolünü kullanarak seviyelere ayrılmış konularda abonelik ve veri iletimini başarılı bir şekilde gerçekleştiği görülmüştür. Bu özelliklerin veri sınıflandırılması ve iletişim cihazlarında enerji ve işlem verimliliği sağlanması açısından önemli olduğu desteklenmektedir.

Özdoğan ve Erdem (2020) tarafından gerçekleştirilen "*Nesnelerin İnterneti için Hibrit Uygulama Katmanı Protokol Tasarımı*" adlı çalışmada IoT alanında kullanılan mesajlaşma protokollerinin iletişim performans ve başarının etkileyen en önemli faktörlerden olduğu vurgulanmıştır. Bu kapsamda çalışma haberleşme protokollerinin incelenerek hibrit bir protokol oluşumunda bir prototip ortaya koyabilmeyi hedeflemiştir. MQTT, XMPP (Extensible Messaging and Presence Protocol), AMQP (Advanced Message Queuing Protocol), DDS (Data Distribution Service) ve CoAP (Constrained Application Protocol) gibi yaygın iletişim protokolleri, farklı iletişim yapılarına sahiptirler. Bu durumun IoT'nin giderek yaygınlaşması ve çeşitli kullanım senaryolarının ortaya çıkması, veri iletişimi için farklı ve yeni yaklaşımların geliştirilmesini bir zorunluluk haline getirdiğine değinilmiştir. Tüm bu durumların çerçevesinde şekillenen çalışmada hem merkezi sunucu aracılığı ile hem de doğrudan iletişim sağlayabilmeyi mümkün kılan UDP (User Datagram Protocol) tabanlı bir hibrit protokol tasarlanmıştır. Geliştirilen bu protokol, geliştirilen gerçek zamanlı uygulamalarda gecikme sürelerinin

azaltılmasına odaklanarak, MQTT protokolü ile kıyaslanmıştır. Protokol tasarımının sahip olduğu mimaride isteğe bağlı iletişim ve güç tüketiminin azaltılması amaçlanmıştır. Ayrıca tasarlanan protokol sayesinde cihazların doğrudan birbirleri ile iletişim kurabilmeleri sayesinde sunuculara olan bağımlılığın ortadan kaldırılması, tek bir hata noktasının oluşmasını engelleyerek ağ geçişi üzerindeki trafik yükünü azaltarak ve dağıtarak darboğaz oluşmasını engellemiştir. Birçok farklı cihaz ve veri tipinin IoT senaryolarında yer alması nedeniyle farklı iletişim protokollerine ihtiyaç duyulmasına ve kullanılacak ortamlara göre uyum sağlayabilecek hibrit bir protokol ihtiyacı varlığını açıkça göstermektedir. Sonuç olarak çalışma, farklı IoT senaryolarını desteklemek üzere tasarlanmış hibrit bir protokolün önemini vurgulamakta ve gelecekteki çalışmalara yönelik bir temel oluşturmayı hedefleyerek bu çalışmalar için önerilerde bulunmuştur. Özellikle, protokolde güvenlik özelliklerinin (şifreleme, veri bütünlüğü, kimlik doğrulama) geliştirilmesinin ve sınırlı kapasiteli IoT cihazlarının ilerleyen teknolojiyle birlikte öngörülen zorluklar ile başa çıkabilen tasarımlar yapılmasının önemli olabileceğine değinilerek bu konuların incelenmesi hakkında önerilerde bulunulmuştur.

Türkel ve Yeşilkuş (2020) tarafından gerçekleştirilen "*Dijital Dönüşüm Paradigması: Endüstri 4.0*" adlı çalışmada Endüstri 4.0 ve bileşenleri ile lojistik, Ar-Ge ve eğitim alanlarında yaygın olarak görülen etkileri incelenmiştir. Endüstri 4.0 ve beraberinde gelişen teknolojiler kullanılarak lojistik alanında süreçsel ve teknik faaliyetlerin entegrasyonunu, firmaların Endüstri 4.0 için Ar-Ge çalışmalarına verdikleri önemi ve tüm bu değişimlerin yanında eğitimi Eğitim 4.0 kavramı üzerinden güçlü ve zayıf yönlerini ortaya koyarak incelemiştir. Neticede Endüstri 4.0 kavramının şirketler üzerindeki etkilerinin neler olabileceği ve nasıl çözümleneceği konularına açıklık getirilmeye çalışılmıştır. Endüstri 4.0 kavramının iş süreçlerinde benimsenmesi gerektiği ve benimsenmediği durumlarda çalışanlarda stres kaynaklı motivasyon bozuklukları yaşanabileceği, Endüstri 4.0 için hazırlanırken geliştirilen ve uygulanan projelerin yanında çalışanlarında psikolojik olarak hazırlanılması gerektiğinden bahsedilmektedir. Teknolojik gelişmeler sonucunda Endüstri 4.0'ın etkilerinin sadece şirketlerin faaliyetlerini kapsamakla kalmadığı, şirketlerin Endüstri 4.0 beraberinde gelen yeniliklere ne kadar açık ve çalışanların değişime ne kadar hazır olduklarının ölçülmesi konularının da önemli olduğu sonucu ortaya çıkmaktadır.

Yıldırım (2020) tarafından gerçekleştirilen "*Farklı Disiplinlerde Endüstri 4.0*" adlı çalışma Endüstri 4.0 kavramının disiplinler arasındaki etkisini ve ilişkisini incelemeyi amaçlamıştır. Çıkış amacı olarak Endüstri 4.0 kavramı üretim ve üretim araçları ile bağdaştırılsa da bu kavramın sadece üretim ile ilgili olmadığını ve hayatımızın bir parçası olduğunu ve hukuk, sağlık, çevre, inşaat, reklam, politika gibi birçok disiplinde kendisine yer bulduğundan bahsedilmiştir. Bu çalışma, insan kaynakları, pazarlama, yalın üretim, işletme yönetimi, finans/yatırım ile gelecek stratejileri açısından ele alınan Endüstri 4.0'ı, etkilerini ve ilişkisini incelemiştir. Bu çalışma ilk olarak insan kaynaklarını ele almıştır ve insan kaynaklarının bu kavramdan en fazla etkilenen disiplin olduğunu öne çıkarmıştır. Endüstri 4.0 kavramı ile gündeme gelen ve gelişen teknolojik gelişmeler, otonom ve robotik sistemleri de beraberinde getirmiş ve karanlık fabrikaların böylece ortaya çıktığı ele alınmaktadır. Bunun neticesinde insan kaynağına olan ihtiyacın azalacağı ve insanların yaptığı işlerin büyük bir kısmını robotların yapabileceği varsayılmıştır. Ancak, teknolojinin gelişmesine rağmen insana olan ihtiyacın hep var olacağını, Endüstri 4.0'ın düşünce yapısına bağlayarak savunmuştur. Çalışmanın savunduğu ve etkilerini incelediği bir diğer disiplin ise pazarlamadır. Endüstri 4.0'ın beraberinde gelişimlerini sağladığı IoT, Büyük Veri gibi teknolojiler kullanılarak müşterilerden toplanan verilerin analiz edilerek ve işlenerek kullanıcılara daha iyi hizmet verebilmek için geliştirilecek olan pazarlama stratejilerinde kullanılabildiğinden yola çıkarak Endüstri 4.0'ın Pazarlama üzerindeki etkileri incelenmiştir. Sonuç olarak bu çalışmada Endüstri 4.0'ın beraberinde bulunan zorluklar ve aynı zamanda uygulayıcılar ve işletmeler için oluşturduğu fırsatlar ve yenilikler ele alınırken; İnsan kaynakları, yalın üretim, pazarlama, finans yatırım gibi farklı disiplinler üzerinde Endüstri 4.0 kavramının etkileri incelenmektedir.

Yiğitol ve Sarı (2020) tarafından gerçekleştirilen "*Küresel Salgınlar ile Mücadelede Endüstri 4.0 Teknolojilerinin Gücü*" adlı çalışmada Endüstri 4.0 teknolojilerinin global pandemi krizinde gerçekleştirilen çalışmaların uygulama alanlarını göstermeyi ve yaygınlaştırılması için temel oluşturmayı amaçlamıştır. Endüstri 4.0 teknolojilerinin endüstriyel üretim ve yönetim faaliyetlerinin yanında küresel çapta büyük etkiler yaratan pandemi ile mücadelede kullanılmasını ele almıştır. Endüstri 4.0 teknolojilerinin sağlık alanında tanı ve tedavi, uzaktan eğitim uygulamaları, e-ticaret ve imalat süreçlerinde pandeminin yaratmış olduğu etkileri minimuma indirdiği ve en az hasar ile bu sürecin

atlatıldığı neticeleri değerlendirilmiştir. Sonuç olarak değerlendirildiğinde çalışma, pandemi sürecinde günlük hayatımızda karşılaştığımız Endüstri 4.0 uygulamalarını açıklayarak bu süreçte hayatımıza katkılarını açıkça ortaya koymuştur. Salgın sürecinin olumsuz etkilerinin yanı sıra, birçok alanda kullanılan Endüstri 4.0 ve teknolojileri sayesinde salgın sürecinde olumlu bir dönüşümün başladığı ifade edilmiştir. Salgın sürecinde kaynak kullanımlarını optimize ederek enerji ve kaynak tasarrufu sağlanabileceği, işgücü akışının değişebileceği ve firmaların gelecekte teknolojik altyapılarını güçlendirmeye yönelebileceklerini öngörmektedir. Bu çalışma özet olarak literatürde salgın sürecinin sektörlerin etkileri üzerine odaklandığını ve sonuç olarak olumsuz sonuçlara sıkılıkla karşılaşıldığı incelenmiş ancak bunların üzerine bu çalışmanın olumsuz sonuçlar üzerine Endüstri 4.0 teknolojilerinin yaratabileceği olumlu etki ele alınmaktadır.

Gönen, Yılmaz ve ark. (2021) tarafından hazırlanan "*Endüstri 4.0'ın Gelişim Sürecinde Unutulan Bileşen: Siber Güvenlik*" adlı çalışmada Endüstri 4.0 ve teknolojilerinin gelişim sürecinde genellikle göz ardı edilen ancak tüm bu gelişim sürecinde önemli bir rolü olan siber güvenlik konusu incelenmiştir. Endüstri 4.0 otomasyon ve dijitalleşme ile veri kaynaklarının kullanımını ve paylaşılmasını artırmıştır. Bu durum bir devrimi ve dönüşümü beraberinde getirmiştir. Ancak avantajlı yanları çok olan bu dönüşüm farklı alan ve durumlarda tehditler oluşturmuştur. Bu tehditlerden bir tanesi ise dönüşen sistemlerin güvenliğidir ve bu durum ilgili çalışmada Endüstri 4.0 çalışmalarında siber güvenliğin önemine değinilerek incelenmiştir. Çalışma Endüstri 4.0 alanındaki avantajları incelemenin yanında siber güvenlik açıklarını da ön plana çıkarmıştır. Üst seviye bağlantılar bulunan ve yüksek düzeyde veri odaklı transferlerin gerçekleştiği sistemlerin, siber saldırılara karşın güvensiz kalabileceği riskini ortaya koymaktadır. Çalışma, Endüstri 4.0 uygulamalarının geliştirilmesi ve tasarlanması sürecinde siber güvenlik önlemlerinin göz önünde bulundurularak projelerin temelinde yer alması gerektiğini, böylece tüm altyapı ve sistemlerin siber saldırılara karşı daha güvenli, dirençli ve hazırlıklı bir ortam sunulabileceğini savunmuştur. Sonuç olarak çalışmada, siber güvenliğin ne derecede önemli olduğu vurgulanmaktadır.

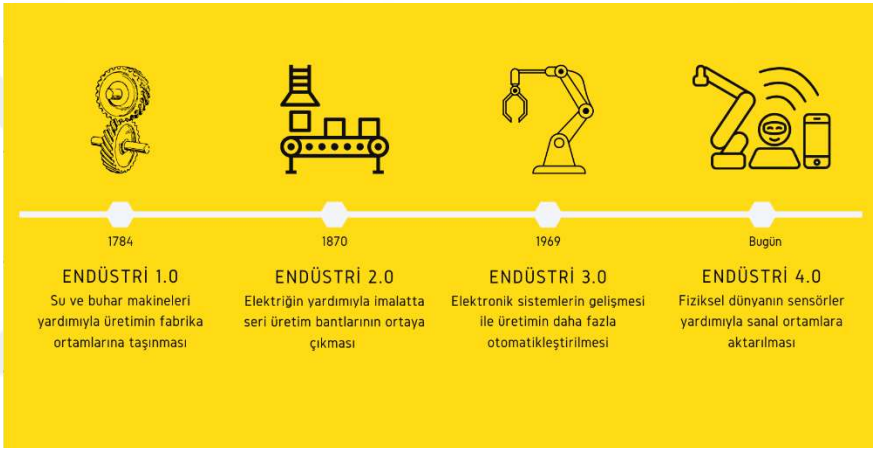
Petekci (2021) tarafından gerçekleştirilen "*Endüstri 4.0: Fırsat mı Tehlike mi?*" adlı çalışma Endüstri 4.0'ın niteliğini ve beraberinde getirdikleri yeniklerin potansiyel avantaj ve dezavantajlarını incelemiştir. Çalışma, Endüstri 4.0'ın önceki endüstri devrimlerinden nasıl ayrıştığına dair analiz yaparken, gelecekte bu kavramın sebep olabileceği olası etkilerini değerlendirmeyi amaçlamıştır. Bu doğrultuda Endüstri 4.0'ın nihai sonuçlarından olması beklenen karanlık üretim kavramı ve bu kavramın olumlu ve olumsuz sonuçları ele alınmıştır. Ancak, çalışmanın gerçekleştiği tarih itibariyle Endüstri 4.0'ın dünya çapında etkilerinin belirsiz olması nedeniyle öngörüsel sonuçlara varıldığı belirtilmiştir. Bulut teknolojileri ve büyük veri kavramlarının hızla hayatımıza entegre olmaya başladığı ve bu entegrasyonun gelecekte gelişmiş robotların üretiminin yanı sıra günlük hayatımızda önemli rol oynayabileceği varsayımında bulunulmuştur. Bu kavram ile ortaya çıkan yeniliklere yönelmeyi hedef edinen girişimciler için çeşitli fırsatlar doğabileceği ve dünya çapında yeni olanaklar yaratma potansiyeli olduğu bahsedilmiştir. Tüketicilerden elde edilen verilerin kullanılmasıyla tüketiciye en doğru önerilerin sunulabilmesi ile rekabet gücünü artırıcı etkilerin var olabileceğine değinilirken, bu verilerin kötü amaçlı kullanıldığında savaşlara neden olabileceğinden de bahsedilmiştir. Endüstri 4.0 ve teknolojilerine doğru yatırımları yapabilen ülkelerin, ciddi düzeyde sağlanabilecek verimlilik ve etkinlik artışı ile küresel çapta pazar payını önemli seviyelere çekme şansına sahip olabileceklerine değinilmiştir. Sonuç olarak karanlık üretim neticesinde, üretim maliyetlerinin azalması, kendi hatalarını tespit eden ve iyileştirebilen sistemlerin ortaya çıkabilmesi, enerji tasarrufu sağlanabilmesi, üretimin 24 saat devam edebilmesi, insan kaynaklı duruş ve hataların azalması, kaliteli ve verimli üretimin artırılması gibi potansiyel avantajların ortaya çıkabileceği vurgulanmıştır. Ancak çalışma, Endüstri 4.0 süreçleri için gereken yatırımların boyutu nedeniyle gelişmiş ülkeler ile gelişmekte olan ülkeler arasındaki farkın açılabilmesi, insan gücünün azalması ile sosyoekonomik nedenlerin sosyal çöküntülere neden olabilmesi, güvenlik açıkları oluşabilmesi ve mesleklerde yaşanabilecek dönüşümler neticesinde birçok alanda reformların gerekmesi gibi potansiyel dezavantajlarının varlığı da gözler önüne serilmektedir.

3. GENEL BİLGİLER

3.1. Endüstri 4.0 ve Teknolojileri

3.1.1. Endüstri devrimlerinin tarihsel gelişimleri ve Endüstri 4.0

İnsanlık tarihindeki en önemli kırılma noktalarından biri de endüstriyel alanda yaşanan devrimlerdir. Bu devrimler, sanayi öncesi toplumları modern endüstriyel toplumlara dönüştürmekte ve toplumsal, ekonomik ve teknolojik alanlarda birçok radikal değişikliklere yol açmaktadır. Şekil 3.1'de endüstri devrimlerinin tarihsel gelişimi gösterilmektedir.



Şekil 3.1. Endüstri devrimlerinin tarihsel gelişimi (Merpazar. Endüstri 4.0 Nedir?Erişim 7.11.2023)

Endüstri alanındaki devrimlerden ilki olan ve Endüstri 1.0 olarak adlandırılan birinci endüstri devrimi, 1784 ile 1840 yılları arasında gerçekleşmiştir. Tarımın modernleşmesi, nüfus artışı, enerji kaynaklarının kullanılmaya başlanması ve bunların beraberinde gerçekleşen teknolojik ilerlemeler birinci endüstri devriminin ortaya çıkmasına neden olmuştur. Buhar makinesi, tekstil endüstrisi içerisinde gelişen makineler, demir üretimindeki gelişmeler gibi teknolojik ilerlemeler, üretim süreçlerinde büyük değişikliklere yol açmakla beraber, kentleşme ve işçi oluşumu gibi toplumsal etkileri de beraberinde getirmiştir. Tüm bunlar üretimin fabrika ortamlarına taşınmasına neden olmuştur ve birinci endüstri devriminin temeli bu gelişmeler eşliğinde atılmıştır.

1870'li yıllara gelindiğinde petrol ve elektrik gibi enerji kaynaklarının daha yaygın kullanılmaya başlanması ile Endüstri 2.0 doğmuştur. Ayrıca, demiryolu ağlarının genişlemesi, telgrafın yaygınlaşması gibi faktörler, ulaşım ve iletişimde büyük

değişimlerin temelini oluşturmuştur. Bunların beraberinde kimya endüstrisi ve çelik üretiminde yaşanan gelişmeler, sanayileşme sürecine hız kazandırmıştır. Sanayileşmenin hızlanması ve enerji kaynaklarının daha yaygın olarak kullanılması imalatta seri üretim bantlarının ortaya çıkmasına neden olmuştur. Bunların neticesinde ikinci endüstri devrimi 1969'lu yıllara kadar sürmüştür.

Endüstri 3.0, bilgi teknolojileri ve mikroçip alanlarında yaşanılan gelişmeler sonucunda ortaya çıkmıştır. Yaşanan bu gelişmeler endüstri alanında otomasyonun yaygınlaşmasına neden olmuştur ve bu durum üretim için yeni bir yol ayrımı oluşturmuştur. Ayrıca, küreselleşme, dünya ekonomisinin küreselleşmesini hızlandırmış, ticaretin ve iletişimin uluslararası düzeyde daha hızlı ve etkili bir şekilde gerçekleşmesine olanak sağlamıştır. Yaşanan bu gelişmelerin sonucunda üçüncü endüstri devrimi doğmuştur ve 2010'lu yıllara kadar etkilerini sürdürmüştür.

2010'lu yıllara gelindiğinde gelişen sensör ve internet teknolojisi ile üretim alanlarında bulunan fiziksel veriler sanal ortamlara aktarılabilmiştir. Bunun beraberinde teknolojinin hızla gelişmesi birçok yeni teknolojik kavramın doğmasına neden olmuştur. Tüm bunlar beraberinde Endüstri 4.0 kavramını şekillendirmiştir. Bu devrim beraberinde gelişen teknolojiler, endüstriyel üretimde önemli değişikliklere yol açmış ve gelecekteki endüstri devrimleri için bir temel oluşturmuştur. Endüstri 4.0 kavramının etkileri günümüzde halen devam etmektedir.

Çalışmanın da üzerine şekillendiği Endüstri 4.0 kavramı 2010'lu yılların başında ortaya çıkarken Avrupa coğrafyasında faaliyet gösteren, sektör devi firmaların, uzak doğu firmaları ile rekabeti artırmak amacı ile ortaya çıkmıştır. Endüstri 4.0 kavramı üretimde hız, esneklik ve verimliliği artırmaya amaçlamaktadır. Bu nedenle Endüstri 4.0 beraberinde, bu kavramın amaçlarına hizmet edebilecek birçok teknolojinin de gelişmesine neden olmuştur. Sonuç olarak endüstri devrimleri, insanlık için kritik dönüm noktalarıdır. Bu devrimlerin tarihsel gelişimi, modern dünyanın şekillenmesinde önemli roller oynamaktadır ve gelecekteki endüstri devrimleri için temel oluşturmaktadır.

3.1.2. Endüstri 4.0 beraberinde gelişen teknolojiler



Şekil 3.2. Endüstri 4.0 teknolojileri (TalentCoders. Endüstri 4.0 Nedir?Erişim 7.11.2023)

Endüstri 4.0 kavramının temelinde fiziksel olarak hayatımızda yer alan bileşenlerin sanal ortamlara aktarılması yer almaktadır. Bu kavram, hayatımızın birçok alanında bulunan veri kaynaklarından toplanan verilerin, sanal ortamlara aktarılmasını ve Endüstri 4.0 kavramı beraberinde gelişen teknolojiler yardımı ile hayatımızın her evresini kolaylaştırabilecek çözümler sunulmasını sağlamaktadır. Endüstri 4.0 kapsamında yapılan çalışmalarda; IoT, büyük veri, otonom (özerk) robotlar, bulut bilişim, eklemeli imalat, artırılmış gerçeklik, simülasyon ve dijital ikiz, siber güvenlik ve sistem entegrasyonu teknolojilerinden sıkça faydalanılmaktadır. (Şekil 3.2) Ancak tüm bu teknolojilerin verimli sonuçlar elde edebilmesi için verilerin sanal ortama taşınması gerekmektedir. Endüstri 4.0 kavramının en önemli teknolojilerinden birisi olarak IoT teknolojisi çalışmanın temelini oluşturmaktadır.

3.2. Endüstri 4.0 Teknolojisi Olarak Nesnelerin İnterneti

3.2.1. IoT teknolojisi

IoT, günlük yaşantımıza entegre olan bir teknoloji olarak ön plana çıkmaktadır. Bu teknoloji, akıllı cihazlar, kullanılan protokoller, uygulamalar, toplanan verilerin entegrasyonu gibi çeşitli adımları içermektedir. IoT teknolojisinin günlük hayatımıza getirdiği ekipmanlar, akıllı cihazlar kategorisinde önemli bir yer tutmaktadır. Bu cihazlar arasında akıllı termostatlar, akıllı lambalar, akıllı ev güvenlik sistemleri, giyilebilir teknolojiler ve akıllı tarım ekipmanları gibi birçok örnek bulunmaktadır. Bu ekipmanlar, kullanıcılara daha akıllı ve bağlantılı bir yaşam tarzı sunmaktadır.

IoT Uygulamalarında kullanılan protokoller, cihazlar arasındaki iletişimi sağlamak için önemlidir. MQTT, CoAP ve HTTP gibi protokoller, veri iletiminde ve cihazlar arası etkileşimde kullanılmaktadır. Bu protokoller, güvenilir ve hızlı iletişim sağlayarak IoT ekosistemini desteklemektedir.

IoT Uygulamaları, çeşitli sektörlerde kullanılan ve günlük hayatımızı kolaylaştıran uygulamalardan oluşur. Akıllı şehir uygulamaları, sağlık hizmetlerinde kullanılan takip sistemleri, enerji yönetimi uygulamaları ve perakende sektöründeki akıllı etiketleme uygulamaları bu kategoride yer almaktadır. Bu uygulamalar, verimliliği artırarak ve kaynakları daha etkili kullanarak yaşam kalitesini iyileştirmektedir.

IoT, büyük miktarda veri üretmektedir. Bu veriler, kullanıcılara özelleştirilmiş deneyimler sunmak, işletmeler için verimliliği artırmak ve sistemler arası entegrasyonu sağlamak amacıyla kullanılmaktadır. Veri analitiği, yapay zekâ ve bulut bilişim gibi teknolojilerle entegre edilen bu veriler, akıllı kararlar alınmasını sağlamaktadır ve IoT ekosistemini daha güçlü kılmaktadır.

3.2.2. IIoT teknolojisi

IIoT, endüstriyel süreçlerin optimize edilmesi ve verimliliğin artırılması için tasarlanmış bir teknoloji alanını temsil etmektedir. Bu kavram, endüstriyel ekipmanların akıllı sensörlerle donatılması, veri analitiği ve otomasyon gibi teknolojilerin kullanılmasıyla şekillenmektedir.

IIoT teknolojisinin temelini, endüstriyel süreçlerde kullanılan akıllı ekipmanlar oluşturmaktadır. Bu ekipmanlar arasında sensörlü makineler, akıllı üretim hatları, otomatik veri toplama sistemleri ve endüstriyel robotlar gibi örnekler bulunmaktadır. Bu ekipmanlar, üretim süreçlerini izlemek, veri toplamak ve daha verimli çalışmak için tasarlanmaktadırlar. IIoT Uygulamalarında kullanılan protokoller, endüstriyel ekipmanların birbirleriyle ve merkezi kontrol sistemleriyle iletişimini sağlamak amacıyla kritik öneme sahiptir. OPC UA, MTConnect, MODBUS gibi protokoller, endüstriyel veri iletiminde ve entegrasyonunda kullanılmaktadır. IIoT Uygulamaları, endüstriyel sektörde verimliliği artırmak ve sürdürülebilirliği desteklemek için kullanılmaktadır. Akıllı üretim, öngörücü bakım, tedarik zinciri yönetimi ve enerji yönetimi gibi uygulamalar, IIoT'nin

endüstriyel süreçlere getirdiği yenilikçi çözümler arasında yer almaktadır. IIoT, endüstriyel tesislerde büyük miktarda veri üretmektedir. Bu veriler, üretim süreçlerinin optimize edilmesi, bakımın planlanması ve kalite kontrolünün iyileştirilmesi gibi alanlarda kullanılmaktadır. Veri entegrasyonu, endüstriyel sistemlerin daha akıllı ve esnek hale gelmesine yardımcı olmaktadır.

3.2.3. IoT ve IIoT farkları

Kapsam ve uygulama alanları kaynaklı farklar:

- IoT, genellikle günlük yaşamda ve tüketici odaklı kullanımlarda karşımıza çıkmaktadır. Evdeki akıllı termostatlar, giyilebilir teknolojiler, akıllı buzdolapları gibi cihazlar IoT'nin örnekleri arasındadır. Bu teknoloji, geniş bir kullanım yelpazesi sunarak evlerden şehirlere, sağlık hizmetlerinden ulaşıma kadar birçok alanda kullanılabilmektedir.
- IIoT ise endüstriyel süreçleri optimize etmek ve endüstriyel sistemlerde verimliliği artırmak amacıyla tasarlanmıştır. Akıllı fabrikalar, otomasyon sistemleri, enerji yönetimi gibi endüstriyel uygulamalarda IIoT teknolojisi kullanılmaktadır.

Güvenlik ve dayanıklılık kaynaklı farklar:

- IoT cihazları genellikle tüketicilere yönelik olduğu için güvenlik ve dayanıklılık önemlidir ancak endüstriyel sistemler kadar sıkı standartlara tabi değildirler.
- IIoT, endüstriyel sistemlerde kullanıldığı için daha sıkı güvenlik ve dayanıklılık standartlarına tabidir. Endüstriyel süreçlerde güvenlik açıklarının veya hataların ciddi sonuçlara yol açabileceği düşünüldüğünden, IIoT cihazları genellikle daha yüksek güvenlik standartlarına sahiptir.

Veri işleme ve analiz kaynaklı farklar:

- IoT genellikle büyük miktarda veri üretir ancak bu veriler genellikle bireysel cihazlarla ilgili bilgileri içermektedir. Veri işleme genellikle bulut tabanlı hizmetlerle gerçekleştirilmektedir.
- IIoT, endüstriyel süreçlerden elde edilen büyük veri setlerini işlemektedir. Veri analitiği, üretim süreçlerini optimize etmek, bakım faaliyetlerini planlamak ve genel sistem performansını iyileştirmek için kullanılmaktadır. Bu nedenle, IIoT genellikle daha karmaşık ve büyük ölçekli veri analitiği uygulamalarını içermektedir.

3.3. Üretimde Kullanılan Tezgahlar, Çeşitleri ve Yapıları

Çalışma kapsamında değerlendirilen ekipmanlar, çeşitleri ve yapıları bakımından farklılıklar oluşturmaktadır. Bu farklılıklar neticesinde ekipmanlardan elde edilen veriler karmaşık yapılara sahip olmaktadır. Bu nedenle ekipmanların gruplandırılması gerekmektedir. Ancak bu gruplandırma, ekipmanların kabiliyet ve yeteneklerine göre değil, bu kabiliyetlerin temelinde yatan teknik özelliklere göre gerçekleştirilmektedir. Bu gruplandırma, ekipmanın kontrolcü türüne bağlı olarak değişkenlik gösterecek olan veri protokollerinin belirlenebilmesi amacı ile gerçekleştirilmektedir. Bu nedenle ekipmanlar IIoT kapsamında incelenirken, ekipmanların ne iş yaptıklarından daha çok, bu kabiliyetleri yerine getirirken kullanmış oldukları kontrolcünün türü önemli olmaktadır. Bu nedenle kullanılan kontrolcü yapısına göre ekipmanlar;

1. Programlanabilir Lojik Kontrolcülere (PLC) sahip Konvansiyonel Ekipmanlar
2. Bilgisayar Destekli Nümerik Kontrolcü (CNC) veya Nümerik Kontrolcülere (NC) sahip CNC/NC Ekipmanlar
3. Kontrolcüye Sahip Olmayan Röle Destekli Manuel Ekipmanlar

olmak üzere üç ana başlıkta gruplandırılmaktadır. Bu başlıklar aşağıda açıklanmaktadır.

3.3.1. Konvansiyonel ekipmanlar

Konvansiyonel ekipmanlar, yarı otomatik ya da otomatik olarak çalışan, CNC ya da NC ile çalışan ekipmanlara göre çok daha az karmaşık bir yapıya sahip olan ekipmanlardır. Bu ekipmanlar, fırın, otoklav, torna tezgâhları olabilmektedirler. Genellikle bu tip ekipmanlarda kontrolcü olarak PLC'ler bulunmaktadır. Bu çalışma kapsamında genelleme yapılabilmesi adına PLC bulunan ve üretim faaliyetlerinde aktif rol oynayan, tezgâh olarak nitelendirilebilecek olan ekipmanlar genel olarak konvansiyonel ekipmanlar olarak adlandırılmaktadır. Şekil 3.3'te konvansiyonel tezgahlara örnek verilmiştir.



Şekil 3.3. Konvansiyonel silindirik taşlama tezgâhı (FormCNC. S30 Konvansiyonel Silindirik Taşlama Tezgahı Erişim 8.11.2023)

Konvansiyonel ekipmanlardan veri almanın genel olarak iki farklı yolu vardır. Bunlardan birincisi ve en kolay, az maliyetli ve en verimli veri toplama yöntemi ekipmanların kontrolcüsü olan PLC'lerin programlandıktan sonra verilerin MODBUS ya da TCP/IP gibi protokoller ile yayınlanmasıdır. Bu yöntem ile veri toplayabilmek için ekipmanlar üzerinden yayınlanan verilerin, adreslendikleri veri blok adreslerinin biliniyor ve bahsi geçen protokoller ile yayınlanıyor olması gerekmektedir. Bu nedenle bu yöntem ile veri toplayabilmek için ekipman üreticisi firmanın, ekipmanın devreye alınması sırasında kontrolcü üzerinde uygun yapılanmayı kurması ve teslim sırasında ilgili veri blok adreslerini ilgili firma yetkilileri ile paylaşması ve MODBUS yayınının sağlanması gerekmektedir. Bu sayede bilinen veri blok adresleri aracılığı ile PLC üzerinde işlenen tüm verilerin toplanması sağlanabilmektedir. Daha sonrasında bu yöntem ile veri toplanmak istenirse bu durum çok zor ve zaman alan bir çalışma gerektirmektedir. Inputların giriş adreslerinin tespit edilmesi için elektrik plana, PLC program yedeklerine ve anlık PLC üzerinde koşan programa ihtiyaç duyulabilmektedir. PLC üzerinden o an çalışan program çekilmek istendiğinde firmanın bilgi birikimini korumak amacıyla şifreleme yapmış olduğu durumlar ile karşılaşılabilmektedir. Bu nedenle eğer ilgili verilerin veri blok adresleri bilinmiyor ya da yayın olarak veriler sağlanmıyorsa sonrasında bu yöntemi kullanmak çok sağlıklı ve kolay olmamaktadır. Ekipmana bahsedilen müdahaleler sonucunda firma garantisinden çıkması, tezgâh programının kitlenmesi ya da haberleşmeyi bozan herhangi bir değişiklik nedeniyle eskisi gibi çalışmaması gibi sorunlar yaşanabilmektedir ve bir inputun adresinin tespit edilmesi bile durumuna göre aylar süren çalışmalar gerektirebilmektedir. Bu nedenle veri blok

adresleri bilinmeyen tezgahlardan veri alınmak istenirse, istenilen verilere bağlı olmak koşulu ile ekipmana eklenecek ek donanımlar ile veriler kuru kontak yöntemleri ile alınabilmektedir. Ekipmanda belli verilerin kaynak noktaları tespit edilerek bu noktalardan paralel bağlantılar çekilerek yeni eklenecek donanıma (PLC, 3. Parti veri kaydediciler vb.) dahil edilerek veriler okunabilmektedir. Bu çalışma maliyet açısından yüksek bir çalışma olmasının yanında ekipmana ait tüm verilere erişebilme şansı olmamaktadır. Bu yöntem ile ekipmana eklenen donanımlar sayesinde veriler MODBUS ya da alternatif protokoller (Siemens TCP/IP vb.) ile adreslenerek yayınlanabilmektedirler.

3.3.2. CNC/NC ekipmanlar



Şekil 3.4. CNC torna tezgâhı (Etasis. CNC Torna Tezgahı Erişim 8.11.2023)

Bu gruba dahil ekipmanlar PLC ile NC farkından kaynaklı, farklı yetkinliklere sahip ekipmanları içermektedir. Kabaca PLC'ler lojikler kullanılarak programlanırken, NC'ler nümerikleri kullanarak programlanmaktadırlar. Bu nedenle kabiliyetleri bir PLC'ye oranla daha fazla olmaktadır. Ancak kullanım alanına ve ihtiyaca bağlı olarak değişmek ile NC'ler beraberinde CNC sistemleri de getirmişlerdir. Kabiliyet, hesaplama yetenekleri, hassasiyet ve esneklik gibi durumlarda daha kompleks uygulamalara çözüm olabilen CNC'ler sıkça karşımıza çıkmaktadır. NC kontrolcülere sahip ekipmanlar genellikle frezeleme, tornalama, kesme ve taşlama gibi işlemleri yerine getirebilmek için özelleşmiş ekipmanlardır. CNC kontrolcülere sahip tezgahlar ise birden fazla kabiliyete sahip ve daha kompleks işlemleri yerlerine getirebilen ekipmanlardır. Bu ekipmanlar genellikle birden fazla işlemi yapan ve birkaç eksenli ekipmanlardır. (Şekil 3.4.)

