

ระบบฝาแฝดดิจิทัลสำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัติด้วยเทคโนโลยีโลกเสมือนจริง



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต  
สาขาวิชาระบบกายภาพที่เชื่อมประสานด้วยเครือข่ายไซเบอร์ ภาควิชาวิศวกรรมเครื่องกล  
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

ปีการศึกษา 2565

ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Digital Twin for Automatic Sorting Machine with Virtual Reality Technology



Mr. Krid Sakunkawinkorn

A Thesis Submitted in Partial Fulfillment of the Requirements  
for the Degree of Master of Engineering in Cyber-Physical System

Department of Mechanical Engineering

FACULTY OF ENGINEERING

Chulalongkorn University

Academic Year 2022

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	ระบบฝาแฝดดิจิทัลสำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัติด้วยเทคโนโลยีโลกเสมือนจริง
โดย	นายกฤษฎ์ สุกุลวินกร
สาขาวิชา	ระบบกายภาพที่เชื่อมประสานด้วยเครือข่ายไซเบอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ดร.ณัฐพล คำรงค์พลาสีทธิ

---

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้หัวข้อวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรมหาบัณฑิต

.....	คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)	
คณะกรรมการสอบวิทยานิพนธ์	ประธานกรรมการ
.....	
(รองศาสตราจารย์ ดร.รัชทิน จันทร์เจริญ)	อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
.....	
(ดร.ณัฐพล คำรงค์พลาสีทธิ)	กรรมการ
.....	
(รองศาสตราจารย์ ดร.กฤษฎา พนมแข็ง)	กรรมการภายนอกมหาวิทยาลัย
.....	
(ดร.สุวัฒน์ โสภิตพันธ์)	

CHULALONGKORN UNIVERSITY

กฤษฎี สกกุลวินกร : ระบบฝาแฝดดิจิทัลสำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัติด้วยเทคโนโลยีโลกเสมือนจริง. ( Digital Twin for Automatic Sorting Machine with Virtual Reality Technology) อ.ที่ปรึกษาหลัก : ดร.ณัฐพล ดำรงค์พลาสีทธิ์

ระบบฝาแฝดดิจิทัลหรือ คู่ฝาแฝดดิจิทัล นั้นเข้ามามีบทบาทสำคัญในก้าวพัฒนาระบบอุตสาหกรรมไปยังอุตสาหกรรม 4.0 โดยตัวอย่างที่เห็นได้อย่างชัดเจนคืองานในกระบวนการผลิตที่สามารถนำระบบฝาแฝดดิจิทัลเข้ามาประยุกต์ใช้เพื่อเพิ่มศักยภาพในการผลิตได้ โดยระบบฝาแฝดดิจิทัลนั้นสามารถรับส่งข้อมูลกันระหว่างคู่ฝาแฝดดิจิทัล และ ชุดอุปกรณ์จริง ในเวลาจริงได้ โดยสื่อสารผ่านระบบที่มีศักยภาพในการรับส่งข้อมูล ทั้งในเรื่องของ ค่าความหน่วง และ เสถียรภาพในการรับส่ง และ ด้วยระบบการสื่อสารแบบ IoT นั้นจะช่วยให้ระบบฝาแฝดดิจิทัลสามารถตรวจสอบการทำงานของชุดอุปกรณ์เซนเซอร์ต่างๆได้อย่างหลากหลายมากยิ่งขึ้น โดยการนำเทคโนโลยี VR เข้ามาประยุกต์ใช้ในระบุนั้นจะสามารถช่วยให้ผู้ใช้งานรู้สึกเหมือนอยู่หน้าชุดอุปกรณ์และสามารถควบคุม และ ตรวจสอบระบบต่างๆผ่านระบบเครือข่าย IoT จากทุกที่ที่สามารถเชื่อมต่อกับระบบสื่อสารอินเทอร์เน็ตได้

โดยวิทยานิพนธ์ฉบับนี้ได้นำเสนอกระบวนการพัฒนาระบบฝาแฝดดิจิทัลสำหรับชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติสำหรับสาธิตด้วยเทคโนโลยีโลกเสมือนจริง โดยตัวชุดอุปกรณ์คัดแยกวัตถุอัตโนมัตินั้นจะประกอบไปด้วยชุดอุปกรณ์เซนเซอร์และชุดอุปกรณ์ตัวส่งการที่ถูกควบคุมการทำงานผ่านระบบไร้สายแบบ UDP(User Datagram Protocol) และ TCP(Transmission Control Protocol) บนเครือข่ายไร้สายเดียวกันกับชุดบอร์ดคอนโทรลเลอร์ โดยจะมีตัวประมวลผลกลาง Node-RED ที่ติดตั้งเป็นชุดประมวลผลหลักโดยจะเก็บรวบรวมข้อมูลจากทุกชุดของบอร์ดคอนโทรลเลอร์บนชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ และทำการเชื่อมต่อข้อมูลไปยังแอปพลิเคชัน VR ผ่านระบบการสื่อสารแบบ MQTT ผ่านคลาวด์แพลตฟอร์มของ NETPIE2020 โดยจะนำค่าความหน่วงและเสถียรภาพของระบบสื่อสารที่กล่าวมาข้างต้นมานำเสนอและวิเคราะห์ประมวลผล

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

สาขาวิชา ระบบกายภาพที่เชื่อมประสานด้วย ฝายมือชื่อนิสิต .....

เครือข่ายไซเบอร์

ปีการศึกษา 2565 ฝายมือชื่อ อ.ที่ปรึกษาหลัก .....

# # 6370013121 : MAJOR CYBER-PHYSICAL SYSTEM

KEYWORD: Digital Twin, Virtual Reality, VR, IoT, Internet of things, NETPIE2020, MQTT,  
Node-RED

Krid Sakunkawinkorn : Digital Twin for Automatic Sorting Machine with Virtual Reality  
Technology. Advisor: NATTAPOL DAMRONGPLASIT

Digital Twin will play an important role in transforming many of the current industries into Industry 4.0. To realize such system, a Digital Twin needs to be able to share and update data in real-time between physical and virtual world through communication protocol that are optimized for both latency and reliability. IoT-enabled sensors and actuators deployed in the physical system will allow for constant monitoring of machine's state and for making adjustment to the controlling parameters. With the integration of Virtual Reality (VR) technology, Digital Twin can increase work mobility by providing an immersive experience for the worker to operate the machine remotely without the need for being physically present in the factory.

This work presents a framework towards a digital twin implementation of an automatic sorting machine with virtual reality technology integration. The physical machine is equipped with various sensors and actuators that communicate wirelessly using UDP (User Datagram Protocol) and TCP (Transmission Control Protocol) protocol with a local server. Node-RED running on a local server is used for processing acquired sensor data and for controlling the operating sequences. In a virtual world, a digital twin is created inside Unity3D environment that allows for operation using VR headset. The synchronization of data between the physical and virtual world is done using MQTT protocol via NETPIE2020 cloud platform. System latency and reliability of the proposed digital twin are measured and analyzed.

Field of Study: Cyber-Physical System

Student's Signature .....

Academic Year: 2022

Advisor's Signature .....

## กิตติกรรมประกาศ

รายงานวิทยานิพนธ์ฉบับนี้ สำเร็จลุล่วงได้ด้วยดีด้วยความกรุณาจาก ดร.ณัฐพล ดำรงค์พลาสิทธิ์ อาจารย์ที่ปรึกษาวิทยานิพนธ์ ที่ได้ให้คำปรึกษาแนะนำแก้ไขปัญหาต่างๆที่เกิดขึ้นพร้อมทั้งแนวคิดการเพิ่มเติมในการศึกษามาโดยตลอด ผู้จัดทำขอกราบขอบพระคุณเป็นอย่างสูงไว้ ณ โอกาสนี้

ขอกราบขอบพระคุณ ดร.สุวัฒน์ โสภิตพันธ์ วิศวกรอาวุโส และ คุณณัฐพล ต้นสังวรรณ ที่ได้ให้คำปรึกษาในเชิงลึกกับการใช้งานแพลตฟอร์ม NETPIE2020 และ ให้การสนับสนุนชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติเพื่อนำมาพัฒนาต่อยอดสร้างเป็นระบบฝาแฝดดิจิทัลในวิทยานิพนธ์ฉบับนี้ และ กรุณาเป็นเกียรติเป็นกรรมการสอบวิทยานิพนธ์ในครั้งนี้

ขอกราบขอบพระคุณรองศาสตราจารย์ ดร.รัชทิน จันทร์เจริญ และ รองศาสตราจารย์ ดร.กฤษฎา พนมเชิง ที่ได้ให้ความอนุเคราะห์เป็นกรรมการสอบและให้คำแนะนำสำหรับการปรับปรุงวิทยานิพนธ์ฉบับนี้

ขอกราบขอบพระคุณรองศาสตราจารย์ ดร.ลัญฉกร วุฒิสัทธาธิกุลกิจ และ คุณ สืบพงศ์ น้อยศรี ที่ได้คำชี้แนะแนวทางการใช้งานเทคโนโลยีโลกเสมือนจริงเรื่อยมา

สุดท้ายนี้ผู้จัดทำหวังว่าวิทยานิพนธ์ฉบับนี้จะเป็นประโยชน์สำหรับผู้สนใจนำไปใช้ในการพัฒนาระบบต่อไป

จุฬาลงกรณ์มหาวิทยาลัย  
CHULALONGKORN UNIVERSITY

กฤษฎ์ สุกุลกวินกร

## สารบัญ

	หน้า
.....	ก
บทคัดย่อภาษาไทย .....	ก
.....	ง
บทคัดย่อภาษาอังกฤษ .....	ง
กิตติกรรมประกาศ.....	จ
สารบัญ.....	ฉ
สารบัญตาราง.....	ญ
สารบัญภาพ .....	ฎ
บทที่ 1 .....	1
บทนำ.....	1
1.1 ที่มาและความสำคัญของวิทยานิพนธ์ .....	1
1.2 วัตถุประสงค์ของวิทยานิพนธ์ .....	2
1.3 ขอบเขตของวิทยานิพนธ์ .....	2
1.4 ขั้นตอนการดำเนินงาน โดยสรุป.....	3
บทที่ 2 .....	4
งานวิจัยที่เกี่ยวข้อง .....	4
2.1 กลุ่มงานวิจัยระบบการสื่อสาร ระหว่าง โลก Physical และ Digital .....	4
2.1 กลุ่มงานวิจัย Virtual Reality ในงานอุตสาหกรรม.....	7
2.3 กลุ่มงานวิจัย Digital Twin ในงานอุตสาหกรรม .....	9
บทที่ 3 .....	12
แนวคิดและการดำเนินงาน .....	12

3.1 การพัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Physical .....	13
3.1.1 อุปกรณ์ที่ใช้สำหรับชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ .....	13
3.1.1.1 ชุดอุปกรณ์ประมวลผลหลัก .....	13
3.1.1.2 ชุดบอร์ดคอนโทรลเลอร์ .....	13
3.1.1.3 ชุดอุปกรณ์เซนเซอร์ในระบบ .....	14
3.1.1.4 ชุดอุปกรณ์ตัวส่งงานในระบบ .....	14
3.1.2 กระบวนการประมวลผลการทำงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ .....	15
3.1.2.1 ชุดโมดูลที่ 1 .....	16
3.1.2.2 ชุดโมดูลที่ 2 .....	18
3.1.2.3 ชุดโมดูลที่ 3 .....	20
3.1.2.4 ชุดโมดูลที่ 4 .....	22
3.1.2.5 ชุดประมวลผลหลัก Node-RED .....	26
3.2 การพัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital หรือ กลุ่มแพลตฟอร์ม (VR).....	27
3.2.1 การเลือกอุปกรณ์ในการพัฒนาสำหรับ โลก Digital .....	27
3.2.2 ซอฟต์แวร์ และ แพลตฟอร์ม ในการพัฒนา.....	29
3.2.3 อุปกรณ์คัดแยกวัตถุในโลก Digital (VR) .....	30
3.3 ระบบการสื่อสารของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ .....	32
3.3.1 UDP (User Datagram Protocol) .....	32
3.3.2 TCP (Transmission Control Protocol) .....	33
3.3.3 MQTT (Message Queue Telemetry Transport) .....	33
3.3.1.1 NETPIE 2020.....	34
3.3.4 การเชื่อมต่อข้อมูลระหว่างกลุ่มแพลตฟอร์ม .....	35
3.3.4.1 การเชื่อมต่อ ESP8266 กับ Node Red .....	35
3.3.4.2 การเชื่อมต่อ Node-RED กับ NETPIE 2020 .....	35



3.3.4.3 การเชื่อมต่อ NETPIE 2020 กับ คู่ฝาแฝดดิจิทัลผ่านโปรแกรม Unity.....	35
3.4 วิธีการทดสอบความหน่วงและเสถียรภาพของระบบการสื่อสาร.....	39
3.4.1 ซอฟต์แวร์ Wireshark .....	39
3.4.2 กระบวนการทดสอบ ความหน่วง และ เสถียรภาพ ของระบบสื่อสาร .....	40
3.4.2.1 การรับส่งข้อมูล Node-RED และ ESP8266 ผ่าน UDP/TCP .....	40
3.4.2.2 การรับส่งข้อมูล Node-RED และ NETPIE 2020.....	42
3.4.2.3 การรับส่งข้อมูลระหว่างคู่ฝาแฝดดิจิทัล .....	45
บทที่ 4 .....	47
ผลการทดลอง .....	47
4.1 ผลการทดลอง ความหน่วง และ เสถียรภาพ Node-RED และ ESP8266.....	47
4.1.1 ค่าความหน่วงของระบบการสื่อสารผ่านระบบ UDP และ TCP .....	48
4.1.2 เสถียรภาพของระบบการสื่อสารผ่านระบบ UDP และ TCP .....	50
4.2 ผลการทดลอง ความหน่วง และ เสถียรภาพ Node-RED และ NETPIE 2020.....	51
4.2.1 ค่าความหน่วงของระบบการสื่อสารบน แพลตฟอร์ม NETPIE 2020.....	52
4.2.2 เสถียรภาพของระบบการสื่อสารบน แพลตฟอร์ม NETPIE 2020.....	53
4.3 ผลการทดลอง ความหน่วง และ เสถียรภาพ ของคู่ฝาแฝดดิจิทัล .....	54
4.3.1 ค่าความหน่วงของระบบการสื่อสารของคู่ฝาแฝดดิจิทัลผ่านระบบ UDP ,TCP และ MQTT .....	55
4.3.2 เสถียรภาพของระบบการสื่อสารของคู่ฝาแฝดดิจิทัลผ่านระบบ UDP ,TCP และ MQTT .....	59
4.4 สาขิตรบบปฏิบัติการของระบบฝาแฝดดิจิทัลสำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัติ .....	60
4.4.1 กระบวนการทำงานขั้นต้น .....	60
4.4.2 กระบวนการทำงานของวัตถุสีแดง .....	62
4.4.3 กระบวนการทำงานของวัตถุสีเหลือง .....	63

4.4.3	กระบวนการทำงานของวัตถุสีเขียวชมพู .....	64
4.4.4	กระบวนการทำงานของวัตถุสีน้ำเงิน.....	66
บทที่ 5	.....	68
สรุปผล และ ข้อเสนอแนะ	.....	68
5.1	สรุปผลการดำเนินงาน .....	68
5.1.1	ค่าความหน่วงและเสถียรภาพของการสื่อสารแบบ Wireless LAN ผ่านระบบ UDP/TCP ระหว่าง Node-RED และ ESP8266 .....	68
5.1.2	ค่าความหน่วงและเสถียรภาพของการสื่อสารแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020.....	69
5.1.3	ค่าความหน่วงและเสถียรภาพของระบบฝาแฝดดิจิทัล ผ่านการสื่อสารแบบ Wireless LAN UDP /TCP และ MQTT บนแพลตฟอร์มของ NETPIE 2020.....	69
5.1.4	สาธิตระบบปฏิบัติการของระบบฝาแฝดดิจิทัลสำหรับชุดอุปกรณ์ตัดแยกวัตถุอัตโนมัติ .....	70
5.2	ข้อเสนอแนะเพิ่มเติม.....	71
5.2.1	ระบบสื่อสารแบบไร้สายของชุดประมวลผลหลัก Node-RED และ ESP8266 .....	71
5.2.1	ระบบสื่อสารบน แพลตฟอร์มของ NETPIE 2020 .....	71
5.2.2	อุปกรณ์ตัดแยกวัตถุอัตโนมัติในโลก Digital (VR).....	72
บรรณานุกรม	.....	73
ประวัติผู้เขียน	.....	78

## สารบัญตาราง

	หน้า
ตาราง 1 .....	21
ตาราง 2 ค่ามัธยฐาน (Median) ของความหน่วง ของระบบการสื่อสารแบบ UDP และ TCP ในชุดข้อมูลที่แตกต่างกัน .....	48
ตาราง 3 ค่าความเสถียรภาพ ของระบบการสื่อสารแบบ UDP และ TCP ในชุดข้อมูลที่แตกต่างกัน .....	50
ตาราง 4 ค่ามัธยฐาน ( Median ) ของความหน่วง ของระบบสื่อสารแบบ MQTT ระหว่าง Node-RED และ NETPIE 2020 .....	52
ตาราง 5 ค่าความเสถียรภาพ ของระบบสื่อสารแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020 ..	54
ตาราง 6 ค่ามัธยฐาน (Median) ของความหน่วงของระบบสื่อสารของคู่ฝาแฝดดิจิทัล ผ่านระบบ UDP และ TCP .....	56
ตาราง 7 ค่ามัธยฐาน ( Median ) ของความหน่วง ของระบบสื่อสารแบบ MQTT ระหว่าง Unity และ NETPIE 2020 .....	57
ตาราง 8 การเปรียบเทียบระบบสื่อสารที่พัฒนาขึ้นกับงานวิจัยที่ใกล้เคียง .....	58
ตาราง 9 ค่าความเสถียรภาพ ของระบบสื่อสารแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020 ..	59

## สารบัญภาพ

หน้า

รูป 1 ระยะเวลาในการวางแผนพัฒนาระบบเบื้องต้น.....	3
รูป 2 ภาพรวมระบบประมวลผลและการสื่อสารระหว่าง เครื่องจักร และ Microsoft HoloLens [5]..	4
รูป 3 ภาพการแสดงผลบนแว่น Microsoft HoloLens ที่ซ้อนทับกับอุปกรณ์ใน โลก Physical [5] .....	5
รูป 4.สภาพแวดล้อม โลก Digital และ การสั่งงานผ่านอุปกรณ์โลก Physical [6] .....	6
รูป 5 กราฟเปรียบเทียบความรวดเร็วในการส่งข้อมูลระหว่าง LAN และ Cloud Based [6].....	6
รูป 6 กราฟเปรียบเทียบความรวดเร็วในการส่งข้อมูลระหว่าง Restful API และ MQTT [6] .....	7
รูป 7 ภาพจำลอง Digital มอเตอร์รถยนต์ไฟฟ้า และ ภาพจำลอง Digital ชุดเกียร์รถยนต์ไฟฟ้า [7] ..	8
รูป 8 สภาพแวดล้อม ที่ใช้ในการฝึกอบรมในโลกของ Digital และ Physical [7] .....	8
รูป 9 Hermle C42U จากโลก Physical และ Hermle C42U ในโลก Digital [8] .....	9
รูป 10 ระบบฝาแฝดดิจิทัล สำหรับกระบวนการทำงานจำลองในงานอุตสาหกรรม [9] .....	10
รูป 11 ระบบฝาแฝดดิจิทัล สำหรับเก็บแขนหุ่นยนต์สำหรับงานเชื่อม [10] .....	10
รูป 12 ภาพการออกแบบโดยรวมของระบบฝาแฝดดิจิทัล .....	12
รูป 13 (ก) Asus TUF Gaming FX504GM (ข) Skyworth_GN542VF 3 .....	13
รูป 14 NodeMCU V3 base on ESP8266 [14] .....	14
รูป 15 Sharp GP270D810Z0F , Omron EE-SX47 , 16 TCS 3200 และ Mouse Scroll Wheel Encoder [16] [17] [18][16].....	14
รูป 16 Servomotor MG996R , Motor Driver L298N และ DC motor 12V [20] [21] [22] .....	15
รูป 17 การเชื่อมต่อ โดยรวมของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ .....	15
รูป 18 อุปกรณ์คัดแยกวัตถุอัตโนมัติและการแบ่งโมดูลต่างๆของอุปกรณ์ .....	16
รูป 19 ภาพขยายอุปกรณ์ต่างๆใน โมดูลที่ 1 .....	16
รูป 20 Flow Chart การทำงานของ Module 1 .....	17

รูป 21 ภาพขยายอุปกรณ์ต่างๆใน โมดูลที่ 2 .....	18
รูป 22 Flow Chart การทำงานของ Module 2 .....	19
รูป 23 ภาพขยายอุปกรณ์ต่างๆใน โมดูลที่ 3 .....	20
รูป 24 วัตถุประสงค์สำหรับทดสอบระบบการคัดแยก .....	20
รูป 25 Flow Chart การทำงานของ Module 3 .....	22
รูป 26 ภาพขยายอุปกรณ์ต่างๆใน โมดูลที่ 4 .....	23
รูป 27 ภาพขยายชุดผลึกวัตถุที่ 1 และ 2.....	23
รูป 28 การทำงานของชุดผลึกวัตถุที่ 1 จากคำสั่งที่ 1.....	24
รูป 29 การทำงานของชุดผลึกวัตถุที่ 2 จากคำสั่งที่ 2.....	24
รูป 30 การทำงานของชุดผลึกวัตถุที่ 2 จากคำสั่งที่ 3.....	25
รูป 31 กลุ่มการคัดแยกวัตถุที่ถูกกำหนดไว้.....	25
รูป 32 Flow Chart การทำงานของ Module 4 .....	26
รูป 33 Flow Chart การทำงานของ Node-RED บนชุดอุปกรณ์คอมพิวเตอร์พกพา.....	27
รูป 34 ชุดแว่นตาสำหรับสวมใส่ และ ชุดคอนโทรลเลอร์ Meta Quest 2 [26].....	29
รูป 35 โมเดลตั้งต้นเดิมจาก Sketch Up และ โมเดล 3D หลังแก้ไขใน Unity 3D.....	30
รูป 36 ตัวอย่างอนิเมชันการเคลื่อนไหววัตถุบนสายพาน.....	30
รูป 37 ตัวอย่างอนิเมชันการเคลื่อนไหวของชุดผลึกวัตถุ .....	31
รูป 38 ตัวอย่างอนิเมชันการเปลี่ยนสี .....	31
รูป 39 กล้องแสดงระยะทางการเคลื่อนที่ของสายพาน .....	32
รูป 40 ตำแหน่ง Scene ตัวอย่างของ M2MqttUnity .....	36
รูป 41 ตำแหน่ง Scripts สำหรับการตั้งค่าการเชื่อมต่อ Client MQTT Unity .....	36
รูป 42 การแก้ไข clientID ใน M2MqttUnityClient.cs .....	37
รูป 43 การใส่ข้อมูล Broker Address , Port , Token (User Name) และ Secret (Password).....	37
รูป 44 ตัวอย่างการสร้าง Variables สำหรับเชื่อมต่อข้อมูล NETPIE 2020 .....	38

รูป 45 ตัวอย่างการตั้งค่า Decode Message , Public Message และ Subscribe to Topic .....	38
รูป 46 ตัวอย่างการใช้งานซอฟต์แวร์ Wireshark.....	40
รูป 47 ภาพจำลองการทดสอบการรับส่งข้อมูลผ่านระบบ UDP และ TCP .....	41
รูป 48 ตัวอย่างการวัดค่าความหน่วงของ UDP 500 bytes.....	42
รูป 49 ตัวอย่างการวัดค่าความหน่วงของ TCP 500 bytes .....	42
รูป 50 ภาพจำลองการทดสอบการรับส่งข้อมูล Protocol MQTT .....	43
รูป 51 ตัวอย่างการวัดค่าความหน่วงของระบบ MQTT , QoS 0 , QoS 1 , QoS 2.....	44
รูป 52 ภาพจำลองการทดสอบการรับส่งข้อมูล Protocol MQTT .....	45
รูป 53 ภาพจำลองแบบย่อสำหรับการทดสอบการรับส่งข้อมูลบน Protocol UDP และ TCP .....	47
รูป 54 กราฟกล่องสำหรับค่าความหน่วงของระบบการสื่อสารระหว่าง Node-RED และ ESP8266 ผ่านระบบ UDP และ TCP .....	48
รูป 55 ตัวอย่างการรันตีข้อมูลจากโปรแกรม Wireshark.....	49
รูป 56 กราฟกล่องสำหรับค่าความหน่วงของระบบการสื่อสารระหว่าง Node-RED และ ESP8266 ผ่านระบบ TCP เพิ่มเติม .....	49
รูป 57 กราฟแสดงค่าความเสถียรของระบบการสื่อสารระหว่าง Node-RED และ ESP8266 ผ่าน ระบบ UDP และ TCP.....	50
รูป 58 ภาพจำลองแบบย่อสำหรับการทดสอบการรับส่งข้อมูลบน แพลตฟอร์ม NETPIE 2020 .....	51
รูป 59 กราฟกล่องแสดงค่าความหน่วงของระบบการสื่อสาร MQTT บนแพลตฟอร์ม NETPIE 2020.....	52
รูป 60 กราฟแสดงค่าความเสถียรของระบบการสื่อสารระหว่างแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020 .....	53
รูป 61 ภาพจำลองแบบย่อสำหรับการทดสอบรับส่งข้อมูลของฟาสต์แลตติจี้ผ่านระบบสื่อสาร MQTT กับ protocol UDP/TCP .....	55
รูป 62 กราฟกล่องสำหรับค่าความหน่วงของระบบการสื่อสารระหว่าง คู่ฟาสต์แลตติจี้ ผ่านระบบ UDP และ TCP .....	55

รูป 63 ค่าความหน่วงจากการทดสอบการสื่อสารแบบ MQTT ระหว่าง Unity และ NETPIE 2020	57
รูป 64 กราฟแสดงค่าความเสถียรของระบบการสื่อสารของกลุ่มแอปพลิเคชันผ่านระบบ UDP , TCP และ MQTT .....	59
รูป 65 สาคิตกระบวนการทำงานของอุปกรณ์คัดแยกวัตถุอัตโนมัติ.....	61
รูป 66 สาคิตกระบวนการทำงานของวัตถุสีแดง .....	62
รูป 67 สาคิตกระบวนการทำงานของวัตถุสีเหลือง .....	63
รูป 68 สาคิตกระบวนการทำงานของวัตถุสีเขียวชมพู.....	65
รูป 69 สาคิตกระบวนการทำงานของวัตถุสีน้ำเงิน .....	67



## บทที่ 1

### บทนำ

#### 1.1 ที่มาและความสำคัญของวิทยานิพนธ์

ในการก้าวเข้าสู่ยุคอุตสาหกรรม 4.0 ( Industry 4.0 ) หนึ่งในหัวใจสำคัญก็คือการเชื่อมต่อ ผสานข้อมูลกันระหว่างข้อมูลในรูปแบบ Physical และ ข้อมูลในรูปแบบ Digital เข้าด้วยกันโดย และสามารถนำข้อมูลสื่อสารผ่านระบบแพลตฟอร์ม อินเทอร์เน็ตบนทุกสรรพสิ่ง (Internet of things หรือ IoT ) ซึ่งจะทำให้ผู้ใช้งานสามารถเข้าถึงข้อมูลของระบบได้สะดวกและหลากหลายช่องทางมากยิ่งขึ้น โดย การเชื่อมต่อผสานข้อมูลในรูปแบบ Physical และ Digital เข้าด้วยกัน เป็นที่มาของ แนวคิดเทคโนโลยี ผาแฝดดิจิทัล Digital Twin [1] โดย ที่มีการสร้างผาแฝดของระบบการทำงาน ต่างๆไปไว้ในโลก Digital เพื่อที่จะสามารถนำข้อมูลในโลก Digital ไปใช้เพื่อวิเคราะห์เพื่อใช้ในการ พัฒนาระบบให้มีประสิทธิภาพมากยิ่งขึ้น

โดยการสร้าง แอปพลิเคชัน Digital ในปัจจุบันนั้นเทคโนโลยีความเป็นจริงเสมือน ( Virtual Reality ) หรือ วีอาร์ (VR)ได้รับความนิยมมากขึ้นอย่างต่อเนื่องโดยโดยจากสถิติ [2] กล่าวว่ามี ผู้ใช้งานอุปกรณ์ VR ในอเมริกาถึง 50 ล้านคน และ กว่า 47% ของชาวอเมริกา นั้นคุ้นเคยหรือมี ประสบการณ์การใช้งานเกี่ยวกับเทคโนโลยี VR และมีแนวโน้มที่ จะเพิ่มขึ้นในอนาคตอย่างต่อเนื่อง โดยเราจะเห็นได้ว่าในปัจจุบันนี้ บริษัทชั้นนำของโลกเช่น Google , Apple , Microsoft หรือ แม้แต่ผู้ให้บริการ Social Media อย่าง META (ชื่อเดิม FACEBOOK) และ บริษัทอื่นๆ เริ่มให้ ความสำคัญและลงทุน วิจัยเกี่ยวข้องกับผลิตภัณฑ์ และเทคโนโลยี VR อย่างต่อเนื่อง โดยในปี 2021 บริษัท เมตา ได้มีการลงทุนวิจัย เกี่ยวกับเทคโนโลยีดังกล่าว เป็นเงินจำนวนกว่า 10,000 ล้าน ดอลลาร์สหรัฐ และ มีความตั้งใจที่จะผลักดันเทคโนโลยีดังกล่าว ให้เข้ามามีบทบาทสำคัญต่อการใช้ ชีวิตในอนาคตโดยให้คำนิยามชื่อแพลตฟอร์มที่รวบรวมเทคโนโลยีที่เกี่ยวข้องกับการผสานโลก Physical และ Digital เข้าด้วยกันว่า เมตาเวิร์ส (Metaverse) หรือ จักรวาลนฤมิต โดยคาดการณ์ว่า ในอนาคตจะเป็นเทคโนโลยี และ แพลตฟอร์ม ที่จะเข้ามามีบทบาทสำคัญ สำหรับการดำเนินชีวิต และ อาจส่งผลกระทบต่อเทคโนโลยีหลายๆอย่างที่ใช้งานในปัจจุบัน โดยในปัจจุบันนี้ได้เริ่มมีการ นำเอาเทคโนโลยี VR ไปใช้เพื่อพัฒนาองค์ความรู้ในหลายๆสาขาวิชาชีพไม่ว่าจะเป็น ทางการแพทย์ ทางด้านศิลปะการออกแบบ ทางด้านการศึกษา และ ที่สำคัญคือทางด้านงานอุตสาหกรรม ซึ่งจะ เป็น หัวใจหลักที่ทางผู้วิจัยให้ความสนใจในการพัฒนา

โดยทางผู้วิจัยได้สังเกตเห็นถึงศักยภาพในการใช้เทคโนโลยี VR จากงานวิจัยต่างๆที่สามารถ



ช่วยเพิ่มประสิทธิภาพในงานอุตสาหกรรม ไม่ว่าจะเป็น การลดต้นทุนการฝึกอบรมสำหรับอุปกรณ์ที่มีราคาแพง , การกระจายองค์ความรู้ในลักษณะแอปพลิเคชันสำเร็จรูป , การจำลองระบบและประเมินผลการแก้ไขการทำงานของอุปกรณ์ในงานอุตสาหกรรมโดยไม่จำเป็นต้องพึ่งพาอุปกรณ์จริง ซึ่งหากนำ เทคโนโลยี VR ที่กล่าวมาข้างต้นเข้ามาผสมผสานกับเทคโนโลยี ฝาแฝดดิจิทัล ( Digital Twin ) เราจะสามารถยกระดับ ระบบฝาแฝดดิจิทัล ให้มีประสิทธิภาพ และ ให้ผลลัพธ์ในการพัฒนา ระบบให้หลากหลายมากยิ่งขึ้น, โดยอีกหนึ่งระบบสำคัญในการเชื่อมต่อ โลก Physical และ Digital เข้าด้วยกันคือ ระบบการสื่อสารระหว่าง Machine to Machine (M2M) โดยการพัฒนาวิทยานิพนธ์ฉบับนี้นั้นทางผู้วิจัยได้เลือกใช้งานแพลตฟอร์ม NETPIE 2020[3] ที่พัฒนาโดย ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ( National Electronic and Computer Technology Center ) หรือ เนคเทค (NECTEC) มาเป็นอีกหนึ่งปัจจัยหลักในการวิจัยเนื่องจาก NETPIE 2020 เป็นแพลตฟอร์มที่ถูกพัฒนาและมีจุดประสงค์ในพัฒนาระบบอุตสาหกรรม 4.0 ในประเทศไทย.

ในวิทยานิพนธ์ฉบับนี้ทางผู้วิจัยได้นำเสนอถึงกระบวนการพัฒนาระบบ ฝาแฝดดิจิทัล ของอุปกรณ์คัดแยกวัตถุอัตโนมัติ ในรูปแบบของ โลก Physical และ โลก Digital (VR) และผสานอุปกรณ์ทั้งสองโลกให้ทำงานประสานกันแบบผ่านการสื่อสารบนแพลตฟอร์มของ NETPIE 2020.

