การพัฒนาระบบจำลอง เรดาร์ เพื่อการควบคุมจราจรทางอากาศ

นาย เมธี เสรีอรุโณ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิทยานิพนธ์นี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

ภาควิชาวิศวกรรมคอมพิว เตอร์

บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย

พ.ศ. 2537

ISBN 974-584-291-5

ลิขสิทธิ์ของบัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย

DEVELOPMENT OF A RADAR SIMULATOR FOR AIR TRAFFIC CONTROL

MR. MATEE SEREARUNO

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

A Thesis Submitted in Partial Fulfillment of the Requirements

for the Degree of Master of Science

Department of Computer Engineering

Graduate School

Chulalongkorn University

1994

ISBN 974-584-291-5

หัวข้อวิทยานิพนธ์	การพัฒนาระบบจำลองเรดาร์เพื่อการควบคุมจราจรทางอากาศ
โดย	นายเมธี เสรีอรุโณ
ภาควิชา	วิศวกรรมคอมพิวเตอร์
อาจารย์ที่ปรึกษา	อาจารย์ ดร. สืบสกุล พิภพมงคล

บัณฑิตวิทยาลัย จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญามหาบัณฑิต

..... คณบดีบัณฑิตวิทยาลัย

(ศาสตราจารย์ ดร. ถาวร วัชราภัย)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ

(อาจารย์ ดร. ฐิต ศิริบูรณ์)

..... อาจารย์ที่ปรึกษา

(อาจารย์ ดร. สืบสกุล พิภพมงคล)

..... กรรมการ

(อาจารย์ ดร. สมชาย ประสิทธิ์จูตระกูล)

..... กรรมการ

(อาจารย์ นงลักษณ์ โควาวิสารัช)

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

พิมพ์ต้นฉบับบทคัดย่อวิทยานิพนธ์ภายในกรอบสีเขียวนี้เพียงแผ่นเดียว

เมธี เสรีอรุโณ : การพัฒนาระบบจำลองเรดาร์เพื่อการควบคุมจราจรทางอากาศ (DEVELOPMENT OF A RADAR SIMULATOR FOR AIR TRAFFIC CONTROL) อ.ที่ปรึกษา: ดร.สืบสกุล พิภพมงคล, 101 หน้า. ISBN 974-584-291-5

การฝึกหัดในภาคปฏิบัติ ของผู้ควบคุมจราจรทางอากาศของประเทศไทยในปัจจุบันสามารถทำการฝึกได้แต่เฉพาะการควบคุมแบบไม่ใช้เรดาร์เท่านั้น ทั้งนี้ เนื่องจากอุปกรณ์เรดาร์จำลองมีราคาแพง อีกทั้ง ระบบเรดาร์ที่ใช้งานจริงจะถูกใช้งานตลอด 24 ชั่วโมง ทำให้ไม่สามารถใช้ในการฝึกได้ บทวิทยานิพนธ์นี้ได้เสนอระบบฝึกหัดการควบคุมจราจรทางอากาศที่สามารถจำลองข้อมูลของเรดาร์ ระบบนี้ประกอบด้วยเครื่องคอมพิวเตอร์ส่วนบุคคล 2 เครื่อง โดยเครื่องหนึ่งทำหน้าที่จำลองข้อมูลการบินของเครื่องบินเพื่อให้ได้ข้อมูลเช่นเดียวกับที่มีในระบบเรดาร์จริง เครื่องนี้จะถูกควบคุมโดยครูผู้ฝึกซึ่งทำหน้าที่เป็นนักบินสมมุติ ข้อมูลที่จำลองขึ้นจะถูกส่งให้เครื่องคอมพิวเตอร์อีกเครื่องหนึ่ง ผ่านทางช่องทางสื่อสารแบบอนุกรม เพื่อทำหน้าที่แสดงผลบนหน้าจอในลักษณะเดียวกับจอเรดาร์ที่ใช้งานจริง โดยผู้ที่รับการฝึกจะใช้ข่าวสารที่ได้รับจากจอภาพของเครื่องคอมพิวเตอร์นี้ประกอบการตัดสินใจเพื่อฝึกหัดการแนะนำลักษณะการบินให้กับนักบิน ระบบนี้มีข้อดีกว่าระบบเรดาร์จริงคือสามารถจำลองเหตุการณ์ฉุกเฉิน ฝึกบทเรียนซ้ำ ๆ กัน ฯลฯ ซึ่งเป็นประโยชน์อย่างมากในการฝึกสอน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา 2536 ลายมือชื่ออาจารย์ที่ปรึกษาร่วม .....

C417555 : MAJOR COMPUTER SCIENCE
KEY WORD: RADAR/ SIMULATION/ TRAINING
MATEE SEREARUNO : DEVELOPMENT OF A RADAR SIMULATOR FOR AIR TRAFFIC CONTROL. THESIS ADVISOR : SUEBSKUL PHIPHOBMONGKOL, Ph.D. 101 pp. ISBN 974-584-291-5

Air Traffic Control Practical Training in Thailand, at the present time, can be done only with non-radar control because radar simulation equipments are still expensive. Furthermore, the real radar systems are operated 24 hours a day and cannot be used for this training. This thesis presents an Air Traffic Control Training System that can simulate the radar information. The system consists of 2 personal computers. One computer performs the simulation of the aircraft data which is similar to the radar data. The simulated data will be transmitted to another computer via a serial communication port and used to construct the screen similar to the real radar screen. This system has more advantages than the real radar system, for examples, the simulation of emergency situation and the repeated exercise execution, which are very useful for the training.

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ภาควิชา วิศวกรรมคอมพิวเตอร์ ลายมือชื่อนิสิต .....
สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่ออาจารย์ที่ปรึกษา .....
ปีการศึกษา 2536 ลายมือชื่ออาจารย์ที่ปรึกษาร่วม .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความช่วยเหลือทางด้านข้อมูล เอกสาร และคำแนะนำอย่างดียิ่งของคุณเสริมศักดิ์ วิทยาพูน อดีตหัวหน้าเวรผู้ควบคุมจราจรทางอากาศ ซึ่งปัจจุบันดำรงตำแหน่งหัวหน้างานฝึกอบรม ศูนย์ควบคุมจราจรทางอากาศ บริษัทวิทยุการบินแห่งประเทศไทย จำกัด ซึ่งได้ให้ข้อคิดเห็นต่างๆ รวมถึงตอบข้อสงสัยเกี่ยวกับการควบคุมจราจรทางอากาศเป็นอย่างดีและพร้อมกันนี้ขอขอบพระคุณอาจารย์ ดร. สืบสกุล พิภพมงคล อาจารย์ที่ปรึกษาวิทยานิพนธ์ ซึ่งท่านได้ให้คำปรึกษา ตรวจทาน และให้ข้อคิดเห็นในการจัดทำวิทยานิพนธ์มาโดยตลอด รวมถึงบริษัทวิทยุการบินแห่งประเทศไทยที่ได้สนับสนุนทุนการศึกษา และอุปกรณ์เครื่องมือที่ใช้ในการวิจัยครั้งนี้ ท้ายนี้ผู้วิจัยใคร่ขอกราบขอบพระคุณบิดามารดา ซึ่งได้สนับสนุนการศึกษามาโดยตลอดและคุณศศิธร เสรีอรุโณ ที่เป็นผู้ช่วยในการจัดพิมพ์วิทยานิพนธ์นี้จนสำเร็จลุล่วงไปด้วยดี

เมธี เสรีอรุโณ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญ

หน้า

บทคัดย่อไทย..... ง

บทคัดย่ออังกฤษ..... จ

กิตติกรรมประกาศ..... ฉ

สารบัญตาราง..... ฌ

สารบัญภาพ..... ญ

บทที่

1. บทนำ..... 1

- ความเป็นมาและความสำคัญของปัญหา..... 1
- แนวทางแก้ไข..... 2
- วัตถุประสงค์..... 3
- ขอบเขตในการวิจัย..... 4
- ขั้นตอนและวิธีการดำเนิน..... 4
- ประโยชน์ที่คาดว่าจะได้รับ..... 5

2. การควบคุมจราจรทางอากาศ..... 6

- เครื่องมือที่ใช้ในการควบคุมจราจรทางอากาศ..... 8
- ระบบเรดาร์ควบคุมจราจรทางอากาศ..... 10
- ขั้นตอนการปฏิบัติในการทำการบิน..... 12
- การฝึกหัดผู้ควบคุมจราจรทางอากาศ..... 14
- การนำเครื่องคอมพิวเตอร์มาช่วยในการฝึก..... 16