Bu ekipmanlardan IoT mimarisine dahil edilerek veri toplanmak istenildiğinde gruplandırma için ekipmanın kabiliyetlerinden çok sahip olduğu kontrolcünün markası, modeli ve türü ile gruplandırılmaktadır. Farklı marka ya da modelde kontrolcüye sahip

ekipmanlar farklı protokoller ile veri yayını sağladıkları için doğru protokolün ve yöntemin tespit edilebilmesi, buna bağlı uygulanacak adımların belirlenebilmesi için veri toplanması istenilen ekipmanların bu kriterler doğrultusunda gruplandırılması daha doğru olmaktadır. Bu noktada kontrolcü markalarından sık karşılaşılması mümkün markalar, Siemens, Fanuc ve Mitsubishi olmaktadır ve bu markalar ile ilgili gruplandırmalar aşağıda yapılmaktadır. Tüm bunlardan bağımsız olarak ilgili marka ve modele ait ekipmanların herhangi bir veri yayınlama protokolüne destek sunmaları oldukça önemlidir. Bu protokollere destek sunamayan kontrolcülere sahip ekipmanlar, CNC ya da NC kontrolcülere sahip olsalar dahi veri toplama aşamalarında konvansiyonel tezgahlar için kullanılan veri toplama yöntemleri kullanılmaktadır. Açıklanan bu durum marka ve model bazlı gruplandırma yapılmasına neden olmaktadır.

3.3.2.1. Siemens kontrolcülü tezgahlar



Şekil 3.5. Siemens kontrolcülü CNC tezgâh (DMGMori. DMU 40 Erişim 8.11.2023)

Siemens kontrolcüye sahip tezgahlar diğer marka kontrolcüye sahip tezgahlara nazaran daha yaygın bir kullanıma sahiptir. Bu nedenle incelendiğinde çok farklı model, versiyon ve konsepte sahip ekipmanlar ile karşılaşılabilmektedir. Farklı tip tezgahlara örnek Şekil 3.5'te verilmiştir. Siemens kontrolöre sahip ekipmanlardan tek bir yöntem ile veri alınabileceğini söylemek çok doğru olmayacaktır. Veri toplanması istenilen ekipman Siemens kontrolcülü bir tezgâh ise birden fazla bağlantı yöntemine sahiptir. Siemens için yaygın olarak kullanılan protokol genellikle OPC UA olmaktadır. Ancak bu protokol ekipman üzerinde bulunan kontrolcünün her versiyon ve modeline destek vermemektedir. Bu nedenle Siemens kontrolcüye sahip bir tezgâh olsa bile OPC UA ile kesin olarak veri almak mümkün olmamaktadır. Bu nedenle Siemens kontrolcülü tezgahlarda kendi içerisinde gruplandırılmaktadır.

Siemens kontrolcülü tezgahlarda Siemens tarafından Numerical Controller Unit (NCU) olarak adlandırılmış CNC kontrol üniteleri bulunmaktadır. İlk olarak bu ünitenin modeli tespit edilmelidir. Tezgâh panosu ve menüsü üzerinden ilgili kısımlar kontrol edilerek tezgâh kontrolcü modeline ulaşılır. Bu modeller arasında sık karşılaşılan modeller Sinümerik 840D Solution Line (SL), Sinümerik 840 D Power Line (PL) olmaktadır. Farklı modeller var ise onlarda ayrıca incelenmelidir. İlk olarak tezgahlar bu NCU modellerine göre tekrar sınıflandırılır. Tezgâh Sinümerik 840D PL bir NCU'ya sahip ise bu tezgâh OPC UA desteği sunamayacaktır. İkinci bir sınıflandırma ise Sinümerik 840D SL bir NCU'ya sahip tezgahlar arasında olmalıdır. Bu NCU'ya sahip tezgahların Sinümerik Operate versiyonları OPC UA desteği açısından önemlidir. Tezgahların ilgili menü sayfası üzerinden versiyonlar kontrol edildiğinde V4.5 SP2 (Version 4.5 Service Pack 2) ve üzeri bir versiyona sahip olduğu tespit edilen tezgahlar OPC UA desteği sunmaktadır. Ancak diğer versiyonlar PL tezgahlar ile aynı sınıflandırmaya sahip olacaktır. Sonuç olarak Siemens tezgahlar kendi içlerinde iki gruba ayrılmış olacaklardır. Bu sınıflandırma OPC UA desteği sunan ve sunmayan tezgahlar şeklinde olacaktır. Şekil 3.6'da Siemens marka 840D SL ürün ailesi görülmektedir.



Şekil 3.6. Siemens 840D SL ürün ailesi (Siemens. Sinümerik 840D SL Catalog Erişim 8.11.2023)

OPC UA desteği sunmayan tezgahlara ise Siemens'in bir çözümü olan BFC Gateway ya da diğer firmaların alternatif MTConnect tabanlı çözümleri gibi çözümler ile veri yayınlama kabiliyetine sahip olabilirler. Ancak bu çalışmalar incelendiğinde en sağlıklı yöntemin BFC Gateway çözümü olduğu görülmüştür. Bu çözümler anahtar teslim çözümler olarak sunulmakta olup OPC UA 'ya göre daha kısıtlı bir çözüm sunmaktadırlar. Bu nedenle ilk tercih hem maliyet hem kabiliyet açısından değerlendirildiğinde OPC UA olmaktadır. Ancak OPC UA desteği sunmayan Siemens marka kontrolcüye sahip tezgahlarda BFC Gateway çözümü kullanılabilmektedir.

3.3.2.2. Mitsubishi/Mazak kontrolcülü tezgahlar

Mitsubishi ve Mazak kontrolcülü tezgahlar MTConnect protokolünü desteklemektedirler. Bu marka kontrolcülere sahip Ethernet bağlantısı kurulabilen tezgahlardan MTConnect ile veri toplanması mümkündür. Bu çalışmanın yapılabilmesi için makine dilini algılayarak anlamlı ve standart bir yapıda verinin yayınlanmasını sağlayan bir adaptöre ihtiyaç duyulmaktadır. Adaptörler çeşitli firmalar tarafından geliştirilmekte olup yeni nesil tezgahlara opsiyon olarak satın alımda ekletilebilmektedir. Bu aşamada satın alınan adaptörün en uygun verileri barındırması önemlidir. Bu noktada adaptörün satın alındığı firmanın bilgisi ve yeterliliği önemlidir ve satın alınan adaptör sağlıklı olmaz ise hem sağlıklı veri almak hem ihtiyaç duyulan tüm verilerin tezgâhtan toplanması mümkün olmayacaktır.



Şekil 3.7. Mazak kontrolcülü CNC tezgâh (Mazak. Variaxis i-500 Erişim 8.11.2023)

MTConnect protokolü farklı markaların kontrolcülerini içeren tezgahlara da entegre edilebilmektedir. Ancak hem kurulumda yaşanan zorluklar hem veri çeşitliliği gibi nedenler ile tercih edilmemektedir. Kontrolcüden bağımsız MTConnect desteği sunan tezgahlar ayrı bir grup olarak değerlendirilebilmektedir. Ancak incelendiğinde bu protokolün ilgili adaptörlerinin piyasadaki yaygınlığı, sunduğu veri çeşitliliği ve kalitesinin en verimli olduğu kontrolcülerin Mitsubishi ve Mazak markalarının kontrolcüleri olduğu tespit edilmiştir. Şekil 3.7'de örnek bir Mazak marka tezgah gösterilmektedir.

3.3.2.3. Fanuc kontrolcülü tezgahlar



Şekil 3.8. Fanuc kontrolcülü CNC tezgâh (Tandem. Fanuc Robodrill CNC Dik İşleme Merkezi Erişim 8.11.2023)

Fanuc kontrolcülü tezgahların veri yayınlama kabiliyeti açısından en esnek tezgahlar oldukları söylenebilmektedir. Fanuc kontrolcü serilerinin tamamına yakını FOCAS ile haberleşme yeteneğine sahiptir. Bu haberleşme özelinde geliştirilmiş olan FOCAS kütüphanesi aracılığı ile geliştirilen uygulamalar ya da hazır ara yazılımlara kütüphane eklenerek uygun driver aracılığı ile veri toplanabilmesi sağlanmaktadır. Bu haberleşmenin sağlanabilmesi için kritik olan durum tezgâhın çok eski olmaması ve Ethernet bağlantısına sahip olmasıdır. Tezgâhın bir Ethernet bağlantısı ile fabrika ağına bağlanması ve tezgâh üzerinden ilgili ağ ayarlarının yapılması ile veri iletişimi sağlanmaktadır. Şekil 3.8'de örnek bir Fanuc kontrolcülü CNC tezgah gösterilmektedir.

3.3.2.4. Diğer tezgahlar

Tezgahlardan beklenen kabiliyetler, yaptıkları iş ve sektöre göre değişiklik göstermesi mümkün olmasına rağmen genellikle nadiren karşılaşılan kontrolör markaları bu başlık altında incelenmektedir. Bu tezgahlar Heindenheim, ABB gibi kontrolcülere sahip olabilirler. Bu tip veri protokol desteği sunmayan markaların kontrolcülerine sahip tezgahlar veri yayınlama kabiliyeti ve bu kabiliyetin kazandırılabilmesi açısından çok daha zahmetli ve bu duruma daha kapalı tezgahlar olmaktadırlar. Daha önce bahsedildiği gibi bu kontrolcülere sahip tezgahlardan uygun sürücüler ve data blok adresleri ile veri okunabilmesi mümkün olabilmektedir. Ancak bu yöntem data blok adreslerine zor erişilmesi gibi nedenler ile uygulanabilirlik açısından çok mümkün olmamaktadır. Bu nedenle genelde bu kontrolcüler özelinde çalışarak MTConnect adaptörü geliştirmiş

firmalardan ya da BFC Gateway gibi çözümlerden faydalanarak veri toplanması mümkün olabilmektedir. Fakat bu yöntemler kullanılarak veri toplayabilmek tezgâh üzerinde yapılacak uzun Ar-Ge çalışmaları gerektirmesi nedeni ile bu yöntemleri geliştiren firmalara sık rastlanmamaktadır. Bahsedilen durumlar neticesinde eğer uygun bir protokol desteği bulunamaz ise ilgili markalara sahip tezgahlar konvansiyonel ekipmanlar için geliştirilen yöntemler kullanılarak veri toplanması sağlanmaktadır.

3.3.3. Manuel ekipmanlar



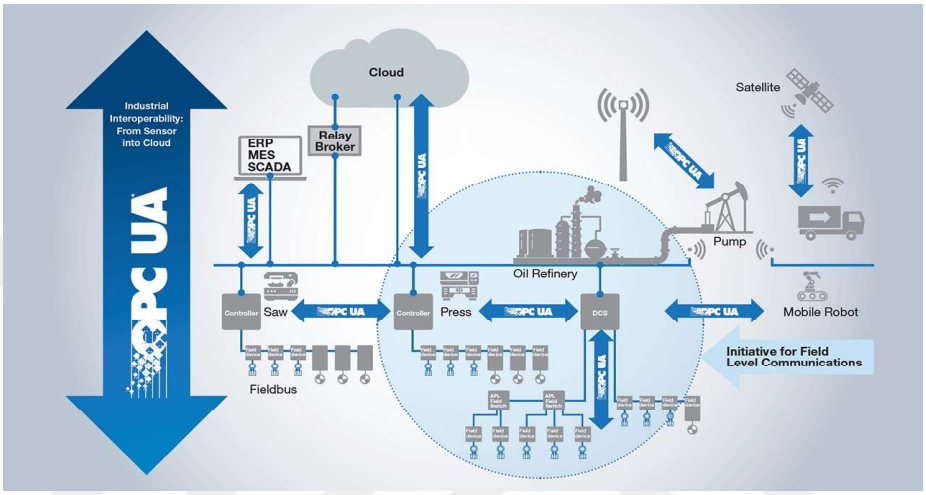
Şekil 3.9. Yarı otomatik freze tezgâhı (CombaKalıp. Konvansiyonel Tezgahlar Erişim 8.11.2023)

Herhangi bir kontrolcü bulunmayan, kontaklar ile ya da basit gömülü kart devreleri ile kontrol edilmesi mümkün, yarı otomatik çalışan ekipmanlar bu kapsama dahil olmaktadır. Bu ekipmanlara ek bir donanım eklenerek veri alınması mümkün olabilmektedir. Kontak ya da sürücü gibi sistemler bulundurmayan ekipmanlardan ise uygun sensörler eklenerek veri alınabilmektedir. Ancak bu tezgahlar sektörel farklılık göstermekle birlikte genelde üretim için takip edilmesi ve veri toplanması çok kritik tezgahlar olmamaktadırlar. Bu nedenle bu tezgahlardan veri toplanması öncelikli olmamaktadır. Çalışmamız kapsamında benzer tezgahlar üzerine yerleştirilmiş olan enerji analizörlerinden veri alınarak tezgâh ile ilgili anlamlı veriler elde edilmiştir. Şekil 3.9'da örnek bir yarı otomatik freze tezgahı gösterilmektedir.

3.4. Nesnelerin İnterneti Entegrasyonunda Kullanılan Protokol ve Yazılımlar

Bu başlık altında, gruplandırma yapılmış olan ekipman marka modellerine göre kullanılan protokoller ve bu protokoller aracılığı ile IoT entegrasyonunun sağlanması için üst katmanlarda kullanılan yazılımlar anlatılmaktadır.

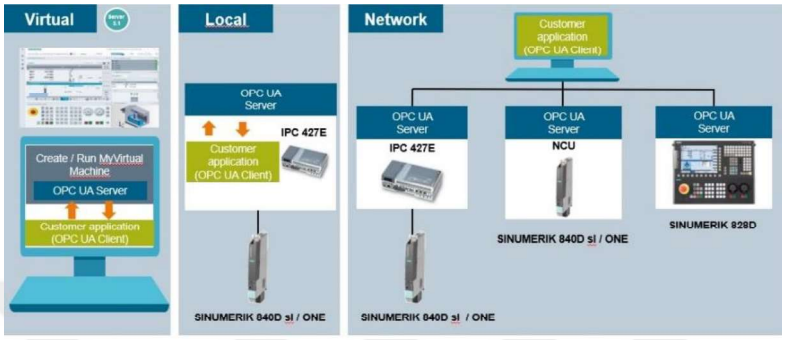
3.4.1. OPC UA protokolü



Şekil 3.10. OPC UA mimarisi (Geelen, Medium. Industry 4.0: is the complex interconnected web of devices resolvable with standardization?Erişim 8.11.2023)

OPC UA, endüstriyel otomasyon ve haberleşme dünyasında önemli bir rol oynamaktadır. Bu standardın en temel özelliği çift yönlü veri akışını desteklemesi ve açık kaynaklı olmasıdır. Özellikle makineler içindeki cihazlarda, makineler arasında ve makineler ile IT (Information Technology) ve OT (Operational Technology)'yi bir araya getiren sistemler arasındaki iletişim için geçerlidir (GSL.OPC-UA nedir?Erişim 16.10.2023). OPC UA'nın herhangi bir yazılım ile entegre çalışabilir olması ve bu nedenle ölçeklenebilirliğinin artırılması en büyük avantajlarından biri olarak gösterilebilmektedir. Esnek yapısı ile farklı cihazlarda sorunsuz çalışabilmektedir ve beraberinde güvenlik protokollerini de barındırmaktadır. Kapalı ağlarda ve internete açık ortamlarda güvenli bir şekilde veri akışını sağlayabiliyor olması endüstriyel uygulamalarda geniş kapsamlı bir kabul görmesine olanak tanımaktadır. OPC UA'nın merkezileşme, platform bağımsızlığı, ölçeklenebilirlik ve ortak çalışabilirlik gibi sağlamış olduğu avantajlar, endüstrinin çeşitli sektörlerinde önemli bir yer bulmasını sağlamaktadır. Bu nedenle, IIoT ve Endüstri 4.0'a geçiş süreçlerinde firmalara önemli bir destek sunmaktadır. Merkezi bir

yapı kurulmasını sağlayarak ekipmanları üst katmanlara bağlanma yeteneği OPC UA'nın ölçeklenebilirliğini göstermektedir. Ayrıca, tak-çalıştır özelliği sayesinde ağlarda kolayca keşfedilebilir. Ancak bu özellik gerekli altyapının sağlanması ve ekipmanların uygun şekilde kurgulanmış olmasına bağlı olmaktadır.

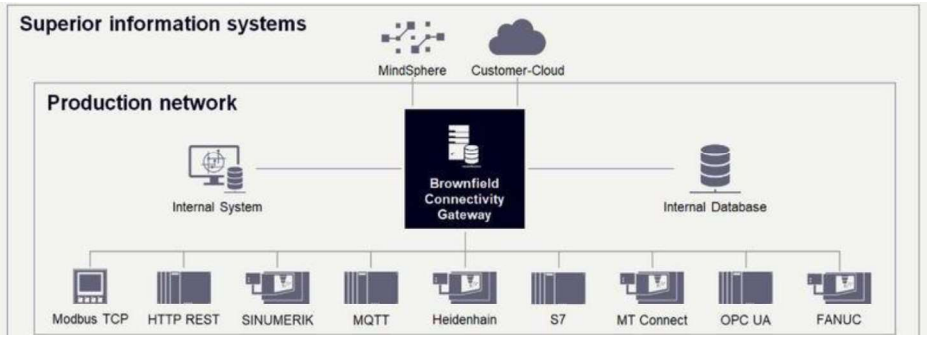


Şekil 3.11. Siemens OPC UA mimari örneği (Siemens. Access MyMachine /OPC UA 828/840 - WLM Erişim 8.11.2023)

OPC UA Endüstri 4.0'ın temel ihtiyacı olan veriye, düşük seviyeli cihazlardan kolayca erişim sağlayabilme özelliği sayesinde firmaların dönüşüm süreçlerinde önemli roller oynamaktadır. Bir işletmenin geniş bir ekipman yelpazesine entegre olabilme yeteneği ile, üretim alanlarından elde edilen büyük verinin aktarılması, depolanması ve çift yönlü akışı sayesinde verilerin analiz edilmesi olanaklarını sağlamaktadır. Tüm bu özellikler neticesinde OPC UA kurumsal uygulamalarda gelişmiş analiz ve görselleştirme çalışmalarının temelini oluşturan verilerin elde edilmesine katkı sağlayarak, endüstriyel süreçlerin daha etkili bir şekilde yönetilmesine olanak sağlamaktadır.

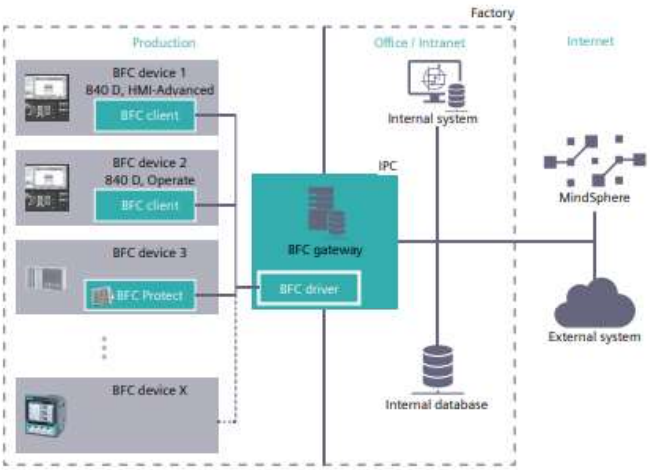
3.4.2. BFC Gateway yazılımı

BFC Gateway, Siemens firması tarafından geliştirilmiş olan geniş bir endüstriyel otomasyon ürün yelpazesine bağlanabilirlik konusunda çözümler sunan bir ara katman yazılımıdır. BFC genellikle endüstriyel tesislerin mevcut bulunan ve daha eski ekipmanlarının modernizasyonu ve dijitalleşmesi süreçlerini ele almaktadır. Bu bağlamda geliştirilen bir ağ geçidi ile farklı endüstriyel cihazların, protokollerin veya sistemlerin haberleşmesini sağlayan bir ara yazılımdır. BFC Gateway birçok farklı marka ve model ekipmanın veri yayınlama protokolleri aracılığı ile verilere erişerek verilerin üst katmanlara aktarılmasını sağlamaktadır. Daha eski Siemens marka kontrolcülere sahip tezgahlarda tercih sebebi olmasının yanında farklı protokoller için de destek sunmaktadır.



Şekil 3.12. BFC Gateway mimarisi ve desteklediği protokoller (Siemens, 2021:76)

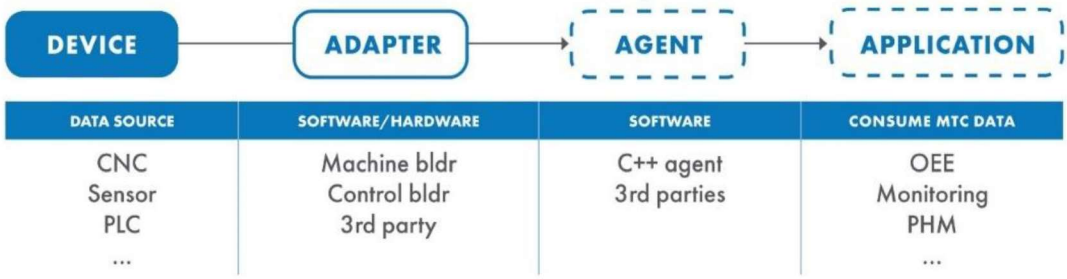
BFC Gateway ekipmanlar üzerine kurulan istemci yazılımlar ile veri yayınlama kabiliyeti bulunmayan kontrolcülerin bu kabiliyeti kazanmasını sağlamaktadır. Her bir ekipman için istemci yazılımı özelleştirilerek ekipman üzerine kurulmaktadır ve bu durum kurulum süreçlerini etkilemektedir. Bu nedenle alternatif ve kolay yöntemler ile veri toplanması mümkün olmayan ve BFC Gateway 'in desteklediği kontrolör versiyonları içerisinde bulunan eski tip ekipmanlardan veri toplamak için kullanılması daha uygun olmaktadır. Ancak anahtar teslimi bir çözüm isteyen firmalar desteklenen tüm ekipmanlarını BFC Gateway kullanarak ağlarına entegre edebilir ve topladıkları verileri OPC UA ile yayınlayabilir ya da doğrudan veri tabanlarına ya da bulut sistemlerine aktarabilmektedirler. Bu çalışmada BFC Gateway, OPC UA ve diğer protokoller için destek sunmayan kontrolörlere sahip olan ekipmanlardan verilerin toplanması ve OPC UA protokolü ile üst katmanlara aktarılması için kullanılmaktadır. Şekil 3.12 ve 3.13'te BFC Gateway yazılımının mimarisi ve desteklediği protokoller gösterilmektedir.



Şekil 3.13. BFC Gateway mimarisi (Siemens, 2022:13)

3.4.3. MTConnect protokolü

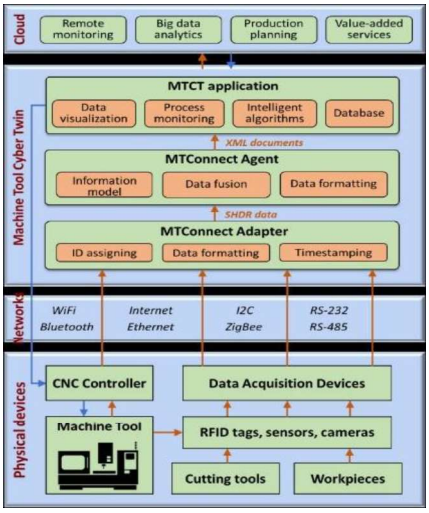
MTConnect, nümerik kontrol sistemlerinden üretim verilerini yayınlamak ve endüstriyel üretim alanlarında gerçek zamanlı veri toplamak amacı ile kullanılan bir protokoldür. Açık kaynaklı bir protokol olan MTConnect, XML (eXtensible Markup Language) ve HTTP teknolojilerine dayanmaktadır. MTConnect, üretim süreçlerini daha izlenebilir ve şeffaf hale getirmeyi, verileri standartlaştırmayı ve bu verileri üst katmanlara aktarmak için anlaşılabilir bir halde yayınlamayı amaçlamaktadır. (Şekil 3.14)



Şekil 3.14. MTConnect (IMTS. Digital Manufacturing Automation Using a Standards-Based Approach: An MTConnect Use Case Erişim 8.11.2023)

MTConnect protokolünün özellikleri üç ana bileşen olarak incelenebilir. Bu bileşenler;

- ✓ XML şemaları ve veri yapıları: MTConnect, XML şemaları kullanarak nümerik kontrol sistemlerinden gelen verileri yorumlamaktadır. Bu sayede, veri yapıları hakkında bilgi edinmekte ve bu verilerin anlamlı bir biçimde nasıl temsil edileceğini belirlemektedir.
- ✓ Tezgâh bileşenleri ve veri anlamları: MTConnect protokolü ile desteklenen tezgâh bileşenlerini ve bu bileşenlerden gelen verilerin anlamlarını içermektedir. Bu özelliği sayesinde farklı tip ve özellikte tezgahlardan elde edilecek verilerin standartlaşmasını sağlamaktadır.
- ✓ Veri akış organizasyonu: Üretim ekipmanlarından toplanan verilerin nasıl organize edileceğini belirlemektedir. MTConnect mimarisi ve verilerin akış diyagramları Şekil 3.15'te görülmektedir. Bu sayede, MTConnect protokolü veri akışının düzenli, gerçek zamanlı ve anlaşılır bir yapıda yayınlanmasını ve toplanmasını sağlamaktadır.



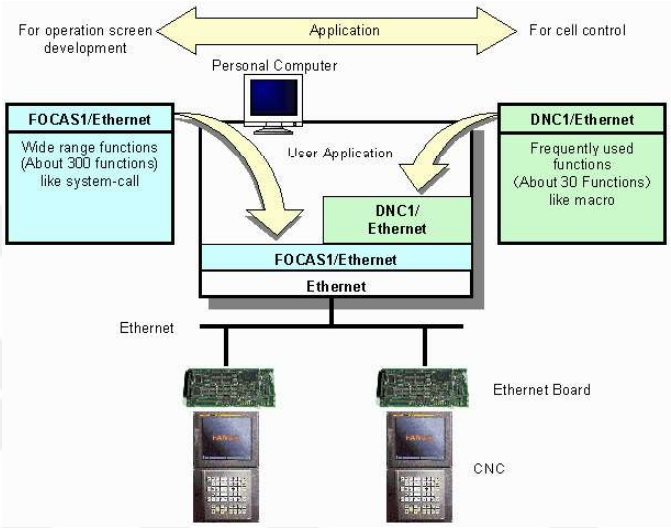
Şekil 3.15. MTConnect mimarisi ve diyagramı (Liu ve ark., 2018:16).

MTConnect, endüstriyel ekipmanların gereksinimlerini belirlemek adına dereceli bir yaklaşım benimsemektedir. Her bir üretim ekipmanı için ihtiyaç duyulan veriler, önceden ekipmanlar üzerinde tanımlanmış ve adreslenmiş olmamaktadır. MTConnect protokolü özellikle metal işleme tezgâhları için standartlaştırılmış bir veri diyagramı sunmaktadır. Ancak diğer ekipmanlar için özelleştirmeye ve geliştirmeye açık XML şemaları sağlanmaktadır. Bu şemalar kullanılarak veriler tezgahlardan yayınlanmakta ve üst katmanlara aktarılmaktadır. Bu sayede, MTConnect, endüstriyel üretim aşamalarında geniş bir kullanım alanına sahip olmaktadır. MTConnect, endüstriyel otomasyon süreçlerinde ve Endüstri 4.0 aşamalarında üretim alanlarının dijitalleşmesinde ve veri odaklı üretimde önemli bir rol oynamaktadır.

3.4.4. Fanuc FOCAS protokolü

Fanuc FOCAS, Fanuc CNC sistemleri ile tezgahlara erişim sağlamak ve verileri toplamak amacıyla kullanılan açık kaynaklı bir API (Application Programming Interface) protokoldür. Bu protokol, Fanuc CNC kontrolörleri üzerindeki verilere, fonksiyonlara ve diğer özelliklere programlı bir şekilde erişim sağlamak amacı ile tasarlanmıştır. Fanuc FOCAS, CNC makineleri ile diğer cihazlar arasında iletişim kurulabilmesi ve CNC sistemi üzerinde bulunan verilere ulaşılarak özel uygulamalar geliştirilebilmesi amacıyla birtakım fonksiyonlar sağlamaktadır. FOCAS, CNC kontrolör üzerindeki çeşitli verilere çift yönlü bir erişim sağlamaktadır. Aynı zamanda FOCAS, CNC kumandası üzerinde bulunan çeşitli fonksiyonlara erişilebilmesine ve çağırılabilmesine olanak sağlamaktadır.

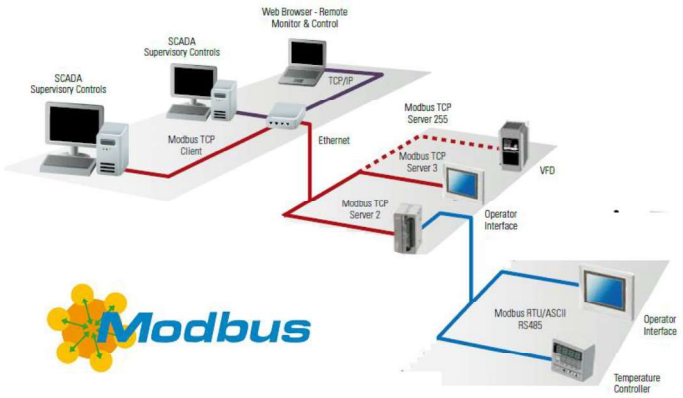
Bu özellikleri sayesinde Fanuc FOCAS protokolü, Fanuc marka kontrolcülere sahip ekipmanlardan veri toplanabilmesine ve bu ekipmanlara entegre uygulamalar geliştirilmesine olanak sağlamaktadır. Bu protokolün en önemli dezavantajı ise Fanuc marka için özelleşmiş olmasıdır. Bu nedenle sadece Fanuc marka kontrolcüye sahip ekipmanlardan bu protokol kullanılarak veri toplanabilmektedir. Şekil 3.16' da FOCAS mimarisi gösterilmektedir.



Şekil 3.16. Fanuc FOCAS mimarisi (Inventcom. Communication With Ethernet Board Erişim 8.11.2023)

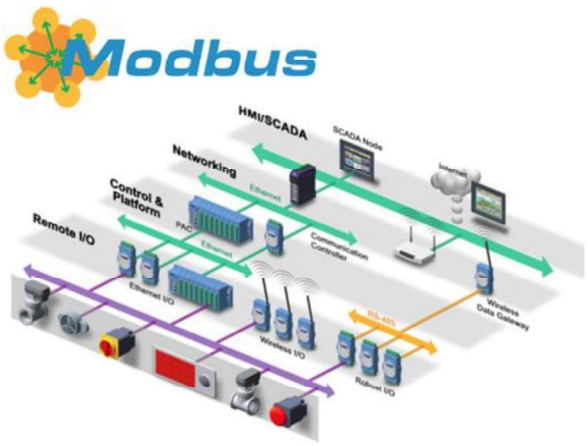
3.4.5. MODBUS protokolü

MODBUS, endüstriyel otomasyon sistemlerinde veri iletişiminin sağlanabilmesi amacı ile geliştirilmiş bir seri haberleşme protokolüdür. 1979 yılında Gould-Modicon şirketi tarafından geliştirilmiş olan protokol, genellikle endüstriyel cihazlar, sensörler ve kontrol cihazları gibi otomasyon ekipmanları arasında veri iletişimi için kullanılmıştır. Günümüzde MODBUS bu ekipmanların kendi aralarında yapmış oldukları veri iletişimini üst katmanlara aktarmak içinde kullanılmaktadır. MODBUS, genellikle seri iletişim bağlantıları üzerinden çalışmaktadır ve temelde iki versiyona sahiptir ve bu versiyonlar MODBUS RTU (Remote Terminal Unit) ve MODBUS ASCII (American Standart Code for Information Interchange) olarak adlandırılmaktadır. MODBUS protokolü, istemci-sunucu modeline dayanmaktadır. Yani bir cihaz (istemci) diğer bir cihazdan (sunucu) veri talebi yapmaktadır ve bu talepler doğrultusunda belirli fonksiyon kodları kullanılarak veriler iletilir. Şekil 3.17'de MODBUS mimarisi gösterilmektedir.



Şekil 3.17. MODBUS mimarisi (3eendustiyel. MODBUS ve Çalışma Mantığı Erişim 8.11.2023)

MODBUS RTU, 8 bitlik bir format içerisinde bir cihazdan diğer cihaza verileri bir çerçeve (frame) olarak iletmektedir. Genellikle RS-232 veya RS-485 gibi seri haberleşme bağlantılarında kullanılmaktadır. MODBUS ASCII, cihazlar arasında metin tabanlı bir formatta ASCII karakterlerini kullanarak veri iletimini sağlamaktadır. Bu iki MODBUS versiyonu genellikle endüstriyel ekipmanların ve bileşenlerin kendi aralarında haberleşmesinde kullanılmaktadır. MODBUS TCP/IP ise verinin üst katmanlara akışını sağlayan MODBUS versiyonudur. MODBUS TCP/IP, MODBUS protokolünün Ethernet üzerinden iletişimini sağlamaktadır. Bu yapı endüstriyel otomasyon sistemlerinde ve IIoT uygulamalarında geniş bir kullanıma sahiptir. TCP/IP tabanlı bu versiyon, endüstri standardı olan Ethernet ve TCP/IP protokollerini kullanarak veri iletimini hızlandırmaktadır ve daha geniş ağlarda kullanıma uygun hale getirmektedir. Şekil 3.18'de MODBUS TCP/IP protokolünün mimarisi gösterilmektedir.



Şekil 3.18. MODBUS TCP/IP mimarisi (ProgWorld. How Machines Communicate: MODBUS Protocol Erişim 9.11.2023)

MODBUS protokolü basit ve yaygın kullanıma sahiptir. Sistemlere kolay entegre edilir ve konfigürasyon yapılması kolaydır. Birçok çeşitli endüstriyel cihaz ve sensör bu protokol ile uyumludur. Ancak bu protokol ile üretimde kullanılan ekipmanlardan veri toplanmak istenildiğinde, toplanması istenilen verilerin ekipmanın kontrolcüsü tarafında MODBUS yayını için adreslenmiş olması gerekmektedir. Eğer ekipmanın üreticisi bu adresleme işlemlerini yerine getirir ve son kullanıcının üst katmanlara veriyi iletebilmesi için veri adreslerini paylaşır ise bu yöntemle veriler toplanabilmektedir.

