

การออกแบบระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก  
โดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ  
LOW-COST ATTITUDE DETERMINATION AND CONTROL SYSTEM DESIGN  
BY USING CMOS SUN SENSOR FOR CUBESAT



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต  
ภาควิชาวิศวกรรมโทรคมนาคม  
คณะวิศวกรรมศาสตร์  
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง  
ปีการศึกษา 2567

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก  
โดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ  
LOW-COST ATTITUDE DETERMINATION AND CONTROL SYSTEM DESIGN  
BY USING CMOS SUN SENSOR FOR CUBESAT



ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2567

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2567

ภาควิชาวิศวกรรมโทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การออกแบบระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก โดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ

LOW-COST ATTITUDE DETERMINATION AND CONTROL SYSTEM DESIGN BY USING CMOS SUN SENSOR FOR CUBESAT

ผู้จัดทำ

1. นางสาวศุภัญญา ชีระภู่งวงน 64010083
2. นางสาวชिरาภรณ์ เทียนชัย 64010750
3. นางสาวกวิตา แสนใจบาล 64011045



อาจารย์ที่ปรึกษา

( ดร. พิระเมศร์ โชติทวีกิจญาติ )

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



การออกแบบระบบควบคุมตำแหน่ง และทิศทางของ  
ดาวเทียมขนาดเล็ก โดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทาง  
แสงอาทิตย์ต้นทุนต่ำ

LOW-COST ATTITUDE DETERMINATION AND  
CONTROL SYSTEM DESIGN BY USING CMOS SUN  
SENSOR FOR CUBESAT

โดย นางสาวคุณัญญา ชีระภูสงวน 64010083  
นางสาวชिरาภรณ์ เทียนชัย 64010750  
นางสาวจิตา แสนใจบาล 64011045

อาจารย์ที่ปรึกษา ดร.พีระเมศร์ โชติทวีกิจญาดา

#### บทคัดย่อ

ปัจจุบันดาวเทียมจำเป็นต้องใช้เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์เพื่อหาตำแหน่งการวางตัว และทิศทางการหมุนของดาวเทียม โดยการเปรียบเทียบทิศทางของแสงจากดวงอาทิตย์ที่ตกกระทบมาที่เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ซึ่งถูกติดตั้งบนตัวดาวเทียม โครงการนี้มีจุดประสงค์เพื่อศึกษาการทำงานของระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก โดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ และศึกษาการจำลองระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย (TC & TM Subsystem) โดยให้สัญญาณที่ได้จากการตกกระทบของแสงแสดงผลบนคอมพิวเตอร์ เพื่อคำนวณหาตำแหน่งที่เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์จะทำงานได้อย่างมีประสิทธิภาพสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ABSTRACT

Satellites utilize sun sensors to determine their orientation and rotation by comparing the sunlight direction detected by sun sensors mounted on the satellite. This project focuses on the study of an attitude control system for small satellites, employing low-cost CMOS sensors for solar direction detection. Furthermore, the project examines a simulation of the wireless communication subsystem (TC & TM Subsystem) between the small satellite and ground station. The sunlight impact data is displayed on a computer, allowing for calculations that determine the optimal positioning for the sun sensor to achieve maximum efficiency in operation.



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ

	หน้า
กิตติกรรมประกาศ	I
บทคัดย่อ	II
สารบัญ	IV
สารบัญรูป	VII
สารบัญตาราง	XVI
<b>บทที่ 1</b>	
<b>บทนำ</b>	
1.1 ความเป็นมาและความสำคัญของปัญหา	1
1.2 วัตถุประสงค์	1
1.3 ขอบเขตของปริญญาานิพนธ์	1
<b>บทที่ 2</b>	
<b>ทฤษฎีและหลักการที่เกี่ยวข้อง</b>	
2.1 ทฤษฎีเกี่ยวกับเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์กับการใช้งานบนดาวเทียม	2
2.2 ทฤษฎีการตกกระทบของแสงบนเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์	3
2.3 ทฤษฎีเกี่ยวกับการประมวลผลภาพ (IMAGE PROCESSING)	4
2.4 ทฤษฎีเกี่ยวกับการคำนวณมุม AZIMUTH (AZ) และ ELEVATION (EL)	6
2.5 ข้อมูลเกี่ยวกับระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย	7
2.6 ข้อมูลเกี่ยวกับระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย	8
2.7 ข้อมูลระบบการรักษาสมดุลของดาวเทียม (SATELLITE ATTITUDE CONTROL)	9
2.8 ระบบที่ใช้ในการกำหนดและควบคุมทิศทางของดาวเทียม (ATTITUDE DETERMINATION AND CONTROL SYSTEM)	12

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า
<b>บทที่ 3</b>	<b>การออกแบบและการจัดทำปริญญาบัตร</b>
3.1	การออกแบบ 18
3.2	เครื่องมือที่ใช้ในการทดลอง 29
3.3	โปรแกรมที่ใช้ในการทดลอง 43
3.4	การจัดเก็บผลการทดลอง 49
<b>บทที่ 4</b>	<b>ผลการทดลอง</b>
4.1	การทดสอบการใช้งานชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ 51
4.2	การทดสอบถ่ายภาพด้วยชุดทดลอง 52
4.3	ผลการทดสอบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ 54
4.4	ผลการทดสอบทดสอบโดยไม่ปรับค่า THRESHOLD และปรับค่า THRESHOLD 58
4.5	ผลการทดสอบความแม่นยำ (ACCURACY) 90
4.6	การทดสอบการคำนวณมุม 95
4.7	การทดสอบระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย (TC & TM SUBSYSTEM) 96
4.8	การทดสอบการหมุนของดาวเทียมขนาดเล็กด้วยมอเตอร์ 99
4.9	การทดสอบระบบต่าง ๆ ของดาวเทียมขนาดเล็กเมื่อนำระบบมาใช้งานร่วมกัน 104
<b>บทที่ 5</b>	<b>สรุปผลและข้อเสนอแนะ</b>
5.1	สรุปผล 105
5.2	ข้อเสนอแนะ 105
<b>บรรณานุกรม</b>	<b>106</b>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญ (ต่อ)

	หน้า	
ภาคผนวก ก	โค้ดการทดสอบโปรไฟล์ความเข้มแสง	111
ภาคผนวก ข	โค้ดการคำนวณข้อผิดพลาด (ERROR) ในการวัดมุม	117
ภาคผนวก ค	โค้ดการถ่ายภาพ	122
ภาคผนวก ง	โค้ดกล้องสำหรับถ่ายภาพ และคำนวณมุม	126
ภาคผนวก จ	โค้ด LORA ฝั่งส่ง	131
ภาคผนวก ฉ	โค้ด LORA ฝั่งรับ	133
ภาคผนวก ช	โค้ดเซิร์ฟเวอร์ของเว็บไซต์	136
ภาคผนวก ซ	โค้ดสำหรับฝั่งผู้ใช้งานของเว็บไซต์	140

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป

รูปที่	หน้า
2.1 ทฤษฎีเกี่ยวกับการตกกระทบของแสง	3
2.2 ช่วงของแอมพลิจูดระหว่าง 0 (สีดำ) ถึง 255 (สีขาว)	4
2.3 มุมราบ และมุมเงย	7
2.4 ระบบการสื่อสารแบบไร้สายผ่าน LORA	9
2.5 การหมุนรอบแกนของดาวเทียม	11
3.1 บล็อกไดอะแกรมภาพรวมของระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็กโดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ	18
3.2 ผังงานภาพรวมของระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็กโดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ	19
3.3 การออกแบบรับแสงในรูปแบบวงกลม	20
3.4 ฐานข้างล่าง	20
3.5 ฐานข้างบน	21
3.6 พื้นที่สำหรับใช้ติดตั้งกล้อง PI CAMERA	21
3.7 ตัวล้อมอเตอร์ฝั่งซ้าย	21
3.8 ตัวล้อมอเตอร์ฝั่งขวา	22
3.9 หน้ากาก (MASK)	22
3.10 ชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์แบบเสร็จสมบูรณ์	22
3.11 ออกแบบระบบการสื่อสารทางด้านภาคส่ง	23
3.12 ออกแบบระบบการสื่อสารทางด้านภาครับ	23
3.13 เว็บไซต์ในส่วนแสดงค่ามุม AZIMUTH และ ELEVATION	24
3.14 เว็บไซต์ในส่วนเก็บค่าข้อมูลมุม 10 มุมล่าสุดไว้ในบันทึกฐานข้อมูล (DATA LOG)	25
3.15 โครงสร้างภาพรวมของดาวเทียมขนาดเล็ก	25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.16 โครงสร้างสามมิติส่วนบน	26
3.17 โครงสร้างส่วนด้านล่าง	26
3.18 โครงสร้างส่วนที่ติดตั้งเซ็นเซอร์	26
3.19 โครงสร้างส่วนติดมอเตอร์	27
3.20 พิมพ์โมเดลด้วยเครื่องพิมพ์สามมิติ	27
3.21 โครงสร้างภาพรวมดาวเทียมขนาดเล็กฝั่งเซ็นเซอร์	28
3.22 โครงสร้างภาพรวมดาวเทียมขนาดเล็กฝั่งมอเตอร์	28
3.23 รูปบอร์ด RASPBERRY PI 5	31
3.24 รูป PI CAMERA MODULE 1.3	33
3.25 รูปบอร์ด ESP32-S2 MINI	34
3.26 รูป SERVO MOTOR	37
3.27 โมดูล LORA ส่งข้อมูลไร้สายระยะไกล ความถี่ 433 เมกะเฮิรตซ์	38
3.28 บอร์ด ARDUINO UNO	39
3.29 ไมโครคอนโทรลเลอร์ STM32 รุ่นชิป STM32F103C8T6	40
3.30 บอร์ดไมโครคอนโทรลเลอร์ MPU-6050	41
3.31 มอเตอร์ไฟฟ้ากระแสตรง	42
3.32 รูปเส้นพลาสติก FILAMENT PLA	43
3.33 โปรแกรม SHAPR3D	43
3.34 โปรแกรม ULTIMAKER CURA 5.6.0	44
3.35 โปรแกรม ARDUINO IDE	45
3.36 เทอร์มินอลของ RASPBERRY PI 5	46
3.37 โปรแกรม VISUAL STUDIO CODE	47
3.38 โปรแกรม MATLAB	47

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
3.39 โปรแกรม DB BROWSER FOR SQLITE	48
4.1 รูปการทดลอง SERVO MOTOR ร่วมกับเครื่อง POWER SUPPLY โดยควบคุมผ่านโปรแกรม ARDUINO IDE	51
4.2 โปรแกรม ARDUINO IDE แสดงค่ามุมผ่านซีเรียลมอนิเตอร์	52
4.3 การทดลองภาพรวม	53
4.4 การถ่ายภาพด้วยชุดทดลองผ่าน PI CAMERA	53
4.5 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา	54
4.6 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 0 องศา	54
4.7 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 0 องศา	55
4.8 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	55
4.9 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION -15 องศา	56
4.10 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION -15 องศา	56
4.11 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 15 องศา	57
4.12 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 15 องศา	57

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.13 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 15 องศา	58
4.14 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 0 องศา	60
4.15 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 0 องศา	60
4.16 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 0 องศา	61
4.17 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 0 องศา	61
4.18 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 0 องศา	62
4.19 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 0 องศา	62
4.20 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 0 องศา	63
4.21 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 0 องศา	63
4.22 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 0 องศา	64
4.23 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	64
4.24 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	65

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.25 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	65
4.26 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION -15 องศา	66
4.27 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION -15 องศา	66
4.28 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION -15 องศา	67
4.29 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION -15 องศา	67
4.30 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION -15 องศา	68
4.31 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION -15 องศา	68
4.32 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 15 องศา	69
4.33 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 15 องศา	69
4.34 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 15 องศา	70
4.35 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 15 องศา	70

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.36 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 15 องศา	71
4.37 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 15 องศา	71
4.38 การทดสอบโดยไม่ปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 15 องศา	72
4.39 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 15 องศา	72
4.40 การทดสอบโดยไม่ปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 15 องศา	73
4.41 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 0 องศา	73
4.42 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 0 องศา	74
4.43 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 0 องศา	74
4.44 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 0 องศา	75
4.45 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 0 องศา	75
4.46 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 0 องศา	76
4.47 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 0 องศา	76

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.48 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 0 องศา	77
4.49 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 0 องศา	77
4.50 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	78
4.51 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	78
4.52 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	79
4.53 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION -15 องศา	79
4.54 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION -15 องศา	80
4.55 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION -15 องศา	80
4.56 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION -15 องศา	81
4.57 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION -15 องศา	81
4.58 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION -15 องศา	82
4.59 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION -15 องศา	82

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.60 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 15 องศา	83
4.61 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH 0 องศา และมุม ELEVATION 15 องศา	83
4.62 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 15 องศา	84
4.63 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 15 องศา	84
4.64 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม AZIMUTH -20 องศา และมุม ELEVATION 15 องศา	85
4.65 การทดสอบโดยปรับค่า THRESHOLD ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 15 องศา	85
4.66 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 2D อยู่ในมุม AZIMUTH 20 องศา และมุม ELEVATION 15 องศา	86
4.67 การทดสอบโดยปรับค่า THRESHOLD การทำ UV PATTERN 3D อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา	86
4.68 การเปรียบเทียบข้อผิดพลาดของมุม AZIMUTH และ ELEVATION ผลของการใช้ THRESHOLD ต่อความแม่นยำในการวัดทิศทาง	90
4.69 ผลการทดสอบข้อผิดพลาดในการวัดมุม AZIMUTH และ ELEVATION	94
4.70 รูปการทดสอบการคำนวณมุมโดยใช้โปรแกรม VISUAL STUDIO CODE ฝั่งภาคส่ง	95
4.71 รูปการทดสอบการรับมุมโดยใช้โปรแกรม VISUAL STUDIO CODE ฝั่งภาครับ	96
4.72 ทดสอบระบบการสื่อสารทางด้านภาคส่ง	97
4.73 แสดงผลลัพธ์การทดสอบการส่งข้อมูล	97

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญรูป (ต่อ)

รูปที่	หน้า
4.74 ทดสอบระบบการสื่อสารทางด้านภาครับ	98
4.75 แสดงผลลัพธ์การทดสอบการรับข้อมูล	98
4.76 ทดสอบระบบสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดิน แบบไร้สายระหว่างภาคส่ง และภาครับ	99
4.77 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน ROLL (X-AXIS) เมื่อเอียง ดาวเทียมขนาดเล็กไปทางซ้าย	100
4.78 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน Roll (X-axis) เมื่อเอียง ดาวเทียมขนาดเล็กไปทางขวา	101
4.79 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน PITCH (Y-AXIS) เมื่อเอียง ดาวเทียมขนาดเล็กไปทางด้านหน้า	101
4.80 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน PITCH (Y-AXIS) เมื่อเอียง ดาวเทียมขนาดเล็กไปทางด้านหลัง	102
4.81 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน YAW (Z-AXIS) เมื่อดาวเทียม ขนาดเล็กอยู่ในจุดเริ่มต้น	102
4.82 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน YAW (Z-AXIS) เมื่อหมุน ดาวเทียมขนาดเล็กตามเข็มนาฬิกา	103
4.83 มอเตอร์จะเริ่มหมุนเมื่อดาวเทียมไม่อยู่ในจุดสมดุคที่กำหนด	104
4.84 มอเตอร์หมุนช้าลงเมื่อดาวเทียมขนาดเล็กเข้าใกล้จุดสมดุค	104
4.85 ทดสอบการทำงานเมื่อนำทุกระบบไปใช้งานร่วมกัน	105

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญตาราง

ตารางที่	หน้า	
3.1	รายละเอียดขาของ RASPBERRY PI 5	31
3.2	รายละเอียดขาของ ESP32-S2-MINI	34
3.3	การเชื่อมต่อพอร์ตระหว่าง SERVO MOTOR และ ESP32-S2-MINI	37
3.4	รายละเอียดการต่อระหว่าง LORA และ ARDUINO	38
4.1	เปรียบเทียบค่ามุม AZIMUTH และมุม ELEVATION โดยอ้างอิง REFERENCE การไม่ปรับค่า THRESHOLD และการปรับค่า THRESHOLD	87
4.2	การเปรียบเทียบข้อผิดพลาดของมุม AZIMUTH และ ELEVATION ผลของการใช้ THRESHOLD ต่อความแม่นยำในการวัดทิศทาง	91
4.3	ผลการทดสอบข้อผิดพลาดในการวัดมุม AZIMUTH และ ELEVATION	94

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1

## บทนำ

### 1.1 ความเป็นมาและความสำคัญของปัญหา

เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ เป็นหนึ่งในองค์ประกอบพื้นฐานของดาวเทียมที่ใช้ในการกำหนดตำแหน่งของดวงอาทิตย์ เพื่อให้สามารถนำไปใช้ในการบอกตำแหน่งทิศทาง และการควบคุมทิศทางการหมุนของดาวเทียมได้อย่างถูกต้อง ปรินูญานิพนธ์นี้จัดทำขึ้นเพื่อศึกษาเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ การออกแบบระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก โดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ เพื่อพัฒนาการทำงานของระบบ เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ให้มีประสิทธิภาพ ต้นทุนต่ำ และมีความแม่นยำ

### 1.2 วัตถุประสงค์

- 1) เพื่อศึกษา และออกแบบการทำงานของระบบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ โดยใช้ต้นทุนที่ต่ำลง
- 2) เพื่อออกแบบระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก
- 3) เพื่อออกแบบพัฒนาอุปกรณ์เครื่องมือให้มีความแม่นยำ และง่ายต่อการใช้งาน
- 4) เพื่อศึกษาการจำลองระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย (TC & TM Subsystem)

### 1.3 ขอบเขตของปรินูญานิพนธ์

ศึกษาการทำงานของระบบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ ทดสอบ และปรับปรุงประสิทธิภาพของระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก โดยทดลองใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ ติดตั้งชุดทดลอง และทดสอบระบบควบคุมการหมุนของดาวเทียมขนาดเล็ก จากนั้นพัฒนาระบบจัดการคำสั่ง และข้อมูลด้วย Raspberry Pi เพื่อควบคุมการทำงานของดาวเทียมขนาดเล็กให้เกิดประสิทธิภาพสูงสุด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 2

### ทฤษฎีและหลักการที่เกี่ยวข้อง

#### 2.1 ทฤษฎีเกี่ยวกับเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์กับการใช้งานบนดาวเทียม

##### 2.1.1 ระบบควบคุมทิศทางของดาวเทียม (Attitude Control)

เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ สามารถใช้ในระบบควบคุมทิศทางของดาวเทียม (Attitude Control System) เพื่อให้สามารถใช้ควบคุมทิศทาง หรือการวางทิศทางของดาวเทียม ต่อแสงอาทิตย์ โดยระบบประกอบไปด้วย เซ็นเซอร์แอกชูเอเตอร์ (Actuator) ซึ่งเป็นอุปกรณ์ที่สามารถสร้างแรงเพื่อให้ดาวเทียมสามารถหมุนตัวได้ รวมไปถึงสามารถเคลื่อนที่ในทิศทางตามที่ต้องการ และซอฟต์แวร์ระบบควบคุมทิศทางของดาวเทียม นั้นใช้ควบคุมการวางทิศทางที่ติดตั้งบนแกนสามแกนที่สามารถทำให้ดาวเทียมมีทิศทางที่คงที่ต่อโลกในทุกโหมดการทำงานของดาวเทียม เพื่อใช้วัดอัตราการหมุนของดาวเทียม และตำแหน่งวงโคจร

##### 2.1.2 บอกรichtungของแผงโซลาร์เซลล์ (Solar array pointing)

เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ สามารถใช้งานในการบอกรichtungของแผงโซลาร์เซลล์ เพื่อกำหนดการวางทิศทางของระบบเก็บพลังงานแสงอาทิตย์ เช่น แผงโซลาร์เซลล์ในดาวเทียมขนาดเล็ก ให้สามารถรับแสงอาทิตย์ได้อย่างมีประสิทธิภาพสูงสุด และยังสามารถมีบทบาทในกระบวนการกู้คืน (Recovery) ด้วยระบบการทำงานที่ปลอดภัย และการอัปเดตจากยานอวกาศอุปกรณ์หมุนรอบ (Gyroscope) ได้

##### 2.1.3 การอัปเดตข้อมูลของไจโรสโคป (Gyro updating)

เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ สามารถใช้ในการอัปเดตข้อมูลของไจโรสโคปได้ โดยการให้ข้อมูลทิศทางของแสงอาทิตย์ที่เข้ามา ซึ่งไจโรสโคปนั้นมีหน้าที่ในการตรวจจับการเคลื่อนที่เชิงมุมของดาวเทียม ดังนั้นการอัปเดตไจโรสโคปนั้นมีความสำคัญอย่างยิ่งในการรักษาความแม่นยำของระบบนำทาง หรือควบคุมดาวเทียม [1]

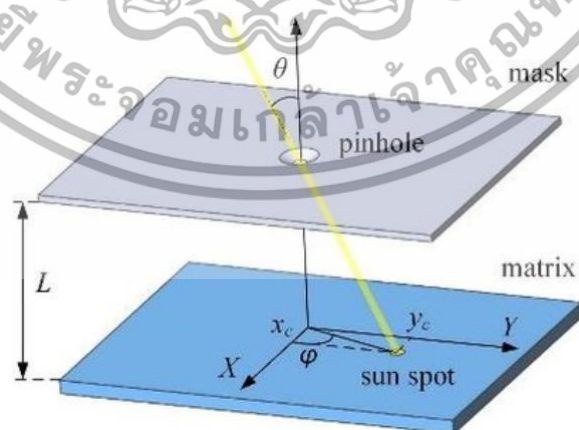
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.1.4 การกู้คืนอย่างปลอดภัยเมื่อมีความผิดพลาด (Fail-safe recovery)

เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ สามารถใช้งานเป็นหนึ่งในเซ็นเซอร์ของระบบการกู้คืนอย่างปลอดภัยบนดาวเทียมซึ่งเป็นระบบที่ใช้ในการสำรองข้อมูลออกมาเพื่อป้องกันหรืออนุญาตให้มีการกู้คืนเมื่อมีความผิดพลาดของระบบหลักเกิดขึ้นหากระบบหลักนั้นเกิดข้อผิดพลาดขึ้นในทางใดทางหนึ่งจะทำให้สามารถป้องกันไม่ให้ระบบการทำงานล่มได้ [2]

## 2.2 ทฤษฎีการตกกระทบของแสงบนเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์

เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ โดยพื้นฐานจะมีการบรรจุเซ็นเซอร์รูปภาพ (Image Sensor) ในตำแหน่งที่ตั้งอยู่ระยะห่าง  $L$  ที่เรียกว่า ระยะโฟกัส จากพิกัดของมาส์คเซ็นเซอร์ (Mask) มาส์คเซ็นเซอร์ประกอบด้วยรูรับแสง (pinhole) หรือรูหลายช่องที่ช่วยควบคุมแสงอาทิตย์ที่ตกกระทบลงบนมาส์คเซ็นเซอร์ไปสู่อาร์เรย์ของเซ็นเซอร์รูปภาพ เซ็นเซอร์รูปภาพประกอบด้วยพื้นที่เซลล์ที่เรียกว่าพิกเซล (Pixel) ส่วนที่ได้รับแสงจากดวงอาทิตย์ในพื้นที่พิกเซลนั้นทำงานอยู่บนอาร์เรย์ของเซ็นเซอร์รูปภาพพื้นที่ที่ได้รับแสงจากดวงอาทิตย์สามารถขยายออกไปได้ตั้งแต่หนึ่งพิกเซลจนถึงหลายพิกเซลซึ่งมีความสัมพันธ์ตรงกับขนาดของรูรับแสง พิกัดของจุดศูนย์กลางของพื้นที่ที่ได้รับแสงตกกระทบจากดวงอาทิตย์สามารถคำนวณได้ด้วยอัลกอริทึมที่เหมาะสมมุมราบ (Azimuth) และมุมเงย (Elevation) จึงถูกกำหนดตามแกนอ้างอิงของระบบพิกัดบนแกนอ้างอิง  $X$ ,  $Y$  และ  $Z$  การตกกระทบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ และพื้นที่เซลล์ส่วนที่ได้รับแสงจากดวงอาทิตย์ได้ [3] โดยทฤษฎีเกี่ยวกับการตกกระทบของแสงดังรูปที่ 2.1 [4]



รูปที่ 2.1 ทฤษฎีเกี่ยวกับการตกกระทบของแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.3 ทฤษฎีเกี่ยวกับการประมวลผลภาพ (Image Processing)

การประมวลผลภาพนั้นเป็นกระบวนการสำคัญที่ใช้ในการจัดการ และวิเคราะห์รูปภาพให้เป็นข้อมูลในรูปแบบดิจิทัล โดยใช้อัลกอริทึมคอมพิวเตอร์เพื่อปรับปรุงรูปภาพให้ดีขึ้น และแก้ไขปัญหาต่าง ๆ ที่เกิดขึ้นได้ ไม่ว่าจะเป็นการกำจัดสัญญาณรบกวนออกจากภาพ การเพิ่มความคมชัด การกำจัดความเบลอ การลบจุด หรือส่วนที่ไม่ต้องการออก การฟื้นฟูภาพในกรณีที่มีความเสียหายเกิดขึ้นกับภาพ การค้นหาวัตถุที่สนใจ การแบ่งส่วนภาพ การลดขนาดไฟล์ภาพ รวมถึงการกำจัดความผิดปกติของแสงบนภาพ ให้สามารถนำข้อมูลไปประมวลผล และวิเคราะห์ได้ เช่น การตรวจจับวัตถุ การจดจำใบหน้า และภาพถ่ายทางอากาศ เป็นต้น [5]

โดยในรูปภาพแบบดิจิทัลนั้น ประกอบไปด้วยช่องพิกเซลหลายพิกเซลที่เรียงกันเป็นรูปภาพ ซึ่งค่าความสว่างของพิกเซลจะอยู่ในช่วงค่าระหว่าง 0 ถึง 255 สำหรับภาพระดับสีเทา (Grayscale Image) 8 บิต แต่ละพิกเซลจะมีช่วงของแอดสีอยู่ระหว่าง 0 (สีดำ) ถึง 255 (สีขาว) ซึ่งระดับสีเทา 256 ระดับนั้นได้ดังรูปที่ 2.2 [6]



รูปที่ 2.2 ช่วงของแอดสีอยู่ระหว่าง 0 (สีดำ) ถึง 255 (สีขาว)

### 2.3.1 การคำนวณความเข้มแสงตามแนวแกน X และแนวแกน Y

การคำนวณความเข้มแสงบนรูปภาพแบบดิจิทัลนั้น ในแต่ละช่องของพิกเซลอ้างอิงถึงพิกัดบนระนาบแนวแกน X และแนวแกน Y ซึ่งค่าความสว่างของพิกเซลนั้นสามารถนำไปคำนวณค่าความเข้มแสงได้ โดยหลักการคำนวณจากพิกเซลที่มีความเข้มแสงในแนวแกน X ซึ่งเป็นค่าเฉลี่ยความเข้มแสงของพิกเซลในแต่ละคอลัมน์บนภาพ และในแนวแกน Y ซึ่งเป็นค่าเฉลี่ยความเข้มแสงของพิกเซลในแต่ละแถวบนภาพ [7] โดยมีสูตรการคำนวณความเข้มแสงเฉลี่ยในแนวแกน X ดังสมการที่ 2.1 การคำนวณความเข้มแสงในแนวแกน Y ดังสมการที่ 2.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$xProfile(i) = \frac{1}{M} \sum_{j=1}^M I(j,i) \quad (2.1)$$

โดย  $xProfile(i)$  คือ ความเข้มแสงเฉลี่ยในแนวแกน X ซึ่งเป็นค่าเฉลี่ยความเข้มแสงของพิกเซลของคอลัมน์  $i$

$I(j,i)$  คือ ความเข้มแสงของพิกเซลที่อยู่ในตำแหน่งแถว  $j$  และคอลัมน์  $i$

$M$  คือ จำนวนแถวของภาพ (จำนวนพิกเซลในแนวแกน Y)

$$yProfile(j) = \frac{1}{N} \sum_{i=1}^N I(j,i) \quad (2.2)$$

โดย  $yProfile(j)$  คือ ความเข้มแสงเฉลี่ยในแนวแกน Y ซึ่งเป็นค่าเฉลี่ยความเข้มแสงของพิกเซลของแถว  $j$

$I(j,i)$  คือ ความเข้มแสงของพิกเซลที่อยู่ในตำแหน่งแถว  $j$  และคอลัมน์  $i$

$N$  คือ จำนวนคอลัมน์ของภาพ (จำนวนพิกเซลในแนวแกน X)

### 2.3.2 การคำนวณตำแหน่งเซนทรอยด์ (Centroid)

การคำนวณเซนทรอยด์ (Centroid) หรือจุดศูนย์กลางเรขาคณิตของพื้นที่วัตถุ มีความสำคัญในการวิเคราะห์ตำแหน่งของวัตถุบนภาพ หลักการคำนวณเซนทรอยด์บนภาพนั้น พิจารณาได้จากค่าการกระจายตัวของพิกเซลในพื้นที่ในภาพ [8] ค่าความเข้มของพิกเซลในแนวแกน X และในแนวแกน Y โดยมีสูตรการคำนวณตำแหน่งเซนทรอยด์ในแนวแกน X ดังสมการที่ 2.3 การคำนวณตำแหน่งเซนทรอยด์ในแนวแกน Y ดังสมการที่ 2.4

$$C_x = \left( \frac{\sum X_i}{N} \right) \quad (2.3)$$

โดย  $C_x$  คือ พิกัดของจุด Centroid ในแนวแกน X

$X_i$  คือ ตำแหน่งของพิกเซลของวัตถุบนภาพ

$N$  คือ จำนวนของพิกเซลทั้งหมดในภาพ

$$C_y = \left( \frac{\sum Y_i}{N} \right) \quad (2.4)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย  $C_y$  คือ พิกัดของจุด Centroid ในแนวแกน Y

$Y_i$  คือ ตำแหน่งของพิกเซลของวัตถุบนภาพ

$N$  คือ จำนวนของพิกเซลทั้งหมดในภาพ

### 2.3.3 การทำ Gaussian Blur

Gaussian Blur คือการใช้ Gaussian function ในการเบลอภาพ เทคนิคนี้จะใช้การคำนวณค่าเฉลี่ยน้ำหนัก (Weighted Average) ของพิกเซลในบริเวณใกล้เคียงของพิกเซลเป้าหมาย พิกเซลที่อยู่ใกล้เคียงจะมีอิทธิพลมากกว่าพิกเซลที่อยู่ไกลออกไป การกระจาย Gaussian จะมีลักษณะเป็นระฆังคว่ำ ซึ่งบ่งบอกถึงการลดอิทธิพลของพิกเซลตามระยะทาง วิธีการนี้ช่วยลดสัญญาณรบกวน และรายละเอียดที่ไม่สำคัญในภาพออกไป Gaussian Blur เป็นวิธีที่ง่ายและประสิทธิภาพสูงในการลดสัญญาณรบกวน ทำให้ภาพมีความชัดเจน ข้อควรระวังในการใช้คือขนาดของ Kernel ที่ใหญ่เกินไปอาจทำให้ภาพเสียรายละเอียดที่สำคัญ และการเบลอภาพมากเกินไปอาจจะทำให้ภาพดูไม่คมชัด ทางผู้จัดเลือกใช้ Gaussian Blur ในการช่วยลดสัญญาณรบกวนในภาพ ทำให้ขั้นตอน Threshold และการตรวจจับขอบเขตมีความแม่นยำมากขึ้น ในโค้ดได้กำหนดขนาดของพื้นที่รอบพิกเซลที่ต้องการเบลอ โดยขนาดของ Kernel คือ 15x15 พิกเซล [9]

## 2.4 ทฤษฎีมุมราบ และมุมเงย

### 2.4.1 มุมราบ

มุมแอสซิมาท (Azimuth) คือ มุมราบที่อยู่ระหว่าง 0 ถึง 360 องศา เป็นมุมแสดงตำแหน่งของวัตถุท้องฟ้าวัดจากแนวทิศเหนือไปตามแนวขอบฟ้าทางทิศตะวันออก หรือตามเข็มนาฬิกาจนถึงตำแหน่งจุดตัดของเส้นที่ลากจากวัตถุนั้นลงมาตัดและตั้งฉากกับเส้นขอบฟ้า

มุมแบริง (Bearing) คือ มุมราบที่อยู่ระหว่าง 0 ถึง 90 องศา เป็นมุมในแนวราบที่วัดจากทิศเหนือ หรือทิศใต้ไปทางทิศตะวันตก หรือตะวันออกถูกกำหนดโดยทิศทางหลักค่าของมุมวัดได้ทั้งตาม และทวนเข็มนาฬิกา [10]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 2.4.1 มุมเงย

มุมเงย (Elevation) คือ มุมเงยที่วัตถุทำกับขอบฟ้า วัตถุที่ดูเหมือนตะขบฟ้าจะมีความสูง 0 องศา ในขณะที่วัตถุที่อยู่ตรงเหนือเราจะมีมุมสูง 90 องศา วัตถุใด ๆ ที่อยู่ต่ำกว่าขอบฟ้าจะมีมุมลบ โดยที่ -90 องศา แสดงถึงตำแหน่งที่ตรงลงมา ในระบบพิกัดท้องฟ้า ตำแหน่งที่ตรงเหนือเราเรียกว่ามุมเซนิท (Zenith) ในขณะที่จุดที่อยู่ใต้เราพอดีเรียกว่ามุมแนเดียร์ (Nadir) [11] มุมราบ และมุมเงยดังรูปที่ 2.3 [12]



รูปที่ 2.3 มุมราบ และมุมเงย

### 2.5 ทฤษฎีเกี่ยวกับการคำนวณมุม Azimuth (Az) และ Elevation (EL)

การคำนวณมุม Azimuth (Az) และ Elevation (EL) โดยใช้ข้อมูลตำแหน่งจุดศูนย์กลางของแสง (centroid) บนภาพ และฟิลด์ออฟวิว (Field of View, FOV) ของกล้องในการเปรียบเทียบตำแหน่งบนเฟรมภาพ เป็นหลักการที่ใช้ในงานวิศวกรรมคอมพิวเตอร์กราฟิก และการวิเคราะห์ภาพเพื่อแปลงพิกัด 2D ของภาพไปเป็นมุมมอง 3D โดยมีสูตรการคำนวณดังนี้

มุม Azimuth (Az) คำนวณจากความต่างระหว่างจุดศูนย์กลางแนวนอนของภาพ และตำแหน่ง X ของ centroid ที่ตรวจจับได้ แล้วเทียบกับมุมมองแนวนอนของกล้อง (FOV\_H) ดังสมการที่ 2.5

$$Az = \left( \frac{(X - X_{center})}{width} \right) \times F_{horizontal} \quad (2.5)$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดย  $X$  คือ พิกัดในแนวแกน X ของ Centroid

$X_{center}$  คือ จุดศูนย์กลางของเฟรมในแนวแกน X

$width$  คือ ความกว้างของเฟรม กำหนดให้เท่ากับ 640

$F_{horizontal}$  คือ มุมมองแนวนอนของกล้อง สำหรับกล้อง Raspberry Pi Camera 1.3 จะมีค่ามุมประมาณ 53.5 องศา

มุม Elevation (EL) คำนวณจากความต่างระหว่างจุดศูนย์กลางแนวตั้งของภาพ และตำแหน่ง Y ของ centroid ที่ตรวจจับได้ แล้วเทียบกับมุมมองแนวตั้งของกล้อง (FOV\_V) ดังสมการที่ 2.6

$$EL = \left( \frac{(Y - Y_{center})}{height} \right) \times F_{horizontal} \quad (2.6)$$

โดย Y คือ พิกัดในแนวแกน Y ของ Centroid

$Y_{center}$  คือ จุดศูนย์กลางของเฟรมในแนวแกน Y

$height$  คือ ความสูงของเฟรม กำหนดให้เท่ากับ 480

$F_{vertical}$  คือ มุมมองแนวตั้งของกล้อง สำหรับกล้อง Raspberry Pi Camera 1.3 จะมีค่ามุมประมาณ 41.41 องศา

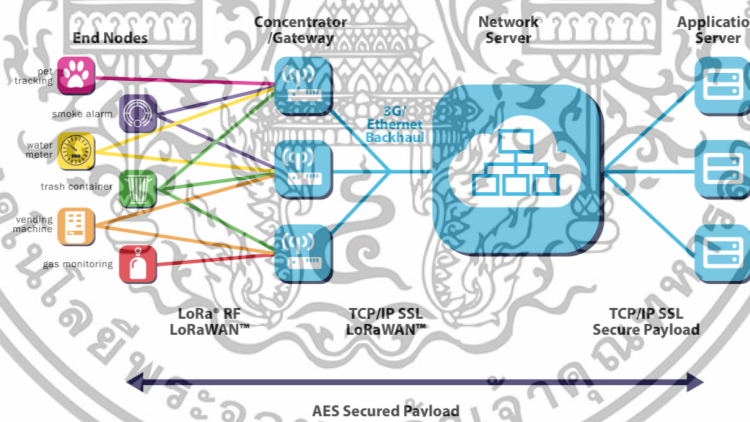
## 2.6 ข้อมูลเกี่ยวกับระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย

ปฏิญญาพนธ์นี้ใช้ระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สายผ่าน LoRa ย่อมาจาก Long Range Radio เป็นโปรโตคอลไร้สายรูปแบบใหม่ที่ออกแบบมาอย่างแม่นยำสำหรับการเชื่อมต่อระยะไกล และการสื่อสารที่ใช้พลังงานต่ำ โดยการใช้เซ็นเซอร์ และแอคทูเอเตอร์สำหรับการใช้งาน M2M และ Internet of Things ระบบจะใช้สเปกตรัมวิทยุในช่วงคลื่นไม่ลงทะเบียนทางอุตสาหกรรม วิทยาศาสตร์ และการแพทย์ (ISM) เพื่อทำการสื่อสารในบริเวณกว้างที่พลังงานต่ำระหว่างเซ็นเซอร์ระยะไกล และเกตเวย์ที่เชื่อมต่อกับเครือข่ายแสดงระบบการสื่อสารแบบไร้สายผ่าน LoRa ดังรูปที่ 2.4 [13]

โดยใช้เทคโนโลยีกระจายสเปกตรัมที่มีช่วงคลื่นกว้างกว่า สัญญาณมอดูเลตความถี่จะใช้แกนเข้ารหัสเพื่อเพิ่มความไวของตัวรับเทคโนโลยี LoRa จึงเหมาะสำหรับงาน Internet of Thing (IoT) ที่ข้อมูลหลัก ๆ มาจากเซ็นเซอร์รับข้อมูลต่าง ๆ โดยเป็นเทคโนโลยีที่มีการใช้งาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แพร่หลาย บริษัทชั้นนำผู้พัฒนาไมโครคอนโทรลเลอร์พัฒนาอุปกรณ์รองรับการเชื่อมต่อ LoRa โดยมีจุดเด่นในเรื่องระยะทางในการสื่อสาร สามารถรับ และส่งข้อมูลได้ในระยะไกล 15-20 กิโลเมตร สามารถเชื่อมต่อเข้ากับโหนดนับล้าน มีอายุการใช้งานแบตเตอรี่ยาวนานกว่า 10 ปี อีกทั้งประหยัดพลังงานจากการใช้พลังงานในช่วงรับ-ส่งข้อมูลค่อนข้างต่ำ ในช่วงที่ไม่ส่งข้อมูลสามารถกำหนดให้อุปกรณ์เข้าสู่โหมดประหยัดพลังงานได้ ทำให้ประหยัดค่าใช้จ่าย เทคโนโลยีนี้จะช่วยให้เครือข่ายหลายผู้เช่า หรือเครือข่ายสาธารณะเชื่อมต่อแอปพลิเคชันจำนวนมากที่ทำงานบนเครือข่ายเดียวกันได้ นอกจากนี้เครือข่ายของ LoRa ยังสามารถให้การครอบคลุมในแบบเดียวกับเครือข่ายมือถือได้ ซึ่งผู้ให้บริการ LoRa จะเป็นผู้ให้บริการเครือข่ายมือถือที่สามารถใช้เสาสัญญาณที่มีอยู่เพื่อติดตั้งเสาอากาศ LoRa ได้ ในบางครั้งเสาอากาศ LoRa อาจผสมรวมกับเสาอากาศเครือข่ายมือถือได้เนื่องจากมีความถี่ใกล้เคียงกัน การรวมเสาอากาศทั้งสองเข้าด้วยกันจะช่วยลดต้นทุนได้อย่างมาก LoRa นั้นเป็นสัญญาณที่มีให้เลือกหลายความถี่ต้องเลือกให้เหมาะสมกับการใช้งาน และถูกกฎหมาย ทางผู้จัดทำได้เลือกใช้โมดูลความถี่ 433 เมกะเฮิร์ตซ์ ซึ่งเป็นความถี่ที่ กสทช. ปล่อยให้ใช้โดยไม่ต้องขออนุญาต หากมีกำลังส่งไม่เกิน 15 มิลลิวัตต์ [14]



รูปที่ 2.4 ระบบการสื่อสารแบบไร้สายผ่าน LoRa

## 2.7 ข้อมูลระบบการรักษาสมดุลของดาวเทียม (Satellite Attitude Control)

การรักษาสมดุลของดาวเทียมหมายถึงการควบคุมทิศทาง และการหมุนของดาวเทียมในอวกาศ เพื่อให้สามารถทำงานได้อย่างแม่นยำ โดยไม่ให้เกิดการหมุนที่ไม่ต้องการ ซึ่งอาจทำให้ระบบต่าง ๆ ทำงานผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

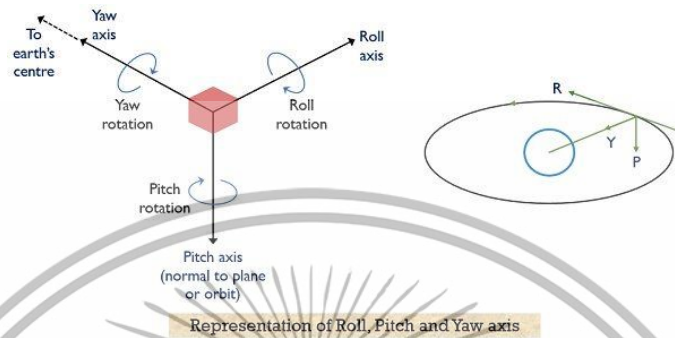
### 2.7.1 ระบบการรักษาสมดุลของดาวเทียม (Satellite Attitude Control)