## 1.2 วัตถุประสงค์ของวิทยานิพนธ์

1.2.1. พัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติสำหรับบริหารจัดการทำงานระบบอัตโนมัติที่มีคู่แฝดดิจิทัล

1.2.2. พัฒนาเทคนิคการสื่อสารระหว่างเครื่องคัดแยกวัตถุอัตโนมัติจริงกับคู่แฝดดิจิทัลในเวลาจริง

1.2.3. วิเคราะห์ความหน่วงและความเสถียรของการสื่อสารของระบบที่พัฒนาขึ้น

## 1.3 ขอบเขตของวิทยานิพนธ์

1.3.1 พัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Physical ให้สามารถส่งผ่านข้อมูล ผ่านระบบการสื่อสารแบบ UDP และ TCP กับชุดประมวลผล Node-RED และเชื่อมต่อเข้ากับระบบ IoT บน แพลตฟอร์มของ NETPIE 2020 ได้

1.3.2 พัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital (VR) ผ่านแพลตฟอร์มการพัฒนา Unity3D ให้สามารถส่งผ่านข้อมูลผ่านระบบ MQTT บนแพลตฟอร์มของ NETPIE 2020 ได้

1.3.3 ผู้ใช้งานสามารถ สั่งงาน ตรวจสอบข้อมูลการทำงาน และ ดูกระบวนการทำงานในรูปแบบ Digital (VR) ของอุปกรณ์คัดแยกวัตถุอัตโนมัติได้

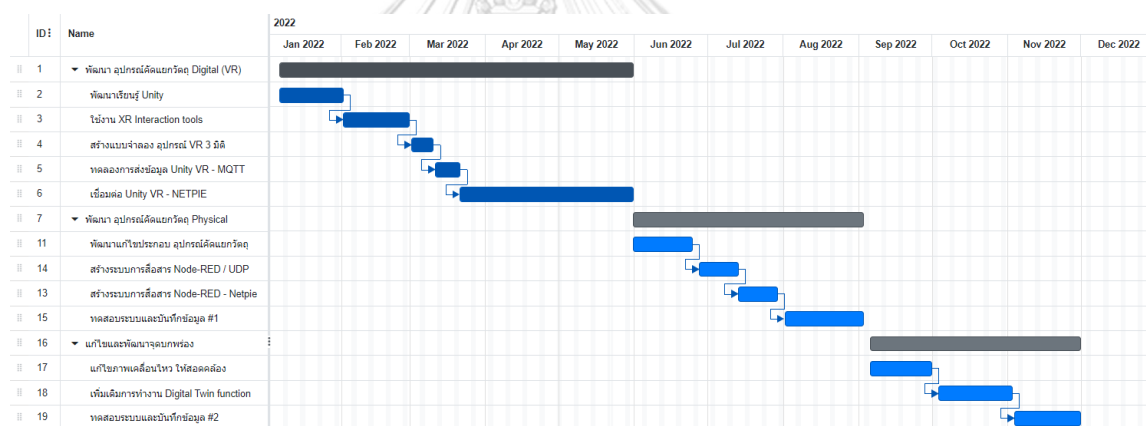
#### 1.4 ขั้นตอนการดำเนินงานโดยสรุป

1.4.1 สร้างและพัฒนาระบบ อุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Physical ให้รองรับการสื่อสารบนแพลตฟอร์ม IoT Node-RED และ NETPIE 2020

1.4.2. สร้างและพัฒนาระบบ อุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital (VR) ให้รองรับการสื่อสารบนแพลตฟอร์ม IoT NETPIE 2020

1.4.3. เชื่อมต่อข้อมูลและกระบวนการทำงานของ อุปกรณ์คัดแยกวัตถุอัตโนมัติ ในโลก Physical และ Digital บน Node-Red และ NETPIE 2020

1.4.4. จัดเก็บข้อมูล เปรียบเทียบการทำงานของ โลก Physical และ Digital



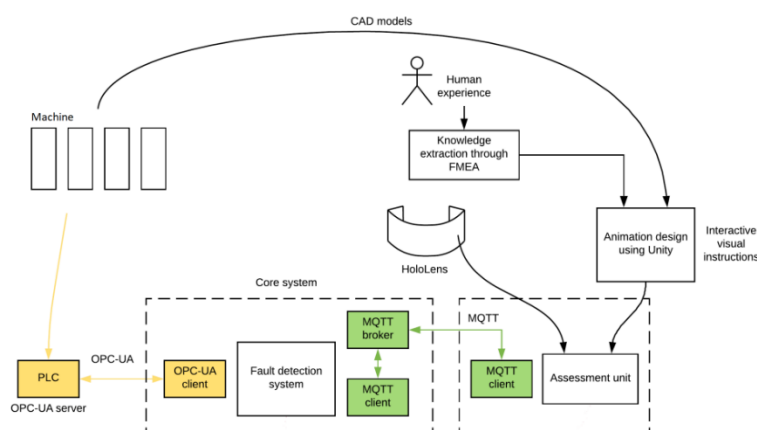
รูป 1 ระยะเวลาในการวางแผนพัฒนาระบบเบื้องต้น

## บทที่ 2 งานวิจัยที่เกี่ยวข้อง

ในบทนี้จะเป็นการกล่าวถึงงานวิจัยเกี่ยวกับเทคโนโลยี Virtual Reality (VR) , Augment Reality (AR) , Mixed Reality (MR) และ ฝาแฝดดิจิทัล Digital Twin โดยจะมุ่งเน้นไปที่งานวิจัยที่มีความเกี่ยวข้องในงานด้านอุตสาหกรรมที่มีผลการทดลองใช้งานจริง ทั้งในเรื่องของ การฝึกอบรม การนำข้อมูลจากโลก Digital ไปใช้งาน และ ขั้นตอนการเลือก ช่องทางการสื่อสารกันระหว่าง โลก Physical และ Digital

### 2.1 กลุ่มงานวิจัยระบบการสื่อสาร ระหว่าง โลก Physical และ Digital

โดย V.Rajan [4] ได้แสดงถึงขั้นตอนของการวางระบบการเก็บรวบรวมข้อมูลเซนเซอร์ต่างๆ ผ่านทางบอร์ดคอนโทรลเลอร์ Node MCU รับส่งข้อมูลผ่านการสื่อสารแบบ MQTT โดยมีตัวกลางในการทดสอบทั้ง HiveMQ , CloudMQTT เพื่อส่งข้อมูลไปยังแอปพลิเคชัน AR บนมือถือเพื่อใช้ในการแสดงค่า อุณหภูมิ ดิจิทัล โดยมีการพัฒนาแอปพลิเคชัน บนแพลตฟอร์มของ โปรแกรม Unity ซึ่งเป็นโปรแกรมที่ได้รับความนิยมในการพัฒนา แอปพลิเคชัน AR VR MR เนื่องจากมีความยืดหยุ่นในการใช้งานและสามารถพัฒนาแอปพลิเคชันบนแพลตฟอร์มได้หลากหลาย , Fernando Arevalo [5] ได้สร้าง ระบบ Mixed-Reality ดังรูปที่ 2. ที่สามารถแจ้งเตือน ผลการวินิจฉัยข้อผิดพลาด ในการทำงานของเครื่องจักรโดยมีการแสดงผลในลักษณะของภาพ Hologram ซ้อนทับไปกับวัตถุจริง โดยมีการพัฒนาแอปพลิเคชันบนอุปกรณ์ Microsoft HoloLens โดยผู้สวมใสจะสามารถเห็น ข้อความการแจ้งเตือนข้อผิดพลาด และ ขั้นตอนของการแก้ปัญหา แบบ real-time โดยเป็นภาพ Hologram ซ้อนทับไปบนวัตถุ



รูป 2 ภาพรวมระบบประมวลผลและการสื่อสารระหว่าง เครื่องจักร และ Microsoft HoloLens [5]

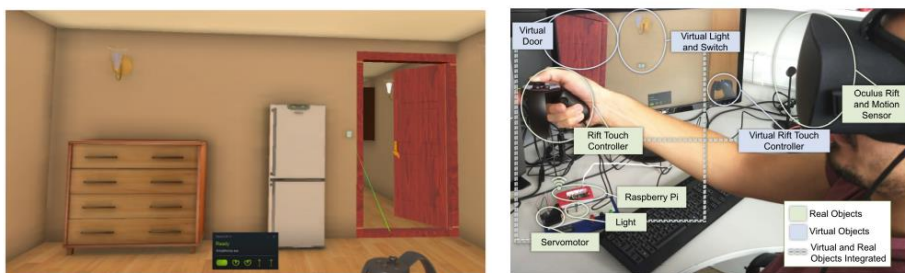
โดยในการแสดงผลภาพ 3 มิติ แบบ Digital ให้ตรงกับ วัตถุจริงใน โลก Physical นั้น ทางผู้วิจัยได้มีการกำหนดจุดมาร์คแบบ QR code ไว้ที่บริเวณ ด้านมุม ของเครื่องจักร จึงทำให้โปรแกรมสามารถวิเคราะห์เพื่อ แสดงภาพ 3 มิติ ซ้อนทับกับวัตถุ ในโลก Physical ได้อย่างใกล้เคียง ดังรูปที่ 3. โดยกระบวนการสื่อสารกันระหว่าง เครื่องจักร และ อุปกรณ์ MR นั้น ทางผู้วิจัยได้เลือกการสื่อสารแบบ MQTT ซึ่งจากผลการทดลองนั้นจะเห็นได้ว่า ระยะเวลาในการรับส่งข้อมูลกันระหว่าง เครื่องจักรและ อุปกรณ์ MR นั้น ใช้ระยะเวลาเพียง 20.509 ms โดยหาก นำไปรวมกับระยะเวลา ประเมินผลของ ส่วนที่มีการทำงานผิดพลาดในตัวเครื่องจักร อีก 30.717 ms นั้น จะได้ระยะเวลาโดยประมาณ ตั้งแต่ตอนที่เครื่องจักรเกิดข้อผิดพลาดจนถึงมีการแจ้งเตือนบนหน้าจอของผู้ใช้อยู่ที่ 51.226 ms ซึ่งถือว่าเป็นระบบการแจ้งเตือนที่รวดเร็ว และสามารถช่วยเหลือผู้ใช้งานในการแก้ปัญหาการทำงานของ อุปกรณ์ได้เป็นอย่างดี



รูป 3 ภาพการแสดงผลบนแว่น Microsoft HoloLens ที่ซ้อนทับกับอุปกรณ์ในโลก Physical [5]

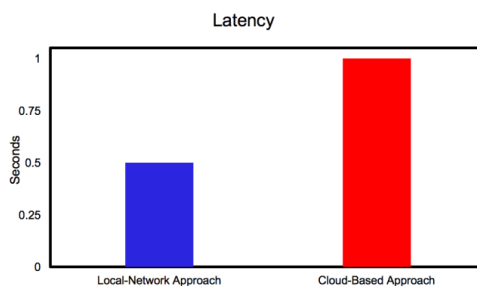
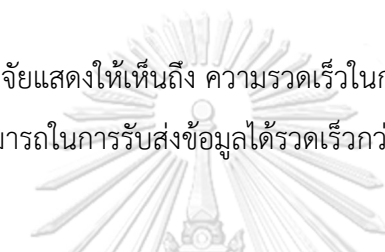
### จุฬาลงกรณ์มหาวิทยาลัย

อีกหนึ่งงานวิจัยที่สามารถแสดงผลของการเลือกระบบการเชื่อมต่อระหว่างโลก Physical และ Digital ได้อย่างน่าสนใจ Anderson Augusto [6] ได้ทำการสร้างระบบการเชื่อมต่อระหว่าง โลก Digital (VR) และ อุปกรณ์ IOT ดังรูปที่ 4. โดยได้ทำการทดลองการใช้ระบบการสื่อสารผ่านระบบ Cloud Base และ Local Area Network โดยได้มีการทดลอง วัดประสิทธิภาพในการรับส่งข้อมูล ในเรื่องของความเร็วในการรับส่ง และ ปริมาณจำนวนข้อมูลในการรับส่ง ระหว่างการสื่อสารแบบ MQTT และ RESTful API โดยการทดลองนั้นผู้วิจัยทำการสร้างสภาพแวดล้อม Digital จำลองห้องที่ประกอบไปด้วยอุปกรณ์สั่งงานแบบ IoT



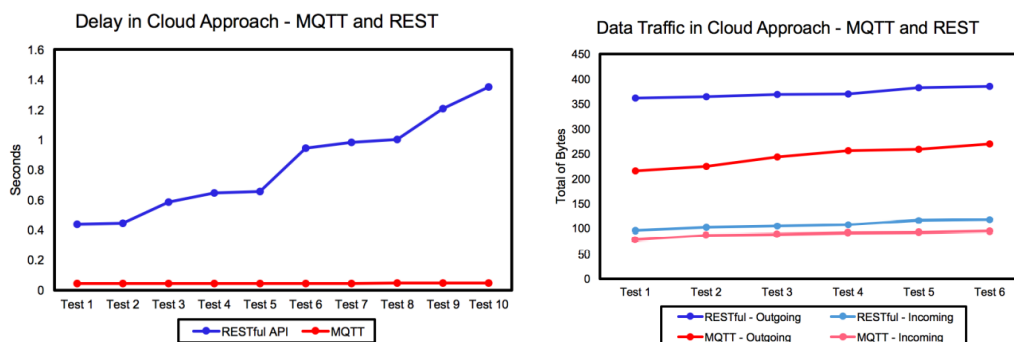
รูป 4.สภาพแวดล้อมโลก Digital และ การทำงานผ่านอุปกรณ์โลก Physical [6]

โดยในผลการทดลองของผู้วิจัยแสดงให้เห็นถึง ความรวดเร็วในการส่งข้อมูล แบบ Local Area Network นั้นจะมีความสามารถในการรับส่งข้อมูลได้รวดเร็วกว่า การใช้งานการสื่อสารผ่านระบบ Cloud-Based ดังรูปที่ 5.



รูป 5 กราฟเปรียบเทียบความรวดเร็วในการส่งข้อมูลระหว่าง LAN และ Cloud Based [6]

โดยระบบการส่งข้อมูลผ่าน Cloud Base ได้มีการเปรียบเทียบ API ที่ใช้ในการสื่อสารระหว่างการ MQTT และ RESTful โดยจากผลการทดสอบการใช้งานผ่าน MQTT มีศักยภาพ มากกว่า Restful API ทั้งในเรื่องของ ความรวดเร็วในการส่งข้อมูลที่รวดเร็วกว่า และ ใช้งาน Data Traffic บนระบบ Cloud น้อยกว่า ดังรูปที่ 6. ซึ่งแสดงให้เห็นว่า การใช้งานผ่านระบบ MQTT นั้น เป็นตัวเลือกที่มี ศักยภาพในการพัฒนาระบบการรับส่งข้อมูล

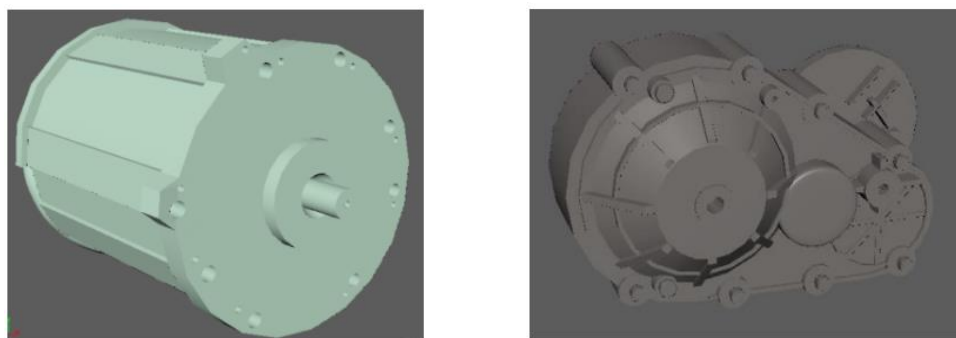


รูป 6 กราฟเปรียบเทียบความเร็วในการส่งข้อมูลระหว่าง Restful API และ MQTT [6]

โดยจากงานวิจัยที่ได้กล่าวมาข้างต้นนั้น จะเห็นได้ว่าการสื่อสารกันระหว่าง อุปกรณ์ในโลก Physical และ Digital นั้น การสื่อสารแบบ Local Area Network จะมีประสิทธิภาพสูงที่สุดในการทำงาน แต่จะขาดความหลากหลายในการเชื่อมต่อกับอุปกรณ์นอกเครือข่าย เพื่อพัฒนาระบบให้สอดคล้องกับงานอุตสาหกรรม 4.0 ทางผู้วิจัย จึงได้เล็งเห็นศักยภาพในการใช้ IoT platform ในการพัฒนาระบบการสื่อสารกันระหว่าง อุปกรณ์ในโลก Physical และ Digital โดย การสื่อสารแบบ MQTT มีประสิทธิภาพการทำงานที่รวดเร็ว ใช้ทรัพยากรน้อย และ มีความยืดหยุ่นในการเชื่อมต่อ อุปกรณ์และ แอปพลิเคชันมาดำเนินการใช้งาน

## 2.1 กลุ่มงานวิจัย Virtual Reality ในงานอุตสาหกรรม

กลุ่มงานวิจัยในหมวดหมู่นี้จะเป็นแนวทางในการใช้งาน เทคโนโลยี VR ในงานอุตสาหกรรม ซึ่งได้มีการนำไปทดลองใช้งานเพื่อศึกษา และ เก็บข้อมูลจากการทดลองใช้งานเพื่อแสดงให้เห็นถึง ศักยภาพในการใช้เทคโนโลยี VR เพื่อนำมาประยุกต์ใช้กับงานทางด้านอุตสาหกรรม 4.0 Prawit Boonmee [7] ได้เห็นความสำคัญสำหรับกระบวนการฝึกอบรมการประกอบชิ้นส่วนรถยนต์ไฟฟ้าที่กำลังจะเกิดขึ้นในโครงการพัฒนาระเบียงเศรษฐกิจพิเศษภาคตะวันออก หรือ EEC (Eastern Economic Corridor) โดยทางผู้วิจัยนั้นได้จัดทำระบบการฝึกจำลองการประกอบชิ้นส่วนรถยนต์ไฟฟ้าในโลก Digital ซึ่งประกอบไปด้วย การประกอบ มอเตอร์รถยนต์ไฟฟ้า ดังรูปที่ 6 และ ชุดเกียร์รถยนต์ไฟฟ้า ดังรูปที่ 7. โดยได้ออกแบบ ระบบการฝึกอบรมโดยอิงจากกระบวนการเรียนรู้จาก ADDIE model ผ่านตัว แอปพลิเคชัน VR เพื่อสร้างสภาพแวดล้อมจำลองการประกอบชิ้นส่วน โดยไม่ต้องพึ่งพาอุปกรณ์ จริง ในการฝึกฝน ดังรูปที่ 8.



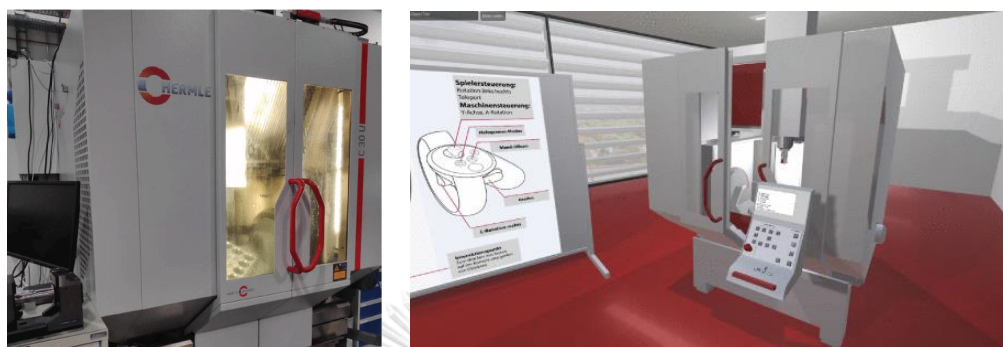
รูป 7 ภาพจำลอง Digital มอเตอร์รถยนต์ไฟฟ้า และ ภาพจำลอง Digital ชุดเกียร์รถยนต์ไฟฟ้า [7]



รูป 8 สภาพแวดล้อม ที่ใช้ในการฝึกอบรมในโลกของ Digital และ Physical [7]

โดยกระบวนการในการฝึกอบรมในแอปพลิเคชัน VR นั้นได้มีการแบ่งกระบวนการฝึกอบรมออกเป็น 3 โหมดได้แก่ 1.โหมดการเรียนรู้ โดย ภายในแอปพลิเคชันจะอธิบายถึงชิ้นส่วนต่างๆ เป็นข้อมูลให้ผู้เรียนรู้ได้เข้าใจกระบวนการทำงานของชิ้นส่วนที่จะต้องทำการฝึกประกอบ 2.โหมดการฝึกฝนตามระบบ โดยระบบจะมีการชี้แนะ ขั้นตอน ทีละขั้นตอน แสดงออกมาให้ผู้ใช้งานเห็นกระบวนการการประกอบชิ้นส่วนรถยนต์ไฟฟ้า 3.โหมดการสอบ โดยผู้เข้าอบรมต้องทำการประกอบชิ้นส่วนอุปกรณ์รถยนต์ไฟฟ้าตามกระบวนการที่ได้ฝึกอบรมมาให้ถูกต้องโดยจากผลการทดสอบผู้เข้าร่วมอบรมจำนวน 10 คน โดยใช้เวลาในการฝึกอบรมเป็นเวลา 30 นาที ผลคะแนน ประเมินความพึงพอใจโดยรวมออกมาอยู่ในเกณฑ์ ที่ดี โดยผลการทดสอบชี้แจ้งว่าการฝึกอบรมด้วย วีอาร์ นั้นช่วยให้ผู้ฝึกอบรมมีความกล้าตัดสินใจในการปฏิบัติการ สามารถนำไปใช้พัฒนาในการปฏิบัติงานได้จริง และเพิ่มความรู้สึกรอยากรู้การทำงานของอุปกรณ์ใหม่ๆ ในอนาคต และอีกหนึ่งงานวิจัยที่โดดเด่นในการสร้างโลกจำลองเสมือนเพื่อใช้ในการฝึกอบรม Dustin KeBler [8] ได้ใช้เทคโนโลยี VR เข้ามาเพื่อใช้ในการฝึกอบรมการใช้งานของเครื่อง CNC Hermle CNC รุ่น C42U รูปที่ 9. โดยมุ่งเน้นโหมดการฝึกอบรม การใช้งานเครื่องโดยเน้นไปที่กระบวนการใช้งานเครื่อง ทั้งการ 1.การ เปิด ปิดอุปกรณ์

และการ ปิดการทำงานฉุกเฉิน 2.การใช้งานแผงควบหน้าอุปกรณ์ 3.การเปลี่ยนแผ่นวัตถุภายในตู้ อุปกรณ์ 4.การเปลี่ยนใบมีดสำหรับงานตัด 5.การปรับตำแหน่ง จุด ออริจิ้น สำหรับหัวเครื่องตัด 6. การปรับเปลี่ยนแปลงจุดอ้างอิงของวัตถุ



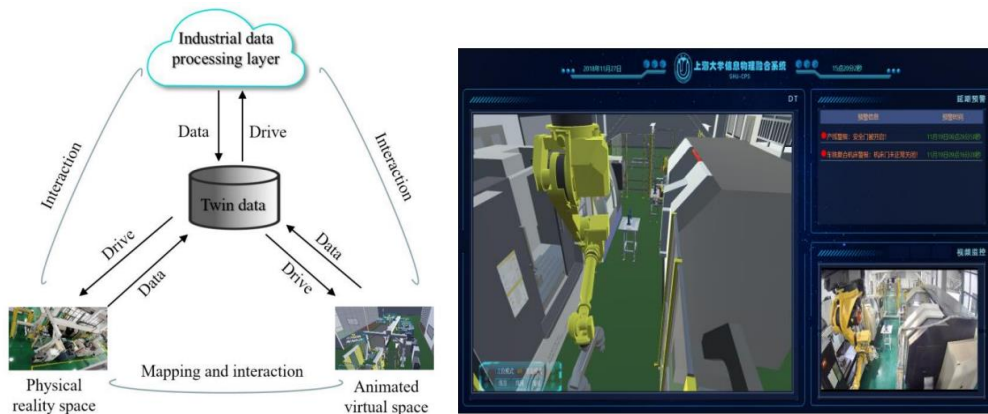
รูป 9 Hermle C42U จากโลก Physical และ Hermle C42U ในโลก Digital [8]

โดยในการทดลองนั้นได้มีการเชิญนักศึกษาวิศวกรรมศาสตร์ สาขาเครื่องกล จำนวน 24 คน เพื่อเข้าไปเรียนรู้ตามกระบวนการดังกล่าวข้างต้น เป็นเวลา 30 นาที และวัตถุที่ได้จากแบบสอบถามประเมินความเข้าใจในการใช้งานอุปกรณ์ของเครื่อง โดยผลการทดลองนั้นแสดงให้เห็นว่าการฝึกอบรมด้วยเทคโนโลยี VR ช่วยให้ผู้ใช้เข้าร่วมสามารถเข้าใจการทำงานของอุปกรณ์ CNC Hermle C42U ตามฟังก์ชันพื้นฐานที่กล่าวมาข้างต้นได้อย่างน่าพึงพอใจ และ ยังเป็นแนวทางในการแก้ปัญหาอุปกรณ์เครื่องจักรที่มีราคาแพงไม่เพียงพอต่อการฝึกอบรม ลดจำนวนผู้เชี่ยวชาญในการทำการฝึกอบรมที่มีผู้เข้าร่วมจำนวนมาก ช่วยสร้างความมั่นใจสำหรับผู้เข้าร่วมอบรมก่อนได้มีโอกาสในการใช้งานกับอุปกรณ์จริงที่มีโอกาสเกิดอุบัติเหตุจากการใช้งานที่ผิดพลาดได้

### 2.3 กลุ่มงานวิจัย Digital Twin ในงานอุตสาหกรรม

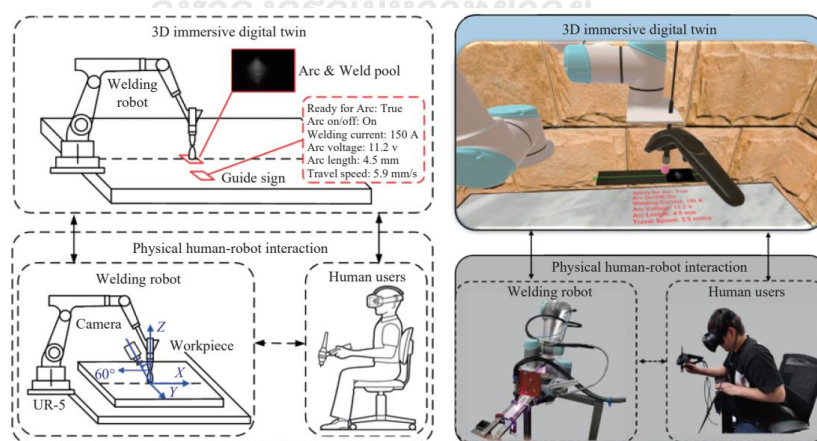
Penfei Wu [9] ได้ทำการพัฒนาระบบ ผาแฝดดิจิทัล (Digital Twin) ที่เชื่อมผสานข้อมูลของกระบวนการทำงานจำลองในงานอุตสาหกรรมระหว่างโลก Physical และ โลก Digital ในรูปแบบของภาพ 3 มิติที่พัฒนาบน Unity 3D ดังรูปที่ 10. โดยระบบฝาแฝดดิจิทัลสามารถแสดงสถานะการทำงานของอุปกรณ์ในแบบ real-time ผ่านการรวบรวมข้อมูล และจัดส่งข้อมูล ในรูปแบบ ของ JSON ซึ่งมีคุณลักษณะการใช้พื้นที่ข้อมูลในการรับส่งที่น้อย ซึ่งทำให้กระบวนการในการรับส่งข้อมูลนั้นมีความรวดเร็วในการรับส่งข้อมูล





รูป 10 ระบบฝาแฝดดิจิทัล สำหรับกระบวนการทำงานจำลองในงานอุตสาหกรรม [9]

และ Qiye Wang [10] ได้จำลองระบบฝาแฝดดิจิทัลสำหรับแขนหุ่นยนต์สำหรับการเชื่อม ดังรูปที่ 11. โดยผู้ใช้งานสามารถควบคุมผ่านอุปกรณ์ วีอาร์ ในโลกความเป็นจริงเสมือน ที่จำลองแขนกลดิจิทัลที่แนบกับอุปกรณ์ที่ใช้ในการเชื่อม โดยเมื่อ ผู้ใช้งานขยับ หรือ สั่งการ เชื่อม ผ่านโลกดิจิทัล อุปกรณ์แขนกลพร้อมที่เชื่อมในโลกแห่งความเป็นจริง ก็จะทำกรปรับเปลี่ยนตำแหน่งให้ และ สั่งการ เชื่อม ให้สอดคล้องกับโลกดิจิทัลที่ผู้ควบคุมได้สั่งงาน โดยผ่านการรับส่งข้อมูลการสื่อสาร แบบ Local Area Network (TCP/IP) และ ในโลกความเป็นจริง นั้นยังมี อุปกรณ์ตรวจจับเซนเซอร์ กระแสไฟฟ้าที่ใช้ในการเชื่อม และ กล้องตรวจจับในอุตสาหกรรมเพื่อส่งข้อมูลกลับไปยังโลกดิจิทัล เพื่อสร้างภาพจำลองเสมือนตำแหน่งของแขนกล ให้แก่ผู้ใช้งาน



รูป 11 ระบบฝาแฝดดิจิทัล สำหรับเก็บแขนหุ่นยนต์สำหรับงานเชื่อม [10]

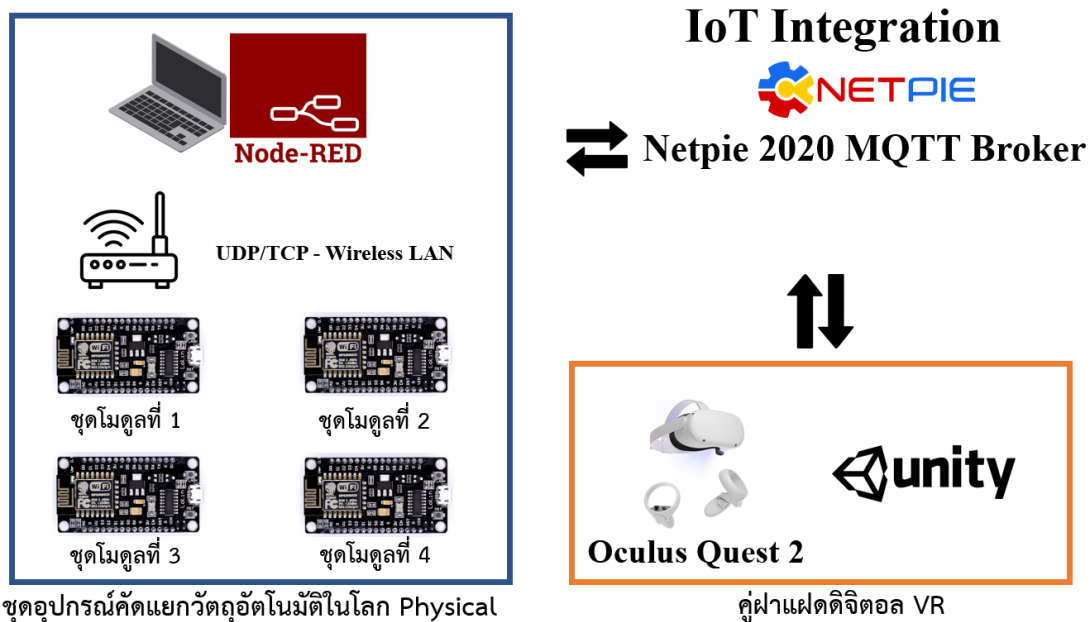
โดยมีการเก็บรวบรวมข้อมูลของการใช้งานโดยมาทำการสร้างโมเดลในการประเมินผลเพิ่มเติมโดยทางผู้วิจัยได้เรียนเชิญผู้ร่วมทดลองที่มีความชำนาญ และไม่ชำนาญ ในงานเชื่อม เพื่อมาทดลองการใช้งานหุ่นยนต์เชื่อมผ่านระบบดิจิทัล และนำข้อมูลเพื่อมาศึกษาเพื่อให้อาจสามารถพัฒนาระบบงานเชื่อมด้วยหุ่นยนต์ให้มีศักยภาพมากขึ้นเทียบเท่ากับ บุคคลที่มีความชำนาญในงานเชื่อม



### บทที่ 3

#### แนวคิดและการดำเนินงาน

ในบทนี้ทางผู้วิจัยจะกล่าวถึงขั้นตอนกระบวนการในการสร้างและพัฒนาระบบฝาแฝดดิจิทัล สำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัติ โดยจะแบ่งออกเป็น 4 หมวดย่อย ได้แก่ 1.การพัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Physical 2. การพัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital (VR) หรือ คู่ฝาแฝดดิจิทัล (VR) 3.ระบบการสื่อสาร และ 4.ขั้นตอนการทดสอบความหน่วงและเสถียรภาพของระบบสื่อสาร โดย ในการวางแผนการพัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ และ คู่ฝาแฝดดิจิทัล พร้อมทั้งระบบสื่อสารนั้น ทางผู้จัดทำได้แสดงภาพออกแบบโดยรวมของระบบฝาแฝดดิจิทัล สำหรับชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติไว้ดังรูปที่12.



รูป 12 ภาพการออกแบบโดยรวมของระบบฝาแฝดดิจิทัล

โดยภาพรวมของระบบฝาแฝดดิจิทัลสำหรับชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ นั้น ทางผู้จัดทำจะใช้ตัวประมวลผลกลาง Node-RED [11] ที่ติดตั้งอยู่บนชุดอุปกรณ์คอมพิวเตอร์พกพาเพื่อสั่งงานชุดบอร์ดคอนโทรลเลอร์ที่ได้ทำการแบ่งหน้าที่การทำงานในส่วนต่างๆของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Physical และ สร้างคู่ฝาแฝดดิจิทัลด้วยเทคโนโลยีโลกเสมือนจริง VR และทำการเชื่อมต่อข้อมูลของชุดอุปกรณ์และคู่ฝาแฝดดิจิทัลในเวลาจริง โดยรายละเอียดในกระบวนการดำเนินงานจะกล่าวในหัวข้อย่อยต่อไป

### 3.1 การพัฒนาชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Physical

#### 3.1.1 อุปกรณ์ที่ใช้สำหรับชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ

ในการออกแบบชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ นั้น ทางผู้วิจัยได้มีแนวคิดในการแบ่งการทำงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ ออกเป็นโมดูลย่อย ๆ ทั้งหมด 4 โมดูล เพื่อแบ่งหน้าที่ในการทำงาน และ เพื่อเป็นการเพิ่มความยืดหยุ่นในการปรับแต่งการทำงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในอนาคตโดยชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติประกอบไปด้วยอุปกรณ์ต่างๆดังนี้

##### 3.1.1.1 ชุดอุปกรณ์ประมวลผลหลัก

ในการออกแบบระบบการทำงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ นั้นทางผู้ออกแบบเลือกใช้ Node-RED เป็นตัวประมวลผลหลักเพื่อส่งงานชุดบอร์ดคอนโทรลเลอร์อีกที โดยได้ติดตั้ง Node-RED อยู่บนอุปกรณ์คอมพิวเตอร์พกพา Asus TUF Gaming FX504 GM [12] ดังรูปที่ 13(ก) โดยมีคุณสมบัติเบื้องต้น ระบบประมวลผล Intel Core i7-8750H RAM 16.0 GB บนระบบปฏิบัติการ Windows 10 Pro 64-bit เป็นตัวประมวลผลหลักในการทำงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ โดย Node-RED ที่ติดตั้งอยู่บนคอมพิวเตอร์จะเชื่อมต่อเข้าสู่ระบบ Wi-Fi 2.4 GHz ผ่าน Router รุ่น Skyworth\_ GN542VF[13] ดังรูปที่ 13(ข). โดยติดตั้งห่างจากชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ และ ชุดอุปกรณ์คอมพิวเตอร์พกพาสำหรับประมวลผล ในระยะในรัศมี 10 เมตร



(ก)



(ข)

รูป 13 (ก) Asus TUF Gaming FX504GM (ข) Skyworth\_ GN542VF 3

##### 3.1.1.2 ชุดบอร์ดคอนโทรลเลอร์

ผู้วิจัยได้เลือกใช้ ชุดบอร์ดคอนโทรลเลอร์ NodeMCU V3 base on ESP8266 [14] รูปที่ 14. ซึ่งรองรับการสื่อสารผ่าน ระบบ Wi-Fi 2.4Ghz 802.11b/g/n , 10 GPIO port รองรับทั้ง PWM , I2C ซึ่งเป็นชุดบอร์ดคอนโทรลเลอร์ที่เหมาะสมสำหรับการเริ่มพัฒนาระบบ IoT โดยเขียนและพัฒนานบนแพลตฟอร์มของ Arduino IDE 2.0.2 [15] โดยจะเป็นชุดบอร์ดคอนโทรลเลอร์หลักที่ใช้ในการเก็บค่าจากเซนเซอร์ต่างๆ และ และส่งงานอุปกรณ์ต่างๆติดตั้งอยู่บนชุดอุปกรณ์คัด

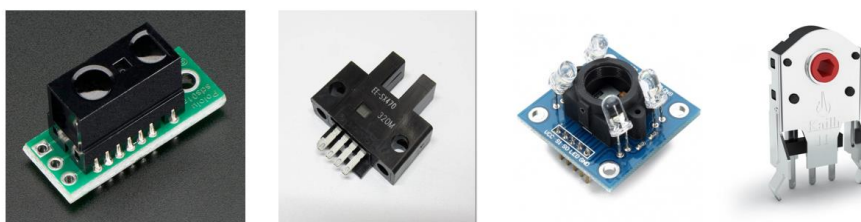
แยกวัตถุอัตโนมัติ โดยชุดคอนโทรลเลอร์ในแต่ละโมดูลจะมีหน้าที่ทำงานที่แตกต่างกัน ซึ่งจะกล่าวถึงในหัวข้อถัดไป



รูป 14 NodeMCU V3 base on ESP8266 [14]

### 3.1.1.3 ชุดอุปกรณ์เซนเซอร์ในระบบ

ในระบบชุดอุปกรณ์คัดแยกวัตถุประกอบไปด้วย Sensor สำหรับตรวจจับวัตถุทั้งหมด 4 ชนิดดังรูปที่ 15. จากซ้ายมือไปขวามือ ได้แก่ 1.Distance Measuring Sensor Sharp GP270D810Z0F [16] หรือ เซนเซอร์ตรวจระยะทางของวัตถุด้วยแสงเลเซอร์ทำหน้าที่ตรวจจับวัตถุในหลอดใส่วัตถุ 2.Photomicrosensors Omron EE-SX47 [17] หรือ เซนเซอร์ตรวจจับวัตถุชนิดแสงทำหน้าที่ตรวจจับชุดผลึกวัตถุ 3.Color Sensor TCS34725 [18] เซนเซอร์ทำหน้าที่ตรวจจับสีของวัตถุ 4.Mouse Scroll Wheel Encoder [19] ติดตั้งสำหรับการคำนวณระยะการเดินทางของระบบสายพาน



รูป 15 Sharp GP270D810Z0F , Omron EE-SX47 , 16 TCS 3200 และ Mouse Scroll Wheel Encoder [16] [17] [18][16]

### 3.1.1.4 ชุดอุปกรณ์ตัวสั่งงานในระบบ

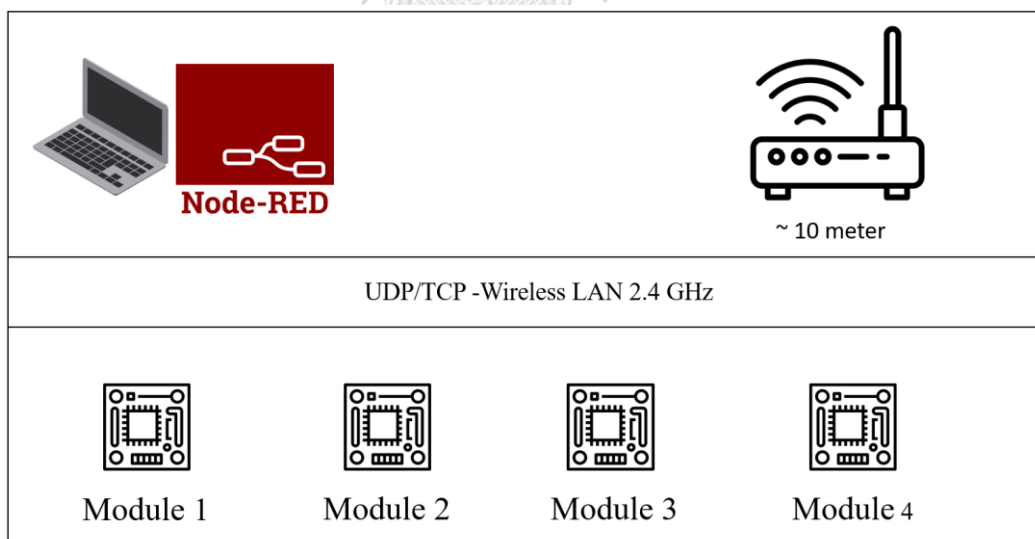
ชุดอุปกรณ์ที่สามารถสั่งการได้ในชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ ดังรูปที่ 16. จากซ้ายมือไปขวามือ ได้แก่ 1.ServoMotor MG996R [20] ทำหน้าที่ขยับตำแหน่งการปล่อยวัตถุจากหลอดใส่วัตถุ 2.Motor Driver L298N [21] เชื่อมต่อกับ DC มอเตอร์ 12V [22] สำหรับขับเคลื่อนระบบสายพาน และ ชุดอุปกรณ์ผลึกวัตถุ



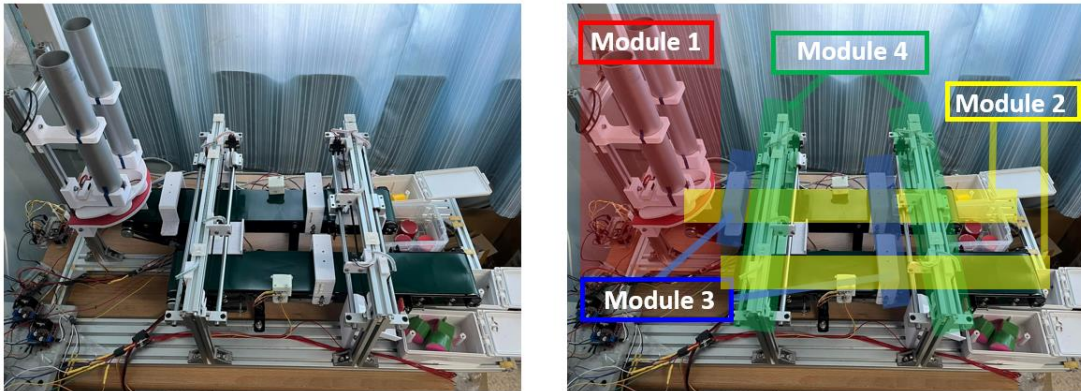
รูป 16 Servomotor MG996R , Motor Driver L298N และ DC motor 12V [20] [21] [22]

### 3.1.2 กระบวนการประมวลผลการทำงานของชุดอุปกรณ์คัตแยกวัตต์อัตโนมัติ

กระบวนการทำงานของชุดอุปกรณ์คัตแยกวัตต์อัตโนมัติ นั้น ระบบประมวลผลหลัก Node-RED ที่ติดตั้งอยู่บนชุดอุปกรณ์คอมพิวเตอร์พกพา จะทำการรับส่งข้อมูล และ ชุดคำสั่ง จาก Node-RED ยังไปชุดบอร์ดคอนโทรลเลอร์ ESP8266 ทั้ง หมด 4 โมดูลผ่านระบบการสื่อสารแบบ UDP[23] หรือ TCP[24] โดยหวังผลมุ่งเน้นการรับส่งข้อมูลที่รวดเร็วเป็นเป้าหมายหลัก ดังรูป ที่17. และ โดยหน้าที่ในการทำงานส่วนต่างๆของชุดอุปกรณ์คัตแยกวัตต์อัตโนมัติ นั้นจะแบ่งออกเป็น 4 โมดูลดังรูปที่ 18.