3. การจัดเก็บแฟ้มข้อมูลที่ใช้ในการเดินอากาศ..... 20

- การแปลงข้อมูลทางภูมิศาสตร์เป็นตำแหน่งบนหน้าจอ..... 20

4. โปรแกรมจำลองข้อมูลการบิน..... 34

- การทำงานของนักบิน..... 34

- การออกแบบโปรแกรมจำลองข้อมูลการบิน..... 37
- การจำลองการทำงานของ เครื่องบิน..... 42
- การคำนวณตำแหน่งของ เครื่องบิน..... 43
- การแสดงผลการทำงาน..... 52
- การรับคำสั่งควบคุมการบิน..... 54
- การส่งข้อมูลไปยังจอแสดงผล..... 58
- การทำงานของโปรแกรมหลัก..... 59
- การรับส่งข้อมูลผ่านพอร์ทอนุกรม..... 61

5. โปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์..... 64
- คำสั่งสำหรับจอแสดงผล..... 68
- โครงสร้างทางด้านฮาร์ดแวร์..... 73
- การออกแบบระบบซอฟท์แวร์..... 76
- โครงสร้างโปรแกรมในส่วนการแสดงผลข้อมูลการบินด้วย เรดาร์..... 77
- ผังการทำงานของโปรแกรม..... 81

6. สรุปผลและ เสนอแนะ..... 92
- วิธีการทดสอบการทำงาน..... 92
- ผลการทดสอบการใช้งาน..... 93
- สรุปผลการทดสอบ..... 95
- ข้อ เสนอแนะในการพัฒนาระบบต่อไป..... 96
- สรุปผลการทำวิทยานิพนธ์..... 97

บรรณานุกรม..... 99

ประวัติผู้ เขียน..... 101

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

สารบัญภาพ

หน้า

รูปที่ 1 ห้องปฏิบัติการควบคุมจราจรทางอากาศ..... 9

รูปที่ 2 จอแสดงผล เรดาร์ที่ใช้ในการควบคุมจราจรทางอากาศ..... 11

รูปที่ 3 โครงสร้างของระบบจำลองข้อมูล เรดาร์..... 17

รูปที่ 4 แผนที่ เส้นทางการบินใน เขตควบคุมการบินกรุง เทพฯ..... 19

รูปที่ 5 โปรแกรมจำลองข้อมูลการบิน..... 36

รูปที่ 6 การคำนวณตำแหน่งของ เครื่องบิน..... 43

รูปที่ 7 อิทธิพลของกระแสลมที่มีต่อ เครื่องบิน..... 44

รูปที่ 8 หน้าจอของโปรแกรมนักบินสมมุติ..... 52

รูปที่ 9 ผังการทำงานของโปรแกรมจำลองข้อมูลการบิน..... 60

รูปที่ 10 หน้าจอของโปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์..... 66

รูปที่ 11 สัญญลักษณ์ของ เครื่องบินที่ปรากฏบนจอ เรดาร์..... 67

รูปที่ 12 เส้นสำหรับวัดระยะทางและมุม..... 70

รูปที่ 13 กราฟิกการ์ดที่มีหน่วยประมวลผลช่วยทางด้านกราฟิก..... 75

รูปที่ 14 โปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์..... 77

รูปที่ 15 ผังการทำงานของโปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์..... 80

รูปที่ 16 สัญญลักษณ์ของ เครื่องบินที่ไม่ได้รับข้อมูลใหม่..... 87

รูปที่ 17 การรับข้อมูลจากนักบินสมมุติมาประมวลผล..... 90

สารบัญตาราง

หน้า

ตารางที่ 1 โครงสร้างแฟ้มข้อมูล เส้นแสดงขอบ เขตควบคุมการบิน..... 22

ตารางที่ 2 โครงสร้างแฟ้มข้อมูลแสดงจุดรายงานการบิน..... 23

ตารางที่ 3 โครงสร้างแฟ้มข้อมูล เส้นทางการบิน..... 24

ตารางที่ 4 โครงสร้างแฟ้มข้อมูล เส้นขอบ เขตสนามบิน..... 25

ตารางที่ 5 โครงสร้างแฟ้มข้อมูลแบบฝึกหัดการบิน..... 27

ตารางที่ 6 โครงสร้างแฟ้มข้อมูลสมรรถนะการบินของ เครื่องบิน..... 29

ตารางที่ 7 โคร่งสร้างแฟ้มข้อมูลสมรรถนะการบินแบ่งตามกลุ่ม..... 30

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 1

บทนำ

ความเป็นมาและความสำคัญของปัญหา

การคมนาคมทางอากาศในปัจจุบันได้ก้าวหน้าไปมาก สังเกตได้จากปริมาณเที่ยวบินที่ได้เพิ่มสูงขึ้นทุกๆปี * ดังนั้นงานให้บริการการควบคุมการจราจรทางอากาศ (Air Traffic Control หรือเรียกย่อว่า ATC) ก็จะยิ่งทวีความสำคัญมากขึ้นตามไปด้วย เพราะเมื่อปริมาณเที่ยวบินสูงขึ้นจะมีผลให้การจราจรทางอากาศมีความหนาแน่นมากขึ้นในแต่ละเส้นทางบิน ดังนั้นผู้ควบคุมการจราจรทางอากาศ (Air Traffic Controller) จะต้องมีความสามารถในการควบคุมจราจรทางอากาศได้เป็นอย่างดี สามารถจัดการควบคุมการจราจรทางอากาศได้ตามระเบียบแบบแผนของมาตรฐานสากล อีกทั้งยังต้องมีสุขภาพทั้งทางด้านร่างกายและจิตใจดี เพื่อให้สามารถปฏิบัติหน้าที่ได้อย่างมีประสิทธิภาพอยู่ตลอดเวลา

เนื่องจากการคมนาคมทางอากาศดำเนินอยู่ตลอด 24 ชั่วโมง ดังนั้นการทำงานของพนักงานควบคุมการจราจรทางอากาศก็ยิ่งมีความสำคัญ เนื่องจากเป็นงานที่ต้องมีความรอบคอบอย่างสูงอยู่ทุกขณะ ความผิดพลาดในการทำงานอาจก่อให้เกิดอุบัติภัยทางอากาศอย่างร้ายแรงทั้งต่อชีวิตและทรัพย์สิน สิ่งหนึ่งที่จะช่วยเพิ่มความปลอดภัยได้ก็คือการมีผู้ควบคุมการจราจราจรทางอากาศที่มีคุณภาพในปริมาณที่เพียงพอต่อการทำงานที่ไม่หนักจนเกินไปในแต่ละวัน ซึ่งในปัจจุบันมี

---

* ในปี 2535 มีปริมาณเที่ยวบินสูงถึง 254,820 เที่ยวบิน เทียบกับปี 2534 ซึ่งมีเพียง 224,471 เที่ยวบิน เพิ่มขึ้นถึง 30,349 เที่ยวบิน คิดเป็นร้อยละ 13.52 เฉลี่ยในแต่ละวันจะมีปริมาณเที่ยวบินสูงถึงวันละเกือบ 700 เที่ยวบินต่อวัน

พนักงานที่สามารถทำงานในตำแหน่งพนักงานควบคุมจราจรทางอากาศประมาณ 50 คน เท่านั้น ซึ่งไม่เพียงพอในการรับหน้าที่ในปัจจุบัน อีกทั้งภาวะการเจริญเติบโตทางเศรษฐกิจของประเทศไทย มีอัตราเติบโตค่อนข้างสูงทำให้ปริมาณเที่ยวบินเพิ่มมากขึ้นทุกปี จึงส่งผลให้การควบคุมจราจรทางอากาศอย่างมีประสิทธิภาพและปลอดภัย ทำได้ลำบากขึ้น