เป็นระบบคำนวณ และควบคุมการหมุนการทรงตัวรอบแกนของดาวเทียม ในที่นี้แกนดาวเทียมคือ X, Y และ Z สำหรับการทรงตัว ส่วนการหมุนรอบแกนนั้นจะเป็น Yaw, Pitch และ Roll ตามลำดับ ดังรูปที่ 2.5 [15] สิ่งที่จะทำให้ระบบย่อยนี้ทำงานให้บรรลุวัตถุประสงค์ได้อย่างสมบูรณ์ตามภารกิจที่ได้รับการออกแบบจะต้องประกอบด้วยส่วนสำคัญสามส่วนคือ Navigation, guidance และ Control โดยในส่วนของ Navigation จะบอกให้เราทราบว่าตอนนี้อยู่ที่ตำแหน่ง และความเร็วของดาวเทียมในวงโคจรเท่าไร, หมุนรอบตัวเองในอัตรากี่องศาต่อวินาทีของแต่ละแกน, ทรงตัวที่กึ่งศาเทียบแกนอ้างอิงของแต่ละแกนด้วยเซ็นเซอร์ชนิดต่าง ๆ เช่น Sun sensor/Star-tracker, Accelerometer/Gyro meter และ GNSS (หรือ GPS) โดยผ่านกระบวนการ Orbit and Attitude Estimation เช่น Extended Kalman filter (EKF), unscented Kalman filter (UKF) และ unscented particle filter (UPF) โดย Guidance นั้นจะรับข้อมูลจากระบบ Navigation เช่น ตำแหน่ง และความเร็วของดาวเทียมบนวงโคจรที่ได้จาก GNSS/Star-tracker ค่าความเร็วในการหมุนรอบแกนเชิงมุม และการทรงตัวในแนวแกนจาก Accelerometer/Gyro meter โดยเพิ่มความแม่นยำจาก Sun sensor และ Star-tracker เพื่อคำนวณ และสร้างสัญญาณอ้างอิงต่าง ๆ ร่วมกับคำสั่งจากภาคพื้นดิน ส่งให้กับระบบควบคุมเพื่อควบคุมกลไกตัวขับเคลื่อนต่าง ๆ ของดาวเทียม

Control system คือส่วนระบบควบคุม จะประกอบด้วยส่วนของอัลกอริทึมในซอฟต์แวร์ (เฟิร์มแวร์) และฮาร์ดแวร์ตัวขับเคลื่อน ชนิดต่าง ๆ (Actuators) ในส่วนของอัลกอริทึมซอฟต์แวร์ที่สำคัญคือ ส่วนของตัวควบคุมระบบ เช่น ตัวควบคุมแบบ PID, PD, Phase lead, PD-based sliding mode control, MPC, LQR, LQG, robust control  $H_\infty$ , Fractional PID หรือ Fractional robust control : CRONE เป็นต้น การเลือกใช้ขึ้นกับประสบการณ์ของผู้ออกแบบพัฒนา ความยากง่ายในการ Implement ความเสถียร และสมรรถนะที่ต้องการ (Gain/Phase margin) นอกจากนี้ยังต้องพิจารณาถึงความยากง่ายในเฟส Orbit Tuning ซึ่งเป็นเฟสของการปรับแต่งระบบ บนวงโคจรหลังนำส่งดาวเทียมขึ้นสู่อวกาศให้พร้อมก่อนปฏิบัติการกิจที่ได้รับมอบหมายจริงจากสถานีควบคุมภาคพื้นดิน ส่วนตัวขับเคลื่อน (Actuators) ใช้เพื่อสร้างแรงบิดในการหมุนดาวเทียมหรือเพื่อต้านแรงบิดที่กระทำต่อดาวเทียมจากภายนอก เช่น Air drag, solar Radiation pressure, Earth magnetic หรือ Earth, moon gravity นั้น โดยทั่วไป จะใช้ Magnetorquer (Magnetic-torque), Reaction wheel หรือ Control Momentum Gyroscope

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(CMG) โดยแต่ละแบบจะให้แรงบิดมากน้อยต่างกัน การใช้งานขึ้นอยู่กับความต้องการของระบบดาวเทียม



รูปที่ 2.5 การหมุนรอบแกนของดาวเทียม

### 2.7.2 ความสำคัญของการควบคุมดาวเทียม

ความสำคัญของการควบคุมดาวเทียมเนื่องจากในอวกาศไม่มีแรงเสียดทานเมื่อดาวเทียมถูกส่งขึ้นไป อาจหมุนไปเรื่อย ๆ ตามแรงที่ได้รับ ดังนั้นจึงต้องมีระบบควบคุมเพื่อให้ดาวเทียมสามารถใช้งานได้ดังนี้

- 1) ชี้ไปยังตำแหน่งที่กำหนด เช่น ชักล่องถ่ายภาพโลกหรือสังเกตการณ์ดาวฤกษ์
- 2) ควบคุมแผงโซลาร์เซลล์ให้รับแสงอาทิตย์สูงสุด เพื่อผลิตพลังงาน
- 3) รักษาตำแหน่งของเสาอากาศ เพื่อให้การสื่อสารกับโลกเสถียร
- 4) ลดการหมุนที่ไม่ต้องการ ซึ่งอาจเกิดจากแรงโน้มถ่วงของโลก, แรงจากแสงอาทิตย์ หรือแรงจากชั้นบรรยากาศเบาบาง

### 2.7.3 ตัวอย่างการใช้งานจริงของการควบคุมสมดุสดาวเทียม

- 1) ดาวเทียมสำรวจโลก (Earth Observation Satellites)  
ต้องควบคุมให้กล้องชี้ไปยังพื้นโลกตลอดเวลา เพื่อให้ได้ภาพที่คมชัดและต่อเนื่อง
- 2) สถานีอวกาศนานาชาติ (ISS - International Space Station)  
ใช้ CMG (Control Moment Gyroscope) เพื่อรักษาสมดุสดตลอดเวลา เพราะไม่มีเชื้อเพลิงสำหรับปรับทิศทางตลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 3) ดาวเทียม GPS ต้องรักษาตำแหน่งและทิศทางของเสาอากาศ เพื่อให้ส่งสัญญาณ GPS ไปยังพื้นโลกได้ [16]

## 2.8 ระบบที่ใช้ในการกำหนดและควบคุมทิศทางของดาวเทียม (Attitude Determination and Control System)

ระบบ Attitude Determination and Control System (ADCS) เป็นระบบที่ใช้ในการกำหนด และควบคุมทิศทางของดาวเทียม เพื่อให้สามารถรักษาตำแหน่ง หรือหมุนไปยังทิศทางที่ต้องการได้ โดยระบบนี้มีความสำคัญอย่างมากต่อการทำงานของดาวเทียม และยานอวกาศต่าง ๆ เช่น CubeSat, ดาวเทียมสื่อสาร, กล้องโทรทรรศน์อวกาศ, และยานสำรวจอวกาศ องค์ประกอบของระบบ ADCS มีสองส่วนหลัก ได้แก่ การกำหนดทิศทาง (Attitude Determination) ใช้เซ็นเซอร์วัดค่าต่าง ๆ เพื่อกำหนดว่าตอนนี้ยานอวกาศหันไปทางไหน และการควบคุมทิศทาง (Attitude Control) ใช้อุปกรณ์สร้างแรงบิดเพื่อปรับทิศทางของยานให้เป็นที่ต้องการ [17]

### 2.8.1 การกำหนดทิศทาง (Attitude Determination)

ความแม่นยำของการคำนวณทิศทางของยานอวกาศเป็นผลมาจากทั้งความสามารถของอัลกอริธึมการกำหนดทิศทาง และความแม่นยำของเซ็นเซอร์ที่ใช้ในการวัดค่า การกำหนดทิศทางของยานอวกาศทำได้โดยใช้ เซ็นเซอร์ตรวจจับสภาพแวดล้อม แล้วนำข้อมูลมาประมวลผลเพื่อตรวจสอบว่ายานอวกาศกำลังหันไปทางไหน ข้อมูลจากเซ็นเซอร์เหล่านี้จะถูกส่งไปยัง Onboard Computer (OBC) ซึ่งใช้ อัลกอริธึมประมวลผล เช่น Kalman Filter หรือ Extended Kalman Filter (EKF) เพื่อรวมข้อมูลจากหลายแหล่งเข้าด้วยกันและคำนวณตำแหน่ง และทิศทางที่แน่นอนของยานอวกาศ เซ็นเซอร์ที่ใช้ในระบบ ADCS ซึ่งเป็นที่นิยม และพบได้มากที่สุดมีดังนี้

#### 2.8.1.1 เซ็นเซอร์วัดดวงอาทิตย์ (Sun Sensor)

เซ็นเซอร์วัดดวงอาทิตย์ที่พบในดาวเทียมขนาดเล็กส่วนใหญ่มักเป็นเซ็นเซอร์วัดดวงอาทิตย์แบบหยาบ (Coarse Sun Sensors - CSS) ซึ่งโดยหลักแล้วจะเป็นเซลล์โฟโตอิเล็กทริกทำหน้าที่เปลี่ยนพลังงานจากโฟตอนที่ตกกระทบเซ็นเซอร์ให้เป็นกระแสไฟฟ้า จากนั้นกระแสไฟฟ้าจะถูกวัด และแปลงเป็นสัญญาณดิจิทัล สัญญาณดิจิทัลจาก CSS แต่ละตัวจะถูกเชื่อมโยงกับตำแหน่งที่แน่นอนบนยานอวกาศที่ติดตั้งเซ็นเซอร์นั้นไว้ และจะส่งค่าความเข้มของแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่ตรวจจับได้ในระนาบนั้นๆ ออกมา เมื่อมีการติดตั้ง CSS หลายตัวในระนาบที่แตกต่างกันในแนวแกน X, Y และ Z ทั้งในทิศทางบวก และลบ ระบบจะสามารถสร้างข้อมูลเวกเตอร์ของทิศทางที่ดวงอาทิตย์ส่องเข้ามาในรูปแบบสามมิติ และส่งข้อมูลไปยังระบบ ADCS เพื่อใช้กำหนดทิศทางของยานอวกาศ

#### 2.8.1.2 แมกนีโทมิเตอร์ (Magnetometer)

แมกนีโทมิเตอร์เป็นอุปกรณ์ที่ใช้ตรวจจับสนามแม่เหล็ก โดยพื้นฐานแล้วประกอบด้วยขดลวดที่สามารถตรวจจับการเปลี่ยนแปลงของกระแสไฟฟ้าภายในขดลวด อันเนื่องมาจากการเปลี่ยนแปลงของสนามแม่เหล็กโดยรอบ แมกนีโทมิเตอร์ให้ข้อมูลที่สำคัญสองประเภท ได้แก่ ระบบ ADCS อย่างแรกคือการวัดค่าความเข้มของสนามแม่เหล็กโดยรอบในกรอบอ้างอิงของตัวเอง ค่านี้เมื่อรวมกับค่าที่ทราบล่วงหน้าของสนามแม่เหล็กโลก และค่าของสนามแม่เหล็กที่อาจเกิดจากตัวยานอวกาศเอง จะสามารถใช้คำนวณเป็นเวกเตอร์ชี้ทิศทางได้ นอกจากนี้แมกนีโทมิเตอร์ยังสามารถวัดอัตราการเปลี่ยนแปลงของสนามแม่เหล็ก ซึ่งช่วยให้ระบบ ADCS ทราบถึงอัตราการหมุนของยานอวกาศรอบแกนของตัวเอง แมกนีโทมิเตอร์มีให้เลือกใช้ทั้งแบบหลายแกนและแบบแกนเดียว แต่ระบบ ADCS ที่เรียบง่ายที่สุดมักจะใช้แมกนีโทมิเตอร์แบบแกนเดียวติดตั้งไว้ในแต่ละแกนหลักของยานอวกาศ

#### 2.8.1.3 ไจโรสโคป (Gyroscope)

ไจโรสโคปเป็นเซ็นเซอร์ที่ใช้กันอย่างแพร่หลายในการวัดอัตราการเปลี่ยนแปลงเชิงมุมของยานอวกาศ หลักการทำงานของไจโรสโคปอาจแตกต่างกันไป เช่น การใช้แผ่นหมุนทางกายภาพ (spinning plates) หรือการใช้ระบบเซ็นเซอร์เชิงแสง (optical sensing systems) รวมถึงมีให้เลือกใช้ทั้งแบบแกนเดียว และหลายแกน ในแต่ละประเภทของไจโรสโคป ไม่ว่าจะเป็นระบบกายภาพที่ใช้วัดการเปลี่ยนแปลงของความเร็วเชิงมุมของมวลที่หมุน หรือระบบแสงที่ใช้วัดเวลาที่แสงเดินทางผ่านเส้นทางที่กำหนด การเปลี่ยนแปลงของค่าที่วัดได้เมื่อเทียบกับเวลา สามารถนำไปใช้คำนวณอัตราการหมุนของยานอวกาศรอบแกนการหมุนของตัวเองได้ เมื่อมีการติดตั้งไจโรสโคปแบบแกนเดียวในแต่ละแกนหลักของยานอวกาศ ระบบจะสามารถวัดอัตราการหมุนของยานอวกาศในทุกทิศทางได้

#### 2.8.1.4 สตาร์แทร็กเกอร์ (Star Tracker)

สตาร์แทร็กเกอร์ หรือกล้องตรวจจับดาว (star cameras) เป็นเซ็นเซอร์ออปติคัลที่สามารถให้ข้อมูลทั้งอัตราการเปลี่ยนแปลงเชิงมุม และเวกเตอร์ชี้ทิศทางไปยังระบบ ADCS สตาร์แทร็กเกอร์ในปัจจุบันสามารถทำงานโดยอ้างอิงกับฐานข้อมูลดาวที่ถูกโหลดไว้ล่วงหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภายในสตาร์แทร็กเกอร์ จะรวบรวมข้อมูลภาพของท้องฟ้า และใช้กระบวนการเปรียบเทียบกับฐานข้อมูลดาวที่มีอยู่ เพื่อคำนวณเวกเตอร์การชี้ตำแหน่งของยานอวกาศเมื่อเทียบกับระบบพิกัดของดวงดาว เมื่อสตาร์แทร็กเกอร์ทำการสแกนดาวในท้องฟ้าได้ครั้งแรก ระบบจะสามารถใช้ข้อมูลนั้นเป็นจุดอ้างอิง และเมื่อมีการเคลื่อนที่ของยานอวกาศ สตาร์แทร็กเกอร์สามารถคำนวณอัตราการหมุนของยานอวกาศรอบกลุ่มดาวที่ตรวจพบได้ โดยทั่วไปสตาร์แทร็กเกอร์เป็นเซ็นเซอร์ที่ให้ค่าความแม่นยำสูงสุดในบรรดาเซ็นเซอร์ทั้งหมดที่ใช้ในระบบ ADCS

#### 2.8.1.5 เซ็นเซอร์ขอบฟ้าโลก (Earth Horizon Sensors)

เซ็นเซอร์ตรวจจับโลกทำงานโดยการรวบรวม และประมวลผลสัญญาณแสง (โดยทั่วไปอยู่ในช่วงอินฟราเรด) ที่แผ่ออกมาจากโลก และใช้ข้อมูลเหล่านี้ในการกำหนดตำแหน่งและทิศทางสัมพันธ์ของดาวเทียบกับทิศทางของสัญญาณดังกล่าว กระบวนการนี้ช่วยให้สามารถคำนวณเวกเตอร์แนวตั้งในท้องถื่น (nadir vector) หรือทิศทางระหว่างตัวดาวเทียบกับโลกได้อย่างแม่นยำ กระบวนการนี้มักถูกเรียกว่า “limb sensing” ซึ่งหมายถึงการตรวจจับขอบของชั้นบรรยากาศ และเป็นคำที่ใช้แทนความหมายเดียวกับการตรวจจับขอบฟ้าโลก [18]

#### 2.8.1.6 ระบบนำทางด้วยดาวเทียม (Global Navigation Satellite System)

ระบบ Global Navigation Satellite System (GNSS) ซึ่งเป็นคำที่ทั่วโลกใช้เรียกระบบดาวเทียมที่มีการเปิดให้บริการอยู่ในปัจจุบัน และระบบดาวเทียมที่มีการวางแผนจะเปิดให้บริการในอนาคต ซึ่งดาวเทียมต่าง ๆ ในระบบ GNSS ประกอบด้วย GPS ย่อมาจาก Global Positioning System ซึ่งเป็นดาวเทียมระบบแรกของโลกที่ออกแบบโดยประเทศสหรัฐอเมริกา มีดาวเทียมทั่วโลกทั้งหมด 28 ดวง GLONASS เป็นระบบดาวเทียมของประเทศรัสเซียมีดาวเทียมทั่วโลกทั้งหมด 24 ดวง และ Galileo เป็นระบบดาวเทียมของสหภาพยุโรป ซึ่งทั้งระบบจะมีดาวเทียมทั่วโลกทั้งหมด 30 ดวง ภายในปี 2020 [19]

### 2.8.2 การควบคุมทิศทาง (Attitude Control)

เมื่อรู้ตำแหน่งของยานอวกาศแล้วระบบจะใช้อุปกรณ์ควบคุมทิศทาง เพื่อให้ยานหมุนไปตามที่ต้องการ ค่าของแรงควบคุม หรือแรงบิดที่คำนวณได้จากฟังก์ชันควบคุมให้ค่าปริมาณเชิงตัวเลขที่จำเป็นในการเปลี่ยนทิศทางของยานอวกาศ ค่าปริมาณเชิงตัวเลขนี้ขึ้นอยู่กับคุณสมบัติทางกายภาพที่เป็นไปได้ของกลไกควบคุม เช่น น้ำหนักทางกายภาพ และอัตราการหมุนสูงสุดของล้อปฏิกิริยา ตำแหน่งของกลไกภายในกรอบอ้างอิง และตำแหน่งเมื่อเทียบกับโมเมนต์ความเฉื่อยเป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สิ่งจำเป็นสำหรับการคำนวณแชนโมเมนต์ และการใช้แรงบิด ในบางกรณี เช่น ระบบขับเคลื่อน ทิศทางของกลไกก็เป็นปัจจัยที่ต้องพิจารณาด้วย อุปกรณ์ควบคุมทิศทางที่ใช้ใน ADCS มีตัวอย่างดังนี้

### 2.8.2.1 ล้อปฏิกิริยา (Reaction Wheels)

ล้อปฏิกิริยา (Reaction wheels) ทำงานโดยใช้มอเตอร์ไฟฟ้าเพื่อหมุนล้อที่มีน้ำหนักภายในโครงสร้างของยานอวกาศ ซึ่งจะสร้างค่าโมเมนตัมสะสมขึ้น เมื่อมีการเบรกที่ล้อ โมเมนตัมนี้จะถูกถ่ายทอดไปยังตัวถังของยานอวกาศ ทำให้เกิดแรงบิดที่ช่วยหมุนยาน โดยทั่วไปล้อปฏิกิริยาจะถูกติดตั้งตามแกนหลักทั้งสามของยานอวกาศ เพื่อให้สามารถควบคุมการเคลื่อนที่ในทุกทิศทางได้ อัตราการหมุนของล้อปฏิกิริยาสามารถปรับเปลี่ยนได้อย่างต่อเนื่องจนถึงค่าความเร็วสูงสุด ซึ่งช่วยให้สามารถควบคุมแรงบิดที่ใช้กับยานอวกาศได้อย่างแม่นยำ อย่างไรก็ตาม ข้อจำกัดของล้อปฏิกิริยาคือมีอัตราการหมุนสูงสุด เมื่อถึงจุดอิ่มตัว (Saturation) จะไม่สามารถสร้างแรงบิดเพิ่มเติมได้อีก ซึ่งจำเป็นต้องมีระบบควบคุมเสริมเพื่อช่วยระบายโมเมนตัมหรือทำให้โมเมนตัมที่สะสมอยู่ลดลง

### 2.8.2.2 แท่งแรงบิดแม่เหล็ก (Magnetorquers)

แท่งแรงบิดแม่เหล็กทำงานบนหลักการของโมเมนต์ไดโพลแม่เหล็ก กลไกการควบคุมเริ่มต้นเมื่อมีกระแสไฟฟ้าไหลผ่านขดลวดแม่เหล็กที่ติดตั้งอยู่บนยานอวกาศ ซึ่งจะสร้างสนามแม่เหล็กขึ้นบนตัวยาน เมื่อสนามแม่เหล็กที่เกิดขึ้นมีปฏิสัมพันธ์กับสนามแม่เหล็กของโลก แรงบิดจะถูกสร้างขึ้นโดยกระทำรอบจุดศูนย์กลางมวล (COG) ของยานอวกาศ ทำให้สามารถปรับทิศทางของยานได้ อุปกรณ์แท่งแรงบิดแม่เหล็กต้องถูกติดตั้งตามแกนหลักทั้งสามของยานอวกาศ เพื่อให้สามารถควบคุมการเคลื่อนที่ได้ในทุกทิศทางของอวกาศ ขนาดของแรงบิดที่เกิดขึ้นสามารถปรับเปลี่ยนได้โดยการควบคุมกระแสไฟฟ้าที่จ่ายให้กับขดลวด ซึ่งจะกำหนดความแรงของสนามแม่เหล็กที่สร้างขึ้น นอกจากนี้ การกลับทิศทางของกระแสไฟฟ้าผ่านขดลวด จะทำให้ทิศทางของโมเมนต์ไดโพลแม่เหล็กกลับด้าน ส่งผลให้สามารถสร้างแรงบิดทั้งในทิศทางบวก และลบของแกนที่กำหนดได้ แท่งแรงบิดแม่เหล็กเป็นอุปกรณ์ที่ใช้พลังงานต่ำมาก แต่โดยธรรมชาติแล้วอาจต้องใช้ระยะเวลาในการสร้างแรงบิดที่มีนัยสำคัญเพียงพอสำหรับการควบคุมทิศทางของยานอวกาศ

### 2.8.2.3 อุปกรณ์ขับเคลื่อนของยานอวกาศ (Thruster)

Thruster เป็นอุปกรณ์ขับเคลื่อนของยานอวกาศที่ใช้สำหรับการรักษาตำแหน่งในวงโคจร (orbital station-keeping), การควบคุมทิศทาง (attitude control) หรือการเร่งความเร็วแบบแรงขับต่ำในระยะเวลาอันยาวนาน (long-duration, low-thrust acceleration) โดยมักเป็นส่วนหนึ่งของระบบควบคุมปฏิกิริยา (Reaction control system) Vernier Thruster หรือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Gimballed Engine เป็นกรณีพิเศษของระบบขับเคลื่อนที่ใช้ในจรวดขนส่งอวกาศ ซึ่งอาจเป็นเครื่องยนต์จรวดเสริม (secondary rocket engine) หรืออุปกรณ์ที่ให้แรงขับสูงอื่น ๆ เพื่อใช้ควบคุมทิศทางของจรวด ขณะที่เครื่องยนต์หลัก (primary thrust engine) ซึ่งโดยทั่วไปเป็นเครื่องยนต์จรวดจะถูกยึดอยู่กับตัวจรวด และทำหน้าที่สร้างแรงขับหลัก

#### 2.8.2.4 อุปกรณ์กระจายพลังงาน (Nutation Dampers)

Nutation Dampers เป็นอุปกรณ์แบบพาสซีฟที่ใช้ในการกระจายพลังงานส่วนเกินออกจากระบบ โดยทั่วไปจะอยู่ในรูปของท่อที่บรรจุของเหลวที่มีความหนืด (viscous fluid) และมีมวลที่ติดตั้งอยู่กับสปริง ในบางกรณี nutation dampers อาจมีเพียงของเหลว โดยเติมของเหลวเพียงบางส่วนของท่อเพื่อให้สามารถเคลื่อนที่ และกระเพื่อมได้ จุดประสงค์หลักของอุปกรณ์นี้คือ การกระจายพลังงานผ่านแรงเสียดทานของของเหลว การสั่นไหวของยานอวกาศที่เกิดจากความไม่สอดคล้องของความเร็วเชิงมุม (angular velocity) กับแกนหมุนหลัก (principal spin axis) จะกระตุ้นให้เกิดความเร่งทั่วทั้งตัวดาวเทียม ส่งผลให้เกิดการกระเพื่อมของของเหลว การยึดตัว และการบิดงอของโครงสร้างที่ไม่แข็งตัว ซึ่งทั้งหมดนี้สามารถช่วยในการกระจายพลังงานได้ Nutation dampers ถูกติดตั้งเพื่อเพิ่มประสิทธิภาพในการกระจายพลังงานโดยเจตนา ซึ่งเป็นปัจจัยสำคัญในการรักษาเสถียรภาพของยานอวกาศประเภท oblate single spinners และ dual-spin spacecraft [20]

#### 2.8.2.5 Control Moment Gyroscope

Control Moment Gyroscope (CMG) เป็นอุปกรณ์ควบคุมทิศทางที่ใช้ในระบบควบคุมทัศนคติของยานอวกาศ (Attitude Control System) โดยทั่วไปจะประกอบด้วย rotor ที่หมุนด้วยความเร็วสูง และ gimbal หนึ่งตัวหรือมากกว่าที่สามารถปรับมุมเอียงของ rotor ได้ เมื่อ rotor เปลี่ยนมุมเอียง ทิศทางของโมเมนตัมเชิงมุม (angular momentum) ของมันก็จะเปลี่ยนแปลงไป ทำให้เกิดแรงบิดใจโรสโคป (gyroscopic torque) ซึ่งส่งผลให้ยานอวกาศหมุนไปในทิศทางที่ต้องการ CMG จึงเป็นอุปกรณ์ที่มีประสิทธิภาพสูงในการควบคุมทิศทางของยานอวกาศ เนื่องจากสามารถสร้างแรงบิดได้มากโดยใช้พลังงานค่อนข้างต่ำเมื่อเทียบกับอุปกรณ์ควบคุมทิศทางประเภทอื่น ๆ

### 2.8.3 ระบบประมวลผลและอัลกอริธึมควบคุม (Processor)

ในระบบควบคุมทัศนคติของยานอวกาศ (Attitude Determination and Control System - ADCS) ข้อมูลจากเซ็นเซอร์และอุปกรณ์ควบคุมจะถูกส่งไปยัง Onboard Computer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(OBC) ซึ่งทำหน้าที่เป็นหน่วยประมวลผลหลักของระบบ OBC ใช้อัลกอริธึมควบคุม (Control Algorithms) เพื่อตัดสินใจว่าควรปรับทิศทางของยานอวกาศอย่างไรให้เหมาะสมกับเป้าหมายของภารกิจ

### 2.8.3.1 Onboard Computer (OBC)

หน่วยประมวลผลกลางที่ควบคุมการทำงานของ ADCS มีหน้าที่หลักดังนี้

- 1) รับข้อมูลจากเซ็นเซอร์ที่ใช้ในการวัดทิศทางของยานอวกาศ
- 2) ประมวลผลข้อมูลทิศทาง โดยใช้ อัลกอริธึมการกำหนดทิศทาง (Attitude Determination Algorithms) เพื่อตรวจสอบว่าทิศทางปัจจุบันของยานอวกาศเป็นไปตามที่กำหนดหรือไม่
- 3) คำนวณ และส่งคำสั่งไปยังอุปกรณ์ควบคุมเพื่อทำการปรับมุมของยานอวกาศให้ถูกต้อง

Onboard Computer ต้องทำงานแบบเรียลไทม์เพื่อให้ยานอวกาศสามารถรักษาทิศทางและเสถียรภาพได้อย่างแม่นยำ ซึ่งมีความสำคัญอย่างยิ่งในภารกิจที่ต้องการความแม่นยำสูง เช่น ดาวเทียมสำรวจโลก, ยานสำรวจอวกาศ และดาวเทียมสื่อสาร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

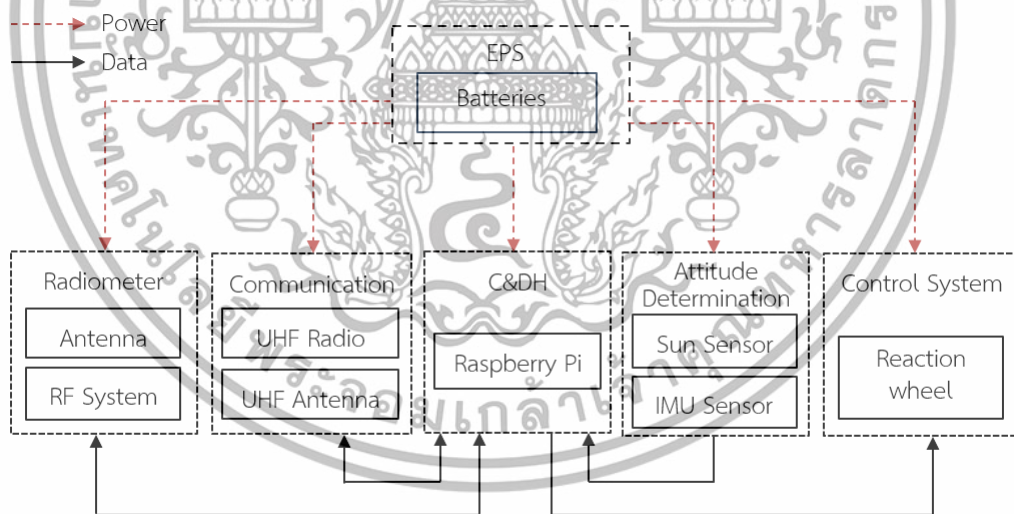
## บทที่ 3

### การออกแบบและการจัดทำปริญญาานิพนธ์

#### 3.1 การออกแบบ

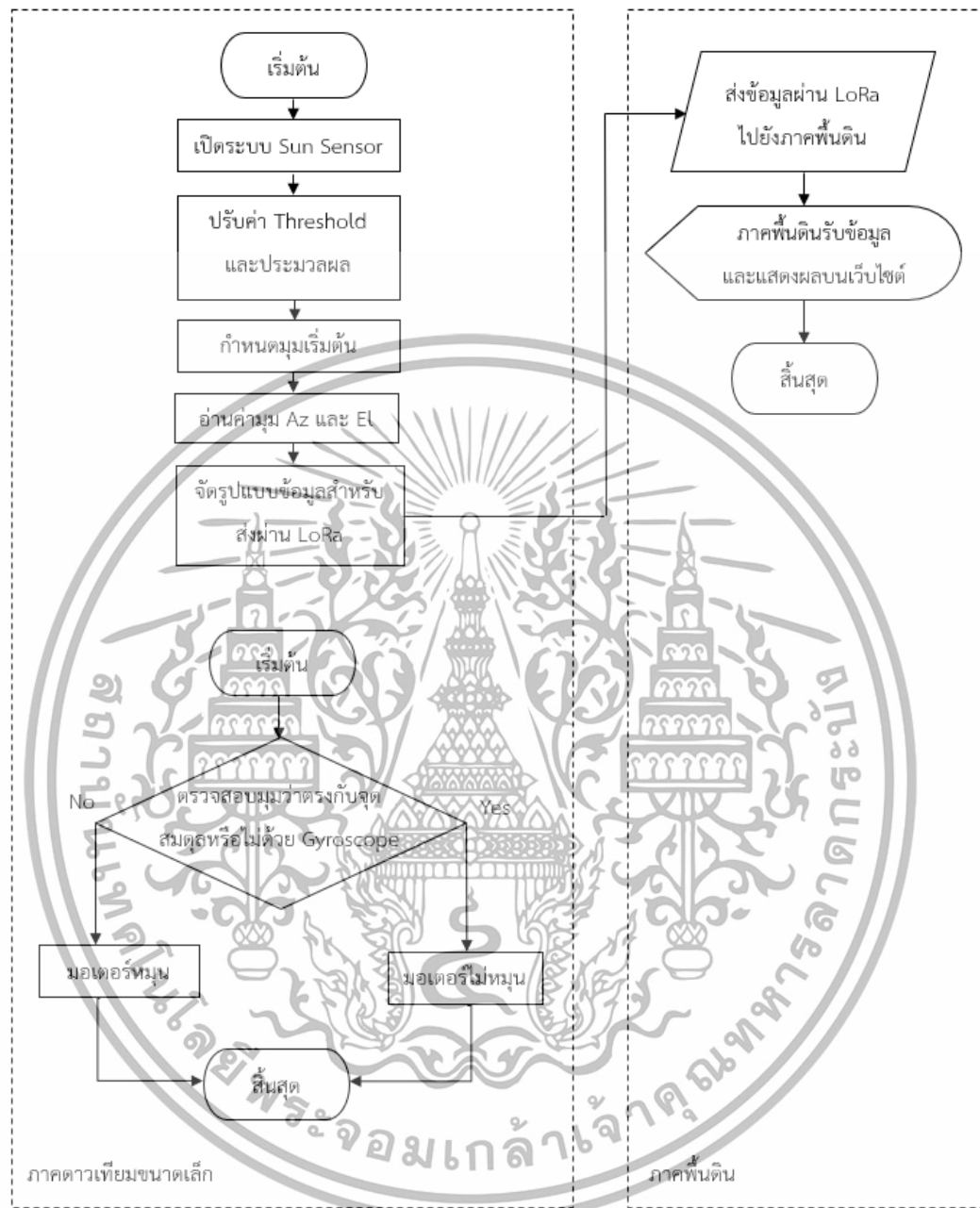
##### 3.1.1 การออกแบบระบบ

ปริญญาานิพนธ์นี้ทำการออกแบบการทำงานของระบบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ โดยใช้ต้นทุนที่ต่ำลง ทำการออกแบบระบบ ทดสอบ และปรับปรุงประสิทธิภาพของระบบควบคุมตำแหน่ง และทิศทาง จากนั้นติดตั้งเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ และทดสอบระบบควบคุมการหมุนของดาวเทียมขนาดเล็ก พัฒนาระบบจัดการคำสั่ง และข้อมูลด้วย Raspberry Pi เพื่อควบคุมการทำงาน ออกแบบระบบจัดการพลังงานให้เกิดประสิทธิภาพสูงสุด และศึกษาการจำลองระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย (TC & TM Subsystem) โดยบล็อกไดอะแกรมภาพรวมของระบบ และส่วนผังงานดังรูปที่ 3.1 และ 3.2



รูปที่ 3.1 บล็อกไดอะแกรมภาพรวมของระบบระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็กโดยใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

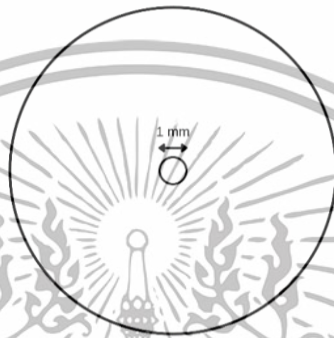


รูปที่ 3.2 ผังงานภาพรวมของระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็กโดยใช้เซ็นเซอร์ CMOS ตรวจสอบทิศทางแสงอาทิตย์ต้นทุนต่ำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.2 การออกแบบสำหรับหน้ากากของตรวจจับทิศทางแสงอาทิตย์

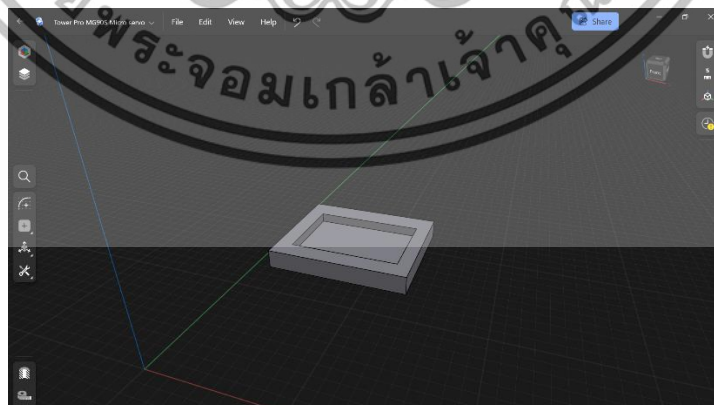
ออกแบบรูรับแสงบนหน้ากากของเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์นั้น ได้ออกแบบรูรับแสงในรูปแบบวงกลมซึ่งมีเส้นผ่านศูนย์กลางที่ขนาด 1 มิลลิเมตร เพื่อให้สามารถเก็บผลลัพธ์ได้อย่างมีประสิทธิภาพดังรูปที่ 3.3



รูปที่ 3.3 การออกแบบรูรับแสงในรูปแบบวงกลม

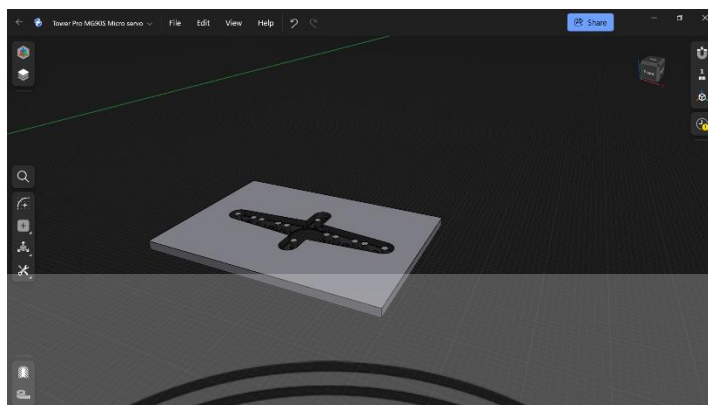
### 3.1.3 การออกแบบโครงสร้างชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์

ออกแบบโครงสร้างชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ ผ่านโปรแกรม Shapr3D เพื่อนำไปพิมพ์เป็นสามมิติผ่านโปรแกรม Ultimaker Cura จำลองชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ โดยการออกแบบโครงสร้างสามมิติดังรูปที่ 3.4, 3.5, 3.6, 3.7, 3.8, 3.9 และ 3.10

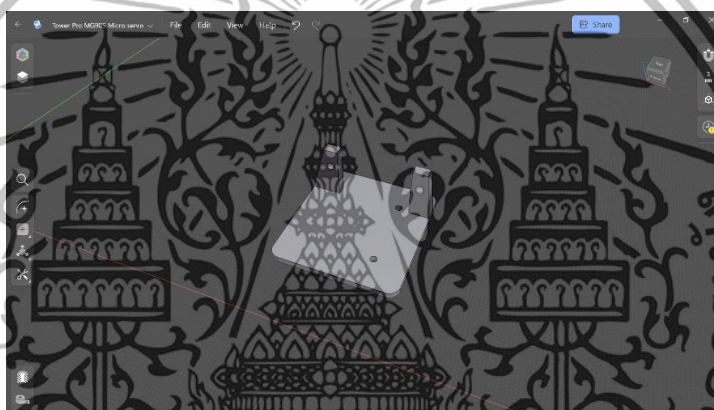


รูปที่ 3.4 ฐานข้างล่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.5 ฐานข้างบน

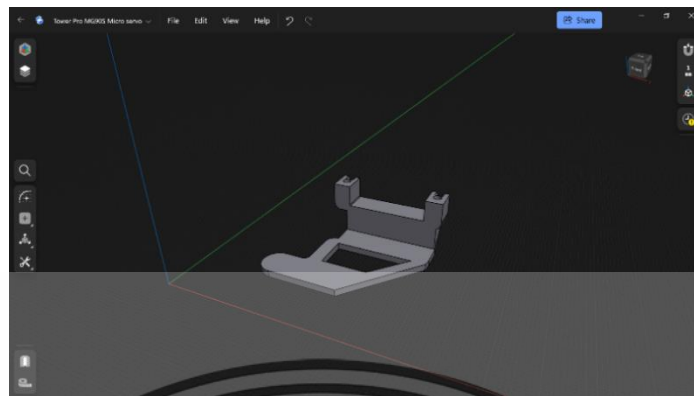


รูปที่ 3.6 พื้นที่ใช้ติดตั้งกล้อง PI Camera

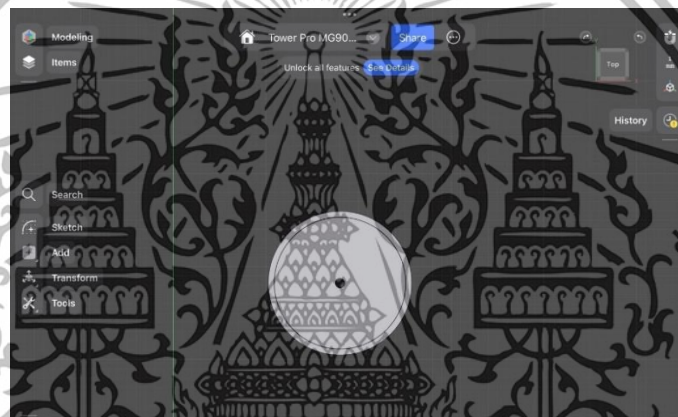


รูปที่ 3.7 ตัวลอคมอเตอร์ฝั่งซ้าย

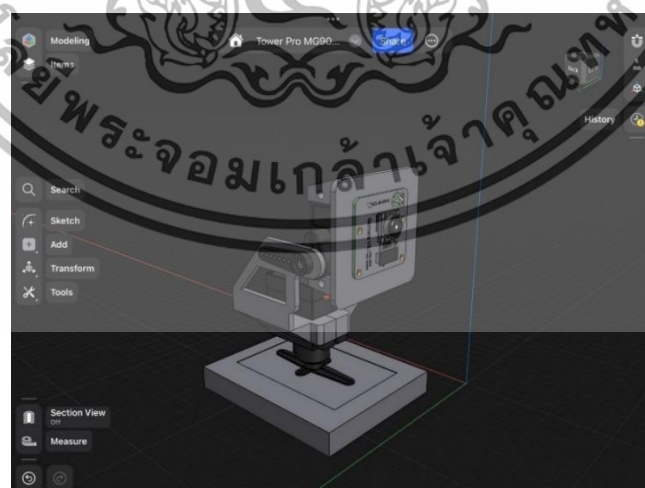
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.8 ตัวล๊อคมอเตอร์ฝั่งขวา



รูปที่ 3.9 หน้ากาก (Mask)



