

ASSUMPTION COLLEGE

CANSAT

AC-CS001

ASSUMPTION COLLEGE CANSAT

THE BASIC GUIDELINES FOR CANSAT CONSTRUCTION

2019 EDITION

Edited by Vivatsathorn Thitasirivit

สารบัญ

สารบัญ.....	1
I. Electronics, Sensors & Arduino Programming.....	3
บอร์ด Arduino คืออะไร.....	3
องค์ประกอบของบอร์ด Arduino.....	4
Port ต่าง ๆ บนบอร์ด Arduino ใช้งานอะไรได้บ้าง.....	6
คำสั่งในการเขียนโปรแกรม Arduino เบื้องต้น.....	11
ตัวอย่างเซนเซอร์และโมดูลที่ใช้กับ Arduino	14
ตัวอย่างโค้ดภาคส่ง.....	22
ตัวอย่างโค้ดภาครับ	23
II. Wireless Communication	24
การรับส่งสัญญาณแบบไร้สาย (Wireless Communication).....	24
คลื่นความถี่ของสัญญาณ (Signal Frequency)	24
ประเภทของเสาสัญญาณที่ควรรู้ (Types of Antennas)	25
1. Monopole Antenna (เสาเดี่ยว)	25
2. Yagi-Uda Antenna (เสาก้างปลา)	26
3. Satellite Dish (จานดาวเทียม).....	26
Radiation Pattern (เพิ่มเติม)	27
III. Parachute Physics & Design	31
1. การตกอย่างอิสระ.....	31
2. ร่มชูชีพ (Parachute)	32
แบบร่าง Cansat + Parachute	34
หารัศมีของ Parachute	34
IV. Surveying and Geography	37
V. Data Analysis & Analytics.....	49
1. ข้อมูลเชิงปริมาณ (Quantitative Data).....	49

1.1. การดูแนวโน้มของข้อมูลเชิงปริมาณ (Data Trends).....	49
1.2. การวิเคราะห์ข้อมูลเชิงปริมาณ	49
1.2.1. ค่าเฉลี่ยเลขคณิต (Arithmetic Mean).....	50
1.2.2. พิสัย (Range)	50
1.2.3. การสร้างความสัมพันธ์ของข้อมูลเพื่อเปรียบเทียบ	50
1.2.4. ค่าสัมประสิทธิ์สหสัมพันธ์ (Coefficient of Correlation, r).....	51
1.3. การนำเสนอข้อมูลเชิงปริมาณ (Data Visualization).....	51
1.3.1. ตาราง (Table)	51
1.3.2. กราฟ (Graph).....	52
1.3.3. แผนภูมิแท่ง (Bar Chart)	54
1.4. ไฟล์ประเภท CSV (Comma-separated Value)	54
1.5. การใช้ซอฟต์แวร์ Microsoft Excel เพื่อวิเคราะห์ผลข้อมูลเชิงปริมาณ	55
1.5.1. General User Interface	55
1.5.2. ชุดคำสั่ง (Functions).....	55
2. ข้อมูลเชิงคุณภาพ (Qualitative Data).....	59
2.1. การวิเคราะห์ด้วยสายตา (Visual Interpretation)	59
2.2. การแบ่งเกณฑ์ (Criteria).....	59
2.3. การใช้ซอฟต์แวร์ Google Earth Pro เพื่อพลอตพิกัดลงแผนที่.....	60
2.3.1. พลอตจากค่าพิกัด	60
2.3.2. พลอตจากไฟล์ CSV/TXT	60
CanSat Missions.....	61
ข้อกำหนด CanSat	61
ภารกิจหลัก (Primary Missions)	61
เพิ่มเติม	61
EXTRA NOTE	62
References	63

I. Electronics, Sensors & Arduino Programming

by Bhavat Ngamdeevilaisak

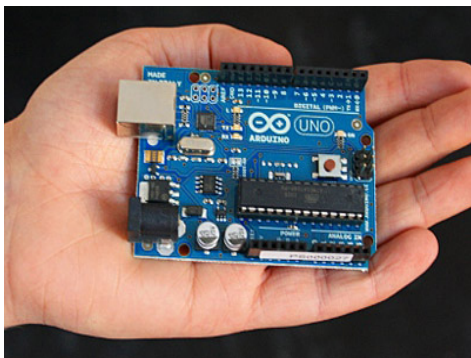
ภาคอิเล็กทรอนิกส์ (Electronics) สำหรับการสร้าง CanSat นั้น ถือว่าสำคัญเป็นอันดับต้น ๆ เพราะหากไม่มีอิเล็กทรอนิกส์เป็นส่วนประกอบในตัว CanSat คงต้องติดเครื่องวัดแบบเข็มขึ้นไปสำรวจเหมือนกับยุคเริ่มต้นของวิทยาศาสตร์ แต่ในสมัยนี้ การวัดด้วยอุปกรณ์อิเล็กทรอนิกส์มีความแม่นยำ น่าเชื่อถือ และทนทานต่อบางสภาวะมากกว่าอุปกรณ์ในสมัยก่อนมาก

ในบทนี้ จะพูดถึงอิเล็กทรอนิกส์พื้นฐาน อุปกรณ์ที่ใช้ในกระบวนการทำงานของ CanSat เช่น เซอร์คิวต์บอร์ด และการเขียนโปรแกรมเพื่อให้ CanSat ทำงานตามจุดประสงค์ที่ต้องการของภารกิจ (Mission) นั้น ๆ

ส่วนประกอบที่สำคัญ และเป็นตัวกลาง ส่วนกลางการประมวลผล และประสานงานการทำงานของระบบ ซึ่งเราจะ focus ในที่นี้ คือ Arduino

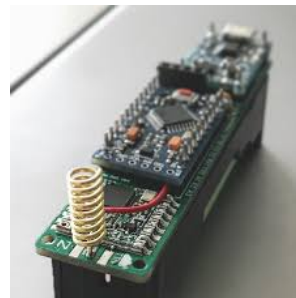
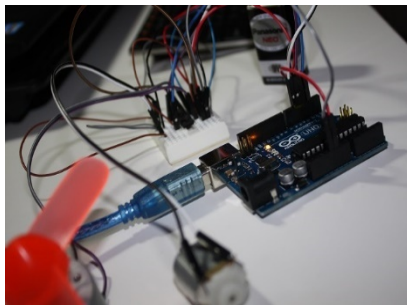
บอร์ด Arduino คืออะไร

บอร์ดไมโครคอนโทรลเลอร์ตระกูล Arduino คือบอร์ดที่ทำขึ้นโดยใช้ชิปไมโครคอนโทรลเลอร์ที่สามารถโปรแกรมผ่าน Arduino IDE ได้และง่ายต่อการใช้งาน บอร์ด Arduino มีหลายรุ่นซึ่งในเอกสารฉบับนี้จะมีรายละเอียดอยู่ 2 รุ่นซึ่งเป็นที่นิยมใช้ในงานทั่วไป คือ Arduino UNO และ Arduino Pro Mini ในบอร์ดทั้งสองรุ่นแม้จะใช้ชิปไมโครคอนโทรลเลอร์ ATMEGA328P เหมือนกันแต่ก็มีรายละเอียดต่าง ๆ ที่แตกต่างกัน ซึ่งจะชี้แจงในหัวข้อต่อไป



รูปภาพแสดงความแตกต่างทางกายภาพของ Arduino ทั้งสองรุ่น Arduino UNO (ซ้าย) Arduino Pro mini (ขวา)

จากภาพจะสังเกตว่า Arduino ทั้งสองบอร์ดนั้นมีขนาดที่แตกต่างกันเป็นอย่างมากซึ่งทำให้การใช้งานส่วนใหญ่ของ Arduino ทั้งสองบอร์ดแตกต่างกันไปด้วย ยกตัวอย่างเช่น Arduino UNO มักใช้ในงานออกแบบและการพัฒนางจร (การทำ Prototype) เนื่องจากมีขนาดใหญ่และสามารถเชื่อมต่อสายไฟได้ง่าย ส่วน Arduino Pro Mini นั้นมักใช้ในงานที่มีข้อจำกัดในเรื่องของพื้นที่ซึ่งมักจะเป็นงานที่ออกแบบเสร็จเรียบร้อยแล้วและกำลังจะนำไปประกอบเป็นวงจรเพื่อใช้งานจริง หรือในบางกรณีผู้ออกแบบวงจรอาจนำแคชชีฟ มาใช้โดยไม่ใช้ตัวบอร์ดเลยเพื่อลดขนาดอีกก็ได้เช่นกัน



ตัวอย่างการทำ Prototyping ด้วย Arduino UNO (ซ้าย)

ตัวอย่างวงจรขนาดเล็กที่ประกอบเสร็จโดยใช้ Arduino Pro Mini (ขวา)

องค์ประกอบของบอร์ด Arduino

องค์ประกอบของบอร์ด Arduino นั้น มีหลายชิ้นส่วน โดยผู้เขียนจะทำการจับกลุ่มให้เป็น 4 กลุ่มหลักให้เข้าใจได้ง่ายตามหน้าที่ของอุปกรณ์ คือ

1. Microcontroller Chip [เป็นส่วนหัวใจของบอร์ด Arduino ซึ่งทำหน้าที่ประมวลผลข้อมูลต่าง ๆ ตามโปรแกรมที่ผู้ใช้เขียนและต้องการ]

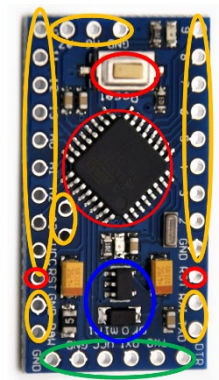
1.1. Microcontroller Chip

1.2. Reset Button & reset pin

2. USB to Serial Interface [เป็นเหมือนปากของ Arduino ที่ใช้พูดคุยสื่อสารและรับโปรแกรมกับคอมพิวเตอร์ผ่านทางพอร์ต USB]

2.1. USB Port

2.2. USB to Serial Converter chip

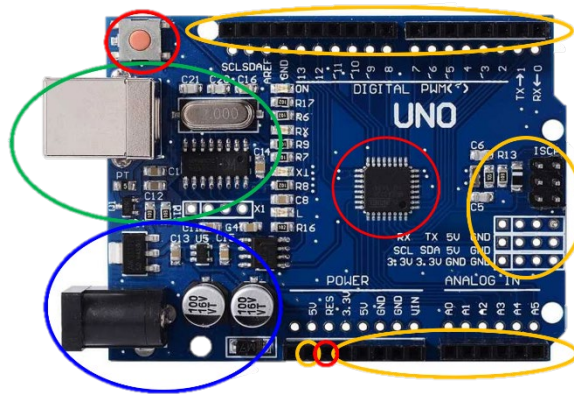


3. I/O & Power Connection [เป็นเหมือนแขนขาของ Arduino ที่ใช้รับอินพุตแสดงผล (OUTPUT) รวมไปถึงการสื่อสารกับเซนเซอร์ (Slave Device) และให้พลังงานกับเซนเซอร์ต่าง ๆ ด้วย]

- 3.1. Input/output (I/O) Connector
- 3.2. Serial (Rx/Tx) I²C (SCL/SDA) interface connector
- 3.3. V_{in} GND +5VDC +3.3VDC
- 3.4. Analog Reference (A_{ref}) pin

4. Power Management

- 4.1. DC Jack
- 4.2. Regulator Chip



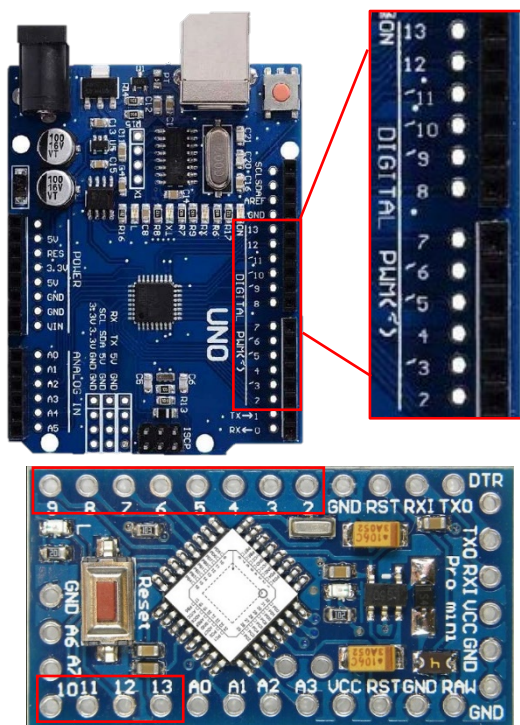
โดยที่ 4 กลุ่มนี้คือ 4 กลุ่มหลัก ๆ ขององค์ประกอบทั่วไปบนบอร์ด Arduino รุ่นมาตรฐานที่นิยมใช้กัน ขณะที่บอร์ด Arduino รุ่นพิเศษอื่น ๆ ก็จะมีองค์ประกอบพิเศษที่แตกต่างออกไป เช่น ส่วนติดต่อสื่อสารผ่านระบบไร้สาย เป็นต้น

*ยกเว้น Arduino Pro Mini จะไม่มีพอร์ตเชื่อมต่อกับUSB เนื่องจากตัวบอร์ดมีขนาดเล็กและจะใช้วิธีการโปรแกรมด้วย Arduino รุ่นอื่นผ่าน Serial Interface ทางด้านหลังของบอร์ดแทน

Port ต่าง ๆ บนบอร์ด Arduino ใช้งานอะไรได้บ้าง

ในบทนี้จะกล่าวถึงพอร์ต I/O และ Power Connection ที่ใช้ในการต่อเซนเซอร์ต่างๆรวมถึงคุณสมบัติของพอร์ตต่างๆว่ามีความแตกต่างกันอย่างไร เพราะบอร์ด Arduino นั้น 1 pin สามารถใช้งานได้มากกว่า 1 ฟังก์ชัน

1. **Digital pin** ใช้รับ/ส่งสัญญาณแบบดิจิทัลโดยมีสายไฟนำสัญญาณเป็นสื่อ (ความเป็นไปได้ของสัญญาณมี 2 แบบแต่สามารถเขียนแทนได้หลายลักษณะคือ [0,1], [HIGH, LOW], [+5V,0V])



Digital Pin นั้นมักนิยมใช้ในการรับค่าอินพุตที่เป็นลอจิก 0,1 เช่นจากสวิตช์กด จากเซนเซอร์ที่ Output เป็น ลอจิก 0,1 แต่ก็มีบางกรณีที่น่ามาใช้เป็นขาสื่อสารระหว่างอุปกรณ์โดยประยุกต์เอาเทคนิคต่างๆมาใช้ เช่นในกรณีของการควบคุม Servo Motor

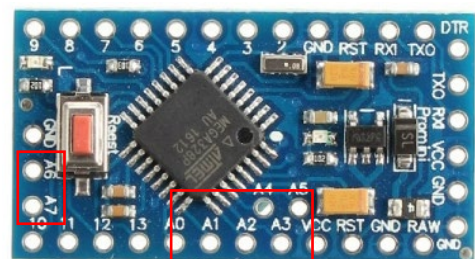
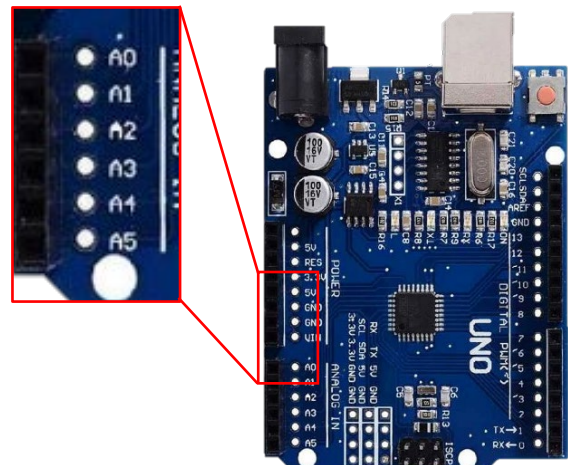
Digital pin จากการใช้งานข้างต้น จะมีคำสั่งที่พบได้บ่อยและควรทำความเข้าใจไว้คือ `digitalRead()` และ `digitalWrite()` โดยก่อนจะใช้งานในขั้นตอนของการเขียนโปรแกรมจะต้องมีคำสั่งบอกสถานะว่าจะใช้ pin นั้นไปทำอะไร เช่น `pinMode(2,OUTPUT)` หมายถึงจะใช้ digital pin 2 ของ Arduino เป็น OUTPUT และ `pinMode(3,INPUT)` หมายถึงจะใช้ digital pin 3 ของ Arduino เป็น INPUT เป็นต้น

2. **Analog pin** ใช้รับสัญญาณทางไฟฟ้าที่มาจากเซ็นเซอร์แอนะล็อก (แรงดันที่สามารถเปลี่ยนค่าได้ตั้งแต่ 0-5V เช่น 3.2V เป็นต้น) โดยขาอินพุตเหล่านี้จะนำสัญญาณแบบ analog ที่ได้ไปแปลงเป็นชุดข้อมูลแบบดิจิทัลด้วย ADC (Analog to Digital Converter) ซึ่ง ADC ของ Arduino นั้นมีความละเอียด 10 บิต หรือ 1024 ระดับของแรงดัน แปลว่า 1 ระดับจาก 1024 ระดับจะมีค่าเท่ากับ 4.9 mV ($5V \div 1024 \text{ step} = 0.00488V/\text{Step}$) เช่น อ่านค่าจาก ADC ได้ 135 แปลว่าที่อินพุตนั้นมีแรงดันเท่ากับ $0.00488 \times 135 \approx 0.66V$ เป็นต้น

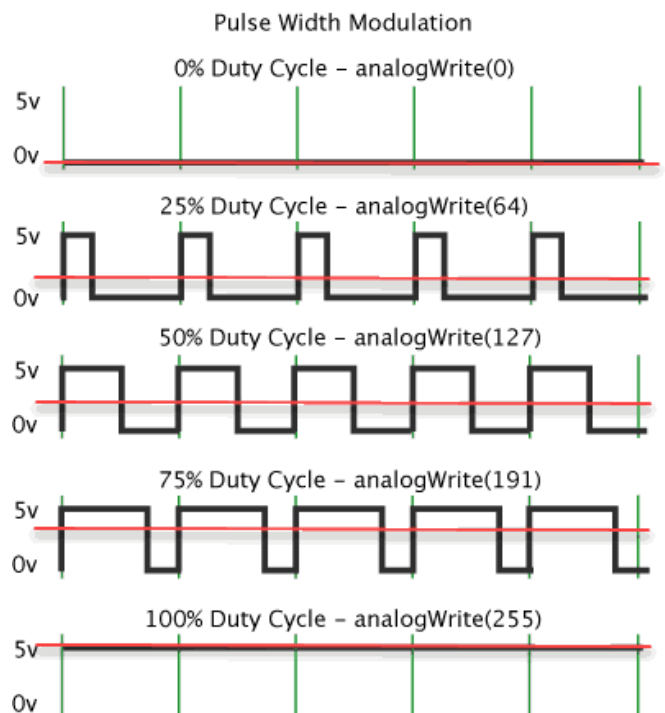
Analog pin นั้นมากใช้ในงานที่ต้องการอินพุตข้อมูลที่มีมากกว่าแค่ 0 กับ 1 โดยสามารถใช้เป็นอินพุตแบบดิจิตอลได้ด้วยเช่นกันโดยใช้คำสั่ง `analogRead()` หรือ `digitalRead()` ต่างกับ Digital pin ที่อินพุตได้แค่ข้อมูลแบบดิจิตอลเท่านั้น แต่ Analog pin ไม่สามารถใช้ในการ Output ข้อมูลได้

บน Arduino Pro Mini นั้นจะมี Analog input pin มากกว่า Arduino UNO อยู่ 2 pin คือ A6 และ A7

****กรณีที่ไม่มีขา SCL และ SDA สำหรับใช้เชื่อมต่อระบบ I²C จะสามารถใช้ขา A4 และ A5 แทนได้ (A4=SDA, A5=SCL)**



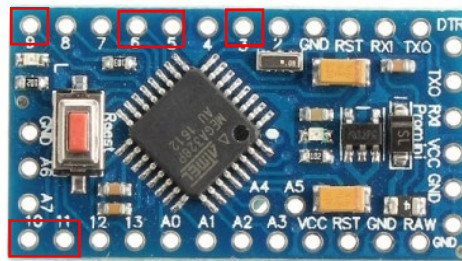
- PWM Pin** ตรงข้ามกับหัวข้อที่ผ่านมา PWM pin นั้นใช้ในการสร้างคลื่นไฟฟ้าแบบแอนะล็อกหรือ OUTPUT นั้นเองโดยใช้เทคนิคที่เรียกว่า Pulse Width Modulation คือการทำให้สัญญาณดิจิตอลติดกันด้วยความเร็วสูงโดยในหนึ่งช่วงคลื่นจะมีการควบคุมเวลาการติด-ดับให้ไม่เท่ากันเช่น สมมติให้แรงดันสูงสุดเป็น 5V หากหนึ่งช่วงคลื่นมีแรงดันลอจิก "1" (5V) 50% และ แรงดันลอจิก "0" 50% จะทำให้ค่าเฉลี่ยแรงดันรวมเป็น 2.5 V เป็นต้น โดย%อัตราการติดดับของสัญญาณไฟฟ้าต่อหนึ่งช่วงคลื่นจะเรียกว่า Duty Cycle เช่น Duty Cycle 80% หมายความว่าในหนึ่งช่วงคลื่นมีลอจิก "1" 80% และ ลอจิก "0" 20% โดยการสร้างคลื่น PWM จาก Arduino จะใช้ขาดิจิตอลที่มีสัญลักษณ์ "~" นำหน้าเท่านั้น หากใช้ขาอื่น จะไม่สามารถทำได้ โดยระดับความละเอียดของสัญญาณที่ออกมาจาก PWM Pin ของบอร์ด Arduino