การฝึกหัดผู้ควบคุมจราจรทางอากาศในแต่ละรุ่นจะมีผู้ผ่านการฝึกเพียงไม่กี่คน เท่านั้น ทั้งยังต้องไปฝึกหาประสบการณ์ในการทำงานกับระบบจริงอีกเป็นเวลาหลายปี กว่าจะขึ้นมารับหน้าที่ได้ นอกจากปัญหาในด้านจำนวนของผู้ผ่านการฝึกมีน้อยแล้ว ปัญหาที่สำคัญมาอีกประการหนึ่งก็คือ <u>การขาดแคลนเครื่องมือที่ใช้ในการฝึก</u> โดยเฉพาะเครื่องจำลองเรดาร์ (Radar Simulator) ซึ่งเป็นการจำลองการทำงานของระบบเรดาร์ควบคุมจราจรทางอากาศที่คล้ายกับระบบเรดาร์ที่ใช้งานจริง เพื่อให้ผู้รับการฝึกมีความชำนาญในการทำงานกับเรดาร์ แต่เนื่องจากระบบนี้มีราคาค่อนข้างสูง จึงยังมิได้จัดหาไว้ใช้ในปัจจุบัน

<u>แนวทางแก้ไข</u>

จากปัญหาดังกล่าวทางบริษัทวิทยุการบินฯ จึงมีความต้องการจะพัฒนาระบบโปรแกรมเพื่อใช้ในการจำลองการทำงานของระบบเรดาร์ โดยมีจุดมุ่งหมายหลักเพื่อใช้ในการฝึกหัดการควบคุมจราจรทางอากาศด้วยเรดาร์ ทั้งนี้เนื่องจากในการฝึกหัดพนักงานควบคุมฯใหม่นั้นจะไม่สามารถทำการฝึกหัดกับระบบที่ใช้งานจริงโดยตรงได้ เพราะนอกจากระบบจริงจะถูกใช้งานตลอด 24 ชั่วโมงแล้ว ยังเป็นการรบกวนการทำงานของพนักงานควบคุมการจราจรทางอากาศคนอื่นๆซึ่งส่งผลกระทบต่อการทำงานของระบบอันเป็นการเสี่ยงต่อการควบคุมจราจรทางอากาศ การฝึกสอนด้วยเครื่องจำลองเรดาร์นี้ จะทำให้ผู้เข้ารับการฝึกมีความชำนาญในการทำงานกับจอเรดาร์ที่คล้ายกับระบบที่ใช้งานจริง อันจะส่งผลดีต่อการเข้ารับการฝึกขั้นสูงต่อไป โปรแกรมจำลองเรดาร์ต้นแบบที่ถูกพัฒนาขึ้นนี้เป็นระบบที่จำลองการทำงานของระบบเรดาร์ควบคุมจราจรทางอากาศที่ใช้งานอยู่จริงโดยได้รวบรวมความสามารถพื้นฐานที่จำเป็นต่างๆมาประยุกต์เข้ากับความต้องการในการใช้งานโดยคำนึงถึงความเป็นไปได้ทางเทคโนโลยีที่มีอยู่ เพื่อที่จะสามารถจัดหาอุปกรณ์เครื่องมือต่างๆได้โดยง่าย และมีราคาต่ำกว่าการซื้อจากต่างประเทศ

ซอฟต์แวร์จำลอง เรดาร์ เพื่อการฝึกหัดการควบคุมจราจรทางอากาศนี้ ประกอบด้วยโปรแกรม 2 ส่วน ซึ่งจะทำงานบน เครื่องคอมพิว เตอร์ 2 เครื่อง ที่ตั้งอยู่ห่างจากกันและ เชื่อมโยงข้อมูลระหว่างกันด้วยสายสื่อสารแบบอนุกรม โดย เครื่องคอมพิว เตอร์ เครื่องหนึ่ง เรียกว่า "ผู้รับการฝึก" (ATC Trainee) ทำหน้าที่รับข้อมูลการบินมาแสดงผลบนจอภาพในแบบกราฟิก เพื่อ เลียนแบบการแสดงผลของระบบ เรดาร์ที่ใช้งานจริง โดยจะรับข้อมูลจากคอมพิว เตอร์อีกเครื่องหนึ่ง เรียกว่า "นักบินสมมุติ" (Pseudo Pilot) ซึ่งทำหน้าที่สร้างข้อมูลของ เรดาร์ในส่วนของ เครื่องบินให้บินด้วยรูปแบบที่ต้องในการฝึก เช่น การไต่หรือลดระดับความสูง การ เลี้ยว การ เพิ่มหรือลดความ เร็วของ เครื่องบิน ฯลฯ ทั้งนี้ เพื่อ เป็นการสร้างสถานการณ์ให้ผู้รับการฝึกได้ฝึกหัดการควบคุมจราจรทางอากาศในสถานการณ์แบบต่างได้

โปรแกรมต้นแบบนี้ สามารถแสดงผลการจำลอง เรดาร์ในด้านรูปแบบและสัญญลักษณ์ เช่น เดียวกับระบบ เรดาร์ที่ใช้งานจริงแต่ทำงานโดย เป็นอิสระ นอกจากนี้ยังสามารถอ่านข้อมูลของแบบฝึกหัดการบินจากที่ เก็บไว้มา เป็นข้อมูล เริ่มต้นในการจำลอง ระบบนี้จะคำนวณตำแหน่งของ เครื่องบินและข้อมูลอื่นๆที่ เกี่ยวข้อง โดยอาศัยข้อมูลสมรรถนะการบินของ เครื่องบินแต่ละรุ่นที่ได้ เก็บไว้ในฐานข้อมูล ทำให้สามารถ เปลี่ยนแปลงหรือแก้ไขรูปแบบการฝึกได้โดยง่าย อีกทั้งสามารถ เข้าทำการควบคุมในส่วนนักบินจำลองด้วยการป้อนคำสั่งผ่านแป้นพิมพ์ เช่น ให้ เครื่องบิน เลี้ยวออกนอก เส้นทางที่ได้กำหนดไว้ เพื่อ เป็นการสร้างสถานการณ์ให้ผู้รับการฝึกได้ฝึกหัดการควบคุมการจราจรทางอากาศในสถานการณ์ที่หลากหลาย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

<u>วัตถุประสงค์</u>

1. เพื่อจัดสร้างโปรแกรมต้นแบบ เพื่อจำลอง เรดาร์ที่ใช้ในการฝึกการควบคุมจราจรทางอากาศ ที่สามารถทำงานในลักษณะ เช่น เดียวกับระบบ เรดาร์ที่ใช้งานจริง
2. ศึกษาการทำงานและการใช้งานของระบบคอมพิว เตอร์กราฟิกความละ เอียดสูง
3. ศึกษาและออกแบบวิธีการจัด เก็บข้อมูลทางด้านภูมิศาสตร์
4. ศึกษาและออกแบบการจัด เก็บข้อมูลสมรรถนะการบินของ เครื่องบิน
5. ศึกษาและออกแบบการส่งผ่านข้อมูลระหว่างคอมพิว เตอร์ผ่านช่องสื่อสารอนุกรม

ขอบเขตในการวิจัย

1. พัฒนาโปรแกรมบนเครื่องคอมพิวเตอร์ส่วนบุคคลที่ใช้หน่วยประมวลผลกลางเบอร์ 80x86 หรือเทียบเท่า โดยใช้หน่วยแสดงผลทางด้านกราฟิกพร้อมจอแสดงผลชนิดความละเอียดสูง 1280x1024 จุดภาพ

2. การเชื่อมโยงข้อมูล จะใช้ผ่านพอร์ทอนุกรมมาตรฐานของเครื่องคอมพิวเตอร์

3. สามารถจำลองการแสดงผลเลียนแบบจอเรดาร์ที่ใช้งานจริง*

ขั้นตอนและวิธีการดำเนินการ

1. ศึกษาการทำงานและเลือกใช้อุปกรณ์ฮาร์ดแวร์ และซอฟต์แวร์ที่เกี่ยวข้องรวมถึงอุปกรณ์แสดงผลทางด้านกราฟิกที่ช่วยในการแสดงผลด้วยความละเอียดสูง

2. ศึกษาการทำงานของระบบแสดงผลเรดาร์ที่ใช้งานจริง และออกแบบระบบซอฟต์แวร์เบื้องต้น

3. พัฒนาโปรแกรมเพื่อเก็บข้อมูลทางภูมิศาสตร์การบินและการแสดงผลทางจอภาพ

4. พัฒนาโปรแกรมแสดงสัญญลักษณ์และข้อมูลต่างๆ เพื่อแทนตำแหน่งและการเคลื่อนที่ของเครื่องบินบนจอแสดงผลเรดาร์