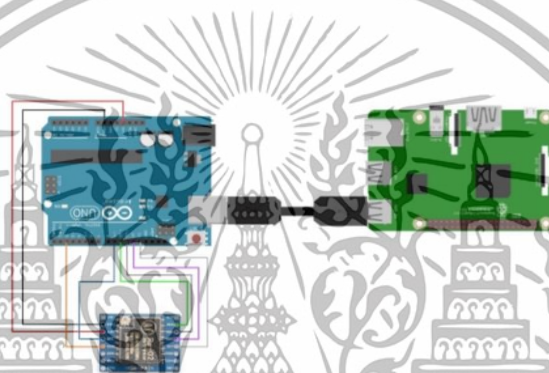
รูปที่ 3.10 ชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์แบบเสรีจสมบูรณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.4 การออกแบบระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย ( TC & TM Subsystem )

#### 3.1.4.1 การออกแบบระบบการสื่อสารทางด้านภาคส่ง

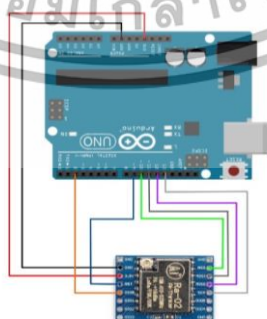
ออกแบบระบบการสื่อสารทางด้านภาคส่ง โดยออกแบบการเชื่อมต่อวงจร ดังรูปที่ 3.11 นำโมดูล LoRa เชื่อมต่อบอร์ด Arduino UNO แล้วเชื่อมต่อเข้ากับบอร์ด Raspberry Pi 5 ซึ่งใช้สำหรับจำลองการส่งข้อมูลจากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์บนดาวเทียมขนาดเล็กไปยังสถานีภาคพื้นดินแบบไร้สาย



รูปที่ 3.11 ออกแบบระบบการสื่อสารทางด้านภาคส่ง

#### 3.1.4.2 การออกแบบระบบการสื่อสารทางด้านภาครับ

ออกแบบระบบการสื่อสารทางด้านภาครับ โดยออกแบบการเชื่อมต่อวงจร ดังรูปที่ 3.12 นำโมดูล LoRa เชื่อมต่อบอร์ด Arduino UNO แล้วเชื่อมต่อเข้ากับคอมพิวเตอร์ PC ซึ่งใช้สำหรับจำลองการรับข้อมูล



รูปที่ 3.12 ออกแบบระบบการสื่อสารทางด้านภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.1.5 การออกแบบเว็บไซต์สำหรับแสดง และเก็บค่าข้อมูลของดาวเทียมขนาดเล็ก

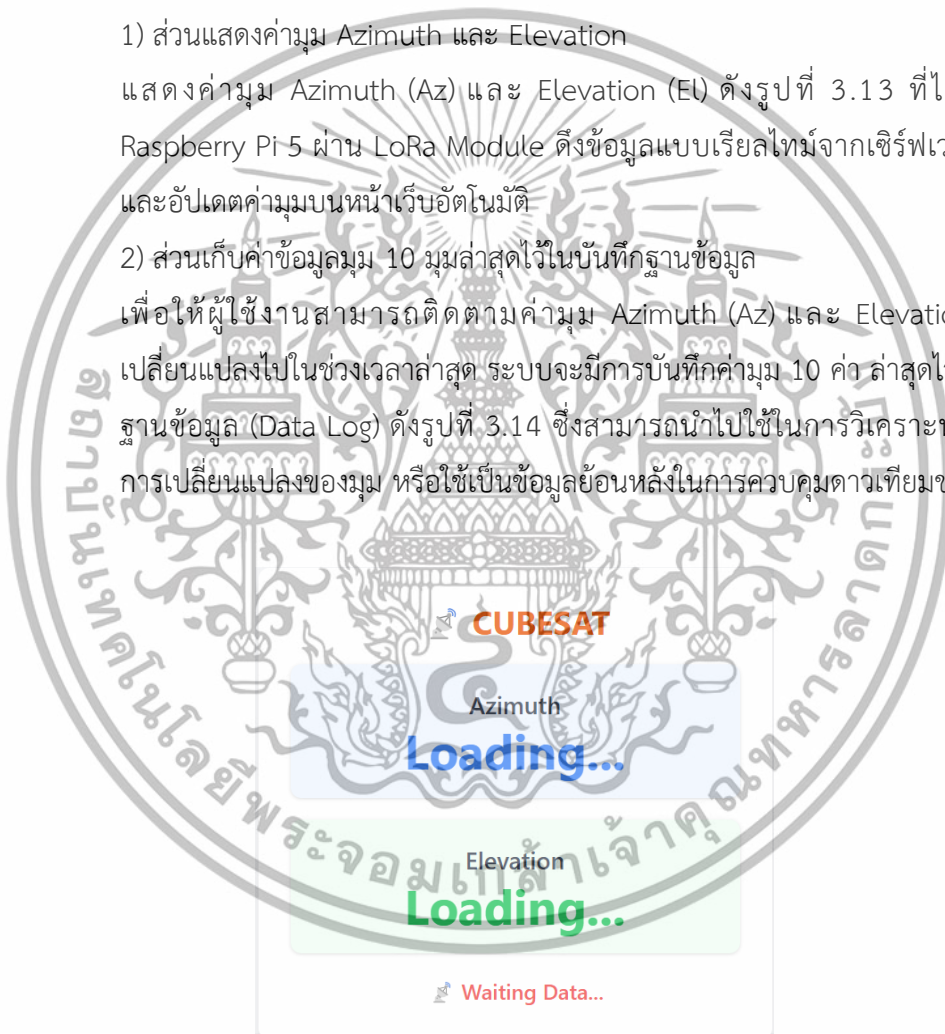
ออกแบบเว็บไซต์ ด้วยภาษา Python บนโปรแกรม Microsoft Visual Studio Code โดยตัวระบบพัฒนาด้วย Flask (ภาษา Python) สำหรับฝั่งเซิร์ฟเวอร์ และสำหรับฝั่งผู้ใช้งาน (Frontend) HTML, CSS และ JavaScript เพื่อใช้แสดงค่ามุมที่ส่งมาจากดาวเทียมขนาดเล็ก (CubeSat) และเก็บค่าข้อมูลมุม 10 มุมล่าสุดไว้ใน data log องค์กรประกอบหลักของเว็บไซต์ ประกอบด้วยองค์ประกอบหลักดังนี้

1) ส่วนแสดงค่ามุม Azimuth และ Elevation

แสดงค่ามุม Azimuth (Az) และ Elevation (EL) ดังรูปที่ 3.13 ที่ได้รับจาก Raspberry Pi 5 ผ่าน LoRa Module ดึงข้อมูลแบบเรียลไทม์จากเซิร์ฟเวอร์ Flask และอัปเดตค่ามุมบนหน้าเว็บอัตโนมัติ

2) ส่วนเก็บค่าข้อมูลมุม 10 มุมล่าสุดไว้ในบันทึกฐานข้อมูล

เพื่อให้ผู้ใช้งานสามารถติดตามค่ามุม Azimuth (Az) และ Elevation (EL) ที่เปลี่ยนแปลงไปในเวลาล่าสุด ระบบจะมีการบันทึกค่ามุม 10 ค่า ล่าสุดไว้ในบันทึกฐานข้อมูล (Data Log) ดังรูปที่ 3.14 ซึ่งสามารถนำไปใช้ในการวิเคราะห์แนวโน้มการเปลี่ยนแปลงของมุม หรือใช้เป็นข้อมูลย้อนหลังในการควบคุมดาวเทียมขนาดเล็ก



รูปที่ 3.13 เว็บไซต์ในส่วนแสดงค่ามุม Azimuth และ Elevation

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

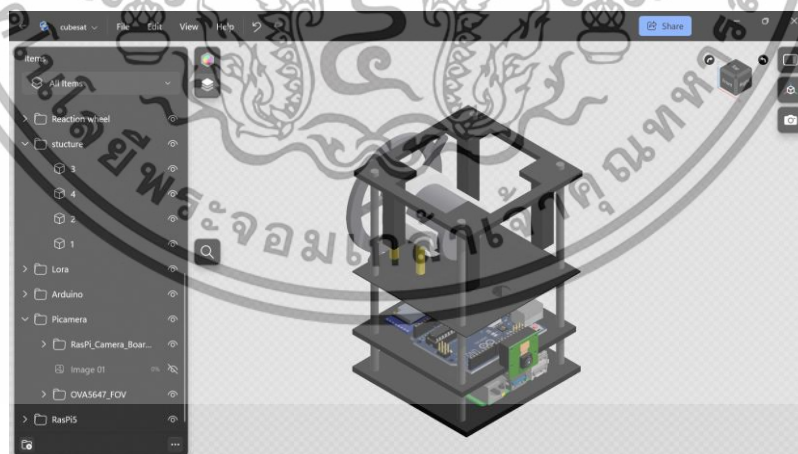
**Data Log (Latest 10 Entries)**

Timestamp	Azimuth	Elevation
2025-02-24 00:30:15	-5.1	6.04
2025-02-24 00:30:13	-5.1	6.04
2025-02-24 00:30:11	-5.1	6.04
2025-02-24 00:30:09	-5.1	6.04
2025-02-24 00:30:01	-5.1	6.04
2025-02-24 00:29:49	-4.51	5.95
2025-02-24 00:29:39	-4.51	6.04
2025-02-24 00:29:35	-4.51	6.04
2025-02-24 00:29:33	-12.46	4.92
2025-02-24 00:29:31	-21.73	-1.81

รูปที่ 3.14 เว็บไซต์ในส่วนเก็บค่าข้อมูล 10 มุมล่าสุดไว้ในบันทึกฐานข้อมูล (Data Log)

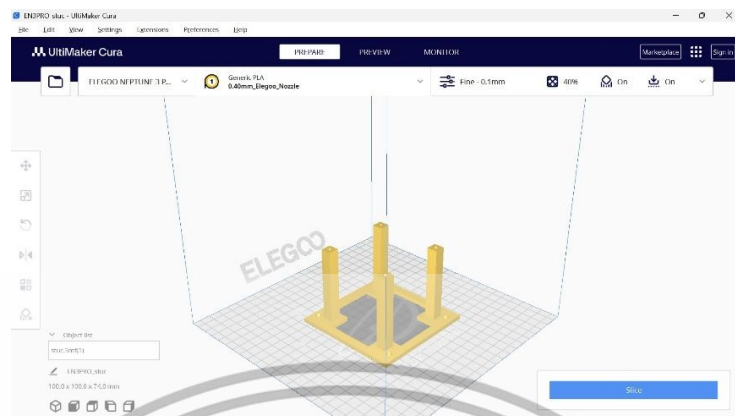
### 3.1.6 การออกแบบโครงสร้างดาวเทียมขนาดเล็ก

ออกแบบโครงสร้างของดาวเทียมขนาดเล็กผ่านโปรแกรม Shapr3D เพื่อนำไปพิมพ์สามมิติผ่านโปรแกรม Ultimaker Cura โดยการจำลองดาวเทียมขนาดเล็กนั้นจะทำการจำลองดาวเทียมขนาด 1.5 ยูนิต การออกแบบโครงสร้างดังในรูปที่ 3.15, 3.16, 3.17, 3.18 และ 3.19



รูปที่ 3.15 โครงสร้างภาพรวมของดาวเทียมขนาดเล็ก

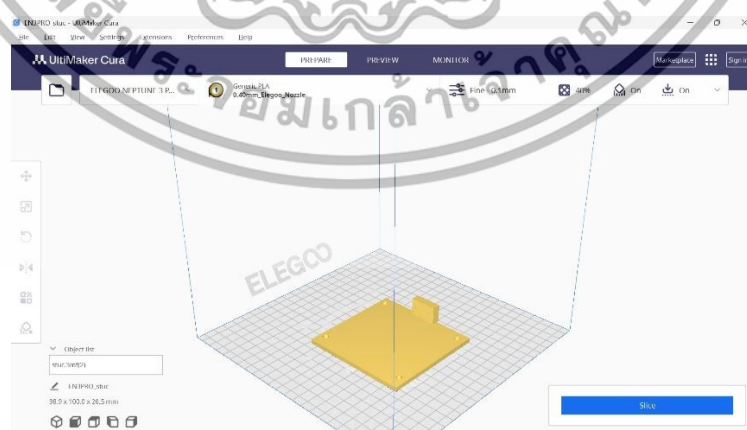
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.16 โครงสร้างสามมิติส่วนบน

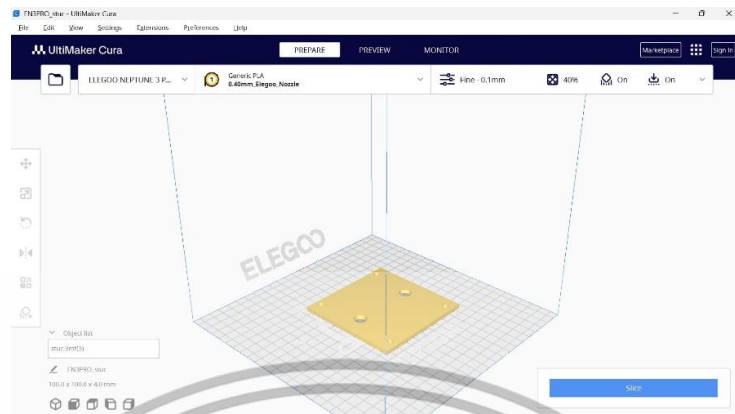


รูปที่ 3.17 โครงสร้างส่วนด้านล่าง



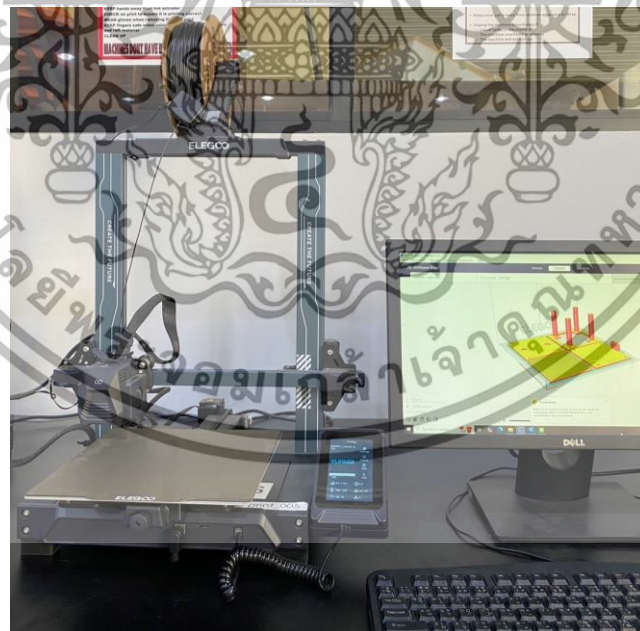
รูปที่ 3.18 โครงสร้างส่วนที่ติดตั้งเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



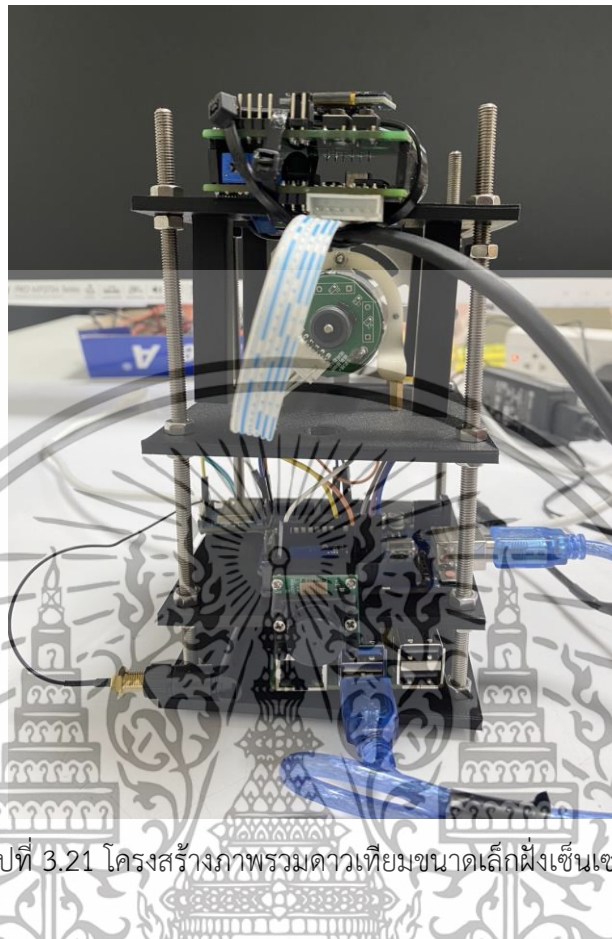
รูปที่ 3.19 โครงสร้างส่วนติดมอเตอร์

หลังจากที่ทำการออกแบบทุกชิ้นส่วนในโครงสร้างสามมิติของดาวเทียมขนาดเล็กเสร็จสิ้นเรียบร้อยแล้ว จึงนำไปพิมพ์โมเดลด้วยเครื่องพิมพ์สามมิติแล้วประกอบโครงสร้างดาวเทียมดังในรูปที่ 3.20 และ 3.21 โดยในโครงสร้างภาพรวมดาวเทียมขนาดเล็กจะติดตั้ง Raspberry Pi, เซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์, บอร์ด Arduino, โมดูล LoRa, Reaction wheel และ MPU-6050 ดังรูปที่ 3.22

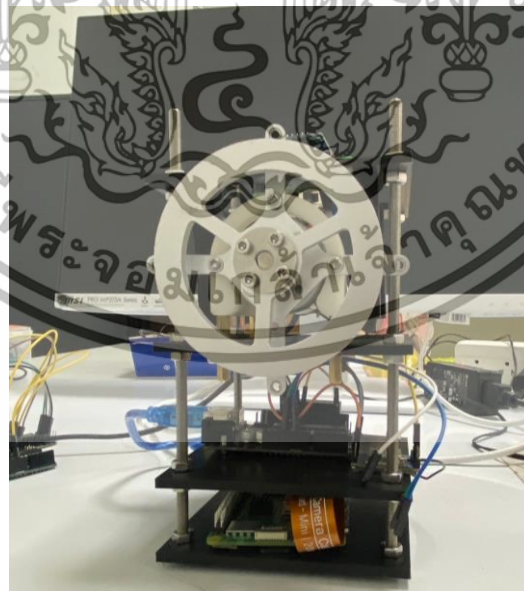


รูปที่ 3.20 พิมพ์โมเดลด้วยเครื่องพิมพ์สามมิติ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.21 โครงสร้างภาพรวมดาวเทียมขนาดเล็กฝังเซ็นเซอร์



รูปที่ 3.22 โครงสร้างภาพรวมดาวเทียมขนาดเล็กฝังมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 3.2 เครื่องมือที่ใช้ในการทดลอง

ในปฏิญานิพนธ์นี้มีอุปกรณ์ และเครื่องมือที่ใช้ในการทดลองดังนี้

### 3.2.1 Raspberry Pi

ปฏิญานิพนธ์นี้เลือกใช้บอร์ด Raspberry Pi 5 แสดงดังรูปที่ 3.23 [21] ซึ่งเป็นบอร์ดรุ่นล่าสุดจาก Cytron Technologies Raspberry Pi เป็น Single Board Computer ที่ถูกพัฒนาเพื่อส่งเสริมการสอนวิทยาการคอมพิวเตอร์ขั้นพื้นฐาน Raspberry Pi เป็นไมโครคอมพิวเตอร์ที่มีความน่าเชื่อถือ โดยตัวบอร์ด Raspberry Pi 5 เลือกใช้งาน CPU แบบ ARM Cortex-A76 ที่สามารถโอเวอร์คล็อกความเร็วได้สูงสุด 2.4 จิกะเฮิร์ตซ์ มาพร้อมกับ GPU VideoCore VII ทำให้ประสิทธิภาพการทำงานของตัวบอร์ด Raspberry Pi 5 นั้นเร็วกว่า Raspberry Pi รุ่นก่อน ซึ่ง Raspberry Pi 5 นั้นเหมาะสำหรับการใช้งานที่ต้องการใช้พลังการประมวลผลสูงซึ่งมีรายละเอียดดังนี้

- 1) หน่วยประมวลผล
  - Broadcom BCM2712
  - 64 บิต ARM Cortex-A76 (ARMv8.2)
  - Quad-core
  - 16 นาโนเมตร Processor SoC
  - แคช L2 ขนาด 512 กิโลไบต์ต่อคอร์ และแคช L3 ขนาด 2 เมกะไบต์ที่ใช้ร่วมกัน
  - ตัวเครื่องผลิตจากโลหะเพื่อการระบายความร้อนที่ดีขึ้น
- 2) SDRAM ใช้ LPDDR4X-4267
- 3) พอร์ตเชื่อมต่อ
  - x พอร์ต USB 3.0 รองรับการถ่ายโอนข้อมูลที่ 5 จิกะบิตต่อวินาที พร้อมกัน
  - x พอร์ต USB 2.0
- 4) GPU
  - VideoCore VII GPU รองรับ OpenGL ES 3.1 และ Vulkan 1.2
  - พอร์ต micro-HDMI แบบคู่ รองรับความละเอียด 4Kp60
  - ตัวถอดรหัส HEVC 4Kp60
- 5) การเชื่อมต่อภาพ/กล้อง 2 x 4-lane MIPI 22-way CSI/DSI port

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 6) พื้นที่จัดเก็บ

- ช่องใส่ MicroSD รองรับโหมดความเร็วสูง SDR104
- ช่องเชื่อมต่อเพิ่มเติมผ่าน PCIe socket รองรับ NVMe SSD

## 7) เครือข่าย

- Ethernet แบบ True Gigabit พร้อมรองรับ PoE+ HAT
- การเชื่อมต่อไร้สายรองรับ 2.4 จิกะเฮิร์ตซ์ และ 5 จิกะเฮิร์ตซ์ IEEE 802.11.b/g/n/ac LAN ไร้สาย , Bluetooth 5.0 และ BLE (Bluetooth Low Energy)

## 8) อินเทอร์เฟซ PCIe 2.0 x 1 รองรับอุปกรณ์ต่อพ่วงความเร็วสูง เช่น NVMe SSD

## 9) พอร์ต UART สำหรับการดีบั๊กโดยเฉพาะ แยกจากพอร์ต 40 พิน GPIO

## 10) พอร์ตพัลส์สำหรับการระบายความร้อน รองรับ PWM และการตอบกลับของ Tachometer

## 11) ส่วนหัว (Header) GPIO 40 พิน

- มีพอร์ต GPIO จำนวน 27 พอร์ต
- UART
- I2C bus, SPI bus พร้อมสองตัวเลือกชิป
- แรงดันไฟฟ้า +3.3 โวลต์, +5 โวลต์ และ Ground

## 12) ระบบ RTC (Real-Time Clock) ที่ถูกติดตั้งบน PMIC (Power Management IC) มาพร้อมกับช่องเชื่อมต่อสำหรับแบตเตอรี่แบบเหรียญที่สามารถชาร์จใหม่ได้

## 13) ปุ่มพาวเวอร์สำหรับการปิดเครื่องที่ถูกต้อง การรีเซ็ต หรือการเปิดเครื่องใหม่

## 14) แหล่งจ่ายไฟ 5 โวลต์/5.0 แอมป์ ผ่านตัวเชื่อมต่อ USB type C โดยรองรับเทคโนโลยี PD (Power Delivery)

## 15) สามารถบูตได้ด้วยระบบปฏิบัติการ Raspberry Pi OS Bookworm

## 16) ขนาดของบอร์ดคือ 88 มิลลิเมตร x 56 มิลลิเมตร x 17 มิลลิเมตร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.23 รูปบอร์ด Raspberry Pi 5

ตารางที่ 3.1 รายละเอียดขาของ Raspberry Pi 5

Pin no.	ชื่อ	ฟังก์ชันหลัก	ฟังก์ชันอื่น ๆ
1	3.3 โวลต์	Power	-
2	5 โวลต์	Power	-
3	GPIO 2	SDA1 (I2C1 Data)	GPIO, I2C, ID_SD
4	5 โวลต์	Power	-
5	GPIO 3	SCL1 (I2C1 Clock)	GPIO, I2C, ID_SC
6	GND	Ground	-
7	GPIO 4	GPIO	GPCLK0, I2C
8	GPIO 14	UART0_TXD	UART Transmit, GPIO
9	GND	Ground	-
10	GPIO 15	UART0_RXD	UART Receive, GPIO
11	GPIO 17	GPIO	SPI1_CE1_N
12	GPIO 18	PCM_CLK	GPIO, PWM0, SPI1_CE0_N
13	GPIO 27	GPIO	PCM_DIN, I2C, SPI
14	GND	Ground	-
15	GPIO 22	GPIO	-
16	GPIO 23	GPIO	-
17	3.3V	Power	-
18	GPIO 24	GPIO	-

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin no.	ชื่อ	ฟังก์ชันหลัก	ฟังก์ชันอื่น ๆ
19	GPIO 10	SPI0_MOSI	GPIO, I2C, SPI
20	GND	Ground	-
21	GPIO 9	SPI0_MISO	GPIO, I2C, SPI
22	GPIO 25	GPIO	-
23	GPIO 11	SPI0_SCLK	GPIO, I2C, SPI
24	GPIO 8	SPI0_CE0_N	GPIO, I2C, SPI
25	GND	Ground	-
26	GPIO 7	SPI0_CE1_N	GPIO, I2C, SPI
27	GPIO 0	ID_SD	GPIO, I2C, EEPROM I2C Data
28	GPIO 1	ID_SC	GPIO, I2C, EEPROM I2C Clock
29	GPIO 5	GPIO	
30	GND	Ground	-
31	GPIO 6	GPIO	
32	GPIO 12	PWM0	GPIO, PCM_DOUT
33	GPIO 13	PWM1	GPIO, PCM_DIN
34	GND	Ground	-
35	GPIO 19	PCM_FS	GPIO, SPI1_MISO
36	GPIO 16	GPIO	SPI1_CE2_N
37	GPIO 26	GPIO	-
38	GPIO 20	PCM_DOUT	GPIO, SPI1_MOSI
39	GND	Ground	-
40	GPIO 21	PCM_DIN	GPIO, SPI1_SCLK

### 3.2.2 Raspberry Pi Camera

โครงการนี้เลือกใช้ Raspberry Pi Camera Module 1.3 แสดงดังรูปที่ 3.24 [22] ซึ่งเป็นโมดูลกล้องที่มีความสามารถในการถ่ายภาพที่ความละเอียด 5 ล้านพิกเซล จากเซ็นเซอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

OmniVision OV5647 โดยมีขนาดเล็ก และน้ำหนักเบา เหมาะสำหรับโปรเจกต์ถ่ายภาพ และวิดีโอผ่าน Raspberry Pi ซึ่งรายละเอียดของโมดูลนี้มีดังนี้

- 1) ใช้เซ็นเซอร์รับภาพแบบ CMOS 5-megapixel (OmniVision OV5647)
- 2) ความละเอียดภาพถ่ายที่ความละเอียดสูงสุด 2592 x 1944 พิกเซล
- 3) การถ่ายวิดีโอ รองรับการถ่ายวิดีโอที่ความละเอียด
  - 1080p ที่ 30 เฟรมต่อวินาที
  - 720p ที่ 60 เฟรมต่อวินาที
  - 480p ที่ 90 เฟรมต่อวินาที
- 4) เลนส์มีค่ารับแสง f/2.9 และความยาวโฟกัส 3.6 มม. มุมมองภาพ (Field of View) คือ 53.5 องศาในแนวนอน และ 41.41 องศาในแนวตั้ง
- 5) การเชื่อมต่อใช้สายเคเบิลแบบแบน (FFC) 15พิน สำหรับเชื่อมต่อกับพอร์ต CSI (Camera Serial Interface) บน Raspberry Pi
- 6) เซ็นเซอร์มีขนาด 1/4 นิ้ว และขนาดพิกเซลที่ 1.4 ไมโครเมตร x 1.4 ไมโครเมตร
- 7) รองรับการทำงานกับ Raspberry Pi ทุกรุ่นที่มีพอร์ต CSI
- 8) สามารถใช้งานร่วมกับไลบรารีเช่น Picamera หรือ libcamera ของภาษา Python สำหรับการถ่ายภาพ และควบคุมกล้อง
- 9) ตัวโมดูลมีขนาด 25 มิลลิเมตร x 24 มิลลิเมตร x 9 มิลลิเมตร และน้ำหนัก 3 กรัม
- 10) โมดูลนี้ใช้พลังงานต่ำ เหมาะสำหรับโปรเจกต์ที่ต้องการประหยัดพลังงาน เช่น ระบบเฝ้าระวังหรือการใช้งานระยะไกล



รูปที่ 3.24 รูป Pi Camera Module 1.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.3 ESP32-S2 Mini

โครงการนี้ใช้บอร์ด ESP32-S2 Mini ซึ่งใช้ชิป ESP32-S2 เป็นชิปรุ่นใหม่จาก Espressif มีความสามารถที่เพิ่มขึ้น และลดลงจาก ESP32 รุ่นปกติ มาพร้อมหน่วยประมวลผลสถาปัตยกรรม Xtensa 32-bit LX7 ทำงานที่ความถี่ 240 เมกะเฮิร์ตซ์ รมม 4 เมกะไบต์ และแรม 320 กิโลไบต์ (ภายใน) มาพร้อม WiFi 2.4 จิกะเฮิร์ตซ์ มาตรฐาน b/g/n เชื่อมต่ออุปกรณ์ภายนอกผ่าน GPIO, SPI, LCD, UART, I2C, I2S, Camera interface, IR, pulse counter, LED PWM, USB OTG 1.1, ADC และDAC อัปโหลดโปรแกรมตรงผ่าน USB Type C เข้าชิป ESP32-S2 เขียนโปรแกรมสั่งงานได้ด้วย Arduino IDE, PlatformIO และ MicroPython โดยมีรายละเอียดขาของ ESP32-S2-MINI แสดงดังตารางที่ 3.2 และบอร์ด ESP32-S2 Mini แสดงดังรูปที่ 3.25 [23]

รูปที่ 3.25 รูปบอร์ด ESP32-S2 Mini

ตารางที่ 3.2 รายละเอียดขาของ ESP32-S2-MINI

Pin no.	ชื่อ	ฟังก์ชันหลัก	ฟังก์ชันอื่น ๆ
1	GND	Ground	
2	3.3 โวลต์	Power Supply	
3	EN	Enable	
4	IO36	GPIO36	ADC1_CH0
5	IO37	GPIO37	ADC1_CH1
6	IO38	GPIO38	ADC1_CH2
7	IO39	GPIO39	ADC1_CH3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin no.	ชื่อ	ฟังก์ชันหลัก	ฟังก์ชันอื่น ๆ
8	IO34	GPIO34	ADC1_CH4
9	IO35	GPIO35	ADC1_CH5
10	IO32	GPIO32	ADC1_CH6
11	IO33	GPIO33	ADC1_CH7
12	IO25	GPIO25	DAC1, ADC2_CH8
13	IO26	GPIO26	DAC2, ADC2_CH9
14	IO27	GPIO27	ADC2_CH7
15	IO14	GPIO14	ADC2_CH6
16	IO12	GPIO12	ADC2_CH5
17	IO13	GPIO13	ADC2_CH4
18	IO9	GPIO9	
19	IO10	GPIO10	
20	IO11	GPIO11	
21	IO6	GPIO6	
22	IO7	GPIO7	
23	IO8	GPIO8	
24	IO1	GPIO1	UART0_TXD
25	IO2	GPIO2	UART0_RXD
26	IO3	GPIO3	UART0_RTS
27	IO4	GPIO4	UART0_CTS
28	IO5	GPIO5	
29	IO21	GPIO21	
30	IO19	GPIO19	
31	IO18	GPIO18	
32	IO17	GPIO17	
33	IO16	GPIO16	
34	IO15	GPIO15	

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Pin no.	ชื่อ	ฟังก์ชันหลัก	ฟังก์ชันอื่น ๆ
35	IO0	GPIO0	
36	IO22	GPIO22	
37	IO23	GPIO23	

รายละเอียดเพิ่มเติมจากตารางที่ 3.2 มีดังนี้

- 1) GPIO (General Purpose Input/Output) ใช้สำหรับการตั้งค่าให้เป็นขารับหรือขาส่งสัญญาณได้ตามความต้องการของผู้ใช้
- 2) ADC (Analog to Digital Converter) ใช้สำหรับแปลงสัญญาณอนาล็อกเป็นสัญญาณดิจิทัล
- 3) DAC (Digital to Analog Converter) ใช้สำหรับแปลงสัญญาณดิจิทัลเป็นสัญญาณอนาล็อก
- 4) UART (Universal Asynchronous Receiver-Transmitter) ใช้สำหรับการสื่อสารแบบอนุกรม

#### 3.2.4 Servo Motor

เซอร์โวมอเตอร์ (Servo Motor) คือมอเตอร์ที่มีระบบควบคุมตำแหน่งอย่างแม่นยำ โดยภายในมีวงจรควบคุมที่สามารถปรับตำแหน่งของแกนมอเตอร์ตามสัญญาณควบคุมที่ได้รับ ทำให้สามารถหมุนไปยังตำแหน่งที่ต้องการได้อย่างรวดเร็ว และแม่นยำ มักใช้ในงานที่ต้องการการควบคุมตำแหน่ง ดังในรูปที่ 3.26 [24] และการเชื่อมต่อพอร์ตระหว่าง Servo Motor และ ESP32-S2-MINI ดังในตารางที่ 3.3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.26 รูป Servo Motor

## ตารางที่ 3.3 การเชื่อมต่อพอร์ตระหว่าง Servo Motor และ ESP32-S2-MINI

Servo-Motor	ESP32-S2 Mini
VCC (5 โวลต์)	5 โวลต์ (หรือแหล่งจ่ายไฟภายนอก)
GND	GND
Signal (สัญญาณ)	GPIO18 (หรือพิน GPIO อื่น ๆ ที่ต้องการใช้)

## 3.2.5 LoRa module 433 เมกะเฮิร์ตซ์ Ra-02 SX1278

ปริญญาณิพนธ์นี้ใช้โมดูลส่งข้อมูลไร้สายระยะไกล ความถี่ 433 เมกะเฮิร์ตซ์ ใช้ไฟเลี้ยง 1.8-3.7 โวลต์ สามารถตั้งการทำงานเป็นตัวส่งข้อมูลหรือจะทำงานเป็น เกตเวย์ คอยรับข้อมูลเพื่อส่งเข้า อินเทอร์เน็ต โดยใช้ ESP8266, ESP32 หรือ Raspberry Pi สื่อสารด้วย SPI โดยมีรายละเอียดดังนี้

- 1) ระยะการสื่อสารไกลพิเศษ 15 กิโลเมตร
- 2) กำลังส่ง RF คงที่แม้แรงดันไฟฟ้าจะเปลี่ยนแปลงในช่วง +20 เดซิเบลมิลลิวัตต์ - 100 มิลลิวัตต์
- 3) ความไวสูง: ต่ำสุด -148 เดซิเบลมิลลิวัตต์
- 4) การสื่อสารแบบ Half-duplex SPI
- 5) อัตราบิตที่ตั้งโปรแกรมได้สูงสุด 300 กิโลบิตต่อวินาที
- 6) รองรับรูปแบบการมอดูเลต FSK, GFSK, MSK, GMSK, LoRa™ และ OOK

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- 7) ช่วงไดนามิก RSSI 127dB
  - 8) การตรวจจับสัญญาณ RF อัตโนมัติ, โหมด CAD และ AFC ความเร็วสูงพิเศษ
  - 9) เครื่องมือประมวลผลแพ็กเก็ตพร้อม CRC รองรับสูงสุด 256 ไบต์
  - 10) ขนาดเล็กแบบแพ็กเกจแผ่นวงจร พร้อมฝาครอบป้องกันสัญญาณรบกวน
  - 11) เสออากาศแบบสปริง (Ra-01) หรือเสออากาศภายนอกแบบIPEX (Ra-02)
- แสดงโมดูล LoRa ส่งข้อมูลไร้สายระยะไกล ความถี่ 433 เมกะเฮิร์ตซ์ ดังรูปที่ 3.27 [25] และตารางการเชื่อมต่อระหว่างต่อโมดูล LoRa และ Arduino ดังตารางที่ 3.4



รูปที่ 3.27 โมดูล LoRa ส่งข้อมูลไร้สายระยะไกล ความถี่ 433 เมกะเฮิร์ตซ์  
ตารางที่ 3.4 รายละเอียดการต่อระหว่าง LoRa และ Arduino

LoRa Pin	Arduino Pin
NSS (CS)	Pin 10
RESET	Pin 9
DIO0	Pin 2
MISO	Pin 12
MOSI	Pin 11
SCK	Pin 13
GND	GND
3.3 โวลต์	3.3 โวลต์

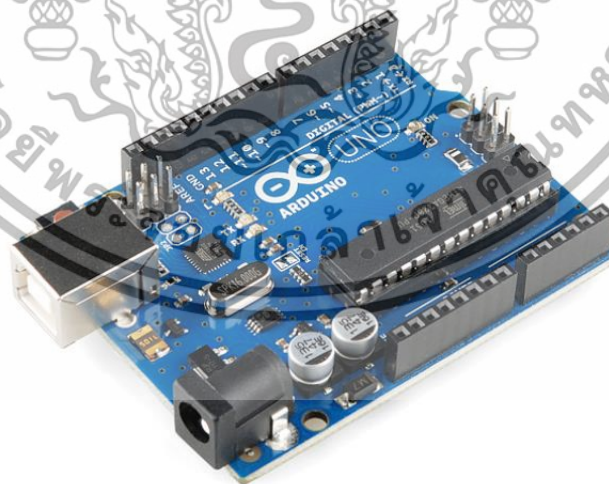
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.6 Arduino UNO

โครงการนี้เลือกใช้บอร์ด Arduino UNO ซึ่งเป็นบอร์ด Arduino ที่ได้รับความนิยมมากเนื่องจากเป็นขนาดที่มีเหมาะสำหรับผู้เริ่มต้นเรียนรู้ Arduino และง่ายต่อการพัฒนาเพราะมีรูปแบบคำสั่งพื้นฐาน การใช้งาน ที่ไม่ซับซ้อน และราคาไม่แพง โดยบอร์ด Arduino รุ่น UNO สามารถใช้งานได้กับโมดูล และเซ็นเซอร์ได้หลายชนิดซึ่งมีรายละเอียดดังนี้

- 1) ใช้ชิปไอซีไมโครคอนโทรลเลอร์ ATmega328P
- 2) มีขาดิจิทัลอินพุต/เอาต์พุต 14 ขา ( รองรับเอาต์พุตแบบ PWM ได้ 6 ขา )
- 3) มีขาแอนะล็อกอินพุต 6 ขา
- 4) ขนาดบอร์ด 68.8 x 53.4 มิลลิเมตร
- 5) มีน้ำหนัก 25 กรัม
- 6) ใช้แรงดันไฟฟ้า 5 โวลต์
- 7) พื้นที่โปรแกรมภายใน 32 กิโลบิต และทำงานที่ความถี่ 16 เมกะเฮิรตซ์
- 8) รองรับการจ่ายแรงดันไฟฟ้า 7 - 12 โวลต์
- 9) กระแสไฟที่จ่ายได้ในแต่ละพอร์ต อยู่ที่ 40 มิลลิแอมป์
- 10) กระแสไฟที่จ่ายได้ในพอร์ต 3.3 โวลต์ อยู่ที่ 50 มิลลิแอมป์

ซึ่งบอร์ดสามารถเชื่อมต่อกับคอมพิวเตอร์ด้วยสายเคเบิล USB ดังในรูปที่ 3.28 [26]



รูปที่ 3.28 บอร์ด Arduino UNO

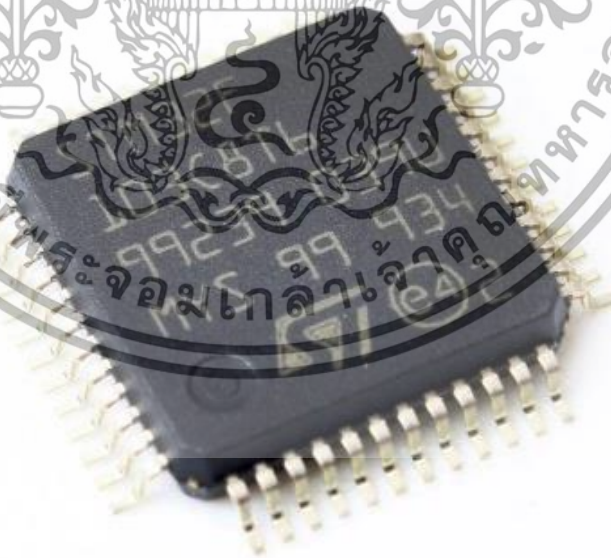
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.7 ไมโครคอนโทรลเลอร์ STM32 32-bit Arm Cortex MCUs

ใช้ไมโครคอนโทรลเลอร์ STM32 รุ่นชิป STM32F103C8T6 เนื่องจากสามารถใช้งานได้หลากหลาย เช่น ระบบขับเคลื่อนมอเตอร์, ควบคุมอุปกรณ์พกพา, ใช้งานในอุตสาหกรรมและอื่น ๆ โดยมีรายละเอียดดังนี้

- 1) หน่วยประมวลผล Arm® Cortex®-M3 แบบ 32 บิต
- 2) ความเร็วสูงสุด 72 เมกะเฮิรตซ์
- 3) ช่วงแรงดันไฟฟ้า 2.0 – 3.6 โวลต์ สำหรับระบบ
- 4) วงจรตรวจจับแรงดันไฟฟ้า (POR, PDR และPVD)
- 5) รองรับโหมดพลังงานต่ำ (โหมด Sleep, Stop และ Standby)
- 6) รองรับการใช้งานตัวจับเวลา (Timers), ADC, SPIs, I<sup>2</sup>Cs และ USARTs
- 7) โหมดดีบั๊กผ่าน Serial wire debug (SWD) และ JTAG อินเทอร์เฟซ
- 8) มีการคำนวณตรวจสอบข้อผิดพลาดแบบวนซ้ำ (CRC)
- 9) มี Timer สำหรับควบคุมมอเตอร์แบบ 16 บิต พร้อมฟังก์ชันต่าง ๆ
- 10) มีเซ็นเซอร์อุณหภูมิภายใน

แสดงไมโครคอนโทรลเลอร์ STM32 รุ่นชิป STM32F103C8T6 ดังรูปที่ 3.29 [27]



รูปที่ 3.29 ไมโครคอนโทรลเลอร์ STM32 รุ่นชิป STM32F103C8T6

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.8 ไมโครคอนโทรลเลอร์ MPU-6050