3.4.6. Kepware Kepserver Ex yazılımı

Endüstriyel otomasyonun ve beraberinde IIoT teknolojisinin hızla gelişmesi sonucunda, farklı cihazların ve kontrol sistemlerinin uyumlu bir şekilde çalışabilmesi büyük bir önem taşımaktadır. Bu noktada, Kepware Kepserver Ex gibi iletişim platformları, farklı protokoller arasında veri alışverişini sağlamak ve endüstriyel otomasyon sistemlerini entegre etmek için önemli bir rol oynamaktadır. Kepware Kepserver Ex, endüstriyel otomasyon ve IIoT uygulamalarında kullanılmak üzere geliştirilmiş bir endüstriyel iletişim platformudur. Geniş bir çapta iletişim protokollerini desteklemesi, çeşitli cihazlar ve sistemler arasında veri entegrasyonunu kolaylaştırması ve endüstri standardı sürücü kütüphaneleri ile uyumlu olması, Kepware'i özellikle IIoT projeleri için vazgeçilmez kılmaktadır.

Kepware uygulamasının çalışma prensibini anlayabilmek için kabiliyetlerine detaylı bir şekilde bakmak gerekmektedir. Aşağıda Kepware 'in kabiliyetleri anlatılmaktadır.

- Protokol Dönüştürme: Kepware Kepserver Ex, endüstri standardı iletişim protokollerini destekleyerek, farklı cihazlar arasında iletişim kurabilmektedir. MODBUS, OPC UA, MTConnect gibi protokollerin desteği, geniş bir cihaz yelpazesi ile entegrasyonu mümkün kılmaktadır.
- Sürücü Kütüphanesi: Yazılımın sürücü kütüphanesi, çeşitli endüstriyel cihazlar ile uyumlu sürücüler içermektedir. Allen-Bradley, Siemens, Schneider Electric gibi önde gelen üreticilere özel sürücüler, geniş bir uyumluluk sağlamaktadır.
- Cihaz Entegrasyonu: Kepware, PLC'ler, sensörler, NCU'lar gibi endüstriyel cihazlar ile iletişim kurarak, bu cihazların bir araya gelebilmesini ve entegrasyonunu sağlamaktadır.

- Modüler Yapı ve Esnek Konfigürasyon: Modüler yapısı, kullanıcılara ihtiyaca yönelik sürücüleri ekleyip çıkarma esnekliği sağlamaktadır. Bu durum, projelerin ölçeklenebilir ve özelleştirilebilir olmasını sağlamaktadır.
- Veri Toplama ve İşleme: Kepware, bağlı cihazlardan sürekli olarak veri toplamaktadır. Bu veriler, endüstriyel otomasyon süreçlerinin izlenmesi, kontrol edilmesi ve raporlanması için kullanılabilmektedir.
- Güvenlik Özellikleri: Yazılım, OPC Security protokolü ile veri trafiğini şifreleyerek yetkilendirme sağlamaktadır. Bu sayede, veri iletimi sırasında güvenlik en üst düzeye çıkarılabilmektedir.
- Endüstri 4.0 Uyumluluğu: Kepware, Endüstri 4.0 ve IoT uygulamalarına uygun olarak tasarlanmıştır. Toplanan verilerin işlenerek kullanılabilmesine imkân sunarak, işletmelerin verimliliğinin artırılmasına katkı sağlamaktadır.

Kepware uygulaması genel olarak incelendiğinde avantaj ve dezavantajları Tablo 3.1'de anlatılmaktadır.

Tablo 3.1. Kepware Kepserver Ex avantaj ve dezavantajları

Avantajları	**Dezavantajları**
Çeşitli iletişim protokollerini destekleme kabiliyeti	Özellikle küçük ölçekli projeler için maliyeti yüksektir.
Geniş sürücü kütüphanesi ve cihaz uyumluluğu	Büyük projelerde konfigürasyonun karmaşık olması
Esnek konfigürasyon ve modüler yapı	
Veri güvenliği ve endüstri standardı uyumluluğu	

3.4.7 Microsoft SQL Server

Veri tabanı yönetim sistemleri (DBMS), günümüzde kurumların veri yönetimi ihtiyaçlarına karşı çeşitli çözümler sunmaktadır. Bu bağlamda, Microsoft SQL Server, veri depolama, yönetim ve analiz konularında endüstri standardı bir çözüm sunan bir DBMS olarak öne çıkmaktadır. Microsoft SQL Server, Microsoft tarafından geliştirilen

ve dağıtılan bir veri tabanı yönetim sistemidir. İşletmelerin veri tabanı yönetiminden, iş zekâsı uygulamalarına kadar geniş bir yelpazede kullanılan bu yazılım, güçlü performansı ve kapsamlı özellikleri ile dikkat çekmektedir.

Microsoft SQL Server'ın çalışma prensibini anlayabilmek için kabiliyetlerine detaylı bir şekilde bakmak gerekmektedir. Aşağıda Microsoft SQL Server'ın kabiliyetleri anlatılmaktadır.

- Veri Depolama ve Yönetim: SQL Server, ilişkisel veri tabanı yönetim sistemleri için özelleştirilmiş bir çözümdür. Tablolar, sütunlar ve ilişkiler aracılığıyla yapılandırılan veri depolama modeli, verilerin etkili bir şekilde yönetilmesini sağlamaktadır.
- T-SQL ce Stored Procedures: Transact-SQL (T-SQL), SQL Server ile etkileşimde bulunmak için kullanılan bir sorgu dili olarak öne çıkmaktadır. Stored Procedures gibi özellikler, veri manipülasyonu ve iş mantığı uygulamak için kullanılmaktadır.
- Güvenlik ve Yetkilendirme: SQL Server, kullanıcı ve rol tabanlı güvenlik modeli ile veri güvenliğini sağlamaktadır. Veri tabanına erişim kontrolü ve yetkilendirme, hassas bilgilerin korunmasına yardımcı olmaktadır.
- Yedekleme ve Kurtarma: SQL Server, düzenli yedekleme ve kurtarma özellikleri ile veri kaybını önlemektedir. Yedekleme planları ve geri yükleme seçenekleri, iş sürekliliğini güvence altına almaktadır.
- İş Zekâsı ve Raporlama: SQL Server, iş zekâsı ve raporlama için entegre araçlar sunmaktadır. SQL Server Reporting Services (SSRS) ve SQL Server Analysis Services (SSAS) gibi bileşenler, verilerin analizini ve raporlanmasını kolaylaştırmaktadır.
- Uygulama Geliştirme ve Entegrasyon: SQL Server, .NET Framework ile entegre olabilen bir platform sunmaktadır. Bu, uygulama geliştiricilerin veri tabanına kolayca erişmesini ve iş mantığı uygulamalarını entegre etmesini sağlamaktadır.

Microsoft SQL Server uygulaması genel olarak incelendiğinde avantaj ve dezavantajları Tablo 3.2'de anlatılmaktadır.

Tablo 3.2. Microsoft SQL Server avantaj ve dezavantajları

Avantajları	**Dezavantajları**
SQL Server, yüksek performans ve ölçeklenebilirlik sunmaktadır, böylece büyük veri hacimleriyle başa çıkabilmektedir.	SQL Server, diğer bazı DB MS'lerle karşılaştırıldığında yüksek lisans maliyetlerine sahiptir.
Kullanıcı ve rol tabanlı güvenlik modeli, hassas verilerin korunmasını sağlamaktadır.	Bazı durumlarda, özellikle küçük ölçekli projelerde, yapılandırma ve yönetim karmaşıklığı artmaktadır.
Entegre iş zekâsı araçları, verilerin analizini ve raporlanmasını kolaylaştırmaktadır.	
.NET Framework entegrasyonu, uygulama geliştirme süreçlerini desteklemektedir.	

3.4.8. Grafana yazılımı

Günümüzde, büyük veri setleri ve karmaşık sistemlerle çalışan işletmeler için veri analitiği ve görselleştirme giderek daha önemli hale gelmektedir. Bu noktada, Grafana gibi güçlü bir araç, kullanıcılara verilerini etkili bir şekilde anlama ve görselleştirme imkânı sunmaktadır.

Grafana, açık kaynaklı bir görselleştirme ve analitik platformudur. Veri kaynaklarından gelen bilgileri çekerek, bu verileri anlamlı ve etkili grafiklere dönüştürme yeteneği ile öne çıkmaktadır. Grafana, birden çok veri kaynağından veri alabilmektedir ve bu verileri tek bir konsol üzerinden izleme ve analiz etme imkânı sunmaktadır.

Grafana' nın çalışma prensibini anlayabilmek için kabiliyetlerine detaylı bir şekilde bakmak gerekmektedir. Aşağıda Grafana 'nın kabiliyetleri anlatılmaktadır.

- Veri Kaynakları ile Entegrasyon: Grafana, çeşitli veri kaynakları ile entegre olabilir. Bu kaynaklar arasında SQL veri tabanları, Prometheus, InfluxDB, Elasticsearch, CloudWatch gibi yaygın olarak kullanılan sistemler bulunmaktadır.

- Görselleştirme Panel ve Grafikler: Kullanıcılar, Grafana üzerinde paneller oluşturarak çeşitli veri kaynaklarından gelen verileri görselleştirebilmektedirler. Grafikler, çizgi grafiklerinden haritalara kadar birçok farklı türde olabilmektedir.
- Anlık Uyarılar ve Bildirimler: Grafana, belirli koşulların sağlanması durumunda anlık uyarılar ve bildirimler gönderme yeteneğine sahiptir. Bu özellik, operasyonel durumları anında takip etmeyi kolaylaştırmaktadır.
- Esnek Eklenti ve Geniş Topluluk Desteği: Grafana, zengin eklenti ve entegrasyon seçenekleri sunmaktadır. Grafana panel eklentileri, kullanıcıların farklı veri kaynaklarını daha etkili bir şekilde görselleştirmelerine olanak tanımaktadır. Ayrıca, geniş bir topluluk tarafından desteklenmektedir ve sürekli olarak geliştirilmektedir.

Grafana uygulaması genel olarak incelendiğinde avantaj ve dezavantajları Tablo 3.3'te anlatılmaktadır.

Tablo 3.3. Grafana avantaj ve dezavantajları

Avantajları	**Dezavantajları**
Grafana, birçok farklı veri kaynağı ile entegre olabilmektedir, bu da kullanıcılara geniş bir veri çeşitliliği sağlamaktadır.	Grafana' nın tam potansiyelinden yararlanabilmek için, kullanıcıların platforma aşina olmaları ve çeşitli özellikleri öğrenmeleri gerekmektedir.
Grafana, kullanıcılara çeşitli görselleştirme seçenekleri sunmaktadır. Bu, verilerin farklı şekillerde ve grafik türlerinde gösterilebilmesini sağlamaktadır.	Büyük veri setleriyle çalışırken, bazı durumlarda performans sorunları ortaya çıkabilmektedir.
Belirli koşulların sağlanması durumunda anlık uyarılar ve bildirimler, operasyonel hataların hızla tespit edilmesine olanak tanımaktadır.	
Geniş bir topluluk ve eklenti ekosistemi, kullanıcılara sürekli güncellenen ve gelişen bir platform sunmaktadır.	

Grafana, görselleştirme ve analitik konularında güçlü bir platform olarak öne çıkmaktadır. Geniş veri kaynağı desteği, esnek görselleştirme seçenekleri ve anlık bildirim yetenekleri, kullanıcılara verilerini daha etkili bir şekilde anlama ve değerlendirme fırsatı sunmaktadır.

3.5. Tezgahlar ile Nesnelerin İnterneti Teknolojisi Entegrasyonu

3.5.1. Endüstriyel nesnelerin interneti altyapısının kurulması ve mimari tasarımı

IIoT mimarisi, birçok cihaz, sensör, yazılım ve ağ teknolojilerini barındıran karmaşık bir yapı olarak karşımıza çıkmaktadır. Genel olarak IIoT mimarisi üç temel katmandan oluşmaktadır. Bu katmanlar; cihaz katmanı, ağ katmanı ve bulut katmanıdır. Bu katmanlar IIoT'nin en temel mimarisini oluşturmaktadır. Ancak bu mimari detaylandırıldığında beş ana katmandan oluşmaktadır. Bu katmanlar ise Şekil 3.19'da görüleceği üzere; Cihaz Katmanı, İletişim Katmanı, Veri-Bilgi Katmanı, Analiz Katmanı, Süreç Entegrasyon Katmanı olarak adlandırılmaktadır.



Şekil 3.19. IIoT katmanları

Aşağıda Şekil 3.19'da görülen IIoT katmanları açıklanmaktadır.

- Cihaz Katmanı: Bu katmanda sahadan veri toplanması muhtemel olan tüm ekipmanlar yer almaktadır (Farooq, Waseem, Mazhar, Khairi, Kamal, 2015:2). Bu ekipmanlar, veri kaynağı olarak değerlendirilen, endüstriyel uygulamalarda kullanılan cihazlar, sensörler ve diğer ekipmanlar olabilmektedir. Şekil 3.19'da görüldüğü üzere bu katman IIoT mimarisinin en alt katmanında yer almaktadır. Temel amacı veriyi üst katmanlara iletmek olan bu mimarinin en önemli ve temel katmanı Cihaz Katmanı olmaktadır.

- İletişim Katmanı: Cihaz Katmanında elde edilen verilerin üst katmanlara iletilmesi İletişim Katmanı aracılığı ile gerçekleşmektedir. Toplanan veriler bu katman aracılığı ile kullanılan güvenli iletişim protokolleri sayesinde üst katmanlara aktarılmaktadır (Mohamed, 2019:53). Bu katman, kablolu (Ethernet) veya kablosuz (Wifi, Bluetooth) iletişim protokolleri aracılığı ile veri yayınlama (OPC-UA) protokollerini kullanarak verileri üst katmanlara ileterek cihazlar ve sunucular arasında iletişimi sağlamaktadır. İletişim katmanı güvenilir, gecikme zamanı düşük ve geniş bant aralığına sahip bağlantılar gerektiren endüstriyel ortamlar için optimize edilmektedir.
- Veri – Bilgi Katmanı: Üst katmanlarda herhangi bir amaca hizmet etmek üzere toplanan verilerin uygun veri tabanlarına ya da bulut sistemlerine kaydedilmesini hedefleyen katmandır. Bu katmanda ölçeklenebilir ve güvenilir bir saklama alanı olmalıdır (Grgić ve ark., 2016:252). Bu saklama alanları, verilerin gerçek zamanlı olması, veri büyüklüğü ve veri toplama sıklığına göre değerlendirilerek belirlenmektedir. Genellikle IIoT aracılığı ile toplanan veriler büyük veriler olmaktadır. Bu nedenle en uygun sistemler Bulut sistemleri ya da No-SQL veri tabanları olmaktadır. Ancak çalışmamızda olduğu gibi özelleşmiş çalışmalarda SQL veri tabanları kullanılabilmektedir.
- Analiz Katmanı: Bu katman event alarm tespiti ve bildirimi, anormallik tespiti, kestirimci bakım gibi Makine Öğrenmesi uygulamaları; performans, kalite, kullanılabilirlik ve bunları içeren GEE takibi, alan durumlarının takibi gibi uygulamaları içermektedir. Bahsedilen uygulamalar bu katmanda yer almaktadır ve önceki katmandan iletilen veriler kullanılmaktadır (Bello ve ark., 2017:54). IIoT mimarisi kullanılarak ekipmanlardan toplanan veriler bu katmanda anlam kazanmaktadır.
- Süreç – Entegrasyon Katmanı: Önceki katmanlarda toplanarak anlamlı hale getirilen veriler, bu katmanda görselleştirilmektedir. Bu katman sayesinde görselleştirilen veriler ile üretim süreçlerinin takibinin yapılmasını sağlayacak gösterge tabloları oluşturulabilmektedir ve son kullanıcıya veriler sunulmaktadır (Fortino ve ark., 2022:27). Şekil 3.19'da görüldüğü gibi bu katman mimarinin en üst konumunda yer almaktadır. Bu mimarinin asıl hedefi ekipmanlardan toplanan verilerin işlenerek ve anlamlandırılarak bu katmanda son kullanıcılara sunulmasıdır. Bu katman içerisinde görselleştirme ve rapor araçlarını (Grafana, Power-BI) barındırmaktadır. Genellikle

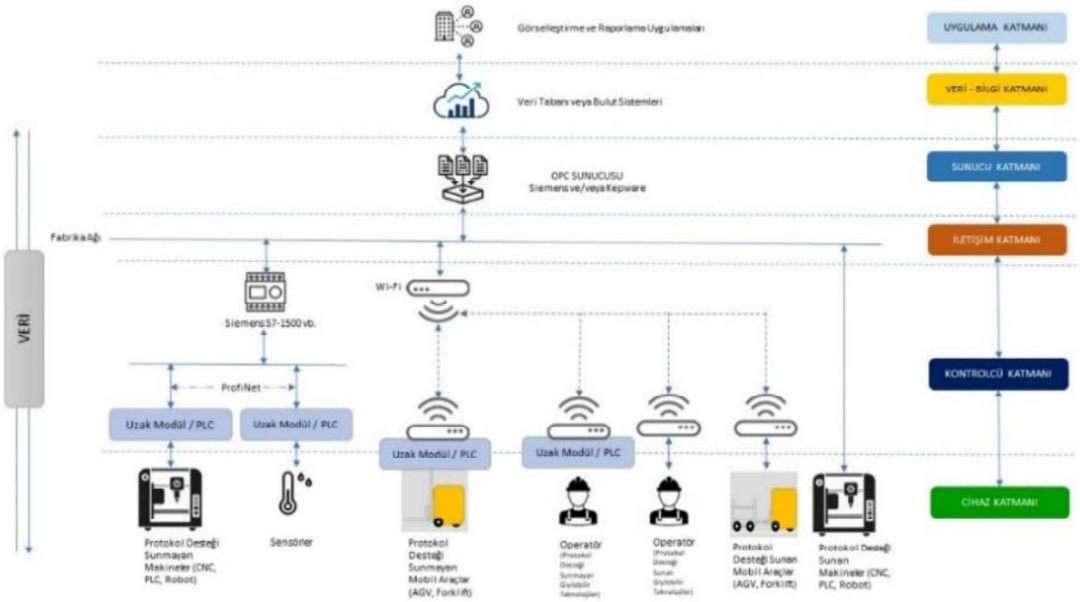
kullanılan bu araçlar ile veriler anlamlı bir şekilde son kullanıcıya raporlanmaktadır. Ayrıca bu katmanda ERP (Kurumsal Kaynak Planlaması), MES (Üretim Yürütme Sistemi), PLM (Ürün Yaşam Yönetim Döngüsü) gibi firmaların kurumsal uygulamalarına da entegrasyon sağlanmaktadır.

Bu çalışmada yukarıda anlatılan mimari temel alınmıştır, ancak çalışmamız özelinde şekillendirilen katmanlar neticesinde konsept bir mimari ortaya çıkmaktadır. Çalışmanın altyapısı aşağıda anlatılan ve Şekil 3.20'de görülen mimari üzerine kurulmuştur.



Şekil 3.20. Tasarlanan IIoT mimarisi

Tasarlanan mimari çalışmanın amacına yönelik olarak şekillenmiştir. Bu doğrultuda bir, üç ve beşinci katmanlar yukarıda anlatılan mimari ile aynı yapılara sahiptirler. Altıncı katman ise yukarıda anlatılan mimarinin Süreç – Entegrasyon Katmanına benzerdir. Tasarlanan mimari iki ve dördüncü katmanlarda farklılıklar oluşturmaktadır. Şekil 3.21'de çalışmamızda uygulanan altyapı görülmektedir.



Şekil 3.21. Tasarlanan IIoT altyapı mimarisi

- Cihaz Katmanı: Tasarlanan mimarinin temelinde yer alır ve yukarıda anlatılan özelliklere sahiptir. Çalışmamızda bu katmanda üretim alanlarında kullanılan ekipmanlar bulunmaktadır. Bu ekipmanlar farklı tiplerden birer tane olmak üzere seçilmiştir. Şekil 3.22'de örnek bir otomatik dikey işleme tezgâhı görülmektedir.



Şekil 3.22. DMG Mori DMU 65 monoBlock 5 eksen işleme tezgâhı (DmgMori. DMU 75 (FD) Mono Block 2. Nesil Erişim 9.11.2023)

- Kontrolcü Katmanı: Verilerin toplanması için ekipmanların hangi protokoller ile veri yayınlama kabiliyetine sahip olduklarının belirlenmesi önemlidir. Bu nedenle sonrasında değişmesi muhtemel bağlantı politikalarının daha net belirlenebilmesi için

bu katman tasarlanmıştır. Ekipmanlar incelenerek üzerlerinde bulunan kontrolcüler tespit edilmektedir. Bu kontrolcüler ise İletişim Katmanında bulunacak olan veri yayınlama protokollerini ve veri toplama yöntemlerini belirleyecektir. Kontrolcü Katmanında oluşabilecek farklılıkların iletişim yöntemlerinde nasıl farklılıklar doğurabileceği Şekil 3.21'de gösterilmektedir. Şekil 3.23'de ise örnek bir kontrolcü ürün ailesi yer almaktadır.



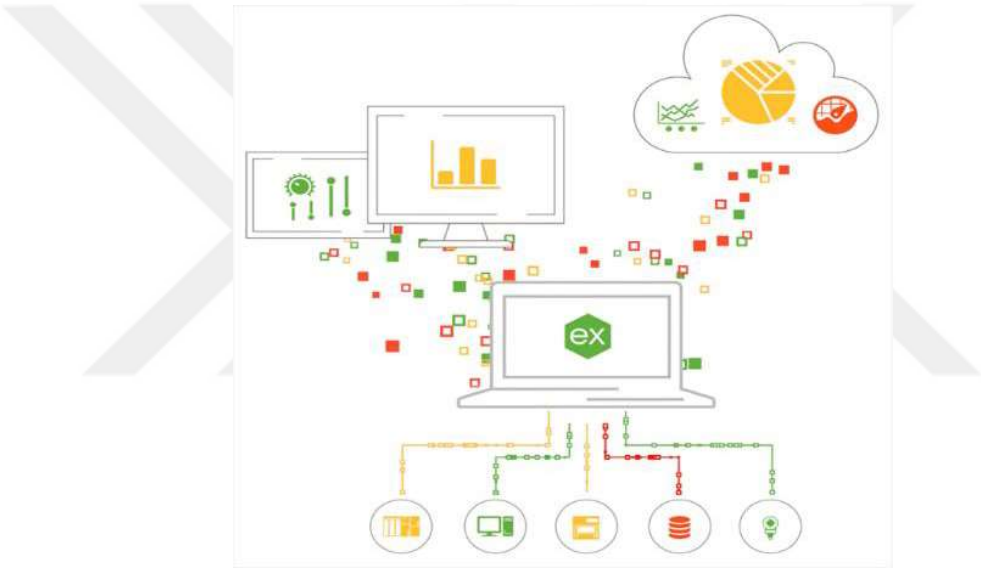
Şekil 3.23. Siemens 840D SL NCU ve operatör paneli (Indiamart. Sinümerik 840D SL Controller Erişim 9.11.2023)

- İletişim Katmanı: Yukarıda anlatılmış olan İletişim Katmanı ile ilgili özelliklerin tümüne sahiptir. Bu katmanda yer alan veri yayınlama protokolleri Bölüm 3.4'de anlatılmaktadır. Bu protokoller aracılığı ile yayınlanan verilerin üst katmanlara iletilebilmesi için ise çalışmanın gerçekleştirildiği üretim alanının politikaları gereği kablolu olarak ve Ethernet protokolü kullanılarak gerçekleştirilmiştir. Şekil 3.24'te örnek bir Ethernet – ağ anahtarı bağlantısı gösterilmektedir.



Şekil 3.24. Ethernet ve ağ anahtarı bağlantısı (Karel. Ethernet Nedir, Nasıl Çalışır? Erişim 9.11.2023)

- Sunucu Katmanı: Yukarıda anlatılan katmanlar aracılığı ile fabrika ağına çıkarılan verilerin üst katmanlarda depolanabilmesi ve görselleştirilmesi için verilerin sanal bir sunucu ortamına çıkarılması sağlanmaktadır. Sunucu ortamında çalışan ve üçüncü parti Kepware Kepserver Ex yazılımı kullanılmaktadır. (Şekil 3.25) Bu yazılım kullanılarak izlenebilen veriler yine bu yazılım aracılığı ile belirlenen zaman aralıkları ile servis aracılığı ile yayınlanmaktadırlar. Yayınlanan bu veriler geliştirilen Java uygulaması ile anlamlı hale getirilmektedir. Geliştirilen uygulama verilerin standardizasyonunu sağlamakla beraber üst katmanlara anlamlı verilerin aktarılmasına hizmet etmektedir.



Şekil 3.25. Kepware Kepserver Ex şema örneği (Alba, Engineering. KepServerEX Adds New MQTT Client Driver Erişim 9.11.2023)

- Veri – Bilgi Katmanı: Sunucu katmanında anlamlandırılan ve standartlaştırılan veriler bu katmanda bir SQL veri tabanına kaydedilmektedir. Bu katman anlam bakımından yukarıda anlatılan mimari ile aynıdır.
- Uygulama Katmanı: Anlamlı bir şekilde veri tabanlarına kaydedilmiş olan veri, son kullanıcılar için bu katmanda uygulamalara dönüştürülmektedir. Çalışmamızda anlamlı veriler Grafana adlı platform kullanılarak görselleştirilmektedir. (Şekil 3.26) Tasarlanan mimaride bu katman, yukarıda anlatılan mimarinin Süreç – Entegrasyon katmanı ile verilerin görselleştirilerek son kullanıcılara iletilmesi ve herhangi bir kurumsal uygulama ile entegre olmaması gibi nedenler ile farklılıklar içermektedir.



Şekil 3.26. Grafana rapor paneli örneği (Duranek, WitaliSoft. Netdata Data Source For Grafana Erişim 9.11.2023)

3.5.2. Ekipmanların belirlenmesi

IIoT altyapısının kurulması istenilen ekipmanların belirlenmesinde ilk öncelik üretim alanlarının ihtiyaçları doğrultusunda hareket etmektedir. Çalışmamız kapsamında üretimde kullanılan ve üretim alanlarının en değerli ekipmanlarından olan üretim tezgâhları üzerinde IIoT altyapısı kurulması hedeflenmektedir. Bu nedenle seçilen ekipmanlar üretim tezgahlarıdır. İlk olarak üretim alanında mevcut olan tezgahlar daha önce anlatılan incelemeler kapsamında gruplandırılmaktadır. Gruplandırılan tezgahlar kendi içlerinde alanların belirlemiş olduğu kritiklik seviyelerine göre önceliklendirilmektedir. Bunların neticesinde alt yapı kurulumunun gerçekleştirileceği tezgahlar belirlenmektedir. Gruplandırılan tezgahlarda alt yapı kurulması için benzer gruplar içerisinde yer alan tezgahlar için benzer adımlar izlenmektedir.

Çalışma kapsamında altı farklı tezgâh için IIoT altyapısı kurulmaktadır. Bu tezgahlar, farklı tip protokolleri destekler ve farklı kabiliyetlere sahiptirler. Tezgahlar belirlenirken ilk olarak üretim parkurunda bulunan tezgahlar destekledikleri protokollere göre gruplandırılmaktadır. Gruplandırılan tezgahlar içerisinden yaptıkları iş, üretim yoğunluğu gibi kriterler göz önünde bulundurularak üretim için en kritik tezgahlar belirlenmektedir. Bu adımların ardından belirlenen tezgahlar; Breton marka beş eksen dikey işleme tezgâhı, Mazak beş eksen dikey işleme tezgâhı, Mori Seiki marka üç eksen torna tezgâhı, Broetje marka c-frame panel montaj tezgâhı, Despatch marka ısıl işlem fırını ve Avure marka form pres tezgahıdır. Bu tezgahların detayları aşağıda anlatılmaktadır.

3.5.2.1. Breton Beş Eksen Dikey İşleme Tezgâhı



Şekil 3.27. Breton marka beş eksen dikey işleme tezgâhı (CNCBul. Breton CNC Makine Modelleri Erişim 9.11.2023)

Bu tezgâh üzerinde yapılan incelemeler neticesinde, tezgâhın CNC/NC tezgahlar grubunda yer aldığı ve OPC UA protokolünü desteklediği belirlenmiştir. Protokol desteği belirlenirken ilk olarak tezgâhın sahip olduğu kontrol marka ve modeli belirlenmektedir. İlgili tezgâh Siemens marka 840D SL modelinde bir NCU'ya sahiptir. İkinci adım olarak Siemens marka kontrolcülere sahip tezgahlar üzerinde çalışan bir tezgâh işletim sistemi olan Sinümerik versiyonu kontrol edilmektedir. Bu tezgâh "V4.7 SP.2" versiyonuna sahiptir. Bu tezgâh ilgili versiyon ve model kapsamında OPC UA desteği sağlamaktadır. Tezgâh menüsü üzerinden OPC UA lisansının aktiflik durumu ve OPC UA sunucusunun durumu kontrol edilmektedir.

İncelemeler sonucunda tezgâh üzerinde OPC UA lisansı bulunmaktadır ve OPC UA sunucusu kuruludur. Ancak OPC UA lisansı ve sunucu bulunmasına rağmen tezgâh üzerinde lisans ve sunucunun aktifleştirilmediği tespit edilmiştir. OPC UA protokolünün tezgâh üzerinde aktifleştirilmesi için gerçekleştirilen adımlar ilerleyen bölümlerde anlatılmaktadır.

Ayrıca tezgâh üzerinde yapılan işin önemi ve tezgâh ilk maliyetinin yüksek olması nedeni ile üretim alanı tarafından kritiklik değeri yüksek bir tezgahtır. Bu durumlar göz önünde bulundurularak Breton tezgâhı çalışma kapsamında değerlendirilmektedir (Şekil 3.27). Bu tezgâh çalışmanın ilerleyen kısımlarında 001 olarak numaralandırılmaktadır ve belirlenen isimlendirme metodu ile 001_Breton_OPCUA olarak isimlendirilmektedir.

3.5.2.2. Mazak beş eksen dikey işleme tezgâhı



Şekil 3.28. Mazak beş eksen dikey işleme tezgâhı (Mazak. Variaxis i-1050 Erişim 9.11.2023)

Mazak marka tezgahlar genellikle Endüstri 4.0 gereksinimlerini karşılayabilmek ve üretim süreçlerini daha verimli hale getirebilmek amacıyla MTConnect protokolü desteği sunmaktadır. İlgili tezgâh CNC/NC tezgahlar grubunda yer almaktadır ve Mazak marka Mazatrol 640M model kontrol ünitesine sahiptir. Bu tezgâh MTConnect protokolü desteği sunmaktadır. Bu marka tezgahlarda ise OPC UA desteği sunan ve özellikle Siemens marka kontrolcülere sahip tezgahlar gibi MTConnect bir opsiyon sunmamaktadır. Ancak MTConnect protokolünün devreye alınması için tezgâh üzerinde MTConnect adaptörü bulunması gerekmektedir. MTConnect adaptörleri eğer daha öncesinde tezgâha kurulmadı ise OPC UA lisansına benzer bir şekilde ilerleyen zamanlarda MTConnect adaptör yazılımı geliştiren üçüncü parti firmalardan temin edilebilmektedir.

İlgili tezgâh üzerinde bulunan Mazatrol 640M model kontrolcü Windows işletim sistemine sahiptir. MTConnect adaptörünün durumunun kontrol edilebilmesi için Windows işletim sistemi tezgâh üzerinde açılmalıdır. Windows üzerinden “C://ProgramFiles” klasörü içerisinde MTConnect adında bir klasörün varlığının kontrol edilmesi gerekmektedir. Bu klasör varlığı MTConnect adaptörünün kurulu olduğu anlamına gelmektedir. MTConnect kurulumu ve devreye alınması ile ilgili adımlar ilerleyen bölümlerde detaylı olarak anlatılmaktadır.

Tezgâhın yaptığı işlerin değeri ve tezgâh maliyetinin yüksek olması nedeni ile belirlenen bu tezgâh üretim için oldukça kritik tezgahlardan bir tanesidir. Bu tezgâh MTConnect protokolünü desteklemesi ve kritik bir tezgâh olması nedeni ile çalışmaya dahil edilmiştir. Çalışmanın ilerleyen bölümlerinde bu tezgâh 002_Mazak_MTConnect olarak adlandırılmaktadır. (Şekil 3.28)

3.5.2.3. Mori Seiki üç eksen torna tezgâhı



Şekil 3.29. Mori Seiki marka üç eksen dikey işleme tezgâhı (Petru ve Ark., 2017)

Seçilen bu tezgâh üretim parkuruna dahil olduğu tarih itibari ile Mori Seiki markası ile yer almıştır. Ancak daha sonralarında gerçekleşmiş olan DMG ve Mori Seiki markalarının ortaklığı ile günümüzde DMG Mori markası ile yoluna devam etmektedir. İlgili tezgah Şekil 3.29'da gösterilmektedir. Dolayısı ile çok geniş bir ürün yelpazesine sahip olan firma genellikle CNC/NC tezgâh grubunda yer alan, farklı tip ve yetkinliklere sahip tezgahlarında farklı markaların kontrolcülerini kullanabilmektedir. Bu nedenle ilgili tezgâhın desteklediği protokolün tespit edilebilmesi için kullanılan kontrolcü markası ve modeli önemli olmaktadır. İlgili tezgâh Fanuc markasının 18i modelinde bir kontrolcüye sahiptir. Bu marka ve modele sahip kontrolcüler farklı yöntemler ile veri yayını yapmaktadırlar. Bu yöntemlerden bir tanesi MTConnect protokolü olsa dahi, Fanuc kontrolcüler genellikle FOCAS protokolünü içlerinde barındırmaktadır. Bu tezgahlarda FOCAS protokolünü kullanmak çok daha kolay ve daha az maliyetle devreye alınabilmektedir. FOCAS kütüphanesi ile veri toplanabilmesi için gerekli adımlar ilerleyen bölümlerde anlatılmaktadır.

Yaptığı iş neticesinde kritik olan bu tezgâh Fanuc marka bir kontrolcüye sahip olması ve FOCAS protokolü ile veri yayını sağladığı için çalışma kapsamında yer almaktadır. Tezgâh ilerleyen bölümlerde 003_MoriSeiki_FOCAS olarak adlandırılmaktadır.

3.5.2.4. Broetje C-Frame panel montaj tezgâhı



Şekil 3.30. Broetje C-Frame panel montaj tezgâhı (Aeroexpo. Electric Riverting Machine CPAC Erişim 9.11.2023)

Üretim aşamalarının en önemli aşamalarından biri olan montaj bölümünün en kritik tezgahlarından biri olan bu tezgâh, üretilen panellerin perçinler yardımı ile montajlanmasını sağlamaktadır. Montaj aşamasında bu tezgâhın kullanılması verimlilik ve kalite aşamaları için kritik etkiler oluşturmaktadır. Bu nedenle üretim için fazlası ile kritik olmaktadır ve çalışmada yer almaktadır. Bu tezgâh ilerleyen bölümlerde 004_Broetje_BFC olarak adlandırılmaktadır. (Şekil 3.30)

İlgili tezgâh yapılan gruplandırma çalışmalarında CNC/NC Tezgahlar grubunda yer almaktadır. IIoT altyapısına sahip olabilmesi için aynı grupta yer alan diğer tezgahlarda olduğu gibi tezgâh kontrolcüsünün marka ve modeli hangi protokol ile veri yayınlandığı konusunda yönlendirme yapmaktadır. Tezgâh kontrolcüsü, Siemens marka 840D SL modeline sahiptir. İlk inceleme sonucunda tezgâh ilgili kontrolcü ile OPC UA desteği sunabilir olarak gözükmektedir. Ancak ikinci adım olarak kontrolcü üzerinde bulunan sinümerik versiyonu kontrol edildiğinde tezgâh versiyonunun V4.3 SP1 olduğu belirlenmiştir. Versiyon uyumsuzluğu nedeni ile tezgâh OPC UA desteği sunmamaktadır. OPC UA desteği sunan tezgahlar ile ilgili detaylar önceki başlıklar içerisinde anlatılmaktadır.