5. พัฒนาโปรแกรม เพื่อจำลองข้อมูลที่เกี่ยวข้องกับการเคลื่อนที่ของเครื่องบินตามการควบคุมของนักบินสมมุติ และการควบคุมโดยอัตโนมัติตามแผนการบินที่ได้กำหนดไว้

6. ทดสอบระบบโดยรวม แก้ไขข้อบกพร่อง เพื่อเพิ่มประสิทธิภาพในการทำงาน

7. สรุปผลการวิจัย และข้อเสนอแนะ

---

* ปัจจุบันใช้ระบบแสดงผลเรดาร์รุ่น DDS-80 และ CDS-80 จากประเทศอิตาลี ซึ่งทั้งสองรุ่นนี้มีคุณสมบัติในด้านการใช้งานเช่นเดียวกัน ต่างกันตรงที่จอรุ่น DDS-80 จะเป็นจอแบบเวคเตอร์สแกน และจอ CDS-80 จะเป็นจอสีแบบราสเตอร์สแกน

ประโยชน์ที่คาดว่าจะได้รับ

1. ใช้ในการฝึกพนักงานควบคุมจราจรทางอากาศ ให้มีความชำนาญในการทำงานกับระบบ เรดาร์ เป็นการทดสอบความพร้อมของผู้รับการฝึกก่อนที่จะต้องไปปฏิบัติงานกับระบบที่ใช้งานจริงต่อไป

2. สามารถนำมาใช้แสดงการทำงานของระบบควบคุมจราจรทางอากาศด้วย เรดาร์ แก่ผู้สนใจ เนื่องจากสถานที่ที่ติดตั้งระบบที่ใช้งานจริง เป็น เขตหวงห้ามไม่สามารถให้บุคคลทั่วไปเข้าชมได้ เพราะจะ เป็นการรบกวนการทำงานของ เจ้าหน้าที่

3. ประหยัด เงินตราต่างประ เทศ ทดแทนการนำ เข้าอุปกรณ์จากต่างประ เทศ ซึ่งมีราคาแพง อีกทั้ง เมื่อสามารถพัฒนาจนสามารถใช้งานได้ เป็นอย่างดีแล้ว ก็จะสามารถนำออกจำหน่าย หรือให้ความช่วย เหลือกิจการบินแก่ประ เทศ เพื่อนบ้าน ความช่วย เหลือนี้จะ เป็นผลดีต่อประ เทศไทยในอนาคต โดยจะสนับสนุนการ เป็นศูนย์กลางการบินใน เขตตะวันออก เฉียงใต้

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 2

การควบคุมจราจรทางอากาศ

การควบคุมจราจรทางอากาศ เป็นการให้บริการคำแนะนำในการเดินอากาศเกี่ยวกับเส้นทางการบิน ความปลอดภัยในการบิน รวมถึงการบินขึ้นลงที่สนามบินต่างๆ นอกจากนี้ยังมีบริการเกี่ยวเนื่องต่างๆอีก เช่น การบริการข่าวแผนการบิน (Flight Plan) บริการข่าวอากาศ (Meteorological,MET) ประกาศสำหรับนักบิน (Notice to Airmen,NOTAM) เพื่อให้การเดินทางทางอากาศมีความสะดวกและปลอดภัย

องค์การการบินพลเรือนระหว่างประเทศ (International Civil Aviation Organization)* ได้แบ่งน่านฟ้าของโลกให้แต่ละประเทศทำการควบคุมการบินในเขตน่านฟ้าประเทศของตนเองเรียกว่า <u>เขตควบคุมการบิน</u> (Flight Information Region)** ทั้งน่านฟ้าบนพื้นดินและในทะเล นอกจากนี้ในบางบริเวณที่มีอาณาเขตกว้างขวาง เช่น น่านน้ำ

---

* เรียกย่อว่า ICAO เป็นองค์กรกลางระหว่างประเทศในการกำหนดมาตรฐานการควบคุมการจราจรทางอากาศให้มีรูปแบบเดียวกันทั่วโลก เพื่อให้เกิดความสะดวกทางปฏิบัติในกิจการการเดินอากาศ โดยกำหนดให้แต่ละประเทศจัดตั้งหน่วยงานที่ให้บริการการควบคุมจราจรทางอากาศซึ่งดำเนินกิจการแบบไม่มุ่งหากำไร เพื่อให้ปฏิบัติต่อสายการบินต่างๆอย่างเท่าเทียมกันโดยไม่จำกัดสัญชาติ สำหรับประเทศไทยมีหน่วยงานที่ทำหน้าที่นี้อยู่คือ บริษัทวิทยุการบินแห่งประเทศไทยจำกัด มีฐานะเป็นรัฐวิสาหกิจ สังกัดกระทรวงคมนาคม โดยมีรัฐบาลไทยเป็นผู้ถือหุ้นใหญ่ และสายการบินต่างๆ เป็นผู้ถือหุ้นรายย่อย

** เรียกย่อๆว่า " เขต FIR"

สากล ก็จะมอบหมายให้บางประเทศที่มีศักยภาพในการควบคุมจราจรทางอากาศอย่างเพียงพอทำการควบคุมจราจรทางอากาศเพิ่มเติม เรียกว่า เขตพื้นที่รับผิดชอบเพิ่มเติม (Area Of Responsibility)* โดยที่ข้อมูลรายละเอียดเกี่ยวกับการเดินอากาศทั้งหมดจะถูกประกาศเพื่อใช้งานเป็นสากลในหนังสือที่ชื่อว่า Aeronautical Information Publication** โดยจะมีการกำหนดเส้นทางการบิน (Airways) เพื่อใช้ทำการบินสู่จุดหมายปลายทางซึ่งแต่ละเส้นทางประกอบด้วยจุดรายงานการบิน (Reporting point) อยู่เป็นช่วงๆ*** เมื่อนักบินทำการบินถึงแต่ละจุดรายงานการบิน ก็จะเรียกทางวิทยุมายังศูนย์ควบคุมจราจรทางอากาศเพื่อเป็นการยืนยันตำแหน่งและรับทราบข้อมูลอื่นๆ เพิ่มเติม

ศูนย์ควบคุมจราจรทางอากาศจึงเป็นศูนย์กลางในการติดต่อกับเครื่องบินทุกลำที่กำลังทำการบินอยู่ในเขตควบคุมนั้นๆ นอกจากนี้ยังเป็นศูนย์กลางข่าวแผนการบินอีกด้วย ซึ่งผู้ควบคุมจราจรทางอากาศจะอาศัยข้อมูลนี้ เพื่อจัดการจราจรให้เหมาะสมกับสภาพการจราจรในขณะนั้น เพื่อให้เครื่องบินทุกลำเดินทางโดยปลอดภัย ทั้งนี้เพราะว่าเมื่อนักบินกำลังทำการบินอยู่ จะไม่สามารถทราบถึงสภาพการจราจรทางอากาศในขณะนั้นได้เลย จึงต้องอาศัยคำแนะนำในการทำการบินจากผู้ควบคุมจราจรทางอากาศ ซึ่งนักบินจะต้องปฏิบัติตามอย่างเคร่งครัด เพื่อมิให้รบกวนเส้นทางบินของเครื่องบินลำอื่นหรือเกิดอันตรายได้ นอกจากมีเหตุฉุกเฉินจำเป็นเท่านั้น ซึ่งผู้ควบคุมจราจรทางอากาศจะอำนวยความสะดวกให้เป็นกรณีพิเศษ

---

* เรียกย่อๆว่า "เขต AOR" เช่นในทะเลมหาสมุทร

** เรียกย่อว่า หนังสือ AIP ซึ่งในแต่ละประเทศจะจัดทำขึ้นเป็นการเฉพาะ ดังนั้นเมื่อจะทำการบินไปยังที่ใดก็จะสามารถหารายละเอียดต่างๆได้จากหนังสือ AIP ของประเทศนั้น

*** ซึ่งโดยปกติ จุดรายงานการบินจะอยู่ห่างกันประมาณ 120-250 ไมล์ทะเล ซึ่งจะใช้เวลาทำการบินประมาณ 15-30 นาที แต่อาจจะอยู่ใกล้หรือไกลกว่านี้ก็ได้ ขึ้นอยู่กับสภาพทางภูมิศาสตร์ด้วย และความเหมาะสม