ไมโครคอนโทรลเลอร์ MPU-6050 หน่วยวัดความเฉื่อยหรือ IMU (Inertial Measurement Units) MPU-6050 นี้มีคุณสมบัติของไจโรสโคป และเครื่องวัดความเร่ง โดยวัดได้ทั้ง 3 แกน X, Y และ Z สำหรับการควบคุมทิศทาง โดยมีรายละเอียดดังนี้

- 1) ประเภทอินเทอร์เฟซ I2C
- 2) ค่าอุณหภูมิ -40 - 105 องศาเซลเซียส
- 3) ช่วงการจ่ายแรงดันไฟฟ้า 2.375 - 3.46 โวลต์
- 4) ตรวจจับวัดทิศทางได้ 3 แกน X, Y และ Z
- 5) กระแสไฟทำงานของไจโรสโคป 3.6 มิลลิแอมป์
- 6) กระแสไฟทำงานของเครื่องวัดความเร่ง 500 ไมโครแอมป์
- 7) การป้องกันไฟฟ้าสถิต (ESD) 2 กิโลโวลต์ (HBM); 250 โวลต์ (MM)

แสดงไมโครคอนโทรลเลอร์ MPU-6050 ดังรูปที่ 3.30 [28]



รูปที่ 3.30 บอร์ดไมโครคอนโทรลเลอร์ MPU-6050

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.2.9 DC Motor

เลือกใช้มอเตอร์ไฟฟ้ากระแสตรง (DC Motor) คือมอเตอร์ที่ทำหน้าที่แปลงพลังงานไฟฟ้าเป็นพลังงานกล สามารถนำไปใช้งานได้หลายรูปแบบดังรูปที่ 3.31 [29] โดยมีรายละเอียดดังนี้

- 1) รุ่น RS-545
- 2) เส้นผ่านศูนย์กลางมอเตอร์ 35.8 มิลลิเมตร
- 3) ความสูงของมอเตอร์ 50.2 มิลลิเมตร
- 4) น้ำหนัก 163 กรัม
- 5) แรงดันไฟฟ้า 12 โวลต์
- 6) ความเร็วรอบ 11,000 รอบต่อนาที



รูปที่ 3.31 มอเตอร์ไฟฟ้ากระแสตรง

### 3.2.10 Filament หรือเส้นพลาสติกสำหรับเครื่องพิมพ์ 3D Printer ชนิด PLA

PLA เป็นเส้นพลาสติกที่ถูกใช้งานมากที่สุดในเครื่องพิมพ์แบบสามมิติเส้นพลาสติก PLA เป็นเส้นประเภท Biodegradable ซึ่งคือเส้นที่ทำมาจากพืช เช่น มันสำปะหลัง หรือข้าวโพด เป็นเส้นที่ไม่เป็นพิษภัยต่อสิ่งแวดล้อม เพราะสามารถย่อยสลายได้ เส้นพลาสติก PLA ที่มีความแข็งแรงมาก และเป็นพลาสติกที่เหมาะสมกับพิมพ์งานที่มีขนาดใหญ่ เพราะตัวเส้นมีค่าการหดตัวต่ำ (Thermal Expansion) ทำให้พิมพ์ง่าย เกาะฐานพิมพ์นอกจากพิมพ์ชิ้นงานขนาดใหญ่ได้แล้ว ยังเหมาะกับพิมพ์งานเล็ก ๆ ที่มีรายละเอียดเยอะ ๆ โดยเส้นพลาสติก Filament PLA ดังรูปที่ 3.32 [30]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.32 รูปเส้นพลาสติก Filament PLA

### 3.3 โปรแกรมที่ใช้ในการทดลอง

#### 3.3.1 โปรแกรมสร้าง 3D print

โปรแกรม Shapr3D เป็นโปรแกรมสำหรับการออกแบบรูปสามมิติ สำหรับผู้ใช้ระดับกลางถึงมืออาชีพ โปรแกรมยังสามารถทำงานร่วมกับโปรแกรม CAD หลัก ๆ ได้เกือบทุกโปรแกรม ทั้ง SolidWorks, AutoCAD, Fusion360, Rhino3D และอื่น ๆ นอกจากนี้ยังสามารถนำเข้าไฟล์สกุล .STEP หรือ .IGES และนำออกไฟล์สกุลเดียวกันหรือ STL ที่สามารถเข้าได้กับโปรแกรมปรับแต่ง CAD อื่น ๆ ดังในรูปที่ 3.33

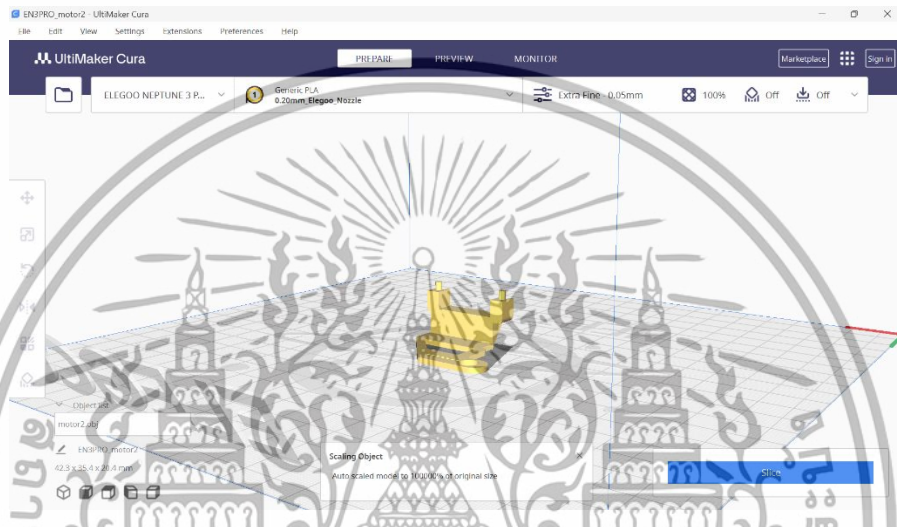


รูปที่ 3.33 โปรแกรม Shapr3D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.2 โปรแกรมที่ใช้กับ 3D printer

โปรแกรม UltiMaker Cura 5.6.0 เป็นโปรแกรมสำหรับเตรียมไฟล์งานพิมพ์สามมิติ ก่อนนำไปพิมพ์ที่เครื่องพิมพ์สามมิติ โดยสามารถทำงานร่วมกับโปรแกรม Shapr3D ที่ทำการออกแบบได้ [31] โปรแกรม UltiMaker Cura 5.6.0 ดังในรูปที่ 3.34



รูปที่ 3.34 โปรแกรม UltiMaker Cura 5.6.0

### 3.3.3 โปรแกรมเขียนโปรแกรมควบคุม

โปรแกรม Arduino IDE เป็นโปรแกรมในการใช้งานลักษณะ Open source ซึ่ง Arduino IDE จะทำหน้าที่ติดต่อระหว่างคอมพิวเตอร์ ไม่ว่าจะ เป็นระบบ Windows, Mac OS X หรือ Linux กับบอร์ด ESP32-S2 Mini ซึ่งโปรแกรมนี้ ออกแบบให้ง่ายต่อการเขียนโค้ด และอัปโหลด โปรแกรมที่เราเขียนเข้าสู่บอร์ด ESP32-S2 Mini [32] โปรแกรม Arduino IDE ดังในรูปที่ 3.35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

sketch_motor.ino
1 #define Servo1 10
2 #define Servo2 13
3
4
5 volatile int angle_servo1 = 0;
6 volatile int angle_servo2 = 0;
7
8 void setup() {
9   Serial.begin(115200);
10
11   ledAttachChannel(servo1, 200, 10, 1);
12   ledAttachChannel(servo2, 200, 10, 2);
13
14   xTaskCreate(SerialTask, "SerialTask", 1000, NULL, 1, NULL);
15   xTaskCreate(servo_1, "servo1", 1000, NULL, 1, NULL);
16   xTaskCreate(servo_2, "servo2", 1000, NULL, 1, NULL);
17
18   // ledwrite(10, (2.7222222 * 0) + 112.53);
19   // ledwrite(13, (2.7222222 * 0) + 112.53);
20 }
21
22 void SerialTask(void *pvParameters) {
23   (void)pvParameters;
24
25   while (1) {
26     if (Serial.available() > 0) {
27       String input_angle = Serial.readStringUntil('\n');
28       char servo_id = input_angle.charAt(0);
29       int angle = input_angle.substring(1).toInt();
30
31       if (servo_id == 'a' || servo_id == 'b') {

```

รูปที่ 3.35 โปรแกรม Arduino IDE

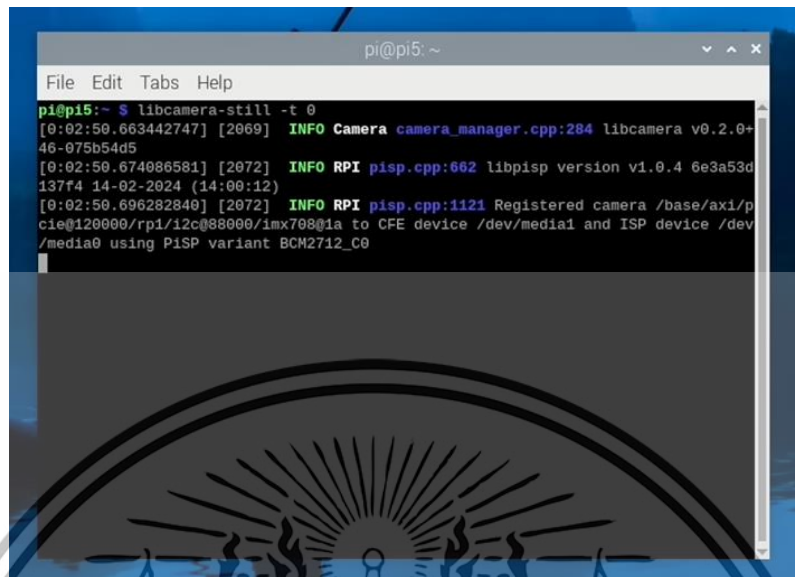
### 3.3.4 โปรแกรมเขียนโปรแกรมเพื่อควบคุมการหมุนของมอเตอร์

การควบคุมมอเตอร์ด้วย FreeRTOS และ PWM บนแพลตฟอร์ม ESP32 เป็นวิธีที่มีประสิทธิภาพในการจัดการมอเตอร์หลายตัวในระบบที่ต้องการการทำงานแบบ multitasking โดยเริ่มจากการกำหนดพิน ตั้งค่าความถี่ และความละเอียดของสัญญาณ PWM เพื่อควบคุมมุมการหมุนของมอเตอร์อย่างแม่นยำ จากนั้น FreeRTOS ถูกนำมาใช้ในการสร้าง Task สามตัวที่ทำงานอิสระ โดย Task แรกอ่านค่ามุมจาก Serial input และจัดเก็บในตัวแปร Task ที่สอง และสามใช้ค่านีเพื่อปรับสัญญาณ PWM และควบคุมการหมุนของมอเตอร์ นอกจากนี้มอเตอร์สองตัวยังคงควบคุมมุม Azimuth และมุม Elevation ของเซ็นเซอร์วัดทิศทางแสงอาทิตย์ตามค่าที่ได้รับ ทำให้ระบบนี้สามารถควบคุมการทำงานหลายอย่างพร้อมกันได้อย่างมีประสิทธิภาพ

### 3.3.5 โปรแกรมสำหรับใช้ร่วมกับ Pi Camera

การใช้งานเทอร์มินอลของ Raspberry Pi 5 ในการใช้งานร่วมกับ Pi Camera และ Raspberry Pi 5 เทอร์มินอลใช้งานสำหรับการสั่งงานด้วยคำสั่งแบบบรรทัด (command-line interface) โดยใช้คำสั่ง libcamera-hello-t0 เพื่อแสดงภาพจากกล้องขึ้นมาบนหน้าจอแบบเรียลไทม์โดยมีดีเลย์น้อยมากดังในรูปที่ 3.36

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



```

pi@pi5: ~
File Edit Tabs Help
pi@pi5:~ $ libcamera-still -t 0
[0:02:50.663442747] [2069] INFO Camera camera_manager.cpp:284 libcamera v0.2.0+
46-075b54d5
[0:02:50.674086581] [2072] INFO RPI pisp.cpp:662 libpisp version v1.0.4 6e3a53d
137f4 14-02-2024 (14:00:12)
[0:02:50.696282840] [2072] INFO RPI pisp.cpp:1121 Registered camera /base/axi/p
cie@120000/rp1/i2c@88000/imx708@1a to CFE device /dev/media1 and ISP device /dev
/media0 using PiSP variant BCM2712_C0

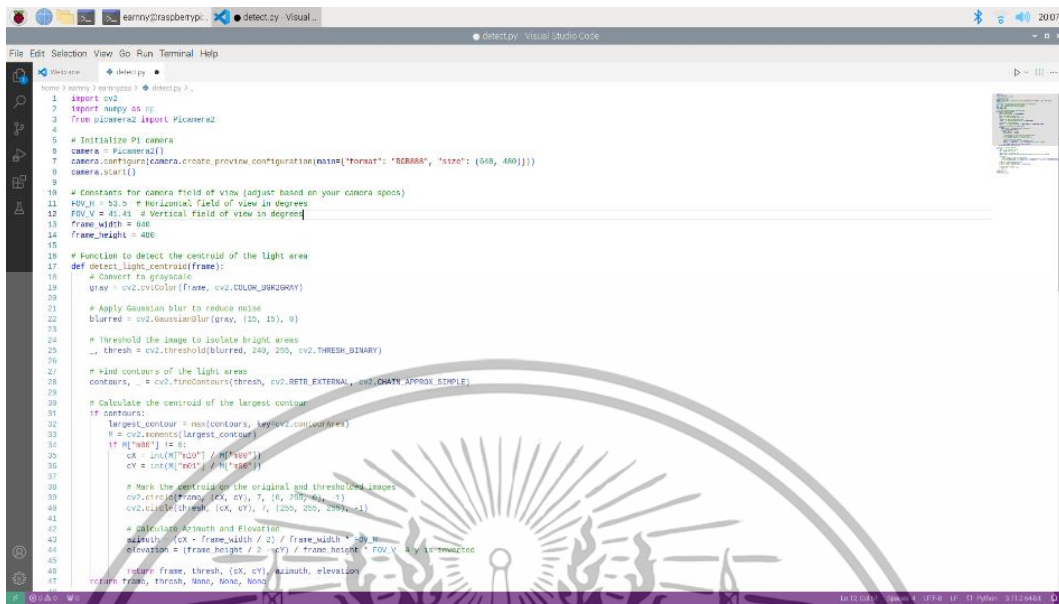
```

รูปที่ 3.36 เทอร์มินอลของ Raspberry Pi 5

### 3.3.6 โปรแกรมคำนวณมุม Visual Studio Code

โปรแกรม Visual Studio Code หรือ VS Code คือ โปรแกรม Code Editor ของ Microsoft ที่ใช้สำหรับเขียน แก้ไข และตรวจสอบความผิดพลาดของโค้ด คุณสมบัติของโปรแกรมมีลักษณะเบา มีประสิทธิภาพสูง มีส่วนขยาย (Extension) เพื่อเพิ่มขีดความสามารถ และสามารถเขียนโค้ดได้หลากหลายภาษา ไม่ว่าจะเป็น JavaScript, C#, Java, C++, Python, HTML หรือ PHP เป็นต้น [33] โปรแกรม Visual Studio Code ดังในรูปที่ 3.37

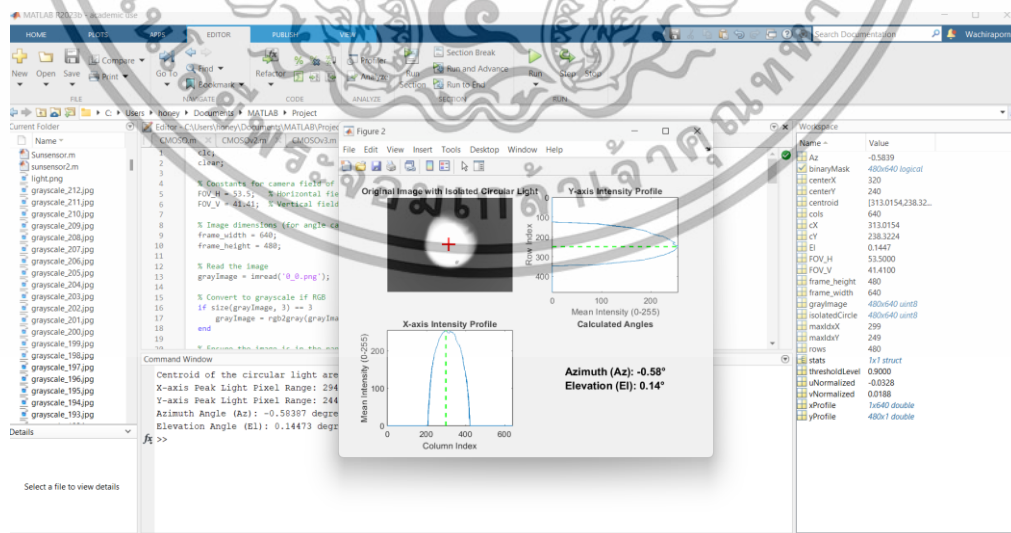
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.37 โปรแกรม Visual Studio Code

### 3.3.7 โปรแกรมคำนวณมุม MATLAB

โปรแกรม MATLAB เป็นแพลตฟอร์มการเขียนโปรแกรมที่ออกแบบมาเฉพาะสำหรับวิศวกร และนักวิทยาศาสตร์ โดยสิ่งสำคัญหลักของ MATLAB คือ ภาษา (Code) ซึ่งใช้เมทริกซ์ช่วยให้สามารถคำนวณทางคณิตศาสตร์ได้เป็นธรรมชาติที่สุด [34] โปรแกรม MATLAB ดังในรูปที่ 3.38

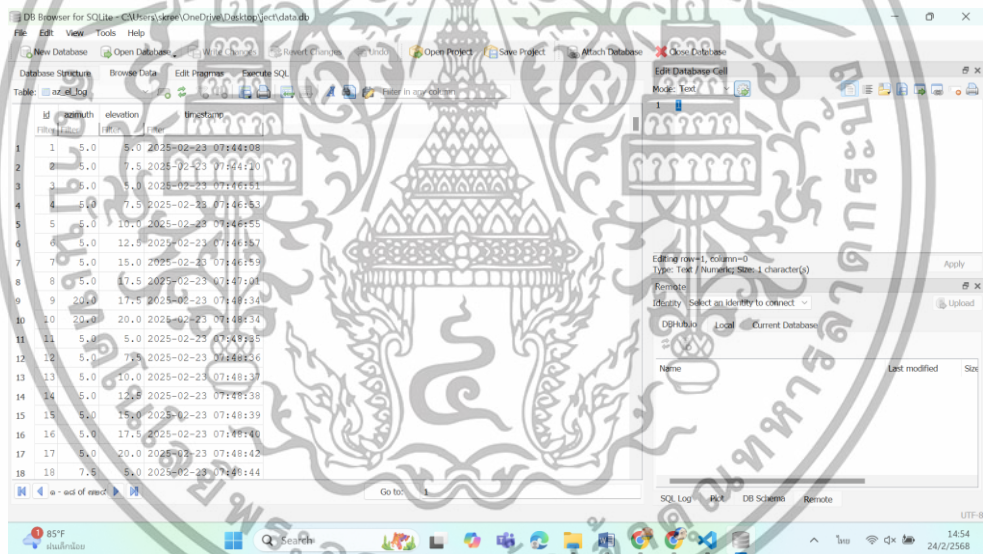


รูปที่ 3.38 โปรแกรม MATLAB

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.3.8 โปรแกรมจัดการฐานข้อมูล DB Browser for SQLite

โปรแกรม DB Browser for SQLite ดังในรูปที่ 3.39 เป็นโปรแกรมที่ใช้ในการจัดการฐานข้อมูล SQLite โดยมีอินเทอร์เฟซแบบกราฟิก (GUI) ที่ช่วยให้ผู้ใช้งานสามารถสร้าง แก้ไข และจัดการฐานข้อมูล SQLite ได้ง่ายขึ้น โดยไม่ต้องใช้คำสั่ง SQL ผ่าน Command Line คุณสมบัติหลักของ DB Browser for SQLite คือ สร้าง และจัดการฐานข้อมูล SQLite สามารถสร้างและแก้ไขตารางในฐานข้อมูล จัดการข้อมูล (Insert, Update, Delete) สามารถรันคำสั่ง SQL ได้โดยตรง นำเข้า (Import) และส่งออก (Export) ข้อมูล อีกทั้งยังใช้งานร่วมกับโปรแกรมอื่น ๆ ได้ง่าย โดยในงานชิ้นนี้ได้นำโปรแกรมนี้มาใช้ในการ สร้างฐานข้อมูล SQLite สำหรับเก็บค่ามุม Azimuth และ Elevation เพิ่มข้อมูลที่ได้รับจาก Raspberry Pi 5 ผ่าน Flask API เรียกดูค่ามุมล่าสุดที่บันทึกไว้ และนำมาใช้ในหน้าเว็บ [35]



รูปที่ 3.39 โปรแกรม DB Browser for SQLite

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4 การจัดเก็บผลการทดลอง

#### 3.4.1 การทดสอบการใช้งานชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์

3.4.1.1 การทดสอบระบบการหมุนของดาวเทียมขนาดเล็กด้วยมอเตอร์

3.4.1.2 การทดสอบการหมุน และการวัดมุม

#### 3.4.2 การทดสอบถ่ายภาพด้วยชุดทดลอง

การทดสอบถ่ายภาพด้วยชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ โดยเชื่อมต่อ Pi Camera เข้ากับ Raspberry Pi 5 เพื่อใช้ในการถ่ายภาพที่แสงตกกระทบเซ็นเซอร์

#### 3.4.3 การทดสอบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์

การถ่ายภาพการทดสอบในมุม Elevation (EU) และ Azimuth (Az) ต่าง ๆ

#### 3.4.4 การทดสอบการโดยไม่ปรับค่า Threshold และปรับค่า Threshold

3.4.4.1 การทดสอบการทดสอบโปรไฟล์ความเข้มแสง (Intensity Profile)

3.4.4.2 การทดสอบการทำ UV pattern

1) การทดสอบการทำ UV pattern 2D

2) การทดสอบการทำ UV pattern 3D

#### 3.4.5 การทดสอบความแม่นยำ (Accuracy)

การทดสอบความแม่นยำ (Accuracy) ของเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ เปรียบเทียบความคลาดเคลื่อนของมุม Azimuth และ Elevation ระหว่างข้อมูลที่ไม่ได้ปรับค่า Threshold กับข้อมูลที่มีการปรับค่า Threshold

#### 3.4.6 การทดสอบการคำนวณมุม

3.4.6.1 ผลลัพธ์การทดสอบการคำนวณมุมฝั่งภาคส่ง

3.4.6.2 ผลลัพธ์การทดสอบมุมฝั่งภาครับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.4.7 การทดสอบระบบการสื่อสารระหว่างดาวเทียมขนาดเล็กและสถานีภาคพื้นดินแบบไร้สาย ( TC & TM Subsystem )

3.4.7.1 ผลลัพธ์การทดสอบระบบการสื่อสารทางด้านภาคส่ง

3.4.7.2 ผลลัพธ์การทดสอบระบบการสื่อสารทางด้านภาครับ

3.4.7.3 ผลลัพธ์การทดสอบระบบการสื่อสารระหว่างภาคส่ง และภาครับ

### 3.4.8 การทดสอบการรักษาความสมดุลของดาวเทียมขนาดเล็กโดยใช้จيروسโคปและมอเตอร์

3.4.8.1 ผลลัพธ์การทดสอบของจيروسโคป

3.4.8.2 ผลลัพธ์การทดสอบมอเตอร์

### 3.4.9 การทดสอบระบบต่าง ๆ ของดาวเทียมขนาดเล็กเมื่อนำระบบมาใช้งานพร้อมกัน

การทดสอบเมื่อนำระบบที่ศึกษาทั้งหมดมาใช้งานพร้อมกันในดาวเทียมขนาดเล็กเดียวกัน เพื่อทดสอบประสิทธิภาพการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4

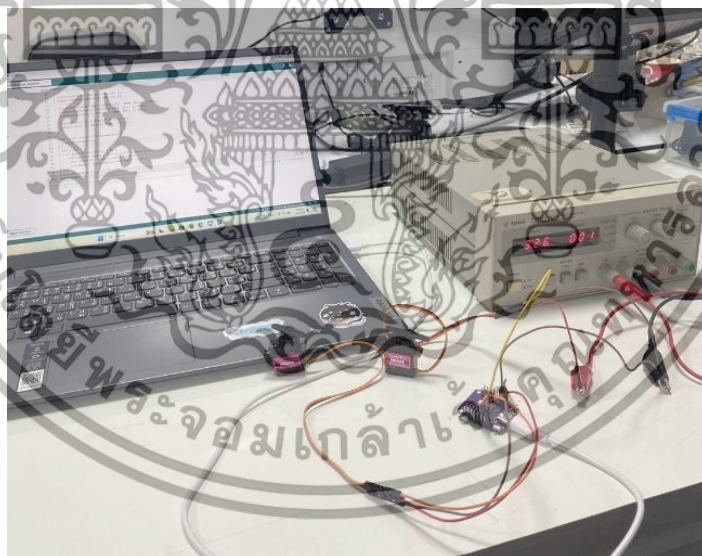
### ผลการทดลอง

ผู้จัดทำได้ทำการทดลอง และรวบรวมผลลัพธ์ที่ได้จากการทำงานของเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ในระบบควบคุมทิศทางของดาวเทียมขนาดเล็กโดยมีการแบ่งการทดสอบออกเป็นส่วนตัวต่าง ๆ และทดสอบผลลัพธ์ที่ได้เพื่อวิเคราะห์ และเปรียบเทียบความแม่นยำการเก็บผลลัพธ์การทำงานของระบบ โดยแบ่งเป็นส่วน ๆ ดังต่อไปนี้

#### 4.1 การทดสอบการใช้งานชุดทดลองเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์

##### 4.1.1 การทดสอบระบบการหมุนของดาวเทียมขนาดเล็กด้วยมอเตอร์

การทดสอบการใช้งานของ Servo Motor ร่วมกับเครื่อง Power Supply โดยควบคุมผ่านโปรแกรม Arduino IDE ดังรูปที่ 4.1



รูปที่ 4.1 รูปการทดลอง Servo Motor ร่วมกับเครื่อง Power Supply โดยควบคุมผ่านโปรแกรม Arduino IDE

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.1.2 การทดสอบการหมุน และการวัดมุม

นำ Servo Motor มาทดสอบการหมุน และการวัดมุมโดยใช้โปรแกรม Arduino IDE ในการเขียนโค้ดควบคุม โปรแกรมจะแสดงผลของการวัดค่ามุม Azimuth (Az) และ Elevation (El) โดยจะแสดงผลค่ามุมบนจอคอมพิวเตอร์ เป็นหน้าต่างของโปรแกรม Arduino IDE โดยจะแสดงผลในส่วนซีเรียลมอนิเตอร์ ดังรูปที่ 4.2