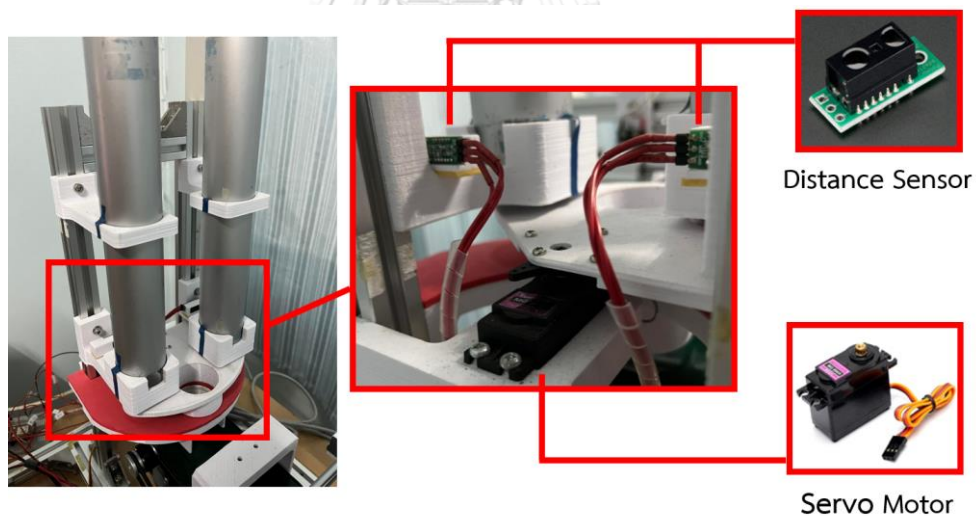
รูป 17 การเชื่อมต่อโดยรวมของชุดอุปกรณ์คัตแยกวัตต์อัตโนมัติ



รูป 18 อุปกรณ์คัตแยกวัตถุอัตโนมัติและการแบ่งโมดูลต่างๆของอุปกรณ์

### 3.1.2.1 ชุดโมดูลที่ 1

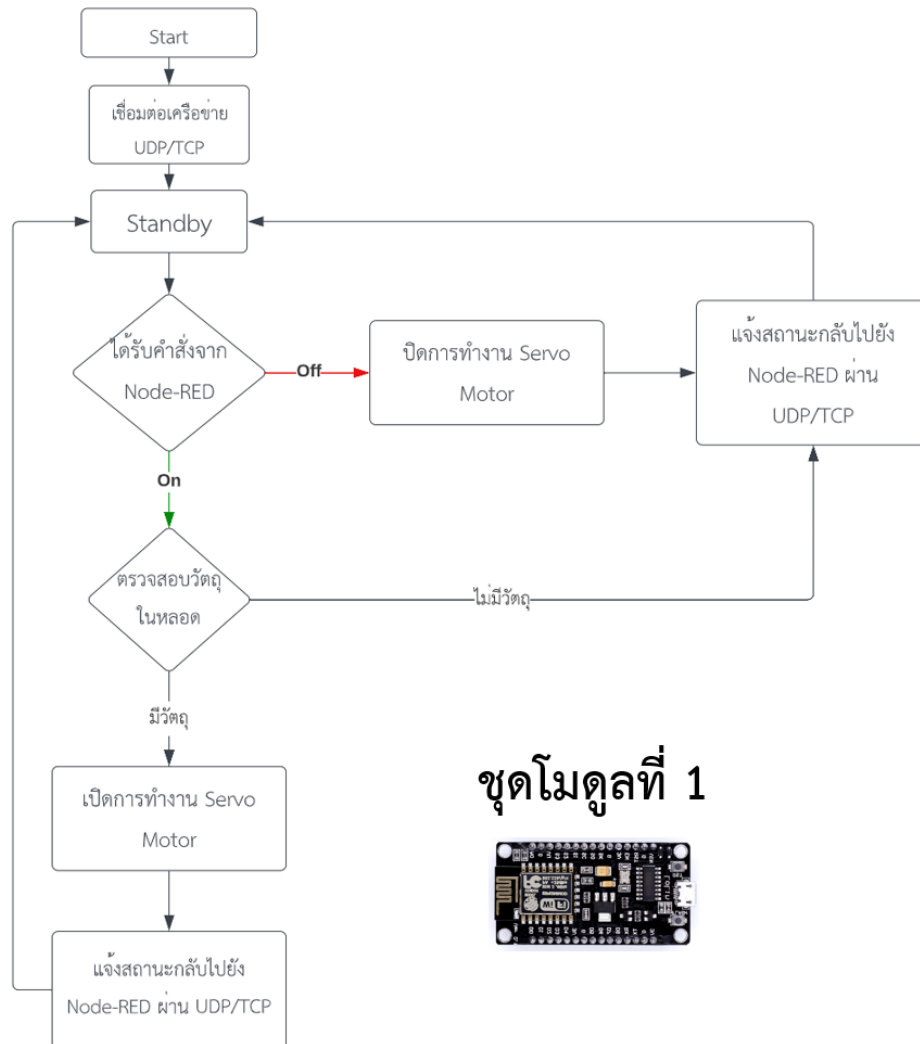
ชุดโมดูลที่ 1 นั้น ESP8266 ของชุดโมดูลที่ 1 ได้ทำการติดตั้ง Servo Motor MG996R สำหรับควบคุมการปล่อยวัตถุจากหลอดใส่วัตถุไปยังสายพาน และ ติดตั้ง Distance Sensor Sharp GP270D810Z0F เพื่อตรวจจับวัตถุในหลอดใส่วัตถุ ดังรูปที่ 19.



รูป 19 ภาพขยายอุปกรณ์ต่างๆในโมดูลที่ 1

โดยในกระบวนการทำงานของโมดูลที่ 1 นั้น นั้นเมื่อ Node-RED สั่งการทำงานมายังชุดคอนโทรลเลอร์ ESP8266 แล้วนั้น ชุดอุปกรณ์โมดูลที่ 1 จะทำการตรวจสอบวัตถุในหลอดใส่วัตถุ หากมีวัตถุพร้อมสำหรับการปล่อยแล้ว จึงจะเริ่มสั่งงาน Servo Motor ให้เริ่มการทำงานโดยการหมุน

สลับแผ่นกั้นหลอดไส้วัตถุ เพื่อให้วัตถุหมุนไปยังสายพาน พร้อมกับแจ้งสถานะกลับไปยัง Node-RED ตามรูป ที่ 20.

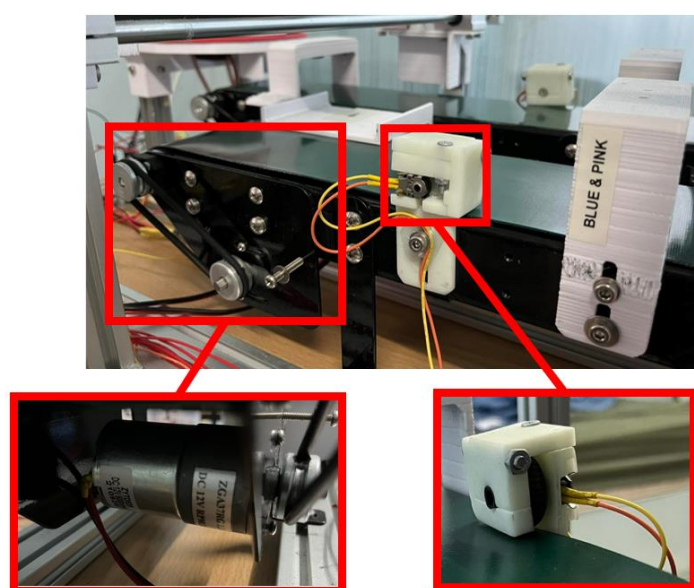


รูป 20 Flow Chart การทำงานของ Module 1



### 3.1.2.2 ชุดโมดูลที่ 2

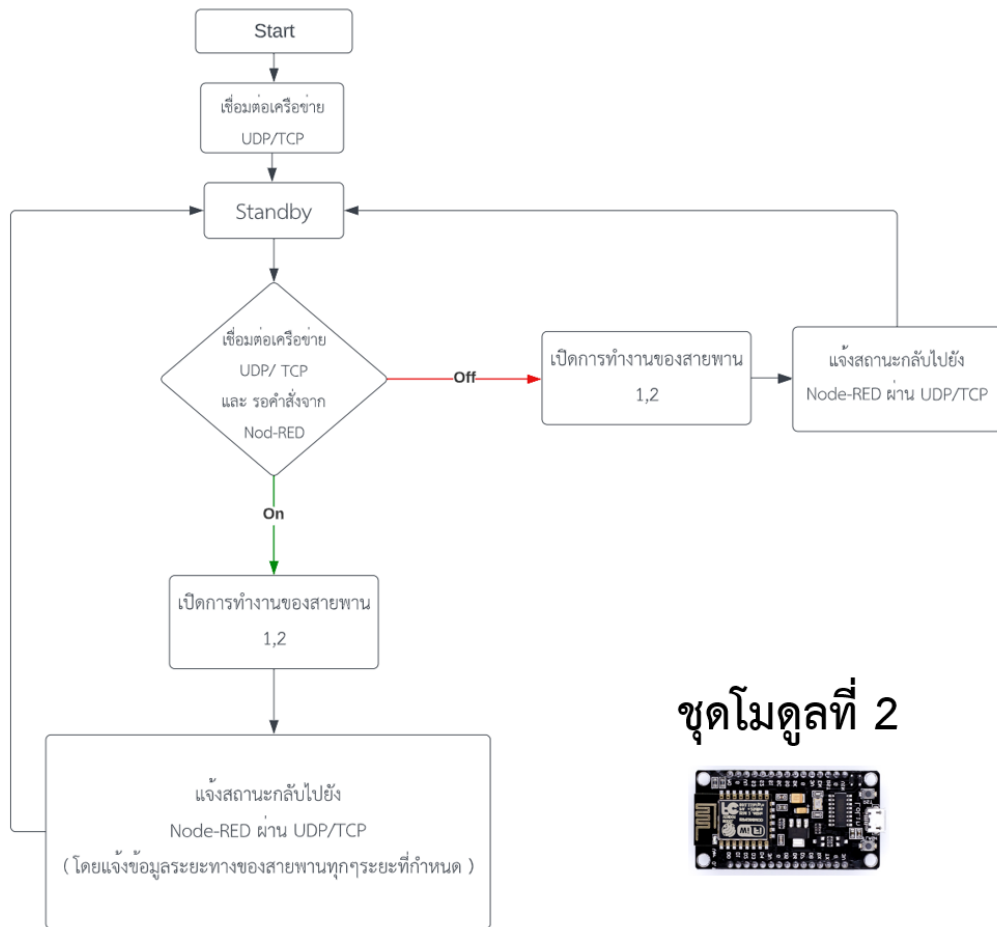
ชุดโมดูลที่ 2 นั้น ESP8266 ของชุดโมดูลที่ 2 ได้ทำการติดตั้ง 12VDC 200 rpm สำหรับควบคุมชุดสายพานในระบบ (สายพานที่ 1 และ 2) และ ติดตั้ง Mouse Scroll Wheel Encoder สำหรับวัดระยะทางการเคลื่อนที่ของชุดสายพาน ดังรูปที่ 21. โดยการทำงานของชุด Mouse Scroll Wheel Encoder นั้น จะติดตั้งล้อลูกกลิ้งและนำไปยึดบริเวณด้านข้างของสายพาน เพื่อให้หน้าสัมผัสของระบบสายพานสัมผัสกับลูกกลิ้ง ทำให้ Mouse Scroll Wheel Encoder จะทำงานเมื่อสายพานเริ่มมีการเคลื่อนที่



12V DC 200 rpm Motor    Mouse Scroll Wheel Encoder

รูป 21 ภาพขยายอุปกรณ์ต่างๆในโมดูลที่ 2

โดยในกระบวนการทำงานของโมดูลที่ 2 นั้นเมื่อ Node-RED สั่งการทำงานมายังชุดคอนโทรลเลอร์ ESP8266 แล้วนั้น ชุดอุปกรณ์โมดูลที่ 2 จะทำการสั่งงาน DC Motor ของระบบสายพานทั้งสองชุดให้เริ่มทำงาน และ ทำการส่งข้อมูลแจ้งเตือนระยะทางของสายพานที่กำลังวิ่ง กลับไปยัง Node-RED ตามระยะทางที่กำหนดไว้ โดยสามารถดูการทำงานโดยรวมของชุดโมดูลที่ 4 ได้ดังรูปที่ 22.



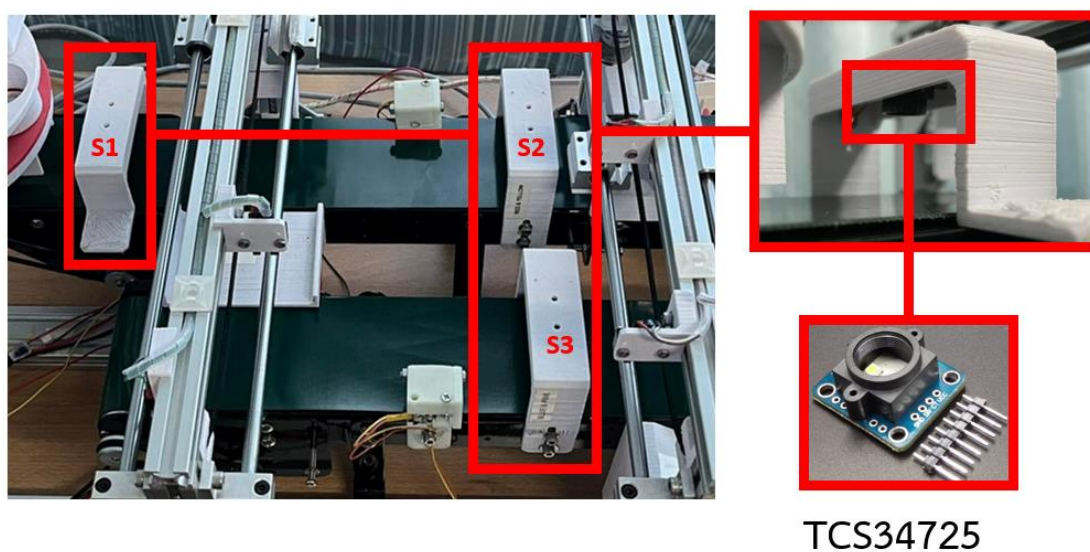
## ชุดโมดูลที่ 2



รูป 22 Flow Chart การทำงานของ Module 2

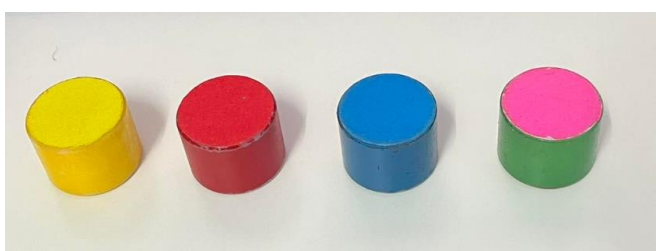
### 3.1.2.3 ชุดโมดูลที่ 3

ชุดโมดูลที่ 3 นั้น ESP8266 ของชุดโมดูลที่ 3 ได้ทำการติดตั้ง Color Sensor TCS34725 รวมทั้งหมด 3 ชุดสำหรับคัดแยกวัตถุโดยติดตั้งบนสายพานทั้งสองข้าง โดยเซนเซอร์สีที่ 1 (S1) จะอยู่ด้านหน้าสุด บนชุดสายพานแรก และ ตามด้วย เซนเซอร์สีที่ 2 (S2) และ 3 (S3) ตามลำดับและ ดังรูปที่ 23.



รูป 23 ภาพขยายอุปกรณ์ต่างๆในโมดูลที่ 3

โดยสำหรับวัตถุที่ใช้เพื่อจำลองระบบคัดแยกนั้น ทางผู้จัดทำได้ใช้วัตถุทรงกระบอก หลากสี โดยวัตถุนั้นแบ่งแยกออกเป็นทั้งหมด 4 สีประกอบด้วย 1.วัตถุสีเหลือง 2.วัตถุสีแดง 3.วัตถุสีน้ำเงิน 4.วัตถุสีเขียวชมพู ดังรูปที่ 24. โดยจุดประสงค์ของอุปกรณ์คัดแยกวัตถุอัตโนมัติในตัวระบบต้องสามารถคัดแยกวัตถุ ทั้ง 4 ชนิดออกจากออกมาเป็นกลุ่มตามที่ได้กำหนดไว้



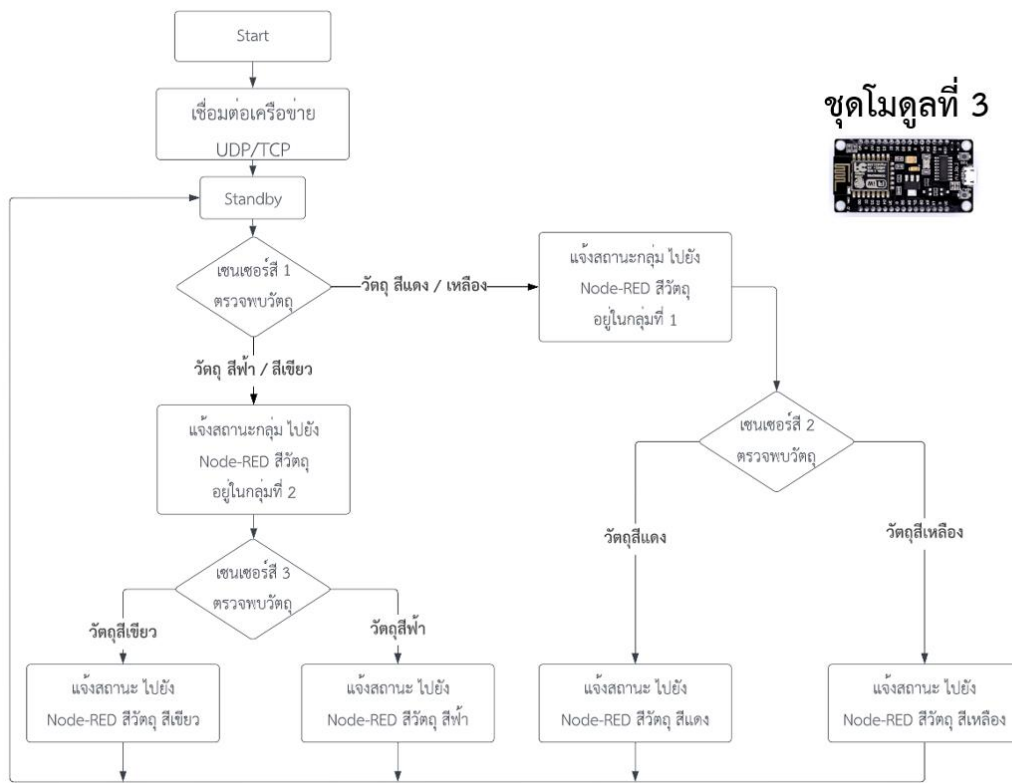
รูป 24 วัตถุสำหรับทดสอบระบบการคัดแยก

โดยในกระบวนการทำงานของโมดูลที่ 3 นั้นเมื่อ ชุดอุปกรณ์โมดูลที่ 3 เริ่มกระบวนการทำงานเพื่อเชื่อมต่อไปยัง Node-RED เสร็จสิ้น ชุดโมดูลที่ 3 จะทำการ stand-by เพื่อรอจนกว่า เซนเซอร์สีที่ 1 จะสามารถตรวจจับวัตถุได้ โดยเมื่อเซนเซอร์สีที่ 1 นั้นตรวจจับวัตถุได้แล้วนั้น เซนเซอร์สี TCS34725 จะทำการบันทึกค่า R G B (ค่าสีแดง สีเขียว สีน้ำเงิน ที่อ่านได้ของวัตถุ) โดยเซนเซอร์ TCS34725 1-3 นั้นจะนำเสนอค่า RGB ที่ไม่เหมือนกันใน และ ค่าที่อ่านได้นั้นมีความผันผวน แปรผันกับสภาพแสง ในสภาพแวดล้อมบริเวณติดตั้งเซนเซอร์ โดยจากการทดลอง ผู้จัดทำได้สรุปผล ค่า RGB ของวัตถุสีต่างๆ จากเซนเซอร์ ที่ 1 ให้ออกมาที่อยู่ในลักษณะที่นำเสนอตามตารางที่ 1.

ตาราง 1

	R	G	B
วัตถุสีแดง	1.00	0.62	0.70
วัตถุสีเหลือง	0.95	1.00	0.60
วัตถุสีน้ำเงิน	0.93	0.78	1.00
วัตถุสีเขียวชมพู	0.27	0.65	1.00

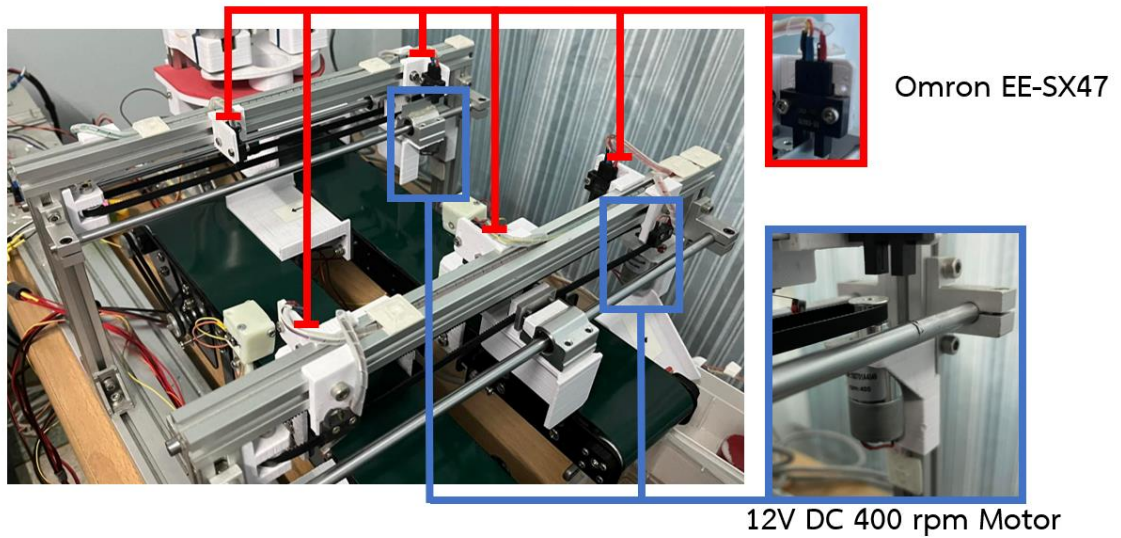
โดยค่าในตารางที่ 1 นั้นนำเสนออัตราส่วนระหว่างค่า R G B ที่อ่านได้เทียบกับค่า R G B ที่มีค่ามากที่สุดของวัตถุนั้นๆ ตัวอย่างเช่น วัตถุสีแดง ค่า R จะเป็นตัวแปรที่มีค่ามากที่สุด (อัตราส่วน 1/1) และ ค่า G อัตราส่วน 0.7 จากค่า R และ ค่า G ในอัตราส่วน 0.62 กับค่า R และ เมื่อนำค่า RGB ที่ได้จากวัตถุไปใช้ในกระบวนการแบ่งกลุ่มของวัตถุนั้น โดยในกลุ่มวัตถุที่ 2 (สีน้ำเงิน , สีเขียวชมพู) จะมีค่าตัวแปร B มากกว่า วัตถุในกลุ่มที่ 1 (สีเหลือง , สีแดง) อย่างมีนัยสำคัญ แต่ว่าค่าอื่นๆนั้นจะมีความใกล้เคียงกันโดยทางผู้จัดทำจึงทำการ แบ่งการแยกวัตถุ ในกลุ่มวัตถุที่ 2 (สีน้ำเงิน , สีเขียวชมพู) ออกมาก่อน โดยเมื่อเซนเซอร์สี ที่ 1 ตรวจจับวัตถุ ที่มีค่าตัวแปร B มากกว่าที่กำหนด จะทำการแจ้งข้อมูลกลุ่มของวัตถุไปยัง Node-RED เพื่อดำเนินการต่อไป และ ในลักษณะการทำงานเดียวกัน เซนเซอร์สีที่ 2 นั้นจะถูกติดตั้งไว้สำหรับแยกวัตถุ สีแดง และ สีเหลือง โดย ใช้ค่าตัวแปร G ในการแบ่งแยก วัตถุสีแดง และ สีเหลือง และ เซนเซอร์สีที่ 3 จะใช้ค่าตัวแปร R ในการแบ่งแยก วัตถุสีเขียวชมพู และสีฟ้า โดยมีกระบวนการทำงานโดยรวมดังรูปที่ 25.



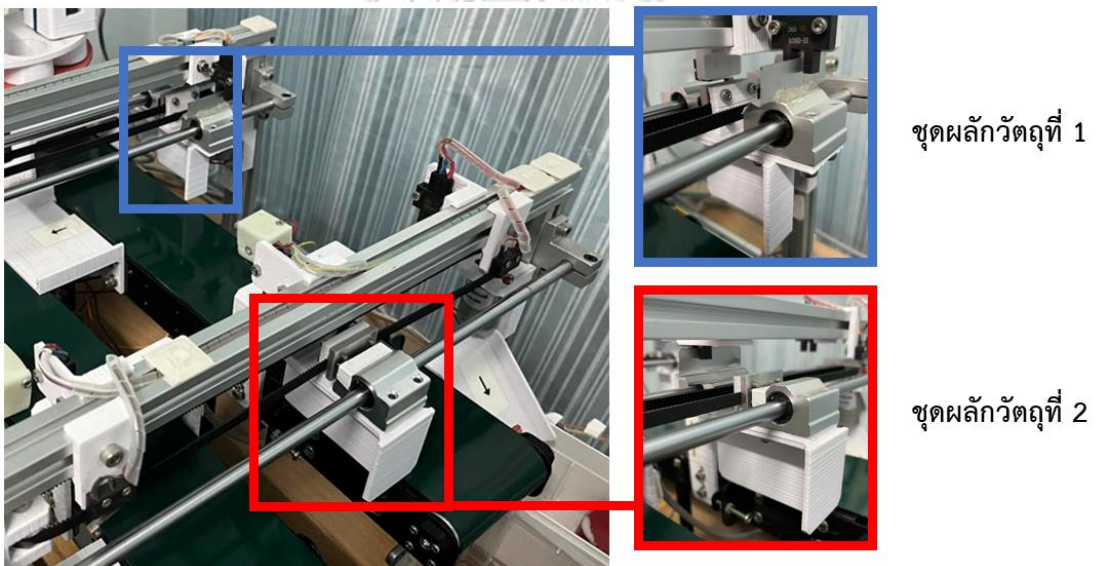
รูป 25 Flow Chart การทำงานของ Module 3

#### 3.1.2.4 ชุดโมดูลที่ 4

ชุดโมดูลที่ 4 นั้น ESP8266 ของชุดโมดูลที่ 3 ได้ทำการติดตั้ง ชุดผลักวัตถุที่ 1 และ ชุดผลักวัตถุที่ 2 และ ติดตั้ง Photo Sensor Omron EE-SX47 ทั้งหมด 5 จุด สำหรับตรวจสอบตำแหน่งการทำงานของชุดผลักวัตถุ ทั้ง 2 ชุด และ Motor 12V DC 400 Rpm 2 ชุด สำหรับสั่งการเคลื่อนไหวของชุดผลักวัตถุ ดังรูปที่ 26. และ 27.



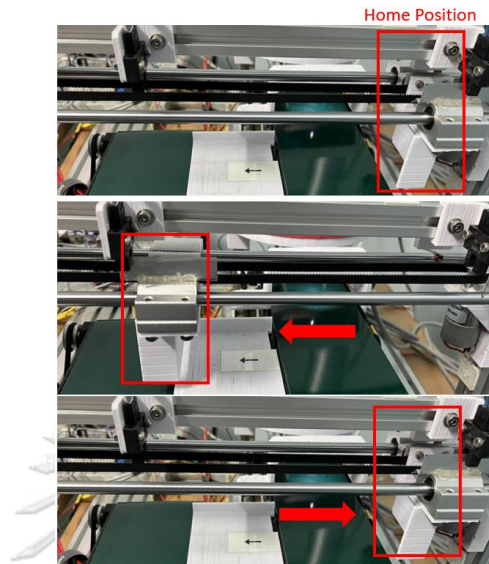
รูป 26 ภาพขยายอุปกรณ์ต่างๆในโมดูลที่ 4



รูป 27 ภาพขยายชุดผลักวัตถุที่ 1 และ 2

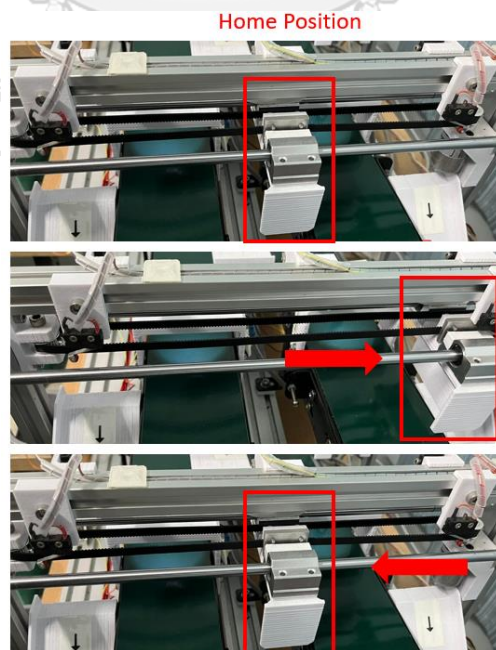
โดยในกระบวนการทำงานของโมดูลที่ 4 นั้นเมื่อ ชุดอุปกรณ์โมดูลที่ 4 เริ่มกระบวนการทำงานแล้วนั้นสามารถเชื่อมต่อการสื่อสารกับ Node-RED ได้ ชุดอุปกรณ์โมดูลที่ 4 จะทำการ Stand-by รอคำสั่งจาก Node-RED โดยชุดคำสั่งที่รอรับจะมีทั้งหมด 3 ชุดคำสั่งได้แก่

คำสั่งที่ 1 จะเป็นการสั่งงานชุดผลักกวัดถูที่ 1 เคลื่อนที่จาก Home Position ของชุดผลักกวัดถูที่ 1 ไปยังสายพานที่ 2 และกลับมายังตำแหน่ง Home Position ชุดผลักกวัดถูที่ 1 ดังรูปที่ 28.



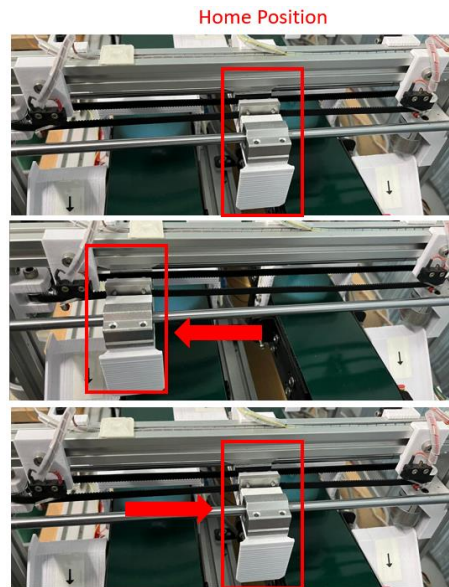
รูป 28 การทำงานของชุดผลักกวัดถูที่ 1 จากคำสั่งที่ 1

คำสั่งที่ 2 จะเป็นการสั่งงานชุดผลักกวัดถูที่ 2 เคลื่อนที่จาก Home Position ของชุดผลักกวัดถูที่ 2 ไปยังสายพานที่ 1 และกลับมายังตำแหน่ง Home Position ของชุดผลักกวัดถูที่ 2 ดังรูปที่ 29.