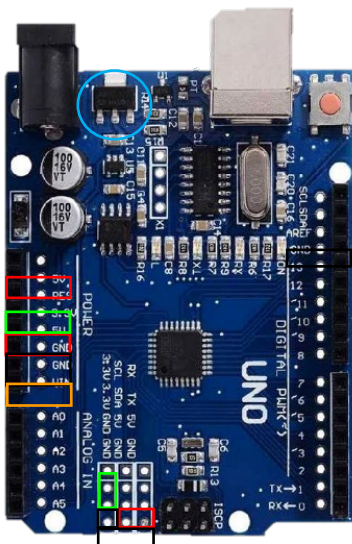
ที่ใช้ไมโครคอนโทรลเลอร์เบอร์ ATMEGA328P นั้นจะมีความละเอียด 8 บิต หรือ 256 ระดับ หมายความว่าบอร์ตลอจิก 5V ของ Arduino เมื่อใช้ฟังก์ชัน `analogWrite()` เพื่อสร้างคลื่น PWM นั้นจะมีความต่างของแรงดันต่อ 1 ระดับที่ $(5V \div 256 \text{ Steps} \approx 0.02 \text{ V/Steps})$ ยกตัวอย่างเช่น PWM ระดับ 135 คือ แรงดันเฉลี่ยที่ประมาณ 2.63V

*ในบอร์ด Arduino บางรุ่นที่มีขนาดเล็กเช่น Arduino Pro Mini จะไม่มีสัญลักษณ์เขียนบอกซึ่งในกรณีนี้ผู้ใช้ต้องอ่านจากคู่มือใน google เองตามรุ่นของบอร์ดนั้น



4. **Power Supply Pin** ในส่วนนี้จะกล่าวถึง Pin ต่างๆบนบอร์ด Arduino ทั้งสองรุ่น ที่ใช้จ่ายไฟเลี้ยงให้กับเซนเซอร์และอุปกรณ์ต่างๆที่ต่ออยู่กับบอร์ดรวมถึงข้อจำกัดของ power supply pin เหล่านี้

โดยในที่นี้ผู้เขียนจะขอจัดหมวดหมู่ไว้ 3 กลุ่มคือ 1.+5V 2.+3.3V 3.GND 4.Vin

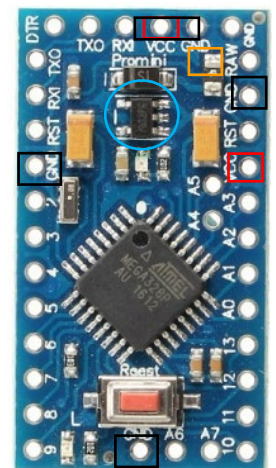


1.+5V คือแรงดันไฟเลี้ยง 5Vdc สำหรับเซนเซอร์และอินพุตต่างๆ หรับใน Arduino UNO เป็นการจ่ายไฟที่มาจากพอร์ต USB หรือ ไฟที่ผ่านการ Regulated มาจาก IC Regulator (ที่วงกลมสีฟ้า)

โดยการจ่ายกระแสสูงสุดขึ้นอยู่กับรุ่นของไอซีเรกกูเลเตอร์นั้นๆ เช่นของ Arduino UNO อยู่ที่ประมาณ 800 mA และของ Arduino Pro Mini อยู่ที่ประมาณ 150mA

2. +3.3V คือแรงดันที่ใช้จ่ายให้กับเซนเซอร์และโมดูลบางประเภทเช่น LoRa BME280

หรือโมดูลอื่น ๆ ที่ใช้แรงดัน 3.3Vซึ่งแรงดันนี้เกิดจาก ไอซีเรกกูเลเตอร์อีกชุด ซึ่งจ่ายกระแสได้สูงสุด



ขึ้นอยู่กับรุ่นของบอร์ด ของ Arduino UNO อยู่ที่ 150 mA และของ Arduino Pro Mini อยู่ที่ประมาณ 150mA

****กรณีของ Arduino Pro Mini จะมี 2 รุ่นคือ รุ่น 5V และ 3.3V โดย Vcc หรือขาไฟเลี้ยงเอาต์พุต นั้นจะขึ้นอยู่กับรุ่นของ Arduino เช่นรุ่น 3.3V ที่ขา Vcc ก็จะมีแรงดัน 3.3V ตลอดเป็นต้น**

3. GND กราวด์หรือขั้วลบ ใช้ต่อกับขั้วลบของโมดูลทุกโมดูล หรือในบางกรณีใช้เชื่อม Arduino มากกว่า 2 บอร์ดเข้าด้วยกันหรือบอร์ดลอจิกต่างๆเพื่อให้ใช้จุดเทียบความต่างศักย์เดียวกัน (Same Voltage Reference) บนบอร์ด Arduino UNO หรือบอร์ด Arduino รุ่นอื่นที่มีการจ่ายไฟเลี้ยงทั้ง 3.3V และ 5V บนบอร์ดเดียวกัน ก็สามารถใช้กราวด์ขาไหนก็ได้เหมือนกันเพราะทั้งสองความต่างศักย์ใช้กราวด์เดียวกันอยู่ไม่จำเป็นต้องแยกกราวด์แต่อย่างใด

4. Vin หรือบนบอร์ด Arduino Pro Mini เรียกว่า RAW ใช้ในการจ่ายไฟเข้านอกจากการเสียบสาย USB หรือพอร์ตโปรแกรม ซึ่งในบอร์ดทั้งสองรุ่นจะสามารถจ่ายไฟเข้าได้เหมือนกับ DC jack คือ 7-12V สำหรับ Arduino UNO และ 3.34-12V หรือ 5.5-12V สำหรับ Arduino Pro Mini รุ่น 3.3V และ 5V ตามลำดับโดยแรงดันเหล่านี้จะไปผ่าน Regulator ของบอร์ดเพื่อจ่ายเป็นไฟเลี้ยงแบบปกติต่อไป **อันที่จริงเราสามารถป้อนไฟเข้าผ่านทางพอร์ต 5V ได้เช่นกันแต่ทางผู้เขียนไม่แนะนำให้ทำเนื่องจากอาจสามารถทำให้ชิพเสียหายอย่างถาวรหากเกิดความผิดพลาดขึ้น**

5. Others Protocol Pin ในหัวข้อนี้จะกล่าวถึงพอร์ตเชื่อมต่อที่ใช้เชื่อมต่อเซนเซอร์หรือโมดูลอเนกประสงค์ต่างๆเข้ากับบอร์ด Arduino โดยใช้โปรโตคอลพิเศษ ซึ่งที่ Arduino รุ่นทั่วไปซัพพอร์ตคือ 1. I²C 2. SPI 3. UART โดยจะอธิบายทีละอย่างดังต่อไปนี้ ****ชื่อเรียกสายสัญญาณอาจแตกต่างกันตามบริษัทผู้ผลิต เช่น SCL บางบริษัทอาจใช้ SCK เป็นต้น**

1. I²C เป็นโปรโตคอลที่ใช้สายไฟสองเส้น คือ SDA(Serial Data) และ SCL(Serial Clock) โดยลักษณะการใช้งานจะเป็นการนำอุปกรณ์ Slave มาต่อขนานกันบนสาย SDA และ SCL กับอุปกรณ์ Master และ Slave ตัวอื่นๆ โดยที่ SCL ใช้กำหนดจังหวะการรับ/ส่งข้อมูล และ SDA ใช้ส่งข้อมูล เช่นในกรณีที่ SCL เป็นลอจิก “1” MCU จะทำการอ่านข้อมูลที่ SDA หากในขณะนั้นที่ SDA เป็น “0” ก็จะหมายความว่าบิตนั้นเป็นลอจิก “0” เป็นต้น

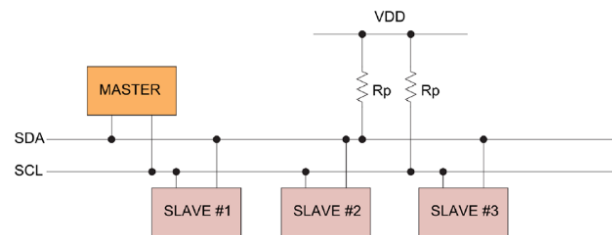
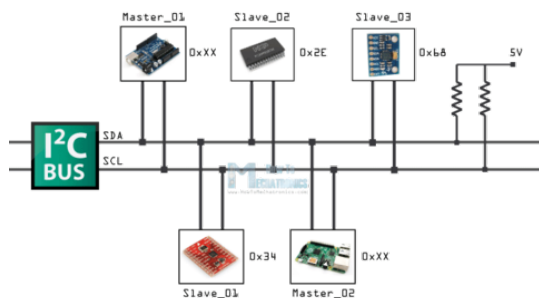
ข้อดี : เชื่อมต่ออุปกรณ์ได้หลายชิ้นโดยใช้สายแค่สองเส้น

มีความซับซ้อนต่ำทำให้สามารถตรวจดูแลและซ่อมแซมได้ง่าย

ข้อเสีย : มีความเร็วในการส่งสัญญาณต่ำทำให้ไม่สามารถส่งข้อมูลปริมาณมากๆได้

หากอุปกรณ์ตัวใดตัวหนึ่งทำงานพลาดจะทำให้ทั้งระบบรวนได้ง่าย

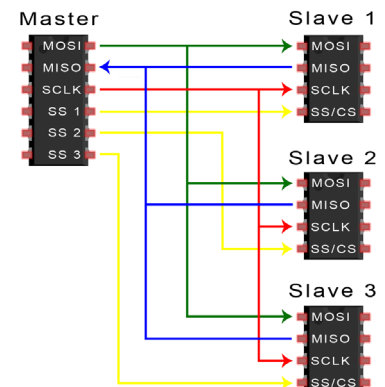
ความเร็วในการสื่อสารจะขึ้นกับอุปกรณ์ในระบบหากมีตัวใดทำงานช้าจะทำให้ทั้งระบบช้าไปด้วยและความเร็วในการสื่อสารของทั้งระบบจะมีค่าเท่ากับความเร็วในการสื่อสารของโมดูลที่ทำงานได้ช้าที่สุด



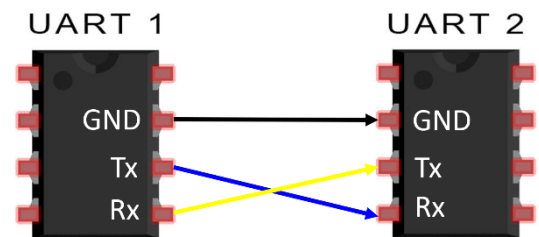
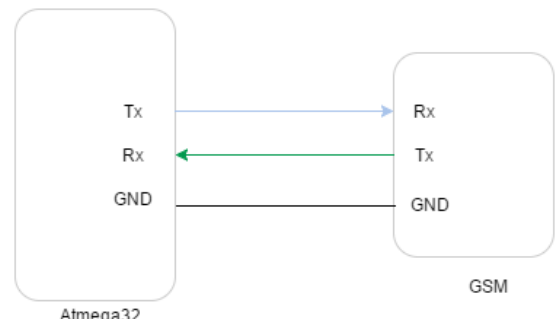
2. SPI เป็นโปรโตคอลที่ใช้สายไฟ 4 เส้นคือ SCK (Serial Clock) MISO (Master IN Slave OUT) MOSI (Master Out Slave In) SS/CS (Slave Select/Chip Select) สำหรับลักษณะการใช้งานจะคล้ายกับโปรโตคอล I²C คือนำอุปกรณ์ Slave ไปต่อแบบขนานบนสาย SCK MISO และ MOSI แต่จะแตกต่างตรง CS คืออุปกรณ์ 1 ชิ้นจะต้องใช้ ขา I/O ของ MCU 1 ขา หรือก็คือนอกจากสายสื่อสารที่ใช้ร่วมกัน 3 เส้นแล้วจะต้องมีสายที่ต้องใช้เพิ่มตามจำนวนอุปกรณ์ Slave อุปกรณ์ละ 1 เส้นนั่นเอง

ข้อดี : มีความเร็วในการสื่อสารที่มากกว่า I²C มากโดยที่ยังคงแนวคิดที่ใช้สายไฟน้อยแต่เชื่อมต่อกับอุปกรณ์ได้มากไว้บางส่วน

ข้อเสีย : หากมีอุปกรณ์ Slave ที่เยอะเกินไป จะไม่สามารถทำงานได้เพราะไม่มี I/O ที่มากพอให้ติดตั้งสาย CS



3. UART เป็นโปรโตคอลที่นิยมใช้กับการสื่อสารที่ต้องการความเร็วสูงหรือต้องการส่งข้อมูลปริมาณมาก รวมถึงเป็นโปรโตคอลที่ใช้แปลงสัญญาณจากพอร์ต USB ไปเป็น สัญญาณที่เขียนใช้โปรแกรม หรือ ให้ Arduino สื่อสารกับคอมพิวเตอร์ ด้วย โดยในโปรโตคอลหากเป็นการสื่อสารทางเดียวสามารถใช้ได้โดยมีสายไฟเพียงเส้นเดียวเท่านั้น แต่ในส่วนใหญ่จะมีสายไฟสองเส้นเพื่อความเร็ว คือสายส่ง สายรับ และสายกราวด์ในทุกกรณี เพื่อให้มีจุดเปรียบเทียบแรงดันเดียวกัน สายส่งสัญญาณจะเรียกว่า Tx (Transmitter) Rx (Receiver) โดยที่ Tx ของตัวส่งจะต้องต่อกับ Rx ของตัวรับ ตัวอย่างเช่น เซอร์ที่ ใช้โปรโตคอลนี้เช่น GPS (ATGM336H)



ข้อดี : รวดเร็วและใช้สายไฟน้อย

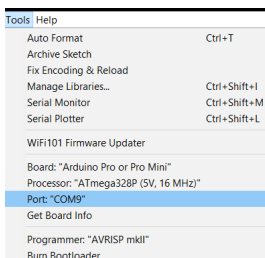
ข้อเสีย : มีโอกาสสูญเสียข้อมูลระหว่างการส่งสัญญาณและความเร็วสูงสุดจำกัดที่ความเร็วของอุปกรณ์ที่ช้าที่สุด


คำสั่งในการเขียนโปรแกรม Arduino เบื้องต้น

ในบทนี้จะกล่าวถึงคำสั่งพื้นฐานในการเขียนโปรแกรม Arduino โดยอ้างอิงพื้นฐานความรู้จากภาษา C และจะกล่าวถึงเฉพาะคำสั่งพื้นฐานที่ใช้ทั่วไปเท่านั้นจะไม่กล่าวถึงคำสั่งเฉพาะจาก Library อื่นที่ใช้เฉพาะงาน

การเขียนโปรแกรม Arduino จะต้องอาศัยโปรแกรมอินเทอร์เฟซที่เรียกว่า Arduino IDE สำหรับพัฒนาและเป็น Compiler ซึ่งสามารถดาวน์โหลดได้ที่ <https://www.arduino.cc/en/Main/Software>

1.การใช้งาน Arduino IDE เมื่อเขียนโปรแกรมเสร็จแล้วก็ต้องมีการอัปโหลดโปรแกรม ซึ่งทำได้ด้วย



การกด Ctrl+U หรือคลิกที่  บริเวณด้านบนซ้ายของจอ โดยก่อนการ Upload จะต้องเลือกประเภทของบอร์ดและพอร์ตสื่อสารให้ถูกต้อง โดยการกด Tools -> Board -> เลือกบอร์ดที่เราใช้ และกด Tools -> Board -> port ที่ต่ออยู่ (ในภาพจะเป็น COM9 แต่ไม่จำเป็นต้องเป็น COM9 เสมอไป แต่จะไม่ใช้ COM1 แน่นนอน) * ข้อสังเกตคำสั่งต่างๆในโปรแกรมจะมีสีต่างกันไปตามประเภทดังนั้นหากพิมพ์แล้วไม่มีสี

หรือสีไม่เหมือนคำสั่งในประเภทเดียวกันให้สงสัยว่าพิมพ์ผิด

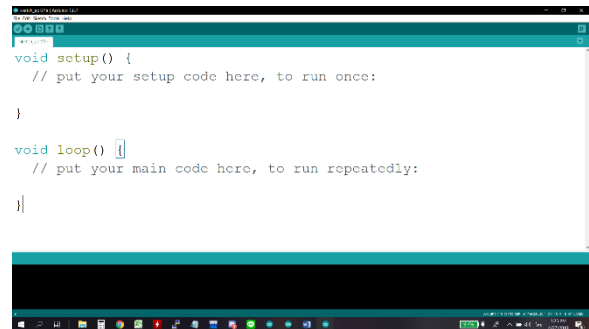
2. คำสั่งพื้นฐานในภาษา Arduino สำหรับการเขียนโปรแกรมในภาษานี้ ไวยากรณ์หรือ Syntax จะคล้ายๆกับภาษาซี เนื่องจากเป็นภาษาที่พัฒนามาจากภาษาซี ทำให้ง่ายต่อการเข้าใจสำหรับผู้ที่เคยผ่านตาภาษาซีมาบ้าง เพราะบางคำสั่งใช้เหมือนกัน 100% เช่น for loop while loop if...else int float เป็นต้น สำหรับคำสั่งที่ไม่มีในภาษาซีก็จะมีการใช้งานที่คล้ายๆกันคือชื่อคำสั่งจะบ่งบอกการทำงาน และต้องมี “;” ต่อท้ายคำสั่งยกเว้นกรณีที่เป็นคำสั่งที่มีการทำงานมากกว่า 1 บรรทัดเช่น if while for เป็นต้น และการอธิบายในบทนี้จะข้ามคำสั่งพื้นฐานที่ซ้ำกับคำสั่งในภาษาซีเช่น int char #include #define float long bool while for if else ฯลฯ ไป



***ข้อควรระวังคำสั่งด้านล่างอักขรพิมพ์เล็ก-ใหญ่มีความสำคัญห้ามเปลี่ยนแปลงอาจทำให้โปรแกรมไม่ทำงาน**

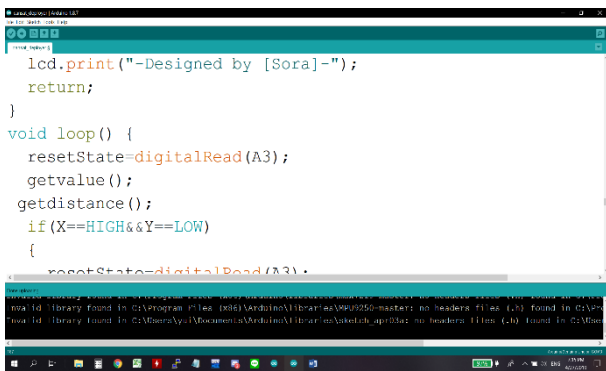
2.1 void setup(), void loop()

เป็นคำสั่งเริ่มต้นและเป็นฟังก์ชันหลัก คล้ายกับ int main() ในภาษาซี ทันทีกด new file ก็จะมีมาให้โดย void setup() ใช้ประกาศคำสั่งต่างๆที่ต้องการให้ทำงานเพียงครั้งเดียวเช่น pinMode void loop() ใช้เขียนคำสั่งทั่วไปเหมือน int main() แต่ตรงจุดนี้จะต่างกับภาษาซีตรงที่ ในภาษาซีจะมีการรันเพียงครั้งเดียวแต่ในคำสั่งนี้เมื่อทำการรันคำสั่งจากบนลงล่างแล้วจะวนกลับขึ้นมาบนใหม่แต่จะไม่ขึ้นไปถึง void setup() โดยจะอยู่ในแค่ void loop() เท่านั้น (void loop = for(x=0;x==0;x=0))



2.1 pinMode ดังที่กล่าวไว้ในบทที่ผ่านมาการจะควบคุม I/O Pin ต้องมีการบอกว่าจะใช้ pin นั้นเป็น Input หรือ Output โดยคำสั่งนี้จะประกาศใน void setup() มีรูปแบบการใช้งานคือ pinMode(Pin Number, Pin State); เช่น pinMode(2, OUTPUT); หรือ pinMode(3, INPUT); โดยสามารถนำตัวแปรมาตั้งชื่อ pin ได้ เช่น int x=2; pinMode(x, OUTPUT); แบบนี้ก็ใช้ได้เช่นกัน

```
pinMode (pwmL, OUTPUT);
pinMode (trigPin, OUTPUT);
pinMode (13, OUTPUT);
pinMode (echoPin, INPUT);
pinMode (LE, OUTPUT);
pinMode (LB, OUTPUT);
pinMode (LEDA, OUTPUT);
pinMode (LEDB, OUTPUT);
```



```

lcd.print("--Designed by [Sora]--");
return;
}
void loop() {
  resetState=digitalRead(A3);
  getValue();
  getdistance();
  if (X==HIGH&&Y==LOW)
  {
    resetState=digitalRead(A3);
  }
}

```

เท่ากับลอจิกที่ขา 2 แทนที่

2.3 **digitalWrite** คำสั่งนี้จะตรงข้ามกับคำสั่งที่แล้วตามชื่อคือคำสั่งนี้จะใช้ส่งสัญญาณหรือควบคุมอุปกรณ์ต่างๆที่ต่ออยู่กับมัน โดยรูปแบบการใช้งานมักจะใช้ในการควบคุมการติดหรือดับของอุปกรณ์ต่างๆ มีหลักไวยากรณ์ว่า **digitalWrite(Pin Number,Logic);** เช่น **digitalWrite(2,HIGH);** ก็จะทำให้มีลอจิก 1 อยู่ที่ขาสอง และสถานะนั้นจะคงอยู่ตลอดไปจนกว่าจะรีเซ็ตไมโครคอนโทรเลอร์หรือมีคำสั่ง **digitalWrite(2,LOW);** มาทำให้กลับไปเป็นลอจิก 0

2.4 **analogRead** คำสั่งนี้จะเหมือนกับข้อที่ 2.2 ต่างกันตรงที่จะสามารถใช้ได้เฉพาะกับ Analog Input ที่มีระบบ ADC เท่านั้นหลักการใช้เหมือน **digitalRead** ทุกประการแต่จะเก็บค่าเป็นตัวเลขแสดงรับแรงดันแบบ Analog 0-1024 แทน (0.00488V/Step)