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

เครื่องมือที่ใช้ในการควบคุมจราจรทางอากาศ

ในปัจจุบันการคมนาคมทางอากาศได้ทวีความสำคัญมากขึ้นเรื่อยๆ ทำให้ปริมาณเที่ยวบินเพิ่มสูงขึ้นมากทุกปีส่งผลให้การควบคุมจราจรทางอากาศได้ทวีความยุ่งยากมากขึ้นตามไปด้วย จึงได้มีการพัฒนาเครื่องมือและวิธีการควบคุมจราจรทางอากาศแบบต่างๆ เพื่อช่วยในการควบคุมการจราจรมาเป็นลำดับ ดังนี้

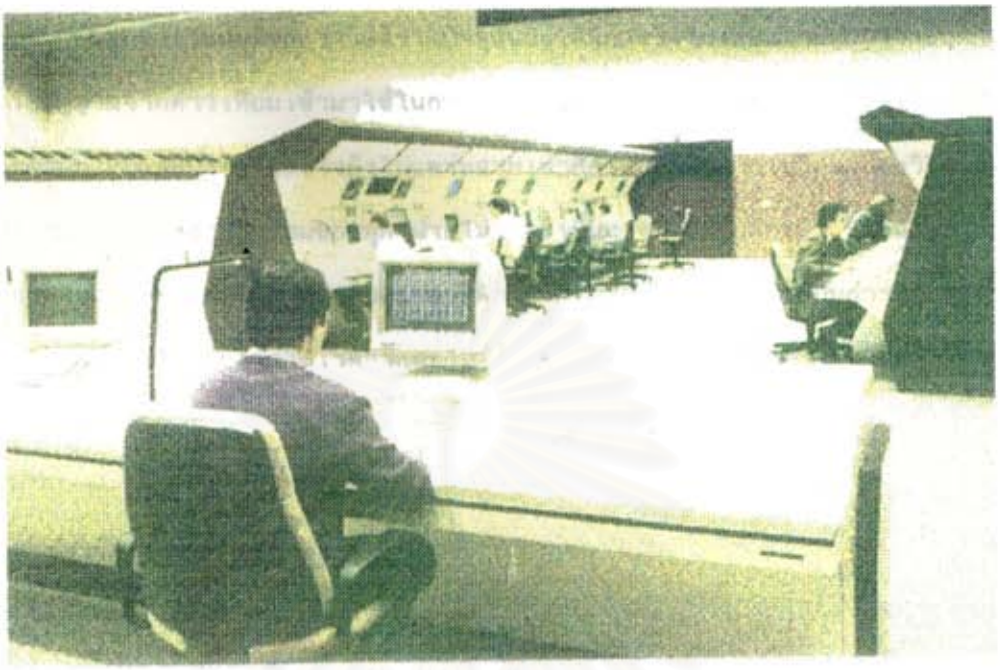
1. การควบคุมจราจรทางอากาศแบบทั่วไป (Conventional Control)* โดยการใช้วิทยุสื่อสารเป็นหลักในการติดต่อประสานงานระหว่างผู้ควบคุมจราจรทางอากาศกับนักบิน ซึ่งระบบนี้ใช้ได้ดีในบริเวณที่การจราจรทางอากาศมีความหนาแน่นไม่มากนัก เพราะใช้อุปกรณ์ที่มีราคาไม่แพงและมีความน่าเชื่อถือสูง สามารถใช้งานได้ดีทุกสภาพอากาศ แต่มีข้อเสียคือไม่สามารถระบุตำแหน่งพิกัดที่แน่นอนของเครื่องบินได้ ดังนั้นเพื่อความปลอดภัยในการบินจึงจำเป็นต้องเว้นระยะต่อมากทั้งทางด้านความสูง (Vertical Separation) และทางด้านในแนวราบ (Horizontal Separation)** ดังนั้นเมื่อปริมาณเที่ยวบินในเส้นทางบินต่างๆ เพิ่มสูงขึ้นทำให้เที่ยวบินต้องรอคอยการจัดการจราจรเป็นเวลานาน

2. การควบคุมจราจรทางอากาศแบบใช้เรดาร์ (Radar control) เป็นการใช้ระบบเรดาร์ในการควบคุมการจราจรทางอากาศร่วมกับวิทยุสื่อสาร เพื่อเพิ่มความสามารถในการควบคุมการจราจรทางอากาศให้มีประสิทธิภาพมากขึ้น ทั้งนี้เนื่องจากการใช้วิทยุสื่อสารแต่

---

* เรียกอีกชื่อหนึ่งว่า Non-Radar control

** ปกติระยะต่อทางด้านความสูงจะใช้ประมาณ 2000 ฟิต นอกจากกรณีฉุกเฉินสามารถลดได้เหลือ 1000 ฟิต และเมื่อใช้ระบบวิทยุในการควบคุมจราจรทางอากาศแต่เพียงอย่างเดียว จะต้องเว้นระยะต่อในแนวราบระหว่างเครื่องบินที่บินตามกันในทิศทางและระดับความสูงเดียวกัน ไม่น้อยกว่า 15 นาทีบิน หรือ 120 ไมล์ทะเล (ปกติเครื่องบินบินด้วยความเร็วเฉลี่ยประมาณ 480 ไมล์ทะเลต่อชั่วโมง หรือนาทีละ 8 ไมล์ทะเล)



รูปที่ 1 ห้องปฏิบัติการควบคุมจราจรทางอากาศ

เพียงอย่างเดียว ผู้ควบคุมจราจรทางอากาศจะไม่สามารถทราบตำแหน่งพิกัดที่แน่นอนของเครื่องบินได้เลย ระบบเรดาร์ช่วยให้ผู้ควบคุมจราจรทางอากาศทราบตำแหน่งพิกัดที่แน่นอนของเครื่องบินแต่ละลำได้อย่างแม่นยำ* ดังนั้น เมื่อใช้ระบบเรดาร์ในการควบคุมจราจรทางอากาศจะสามารถลดระยะต่อระหว่างเครื่องบินในแนวราบ ลงมาได้เหลือเพียง 5 ไมล์ทะเลเท่านั้น ทำให้สามารถเพิ่มปริมาณเที่ยวบินที่จะทำการบินในแต่ละเส้นทางบินและความสูงเดียวกันได้เป็นอย่างมาก

แต่เนื่องจากระบบเรดาร์มีราคาแพง ดังนั้นความเหมาะสมในการนำระบบเรดาร์มาใช้งาน จึงขึ้นอยู่กับปริมาณเที่ยวบินเทียบกับราคาของระบบเรดาร์ ซึ่งถ้าปริมาณเที่ยวบินไม่มากนัก การใช้ระบบเรดาร์ควบคุมจราจรทางอากาศจะไม่มีความคุ้มค่าทางเศรษฐกิจเพียงพอ

---

* ระบบเรดาร์อาศัยหลักการสะท้อนของคลื่นวิทยุที่ส่งออกไปกระทบกับลำตัวเครื่องบินและสะท้อนกลับมาสู่เครื่องรับ ซึ่งสามารถคำนวณตำแหน่งพิกัดของเครื่องบินแต่ละลำได้จากระยะเวลาที่คลื่นวิทยุเดินทางไปกลับและตำแหน่งทิศทางของสายอากาศในขณะนั้น

นอกจากระบบดังกล่าวนี้แล้วในปัจจุบันยังได้มีการริเริ่มเพื่อนำระบบการระบุตำแหน่งโดยใช้สัญญาณจากดาวเทียมเข้ามาใช้ในการควบคุมจราจรทางอากาศอีกด้วย แต่เนื่องจากเป็นอุปกรณ์สมัยใหม่และมีราคาแพงจึงยังไม่แพร่หลายเท่าที่ควร อีกทั้งการเปลี่ยนแปลงวิธีที่ใช้ปฏิบัติจะต้องกระทำโดยความยินยอมของทุกๆฝ่ายไม่ว่าจะเป็นทางด้านสายการบิน นักบินและผู้ควบคุมจราจรทางอากาศ จึงยังมิอาจเปลี่ยนแปลงหลักปฏิบัติโดยทันทีได้ ดังนั้นระบบการควบคุมจราจรทางอากาศแบบเดิมด้วยการใช้เรดาร์และไม่ใช้เรดาร์ จึงยังคงมีความจำเป็นในการใช้งานอยู่