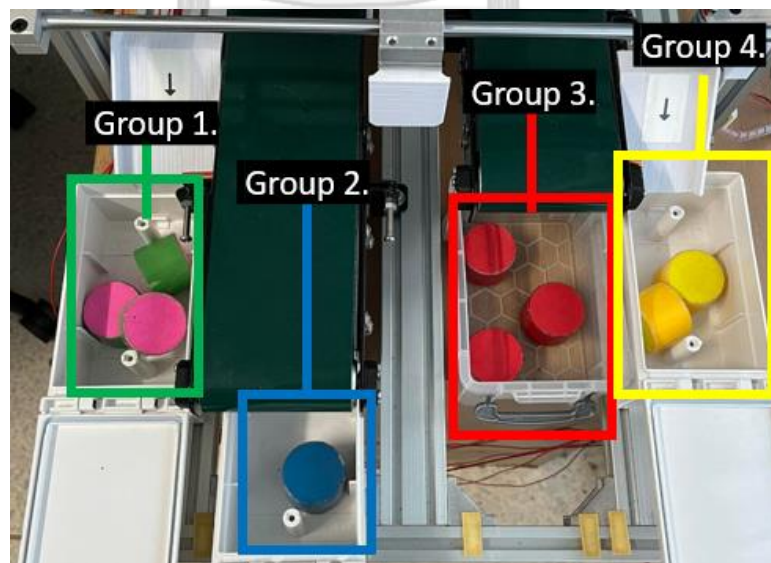
รูป 29 การทำงานของชุดผลักกวัดถูที่ 2 จากคำสั่งที่ 2

คำสั่งที่ 3 จะเป็นการสั่งงานชุดผลักวัตถุที่ 2 เคลื่อนที่จาก Home Position ของชุดผลักวัตถุที่ 2 ไปยังสายพานที่ 1 และกลับมายังตำแหน่ง Home Position ของชุดผลักวัตถุที่ 2 ดังรูปที่ 30.



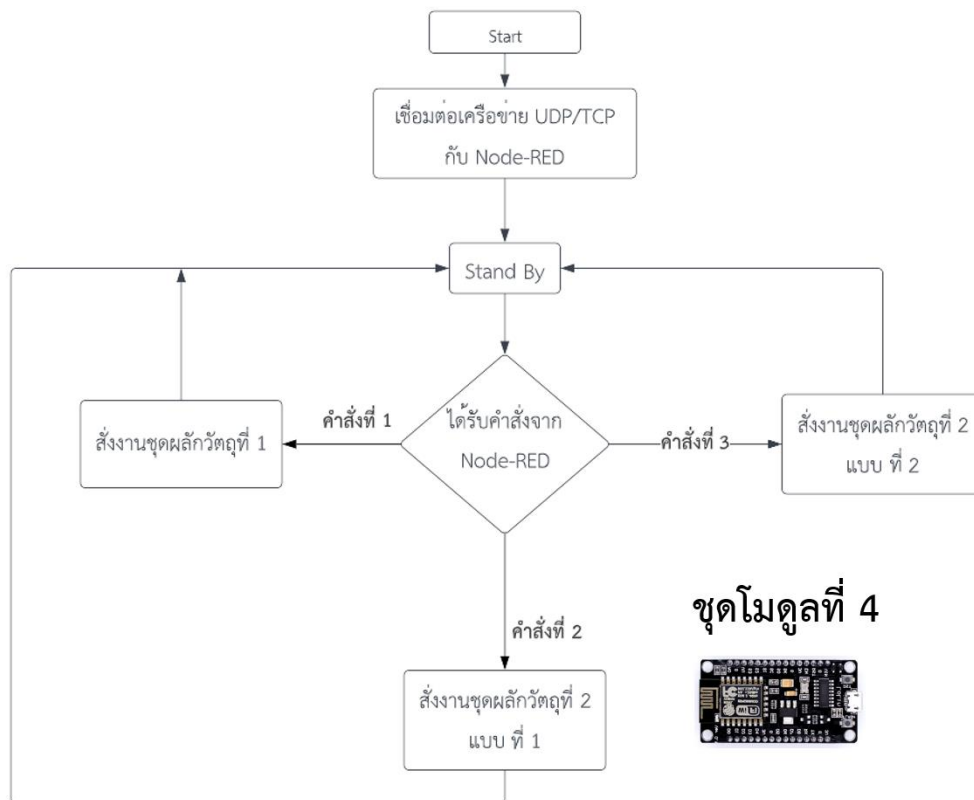
รูป 30 การทำงานของชุดผลักวัตถุที่ 2 จากคำสั่งที่ 3

โดยเป้าหมายในการทำงานของชุดโมดูลที่ 4 คือการแยกวัตถุหลากสีให้ถูกต้องตามกลุ่มที่ได้กำหนดไว้ ดังรูป 31 โดยสามารถดูการทำงานโดยรวมของชุดโมดูลที่ 4 ได้ดังรูปที่ 32.



รูป 31 กลุ่มการคัดแยกวัตถุที่ถูกกำหนดไว้

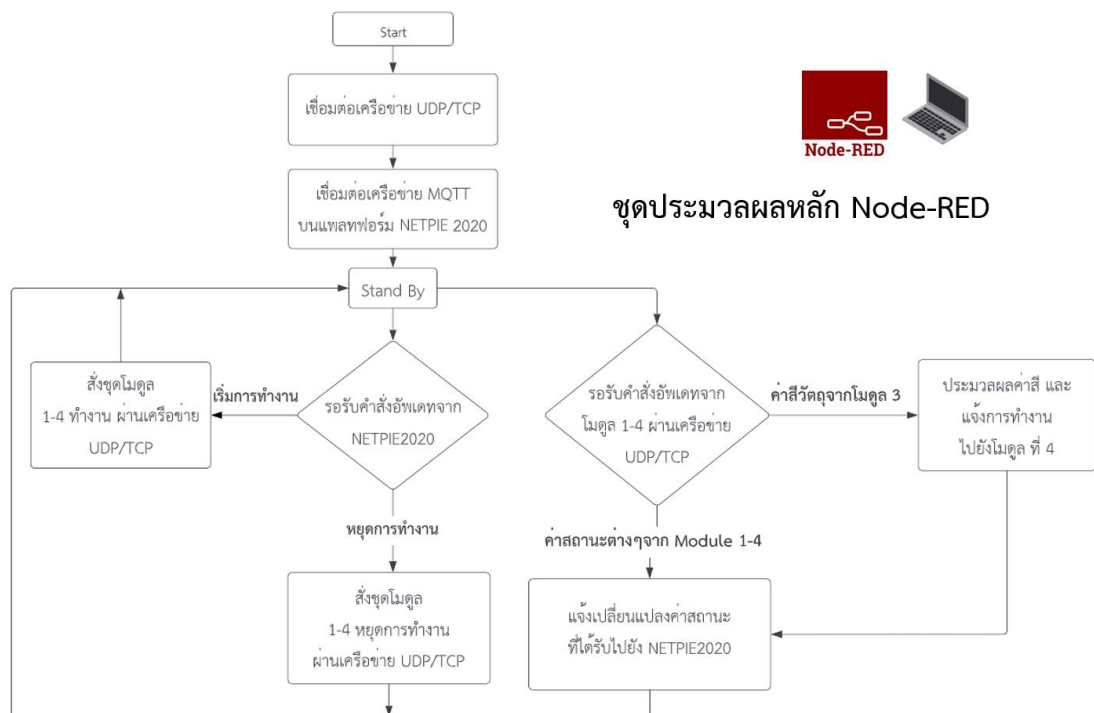




รูป 32 Flow Chart การทำงานของ Module 4

### 3.1.2.5 ชุดประมวลผลหลัก Node-RED

โดยชุดประมวลผลหลัก Node-RED จะถูกติดตั้งอยู่บนชุดอุปกรณ์พกพา คอมพิวเตอร์ และ เชื่อมต่อไปยังชุดโมดูลที่ 1-4 ผ่านระบบไร้สาย Wireless Lan โดยชุดอุปกรณ์ประมวลผลหลักจะทำหน้าที่ส่งงาน และ อัปเดตค่าสถานะ จากชุดโมดูลต่างๆเพื่อนำค่าสถานะที่ได้มา อัปเดตขึ้นระบบคลาวด์ ของ NETPIE 2020 ผ่านการสื่อสารแบบ MQTT โดยสามารถดูกระบวนการทำงานโดยรวมของชุดประมวลผลหลัก Node-RED ได้ดังรูปที่ 33.



รูป 33 Flow Chart การทำงานของ Node-RED บนชุดอุปกรณ์คอมพิวเตอร์พกพา

### 3.2 การพัฒนาชุดอุปกรณ์ตัดแยกวัตถุอัตโนมัติในโลก Digital หรือ คู่ฝาแฝดดิจิทัล (VR)

ในขั้นต้นในการเลือกแพลตฟอร์มในการพัฒนาชุดอุปกรณ์ตัดแยกอัตโนมัติสำหรับโลก Digital นั้นทางผู้วิจัย ได้ทำการศึกษาข้อมูลเพื่อเปรียบเทียบในส่วน ของ ตัวเลือก ในการพัฒนาโลก Digital ได้ 3 กลุ่มใหญ่ๆได้แก่ Virtual Reality , Augmented Reality , Mix Reality

#### 3.2.1 การเลือกอุปกรณ์ในการพัฒนาสำหรับ โลก Digital

โดยทางผู้วิจัยได้ตัดสินใจเลือกใช้ เทคโนโลยี Virtual Reality ในการมาพัฒนาโลก Digital สาเหตุเพราะ มีความหลากหลายของอุปกรณ์ให้เลือกใช้ , เป็นเทคโนโลยีที่กำลังได้รับความนิยม และ จากผลสำรวจของงานวิจัยเทคโนโลยี VR ได้รับผลตอบรับในทางที่ดี และ ผู้ใช้งานมีความพึงพอใจในระดับสูง โดยอุปกรณ์ ที่ใช้ในการใช้งาน Virtual Reality มักเป็นอุปกรณ์ลักษณะเหมือน หน้ากากสวมใส่ครอบดวงตาผู้ใช้งานเพื่อให้ผู้ใช้งานได้เห็นภาพจำลอง 3 มิติ และทำให้ตัวผู้ใช้งานนั้น รู้สึกเหมือนอยู่ในภาพแวดล้อมนั้นๆ และในปัจจุบันมักจะประกอบไปด้วย อุปกรณ์ Controller สำหรับถือ โดยจะจำลอง มือของผู้ใช้งานในโลก Digital โดยใน ปี 2022 นั้น มี อุปกรณ์ ที่รองรับการใช้งาน Virtual Reality มากมาย ได้ถูกจัดจำหน่ายออกมาสู่ท้องตลาด โดย อุปกรณ์ส่วนใหญ่ มักต้องพึ่งพาการใช้งานด้วย คอมพิวเตอร์ เพื่อใช้เป็นตัวประมวลผลกล่าวคือ ผู้ใช้งานจะไม่สามารถใช้

งานอุปกรณ์ VR ได้หากปราศจาก คอมพิวเตอร์ ประมวลผล ซึ่งถือเป็นอีกหนึ่งจุดที่ทำให้การใช้เทคโนโลยี VR มีความลำบากในการใช้งาน แต่ในปี 2020 เดือน ตุลาคม ทาง บริษัท META [25] นั้น ได้จัดจำหน่ายอุปกรณ์ วีอาร์ ชื่อว่า Meta Quest 2 [26] โดยเป็นอุปกรณ์ VR ที่สามารถใช้งานบนตัวอุปกรณ์โดยไม่ต้องพึ่งพา คอมพิวเตอร์ ในการประมวลผล กล่าวคือ Meta Quest 2 นั้นนอกจากจะสามารถรับภาพที่ประมวลผลแล้วนำมาแสดงให้ผู้ใช้งานได้เห็นเหมือนกับ อุปกรณ์ VR โดยทั่วไป ตัวอุปกรณ์เองนั้นก็ยังสามารถที่จะประมวลผลสภาพแวดล้อมในโลก 3 มิติ ด้วยตัวเองได้เช่นกัน โดยเมื่อทางผู้วิจัย ได้ทำการตรวจสอบ คุณสมบัติ Meta Quest 2 เทียบกับ อุปกรณ์ VR อื่นๆในท้องตลาด เพื่อจัดหามาพัฒนา จึงได้ สรุป เลือกใช้ อุปกรณ์ VR เป็น Meta Quest 2 เนื่องจาก เป็นอุปกรณ์ VR ที่มีการใช้งานมากที่สุดท้องตลาดโดยราคาเปิดตัวของอุปกรณ์ อยู่ที่ประมาณ 299 ดอลลาร์สหรัฐ ในอเมริกา หรือ จัดจำหน่ายในไทยประมาณ 12,000 – 12,500 บาท โดยหากเมื่อเปรียบเทียบกับ อุปกรณ์ VR อื่นๆในตลาด ที่มีคุณสมบัติการแสดงผลภาพในความละเอียดสูงใกล้เคียงกันนั้น จะอยู่ที่ราคาประมาณ 30,000 – 40,000 บาท ( HTC Vive, Valve Index , HP Reverb G2 ) [27] [28] [29] และMeta Quest 2 เป็นอุปกรณ์ VR ที่ Controller รองรับในตัว และสามารถที่จะประมวลผลใช้งานแอปพลิเคชัน วีอาร์ โดยไม่ต้องพึ่งพา คอมพิวเตอร์เพื่อเป็นตัวประมวลผล โดยในส่วนของอุปกรณ์ Meta Quest 2 ดังรูปที่ 34. นั้น ประกอบไปด้วย

1.ชุดแว่นตาสำหรับสวมใส่ ( VR head set ) ใช้สำหรับให้ผู้สวมใส่เห็นภาพจำลองสามมิติในโลกเสมือนจริง โดย ชุดแว่นตาสำหรับสวมใส่ภายในขับเคลื่อนด้วยชิปประมวลผล Qualcomm SnapdragonXR2 platform แสดงผลบน จอ LCD ความถี่ 72 – 120 Hz ที่ความละเอียด 1832x1920 pixel per eye และ ติดตั้งกล้องอินฟราเรด 4 ตัว เพื่อใช้สำหรับ วัดระยะพื้นผิวและชุดคอนโทรลเลอร์ ทั้งสองข้าง

2.ชุดคอนโทรลเลอร์ ( Controller ) จำนวนสองชิ้น โดยชุดควบคุมประกอบด้วยชุดด้ามจับจอยสติค และ ปุ่มกด โดยภายในตัวชุดควบคุมได้ทำการติดตั้ง ระบบ อินฟราเรดเพื่อใช้สำหรับส่งสัญญาณให้กับชุดแว่นตาสำหรับสวมใส่ประมวลผล พร้อม กับติดตั้ง มิเตอร์วัดความเร่ง และ เครื่องมือวัดการหมุนรอบ ไจโรสโคป เพื่อสามารถแสดงตำแหน่งของ ชุดคอนโทรลเลอร์ บนแว่นตาสำหรับสวมใส่ได้



รูป 34 ชุดแว่นตาสำหรับสวมใส่ และ ชุดคอนโทรลเลอร์ Meta Quest 2 [26]

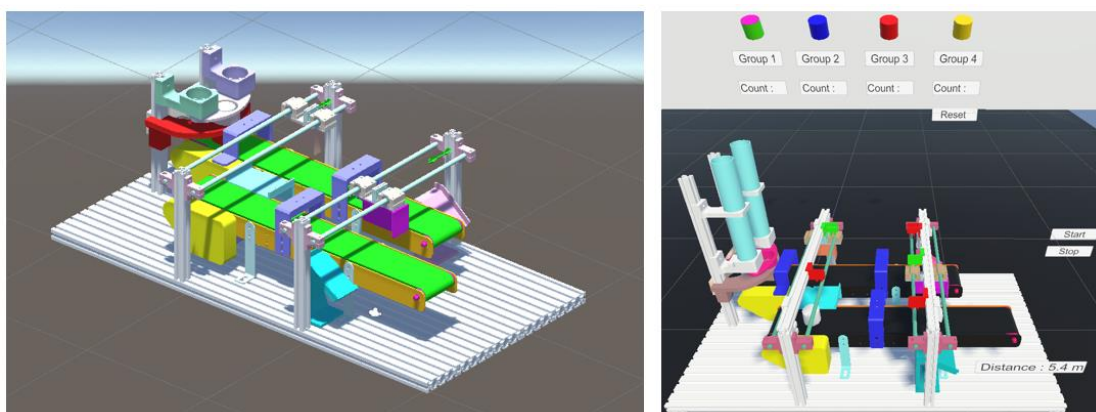
### 3.2.2 ซอฟต์แวร์ และ แพลตฟอร์ม ในการพัฒนา

ในการพัฒนา แอปพลิเคชัน สำหรับอุปกรณ์ VR Meta Quest 2 นั้น Game Engine หรือ แพลตฟอร์มที่สามารถพัฒนา แอปพลิเคชัน ที่สามารถ แก้ไข และดูผลการทำงานได้ แบบ Real Time ที่ได้รับความนิยมได้แก่ Unreal Engine 5 [30] พัฒนาโดย บริษัท Epic Games , เป็นซอฟต์แวร์ที่สามารถพัฒนา แอปพลิเคชัน 3 มิติ โดย ผู้พัฒนาสามารถใช้งานแพลตฟอร์มได้ฟรี โดย จะคิดค่าใช้งานอุปกรณ์ต่อเมื่อผู้พัฒนาสามารถทำเงินจาก แอปพลิเคชันที่พัฒนาบนแพลตฟอร์มได้ เกิน 1 ล้าน ดอลลาร์สหรัฐ และ Unity Engine [31] พัฒนาโดย Unity Technologies , เป็นซอฟต์แวร์ที่รองรับการพัฒนาแอปพลิเคชัน 3 มิติ โดยมีการกำหนด กำหนดราคา แต่ละ เวอร์ชัน ให้ ผู้ใช้งาน โดยเปิดให้ใช้งานฟรีสำหรับแอปพลิเคชันที่ใช้งานส่วนตัวทั่วไปสำหรับ การทดลองและการ พัฒนาโดยทางผู้วิจัยได้เลือกใช้งาน Unity Engine นั้นเพราะจากการศึกษาข้อมูลการใช้งานเบื้องต้น นั้น แพลตฟอร์ม Unity นั้นเหมาะสมสำหรับผู้วิจัยในการเริ่มพัฒนา แอปพลิเคชัน 3 มิติมากกว่า ทั้ง ในเรื่องของการมี Library สำหรับอุปกรณ์ VR ที่รองรับไว้แล้ว โดยมีพัฒนา และ อัปเดต เวอร์ชัน ของตัว Library ตลอดโดย Library ในการพัฒนาApplicationพลิเคชัน วีอาร์ นั้น อาจจะถูกผูกติด กับอุปกรณ์เฉพาะ วีอาร์ ที่เฉพาะเจาะจงจากทางทางผู้ผลิตเช่น Oculus Integration [32] - เป็น Library สำหรับใช้งานอุปกรณ์ วีอาร์ Meta Quest เท่านั้นโดยเฉพาะ จะไม่สามารถนำไปใช้ร่วมกับ อุปกรณ์วีอาร์จากผู้ผลิตอื่นได้ XR interaction toolkit [33] เป็น Library กลางที่ผู้พัฒนาสามารถใช้ พัฒนบนอุปกรณ์แพลตฟอร์มอื่นๆได้ หรือ แม้แต่ การสร้างแอปพลิเคชันเพื่อทดสอบ หรือ จำลอง โดยไม่ต้องมีอุปกรณ์ วีอาร์ เพื่อใช้ในการทำสอบจริงๆได้

โดยจากที่กล่าวมาข้างต้น นั้น เมื่อใช้ แพลตฟอร์ม Unity ร่วมกับ Library ในการ พัฒนา ที่มีโปรแกรมจะช่วยให้การพัฒนา โลก Digital VR ได้สมบูรณ์สอดคล้องกับอุปกรณ์ Meta Quest 2 ได้มีประสิทธิภาพมากยิ่งขึ้น

### 3.2.3 อุปกรณ์คัดแยกวัตถุในโลก Digital (VR)

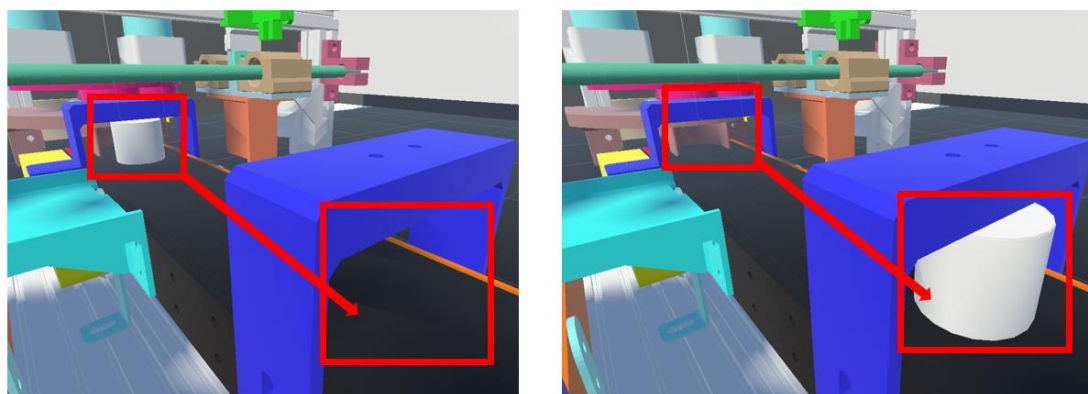
ทางผู้วิจัยได้ทำการปรับแต่ง อุปกรณ์ระบบคัดแยกวัตถุอัตโนมัติ 3 มิติ เดิมจากโปรแกรม Sketch Up [34] และนำเข้าสู่โปรแกรม Unity3D เพื่อใช้ในการสร้างแพลตฟอร์มดิจิทัลโดยทางผู้วิจัยได้ทำการแก้ไข อุปกรณ์บางส่วนบนตัวโมเดลให้มีความสอดคล้องกับ อุปกรณ์จำลองการคัดแยกวัตถุจริง และ เพิ่ม UI ( User Interface ) เพื่อให้ผู้ใช้งานสามารถสั่งการทำงานของอุปกรณ์ภายในโลก Digital (VR) ดังรูปที่ 35.



รูป 35 โมเดลตั้งต้นเดิมจาก Sketch Up และ โมเดล 3D หลังแก้ไขใน Unity 3D

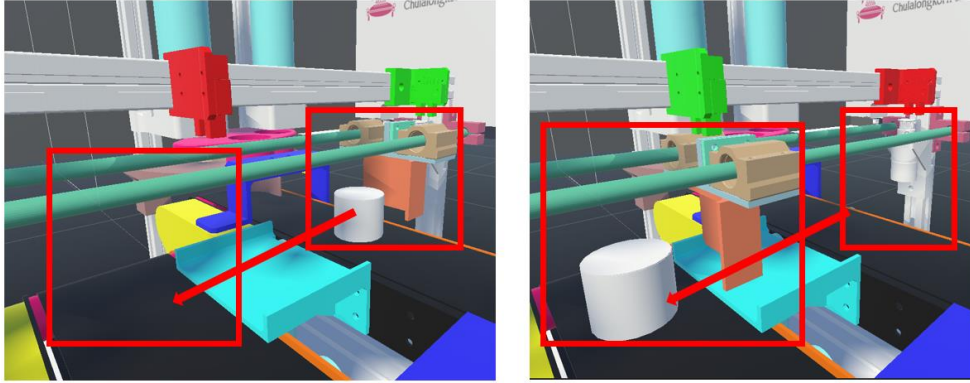
โดยอุปกรณ์คัดแยกวัตถุในโลก Digital ผู้วิจัยได้พัฒนาภาพเคลื่อนไหว สามมิติ บางส่วนบนอุปกรณ์คัดแยกวัตถุอัตโนมัติ ในโลก Digital (VR) ได้แก่

1.ภาพเคลื่อนไหวของวัตถุบนสายพาน โดยภาพเคลื่อนไหวจะแสดงวัตถุบนสายพานเคลื่อนที่จากจุดเริ่มต้นบริเวณใต้เซนเซอร์สี TCS34725 ไปยังจุดอื่นๆที่กำหนดดังรูปที่ 36.



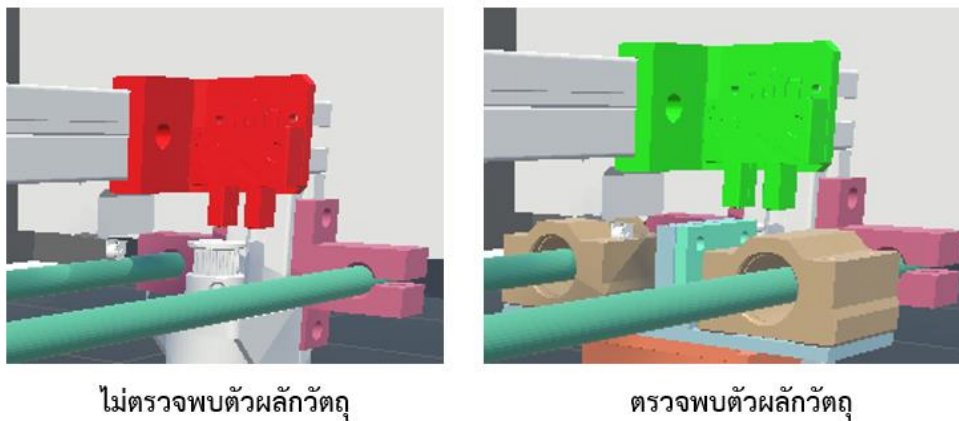
รูป 36 ตัวอย่างอนิเมชันการเคลื่อนไหววัตถุบนสายพาน

2.ภาพเคลื่อนไหวของชุดผลักวัตถุ โดยภาพเคลื่อนไหวจะแสดงอนิเมชันชุดผลักวัตถุเคลื่อนที่จากจุดที่กำหนดพร้อมด้วย วัตถุบนสายพานเช่นกัน ดังรูป 37.



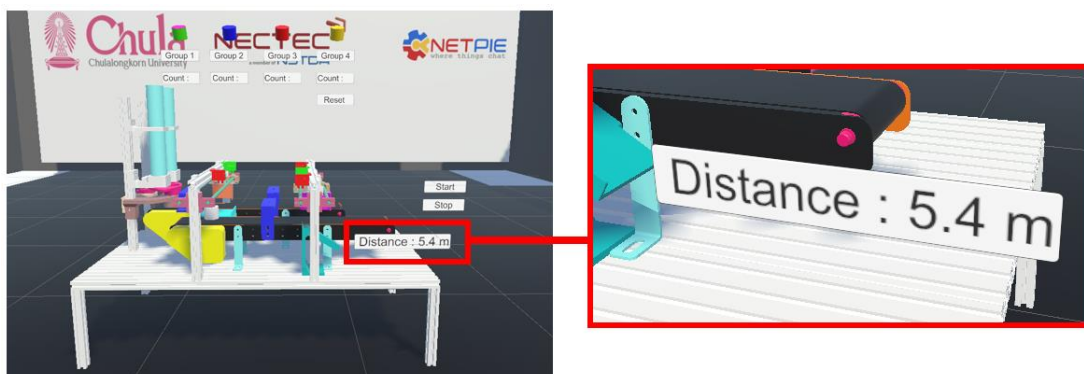
รูป 37 ตัวอย่างอนิเมชันการเคลื่อนไหวของชุดผลักวัตถุ

3.การเปลี่ยนสีแสดงสถานะของ Sensor Omron-EESX47 บนชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ โดยสีแดงจะแสดงสถานะเซนเซอร์ตรวจจับไม่พบชุดผลักวัตถุสีเขียวแสดงสถานะเซนเซอร์ตรวจจับตรวจพบชุดผลักวัตถุ (ซ้าย, สีแดง) ไม่ตรวจพบวัตถุ และ (ขวา , สีเขียว) ตรวจพบวัตถุดังรูปที่ 38.



รูป 38 ตัวอย่างอนิเมชันการเปลี่ยนสี

4. กล้องแสดงข้อมูลระยะทางของสายพาน ดังรูปที่ 39. โดยข้อมูลที่แสดงในกล้องนั้นสามารถอัปเดตค่าสถานะได้ในเวลาจริงเมื่อระบบเริ่มปฏิบัติการ.



รูป 39 กล้องแสดงระยะทางการเคลื่อนที่ของสายพาน

โดยเพื่อให้เห็นภาพรวมของการทำงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital(VR) ให้เห็นภาพมากยิ่งขึ้นทางผู้จัดทำจะใส่เรียงสาริตลำดับการทำงาน พร้อมทั้งอธิบายระบบการสื่อสารระหว่างชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลกดิจิทัล ในบทถัดไป

### 3.3 ระบบการสื่อสารของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ

ในการเลือกใช้กระบวนการเชื่อมต่อสื่อสารเพื่อสร้างการทำงานในลักษณะของคู่แฝดดิจิทัลนั้นทางผู้จัดทำได้พิจารณาเลือกการใช้งานของระบบสื่อสารไร้สายในเครือข่ายแบบ Wireless LAN บน UDP (User Datagram Protocol) และ TCP(Transmission Control Protocol) สำหรับการส่งงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัตินั้นจะทำการเชื่อมต่อข้อมูลเข้าสู่คู่แฝดดิจิทัล(VR)ด้วยการสื่อสารผ่านระบบ MQTT(Message Queuing Telemetry Transport) โดยมุ่งเน้นประสิทธิภาพในเรื่องของความหน่วง และ เสถียรภาพในการส่งข้อมูลเป็นหลัก

#### 3.3.1 UDP (User Datagram Protocol)

เป็นมาตรฐานการส่งข้อมูลอย่างเรียบง่ายโดยเป็นการรวบรวม Package จัดส่งได้รวดเร็ว แต่มีข้อเสียคือไม่มีการรับประกันว่าชุดข้อมูลที่จัดส่งไปนั้นถึงผู้รับได้ถูกต้องตามลำดับขั้นตอน โดยในกระบวนการส่งงานด้วย UDP ในเครือข่าย Wireless LAN นั้น ทางผู้จัดทำได้ใช้ในการรับ-ส่ง ชุดคำสั่งสำหรับเปิดการทำงานของชุดคอนโทรลเลอร์บนโมดูลต่างๆ รวมถึง ข้อมูลการทำงานของชุดเซนเซอร์และอุปกรณ์ที่ติดตั้งอยู่บนชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ โดยในการใช้งานนั้น ทางผู้จัดทำได้ใช้งานชุด Library สำหรับ บอร์ดคอนโทรลเลอร์ โดยอ้างอิง Library WiFiUdp.h [35] และ

ใช้งาน Node Function UDP ซึ่งเป็น Node Function พื้นฐานที่ติดตั้งมาให้สามารถใช้งานได้บนชุดประมวลผลหลัก Node-RED ด้วย ชุดคำสั่ง ทั้งสองชุดที่กล่าวมาผู้จัดทำสามารถสร้างกระบวนการรับส่งข้อมูลในชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลกแห่งความจริง ด้วยการสื่อสารแบบ UDP Wireless LAN ได้

### 3.3.2 TCP (Transmission Control Protocol)

เป็นมาตรฐานการส่งข้อมูลที่มีการควบคุมการรับส่งข้อมูลระหว่างผู้รับกับผู้ส่ง โดยในการส่งข้อมูลมีการตรวจสอบและยืนยันให้แน่ใจว่าข้อมูลนั้น ถึงผู้รับและเป็นไปตามลำดับขั้นตอนอย่างถูกต้อง โดยทางผู้จัดทำได้เลือกใช้การสื่อสารแบบ TCP Wireless LAN สำหรับการทดสอบเท่านั้นเพื่อนำมาเปรียบเทียบความหน่วงและเสถียรภาพ กับการใช้งานการสื่อสาร แบบ UDP สำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัติ โดยในกระบวนการใช้งานนั้น ทางผู้จัดทำได้ใช้งานชุด Library สำหรับบอร์ดคอนโทรลเลอร์ คือ Wi-Fi Library for Arduino [36] และใช้งาน Node TCP ซึ่งเป็น Node Function พื้นฐานที่ติดตั้งมาให้สามารถใช้งานได้บนชุดประมวลผลหลัก Node-RED เช่นเดียวกับชุดคำสั่ง UDP ที่กล่าวมาก่อนในข้างต้น

### 3.3.3 MQTT (Message Queue Telemetry Transport)

เป็นช่องทางการติดต่อสื่อสารระหว่าง M2M ( machine – to machine ) โดยเป็นส่วนหนึ่งในเทคโนโลยีของ IoT(Internet of Things ) ซึ่งเป็นเทคโนโลยีที่เชื่อมต่อการติดต่อสื่อสารกันระหว่างอุปกรณ์ต่างๆ รวมทั้งการส่งต่อข้อมูลผ่านแอปพลิเคชันข้ามแพลตฟอร์ม โดยผู้จัดทำได้ใช้ระบบการสื่อสารแบบ MQTT สำหรับการสื่อสารกันระหว่าง Application Unity บนอุปกรณ์คอมพิวเตอร์พกพา , อุปกรณ์แว่น VR และ Node-RED ผ่านผู้ให้บริการ MQTT Broker NETPIE 2020 โดย MQTT มีส่วนประกอบ 3 องค์ประกอบสำหรับการรับส่งข้อมูลคือ

ผู้ส่ง (Publisher) มีหน้าที่ในการส่งข้อความไปยังตัวกลาง โดยข้อความที่จัดส่งไปนั้นประกอบไปด้วย หัวข้อ (Topic) และ ข้อความ (Message)

2.ผู้รับ (Subscriber) มีหน้าที่ติดตามไปยัง หัวข้อ ที่ผู้รับสนใจ และรอรับ ข้อความจากตัวกลางที่ในการส่งข้อมูล

3.ตัวกลาง (MQTT broker) มีหน้าที่รับข้อความ จากผู้ส่ง แล้วทำการกระจาย (Broadcast)ข้อความไปยัง ผู้รับ ตามหัวข้อ ที่ได้มีการกำหนดไว้



โดยการสื่อสารด้วย MQTT protocol มีการรับประกันการส่งข้อมูล (QoS , Quality of Service ) อยู่ 3 ระดับคือ

1.QoS Level 0 คือ การรับประกันการส่งข้อมูลในระดับต่ำสุด เน้นข้อมูลที่ต้องการความรวดเร็วไม่ต้องการตอบกลับว่าข้อมูลนั้นถึง ตัวกลาง เรียบร้อยแล้ว

2.QoS Level 1 คือ การใช้ระบบการรับประกันการส่งข้อมูล โดยทุกครั้งของการส่งข้อมูลนั้นต้องมีการตอบกลับยืนยันว่าข้อมูลนั้นได้ส่งไปยัง ตัวกลาง แล้ว โดยถ้าหากไม่มีการตอบกลับทางผู้ส่งจะทำการส่งข้อมูลไปใหม่ จนกว่าจะได้รับการยืนยันการกลับมา โดย เป็นการรับประกันว่าตัวกลาง จะได้รับข้อมูลจาก ผู้ส่ง ข้อมูล อย่างน้อย 1 ครั้ง และ อาจจะมีมากกว่า เช่นเดียวกัน

3.QoS Level 2 คือ การใช้ระบบการรับประกันการส่งข้อมูลสูงสุด โดยทุกครั้งที่มีการส่งข้อมูลจะมีการยืนยันว่าถูกส่งไปยัง ตัวกลาง อย่างแน่นอน และ และ ตัวกลาง จะได้รับการส่งข้อมูลอย่างแน่นอน

#### 3.3.1.1 NETPIE 2020

คือแพลตฟอร์มที่พัฒนาขึ้นมาโดยศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (National Electronic and Computer Technology Center ) หรือ เนคเทค (NECTEC) เพื่อตอบสนองผู้ใช้งานในด้าน อุตสาหกรรม 4.0 ในประเทศไทย โดยคุณสมบัติความสามารถที่โดดเด่นหลักๆของแพลตฟอร์ม NETPIE 2020 นั้นคือ

1. แสดงค่าข้อมูลจากเซ็นเซอร์ หรือ อุปกรณ์ ได้แบบ Real-time (Monitoring)
2. สามารถควบคุมการทำงานของอุปกรณ์ต่างๆ ผ่านระบบCloud Platform (Controlling)
3. ดำเนินการจัดเก็บค่า และ ข้อมูลต่างๆ ของอุปกรณ์ (Data Storage) ในระบบ
4. กำหนดการแจ้งเตือนผิดปกติของอุปกรณ์ หรือ เซนเซอร์ ตามที่ระบบได้กำหนดไว้ (Notification)
- 5.แสดงผลและควบคุมการทำงานของอุปกรณ์ ผ่านระบบ Dashboard (Dashboard for Monitor & Control)

ซึ่งทาง ระบบ แพลตฟอร์ม NETPIE 2020 รูปที่ 21. นั้นมีการตั้งใจให้ผู้ใช้งานทั่วไปใช้งานได้ โดยไม่จำเป็นต้องเสียค่าใช้จ่าย (ในขอบเขตที่กำหนด )เพื่อเป็นอีกหนึ่งช่องทางในการเปิดโอกาสให้