İlgili tezgâhta OPC UA desteği sunmaması nedeni ile alternatif bir çözüm olan BFC Gateway kullanılmaktadır. BFC Gateway çözümü Siemens firması tarafından sağlanmaktadır. Satın alınan çözümün tezgâh üzerinde devreye alınması ile tezgâha veri yayınlama kabiliyeti kazandırılabilmektedir. BFC Gateway çözümünün devreye alınması ile ilgili detaylar ilerleyen bölümlerde anlatılmaktadır.

3.5.2.5. Despatch ısıl işlem fırını



Şekil 3.31. Despatch ısıl işlem fırını (Despatch. Raf/Rff Hıgh-Performance Batch Furnaces Erişim 9.11.2023)

Üretimin bazı aşamalarında ısıl işlem uygulanması gerekmektedir. Kritik olan bu işlemden dolayı ilgili fırının verilerine erişilmesi önem kazanmaktadır. Bu nedenle çalışma kapsamına dahil edilen bu fırın konvansiyonel ekipmanlar grubunda yer almaktadır. İlgili fırın Siemens marka S7-1500 model PLC'ye sahiptir ve bu PLC MODBUS desteği sunmaktadır. Ancak bu yöntem ile veri alınabilmesi için daha önce bahsedildiği gibi verilerin data blok adreslerinin bilinmesi gerekmektedir. Şirket içerisinde ilgili fırının satın alım sırasında teslim edilen dokümanları incelendiğinde, fırın verilerine ait data blok adreslerinin dokümanlar aracılığı ile paylaşıldığı tespit edilmiştir. Bunların neticesinde bu fırından MODBUS protokolü kullanılarak veri toplanmaktadır. Bu fırın 005_Despatch_MODBUS olarak adlandırılmaktadır. (Şekil 3.31)

3.5.2.6. Avure form pres tezgâhı



Şekil 3.32. Avure form pres tezgâhı (Ismr. Piper Aircraft selects hydroform press Erişim 9.11.2023)

Üretimin kritik olarak değerlendirilen aşamalarından biri olan pres işleminin gerçekleştirilebilmesi için kullanılan Avure marka form pres tezgâhı çalışma kapsamında 006_Avure_MODBUS_EA olarak adlandırılmaktadır ve konvansiyonel tezgahlar grubunda yer almaktadır. (Şekil 3.32)

İlgili tezgâh Siemens marka S7-1200 model PLC'ye sahiptir. Ancak tezgâha ait dokümanlar incelendiğinde, tezgâhın MODBUS protokolünü destekliyor olmasına rağmen verilerin data blok adreslerinin dokümanlarda yer almadığı tespit edilmiştir. Bu nedenle ilgili tezgâhtan üzerinde bulunan PLC aracılığı ile veri alınamadığı için manuel bir tezgâh gibi değerlendirilmektedir. Bu nedenle bu tezgâhtan veri alınabilmesi için ek bir donanım eklenmesi gerekmektedir. Tezgâh üzerinde yapılan incelemeler sonucunda, tezgâhın enerji verisinin takip edilebilmesi için tezgâha enerji analizörü eklendiği tespit edilmiştir. İlgili enerji analizörü Entes marka MPR46S modeline sahiptir. Enerji analizörüne sahip dokümanlar incelendiğinde bu donanımın MODBUS protokolü aracılığı ile veri yayınlayabildiği tespit edilmiştir. Dokümanlarda enerji analizörü üzerinden alınabilecek verilerin data blok adresleri de yer almaktadır. Bu nedenle enerji analizörü kullanılarak tezgâh ile ilgili veriler elde edilebilmektedir.

3.5.3. Ağ altyapısının kurulması ve desteklenen protokollerin devreye alınması

Ekipmanlar üzerinde IIoT mimarisi kurulabilmesi için ekipmanlar veri yayınlama protokollerine sahip olmalıdırlar. Ekipmanın veri yayınlama protokolüne sahip olmadığı durumlarda, ekipmana uygun protokoller eklenerek veri yayınlama kabiliyeti kazandırılmaktadır. Çalışma kapsamında incelenmiş olan protokollerden bazıları için tezgâhın uyumluluk durumuna göre sonradan kurulum yapılması mümkün olabilmektedir. Bu protokoller OPC UA, MTConnect protokolleri ve BFC Gateway'dir. MODBUS protokolü ise bunların aksine ekipmanın ilk kurulumunda üretici firma tarafından devreye alınması gereken ve veriler ile ilgili data blok adreslerinin ekipman sahibi ile paylaşılması gerekmektedir. Bunlara ek olarak enerji analizörleri vb. ekipmanlar ile veri toplanmak istenilirse, bazı cihazlar açık kaynaklı olarak MODBUS yayını yapmaktadır ve bu cihazlara ait veri adreslerine ilgili cihazların dokümanları aracılığı ile ulaşılabilmektedir.

Aşağıda veri yayınlama kabiliyeti olmayan ancak OPC UA, MTConnect ve BFC Gateway desteği sunabilen ekipmanlar için kurulumların nasıl yapılması gerektiği anlatılmaktadır.

3.5.3.1. OPC UA protokolünün devreye alınması

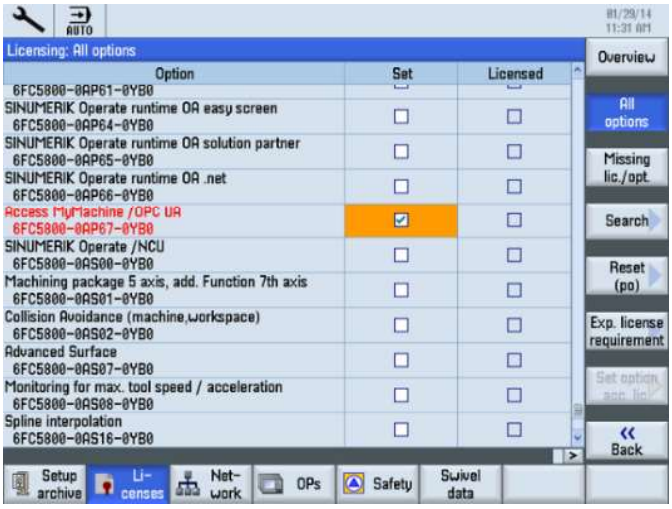
Siemens marka kontrolcülere sahip tezgahlar OPC UA protokolü ile veri yayınlarını desteklemektedir. Ancak bu durum her tezgâh için geçerli olmamaktadır. Bu nedenle uygun versiyona sahip tezgahlarda Siemens tarafından opsiyon olarak sunulan OPC UA lisansının satın alınmış olması gerekmektedir. Bu lisans tezgâhın ilk satın alımı sırasında alınmamış ise sonrasında satın alınarak tezgâh üzerinde aktif edilmesi gerekmektedir. Lisans aktif edilmeden önce olası sorunların önüne geçilebilmesi amacı ile ilk olarak tezgâh diskinin yedeğinin alınması gerekmektedir. Aşağıda yedekleme işleminin nasıl gerçekleştireceği ile ilgili adımlar anlatılmaktadır.

- ✓ Tezgâh ana menüsü 'Setup' ayarları üzerinden 'Password' seçeneği ardından 'Set Password' kullanılarak servis şifresi girilerek ilgili menüler aktif edilmektedir.
- ✓ 'Setup' ve ardından 'Setup Archive' yolundan 'Create Setup Archive' ile yedek alma işlemi başlatılmaktadır.

- ✓ Açılan sayfada sırasıyla ‘NC’ seçildikten sonra ‘PLC’ seçeneği seçilmektedir ve ekrandaki tüm seçenekler için bu işlem tekrarlanmaktadır. Yedekleme alınacak konum belirtilerek yedek alma işlemi tamamlanmaktadır.

Yedekleme işlemi tamamlandıktan sonra lisans aktifleştirilmesi için gerekli olan adımlar ise aşağıda açıklanmaktadır.

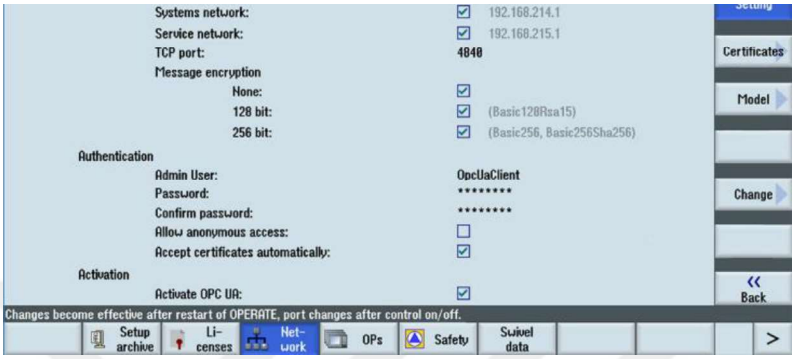
- ✓ İnternete bağlı bir bilgisayar kullanılarak Siemens’in internet sayfası üzerinden ‘www.siemens.com/automation/license’ adresine girilir.
- ✓ Açılan sayfada ‘Direct Access’ bölümüne girilir ve satın alınan lisans bilgileri ilgili yerlere girilerek ilerlenir.
- ✓ Lisansın kurulmasının istendiği tezgâha ait model ve bilgiler ilgili bölümlere tanıtılır.
- ✓ Satın alınan lisansın OPC UA opsiyonu kontrol edilmelidir.
- ✓ Sonraki adımda yeni lisans oluşturulur ve yeni lisans tezgâha girilir.
- ✓ Tezgâh yeniden başlatılır ve tezgâh menüsünde bulunan ‘License’ sayfasından ‘mymachine/ OPCUA’ seçeneği kontrol edilir.
- ✓ Lisans kurulumu tamamlanan tezgâh üzerinde OPC UA lisansı aktif edilir. (Şekil 3.33)



Şekil 3.33. Sinümerik lisans menüsü ve OPC UA opsiyonu (Siemens, 2018:18)

- ✓ OPC UA yayınının başlayabilmesi için OPC UA sunucusunun tezgâh üzerinde ayarlanması gerekmektedir.

- ✓ Tezgâh menüsü üzerinde bulunan diagnostic sayfasının altında OPC UA seçeneği bulunmaktadır.
- ✓ Bu menü üzerinden tezgâha ait IP bilgileri, OPC UA yayını şifre bilgileri tanımlanarak OPC UA yayını aktifleştirilmelidir. (Şekil 3.34)



Şekil 3.34. Sinümerik Operate OPC UA konfigürasyon sayfası (Siemens, 2018:20)

- ✓ Bu işlemler tamamlandığında tezgâh OPC UA ile veri yayınlamaya başlayacaktır.

3.5.3.2. MTConnect protokolünün devreye alınması

MTConnect Protokolü ile veri toplanmak istenildiğinde, eklenecek her bir tezgâh için bir adaptör kurulması ve sunucuda bir agent yazılımı olması gerekmektedir. Tezgahlar için agent tarafında pek bir değişiklik olmasa bile adapter ihtiyacı kaynaklı firmalardan satın almalar gerekmektedir. Aşağıda bahsi geçen kurulum MTConnect kurulumlarının genelinde aynı şekilde yapılmaktadır.

- ✓ Tezgâhın fabrika ağı ile bağlantısı kurularak IP ayarları yapılmalıdır.
- ✓ MTConnect adaptör kurulumu Windows tabanlı tezgahlara ya da sanal bir sunucu üzerine yapılabilmektedir.
- ✓ MTConnect agent kurulumu bir kere yapılacaktır ancak makine sayısı kadar MTConnect adaptör kurulumu yapılması gerekmektedir.
- ✓ MTConnect adaptör dosyası sanal sunucu ya da tezgâh üzerinde ‘Program Files’ konumundaki ‘MTConnect’ dosyasının içine atılmalıdır.
- ✓ Dosyalar atıldıktan sonra ‘CMD: Komut İstemi’ yönetici olarak çalıştırılarak bu adaptöre ait hizmetin oluşturulması gerekmektedir.
- ✓ Komut çalıştırıldıktan sonra servis kurulmuş olmaktadır.

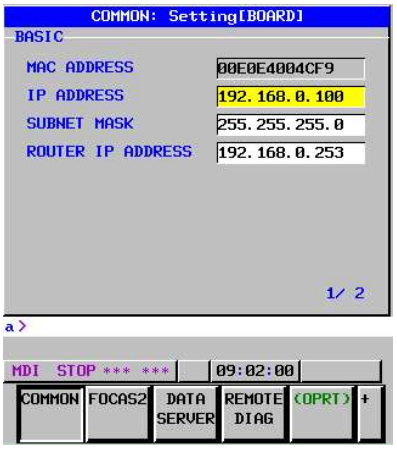
- ✓ Tezgâha ait bilgiler kurulan adaptör dosyasının içindeki ‘Machines.xml’ dosyasının içine doldurulmalıdır.
    - ID: Adaptör sayısı arttıkça artırılmaktadır.
    - KOD: Tezgâha ait isim kodu bu alana girilmektedir.
    - IP: Tezgâha verilen IP adresi bu alandan tanımlanmaktadır.
    - PORT: Her adaptörün portu farklı olmalıdır. Agentla haberleşirken portlardan hangi tezgâh olduğu ayırt edilmektedir. (Örneğin: 7878, 7879)
    - KONTROL ÜNİTESİ: Tezgâhın modeli buradan tanımlanmaktadır.
- ✓ ‘App.config’ dosyası içerisine lisans anahtarı girilmelidir.
- ✓ Adaptör başlatıldığında ‘log.txt’ dosyası içerisinde bulunan cihazın seri numarası kullanılarak adaptörün satın alındığı firmadan yeni bir lisans key talep edilmektedir.
- ✓ Oluşturulan lisans dosyasının app.config dosyası üzerine yüklenmesi gerekmektedir.
- ✓ Bu aşamalardan sonra MTConnect agent kurulumu gerçekleştirilebilmektedir.
- ✓ MTConnect Agent dosyasının içinde sırasıyla ‘cppagent1.7.0.2\install \Windows\src’ klasörüne gidilerek ‘MTConnectAgentInstall1.2.exe’ uygulamasının yönetici olarak çalıştırılması gerekmektedir.
- ✓ Açılan ekranda Agent’ın kurulumunun yapılacağı dosya konumu seçilmektedir ve ilerle butonu kullanılarak kuruluma devam edilmektedir.
- ✓ Agent kurulumundan sonra açılan sayfada adaptör konfigürasyonları yapılmaktadır.
- ✓ Adaptör konfigürasyonuna devam etmeden önce MTConnect agent kurulum dosyasının içinde bulunan ‘Devices.txt’ dosyası ‘C:\ProgramFiles\MTConnect\ MTConnectAgent \Devices\Mazak’ adresine taşınmaktadır.
- ✓ İlgili dosya taşındıktan sonra adaptör konfigürasyonuna devam edilmektedir.
- ✓ Makine seçimi sayfasında ‘Devices.txt’ dosyasının atıldığı klasör adresi seçilmektedir.
- ✓ İlerleyen adımda tezgâha ait kod, IP ve port tanımlanmaktadır.
- ✓ Ve ardımdan konfigürasyon kurulumu tamamlanmaktadır.
- ✓ MTConnect agent’ın kurulduğu dosya adresinde bulunan Agent.cfg dosyasına ulaşılmaktadır.
- ✓ Agent.cfg dosyası açılarak ve ne kadar makine varsa kopyala-yapıştır yaparak adaptörler konfigürasyon dosyasına eklenmektedir.

- ✓ Tezgâha ait isim kodu yazılmaktadır ve parantez içerisinde Host kısmına haberleşmeyi sağlayacak olan sunucunun IP'si yazılmaktadır. Port kısmına ise adaptör kurulumu sırasında Machines.xml dosyası içerisinde verilen port numarası yazılmaktadır.
- ✓ İşlem tamamlanarak, MTConnect agent ve adaptörleri sırasıyla servis olarak çalıştırılmaktadır.
- ✓ Ardından adaptör kurulumu yapılan ve agent konfigürasyonları tamamlanan tezgahlardan veri toplanmaya başlanmaktadır.

3.5.3.3. FOCAS protokolünün devreye alınması

FOCAS protokolü Fanuc marka kontrolcüye sahip tezgahlar özelinde kullanılabilmektedir. Bu protokolün devreye alınabilmesi için aşağıda anlatılan adımlar takip edilmektedir.

- ✓ Tezgâh kontrolcüsünün Ethernet portu kontrol edilerek fabrika ağına bağlanması gerekmektedir.
- ✓ Ağ bağlantısı kurulduğunda, tezgâh bağlantısının kurulabilmesi için tezgâha IP ataması gerçekleştirilmelidir. Eğer tezgâh daha öncesinde fabrika ağına bağlandı ise FOCAS protokolünün kullanılabilmesi için IP adresinin öğrenilmesi gerekecektir.
- ✓ Tezgâh menüsünden Sistem klasörüne girilir.
- ✓ Klasör içerisindeki sayfalar arasında Embeded Port sayfası bulunmaktadır.
- ✓ Bu sayfa içerisinde Common ve FOCAS2 olmak üzere iki menü görüntülenmektedir.



Şekil 3.35. Fanuc tezgah kontrolcüsü common menüsü(Scytec. Configuring the Fanuc Control Erişim 10.11.2023)

- ✓ Common menüsü üzerinden tezgâh IP adresine ulaşılabilmektedir. Eğer IP adresinin güncellenmesi ya da yeni bir IP adresi atanması gerekiyor ise bu sayfa üzerinden ilgili Ethernet ayarları yapılmaktadır. (Şekil 3.35)
- ✓ İlgili IP ayarları tamamlandıktan sonra FOCAS2 menüsü üzerinden FOCAS protokolünün iletişim ayarları görüntülenmektedir. (Şekil 3.36)
- ✓ Genellikle FOCAS protokolü 8193 haberleşme portunu kullanmaktadır. Ancak olası bir durumda ilgili port numarası ilgili menü üzerinden güncellenmektedir.



Şekil 3.36. Fanuc tezgâh kontrolcüsü FOCAS menüsü (Scytec. Configuring the Fanuc Control Erişim 10.11.2023)

- ✓ İlgili ayarlar tamamlandığında tezgâh FOCAS protokolü ile veri yayınlamaya başlayacaktır.

3.5.3.4. BFC Gateway yazılımının devreye alınması

BFC Gateway Siemens tarafından OPC UA desteği sunmayan marka ve model tezgahlardan veri toplanabilmesi için geliştirilmiş bir yazılımdır. BFC Gateway Siemens tarafından bir hizmet olarak satın alınmaktadır. Kabaca BFC Gateway kurulumunun adımları şöyledir;

- ✓ İlk olarak tezgâhın ağ bağlantısı kontrol edilmektedir. Fiziksel olarak tezgâh fabrika ağına bağlandı ise IP adresi ayarları kontrol edilecektir.
- ✓ Siemens Sinümerik Operate menüsünden "Diagnostic/Network/Bus TCP-IP" sayfasından IP ayarları yapılmaktadır.
- ✓ Tezgâh üzerine BFC Gateway istemci uygulaması kurulmaktadır.

- ✓ Fabrika ağı üzerine bağlı olan bir sanal sunucu üzerine BFC Gateway uygulaması kurulmaktadır. İlgili uygulama sayfası Şekil 3.37'de gösterilmektedir.
- ✓ BFC Gateway uygulamasına tezgâh IP adresleri tanımlanarak, tezgâh üzerinde kurulan istemci uygulaması ile haberleştirilir.
- ✓ BFC Gateway uygulaması üzerinden tezgâhtan alınacak verilerin data setleri tanımlanmaktadır.
- ✓ Verilerin üst katmanlarda kullanılabilmesi için OPC Ua protokolü aracılığı ile sunucudan veri yayını yapılmaktadır.



Şekil 3.37. BFC Gateway sunucu uygulama sayfası (Siemens, 2021:77)

3.5.3.5 MODBUS protokolünün devreye alınması

MODBUS protokolünün devreye alınabilmesi için süreç PLC programlaması adımından başlamaktadır. Üretici firma çoğunlukla MODBUS protokolü ile tezgâh üzerinde bulunan tüm donanımları haberleştirebilmektedir. Bu nedenle genellikle tezgahlar üzerinde MODBUS protokolü aktif olarak kullanılmaktadır. Ancak verilerin toplanabilmesi için data blok adreslerinin bilinmesi gerekmektedir.

Aktuel_Sa	Int	52.0	0
Aktuel_Dk	Int	54.0	0
Aktuel_Sn	Int	56.0	0
Toplam_Sa	Int	58.0	0
Toplam_Dk	Int	60.0	0
Toplam_Sn	Int	62.0	0
Aktuel_Durum	Int	64.0	0

Şekil 3.38. Örnek MODBUS adres tablosu

Bu adreslerin tespiti mümkün olmamaktadır. Bu nedenle üretici firma ancak verilerin data blok adreslerini dokümante etmiş ya da paylaşmış ise MODBUS ile veri toplanması mümkün olacaktır. Şekil 3.38'de MODBUS adresleri içeren bir örnek gösterilmektedir.

Fabrika ağına bağlı olan bir ekipman üzerinden MODBUS ile veri okunabilmesi için ek bir kurulum ya da devreye alma gerekmemektedir. Sunucu tarafından tezgâha fabrika ağı kullanılarak veri adreslerine sorgu atıldığında MODBUS protokolü yanıt vermektedir. Bu işlem sunucu ortamında bulunan ve MODBUS sürücüsüne sahip uygulamalar aracılığı ile gerçekleştirilmektedir.

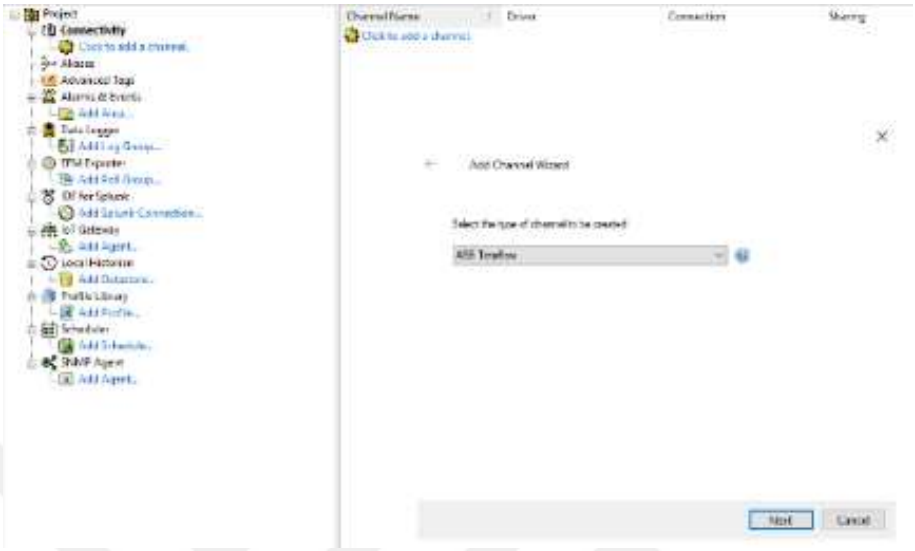
3.5.4. Verilerin sunucu ortamına aktarılması

Yukarıda bulunan bölümlerde anlatılan işlemler tamamlandığında tezgahlar, destekledikleri protokoller yardımı ile veri yayınlamaya başlamaktadır. Bu verilerin üst katmanlara taşınarak anlamlı bir hale getirilebilmesi, uygulamalarda kullanılabilmesi, depolanabilmesi ve görselleştirilebilmesi için fabrika ağına bağlı olan bir sunucu üzerinde verilerin toplanması gerekmektedir. Bu aşamada üçüncü parti bir yazılım olan Kepware Kepserver Ex yazılımı kullanılmaktadır. Bu yazılım sayesinde veri yayını yapan tezgahlardan veri isteminde bulunabilmektedir. Tezgahlardan bu yöntem ile toplanan veriler yine Kepware uygulaması üzerinden rest API yardımı ile JSON (JavaScript Object Notation) formatında anlık olarak yayınlanabilmektedir. Bu eklenti IoT Gateway olarak adlandırılmaktadır. Ayrıca tezgahlardan toplanan ham veriler depolanmak istenilirse Datalogger adı verilen eklenti aracılığı ile veriler istenilen veri tabanlarına doğrudan kaydedilebilmektedir. Aşağıda tezgahların Kepware yazılımına eklenmesi için gerekli adımlar anlatılmaktadır.

3.5.4.1 Tezgahların Kepware yazılımına eklenmesi

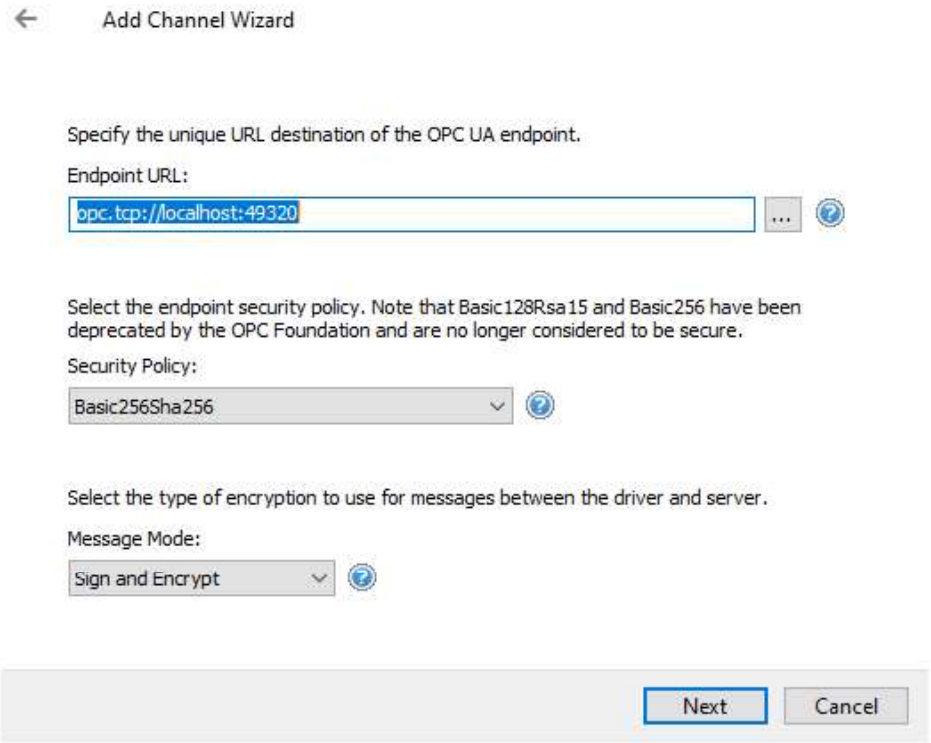
Kepware yazılımına ilgili tezgâhın eklenebilmesi için Kepware yazılımının bu protokolü destekleyen sürücülere sahip olması gerekmektedir. Kepware yazılımı birden fazla protokol için sürücü desteği sunmaktadır ve çalışma kapsamında kullanılan protokollerin tümüne Kepware yazılımı ile bağlantı kurulabilmektedir. Aşağıda Kepware yazılımına OPC UA protokolünü destekleyen bir tezgâhın nasıl eklendiği anlatılmaktadır. Ancak kullanılan diğer protokoller de benzer şekilde uygulamaya eklenmektedir. Kepware yazılımına eklenmesi istenilen tezgahların IP adresleri belirlenerek sunucu ile erişimi test edilmektedir. Erişim sıkıntısı olmadığı tespit edildiğinde kurulum adımlarına başlanılmaktadır.

- ✓ Öncelikle arayüz üzerinden “New Channel” kullanılarak tezgâh için yeni bir kanal eklenmektedir. Bu işlem Şekil 3.39’da görülmektedir.



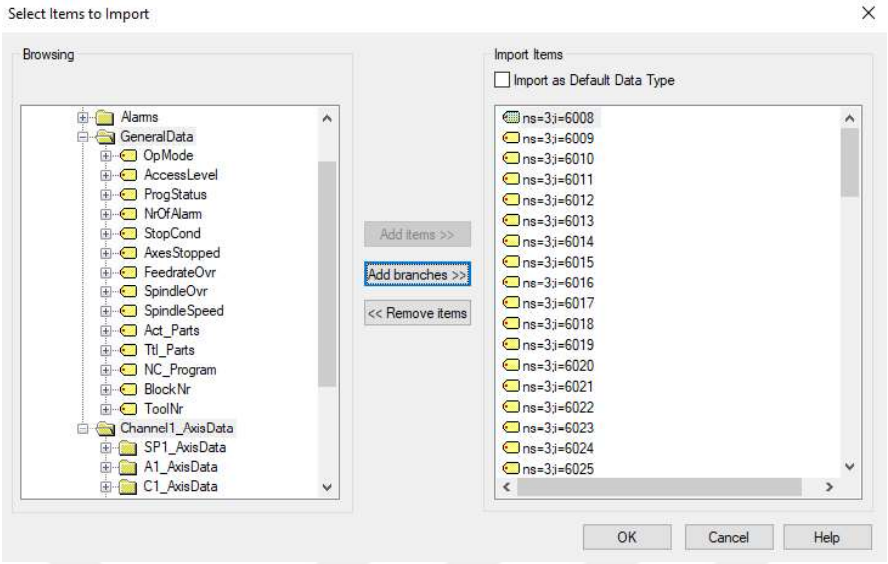
Şekil 3.39 Kepware yeni kanal eklenmesi

- ✓ Açılan sayfa üzerinden ilgili tezgâhın bağlı olduğu protokolün seçimi yapılmaktadır. OPC Ua protokolünü destekleyen bir tezgâh eklenmek istenildiğinde “OPC UA İstemci” seçimi yapılmaktadır.
- ✓ Ardından tezgâhın Kepware üzerinde görüntülenmesi istenilen tezgâh adı girilmektedir. Çalışmamızda OPC UA ile eklenen tezgâh “001_Breton_OPCUA” olarak adlandırılmaktadır.
- ✓ İlerleyen sayfada tezgâh üzerinde yapılan OPC UA konfigürasyonunda belirlenen IP adresi ve port adresi kullanılarak iletişim konfigürasyonu yapılmaktadır. Eğer tezgâh üzerinde yapılan konfigürasyonlarda şifreleme tipi seçilmişse bu bilgilerde bu sayfa üzerinde düzenlenmektedir. Şekil 3.40 da ilgili sayfa görünmektedir.



Şekil 3.40. Kepware OPC UA konfigürasyonu

- ✓ Tezgâh konfigürasyonu sırasında OPC UA yayını için belirlenen kullanıcı adı ve şifre sonraki sayfada eklenmektedir ve ardından tezgâh için ilgili OPC UA kanalı oluşturulmaktadır.
- ✓ Kanal üzerinden tezgâh verilerine erişilebilmesi için bir cihaz eklenmesi gerekmektedir. Channel üzerinden "New Device" seçilerek cihazın eklenmesine başlanmaktadır.
- ✓ Son olarak açılan sayfada "Select Import Items" butonuna tıklanarak tezgâh üzerinde bulunan kontrolcünün anlık olarak yayın yaptığı verilere erişilebilmektedir. Açılan sayfa kullanılarak toplanması istenilen veriler Kepware uygulamasına eklenmektedir. İlgili ekran Şekil 3.41'de gösterilmektedir.



Şekil 3.41. Kepware veri seçimi

✓ Ardından kurulum tamamlanmaktadır ve Kepware kullanılarak veriler canlı olarak izlenebilmektedir. (Şekil 3.42)

Item ID	Data Type	Value	Timestamp	Quality
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.Ttl_Parts	Double	876	16:46:19.241	Good
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.ToolNr	String	7	16:46:19.236	Good
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.StopCond	Word	0	17:11:29.313	Good
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.SpindleSpeed	Double	20001.8	17:13:25.271	Good
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.SpindleOvr	Double	100	16:46:19.241	Good
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.ProgStatus	Word	3	16:38:08.314	Good
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.OpMode	Word	2	16:37:59.260	Good
001_Breton_OPCUA.Siemens.647-03_BRETON.GeneralData.NrOfAlarm	Word	0	13:44:16.386	Good

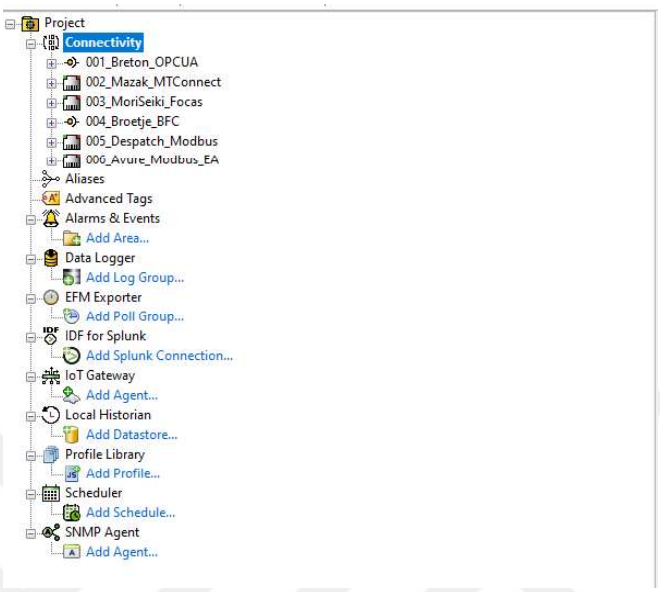
Şekil 3.42. Kepware üzerinde verilerin anlık görüntülenmesi

3.5.4.2. Tezgâh verilerinin IoT Gateway servisine eklenmesi

IoT Gateway servisi Kepware yazılımının bir eklentisi olarak satın alınmaktadır. IoT Gateway servisi kullanılarak, tezgahlardan toplanan anlık verinin istenilen zaman aralıklarında Rest servisi olarak ya da MQTT protokolü kullanılarak verilerin üst katmanlarda kullanılabilmesi için iletişim sağlanmaktadır. Çalışmamız kapsamında verilerin üst katmanlarda anlamlandırılarak standart bir hale getirilebilmesi amacıyla IoT Gateway servisi kullanılmaktadır. Çalışmada IoT Gateway Rest servisi olarak çalışmakta ve JSON formatında yayın yapmaktadır. Aşağıda IoT Gateway konfigürasyonlarının nasıl yapıldığı anlatılmaktadır.