ระบบเรดาร์ควบคุมจราจรทางอากาศ

ระบบเรดาร์ จะแสดงตำแหน่งพิกัดของเครื่องบินแต่ละลำที่ทำการบิน อยู่ภายในรัศมีครอบคลุมของเรดาร์นั้นๆได้อย่างแม่นยำ ซึ่งจะแสดงผลทุกๆคาบเวลารอบการหมุนของเรดาร์เรียกว่าเรดาร์สวีพ (Radar Sweep)** เรดาร์ที่ใช้ในการควบคุมจราจรทางอากาศจะมีอยู่ 2 ชนิดคือ

1. เรดาร์ปฐมภูมิ (Primary Radar) เป็นเรดาร์ที่อาศัยหลักการส่งลำคลื่นวิทยุความถี่สูงในย่านไมโครเวฟไปในทิศทางหนึ่งๆ ถ้าในบริเวณนั้นมีเครื่องบินทำการบินอยู่คลื่นวิทยุจะสะท้อนกับลำตัวของเครื่องบิน(ซึ่งทำด้วยโลหะและสะท้อนคลื่นวิทยุได้ดี) ลำคลื่นวิทยุที่สะท้อนกลับมาที่เครื่องรับของเรดาร์จะถูกประมวลผลและแปลความหมายว่าเป็นเครื่องบิน ซึ่งจะถูกนำไปแสดงผลบนจอแสดงผลตามตำแหน่งพิกัดที่คำนวณได้ เรดาร์ชนิดนี้สามารถตรวจจับเครื่องบิน เมฆฝน และสิ่งบินอื่นๆได้ แต่มีข้อจำกัดในด้านการใช้งาน เนื่องจากต้องใช้พลังงานสูงในการส่งคลื่น เพื่อให้ได้สัญญาณที่สะท้อนกลับมามีความแรงเพียงพอสำหรับตรวจจับได้ นอกจากนี้ยังถูกจำกัดด้วยความโค้งของผิวโลก ทำให้มีรัศมีในการตรวจจับประมาณ 80 ไมล์ทะเลเท่านั้น

** ปกติจานสายอากาศของเรดาร์จะหมุนด้วยความเร็วรอบประมาณ 5-12 รอบต่อนาที ซึ่งทำให้คาบเวลาของเรดาร์สวีพจะมีค่า 5-12 วินาทีด้วย

2. <u>เรดาร์ทุติยภูมิ</u> (Secondary Radar) เป็นเรดาร์ที่อาศัยหลักการตอบกลับของอุปกรณ์ทรานสปอนเดอร์ (Transponder) ที่ติดตั้งอยู่บนเครื่องบิน เมื่อเครื่องเรดาร์นี้ส่งสัญญาณวิทยุเป็นรหัสคลื่นออกไป ถ้ามีเครื่องบินทำการบินอยู่ในบริเวณนั้นอุปกรณ์ทรานสปอนเดอร์บนเครื่องบินจะส่งรหัสตอบกลับมาโดยอัตโนมัติ ประกอบด้วยรหัสประจำเที่ยวบิน และระดับเพดานบินมาให้ ส่วนระยะทางระหว่างเครื่องบินถึงสถานีเรดาร์ สามารถคำนวณได้จากระยะเวลาที่คลื่นเดินทางไป ส่วนทิศทางของเครื่องบินจะคำนวณได้จากทิศทางของจานสายอากาศของระบบเรดาร์ในขณะนั้น ข้อดีของเรดาร์แบบนี้ก็คือสามารถทำการตรวจจับเครื่องบินในระยะไกลได้ดี โดยใช้พลังงานในการส่งต่ำกว่าระบบเรดาร์แบบปฐมภูมิ เพราะไม่ได้อาศัยการสะท้อนกลับของคลื่นวิทยุ แต่ใช้การตอบกลับของอุปกรณ์ทรานสปอนเดอร์แทน นอกจากนี้ยังได้รับข้อมูลรหัสประจำเที่ยวบินและความสูงของเครื่องบินอีกด้วย แต่มีข้อเสียคือ เครื่องบินจะต้องติดอุปกรณ์นี้ด้วยทุกๆลำ ทำให้เพิ่มค่าใช้จ่ายในการบิน

<u>รูปที่ 2 จอแสดงผลเรดาร์ที่ใช้ในการควบคุมจราจรทางอากาศ</u>

ระบบเรดาร์ทั้งสองแบบนี้มีข้อดีข้อเสียต่างกัน ปกติจึงมีการใช้งานควบคู่กันไป โดยจะติดตั้งเรดาร์แบบปฐมภูมิที่บริเวณสนามบิน เนื่องจากเป็นระยะที่เครื่องบินอยู่ใกล้กับสนามบิน ซึ่งต้องทำการควบคุมการบินอย่างใกล้ชิดทันท่วงที และเรดาร์ทุติยภูมิเพื่อตรวจจับเครื่องบินในระยะไกล เมื่อผู้ควบคุมจราจรทางอากาศเฝ้าสังเกตการเคลื่อนที่ของเครื่องบินแต่ละครั้งก็จะทราบทิศทางและความเร็วของเครื่องบินได้ ทำให้สามารถตัดสินใจในการควบคุมเครื่องบินได้อย่างถูกต้องมากขึ้นกว่าการใช้ระบบวิทยุสื่อสารแต่อย่างเดียว

ระบบประมวลผลข้อมูลเรดาร์

ระบบประมวลผลข้อมูลเรดาร์ รับข้อมูลมาจากเรดาร์ทั้งแบบปฐมภูมิและแบบทุติยภูมิ โดยเรดาร์ทั้งสองแบบ จะส่งข้อมูลมาเป็นชุด เรียกว่าข้อมูลเรดาร์ (Track Message) ซึ่งประกอบไปด้วยข้อมูลเกี่ยวกับเครื่องบินนั้นๆที่เรดาร์ตรวจจับได้ เช่นทิศทาง ความเร็ว ความสูง เป็นต้น ข้อมูลนี้จะถูกส่งมาจากสถานีเรดาร์ผ่านทางระบบไมโครเวฟ สายโทรศัพท์ หรือส่งสัญญาณผ่านระบบดาวเทียม เพื่อนำมาประมวลผลต่อไป

แต่เนื่องจากรัศมีการตรวจจับของเรดาร์มีระยะจำกัด ดังนั้นเพื่อให้รัศมีเรดาร์ครอบคลุมพื้นที่ในการควบคุมจราจรทางอากาศ จึงอาจจะต้องมีสถานีเรดาร์หลายๆจุดในภูมิภาคต่างๆ ดังนั้นที่ศูนย์ระบบประมวลผลข้อมูลเรดาร์ จะต้องรวบรวมข้อมูลที่ถูกส่งมาจากสถานีเรดาร์ต่างๆ เพื่อประมวลผลและแยกแยะเป้าที่ถูกตรวจจับ เป้าทั้งหมดจะถูกวิเคราะห์และถูกส่งไปยังระบบแสดงผล เพื่อแสดงผลออกทางจอภาพขนาดใหญ่ เพื่อใช้ในการควบคุมจราจรทางอากาศต่อไป

ขั้นตอนการปฏิบัติในการทำการบิน

สำหรับเที่ยวบินขาออก ก่อนที่นักบินจะทำการบินประมาณ 1-2 ชั่วโมง นักบินจะไปที่ห้องต้นหน (Briefing room) เพื่อซักซ้อมความเข้าใจ รับทราบข้อมูลข่าวอากาศ ประกาศสำหรับนักบิน และกรอกแบบฟอร์มแผนการบินเกี่ยวกับเที่ยวบินนั้นๆ เรียกว่าแผนการบิน ซึ่งมีราย

ละเอียดที่สำคัญได้แก่ ชื่อเรียกขานเที่ยวบิน รหัสเที่ยวบิน แบบหรือชนิดของเครื่องบิน น้ำหนักบรรทุก จำนวนผู้โดยสาร เวลาเดินทางโดยประมาณ เส้นทางการบิน และจุดหมายปลายทาง ซึ่งพนักงานสื่อสารการบินจะนำข้อมูลเหล่านี้ ส่งผ่านเครือข่ายสื่อสารการบินระหว่างประเทศ (Aeronautical Fixed Telecommunication Network)* เพื่อส่งไปยังผู้เกี่ยวข้อง เช่น สนามบินที่จะทำการบินขึ้น ศูนย์ควบคุมจราจรทางอากาศ สนามบินปลายทาง เป็นต้น ดังนั้นก่อนที่นักบินจะนำเครื่องบินขึ้นนั้นผู้เกี่ยวข้องทั้งหมดจะทราบข้อมูลเกี่ยวกับเที่ยวบินนี้ได้ล่วงหน้า