ผู้ประกอบการ เจ้าของโรงงาน และ นักพัฒนา ได้มีช่องทางในการพัฒนา งานอุตสาหกรรม 4.0 ในประเทศไทย

### 3.3.4 การเชื่อมต่อข้อมูลระหว่างคู่ฝาแฝดดิจิทัล

โดยจากกระบวนการสื่อสารที่กล่าวมาในข้างต้นนั้นผู้จัดทำจึงสามารถทำการเชื่อมต่อประสานข้อมูลระหว่างชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Physical และ คู่ฝาแฝดดิจิทัลได้ โดยขออธิบายการเชื่อมต่อชุดข้อมูลของคู่ฝาแฝดดิจิทัลออกเป็นสามหมวดหมู่ได้แก่

#### 3.3.4.1 การเชื่อมต่อ ESP8266 กับ Node Red

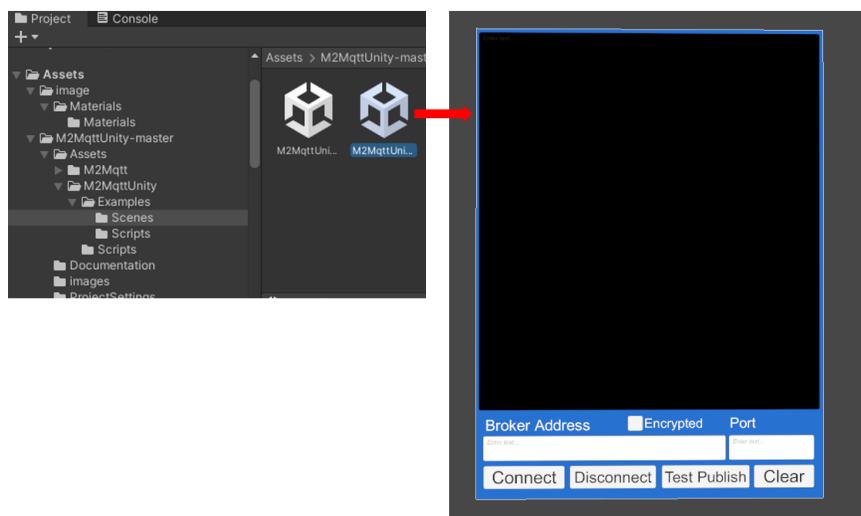
โดยในกระบวนการเชื่อมต่อชุดบอร์ดคอนโทรลเลอร์ และ ชุดประมวลผลหลัก Node-RED นั้น ทางผู้จัดทำได้ทำการใช้ Library ฟังก์ชัน WiFiUdp.h สำหรับการส่งข้อมูลแบบ UDP และ Wi-Fi Library สำหรับการรับส่งข้อมูลแบบ TCP โดยมีกำหนด IP. Address ของชุดบอร์ดคอนโทรลเลอร์ให้ชัดเจน และ ทำการเลือก Port สำหรับการสื่อสารของชุดอุปกรณ์ โดยการใช้ Node UDP IN และ Node TCP IN บน Node-RED นั้นจะทำให้ Node-RED สามารถรับข้อมูลที่จัดส่งมาจากชุดบอร์ดคอนโทรลเลอร์ได้

#### 3.3.4.2 การเชื่อมต่อ Node-RED กับ NETPIE 2020

โดยในกระบวนการเชื่อมต่อชุดประมวลผลหลัก และ Platform NETPIE 2020 นั้นทางผู้จัดทำสามารถเลือกใช้ Node MQTT บนชุดประมวลผลหลัก Node-RED ได้ โดยผู้ใช้จะต้องตั้งค่าตามคู่มือการใช้งานของ NETPIE 2020 [35] โดยจุดที่ต้องให้ความสำคัญคือ Client ID โดย Client ID นั้น 1 ชุดอุปกรณ์จะสามารถเชื่อมต่อกับ Device บนแพลตฟอร์มของ NETPIE 2020 ได้ทีละ 1 ชุดอุปกรณ์เท่านั้น หากต้องการทำการเชื่อมต่อหลายชุดอุปกรณ์บนแพลตฟอร์มของ NETPIE 2020 จำเป็นจะต้องสร้าง Device บนแพลตฟอร์มของ NETPIE 2020 ให้ตรงกับชุดอุปกรณ์จริง โดยในที่นี้คือชุดประมวลผลหลัก Node-RED

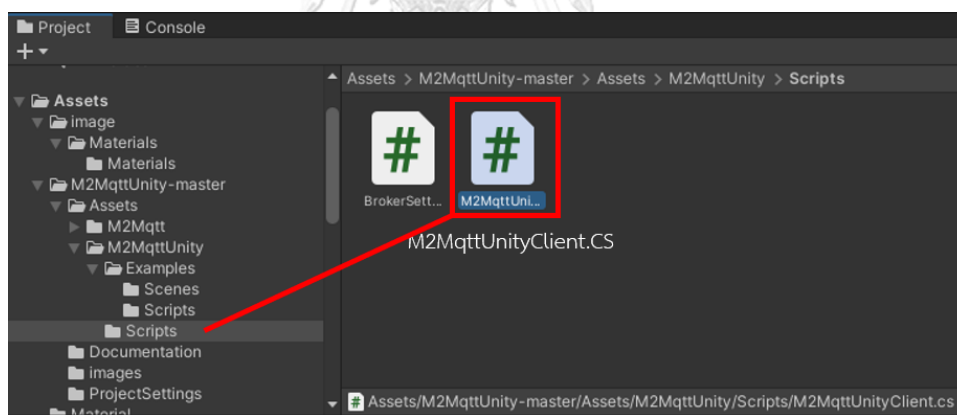
#### 3.3.4.3 การเชื่อมต่อ NETPIE 2020 กับ คู่ฝาแฝดดิจิทัลผ่านโปรแกรม Unity

โดยในกระบวนการเชื่อมต่อ NETPIE 2020 กับคู่ฝาแฝดดิจิทัลผ่านโปรแกรม Unity นั้น ผู้จัดทำได้เลือกใช้งาน Library การสื่อสารแบบ MQTT สำหรับโปรแกรม Unity M2MQTT for Unity [36] ซึ่งสามารถนำไปใช้รองรับกับโปรแกรม Unity เวอร์ชัน 2017.1 หรือสูงกว่าได้โดยในการนำ Library มาใช้งานนั้น ผู้ใช้งานสามารถดึงหน้า UI สำหรับทดสอบการเชื่อมต่อข้อมูลแบบ MQTT ใน Assets -> M2MqttUnity->Examples -> Scenes ดังรูปที่ 40.



รูป 40 ตำแหน่ง Scene ตัวอย่างของ M2MqttUnity

และผู้ใช้งานจะต้องตั้งค่าการเชื่อมต่อกับ NETPIE 2020 ผ่านการแก้ไข Script การ Setting การเชื่อมต่อด้วย MQTT ใน Scripts file ใน Folder ดังรูปที่ 41.



รูป 41 ตำแหน่ง Scripts สำหรับการตั้งค่าการเชื่อมต่อ Client MQTT Unity

โดย Default Library ใน Script การเชื่อมต่อของ m2mqttUnityClient นั้นไม่ได้มีการกำหนดค่า ClientID ให้สอดคล้องกับ ClientID บนแพลตฟอร์มของ NETPIE มาให้ จึงทำให้ผู้ใช้งานไม่สามารถเชื่อมต่อ การสื่อสารแบบ MQTT จาก Game Inspector ใน Unity ได้โดยตรง โดยผู้ใช้งานจะต้องแก้ไขการสร้าง Variable “clientId” ใน Script M2MqttUnityClient.cs บริเวณบรรทัดที่ 305 โดยการแก้ไขการสร้าง Variable “clientId” อัตโนมัติ เป็นการ Fix “clientId” ให้สอดคล้องแพลตฟอร์ม NETPIE 2020 ดังรูปที่ 42.

```

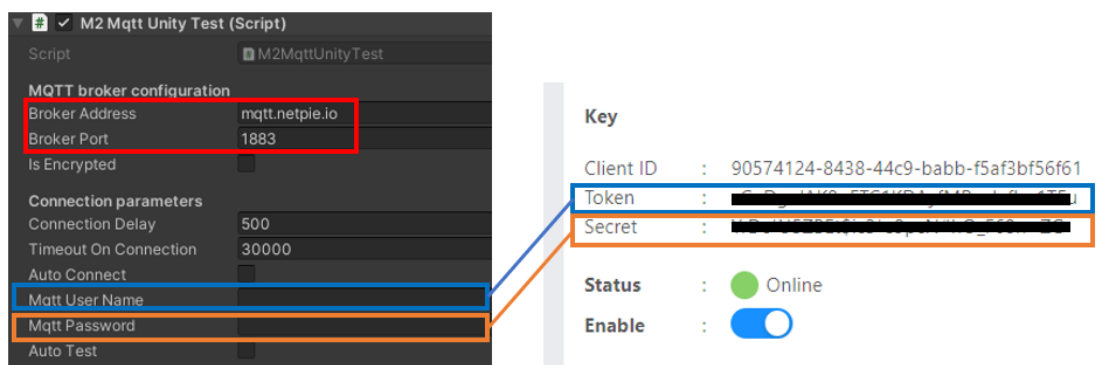
client.Settings.TimeoutOnConnection = timeoutOnConnection;
string clientId = Guid.NewGuid().ToString();
//string clientId = "27bdf73b-6ae7-4bfd-8a3b-1003e76c3ffb";

//string clientId = Guid.NewGuid().ToString();
string clientId = "27bdf73b-6ae7-4bfd-8a3b-1003e76c3ffb";

```

รูป 42 การแก้ไข clientId ใน M2MqttUnityClient.cs

โดยการตั้งค่าในส่วนอื่นๆของการเชื่อมต่อนั้น จะอยู่ใน Script M2MqttUnityTest.cs สำหรับตัวอย่างการใช้งานนั้นโดยผู้ใช้งานสามารถแก้ไข ข้อมูลจาก Game Inspector ได้เลย หรือ จะทำการแก้ไข Variable ใน Script ก็ได้เช่นกัน โดยนำชุดข้อมูล Token ( Mqtt User name ) และ Secret ( Mqtt Password ) บนระบบ NETPIE 2020 มาใส่ใน Game Inspector ตามรูปที่ 43. หรือ แก้ไขใน Script “M2MqttUnityClient.cs” ได้เช่นกัน



รูป 43 การใส่ข้อมูล Broker Address , Port , Token (User Name) และ Secret (Password)

โดยเมื่อแก้ไขการเชื่อมต่อได้แล้ว ผู้ใช้งานสามารถสร้าง class object สำหรับรับค่าตัวแปร โดยลำดับ variable class ชุดข้อมูลที่รับส่งนั้นควรให้สอดคล้องกับชุดข้อมูลในการรับส่ง โดยข้อมูลที่รับส่งจาก NETPIE 2020 นั้นจะประกอบไปด้วย 2 ชุด ข้อมูลใหญ่ๆ คือ 1.deviceid , 2.data โดยผู้ใช้งานควรตั้งค่า data ให้สอดคล้องกับข้อมูลที่ต้องการรับส่ง ดังตัวอย่างตามรูปที่ 44. และสามารถตั้งค่าการ Decode message จาก MQTT broker และ การส่งข้อมูลกลับ โดยจำเป็นต้องตั้งค่าการ Subscribe และ Unsubscribe หัวข้อที่ต้องการด้วย ดังรูปที่ 45.

```

6 [Serializable]
7 public class MyObject
8 {
9     public string deviceid;
10    public data data;
11 }
12 [Serializable]
13 public class data
14 {
15     public Actuator Actuator;
16     public Sensor Sensor;
17     public Animation Animation;
18 }
19
20 [Serializable]
21 public class Sensor
22 {
23     public bool S_Sensor_1;
24     public bool S_Sensor_2;
25     public bool S_Sensor_3;
26     public bool S_Sensor_4;
27     public bool S_Sensor_5;
28     public int C_Distance;
29     public int C_Red;
30     public int C_Yellow;
31     public int C_Green;
32     public int C_Blue;
33 }

```

รูป 44 ตัวอย่างการสร้าง Variables สำหรับเชื่อมต่อข้อมูล NETPIE 2020

Decode Message From MQTT Broker

```

public MyObject myObject;
2 references
protected override void DecodeMessage(string topic, byte[] message)
{
    string msg = System.Text.Encoding.UTF8.GetString(message);
    StoreMessage(msg);

    myObject = JsonUtility.FromJson<MyObject>(msg);
}

```

Public message to MQTT Broker

```

public void TestPublish()
{
    string json = JsonUtility.ToJson(myObject);
    client.Publish("@msg/com", System.Text.Encoding.UTF8.GetBytes(json), MqttMsgBase.QOS_LEVEL_AT_MOST_ONCE, false);
    Debug.Log(json);
}

```

Subscribe / Unsubscribe to topics setting

```

protected override void SubscribeTopics()
{
    client.Subscribe(new string[] { "@msg/com" }, new byte[] { MqttMsgBase.QOS_LEVEL_AT_MOST_ONCE });
}
2 references
protected override void UnsubscribeTopics()
{
    client.Unsubscribe(new string[] { "@msg/com" });
}

```

รูป 45 ตัวอย่างการตั้งค่า Decode Message , Public Message และ Subscribe to Topic

โดยเมื่อทำการตั้งค่าการเชื่อมต่อและตัวแปรเรียบร้อยแล้วนั้น ผู้ใช้งานสามารถทดสอบการทำงานของ M2MqttUnity ได้โดยการสั่งชุดคำสั่ง Connected () จาก Script ไฟล์ M2MqttUnityTest.cs เพื่อทดสอบการเชื่อมต่อแอปพลิเคชันไปยัง MQTT Broker โดยการทำงานของ Library MQTT นั้นเมื่อ MQTT Broker ได้มีการทำการจัดส่งข้อมูลมา ตัว Library จะทำการ Decode Message และ ทำการอัปเดต ค่าของ Variable ที่ได้รับมาเพื่อนำไปพัฒนาระบบต่อไป

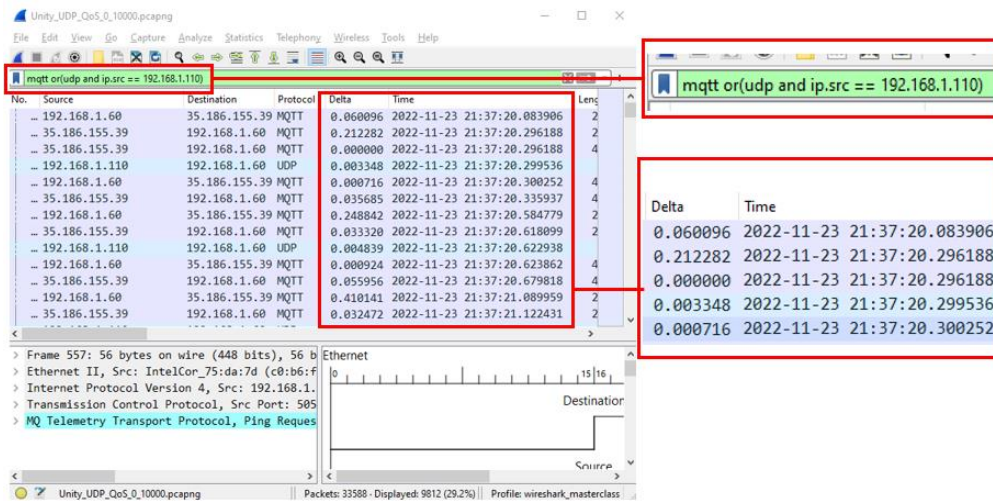
### 3.4 วิธีการทดสอบความหน่วงและเสถียรภาพของระบบการสื่อสาร

ในขั้นตอนทดสอบความหน่วงและเสถียรภาพของระบบสื่อสารที่ทางผู้จัดทำได้เลือกใช้ในการพัฒนาระบบผ่านแพลตฟอร์มสำหรับอุปกรณ์คัตแยกวัตถุอัตโนมัติ นั้นเพื่อให้เกิดความแม่นยำในการวัดความหน่วง และการเก็บข้อมูลการสื่อสารระหว่างแพลตฟอร์มให้ถูกต้องที่สุด ทางผู้จัดทำจึงได้ทำการเลือกใช้ Software Wireshark [37] ซึ่งเป็น Software สำหรับการวิเคราะห์ข้อมูลในระบบ Network โดยสามารถ ตรวจสอบและดักจับข้อมูลได้หลากหลาย protocol ซึ่งทางผู้จัดทำได้นำมาใช้ในการตรวจสอบข้อมูลผ่านระบบ UDP , TCP และ MQTT เพื่อวิเคราะห์ความหน่วง และเสถียรภาพ ในการทำงานของระบบการสื่อสารที่ผู้จัดทำได้ทำการพัฒนาขึ้นบนอุปกรณ์คัตแยกวัตถุอัตโนมัติ ทั้งในโลกแห่งความจริง และ ในโลก Digital (VR)

#### 3.4.1 ซอฟต์แวร์ Wireshark

โดยในวิทยานิพนธ์ฉบับนี้ทางผู้จัดทำได้ใช้ ซอฟต์แวร์ Wireshark version 4.0.0 ติดตั้งเพื่อตรวจจับข้อมูลบนอุปกรณ์คอมพิวเตอร์พกพา Asus TUF gaming fx504 gm ระบบประมวลผล Intel Core i7-8750H RAM 16.0 GB บนระบบปฏิบัติการ Windows 10 Pro 64-bit โดยทำการตรวจจับข้อมูลบนระบบผ่านระบบสื่อสารไร้สายบนตัวอุปกรณ์คอมพิวเตอร์พกพา Intel® Wireless-AC 9560 160MHZ บนความถี่ Wi-Fi 2.4Ghz ดังรูปที่ โดยระยะห่างจาก Router Skyworth\_ GN542VF ในรัศมี 10 เมตร บนเครือข่ายอินเทอร์เน็ต True Fiber 300/300Mbps.

โดยในขั้นกระบวนการใช้งานนั้นทางผู้จัดทำใช้กระบวนการกรอง Filter เพื่อเก็บข้อมูลข้อมูลเฉพาะในระบบ protocol ที่ต้องการนำมาวิเคราะห์คือ UDP, TCP และ MQTT โดยทำการเก็บข้อมูลบันทึกเวลา เพื่อนำมาหาค่าความหน่วงระหว่างการรับส่งชุดข้อมูล ดังรูปที่ 46.เป็นตัวอย่างชุดข้อมูลที่ สามารถตรวจสอบผ่านโปรแกรมWiresharkได้



รูป 46 ตัวอย่างการใช้งานซอฟต์แวร์ Wireshark

### 3.4.2 กระบวนการทดสอบ ความหน่วง และ เสถียรภาพ ของระบบสื่อสาร

ในกระบวนการทดสอบ ความหน่วง และ เสถียรภาพของระบบสื่อสาร นั้นทางผู้จัดทำได้ออกแบบการทดสอบโดยออกแบบการรับส่งข้อมูลที่มีขนาดแตกต่างกันโดยแบ่งการทดสอบออกเป็น 2 หมวดหมู่ได้แก่

#### 3.4.2.1 การรับส่งข้อมูล Node-RED และ ESP8266 ผ่าน UDP/TCP

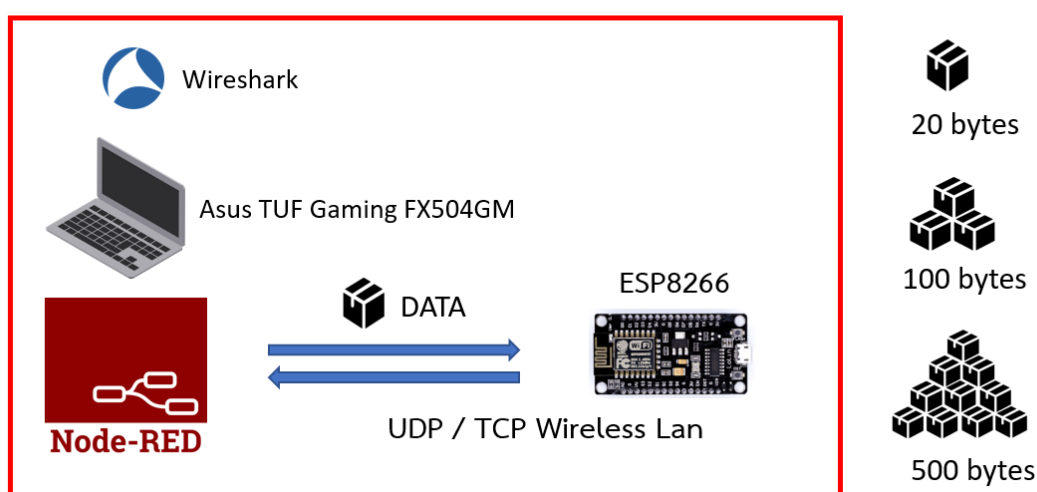
โดยในกระบวนการทดสอบนั้นจะเป็นการจำลองการรับส่งข้อมูลเพื่อเปรียบเทียบความหน่วงและเสถียรภาพ ระหว่างการสื่อสารผ่าน Wireless LAN ผ่านระบบ UDP และ protocol TCP โดยจะทำการวัดความหน่วงในการรับส่งข้อมูลระหว่างชุดประมวลผลหลัก Node-RED ที่ติดตั้งอยู่บนคอมพิวเตอร์พกพา และ ชุดบอร์ดคอนโทรลเลอร์ ESP8266 สำหรับส่งงานอุปกรณ์ต่างๆบนชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ โดยชุดอุปกรณ์ทั้งหมดเชื่อมต่อผ่านระบบ Wireless LAN 2.4 GHz ในรัศมี 10 เมตรจาก Router โดยการทดสอบนั้นจะทำการแบ่งชุดข้อมูลในการรับส่งออกเป็น 3 ชุดข้อมูล ได้แก่

- 1.ชุดข้อมูลคำสั่งขนาดเล็ก มีขนาดชุดข้อมูลทั้งหมดประมาณ 20 byte
- 2.ชุดข้อมูลคำสั่งขนาดกลาง มีขนาดชุดข้อมูลทั้งหมดประมาณ 100 byte
- 3.ชุดข้อมูลคำสั่งขนาดใหญ่ มีขนาดชุดข้อมูลทั้งหมดประมาณ 500 byte

โดยจำนวนครั้งในการทดสอบรับส่งข้อมูล คือ จัดส่งข้อมูลทั้งหมด 10,000 ครั้ง โดยใช้เวลา 500 ms ระหว่างชุดข้อมูล โดยในการทดลองนั้นจะตรวจสอบความหน่วงเวลาในการรับส่งข้อมูล ไป-กลับ (Round Trip Time) และ จำนวนข้อมูลที่มีการรับส่งสำเร็จจากชุดประมวลผลหลัก Node-RED และ ESP8266 ของทั้งการสื่อสารแบบ UDP และ TCP ดังรูปที่ 47.



WIFI 2.4Ghz True Fiber 300/300 Mbps  
รัศมี 10 เมตร



รูป 47 ภาพจำลองการทดสอบการรับส่งข้อมูลผ่านระบบ UDP และ TCP

โดยในกระบวนการเก็บข้อมูลนั้น ทางผู้จัดทำได้ใช้โปรแกรม Wireshark ในการตรวจสอบ ชุดข้อมูลที่ทำการรับส่งข้อมูลไปจากชุดอุปกรณ์คอมพิวเตอร์พกพา (Node-RED) โดยโปรแกรม Wireshark นั้นจะสามารถบันทึกเวลา ณ ตอนที่มีการรับส่งข้อมูลเอาไว้ได้พร้อมที่ยังแจจรายละเอียดจำนวน bytes ของ Data ที่มีการจัดส่ง โดยในการนำข้อมูลมาบันทึกผลนั้นใน 1 ชุดคำสั่งจะเป็นการส่ง ชุดข้อมูลออกไปจาก ชุดประมวลผลหลัก Node-RED (192.168.1.60) ไปยัง บอร์ดคอนโทรลเลอร์ ESP8266 สำหรับทดสอบ (192.168.1.110) และ รอรับชุดข้อมูล ตอบกลับมาจาก บอร์ดคอนโทรลเลอร์ ESP8266 (192.168.1.110) กลับมายังชุดประมวลผลหลัก Node-RED (192.168.1.60) โดยจะนำค่า Delta หรือ ค่าส่วนต่างของช่วงเวลา มาบันทึกผลในการนำเสนอ ค่าความหน่วงแบบ Round Trip Time ของการรับส่งข้อมูล โดยสามารถชมตัวอย่างในโปรแกรม Wireshark ในตามรูปที่ 48. และ 49.



No.	Time	Source	Destination	Delta	Protocol	Length	Info
3...	2022...	192.168.1.60	192.168.1.110	0.170008	UDP	542	58202 → 4210 Len=500
3...	2022...	192.168.1.110	192.168.1.60	0.125154	UDP	542	4210 → 4210 Len=500
3...	2022...	192.168.1.60	192.168.1.110	0.077857	UDP	542	58202 → 4210 Len=500
3...	2022...	192.168.1.110	192.168.1.60	0.075342	UDP	542	4210 → 4210 Len=500
3...	2022...	192.168.1.60	192.168.1.110	0.176387	UDP	542	58202 → 4210 Len=500
3...	2022...	192.168.1.110	192.168.1.60	0.077844	UDP	542	4210 → 4210 Len=500
3...	2022...	192.168.1.60	192.168.1.110	0.181925	UDP	542	58202 → 4210 Len=500
3...	2022...	192.168.1.110	192.168.1.60	0.021441	UDP	542	4210 → 4210 Len=500
3...	2022...	192.168.1.60	192.168.1.110	0.181422	UDP	542	58202 → 4210 Len=500
3...	2022...	192.168.1.110	192.168.1.60	0.019575	UDP	542	4210 → 4210 Len=500
3...	2022...	192.168.1.60	192.168.1.110	0.182855	UDP	542	58202 → 4210 Len=500
3...	2022...	192.168.1.110	192.168.1.60	0.050843	UDP	542	4210 → 4210 Len=500
3...	2022...	192.168.1.60	192.168.1.110	0.165358	UDP	542	58202 → 4210 Len=500
3...	2022...	192.168.1.110	192.168.1.60	0.022877	UDP	542	4210 → 4210 Len=500

Node-RED (Red square)  
ESP8266 (Blue square)

Delta time หลังจากส่งข้อความ

รูป 48 ตัวอย่างการวัดค่าความหน่วงของ UDP 500 bytes

Source	Destination	Delta	Protocol	Length	Info
192.168.1.60	192.168.1.110	0.000000	TCP	554	1025 → 61192 [PSH, ACK] Seq=1 Ack=1 Win=64888 Len=500
192.168.1.110	192.168.1.60	0.048065	TCP	554	61192 → 1025 [PSH, ACK] Seq=1 Ack=501 Win=1144 Len=500
192.168.1.60	192.168.1.110	0.466481	TCP	554	1025 → 61192 [PSH, ACK] Seq=501 Ack=503 Win=65390 Len=500
192.168.1.110	192.168.1.60	0.046714	TCP	554	61192 → 1025 [PSH, ACK] Seq=503 Ack=1001 Win=1644 Len=500
192.168.1.60	192.168.1.110	0.467415	TCP	554	1025 → 61192 [PSH, ACK] Seq=1001 Ack=1005 Win=64888 Len=500
192.168.1.110	192.168.1.60	0.043314	TCP	554	61192 → 1025 [PSH, ACK] Seq=1005 Ack=1501 Win=1144 Len=500
192.168.1.60	192.168.1.110	0.471886	TCP	554	1025 → 61192 [PSH, ACK] Seq=1501 Ack=1507 Win=65390 Len=500
192.168.1.110	192.168.1.60	0.092020	TCP	554	61192 → 1025 [PSH, ACK] Seq=1507 Ack=2001 Win=1644 Len=500
192.168.1.60	192.168.1.110	0.421008	TCP	554	1025 → 61192 [PSH, ACK] Seq=2001 Ack=2009 Win=64888 Len=500
192.168.1.110	192.168.1.60	0.039348	TCP	554	61192 → 1025 [PSH, ACK] Seq=2009 Ack=2501 Win=1144 Len=500
192.168.1.60	192.168.1.110	0.471867	TCP	554	1025 → 61192 [PSH, ACK] Seq=2501 Ack=2511 Win=65390 Len=500

Node-RED (Red square)  
ESP8266 (Blue square)

Delta time หลังจากส่งข้อความ

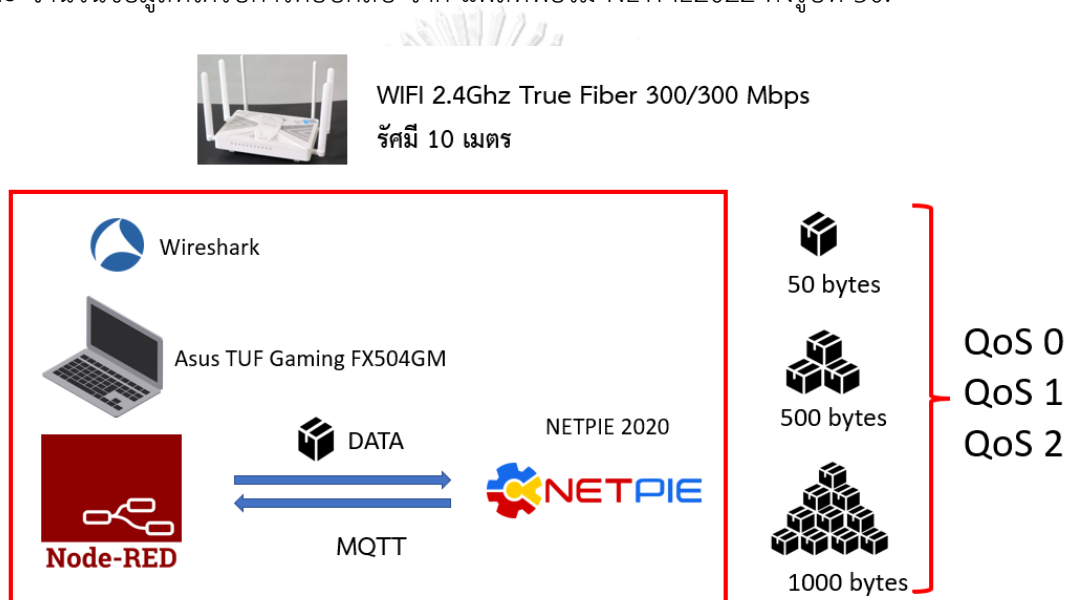
รูป 49 ตัวอย่างการวัดค่าความหน่วงของ TCP 500 bytes

### 3.4.2.2 การรับส่งข้อมูล Node-RED และ NETPIE 2020

โดยในกระบวนการทดสอบสำหรับ protocol MQTT นั้น จะเป็นการจำลองการรับส่งข้อมูลเพื่อเปรียบเทียบความหน่วงและเสถียรภาพของระบบสื่อสารของ protocol MQTT บนแพลตฟอร์มของ NETPIE 2020 โดยจะทำการวัดความหน่วงในการรับส่งข้อมูลระหว่างชุดประมวลผลหลัก Node-RED ที่ติดตั้งอยู่บนคอมพิวเตอร์พกพา ที่เชื่อมต่อผ่านระบบ Wireless LAN 2.4Ghz ในรัศมี 10 เมตรจาก Router กับ แพลตฟอร์ม NETPIE 2020 ผ่านเครือข่าย True Fiber 300/300 Mbps โดยการทดสอบนั้นจะทำการแบ่งชุดข้อมูลในการรับส่งออกเป็น 3 ชุดข้อมูลในลักษณะของ JSON ( JavaScript Object notation ) [38] เพื่อใช้ในการรับส่งข้อมูลสำหรับ protocol MQTT ได้แก่

- 1.ชุดข้อมูลคำสั่งขนาดเล็ก มีขนาดชุดข้อมูลทั้งหมดประมาณ 50 byte
- 2.ชุดข้อมูลคำสั่งขนาดกลาง มีขนาดชุดข้อมูลทั้งหมดประมาณ 500 byte
- 3.ชุดข้อมูลคำสั่งขนาดใหญ่ มีขนาดชุดข้อมูลทั้งหมดประมาณ 1,000 byte

โดยจำนวนครั้งในการทดสอบรับส่งข้อมูล คือ จัดส่งข้อมูลทั้งหมด 10,000 ครั้ง โดยใช้เวลา 500 ms ระหว่างชุดข้อมูล และ ทำการทดสอบในทุกการรับประกันการส่งข้อมูล QoS 0 , QoS 1 และ QoS 2 โดยในการทดลองนั้นจะตรวจสอบความหน่วงเวลาในการรับส่งข้อมูล ไป-กลับ (Round Trip Time) และ จำนวนข้อมูลที่ได้รับการตอบกลับ จาก แพลตฟอร์ม NETPIE2022 ดังรูปที่ 50.



รูป 50 ภาพจำลองการทดสอบการรับส่งข้อมูล Protocol MQTT

โดยในกระบวนการเก็บข้อมูลผ่านโปรแกรม Wireshark นั้น ทางผู้จัดทำได้ทำการแบ่งการส่งข้อมูลในชุดข้อมูล และ QoS ที่แตกต่างกัน โดยเป็นการตั้งค่า QoS ทั้ง publish และ subscribe ในระดับเดียวกันเมื่อเชื่อมต่อไปยัง NETPIE 2020 และมีการรับส่งข้อมูลแยก Topic สำหรับ ที่ใช้รับส่งแยกออกตามชุดข้อมูลที่ทดลอง และ QoS เพื่อให้ง่ายในการนำชุดข้อมูลที่ดักจับได้บนชุดอุปกรณ์คอมพิวเตอร์มาตรวจสอบเพื่อหาค่าความหน่วงของแต่ละชุดข้อมูล โดยสามารถชมตัวอย่างการแยกชุดข้อมูล และ QoS ได้ตามรูปที่ 51.