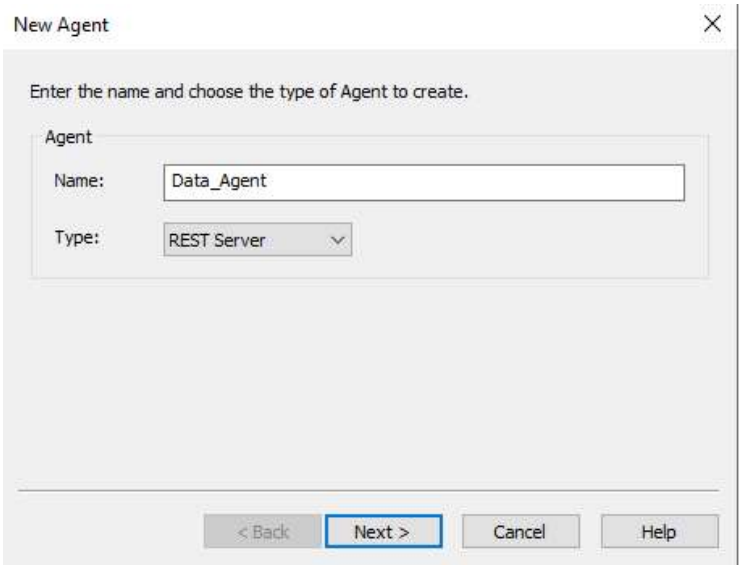
✓ İlk olarak Kepware aracılığıyla verilerin üst katmanlara iletilmesi istenilen tezgahların Kepware'e eklenmiş olması gerekmektedir. Proje klasörünün altında IoT

Gateway eklentisi bulunmaktadır. Buradan yeni agent eklenmelidir. Şekil 3.43'te Kepware proje dosyası gösterilmektedir.



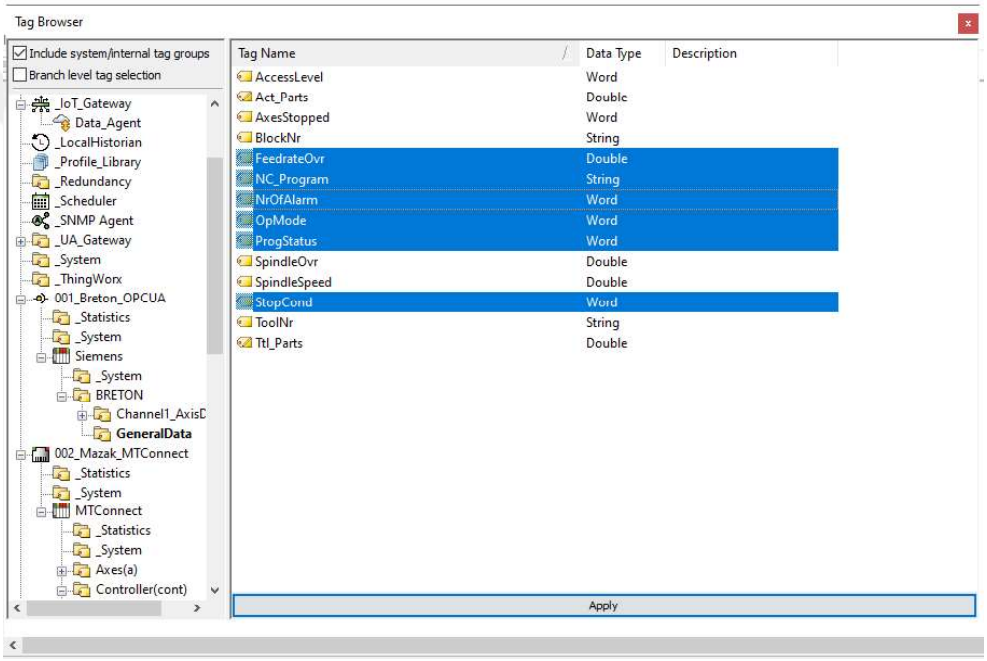
Şekil 3.43. IoT Gateway ve tezgâh bağlantıları

✓ Açılan pencerede oluşturulacak agent'ın adı ve tipi seçilmelidir. Çalışmada veriler Rest Server kullanılarak üst katmanlara aktarılmaktadır. Şekil 3.44'te yeni agent ekleme sayfası gösterilmektedir.



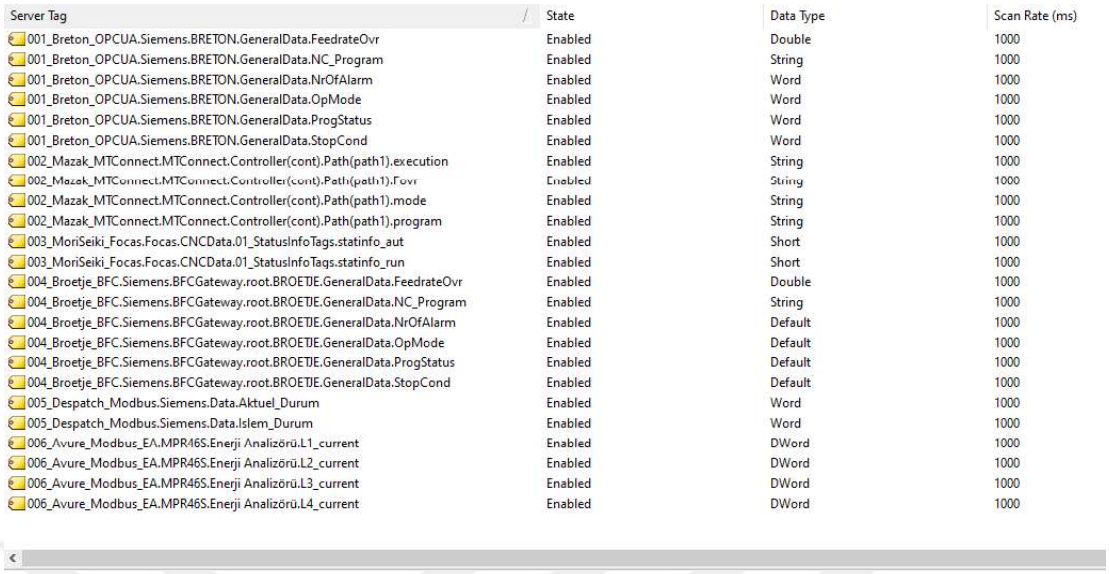
Şekil 3.44. Yeni agent ekleme sayfası

- ✓ İlerleyen sayfada Rest Server'ın yayın yapacağı network ayarları yapılandırılmaktadır. Eklenen her bir agent farklı port adresine sahip olmalıdır. Port adreslerinin çakışması durumunda eklenen agentlar çalışmamaktadır.
- ✓ Gerekli konfigürasyonlar tamamlandığında agent başarılı bir şekilde eklenmiş olacaktır.
- ✓ Agent üzerinden verilerin üst katmanlara aktarılabilmesi için istenilen verilerin agenta eklenmesi gerekmektedir. Şekil 3.44'te verilerin agenta nasıl eklendiği ve Şekil 3.45'te verilerin nasıl seçildiği gösterilmektedir.



Şekil 3.45. IoT Gateway agenta verilerin eklenmesi

- ✓ IoT Gateway ile yayınlanacak veriler eklenildiğinde, Rest Server yayın yapmaya başlayacaktır.
- ✓ Çalışma kapsamında IoT Gateway ile yayınlanan veriler Şekil 3.46'da gösterilmektedir.



Server Tag	State	Data Type	Scan Rate (ms)
001_Breton_OPCUA.Siemens.BRETON.GeneralData.FeedrateOvr	Enabled	Double	1000
001_Breton_OPCUA.Siemens.BRETON.GeneralData.NC_Program	Enabled	String	1000
001_Breton_OPCUA.Siemens.BRETON.GeneralData.NrOfAlarm	Enabled	Word	1000
001_Breton_OPCUA.Siemens.BRETON.GeneralData.OpMode	Enabled	Word	1000
001_Breton_OPCUA.Siemens.BRETON.GeneralData.ProgStatus	Enabled	Word	1000
001_Breton_OPCUA.Siemens.BRETON.GeneralData.StopCond	Enabled	Word	1000
002_Mazak_MTConnect.MTConnect.Controller(cont).Path(path1).execution	Enabled	String	1000
002_Mazak_MTConnect.MTConnect.Controller(cont).Path(path1).Ovr	Enabled	String	1000
002_Mazak_MTConnect.MTConnect.Controller(cont).Path(path1).mode	Enabled	String	1000
002_Mazak_MTConnect.MTConnect.Controller(cont).Path(path1).program	Enabled	String	1000
003_MoriSeiki_Focas.Focas.CNCData.01_StatusInfoTags.statinfo_aut	Enabled	Short	1000
003_MoriSeiki_Focas.Focas.CNCData.01_StatusInfoTags.statinfo_run	Enabled	Short	1000
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.FeedrateOvr	Enabled	Double	1000
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.NC_Program	Enabled	String	1000
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.NrOfAlarm	Enabled	Default	1000
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.OpMode	Enabled	Default	1000
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.ProgStatus	Enabled	Default	1000
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.StopCond	Enabled	Default	1000
005_Despatch_Modbus.Siemens.Data.Aktuel_Durum	Enabled	Word	1000
005_Despatch_Modbus.Siemens.Data.Islem_Durum	Enabled	Word	1000
006_Avure_Modbus_EA.MPR46S.Enerji Analizörü.L1_current	Enabled	DWord	1000
006_Avure_Modbus_EA.MPR46S.Enerji Analizörü.L2_current	Enabled	DWord	1000
006_Avure_Modbus_EA.MPR46S.Enerji Analizörü.L3_current	Enabled	DWord	1000
006_Avure_Modbus_EA.MPR46S.Enerji Analizörü.L4_current	Enabled	DWord	1000

Şekil 3.46. IoT Gateway ile yayınlanan veriler

IoT Gateway kullanılarak, çalışmanın ilerleyen kısımlarında anlamlandırılarak kullanılabilirlik verilerinin elde edilmesinde kullanılacak veriler üst katmanlara aktarılmaktadır. Bu veriler 1000 ms aralıklarla yenilenmektedir. Toplanan verilerin isimleri ve veri tipleri Şekil 3.46'da görülmektedir. Oluşan bu çeşitlilik yapılması gereken standardizasyon çalışmasının önemini göstermektedir.

3.6. Üretim Verilerinin İncelenmesi ve Anlamlandırılması

Çalışmanın bu bölümüne kadar anlatılan kısımlar IIoT altyapısı, protokol ve yazılımları kullanılarak verilerin toplanmasını sağlamaktadır. Kepware uygulamasına ulaşan veriler beşinci başlık altında anlatılan IIoT katmanlarından dördüncüsü olan sunucu katmanına ulaşmış olmaktadır. Çalışmanın bu bölümünden sonrasında toplanan verilerin incelenmesi, anlamlandırılması ve görselleştirilmesi anlatılmaktadır. Toplanan verilerin son kullanıcının anlayabileceği duruma getirilmesi için yapılan çalışma, anlamlandırma çalışması olarak adlandırılmaktadır. Anlamlandırma çalışmasının en doğru şekilde yapılabilmesi için öncelikle tezgahlardan toplanan ham verilerin incelenmesi gerekmektedir.

Ham verinin ne olduğu tespit edildikten sonra doğru anlamlandırmanın yapılabilmesi için alanda tezgahlar incelenerek, hangi durumlarda hangi verileri ürettiği ve hangi veriler kullanılarak anlamlı sonuçlar elde edilebileceği araştırılmaktadır. Aşağıda kullanılan

protokoller aracılığı ile toplanan veriler incelenmektedir. Protokol değişmediği sürece genellikle veriler aynı isimlendirme ve aynı veri tipi ile toplanmaktadır. Bu nedenle çalışma ilerleyen kısımlarda tezgâh bazında değil protokol bazında gruplandırılmaktadır. CNC/NC grubunda yer alan tezgahlardan toplanan ham verilerden çalışmamız kapsamında elde edilmek istenilen verinin tezgâh kullanılabilirliğine hizmet etmesi beklenmektedir. Bu nedenle ilk olarak tezgahların çalışma durumlarının belirlenmesi gerekmektedir. Sahada tezgahların çalışma durumları gözlemlenerek, tezgahların %99 oranında otomatik modda çalıştığı belirlenmiştir. Tezgâhın otomatik modda çalışması üretilecek olan parçaya göre hazırlanmış olan NC programın tezgâha yüklenerek tezgâhın üretimi otomatik olarak gerçekleştirdiği durumdur. Dolayısıyla tezgahların çalışma modu takip edilirken beraberinde tezgahların üzerinde çalışan programların anlık durumlarının da kontrol edilmesi gerekmektedir. Ayrıca tezgahların çalıştığı durumlarda eksenlerin hareketini engelleyen bir durum ile karşılaşılmaması beklenmektedir. Bu nedenle eksenlerin ilerleme hızı sıfır olarak ayarlanamamaktadır. Bazı durumlarda tezgahların durdurulması için ilerleme hızı sıfıra alınarak tezgahlar durdurulabilmektedir. Bu sebeple tezgahların ilerleme hızı da çalışma durumunu etkileyen veriler arasında yer almaktadır. Çalışma kapsamında CNC/NC tezgâh grubunda yer alan 001_Breton_OPCUA, 002_Mazak_MTConnect, 003_MoriSeiki_FOCAS, 004_Broetje_BFC tezgahlarında çalışma durumları yukarıda anlatıldığı gibidir. Ancak farklı protokoller aracılığı ile veri toplanması sebebi ile verilerde farklılıklar meydana gelmektedir. Bu nedenle aşağıda veriler bu çalışmaya hizmet edecek şekilde anlamlandırılmaktadır.

Konvansiyonel tezgahlar grubunda yer alan ve MODBUS protokolü kullanılarak veri toplanan 005_Despatch_MODBUS fırınında ise çalışma durumu, ekipmanların yapısal farklılıkları nedeni ile CNC/NC tezgâh grubunda yer alan tezgahlara göre farklılık göstermektedir. Sahada yapılan incelemeler sonucunda fırının çalışması gereken sıcaklık ayarları operatör ekranı üzerinden belirlenmektedir. Fırın çalışma yaparken iki farklı işlem uygulamaktadır. Bu işlemler rampa ve soaking olarak adlandırılmaktadır. Rampa işlemi, fırının operatör tarafından belirtilen sıcaklık değerine ulaşması için gerekli olan işlemdir. Soaking işlemi ise fırın üzerinde işleme alınan parçanın istenilen sıcaklıkta kalması gereken süre boyunca sabit ısıda tutulması işlemidir. Bazı parçalar iki aşamalı soaking işlemi uygulanmaktadır. Çalışma durumu yukarıda anlatıldığı gibi belirlenen

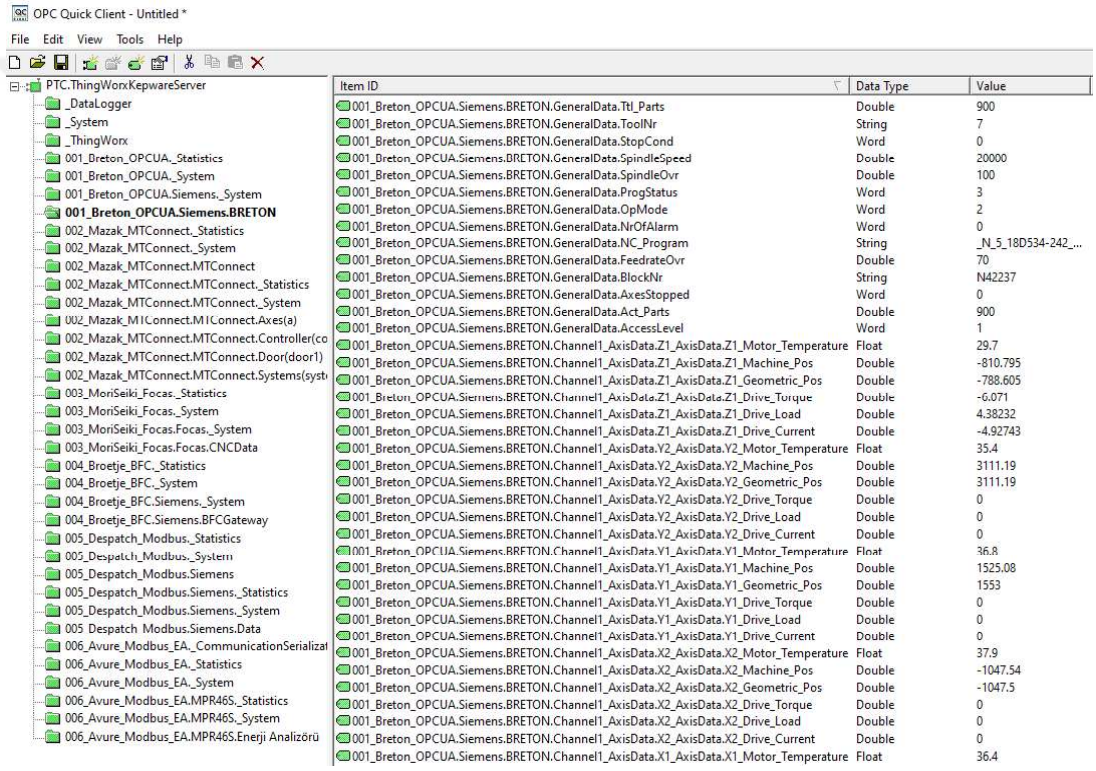
fırının aynı zamanda yaptığı işlemin ne durumda olduğu da kontrol edilmelidir. Çünkü rampa ya da soaking sırasında sıcaklık ya da operatör kaynaklı süreç durabilmektedir. Bu nedenle bu bilginin de kontrol edilmesi kritiktir.

Konvansiyonel tezgahlar grubunda yer almasına rağmen, üzerinde bulunan PLC'nin MODBUS protokolü ile yayın yapmaması ya da data blok adreslerine ulaşılamaması gibi nedenler ile manuel bir tezgâh gibi değerlendirilmiş olan 006_Avure_MODBUS_EA adlı pres tezgahında, tezgâh üzerine yerleştirilmiş olan enerji analizöründen veriler toplanmaktadır. Bu nedenle tezgâhın hangi durumlarda çalıştığından çok, çalıştığı durumlarda ne gibi enerji verileri tükettiği önemlidir. Tezgâhın çalışma durumlarında harcayacağı enerji, tezgâhın çektiği akım miktarına bağlıdır. Bu nedenle enerji analizörü üzerinde bulunan akım verileri kontrol edilmektedir. İncelemeler sonucunda tezgâhın çalıştığı durumlarda yüksek akımlar çektiği gözlemlenmiştir. Tezgâh kapalı durumda iken 0 A akım çekmektedir, duruyorken değişiyor olmakla beraber genellikle 1-5 A aralığında akım çekerken çalışıyorken 50-60 A civarında akım çekmektedir. Bu nedenle tezgâhın 5 A'den büyük akım çekmesi durumunda tezgâh çalışıyor olarak anlamlandırılmaktadır. Bu tezgâhtan enerji analizörü aracılığı ile veri toplandığı için yukarıda anlatılan veriler elde edilebilmektedir. Diğer tezgahlarda olduğu gibi detay bilgilere ulaşılmak istenilirse, verilerin elde edilebileceği uygun konumlara uygun sensörlerin yerleştirilmesi ya da PLC üzerinde bulunan kontakların tespit edilerek alınacak paralel bağlantılar ile verilerin toplanması gerekmektedir.

3.6.1. OPC UA ile toplanan üretim verileri

Tezgahlardan OPC UA protokolü kullanılarak verilerin nasıl toplanabileceği önceki bölümlerde anlatılmaktadır. Siemens marka kontrolcüye sahip olan Breton marka bir beş eksenli dikey işleme tezgahından OPC UA protokolü ile veriler toplanmaktadır. Bu aşamada tezgâhın yayınlamış olduğu veriler önem kazanmaktadır. İlk etapta OPC UA ile yayınlanan veriler incelendiğinde tezgâhtan ulaşılabilen yedi bin adet veri bulunmaktadır. Ancak bu verilerin bir kısmına üretici firmanın koyduğu engeller sonucunda ulaşılamamaktadır ve tüm verilerin toplanması ikili verilerin oluşmasına ve kullanılmayacak verilerin gereksiz yer kaplamasına neden olabilmektedir. Bu nedenle verilerin ne olduğu incelenerek öncelikli toplanması gereken veriler tespit edilmektedir.

Tezgâh tipi, protokol ve üretici firma bazında değişiklik gösterebilmekle beraber çalışma kapsamında tezgâhın kullanılabilirliğini belirleyebilecek, çalışma durumlarının elde edilebileceği ve ilerleyen zamanlarda fayda sağlayabileceği ön görülen veriler öncelikli olarak seçilmektedir. Bu veriler OPC UA protokolü çerçevesinde tezgâhın çalışma modu, alarm sayısı, duruş bilgileri, iş mili ve eksen hızları, çalışan program adı, tool numarası, üretilen parça sayısı, eksenlerin durumu, pozisyonları, yük, akım ve tork bilgileri olmaktadır. Şekil 3.47'de çalışma kapsamında bulunan 001_Breton_OPCUA tezgahından toplanan ham veriler gösterilmektedir.



Şekil 3.47. 001_Breton_OPCUA tezgahından toplanan anlık veriler

Yukarıda OPC UA protokolü ile verileri toplanan 001_Breton_OPCUA adlı tezgâhın hangi durumlarda çalıştığı anlatılmıştır. Bu nedenle tezgâhtan toplanan veriler arasından tezgâhın çalışma modu, programın anlık durumu, eksenin ilerleme hızı ve eğer elde edilebiliyor ise tezgâhın neden durduğu ile ilgili veriler önceliklidir. Yapılan araştırmalar ve tezgâh üzerindeki incelemeler sonucunda OPC UA protokolü ile toplanan verilerden sırası ile opMode, progStatus, feedrateOvr, stopCond adlı veriler öncelikli veri ihtiyacını karşılamaktadır. Tablo 3.4'te toplanan bu veriler ve ne anlama geldikleri açıklanmaktadır.

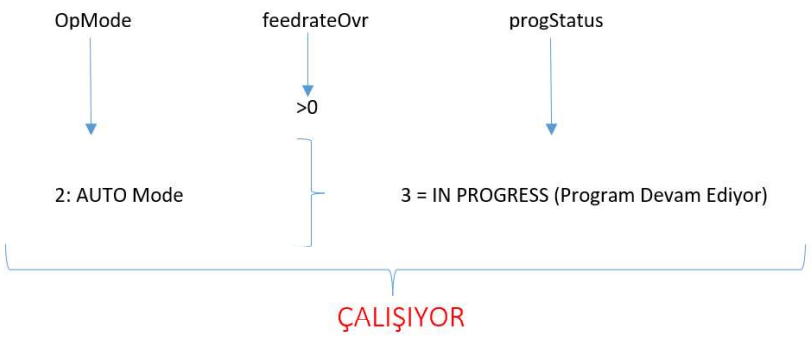
Tablo 3.4. OPC UA verileri ve anlamları

Veri adı	Veri tipi	Veri anlamı	Verinin karşılığı
Opmode	Word	Tezgâhın anlık modunu göstermektedir.	0 = Adım Kontrol (JOG) Modu 1 = Makine Veri Giriş (MDI) Modu 2 = Otomatik (AUTO) Modu
Progstatus	Word	Tezgâh üzerinde çalışan programın anlık durumunu göstermektedir.	1 = Program Kesildi 2 = Program Durdu 3 = Program Devam Ediyor 4 = Program Bekliyor 5 = Program İptal Edildi
Feedrateovr	Double	Tezgâh eksenlerinin ilerleme hızını göstermektedir.	0 ile 120 aralığında değişen veriler okunmaktadır. Değerler yüzde olarak değerlendirilmektedir.
Stopcond	Word	Tezgâh durduğunda oluşan duruş kodlarını göstermektedir.	0 ile 71 arasında değişen duruş kodu bulunmaktadır. 0 = Duruş Durumu Yok 1 = NC Hazır Değil 2 = İşlem Modu Hazır Değil 3 = Acil Duruş Aktif 4 = NC Duruşu ile Alarm Aktif 5 = M0 / M1 Kod Bloğu Aktif …

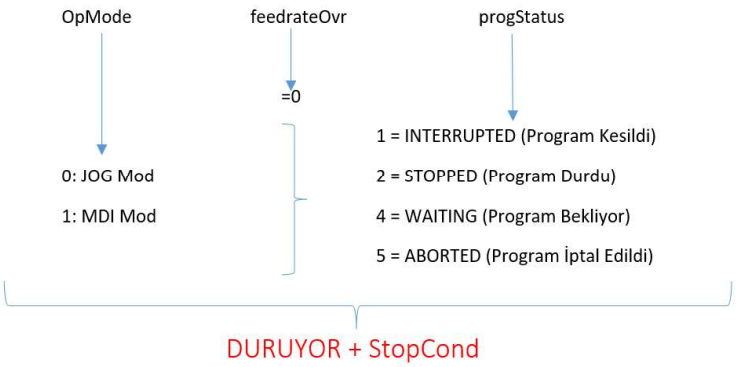
Elde edilen veriler ve anlamları incelendiğinde tezgâhın çalışma durumu;

- opMode = 2
- progStatus = 3
- Feedrate> 0

olarak belirlenmiştir ve diğer tüm durumlarda tezgâh durmaktadır. Tezgâhın durduğu durumlarda ise duruş kodu stopCond verisi üzerinden alınabilmektedir. Tezgâhın çalışma durumu Şekil 3.48'de, tezgâhın duruş durumu ise Şekil 3.49'da gösterilmektedir.



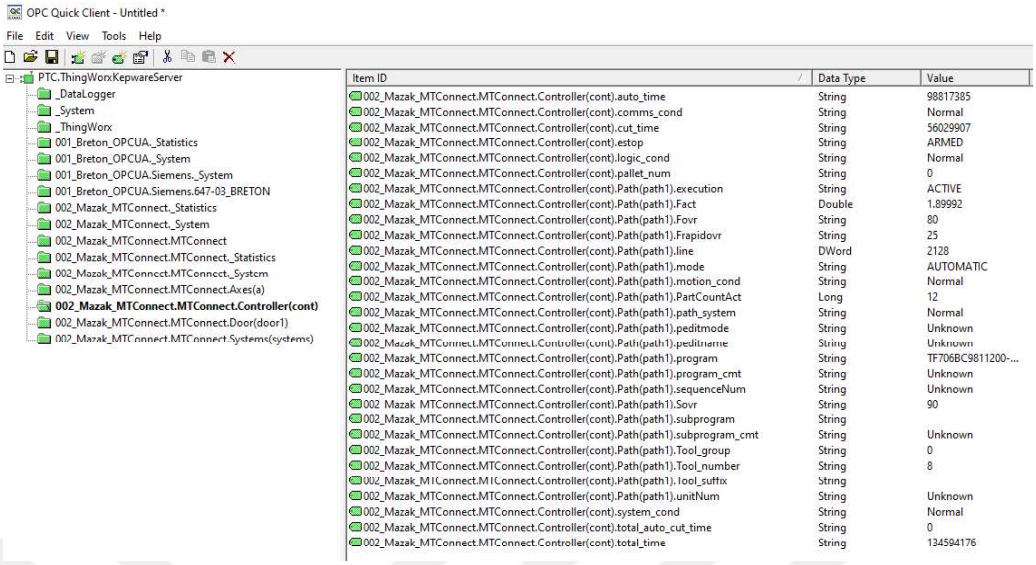
Şekil 3.48. OPC UA çalışma durumu diyagramı



Şekil 3.49. OPC UA duruş durumu diyagramı

3.6.2. MTConnect ile toplanan üretim verileri

MTConnect Protokolü kullanılarak, protokol desteği sunan tezgahlardan nasıl veri toplanabileceği önceki bölümlerde anlatılmaktadır. Kontrolcü ve tezgâh markası Mazak olan beş eksenli dikey işleme tezgâhı olan 002_Mazak_MTConnect tezgahından MTConnect protokolü kullanılarak veri toplanmaktadır. MTConnect protokolü ile veri toplanmak istenildiğinde kurulan agent üzerinde çalışan config dosyası ile belirlenen verilere erişilebilmektedir. Bu veriler MTConnect kurulumu sırasında config dosyasına eklenmektedir. Config dosyasına eklenecek veriler önceliklendirilmelidir. Bu durum tezgâh kontrolcüsünde yaşanacak sorunların ve ağ üzerinde yaşanabilecek gecikme problemlerinin önüne geçilmesini sağlamaktadır. Bu nedenle bu tezgâh ile aynı grupta yer alan 001_Breton_OPCUA tezgahında olduğu gibi tezgâhın çalışma modu, alarm sayısı, duruş bilgileri, iş mili ve eksen hızları, çalışan program adı, tool numarası, üretilen parça sayısı, eksenlerin durumu, pozisyonları, yük, akım ve tork verilerinin toplanması önemlidir. Şekil 3.50'de 002_Mazak_MTConnect tezgahından elde edilen ham veriler gösterilmektedir.



Şekil 3.50. 002_Mazak_MTConnect tezgahından toplanan anlık veriler

Yukarıda CNC/NC grubunda yer alan tezgahların çalışma durumları anlatılmaktadır. MTConnect protokolü ile verileri toplanan 002_Mazak_MTConnect adlı tezgâhtan, OPC Ua tezgahında olduğu gibi, toplanan veriler arasında tezgâhın çalışma modu, programın anlık durumu, eksenin ilerleme hızı ve eğer elde edilebiliyor ise tezgâhın neden durduğu ile ilgili verilerin bulunması gerekmektedir. Yapılan araştırmalar ve tezgâh üzerindeki incelemeler sonucunda MTConnect protokolü ile toplanan verilerden sırası ile mode, execution, fovr adlı veriler öncelikli veri ihtiyacını karşılamaktadır. Ancak MTConnect protokolü ile toplanan veriler arasında duruş durumu ile ilgili veriler bulunamamıştır. Tablo 3.5'te toplanan bu veriler ve ne anlama geldikleri açıklanmaktadır.

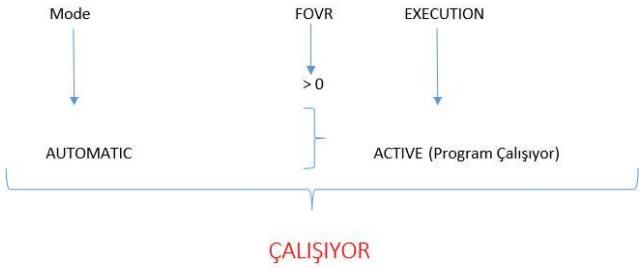
Tablo 3.5. MTConnect verileri ve anlamları

Veri adı	**Veri tipi**	**Veri anlamı**	**Verinin karşılığı**
Mode	String	Tezgâhın anlık modunu göstermektedir.	MANUAL MODE (Manuel Mod) MANUAL_DATA_INPUT MODE (Manuel Veri Giriş Modu) SEMI_AUTOMATIC MODE (Yarı Otomatik Mod) AUTOMATIC MODE (Otomatik Mod) EDIT MODE (Düzenleme Modu)
Execution	String	Tezgâh üzerinde çalışan programın anlık durumunu göstermektedir.	STOPPED (Tezgâh Duruyor) READY (Tezgâh Hazır) PROGRAM STOPPED (Program Duruyor) FEED HOLD (Tezgâh Beklemede) ACTIVE (Program Çalışıyor)
Fovr	String	Tezgâh eksenlerinin ilerleme hızını göstermektedir.	0 ile 120 aralığında değişen veriler okunmaktadır. Değerler yüzde olarak değerlendirilmektedir.

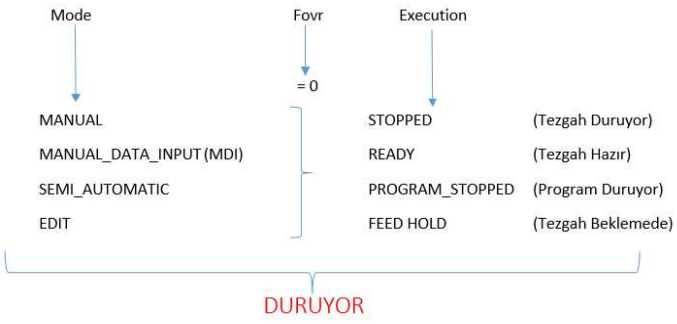
Elde edilen veriler ve anlamları incelendiğinde tezgâhın çalışma durumu;

- Mode = AUTOMATIC
- Execution = ACTIVE
- Fovr> 0

olarak belirlenmiştir ve diğer tüm durumlarda tezgâh durmaktadır. MTConnect aracılığı ile toplanan verilerin tipi string olması nedeni ile veriler metin tabanlı olarak toplanmaktadır. Tezgâhın çalışma durumu Şekil 3.51'de tezgâhın durma durumu ise Şekil 3.52'de gösterilmektedir.



Şekil 3.51. MTConnect çalışma durumu diyagramı

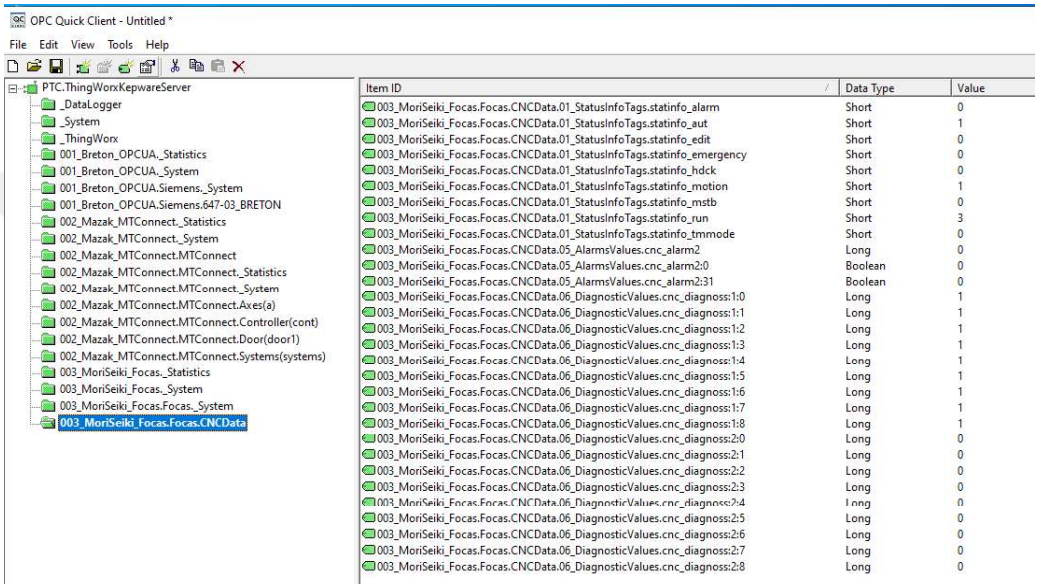


Şekil 3.52. MTConnect duruş durumu diyagramı

3.6.3. FOCAS ile toplanan üretim verileri

FOCAS protokolü kullanılarak, protokol desteği sunan tezgahlardan nasıl veri toplanabileceği önceki bölümlerde anlatılmaktadır. Mori Seiki markasının Fanuc marka kontrolcüsüne sahip üç eksen torna tezgâhı olan bu tezgâhtan FOCAS protokolü kullanılarak veri toplanmaktadır. FOCAS protokolü ile erişilebilecek veriler sınırlı olabilmektedir. Bu durum tezgâh üreticisinin yapmış olduğu parametrik ayarlar sonucunda değişiklik gösterebilmektedir. Ancak genellikle OPC UA ya da MTConnect

protokollerinde olduğu gibi geniş yelpazede veri yayını bulunmamaktadır. Ancak FOCAS protokolü aracılığı ile tezgâh çalışma durumu, program düzenleme durumu, alarm durumları, acil buton durumu, programın anlık durumu, eksen hareketleri gibi önemli verilere ulaşılabilmektedir. Bunların haricinde parametrik adreslerinin bilinmesi durumunda tezgâh ile ilgili diğer verilere de ulaşılabilmektedir. Fakat bu adreslerin belirlenmesi yoğun çaba gerektirmektedir.