เมื่อศูนย์ควบคุมจราจรทางอากาศ รับข่าวแผนการบินจากศูนย์สื่อสารการบินระหว่างประเทศก็จะนำไปกรอกลงในตารางข้อมูลการบิน (Flight Progress Strip)** โดยมีข้อมูลของเครื่องบินแต่ละลำที่ทำแน่งจุดรายงานการบินแต่ละจุดเรียงตามลำดับเวลา ดังนั้นเมื่อผู้ควบคุมจราจรทางอากาศดูจากตารางนี้ ก็จะทราบว่าที่แต่ละจุดรายงานการบินในเวลาหนึ่งๆ มีเครื่องบินกำลังบินเข้ามากี่ลำ ถ้ามีมากกว่าหนึ่งลำระยะห่างเพียงพอหรือไม่ และถ้าไม่พอควรจะทำอย่างไร ทำให้สามารถวางแผนการจราจรได้ล่วงหน้า

เมื่อถึงเวลาที่ทำการบินขึ้นจริงนักบินจะเรียกวิทยุมายังศูนย์ควบคุมจราจรทางอากาศเพื่อขออนุญาตทำการบิน ซึ่งถ้าผู้ควบคุมจราจรทางอากาศตรวจสอบแล้ว สามารถจัดการจราจรให้ได้ก็จะอนุญาตให้ทำการบินขึ้นได้ (แต่ถ้ามีปัญหา หรือไม่สามารถจัดการจราจรได้ก็จะสั่งให้เลื่อนเวลาบินขึ้นออกไปจนกว่าจะสามารถจัดการจราจรให้ได้)

<u>สำหรับเครื่องบินขาเข้า</u> ที่ทำการบินจากที่อื่นเข้ามายังเขตควบคุมการบิน ที่ทางศูนย์ควบคุมจราจรทางอากาศรับผิดชอบอยู่ เมื่อเครื่องบินเข้ามาในเขตความควบคุม ก็จะเรียก

---

* เรียกย่อๆว่า AFTN

** เรียกย่อๆว่า strip ซึ่งเป็นกระดาษเล็กๆหลายๆใบ ใช้สำหรับตรวจสอบการรายงานการบินและใช้ในการคำนวณว่าเที่ยวบินนี้ จะมาถึงเขตควบคุมการบินและจุดรายงานแต่ละจุดในเวลาใด เพื่อใช้เป็นข้อมูลในการควบคุมจราจรทางอากาศ

มายังศูนย์ควบคุมฯ เพื่อรายงานสภาพการบินในขณะนั้น ผู้ควบคุมจราจรทางอากาศจะสั่งการ เพื่อจัดการจราจรให้ เป็นไปตามหลักวิธีปฏิบัติการบิน ซึ่งโดยทั่วๆ ไปมักจะไม่ค่อยมีปัญหาสำหรับ เที่ยวบินขา เข้า เพราะได้มีการติดต่อประสานงานกันระหว่างศูนย์ควบคุมจราจรทางอากาศ ที่มี เขตควบคุมการบินประชิดกันตลอด เวลาผ่านระบบโทรศัพท์สายตรง (Voice page) ดังนั้นระหว่างที่ เครื่องบินยังไม่ได้ เข้ามายัง เขตความรับผิดชอบ ศูนย์ควบคุมฯที่อยู่ข้าง เคียงจะจัดการจราจรให้โดยมีระยะต่อในแนวราบและระดับความสูงตามที่ได้ตกลงกันไว้ล่วงหน้า ดังนั้น เมื่อ เครื่องบินผ่าน เข้ามาก็จะสามารถทำการบินต่อไปได้โดยไม่ต้องปรับการบินอีก (ในกรณีที่ไม่มีปัญหาอื่นๆ เป็นพิ เศษ)

ผู้ควบคุมจราจรทางอากาศ

ผู้ควบคุมจราจรทางอากาศ จะทำหน้าที่ในการติดต่อกับนักบินผ่านทางวิทยุสื่อสาร ในย่านความถี่การ เดินอากาศ (100-150 MHz) ซึ่งใช้ เป็นมาตรฐานทั่วโลก เพื่อความสะดวกในการติดต่อสื่อสารระหว่างนักบินกับศูนย์ควบคุมจราจรทางอากาศทั่วโลก นอกจากนี้ยังอาจจะมีจอแสดงผล เรดาร์ เพื่อบอกตำแหน่งของ เครื่องบินในการควบคุมจราจรแบบใช้ เรดาร์อีกด้วย ผู้ควบคุมจราจรทางอากาศจะ เป็นผู้ให้คำแนะนำในการบินแก่นักบิน เพราะ เมื่อนักบินกำลังทำการบินอยู่จะไม่สามารถทราบได้ว่าในขณะนี้รอบตัว เขา เครื่องบินอื่นตามมาหรือบินอยู่ข้างหน้าข้างบนข้างล่างหรือไม่ ดังนั้นนักบินจะต้องไว้ใจผู้ควบคุมจราจรทางอากาศ และปฏิบัติตามคำแนะนำอย่าง เคร่งครัด เพราะผู้ควบคุมจราจรทางอากาศได้รับข้อมูลข่าวสารการบินที่ทันสมัยกว่า

การฝึกหัดผู้ควบคุมจราจรทางอากาศ

ผู้ควบคุมจราจรทางอากาศจะต้องดูแลรับผิดชอบ เครื่องบินทุกลำ ที่ทำการบินอยู่ใน เขตควบคุมในขณะนั้น และต้องทำการตัดสินใจ เพื่อจัดการจราจรให้ เหมาะสม ซึ่ง เป็นภาระที่หนักและมีความ เสี่ยง ดังนั้นผู้ควบคุมจราจรทางอากาศจึงต้องได้รับการฝึกหัดมา เป็นอย่างดี มี

ประสบการณ์ในการทำงานสูง สามารถตัดสินใจ เพื่อแก้ไข เหตุการณ์ เฉพาะหน้าได้อย่างรวด เร็ว และมีรอบคอบ การฝึกผู้ที่จะ เป็นผู้ควบคุมจราจรทางอากาศ จึงประกอบด้วยขั้นตอนที่ค่อนข้าง ยากมาก เนื่องจากมีทั้งขั้นตอนในภาคทฤษฎีและภาคปฏิบัติ โดยในภาคทฤษฎี ผู้รับการฝึกจะต้อง เรียนรู้และท่องจำกฎระ เบียบในการ เดินอากาศ สภาพภูมิศาสตร์ และสมรรถนะของ เครื่องบิน ทั้งหมด สามารถคำนวณได้อย่างรวด เร็วและแม่นยำ รวมถึงสามารถใช้ภาษาอังกฤษได้อย่าง คล่องแคล่ว ส่วนในภาคปฏิบัติจะมีการทดสอบในห้องฝึกจำลอง เรดาร์ ตามแบบฝึกหัดที่ที่ได้จัด เตรียมไว้ล่วงหน้า

ปัจจุบันผู้ที่ เข้ารับการฝึกหลักสูตรการควบคุมจราจรทางอากาศ จะรับจากผู้จบปริญญา ตรีหรือจากศูนย์ฝึกการบินพล เรือนแห่งประ เทศไทย มา เข้ารับการฝึกทางด้านการควบคุมจราจร พื้นฐานประมาณ 6 เดือนก็จะมีการทดสอบความสามารถ เมื่อผ่านการทดสอบก็จะสามารถทำงาน เป็นพนักงานป้อนข้อมูลการบิน (Flight Data Operator) โดยมีหน้าที่ เตรียมแผนการบินให้ กับพนักงานควบคุมจราจรทางอากาศ เพื่อให้คุ้น เคยกับสภาพการทำงาน หลังจากทำงานได้ระยะ หนึ่ง(ประมาณ 1-2 ปี) ก็จะ เข้า เรียนต่อในหลักสูตรการควบคุมจราจรทางอากาศขั้นสูงต่อไป เมื่อผ่านการทดสอบทั้งทางด้านทฤษฎีและปฏิบัติก็จะสามารถ เข้าทำงาน เป็นผู้ช่วยผู้ควบคุมจราจร ทางอากาศ (ATC Assistant) ทำหน้าที่จดข้อมูลรายงานการติดต่อกับนักบิน รวมทั้งอำนวย ความสะดวกในการทำงานต่างๆให้กับผู้ควบคุมจราจรทางอากาศจนมีประสบการณ์ในการทำงาน มากพอสมควร ก็จะ เข้ารับการทดสอบความสามารถ เพื่อรับประกาศนียบัตรควบคุมจราจรทาง อากาศ รวมถึงผ่านการตรวจร่างกายและประสาทการรับรู้ทั้งหมด ว่ามีสภาพจิตใจที่สมบูรณ์ ไม่ เป็นคนใจกล้าบ้าบิ่น ชอบ เสี่ยงฯลฯ ซึ่งใช้ระยะ เวลาในการฝึกทั้งหมด ไม่ต่ำกว่า 5 ปี