No.	Time	Source	Destination	Delta	Protocol	Length	Info
3..	202..	35.186.155.39	192.168.1.60	0.037545	MQTT	130	Publish Message [@msg/com/D1]
3..	202..	192.168.1.60	35.186.155.39	0.000363	MQTT	530	Publish Message [@shadow/update/D1]
3..	202..	192.168.1.60	35.186.155.39	0.472398	MQTT	524	Publish Message [@msg/com/D2]
3..	202..	35.186.155.39	192.168.1.60	0.037999	MQTT	524	Publish Message [@msg/com/D2]
3..	202..	192.168.1.60	35.186.155.39	0.000746	MQTT	530	Publish Message [@shadow/update/D2]
3..	202..	192.168.1.60	35.186.155.39	0.471544	MQTT	985	Publish Message [@msg/com/D3]
3..	202..	35.186.155.39	192.168.1.60	0.038751	MQTT	985	Publish Message [@msg/com/D3]
3..	202..	192.168.1.60	35.186.155.39	0.000302	MQTT	530	Publish Message [@shadow/update/D3]
3..	202..	192.168.1.60	35.186.155.39	0.470788	MQTT	132	Publish Message (id=51609) [@msg/com/D4]
3..	202..	35.186.155.39	192.168.1.60	0.035617	MQTT	58	Publish Ack (id=51609)
3..	202..	35.186.155.39	192.168.1.60	0.001845	MQTT	132	Publish Message (id=871) [@msg/com/D4]
3..	202..	192.168.1.60	35.186.155.39	0.000271	MQTT	58	Publish Ack (id=871)
3..	202..	192.168.1.60	35.186.155.39	0.000156	MQTT	532	Publish Message (id=51610) [@shadow/update/D4]
3..	202..	35.186.155.39	192.168.1.60	0.035963	MQTT	58	Publish Ack (id=51610)
2..	202..	192.168.1.60	35.186.155.39	0.442832	MQTT	132	Publish Message (id=49887) [@msg/com/D7]
2..	202..	35.186.155.39	192.168.1.60	0.034693	MQTT	58	Publish Received (id=49887)
2..	202..	192.168.1.60	35.186.155.39	0.000250	MQTT	58	Publish Release (id=49887)
2..	202..	35.186.155.39	192.168.1.60	0.005102	MQTT	132	Publish Message (id=10) [@msg/com/D7]
2..	202..	35.186.155.39	192.168.1.60	0.024693	MQTT	58	Publish Complete (id=49887)
2..	202..	192.168.1.60	35.186.155.39	0.000030	MQTT	58	Publish Received (id=10)
2..	202..	35.186.155.39	192.168.1.60	0.029712	MQTT	58	Publish Release (id=10)
2..	202..	192.168.1.60	35.186.155.39	0.000244	MQTT	58	Publish Complete (id=10)
2..	202..	192.168.1.60	35.186.155.39	0.000252	MQTT	532	Publish Message (id=49888) [@shadow/update/D7]
2..	202..	35.186.155.39	192.168.1.60	0.032062	MQTT	58	Publish Received (id=49888)
2..	202..	192.168.1.60	35.186.155.39	0.000190	MQTT	58	Publish Release (id=49888)

รูป 51 ตัวอย่างการวัดค่าความหน่วงของระบบ MQTT , QoS 0 , QoS 1 , QoS 2

โดยจากการทดลองเบื้องต้น ในการรับส่งชุดข้อมูล ทั้งหมด 10,000 ครั้ง 3 ชุด ข้อมูล และ 3 รับประกันการส่งข้อมูล รวมเป็นจำนวน 90,000 คำสั่งการรับส่งข้อมูล เนื่องจากการทดสอบครั้งนี้มีการรับส่งข้อมูลจำนวนมาก และ สำหรับการรับส่งข้อมูลผ่านระบบ MQTT นั้นต้องดำเนินการผ่านทางระบบอินเทอร์เน็ต เพื่อลดทอนความแตกต่างของข้อมูลในช่วงเวลาที่แตกต่างกันของการทำงานบนระบบอินเทอร์เน็ต ทางผู้จัดทำจึงได้ทำการออกแบบการทดสอบโดยแบ่งเป็นชุดคำสั่งใหญ่ 1 ชุดคำสั่งประกอบไปด้วยการจัดส่งชุดข้อมูล ทั้งหมด 9 ชุดข้อมูลย่อย ( 50bytes ที่ QoS 0-2 , 100bytes ที่ QoS 0-2 , และ 1000 bytes ที่ QoS 0-2 ) โดยจะไล่เรียงการจัดส่งข้อมูลในชุดข้อมูลย่อยให้เรียบร้อยก่อนในเวลาใกล้เคียงกัน และ วนซ้ำ จนครบ จะถือว่า 1 ชุดคำสั่งใหญ่ดำเนินการสำเร็จ โดยทำการทดสอบทั้งหมด 10,000 ชุดคำสั่งใหญ่ โดยการตั้งค่า Default ของ Node-RED ในกระบวนการสื่อสารแบบ MQTT นั้น ไม่ได้ตั้งค่าให้การเชื่อมต่อระหว่าง Node-RED และ NETPIE 2020 เป็นแบบ Persistent Connection [39] ซึ่งอาจจะมีโอกาสทำให้ข้อมูลการรับส่งสูญหายระหว่างกระบวนการทดสอบ ในทุกการรับประกันการส่งข้อมูล

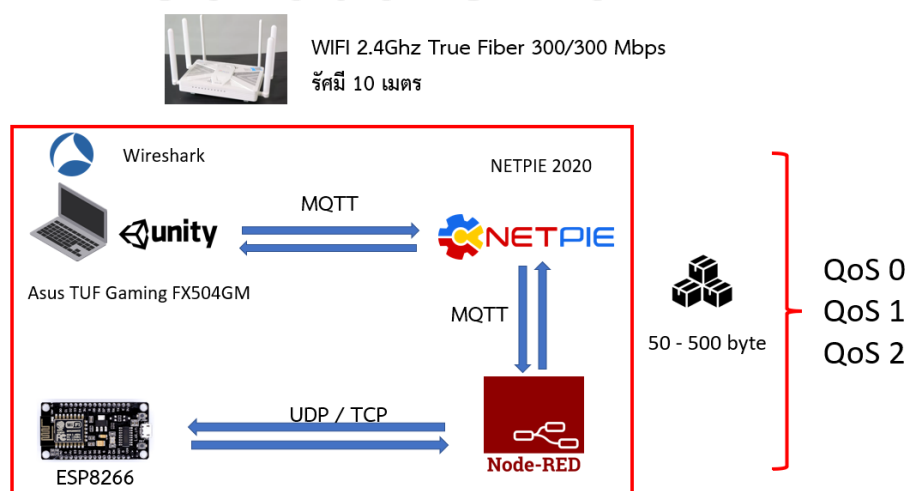
### 3.4.2.3 การรับส่งข้อมูลระหว่างคู่ฝาแฝดดิจิทัล

กระบวนการทดสอบสำหรับการสื่อสารระหว่างคู่ฝาแฝดดิจิทัล ( ชุดอุปกรณ์ คัดแยกวัตถุอัตโนมัติ ในโลก Digital และ ชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลกแห่งความจริง ) นั้น ผู้จัดทำได้สร้างการทดลองในการส่งข้อมูลจาก โปรแกรม Unity ที่ใช้สำหรับ Simulate ชุดอุปกรณ์ คัดแยกวัตถุอัตโนมัติในโลก Digital โดยเชื่อมต่อการทำงานผ่านระบบ Wireless LAN 2.4 GHz ใน รัศมี 10 เมตรจาก Router โดยแบ่งการทดสอบออกเป็น 2 หมวดหมู่ใหญ่ๆได้แก่

1.เชื่อมต่อ Unity และ Node-RED กับ NETPIE 2020 ด้วยการสื่อสารแบบ MQTT และ เชื่อมต่อการสื่อสารระหว่าง Node-RED กับ ESP8266 ด้วยการสื่อสารแบบ UDP

2.เชื่อมต่อ Unity และ Node-RED กับ NETPIE 2020 ด้วยการสื่อสารแบบ MQTT และ เชื่อมต่อการสื่อสารระหว่าง Node-RED กับ ESP8266 ด้วยการสื่อสารแบบ TCP

โดยในแต่ละหมวดหมู่การทดสอบนั้น จะรวมถึงการทดสอบการรับประกันการส่งข้อมูล QoS 0, QoS 1 , QoS 2 สำหรับการเชื่อมต่อข้อมูลแบบ MQTT โดยจัดส่งข้อมูล ทั้งหมด 10,000 ครั้ง โดยใช้เวลา 500 ms ระหว่างชุดข้อมูล โดยขนาด ข้อมูล ที่จัดส่งนั้น มีขนาด หลากหลายในช่วง 50 - 500 byte ต่อ 1 ชุดข้อมูลในการจัดส่ง โดยในการตรวจสอบความหน่วงเวลานั้น ทางผู้จัดทำจะทำการตรวจสอบความหน่วงเวลาในการรับส่งข้อมูล ไป-กลับ (Round Trip Time) และ จำนวนข้อมูลที่ได้รับการตอบกลับ ในโปรแกรม Unity ( ชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital ) และ ชุดคอนโทรลเลอร์ ESP8266 ที่สั่งงานบนชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก แห่งความจริง ดังรูป ที่ 52.



รูป 52 ภาพจำลองการทดสอบการรับส่งข้อมูล Protocol MQTT

โดยในการทดสอบเพื่อบันทึกค่านันท่างผู้จัดทำได้ได้ประสบปัญหาการเก็บข้อมูลด้วย Software Unity ที่ทำการ simulate ชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital จากการทดลองเก็บข้อมูล ในช่วงแรก หากทำงานสั่งงานประมวลผล รับส่งชุดคำสั่งทั้งหมด 60,000 ชุดคำสั่ง ( 30,000 สำหรับ UDP(QoS 0-2) และ 10,000 สำหรับ TCP (QoS 0-2) ) ที่เดิยวนั้น จะเกิดปัญหาความหน่วงเวลา ในการรับส่งไม่เท่าเดิม ในระบบ Unity ทำให้การรับส่งข้อมูลช้าลง จนไม่สามารถทำการทดลอง 60,000 ในการรันโปรแกรมเพียงครั้งเดียวได้ ทางผู้จัดทำ ถึงทำการแบ่ง การเก็บข้อมูลในลักษณะ เป็นรอบ เล็กๆ รอบละ 5,000 คำสั่งโดยสาเหตุมาจากการทดสอบด้วยโปรแกรม Unity เป็นเวลานานนั้น ผู้จัดทำสังเกตเห็นช่วงเวลาระหว่างชุดคำสั่งที่ตั้งไว้ในโปรแกรม Unity ที่ 500 ms เริ่มมีการ คลาดเคลื่อนและเพิ่มขึ้นในเวลาจริงอย่างต่อเนื่องหากเปิดทดสอบการใช้งานเป็นเวลานาน (ประมาณ ชุดคำสั่งที่ 10,000 ค่าความหน่วงที่ตั้งไว้ในโปรแกรม 500 ms แต่เมื่อเทียบกับเวลาจริงในโปรแกรม จะใช้เวลาประมาณ 1 วินาที และมีแนวโน้มเพิ่มขึ้น ) ทำให้การทดสอบรับส่งชุดข้อมูลทั้งหมดในเวลา ไกลกันทีเดียวไม่สามารถดำเนินการได้ โดยสำหรับชุดข้อมูลในแต่ละ การรับประกันการสื่อสารนั้น อาจจะมี ความคลาดเคลื่อนเล็กน้อยเนื่องจากเวลาการใช้งานไม่ต่อเนื่องกันบนระบบอินเทอร์เน็ต สำหรับการสื่อสารแบบ MQTT ในระบบ

## บทที่ 4

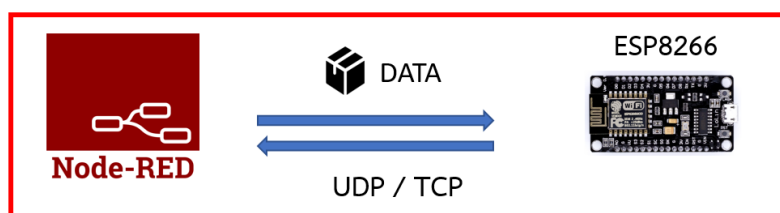
### ผลการทดลอง

ในบทนี้ผู้วิจัยได้แสดงผลของการทดลองตามกระบวนการทดลองที่ได้กล่าวไว้ในบทที่ 3 โดยจะแยกหัวข้อกล่าวถึง ผลการทดลองเพื่อแสดงค่าความหน่วงและเสถียรภาพ ของระบบสื่อสารทั้งหมดในชุดอุปกรณ์คัตแยกวัตต์อู๊ดโนมิตี และ คู่ฝาแฝดดิจิทัล โดยการนำเสนอผลการทดสอบนั้น สำหรับค่าความหน่วงจะอยู่ในรูปแบบของกราฟกล่อง และตารางค่ามัธยฐาน โดยกราฟที่นำมาแสดงนั้น ได้ทำการตัดค่า Outlier ( Lower range limit =  $Q1 - (1.5 * IQR)$  และ Higher range limit =  $Q3 + (1.5 * IQR)$  ) เพื่อให้ผู้อ่านเห็นภาพโดยรวมของกลุ่มข้อมูลได้ดีขึ้น และ นำเสนออัตราความสำเร็จในการส่งข้อมูล หรือ เสถียรภาพ จากการทดสอบในบทที่ 3 ภาพประกอบสาธิตระบบการทำงานของชุดอุปกรณ์คัตแยกวัตต์อู๊ดโนมิตี และ คู่ฝาแฝดดิจิทัล ในเวลาจริง

#### 4.1 ผลการทดลอง ความหน่วง และ เสถียรภาพ Node-RED และ ESP8266

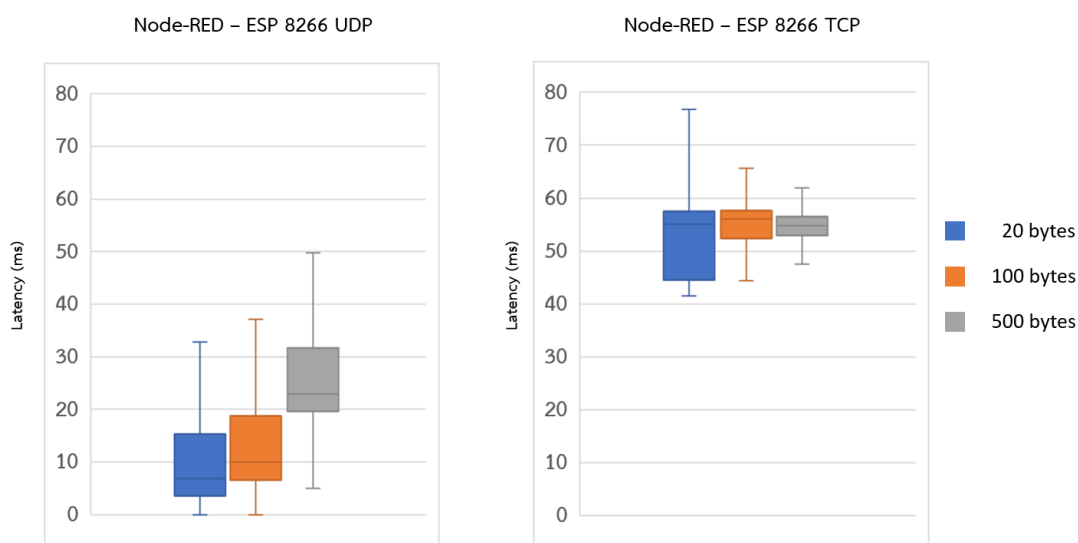
จากกระบวนการทดลองที่ได้กล่าวมาในหัวข้อที่ 3.4.2.1 ถึงกระบวนการทดลองจัดส่งข้อมูล ขนาด 20 bytes , 100 bytes และ 500 bytes จำนวน 10,000 ครั้งระหว่าง ชุดประมวลผลหลัก Node-RED และ บอร์ดคอนโทรลเลอร์ ESP8266 เพื่อหาค่าความหน่วง และ ความเสถียรภาพของระบบสื่อสาร UDP และ TCP ดังรูปที่ 53. โดยจะนำเสนอผลการทดลองออกเป็น 2 หัวข้อย่อยดังนี้

10,000 ครั้ง / 20 bytes , 100 bytes และ 500 bytes



รูป 53 ภาพจำลองแบบย่อสำหรับการทดสอบการรับส่งข้อมูลบน Protocol UDP และ TCP

#### 4.1.1 ค่าความหน่วงของระบบการสื่อสารผ่านระบบ UDP และ TCP



รูป 54 กราฟกล่องสำหรับค่าความหน่วงของระบบการสื่อสารระหว่าง Node-RED และ ESP8266 ผ่านระบบ UDP และ TCP

ตาราง 2 ค่ามัธยฐาน (Median) ของความหน่วง ของระบบการสื่อสารแบบ UDP และ TCP ในชุดข้อมูลที่แตกต่างกัน

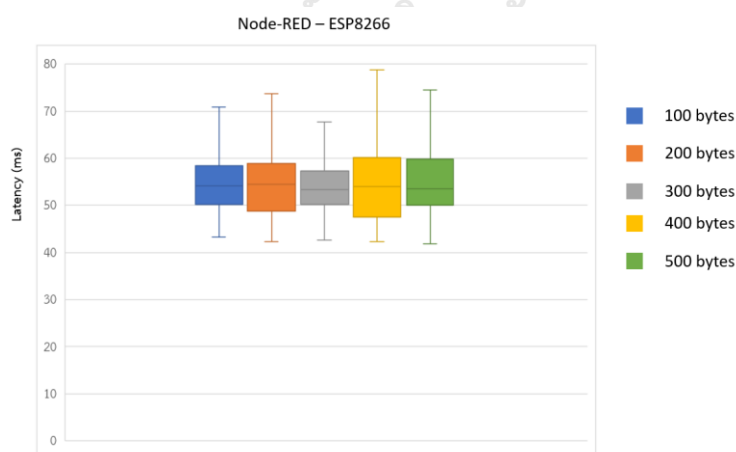
การสื่อสาร	ขนาดข้อมูล		
	20 bytes	100 bytes	500 bytes
UDP	6.903 ms	9.995 ms	22.979 ms
TCP	55.02 ms	56.01 ms	54.828 ms

โดยจากผลการทดสอบในรูปที่ 54. และ ตารางที่ 2. นั้น จะเห็นได้ว่าการสื่อสารแบบ UDP มีค่าความหน่วงในการรับส่งข้อมูลแบบไปกลับอยู่ในช่วงประมาณ 6 - 30ms ขึ้นอยู่กับขนาดข้อมูลที่จัดส่ง และการสื่อสารแบบ TCP จะมีค่าความหน่วงในการรับส่งข้อมูลแบบไปกลับอยู่ในช่วงประมาณ 54-56ms โดยซึ่งมีความใกล้เคียงในการรับส่งชุดข้อมูลกันของทุกชุดขนาดข้อมูล และ มีค่าความหน่วงช้ากว่าการสื่อสารแบบ UDP อยู่ที่ 30-49ms โดยประมาณ เป็นเพราะด้วยระบบการสื่อสารแบบ TCP มีการการันตีการรับส่งข้อมูลกัน โดยผู้จัดทำได้ตรวจสอบการรับส่งข้อมูลผ่านโปรแกรม Wireshark เพิ่มเติม เพื่อตรวจสอบการการันตีการรับส่งข้อมูล ดังรูปที่ 55.

0.001445	TCP	554 54074 → 1027 [PSH, ACK] Seq=1187733 Ack=1185367 Win=1874 Len=500
0.047647	TCP	54 1027 → 54074 [ACK] Seq=1185367 Ack=1188233 Win=64890 Len=0
0.003055	TCP	60 54074 → 1027 [PSH, ACK] Seq=1188233 Ack=1185367 Win=1874 Len=2
0.000288	TCP	555 1027 → 54074 [PSH, ACK] Seq=1185367 Ack=1188235 Win=64888 Len=501
0.036080	TCP	60 54074 → 1027 [ACK] Seq=1188235 Ack=1185868 Win=1909 Len=0
0.001373	TCP	554 54074 → 1027 [PSH, ACK] Seq=1188235 Ack=1185868 Win=1909 Len=500
0.052078	TCP	54 1027 → 54074 [ACK] Seq=1185868 Ack=1188735 Win=65392 Len=0
0.003091	TCP	60 54074 → 1027 [PSH, ACK] Seq=1188735 Ack=1185868 Win=1909 Len=2
0.000288	TCP	555 1027 → 54074 [PSH, ACK] Seq=1185868 Ack=1188737 Win=65390 Len=501
0.046269	TCP	60 54074 → 1027 [ACK] Seq=1188737 Ack=1186369 Win=1944 Len=0

รูป 55 ตัวอย่างการการันตีข้อมูลจากโปรแกรม Wireshark

โดยนอกเหนือจากชุดข้อมูลที่มีการจัดส่ง ไปกลับ ระหว่างชุดประมวล Node-RED และ ชุดบอร์ดคอนโทรลเลอร์ ESP8266 แล้ว ยังมีความหวังในการจัดส่งข้อความเพื่อการันตีการรับส่งข้อความด้วย (PSH,ACK) [24] จึงเป็นเหตุผลที่ทำให้การรับส่งข้อมูลแบบ TCP นั้นมีความหวังที่มากกว่าการรับส่งข้อมูลแบบ UDP และอีกประเด็นที่น่าสนใจสำหรับการสื่อสารแบบ TCP ผู้จัดทำได้สังเกตเห็นว่า ในทุกชุดข้อมูลการจัดส่งนั้น ค่าความหวังในการจัดส่ง จากค่ามัธยฐานนั้น จะอยู่ในช่วงที่มีความใกล้เคียงกัน โดยทางผู้วิจัยจึงตั้งสมมุติฐานว่าการสื่อสารแบบ TCP นั้น การเพิ่มขนาดข้อมูลที่ใช้สำหรับรับส่งนั้น มีผลต่อค่าความหวังไม่มากนัก โดยทางผู้จัดทำได้ทำการทดสอบเพิ่มเติมเพื่อพิสูจน์สมมุติฐานดังกล่าว ทางผู้จัดทำได้ทำการเพิ่มการทดลองโดยเป็นใช้ชุดข้อมูล TCP ในขนาด 100,200,300,400,500 bytes เพื่อตรวจสอบหาแนวโน้มค่าความหวังที่เกิดขึ้น จากการรับส่ง 1,000 ครั้ง

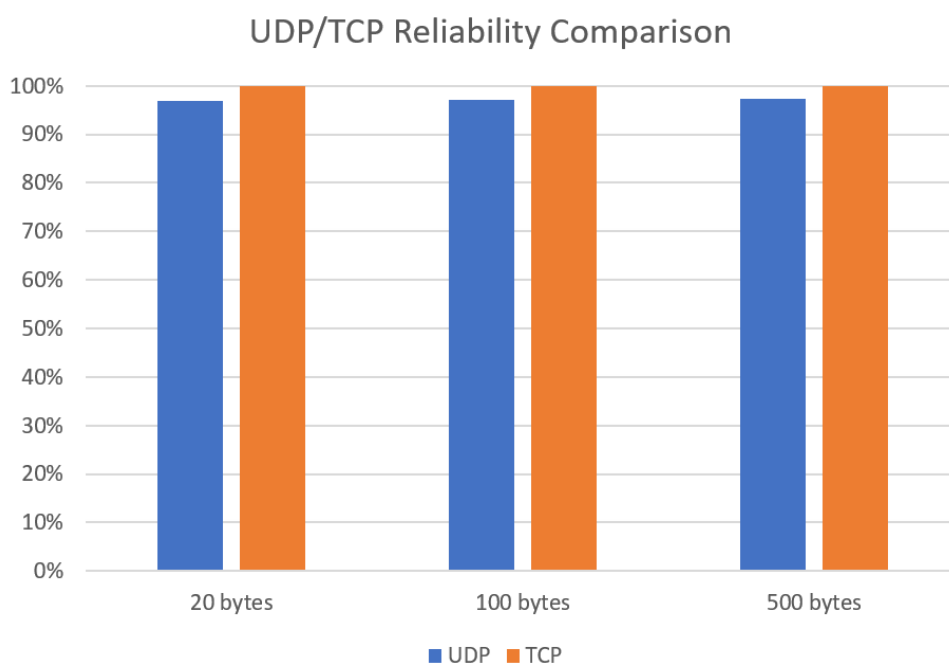


รูป 56 กราฟกล่องสำหรับค่าความหวังของระบบการสื่อสารระหว่าง Node-RED และ ESP8266 ผ่านระบบ TCP เพิ่มเติม



โดยจากรูปที่ 56. นั้นจะเห็นได้ว่าค่าความหน่วงในทุกระดับข้อมูล ( 100,200,300,400,500 bytes) นั้นเป็นการยืนยันได้ว่าการรับส่งข้อมูลจากชุดอุปกรณ์ประมวลผลหลัก Node-RED และ ชุดบอร์ดคอนโทรลเลอร์ ESP8266 จะมีค่าความหน่วงที่ไม่แตกต่างกันมากนักในระบบการสื่อสารแบบ TCP สำหรับการรับส่งข้อมูล 20-500 bytes.

#### 4.1.2 เสถียรภาพของระบบการสื่อสารผ่านระบบ UDP และ TCP



รูป 57 กราฟแสดงค่าความเสถียรของระบบการสื่อสารระหว่าง Node-RED และ ESP8266 ผ่านระบบ UDP และ TCP

ตาราง 3 ค่าความเสถียรภาพ ของระบบการสื่อสารแบบ UDP และ TCP ในชุดข้อมูลที่แตกต่างกัน

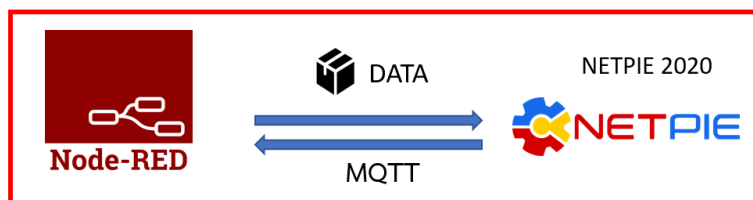
การสื่อสาร	ขนาดข้อมูล		
	20 bytes	100 bytes	500 bytes
UDP	96.98%	97.20%	97.40%
TCP	100%	100%	100%

โดยในผลการทดสอบในรูปแบบที่ 57. และ ตารางที่ 3. นั้น จะเห็นได้ว่าการใช้งานการสื่อสารแบบ UDP ในการรับส่งข้อมูล 10,000 ครั้งนั้นอัตราความสำเร็จในการรับส่งข้อมูลนั้นอยู่ที่ช่วงประมาณ 96.9 – 97.5 % ซึ่งในการรับส่งข้อมูลแบบ UDP นั้นมีบางส่วนของชุดข้อมูลที่ไม่สำเร็จในระหว่างการรับส่งข้อมูลกันระหว่างชุดประมวลผลหลัก Node-RED และ บอร์ดคอนโทรลเลอร์ ESP8266 โดยการสื่อสารแบบ TCP นั้นเสถียรภาพ อัตราความสำเร็จในการรับส่งข้อมูลอยู่ที่ 100% หรือกล่าวคือทุก การส่งข้อมูลจาก ชุดประมวลผลหลัก Node-RED ไปยัง ชุดบอร์ดคอนโทรลเลอร์ ESP8266 ในการ ทดลอง 10,000 ครั้งนั้น มีการรับส่งข้อมูลและตอบกลับมายัง ชุดประมวลผลหลัก Node-RED ทุก ครั้งสำหรับการสื่อสารแบบ TCP

#### 4.2 ผลการทดลอง ความหน่วง และ เสถียรภาพ Node-RED และ NETPIE 2020

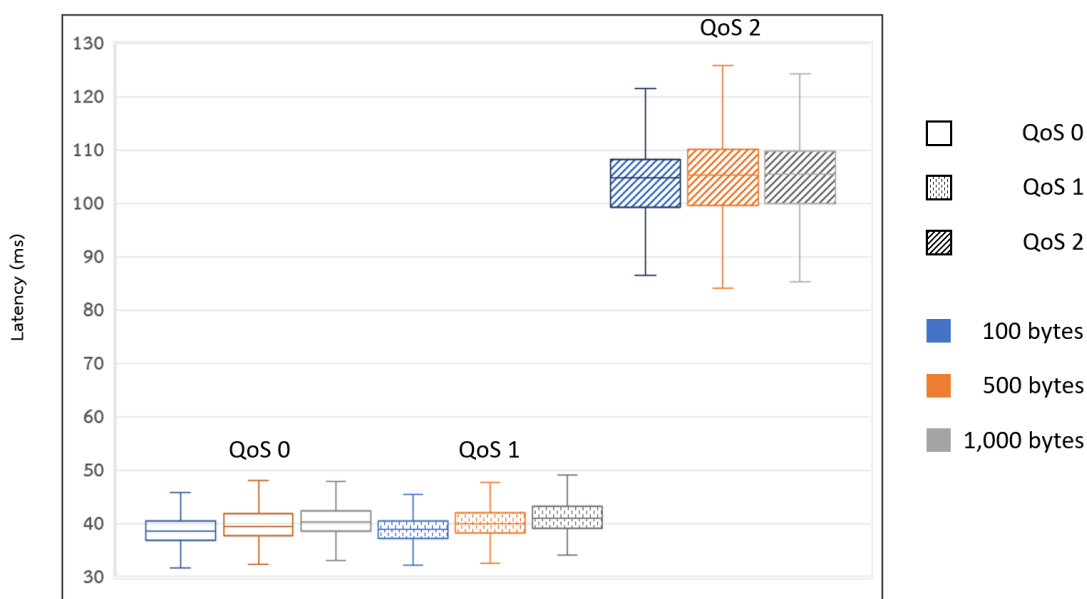
จากกระบวนการทดลองที่ได้กล่าวมาในหัวข้อที่ 3.4.2.2 ถึงกระบวนการทดลองจัดส่ง ข้อมูล ขนาด 100 bytes , 500 bytes และ 1,000 bytes จำนวน 10,000 ครั้งระหว่าง ชุด ประมวลผลหลัก Node-RED และ แพลตฟอร์ม NETPIE2020 เพื่อหาค่าความหน่วง และ ความ เสถียรภาพของระบบสื่อสาร MQTT ในทุกการรับประกันการส่งข้อมูล ( QoS 0 , QoS 1 และ QoS 2 ) ดังรูปที่ 58. สามารถแสดงผลการทดลองได้ดังนี้

10,000 ครั้ง / 50 bytes , 500 bytes และ 1000 bytes



รูป 58 ภาพจำลองแบบย่อสำหรับการทดสอบการรับส่งข้อมูลบน แพลตฟอร์ม NETPIE 2020

#### 4.2.1 ค่าความหน่วงของระบบการสื่อสารบน แพลตฟอร์ม NETPIE 2020



รูป 59 กราฟกล่องแสดงค่าความหน่วงของระบบการสื่อสาร MQTT บนแพลตฟอร์ม NETPIE 2020

ตาราง 4 ค่ามัธยฐาน ( Median ) ของความหน่วง ของระบบสื่อสารแบบ MQTT ระหว่าง Node-RED และ NETPIE 2020

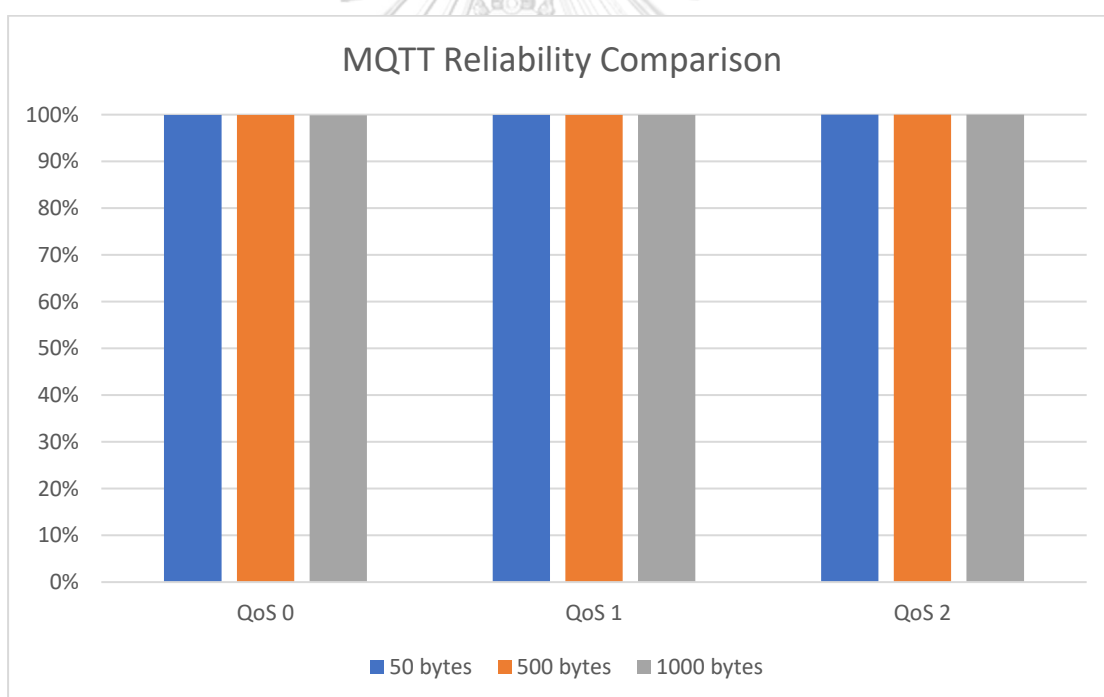
การรับประกันการส่งข้อมูล	ขนาดข้อมูล		
	100 bytes	500 bytes	1,000 bytes
QoS 0	38.52 ms	39.46 ms	40.24 ms
QoS 1	38.85 ms	39.93 ms	40.94 ms
QoS 2	104.76 ms	105.28 ms	105.49 ms

โดยจากผลการทดสอบในรูปที่ 59. และ ตารางที่ 4. นั้น จะเห็นได้ว่าการสื่อสารแบบ MQTT นั้นมีค่าความหน่วงในการรับส่งข้อมูล สำหรับ ทุกชุดข้อมูล 100,500 และ 1000 bytes ที่ไม่แตกต่างกันมากนักในการรับประกันการส่งข้อมูลแบบเดียวกันโดยค่าความหน่วงในแต่ละชุดข้อมูลที่แตกต่างกันมีค่าความหน่วงอยู่ในช่วงประมาณ 1 – 2 ms และ อีกหนึ่งประเด็นที่น่าสนใจคือค่าความหน่วงของการรับส่งข้อมูลสำหรับการรับประกันการส่งข้อมูลแบบ QoS 0 และ QoS 1 มีค่าความหน่วงที่ใกล้เคียงกัน โดยในการจัดส่งข้อมูลแบบการรับประกัน QoS 1 นั้นจะมีค่าความหน่วงที่มากกว่าเล็กน้อยเนื่องจากการตอบรับจากทางผู้รับข้อมูลว่าได้รับข้อมูลเรียบร้อยแล้วดังรูปที่ 51. โดยจะมีค่า

ความหน่วงในการรับส่งข้อมูล แบบไปกลับอยู่ในช่วงประมาณ 38-41 ms โดย สำหรับการรับประกักับการส่งข้อมูลแบบ QoS 2 นั้นเนื่องจากการรับประกัการสื่อสารกันระหว่างผู้ส่งกับผู้รับ โดยทำให้ค่าความหน่วงในการรับส่งข้อมูลแบบไปกลับจะอยู่ในช่วงประมาณ 104-106 ms ซึ่งค่าความหน่วงของการรับประกัการส่งข้อมูลแบบ QoS 2 นั้นมีค่าความหน่วงที่เพิ่มขึ้นมาอย่างมีนัยสำคัญสำหรับกระบวนการรับส่งข้อมูล เมื่อเปรียบเทียบกับ การรับส่งข้อมูล แบบ QoS 0 และ QoS 1 บนแพลตฟอร์มของ NETPIE 2020 และ หากสังเกต ค่าความหน่วง ในทุกช่วงข้อมูลของ การรับประกัการส่งข้อมูลแบบ QoS 2 นั้น ในทุกช่วงการส่งข้อมูล ถึงแม้ว่า จำนวนข้อมูลที่ได้รับส่งจะมีมากขึ้นจาก 100 bytes เป็น 500 และ 1000 bytes แต่ค่าความหน่วงที่วัดได้นั้นก็ไม่ได้มีการเพิ่มอย่างเป็นนัยยะสำคัญเช่นกัน



#### 4.2.2 เสถียรภาพของระบบการสื่อสารบน แพลตฟอร์ม NETPIE 2020



รูป 60 กราฟแสดงค่าความเสถียรของระบบการสื่อสารระหว่างแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020

ตาราง 5 ค่าความเสถียรภาพ ของระบบสื่อสารแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020

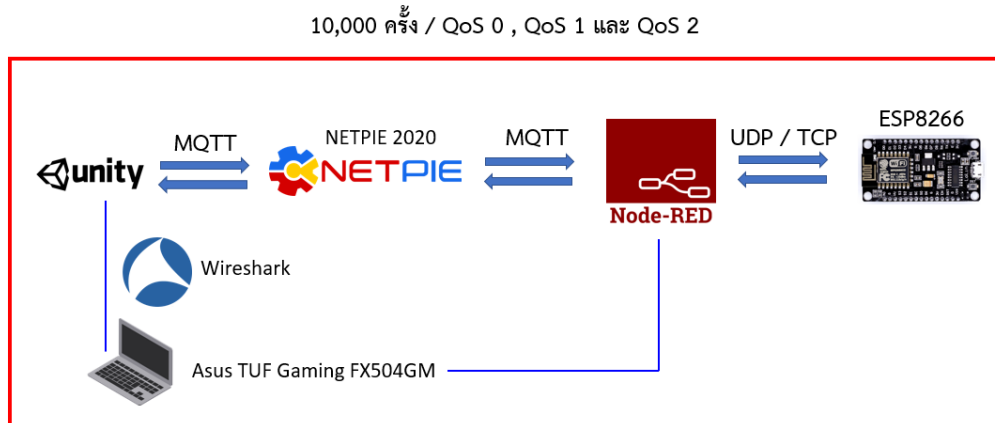
การรับประกันการส่ง ข้อมูล	ขนาดข้อมูล		
	100 bytes	500 bytes	1,000 bytes
QoS 0	99.95 %	99.94 %	99.90 %
QoS 1	99.95 %	99.94 %	99.95 %
QoS 2	100 %	100 %	100 %