Şekil 3.53. 003_MoriSeiki_FOCAS tezgahından toplanan anlık veriler

Yukarıda CNC/NC grubunda yer alan tezgahların çalışma durumları anlatılmaktadır ve Şekil 3.53'te ilgili tezgâhtan toplanan anlık veriler gösterilmektedir. FOCAS protokolü ile verileri toplanan 003_MoriSeiki_FOCAS adlı tezgâhtan, CNC/NC grubunda yer alan diğer tezgahlarda olduğu gibi tezgâhın çalışma modu, programın anlık durumu, eksenin ilerleme hızı ve eğer elde edilebiliyor ise tezgâhın neden durduğu ile ilgili verilerin toplanması gerekmektedir. Tezgâh üzerindeki incelemeler sonucunda FOCAS protokolü ile toplanan verilerden sırası ile statinfo_aut ve statinfo_run adlı veriler öncelikli veri ihtiyacını karşılamaktadır. Ancak FOCAS protokolü ile toplanan veriler arasında duruş durumu ve eksen ilerleme hızı ile ilgili veriler bulunamamıştır. Anlık ilerleme hızının sıfır olması tezgâhın çalışma durumunu etkilemektedir. Ancak Fanuc kontrolcüye sahip bu tezgâh üzerinde eksen hızının sıfır olması durumunda tezgâhın beklemeye geçtiği görülmektedir. Bu nedenle eksen ilerleme hızına ek olarak ihtiyaç kalmamaktadır. Tablo 3.6'da toplanan bu veriler ve ne anlama geldikleri açıklanmaktadır.

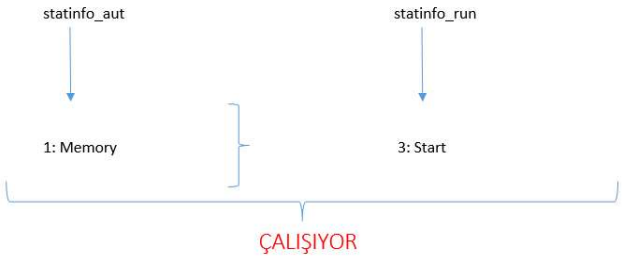
Tablo 3.6. FOCAS verileri ve anlamları

Veri adı	Veri tipi	Veri anlamı	Verinin karşılığı
Statinfo_aut	Short	Tezgâhın anlık modunu göstermektedir.	0 = Manuel Veri Giriş Modu 1 = Hafıza Modu Aynı Zamanda Otomatik Mod 2= Düzenleme Modu 3 = Manivela Kontrol Modu 4 = Adım Kontrol Modu 5 = Elle Öğretme Modu 6 = Manivela İle Öğretme Modu 7 = Artan Besleme Modu 8 = Referanslandırma Modu 9 = Uzaktan Kumanda Modu 10 = Test İşlem Modu
Statinfo_run	Short	Tezgâh üzerinde çalışan programın anlık durumunu göstermektedir.	0 = Program Resetlenmiş 1 = Program Durdurulmuş 2 = Program Beklemede 3 = Program Başlatılmış 4 = Tool Pozisyonlama

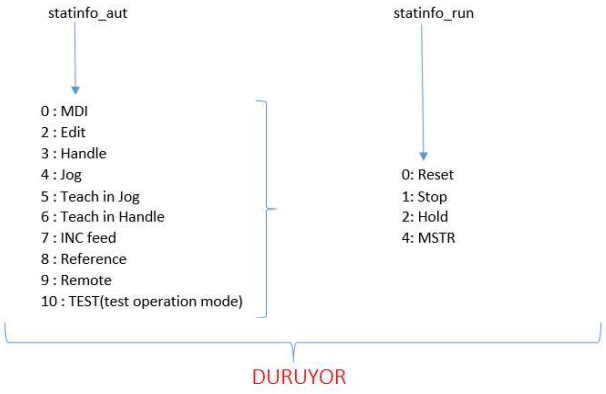
Elde edilen veriler ve anlamları incelendiğinde tezgâhın çalışma durumu;

- statinfo_aut = 1
- statinfo_run = 3

olarak belirlenmiştir ve diğer tüm durumlarda tezgâh durmaktadır. Tezgâhın çalışma durumu Şekil 3.54'te, tezgâhın durma durumu ise Şekil 3.55'te gösterilmektedir.



Şekil 3.54. FOCAS çalışma durumu diyagramı



Şekil 3.55. FOCAS duruş durumu diyagramı

3.6.4. BFC Gateway ile toplanan üretim verileri

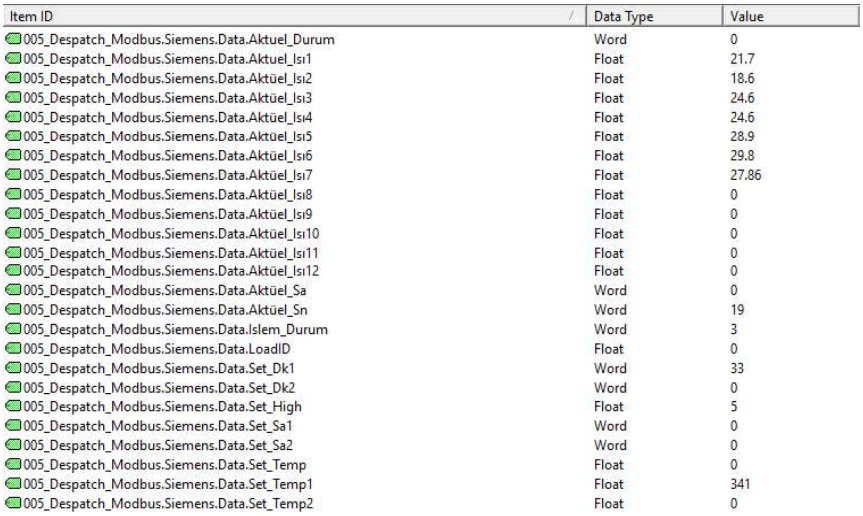
BFC Gateway kullanılarak veri toplanan tezgâh Siemens marka kontrolcüye sahip, Broetje markasının C-frame panel montaj tezgahıdır. Bu tezgâh her ne kadar versiyon kaynaklı OPC Ua desteği sunmasa bile kontrolcü altyapısı kaynaklı 001_Breton_OPCUA tezgahından toplanan veriler ile BFC Gateway ile bu tezgâhtan toplanan veriler aynı yapıya sahiptir. Bu nedenle tezgâh verilerinin anlamlandırılması OPC Ua protokolü ile toplanan veriler ile aynı anlamlandırmaya sahiptir. BFC Gateway ile toplanan üretim verileri 3.6.1 başlığında anlatılan çalışma kapsamında değerlendirilmektedir ve Şekil 3.56'da ilgili tezgâhtan toplanan anlık veriler gösterilmektedir.

Item ID	Data Type	Value
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.Online	Boolean	1
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.MachineStatus.PowerOn	Boolean	0
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.ToolNr	String	
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.StopCond	LLong	8
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.SpindleOvr	Double	0
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.Spindle_Speed	Double	0
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.ProgStatus	LLong	3
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.OpMode	LLong	2
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.NrOfAlarm	LLong	0
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.NC_Program	String	_N_S18_LS_S1_CP2...
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.FeedrateOvr	Double	120
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.BlockNr	String	N160
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.AxesStopped	Boolean	1
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.Act_Parts	LLong	0
004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.AccessLevel	LLong	6

Şekil 3.56. 004_Broetje_BFC tezgahından toplanan anlık veriler

3.6.5. MODBUS ile toplanan üretim verileri

Konvansiyonel tezgâh grubunda yer alan ve MODBUS protokolü kullanılarak veri toplanan 005_Despatch_MODBUS adlı fırının çalışma durumu yukarıda anlatılmaktadır. MODBUS ile toplanan veriler üretici firmaların isimlendirme ve verileri adreslemelerine göre çeşitlilik göstermektedir. Bu nedenle anlatılan veriler ve anlamlandırma çalışması sadece ilgili tezgâh için geçerli olmaktadır. MODBUS protokolü kullanılarak yeni bir tezgâh çalışmaya dahil edilmek istenildiğinde, anlamlandırma çalışmasının tekrarlanması gerekmektedir. Bu kapsamda MODBUS protokolü kullanılarak önemli olarak değerlendirilen, anlık sıcaklık ölçümleri, aktif adımda geçen süre, çalışmanın başından itibaren geçen süre, belirlenen sıcaklık değerleri, anlık çalışma ve işlem durumları, soaking ve rampa adımlarında geçen süreler ve istenilen sıcaklık değerleri gibi verilere ulaşılabilmektedir. Şekil 3.57'de fırından toplanan anlık veriler gösterilmektedir.



Item ID	Data Type	Value
005_Despatch_Modbus.Siemens.Data.Aktuel_Durum	Word	0
005_Despatch_Modbus.Siemens.Data.Aktuel_Isı1	Float	21.7
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı2	Float	18.6
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı3	Float	24.6
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı4	Float	24.6
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı5	Float	28.9
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı6	Float	29.8
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı7	Float	27.86
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı8	Float	0
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı9	Float	0
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı10	Float	0
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı11	Float	0
005_Despatch_Modbus.Siemens.Data.Aktüel_Isı12	Float	0
005_Despatch_Modbus.Siemens.Data.Aktüel_Sa	Word	0
005_Despatch_Modbus.Siemens.Data.Aktüel_Sn	Word	19
005_Despatch_Modbus.Siemens.Data.Islem_Durum	Word	3
005_Despatch_Modbus.Siemens.Data.LoadID	Float	0
005_Despatch_Modbus.Siemens.Data.Set_Dk1	Word	33
005_Despatch_Modbus.Siemens.Data.Set_Dk2	Word	0
005_Despatch_Modbus.Siemens.Data.Set_High	Float	5
005_Despatch_Modbus.Siemens.Data.Set_Sa1	Word	0
005_Despatch_Modbus.Siemens.Data.Set_Sa2	Word	0
005_Despatch_Modbus.Siemens.Data.Set_Temp	Float	0
005_Despatch_Modbus.Siemens.Data.Set_Temp1	Float	341
005_Despatch_Modbus.Siemens.Data.Set_Temp2	Float	0

Şekil 3.57. 005_Despatch_MODBUS fırınından toplanan anlık veriler

Yukarıda konvansiyonel tezgâh olarak çalışma durumu anlatılan bu tezgâh için tezgâhın hangi işlemleri gerçekleştirdiği ve o işlemler sırasında hangi durumlarda kaldığının tespit edilmesi gerekmektedir. Yapılan incelemeler sonucunda sırası ile Aktuel_Durum ve İşlem_Durum verileri çalışma durumunun elde edilebilmesi için gereken öncelikli veri ihtiyacını karşılamaktadır. Tablo 3.7'de toplanan bu veriler ve ne anlama geldikleri açıklanmaktadır.

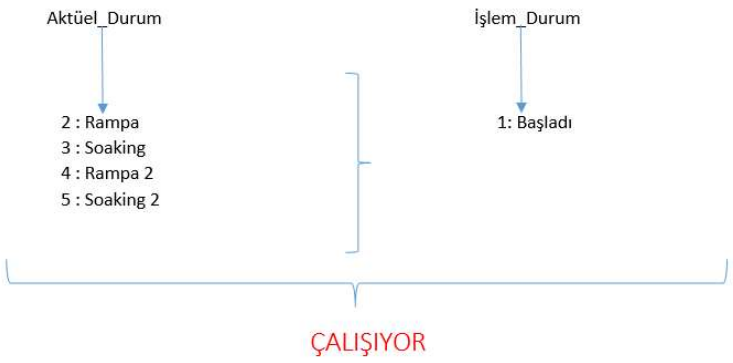
Tablo 3.7. MODBUS verileri ve anlamları

Veri adı	**Veri tipi**	**Veri anlamı**	**Verinin karşılığı**
Aktuel_durum	Word	Fırının aktif adımını göstermektedir.	0 = hazır 1 = çalışıyor 2 = rampa 3 = soaking 4 = rampa 2 5 = soaking 2 6 = işlem sonu 7 = zaman bekliyor 8 = sıcaklık hata
İşlem_durum	Word	Fırının anlık işlem durumunu göstermektedir.	0 = yeni 1 = başladı 2 = tamamlandı 3 = iptal edildi 4 = arıza

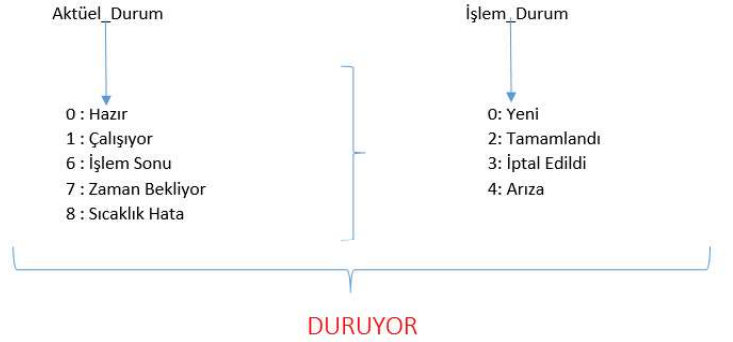
Elde edilen veriler ve anlamları incelendiğinde fırının çalışma durumu;

- Aktuel_Durum = 2 / 3 / 4 / 5
- İşlem_Durum = 1

olarak belirlenmiştir ve diğer tüm durumlarda tezgâh durmaktadır. Tezgâhın çalışma durumu Şekil 3.58'de, tezgâhın durma durumu ise Şekil 3.59'de gösterilmektedir.



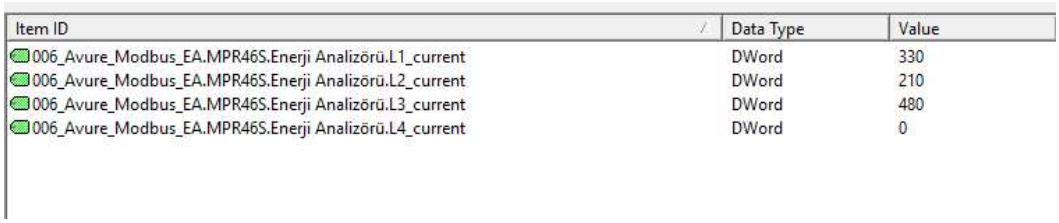
Şekil 3.58. MODBUS çalışma durumu diyagramı



Şekil 3.59. MODBUS duruş durumu diyagramı

3.6.6. Enerji analizörü kullanılarak MODBUS ile toplanan üretim verileri

Konvansiyonel tezgâh grubunda yer alan ancak MODBUS protokolü ile veri toplanabilmesi için ihtiyaç duyulan veri adreslerine ulaşılamaması nedeni ile üzerinde bulunan enerji analizöründen elde edilen veriler ile çalışma verisinin elde edilmesi sağlanan 006_Avure_MODBUS_EA adlı tezgâhtan nasıl veri toplanacağı ve hangi verilere ihtiyaç duyulduğu yukarıda anlatılmaktadır. Enerji analizöründen toplanan veriler enerji analizörünün marka ve modeline bağlı olarak değişebilmektedir. İlgili tezgâh üzerinde Entes marka MPR46S modelinde bir enerji analizörü bulunmaktadır. Bu enerji analizörü, tezgâhın enerji verileri ile alakalı olan her faz için ayrı olmak üzere, voltaj, akım, aktif güç, reaktif güç ve toplam güç gibi verileri MODBUS protokolü ile yayınlamaktadır. Şekil 3.60'ta tezgâh üzerinde bulunan enerji analizöründen toplanan anlık veriler gösterilmektedir.



Şekil 3.60 006_Avure_MODBUS_EA tezgahından toplanan anlık veriler

Enerji analizörü kullanılarak tezgâhın çalışma durumlarının elde edilebilmesi için tezgâhın faz akımları incelenmektedir. Tezgâhın çalışma durumu ve enerji analizörünün MODBUS protokolü ile yayınladığı veriler incelendiğinde sırasıyla L1_Current, L2_Current, L3_Current verileri çalışma verisinin elde edilmesi için gereken verileri sağlamaktadır. Tablo 3.8'de toplanan bu veriler ve ne anlama geldikleri açıklanmaktadır.

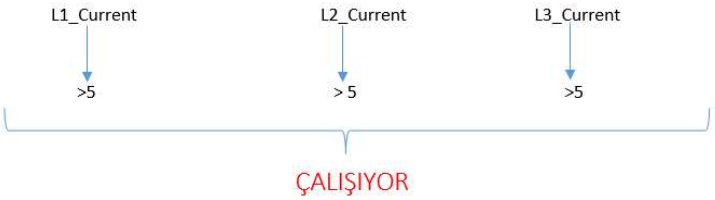
Tablo 3.8. Enerji analizörü MODBUS verileri ve anlamları

Veri Adı	**Veri Tipi**	**Veri Anlamı**	**Verinin Karşılığı**
L1_Current	DWORD	Tezgâhın birinci faz için çektiği akımı göstermektedir.	mA cinsinde 0 ile maksimum çekilen akım değeri
L2_Current	DWORD	Tezgâhın ikinci faz için çektiği akımı göstermektedir.	mA cinsinde 0 ile maksimum çekilen akım değeri
L3_Current	DWORD	Tezgâhın üçüncü faz için çektiği akımı göstermektedir.	mA cinsinde 0 ile maksimum çekilen akım değeri

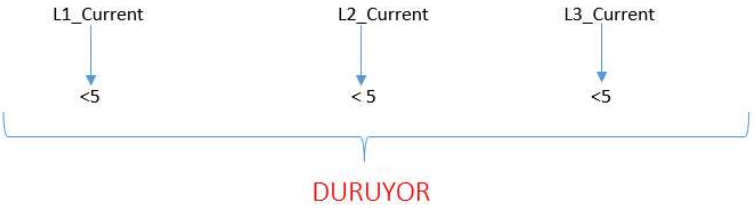
Elde edilen veriler ve anlamları incelendiğinde tezgâhın çalışma durumu;

- L1_Current> 5A
- L2_Current> 5A
- L3_Current> 5A

olarak belirlenmiştir ve diğer tüm durumlarda tezgâh durmaktadır. Tezgâhın çalışma durumu Şekil 3.61'de, tezgâhın durma durumu ise Şekil 3.62'de gösterilmektedir.



Şekil 3.61. Enerji analizörü MODBUS çalışma durumu diyagramı



Şekil 3.62. Enerji analizörü MODBUS duruş durumu diyagramı

4. BULGULAR

4.1. Veri Standardizasyon ve Anlamlandırma Uygulaması

IIoT teknolojisi kullanılarak üretim alanlarında bulunan tezgahlardan nasıl veri toplanabileceği ve toplanan veriler arasında protokol kaynaklı oluşan farklılıklar önceki bölümlerde anlatılmaktadır. Toplanan ham verilerin son kullanıcıya olduğu gibi göstermek, son kullanıcı tarafından anlaşılabilir olmamaktadır. Bu nedenle tezgâhtan toplanan ham verilerin işlenerek anlamlı bir hale getirilmesi gerekmektedir. Çalışma kapsamında tezgahlardan toplanan ham veriler kullanılarak, tezgahların çalışma durumları tespit edilmeye çalışılmaktadır. Çalışma sonucunda anlamlandırılan ve standart hale getirilen veriler tezgahların kullanılabilirlik kapasitelerini ortaya koymaktadır.

Java yazılım dili kullanılarak tezgahlardan toplanan verilerden yukarıda anlatılan çalışma durumları göz önünde bulundurularak, tezgahların çalışma durumları elde edilmektedir. Uygulama içerisinde protokollere göre oluşturulan fonksiyonlar standart bir veri çıktısının oluşmasını sağlamaktadır. Ham veriler kullanarak oluşturulan çalışma durumları ile ilgili veriler bir MS SQL veri tabanına kaydedilmektedir. Buraya kaydedilen veriler ise Grafana uygulaması üzerinde oluşturulan izleme ekranları ile görselleştirilmektedir. Aşağıda verinin hangi aşamalardan geçerek anlamlandırıldığı ve standart hale getirildiği anlatılmaktadır.

4.1.1. Veri standardizasyon uygulaması

Java dili kullanılarak yazılan uygulamanın asıl amacı ham verilerin anlamlı bir sonuç elde edebilecek şekilde standartlaştırılmasını ve depolanmasını sağlamaktır. Bu nedenle çalışmanın önceki bölümlerinde anlatılan Kepware yazılımının IoT Gateway eklentisi kullanılarak JSON formatında yayınlanan ham veriler bu uygulama aracılığı ile işlenerek çalışma durumlarının elde edilmesini sağlamaktadır. Uygulama ilgili protokoller için yapılan anlamlandırma çalışmaları sonucunda belirlenen koşullar sonucunda çalışma durumlarını elde etmektedir.

```
public class VeriStandardizasyonUygulaması implements ApplicationRunner {

    //IoT Gateway Linkinin Standart Kısmını Döndüren Fonksiyon
    public static final String IOTGATEWAY_OKUMA = "http://localhost:39321/iotgateway/read?ids=";

    Database vt = new Database();

        // Siemens
        // Breton
        public static SensorChange sc001 = new SensorChange("001", "","");

        // Mazak
        //MTConnect
        public static SensorChange sc002 = new SensorChange("002", "","");

        //Fanuc
        //Focas
        public static SensorChange sc003 = new SensorChange("003", "","");

        //Siemens
        // BFC
        public static SensorChange sc004 = new SensorChange("004", "","");

        //Siemens
        // MODBUS
        public static SensorChange sc005 = new SensorChange("005", "","");

        //Entes
        // Enerji Analizörü Modbus
        public static SensorChange sc006 = new SensorChange("006", "","");

    //Uygulama Main Döngüsü
    public static void main(String[] args) {

    //Uygulama Fonksiyonlarının Çağırıldığı Ana Fonksiyon, Uygulamanın Çalışmasını Sağlamaktadır.
    public void run(ApplicationArguments args) throws Exception {
```

Şekil 4.1. Uygulama sensör change fonksiyonları

```
//İlgili Protokoller İle Bağlanan Tezgahları İçeren Fonksiyonlar
public void runForOpcUa() {

public void runForMTConnect() {

public void runForFocas() {

public void runForModbus() {

public void runForEAModbus() {

//IoT Gateway'den okunan JSON veriden Veri Kalitesini Alan Fonksiyon
private static String parseJSONForTagQuality(String output) {

//IoT Gateway'den Yayınlanan İlgili Protokol Verilerini Okuyarak Değişkenlere Atayan Fonksiyon
private int setValueForOPCUa(String programStatusUrl, String opModeUrl, String marka, String protokolturu, String ekipmanNo,

private int setValueForDBMTConnect(String executionUrl, String modeUrl, String marka,String protokolturu, String ekipmanNo,

private int setValueForDBFocas(String authUrl, String runUrl, String marka,String protokolturu, String ekipmanNo, SensorChange sc){

private int setValueForDBModbus(String aktuelUrl, String islemUrl, String marka,String protokolturu, String ekipmanNo, SensorChange sc){

private int setValueForDBEAModbus(String L1, String L2, String L3, String marka,String protokolturu, String ekipmanNo, SensorChange sc,

//IoT Gateway Linkine Atılan URL Sorgusunu Döndüren Fonksiyon
//(setValueFor Fonksiyonlarının İçerisinde Çağırılmaktadır.)
private static String getUrlContents(String theUrl) {

//IoT Gateway'den Gelen JSON Veriden İstenilen Value Değerinin Alınmasını Sağlayan Fonksiyon
//(setValueFor Fonksiyonlarının İçerisinde Çağırılmaktadır.)
private static Object parseJSON(String output) {
```

Şekil 4.2. Uygulama anlamlandırma fonksiyonları

```
//Protokollere Göre Tezgah Durumlarını Anlamdırarak Veri Tabanına Kaydeden Fonksiyonlar
public void tezgahDurumKaydetOPC(String programStatus, String opMode, String marka, String protokolturu,

public void tezgahDurumKaydetMTConnect(String execution, String mode, String marka, String protokolturu,

public void tezgahDurumKaydetFocas(String run, String auth, String marka, String protokolturu,

public void tezgahDurumKaydetModbus(String aktuel, String islem, String marka, String protokolturu,

public void tezgahDurumKaydetEAModbus(String L1, String L2, String L3, String marka, String protokolturu,
```

Şekil 4.3. Uygulama kaydetme fonksiyonları

Şekil 4.1, 4.2 ve 4.3 IoT Gateway ile yayınlanan JSON formatındaki verinin okunup, ilgili değerin alınmasını, bu değerlerin kullanılarak çalışma verilerinin elde edilmesini ve veri tabanlarına kaydedilmesini sağlayan, uygulamanın ana fonksiyonlarının adlarını göstermektedir.

```
public class Database {

    Connection con = null;

    //Veri Tabanı Bağlantısının Kurulduğu Fonksiyon
    public Connection connect() {

    //Duruş Açıklamalarını Veri Tabanından Getiren Fonksiyon
    public String getAlarmAciklama(String durusId) {

    //Anlamlandırılan Verileri Veri Tabanına Kaydeden Fonksiyon
    public String InsertRecord(String ekipmanno, String marka, String durum, LocalDateTime nowDate, String stopCond,String durusAciklama,
```

Şekil 4.4. Veri tabanı fonksiyonları

Şekil 4.4'te veri tabanı bağlantılarının kurulmasını, veri tabanı üzerinde bir tabloda tutulan duruş açıklamalarının gerekli durumlarda uygulamaya çekilmesini ve anlamlandırılan verilerin veri tabanı tablolarına kaydedilmesini sağlayan fonksiyon adlarını göstermektedir.

```
public class SensorChange {

    private String sensorAdi;
    private String durum;
    private String kalite;

    //Tezgahların Son Durumlarının Sürekli Olarak Tutulmasını Sağlayan Fonksiyonlar
    public SensorChange(String sensorAdi, String durum, String kalite) {

    public String getSensorAdi() {

    public void setSensorAdi(String sensorAdi) {

    public String getKalite() {

    public void setKalite(String kalite) {

    public String getDurum() {

    public void setDurum(String durum) {
```

Şekil 4.5. Tezgâh durumlarının tutulduğu değişken fonksiyonları

Şekil 4.5'te tezgahların anlamlandırılan durumlarını ve değişikliklerinin tespit edilmesini sağlamak amacı ile oluşturulan "SensorChange" fonksiyonları gösterilmektedir. Uygulamada her bir tezgâh için bir Sensor Change değişkeni oluşturulmaktadır. Bu değişkenler yardımı ile tezgahların anlamlandırılan son durumları bir sınıf içerisinde tutulabilmektedir. Bu fonksiyonlar ilgili sc değişkeni ile eşleşen tezgâhın kalite ve durum bilgilerini kontrol ederek sınıf içerisinde tutulabilmesini sağlamaktadır. Tezgâh durumlarında oluşan her değişiklik sonrasında fonksiyonlar ilgili sc değişkeni ile eşleşerek bu sınıfın içerisinde tutulmaktadır. Şekil 4.6'da sınıf içerisinde bulunan fonksiyonlar gösterilmektedir.

```
public class SensorChange {

    private String sensorAdi;
    private String durum;
    private String kalite;
    //Tezgahların Son Durumlarının Sürekli Olarak Tutulmasını Sağlayan Fonksiyonlar
    public SensorChange(String sensorAdi, String durum, String kalite) {
        super();
        this.sensorAdi = sensorAdi;
        this.durum = durum;
        this.kalite = kalite;
    }
    public String getSensorAdi() {
        return sensorAdi;
    }
    public void setSensorAdi(String sensorAdi) {
        this.sensorAdi = sensorAdi;
    }
    public String getKalite() {
        return kalite;
    }
    public void setKalite(String kalite) {
        this.kalite = kalite;
    }
    public String getDurum() {
        return durum;
    }
    public void setDurum(String durum) {
        this.durum = durum;
    }
}
```

Şekil 4.6. Sensor Change sınıf fonksiyonları

Uygulama ana döngü olan main döngüsü ve run fonksiyonlarını kullanarak sürekli olarak veri değişikliklerini kontrol edebilmektedir. Her bir protokol için ayrıca oluşturulan runForOpcUa, runForMtConnect, runForFocas, runForModbus ve runForEAModbus fonksiyonları ise farklı veri tiplerinde Kepware'in Iot Gateway eklentisinden yayınlanan verilerin ilgili fonksiyonlar altında sorgulanmasını sağlamaktadır. Bu sorgular ilgili protokolden anlamlandırma için toplanması gereken ham verilerin JSON formatında uygulamaya akmasını sağlamaktadır.

```
//İlgili Protokoller İle Bağlanan Tezgahları İçeren Fonksiyonlar
public void runForOpcUa() {

    // OPC UA ile Bağlı Tezgahlar
    // BRETON

    // 001_Breton_OPCUa
    setValueForOPCUa(
            IOTGATEWAY_OKUMA + "001_Breton_OPCUA.Siemens.BRETON.GeneralData.ProgStatus",
            IOTGATEWAY_OKUMA + "001_Breton_OPCUA.Siemens.BRETON.GeneralData. OpMode",
            "Breton", "OPCUa", "001_Breton_OPCUa",
            IOTGATEWAY_OKUMA + "001_Breton_OPCUA.Siemens.BRETON.GeneralData.StopCond",
            IOTGATEWAY_OKUMA + "001_Breton_OPCUA.Siemens.BRETON.GeneralData.FeedrateOvr"
            , sc001);


    // BFC GATEWAY ile Bağlı Tezgahlar
    // BROETJE

    // 004_Broetje_BFC
    setValueForDBOPCUa(
            IOTGATEWAY_OKUMA + "004 Broetje_BFC.Siemens.BFGateway.root.BROETJE.GeneralData.ProgStatus",
            IOTGATEWAY_OKUMA + "004_Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.OpMode",
            "Broetje", "BFC", "004_Broetje_BFC",
            IOTGATEWAY_OKUMA + "004_Broetje_BFC.Siemens.BFGateway.root.BROETJE.GeneralData.StopCond",
            IOTGATEWAY_OKUMA + "004 _Broetje_BFC.Siemens.BFCGateway.root.BROETJE.GeneralData.FeedrateOvr",
            sc004);
}
```

Şekil 4.7. runForOpcUa fonksiyonu

Şekil 4.7'de gösterilen "runForOpcUa" fonksiyonu OPC Ua protokolü ve BFC Gateway kullanılarak verileri toplanan tezgahların ham verilerinin IoT Gateway yayınından sorgulanmasını sağlamaktadır. BFC Gateway ile verileri toplanan 004_Broetje_BFC adlı tezgâh sahip olduğu kontrolcü kaynaklı OPC Ua ile aynı verilere ve aynı anlamlandırmaya sahiptir. Bu nedenle çalışmada OPC Ua için hazırlanan fonksiyonlar içerisinde yer almaktadır. runForOpcUa fonksiyonu, Kepware tarafından Iot Gateway kullanılarak yayınlanan "progStatus, opMode, stopCond ve feedrateOvr" verilerinin JSON formatında sorgulanması için gerekli linklerin oluşturulmasını sağlamaktadır. Bunun beraberinde fonksiyon içerisinde yer alan veriler ve beraberinde tezgâhın markası, protokolü, adı ve sensör change değişkeni parametrik olarak "setValueForOPCUa" adlı fonksiyona gönderilmektedir ve bu fonksiyon Şekil 4.8'de gösterilmektedir.

```
//IoT Gateway'den Yayınlanan İlgili Protokol Verilerini Okuyarak Değişkenlere Atayan Fonksiyon
private int setValueForOPCUa(String programStatusUrl, String opModeUrl, String marka, String protokolturu, String ekipmanNo,
        String stopCondurl, String fovrUrl, SensorChange sc) {
    String opMode = "";
    String programStatus = "";
    String stopCond = "";
    String fovr = "";
    String result = "";

    if (!programStatusUrl.equals("")) {
        String output2 = getUrlContents(programStatusUrl);
        Object object2 = parseJSON(output2);
        result = parseJSONForTagQuality(output2);
        programStatus = object2.toString();
    } else if (programStatusUrl.equals("")) {
        programStatus = "";
    }
    if (!opModeUrl.equals("")) {
        String output2 = getUrlContents(opModeUrl);
        Object object2 = parseJSON(output2);
        opMode = object2.toString();
    } else if (opModeUrl.equals("")) {
        opMode = "";
    }
    if (!stopCondurl.equals("")) {
        String output4 = getUrlContents(stopCondurl);
        Object object4 = parseJSON(output4);
        stopCond = object4.toString();
    } else if (stopCondurl.equals("")) {
        stopCond = "";
    }
    if (!fovrUrl.equals("")) {
        String output6 = getUrlContents(fovrUrl);
        Object object6 = parseJSON(output6);
        fovr = object6.toString();
    } else if (fovrUrl.equals("")) {
        fovr = "100";
    }

    fovr = fovr.split("\\.")[0];
    int sonuc = 0;
    tezgahDurumKaydetOPC(programStatus, opMode, marka, protokolturu, ekipmanNo, stopCond, fovr, sc, result);

    return sonuc;

}
```

Şekil 4.8. setValueForOPCUa fonksiyonu

Parametrik olarak "runForOpcUa" fonksiyonundan gönderilen verilerin sorgulanabilmesi için gerekli linkler "setValueForOPCUa" fonksiyonuna gelmektedir. Bu fonksiyonun amacı JSON formatında IoT Gateway tarafından yayınlanan verileri oluşturulan linkler yardımı ile sorgulayarak istenilen değerleri almak ve bir değişkene atamaktır. Gelen linkler öncelikle "getUrlContents" fonksiyonuna giderek ilgili veriyi sorgulamaktadır.