```

sketch_motor.ino
1 #include <Servo.h>
2 #include <Serial.h>
3
4
5 volatile int angle_servo1 = 0;
6 volatile int angle_servo2 = 0;
7
8 void setup() {
9   Serial.begin(115200);
10
11   pinMode(10, OUTPUT);
12   pinMode(13, OUTPUT);
13
14   attachInterrupt(digitalPinToInterrupt(10), servo1_callback, RISING);
15   attachInterrupt(digitalPinToInterrupt(13), servo2_callback, RISING);
16
17   Serial.println("Servo Motor Control");
18
19   // Servo 1
20   servo1.attach(10);
21   servo1.write(90);
22
23   // Servo 2
24   servo2.attach(13);
25   servo2.write(90);
26
27   // Print initial angles
28   Serial.println("Initial angles: Servo 1 = 90, Servo 2 = 90");
29
30   // Wait for 2 seconds
31   delay(2000);
32
33   // Start receiving angles from serial
34   while (1) {
35     if (Serial.available() > 0) {
36       // Read angle for Servo 1
37       int angle1 = Serial.parseInt();
38       // Read angle for Servo 2
39       int angle2 = Serial.parseInt();
40
41       // Move Servo 1
42       servo1.write(angle1);
43
44       // Move Servo 2
45       servo2.write(angle2);
46
47       // Print current angles
48       Serial.println("Current angles: Servo 1 = " + angle1 + ", Servo 2 = " + angle2);
49
50       // Wait for 1 second
51       delay(1000);
52     }
53   }
54 }
55
56 // Servo 1 callback function
57 void servo1_callback() {
58   // Read angle from serial
59   int angle1 = Serial.parseInt();
60   // Move Servo 1
61   servo1.write(angle1);
62 }
63
64 // Servo 2 callback function
65 void servo2_callback() {
66   // Read angle from serial
67   int angle2 = Serial.parseInt();
68   // Move Servo 2
69   servo2.write(angle2);
70 }

```

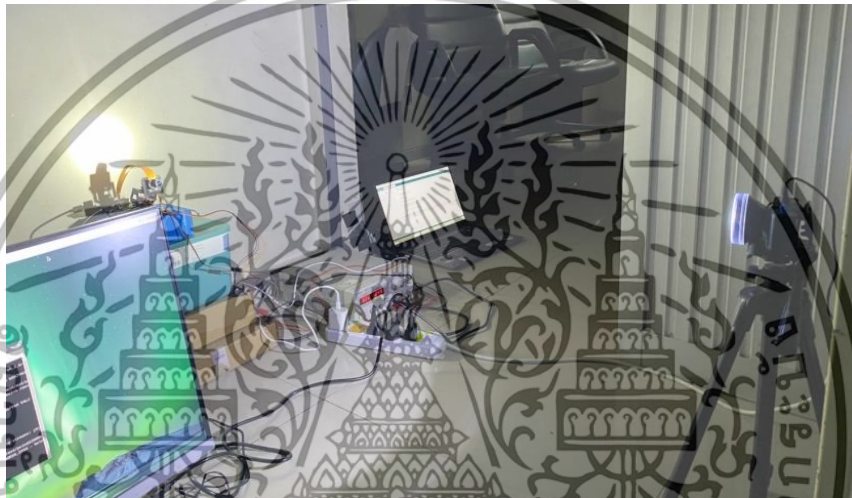
รูปที่ 4.2 โปรแกรม Arduino IDE แสดงค่ามุมผ่าน ซีเรียลมอนิเตอร์

#### 4.2 การทดสอบถ่ายภาพด้วยชุดทดลอง

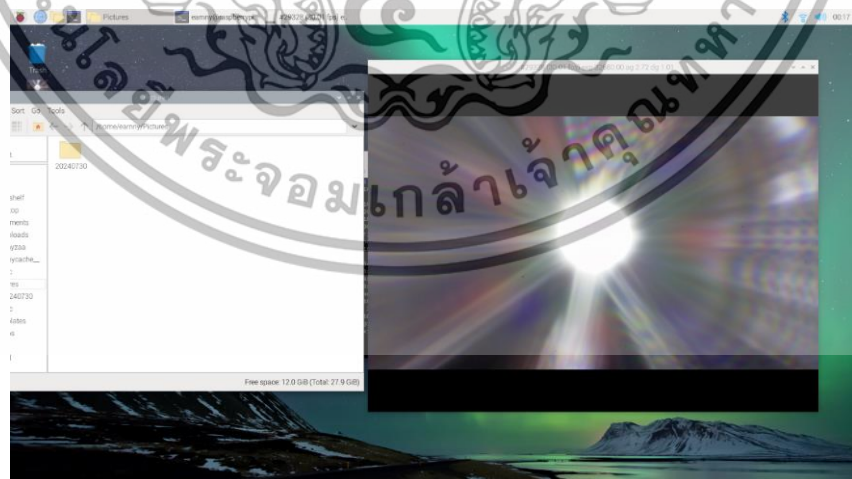
การทดสอบนี้จัดทำในห้องมืดเพื่อลดแสงรบกวนให้ได้มากที่สุด ซึ่งจะช่วยให้เซ็นเซอร์สามารถตอบสนองได้เฉพาะแสงจากแหล่งกำเนิดแสงที่ต้องการเท่านั้น โดยทำการติดตั้งไฟฉายเข้ากับขาตั้งกล้อง และกำหนดระยะห่างระหว่างไฟฉายกับเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ที่ 1 เมตร เพื่อให้แสงตกกระทบเซ็นเซอร์ในตำแหน่งที่ต้องการ สำหรับการถ่ายภาพนั้น ได้เชื่อมต่อ Pi Camera เข้ากับ Raspberry Pi 5 เพื่อใช้ในการถ่ายภาพที่แสงตกกระทบเซ็นเซอร์ โดย Pi Camera จะรับแสงที่ตกกระทบผ่านหน้ากากวงกลมที่เจาะรูขนาด 1 มิลลิเมตรบนเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ เพื่อบันทึกทิศทางและความเข้มของแสง โดย Pi Camera จะถูกควบคุมผ่านจอคอมพิวเตอร์ ซึ่งเชื่อมต่อกับ Raspberry Pi 5 ทำให้สามารถมองเห็นภาพ และข้อมูลที่เกิดขึ้นแบบเรียลไทม์ รวมถึงการปรับตั้งค่าต่าง ๆ ได้ตามต้องการในระหว่างการทดลอง ระบบการเคลื่อนที่ของเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ถูกออกแบบให้สามารถเคลื่อนที่ได้ทั้งในแกน X และแกน Y

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยการใช้ออเตอร์ 2 ตัวที่ควบคุมการเคลื่อนที่ตามแนวซ้าย-ขวา (แกน X) และบน-ล่าง (แกน Y) เพื่อจำลองการรับแสงจากหลายมุม ออเตอร์แต่ละตัวจะควบคุมการทำงานผ่าน ESP32-S2-mini โดยผ่านการตั้งค่าจากโปรแกรม Arduino IDE และมีการใช้งานเครื่อง Power supply ในการจ่ายไฟให้กับระบบเพื่อให้มีแรงบิด และความเสถียรในการควบคุมตำแหน่งของเซ็นเซอร์ การหมุนซ้าย-ขวา และบน-ล่างของมอเตอร์จะถูกตั้งค่าเป็นหน่วยองศาในโปรแกรม Arduino IDE ทำให้ควบคุมตำแหน่งของเซ็นเซอร์ให้หมุนตามองศาที่ต้องการได้อย่างแม่นยำดังในรูปที่ 4.3 และ 4.4



รูปที่ 4.3 การทดลองภาพรวม

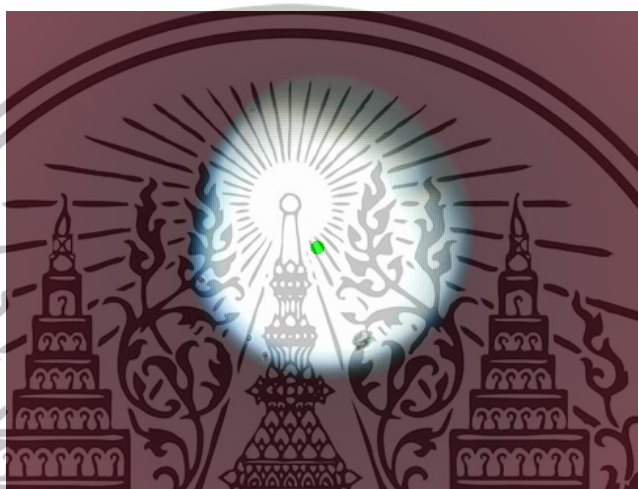


รูปที่ 4.4 การถ่ายภาพด้วยชุดทดลองผ่าน Pi Camera

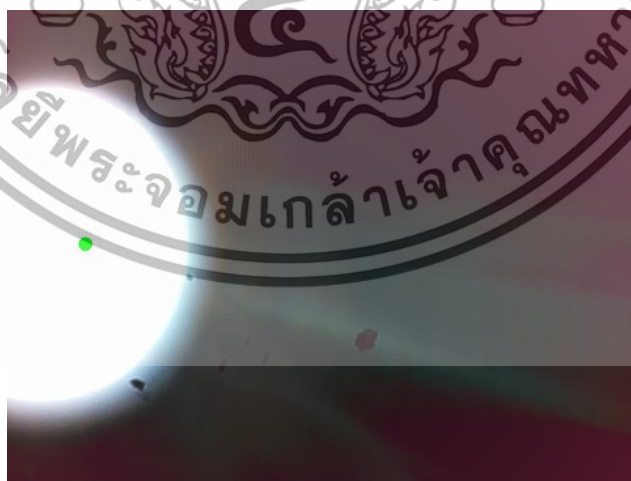
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.3 ผลการทดสอบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์

การทดสอบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ ผลลัพธ์ที่ได้จากการทดสอบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์นั้นเพื่อนำไปวิเคราะห์ ทำการถ่ายภาพการทดสอบในมุม Azimuth (Az) และ Elevation (EL) ต่าง ๆ ที่กำหนด โดยได้ทำการถ่ายภาพผลลัพธ์ทดสอบ ดังในรูปที่ 4.5, 4.6, 4.7, 4.8, 4.9, 4.10, 4.11, 4.12 และ 4.13



รูปที่ 4.5 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา



รูปที่ 4.6 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

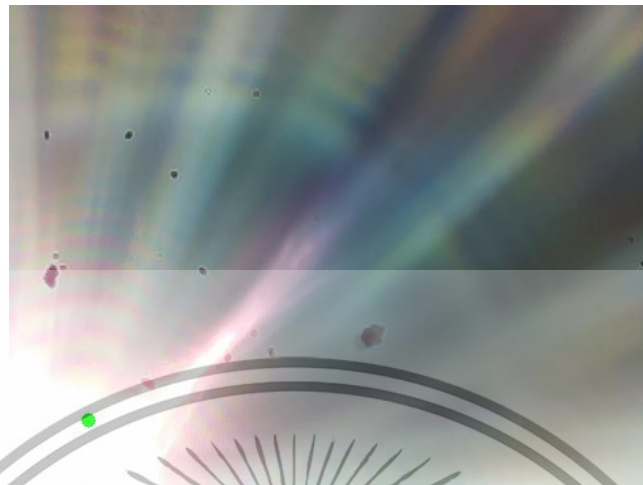


รูปที่ 4.7 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation 0 องศา



รูปที่ 4.8 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

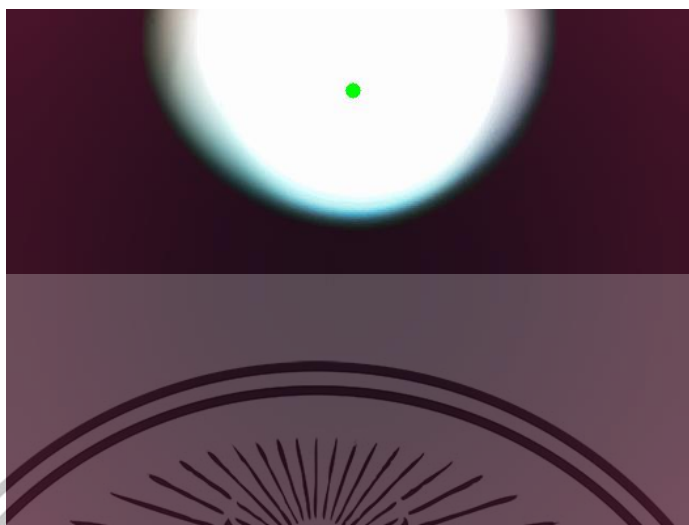


รูปที่ 4.9 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation -15 องศา



รูปที่ 4.10 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

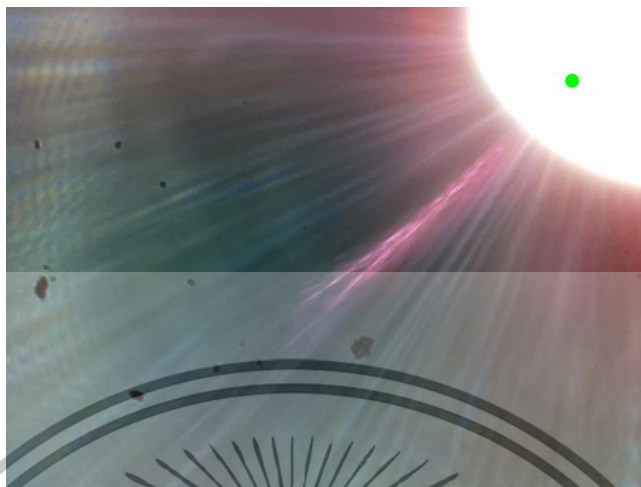


รูปที่ 4.11 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 15 องศา



รูปที่ 4.12 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.13 ผลการทดสอบที่ได้จากเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา

#### 4.4 ผลการทดสอบทดสอบโดยไม่ปรับค่า Threshold และปรับค่า Threshold

การทดสอบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ มีการเปรียบเทียบผลลัพธ์ระหว่างการไม่ปรับค่า Threshold และการปรับค่า Threshold เพื่อประเมินประสิทธิภาพ และความแม่นยำของการตรวจจับทิศทางแสงอาทิตย์ ในกรณีการไม่ปรับค่า Threshold พบสัญญาณรบกวนจากแสงในบริเวณที่ไม่ต้องการ ที่ทำให้การวิเคราะห์ตำแหน่งเซนทรอยด์ มีความคลาดเคลื่อน และเมื่อมีการปรับ Threshold เพื่อกรองแสงรบกวนออก จะพบว่าผลการทดสอบมีความแม่นยำมากขึ้น

##### 4.4.1 ผลการทดสอบโปรไฟล์ความเข้มแสง (Intensity Profile)

ทำการทดสอบโปรไฟล์ความเข้มแสง จากผลลัพธ์ที่ได้จากการทดสอบการถ่ายภาพด้วยเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ โดยใช้โปรแกรม MATLAB ในการคำนวณความเข้มแสงจากพิกเซลตามแนวแกน X แนวแกน Y รวมถึงการคำนวณตำแหน่งเซนทรอยด์ของแสงที่ตกกระทบบนเซ็นเซอร์ ผลการทดสอบ ดังในรูปที่ 4.14, 4.17, 4.20, 4.23, 4.26, 4.29, 4.32, 4.35, 4.38, 4.41, 4.44, 4.47, 4.50, 4.53, 4.56, 4.59, 4.62 และ 4.65 ซึ่งแสดงโปรไฟล์ความเข้มแสงตามแนวแกน X และ Y สำหรับมุมทิศทางแสงต่าง ๆ ที่ได้ทำการทดสอบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.4.2 ผลการทดสอบการทำ UV pattern

การวิเคราะห์ UV pattern ซึ่งเป็นการสร้างรูปแบบของแสง UV ที่ตกกระทบบนเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ เพื่อให้ได้ข้อมูลเชิงลึกเพิ่มเติมเกี่ยวกับการกระจายตัวของแสง UV ในพื้นที่ต่าง ๆ ของเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ การทำ UV pattern เพื่อเพิ่มความแม่นยำในการระบุตำแหน่งและทิศทางของแสง UV รวมถึงการตรวจสอบการกระจายตัวของแสงในมุมต่าง ๆ ซึ่งรูปแบบผลลัพธ์ จะแสดงในรูปแบบของภาพ 2D และ 3D ช่วยให้สามารถเห็นกระจายตัวของแสงในมุมมองที่ชัดเจน และละเอียดมากขึ้น

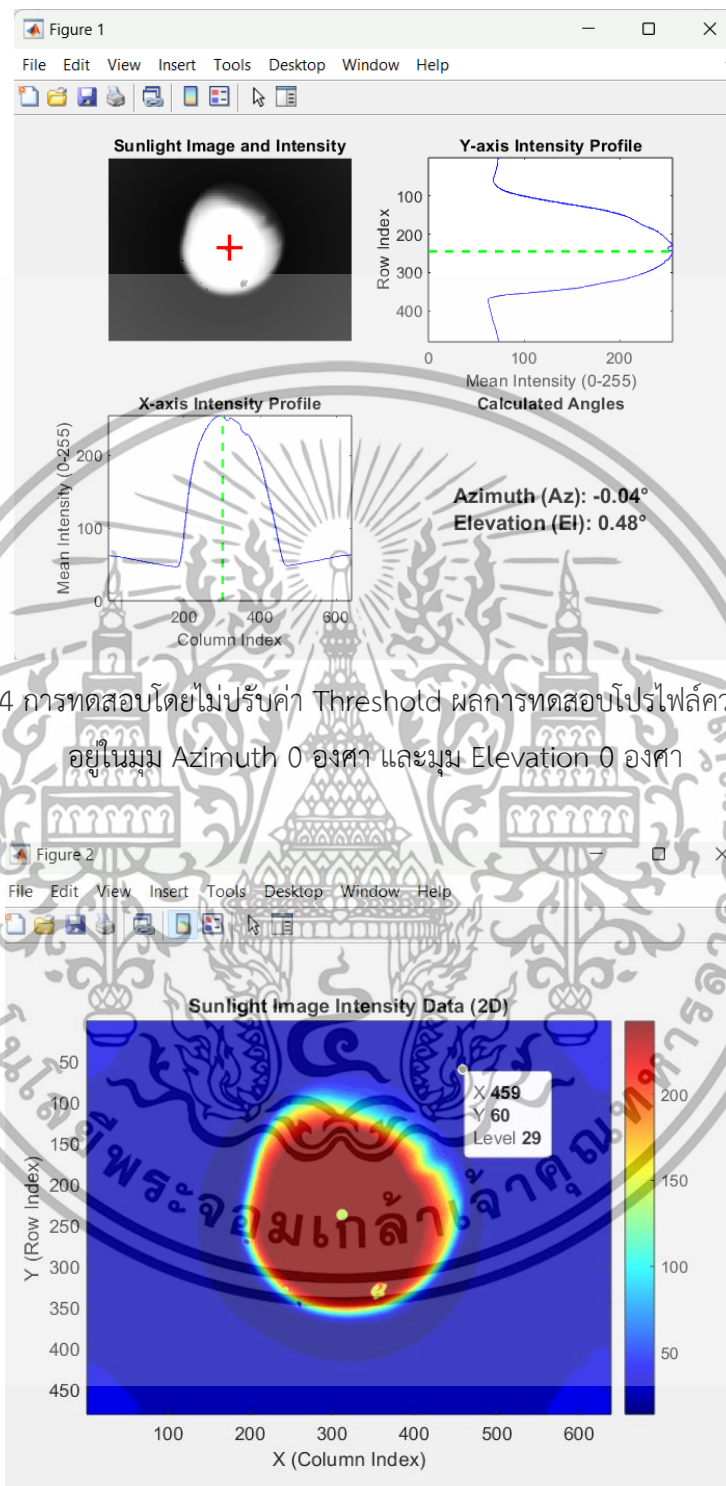
##### 4.4.2.1 2D UV pattern

ใช้ในการแสดงการกระจายตัวของความเข้มแสงในระนาบสองมิติ (แกน X และ Y) ซึ่งจะช่วยให้เห็นการกระจายของแสงในแต่ละตำแหน่งบนเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ สามารถใช้ในการระบุตำแหน่งของจุดศูนย์กลาง และการเปลี่ยนแปลงความเข้มแสงในระนาบได้ ซึ่งผลการทดสอบ ดังในรูปที่ 4.15, 4.18, 4.21, 4.24, 4.27, 4.30, 4.33, 4.36, 4.39, 4.42, 4.45, 4.48, 4.51, 4.54, 4.57, 4.60, 4.63 และ 4.66

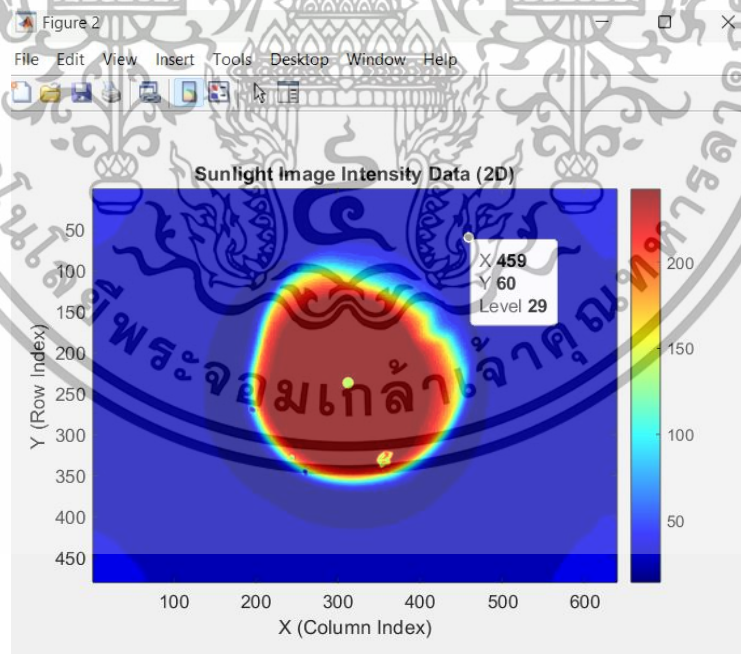
##### 4.4.2.2 3D UV pattern

ช่วยในการแสดงผลความเข้มแสงในเชิงลึก โดยการเพิ่มมิติความเข้มแสงในแนวแกน Z ซึ่งทำให้สามารถเห็นความสูงต่ำของความเข้มแสงในแต่ละตำแหน่งบนเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ ทำให้มองเห็นความเปลี่ยนแปลงของความเข้มแสง และลักษณะการกระจายตัวในเชิงลึกได้อย่างชัดเจนขึ้น ซึ่งผลการทดสอบ ดังในรูปที่ 4.16, 4.19, 4.22, 4.25, 4.28, 4.31, 4.34, 4.37, 4.40, 4.43, 4.46, 4.49, 4.52, 4.55, 4.58, 4.61, 4.64 และ 4.67

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

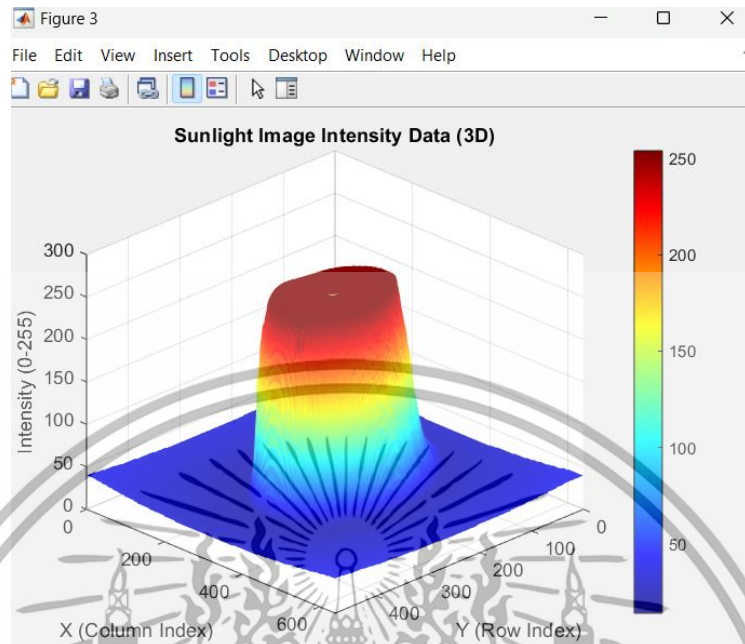


รูปที่ 4.14 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา

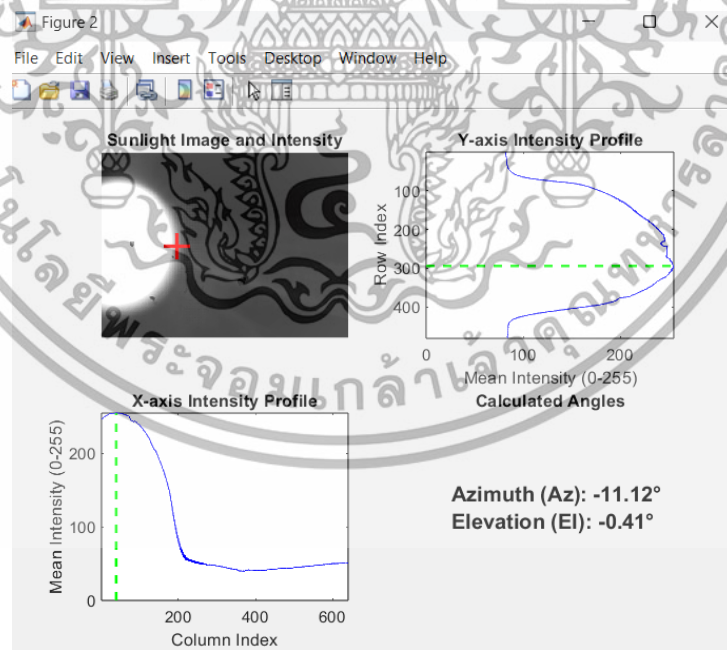


รูปที่ 4.15 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

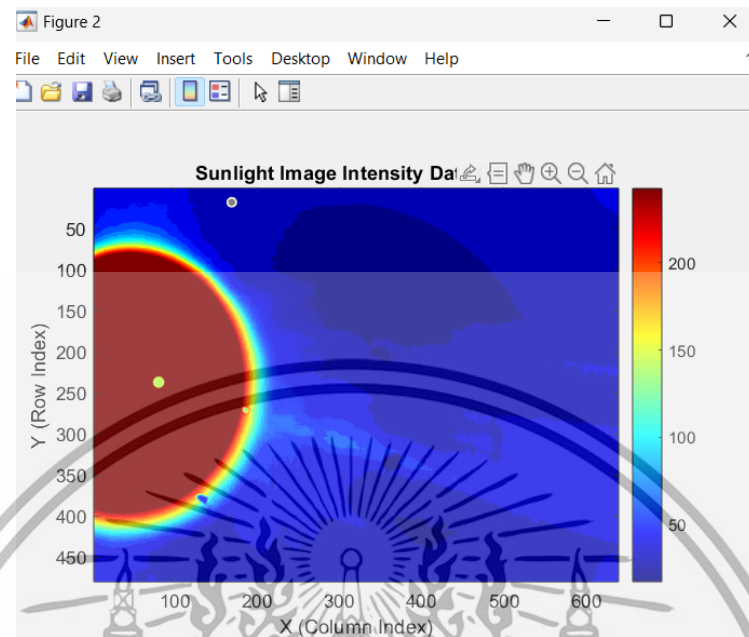


รูปที่ 4.16 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา

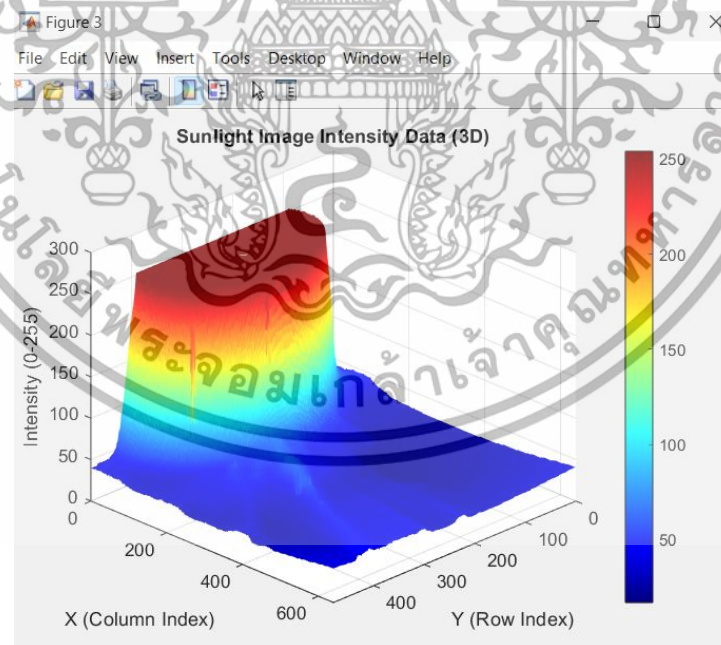


รูปที่ 4.17 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม Azimuth -20 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

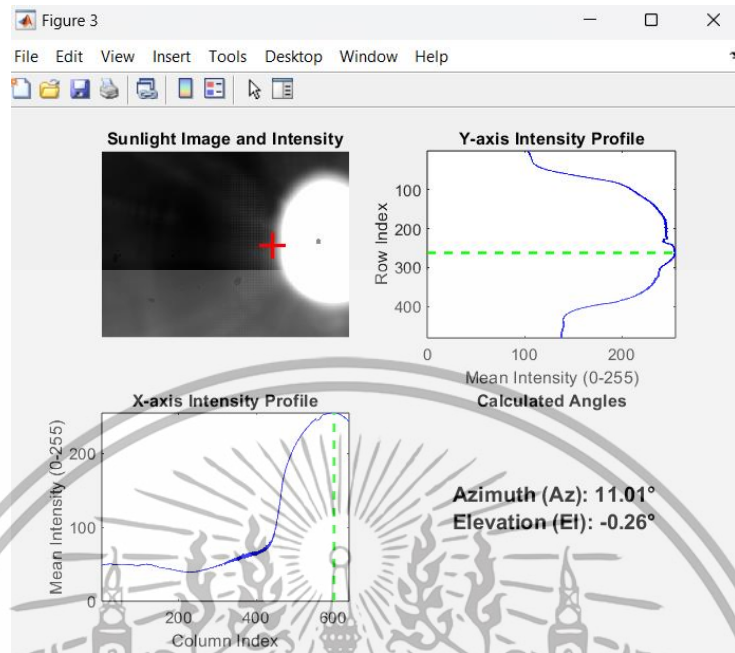


รูปที่ 4.18 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D อยู่ในมุม Azimuth -20 องศา และมุม Elevation 0 องศา

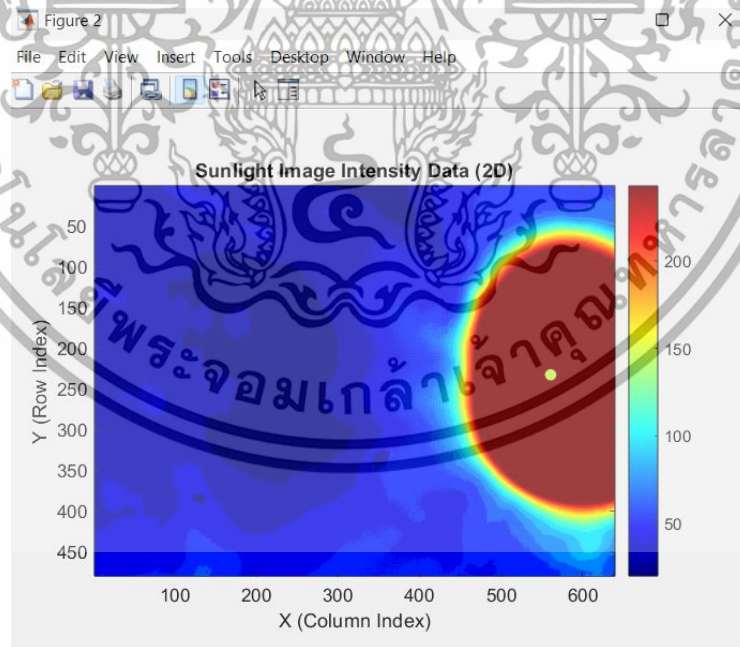


รูปที่ 4.19 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D อยู่ในมุม Azimuth -20 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

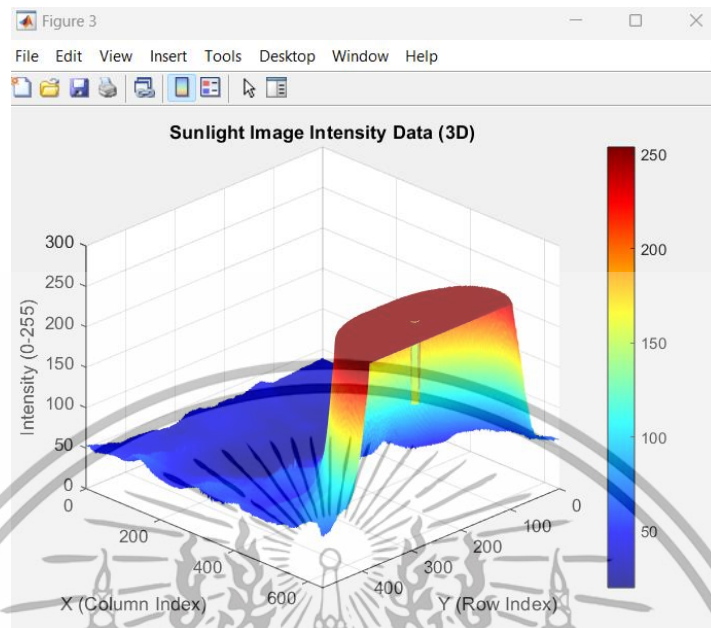


รูปที่ 4.20 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม Azimuth 20 องศา และมุม Elevation 0 องศา

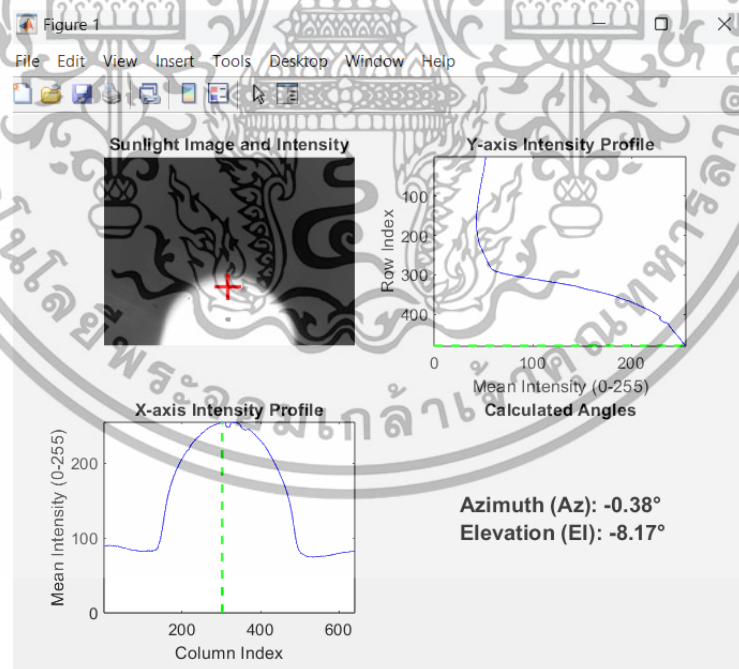


รูปที่ 4.21 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D อยู่ในมุม Azimuth 20 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

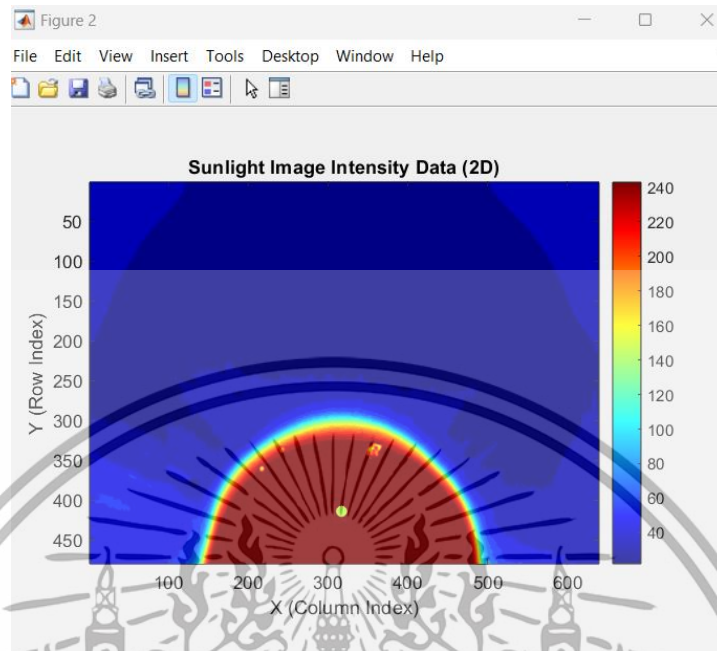


รูปที่ 4.22 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D อยู่ในมุม Azimuth 20 องศา และมุม Elevation 0 องศา

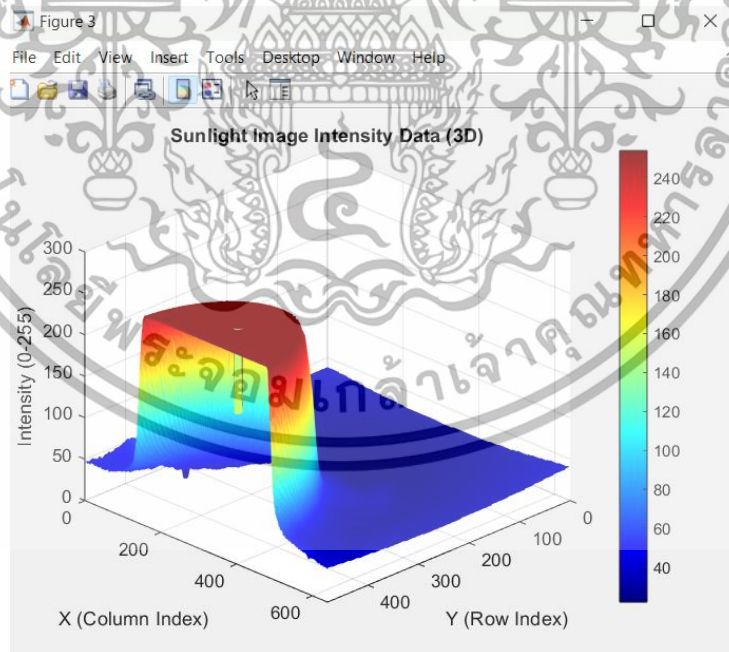


รูปที่ 4.23 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม Azimuth 0 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

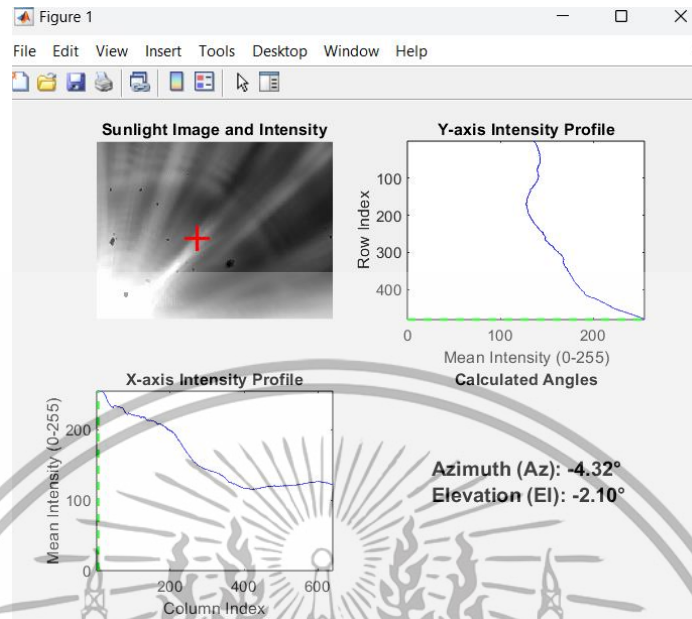


รูปที่ 4.24 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D อยู่ในมุม Azimuth 0 องศา และมุม Elevation -15 องศา

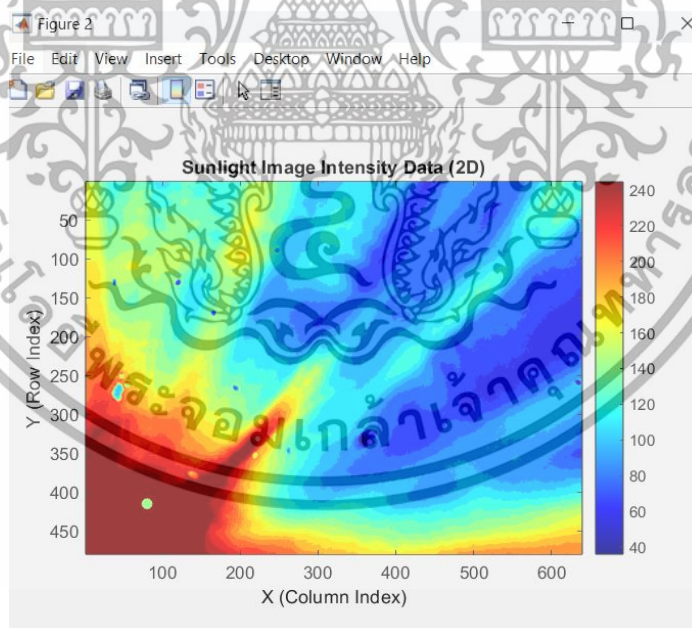


รูปที่ 4.25 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D อยู่ในมุม Azimuth 0 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

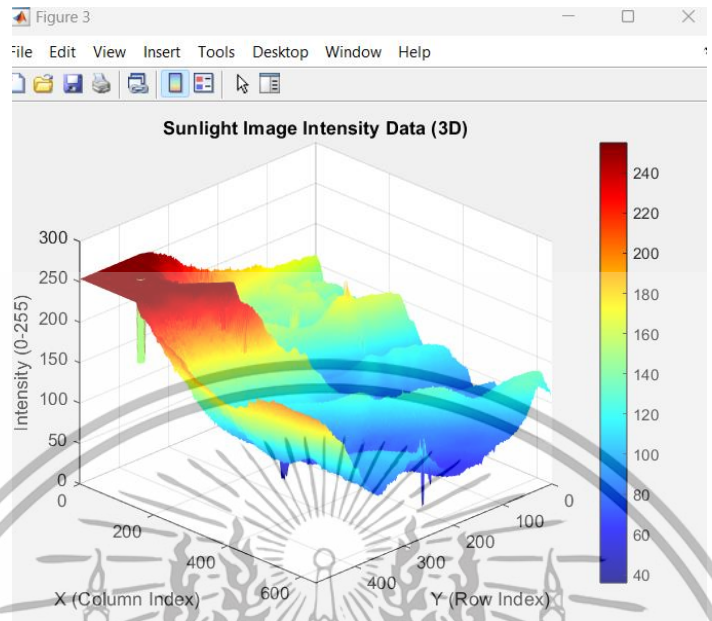


รูปที่ 4.26 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation -15 องศา

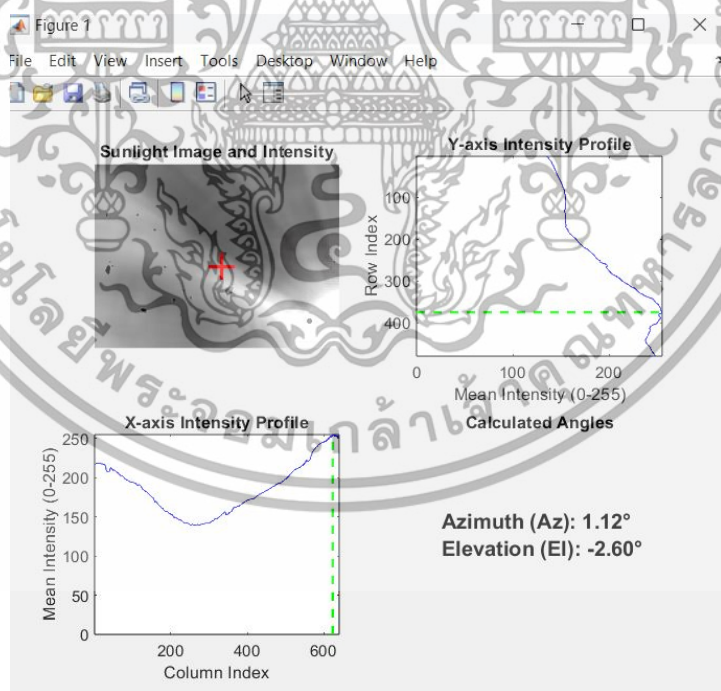


รูปที่ 4.27 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

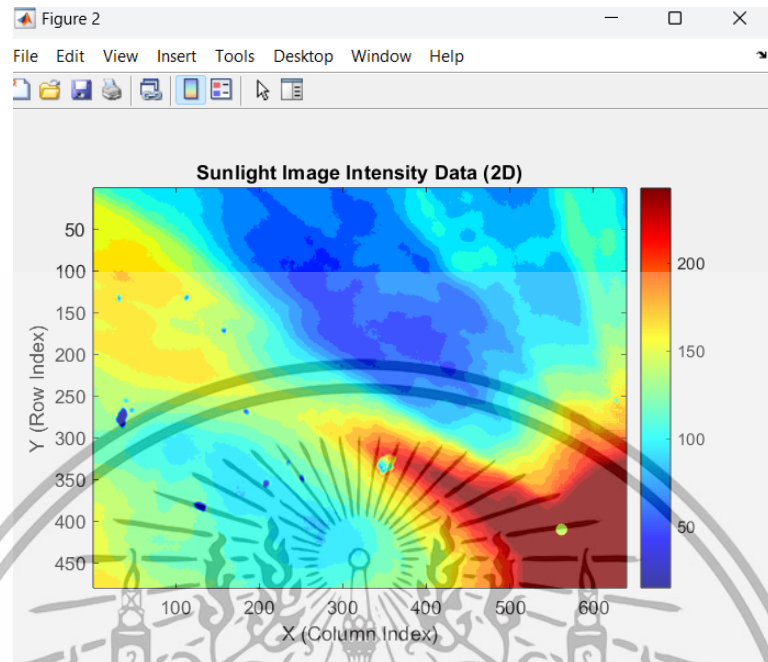


รูปที่ 4.28 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D อยู่ในมุม Azimuth -20 องศา และมุม Elevation -15 องศา

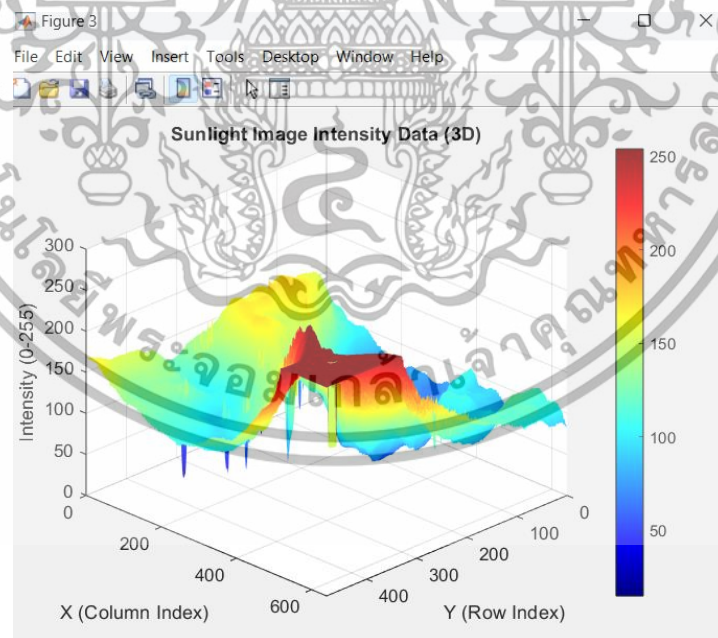


รูปที่ 4.29 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม Azimuth 20 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

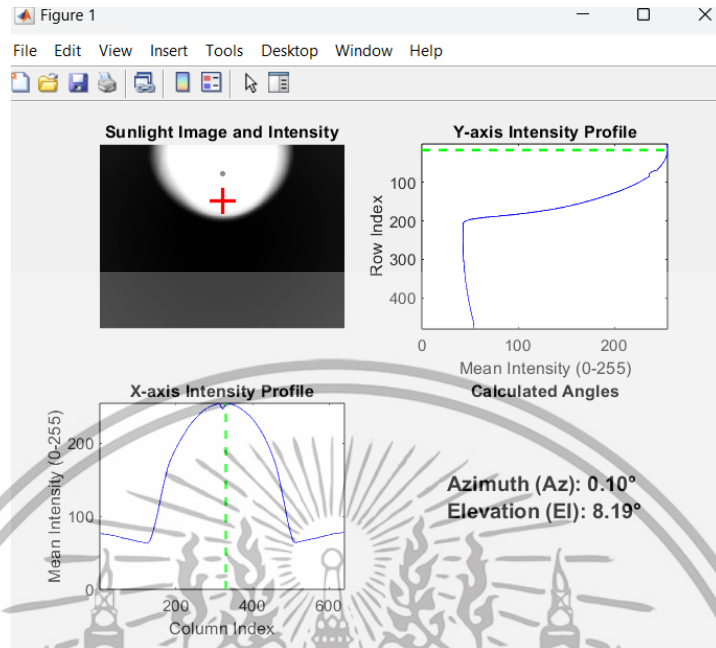


รูปที่ 4.30 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation -15 องศา

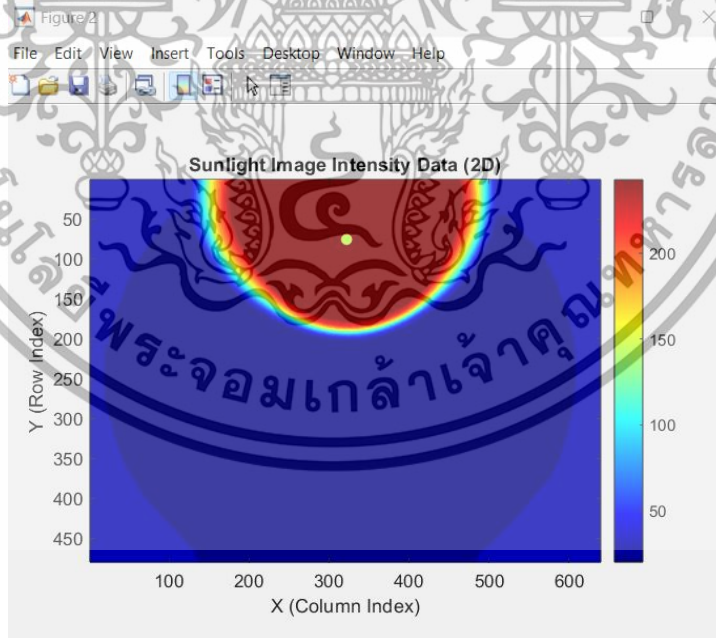


รูปที่ 4.31 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

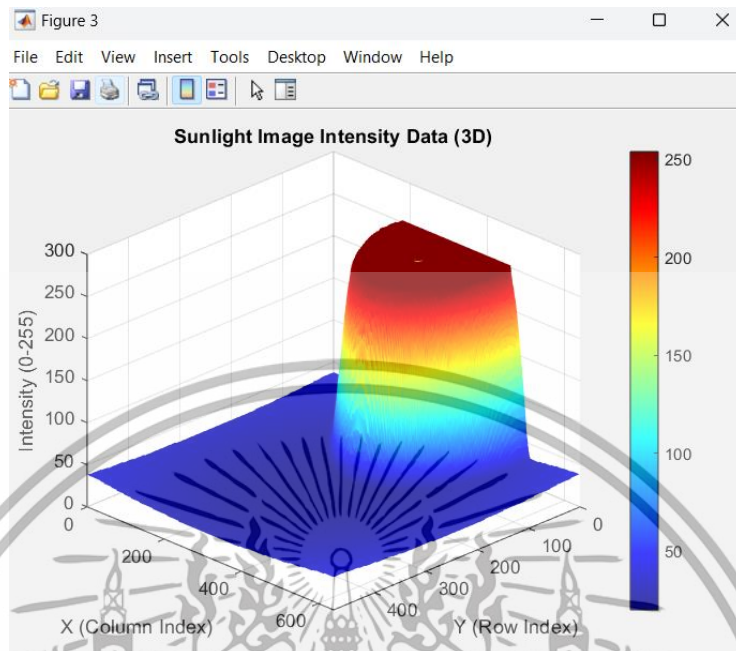


รูปที่ 4.32 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 15 องศา

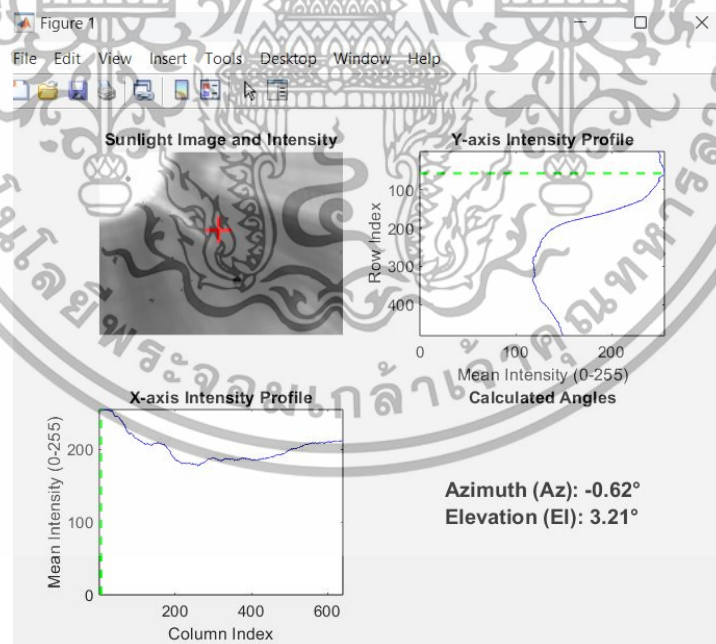


รูปที่ 4.33 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

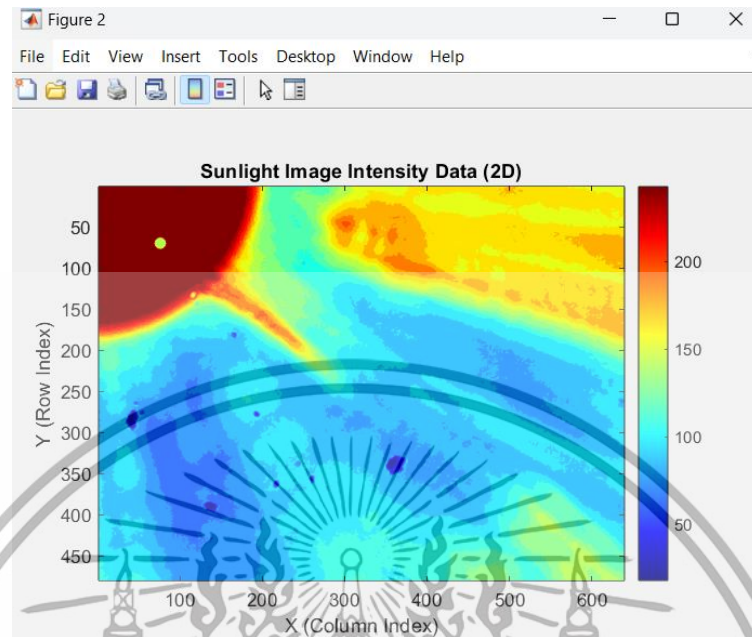


รูปที่ 4.34 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 15 องศา

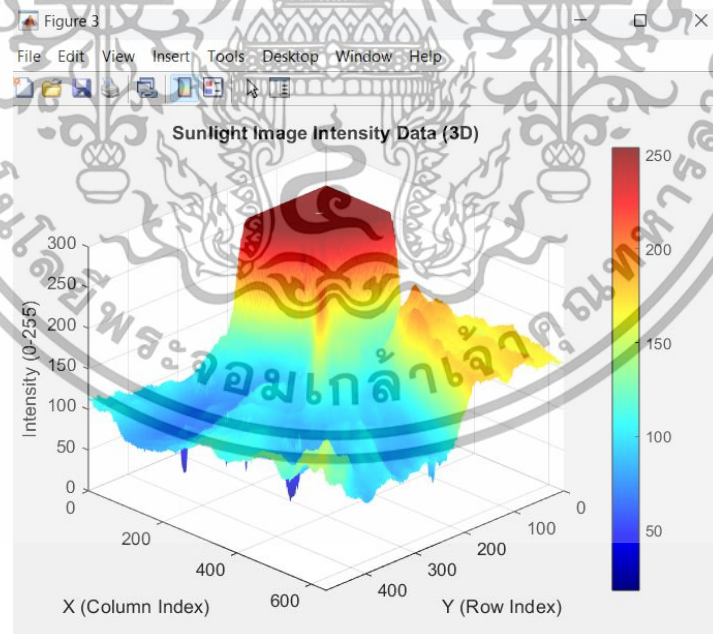


รูปที่ 4.35 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

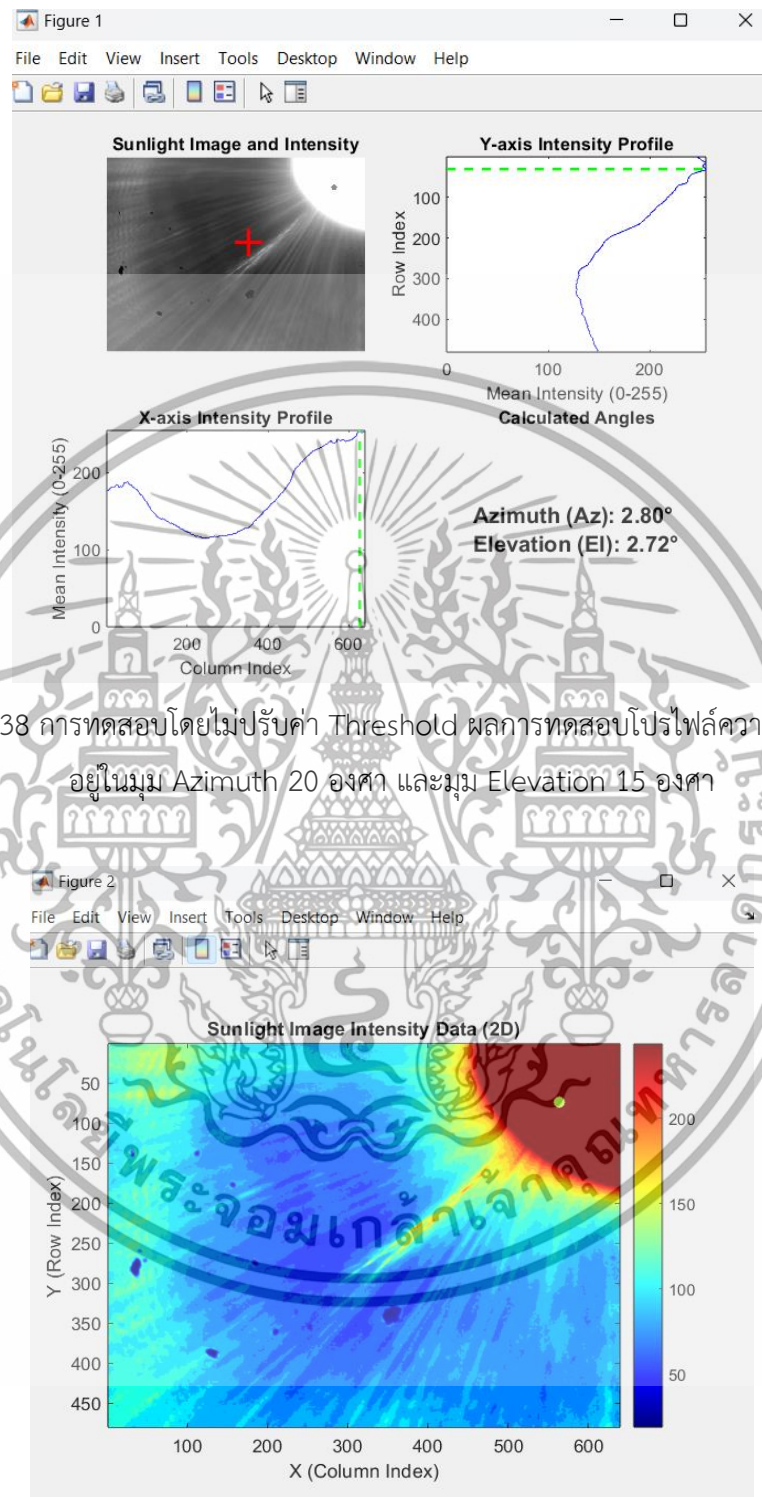


รูปที่ 4.36 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 15 องศา

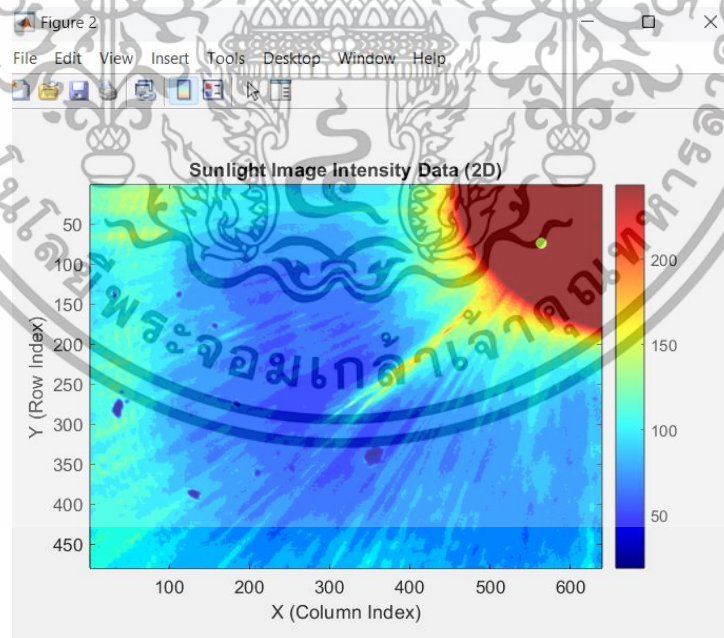


รูปที่ 4.37 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

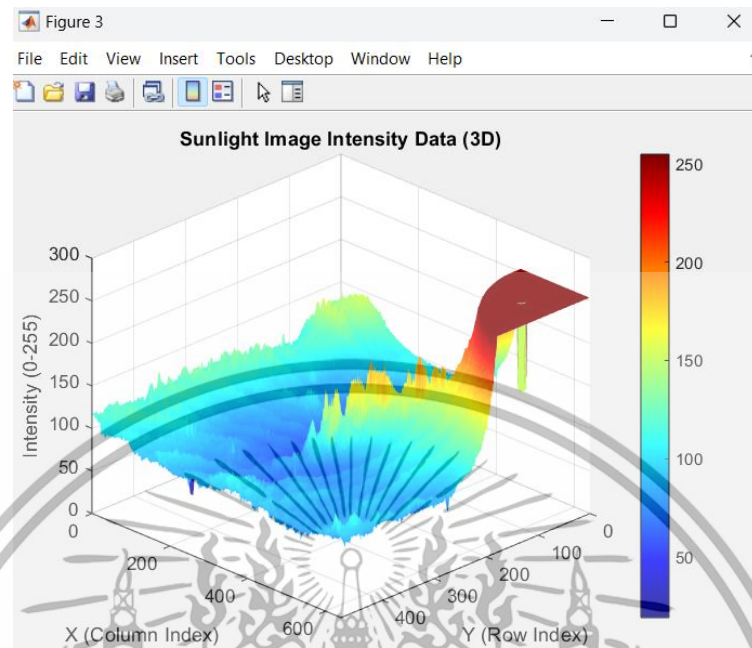


รูปที่ 4.38 การทดสอบโดยไม่ปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา

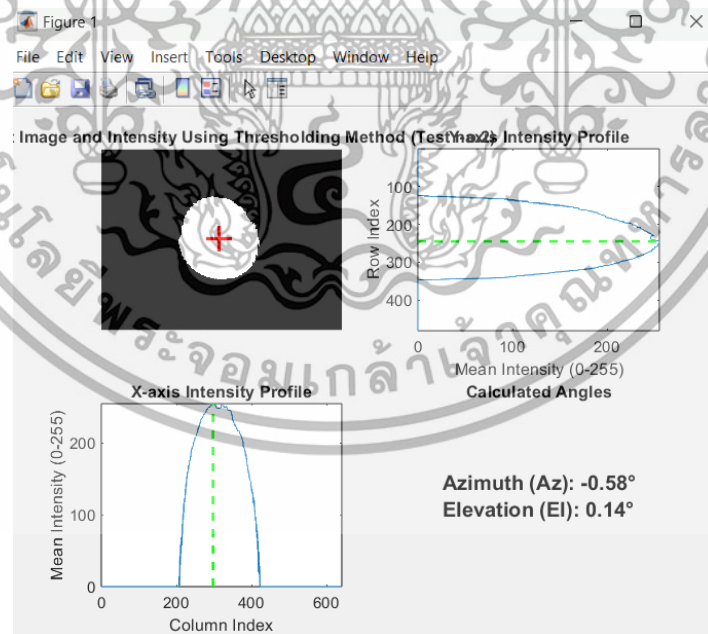


รูปที่ 4.39 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 2D อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

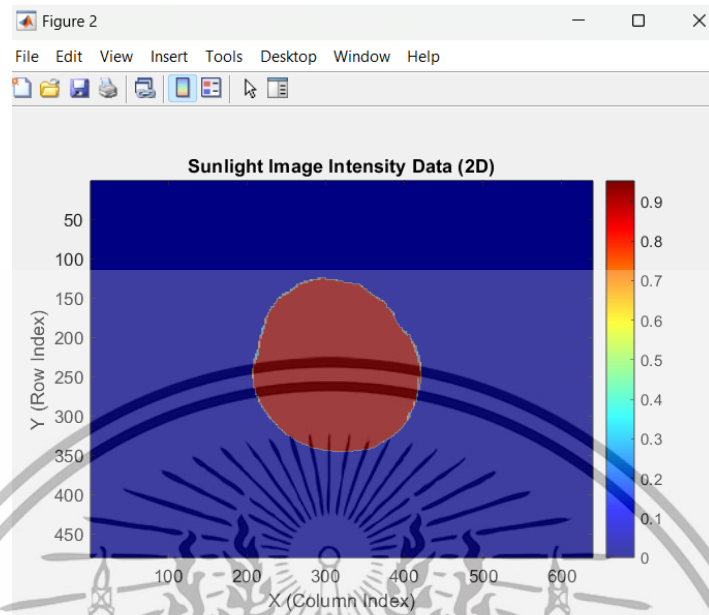


รูปที่ 4.40 การทดสอบโดยไม่ปรับค่า Threshold การทำ UV pattern 3D อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา

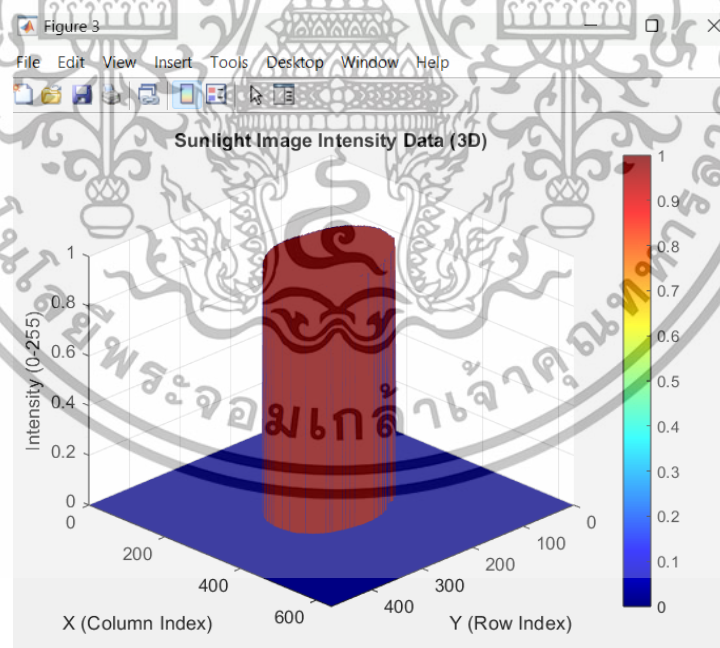


รูปที่ 4.41 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

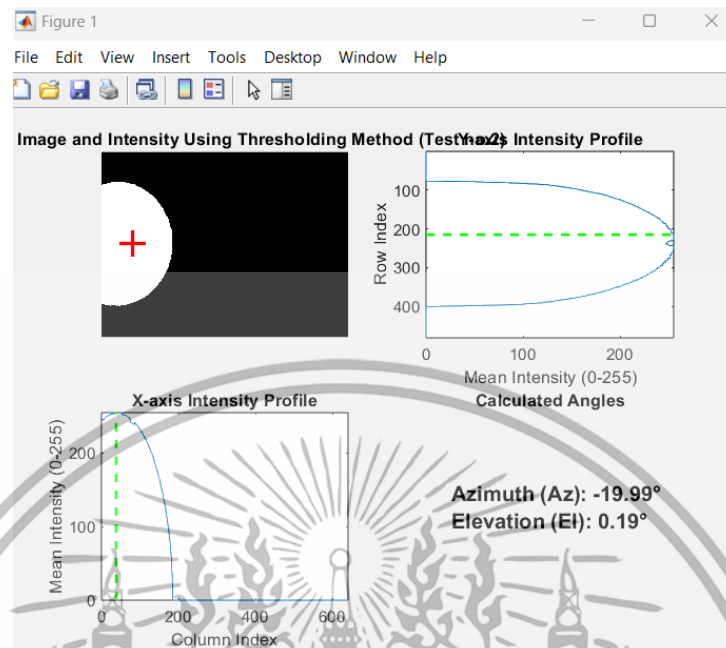


รูปที่ 4.42 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา

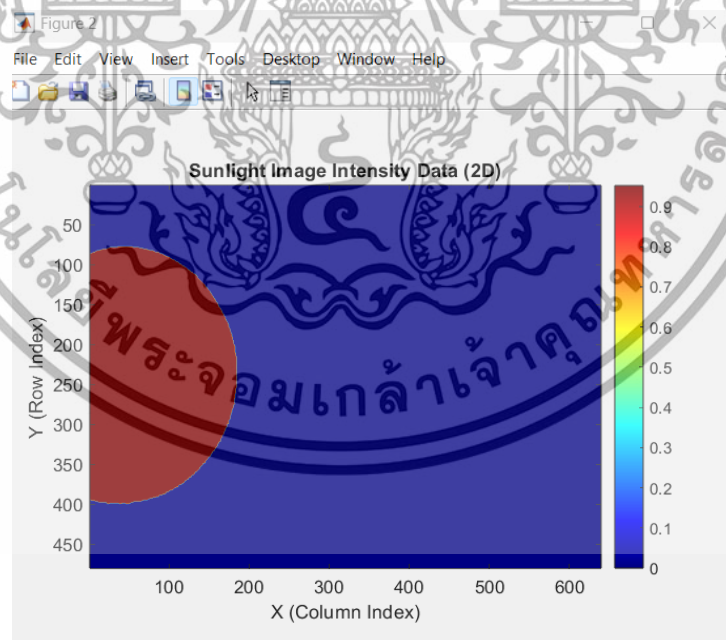


รูปที่ 4.43 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

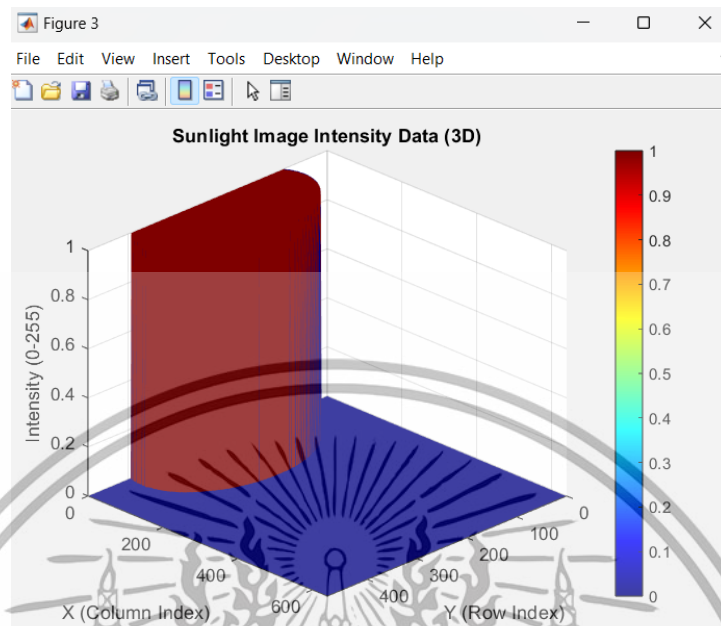


รูปที่ 4.44 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 0 องศา

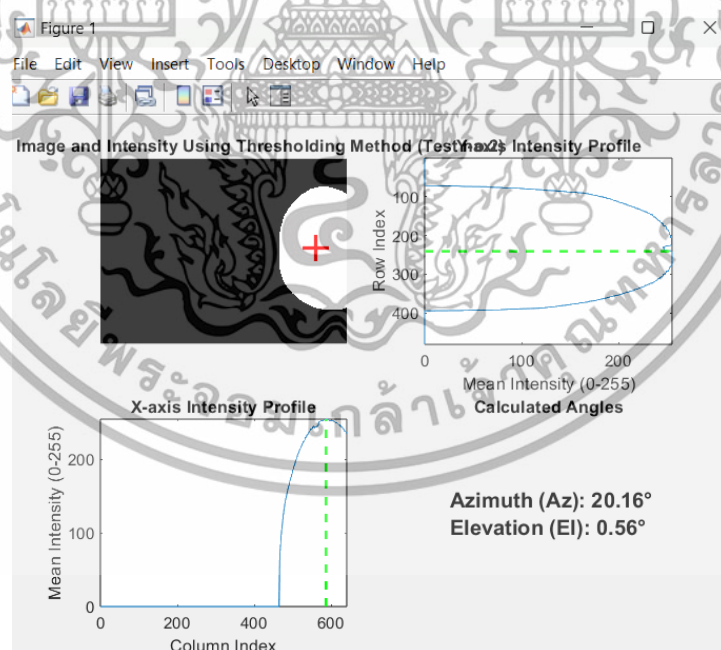


รูปที่ 4.45 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

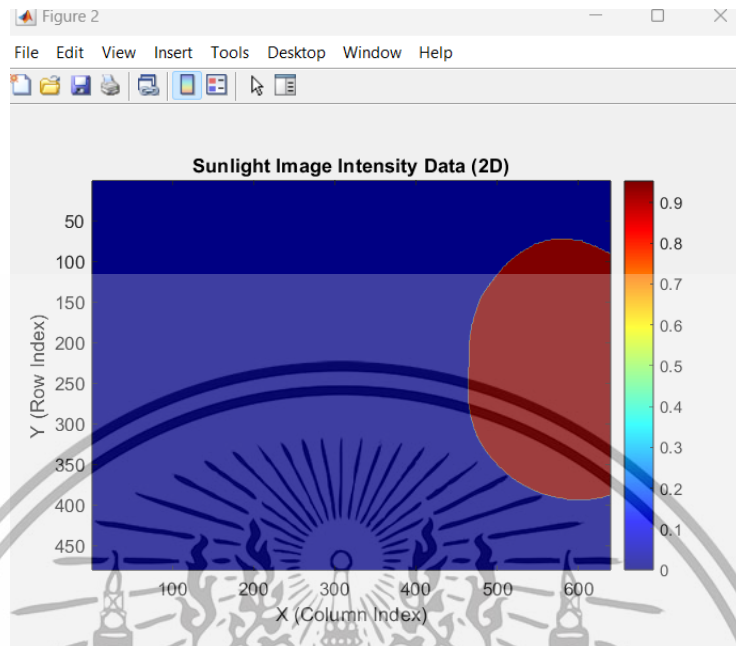


รูปที่ 4.46 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D อยู่ในมุม Azimuth -20 องศา และมุม Elevation 0 องศา

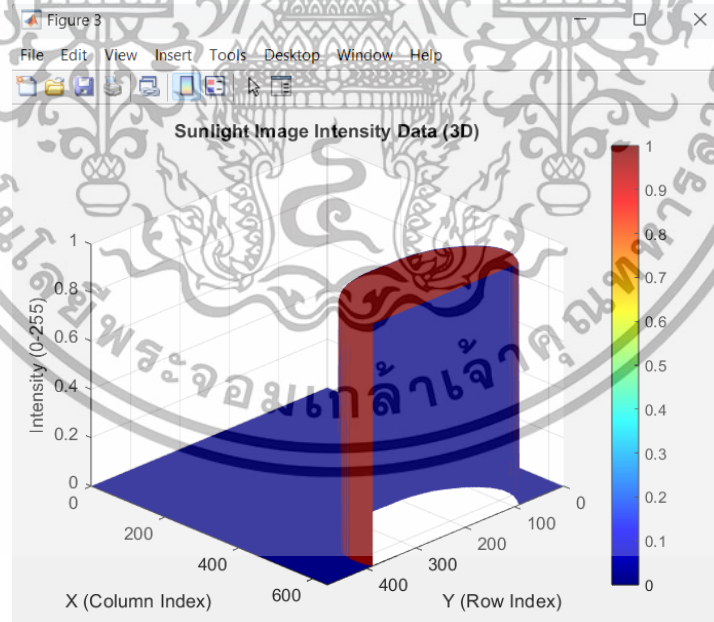


รูปที่ 4.47 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง อยู่ในมุม Azimuth 20 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

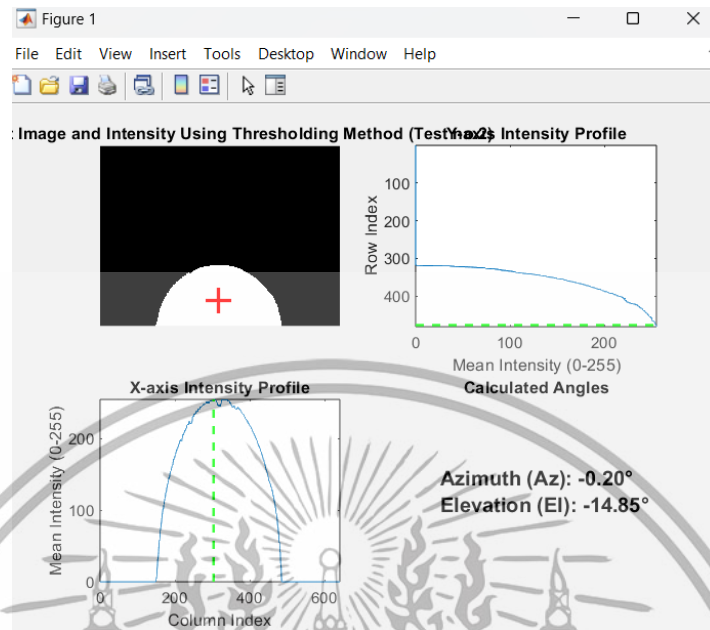


รูปที่ 4.48 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation 0 องศา

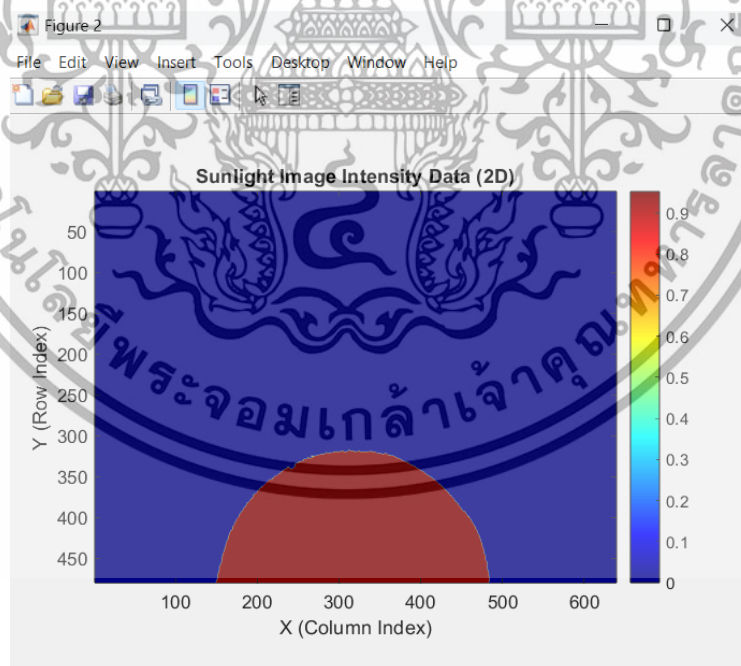


รูปที่ 4.49 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation 0 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

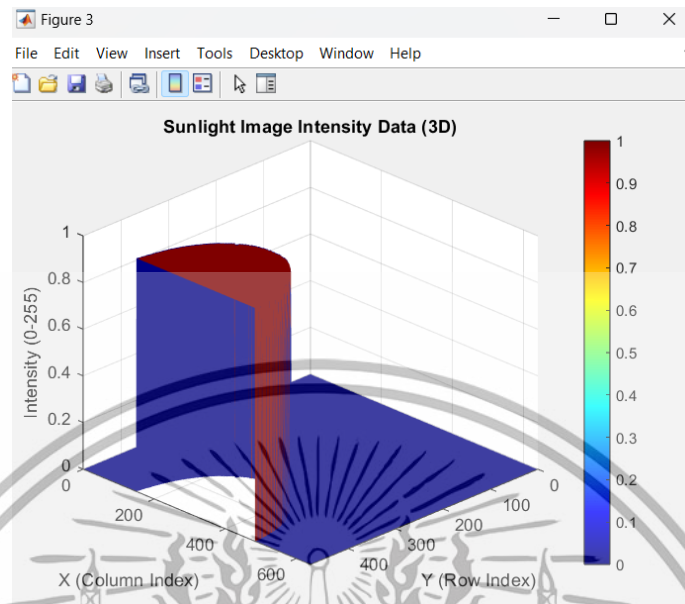