2.5 **analogWrite** คำสั่งนี้มีชื่อ Analog แต่ใช้ได้กับ Digital pin ที่มี PWM support เท่านั้น(อ่านเพิ่มเติมได้ในหัวข้อ PWM pin) รูปแบบการใช้งานคือ **analogWrite(Pin Number,voltage level);** เช่น **analogWrite(11,145);** ที่ขา 11 ก็จะมีระดับแรงดัน $145 \times 0.00488 = 1.938 \text{ V}$ โดยมักนิยมนำไปใช้กับการควบคุมอัตราการรับพลังงานของอุปกรณ์เช่นควบคุมความสว่างของ LED หรือควบคุมความเร็วของมอเตอร์ เป็นต้น

2.6 **Serial** คำสั่งนี้พิเศษตรงที่ไม่ต้องใช้อุปกรณ์ต่อพ่วงใดๆ แต่เป็นคำสั่งสำหรับให้คอมพิวเตอร์สื่อสารกับ Arduino โดยมีแยกย่อยได้หลายคำสั่งแต่จะยกมาเพียงแค่ที่ใช้บ่อย เช่น **Serial.begin(); Serial.println(); Serial.print(); Serial.read();** เป็นต้น

Serial.begin() > จะประกาศใน ถึง **void setup()** จะเขียน Baud rate คือความเร็วที่ใช้ส่งสัญญาณ ซึ่งต้องตั้งค่าให้ตรงกับในหน้าจอ Serial Monitor ด้วย เช่น **Serial.begin(9600);**

No line ending 9600 baud Clear output

Serial.println() และ **Serial.print()** > ใช้เหมือนใน printf ในภาษาซีคือเป็นคำสั่งไว้แสดงผลบนหน้าจอ โดยคำสั่งทั้งสองใช้งานเหมือนกันทุกประการที่เดียวคือ **Serial.println()** เมื่อพิมพ์สิ่งที่ต้องพิมพ์เสร็จจะเว้นบรรทัดแต่ **Serial.print()** จะไม่สามารถใช้ได้สองวิธีคือแสดงข้อความ และ แสดงค่าตัวแปร เช่น

`Serial.println("Sora System");` ก็แสดงคำว่า Sora System โดยเว้นบรรทัดมาให้ใน Serial Monitor หรือ `Serial.print(a);` กรณีนี้จะแสดงค่าที่อยู่ในตัวแปร a โดยไม่เว้นบรรทัดเป็นต้น

`Serial.read()` ใช้อ่านค่าที่พิมพ์มาในช่องพิมพ์ของ Serial Monitor มาเก็บไว้ในตัวแปร โดยมีรูปแบบการใช้งานคือการจั่งตัวแปรมาเก็บค่าเช่น `data = Serial.read();` ซึ่งคำสั่งนี้จะทำให้ตัวแปร data เก็บสิ่งที่ถูกพิมพ์เข้ามาใน Serial Monitor (แล้วแต่ประเภทของตัวแปร)

```
void loop() {
  if(Serial.available()) {
    password=Serial.read();

    if(password == 'G')
    {
      LoRa.beginPacket();
      LoRa.print("G");
      LoRa.endPacket();
      Serial.println("LBD A Activated");
    }

    if(password == 'P')
    {
      LoRa.beginPacket();
      LoRa.print("P");
    }
  }
}
```



ตัวอย่างเซนเซอร์และโมดูลที่ใช้กับ Arduino

โลกนี้มีเซนเซอร์และโมดูลมากมายที่สามารถใช้ได้กับ Arduino เพราะ Arduino นั้นสามารถใช้งานร่วมกับอุปกรณ์หลากหลายโปรโตคอลอีกทั้งเหมาะกับผู้เริ่มต้นจึงทำให้มีการออกแบบเซนเซอร์ใหม่ๆ มากมาย ทั้งวัดค่าสภาพแวดล้อม ตรวจจับสิ่งต่างๆ ไปจนถึง GPS จึงทำให้ในเวลานี้ Arduino เหมาะกับการนำมาทำเป็นอุปกรณ์ประยุกต์ได้อย่างหลากหลายและเหมาะสม ในบทนี้เราจะกล่าวถึงเซนเซอร์ที่นิยมใช้ในเวลานี้ เป็นตัวอย่าง 3 ชนิด

1. BME280

BME280 เป็นเซนเซอร์ที่สามารถวัดค่าได้มากถึง 3 ค่าในตัวเดียวคือ ความชื้นสัมพัทธ์ อุณหภูมิ ความกดอากาศ BME280 จริงๆแล้วเซนเซอร์มีขนาดเล็กแต่บอร์ดจะขยายขนาดและขาเชื่อมต่อให้ง่ายต่อการใช้งาน โดยตัวเซนเซอร์สามารถวัดค่าต่างๆได้ตามนี้

- วัดอุณหภูมิในช่วง -40 ถึง 85°C
- วัดความชื้นในช่วง 0 ถึง 100%
- วัดความดันบรรยากาศในช่วง 300 hPa ถึง 1100 hPa
- รองรับแรงดันไฟที่ 2.6 ถึง 3.6 V (ใช้ 3.3V จาก Arduino หากใช้ 5 V เซนเซอร์อาจเสียได้)
- กระแส Standby อยู่ที่ $0.2\text{ }\mu\text{A}$
- กระแส Sleep Mode อยู่ที่ $0.1\text{ }\mu\text{A}$
- รูปแบบการเชื่อมต่อ : I^2C , SPI



Main features



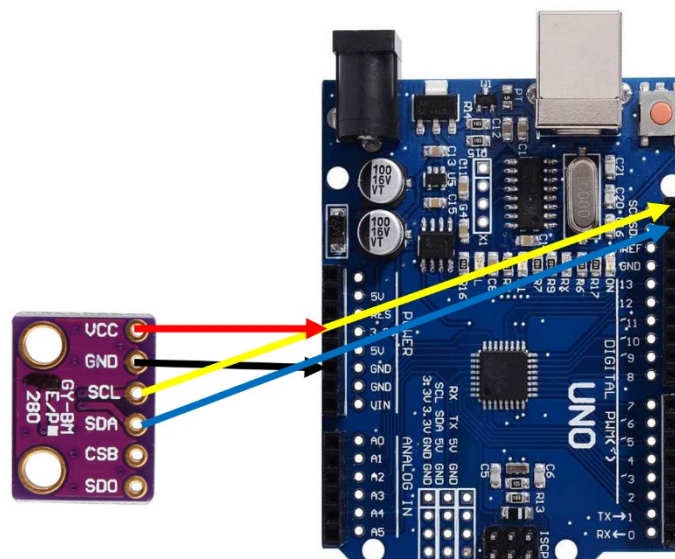
Relative Humidity
Measures relative humidity with a fast response time



Pressure
Measures barometric pressure and altitude



Temperature
Measures ambient temperature



โดยมีรูปแบบการใช้งานตามรูป (Example)

```
#include <Wire.h>
#include <SPI.h>
#include <Adafruit_Sensor.h>
#include <Adafruit_BME280.h>
#define SEALEVELPRESSURE_HPA (1013.25)
```

ในส่วนนี้จะเป็นการ include library ที่จำเป็นมาใช้รวมถึงการกำหนดค่า เพื่อเป็นตัวเทียบให้โปรแกรมด้วย

```
Adafruit_BME280 bme;
```

บรรทัดนี้จะเป็นการบอกว่า ให้ชื่อชุดคำสั่งใน Library ที่ว่า Adafruit_BME280 ย่อลงเหลือแค่ bme เพื่อให้ง่ายต่อการเขียน

```
void setup() {
  Serial.begin(9600);
  bool status;
  status = bme.begin();
  if (!status) {
    Serial.println("Error check wiring!");
    while (1);
  }
  Serial.println("System Start");
}
```

ในส่วนนี้จะเป็นการ setup ค่าต่าง ๆ และทำการตรวจสอบว่า เซนเซอร์ทำงานปกติ และ ระบบ เชื่อมต่อกับ Serial Monitor เรียบร้อยดีหรือไม่

```
void loop() {
  printValues();
  delay(1000);
}
```

ในส่วนนี้จะเป็นการย่อคำสั่งให้ง่ายต่อการเข้าใจ โดยการเรียกฟังก์ชัน printValues(); รอ 1 วินาที แล้ว ทำซ้ำไปเรื่อย ๆ

```

void printValues() {
  Serial.print("Temperature = ");
  Serial.print(bme.readTemperature());
  Serial.println(" *C");

  Serial.print("Pressure = ");

  Serial.print(bme.readPressure() / 100.0F);
  Serial.println(" hPa");

  Serial.print("Humidity = ");
  Serial.print(bme.readHumidity());
  Serial.println(" %");

  Serial.println();
}

```

ฟังก์ชันนี้จะเป็นหัวใจ
สำหรับการทำงานทั้งหมด โดย
`bme.readTemperature()` เป็น
คำสั่งสำหรับอ่านค่าอุณหภูมิจาก
เซนเซอร์ อาจใช้ในรูปแบบ `int u`
`= bme.readTemperature()` ก็ได้
เพื่อให้ค่า `u` มีค่าเป็นอุณหภูมิ

`(bme.readPressure() /`
`100.0F)` ฟังก์ชันนี้ใช้รับค่าความ
กดอากาศและแปลงหน่วยความ
กดอากาศหลักการใช้งานในการ
เก็บเข้าตัวแปรก็เหมือนกับคำสั่ง
อื่น

`bme.readHumidity()`
คำสั่งนี้ใช้อ่านความชื้นสัมพัทธ์

ส่วนคำสั่งอื่นๆช่วยในการ
แสดงผลให้มีรูปแบบที่สวยงาม
และอ่านได้ง่าย

2. ATGM336H (GPS)

โมดูลนี้มีหน้าที่เชื่อมสัญญาณกับดาวเทียมที่อยู่บนวงโคจรเพื่อรับค่าพิกัด GPS (Latitude Longitude) มาใช้งานค่า Parameter ของเซนเซอร์มีดังต่อไปนี้

λ Good positioning and navigation function, support BDS/GPS/GLONASS satellite navigation single system positioning and any combination of multiple system joint positioning, also support QZSS and SBAS system.

λ Support A-GNSS

λ Cold start - Recapture sensitivity : -148dBm

λ Tracking sensitivity : -162dBm

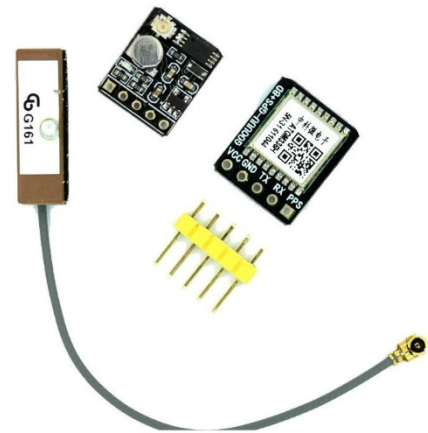
λ Positioning Precision : 2.5m (CEP50)

λ The Time to First Fix: : 32s

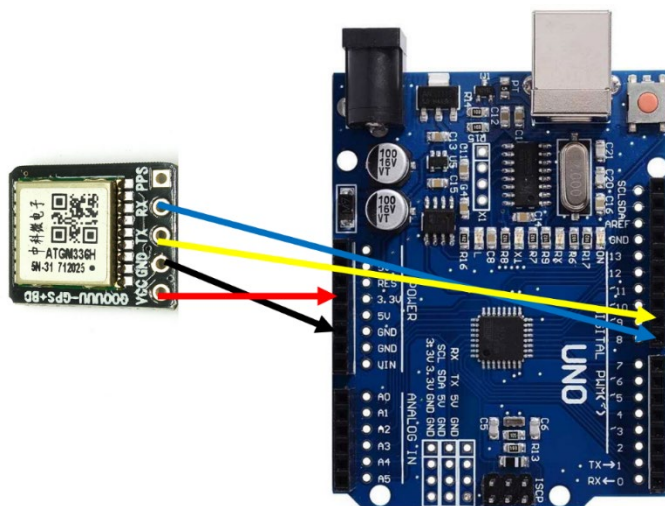
λ ใช้กับไฟ 3.3 VDC เท่านั้น

λ ใช้กระแสสูงสุด 100 mA

λ ใช้โปรโตคอล UART (Tx>Rx,Rx>Tx)



โดยการต่อใช้งานให้ต่องดังรูป



***Tx ของ GPS ต่อกับ Rx ของ Arduino Rx ของ GPS ต่อกับ Tx ของ Arduino โดยจะกำหนดในโค้ดอีกครั้ง

ตัวอย่างโค้ด

```
#include "TinyGPS++.h"
#include "SoftwareSerial.h"
```



ในส่วนนี้จะเป็นการ include library ที่จำเป็นมาใช้

```
SoftwareSerial serial_connection(9,8);
TinyGPSPlus gps;
```



สองบรรทัดนี้คือการกำหนดค่าว่าจะใช้ขาไหนเป็น Tx และ Rx เพื่อใช้สื่อสารซึ่งกำหนดให้ 9=Rx (ต่อกับ Tx ของGPS) และ 8=Tx (ต่อกับ Rx ของGPS) และในบรรทัดที่2เป็นการย่อชื่อฟังก์ชัน

```
void setup()
{
  pinMode(8,OUTPUT);
  pinMode(9,INPUT);
  Serial.begin(9600);
  serial_connection.begin(9600);
  Serial.println("GPS Start");
}
```



ในส่วนนี้จะเป็นการ setup ค่าต่างๆ เริ่มการทำงานของ Serial Monitor และเริ่มการสื่อสารกับโมดูล GPS

```
void loop()
{
  while(serial_connection.available())
  {
    gps.encode(serial_connection.read());
  }

  if(gps.location.isUpdated())
  {
    Serial.println("Satellite Count:");
    Serial.println(gps.satellites.value());
    Serial.println("Latitude:");
    Serial.println(gps.location.lat(), 6);
    Serial.println("Longitude:");
    Serial.println(gps.location.lng(), 6);
  }
}
```

```
Serial.println ("Speed MPH:");
Serial.println (gps.speed.mph());
Serial.println ("Altitude Feet:");
Serial.println (gps.altitude.feet());
Serial.println ("");
}
}
```

ตั้งแต่ `void loop()` ลงมาจะเป็นการรอให้โมดูล GPS สื่อสารกับดาวเทียมและพร้อมทำงานจากนั้นจะทำการส่งค่ามาทาง Serial Monitor ตามคำสั่งต่างๆ ซึ่งวิธีการเก็บค่าที่ได้ไว้ในตัวแปรจะเหมือนกับบทก่อนๆ จึงของละไว้ โดยคำสั่งมีดังต่อไปนี้

`gps.satellites.value()` > ใช้นับจำนวนดาวเทียมที่โมดูลกำลังสื่อสารอยู่ (ยิ่งมีเยอะยิ่งแม่นยำ) จะอยู่ที่ประมาณ 7-15 ดวง

`gps.location.lat()` > ใช้รับค่าละติจูดจากโมดูล

`gps.location.lng()` > ใช้รับค่าลองจิจูดจากโมดูล

`gps.speed.mph()` > ในคำสั่งนี้จะเป็นการนำความเปลี่ยนแปลงของพิกัดมาเทียบเป็นระยะทางแล้วเปรียบเทียบกับหน่วยเวลาแล้วคำนวณออกมาเป็นความเร็วในหน่วย ไมล์ต่อชั่วโมง

`gps.altitude.feet()` > คำสั่งนี้จะคำนวณหาพิกัดแล้วนำไปเทียบในแผนที่เพื่อแสดงระดับความสูงจากระดับน้ำทะเล

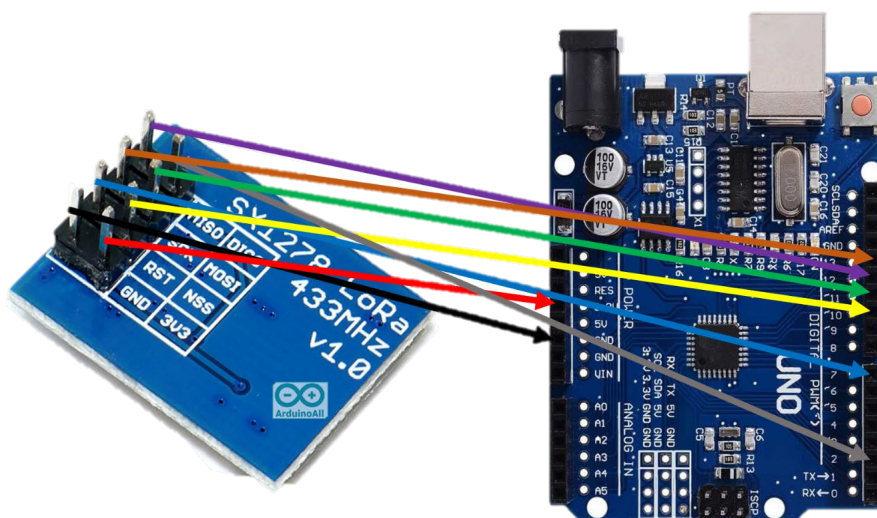
3.LoRa

โมดูลนี้เป็นโมดูลสื่อสารไร้สายด้วยคลื่นวิทยุที่มีหลายรุ่นย่อยแต่ที่จะใช้ในหัวข้อนี้คือรุ่น Ra-02 โมดูลรุ่นนี้ทำงานที่ความถี่ 410-525 MHz สามารถทำการสื่อสารในที่โล่งได้มากกว่า 1 กิโลเมตร และประมาณ 50 เมตรในเมืองโดยมีรายละเอียดจำเพาะต่อไปนี้

- ~ ใช้โปรโตคอล SPI
- ~ ย่านความถี่ที่ใช้ได้ 410-525MHz
- ~ การใช้กระแส Peak > 120 mA Standby > 1.5 mA Receive > 13 mA
- ~ กำลังส่งสูงสุด 18 dBm
- ~ ใช้แรงดัน 3.3 V เท่านั้นผิดจากนี้อาจเสียหายได้

การต่อใช้งานทั้งภาคส่งและภาครับเหมือนกันตามภาพและตารางด้านล่าง

LoRa-Ra 02	Arduino UNO
3V3	3.3 V
GND	GND
NSS	D10
RST	D7
MISO	D12
MOSI	D11
SCK	D13
DIO 0	D2



ตัวอย่างโค้ดภาคส่ง

```
#include <SPI.h>
#include <LoRa.h>
int counter = 0;
```

ในส่วนนี้จะเป็นการ include library ที่จำเป็นมาใช้ และ กำหนดค่าตัวแปรสำหรับตัวเลขที่จะใช้ส่งเป็นตัวอย่าง

```
void setup() {
  Serial.begin(9600);
  while (!Serial);
  LoRa.setPins(10,7,2);
  Serial.println("LoRa Sender");
  if (!LoRa.begin(433E6)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
  LoRa.setTxPower(17);
}
```

ในส่วนนี้จะเป็นการ setup ค่าต่าง ๆ เริ่มการทำงานของ Serial Monitor และเริ่มการสื่อสารกับ โมดูล LoRa พร้อมกับตรวจสอบสภาพการทำงานว่าพร้อมหรือไม่ ในคำสั่ง **LoRa.setPins()**; จะใช้กำหนดขาของ LoRa, ท ที่ ไม่ ถูก lock ไว้ ทั้ง 3 ขา คือ **LoRa.setPins(NSS,RST,DIO0)**; **LoRa begin(433E6)** คือกำหนดความถี่เริ่มต้นที่ 433 MHz เปลี่ยนความถี่ก็จะเป็น **LoRa begin(488E6)** ก็จะทำงานที่ 488MHz ซึ่งในส่วนนี้จะต้องกำหนดตัวรับและตัวส่งให้เป็นค่าเดียวกัน **LoRa setTxPower(17)**; คำสั่งนี้ใช้กำหนดกำลังส่งสัญญาณของ LoRa (เลขยิ่งเยอะยิ่งใช้พลังงานสูง) และ สูงสุดที่ 17

```
void loop() {
  Serial.print("Sending packet: ");
  Serial.println(counter);
  LoRa.beginPacket();
  LoRa.print("hello world");
  LoRa.print(counter);
  LoRa.endPacket();
  counter++;
  delay(1000);
}
```

สำหรับ loop นี้ จะทำการเริ่มต้นการส่งสัญญาณ ด้วยคำสั่ง **LoRa beginPacket()**; ส่งคำว่า hello shiro ตามด้วยเลขที่บวกไปที่ละ 1 ด้วยคำสั่ง **LoRa.print("hello world");** และ **LoRa.print(counter)**; ยุติ การ ส่ง สัญญาณด้วยคำสั่ง **LoRa endPacket()**; เพิ่ม เลขในตัวแปร counter ที่ละ 1 และรอ 1 วินาทีเพื่อทำงานครั้งถัดไป

ตัวอย่างโค้ดภาครับ

```
#include <SPI.h>
#include <LoRa.h>
```

ในส่วนนี้จะเป็นการ include library ที่จำเป็นมาใช้ และกำหนดค่าตัวแปรสำหรับ ตัวเลขที่จะใช้ส่งเป็นตัวอย่าง

```
void setup() {
  Serial.begin(9600);
  while (!Serial);
  LoRa.setPins(10,7,2);
  Serial.println("LoRa Sender");
  if (!LoRa.begin(433E6)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
}
```

ในส่วนนี้จะเป็นการ setup ค่าต่างๆ เริ่มการทำงานของ Serial Monitor และเริ่มการสื่อสารกับ โมดูล LoRa พร้อมกับตรวจสอบ สภาพการทำงานว่าพร้อมหรือไม่ ในคำสั่งที่เหลืองก็จะเหมือนกับ ภา ค ส ง แ ต่ จ ะ ไ ม่ มี **LoRa setTxPower(17);** เพราะไม่ ต้องส่งสัญญาณนั่นเอง

```
void loop () {
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    Serial.print("Received packet ");
    while (LoRa.available()) {
      Serial.print ((char)LoRa.read());
    }
    Serial.print (" with RSSI ");
    Serial.println (LoRa.packetRssi());
  }
}
```

ในส่วนนี้จะเป็นการ อ่านค่าจาก lora ที่ส่ง มาในอากาศ โดยแปลงให้ เป็น ข้อมูล ประเภท ตัวอักษรด้วย **char** กับ **LoRa read()** สามารถใช้ ในรูปแบบเช่น **char X = LoRa read()** ก็จะทำให้ตัว แปร X มีเก็บชุดอักขระที่