โดยในผลการทดสอบในรูปที่ 60. และ ตารางที่ 5. นั้น จะเห็นได้ว่าการใช้งานการสื่อสารแบบ MQTT ในทุกระดับการรับประกันการส่งข้อมูล บนแพลตฟอร์มของ NETPIE 2020 นั้นมีเสถียรภาพในการรับส่งที่ดี โดยสำหรับ การรับประกันการส่งข้อมูล 10,000 ครั้ง สำหรับ QoS 0 และ QoS 1 อัตราความสำเร็จในการรับส่งข้อมูลจะอยู่ที่ประมาณ 99.90 - 99.95 % และ สำหรับ QoS 2 นั้นอัตราความสำเร็จในการรับส่งข้อมูลอยู่ที่ 100% ในทุกขนาดข้อมูลที่ได้มีการจัดส่ง อย่างไรก็ตามในการทดสอบครั้งนี้นั้นการเชื่อมต่อระหว่าง Node-RED และ NETPIE 2020 ได้ถูกตั้งค่าดั้งเดิมใน Node MQTT บน Node-RED ซึ่งไม่ได้มีการตั้งค่าการเชื่อมต่อแบบ Persistent Session ซึ่งอาจจะเป็นเหตุผลหนึ่งที่ทำให้เกิดการรับส่งชุดข้อมูลที่ไม่ครบถ้วนสมบูรณ์ตามทฤษฎี ของการรับประกันการส่งข้อมูล QoS 1 โดยทางผู้จัดทำได้ได้พยายามลองทำการทดสอบเพิ่มเติมโดยการตั้งค่าให้การเชื่อมต่อแบบ Persistent Session แต่ในการทดลองทางผู้จัดทำได้พบเจอปัญหาระหว่างการทดลองจึงยังไม่สามารถนำผลมาเสนอในวิทยานิพนธ์ฉบับนี้ได้ โดยจากผลการทดลองที่ได้นำเสนอขึ้นเมื่อนำไปเปรียบเทียบกับงานวิจัยระบบการสื่อสารแบบ MQTT ที่มีการทดสอบเสถียรภาพของการสื่อสารแบบ MQTT นั้น [40][41] การสื่อสารผ่านระบบเครือข่ายไร้สายบนอินเทอร์เน็ตนั้นผลการทดสอบเสถียรภาพมีความใกล้เคียงกันตามผลลัพธ์ในวิทยานิพนธ์ฉบับนี้ ในการรับประกันการส่งข้อมูลแบบ QoS 0 , QoS 1 และ QoS 2 โดยการรับประกันการส่งข้อมูลแบบ QoS 0 และ QoS 1 นั้น ข้อมูล message loss ประมาณ 1% โดยมีแนวโน้มที่จะเพิ่มขึ้นตามจำนวนชุดข้อมูลที่ได้รับส่ง และ ถึงแม้จะใช้การรับประกันการส่งข้อมูลแบบ QoS 2 ก็ยังคงมีข้อความบางส่วนหายไปในระหว่างการทดลอง

#### 4.3 ผลการทดลอง ความหน่วง และ เสถียรภาพ ของคู่ฝาแฝดดิจิทัล

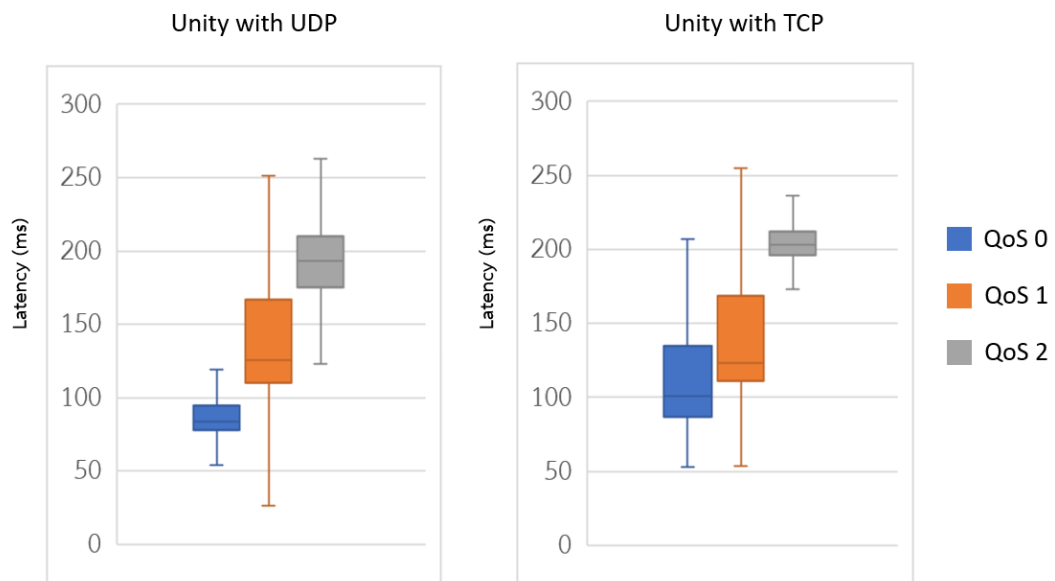
จากกระบวนการทดลองที่ได้กล่าวมาในหัวข้อที่ 3.4.2.3 ถึงกระบวนการทดลองสั่งงานคู่ฝาแฝดดิจิทัล ด้วย จัดส่งข้อมูลผ่านจากโปรแกรม Unity (VR) ที่ Simulate บนอุปกรณ์คอมพิวเตอร์พกพาไปยังแพลตฟอร์ม NETPIE และอัปเดตข้อมูลกลับมายังชุดประมวลผลหลัก Node-RED ที่ติดตั้งอยู่

บนอุปกรณ์คอมพิวเตอร์พกพาเช่นเดียวกัน เพื่อส่งงานบอร์ดคอนโทรลเลอร์ ESP8266 ในชุดอุปกรณ์ คัดแยกวัตถุอัตโนมัติ ในเวลาจริง จำนวน 10,000 ครั้ง ในทุกการรับประกันการส่งข้อมูล (QoS 0 , QoS 1 และ QoS 2 ) ดังรูป ที่ 61. สามารถแสดงผลการทดลองได้ดังนี้



รูป 61 ภาพจำลองแบบย่อสำหรับการทดสอบรับส่งข้อมูลของฝาฝาแฝดดิจิทัลผ่านระบบสื่อสาร MQTT กับ protocol UDP/TCP

#### 4.3.1 ค่าความหน่วงของระบบการสื่อสารของคู่ฝาแฝดดิจิทัลผ่านระบบ UDP ,TCP และ MQTT

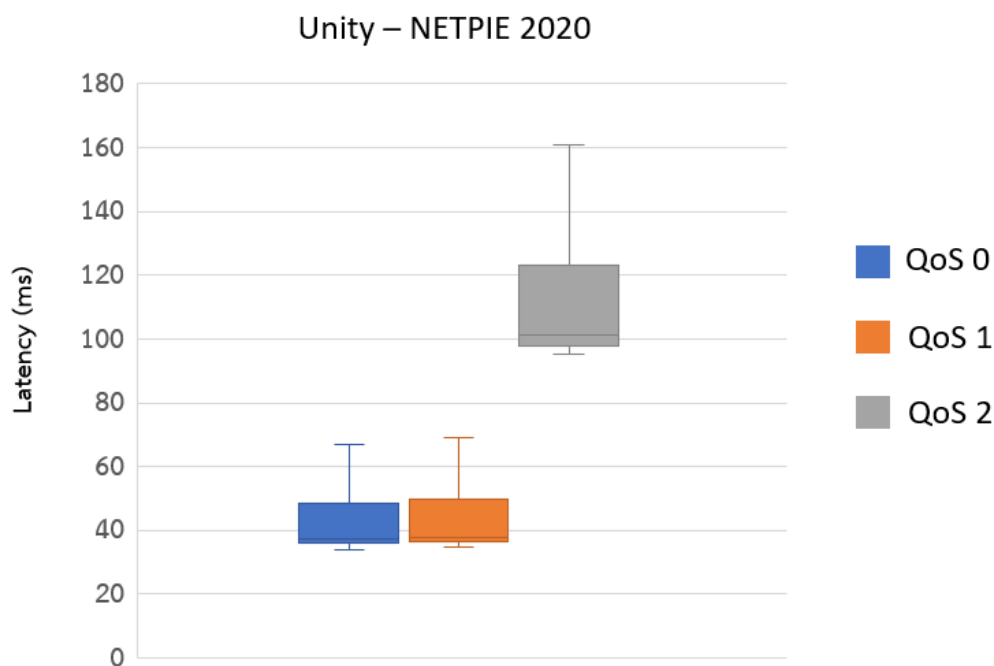


รูป 62 กราฟกล่องสำหรับค่าความหน่วงของระบบการสื่อสารระหว่าง คู่ฝาแฝดดิจิทัล ผ่านระบบ UDP และ TCP

ตาราง 6 ค่ามัธยฐาน (Median) ของความหน่วงของระบบสื่อสารของคู่ฝาแฝดดิจิทัล ผ่านระบบ UDP และ TCP

	การรับประกันการส่งข้อมูลของ MQTT		
	QoS 0	QoS 1	QoS 2
UDP	83.44 ms	125.80 ms	193.07 ms
TCP	101.14 ms	122.97 ms	203.10 ms

โดยจากผลการทดสอบในรูปแบบที่ 62. และ ตารางที่ 6 นั้น ได้แสดงถึงค่าความหน่วงแบบไปกลับ สำหรับระบบสื่อสารของคู่ฝาแฝดดิจิทัล ของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ โดยค่าความหน่วงที่ต่ำที่สุดจะเป็นการรับประกันการสื่อสารที่ QoS 0 ในการสื่อสารแบบ MQTT ผ่านแพลตฟอร์มของ NETPIE 2020 และ ใช้การสื่อสารแบบ UDP สำหรับส่งงานบอร์ดคอนโทรลเลอร์ ESP8266 บนชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติโดยค่าความหน่วงแบบไปกลับอยู่ที่ อยู่ที่ 83.44 ms ซึ่งต่ำกว่า การใช้งานการสื่อสารแบบ TCP สำหรับส่งงานบอร์ดคอนโทรลเลอร์ ESP8266 ประมาณ 17.7 ms และ เมื่อเปรียบเทียบ การรับประกันการส่งข้อมูลในระดับ QoS 0 และ QoS 1 แล้วนั้น ค่าความหน่วงมีการเพิ่มขึ้นประมาณ 20-40 ms โดยเป็นจุดที่น่าสนใจสำหรับผู้จัดทำ โดยจากผลการทดสอบการสื่อสารระหว่างชุดประมวลผลหลัก Node-RED และ NETPIE 2020 นั้น ความแตกต่างระหว่างการรับประกันการรับส่งข้อมูลในระดับ QoS 0 และ QoS 1 นั้นไม่แตกต่างกันมาก แต่ในการทดสอบโดยภาพรวมระหว่างคู่ฝาแฝดดิจิทัลเมื่อมีการใช้งานโปรแกรม Unity ผ่าน Library M2MQTT ด้วยนั้น ค่าความหน่วงระหว่างการรับส่งข้อมูลแบบ QoS 0 และ QoS 1 มีความแตกต่างเพิ่มขึ้นอย่างชัดเจน ดังรูปที่ 62. ซึ่งเป็นประเด็นที่น่าสนใจในการทดสอบความหน่วงของระบบสื่อสารผ่าน Library M2MQTT โดยทางผู้จัดทำได้ทำการทดสอบการสื่อสารระหว่างโปรแกรม Unity และ NETPIE 2020 เพิ่มเติมเพื่อหาข้อสังเกตการเพิ่มขึ้นของค่าความหน่วงในการรับส่งระหว่าง QoS 0 , QoS 1 และ QoS 2 ทั้งหมด 1,000 ครั้ง ด้วยชุดข้อมูล ประมาณ 500 bytes ได้ผลดังรูป ดังรูปที่ 63.



รูป 63 ค่าความหน่วงจากการทดสอบการสื่อสารแบบ MQTT ระหว่าง Unity และ NETPIE 2020

ตาราง 7 ค่ามัธยฐาน ( Median ) ของความหน่วง ของระบบสื่อสารแบบ MQTT ระหว่าง Unity และ NETPIE 2020

	การรับประกันการส่งข้อมูลของ MQTT		
	QoS 0	QoS 1	QoS 2
Unity – NETPIE 2020	37.24 ms	37.64 ms	101.246 ms

จากตารางที่ 7. ค่ามัธยฐาน ค่าความหน่วงจากการทดสอบจากการทดสอบแสดงให้เห็นว่า การสื่อสารด้วย MQTT ระหว่างโปรแกรม Unity กับ NETPIE 2020 นั้น ในทุกการรับประกันการส่งข้อมูลมีความใกล้เคียงกับการสื่อสารระหว่างชุดประมวลผล Node-RED และ NETPIE 2020 โดยค่าความหน่วงของ QoS 0 และ QoS 1 นั้นไม่ได้เพิ่มขึ้นอย่างมีนัยสำคัญแต่อย่างใดจึงเป็นที่น่าสังเกตว่าเหตุใดการปรับระดับ QoS 0 เป็น QoS 1 ในกระบวนการทดสอบภาพรวมระหว่างคู่ฝาแฝดดิจิทัลถึงมีการเพิ่มขึ้นของความหน่วงโดยทางผู้จัดทำยังคงมิได้หาข้อสรุปในวิทยานิพนธ์ฉบับนี้



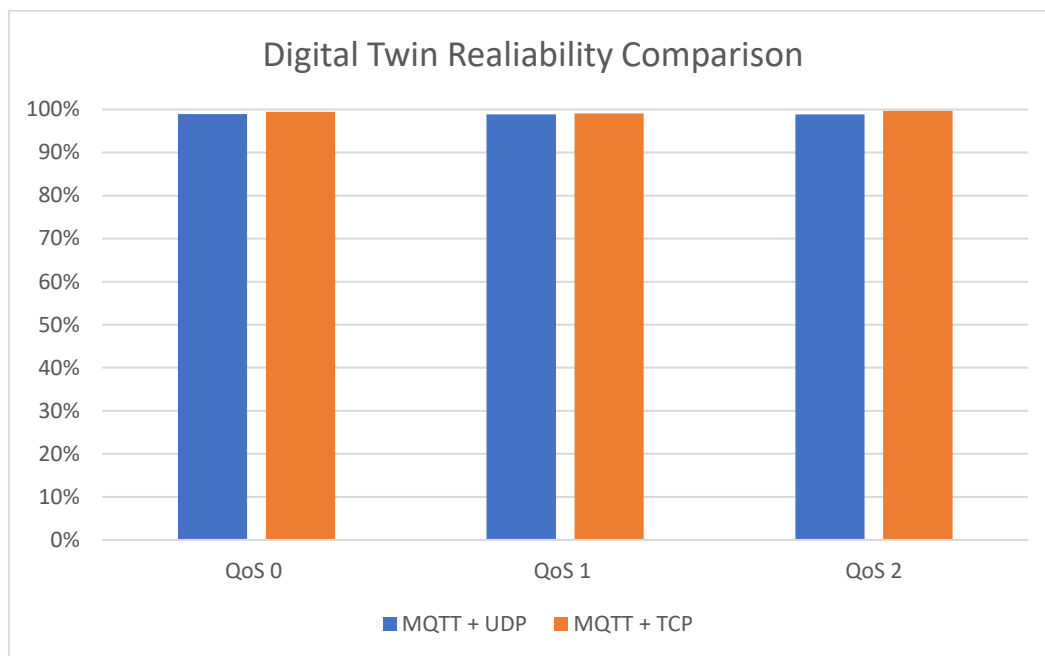
และจากผลการทดลองของระบบสื่อสารผ่านเครือข่ายที่พัฒนาขึ้นเมื่อนำค่าความหน่วงต่ำที่สุด ( ใช้การสื่อสารแบบ UDP รับประกันข้อมูล QoS 0 ) มาเปรียบเทียบกับความเร็วในการทำงานของระบบกับระบบจากงานวิจัยอื่นๆที่ใกล้เคียง ทางผู้จัดทำจึงนำค่าความหน่วงตามค่ามัธยฐานของระบบผ่านเครือข่าย 2 โดยเป็นการอนุมานว่าเป็นค่าความหน่วงส่งข้อมูลแบบฝั่งเดียว จากคู่ผ่านเครือข่ายไปยังชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติเพื่อทำการเปรียบเทียบกับระบบการสื่อสารจากงานวิจัยที่ใกล้เคียง ดังตารางที่ 8

ตาราง 8 การเปรียบเทียบระบบสื่อสารที่พัฒนาขึ้นกับงานวิจัยที่ใกล้เคียง

ระบบ	Latency (ms)
ระบบที่พัฒนาขึ้น	$(83.44 / 2) = 41.72$
VRITNESS [6]	43.7
Interactive Visual Procedure [5]	51.2
RCDT [42]	70

จากการเปรียบเทียบงานวิจัยที่ใกล้เคียงนั้นจะเห็นว่า ค่าความหน่วง ในลักษณะงานที่มีการใช้งานในระบบสื่อสารที่ใกล้เคียงกัน มีค่าความหน่วงที่อยู่ในเกณฑ์ใกล้เคียงกัน โดย VRITNESS [6] ที่นำเสนอวิธีการใช้ MQTT protocol บนแพลตฟอร์มของ Adafruit IO [43] จัดส่งข้อมูลผ่าน Software Unity โดยตรงไปยัง บอร์ดคอนโทรลเลอร์ Raspberry Pi ที่สั่งงานผ่านอุปกรณ์ VR มีค่าความหน่วงเฉลี่ยอยู่ที่ 43.7s ซึ่งมีความใกล้เคียงกับระบบที่พัฒนาขึ้น ถึงแม้ระบบที่พัฒนาขึ้นในวิทยานิพนธ์ฉบับนี้จะมีการสื่อสารแบบ UDP เพิ่มเข้าไปในระบบด้วยก็ตาม

#### 4.3.2 เสถียรภาพของระบบการสื่อสารของคู่ฝาแฝดดิจิทัลผ่านระบบ UDP ,TCP และ MQTT



รูป 64 กราฟแสดงค่าความเสถียรของระบบการสื่อสารของคู่ฝาแฝดดิจิทัลผ่านระบบ UDP , TCP และ MQTT

ตาราง 9 ค่าความเสถียรภาพ ของระบบสื่อสารแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020

การรับประกันการส่งข้อมูล	การรับประกันการส่งข้อมูลของ MQTT		
	QoS 0	QoS 1	QoS 2
UDP	98.93 %	98.84 %	98.89 %
TCP	99.44%	99.12 %	99.68 %

โดยในผลการทดสอบในรูปแบบที่ 64. และ ตารางที่ 9. นั้นจะเห็นได้ว่าอัตราความสำเร็จในการรับส่งข้อมูลระหว่างคู่ฝาแฝดดิจิทัล ในโปรแกรม Unity กับ ชุดอุปกรณ์จริง ด้วยตัวประมวลผล Node-RED นั้นมีอัตราความสำเร็จอยู่ที่ประมาณ 98.8 – 99.7% โดยในการรับส่งข้อมูลที่มีการรับประกันต่ำที่สุดคือ QoS 0 สำหรับ แพลตฟอร์ม NETPIE 2020 และ สื่อสารผ่านระบบ UDP สำหรับชุดอุปกรณ์จริงก็ยังมีพบว่ามีอัตราการ รับส่งข้อมูลสำเร็จมากถึง 98.93 % แต่ในทางกลับกันนั้นการเลือกใช้การรับประกันการส่งข้อมูล กล่าวคือ การเลือกใช้การรับประกันการส่งข้อมูลแบบ QoS 1-2 บนแพลตฟอร์ม

ฟอร์มของ NETPIE 2020 และ ใช้งานการสื่อสารแบบ TCP สำหรับชุดอุปกรณ์จริง โดยในการทดลองครั้งนี้ ก็ยังมีข้อมูลบางส่วนที่ไม่สามารถ รับส่ง ได้สมบูรณ์ครบถ้วน โดยอัตราการรับส่งข้อมูลสำเร็จ อยู่ที่ 99.68% จากการทดลองรับส่งข้อมูลทั้งหมด 10,000 ครั้ง

#### 4.4 สาธิตระบบปฏิบัติการของระบบฝาแฝดดิจิทัลสำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัติ

ในการสาธิตระบบปฏิบัติการของระบบฝาแฝดดิจิทัลสำหรับอุปกรณ์คัดแยกวัตถุอัตโนมัตินั้น ทางผู้จัดทำได้ทำการถ่ายภาพเพื่อเปรียบเทียบกระบวนการทำงานของชุดอุปกรณ์จริงและคู่ฝาแฝดดิจิทัลในเวลาจริง

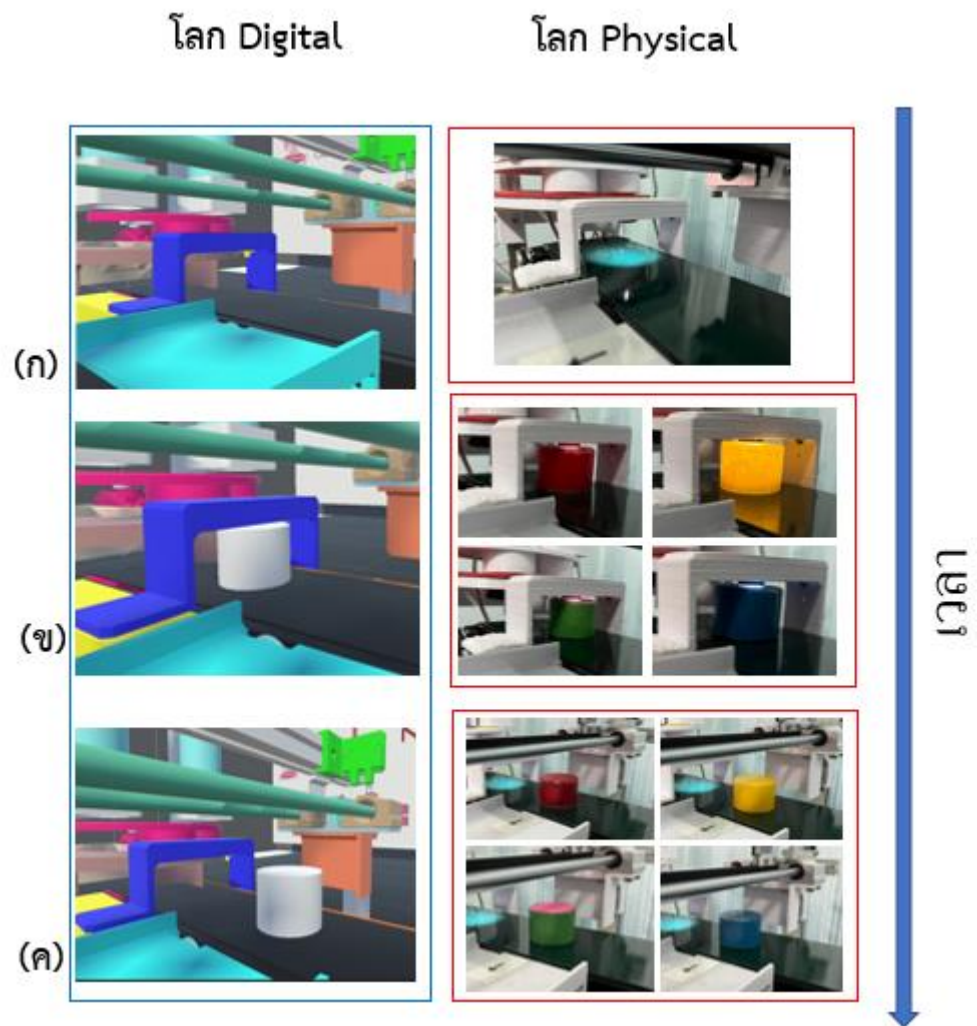
โดยเมื่อผู้ใช้งานเริ่มสั่งการการทำงานของชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ แล้วนั้น ชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติในโลกแห่งความจริงจะเริ่มปฏิบัติการตามกระบวนการของแต่ละโมดูลในหัวข้อที่ 3.1.2 โดยชุดประมวลผลหลัก Node-RED จะทำการส่งข้อมูลอัปเดตสถานะของชุดอุปกรณ์แต่ละอย่างเพื่อเชื่อมต่อไปยัง ฝาแฝดดิจิทัล ตามหัวข้อ 3.2.3 โดยคู่ฝาแฝดดิจิทัลนั้นจะแสดงผลค่าสถานะ และ ภาพเคลื่อนไหวให้สอดคล้องกับอุปกรณ์ในเวลาจริง โดยลำดับการทำงานนั้นจะเป็นไปดังนี้

##### 4.4.1 กระบวนการทำงานขั้นต้น

รูปที่ 65.(ก) เริ่มกระบวนการทำงานของระบบฝาแฝดดิจิทัล

รูปที่ 65.(ข) วัตถุเคลื่อนที่จากหลอดใส่วัตถุ จากโมดูลที่ 1 ลงมายังชุดสายพาน ขอบโมดูลที่ 2 และเคลื่อนตัวไปได้เซนเซอร์ตรวจจับสี บนโมดูลที่ 3 แล้วนั้น ชุดประมวลผล Node-RED จะจัดส่งข้อมูลแจ้งตำแหน่งของวัตถุไปยังคู่ฝาแฝดดิจิทัล โดย ข้อจำกัดสำหรับคู่ฝาแฝดดิจิทัลนั้นเนื่องจากข้อมูลสีที่ได้รับจากชุดอุปกรณ์ในโลกแห่งความจริงยังไม่เพียงพอสำหรับการยืนยันสีวัตถุบนระบบคัดแยกวัตถุอัตโนมัติ สีของวัตถุในคู่ฝาแฝดดิจิทัลจึงแสดงเป็นสีขาว ในทุกวัตถุจริงที่ผ่านเซนเซอร์ตรวจจับสี

รูปที่ 65.(ค) เมื่อวัตถุเคลื่อนที่ผ่านเซนเซอร์สีแล้ว ชุดประมวลผล Node-RED จะนำข้อมูลชุดสีที่ได้รับจากโมดูลที่ 3 และ ตั้งค่าเวลาเพื่อสั่งงานชุดผลักวัตถุในโมดูลที่ 4 โดยในแต่ละสีของวัตถุจะมีกระบวนการทำงานที่แตกต่างกันออกไป



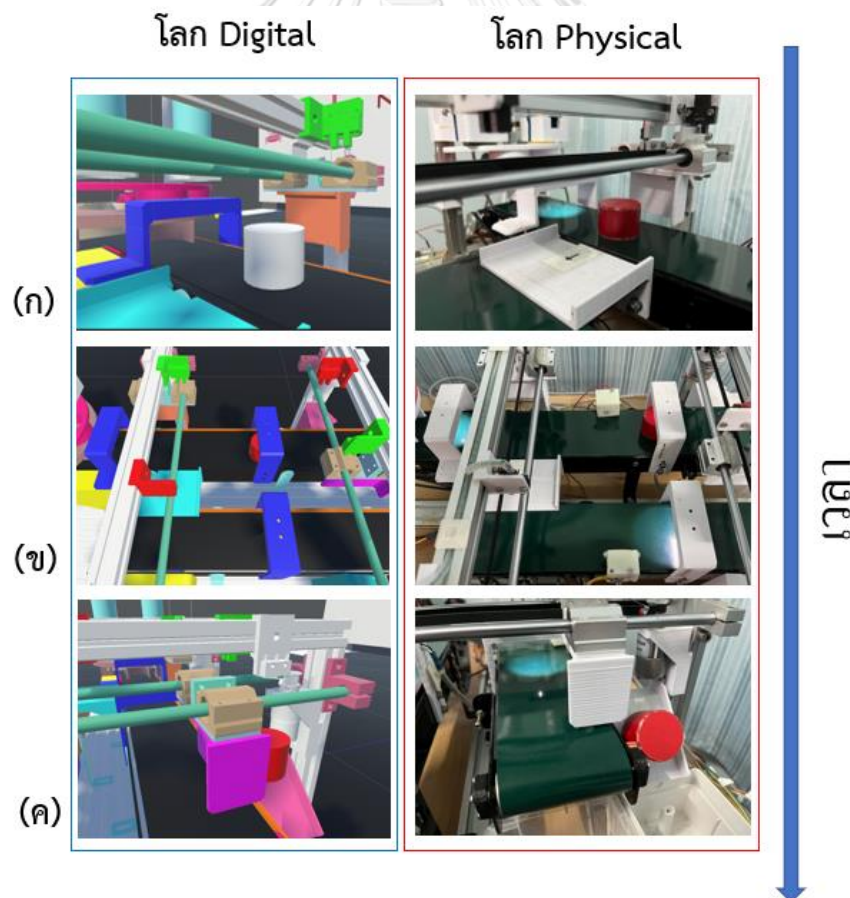
รูป 65 สถิติกระบวนการทำงานของอุปกรณ์คัดแยกวัตถุอัตโนมัติ

#### 4.4.2 กระบวนการทำงานของวัตถุสีแดง

รูป 66.(ก) ต่อ จากกระบวนการทำงานขั้นต้น โดย วัตถุสีแดงเคลื่อนที่ผ่านตัวอุปกรณ์ผลักวัตถุ ที่ 1

รูป 66.(ข) วัตถุสีแดงเคลื่อนที่ผ่านเซนเซอร์ตรวจจับสีของชุดโมดูลที่ 3 ชุดประมวลผลหลัก Node-RED จะทำการยืนยันอัปเดตค่าสถานะสีไปยัง คู่ฝาแฝดดิจิทัล ทำให้สีของวัตถุบน คู่ฝาแฝดดิจิทัลเปลี่ยนสีจากสีขาว เป็นสีแดง ตามข้อมูลที่ได้รับจากชุดอุปกรณ์จริง

รูป 66.(ค) เมื่อได้รับการยืนยันค่าสถานะสีของวัตถุเรียบร้อยแล้ว ชุดประมวลผลหลัก Node-RED จะสั่งการทำงานของชุดผลักอุปกรณ์ของโมดูลที่ 4 ให้ทำงาน เพื่อแยกวัตถุออกตามกลุ่มที่กำหนดไว้ โดย เมื่อ ชุดผลักอุปกรณ์ของโมดูลที่ 4 เริ่มต้นทำงานนั้น ชุดประมวลผลหลัก Node-RED จะทำการส่งข้อมูลสถานะ ให้คู่ฝาแฝดดิจิทัล ทำการแสดงภาพเคลื่อนไหวของ วัตถุและชุดผลักวัตถุให้ตรงกันกับชุดอุปกรณ์จริง



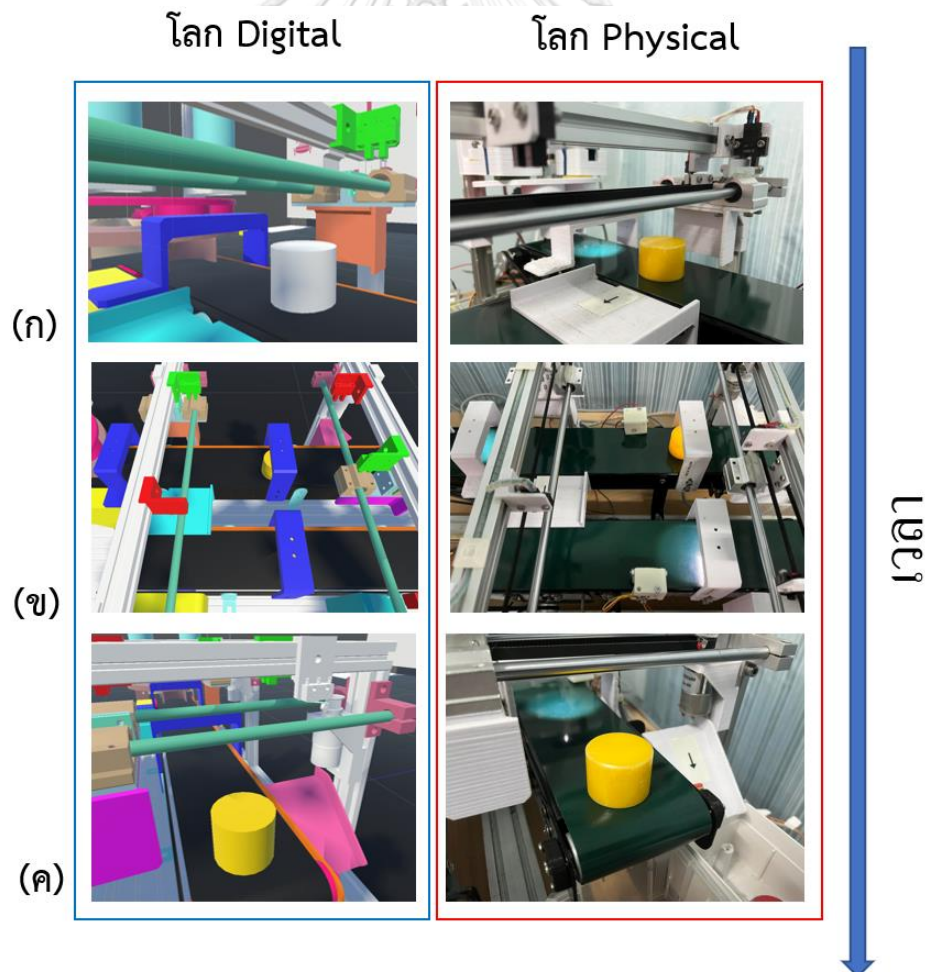
รูป 66 สาธิตกระบวนการทำงานของวัตถุสีแดง

#### 4.4.3 กระบวนการทำงานของวัตถุสีเหลือง

รูป 67.(ก) ต่อ จากกระบวนการทำงานขั้นต้น โดย วัตถุสีเหลืองเคลื่อนที่ผ่านตัวอุปกรณ์ผลักวัตถุ ที่ 1

รูป 67.(ข) วัตถุสีแดงเคลื่อนที่ผ่านเซนเซอร์ตรวจจับสีของชุดโมดูลที่ 3 ชุดประมวลผลหลัก Node-RED จะทำการยืนยันอันดับค่าสถานะสีไปยัง คู่ฝาแฝดดิจิทัล ทำให้สีของวัตถุบน คู่ฝาแฝดดิจิทัลเปลี่ยนสีจากสีขาว เป็นสีเหลือง ตามข้อมูลที่ได้รับจากชุดอุปกรณ์จริง

รูป 67.(ค) เมื่อได้รับการยืนยันค่าสถานะสีของวัตถุเรียบร้อยแล้ว ชุดประมวลผลหลัก Node-RED จะส่งสถานอัปเดตไปยัง ฝาแฝดดิจิทัล เพื่อให้ ฝาแฝดดิจิทัลนั้นแสดงภาพอนิเมชันการเคลื่อนที่ของวัตถุสีเหลือง โดยไม่จำเป็นต้องมีการเคลื่อนที่ของชุดผลักวัตถุ



รูป 67 สาธิตกระบวนการทำงานของวัตถุสีเหลือง

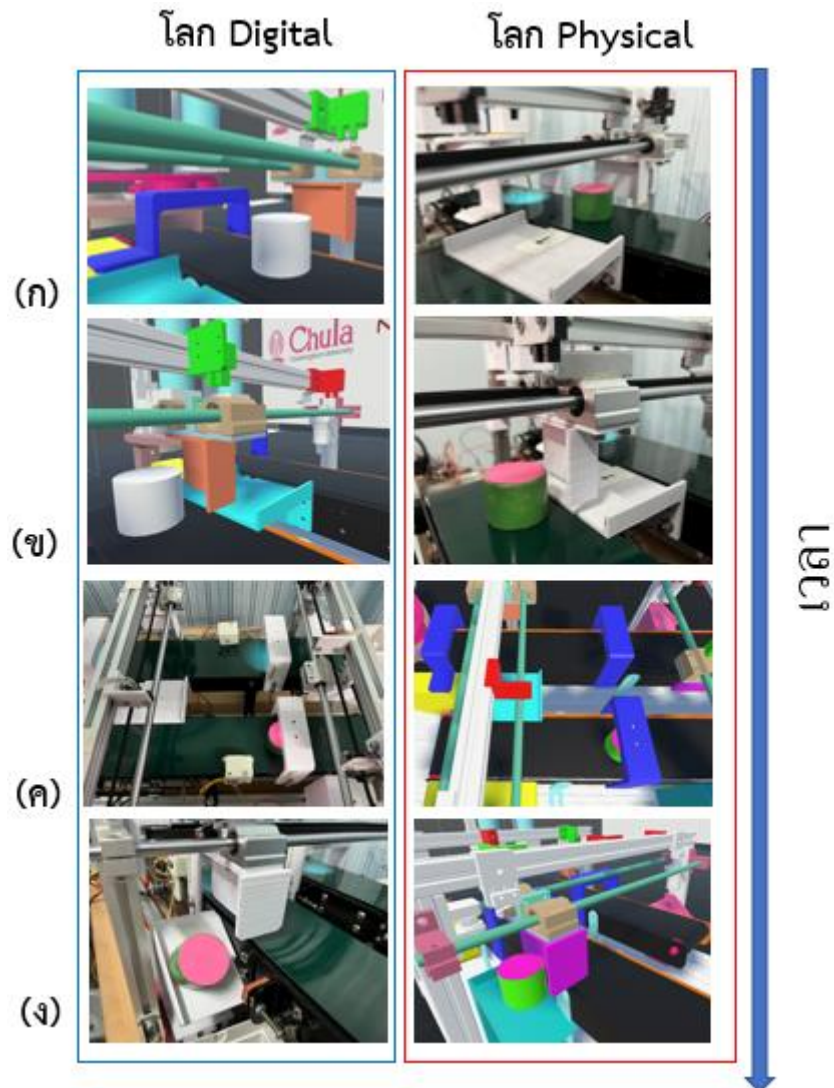
#### 4.4.3 กระบวนการทำงานของวัตถุสีเขียวชมพู