รูปที่ 4.50 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation -15 องศา

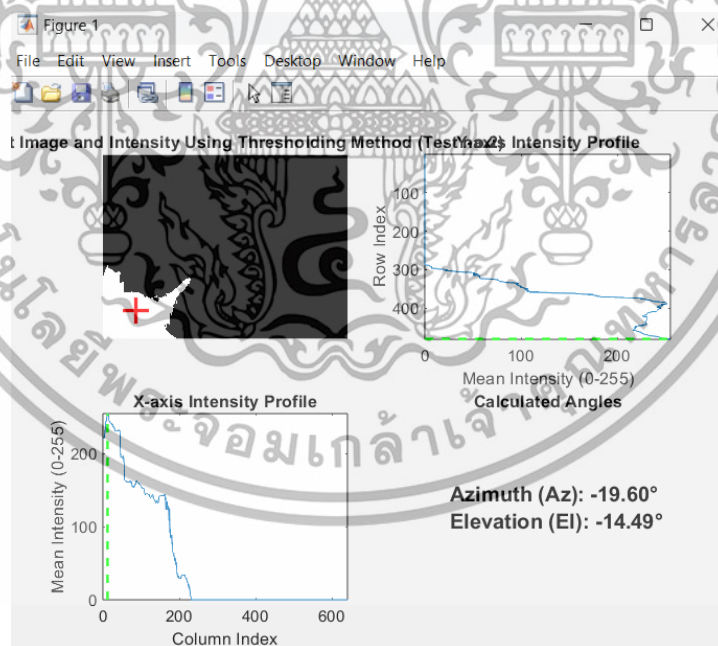


รูปที่ 4.51 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

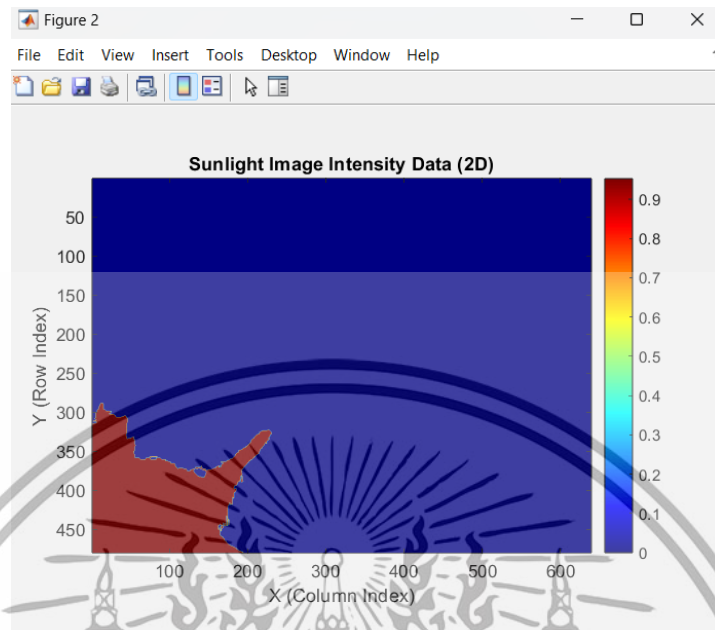


รูปที่ 4.52 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation -15 องศา

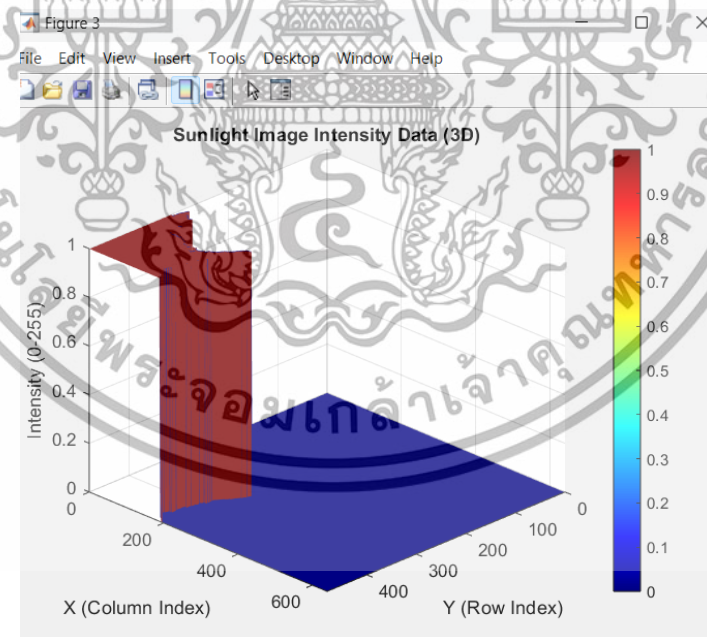


รูปที่ 4.53 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

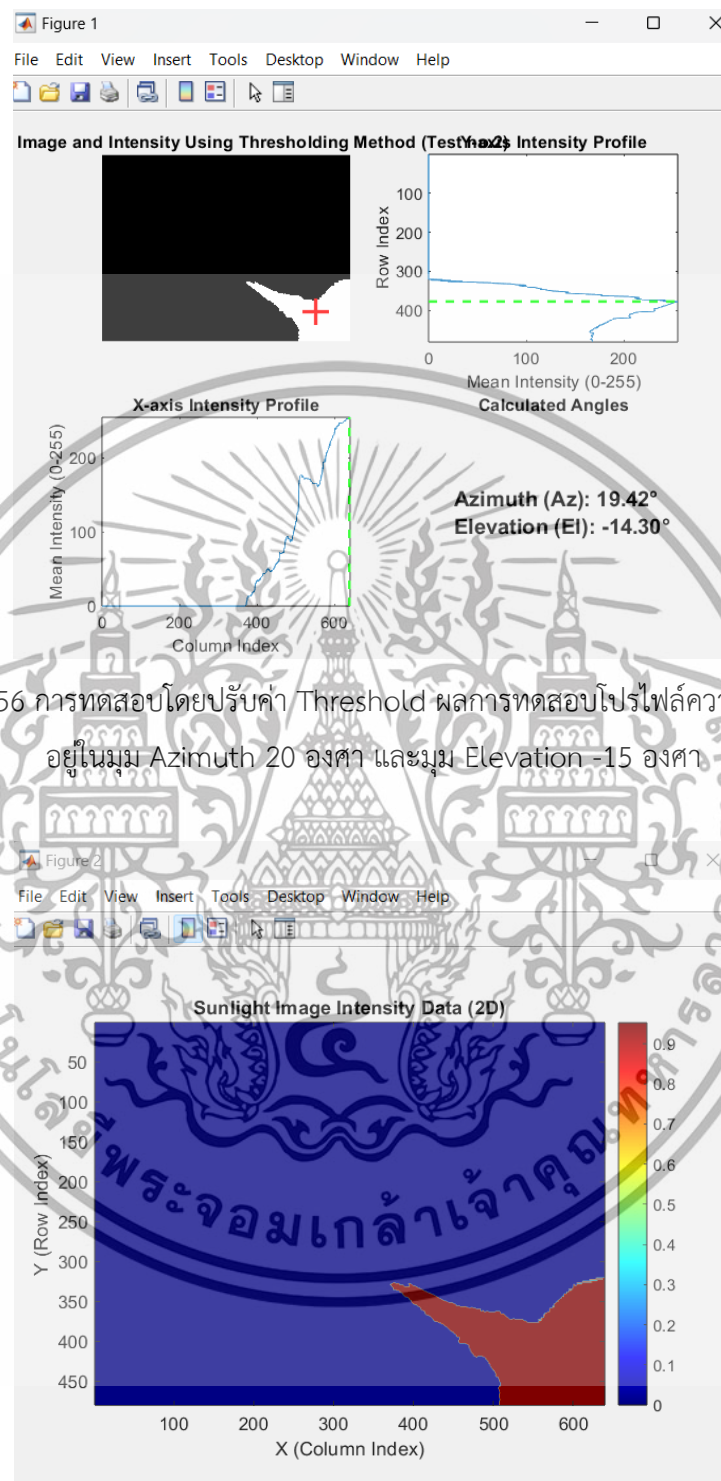


รูปที่ 4.54 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation -15 องศา

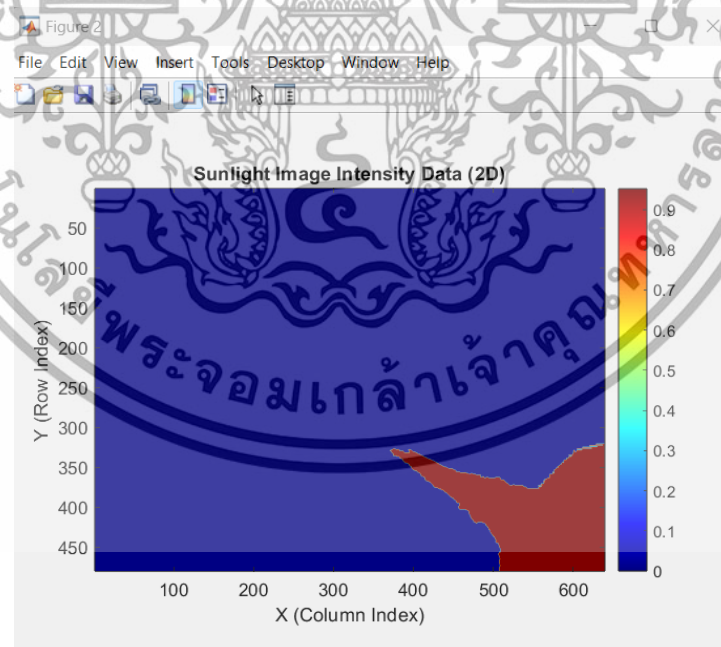


รูปที่ 4.55 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

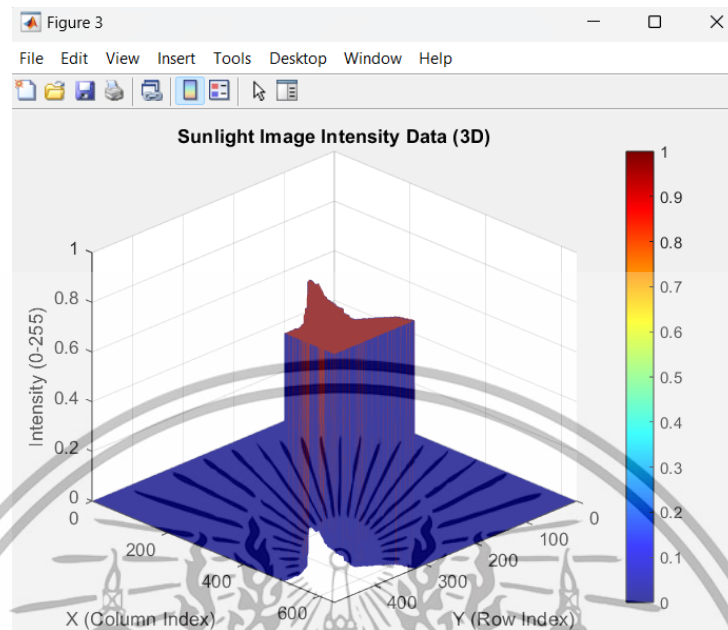


รูปที่ 4.56 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation -15 องศา

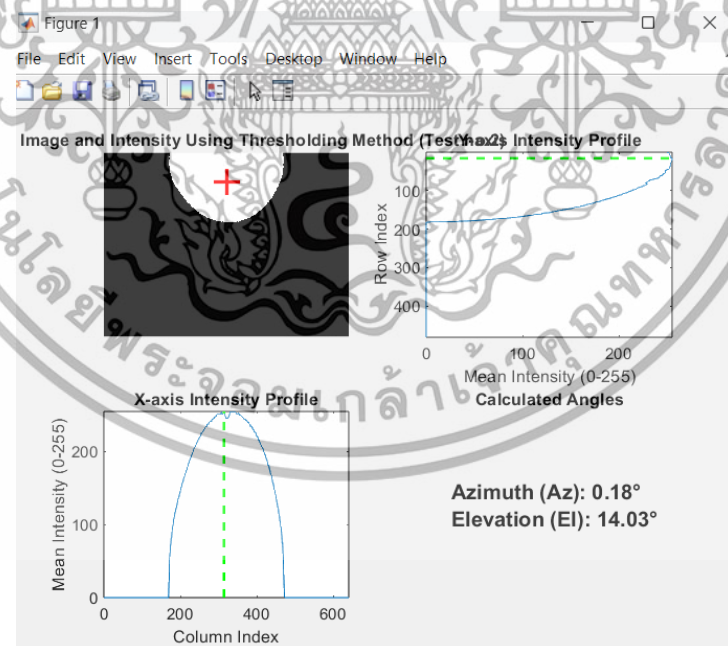


รูปที่ 4.57 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation -15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

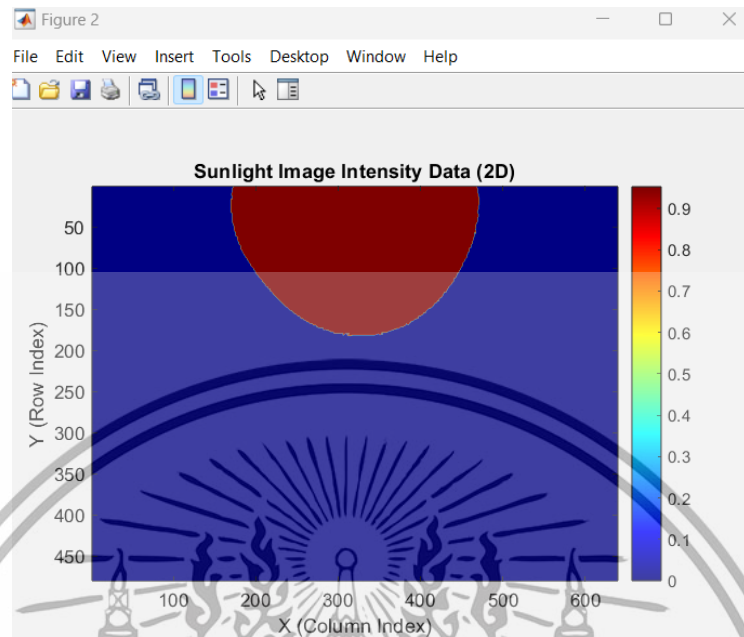


รูปที่ 4.58 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation -15 องศา

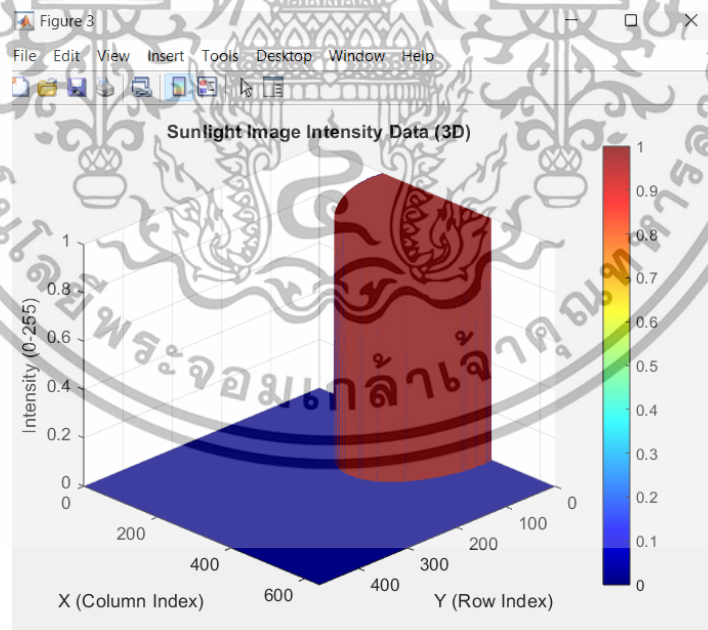


รูปที่ 4.59 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

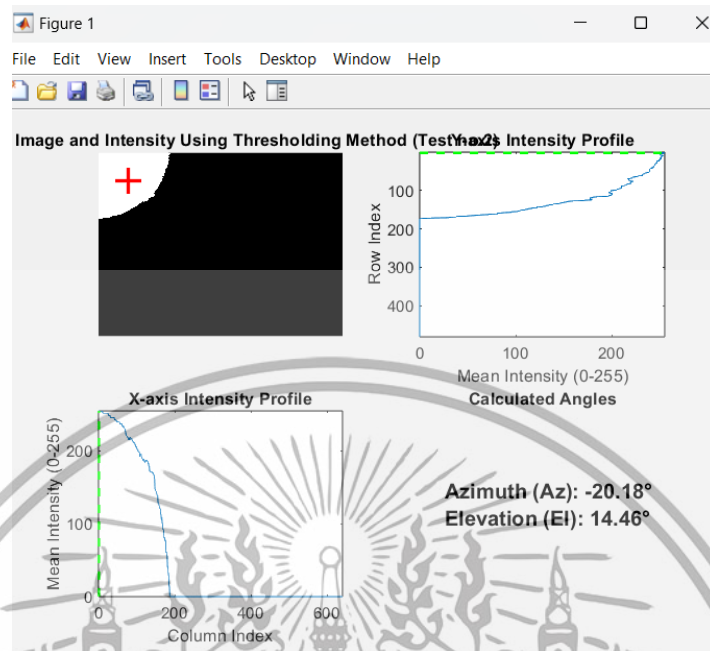


รูปที่ 4.60 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 15 องศา

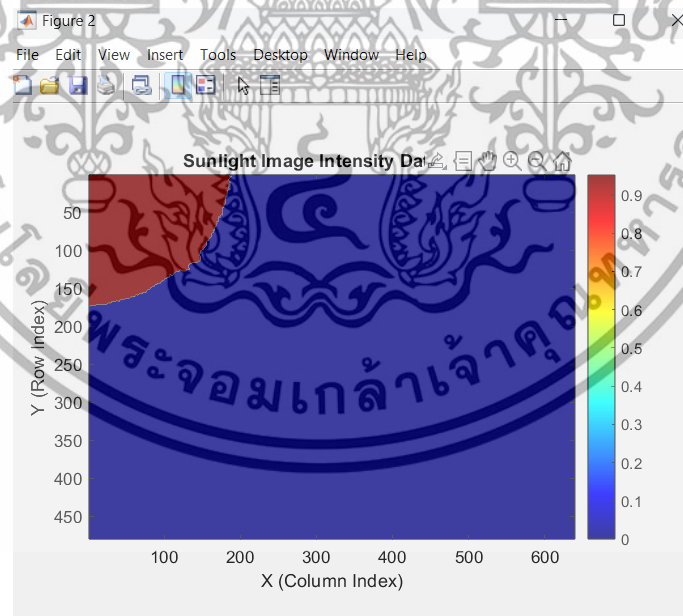


รูปที่ 4.61 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 0 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

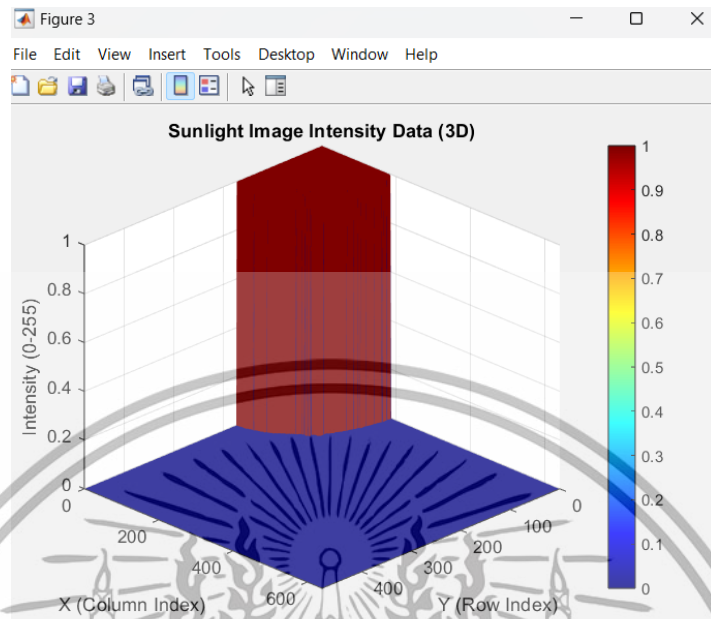


รูปที่ 4.62 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 15 องศา

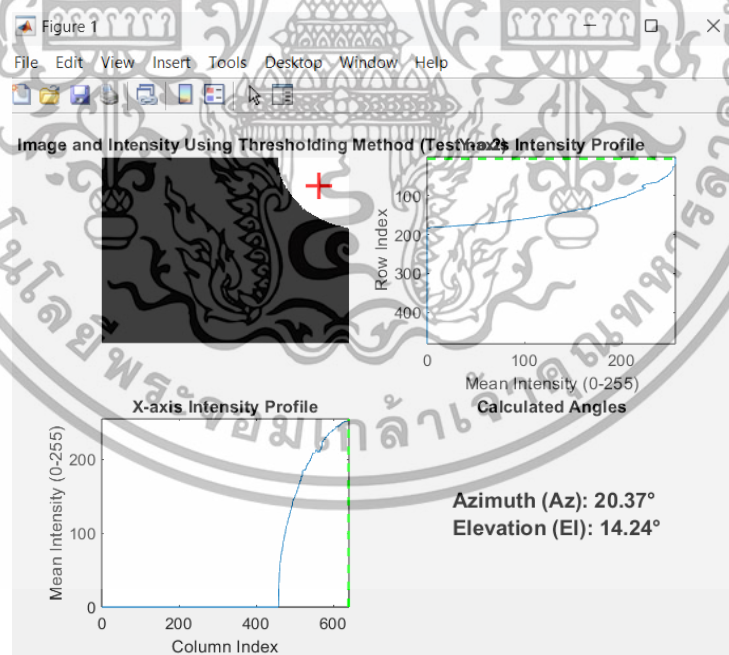


รูปที่ 4.63 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

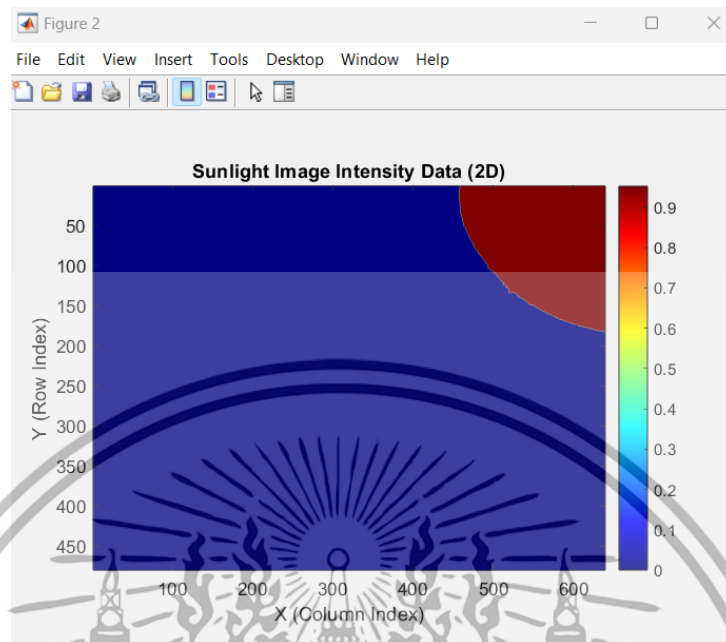


รูปที่ 4.64 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth -20 องศา และมุม Elevation 15 องศา

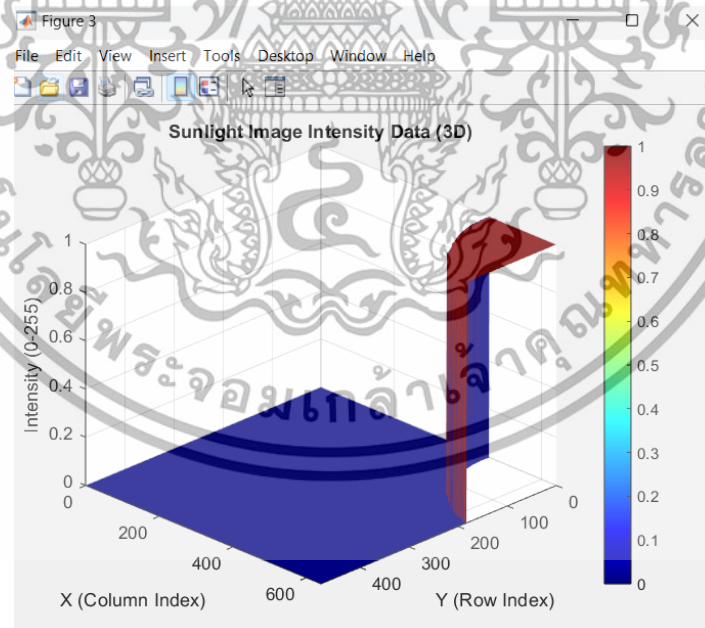


รูปที่ 4.65 การทดสอบโดยปรับค่า Threshold ผลการทดสอบโปรไฟล์ความเข้มแสง  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.66 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 2D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา



รูปที่ 4.67 การทดสอบโดยปรับค่า Threshold การทำ UV pattern 3D  
อยู่ในมุม Azimuth 20 องศา และมุม Elevation 15 องศา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.1 เปรียบเทียบค่ามุม Azimuth และมุม Elevation โดยอ้างอิง Reference การไม่ปรับค่า Threshold และการปรับค่า Threshold

Reference		ไม่ปรับค่า Threshold		ปรับค่า Threshold	
Az	El	Az	El	Az	El
0	0	-0.040915	0.47511	-0.58387	0.14473
-5	0	-3.0439	0.7238	-4.7043	0.82104
-10	0	-6.2545	-0.018896	-10.6117	-0.0030768
-15	0	-9.0621	0.66312	-14.9278	0.085252
-20	0	-11.1176	-0.4097	-19.9941	0.18718
5	0	2.9966	0.32011	5.4627	-0.2293
10	0	5.3466	0.13634	10.0697	-0.13322
15	0	8.6448	0.74645	15.2211	0.77394
20	0	11.006	-0.25538	20.1605	0.55894
0	-5	0.19716	-2.1108	-0.13156	-4.7864
-5	-5	-2.3847	-2.2648	-4.8737	-5.4667
-10	-5	-5.61	-2.7248	-10.0952	-5.1757
-15	-5	-8.9774	-2.778	-15.3795	-5.2382
-20	-5	-10.2735	-4.0208	-19.7904	-4.9057
5	-5	3.0925	-2.393	5.3679	-4.6309
10	-5	6.0574	-2.7319	10.0425	-5.4519
15	-5	9.5979	-2.8328	15.2777	-5.1787
20	-5	11.0653	-2.8513	20.2288	-4.8475
0	-10	0.1122	-5.4914	0.60487	-10.8065
-5	-10	-2.8986	-5.7951	-4.6686	-10.823
-10	-10	-6.2537	-5.1332	-9.8813	-8.9832
-15	-10	-9.8416	-5.979	-15.0062	-10.3267

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Reference		ไม่ปรับค่า Threshold		ปรับค่า Threshold	
Az	El	Az	El	Az	El
-20	-10	-6.9817	-5.9703	-19.556	-10.1219
5	-10	2.5997	-5.7838	5.1271	-10.335
10	-10	5.4982	-6.074	10.1064	-10.7306
15	-10	9.1185	-6.4224	14.8992	-10.3932
20	-10	5.4461	-3.6201	20.0393	-10.2048
0	-15	-0.38036	-8.1662	-0.20189	-14.8473
-5	-15	3.024	-8.2366	-5.3501	-15.1408
-10	-15	4.5061	-7.0494	-10.0096	-15.0742
-15	-15	3.9736	-4.9069	-15.3105	-15.5663
-20	-15	1.1233	-2.5959	-19.6033	-14.4946
5	-15	3.024	-8.2366	5.173	-14.6547
10	-15	4.5061	-7.0494	10.2776	-14.7834
15	-15	3.9736	-4.9096	15.3315	-14.6689
20	-15	1.1233	-2.5959	19.4245	-14.2998
0	5	-0.31301	2.4632	0.33812	4.7934
-5	5	-3.0285	2.758	-4.9231	5.4252
-10	5	-6.166	2.5046	-10.2522	4.6759
-15	5	-9.5484	2.7166	-15.226	5.1835
-20	5	-11.39	2.3366	-19.8589	4.5757
5	5	3.6124	2.3547	5.2536	5.1289
10	5	5.6382	2.5748	9.8689	5.7705
15	5	9.8665	2.1001	15.4124	4.7558
20	5	12.6029	3.2002	19.8441	5.2306
0	10	1.1768	5.5435	0.19503	10.579

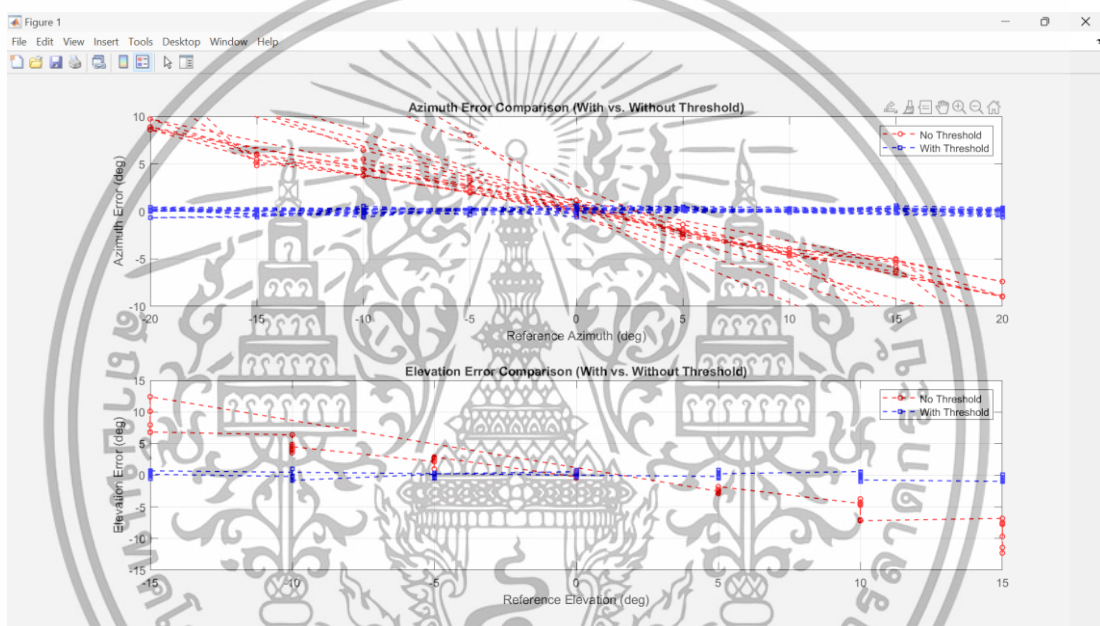
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Reference		ไม่ปรับค่า Threshold		ปรับค่า Threshold	
Az	El	Az	El	Az	El
-5	10	-1.7816	5.2596	-4.9373	10.4895
-10	10	-4.5046	5.4816	-9.4479	10.0254
-15	10	-10.1729	6.2597	-15.6107	10.0476
-20	10	-4.4145	2.9646	-20.6724	10.5767
5	10	2.2707	5.3879	5.2063	9.8456
10	10	6.0667	5.5615	10.204	8.9822
15	10	9.997	5.6979	15.245	9.6405
20	10	7.6795	2.8125	19.6265	9.2635
0	15	0.097831	8.1855	0.18087	14.0251
-5	15	-1.495	7.5507	-4.9106	14.6134
-10	15	-3.5274	7.2265	-10.1224	15.0816
-15	15	-4.4037	5.3281	-14.8212	14.1295
-20	15	-0.61901	3.2083	-20.1846	14.4573
5	15	2.8165	7.4073	5.2139	15.0665
10	15	5.321	7.2374	10.053	14.6042
15	15	4.0421	3.5738	15.5759	14.6742
20	15	2.7998	2.7216	20.3683	14.2401

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.5 ผลการทดสอบความแม่นยำ (Accuracy)

การทดสอบความแม่นยำของเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ ผลลัพธ์ที่ได้จากการทดสอบ ได้ทำการเปรียบเทียบความคลาดเคลื่อนของมุม Azimuth และ Elevation ระหว่างข้อมูลที่ไม่ได้ปรับค่า Threshold กับข้อมูลที่มีการปรับค่า Threshold เพื่อศึกษาว่าการปรับค่าส่งผลอย่างไร ต่อค่าความแม่นยำของการตรวจจับทิศทางแสงอาทิตย์ ดังในรูปที่ 4.68, 4.69 และตารางที่ 4.2, 4.3



รูปที่ 4.68 การเปรียบเทียบข้อผิดพลาดของมุม Azimuth และ Elevation ผลของการใช้ Threshold ต่อความแม่นยำในการวัดทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางที่ 4.2 การเปรียบเทียบข้อผิดพลาดของมุม Azimuth และ Elevation ผลของการใช้ Threshold ต่อความแม่นยำในการวัดทิศทาง

Az		El	
Without Threshold	With Threshold	Without Threshold	With Threshold
-0.040915	-0.58387	0.47511	0.14473
1.9561	0.2957	0.7238	0.82104
3.7455	-0.6117	-0.018896	-0.0030768
5.9379	0.0722	0.66312	0.085252
8.8824	0.0059	-0.4097	0.18718
-2.0034	0.4627	0.32011	-0.2293
-4.6534	0.0697	0.13634	-0.13322
-6.3552	0.2211	0.74645	0.77394
-8.994	0.1605	-0.25538	0.55894
0.19716	-0.13156	2.8892	0.2136
2.6153	0.1263	2.7352	-0.4667
4.39	-0.0952	2.2752	-0.1757
6.0226	-0.3795	2.222	-0.2382
9.7265	0.2096	-0.9792	0.0943
-1.9075	0.3679	2.607	0.3691
-3.9426	0.0425	2.2681	-0.4519
-5.4021	0.2777	2.1672	-0.1787
-8.9347	0.2288	2.1487	0.1525
0.1122	0.60487	4.5086	-0.8065
2.1014	0.3314	4.2049	-0.823
3.7463	0.1187	4.8668	1.0168
5.1584	-0.0062	4.021	-0.3267
13.0183	0.444	4.0297	-0.1219
-2.4003	0.1271	4.2162	-0.335

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

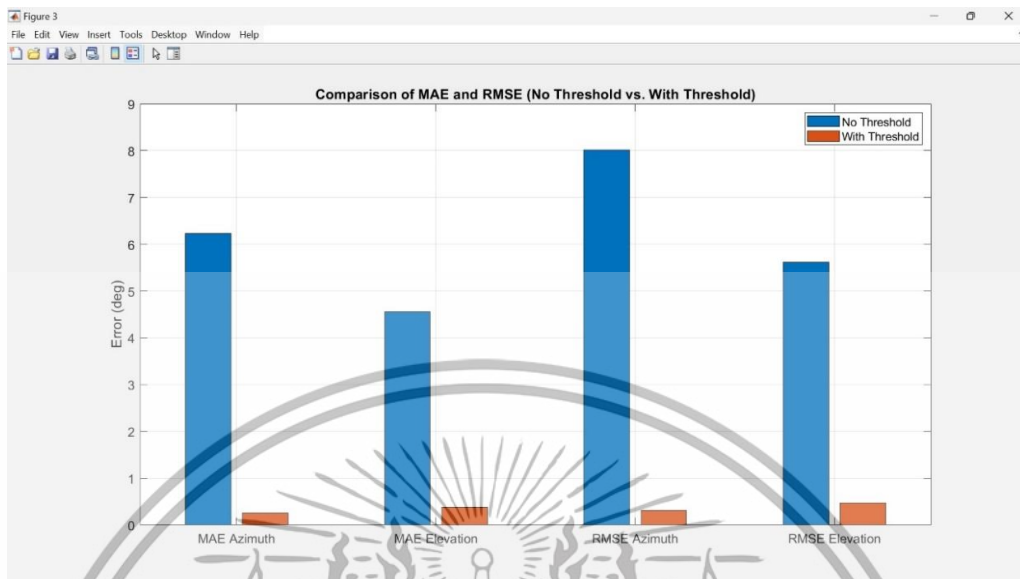
Az		El	
Without Threshold	With Threshold	Without Threshold	With Threshold
-4.5018	0.1064	3.926	-0.7306
-5.8815	-0.1008	3.5776	-0.3932
-14.5539	0.0393	6.3799	-0.2048
-0.38036	-0.20189	6.8338	0.1527
8.024	-0.3501	6.7634	-0.1408
14.5061	-0.0096	7.9506	-0.0742
18.9736	-0.3105	10.0931	-0.5663
21.1233	0.3967	12.4041	0.5054
-1.976	0.173	6.7634	0.3453
-5.4939	0.2776	7.9506	0.2166
-11.0264	0.3315	10.0904	0.3311
-18.8767	-0.5755	12.4041	0.7002
-0.31301	0.33812	-2.5368	-0.2066
1.9715	0.0769	-2.242	0.4252
3.834	-0.2522	-2.4954	-0.3241
5.4516	-0.226	-2.2834	0.1835
8.61	0.1411	-2.6634	-0.4243
-1.3876	0.2536	-2.6453	0.1289
-4.3618	-0.1311	-2.4252	0.7705
-5.1335	0.4124	-2.8999	-0.2442
-7.3971	-0.1559	-1.7998	0.2306
1.1768	0.19503	-4.4565	0.579
3.2184	0.0627	-4.7404	0.4895
5.4954	0.5521	-4.5184	0.0254
4.8271	-0.6107	-3.7403	0.0476
15.5855	-0.6724	-7.0354	0.5767

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Az		El	
Without Threshold	With Threshold	Without Threshold	With Threshold
-2.7293	0.2063	-4.6121	-0.1544
-3.9333	0.204	-4.4385	-1.0178
-5.003	0.245	-4.3021	-0.3595
-12.3205	-0.3735	-7.1875	-0.7365
0.097831	0.18087	-6.8145	-0.9749
3.505	0.0894	-7.4493	-0.3866
6.4726	-0.1224	-7.7735	0.0816
10.5963	0.1788	-9.6719	-0.8705
-2.1835	0.2139	-7.5927	0.0665
-4.679	0.053	-7.7626	-0.3958
-10.9579	0.5759	-11.4262	-0.3258
-17.2002	0.3683	-12.2784	-0.7599

การเปรียบเทียบข้อผิดพลาดของมุม Azimuth และมุม Elevation แสดงให้เห็นว่า การใช้ Threshold มีผลต่อการลดข้อผิดพลาด และเพิ่มความแม่นยำของการตรวจจับทิศทาง แสงอาทิตย์ได้อย่างแม่นยำ ในกรณีที่ไม่มีการใช้ Threshold ค่าข้อผิดพลาดของทั้งมุม Azimuth และมุม Elevation มีการกระจายตัวสูงมาก โดยมุม Azimuth ค่าข้อผิดพลาดกระจายตัวในช่วง ประมาณ  $\pm 10$  องศา และมีแนวโน้มที่จะเพิ่มขึ้นตามมุมอ้างอิงที่สูงขึ้น ส่วนในมุม Elevation ข้อผิดพลาดกระจายตัวมากขึ้นในช่วง  $\pm 15$  องศา ซึ่งแสดงถึงความไม่มีประสิทธิภาพ และความคลาดเคลื่อนที่เพิ่มขึ้นเมื่อค่ามุมอ้างอิงสูง หรือต่ำมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.69 ผลการทดสอบข้อผิดพลาดในการวัดมุม Azimuth และ Elevation

ตารางที่ 4.3 ผลการทดสอบข้อผิดพลาดในการวัดมุม Azimuth และ Elevation

Condition	MAE_Azimuth	MAE_Elevation	RMSE_Azimuth	RMSE_Elevation
Without Threshold	6.225862516	4.561554935	8.017942971	5.614063755
With Threshold	0.253858226	0.384748045	0.307945314	0.471910529

ข้อผิดพลาด (Error) ในการวัดมุม Azimuth กรณีที่ไม่ใช้ Threshold ค่าเฉลี่ยของข้อผิดพลาดแบบสัมบูรณ์ (Mean Absolute Error หรือ MAE) อยู่ที่ 6.23 องศา และค่า Root Mean Square Error (RMSE) อยู่ที่ 8.02 องศา ซึ่งหมายถึงการวัดมุม Azimuth มีข้อผิดพลาดค่อนข้างมากหากไม่มีการใช้ Threshold กรณีที่ใช้ Threshold ค่า MAE ลดลงเหลือ 0.25 องศา และค่า RMSE ลดลงเหลือ 0.31 องศา แสดงว่าการใช้ Threshold มีประโยชน์อย่างมากในการลดข้อผิดพลาดของการวัดมุม Azimuth

ข้อผิดพลาด (Error) ในการวัดมุม Elevation กรณีที่ไม่ใช้ Threshold ค่า MAE อยู่ที่ 4.56 องศา และค่า RMSE อยู่ที่ 5.61 องศา ซึ่งยังคงมีค่าข้อผิดพลาดที่ค่อนข้างสูง กรณีที่ใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

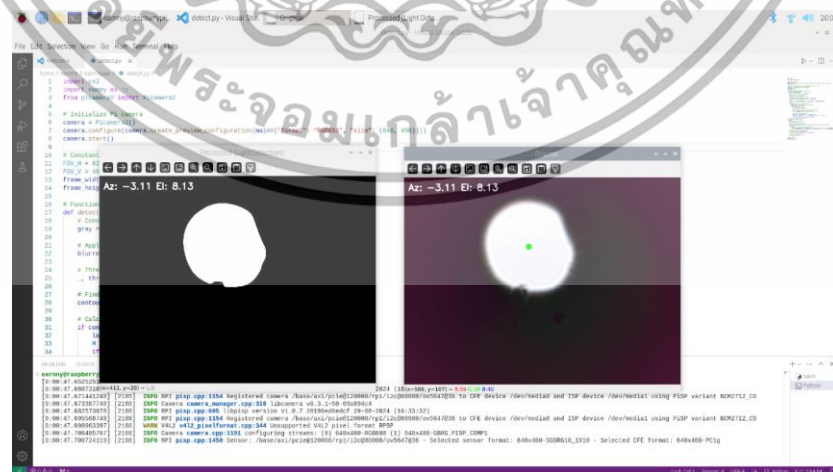
Threshold ค่า MAE ลดลงเหลือ 0.38 องศา และค่า RMSE ลดลงเหลือ 0.47 องศา แสดงให้เห็นว่าการใช้ Threshold ช่วยปรับปรุงความแม่นยำของการวัดมุม Elevation ได้อย่างชัดเจน

## 4.6 การทดสอบการคำนวณมุม

### 4.6.1 ผลลัพธ์การทดสอบการคำนวณมุมฝั่งภาคส่ง

ทดสอบการคำนวณมุมโดยใช้โปรแกรม Visual Studio Code ฝั่งภาคส่งดังรูปที่ 4.70 ผ่าน Raspberry Pi 5 โดยใช้สูตรที่ได้มาจากการวิเคราะห์ข้อมูลจากขั้นตอนที่ 4.4 และ 4.5 มีขั้นตอนดังนี้

- 1) ใช้กล้อง Pi ในการถ่ายภาพ และกำหนดค่าให้จับภาพที่มีขนาด 640x480 พิกเซล
- 2) กำหนด FOV ในแนวนอน 53.5 องศา และแนวตั้ง 41.41 องศา ตามสเปคกล้อง
- 3) แปลงภาพจากกล้องเป็น grayscale
- 4) ลด noise ด้วย Gaussian blur
- 5) ใช้ threshold เพื่อแยกส่วนที่สว่างออกมา
- 6) หา centroid (จุดศูนย์กลาง) ของส่วนที่สว่างที่สุดโดยใช้ contour และโมเมนต์
- 7) คำนวณมุม Azimuth และ Elevation โดยใช้ตำแหน่ง centroid และขนาดเฟรมและมุม FOV ตามสูตร
- 8) แสดงผลวงกลมที่จุด centroid และแสดงค่ามุม Az และ El บนหน้าจอทั้งภาพจริงจากกล้อง และภาพที่ตรวจจับแสงได้ในแบบ grayscale



รูปที่ 4.70 รูปการทดสอบการคำนวณมุมโดยใช้โปรแกรม Visual Studio Code ฝั่งภาคส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

#### 4.6.2 ผลลัพธ์การทดสอบมุมฝั่งภาครับ

ทดสอบการรับมุมโดยใช้โปรแกรม Visual Studio Code ฝั่งภาครับดังรูปที่ 4.71 ผ่าน Arduino แล้วนำไปแสดงผลบนเว็บไซต์ มีขั้นตอนดังนี้

- 1) รับค่ามุม Azimuth และ Elevation จากซีเรียลมอนิเตอร์ ของ Arduino และนำไปแยกค่ามุม Az และ El
- 2) บันทึกข้อมูลลง SQLite
- 3) ให้ API /get\_data และ /get\_logs สำหรับดึงข้อมูลมุมล่าสุด และดึงค่ามุม 10 รายการล่าสุดจากฐานข้อมูล
- 4) แสดงค่ามุมหน้าเว็บ

รูปที่ 4.71 รูปการทดสอบการรับมุมโดยใช้โปรแกรม Visual Studio Code ฝั่งภาครับ

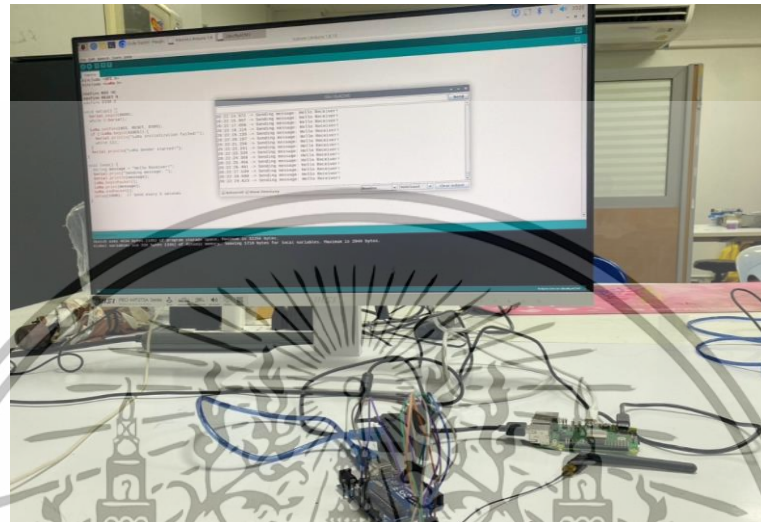
### 4.7 การทดสอบระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย ( TC & TM Subsystem )

#### 4.7.1 ผลลัพธ์การทดสอบระบบการสื่อสารทางด้านภาคส่ง

ทดสอบระบบการสื่อสารทางด้านภาคส่งดังรูปที่ 4.72 โดยทำการเชื่อมต่อดังที่ออกแบบแล้วทดสอบการทำงานของระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคพื้นดินแบบไร้สาย ( TC & TM Subsystem ) โดยเมื่อทำการทดสอบแล้วสามารถส่งข้อมูลแบบ ไร้สายได้จริงแสดงผลการทดสอบการส่งข้อมูลดังรูปที่ 4.73



รูปที่ 4.72 ทดสอบระบบการสื่อสารทางด้านภาคส่ง



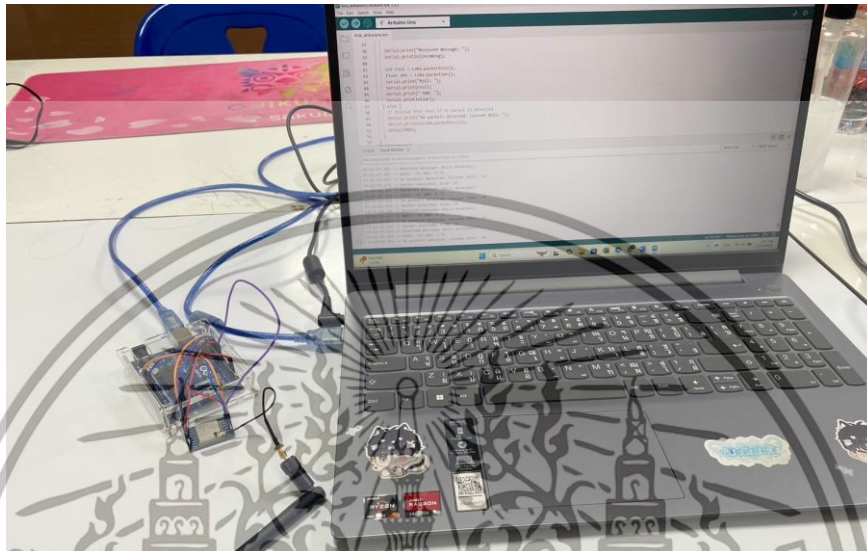
รูปที่ 4.73 แสดงผลลัพธ์การทดสอบการส่งข้อมูล

#### 4.7.2 ผลลัพธ์การทดสอบระบบการสื่อสารทางด้านภาครับ

ทดสอบระบบการสื่อสารทางด้านภาครับดังรูปที่ 4.74 โดยทำการเชื่อมต่อดังที่ ออกแบบแล้วทดสอบการทำงานของระบบการสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคพื้นดินแบบไร้สาย ( TC & TM Subsystem ) โดยเมื่อทำการทดสอบแล้วสามารถรับข้อมูลแบบไร้สายได้จริงแสดงผลการทดสอบการรับข้อมูลดังรูปที่ 4.75



รูปที่ 4.74 ทดสอบระบบการสื่อสารทางด้านภาครับ