II. Wireless Communication

by Dr. Peeramed Chodkaveekityada

นอกจากหน่วยประมวลผล เช่น เซอร์ และ วงจรไฟฟ้า ที่ทำงานรับข้อมูลแล้ว เราจะไม่สามารถรู้ได้เลยว่าข้อมูลที่เก็บมาได้นั้น มีค่าเป็นอย่างไรเมื่อส่งขึ้นไปพร้อมปล่อยแล้ว (2 เมตรคงไม่มีปัญหา แต่ถ้า 300 เมตรล่ะ? หรือเก็บบันทึกข้อมูลก็ได้ แล้วถ้าไม่อยากเก็บกู่ล่ะ?) ดังนั้นการสื่อสารข้อมูลจาก “ภาคส่ง (Sender)” ไปสู่ “ภาครับ (Receiver)” นั้นจำเป็นอย่างมาก และต้องสื่อสารแบบไร้สาย (Wireless Communication) อีกด้วย โดยในบทนี้จะพูดถึง การรับส่งสัญญาณทางเดียว การรับส่งสัญญาณ 2 ทาง คลื่นความถี่ของการรับส่งสัญญาณ เสาอากาศ ฯลฯ

การรับส่งสัญญาณแบบไร้สาย (Wireless Communication)

คือ การส่งข้อมูลระหว่างภาครับและภาคส่งโดยไม่ใช้สาย แต่ใช้การส่งสัญญาณผ่านการแผ่ของคลื่นแม่เหล็กไฟฟ้า (Electromagnetic Wave Radiation) (ต่างจาก Wired Communication) ทำให้การส่งข้อมูลในระยะที่ไกลมาก ๆ ได้ในราคาถูก (เช่น ส่งข้อมูลจาก CanSat ลงมาที่ภาครับ)

ประเภทของ Wireless Communication สามารถแบ่งตามผู้ส่งข้อมูลได้เป็น Simplex (ส่งข้อมูลได้ทิศทางเดียว), Half Duplex (ส่งข้อมูลได้ 2 ทาง / ส่งทีละฝั่ง A แล้ว B), Full Duplex (ส่งข้อมูลได้ 2 ทาง แต่ส่งได้พร้อมกัน / A พร้อมกับ B)

การส่งสัญญาณแบบไร้สายนั้นมีหนึ่งในองค์ประกอบหลักคือ เสาสัญญาณ (Antenna) โดยเสาสัญญาณนั้นมีหลากหลายรูปแบบ ขึ้นอยู่กับการใช้งานแต่ละประเภท และมีกำลังรับ-ส่ง (Gain) ที่ต่างกัน

คลื่นความถี่ของสัญญาณ (Signal Frequency)

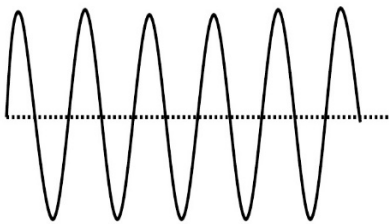
คลื่น (Wave) คือ การถ่ายทอดพลังงานด้วยการสั่น เกิดการสั่นของแหล่งและแผ่ไปยังจุดใด ๆ โดยในการประยุกต์ใช้ส่งข้อมูลแบบไร้สายนั้น ข้อมูลในการส่งจะถูกแปลงให้อยู่ในรูปของ “คลื่น” (Modulation) โดยภาคส่ง, จากนั้นภาครับจึงรับสัญญาณที่ส่งมา แล้วแปลงจากรูปสัญญาณเป็นข้อมูลที่คอมพิวเตอร์หรือระบบดิจิทัลสามารถอ่านได้ (Demodulation) โดยที่คลื่นจะประกอบไปด้วย 3 อย่าง คือ

1. ความถี่ (frequency, f) **** --> จำนวนรอบการสั่นต่อ 1 วินาที มีหน่วยเป็น Hz /
2. ความยาวคลื่น (wavelength, λ)
3. ความเร็ว (velocity, v) **** --> สำหรับคลื่นแม่เหล็กไฟฟ้ามี ความเร็ว = ความเร็วแสง

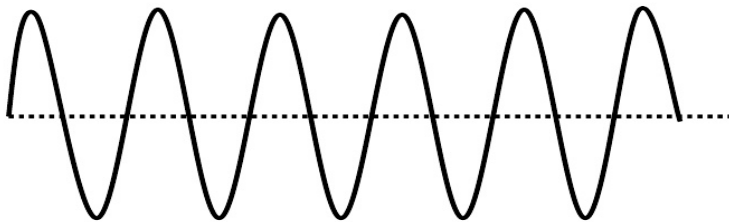
ความสัมพันธ์ของทั้ง 3 องค์ประกอบ สามารถอธิบายได้ด้วยสมการคลื่น:

$$v = f\lambda$$

หรือสามารถอธิบายเชิงการใช้งานได้ว่า “คลื่นมีความเร็วคงที่ คือ ความเร็วแสง แต่จะต่างกันที่ความถี่ของคลื่น ซึ่งยิ่งความถี่สูง จะยิ่งใช้พลังงานมากในการส่ง ทำให้ส่งได้ในระยะทางที่สั้นกว่าคลื่นความถี่ต่ำกว่า เมื่อกำลังส่งของแหล่งเท่ากัน”



คลื่นความถี่สูง เคลื่อนที่ได้ระยะทางสั้น
แต่สามารถส่งข้อมูลจำนวนมาก ๆ ได้เร็ว



คลื่นความถี่ต่ำ เคลื่อนที่ได้ระยะทางไกล
แต่ส่งข้อมูลได้ทีละน้อย ๆ

ประเภทของเสาสัญญาณที่ควรรู้ (Types of Antennas)

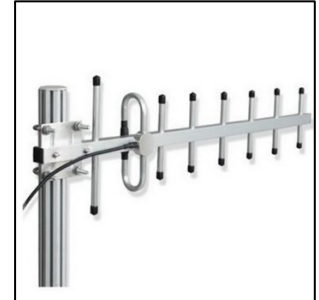
1. Monopole Antenna (เสาเดี่ยว)

มีลักษณะเป็นเสาแท่งเดี่ยว รับ-ส่งสัญญาณจากทั่วทุกทิศทาง (Omnidirectional) แต่เสาประเภทนี้จะมี Gain ที่ค่อนข้างต่ำ เสาชนิดนี้นิยมใช้กับงานทั่วไป ไม่เฉพาะเจาะจง



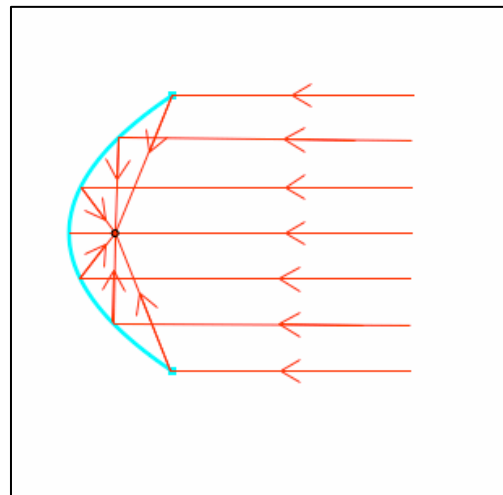
2. Yagi-Uda Antenna (เสาข้างปลา)

มีลักษณะเป็นแท่งยาว (ความยาวจะขึ้นอยู่กับคลื่นความถี่สัญญาณที่ใช้และ Gain ที่ต้องการ) มี element ย่อยวางห่างกันเป็นระยะ ๆ โดยเสาชนิดนี้มีความพิเศษ คือ “มี Gain ที่สูงมาก เฉพาะในทิศทางที่เสาอากาศหันออกเป็นมุมแคบ ๆ (Monodirectional)” เหมาะสมกับการใช้งานส่งสัญญาณในระยะทางมาก



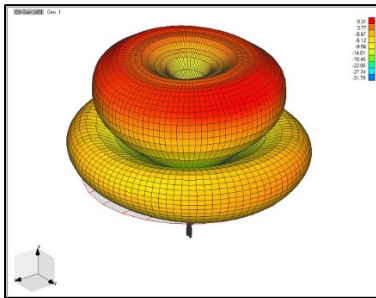
3. Satellite Dish (จานดาวเทียม)

มีลักษณะเป็นจานโค้งรูปพาราโบลา (Concave Parabolic Surface) รับสัญญาณจากทิศทางเดียว เป็นแนวกว้าง ใช้รับสัญญาณที่ส่งมาคงที่ และเหมาะสมกับสัญญาณคลื่นความถี่สูง ๆ ซึ่งจะได้ขนาดของจานที่กำลังพอดี และมี Gain สูง

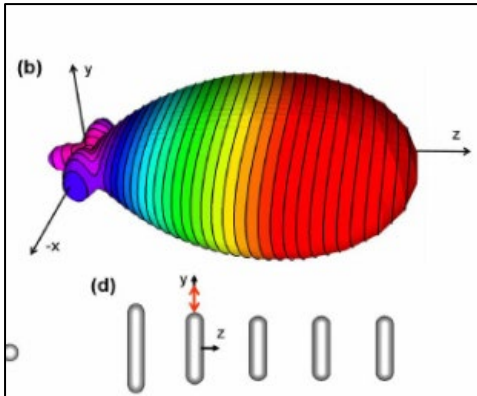


Radiation Pattern (เพิ่มเติม)

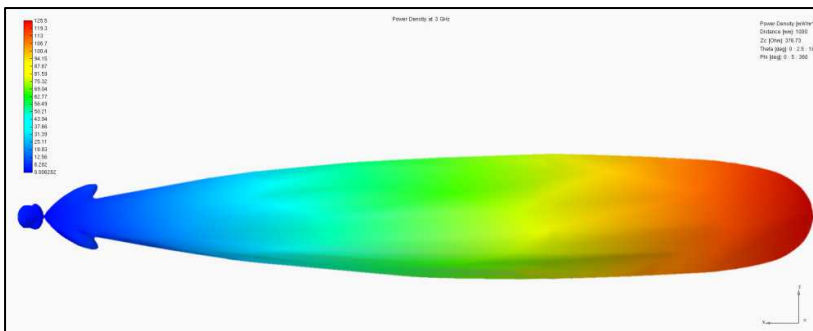
Radiation Pattern (รูปแบบการแพร่กระจายคลื่น) คือ ลักษณะการแพร่คลื่นแม่เหล็กไฟฟ้าจากแหล่งออกไปในอากาศ เพื่อแสดง Gain ของเสาอากาศนั้นในแต่ละทิศ เพื่อหาทิศการหันของเสาอากาศว่าควรหันไปในทิศทางใดเพื่อให้ได้ประสิทธิภาพการสื่อสารข้อมูลสูงสุด (Gain สูงสุด และ Loss* ต่ำ)



Monopole Antenna
Radiation Pattern



Yagi Uda Antenna
Radiation Pattern



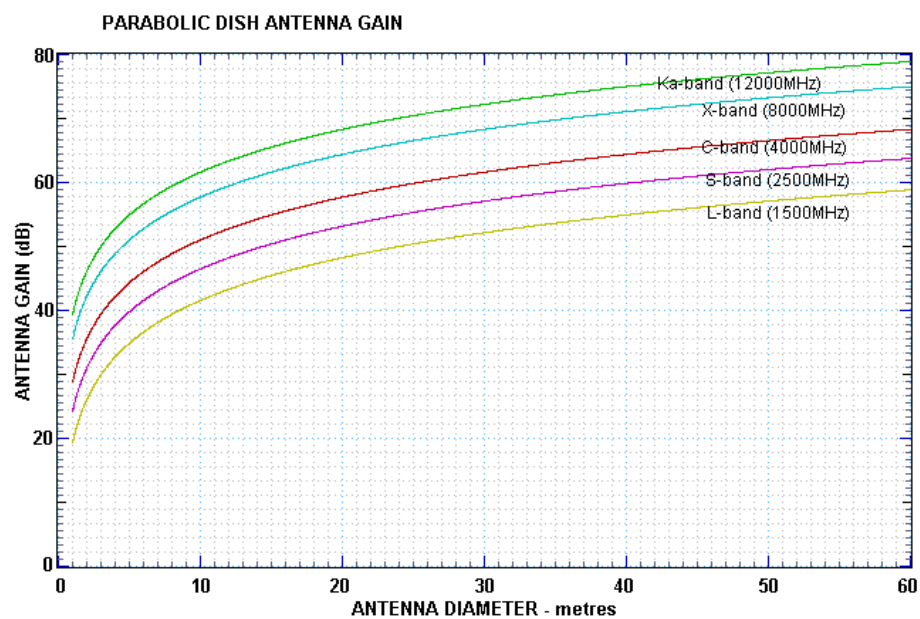
Satellite Dish
Radiation Pattern

*Loss คือ การสูญเสียหรือสูญหายของข้อมูลระหว่างการส่งด้วยปัจจัยต่าง ๆ เช่น ความไม่เสถียรของตัวส่ง การรบกวนกันของคลื่น การหันเสาอากาศไปในทิศที่ Gain ต่ำ ฯลฯ

NOTE

NOTE

NOTE



III. Parachute Physics & Design

by Thanakrit Suelong

หลักการทำงานของ CanSat ส่วนสำคัญ คือ “จะอย่างไรให้ CanSat ชะลอตัวระหว่างตกลงมา” เพื่อลดความเร็วตกให้ไม่เกิดความเสียหายขึ้นเมื่อตกลงถึงพื้น วิธีการนั้นมีอยู่หลากหลายประเภท หนึ่งในนั้นคือ “ร่มชูชีพ (Parachute)”

ร่มชูชีพ เป็นส่วนสำคัญของ CanSat ซึ่งเป็นส่วนในการชะลอความเร็วของตัว CanSat ขณะที่ตกลงสู่พื้นดิน ร่มชูชีพจะต้องทำการทดสอบเรื่อย ๆ เพื่อให้แน่ใจว่ามันสามารถพาตัว CanSat ให้ถึงพื้นแบบปลอดภัยได้ ถ้าจะพูดว่าส่วนนี้ถือว่าเป็นส่วนที่เล็กน้อยมากเมื่อเทียบกับส่วนอิเล็กทรอนิกส์ก็ได้ แต่อย่าลืมว่า ร่มชูชีพนี้แหละ จะเป็นส่วนที่จะพบกับปัญหาต่าง ๆ ค่อนข้างมากเลยทีเดียว ดังนั้นการสร้างร่มชูชีพที่ดี จะเป็นส่วนสำคัญมากในการทำให้ CanSat ตกอย่างปลอดภัย สามารถเก็บค่าที่ต้องการได้ และไม่ทำให้ตัว CanSat พังอีกด้วย

1. การตกอย่างอิสระ

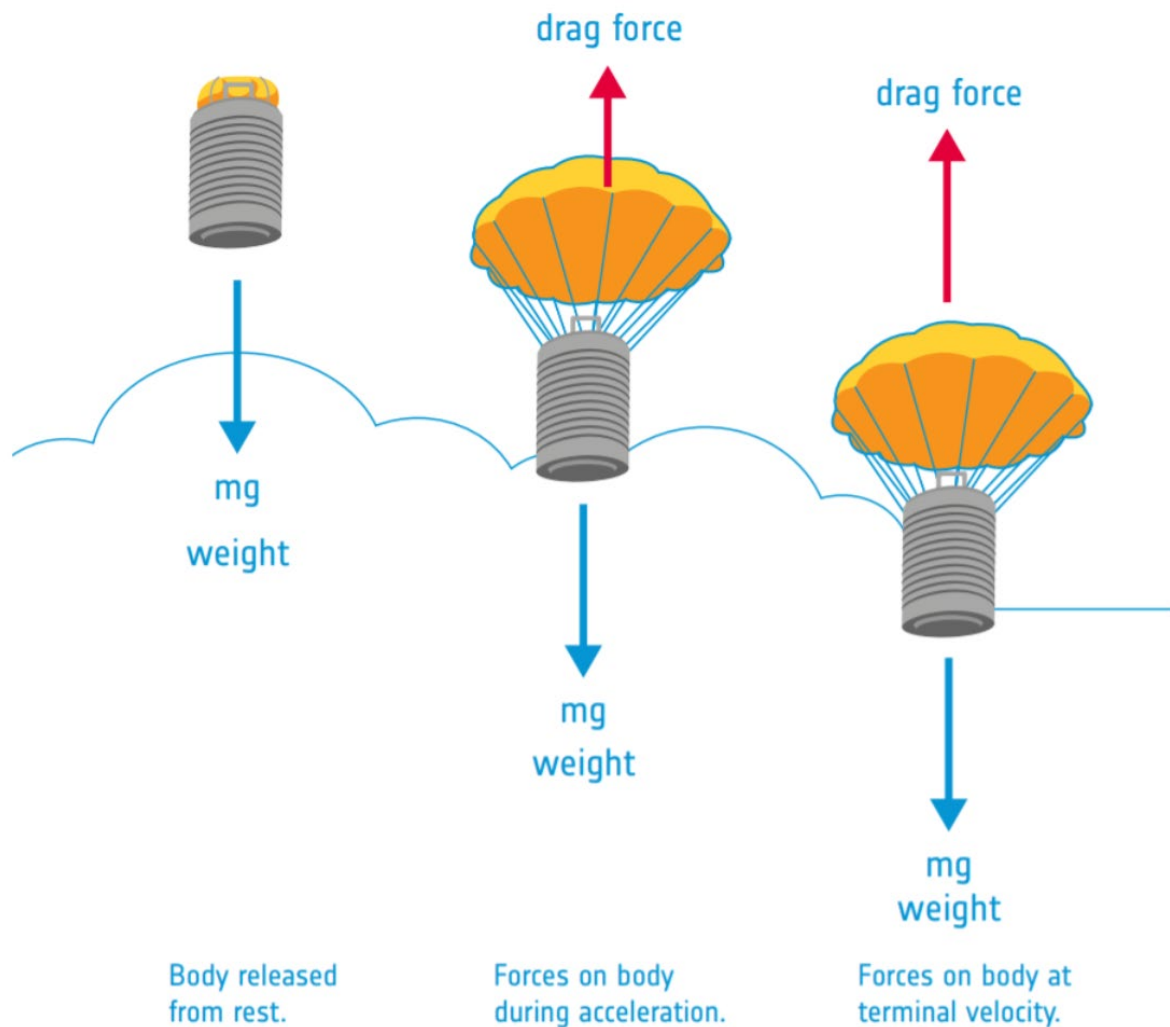
เมื่อเราโยนก้อนหินขึ้นไป ก้อนหินจะลอยตัวขึ้นไปเรื่อย ๆ จนถึง ณ จุด ๆ หนึ่ง มันก็จะเริ่มตกลงมาอย่างอิสระด้วยความเร่งซึ่งเกิดจาก “แรงโน้มถ่วง” (Gravity, g) ของโลก ซึ่งมีค่าประมาณ 9.81 m/s^2 (ส่วนมากจะประมาณเป็น 10 m/s^2) ทำให้ก้อนหินตกเร็วขึ้นเรื่อย ๆ

สมมติ ถ้าก้อนหินมีมวล 2 กิโลกรัม (kg) วัตถุก้อนนี้จะมีน้ำหนัก (Weight) เท่าไร

ตอบ _____ หน่วย _____

ดังนั้น น้ำหนัก = มวล \times ความเร่งเนื่องจากแรงโน้มถ่วงของโลก ($\text{Weight} = \text{Mass} \times g$)

2. ร่มชูชีพ (Parachute)



การออกแบบร่มนั้นจะต้องอาศัยการคำนวณทางฟิสิกส์เพื่อให้ได้ “ขนาด” และ “ชนิด” ของร่มที่เหมาะสมกับภารกิจ

ระหว่างตก แรงที่ทำให้ CanSat ตกลงมา “ขณะปล่อย” มีเพียงแค่ “น้ำหนัก” หรือแรงดึงดูดจากโลก ใช้ตัวแปรแทนว่า $F_{gravity}$ ซึ่งมีค่า

$$F_{gravity} = mg$$

เมื่อ m คือ มวลของ CanSat มีหน่วยเป็นกิโลกรัม (kg)

g คือ ความเร่งเนื่องจากแรงโน้มถ่วงของโลก มีค่าประมาณ 10 m/s^2

และเมื่อ CanSat ตกมาหลังจากปล่อยแล้ว จะมีอีกแรงหนึ่ง ทำให้ร่มชูชีพคลายตัวและกางออก ซึ่งคือ “แรงต้านอากาศ” (Drag Force / Air Resistance Force, ใช้ตัวแปรแทนว่า F_d ซึ่งเป็นไปตามสมการ

$$F_d = \frac{1}{2} \rho c_d A v^2$$

เมื่อ c_d คือ ค่าสัมประสิทธิ์ความต้านอากาศ ขึ้นอยู่กับรูปร่างของร่มชูชีพ มีค่าประมาณ 0.825 N/m^2
 ρ คือ ความหนาแน่นของอากาศ มีค่าประมาณ 1.225 kg/m^3
 A คือ พื้นที่ของร่มชูชีพ มีหน่วยเป็น m^2
 $A = \pi(R^2 - r^2)$ สำหรับร่มวงกลมรัศมี R และมีรูขนาดรัศมี r
 v คือ ความเร็วที่ CanSat ตก มีหน่วยเป็น (m/s)

แต่เมื่อผ่านไปสักพักหลังจากปล่อย แรงต้านอากาศที่มีทิศขึ้น จะค่อย ๆ มีค่ามากขึ้นเรื่อย ๆ จนแรงต้านอากาศมีค่าเท่ากับ น้ำหนักของวัตถุ ดังนั้น CanSat จะตกด้วยความเร็วคงที่ จากสมการ

$$F_{gravity} = F_d$$

$$mg = \frac{1}{2} \rho c_d A v^2$$

$$mg = \frac{1}{2} \rho c_d \pi (R^2 - r^2) v^2$$

$$R^2 - r^2 = \frac{2mg}{\rho c_d \pi v^2}$$

โดยขณะที่ร่มกางแล้วและกำลังตกอยู่ มักใช้เวลานานกว่าขณะกางออก เราจึงสามารถประมาณเวลาที่ใช้ตกด้วยความเร็วคงที่ได้จากสูตร

$$v = \frac{s}{t}$$

แต่ยังต้องพิจารณาถึงลมที่พัดให้ร่มยกตัวอีกด้วย ซึ่งขึ้นอยู่กับความเร็วลม ณ ขณะนั้น