รูป 68.(ก) ต่อ จากกระบวนการทำงานขั้นต้น โดย วัตถุสีเขียวเคลื่อนที่ผ่านตัวอุปกรณ์ผลึกวัตถุ ที่ 1

รูป 68.(ข) ชุดประมวลผลหลัก Node-RED เมื่อได้รับข้อมูลจากเซนเซอร์สีจากโมดูลที่ 3 ในขั้นต้น เมื่อตรวจสอบข้อมูลแล้ว จะทำการส่งงานชุดหลักอุปกรณ์ในโมดูลที่ 4 เพื่อให้ผลึกวัตถุไปยังสายพานที่ 2 พร้อมทั้ง แจ้งสถานะไปยัง ฝาแฝดดิจิทัลให้เพื่อให้เห็นผลภาพอนิเมชันให้ใกล้เคียงกับชุดอุปกรณ์ในเวลาจริง

รูป 68.(ค) วัตถุสีเขียวชมพูเมื่อถูกผลึกมายังสายพานที่ 2 แล้วนั้น วัตถุจะเคลื่อนที่ไปจนกระทั่งเซนเซอร์ตรวจจับวัตถุ ของชุดโมดูลที่ 3 ชุดประมวลผลหลักจะทำการตรวจเช็คสีของวัตถุ และจัดส่งข้อมูลไปยังฝาแฝดดิจิทัลเพื่อให้วัตถุในระบบฝาแฝดดิจิทัลเปลี่ยนสีจากสีขาวเป็นสีเขียว

รูป 68.(ง) เมื่อได้รับการยืนยันค่าสถานะสีของวัตถุเรียบร้อยแล้ว ชุดประมวลผลหลัก Node-RED จะสั่งการทำงานของชุดหลักอุปกรณ์ของโมดูลที่ 4 ให้ทำงาน เพื่อแยกวัตถุออกตามกลุ่มที่กำหนดไว้ โดย เมื่อ ชุดหลักอุปกรณ์ของโมดูลที่ 4 เริ่มต้นทำงานนั้น ชุดประมวลผลหลัก Node-RED จะทำการส่งข้อมูลสถานะ ให้คู่ฝาแฝดดิจิทัล ทำการแสดงผลภาพเคลื่อนไหวของ วัตถุและชุดผลึกวัตถุให้ใกล้เคียงกันกับชุดอุปกรณ์จริง



รูป 68 สถิติกระบวนการทำงานของวัสดุสีเขียวชมพู



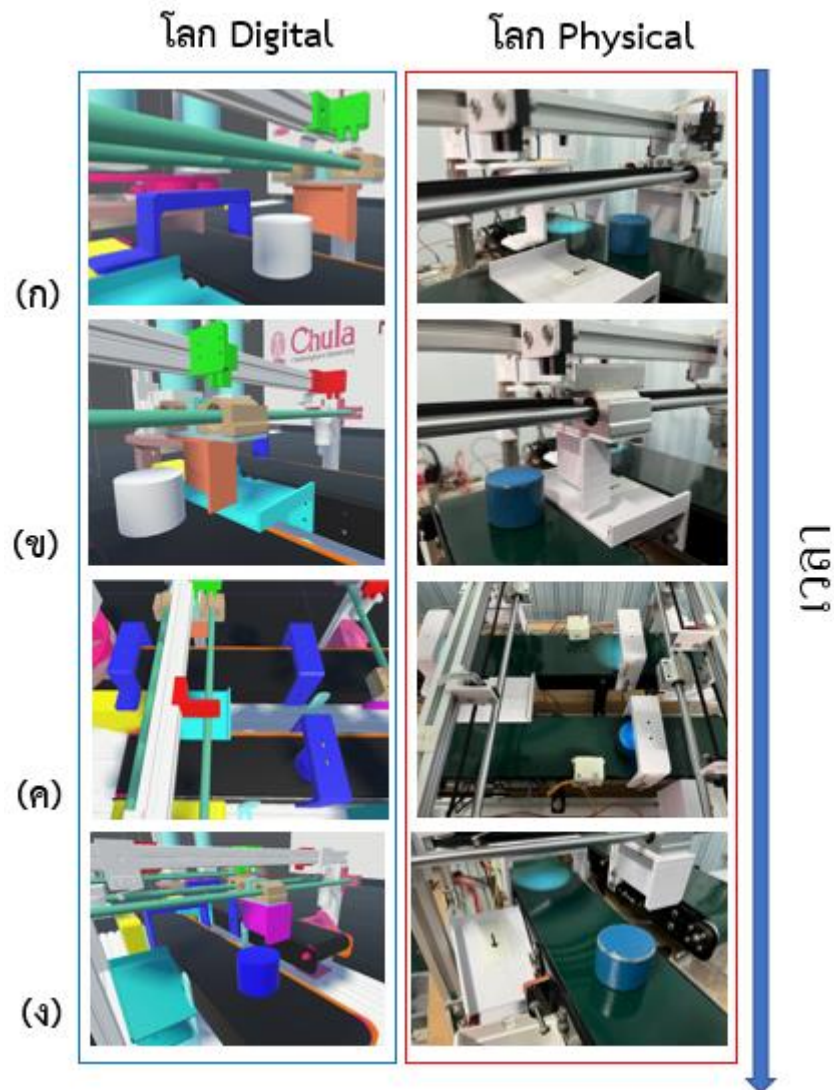
#### 4.4.4 กระบวนการทำงานของวัตถุสีน้ำเงิน

รูป 69.(ก) ต่อ จากกระบวนการทำงานขั้นต้น โดย วัตถุสีน้ำเงินเคลื่อนที่ผ่านตัวอุปกรณ์หลัก วัตถุ ที่ 1

รูป 69.(ข) ชุดประมวลผลหลัก Node-RED เมื่อได้รับข้อมูลจากเซนเซอร์สีจากโมดูลที่ 3 ในขั้นต้น เมื่อตรวจสอบข้อมูลแล้ว จะทำการส่งงานชุดหลักอุปกรณ์โมดูลที่ 4 เพื่อให้ผลวัตถุไปยังสายพานที่ 2 พร้อมทั้ง แจ้งสถานะไปยัง ฝาแฝดดิจิทัลให้เพื่อให้แสดงผลภาพนิเมชันให้ใกล้เคียงกับชุดอุปกรณ์ในเวลาจริง

รูป 69.(ค) วัตถุสีน้ำเงินเมื่อถูกผลกมายังสายพานที่ 2 แล้วนั้น วัตถุจะเคลื่อนที่ไปจนกระทั่งเซนเซอร์ตรวจจับวัตถุ ของชุดโมดูลที่ 3 ชุดประมวลผลหลักจะทำการตรวจเช็คสีของวัตถุและจัดส่งข้อมูลไปยังฝาแฝดดิจิทัลเพื่อให้วัตถุในระบบฝาแฝดดิจิทัลเปลี่ยนสีจากสีขาวเป็นสีน้ำเงิน

รูป 69.(ง) เมื่อได้รับการยืนยันค่าสถานะสีของวัตถุเรียบร้อยแล้ว ชุดประมวลผลหลัก Node-RED จะสั่งการทำงานของชุดหลักอุปกรณ์ของโมดูลที่ 4 ให้ทำงาน เพื่อแยกวัตถุออกตามกลุ่มที่กำหนดไว้ โดย เมื่อ ชุดหลักอุปกรณ์ของโมดูลที่ 4 เริ่มต้นทำงานนั้น ชุดประมวลผลหลัก Node-RED จะทำการส่งข้อมูลสถานะ ให้คู่ฝาแฝดดิจิทัล ทำการแสดงผลภาพเคลื่อนไหวของ วัตถุและชุดหลักวัตถุให้ใกล้เคียงกันกับชุดอุปกรณ์จริง



CHULALONGKORN UNIVERSITY

รูป 69 สาธิตกระบวนการทำงานของวัตถุสีน้ำเงิน

จากกระบวนการการทำงานของชุดฝาแฝดดิจิทัลนั้นทางผู้จัดทำยังคงเห็นถึงข้อจำกัดข้อตัวภาพอนิเมชันสำหรับคู่ฝาแฝดดิจิทัลที่ยังคงไม่สามารถทำให้สอดคล้องกับสภาพวัตถุจริงได้ 100% โดยเฉพาะการเคลื่อนไหวในบางจุดเช่น วัตถุเคลื่อนที่บนสายพาน หรือ วัตถุเคลื่อนที่ขณะโดนอุปกรณ์ผลักวัตถุ ผลัก เนื่องด้วยข้อมูลที่อพเดทสถานะไปยังกระบวนการสร้างอนิเมชัน ยังคงไม่เพียงพอเพื่อให้คู่ฝาแฝดดิจิทัลสามารถแสดงตำแหน่งของวัตถุได้ตรงตามอุปกรณ์ในโลกจริง

## บทที่ 5

### สรุปผล และ ข้อเสนอแนะ

#### 5.1 สรุปผลการดำเนินงาน

วิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์ในการสร้างระบบฝาแฝดดิจิทัลสำหรับชุดอุปกรณ์สาธิตตัดแยกวัตถุอัตโนมัติ และเชื่อมต่อข้อมูลระหว่างชุดอุปกรณ์จริง และ คู่ฝาแฝดดิจิทัลให้สามารถทำงานและแสดงผลควบคู่กัน โดยยังวัดประสิทธิภาพระบบการสื่อสารที่เลือกนำมาใช้ สำหรับชุดอุปกรณ์ตัดแยกวัตถุอัตโนมัติในโลกแห่งความจริงที่รับส่งข้อมูลการทำงานแบบ Wireless LAN ผ่านระบบ UDP /TCP ระหว่าง ชุดประมวลผลหลัก Node-RED ที่ติดตั้งอยู่บนอุปกรณ์คอมพิวเตอร์พกพา Asus TUF Gaming FX504GM และ ชุดบอร์ดคอนโทรลเลอร์ ESP8266 และ การเชื่อมต่อระหว่าง คู่ฝาแฝดดิจิทัล ผ่านระบบการสื่อสาร แบบ MQTT บนแพลตฟอร์มของ NETPIE 2020 โดยสามารถสรุปผลได้ดังนี้

##### 5.1.1. ค่าความหน่วงและเสถียรภาพของการสื่อสารแบบ Wireless LAN ผ่านระบบ UDP/TCP ระหว่าง Node-RED และ ESP8266

การสื่อสารแบบ Wireless LAN ด้วย protocol UDP ระหว่างชุดประมวลผลหลัก Node-RED และ ชุดบอร์ดคอนโทรลเลอร์ ESP8266 นั้น มีค่าความหน่วงที่น้อยกว่า อย่างมีนัยสำคัญ หรือ กล่าวคือ มีความรวดเร็วในการรับส่งข้อมูลที่รวดเร็วกว่า กว่า protocol TCP โดยเฉพาะอย่างยิ่งหากเป็นชุดคำสั่งที่มีขนาดเล็ก เช่น ชุดคำสั่ง เปิด / ปิด อุปกรณ์ จะทำให้ ลักษณะเด่นของการสื่อสารแบบ UDP เห็นผลได้ชัดเจน โดยจากการทดสอบเฉพาะการรับส่งข้อมูล ระหว่างชุดประมวลผลหลัก และ บอร์ดคอนโทรลเลอร์ ESP8266 ด้วยการสื่อสาร แบบ UDP จะมีความรวดเร็วกว่า การสื่อสารแบบ TCP ประมาณ 30ms ต่อชุดคำสั่งหากระบบต้องการการอัปเดตข้อมูลที่รวดเร็วเป็นหลัก การใช้ระบบสื่อสารแบบ UDP จะช่วยตอบโจทย์ในการออกแบบระบบมากกว่าการสื่อสารแบบ TCP แต่หากพิจารณาจากเสถียรภาพที่ได้ทดสอบไป อัตราการส่งข้อมูลสำเร็จของ การสื่อสารแบบ UDP ระหว่างชุดประมวลผลหลักและ บอร์ดคอนโทรลเลอร์ อยู่ที่ประมาณ 97% ซึ่ง การสื่อสารด้วย protocol UDP ยังไม่เหมาะสมสำหรับงานที่ต้องให้ความสำคัญกับทุกข้อมูลที่มีการจัดส่ง แต่เหมาะสมกับระบบที่เน้นความเร็วในการสื่อสารมากกว่า

### 5.1.2 ค่าความหน่วงและเสถียรภาพของการสื่อสารแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020

การสื่อสารแบบ MQTT ผ่านแพลตฟอร์ม NETPIE 2020 นั้น จากผลการทดลองให้ จะเห็นว่าค่าความหน่วงของการรับส่งข้อมูล ในการรับประกันข้อมูล แบบ QoS 0 และ QoS 1 นั้นนั้น มีความรวดเร็วในการรับส่งข้อมูลมาก โดยความหน่วงในการรับส่งข้อมูล นั้นอยู่ในช่วง 40 ms สำหรับการรับส่งข้อมูลไปกลับ ซึ่งสามารถนำไปใช้ในการพัฒนาระบบที่ต้องการความรวดเร็วในการสื่อสารได้ แต่ หากเพิ่มระดับการรับประกันข้อมูลเป็นแบบ QoS 2 นั้น ค่าความหน่วงที่วัดได้จากการทดสอบ จะเพิ่มขึ้นอย่างมีนัยยะสำคัญ โดยเพิ่มค่าความหน่วงอีกประมาณ 60ms ในทุกชุดคำสั่ง โดยเวลาโดยรวมสำหรับการส่งข้อมูลแบบไปกลับ ด้วยรับประกันการส่งข้อมูล QoS 2 ผ่านแพลตฟอร์ม NETPIE 2020 นั้นอยู่ที่ประมาณ 105ms ซึ่งอาจจะไม่เหมาะสำหรับงานที่ต้องการความรวดเร็วในการรับส่งข้อมูล และ จากผลการทดลองเสถียรภาพของ การสื่อสารผ่านระบบ MQTT บนแพลตฟอร์ม NETPIE 2020 เองนั้นยังมีอัตราการส่งข้อมูลที่สำเร็จมากถึง 99.90-100% ในทุกช่วงชุดข้อมูล ถึงแม้ว่าในการทดลองจะใช้การรับประกันการส่งข้อมูล ระดับ QoS 0 ก็ตาม โดยอัตราความสำเร็จในการ รับส่ง ข้อมูล สำหรับ QoS 0 นั้นมากถึง 99.9% และหากเพิ่มระดับการรับประกันการส่งข้อมูล เป็นระดับ QoS 2 จากผลการทดสอบทำให้เห็นว่าสามารถช่วยเพิ่มอัตราการรับส่งข้อมูลสำเร็จถึง 100%

### 5.1.3 ค่าความหน่วงและเสถียรภาพของระบบฝาแฝดดิจิทัล ผ่านการสื่อสารแบบ Wireless LAN UDP /TCP และ MQTT บนแพลตฟอร์มของ NETPIE 2020

ในการสื่อสารแบบไปกลับจาก ฝาแฝดดิจิทัล ไปยัง ชุดบอร์ดคอนโทรลเลอร์ จากผลการทดลองนั้นแสดงให้เห็นถึงค่าความหน่วงที่ต่ำที่สุดคือ การเลือกใช้การสื่อสารระหว่างชุดอุปกรณ์ประมวลผลหลักกับบอร์ดคอนโทรลเลอร์ ESP8266 ผ่าน protocol UDP และ เลือกการรับประกันการรับส่งข้อมูลระดับต่ำสุด QoS 0 สำหรับการสื่อสารแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020 โดยค่าความหน่วงอยู่ที่จากการทดลองอยู่ในช่วง 83.44 ms โดยอนุมาณการส่งข้อมูลแบบฝั่งเดียวจาก คู่ฝาแฝดดิจิทัลไปยัง ชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติอยู่ที่ 41.72 msซึ่งถือว่าตอบโจทย์การสร้างระบบฝาแฝดดิจิทัลที่มีต้องการความรวดเร็วในการรับส่งข้อมูล แต่เป็นที่หน้าสังเกตคือถึงแม้จะทำการ เพิ่มการรับประกันการส่งข้อมูลให้สูงขึ้น ทั้ง เพิ่มระดับ QoS หรือ เปลี่ยน protocol สื่อสาร เป็น TCP จากผลการทดสอบเสถียรภาพของระบบ แสดงให้เห็นว่า ระบบสื่อสารที่ใช้ ยังมีชุดข้อมูลบางส่วนที่จัดส่งชุดข้อมูลไม่สำเร็จโดย หากเลือกใช้ protocol TCP ในการสื่อสารสำหรับชุดอุปกรณ์

ในโลกจริง และ ปรับระดับการรับประกันการส่งข้อมูล เป็น QoS 2 สำหรับการเชื่อมต่อข้อมูลของคู่ ฝาแฝดดิจิทัล จากการทดลอง 4.1 และ 4.2 อัตราความสำเร็จในการส่งข้อมูลไปกลับ ควรจะเป็นที่ 100% แต่จากผลการทดลองนั้น อัตราความสำเร็จอยู่ 99.68% จากการรับส่งข้อมูล 10,000 ครั้งซึ่ง สาเหตุส่วนหนึ่งอาจจะมาจากการตั้งค่าการเชื่อมต่อ Node-RED และ NETPIE 2020 แบบ Clean Session ( Non Persistent Session ) ซึ่งมีโอกาสที่การเชื่อมต่อระหว่างชุดอุปกรณ์และระบบการ สื่อสารขัดข้องและ NETPIE 2020 ไม่ได้ทำการส่งข้อมูลให้ครบตามเงื่อนไขของการรับประกันการส่ง ข้อมูล QoS 1 - 2 ตามที่ได้ตั้งค่าไว้

#### 5.1.4 สาธิตระบบปฏิบัติการของระบบฝาแฝดดิจิทัลสำหรับชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ

ในส่วนของงานสาธิตระบบปฏิบัติการของระบบฝาแฝดดิจิทัลต้องเน้นภาพโดยรวม การทำงานมีความใกล้เคียงกันกับชุดอุปกรณ์จริง ในเวลาจริง ถึงแม้ว่าภาพอนิเมชันการเคลื่อนไหวใน บางส่วนจะยังไม่สอดคล้องกับชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติจริง แบบ 100% โดยเฉพาะอย่างยิ่งตัว วัตถุที่เคลื่อนไหวบนสายพาน เนื่องจากไม่มีเซนเซอร์ หรือ ระบบ ที่สามารถ ระบุตำแหน่งของวัตถุบน สายพานได้อย่างแม่นยำ ภาพที่เห็นผ่านคู่ฝาแฝดดิจิทัลจึงเป็นภาพอนิเมชันที่ถูกสร้างไว้ล่วงหน้าจึงไม่ สามารถแสดงการทำงานให้เหมือนของจริงได้ 100% และ ตัวระบบชุดอุปกรณ์คัดแยกวัตถุอัตโนมัติ เองนั้นยังมีข้อผิดพลาดในการแยกวัตถุให้ถูกต้องโดยข้อผิดพลาดเกิดจากกระบวนการตรวจสอบวัตถุของ โมดูลที่ 3 ที่บางครั้งตัวเซนเซอร์นั้นมีการอ่านค่าสีที่ผิดตำแหน่งของวัตถุทำให้ค่า RGB ที่ได้รับไป วิเคราะห์นั้นผิดเพี้ยนไป และ มีผลให้นำไปประมวลผลส่งงานชุดอุปกรณ์ผิดพลาด

จากผลสรุปที่กล่าวมาวิทยานิพนธ์ฉบับนี้ได้นำเสนอสาธิตกระบวนการพัฒนาระบบการ สื่อสารสำหรับคู่ฝาแฝดดิจิทัลด้วยโปรแกรม Unity ผ่านระบบสื่อสารแบบ Local Area network และ IoT โดยกระบวนการสื่อสารแบบ Local Area Network แบบ UDP นั้น จะเป็นกระบวนการ สื่อสารที่รวดเร็วที่สุด แต่การรับประกันการส่งข้อมูลจะต่ำที่สุดเช่นกัน และ การเชื่อมต่อผ่านระบบ IoT นั้นถึงแม้ความจะมีค่าความหน่วงในการรับส่งข้อมูลมากกว่าการสื่อสารแบบ Local Area Network แต่ก็มีคามยืดหยุ่นในการตั้งค่าการรับประกันการส่งข้อมูล และ การเชื่อมต่อชุดอุปกรณ์ได้ หลากหลายกว่าผ่านแพลตฟอร์ม NETPIE 2020 ซึ่งการนำไปประยุกต์ใช้ควรคำนึงถึงลักษณะงานและ แอปพลิเคชันที่พัฒนาเป็นหลักกว่าผู้ใช้งานเน้นประสิทธิภาพส่วนไหนเป็นสำคัญ

## 5.2 ข้อเสนอแนะเพิ่มเติม

วิทยานิพนธ์ฉบับนี้มีวัตถุประสงค์หลักในการเสนอ และ สาธิต วิธีการพัฒนาระบบฝาแฝดดิจิทัลด้วยเทคโนโลยีโลกเสมือนจริง โดยเน้นพัฒนาระบบสำหรับสั่งงานอุปกรณ์จริงแบบไร้สาย และสามารถรองรับการทำงานแบบ IoT ระบบการสื่อสาร MQTT บนแพลตฟอร์มของ NETPIE 2020 โดยทั้งยังนำเสนอผลทดลองเพื่อหาค่า ความหน่วง และ เสถียรภาพในกระบวนการสื่อสารแบบไร้สายของชุดอุปกรณ์ และ การสื่อสารแบบ MQTT โดยในข้อเสนอแนะเพิ่มเติมจากผู้จัดทำสามารถนำเสนอได้ดังนี้

### 5.2.1 ระบบสื่อสารแบบไร้สายของชุดประมวลผลหลัก Node-RED และ ESP8266

จากผลการทดลองจะเห็นได้ว่าการสื่อสารแบบ UDP นั้นมีค่าความหน่วงที่ต่ำกว่า TCP อย่างมีนัยยะสำคัญสำหรับการจัดส่งชุดข้อมูล ถึงแม้ เสถียรภาพของการสื่อสาร จะไม่การันตีการรับส่งข้อมูลแบบ 100% แต่จากผลการทดสอบ อัตราความสำเร็จในการรับส่งข้อมูล ก็เป็นที่น่าพึงพอใจซึ่งหากนำระบบการสื่อสารแบบ UDP ไปใช้ในลักษณะ การส่งข้อมูลจากฝั่งเดียวจาก บอร์ดคอนโทรลเลอร์ ESP8266 ไปยัง ชุดประมวลผลหลัก Node-RED ในลักษณะ ของการอัปเดต ค่าสถานะต่างๆของระบบเซนเซอร์ หรือ อุปกรณ์ ที่ติดตั้งอยู่บนชุดอุปกรณ์จริง จะทำให้บนชุดประมวลผลหลักมีการอัปเดตค่าใกล้เคียงในลักษณะแบบ Real-Time มากยิ่งขึ้น และ ยังสามารถพัฒนาการส่งข้อมูลแบบ Broadcast ผ่านการสื่อสารแบบ UDP เพื่อทำการเชื่อมต่อข้อมูล ของบอร์ดคอนโทรลเลอร์ในโมดูลต่างๆเพื่อเพิ่มศักยภาพในการทำงานของชุดอุปกรณ์ให้สามารถตรวจสอบข้อมูล ข้ามโมดูลโดยไม่ต้องพึ่งพาชุดประมวลผลหลัก Node-RED เพียงอย่างเดียว

### 5.2.1 ระบบสื่อสารบน แพลตฟอร์มของ NETPIE 2020

สำหรับการใช้งานการสื่อสารบนแพลตฟอร์มแบบ MQTT บนแพลตฟอร์มของ NETPIE 2020 นั้นมีความน่าพึงพอใจในการใช้งานสูงจากผู้จัดทำหากแต่ยังคงขาดกระบวนการเชื่อมต่อ ชุดประมวลผล Node-RED และ NETPIE 2020 แบบ Persistent Session ที่สามารถเพิ่มประสิทธิภาพในกระบวนการรับส่งข้อมูลให้มีประสิทธิภาพมากยิ่งขึ้นโดยบนแพลตฟอร์มของ NETPIE 2020 เองนั้นก็ยังคงสามารถนำข้อมูลที่ได้รับการจัดส่งจาก ชุดประมวลผลหลัก Node-RED และ Unity ส่งผ่านไปยังการสื่อสารแบบอื่นที่รองรับบน NETPIE 2020 อื่นๆได้เช่นกัน เช่น RESTful API , CoAP API โดยหากมีกระวัดประสิทธิภาพในแต่ละ API เพิ่มเติมจะมีประโยชน์สำหรับผู้ใช้งานในการเลือกใช้งานได้ในอนาคต

### 5.2.2 อุปกรณ์คัดแยกวัตถุอัตโนมัติในโลก Digital (VR)

สำหรับคู่มือดิจิทัลที่พัฒนาสำหรับชุดอุปกรณ์โลกเสมือนจริง (VR) นั้น ถึงแม้คู่มือดิจิทัลจะสามารถแสดงผลการทำงานได้ใกล้เคียงกับชุดอุปกรณ์จริง แต่คู่มือดิจิทัลเองนั้นยังไม่มีความสามารถในการนำข้อมูลต่างๆจากชุดอุปกรณ์จริงมาวิเคราะห์และนำมาพัฒนาระบบเพิ่มเติมได้ หากแต่เพียงเป็นคู่มือดิจิทัลที่สามารถแสดงผลการทำงานให้กับผู้ใช้งานได้เห็นภาพรวมของชุดข้อมูลบนชุดอุปกรณ์เท่านั้น โดยหาก พัฒนาระบบที่สามารถ คาดเดาการทำงานในรูปแบบอื่นๆ หรือ จำลอง ปัญหาในระบบที่อาจจะเกิดขึ้น ให้กับผู้ใช้งานได้เห็นในผ่านเทคโนโลยีโลกเสมือนจริงจะช่วยให้ระบบคู่มือดิจิทัลด้วยเทคโนโลยีโลกเสมือนจริงมีประโยชน์มากขึ้นกับงานทางด้านอุตสาหกรรม



## บรรณานุกรม

- [1] ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (เนคเทค). "Digital Twin แบบจำลองเสมือนจริงจากระบบทางกายภาพ,." <https://www.nectec.or.th/news/news-public-document/digital-twin-epub.html> (accessed 2022).
- [2] Ying Lin. "10 Virtual Reality Statistics Every Marketer Should Know In 2022[INFOGRAPHIC]." OBERLO. <https://www.oberlo.com/blog/virtual-reality-statistics#:~:text=This%20marks%20a%2020.9%20percent,to%2060.8%20million%20in%202022.> (accessed 2022).
- [3] N. E. a. C. T. C. (NECTEC). "Netpie 2020 from Makers Nation Toward Smart Nation." <https://netpie.io/> (accessed).
- [4] V. Rajan, N. V. Sobhana, and R. Jayakrishnan, "Machine Fault Diagnostics and Condition Monitoring Using Augmented Reality and IoT," in *2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS)*, 14-15 June 2018 2018, pp. 910-914, doi: 10.1109/ICCONS.2018.8663135.
- [5] F. Arévalo, D. Sunaringtyas, C. Tito, C. Piolo, and A. Schwung, "Interactive Visual Procedure using an extended FMEA and Mixed-Reality," in *2020 IEEE International Conference on Industrial Technology (ICIT)*, 26-28 Feb. 2020 2020, pp. 286-291, doi: 10.1109/ICIT45562.2020.9067296.
- [6] A. A. Simiscuka, T. M. Markande, and G. M. Muntean, "Real-Virtual World Device Synchronization in a Cloud-Enabled Social Virtual Reality IoT Network," *IEEE Access*, vol. 7, pp. 106588-106599, 2019, doi: 10.1109/ACCESS.2019.2933014.
- [7] P. Boonmee, B. Jantarakongkul, and P. Jitngernmadan, "VR Training Environment for Electrical Vehicle Assembly Training in EEC," in *2020 - 5th International Conference on Information Technology (InCIT)*, 21-22 Oct. 2020 2020, pp. 238-242, doi: 10.1109/InCIT50588.2020.9310947.
- [8] D. Keßler, A. Arntz, J. Friedhoff, and S. C. Eimler, "Mill Instructor: Teaching Industrial CNC Procedures Using Virtual Reality," in *2020 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, 14-18 Dec. 2020 2020, pp. 231-234, doi: 10.1109/AIVR50618.2020.00048.



- [9] P. Wu, M. Qi, L. Gao, W. Zou, Q. Miao, and L. I. Liu, "Research on the Virtual Reality Synchronization of Workshop Digital Twin," in *2019 IEEE 8th Joint International Information Technology and Artificial Intelligence Conference (ITAIC)*, 24-26 May 2019 2019, pp. 875-879, doi: 10.1109/ITAIC.2019.8785552.
- [10] Q. Wang, W. Jiao, P. Wang, and Y. Zhang, "Digital Twin for Human-Robot Interactive Welding and Welder Behavior Analysis," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 2, pp. 334-343, 2021, doi: 10.1109/JAS.2020.1003518.
- [11] O. Foundation. "Node-RED Low-code programming for event-driven applications." <https://nodered.org/> (accessed.
- [12] A. C. Inc. "ASUS TUF Gaming FX504." [https://www.asus.com/th/supportonly/asus%20tuf%20gaming%20fx504/helpdesk\\_manual/](https://www.asus.com/th/supportonly/asus%20tuf%20gaming%20fx504/helpdesk_manual/) (accessed.
- [13] T. C. P. C. Limited. "Skyworth GN542VF." [https://www.true.th/support/sitehome\\_internet/3986/product/163?ln=en](https://www.true.th/support/sitehome_internet/3986/product/163?ln=en) (accessed.
- [14] M. Stor. "Comparison of ESP8266 NodeMCU development boards." <https://frightanic.com/iot/comparison-of-esp8266-nodemcu-development-boards/> (accessed.
- [15] Arduino. "Arduino IDE 2.0.2." <https://www.arduino.cc/en/software> (accessed.
- [16] Pololu Robotics & Electronics. <https://www.pololu.com/product/1134> (accessed.
- [17] Omron. "EE-SX47 / SX67." <https://www.omron-ap.co.th/products/family/436/specification.html> (accessed.
- [18] ams OSRAM Group. "TCS3200 Programmable Color Light-to-Frequency Converter." [https://ams.com/documents/20143/36005/TCS3200\\_DS000107\\_3-00.pdf/c53d702f-63aa-eda1-745f-d513aa4f535f](https://ams.com/documents/20143/36005/TCS3200_DS000107_3-00.pdf/c53d702f-63aa-eda1-745f-d513aa4f535f) (accessed.
- [19] Ausmodshop. "Mouse Scroll Wheel Encoder." <https://ausmodshop.com/products/alps-mouse-wheel-encoder> (accessed.
- [20] Components101. "MG996R Servo Motor." <https://components101.com/modules/1293n-motor-driver-module> (accessed.
- [21] Components101. "L298N Motor Driver Module." <https://components101.com/modules/1293n-motor-driver-module> (accessed.
- [22] L. Wuhan Ruidike Electormechanical Equipment Co. "37GA520 DC Motor."

- <https://th.aliexpress.com/i/1005001411290703.html?gatewayAdapt=glo2tha> (accessed.
- [23] L. Rosencrance. "User Datagram Protocol (UDP)." [https://www.techtarget.com/searchnetworking/definition/UDP-User-Datagram-Protocol#:~:text=User%20Datagram%20Protocol%20\(UDP\)%20is,provided%20by%20the%20receiving%20party.](https://www.techtarget.com/searchnetworking/definition/UDP-User-Datagram-Protocol#:~:text=User%20Datagram%20Protocol%20(UDP)%20is,provided%20by%20the%20receiving%20party.) (accessed.
- [24] I. Wikimedia Foundation. "Transmission Control Protocol (TCP)." [https://en.wikipedia.org/wiki/Transmission\\_Control\\_Protocol](https://en.wikipedia.org/wiki/Transmission_Control_Protocol) (accessed.
- [25] Meta. "Introducing Meta: A Social Technology Company." <https://about.fb.com/news/2021/10/facebook-company-is-now-meta/> (accessed.
- [26] META. "META Quest 2 " <https://store.facebook.com/quest/products/quest-2> (accessed.
- [27] HTC, "VIVE Pro 2 Full Kit,," (in English), 2022. [Online]. Available: <https://www.vive.com/us/product/vive-pro2-full-kit/overview/>.
- [28] Valve Corporation. "Valve INDEX VR Kit." <https://store.steampowered.com/valveindex> (accessed.
- [29] L. HP Development Company. "HP Reverb G2 Next-level immersion Hyper-realistic VR." <https://www.hp.com/us-en/vr/reverb-g2-vr-headset.html> (accessed.
- [30] I. Epic Games. "Unreal Engine 5." <https://www.unrealengine.com/en-US/unreal-engine-5> (accessed.
- [31] U. Technologies. "The world's leading platform for real-time content creation." <https://unity.com/> (accessed.
- [32] Unity Technologies. "Oculus Integration SDK." <https://developer.oculus.com/downloads/package/unity-integration/> (accessed.
- [33] Unity Technologies. "XR Interaction Toolkit 2.1.1." <https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.1/manual/index.html> (accessed.
- [34] T. Inc. "SketchUp." <https://www.sketchup.com/> (accessed.
- [35] NETPIE. "คู่มือการใช้งาน NETPIE PLATFORM." <https://docs.netpie.io/mqtt-api.html> (accessed.
- [36] Giovanni Paolo Vigano. "M2MqttUnity." <https://github.com/gpvigano/M2MqttUnity> (accessed.

- [37] WIRESHARK. "Wireshark network protocol analyzer." <https://www.wireshark.org/> (accessed.
- [38] Json.org. "Introducing JSON." <https://www.json.org/json-en.html> (accessed.
- [39] P. S. a. Q. M.-. MQTT. "The HiveMQ Team." <https://www.hivemq.com/blog/mqtt-essentials-part-7-persistent-session-queuing-messages/> (accessed.
- [40] S. Lee, H. Kim, D. k. Hong, and H. Ju, "Correlation analysis of MQTT loss and delay according to QoS level," in *The International Conference on Information Networking 2013 (ICOIN)*, 28-30 Jan. 2013 2013, pp. 714-717, doi: 10.1109/ICOIN.2013.6496715.
- [41] D. Thangavel, X. Ma, A. Valera, H. X. Tan, and C. K. Y. Tan, "Performance evaluation of MQTT and CoAP via a common middleware," in *2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, 21-24 April 2014 2014, pp. 1-6, doi: 10.1109/ISSNIP.2014.6827678.
- [42] I. A. Tsokalo, D. Kuss, I. Kharabet, F. H. P. Fitzek, and M. Reisslein, "Remote Robot Control with Human-in-the-Loop over Long Distances Using Digital Twins," in *2019 IEEE Global Communications Conference (GLOBECOM)*, 9-13 Dec. 2019 2019, pp. 1-6, doi: 10.1109/GLOBECOM38437.2019.9013428.
- [43] Adafruit.io, "Adafruit IO MQTT API." [Online]. Available: <https://io.adafruit.com/api/docs/mqtt.html#adafruit-io-mqtt-api>.



จุฬาลงกรณ์มหาวิทยาลัย  
**CHULALONGKORN UNIVERSITY**

## ประวัติผู้เขียน

ชื่อ-สกุล	นาย กฤษณ์ สุกุลกวินกร
วัน เดือน ปี เกิด	27 กันยายน 2534
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	วิศวกรรมศาสตรบัณฑิต สาขาวิศวกรรมเมคคาทรอนิกส์ มหาวิทยาลัยอัสสัมชัญ กำลังศึกษาต่อที่ วิศวกรรมศาสตรมหาบัณฑิต สาขาวิศวกรรมเครื่องกล จุฬาลงกรณ์มหาวิทยาลัย
ที่อยู่ปัจจุบัน	141/63 ร้านรุ่งเรืองการยาง คูบอน 27 रामอินทรา กม.8 เขตบางเขน แขวง ท่าแร้ง กทม 10220
ผลงานตีพิมพ์	Towards Digital Twin Implementation of an Automatic Sorting Machine