```

11 int rssi = LOW;
12 float snr = LOW;
13 Serial.print("RSSI: ");
14 Serial.print(rssi);
15 Serial.print(" SNR: ");
16 Serial.print(snr);
17
18 // Print the RSSI even if no packets detected. Current RSSI is
19 Serial.print("No packets detected. Current RSSI is ");
20 Serial.print(rssi);
21 Serial.println();
22
23 }

```

Serial Monitor

```

20:12:52.114 -> Packet received, size: 15
20:12:52.737 -> Received message: Hello Received!
20:12:53.769 -> No packets detected. Current RSSI: -82
20:12:53.782 -> Packet received, size: 15
20:12:53.785 -> Received message: Hello Received!
20:12:53.788 -> RSSI: -82 SNR: 9.75
20:12:53.821 -> No packets detected. Current RSSI: -82
20:12:54.780 -> Packet received, size: 15
20:12:54.812 -> Received message: Hello Received!
20:12:54.845 -> RSSI: -82 SNR: 9.75
20:12:54.878 -> No packets detected. Current RSSI: -82
20:12:55.840 -> Packet received, size: 15
20:12:55.872 -> Received message: Hello Received!
20:12:55.904 -> RSSI: -82 SNR: 9.75
20:12:55.938 -> No packets detected. Current RSSI: -82
20:12:56.898 -> Packet received, size: 15
20:12:56.938 -> Received message: Hello Received!
20:12:56.991 -> RSSI: -82 SNR: 11.75
20:12:56.991 -> No packets detected. Current RSSI: -82