ที่มา สามารถหาข้อมูลเพิ่มเติมได้จาก Internet, ค้นหาว่า “Drag Equation”

แบบร่าง Cansat + Parachute

หารัศมีของ Parachute

NOTE

NOTE

สำหรับสายโหดสายเลื่อน

$$\begin{aligned} \text{จาก } m \frac{dv}{dt} &= mg - \beta v^2 ; \beta = \frac{1}{2} \rho c_d A \\ \frac{dv}{dt} &= g - \frac{\beta}{m} v^2 \\ \int_{v=0}^v \frac{1}{g - \frac{\beta}{m} v^2} dv &= \int_{t=0}^t dt \end{aligned}$$

จะได้

$$\begin{aligned} t &= \sqrt{\frac{m}{\beta g}} \operatorname{arccosh} \left\{ \exp \left(\frac{\beta h}{m} \right) \right\} \\ v &= \sqrt{\frac{mg}{\beta}} \tanh \left(\frac{t}{\sqrt{\frac{m}{\beta g}}} \right) \\ v &= \sqrt{\frac{mg}{\beta}} \tanh \left\{ \operatorname{arccosh} \left[\exp \left(\frac{\beta h}{m} \right) \right] \right\} \end{aligned}$$

IV. Surveying and Geography

by Phachara Phumiprathet

การสำรวจและการรังวัด เป็นกระบวนการที่สำคัญในการศึกษาทางภูมิศาสตร์ วิทยาศาสตร์ และวิศวกรรมศาสตร์ เพราะการสำรวจจะช่วยให้ผู้ศึกษาสามารถเข้าใจและรู้ถึงลักษณะที่สำคัญในด้านต่างๆ ของพื้นที่เป็นอย่างดี และนำไปสู่การตั้งคำถาม สมมติฐาน และแนวทางการศึกษา ในการสำรวจพื้นที่นั้นจำเป็นจะต้องอาศัยศาสตร์และเครื่องมือต่างๆ มาเกี่ยวข้องเพื่อช่วยเหลือให้ผู้ศึกษาสามารถสำรวจและศึกษาพื้นที่ได้ง่ายมากยิ่งขึ้น โดยเฉพาะอย่างยิ่ง "ภูมิศาสตร์และภูมิสารสนเทศศาสตร์" ที่จะเข้ามาช่วยให้ผู้ศึกษาสามารถสร้างข้อมูลเชิงพื้นที่ (Geo Database) ได้ ดังนั้นการสร้าง CanSat ขึ้นมาเพื่อศึกษาประเด็นปัญหาที่ผู้ศึกษาสนใจหรือตั้งคำถามจะมีประสิทธิภาพมากขึ้นหากผู้ศึกษามีความรู้และเข้าใจในพื้นที่ศึกษาเป็นอย่างดี

SATELLITE

Satellite หรือ ดาวเทียม คือ อุปกรณ์ที่มนุษย์สร้างขึ้นแล้วปล่อยไว้ในวงโคจรรอบโลก เพื่อใช้ประโยชน์ในด้านต่างๆ เช่น ถ่ายภาพ ตรวจอากาศ โทรคมนาคม และปฏิบัติการทางวิทยาศาสตร์ เป็นต้น ดาวเทียมถูกส่งขึ้นสู่อวกาศโดยติดตั้งบนจรวดหรือยานขนส่งอวกาศ ดาวเทียมดวงแรกของโลกเป็นของสหภาพโซเวียตชื่อ สปุตนิก 1 (Sputnik 1) ถูกส่งขึ้นสู่อวกาศเมื่อวันที่ 4 ตุลาคม 2500

ห้าสิบกว่าปีผ่านไปนับตั้งแต่สปุตนิก 1 ถูกส่งขึ้นสู่อวกาศจนถึงปัจจุบัน ได้มีการส่งดาวเทียมขึ้นสู่วงโคจรรอบโลกมากกว่า 30,000 ดวง เพื่อใช้ประโยชน์ในด้านต่างๆ ดาวเทียมทั้งหลายจึงมีขนาด รูปร่าง ลักษณะแตกต่างกัน อย่างไรก็ตามดาวเทียมส่วนใหญ่มีองค์ประกอบหลักที่คล้ายคลึงกันดังอธิบายประกอบดังนี้

ระบบนำร่อง เป็นระบบคอมพิวเตอร์และไจโรสโคป ซึ่งมีหน้าที่ตรวจสอบตำแหน่งของดาวเทียม โดยการเปรียบเทียบกับตำแหน่งของดาวฤกษ์ สัญญาณวิทยุจากสถานีบนโลกหรือสัญญาณจากดาวเทียมจีพีเอส

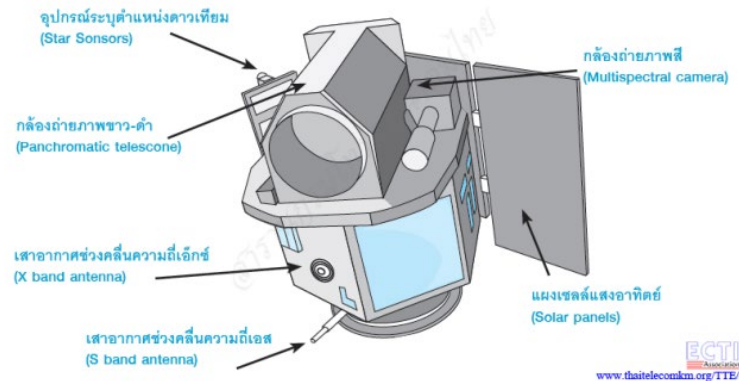
ระบบควบคุมและสื่อสาร ประกอบด้วย คอมพิวเตอร์ที่เก็บรวบรวมข้อมูล และประมวลผลคำสั่งต่างๆ ที่ได้รับจากส่วนควบคุมบนโลก โดยมีอุปกรณ์วิทยุและเสาอากาศ เพื่อใช้ในการรับส่งข้อมูล

ระบบเซ็นเซอร์ และอุปกรณ์วิทยาศาสตร์อื่นๆ ขึ้นอยู่กับวัตถุประสงค์ของภารกิจ เช่น ดาวเทียมสำรวจโลกติดตั้งเซ็นเซอร์ตรวจจับช่วงคลื่นต่างๆ ดาวเทียมปฏิบัติการทางวิทยาศาสตร์ติดตั้งห้องทดลอง ดาวเทียมทำแผนที่ติดตั้งเรดาร์และกล้องถ่ายภาพ ข้อมูลที่ได้จากระบบนี้จะถูกส่งกลับสู่โลกโดยใช้เสาอากาศส่งคลื่นวิทยุ

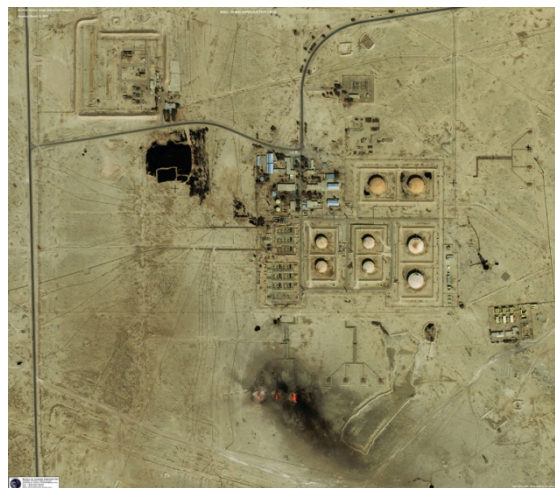
ระบบพลังงาน ทำหน้าที่ผลิตพลังงานและกักเก็บไว้เพื่อแจกจ่ายไปยังระบบไฟฟ้าของดาวเทียม โดยมีแผงรับพลังงานแสงอาทิตย์ (Solar cells) ใ้รับพลังงานจากแสงอาทิตย์เพื่อเปลี่ยนเป็นพลังงานไฟฟ้า แต่ดาวเทียมขนาดใหญ่อาจมีเครื่องปฏิกรณ์นิวเคลียร์

ระบบเครื่องยนต์ ดาวเทียมขนาดใหญ่ที่มีอายุใช้งานยาว จะมีเครื่องยนต์ซึ่งทำงานคล้ายกับเครื่องอัดอากาศ และปล่อยออกทางปลายท่อ มีหน้าที่สร้างแรงขับเคลื่อนเพื่อรักษาระดับความสูงของวงโคจร เนื่องจากที่ระดับวงโคจรในอวกาศยังคงมีโมเลกุลลอยอยู่อย่างเบาบาง แต่ดาวเทียมโคจรด้วยความเร็วสูง โมเลกุลอากาศ

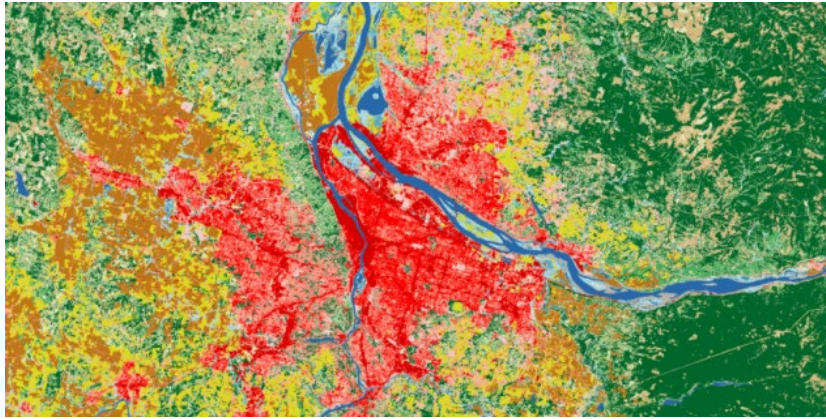
สามารถสร้างแรงเสียดทานให้ดาวเทียมเคลื่อนที่ช้าลงและเคลื่อนที่ต่ำลง หากไม่รักษาระยะสูงไว้ ในที่สุดดาวเทียมก็จะตกลงสู่พื้นโลก



ดาวเทียมทำแผนที่ เป็นดาวเทียมที่มีวงโคจรต่ำ (LEO) ที่ระดับความสูงไม่เกิน 800 กิโลเมตร เพื่อให้ได้ภาพที่มีรายละเอียดสูง และเป็นดาวเทียมที่มีวงโคจรใกล้ขั้วโลก (Polar orbit) เพื่อให้สแกนพื้นผิวถ่ายภาพได้ครอบคลุมทุกพื้นที่ของโลก ภาพถ่ายดาวเทียมที่ได้สามารถนำไปใช้ในการทำแผนที่ ผังเมือง และการทำจากรกรรมสอดแนมทางการทหาร ดาวเทียมทำแผนที่ที่มีชื่อเสียงได้แก่ Ikonos, QuickBird ซึ่งสามารถดูภาพแผนที่ใน Google Maps



ดาวเทียมสำรวจทรัพยากร เป็นดาวเทียมวงโคจรต่ำที่มีวงโคจรแบบใกล้ขั้วโลก (Near Polar Orbit) ที่ระยะสูงประมาณ 800 กิโลเมตร จึงไม่มีรายละเอียดสูงเท่าภาพถ่ายที่ได้จากดาวเทียมทำแผนที่ เพราะเน้นการครอบคลุมพื้นที่เป็นบริเวณกว้าง และทำการบันทึกภาพได้ทั้งในช่วงแสงที่ตามองเห็นและรังสีอินฟราเรดเนื่องจากโลกแผ่รังสีอินฟราเรดออกมา จึงสามารถบันทึกภาพได้แม้ในเวลากลางคืน ดาวเทียมสำรวจทรัพยากรที่มีชื่อเสียงมากได้แก่ Landsat, Terra และ Aqua (MODIS Instruments) ดาวเทียมสำรวจทรัพยากรของไทยมีชื่อว่า ทีโอเอส (Theos)



ดาวเทียมอุตุนิยมวิทยา มีวงโคจรหลายระดับขึ้นอยู่กับการใช้งาน ดาวเทียม NOAA มีวงโคจรต่ำถ่ายภาพรายละเอียดสูง ส่วนดาวเทียม GOES และ MTSAT มีวงโคจรค้างฟ้าอยู่ระดับสูง ถ่ายภาพมุมกว้างครอบคลุมทวีปและมหาสมุทร นักอุตุนิยมวิทยาใช้ภาพถ่ายดาวเทียมในการพยากรณ์อากาศ และติดตามการเคลื่อนที่ของพายุจึงสามารถช่วยป้องกันความเสียหายและชีวิตคนได้เป็นจำนวนมาก

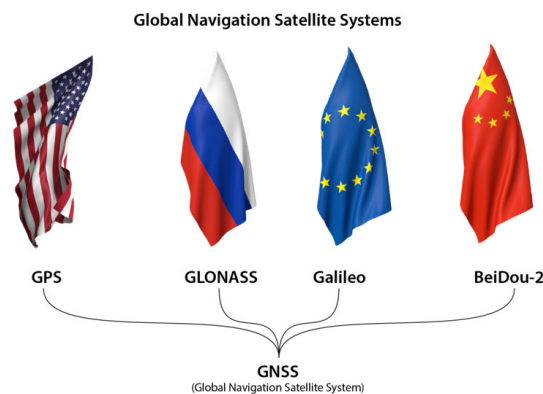
ดาวเทียมเพื่อการนำร่อง Global Navigation Satellite System "GNSS" เป็นระบบบอกตำแหน่ง พิกัดภูมิศาสตร์บนพื้นโลก ซึ่งประกอบด้วยเครือข่ายดาวเทียมจำนวน 32 ดวง โคจรรอบโลกในทิศทางต่างๆ ที่ระยะสูง 20,000 กิโลเมตรส่งสัญญาณมาบนโลกพร้อมๆ กัน แต่เนื่องจากดาวเทียมแต่ละดวงอยู่ห่างจากเครื่องรับบนพื้นโลกไม่เท่ากัน เครื่องรับจึงได้รับสัญญาณจากดาวเทียมแต่ละดวงไม่พร้อมกัน วงจรอิเล็กทรอนิกส์ในเครื่องรับ GNSS นำค่าเวลาที่แตกต่างกันมาคำนวณหาพิกัดภูมิศาสตร์บนพื้นโลก ปัจจุบันเครื่องรับ GNSS เป็นที่นิยมใช้กันในกลุ่มนักเดินทางมีทั้งแบบมือถือ ติดตั้งบนรถ เรือ และเครื่องบิน

ดาวเทียมโทรคมนาคม เช่น Intelsat, Thaicom ส่วนใหญ่เป็นดาวเทียมวงโคจรค้างฟ้า (Geostationary Orbit) เพื่อถ่ายทอดสัญญาณจากทวีปหนึ่งไปยังอีกทวีปหนึ่ง ข้ามส่วนโค้งของโลก ดาวเทียมค้างฟ้า 1 ดวง สามารถส่งสัญญาณครอบคลุมพื้นที่การติดต่อประมาณ 1/3 ของผิวโลก และถ้าจะให้ครอบคลุมพื้นที่ทั่วโลก จะต้องใช้ดาวเทียมในวงโคจรอย่างน้อย 3 ดวง อย่างไรก็ตามดาวเทียมวงโคจรค้างฟ้าจะลอยอยู่ในแนวเส้นศูนย์สูตรโลกเท่านั้น ดังนั้นสัญญาณจะไม่สามารถครอบคลุมบริเวณขั้วโลกได้เลย เนื่องจากประชากรส่วนใหญ่อยู่บริเวณเส้นศูนย์สูตร

ดาวเทียมภารกิจพิเศษ นอกจากดาวเทียมทั่วไปที่ใช้งานเกี่ยวข้องกับชีวิตประจำวันตามที่กล่าวไปแล้ว ยังมีดาวเทียมอีกหลายชนิดที่ส่งขึ้นไปเพื่อปฏิบัติการกิจพิเศษเฉพาะทาง เช่น ดาวเทียมเพื่อการวิจัยทางวิทยาศาสตร์ กล้องโทรทรรศน์อวกาศ ดาวเทียมจารกรรม ดาวเทียมทางทหาร ดาวเทียมประเภทนี้มีระยะสูงและรูปแบบของวงโคจรต่างๆ กันขึ้นอยู่กับวัตถุประสงค์ของการใช้งาน

GNSS

GNSS (Global Navigation Satellite System) หรือระบบระบุตำแหน่งบนพื้นผิวโลกโดยใช้ดาวเทียม หมายถึง การนำคลื่นสัญญาณวิทยุและรหัสจากดาวเทียมบอกตำแหน่งมาใช้บอกค่าพิกัดของสิ่งต่างๆ บนพื้นโลก



ระบบดังกล่าวเกิดจากการส่งดาวเทียมจำนวน 24 ดวงขึ้นสู่อวกาศที่ระดับความสูงประมาณ 2,200 กิโลเมตรเหนือพื้นผิวโลก ดาวเทียมทั้งหมดจะแบ่งออกเป็นวงโคจร 6 วงโคจร แต่ละวงโคจรมีดาวเทียมประจำการอยู่จำนวน 4 ดวง เพื่อทำหน้าที่ส่งคลื่นสัญญาณบอกตำแหน่ง

การโคจรมายังสถานควบคุมดาวเทียมภาคพื้นดินและเครื่องรับสัญญาณจากดาวเทียม โดยเครื่องรับสัญญาณจากดาวเทียมจะนำสัญญาณจากดาวเทียมอย่างน้อย 4 ดวงมาคำนวณหาค่าพิกัดละติจูด (Latitude) และลองจิจูด (Longitude) ของตำแหน่งต่างๆ ทั้งที่อยู่บนพื้นผิวโลกและอยู่เหนือพื้นผิวโลกขึ้นไป ระบบนี้จึงมีประโยชน์ต่อการบอกตำแหน่งและทิศทางในการเดินทางทั้ง ทางบก ทางน้ำ และทางอากาศ

ระบบกำหนดพิกัดบนพื้นผิวโลกประกอบด้วยสามส่วนหลัก คือ

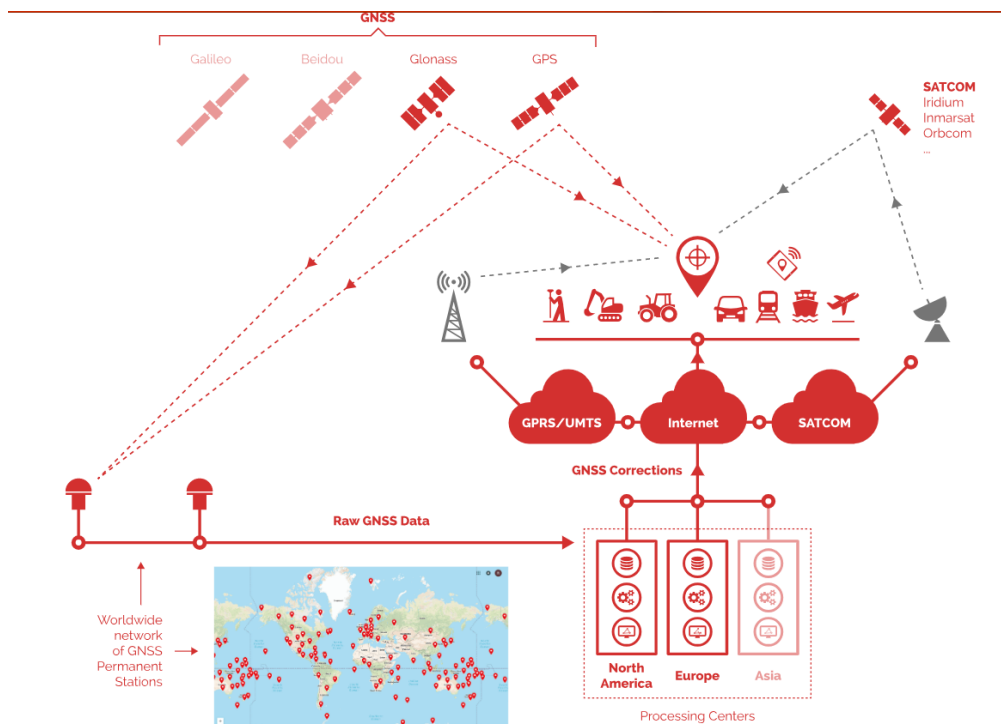
ส่วนอวกาศ (Space Segment) คือ ดาวเทียมที่อยู่บนอวกาศ ทำหน้าที่ส่งสัญญาณ

ส่วนสถานีควบคุม (Control Segment) คือ สถานีภาคพื้นดินที่กระจายอยู่ตามส่วนต่างๆ บนพื้นโลก ทำหน้าที่ในการติดต่อสื่อสารระหว่างผู้ใช้กับดาวเทียม

ส่วนผู้ใช้ (User Segment) คือ เครื่องรับสัญญาณมีหลายขนาด สามารถพกพาไปได้หรือติดไว้กับยานพาหนะ เครื่องนี้ทำหน้าที่แปลงสัญญาณและคำนวณหาพิกัดตำแหน่งบนพื้นโลก

ประโยชน์ของระบบกำหนดพิกัดตำแหน่งบนพื้นโลกมีหลายด้าน ได้แก่ การนำร่อง การติดตามความเคลื่อนไหว การสำรวจรังวัด และการทำแผนที่ การอ้างอิง การวัดเวลาที่เที่ยงตรงที่สุดในโลก และการประยุกต์ใช้ในงานอื่นๆ เช่น การควบคุมเครื่องจักรกล การขนส่งทางน้ำและทางทะเล การเพิ่มประสิทธิภาพของระบบคมนาคม