Sorgu sonucunda JSON formatında dönen veri, “parseJSON” fonksiyonunda istenilen değerin JSON metninden ayrıştırılmasını sağlamaktadır. Her bir parametre için gönderilen veriler aynı fonksiyonlar kullanılarak ayrıştırıldıktan sonra ilgili değişkenler atılmaktadır. Bu değişkenler anlamlandırılmak ve veri tabanlarına kaydedilmek amacıyla parametrik olarak “tezgahDurumKaydetOPC” fonksiyonuna gönderilmektedir ve bu fonksiyon Şekil 4.9’da gösterilmektedir.

```
//Protokollere Göre Tezgah Durumlarını Anlamdırarak Veri Tabanına Kaydeden Fonksiyonlar
public void tezgahDurumKaydetOPC(String programStatus, String opMode, String marka, String protokolturu,
        String ekipmanNo, String stopCond, String fovr, SensorChange sensorChange, String kalite) {
    // 0 - Duruyor
    // 1 - Kapalı
    // 2 - Çalışıyor
    String durum = "0";
    String durusAciklama = "";
    String ariza = "0";

    String durusid = "";
    durusid = stopCond;

    if (programStatus.equals("3") && opMode.equals("2") && !fovr.equals("0")) {
        durum = "2";
    } else if (programStatus.equals("Unknown") || programStatus.equals("") || opMode.equals("Unknown")
            || opMode.equals("")) {

        durum = "1";

    } else {

        durum = "0";
    }

    if (durum == "2") {
        durusAciklama = "Started";
    } else if (durum == "1") {
        durusAciklama = "Off";
    } else if (durum == "0") {
        durusAciklama = vt.getAlarmAciklama(durusid);
        if (durusAciklama.equals("")) {
            durusAciklama = "Unidentified Stop";
        }
    }

    LocalDateTime nowDate = LocalDateTime.now();

    if (!sensorChange.getDurum().equals(durum) || sensorChange.getDurum().equals("") || !sensorChange.getKalite().equals(kalite)) {

        vt.InsertRecord(ekipmanNo, marka, durum, nowDate, stopCond, durusAciklama,
                fovr, ariza, kalite);

        sensorChange.setDurum(durum);
        sensorChange.setKalite(kalite);
    }

}
```

Şekil 4.9. tezgahDurumKaydetOPC fonksiyonu

Ayrıştırılarak ilgili değişkenlere atılan ve parametrik olarak “tezgahDurumKaydetOPC” fonksiyonuna gönderilen veriler, çalışmanın yukarıda anlatılan bölümlerinde elde edilen anlamlandırma çalışması neticesinde koşullandırılmaktadır. Bu koşullar anlık verileri takip ederek tezgâh durumunu, bir değişiklik olması durumunda güncellemektedir. Bu anlamlandırma çalışması kapsamında değerlendirilen 001_Breton_OPCUA ve 004_Broetje_BFC tezgahlarından toplanan program durumu (progStatus) verisinin “3” olduğu ve tezgâh çalışma modunun (opMode) “2” yani otomatik olduğu ve eksen ilerleme hızının (fovr) “0” olmadığı durumlarda tezgâh çalışıyor olarak anlamlandırılmaktır.

Toplanan verilerin “unknown” veya boş gelmesi durumunda ise, tezgâhtan veri kesildiği anlamına gelmektedir ve tezgâh kapalı olarak anlamlandırılmaktadır. Diğer tüm durumlarda ise tezgâh duruyor olarak anlamlandırılmıştır. Anlamlandırılan bu veriler “SensorChange” sınıfı içerisinde kontrol edilerek değişiklik olması durumunda sc değişkenleri güncellenmektedir. Bununla beraber ilgili veri tabanı tablolarına veriler kaydedilmektedir.

Uygulama tüm protokoller için özelleşmiş fonksiyonlarda aynı işlemleri gerçekleştirmektedir. Sırasıyla diğer protokoller için veri linklerini oluşturan fonksiyonlar “runForMTConnect, runForFocas, runForModbus ve runForEAModbus” fonksiyonlarıdır. Bu fonksiyonlar Şekil 4.10, 4.11, 4.12 ve 4.13’te sırasıyla gösterilmektedir.

```
public void runForMTConnect() {
    // MTConnect protokolü ile bağlanan tezgahlar

    // MAZAKLAR
    // 002_Mazak_MTConnect
    setValueForDBMTConnect(
            IOTGATEWAY_OKUMA + "002_ Mazak_MTConnect.MConnect.Controller(cont).Path (path1).execution",
            IOTGATEWAY_OKUMA + "002_Mazak MTConnect.MTConnect.Controller(cont).Path(path1).mode",
            "Mazak","MTConnect", "002_Mazak_MTConnect",
            IOTGATEWAY_OKUMA + "002_Mazak_MTConnect.MConnect.Controller(cont).Path(path1).Fovr",
            sc002);
}
```

Şekil 4.10. runForMTConnect fonksiyonu

```
public void runForFocas() {
    // Focas protokolü ile bağlanan tezgahlar

    // MORI SEIKI
    // 003_MoriSeiki_Focas
    setValueForDBFocas(
            IOTGATEWAY_OKUMA + "003_MoriSeiki_Focas.Focas.CNCData.01_StatusInfo Tags.statinfo_aut",
            IOTGATEWAY_OKUMA + "003_MoriSeiki_Focas.Focas.CNCData.01_Statuslfo Tags.statinfo_run",
            "MoriSeiki", "Focas", "003_MoriSeiki_Focas",
            sc003);
}
```

Şekil 4.11. runForFocas fonksiyonu

```
public void runForModbus() {
    // Modbus protokolü ile bağlanan tezgahlar

    // DESPATCH
    // 005_Despatch_Modbus
    setValueForDBModbus(
            IOTGATEWAY_OKUMA + "005 Despatch_Modbus.Siemens. Data.Aktuel_Durum",
            IOTGATEWAY_OKUMA + "005_Despatch_Modbus.Siemens.Data.Islem_Durum",
            "Despatch", "Modbus", "005_Despatch_Modbus",
            sc005);

}
```

Şekil 4.12. runForModbus fonksiyonu

```
public void runForEAModbus() {
    // Enerji Analizörü Modbus protokolü ile bağlanan tezgahlar

    //AVURE
    //006_AVURE_Modbus_EA
    setValueForDBEAModbus(
            IOTGATEWAY_IP_FOR_READ + "006_Avure_Modbus_EA.MPR46S.Enerji Analizörü.L1_current",
            IOTGATEWAY_IP_FOR_READ + "006_Avure_Modbus_EA.MPR46S.Enerji Analizörü.L2_current",
            IOTGATEWAY_IP_FOR_READ + "006_Avure_Modbus_EA.MPR46S.Enerji Analizörü.L3_current",
            "AVURE", "EA Modbus", "006_AVURE_Modbus_EA",
            sc006, 5, 5, 5);
}
```

Şekil 4.13. runForEAMODBUS fonksiyonu

Bu fonksiyonlar yukarıda anlatılan "runForOpcUa" fonksiyonu ile aynı çalışma mantığına sahiptirler. Aynı çalışma mantığına sahip olan bu fonksiyonlar da "runForOpcUa" fonksiyonunda olduğu gibi ilgili parametreleri sırasıyla "setValueForDBMTConnect, setValueForDBFocas, setValueForDBModbus ve setValueForDBEAModbus" fonksiyonlarına iletmektedir. Bu fonksiyonlar Şekil 4.14, 4.15, 4.16 ve 4.17 'de görülmektedir.

```
private int setValueForDBMTConnect(String executionUrl, String modeUrl, String marka,String protokolturu, String ekipmanNo,
        String fovrUrl, SensorChange sc) {

    String mode = "";
    String execution = "";
    String fovr = "";
    String result = "";

    if (!executionUrl.equals("")) {
        String output2 = getUrlContents(executionUrl);
        Object object2 = parseJSON(output2);
        result = parseJSONForTagQuality(output2);
        execution = object2.toString();
    } else if (executionUrl.equals("")) {
        execution = "";
    }
    if (!modeUrl.equals("")) {
        String output2 = getUrlContents(modeUrl);
        Object object2 = parseJSON(output2);
        mode = object2.toString();
    } else if (modeUrl.equals("")) {
        mode = "";
    }
    if (!fovrUrl.equals("")) {
        String output6 = getUrlContents(fovrUrl);
        Object object6 = parseJSON(output6);
        fovr = object6.toString();
    } else if (fovrUrl.equals("")) {
        fovr = "100";
    }

    fovr = fovr.split("\\.")[0];
    int sonuc = 0;
    tezgahDurumKaydetMTConnect(execution, mode, marka, protokolturu, ekipmanNo, fovr, sc, result);

    return sonuc;

}
```

Şekil 4.14. setValueForDBMTConnect fonksiyonu

```
private int setValueForDBFocas(String authUrl, String runUrl, String marka,String protokolturu, String ekipmanNo, SensorChange sc){

    //

    String auth = "";
    String run = "";
    String result = "";

    if (!authUrl.equals("")) {
        String output2 = getUrlContents(authUrl);
        Object object2 = parseJSON(output2);
        result = parseJSONForTagQuality(output2);
        auth = object2.toString();
    } else if (authUrl.equals("")) {
        auth = "";
    }
    if (!runUrl.equals("")) {
        String output2 = getUrlContents(runUrl);
        Object object2 = parseJSON(output2);
        run = object2.toString();
    } else if (runUrl.equals("")) {
        run = "";
    }
    int sonuc = 0;
    tezgahDurumKaydetFocas(run, auth, marka, protokolturu, ekipmanNo, sc, result);

    return sonuc;

}
```

Şekil 4.15. setValueForDBFocas fonksiyonu

```
private int setValueForDBModbus(String aktuelUrl, String islemUrl, String marka,String protokolturu, String ekipmanNo, SensorChange sc){
    String aktuel = "";
    String islem = "";
    String result = "";

    if (!aktuelUrl.equals("")) {
        String output2 = getUrlContents(aktuelUrl);
        Object object2 = parseJSON(output2);
        aktuel = object2.toString();
    } else if (aktuelUrl.equals("")) {
        aktuel = "";
    }
    // System.out.println(islemUrl);
    if (!islemUrl.equals("")) {
        String output2 = getUrlContents(islemUrl);
        Object object2 = parseJSON(output2);
        // System.out.println("obj2: " + object2.toString());
        result = parseJSONForTagQuality(output2);
        islem = object2.toString();
    } else if (islemUrl.equals("")) {
        islem = "";
    }
    int sonuc = 0;
    tezgahDurumKaydetModbus(aktuel, islem, marka, protokolturu, ekipmanNo, sc, result)

    return sonuc;

}
```

Şekil 4.16. setValueForDBModbus fonksiyonu

```
private int setValueForDBEAModbus(String L1, String L2, String L3, String marka,String protokolturu, String ekipmanNo, SensorChange sc,
        double faz1EsikDeger,double faz2EsikDeger, double faz3EsikDeger) {

    String Faz1 = "";
    String Faz2 = "";
    String Faz3 = "";
    String result = "";

    if (!L1.equals("")) {
        String output2 = getUrlContents(L1);
        Object object2 = parseJSON(output2);
        result = parseJSONForTagQuality(output2);
        Faz1 = object2.toString();
    }

    if (!L2.equals("")) {
        String output4 = getUrlContents(L2);
        Object object4 = parseJSON(output4);
        Faz2 = object4.toString();
    }

    if (!L3.equals("")) {
        String output5 = getUrlContents(L3);
        Object object5 = parseJSON(output5);
        Faz3 = object5.toString();
    }
    int sonuc = 0;
    tezgahDurumKaydetEAModbus(Faz1, Faz2, Faz3, marka, protokolturu, ekipmanNo, sc, result,
            faz1EsikDeger,faz2EsikDeger,faz3EsikDeger);
    return sonuc;

}
```

Şekil 4.17. setValueForDBEAModbus fonksiyonu

Bu fonksiyonlar yukarıda anlatılan “setValueForOPCUa” fonksiyonu ile aynı çalışma mantığına sahiptir. Fonksiyonlara gelen parametrik veriler bu fonksiyon içerisinde sorgulanarak ayrıştırılmaktadır. Ayrıştırılan veriler ilgili değişkenlere atanarak anlamlandırılmak ve veri tabanlarına kaydedilmek amacıyla protokollere göre oluşturulan “tezgahDurumKaydetMTConnect, tezgahDurumKaydetFocas, tezgahDurumKaydetModbus, tezgahDurumKaydetEAModbus” fonksiyonlarına iletilmektedir.

```
public void tezgahDurumKaydetMTConnect(String execution, String mode, String marka, String protokolturu,
        String ekipmanNo,String fovr, SensorChange sensorChange, String kalite) {

    // 0 - Duruyor
    // 1 - Kapalı
    // 2   Çalışıyor

    String durum = "0";
    String durusAciklama = "";
    String ariza = "0";

    String durusid = "0";

    if (execution.equals("ACTIVE") && mode.equals("AUTOMATIC") && !fovr.equals("0")) {
        durum = "2";
    } else if (execution.equals("Unknown") || execution.equals("") || mode.equals("Unknown") || mode.equals("")) {

        durum = "1";
    } else {

        durum = "0";
    }

    if (durum == "2") {
        durusAciklama = "Started";
    } else if (durum == "1") {
        durusAciklama = "Off";
    } else if (durum == "0") {
        durusAciklama = vt.getAlarmAciklama(durusid);
        if (durusAciklama.equals("")) {
            durusAciklama = "Unidentified Stop";
        }
    }

    LocalDateTime nowDate = LocalDateTime.now();
    if (!sensorChange.getDurum().equals(durum) || sensorChange.getDurum().equals("") || !sensorChange.getKalite().equals(kalite)) {

        vt.InsertRecord(ekipmanNo, marka, durum, nowDate, "", durusAciklama,
                fovr, ariza, kalite);

        sensorChange.setDurum(durum);
        sensorChange.setKalite(kalite);
    }
}
```

Şekil 4.18. tezgahDurumKaydetMTConnect fonksiyonu

Şekil 4.18’de gösterilen “tezgahDurumKaydetMTConnect” fonksiyonuna ayrıştırılarak gönderilen ilgili değişkenler çalışmanın yukarıda anlatılan bölümlerinde elde edilen anlamlandırma çalışması neticesinde koşullandırılmaktadır. Bu koşullar anlık verileri takip ederek tezgâh durumunu, bir değişiklik olması durumunda güncellemektedir. Bu anlamlandırma çalışması kapsamında değerlendirilen 002_Mazak_MTConnect tezgahından toplanan program durumu (execution) verisinin “ACTIVE” olduğu ve tezgâh çalışma modunun (Mode) “AUTOMATIC” olduğu ve eksen ilerleme hızının (fovr) “0” olmadığı durumlarda tezgâh çalışıyor olarak anlamlandırılmaktır. Toplanan verilerin “unknown” veya boş gelmesi durumunda ise, tezgâhtan veri kesildiği anlamına gelmektedir ve tezgâh kapalı olarak anlamlandırılmaktadır. Diğer tüm durumlarda ise tezgâh duruyor olarak anlamlandırılmaktadır.

Anlamlandırılan bu veriler “SensorChange” sınıfı içerisinde kontrol edilerek değişiklik olması durumunda sc değişkenleri güncellenmektedir. Bununla beraber ilgili veri tabanı tablolarına veriler kaydedilmektedir.

```
public void tezgahDurumKaydetFocas(String run, String auth, String marka, String protokolturu,
        String ekipmanNo, SensorChange sensorChange, String kalite){
    // 0 - Duruyor
    // 1 - Kapalı
    // 2 - Çalışıyor

    String durum = "0";
    String durusAciklama = "";
    String ariza = "0";

    String durusid = "0";

    if (run.equals("3") && auth.equals("1")) {
        durum = "2";
    } else if (run.equals("Unknown") || run.equals("") || auth.equals("Unknown") || auth.equals("")) {

        durum = "1";
    } else {

        durum = "0";

    }

    if (durum == "2") {
        durusAciklama = "Started";
    } else if (durum == "1") {
        durusAciklama = "Off";
    } else if (durum == "0") {
        durusAciklama = vt.getAlarmAciklama(durusid);
        if (durusAciklama.equals("")) {
            durusAciklama = "Unidentified Stop";
        }
    }

    LocalDateTime nowDate = LocalDateTime.now();

    if (!sensorChange.getDurum().equals(durum) || sensorChange.getDurum().equals("") || !sensorChange.getKalite().equals(kalite)) {

        vt.InsertRecord(ekipmanNo, marka, durum, nowDate, "", durusAciklama,
                "100", ariza, kalite);

        sensorChange.setDurum(durum);
        sensorChange.setKalite(kalite);
    }
}
```

Şekil 4.19. tezgahDurumKaydetFocas fonksiyonu

Şekil 4.19’da gösterilen “tezgahDurumKaydetFocas” fonksiyonuna ayrıştırılarak gönderilen ilgili değişkenler çalışmanın yukarıda anlatılan bölümlerinde elde edilen anlamlandırma çalışması neticesinde koşullandırılmaktadır. Bu koşullar anlık verileri takip ederek tezgâh durumunu, bir değişiklik olması durumunda güncellemektedir. Bu anlamlandırma çalışması kapsamında değerlendirilen 003_MoriSeiki_Focas tezgahından toplanan program durumu (run) verisinin “3” olduğu ve tezgâh çalışma modunun (auth) “1” durumlarda tezgâh çalışıyor olarak anlamlandırılmaktır. Toplanan verilerin “unknown” veya boş gelmesi durumunda ise, tezgâhtan veri kesildiği anlamına gelmektedir ve tezgâh kapalı olarak anlamlandırılmaktadır. Diğer tüm durumlarda ise tezgâh duruyor olarak anlamlandırılmaktadır.

Anlamlandırılan bu veriler “SensorChange” sınıfı içerisinde kontrol edilerek değişiklik olması durumunda sc değişkenleri güncellenmektedir. Bununla beraber ilgili veri tabanı tablolarına veriler kaydedilmektedir.

```
public void tezgahDurumKaydetModbus(String aktuel, String islem, String marka, String protokolturu,
        String ekipmanNo, SensorChange sensorChange, String kalite){
    // 0 - Duruyor
    // 1 - Kapalı
    // 2 - Çalışıyor

    // }
    String durum = "0";
    String durusAciklama = "";
    String ariza = "0";

    String durusid = "";

    if ((aktuel.equals("2") || aktuel.equals("3") || aktuel.equals("4") || aktuel.equals("5"))
            && islem.equals("1")) {
        durum = "2";
    } else if (aktuel.equals("Unknown") || aktuel.equals("") || islem.equals("Unknown") || islem.equals("")) {
        durum = "1";
    } else {
        durum = "0";
    }
    // System.out.println("durum" + durum);

    if (durum == "2") {
        durusAciklama = "Started";
    } else if (durum == "1") {
        durusAciklama = "Off";
    } else if (durum == "0") {
        durusAciklama = vt.getAlarmAciklama(durusid);
        if (durusAciklama.equals("")) {
            durusAciklama = "Unidentified Stop";
        }
    }

    LocalDateTime nowDate = LocalDateTime.now();

    if (!sensorChange.getDurum().equals(durum) || sensorChange.getDurum().equals("") || !sensorChange.getKalite().equals(kalite)) {

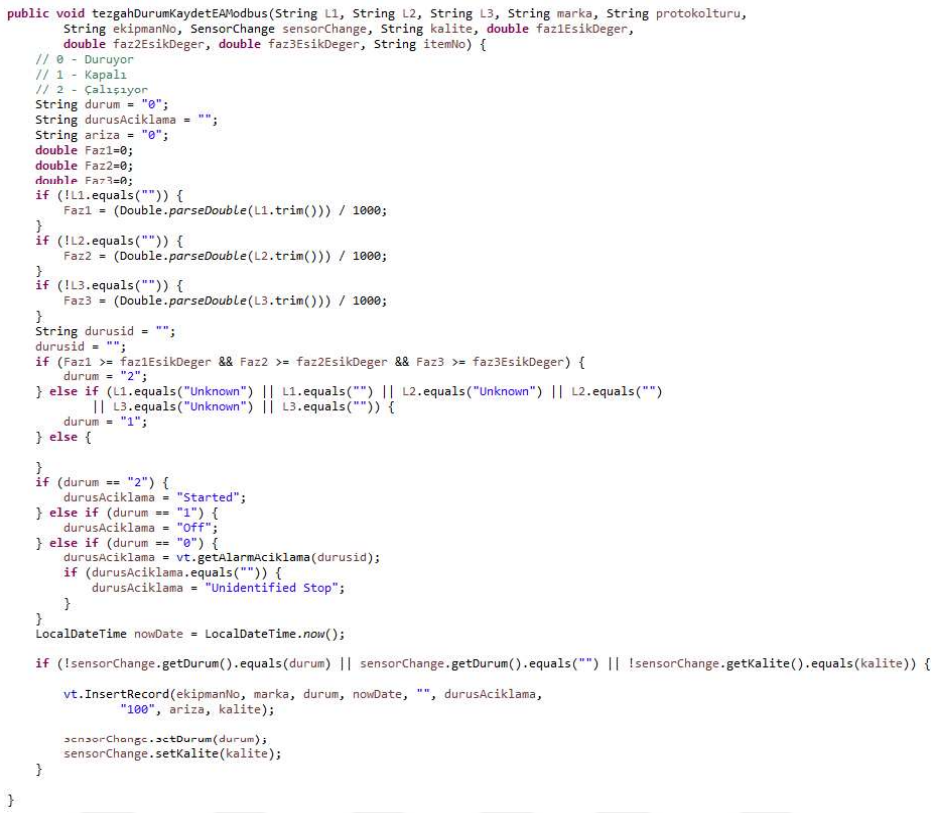
        vt.InsertRecord(ekipmanNo, marka, durum, nowDate, "", durusAciklama,
                "100", ariza, kalite);

        sensorChange.setDurum(durum);
        sensorChange.setKalite(kalite);
    }
}
```

Şekil 4.20. tezgahDurumKaydetMODBUS fonksiyonu

Şekil 4.20'de gösterilen "tezgahDurumKaydetMODBUS" fonksiyonuna ayrıştırılarak gönderilen ilgili değişkenler çalışmanın yukarıda anlatılan bölümlerinde elde edilen anlamlandırma çalışması neticesinde koşullandırılmaktadır. Bu koşullar anlık verileri takip ederek tezgâh durumunu, bir değişiklik olması durumunda güncellemektedir. Bu anlamlandırma çalışması kapsamında değerlendirilen 005_Despatch_MODBUS tezgahından toplanan anlık durum (aktuel) verisinin "2" veya "3" veya "4" veya "5" olduğu ve tezgâh işlem durumunun (islem) "1" olduğu durumlarda tezgâh çalışıyor olarak anlamlandırılmaktır. Toplanan verilerin "unknown" veya boş gelmesi durumunda ise, tezgâhtan veri kesildiği anlamına gelmektedir ve tezgâh kapalı olarak anlamlandırılmaktadır. Diğer tüm durumlarda ise tezgâh duruyor olarak anlamlandırılmaktadır.

Anlamlandırılan bu veriler "SensorChange" sınıfı içerisinde kontrol edilerek değişiklik olması durumunda sc değişkenleri güncellenmektedir. Bununla beraber ilgili veri tabanı tablolarına veriler kaydedilmektedir.



```
public void tezgahDurumKaydetEAModbus(String L1, String L2, String L3, String marka, String protokolturu,
        String ekipmanNo, SensorChange sensorChange, String kalite, double faz1EsikDeger,
        double faz2EsikDeger, double faz3EsikDeger, String itemNo) {
    // 0 - Duruyor
    // 1 - Kapalı
    // 2 - Çalışıyor
    String durum = "0";
    String durusAciklama = "";
    String ariza = "0";
    double Faz1=0;
    double Faz2=0;
    double Faz3=0;
    if (!L1.equals("")) {
        Faz1 = (Double.parseDouble(L1.trim())) / 1000;
    }
    if (!L2.equals("")) {
        Faz2 = (Double.parseDouble(L2.trim())) / 1000;
    }
    if (!L3.equals("")) {
        Faz3 = (Double.parseDouble(L3.trim())) / 1000;
    }
    String durusid = "";
    durusid = "";
    if (Faz1 >= faz1EsikDeger && Faz2 >= faz2EsikDeger && Faz3 >= faz3EsikDeger) {
        durum = "2";
    } else if (L1.equals("Unknown") || L1.equals("") || L2.equals("Unknown") || L2.equals("")
            || L3.equals("Unknown") || L3.equals("")) {
        durum = "1";
    } else {

    }
    if (durum == "2") {
        durusAciklama = "Started";
    } else if (durum == "1") {
        durusAciklama = "Off";
    } else if (durum == "0") {
        durusAciklama = vt.getAlarmAciklama(durusid);
        if (durusAciklama.equals("")) {
            durusAciklama = "Unidentified Stop";
        }
    }
    LocalDateTime nowDate = LocalDateTime.now();

    if (!sensorChange.getDurum().equals(durum) || sensorChange.getDurum().equals("") || !sensorChange.getKalite().equals(kalite)) {

        vt.InsertRecord(ekipmanNo, marka, durum, nowDate, "", durusAciklama,
                "100", ariza, kalite);

        sensorChange.setDurum(durum);
        sensorChange.setKalite(kalite);
    }

}
```

Şekil 4.21. tezgahDurumKaydetEAModbus fonksiyonu

Şekil 4.21'de gösterilen "tezgahDurumKaydetEAModbus" fonksiyonuna ayrıştırılarak gönderilen ilgili değişkenler çalışmanın yukarıda anlatılan bölümlerinde elde edilen anlamlandırma çalışması neticesinde koşullandırılmaktadır. Bu koşullar anlık verileri takip ederek tezgâh durumunu, bir değişiklik olması durumunda güncellemektedir. Bu anlamlandırma çalışması kapsamında değerlendirilen 006_Avure_MODBUS_EA tezgahından toplanan akım verilerinin, belirlenen eşik değerlerden büyük olduğu durumlarda tezgâh çalışıyor olarak anlamlandırılmaktır. Toplanan verilerin "unknown" veya boş gelmesi durumunda ise, tezgâhtan veri kesildiği anlamına gelmektedir ve tezgâh kapalı olarak anlamlandırılmaktadır. Diğer tüm durumlarda ise tezgâh duruyor olarak anlamlandırılmaktadır.

Anlamlandırılan bu veriler "SensorChange" sınıfı içerisinde kontrol edilerek değişiklik olması durumunda sc değişkenleri güncellenmektedir. Bununla beraber ilgili veri tabanı tablolarına veriler kaydedilmektedir.

Uygulama içerisinde anlamlandırılan veriler “Durum” verisi olarak çıktı vermektedir. “Durum” verisi tezgahların anlık durumlarını göstermektedir ve sayısal bir veridir. Durum verisinin “0” geldiği durumlarda tezgâh durmakta, “1” geldiği durumlarda kapalı ve “2” geldiği durumlarda çalışmaktadır. Anlamlandırılan veriler “vt.InsertRecord” fonksiyonu ile parametrik olarak veri tabanı fonksiyonlarına iletilmektedir. Veri tabanına verilerin kaydedilebilmesi için oluşturulan veri tabanı tablosu ile bağlantı kurulmaktadır. Şekil 4.22‘de görülen “Connection connect ()” fonksiyonunda ilgili veri tabanı ile bağlantı kurulmaktadır.

```
public class Database {

    Connection con = null;
    //Veri Tabanı Bağlantısının Kurulduğu Fonksiyon
    public Connection connect() {
        String server = "IIoT";
        int port = 1433;
        String user = "IIoT";
        String password = "***";
        String database = "TEZ";
        String jdbcurl = "jdbc:sqlserver://" + server + ":" + port + ";user=" + user + ";password=" + password
                + ";databasename=" + database + "";
        try {
            Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
            con = DriverManager.getConnection(jdbcurl);
        } catch (ClassNotFoundException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        return con;
    }
```

Şekil 4.22. Veri tabanı bağlantı fonksiyonu

Veri tabanları için oluşturulmuş olan “Database” adlı sınıf içerisinde sorgu fonksiyonları da yer almaktadır. Bu fonksiyonlardan bir tanesi olan “getAlarmAciklama” fonksiyonu ilgili tezgâhın “duruşid” verisine göre veri tabanına önceden kaydedilmiş olan duruş açıklamasını getirmektedir. Elde edilen bu veri tezgâhın anlık duruş durumu ile birleştirilmektedir. Duruş açıklamalarının sayısının fazla olması sebebi ile uygulama içerisine yerleştirilmemektedir. Uygulama içerisinde bu verinin anlamlandırılması uzun bir koşul oluşturulmasına neden olmaktadır. Bu nedenle duruş açıklamaları veri tabanında tutulmaktadır. Şekil 4.23’te görülen “getAlarmAciklama” fonksiyonu veri tabanından ilgili duruş açıklamalarını getirmektedir.

```
//Duruş Açıklamalarını Veri Tabanından Getiren Fonksiyon
public String getAlarmAciklama(String durusId) {

    if (con == null) {
        con = connect();
    }
    String aciklama = "";
    try {
        String sorgu = "SELECT ACIKLAMA FROM DURUS_KOD WHERE ID = '" + durusId + "'";
        PreparedStatement pst = con.prepareStatement(sorgu);
        ResultSet rs = pst.executeQuery();
        if (rs != null) {
            while (rs.next()) {
                aciklama = rs.getString(1);
            }
        } else {
            aciklama = "";
        }
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
    return aciklama;
}
```

Şekil 4.23. getAlarmAciklama fonksiyonu

İlgili Database sınıfı içerisinde yer alan bir diğer fonksiyon ise anlamlandırılan tezgâh verilerinin ilgili veri tabanı fonksiyonlarına kaydedilmesini sağlamaktadır. Şekil 4.24'de görülen "InsertRecord" adlı fonksiyon, "tezgahDurumKaydet(protokol)" fonksiyonlarından "vt.InsertRecord" aracılığı ile gönderilen anlamlandırılmış verileri ilgili veri tabanı fonksiyonlarına kaydetmektedir.

```
//Anlamlandırılan Verileri Veri Tabanına Kaydeden Fonksiyon
public String InsertRecord(String ekipmanno, String marka, String durum, LocalDateTime nowDate, String stopCond,String durusAciklama,
        String fovr, String ariza, String kalite, String durumKodu) {

    if (con == null) {
        con = connect();
    }

    try {

        String sorgu = "INSERT INTO TEZGAH_VERILERI_STD(EKIPMANNO,TARIH_BASLANGIC,DURUM,FEEDRATE,DURUS_ID,DURUS_ACIKLAMA,ARIZA,KALITE) VALUES "
                + "('" + ekipmanno + "','" + String.valueOf(nowDate) + "','" + durum + "','" + fovr + "','" + stopCond + "','" + durusAciklama
                + "','"  + ariza + "','" + kalite + "')";

        PreparedStatement pst = con.prepareStatement(sorgu);
        int rs = pst.executeUpdate();
        if(rs==1) {
            return "Kaydedildi";
        }
        else {
            return "Kaydedilemedi";
        }

    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
        return "Kaydedilemedi";
    }

}
```

Şekil 4.24. Veri tabanı kayıt fonksiyonu

4.2.2. MS SQL veri tabanı kayıtları

Anlamlandırılan ve Durum verisi ile standart hale getirilen çalışma verileri "TEZGAH_VERILERI_STD" adlı veri tabanı tablosuna kaydedilmektedir. Kaydedilen bu veriler tezgâh kapasitesinin belirlenmesinde, anlık durumlarının görüntülenmesinde ve tezgâh verimliliğinin hesaplanmasında kullanılabilmektedir.

```
/****** Script for SelectTopNRows command from SSMS  ******/
SELECT Top (1000) [EKIPMANNO]
      ,[TARIH_BASLANGIC]
      ,[DURUM]
      ,[FEEDRATE]
      ,[DURUS_ID]
      ,[DURUS_ACIKLAMA]
      ,[ARIZA]
      ,[KALITE]
  FROM [DEDW].[dbo].[TEZGAH_VERILERI_STD] order By TARIH_BASLANGIC Desc
```

100 %

Results | Messages

	EKIPMANNO	TARIH_BASLANGIC	DURUM	FEEDRATE	DURUS_ID	DURUS_ACIKLAMA	ARIZA	KALITE
1	001_Breton_OPC	2023-11-30 13:04:27.067	0	69	5	M0 / M1 active	0	1
2	001_Breton_OPC	2023-11-30 13:00:28.047	2	69	11	Çalışıyor	0	1
3	001_Breton_OPC	2023-11-30 13:00:19.727	0	69	0	No stop state	0	1
4	001_Breton_OPC	2023-11-30 12:59:07.483	2	69	0	Çalışıyor	0	1
5	001_Breton_OPC	2023-11-30 12:58:04.930	0	69	0	No stop state	0	1
6	004_Broetje_BFC	2023-11-30 12:52:05.087	0	120	0	No stop state	0	1
7	004_Broetje_BFC	2023-11-30 12:50:25.223	2	120	11	Çalışıyor	0	1
8	004_Broetje_BFC	2023-11-30 12:50:19.163	0	120	5	M0 / M1 active	0	1
9	004_Broetje_BFC	2023-11-30 12:49:04.347	2	120	8	Çalışıyor	0	1
10	004_Broetje_BFC	2023-11-30 12:48:58.893	0	120	5	M0 / M1 active	0	1
11	004_Broetje_BFC	2023-11-30 12:48:13.177	2	120	8	Çalışıyor	0	1
12	004_Broetje_BFC	2023-11-30 12:48:10.473	0	120	5	M0 / M1 active	0	1
13	006_Avure_Modbus_EA	2023-11-30 12:47:11.233	0	0	0	Belirsiz Duruş	0	1
14	004_Broetje_BFC	2023-11-30 12:46:38.180	2	120	8	Çalışıyor	0	1
15	004_Broetje_BFC	2023-11-30 12:46:35.447	0	120	5	M0 / M1 active	0	1
16	006_Avure_Modbus_EA	2023-11-30 12:45:46.363	2	0	0	Çalışıyor	0	1
17	006_Avure_Modbus_EA	2023-11-30 12:45:43.677	0	0	0	Belirsiz Duruş	0	1
18	004_Broetje_BFC	2023-11-30 12:45:43.377	2	120	8	Çalışıyor	0	1
19	004_Broetje_BFC	2023-11-30 12:45:35.290	0	120	5	M0 / M1 active	0	1
20	004_Broetje_BFC	2023-11-30 12:44:52.957	2	120	8	Çalışıyor	0	1
21	004_Broetje_BFC	2023-11-30 12:44:50.237	0	120	5	M0 / M1 active	0	1
22	006_Avure_Modbus_EA	2023-11-30 12:44:20.583	2	0	0	Çalışıyor	0	1
23	004_Broetje_BFC	2023-11-30 12:42:38.577	2	120	8	Çalışıyor	0	1
24	004_Broetje_BFC	2023-11-30 12:42:33.200	0	120	5	M0 / M1 active	0	1
25	004_Broetje_BFC	2023-11-30 12:41:38.520	2	120	8	Çalışıyor	0	1
26	004_Broetje_BFC	2023-11-30 12:41:33.027	0	120	5	M0 / M1 active	0	1
27	004_Broetje_BFC	2023-11-30 12:40:08.417	2	120	8	Çalışıyor	0	1
28	004_Broetje_BFC	2023-11-30 12:40:02.903	0	120	5	M0 / M1 active	0	1
29	004_Broetje_BFC	2023-11-30 12:38:29.533	2	120	8	Çalışıyor	0	1
30	004_Broetje_BFC	2023-11-30 12:38:24.023	0	120	5	M0 / M1 active	0	1
31	004_Broetje_BFC	2023-11-30 12:36:14.177	2	120	8	Çalışıyor	0	1
32	004_Broetje_BFC	2023-11-30 12:36:08.733	0	120	5	M0 / M1 active	0	1

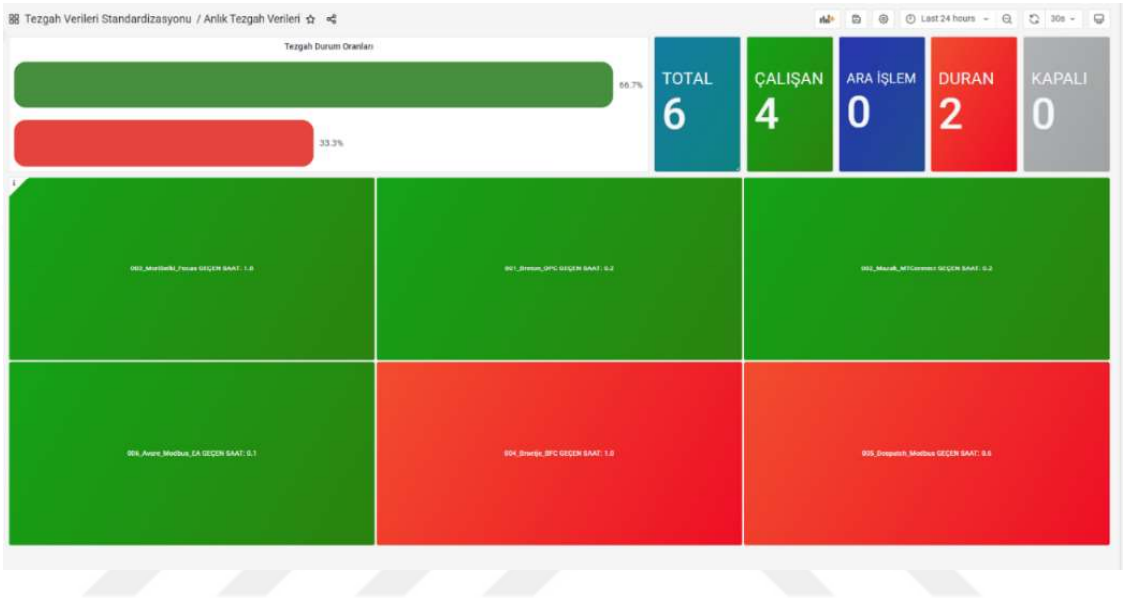
Şekil 4.25. TEZGAH_VERILERI_STD tablosu örnek sorgu

Şekil 4.25'te ilgili tabloya ilişkin örnek bir sorgu ve tablo formatı gösterilmektedir.