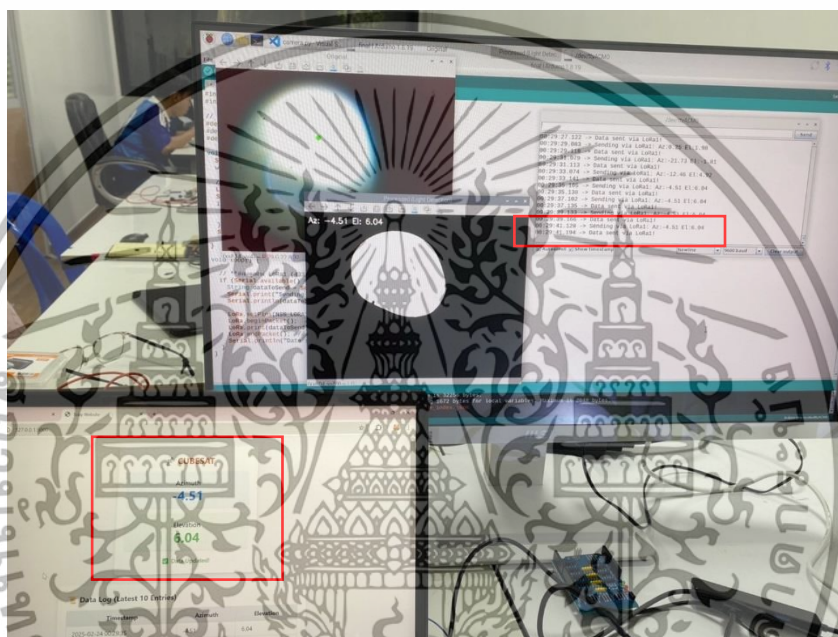
```

รูปที่ 4.75 แสดงผลลัพธ์การทดสอบการรับข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 4.7.3 ผลลัพธ์การทดสอบระบบการสื่อสารระหว่างภาคส่ง และภาครับ

ทดสอบระบบสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย ระหว่างภาคส่ง และภาครับ ดังรูปที่ 4.76 โดยจากรูปจะเห็นได้ว่ามุมที่ภาคส่งส่งมา ซึ่งคือมุมที่มีแสง ตกกระทบเข้ามา และมุมที่ภาครับรับได้ ที่แสดงบนเว็บไซต์เป็นมุมเดียวกัน แสดงให้เห็นว่าการรับส่ง มีประสิทธิภาพ และถูกต้องแม่นยำ



รูปที่ 4.76 ทดสอบระบบสื่อสารระหว่างดาวเทียมขนาดเล็ก และสถานีภาคพื้นดินแบบไร้สาย ระหว่างภาคส่ง และภาครับ

## 4.8 การทดสอบการรักษาความสมดุลของดาวเทียมขนาดเล็กโดยใช้ไจโรสโคป และมอเตอร์

### 4.8.1 ผลลัพธ์การทดสอบไจโรสโคปและวัดการเปลี่ยนแปลงของมุมในแต่ละแกน

ในการทดสอบไจโรสโคปเพื่อรักษาความสมดุลของดาวเทียมขนาดเล็ก โดยใช้เซ็นเซอร์ MPU-6050 ไจโรสโคปทำหน้าที่อ่านค่าความเร็วเชิงมุมในแต่ละแกน ได้แก่ Pitch (Y-axis), Yaw (Z-axis) และ Roll (X-axis) เมื่อดาวเทียมขนาดเล็กอยู่ในตำแหน่งสมดุล ค่าความเร็วเชิงมุมในทุกแกนจะเป็น 0

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการหมุนในแกน Roll (X-axis) พบว่าค่ามุม (Angle) มีการเปลี่ยนแปลง โดยเมื่อเอียงดาวเทียมขนาดเล็กไปทางซ้าย ค่ามุมจะลดลงจาก 360 องศาไปเรื่อย ๆ ตามระดับของการเอียง และเมื่อเอียงไปทางขวา ค่ามุมจะเพิ่มขึ้นจาก 0 องศาไปตามแนวการหมุน ซึ่งแสดงถึงการเปลี่ยนแปลงของมุมในระบบพิกัดเชิงมุมที่ค่ามุมถูกกำหนดให้อยู่ในช่วง 0–360 องศา ดังรูปที่ 4.77 และ 4.78

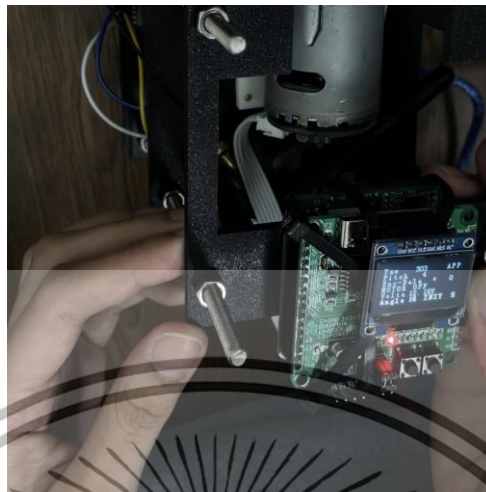
สำหรับการหมุนในแกน Pitch (Y-axis) เมื่อดาวเทียมขนาดเล็กอยู่ที่ตำแหน่งสมดุ ค่ามุมจะเป็น 0 องศา หากมีการเอียงไปทางด้านหน้า ค่ามุมจะเพิ่มขึ้นจาก 0 องศาไปเรื่อย ๆ ตามระดับของการเอียง ในทางกลับกัน หากเอียงไปทางด้านหลัง ค่ามุมจะลดลงโดยถอยกลับจาก 360 องศา ดังรูปที่ 4.79 และ 4.80

ในกรณีของการหมุนในแกน Yaw (Z-axis) ค่ามุมในตำแหน่งสมดุจะเริ่มต้นที่ 360 องศา เมื่อมีการหมุนตามเข็มนาฬิกา ค่ามุมจะลดลงเรื่อย ๆ จนกระทั่งถึง 0 องศา ซึ่งหมายถึงการกลับสู่ตำแหน่งเริ่มต้นของการหมุนแสดงถึงการหมุนครบหนึ่งรอบในระบบพิกัดเชิงมุม ดังรูปที่ 4.81 และ 4.82

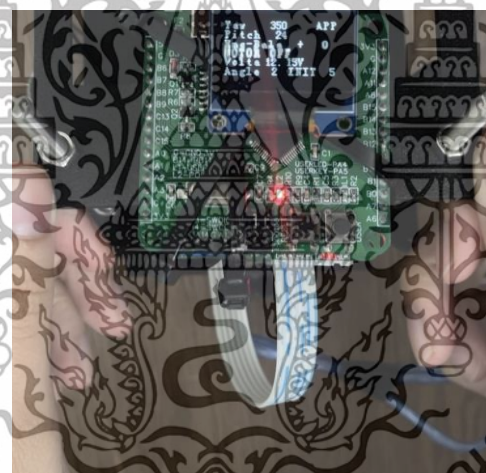


รูปที่ 4.77 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน Roll (X-axis) เมื่อเอียงดาวเทียมขนาดเล็กไปทางซ้าย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.78 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน Roll (X-axis) เมื่อเอียงดาวเทียมขนาดเล็กไปทางขวา

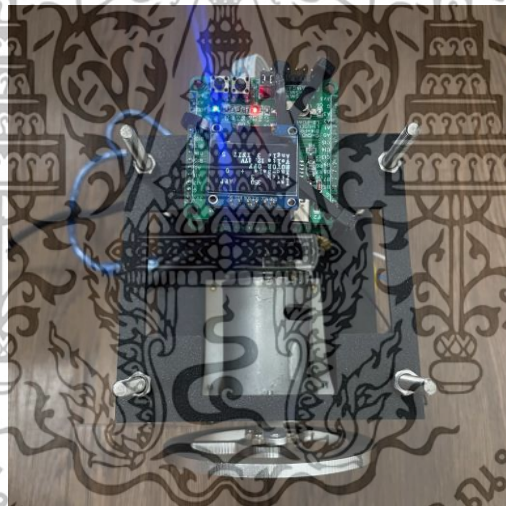


รูปที่ 4.79 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน Pitch (Y-axis) เมื่อเอียงดาวเทียมขนาดเล็กไปทางด้านหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

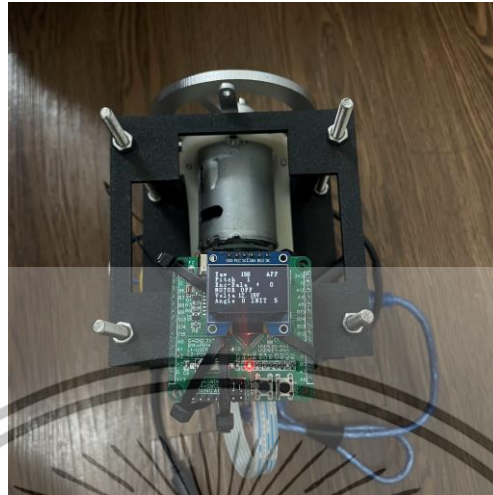


รูปที่ 4.80 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน Pitch (Y-axis) เมื่อเอียงดาวเทียมขนาดเล็ก  
ไปทางด้านหลัง



รูปที่ 4.81 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน Yaw (Z-axis) เมื่อดาวเทียมขนาดเล็กอยู่ใน  
จุดเริ่มต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



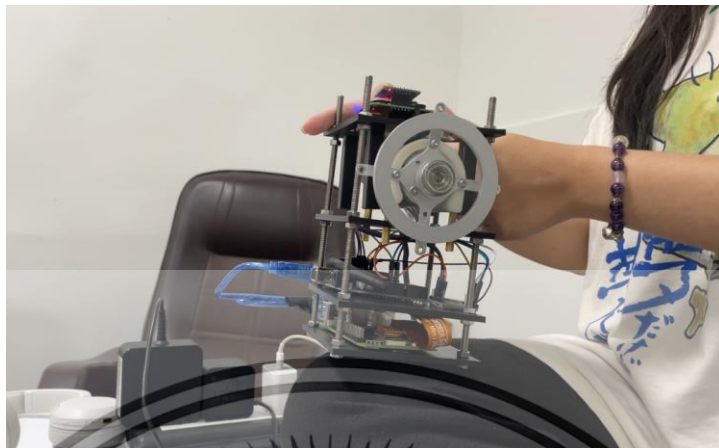
รูปที่ 4.82 จอ LED แสดงค่าการเปลี่ยนแปลงของแกน Yaw (Z-axis) เมื่อหมุนดาวเทียมขนาดเล็กตามเข็มนาฬิกา

#### 4.8.2 ผลลัพธ์การทดสอบมอเตอร์

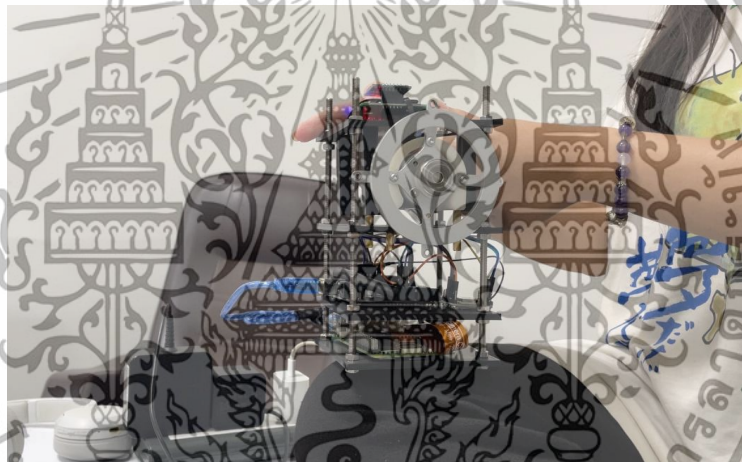
จากการทดสอบใจโรสโคปเบื้องต้น ได้นำใจโรสโคปมาติดตั้งร่วมกับมอเตอร์ (Reaction Wheel) เพื่อทำหน้าที่รักษาสมดุลของดาวเทียมขนาดเล็กในแนวแกน Roll (X-axis) เมื่อค่ามุมแสดงในจอ LED (Angle) มีการเปลี่ยนแปลงโดยเพิ่มขึ้นจาก 0 องศา ไปเรื่อย ๆ ระบบสั่งการให้มอเตอร์เริ่มหมุนในทิศทางที่เหมาะสม เพื่อดำเนินการเคลื่อนที่และปรับสมดุลให้ค่ามุม กลับคืนสู่ 0 องศา ซึ่งเป็นตำแหน่งสมดุลเดิม

ดังในรูปที่ 4.83 เมื่อดาวเทียมขนาดเล็กไม่อยู่ในจุดสมดุล มอเตอร์จะเริ่มหมุนเพื่อปรับทิศทางให้กลับสู่ตำแหน่งเดิม และเมื่อดาวเทียมขนาดเล็กเข้าสู่จุดสมดุล มอเตอร์จะค่อย ๆ หมุนช้าลงดังในรูปที่ 4.84

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.83 มอเตอร์จะเริ่มหมุนเมื่อดาวเทียมไม่อยู่ในจุดสมดุคที่กำหนด

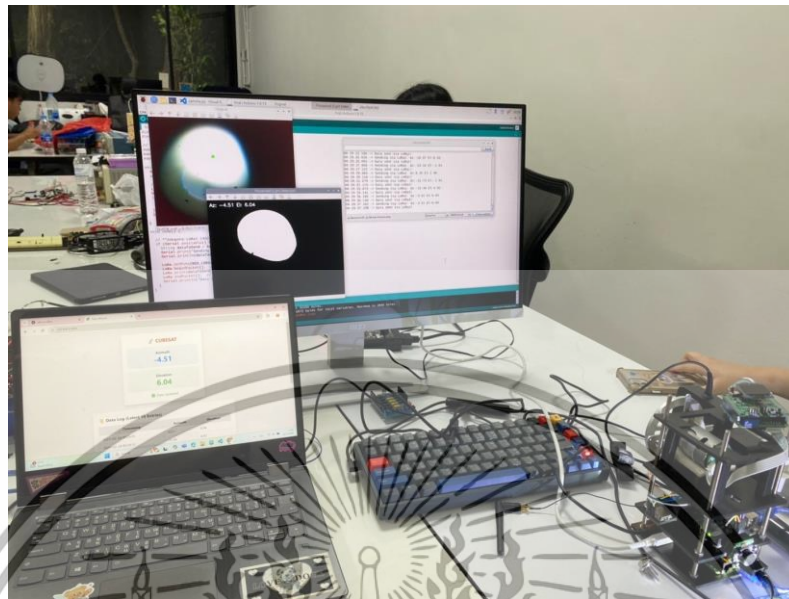


รูปที่ 4.84 มอเตอร์หมุนช้าลงเมื่อดาวเทียมขนาดเล็กเข้าใกล้จุดสมดุค

#### 4.9 การทดสอบระบบต่าง ๆ ของดาวเทียมขนาดเล็กเมื่อนำระบบมาใช้งานพร้อมกัน

การทดสอบเมื่อนำระบบที่ศึกษาทั้งหมดมาใช้งานร่วมกันดังรูปที่ 4.85 พบว่าระบบต่าง ๆ ที่ศึกษาสามารถทำงานพร้อมกันได้อย่างแม่นยำ โดยเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์สามารถตรวจจับแสง แล้วนำไปคำนวณมุมเพื่อส่งค่ามุมกลับมาที่ภาครับ ผ่านระบบสื่อสารได้แบบไร้สาย แล้วนำไปแสดงผลผ่านเว็บไซต์ได้เป็นค่าปัจจุบันที่ส่งมาจากเซ็นเซอร์ และสามารถควบคุมดาวเทียมให้อยู่ในจุดสมดุคได้ผ่านการใช้ไจโรสโคป และมอเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.85 ทดสอบการทำงานเมื่อนำทุกระบบใช้งานร่วมกัน



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5

### สรุปผลและข้อเสนอแนะ

#### 5.1 สรุปผล

ปริญญานิพนธ์นี้มีจุดประสงค์เพื่อนำเสนอการออกแบบ และจำลองระบบควบคุมตำแหน่ง และทิศทางของดาวเทียมขนาดเล็ก โดยระบบแรกเป็นระบบเซ็นเซอร์ตรวจจับทิศทางแสงอาทิตย์ใช้เซ็นเซอร์ CMOS ตรวจจับทิศทางแสงอาทิตย์ต้นทุนต่ำ การทดลองได้ใช้เซ็นเซอร์ CMOS ที่อยู่ภายใน Pi Camera แสดงภาพแสงจากไฟฉายซึ่งจำลองแทนแสงดวงอาทิตย์ที่ตกกระทบเซ็นเซอร์ผ่านหน้ากากวงกลมที่เจาะรูขนาด 1 มิลลิเมตร เพื่อนำไปคำนวณมุมของแสงที่ตกกระทบ จากการทดลองสามารถสรุปได้ว่าการแสดงผลการวัดค่ามุม Azimuth (Az) และ Elevation (El) บนจอมอนิเตอร์มีความแม่นยำของการตรวจจับทิศทางแสงอาทิตย์ ส่วนระบบที่สองเป็นการรักษาสมดุลของดาวเทียม โดยใช้ไมโครคอนโทรลเลอร์ MPU-6050 ที่มีคุณสมบัติของไจโรสโคป และมอเตอร์ในการหมุนในแนว Roll เป็นการเอียงซ้าย-ขวา เทียบจากแกนอ้างอิง X, Y และ Z จะเป็นการหมุนในแนวรอบแกน X เพื่อป้องกันไม่ให้เกิดการหมุนในมุมที่ไม่ต้องการ ซึ่งอาจทำให้ระบบต่าง ๆ ทำงานผิดพลาด ปริญญานิพนธ์นี้สามารถทำตามแผนงานที่วางไว้ได้ ทั้งสองระบบสามารถทำงานได้อย่างแม่นยำ

#### 5.2 ข้อเสนอแนะ

จากการทำปริญญานิพนธ์ ปัญหาที่พบคือ การจำลองการทำงานโดยใช้อุปกรณ์พื้นฐานซึ่งอาจยังขาดความเสถียรเพียงพอ หากต้องการนำไปใช้งานจริงในสภาพแวดล้อมจริง จำเป็นต้องเลือกใช้อุปกรณ์ที่มีคุณภาพสูง และน่าเชื่อถือมากขึ้น รวมถึงการพัฒนาระบบเพิ่มเติมเพื่อเพิ่มความแม่นยำในการทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม

- [1] Sciencedirect. “Attitude and Orbit Control System.”  
<https://gracefo.jpl.nasa.gov/attitude-and-orbit-control-system/#:~:text=The%20satellites'%20%E2%80%9Cattitude%2C%E2%80%9D,spacecraft%20rates%20and%20orbital%20position>
- [2] Josh Kaufman. “What Is A ‘Fail-Safe’?.”  
<https://personalmba.com/fail-safe/#:~:text=durable%20insurance%20policy,-,A%20Fail-Safe%20is%20a%20backup%20system%20designed%20to%20prevent,anywhere%20consistent%20performance%20is%20critical>
- [3] OmniVision Technologies, Inc. “color CMOS QSXGA (5 megapixel) image sensor with OmniBSI technology”  
[https://cdn.sparkfun.com/datasheets/Dev/RaspberryPi/ov5647\\_full.pdf](https://cdn.sparkfun.com/datasheets/Dev/RaspberryPi/ov5647_full.pdf)
- [4] ResearchGate. “High-precision CubeSat sun sensor coupled with infrared Earth horizon detector”  
[https://www.researchgate.net/figure/a-Working-principle-of-the-sun-sensor-b-The-influence-of-the-shape-of-the-hole-in\\_fig1\\_338899488](https://www.researchgate.net/figure/a-Working-principle-of-the-sun-sensor-b-The-influence-of-the-shape-of-the-hole-in_fig1_338899488)
- [5] Nattanon Sornchumni. “ImageProcessing: กราฟ Histogram และเบื้องหลังของภาพดิจิทัล”  
<https://medium.com/@nattanon.s/imageprocessing-1-กราฟ-histogram-และเบื้องหลังของภาพดิจิทัล-53bf9e032287>
- [6] ชัยพร ปานยินดี. Digital Image Processing การประมวลผลภาพดิจิทัล. พิมพ์ครั้งที่ 2. สำนักงานประชาสัมพันธ์ มหาวิทยาลัยเทคโนโลยีราชมงคลรัตนโกสินทร์, 2566.
- [7] The MathWorks, Inc. “intensity-profile-of-images”  
<https://www.mathworks.com/help/images/intensity-profile-of-images.html>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### บรรณานุกรม (ต่อ)

- [8] Young-Keun Chang, Seok-Jin Kang, Byung-Hoon Lee. “High-Accuracy Image Centroiding Algorithm for CMOS-Based Digital Sun Sensors”  
<https://ieeexplore.ieee.org/abstract/document/4388403>
- [9] ntpriinf. “การจัดการกับภาพด้วย OpenCV - การทำ Gaussian Blur ด้วย cv2.GaussianBlur()”  
[https://expert-programming-tutor.com/tutorial/article/KE003569\\_Manipulating\\_images\\_with\\_OpenCV\\_\\_Making\\_Gaussian\\_Blur\\_with\\_cv2GaussianBlur\\_.php](https://expert-programming-tutor.com/tutorial/article/KE003569_Manipulating_images_with_OpenCV__Making_Gaussian_Blur_with_cv2GaussianBlur_.php)
- [10] Chattichai. “บทที่ 5 มุม แบริ่งส์ และแอซิมัท.”  
<https://www.slideshare.net/slideshow/5-4738990/4738990>
- [11] Konstantin Bikos. “Altitude & Azimuth: The Horizontal Coordinate System.”  
<https://www.timeanddate.com/astronomy/horizontal-coordinate-system.html>
- [12] Julien Noua, Rémi Chauvina, Stéphane Thila,b, Stéphane Grieu. “A new approach to the real-time assessment of the clear-sky direct normal irradiance”  
[https://www.researchgate.net/publication/299413323\\_A\\_new\\_approach\\_to\\_the\\_real-time\\_assessment\\_of\\_the\\_clear-sky\\_direct\\_normal\\_irradiance](https://www.researchgate.net/publication/299413323_A_new_approach_to_the_real-time_assessment_of_the_clear-sky_direct_normal_irradiance)
- [13] MKT Admin. “LoRa คืออะไร? เกี่ยวข้องอย่างไรกับ IoT.”  
<https://www.success-network.co.th/what-is-lora/>
- [14] AllNewStep. “LoRa Arduino ESP8266 ESP32 IoT อะไรคือ LoRa ใช้งาน LoRa อย่างไร.”  
<https://www.allnewstep.com/article/136/1-lora-arduino-esp8266-esp32-iot-อะไรคือ-lora-ใช้งาน-lora-อย่างไร>
- [15] Roshni Y. “Attitude Control System.”  
[https://electronicsdesk.com/attitude-control-system.html#google\\_vignette](https://electronicsdesk.com/attitude-control-system.html#google_vignette)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม (ต่อ)

- [16] Gistda. “อะไร? ทำให้ดาวเทียมหมุนรอบตัวเอง เอียงไปเอียงมาได้.”  
[https://gistda.or.th/news\\_view.php?n\\_id=1281&lang=TH](https://gistda.or.th/news_view.php?n_id=1281&lang=TH)
- [17] Steven T. Bednarski. “CubeSat Attitude Determination and Control System (ADACS)”  
[https://s3vi.ndc.nasa.gov/ssri-kb/static/resources/CubeSat%20Attitude%20Determination%20and%20Control%20System%20\(ADACS\)%20Charact.pdf](https://s3vi.ndc.nasa.gov/ssri-kb/static/resources/CubeSat%20Attitude%20Determination%20and%20Control%20System%20(ADACS)%20Charact.pdf)
- [18] Satsearch. “Earth horizon sensors.”  
<https://blog.satsearch.co/2024-01-19-earth-sensors-and-horizon-sensors>
- [19] Cartrack. “GNSS คืออะไร.”  
<https://www.cartrack.co.th/blog/gnss-khuue-aair>
- [20] Howard D. Curtis. “Satellite Attitude Dynamics.”  
<https://www.sciencedirect.com/topics/engineering/nutation-damper#:~:text=Nutation%20dampers%20are%20added%20to,mass%20in%20the%20nutation%20damper.>
- [21] Raspberry Pi. “Raspberry Pi5”  
<https://www.raspberrypi.com/products/raspberry-pi-5/>
- [22] Cytron Technologies. “raspberrypi-camera”  
<https://th.cytron.io/c-raspberry-pi-camera/p-5mp-camera-board-for-raspberry-pi>
- [23] Alibaba.com. “บอร์ด ESP32 S2 Mini.”  
<https://thai.alibaba.com/product-detail/ESP32-S2-Mini-V1-0-0-1600552530818.html>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม (ต่อ)

- [24] Az-delivery. “MG90S Micro Servomotor kompatibel mit Arduino. ”  
<https://www.az-delivery.de/products/mg90s-micro-servomotor?shpxid=5779f24e-67e4-41a2-9a6c-805c83e1a7c9>
- [25] AnalogRead. “LoRa module 433MHz Ra-02 SX1278.”  
<https://www.success-network.co.th/what-is-lora/>
- [26] The invention. “ทำความรู้จัก : บอร์ด ไมโครคอนโทรลเลอร์ ARDUINO UNO R3 คืออะไรใช้งานอะไรได้บ้าง.”  
<https://www.ai-corporation.net/2021/11/19/arduino-uno-r3/>
- [27] Javanelec. “ไมโครคอนโทรลเลอร์ MCU.”  
<https://www.javanelec.com/Shops/ProductDetail/8649>
- [28] TDK invensense. “ไมโครคอนโทรลเลอร์ MPU-6050.”  
<https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/>
- [29] SLYPRO. “DC MOTOR RS-545 12V11000RPM.”  
<https://slypro.net/produit/dc-motor-rs-545-12v11000rpm/>
- [30] Siamreprint. “รูปเส้นพลาสติก Filament PLA.”  
<https://www.siamreprint.com/2020/01/how-to-select-3d-filament-for-3d-printer/#pla-filament>
- [31] I-LOADZONE. “Ultimaker Cura 5.6.0.”  
<https://i-loadzone.com/ultimaker-cura/>
- [32] The Invention. “Arduino IDE คืออะไร.”  
<https://www.ai-corporation.net/2021/11/18/what-is-arduino-ide/>
- [33] Learn Microsoft. “ใช้ส่วนขยาย Visual Studio Code.”  
<https://learn.microsoft.com/th-th/power-pages/configure/vs-code-extension>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บรรณานุกรม (ต่อ)

[34] KMUTT. “รู้จักกับ MATLAB.”

<https://shorturl.at/e03UN>

[35] DH Team. “ติดตั้ง DB Browser และเริ่มใช้งาน SQLite Database.”

<https://devhub.in.th/blog/db-browser-for-sqlite>



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ก

โครงการทดสอบไปรไฟล์ความเข้มแสง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

clc;
clear;

% Constants for camera field of view
FOV_H = 53.5; % Horizontal field of view in degrees
FOV_V = 41.41; % Vertical field of view in degrees

% Image dimensions (for angle calculations)
frame_width = 640;
frame_height = 480;

% Read the image
grayImage = imread('0_0.png');

% Convert to grayscale if the image is RGB
if size(grayImage, 3) == 3
    grayImage = rgb2gray(grayImage);
end

% Ensure the image is in the range [0, 255]
grayImage = uint8(min(max(grayImage, 0), 255));

% Threshold the image to isolate bright circular regions
thresholdLevel = 0.9; % Adjust threshold level if needed
binaryMask = imbinarize(grayImage, thresholdLevel);

% Remove small objects and fill holes in the mask
binaryMask = bwareaopen(binaryMask, 100); % Remove objects with less
than 100 pixels

binaryMask = imfill(binaryMask, 'holes');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

% Apply the mask to the grayscale image
isolatedCircle = grayImage .* uint8(binaryMask);

% Calculate the centroid of the isolated light area
stats = regionprops(binaryMask, 'Centroid');
if isempty(stats)
    disp('Not Found');
    return;
end
centroid = stats.Centroid; % [x, y] coordinates of the centroid

% Calculate the X and Y intensity profiles
xProfile = mean(isolatedCircle, 1);
yProfile = mean(isolatedCircle, 2);

% Normalize the profiles to the [0, 255] range
xProfile = uint8(255 * (xProfile / max(xProfile)));
yProfile = uint8(255 * (yProfile / max(yProfile)));

% Find peak intensity indices in the X and Y profiles
[~, maxIdxX] = max(xProfile);
[~, maxIdxY] = max(yProfile);

% Calculate image center coordinates
centerX = frame_width / 2;
centerY = frame_height / 2;

% Calculate Azimuth and Elevation angles based on the centroid
Az = (centroid(1) - centerX) / frame_width * FOV_H; % Azimuth

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

El = (centerY - centroid(2)) / frame_height * FOV_V; % Elevation (Y-axis
inverted)

% Display results in the console
disp(['Centroid of the circular light area: X = ', num2str(centroid(1)), ', Y = ',
num2str(centroid(2))]);
disp(['Azimuth Angle (Az): ', num2str(Az), ' degrees']);
disp(['Elevation Angle (El): ', num2str(El), ' degrees']);

% Plot the results in a 2x2 subplot layout
figure;
% Display the original image with the detected centroid
subplot(2, 2, 1);
imshow(binaryMask);
title('Sunlight Image and Intensity Using Thresholding Method (Test no.2)');
hold on;
plot(centroid(1), centroid(2), 'r+', 'MarkerSize', 15, 'LineWidth', 2);
hold off;

% Plot the X-axis intensity profile
subplot(2, 2, 3);
plot(xProfile);
title('X-axis Intensity Profile');
xlabel('Column Index');
ylabel('Mean Intensity (0-255)');
ylim([0 255]);
xlim([0 frame_width]);
hold on;
line([maxIdxX maxIdxX], [0 255], 'Color', 'g', 'LineWidth', 1.5, 'LineStyle', '--');
hold off;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

% Plot the Y-axis intensity profile
subplot(2, 2, 2);
plot(yProfile, 1:frame_height);
title('Y-axis Intensity Profile');
ylabel('Row Index');
xlabel('Mean Intensity (0-255)');
xlim([0 255]);
ylim([1 frame_height]);
set(gca, 'YDir', 'reverse'); % Reverse Y-axis to match image orientation
hold on;
line([0 255], [maxIdxY maxIdxY], 'Color', 'g', 'LineWidth', 1.5, 'LineStyle', '-');
hold off;

% Display the calculated Azimuth and Elevation angles
subplot(2, 2, 4);
text(0.1, 0.5, {'Azimuth (Az): ', num2str(Az, '%.2f'), '°'}, ...
      ['Elevation (El): ', num2str(El, '%.2f'), '°'], ...
      'FontSize', 12, 'FontWeight', 'bold');
axis off;
title('Calculated Angles');

% Generate 2D contour plot of intensity
[X, Y] = meshgrid(1:frame_width, 1:frame_height);
Z = double(binaryMask);

figure;
contourf(X, Y, Z, 20, 'LineColor', 'none');
colormap(jet);
colorbar;
title('Sunlight Image Intensity Data (2D)');
xlabel('X (Column Index)');

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

ylabel('Y (Row Index)');
axis equal;
set(gca, 'YDir', 'reverse');
% Generate 3D surface plot of intensity
figure;
surf(X, Y, Z, 'EdgeColor', 'none');
colormap(jet);
colorbar;
title('Sunlight Image Intensity Data (3D)');
xlabel('X (Column Index)');
ylabel('Y (Row Index)');
zlabel('Intensity (0-255)');
view(45, 30);
set(gca, 'YDir', 'reverse');

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

clc;
clear;

% Load data from Excel file
data = readmatrix('accuracy.xlsx', 'Range', 'A2:F100'); % Adjust the range as
needed

% Extract Reference angles (Azimuth and Elevation)
Az_ref = data(:, 1);
El_ref = data(:, 2);

% Extract Azimuth and Elevation without threshold (columns 3 and 4)
Az_no_threshold = data(:, 3);
El_no_threshold = data(:, 4);

% Extract Azimuth and Elevation with threshold (columns 5 and 6)
Az_with_threshold = data(:, 5);
El_with_threshold = data(:, 6);

% Handle NaN values by removing rows with NaNs
valid_indices = ~any(isnan([Az_ref, El_ref, Az_no_threshold,
El_no_threshold, Az_with_threshold, El_with_threshold]), 2);

% Use only valid rows (without NaNs)
Az_ref = Az_ref(valid_indices);
El_ref = El_ref(valid_indices);
Az_no_threshold = Az_no_threshold(valid_indices);
El_no_threshold = El_no_threshold(valid_indices);
Az_with_threshold = Az_with_threshold(valid_indices);
El_with_threshold = El_with_threshold(valid_indices);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

% Calculate errors for Azimuth and Elevation (without and with threshold)
Az_error_no_threshold = Az_no_threshold - Az_ref;
El_error_no_threshold = El_no_threshold - El_ref;
Az_error_with_threshold = Az_with_threshold - Az_ref;
El_error_with_threshold = El_with_threshold - El_ref;

```

```

% Plot Azimuth Error Comparison (With and Without Threshold - Raw
data)

```

```

figure;
% Plot Azimuth Error with and without Threshold in the same plot
subplot(2, 1, 1);
plot(Az_ref, Az_error_no_threshold, 'r--o', 'LineWidth', 1, 'MarkerSize', 4,
'DisplayName', 'No Threshold');
hold on;
plot(Az_ref, Az_error_with_threshold, 'b--s', 'LineWidth', 1, 'MarkerSize', 4,
'DisplayName', 'With Threshold');
xlabel('Reference Azimuth (deg)');
ylabel('Azimuth Error (deg)');
title('Azimuth Error Comparison (With vs. Without Threshold)');
xlim([-20, 20]); % Adjust X-axis limits as needed
ylim([-10, 10]); % Adjust Y-axis limits as needed
legend('show');
grid on;
hold off;

```

```

% Plot Elevation Error with and without Threshold in the same plot
subplot(2, 1, 2);
plot(El_ref, El_error_no_threshold, 'r--o', 'LineWidth', 1, 'MarkerSize', 4,
'DisplayName', 'No Threshold');
hold on;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

plot(El_ref, El_error_with_threshold, 'b--s', 'LineWidth', 1, 'MarkerSize', 4,
'DisplayName', 'With Threshold');
xlabel('Reference Elevation (deg)');
ylabel('Elevation Error (deg)');
title('Elevation Error Comparison (With vs. Without Threshold)');
xlim([-15, 15]); % Adjust X-axis limits as needed
ylim([-15, 15]); % Adjust Y-axis limits as needed
legend('show');
grid on;
hold off;

% Calculate MAE and RMSE for Azimuth and Elevation errors
MAE_Az_no_threshold = mean(abs(Az_error_no_threshold));
MAE_El_no_threshold = mean(abs(El_error_no_threshold));
MAE_Az_with_threshold = mean(abs(Az_error_with_threshold));
MAE_El_with_threshold = mean(abs(El_error_with_threshold));

RMSE_Az_no_threshold = sqrt(mean(Az_error_no_threshold.^2));
RMSE_El_no_threshold = sqrt(mean(El_error_no_threshold.^2));
RMSE_Az_with_threshold = sqrt(mean(Az_error_with_threshold.^2));
RMSE_El_with_threshold = sqrt(mean(El_error_with_threshold.^2));

% Create a table to display MAE and RMSE for each case
results_table = table(...
    ["No Threshold"; "With Threshold"], ...
    [MAE_Az_no_threshold; MAE_Az_with_threshold], ...
    [MAE_El_no_threshold; MAE_El_with_threshold], ...
    [RMSE_Az_no_threshold; RMSE_Az_with_threshold], ...
    [RMSE_El_no_threshold; RMSE_El_with_threshold], ...
    'VariableNames', {'Condition', 'MAE_Azimuth', 'MAE_Elevation',
    'RMSE_Azimuth', 'RMSE_Elevation'});

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

% Display the table
disp(results_table);

% Bar chart to compare MAE and RMSE for Azimuth and Elevation (No
Threshold vs. With Threshold)
figure;
metrics = {'MAE Azimuth', 'MAE Elevation', 'RMSE Azimuth', 'RMSE
Elevation'};
no_threshold_values = [MAE_Az_no_threshold, MAE_El_no_threshold,
RMSE_Az_no_threshold, RMSE_El_no_threshold];
with_threshold_values = [MAE_Az_with_threshold,
MAE_El_with_threshold, RMSE_Az_with_threshold, RMSE_El_with_threshold];
bar_data = [no_threshold_values; with_threshold_values];
bar(bar_data);
set(gca, 'XTickLabel', metrics);
legend('No Threshold', 'With Threshold');
ylabel('Error (deg)');
title('Comparison of MAE and RMSE (No Threshold vs. With Threshold)');
grid on;
set(gca, 'FontSize', 12);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ค

โค้ดการถ่ายภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

import cv2
import numpy as np
from picamera2 import Picamera2

# Initialize Pi camera
camera = Picamera2()
camera.configure(camera.create_preview_configuration(main={"format":
"RGB888", "size": (640, 480)}))
camera.start()

# Constants for camera field of view (adjust based on your camera specs)
FOV_H = 53.5 # Horizontal field of view in degrees
FOV_V = 41.41 # Vertical field of view in degrees
frame_width = 640
frame_height = 480

# Function to detect the centroid of the light area
def detect_light_centroid(frame):
    # Convert to grayscale
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    # Apply Gaussian blur to reduce noise
    blurred = cv2.GaussianBlur(gray, (15, 15), 0)

    # Threshold the image to isolate bright areas
    _, thresh = cv2.threshold(blurred, 240, 255, cv2.THRESH_BINARY)

    # Find contours of the light areas

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

# Ensure that at least one contour was found
if contours:
    # Calculate the centroid of the largest contour
    largest_contour = max(contours, key=cv2.contourArea)
    M = cv2.moments(largest_contour)
    if M["m00"] != 0:
        cX = int(M["m10"] / M["m00"])
        cY = int(M["m01"] / M["m00"])

        # Mark the centroid on the original and thresholded images
        cv2.circle(frame, (cX, cY), 7, (0, 255, 0), -1)
        cv2.circle(thresh, (cX, cY), 7, (255, 255, 255), -1)

        # Calculate Azimuth and Elevation
        azimuth = (cX - frame_width / 2) / frame_width * FOV_H
        elevation = (frame_height / 2 - cY) / frame_height * FOV_V # y is
        inverted

    return frame, thresh, (cX, cY), azimuth, elevation

# If no contours found, return None values
return frame, thresh, None, None, None

# Main loop for real-time detection
while True:
    # Capture frame from the Pi camera
    frame = camera.capture_array()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

# Detect centroid and calculate Az/El
original_frame, processed_frame, centroid, azimuth, elevation =
detect_light_centroid(frame)

if centroid:
    # Display Azimuth and Elevation on both windows
    #cv2.putText(original_frame, f"Az: {azimuth:.2f} El: {elevation:.2f}",
(10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255, 255, 255), 2)
    cv2.putText(processed_frame, f"Az: {azimuth:.2f} El: {elevation:.2f}",
(10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255, 255, 255), 2)

# Display the original and processed frames
cv2.imshow("Original", original_frame)
cv2.imshow("Processed (Light Detection)", processed_frame)

# Press 'q' to exit the loop
if cv2.waitKey(1) & 0xFF == ord('q'):
    break

# Release resources
camera.stop()
cv2.destroyAllWindows()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

import cv2
import numpy as np
from picamera2 import Picamera2
import time
import serial

# ตั้งค่า Serial Communication
ser = serial.Serial('/dev/ttyACM0', 9600, timeout=1) # กำหนด COM_PORT
ให้เป็นพอร์ตที่เชื่อมต่อกับ Arduino

# Initialize Pi camera
camera = Picamera2()
camera.configure(camera.create_preview_configuration(main={"format":
"RGB888", "size": (640, 480)}))
camera.start()

# Constants for camera field of view (adjust based on your camera specs)
FOV_H = 53.5 # Horizontal field of view in degrees
FOV_V = 41.41 # Vertical field of view in degrees
frame_width = 640
frame_height = 480

# Function to detect the centroid of the light area
def detect_light_centroid(frame):
    # Convert to grayscale]
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

    # Apply Gaussian blur to reduce noise
    blurred = cv2.GaussianBlur(gray, (15, 15), 0)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

# Threshold the image to isolate bright areas
_, thresh = cv2.threshold(blurred, 240, 255, cv2.THRESH_BINARY)

# Find contours of the light areas
contours, _ = cv2.findContours(thresh, cv2.RETR_EXTERNAL,
cv2.CHAIN_APPROX_SIMPLE)

# Ensure that at least one contour was found
if contours:
    # Calculate the centroid of the largest contour
    largest_contour = max(contours, key=cv2.contourArea)
    M = cv2.moments(largest_contour)
    if M["m00"] != 0:
        cX = int(M["m10"] / M["m00"])
        cY = int(M["m01"] / M["m00"])

    # Mark the centroid on the original and thresholded images
    cv2.circle(frame, (cX, cY), 7, (0, 255, 0), -1)
    cv2.circle(thresh, (cX, cY), 7, (255, 255, 255), -1)

    # Calculate Azimuth and Elevation
    azimuth = (cX - frame_width / 2) / frame_width * FOV_H
    elevation = (frame_height / 2 - cY) / frame_height * FOV_V # y is
inverted

    return frame, thresh, (cX, cY), azimuth, elevation

# If no contours found, return None values
return frame, thresh, None, None, None

```

# Variables to track time

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

last_print_time = time.time()

# Main loop for real-time detection
while True:
    # Capture frame from the Pi camera
    frame = camera.capture_array()

    # Detect centroid and calculate Az/El
    original_frame, processed_frame, centroid, azimuth, elevation =
detect_light_centroid(frame)

    if centroid:
        # Display Azimuth and Elevation on processed frame
        cv2.putText(processed_frame, f"Az: {azimuth:.2f} El: {elevation:.2f}",
(10, 30), cv2.FONT_HERSHEY_SIMPLEX, 0.7, (255, 255, 255), 2)

        # Print Azimuth and Elevation to the terminal every 2 seconds
        current_time = time.time()
        if current_time - last_print_time >= 2: # Check if 2 seconds have
passed
            print(f"Az: {azimuth:.2f} El: {elevation:.2f}")

            # Send the Azimuth and Elevation values to Arduino via Serial if
both values are not None
            if azimuth is not None and elevation is not None:
                current_data = f"Az:{azimuth:.2f} El:{elevation:.2f}"
                ser.write(current_data.encode()) # ส่งข้อมูลไปยัง Arduino
                print("Sent to LoRa1:", current_data)
            else:
                print("Azimuth or Elevation is None. Skipping data
transmission.")

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

last_print_time = current_time

# Display the original and processed frames
cv2.imshow("Original", original_frame)
cv2.imshow("Processed (Light Detection)", processed_frame)

# Press 'q' to exit the loop
if cv2.waitKey(1) & 0xFF == ord('q'):
    break

# Release resources
camera.stop()
cv2.destroyAllWindows()
ser.close()

```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <SPI.h>
#include <LoRa.h>
// กำหนดพินสำหรับ LoRa (433 MHz)
#define NSS_LORA1 10
#define RESET_LORA1 9
#define DIO0_LORA1 2
void setup() {
  Serial.begin(9600); // เปิดการสื่อสาร Serial
  while (!Serial);
  // เริ่มต้น LoRa (433 MHz)
  LoRa.setPins(NSS_LORA1, RESET_LORA1, DIO0_LORA1);
  Serial.println("Initializing LoRa (433 MHz)...");
  if (!LoRa.begin(433E6)) {
    Serial.println("LoRa initialization failed!");
    while (1);
  }
  Serial.println("LoRa initialized!");
}
void loop() {
  // ส่งข้อมูลผ่าน LoRa1 (433 MHz)**
  if (Serial.available() > 0) {
    String dataToSend = Serial.readString(); // รับข้อมูลจาก Serial Monitor
    Serial.print("Sending via LoRa1: ");
    Serial.println(dataToSend);
    LoRa.setPins(NSS_LORA1, RESET_LORA1, DIO0_LORA1);
    LoRa.beginPacket(); // เริ่มแพ็คเกจ LoRa
    LoRa.print(dataToSend); // ใส่ข้อมูลที่ต้องการส่ง
    LoRa.endPacket(); // ส่งข้อมูล
    Serial.println("Data sent via LoRa1!");
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

#include <SPI.h>
#include <LoRa.h>
// กำหนดพินสำหรับ LoRa (433 MHz)
#define NSS_LORA2 10
#define RESET_LORA2 9
#define DIO0_LORA2 2
void setup() {
  Serial.begin(9600); // เปิด Serial Monitor สำหรับรับข้อมูลจาก Visual Studio
  while (!Serial);
  // ตั้งค่า LoRa (433 MHz)
  LoRa.setPins(NSS_LORA2, RESET_LORA2, DIO0_LORA2);
  Serial.println("Initializing LoRa2 (433 MHz)...");
  if (!LoRa.begin(433E6)) {
    Serial.println("LoRa2 initialization failed!");
    while (1);
  }
  Serial.println("LoRa2 initialized!");
  void loop() {
    // รับข้อมูลจาก Arduino1 ผ่าน LoRa (433 MHz)**
    LoRa.setPins(NSS_LORA2, RESET_LORA2, DIO0_LORA2);
    if (LoRa.parsePacket()) {
      String dataFromArduino1 = "";
      while (LoRa.available()) {
        dataFromArduino1 += (char)LoRa.read();
      }
      Serial.print("Received from Arduino1: ");
      Serial.println(dataFromArduino1);
    }
  }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

from flask import Flask, jsonify, render_template
import serial
import re
import sqlite3
from datetime import datetime, timedelta

app = Flask(__name__)

SERIAL_PORT = "COM5"
BAUD_RATE = 9600

ser = None
last_az = None
last_el = None #

def init_db():
    conn = sqlite3.connect("data.db")
    cursor = conn.cursor()
    cursor.execute("""
CREATE TABLE IF NOT EXISTS az_el_log (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    azimuth REAL,
    elevation REAL,
    timestamp DATETIME DEFAULT CURRENT_TIMESTAMP
)
""")
    conn.commit()
    conn.close()
    print("ฐานข้อมูลพร้อมใช้งาน!")

def save_to_db(az, el):

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

conn = sqlite3.connect("data.db")
cursor = conn.cursor()

utc_now = datetime.utcnow()
thailand_time = utc_now + timedelta(hours=7)

cursor.execute("INSERT INTO az_el_log (azimuth, elevation, timestamp)
VALUES (?, ?, ?)",
               (az, el, thailand_time.strftime('%Y-%m-%d %H:%M:%S')))
conn.commit()
conn.close()
print(f"บันทึกข้อมูลสำเร็จ: Az={az}, El={el}, Time={thailand_time}")

def init_serial():
    global ser
    try:
        if ser is None or not ser.is_open:
            ser = serial.Serial(SERIAL_PORT, BAUD_RATE, timeout=1)
            print("Serial Port เปิดสำเร็จ!")
    except Exception as e:
        print(f"Error opening serial port: {e}")
        ser = None

@app.route('/get_data', methods=['GET'])
def get_data():
    global last_az, last_el

    if ser and ser.is_open:
        ser.flush()
        line = ser.readline().decode('utf-8').strip()

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if line:
    print(" 📡 Received:", line)
    match = re.search(r"Az:(-?\d+\.\d*)s+El:(-?\d+\.\d*)", line)
    if match:
        last_az, last_el = float(match.group(1)), float(match.group(2))
        save_to_db(last_az, last_el)
        return jsonify({"Az": last_az, "El": last_el})

    if last_az is not None and last_el is not None:
        return jsonify({"Az": last_az, "El": last_el})
    return jsonify({"error": "No valid data"}), 400

@app.route('/get_logs', methods=['GET'])
def get_logs():
    conn = sqlite3.connect("data.db")
    cursor = conn.cursor()
    cursor.execute("SELECT azimuth, elevation, timestamp FROM az_el_log
ORDER BY id DESC LIMIT 10")
    data = cursor.fetchall()
    conn.close()

    logs = [{"Azimuth": row[0], "Elevation": row[1], "Timestamp": row[2]} for
row in data]

    return jsonify(logs)

@app.route('/')
def index():
    return render_template("index.html")

```

```
if __name__ == '__main__':
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
init_db()
init_serial()
app.run(host='0.0.0.0', port=5000, debug=False)
```



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



ภาคผนวก ซ

โค้ดสำหรับผู้ใช้งานของเว็บไซต์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<!DOCTYPE html>
<html lang="th">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Toby Website</title>

  <!-- Tailwind CSS CDN -->
  <script src="https://cdn.tailwindcss.com"></script>
</head>
<body class="bg-gray-100 flex flex-col items-center min-h-screen p-6">
  <div class="bg-white shadow-lg rounded-lg p-6 w-96 text-center mb-6">
    <h1 class="text-2 xl font-bold text-orange-6 0 0 mb-4 " >
CUBESAT</h1>
    <div class="mb-4 p-4 bg-blue-50 rounded-lg shadow">
      <p class="text-lg font-semibold text-gray-700">Azimuth</p>
      <span id="az" class="text-4 xl font-bold text-blue-
500">Loading...</span>
    </div>

    <div class="mb-4 p-4 bg-green-50 rounded-lg shadow">
      <p class="text-lg font-semibold text-gray-700">Elevation</p>
      <span id="el" class="text-4 xl font-bold text-green-
500">Loading...</span>
    </div>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    <p id="status" class="mt-4 text-red-5 0 0 font-semibold"> Waiting
Data...</p>
</div>

```

```

<div class="bg-white shadow-lg rounded-lg p-6 w-full max-w-2xl">
  <h2 class="text-xl font-bold text-gray-700 mb-4"> Data Log (Latest
10 Entries)</h2>

```

```

  <table class="min-w-full border-collapse border border-gray-300">
    <thead>
      <tr class="bg-gray-200">
        <th class="border border-gray-3 0 0 px-4 py-
2">Timestamp</th>
        <th class="border border-gray-300 px-4 py-2">Azimuth</th>
        <th class="border border-gray-3 0 0 px-4 py-
2">Elevation</th>
      </tr>
    </thead>
    <tbody id="log-table">
      <tr><td colspan="3" class="text-center py-
2">Loading...</td></tr>
    </tbody>
  </table>
</div>

```

```

<script>
  async function fetchData() {
    try {
      let response = await fetch("/get_data");
      let data = await response.json();
      console.log(" Data received:", data);
    }
  }

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

if (data.Az !== undefined && data.El !== undefined) {
  document.getElementById("az").textContent = data.Az;
  document.getElementById("el").textContent = data.El;
  document.getElementById("status").textContent = " Data
Updated!";

  document.getElementById("status").classList.remove("text-
red-500");

  document.getElementById("status").classList.add("text-green-
500");
}
} catch (error) {
  console.error(" Error fetching data:", error);
  document.getElementById("status").textContent = " Error";
  document.getElementById("status").classList.add("text-red-500");
}
}
}

async function fetchLogs() {
  try {
    let response = await fetch("/get_logs");
    let logs = await response.json();
    console.log(" Logs received:", logs);

    let tableBody = document.getElementById("log-table");
    tableBody.innerHTML = "";
    logs.forEach(log => {
      let row = `
      <tr>
        <td class="border border-gray-3 0 0 px-4 py-
2">${log.Timestamp}</td>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

<td class="border border-gray-3 0 0 px-4 py-
2">${log.Azimuth}</td>

<td class="border border-gray-3 0 0 px-4 py-
2">${log.Elevation}</td>

</tr>

;
tbody.innerHTML += row;
});
} catch (error) {
console.error(" Error fetching logs:", error);
}
}
setInterval(fetchData, 1000); // รีเฟรชค่าล่าสุดทุก 1 วินาที
setInterval(fetchLogs, 5000); // รีเฟรช Log ทุก 5 วินาที
fetchData();
fetchLogs();
</script>
</body>
</html>

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้