ระบบระบุตำแหน่งบนพื้นผิวโลกนี้เป็นเครื่องมือที่สำคัญมากในการสร้าง CanSat เพราะ จะทำให้ผู้ศึกษาทราบว่าดาวเทียมมีตำแหน่งหรือพิกัดอยู่บริเวณใดของโลก และนอกจาก GNSS จะบอกค่าพิกัดทางภูมิศาสตร์และยังบ่งบอกความสูงอีกด้วย ดังนั้นระบบนี้จึงสามารถบอกค่าได้ในแกนสามมิติ คือ X, Y, และ Z ช่วยทำให้เราสามารถซึ่งศึกษาข้อมูลที่เกี่ยวข้องได้สะดวกขึ้น เช่น หากเราต้องการทราบว่าภูเขาสูงเท่าใด GNSS สามารถบอกได้ รวมถึงการนำไปประยุกต์ใช้ในการกำหนดการเดินทางไปยังพิกัดที่ตั้งไว้ได้อีกด้วยระบบ GNSS ในชีวิตประจำวันของเราได้จาก Application ต่างๆ เช่น Google Map, Grab, Pokemon Go, Facebook, และ Instagram เป็นต้น สิ่งเหล่านี้เป็นสิ่งที่อยู่รอบตัวเราแต่เราแทบจะไม่สังเกตว่ามีการใช้ GNSS ในชีวิตประจำวันของเรา นอกจากนี้ยังรวมไปถึง Smart Phone ในปัจจุบันมีระบบแจ้งเตือนพิกัดอยู่ตลอดเวลาเราสามารถสังเกตได้จาก Metadata ของรูปถ่ายได้



Geography

Geography หรือภูมิศาสตร์ มีความหมายว่า Geo (แปลว่าโลก) รวมกับ Graphein (แปลว่า เขียน) จึงรวมกันกลายเป็นศาสตร์ที่เกี่ยวกับการเขียนเกี่ยวกับโลก คำว่า โลก ก็แบ่งออกเป็นสองความหมาย คือ Earth และ World

Earth หมายถึง โลกที่เน้นความหมายทางวิทยาศาสตร์ เป็นโลกที่แบ่งออกเป็น 4 ภาค หรือ Sphere ได้แก่ Atmosphere, Lithosphere, Hydrosphere, และ Biosphere

World หมายถึง โลกที่เน้นความหมายทางสังคมศาสตร์ ศึกษาความสัมพันธ์ระหว่างสังคมกับพื้นที่ ศึกษาพฤติกรรมของมนุษย์ หรือปฏิสัมพันธ์ระหว่างมนุษย์กับพื้นที่หรือสถานที่ ความหมายของ World จึงกว้างและครอบคลุมกว้าง Earth

ภูมิศาสตร์ แบ่งออกเป็น 3 สาขา ได้แก่ ภูมิศาสตร์กายภาพ (Physical Geography) ภูมิศาสตร์มนุษย์ (Human Geography) และเทคนิคทางภูมิศาสตร์ (Technical Geography) ทั้ง 3 สาขาถูกจำแนกออกจากลักษณะที่แตกต่างกันของพื้นที่ ซึ่งเราสามารถศึกษาที่สามสาขาจากพื้นที่ศึกษาเดียวกันได้ เพราะทั้งสามมีความเชื่อมโยงและมีปฏิสัมพันธ์ที่เกี่ยวข้องกัน ยกตัวอย่างเช่น ลักษณะพื้นที่ราบ เหมาะสมกับการเพาะปลูก เพราะเป็นพื้นที่ที่ต่ำจึงทำให้เกิดการสะสมตัวของตะกอนที่มีความอุดมสมบูรณ์จึงทำให้ มนุษย์มักเพาะปลูกบริเวณที่เรียกว่า "Flood Plain" บริเวณนี้จึงก่อให้เกิดวิถีชีวิตและวัฒนธรรมของมนุษย์ที่อาศัยอยู่ร่วมกัน หากเราใช้แผนที่ในการศึกษาเราจะเห็นปฏิสัมพันธ์ของภูมิสัณฐานกับวิถีชีวิตอย่างชัดเจนและหลากหลายระดับมากขึ้น

Physical Geography

Physical Geography หรือภูมิศาสตร์กายภาพ หมายถึง วิชาที่ศึกษาเกี่ยวกับลักษณะของโลก โดยเน้นการศึกษาด้านธรณีวิทยา ปัจจัยที่มีอิทธิพลต่อการเปลี่ยนแปลงของแผ่นเปลือกโลก โดยแบ่งโลกออกเป็นส่วนต่างๆ ในการศึกษา ตามความหมายของ Earth ซึ่งภูมิศาสตร์กายภาพ ยกตัวอย่างเช่น ภูมิศาสตร์สัณฐาน ภูมิศาสตร์ดิน ภูมิอากาศ ภูมิศาสตร์สิ่งแวดล้อม และภูมิศาสตร์ภูมิภาค เป็นต้น



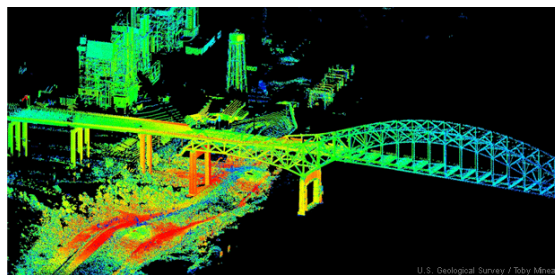
Human Geography

Human Geography หรือภูมิศาสตร์มนุษย์ หมายถึง วิชาที่ศึกษาเกี่ยวกับลักษณะและพลวัตของมนุษย์ และความสัมพันธ์ระหว่างกายภาพกับกิจกรรมของมนุษย์ ตามความหมายของ World ซึ่งยกตัวอย่างเช่น ภูมิศาสตร์เศรษฐกิจ ภูมิศาสตร์เมือง ภูมิศาสตร์วัฒนธรรม ภูมิศาสตร์การเมืองและภูมิศาสตร์เกษตร เป็นต้น



Technical Geography

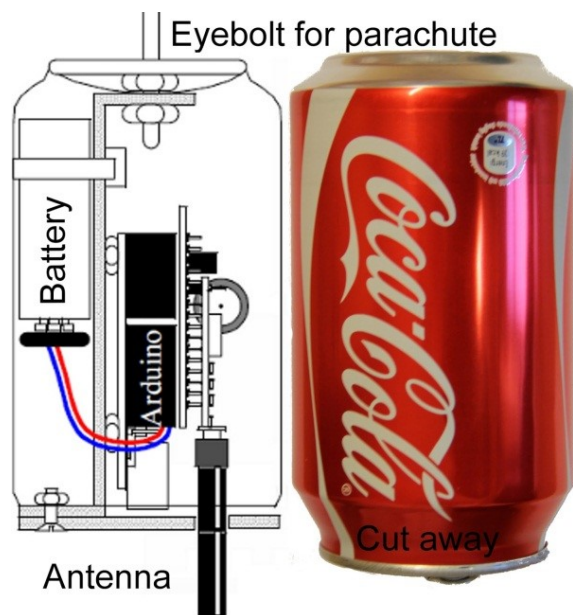
Technical Geography หรือเทคนิคและวิธีการทางภูมิศาสตร์ หมายถึง การใช้เทคโนโลยีมาใช้ในการศึกษาลักษณะทางกายภาพของโลก และศึกษาความสัมพันธ์ของมนุษย์กับพื้นที่ โดยเครื่องมือทางภูมิศาสตร์ในปัจจุบันมีการนำข้อมูลทางพื้นที่มาประยุกต์ใช้ในการศึกษา เรียกว่า ภูมิสารสนเทศศาสตร์ (Geography Information System : GIS) เครื่องมือทางภูมิสารสนเทศที่อยู่ใกล้ตัวของเรา ยกตัวอย่างเช่น แผนที่ ภาพถ่ายทางดาวเทียม ภาพถ่ายทางอากาศ ระบบระบุพิกัดบนโลกจากดาวเทียม เป็นต้น



Conclusion

โดยสรุป การศึกษาลักษณะทางกายภาพและความสัมพันธ์ของมนุษย์กับพื้นที่ จำเป็นที่จะต้องใช้เครื่องมือทางภูมิศาสตร์ "ดาวเทียม" หรือ Satellite เป็นเครื่องมือสำคัญในภูมิศาสตร์สมัยใหม่ เนื่องจากโลกในปัจจุบันมีการพัฒนาของเทคโนโลยีการสื่อสารและการขนส่ง (Telecommunication & Transportation) ทำให้นักภูมิศาสตร์ได้นำเทคโนโลยีเหล่านี้มาศึกษาประเด็นต่างๆ เพื่อใช้ในการพัฒนาความรู้และคุณภาพชีวิตของมนุษย์ให้สามารถอยู่ร่วมกับสิ่งแวดล้อม หรือใช้ในการแก้ไขปัญหาต่างๆ เช่น การนำภาพถ่ายดาวเทียมมาใช้ในการสร้างแผนที่เพื่อนำไปวางผังเมือง การนำ GNSS มาติดตามการเกิดพายุ การใช้ LIDAR ในการสำรวจสถานที่ที่มนุษย์ไม่สามารถเข้าถึงได้

อย่างไรก็ตาม การสร้าง CANSAT เพื่อศึกษาประเด็นที่ผู้ศึกษาสนใจนั้น ผู้ศึกษาจะเป็นจะต้องทราบถึงลักษณะสำคัญของพื้นที่เสียก่อน จึงจะสามารถกำหนดประเด็นที่ผู้ศึกษาสนใจจะศึกษาได้ และนำความรู้จากสาขาวิชาอื่นมาร่วมใช้ในการออกแบบรูปแบบและวิธีการศึกษา ในส่วนของการสร้าง CANSAT ผู้ศึกษาจะต้องนำความรู้ทางด้านวิทยาศาสตร์และสังคมศาสตร์มาร่วมใช้ในการศึกษา ยกตัวอย่างเช่น การสร้างดาวเทียมตรวจจับฝุ่น เพื่อวัดค่าปริมาณฝุ่นละอองและก๊าซคาร์บอนไดออกไซด์ เพื่อใช้ในการกำหนดนโยบายภายในโรงเรียน ผู้ศึกษาจะต้องทราบว่าต้องการข้อมูลใดบ้างและนำข้อมูลเหล่านั้นเข้าสู่กระบวนการวิเคราะห์ที่ใด และผลที่ได้ออกมาสามารถอธิบายเรื่องอะไรได้บ้าง สิ่งเหล่านี้เป็นสิ่งที่สำคัญอย่างยิ่งก่อนเริ่มทำ CANSAT



กิจกรรมฐาน สำรวจและรังวัด

คำสั่ง ให้ผู้เข้าร่วมค่ายค้นหาสถานที่จากค่าพิกัด

1. 13°43'09.9"N 100°15'45.1"E คือสถานที่ _____
2. 13.721324, 100.261680 คือสถานที่ _____
3. จุดที่เรายืนอยู่มีพิกัด Latitude _____ และ Longitude _____

คำสั่ง ให้ผู้เข้าร่วมค่ายรังวัดสนาม

$$\text{สูตรมาตราส่วนแผนที่} = \frac{\text{ระยะในแผนที่ (Map Distance)}}{\text{ระยะทางในภูมิประเทศ (Ground Distance)}}$$

สูตร การหาพื้นที่ สี่เหลี่ยมพื้นภาพ กว้าง * ยาว

(รังวัดโดยภาพดาวเทียม)

ด้านกว้าง _____ เมตร ด้านยาว _____ เมตร ขนาดพื้นที่ _____ ตารางเมตร

(รังวัดโดยการสำรวจ)

ด้านกว้าง _____ เมตร ด้านยาว _____ เมตร ขนาดพื้นที่ _____ ตารางเมตร

คำสั่ง ให้ผู้เข้าร่วมค่ายคิดหัวข้อการวิจัยที่เหมาะสมกับบ้านผู้หว่าน

1. ข้อมูลที่สามารถจัดเก็บได้จากการสำรวจ (โดย CANSAT และ HUMAN)

(1) _____ (2) _____ (3) _____

2. วัตถุประสงค์การของการศึกษา (นำข้อมูลที่เก็บได้มาศึกษาอะไร)

(1) _____

(2) _____

3. ประโยชน์ของการศึกษา _____

THAICOM Public Company LIMITED

THAICOM 6



“นำเทคโนโลยีด้านดาวเทียมการสื่อสาร ผสมผสานกับประสบการณ์อันเชี่ยวชาญของบุคลากร เพื่อ
การทำประโยชน์ ในการสร้างโอกาสให้แก่ประชาชนและชุมชน เพื่อการพัฒนามาตรฐานความเป็นอยู่
ประชาชน เสริมสร้างสิ่งแวดล้อม คุณภาพชีวิตและสังคมที่แข็งแกร่งและมั่นคง”

- บริษัท ไทยคม จำกัด (มหาชน)

บริษัท ไทยคม จำกัด (มหาชน) ก่อตั้งขึ้นเมื่อวันที่ 7 พ.ย. 2534 ด้วยความภาคภูมิใจในชื่อ “ไทยคม” (THAICOM) ที่ได้รับพระราชทานนามดาวเทียมสื่อสารแห่งชาติดวงแรก จากพระบาทสมเด็จพระเจ้าอยู่หัวภูมิพลอดุลยเดช เมื่อวันที่ 17 ธันวาคม 2534 ชื่อไทยคมนั้น มาจากคำว่า Thai Communications หรือ ไทยคมนาคม ซึ่งเป็นสัญลักษณ์เชื่อมโยงประเทศไทยกับเทคโนโลยีสมัยใหม่จนถึงปัจจุบัน

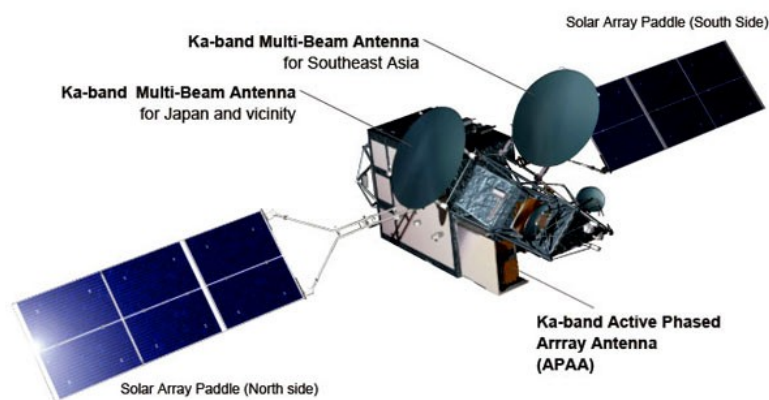
บริษัท ไทยคม จำกัด (มหาชน) เป็นบริษัทจดทะเบียนในตลาดหลักทรัพย์แห่งประเทศไทย ภายใต้ชื่อ “THCOM” ปัจจุบันบริษัท มีหุ้นสามัญจำนวน 1,096,101,954 หุ้น แบ่งเป็นหุ้นที่ออกและชำระแล้ว 1,096,101,954 หุ้น มูลค่าที่ตราไว้หุ้นละ 5 บาท ส่งผลให้บริษัทมีทุนจดทะเบียน จำนวน 5,499,884,200.00 บาท และมีทุนจดทะเบียนชำระแล้วจำนวน 5,480,509,770.00 บาท

ตลอดระยะเวลา 24 ปีที่ผ่านมา บริษัท ไทยคม จำกัด (มหาชน) มุ่งสร้างสรรค์นวัตกรรม ทางธุรกิจและเทคโนโลยี สร้างองค์ความรู้และใช้ความเชี่ยวชาญ ตลอดจนทรัพยากรของบริษัทในการยกระดับธุรกิจดาวเทียมและการสื่อสาร เพื่อรองรับการเติบโตและการให้บริการในยุคเศรษฐกิจดิจิทัลและนวัตกรรม ซึ่งเน้นการได้รับข้อมูลที่รวดเร็วและสามารถนำมาสร้างคุณค่าในเชิงเศรษฐกิจควบคู่กับการพัฒนาคุณภาพชีวิต ไทยคมได้สร้างสรรค์นวัตกรรมทางธุรกิจ และต่อยอดเทคโนโลยีใหม่ๆ อยู่เสมอ โดยมุ่งตอบสนองต่อความท้าทายและการเปลี่ยนแปลงทั้งในปัจจุบันและอนาคต ด้วยแนวคิดความยั่งยืน ไทยคมจึงเป็นผู้นำธุรกิจดาวเทียมแห่งเอเชียที่ไม่เพียงให้บริการการสื่อสารผ่านโครงข่ายโทรคมนาคมผ่านดาวเทียมระดับโลก แต่ยังสร้างการเติบโตที่ยั่งยืนอีกด้วย

การสื่อสารเป็นปัจจัยขั้นพื้นฐานในการดำรงชีวิตที่มีความสำคัญอย่างยิ่งต่อสภาพสังคมในทุกระดับจึงปฏิเสธไม่ได้ว่า การสื่อสารเป็นส่วนสำคัญในการพัฒนาเศรษฐกิจและสังคม และมีส่วนช่วยสนับสนุนให้เกิดความสมดุลและความเจริญก้าวหน้าของสิ่งแวดล้อมในทุกภาคส่วนอย่างไม่หยุดยั้ง

อย่างไรก็ตามมีกลุ่มคนจำนวนไม่น้อยที่ยังไม่สามารถเข้าถึงระบบการสื่อสารที่ทันสมัยในปัจจุบัน จึงเป็นความท้าทายประการสำคัญของไทยคม ในการมีส่วนช่วยพัฒนาศักยภาพของทรัพยากรทางเศรษฐกิจ สังคม และสิ่งแวดล้อม เพื่อให้มั่นใจคนทุกกลุ่มจะมีโอกาสเข้าถึงการสื่อสารอย่างไร้ขีดจำกัด เพื่อให้ทุกคนสามารถพัฒนาคุณภาพชีวิตและความเป็นอยู่ที่ดียิ่งขึ้น ไทยคมตระหนักถึงความสำคัญของการดำรงอยู่ร่วมกัน และเชื่อว่าพลังอันแข็งแกร่งจากทุกภาคส่วน คือรากฐานสำคัญในการขับเคลื่อนองค์กรไปสู่อนาคต พร้อมก้าวสู่การเติบโตอย่างยั่งยืนร่วมกัน

บริษัทเป็นผู้ให้บริการและดำเนินโครงการดาวเทียมสื่อสารแห่งชาติ ภายใต้สัญญาดำเนินการดาวเทียมสื่อสารภายในประเทศระหว่างบริษัทกับกระทรวงเทคโนโลยีสารสนเทศและการสื่อสาร (ปัจจุบันเปลี่ยนชื่อเป็นกระทรวงดิจิทัลเพื่อเศรษฐกิจและสังคม) (“กระทรวงฯ”) เป็นระยะเวลา 30 ปี สิ้นสุดปี พ.ศ. 2564 จนถึงปัจจุบันบริษัทได้จัดส่งดาวเทียมขึ้นสู่วงโคจรแล้วจำนวนทั้งสิ้น 8 ดวง คือ ดาวเทียมไทยคม 1 ไทยคม 2 ไทยคม 3 (ดาวเทียมไทยคม 1 - 3 ได้ปลดระวางเป็นที่เรียบร้อยแล้ว) ไทยคม 4 ไทยคม 5 ไทยคม 6 ไทยคม 7 และไทยคม 8



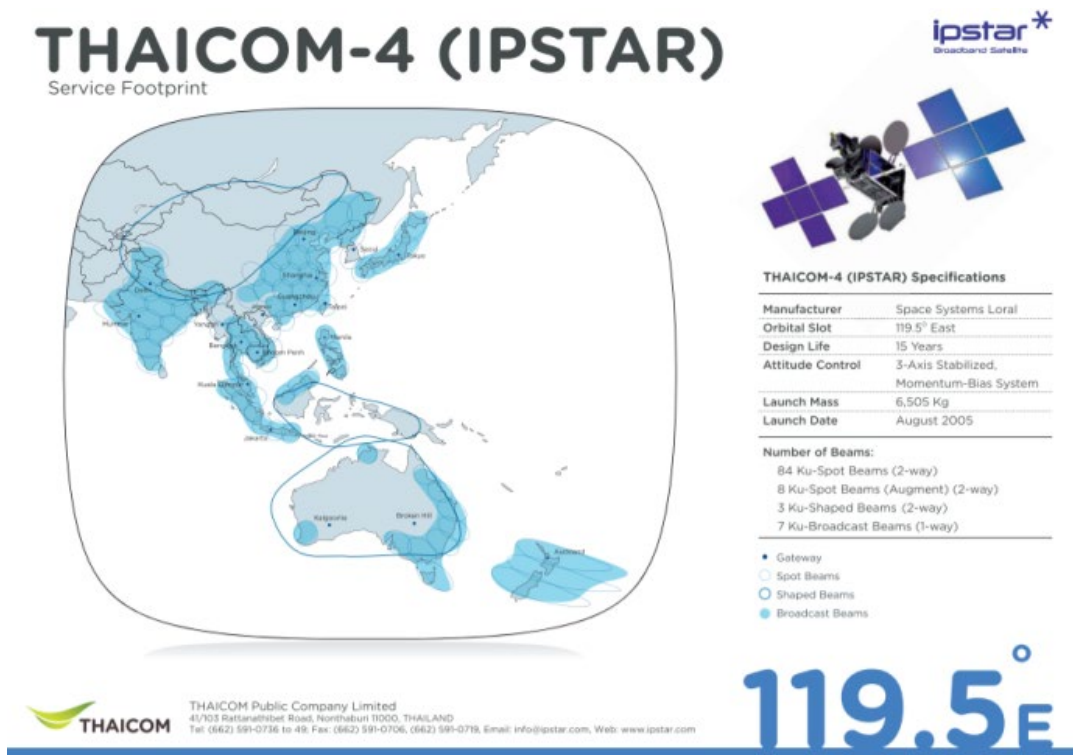
ส่วนประกอบของดาวเทียม KIZUNA

ดาวเทียมสื่อสาร (ไทยคม)

ดาวเทียมสื่อสารเป็นดาวเทียมที่ต้องทำงานอยู่ตลอดเวลา เรียกได้ว่าทำงานตลอด 24 ชม. ไม่มีวันหยุด เพื่อที่จะเชื่อมโยงเครือข่ายการสื่อสารของโลกเข้าไว้ด้วยกัน ดาวเทียมสื่อสารเมื่อถูกส่งเข้าสู่วงโคจร ดาวเทียมพร้อมทำงานได้ทันที ดาวเทียมส่งสัญญาณกลับไปยังสถานีภาคพื้นดิน สถานีภาคพื้นดินรับสัญญาณ โดยใช้อุปกรณ์ ที่เรียกว่า "Transponder" ซึ่งเป็นอุปกรณ์ที่ทำหน้าที่พักสัญญาณ แล้วกระจายสัญญาณไปยังจุดรับสัญญาณต่างๆ บนพื้นโลก ดาวเทียมสื่อสารสามารถส่งผ่านสัญญาณโทรศัพท์ ข้อมูลต่างๆ รวมถึงสัญญาณภาพโทรทัศน์ได้ไปยังพื้นที่ต่างๆ ที่การให้บริการครอบคลุม