4.2.3 Grafana ile verilerin görselleştirilmesi

Tezgahlardan toplanan veriler, kullanılan IIoT protokollerine göre farklılıklar oluşturabilmektedir. Ancak yapılan çalışma neticesinde gerekli saha incelemeleri ve uygulamalar ile verilerin standartlaştırılabildiği görülmektedir. IIoT altyapısı üretim alanlarında kullanılan tezgahlar üzerinde kurularak, tezgahların desteklediği protokoller aracılığı ile toplanan farklı tip veriler, yapılan çalışma neticesinde standart bir veri formatına getirilerek veri tabanlarına kaydedilmiştir. Bu durumun sonucunda elde edilen çalışma verileri, durum verisi olarak veri tabanlarında tutulmaktadır. Bu veriler Grafana uygulaması yardımı ile görselleştirilerek son kullanıcıya sunulmaktadır.

Tezgahların veri tabanına kaydedilmiş verileri için oluşturulan uygun sorgular ve Grafana uygulaması üzerinde oluşturulan izleme ekranları ile tezgâh verileri anlık olarak izlenebilmektedir. Elde edilen verilerin son kullanıcılar tarafından takip edilebilmesi amacı ile Grafana üzerinde izleme ekranları oluşturulmaktadır. Şekil 4.26'da tezgahların anlık durumlarını göstermek için tasarlanan Grafana ekranı gösterilmektedir.

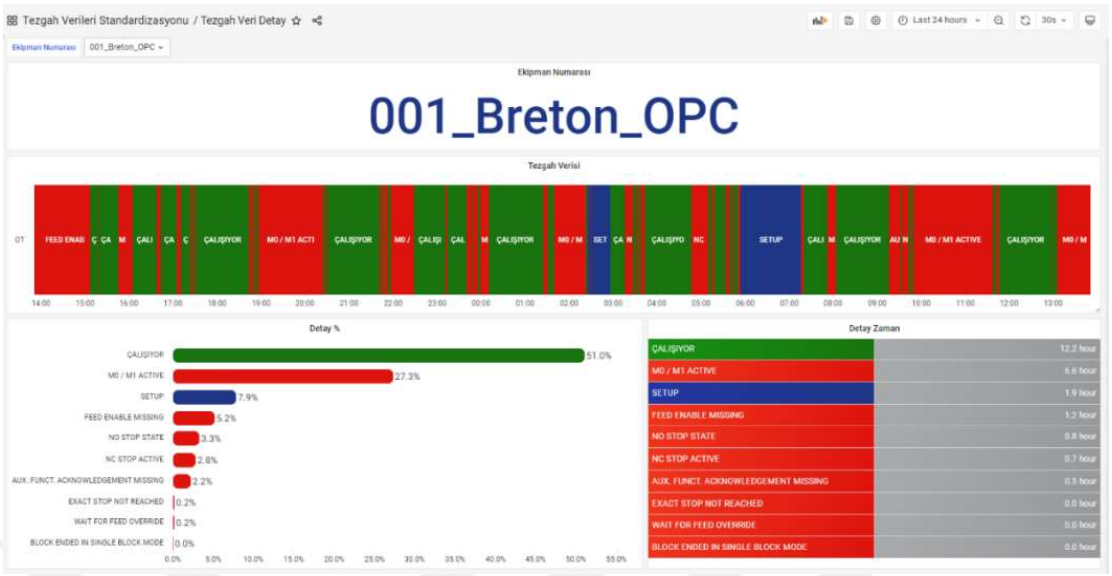


Şekil 4.26. Anlık tezgâh verileri Grafana ekranı

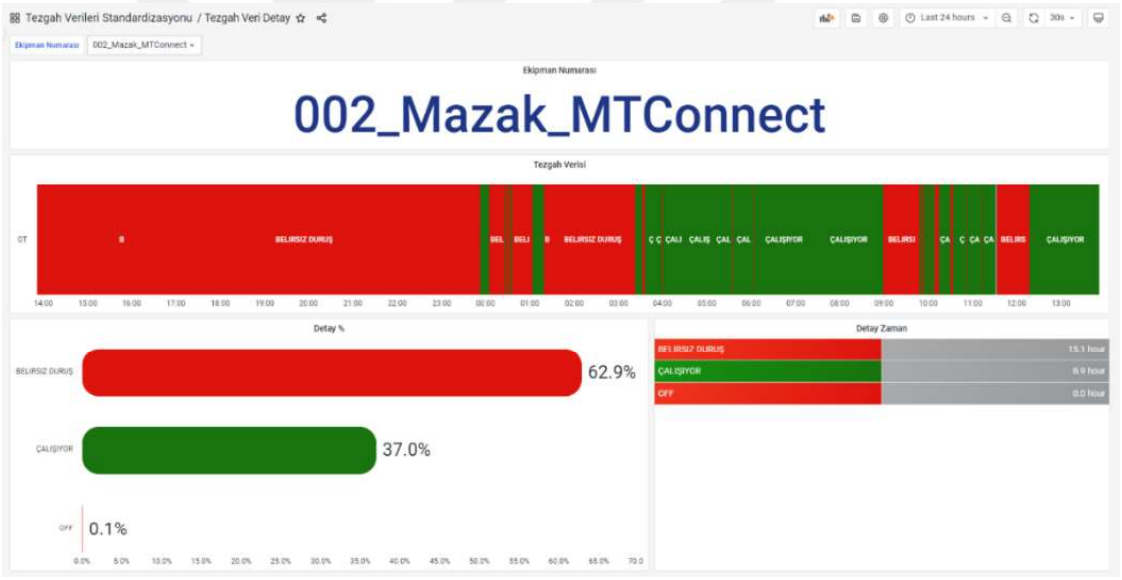
Bu ekran üzerinde tüm tezgahlar gösterilmektedir. Bu tezgahlar çalışırken "yeşil", duruyorken "kırmızı" ve kapalı iken "gri" olarak renklendirilmektedir. Tezgahlar için küçük duruş olarak görülen ve operatörün ara işlemler gerçekleştirdiği beş dakika ve altındaki duruşlar ise ara işlem olarak gösterilmektedir ve " mavi " olarak renklendirilmektedir.

Bu ekranın haricinde her tezgâh özelinde tezgahların detaylı olarak izlenebileceği ekranlarda oluşturulmaktadır. Bu ekranlar her tezgâhın anlık ve geçmiş verilerini grafiksel olarak göstermektedir ve istenilen tarih-zaman aralığında filtrelenebilmektedir. Şekil 4.27, 4.28, 4.29, 4.30, 4.31 ve 4.32'de her tezgâh için özelleştirilmiş ekranlar gösterilmektedir.

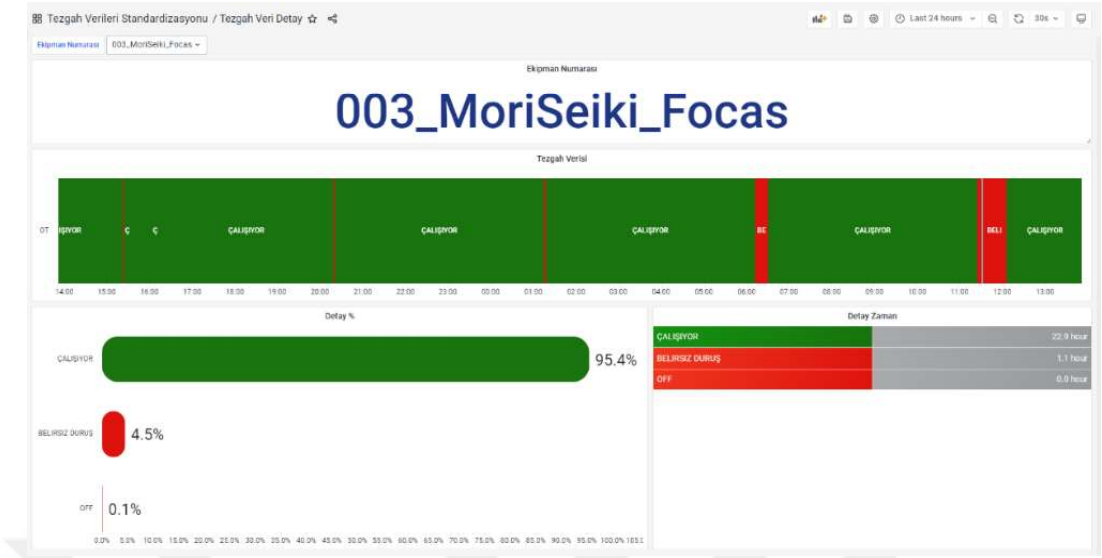
Çalışmanın en önemli ve asıl amacını taşıyan sonucu olarak, aşağıda yer alan şekillerde, çalışmanın tamamında yer alan farklı tip ve değer taşıyan verilerin, aynı grafiğe yerleştirebildiği ortaya koyulmuştur.



Şekil 4.27. 001_Breton_OPC tezgâh durumu görüntüleme ekranı



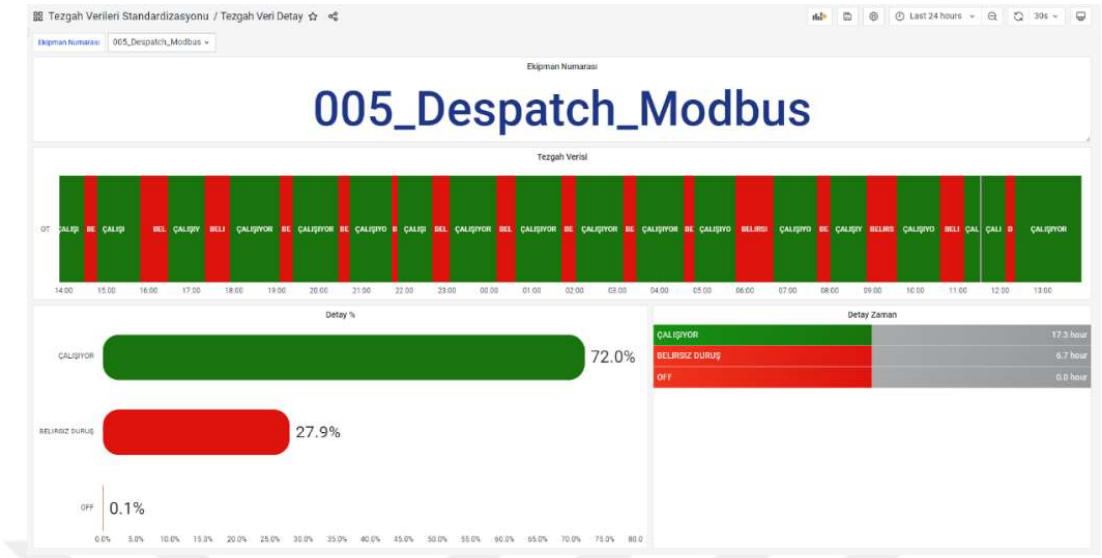
Şekil 4.28. 002_Mazak_MTConnect tezgâh durumu görüntüleme ekranı



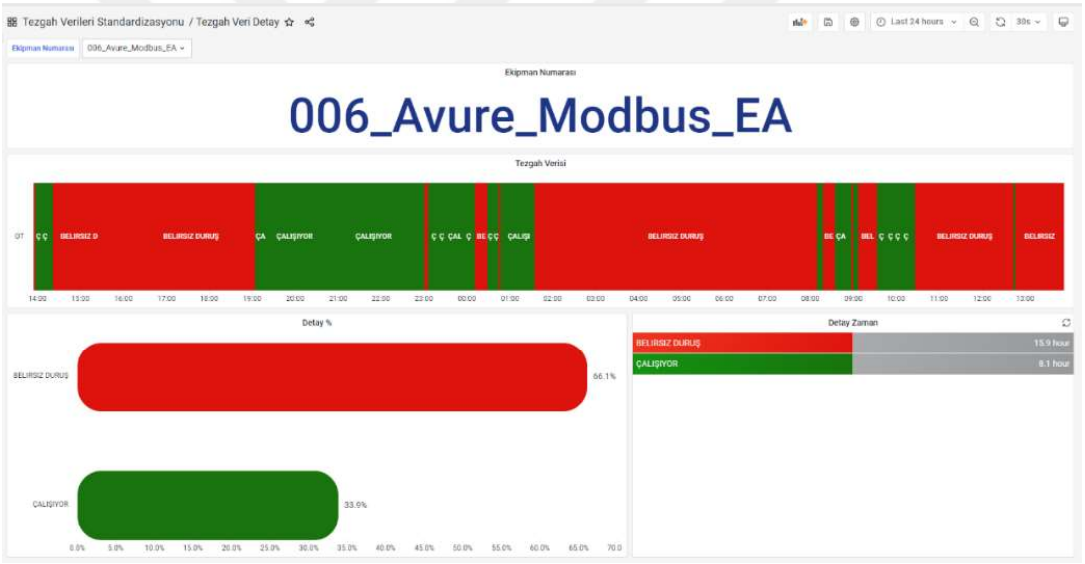
Şekil 4.29. 003_MoriSeiki_Focas tezgâh durumu görüntüleme ekranı



Şekil 4.30. 004_Broetje_BFC tezgâh durumu görüntüleme ekranı



Şekil 4.31. 005_Despatch_Modbus tezgâh durumu görüntüleme ekranı



Şekil 4.32. 006_Avure_MODBUS_EA tezgâh durumu görüntüleme ekranı

5. SONUÇ VE ÖNERİLER

İncelenen literatür çalışmaları kapsamında Endüstri 4.0 kavramının yıllar boyunca gelişimi gözlemlenmektedir. Ayrıca bu kavramın, hayatımızın her evresini derinden etkilediği ve bu etkilerin ilerleyen dönemlerde büyüyerek devam edeceği net olarak görülmektedir. Bu kavramın bileşenlerinden en önemlisinin ise IoT teknolojisinin olduğu çıkarımı yapılabilmektedir. Bu doğrultuda gerçekleştirilen çalışma bir çok açıdan IoT teknolojisinin Endüstri'de kullanılması için bir rehber niteliği taşımaktadır. Bu çalışma sayesinde firmalar, Endüstri 4.0 süreçleri için ihtiyaç duyulan verilere daha kolay ve anlaşılabilir bir şekilde ulaşabilmektedir.

Üretim tezgahlarından toplanan verilerin standardizasyonu, üretim tezgahlarından elde edilen ve farklı protokoller ile toplanana verilerin, ortak bir amaç için standartlaştırılmasını sağlamaktadır. Elde edilen verilerin standartlaştırılması, birbirinden farklı tezgah ve protokollerden toplanan verilerin son kullanıcı tarafından izlenebilirliğini artırmaktadır. Bu kapsamda öncelik tezgahların kullanılabilirliğini ortaya koyan çalışma verisidir. Bu standardizasyon, tezgahlardan toplanan ham verilerde oluşabilecek farklılıkları ortadan kaldırarak tüm yapılardaki üretim tezgahları için standart bir kullanılabilirlik verisinin elde edilmesini sağlamaktadır. Kullanılabilirlik verisi kullanılarak, tezgah doluluk, tezgah takip, tezgah verimliliği, üretim ve yatırım planlamaları gibi raporlar elde edilebilmektedir.

Üretim alanlarında bulunan ekipmanlardan verilerin toplanabilmesi için, ilgili alanlarda IIoT altyapısı kurulmaktadır. Tasarlanan altyapı mimarisi katmanlı bir yapıya sahiptir. Bu yapının temelinde cihazlar yer almaktadır ve bu katman cihaz katmanı olarak adlandırılmaktadır. Çalışma kapsamında bu katmanda üretimde kullanılan farklı tip ve yetkinliğe sahip tezgahlar yer almaktadır. Bu tezgahlar veri yayınlama kabiliyeti ve veri toplama yöntemleri açılarından oluşturdukları farklılıklar neticesinde gruplandırılmaktadır. Bu gruplandırma neticesinde tezgahların destekledikleri protokoller tespit edilmektedir. Bu nedenle bu gruplandırma çalışması altyapının ikinci katmanında yer almaktadır ve kontrolcü katmanı olarak adlandırılmaktadır. Kontrolcü katmanında gruplandırılan bu tezgahların destekledikleri protokoller tespit edilmektedir. Bu protokoller ve haberleşme protokolleri kullanılarak, üretim verilerinin üst katmanlara

taşınması iletişim katmanında olmaktadır. Bu katman yapıları nedeni ile iki farklı protokol içermektedir. Bu protokollerden ilki tezgahların veri yayınlamasını sağlayan protokol iken, diğeri ise bu verilerin sunucu ortamına aktarılmasını sağlayan protokoldür. Üst katmanlara aktarılan bu veriler mimarinin bir üst katmanı olan sunucu katmanında toplanmaktadır. Yayınlanan bu veriler ister ham halleri ile istenilirse farklı uygulamalar kullanılarak işlenerek veri-bilgi katmanına aktarılmaktadır. Veriler bu katmanda uygun veri tabanlarında depolanmaktadır. Depolanan bu veriler uygulama katmanında geliştirilen çeşitli uygulamalar ile son kullanıcılara sunulmaktadır.

Tasarlanan bu mimari çerçevesinde oluşan bu çalışmada, cihaz katmanında altı tezgah yer almaktadır. Bu tezgahlar; Breton ve Mazak markalarına ait beş eksen dikey işleme, Mori Seiki marka üç eksen torna, Broetje marka C-Frame panel montaj, Despatch marka ısıl işlem ve Avure marka form pres tezgahlarıdır. Tezgahlar kontrolcü katmanında incelenerek gruplandırıldığında ilk dört tezgah CNC / NC, diğer iki tezgah ise konvansiyonel tezgah gruplarında yer almaktadır. Ancak bu tezgahlar aynı grup içerisinde yer almalarına rağmen kontrolcülerinin aynı tip fakat farklı markalara sahip olması nedeni ile iletişim katmanında kullanılan protokoller farklılaşmaktadırlar. İlgili tezgahlardan toplanacak olan veriler Ethernet altyapısı kullanılarak sağlanan sunucu haberleşmesi sayesinde, tezgahların desteklediği sırası ile OPC Ua, MTConnect, FOCAS, BFC Gateway, MODBUS protokolleri ve ek donanım olarak eklenen enerji analizöründen MODBUS protokolü ile sunucu katmanına iletilmektedir. Sunucu katmanında Kepware yazılımı kullanılarak veriler bir platform üzerinde toplanmaktadır ve bu yazılımın bir eklentisi olan IoT Gateway servisi kullanılarak veriler yayınlanmaktadır. Bu veriler tezgahlar üzerinde yapılan incelemeler ile kullanılabilirlik durumlarının tespitini sağlamak amacı ile anlamlandırılmaktadır. Bu anlamlandırma çalışması ışığında ham veriler, geliştirilen Java uygulaması tarafından standartlaştırılmaktadır. Standartlaştırılan bu veriler veri-bilgi katmanında SQL veri tabanlarına kaydedilmektedir. Bu katmanda kaydedilen veriler uygulama katmanında geliştirilen Grafana raporları ile görselleştirilerek son kullanıcılara anlamlı ve standart bir şekilde verilerin izlenebilirliğini ve tezgahların takibini sağlamaktadır.

Bu çalışma ilerleyen dönemlerde geliştirilerek, diğer Endüstri 4.0 bileşenlerine hizmet edecek verilerin standartlaştırılması için bir zemin oluşturabilecektir. Bu çalışmada

geliştirilen adımlar takip edilerek, bir yapay zeka uygulaması olan kestirimci bakım çalışmaları için gerekli olabileceği öngörülen tezgah eksenlerine ait motorların, akım, sıcaklık, tork benzeri verileri toplanarak standartlaştırılabilecektir. Bunun haricinde ortaya çıkabilecek ihtiyaçlar çerçevesinde belirlenecek olan verilerin standartlaştırılması için de yol gösterebilecektir. Ayrıca anlatılan mimari kullanılarak farklı tip cihazlardan veriler toplanarak, bu cihazların verilerinin de standartlaştırılması sağlanabilecektir.

Sonuç olarak bu çalışma IIoT altyapısı kurulan firmalarda yer alan farklı tip ekipmanlardan nasıl veri toplanabileceğini anlatmaktadır. Bunun beraberinde ortaya çıkan protokol ve veri farklılıkları kaynaklı karışıklıkları, geliştirilen uygulama ile ortadan kaldırmaktadır. Oluşturulan rapor ekranları sayesinde ise ekipmanların izlenebilirliğini artırarak, ekipmanların kullanılabilirliğini son kullanıcıların anlaşılabilir şekilde takip edebilmesini sağlamaktadır.

Bu çalışma üretim alanlarında denenmiş ve kullanılmaktadır. Ancak yapılan gizlilik anlaşmaları nedeni ile firma bilgileri paylaşılamamaktadır.

KAYNAKLAR

Akben, İ., & Avşar, İ. İ. (2018). Endüstri 4.0 ve Karanlık Üretim: Genel Bir Bakış. Türk Sosyal Bilimler Araştırmaları Dergisi, 3(1), 26-37.

Akram, V., & Dağdeviren, O. (2020). Nesnelerin İnterneti İçin Gerçek Zamanlı Tasarsız Veri Toplama Platformu. Bilişim Teknolojileri Dergisi, 13(4), 451-462.

Alba,M.,2017.” KepServerEX Adds New MQTT Client Driver”. Engineering. https://www.engineering.com/story/kepserverex-adds-new-mqtt-client-driver (9.11.2023)

Altuntaş, E. Y. (2018). Dijital Dönüşüm Uygulamalarının Kurumların Marka Değeri Üzerindeki Etkisi. Ege Üniversitesi İletişim Fakültesi Medya ve İletişim Araştırmaları Hakemli E-Dergisi, (2), 1-18.

Anonim,2018.“Opc Ua Nedir, Neden Kullanılır? ”Gsl. Https://Www.Gsl.Com.Tr/Blog/Dijital-Devrim-Cagnda-Opc-Ua-Nedir-Neden-Kullanlr(16.10.2023)

Anonim,2020.“Endüstri 4.0 Nedir? ”Merpazar. https://blog.merpazar.com/endustri-40 nedir/(7.11.2023)

Anonim,2022.“Digital Manufacturing Automation Using a Standards-Based Approach: An MTConnect Use Case”.IMTS. https://www.imts.com/read/article-details/Digital-Manufacturing-Automation-Using-a-Standards-Based-Approach-An-MTConnect-Use-Case/1506/type/Read/1 (8.11.2023)

Anonim,“S30 Konvansiyonel Silindirik Taşlama Tezgahı”FormCNC. http://formcnc.com.tr/formcnc-urunler90.html (8.11.2023)

Anonim,“CNC Torna Tezgahı”Etasis. https://www.etasis.com.tr/cnc-torna-tezgahi/ (8.11.2023)

Anonim,“DMU 40”DMGMori. https://en.dmgmori.com/products/machines/milling/5-axismilling/dmu/dmu-40 (8.11.2023)

Anonim,“Sinümerik 840DSL Catalog”Siemens.https://mall.industry.siemens.com/mall/es/WW/Catalog/Products/10233717 (8.11.2023)

Anonim,“Variaxis i-500” Mazak. https://www.mazakeu.com/machines/variaxis-i-500/ (8.11.2023)

Anonim,”Fanuc Robodrill CNC Dik İşleme Merkezi”Tandem. https://tandem.com.tr/en/Product/ Fanuc-CNC-Dik- Isleme-Merkezi (8.11.2023)

Anonim,”Konvansiyonel Tezgahlar”CombaKalıp. http://www.combakalip.com/hizmet/konvansiyonel-tezgahlar.html (8.11.2023)

Anonim,” Access MyMachine /OPC UA 828/840 - WLM”Siemens. https://www.dex.siemens.com/ industrialsoftware/machine-tool-software/access-mymachine-opc-ua-828-840-wlm?cclcl=de_DE (8.11.2023)

Anonim.”MODBUS ve Çalışma Mantığı”.3eendustiyel. https://www.3eendustriyel.com.tr/BlogModbus (8.11.2023)

Anonim.”How Machines Communicate: MODBUS Protocol”. ProgWorld. https://prog.world/how-machines-communicate-modbus-protocol/ (9.11.2023)

Anonim.”DMU 75 (FD) Mono Block 2. Nesil”.DmgMori. https://tr.dmgmori.com/ueruenler/makineler/freze/5-eksenli-freze/monoblock/dmu-75-fd-monoblock-2nd (9.11.2023)

Anonim.”Sinümerik 840D SL Controller”.Indiamart. https://m.indiamart.com/proddetail/sinumerik-840d-sl-controller-11435774797.html (9.11.2023)

Anonim.”Ethernet Nedir, Nasıl Çalışır?”.Karel. https://www.karel.com.tr/blog/ethernet-nedir-nasil-calisir (9.11.2023)
Anonim.”Breton CNC Makine Modelleri”.CNCBul. https://www.cncbul.com/breton-cnc-makina-modelleri/ (9.11.2023)
Anonim.”Variaxis i-1050”.Mazak. https://www.mazak.com.tr/tr/machines/variaxis-i-1050/ (9.11.2023)
Anonim.”Electric Riverting Machine CPAC”.Aeroexpo. https://www.aeroexpo.online/prod/broetje-automation/product-173789-20114.html (9.11.2023)
Anonim.”Raf/Rff Hıgh-Performance Batch Furnaces”.Despatch. https://despatchdev.com/RAF-RFF-Batch-Furnace.html (9.11.2023)
Anonim.”Configuring the Fanuc Control”.Scytec. https://dx1.scytec.com/help/dataXchange.htm? configuring_the_fanuc_control (10.11.2023)
Bayuk, M. N., & Demir, B. N. (2019). Endüstri 4.0 Kapsamında Yapay Zekâ ve Pazarlamanın Geleceği. International Journal Of Social, Humanities And Administrative Sciences, 5(19), 781-799.
Bello, O. Zeadally, S. Ve Badra, M. (2017). Network Layer Inter-Operation of Device-To-Device Communication Technologies in Internet of Things (Iot). Ad Hoc Networks, 57:52-62.
Calp, M. H., Bahçekapılı, E., & Berigel, M. (2019). Endüstri 4.0 Kapsamında Akıllı Fabrikaların İncelenmesi.
Çark, Ö. (2020). İşletmelerin Dijital Dönüşüm Sürecinde “Nesnelerin İnterneti” Teknolojisinin Etkisi. Turkish Studies-Economy, 15(3), 1247-1266.
Çavdar, T., & Öztürk, E. (2018). Nesnelerin İnterneti İçin Yeni Bir Mimari Tasarımı. Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 22(1), 39-48.
Dengiz, O. (2017). Endüstri 4.0: Üretimde kavram ve algı devrimi. Makina Tasarım ve İmalat Dergisi, 15:38-45.
Duranek, W.,2023,”Netdata Data Source For Grafana”.WitaliSoft. https://github.com/netdata/netdata-grafana-datasource-plugin (9.11.2023)
Ercan, T., & Kutay, M. (2016). Endüstride Nesnelerin İnterneti (Iot) Uygulamaları. Afyon Kocatepe Üniversitesi Fen ve Mühendislik Bilimleri Dergisi, 16(3), 599-607.
Fanuc.”Communication With Ethernet Board”.Inventcom. https://www.inventcom.net/fanuc-focas-library/general/dnc1(8.11.2023)
Farooq, M. U. Waseem, M. Mazhar, S. Khairi, A. Ve Kamal, T. (2015). A Review on Internet of Things (Iot). International Journal of Computer Applications, 113(1):1-7.
Fortino, G. Guerrieri, A. Savaglio, C. Ve Spezzano, G. (2022). A Review of Internet of Things Platforms Through the Iot-A Reference Architecture. Intelligent Distributed Computing Xiv, 1026:25-35. Doi: 10.1007/978-3-030-96627-0_3
Gabaçlı, N., & Uzunöz, M. (2017, October). IV. Sanayi Devrimi: Endüstri 4.0 ve Otomotiv Sektörü. In Icpess (International Congress On Politic, Economic And Social Studies) (No. 3).
Geelen, P.,2021.” Industry 4.0: is the complex interconnected web of devices resolvable withstandardization?”Medium.https://medium.com/sitechassethealthcenter/industry-4-0-is-the-complex-interconnected-web-of-devices-resolvable-with-standardization-c39522870c38 (8.11.2023)

Göv, S. A., & Erdoğan, D. (2020). Dördüncü Endüstri Devriminin (Endüstri 4.0) Neresindeyiz? İstanbul Gelişim Üniversitesi Sosyal Bilimler Dergisi, 7(2), 299-318.

Grgić, K. Špeh, I. Ve Heđi, I. (2016). A Web-Based Iot Solution for Monitoring Data Using MQTT Protocol. In 2016 International Conference on Smart Systems and Technologies, IEEE, 249-253. Doi: 10.1109/Sst.2016.7765668

Gündüz, M. Z., & Resul, D. A. Ş. (2018). Nesnelerin İnterneti: Gelişimi, Bileşenleri ve Uygulama Alanları. Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi, 24(2), 327-335.

Gürsoy, M. Ü. Çolak, U. C. Gökçe, M. H. Akkulak, C. Ötleş, C. (2019). Endüstri için kestirimci bakım. Uluslararası 3B Yazıcı Teknolojileri ve Dijital Endüstri Dergisi, 3(1):56-66.

Liu, C. Vengayil, H. Zhong, R. Y. Xu, X. (2018). A systematic development method for cyber-physical machine tools. Journal of Manufacturing Systems, 48(C):13-24.

Mısır, O., & Gökrem, L. (2020). Nesnelerin İnterneti İçin MQTT ile Hiyerarşik Haberleşme. Journal Of New Results İn Engineering And Natural Sciences, (12), 1-11.

Mohamed, K. S. (2019). Iot Networking and Communication Layer. The Era of Internet of Things: Towards A Smart World, 49-70.

Oral, O., & Çakır, M. (2017). Nesnelerin İnterneti Kavramı ve Örnek Bir Prototipin Oluşturulması. Mehmet Akif Ersoy Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 1, 172-177.

Özdoğan, E., & Erdem, O. A. (2020). Nesnelerin İnterneti İçin Hibrit Uygulama Katmanı Protokol Tasarımı. Mühendislik Bilimleri ve Tasarım Dergisi, 8(1), 285-304.

Petekci, A. R. (2021). Endüstri 4.0: Fırsat Mı Tehlike Mi? Bilgisayar Bilimleri ve Teknolojileri Dergisi, 2(1), 7-15.

Petru, J., Zlamal, T., Cep, R., & Sadílek, M. (2017). The Effect Of Feed Rate On Durability And Wear Of Exchangeable Cutting İnserts During Cutting Ni-625. Teknik Dergi, 24, 1-6.

Soylu, A. (2018). Endüstri 4.0 ve Girişimcilikte Yeni Yaklaşımlar. Pamukkale Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, (32), 43-57.

Siemens.(2018). "Sinümerik 840DSL/828D Sinümerik Access MyMachine / Opc Ua"(s.18-20). Siemens Konfigürasyon Manueli No: 6FC5397-1DP41-0BA1. Siemens.

Siemens.(2021). "Brownfield Connectivity BFC Analytics"(s.76-77). Siemens Operasyon Manueli No: A5E51374603B AA. Siemens.

Siemens.(2022). "Brownfield Connectivity-Gateway"(s.13). Siemens Fonksiyon Manueli No: A5E49457327B AE. Siemens.

Şen, E.,2023."Endüstri 4.0 Nedir?"TalentCoders. https://www.talentcoders.co/endustri-4-0-nedir/ (7.11.2023)

Taş, H. Y. (2018). Dördüncü Sanayi Devrimi'nin (Endüstri 4.0) Çalışma Hayatına ve İstihdama Muhtemel Etkileri. Opus International Journal Of Society Researches, 9(16), 1817-1836.

Toker, K. (2018). Endüstri 4.0 ve Sürdürülebilirliğe Etkileri. İstanbul Üniversitesi İşletme Fakültesi İşletme İktisadı Enstitüsü Yönetim Dergisi, 29(84), 51-64.

Türkel, S., & Yeşilkuş, F. (2020). Dijital Dönüşüm Paradigması: Endüstri 4.0. Avrasya Sosyal ve Ekonomi Araştırmaları Dergisi, 7(5), 332-346.

Yiğitol, B., & Sarı, T. (2020). Küresel Salgınlar ile Mücadelede Endüstri 4.0 Teknolojilerinin Rolü. Pamukkale Üniversitesi Sosyal Bilimler Enstitüsü Dergisi, (41), 53-73.

Yıldırım, Y. (2020). Farklı Disiplinlerde Endüstri 4.0. Opus International Journal Of Society Researches, 15(21), 756-789.

Yıldız, A. (2018). Endüstri 4.0 ve Akıllı Fabrikalar. Sakarya University Journal Of Science, 22(2), 546-556.

Yılmaz, E. N., Gönen, S., Şanoğlu, S., Karacayılmaz, G., & Özbirinci, Ö. (2021). Endüstri 4.0'ın Gelişim Sürecinde Unutulan Bileşen: Siber Güvenlik. Düzce Üniversitesi Bilim ve Teknoloji Dergisi, 9(4), 1142-1158.

Waddington,S.," Piper Aircraft selects hydroform press".Ismr. https://ismr.net/piper-aircraft-selects-hydroform-press (9.11.2023)