จุดประสงค์ในการสร้าง

เป็นดาวเทียมที่มีจุดประสงค์เพื่อการสื่อสารและโทรคมนาคม จะถูกส่งไปในช่วงของอวกาศเข้าสู่วงโคจรโดยมีความห่างจากพื้นโลกโดยประมาณ 35.786 กิโลเมตร ซึ่งความสูงในระดับนี้จะทำให้เกิดแรงดึงดูดระหว่างโลกกับดาวเทียม ในขณะที่โลกหมุนก็จะส่งแรงเหวี่ยง ทำให้ดาวเทียมเกิดการโคจรรอบโลกตามการหมุนของโลก



V. Data Analysis & Analytics

by Vivatsathorn Thitasirivit

ในทางวิทยาศาสตร์นั้น เมื่อเราเกิดคำถาม แล้วต้องการทราบคำตอบหรือแนวโน้ม เราเริ่มจากการเก็บข้อมูลที่ต้องการศึกษา แล้วนำมาวิเคราะห์ผลเพื่อหาแนวโน้มหรือคำตอบที่แท้จริง และหาคำตอบของข้อสงสัยที่ตั้งไว้ตามวัตถุประสงค์ของการทดลอง ในบทนี้จะพูดถึงการวิเคราะห์ผลข้อมูลทั่วไป (Data Analysis) และการวิเคราะห์ผลเชิงสถิติ (Analytics)

ข้อมูลที่เราศึกษานั้น มีอยู่ในหลากหลายรูปแบบ เช่น ตัวเลข, ปริมาณไอน้ำในอากาศ, อุณหภูมิ, ความดันไอน้ำอิมตัว, ภาพถ่ายทางอากาศ, พิกัด ซึ่งข้อมูลเหล่านี้มีความแตกต่างอย่างชัดเจนในด้านวิธีการวิเคราะห์ ข้อมูลเหล่านี้แบ่งโดยทั่วไปได้ 2 รูปแบบ คือ ข้อมูลเชิงปริมาณ (Quantitative Data) และข้อมูลเชิงคุณภาพ (Qualitative Data) ซึ่งจะอธิบายในรายละเอียดดังนี้

1. ข้อมูลเชิงปริมาณ (Quantitative Data)

ข้อมูลเชิงปริมาณ คือ ข้อมูลที่สามารถวัดได้ คำนวณได้ และแสดงออกมาเป็นค่าของตัวเลขได้ เช่น อุณหภูมิ, ปริมาตรของน้ำในแก้ว, มุมของการหักเหของแสง ซึ่งในการวิเคราะห์ข้อมูลเหล่านี้ นิยมวิเคราะห์ในเชิงการคำนวณและเชิงสถิติ

1.1. การดูแนวโน้มของข้อมูลเชิงปริมาณ (Data Trends)

เราวิเคราะห์แนวโน้มของข้อมูลเพื่อการทำนายว่า ข้อมูลชุดนั้น ๆ มีทิศทางของข้อมูลไปอย่างไร เช่น เมื่อเราเทียบคันเร่งรถยนต์แล้วดูมิเตอร์อัตราเร็วทุก ๆ 1 วินาทีได้ข้อมูลคือ 0, 10, 30, 70, 120 km/h เราสามารถทำนายข้อมูลชุดนี้ได้ว่า มีแนวโน้มที่จะเพิ่มขึ้นเรื่อย ๆ และกำลังเพิ่มขึ้นด้วยอัตราที่เร็วขึ้นเรื่อย ๆ

การดูแนวโน้มนั้น เรายังสามารถใช้หาได้ว่า ข้อมูลนั้น มีอัตราการเปลี่ยนแปลงที่ค่อย ๆ ช้าลง คงตัว หรือเร็วขึ้นเรื่อย ๆ ตามเวลาที่เปลี่ยนไปโดยยังไม่จำเป็นต้องคำนวณอะไรทั้งสิ้น ใช้การอนุมาน (Inference) ได้

1.2. การวิเคราะห์ข้อมูลเชิงปริมาณ

ประโยชน์ของข้อมูลเชิงปริมาณ คือ สามารถนำมาคำนวณทางคณิตศาสตร์ได้ เช่น ถ้าเราเก็บข้อมูลความสูงของวัตถุหนึ่งทุก ๆ 1 วินาที ได้ข้อมูลดังนี้

ความสูง (m)	250	240	230	220	210	200	190	180	170
เวลา (s)	0	1	2	3	4	5	6	7	8

เราสามารถหาอัตราเร็วเฉลี่ยของการตกของวัตถุนี้ได้, อัตราเร็วเฉลี่ยมีนิยามคือการเปลี่ยนไปของตำแหน่งต่อ 1 หน่วยเวลา หรือในรูปสมการ $v_{\text{เฉลี่ย}} = \frac{\Delta}{t}$ หน่วยเป็น m/s (เมตรต่อวินาที)

การวิเคราะห์ข้อมูลประเภทนี้มีหลายวิธีไม่จำกัด โดยจะยกตัวอย่างมาบางรูปแบบที่ใช้บ่อย ๆ

1.2.1. ค่าเฉลี่ยเลขคณิต (Arithmetic Mean)

ค่าเฉลี่ยเลขคณิตมักใช้เมื่อต้องการตัวแทนของข้อมูลช่วงหนึ่งมาเพียงค่าเดียว เพื่อบ่งบอกสภาพของกลุ่มนั้น ๆ เช่น เก็บข้อมูลความสูงของนักเรียน 3 คนได้คือ 150, 165, และ 180 เมตร ต้องการทราบว่า 3 คนนี้สูงประมาณเท่าใด ลักษณะนี้จะใช้ค่าเฉลี่ยเพื่อเป็นตัวแทนของข้อมูลชุดนั้น ๆ การหาค่าเฉลี่ยสามารถคำนวณได้โดยใช้สูตร

$$\bar{x} = \frac{\sum x}{N}$$

โดยที่ \bar{x} คือ ค่าเฉลี่ยเลขคณิต, $\sum x$ คือผลรวมของข้อมูล, และ N คือจำนวนข้อมูลในชุดนั้น ๆ เช่น จากตัวอย่างข้างต้น สามารถหาความสูงเฉลี่ยได้คือ

$$\bar{x} = \frac{150+165+180}{3} = \frac{495}{3} = 165 \text{ เมตร}$$

1.2.2. พิสัย (Range)

พิสัย คือผลต่างระหว่างค่าสูงสุดกับค่าต่ำสุดของข้อมูล ($R = \text{MAX} - \text{MIN}$) นิยมใช้เมื่อต้องการหาความต่างที่มีค่ามากที่สุดที่สุดในชุดข้อมูล เช่น เก็บข้อมูลอุณหภูมิในห้องเมื่อระยะเวลาหนึ่ง ได้ข้อมูลอุณหภูมิคือ 24, 27, 26, 23, 25, 25, 26 °C มา แล้วอยากทราบว่า อุณหภูมิสูงสุดที่เป็นไปได้และอุณหภูมิต่ำสุดที่เป็นไปได้ในห้องนี้มีค่าเท่าใด ก็คือ $27 - 23 = 4$ °C ซึ่งพิสัยสามารถตีความถึงความเหวี่ยงของข้อมูลคร่าว ๆ ได้ ว่าข้อมูลชุดนั้นมีความคงที่มากเพียงใด เสถียรหรือไม่ จะเห็นได้ว่าข้อมูลชุดนี้บอกว่าแอร์ตัวนี้ไม่มีประสิทธิภาพ เพราะอุณหภูมิมีความเหวี่ยงมากถึง 4 °C

1.2.3. การสร้างความสัมพันธ์ของข้อมูลเพื่อเปรียบเทียบ

โดยส่วนใหญ่จะใช้เมื่อต้องการทราบว่า ข้อมูลที่เก็บมา มีความตรงมากเพียงใดเมื่อเทียบกับข้อมูลต้นแบบ มักใช้การคูณกันแล้วหาว่า มีความเที่ยงตรงที่ % โดยใช้ความสัมพันธ์ว่า

$$\text{Accuracy} = \frac{A \times B \times C \times D \times \dots}{A_0 \times B_0 \times C_0 \times D_0 \times \dots} \times 100\%$$

โดย A คือข้อมูลที่เก็บมาได้หรือตัวแทนของข้อมูลที่เก็บมาได้ และ A_0 คือข้อมูลต้นแบบที่ต้องการใช้เทียบ แต่ส่วนมาก จะแทนค่าบนตัวเศษให้น้อยกว่าตัวส่วนเสมอเพื่อความ make sense และมักแทน A เป็น $A_0 - |A - A_0|$, Accuracy อาจหมายถึง Relative Value ก็ได้

1.2.4. ค่าสัมประสิทธิ์สหสัมพันธ์ (Coefficient of Correlation, r)

เป็นการสร้างความสัมพันธ์ของข้อมูลเพื่อหาแนวโน้มจากความสัมพันธ์ของตัวแปร 2 ตัวแปร (การใช้งานค่าสัมประสิทธิ์สหสัมพันธ์มีเงื่อนไขค่อนข้างยุ่งยากและซับซ้อน จึงไม่ขออธิบายในคู่มือนี้ สามารถหาศึกษาอ่านทฤษฎีและการนำมาใช้ได้จาก Internet)

1.3. การนำเสนอข้อมูลเชิงปริมาณ (Data Visualization)

เมื่อเก็บค่าได้มาแล้วหรือวิเคราะห์ผลมาแล้ว การนำเสนอข้อมูลชนิดนี้ให้เข้าใจง่ายก็เป็นเรื่องสำคัญเช่นกัน เพราะหากแสดงข้อมูลดิบ กรณีทั่วไปคงไม่มีใครอยากรู้

การนำเสนอข้อมูลชนิดนี้ขึ้นอยู่กับประเภทของข้อมูลว่าเป็นข้อมูลแบบไหน โดยทั่วไปมักใช้กราฟเมื่อมีความสัมพันธ์ของข้อมูล หรือมี Trends ข้อมูล หรืออาจใช้ชนิดอื่น ๆ เช่น Charts, Table, Infographics, Distribution Map, Heat Map, Timeline สำหรับข้อมูลชนิดอื่น ๆ โดยจะอธิบายในส่วนของการกราฟและแผนภูมิเป็นหลัก

1.3.1. ตาราง (Table)

ตาราง คือ รูปแบบของการเก็บข้อมูลเมื่อมีอย่างใดอย่างหนึ่งคงตัว เรียกข้อมูลตามแนวตั้งว่า “คอลัมน์ (Column)” และ เรียกข้อมูลตามแนวนอนว่า “แถว (Row)” เช่น

ตัวอย่างตารางข้อมูลที่มีหัวข้อของแต่ละคอลัมน์

ข้อมูลชุดที่	อุณหภูมิ (°C)	ความชื้น (%)	ความดัน (mmHg)
1	30	65	765
2	26	61	764
...
N	27	58	763

แถว

คอลัมน์

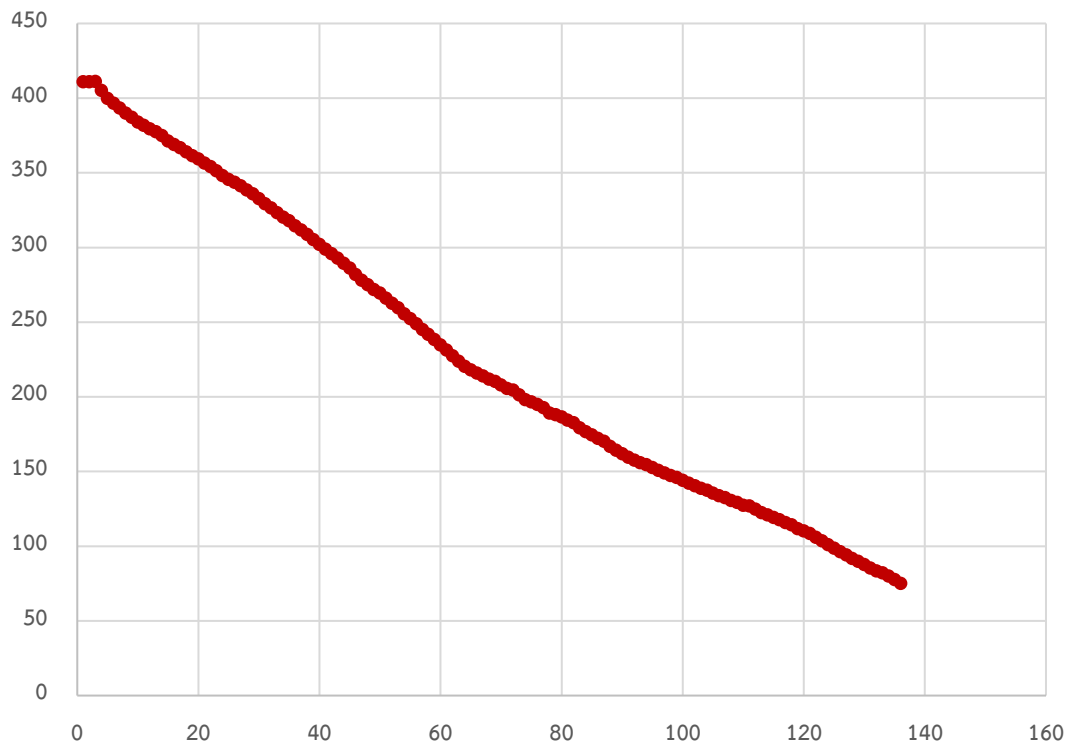
ซอฟต์แวร์ที่คนส่วนใหญ่ใช้เพื่อสร้างตาราง แกะไขข้อมูลในตารางก็คือ “Microsoft Excel”

จะเห็นได้ว่า ตารางข้างบนนั้น มีหัวข้อของแต่ละ column เช่น column ที่ 1 บอกถึงข้อมูลชุดที่, column ที่ 2 บอกถึงอุณหภูมิ ฯลฯ และ ข้อมูลที่สอดคล้องกับหัวข้อก็จะถูกเขียนในแถวถัดมาเรื่อย ๆ ตารางจึงเป็นการแสดงข้อมูลพื้นฐานที่ต้องทราบและเราสามารถเอาข้อมูลในตารางมาพลอตกราฟต่อไปได้

1.3.2. กราฟ (Graph)

กราฟ ใช้สื่อความหมายเชิงความสัมพันธ์ระหว่างข้อมูลหนึ่ง กับข้อมูลหนึ่ง โดยทั่วไปกราฟ 2 มิติ มี 2 แกน คือ แกนตั้ง และ แกนนอน, โดยส่วนใหญ่ตีความว่า ข้อมูลในแกนตั้ง มีความสัมพันธ์กับข้อมูลในแกนนอนอย่างไร เช่น ความสูงของวัตถุตามเวลา

กราฟความสัมพันธ์ระหว่างความสูง (แกนตั้ง, เมตร) - เวลา (แกนนอน, วินาที)



ข้อมูลชุดนี้เมื่อพลอตออกมาเป็นแกนตั้ง-แกนนอนแล้ว เราจะเห็นความพิเศษของกราฟ คือ มันเป็นกราฟเส้นตรง (Linear Graph) ซึ่งอาจไม่ตรงเป๊ะ แต่เราดูแนวโน้ม เนื่องจากการเบี่ยงของกราฟอาจเกิดจากความไม่พอดีของข้อมูล หรืออาจเกิดจากความผิดพลาด (Error) ในการวัด

Linear Graph มีความพิเศษคือ สามารถหาความชันของกราฟ (Slope) ได้ โดย slope นั้นขึ้นอยู่กับความหมายของแกนตั้ง-แกนนอน สรุป slope ของกราฟคืออัตราส่วนระหว่างแกนตั้งต่อแกนนอน

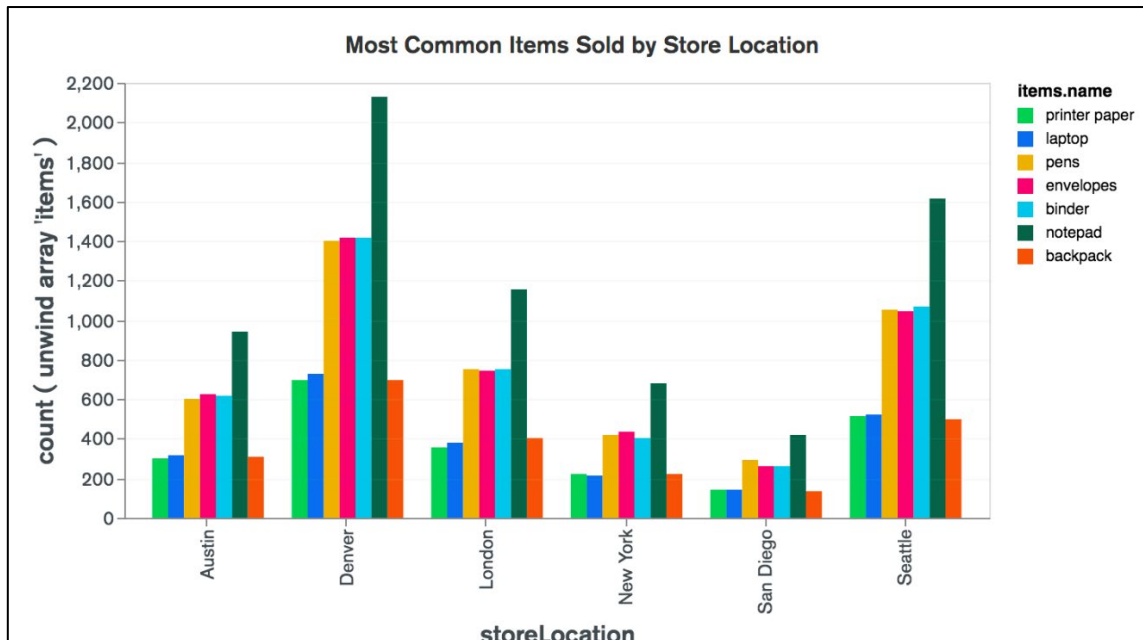
$$\text{Linear Slope} = \frac{\Delta y}{\Delta x}, \quad \Delta X = X_2 - X_1$$

เช่นในกรณีนี้ แกนตั้งคือ ความสูง, แกนนอนคือเวลา, ดังนั้น slope คือ การเปลี่ยนไปของ ความสูง (h) ต่อ การเปลี่ยนไปของ เวลา (t) หรือว่า $\frac{\Delta h}{\Delta t}$ ซึ่งมีหน่วยเป็น เมตรต่อวินาที, (อะไรล่ะ? มีหน่วยเป็นเมตรต่อวินาที)

ลองคำนวณ slope โดยประมาณจากกราฟ

1.3.3. แผนภูมิแท่ง (Bar Chart)

แผนภูมิแท่ง (Bar Chart) จะใช้เมื่อมีข้อมูลแบบแบ่งกลุ่ม และในแต่ละกลุ่มมีปริมาณที่ต้องการเปรียบเทียบ เช่น



นอกจากแผนภูมิแท่งแล้ว ยังมีแผนภูมิต่อชนิดอื่น ๆ อีก เช่น Pie Chart ฯลฯ

1.4. ไฟล์ประเภท CSV (Comma-separated Value)

ไฟล์ CSV คือไฟล์ที่เก็บ “ตาราง (Table)” ในรูปแบบของการขึ้นแต่ละค่าด้วยเครื่องหมาย comma “,” โดยสามารถอธิบายได้ในรูปแบบ “A, B, C, D, E, ..., Z,” จะสังเกตได้ว่า ค่าแต่ละค่าจะตามหลังด้วย comma เสมอ จนไปถึงตัวสุดท้าย (บางครั้งไม่ต้องใส่หลังตัวสุดท้ายก็ได้ ไม่ผิด! แต่ไม่สม่ำเสมอเท่านั้นเอง) เมื่อจบไป 1 แถวก็จะขึ้นบรรทัดใหม่เป็นการเริ่มแถวถัดไป อธิบายด้วยภาพได้ดังนี้

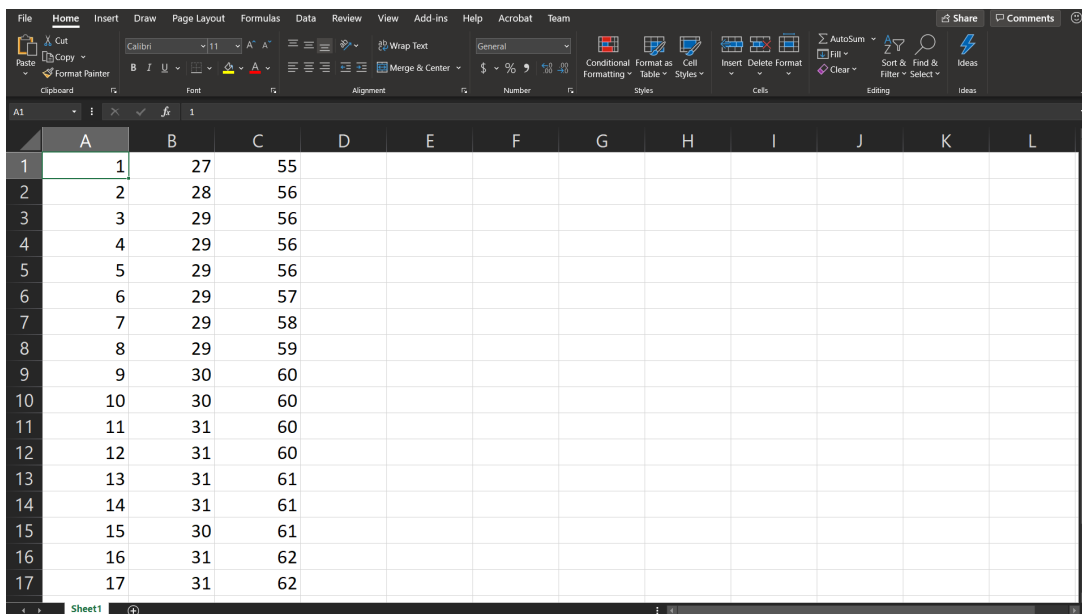
Table					CSV
N	T	RH			1,30,65,
1	30	65			2,29,63,
2	29	63			3,28,61,
3	28	61			4,27,58,
4	27	58			5,27,56,
5	27	56			

1.5. การใช้ซอฟต์แวร์ Microsoft Excel เพื่อวิเคราะห์ผลข้อมูลเชิงปริมาณ

1.5.1. General User Interface

ซอฟต์แวร์ Microsoft Excel อย่างที่บอกไปใน IV-1.3.1. เป็นที่นิยมในการประมวลผลเชิงตารางอย่างมาก, การคำนวณนั้นสามารถคำนวณได้หลากหลายรูปแบบ เช่น ค่าเฉลี่ย ส่วนเบี่ยงเบนมาตรฐาน มัธยฐาน ฯลฯ

User Interface ของ Microsoft Excel



ข้อมูลจะเรียงเป็นแนว 1, 2, 3, ... ตามแถว และ A, B, C, ... ตามหลัก จะเรียกตำแหน่งของแต่ละช่องว่า “หลัก+แถว” เช่น A1 มีค่า 1, C13 มีค่า 61, B7 มีค่า 29 โดยแต่ละตำแหน่ง (ช่อง) สามารถบรรจุค่าประเภทใดก็ได้ ทั้งจำนวนเต็ม จำนวนทศนิยม ค่า ฯลฯ

1.5.2. ชุดคำสั่ง (Functions)

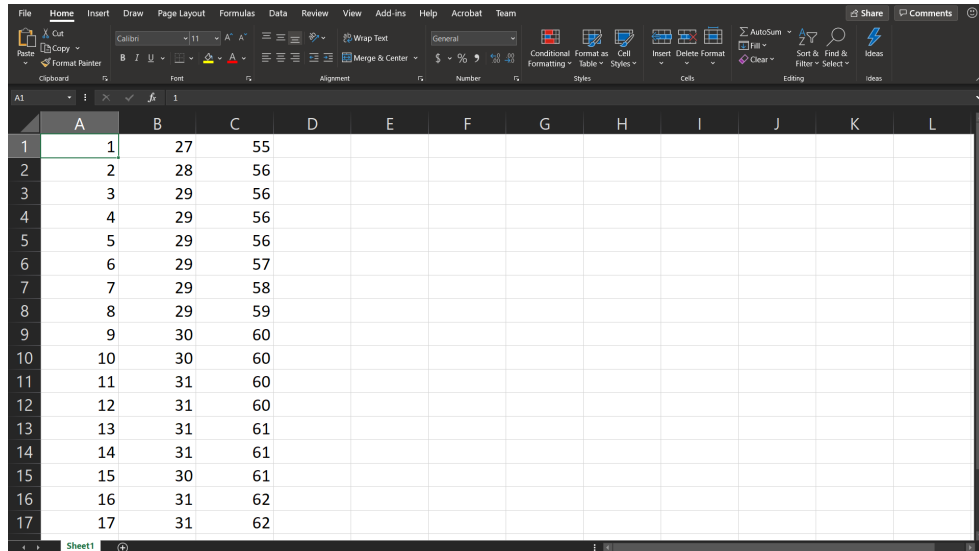
คำสั่งใน Microsoft Excel ใช้เพื่อการคำนวณโดยไม่ใช้การกดคำนวณเอง แต่ว่า ให้โปรแกรมคำนวณให้อัตโนมัติ โดยใช้ function ซึ่งจะกล่าวคร่าว ๆ ดังนี้

1. Formula (Manual Input)

- คลิกในช่องข้อมูลที่ต้องการให้ผลลัพธ์อยู่
- พิมพ์ = ลงไปแล้วตามด้วยสมการที่ต้องการ เช่น ต้องการหาผลลัพธ์ของ $1 + 2 + 3$ ก็พิมพ์ว่า $=1+2+3$ หรือ ต้องการหาผลลัพธ์ของ $(1 + 5) \div 2$ ก็พิมพ์ว่า $=(1+5)/2$
- กด Enter, แล้วผลลัพธ์ก็จะแสดงในช่องนั้น

2. Summation (ผลรวม)

ใช้คำสั่ง **=SUM(ช่องแรก:ช่องสุดท้าย)** หมายถึง บวกตัวเลขกันตั้งแต่ตัวแรกถึงตัวสุดท้าย
เช่น

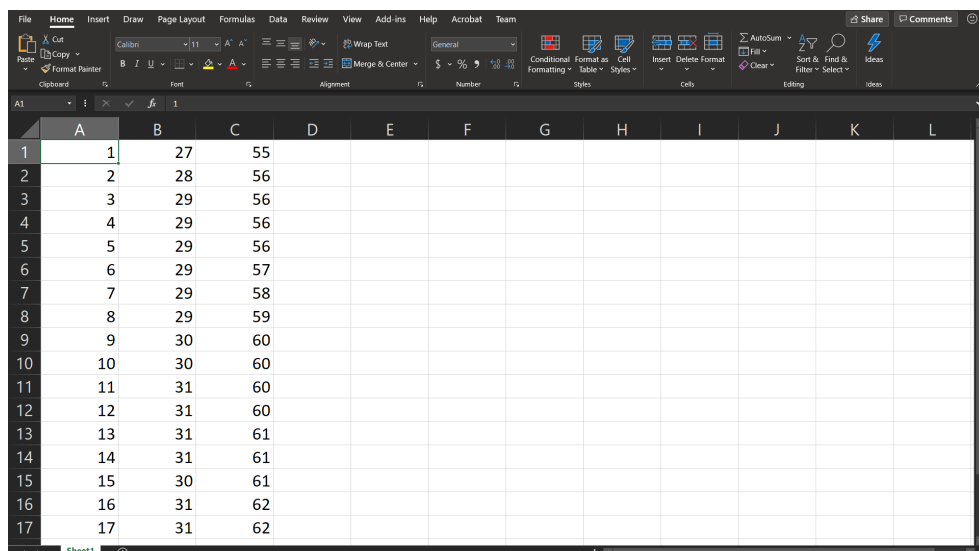


	A	B	C	D	E	F	G	H	I	J	K	L
1	1	27	55									
2	2	28	56									
3	3	29	56									
4	4	29	56									
5	5	29	56									
6	6	29	57									
7	7	29	58									
8	8	29	59									
9	9	30	60									
10	10	30	60									
11	11	31	60									
12	12	31	60									
13	13	31	61									
14	14	31	61									
15	15	30	61									
16	16	31	62									
17	17	31	62									

=SUM(A1:A17) จะได้ $1 + 2 + 3 + \dots + 16 + 17$

3. Average (ค่าเฉลี่ยเลขคณิต)

ใช้คำสั่ง **=AVERAGE(ช่องแรก:ช่องสุดท้าย)** หมายถึง บวกตัวเลขกันตั้งแต่ตัวแรกถึงตัวสุดท้ายแล้วหารด้วยจำนวนของตัวเลขในช่วงที่เลือกเพื่อหาค่าเฉลี่ย เช่น



	A	B	C	D	E	F	G	H	I	J	K	L
1	1	27	55									
2	2	28	56									
3	3	29	56									
4	4	29	56									
5	5	29	56									
6	6	29	57									
7	7	29	58									
8	8	29	59									
9	9	30	60									
10	10	30	60									
11	11	31	60									
12	12	31	60									
13	13	31	61									
14	14	31	61									
15	15	30	61									
16	16	31	62									
17	17	31	62									

=AVERAGE(A1:A17) จะได้ $\frac{1+2+3+\dots+16+17}{17}$

4. Maximum – Minimum (ค่าสูงสุด - ต่ำสุด)

ใช้คำสั่ง =MAX(ช่องแรก:ช่องสุดท้าย) เพื่อหาค่าสูงสุดของช่วง

และใช้ =MIN(ช่องแรก:ช่องสุดท้าย) เพื่อหาค่าต่ำสุดของช่วง

	A	B	C	D	E	F	G	H	I	J	K	L
1	1	27	55									
2	2	28	56									
3	3	29	56									
4	4	29	56									
5	5	29	56									
6	6	29	57									
7	7	29	58									
8	8	29	59									
9	9	30	60									
10	10	30	60									
11	11	31	60									
12	12	31	60									
13	13	31	61									
14	14	31	61									
15	15	30	61									
16	16	31	62									
17	17	31	62									

=MAX(B1:B17) จะได้ 31 ซึ่งเป็นค่าที่สูงที่สุดของช่วงที่เลือก

=MIN(B1:B17) จะได้ 27 ซึ่งเป็นค่าที่ต่ำที่สุดของช่วงที่เลือก

5. Conditional Count (นับเมื่อตรงกับเงื่อนไข)

ใช้คำสั่ง =COUNTIF(ช่องแรก:ช่องสุดท้าย,เงื่อนไข)

ข้อมูล	ข้อมูล
apples	32
oranges	54
peaches	75
apples	86

สูตร	คำอธิบาย
=COUNTIF(A2:A5,"apples")	นับจำนวนเซลล์ที่มี apples ในเซลล์ A2 ถึง A5 ผลลัพธ์คือ 2
=COUNTIF(A2:A5,A4)	นับจำนวนเซลล์ที่มี peaches (ค่าใน A4) ในเซลล์ A2 ถึง A5 ผลลัพธ์คือ 1
=COUNTIF(A2:A5,A2)+COUNTIF(A2:A5,A3)	นับจำนวนเซลล์ที่มี apples (ค่าใน A2) และ oranges (ค่าใน A3) ในเซลล์ A2 ถึง A5 ผลลัพธ์คือ 3 สูตรนี้ใช้ COUNTIF สองครั้งเพื่อระบุเกณฑ์หลายเกณฑ์ หนึ่งเกณฑ์ต่อนิพจน์
=COUNTIF(B2:B5,">55")	นับจำนวนเซลล์ที่มีค่ามากกว่า 55 ในเซลล์ B2 ถึง B5 ผลลัพธ์คือ 2
=COUNTIF(B2:B5,"<>"&B4)	นับจำนวนเซลล์ที่มีค่าไม่เท่ากับ 75 ในเซลล์ B2 ถึง B5 เครื่องหมาย และ (&) จะผสมกับตัวดำเนินการเปรียบเทียบ ไม่เท่ากับ (<>) และค่าใน B4 เพื่ออ่าน =COUNTIF(B2:B5,"<>75") ผลลัพธ์คือ 3
=COUNTIF(B2:B5,">=32")-COUNTIF(B2:B5,">85")	นับจำนวนเซลล์ที่มีค่ามากกว่า (>) หรือเท่ากับ (=) 32 และน้อยกว่า (<) หรือเท่ากับ (=) 85 ในเซลล์ B2 ถึง B5 ผลลัพธ์คือ 3
=COUNTIF(A2:A5,"*")	นับจำนวนเซลล์ที่มีข้อความใดๆ ก็ตามที่อยู่ในเซลล์ A2 ถึง A5 เครื่องหมายดอกจัน (*) จะถูกใช้เพื่อจับคู่ค่าที่ตรงกับอักขระใดๆ ผลลัพธ์คือ 4
=COUNTIF(A2:A5,"?????es")	นับจำนวนเซลล์ที่มี 7 อักขระพอดี และลงท้ายด้วยตัวอักษร "es" ในเซลล์ A2 ถึง A5 โดยเครื่องหมายคำถาม (?) ถูกใช้เป็นตัวแทนเพื่อให้ตรงกับอักขระแต่ละตัว ผลลัพธ์คือ 2

REFERENCE: Microsoft Office Support

<คำสั่ง สามารถหาข้อมูลเพิ่มเติมได้จาก Google/ Office Support>

2. ข้อมูลเชิงคุณภาพ (Qualitative Data)

ข้อมูลเชิงคุณภาพ มีความพิเศษคือ ไม่สามารถนำมาวิเคราะห์ด้วยการคำนวณโดยตรงได้ อาจมาในรูปแบบของเกณฑ์ของข้อมูล ความถี่ของข้อมูล หรือว่าลักษณะทางแผนภาพ เป็นต้น

ข้อมูลในลักษณะนี้ เรามีวิธีการวิเคราะห์ได้หลากหลายรูปแบบ โดยจะยกมาแค่บางรูปแบบเท่านั้นเพื่อความกะทัดรัด (ผู้ที่ต้องการศึกษาเพิ่มเติม สามารถศึกษาได้จาก Internet) ดังนี้

2.1. การวิเคราะห์ด้วยสายตา (Visual Interpretation)

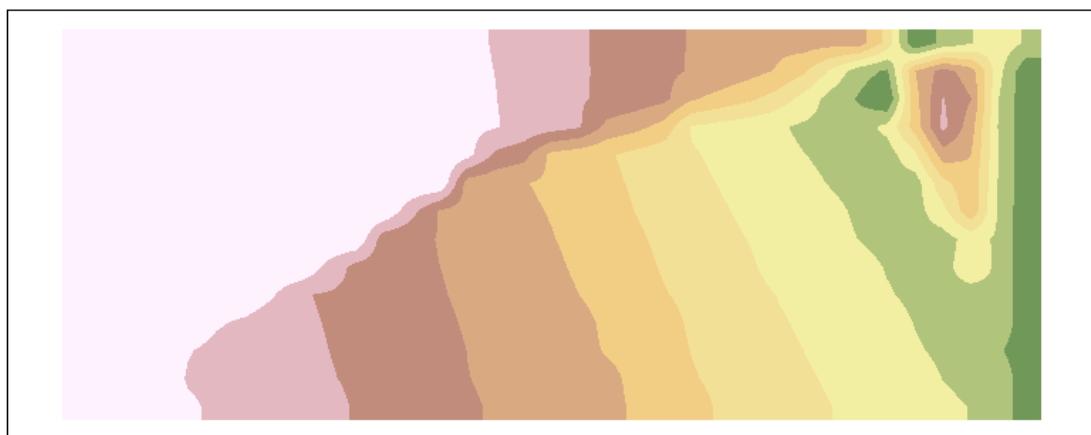
นิยมใช้สำหรับข้อมูลประเภทภาพถ่ายทางอากาศ แผนที่ แผนที่ที่พลอตพิกลงมาแล้ว เป็นต้น การวิเคราะห์ด้วยสายตานั้น เราไม่สามารถทำนายแนวโน้มของข้อมูลได้โดยตรง แต่สามารถดูลักษณะของพื้นที่หรือบริเวณที่เราต้องการศึกษาได้ว่ามีลักษณะแบบใด บริเวณใดบ้าง อีกทั้งยังสามารถใช้ตรวจสอบความเหมาะสม ความ make sense และความถูกต้องของข้อมูลได้ ว่าข้อมูลเชิงปริมาณที่เราเก็บมา หน้าเชื่อถือหรือไม่ เช่น ตั้งสถานีเก็บข้อมูลแล้วปล่อยให้มันกลิ้งตกเขามา เก็บค่าทุก ๆ 1 วินาที ปรากฏว่า ข้อมูลชุดแรก ๆ บอกระดับความสูงน้อยกว่าข้อมูลชุดหลัง ๆ ก็รู้อยู่แก่ใจว่าเป็นไปไม่ได้ เพราะตกมาจากเขา และไปหาสาเหตุของปัญหานั้นได้ เช่น เซ็นเซอร์เสีย สูตรคำนวณความสูงพลาด ฯลฯ

2.2. การแบ่งเกณฑ์ (Criteria)

เมื่อทำการเก็บข้อมูลเรียบร้อยแล้ว พบว่าต้องการแบ่งข้อมูลเป็นเกณฑ์เช่น สูง กลาง ต่ำ โดยใช้เกณฑ์การแบ่งเป็นระดับ เช่น 1-10 คือ ต่ำ, 11-20 คือ ปานกลาง, และ 21-30 คือ สูง หรือแบ่งเป็น ร้อน หนาว

การแบ่งเกณฑ์นั้น ส่วนใหญ่จะทำเมื่อต้องการวัดว่า ณ จุดที่วัดนั้น ค่าที่ได้ออกมาเป็นเท่านี้ และมีเกณฑ์ที่ได้มาตรฐาน จึงทำการแบ่งเกณฑ์เพื่อลดการสื่อความหมายด้วยตัวเลข ตัวอย่าง แผนที่ความกดอากาศด้านล่าง ใช้ระดับสีในการแบ่งเกณฑ์ว่า x_1 ถึง x_2 ให้เป็นสีหนึ่ง เพื่อสื่อความหมายให้ง่ายต่อความเข้าใจ และไล่เป็นระดับ ๆ (จากแผนที่ สามารถตีความอะไรได้บ้าง?)

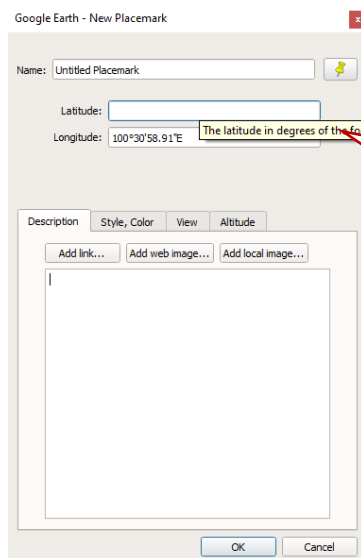
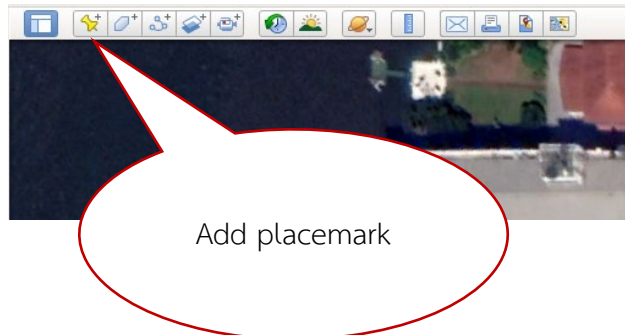
แผนที่ความกดอากาศ 1:6,000



406.9411621 - 490.0593397	656.2956951 - 739.4138726	905.650228 - 988.7684055
490.0593398 - 573.1775174	739.4138727 - 822.5320502	988.7684056 - 1,071.886583
573.1775175 - 656.295695	822.5320503 - 905.6502279	1,071.886584 - 1,155.004761

2.3. การใช้ซอฟต์แวร์ Google Earth Pro เพื่อพลอตพิกัดลงแผนที่

2.3.1. พลอตจากค่าพิกัด

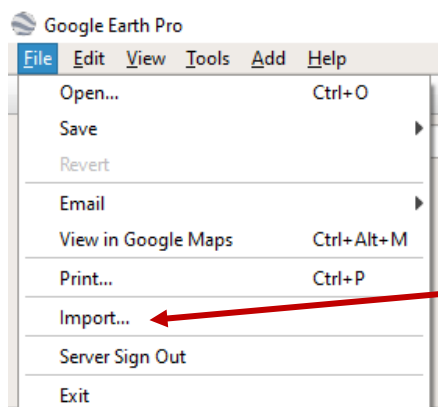


พิกัด Latitude/Longitude

ทศนิยม 6 ตำแหน่ง เช่น

13.023786

2.3.2. พลอตจากไฟล์ CSV/TXT



CanSat Missions

ข้อกำหนด CanSat

1. น้ำหนักไม่เกิน 250 g
2. ไม่จำกัดรูปร่างและขนาด
3. ปลอยจากความสูง 200 ~ 300 m
4. มี terminal velocity ประมาณ 5 ~ 10 m/s
5. สามารถขอลอยเพิ่มได้เพียง 1 ครั้ง (มีโควตาลอย 2 ครั้งต่อทีม)
6. ส่งข้อมูลกลับในรูปแบบดังนี้

<รหัสทีม>,<ข้อมูลชุดที่...>,<T>,<RH>,<P>,<Lat>,<Lon>

ภารกิจหลัก (Primary Missions)

1. สามารถเก็บค่า Temperature (T), Humidity (RH), และ Pressure (P) จาก BME280 ได้
2. สามารถรับค่าพิกัดจาก GPS (Lat), (Lon) ได้
3. สามารถรับสัญญาณและข้อมูลจาก CanSat สู่ Ground Station ได้
4. ทำการกู้ CanSat กลับมาได้ (ความสมบูรณ์มีผลต่อแต้ม)

ภารกิจรอง (Secondary Mission)

อาจมีหรือไม่มีก็ได้ ภารกิจรองอาจเป็นการนำค่าที่เก็บได้มาประมวลผล นำเสนอ วิเคราะห์ผล ตีความ ข้อมูล หาความสัมพันธ์ หรือพิสูจน์ทฤษฎีก็ได้ หรือทำอย่างอื่นเพิ่มเติมบนตัว CanSat ก็ได้

เพิ่มเติม

นำผลสรุปมานำเสนอ (Presentation) ในวันที่สี่เวลา 19.00 ~ 20.00 น. (ทีมละประมาณ 5 ~ 10 นาที)

EXTRA NOTE

References

ศึกษาข้อมูลเกี่ยวกับ CanSat เพิ่มเติม

http://unisec.jp/library/i-cansat/manual_CanSat_textbook_eng_v5.pdf

ศูนย์วิทยาศาสตร์โลกและอวกาศ (LESA). (2550). *ดาวเทียม*. สืบค้นเมื่อวันที่ 27 เมษายน 2562.
จาก <http://www.lesa.biz/>

เฝ้าไท่ สีนอำพน. (2561). ภูมิศาสตร์มนุษย์ในกระแสโลกาภิวัตน์. คณะสังคมศาสตร์
ภาควิชาภูมิศาสตร์ มหาวิทยาลัยเชียงใหม่. สำนักพิมพ์ มหาวิทยาลัยเชียงใหม่. เชียงใหม่.

ไทยคม (Thaicom). (2562). *ข้อมูลเกี่ยวกับบริษัทไทยคม จำกัด (มหาชน)*. สืบค้นเมื่อวันที่ 27
เมษายน 2562