ประ เทศไทยในปัจจุบันมีการฝึกสอนการควบคุมจราจรทางอากาศในภาคปฏิบัติ เฉพาะ แบบการควบคุมที่ไม่ใช้ เรดาร์ เท่านั้น ซึ่งมี เพียงระบบติดต่อสื่อสารทางสาย (intercom) ที่ ใช้แทนระบบวิทยุสื่อสารระหว่างผู้ควบคุมจราจร กับนักบินสมมุติซึ่งทำหน้าที่โดยครูผู้ฝึก โดยใน ขั้นตอนการฝึก ครูฝึกจะคำนวณตำแหน่งของ เครื่องบินทุกๆลำในแต่ละช่วง เวลา 1 นาที และจด บันทึกลงในตาราง ไว้บนกระดาษ เพื่อใช้ เป็นข้อมูลในการจำลองการฝึก

เมื่อเริ่มทำการฝึก(ครั้งละ 1 คน) จะแบ่งผู้รับการฝึกบางส่วนที่ยังไม่ได้ฝึกในเวลานั้นไปทำหน้าที่เป็นนักบินสมมุติ โดยมีครูผู้ฝึกเป็นผู้ควบคุมการฝึกอีกทีหนึ่ง* นักบินสมมุติจะใช้ข้อมูลที่ได้จดบันทึกไว้และคำนวณข้อมูลอื่นๆตามการสั่งการจากผู้รับการฝึก (ซึ่งทำหน้าที่เป็นผู้ควบคุมจราจรทางอากาศ) เพื่อใช้ในการรายงานการบินต่อไป

การฝึกสอนด้วยวิธีนี้ มีข้อจำกัดหลายประการ เช่น

1. ต้องใช้นักบินสมมุติหลายคนช่วยในการฝึก เพราะนักบินสมมุติแต่ละคนจะควบคุมเครื่องบินได้ 1-2 ลำ เท่านั้น เนื่องจากคำนวณไม่ทัน

2. ไม่สามารถพลิกแพลงการฝึกได้มาก เพราะข้อมูลการฝึกจะต้องคำนวณไว้ล่วงหน้าในบางส่วน มิฉะนั้นจะคำนวณไม่ทัน ในการระหว่างการฝึกจริง

3. ความสามารถของนักบินสมมุติแต่ละคน มีผลต่อการฝึกเป็นอย่างมาก จึงอาจต้องฝึกการทำหน้าที่เป็นนักบินสมมุติเพิ่มเติม เพื่อให้ทำหน้าที่ได้อย่างคล่องแคล่ว

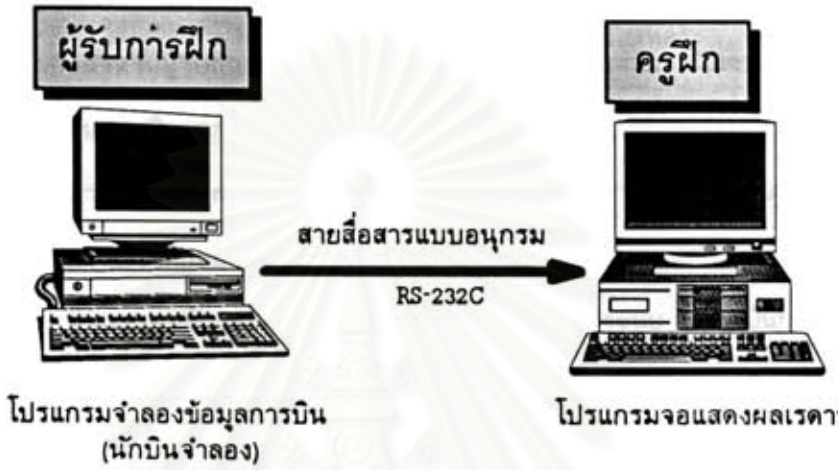
<u>การนำเครื่องคอมพิวเตอร์มาช่วยในการฝึก</u>

จากวิธีการฝึกข้างต้นพบว่าสามารถนำเครื่องคอมพิวเตอร์มาช่วยในการฝึกได้ โดยแทนที่ครูผู้ฝึกและนักบินสมมุติจะทำการคำนวณตำแหน่งและข้อมูลการบินอื่นๆ เอง ก็จะให้เครื่องคอมพิวเตอร์คำนวณให้และแสดงผลทางหน้าจอ เพื่อให้นักบินสมมุติใช้เป็นข้อมูลในการรายงานการบินต่อไป นอกจากนี้ยังสามารถปรับปรุงให้สามารถใช้ในการฝึกแบบใช้เรดาร์ได้อีกด้วยโดยการเพิ่มเติมในส่วนแสดงผล ให้สามารถแสดงผลข้อมูลการบินในรูปแบบกราฟิกลักษณะเดียวกับ

---

* เนื่องจากในแต่ละแบบฝึกหัดอาจจะมีเที่ยวบินเป็นจำนวนมาก โดยที่นักบินสมมุติแต่ละคนจะสามารถควบคุมเครื่องบินได้ 1-2 ลำ เท่านั้น แต่ครูผู้ฝึกมีจำนวนน้อยไม่เพียงพอในการทำหน้าที่นักบินสมมุติได้หลายๆคนพร้อมกัน จึงต้องใช้ผู้รับการฝึกคนอื่นๆมาทำหน้าที่แทน

จอ เรดาร์ที่ใช้งานจริง เพื่อให้ผู้รับการฝึกมีความคุ้น เคยกับระบบ เรดาร์ ซึ่งจะส่งผลดีในการทำงานกับระบบจริงต่อไป



รูปที่ 3 โครงสร้างของระบบจำลองข้อมูล เรดาร์

ระบบจำลองข้อมูล เรดาร์นี้ ประกอบด้วยโปรแกรมที่สำคัญ 2 ส่วน ได้แก่

1. โปรแกรมจำลองข้อมูลการบิน(นักบินสมมุติ) สำหรับผู้ควบคุมการฝึก
2. โปรแกรมแสดงผลข้อมูล เรดาร์ สำหรับผู้รับการฝึก

โดยที่โปรแกรมนักบินสมมุติทำหน้าที่ เป็นตัวสร้างข้อมูลการบิน เพื่อจำลองการ เคลื่อนที่ของ เครื่องบินตามแบบฝึกหัดที่ได้กำหนดไว้(ซึ่งครูผู้ฝึกสามารถควบคุมได้) ข้อมูลนี้จะ เปรียบได้กับข้อมูลที่ได้รับจาก เครื่องรับของ เรดาร์จริงๆ ซึ่งจะถูกส่งไปยังโปรแกรมจอแสดงผล เพื่อแสดงผลในลักษณะ เดียวกับจอ เรดาร์ที่ใช้งานจริง เพื่อให้ผู้รับการฝึกทราบ โดยที่ทั้งครูผู้ฝึกและผู้รับการฝึก(ที่ทำหน้าที่ เป็นผู้ควบคุมจราจรทางอากาศ) จะติดต่อสื่อสารกันได้ผ่านระบบสื่อสารทางสาย ซึ่งใช้แทนวิทยุสื่อสารระหว่างนักบินและผู้ควบคุมจราจรทางอากาศ

นอกจากนี้ยังมีข้อดีในการฝึกหัดการควบคุมจราจรทางอากาศ ดังนี้

1. สามารถจำลองเหตุการณ์ต่างๆได้ตามความต้องการ เช่น เครื่องบิน บินเข้าหากันที่ระดับความสูงเดียวกัน เครื่องบินโดนจี้บังคับ เครื่องบินขัดข้อง เป็นต้น เพื่อเป็นการฝึกความพร้อมในการรับสถานการณ์ฉุกเฉินเฉพาะหน้า

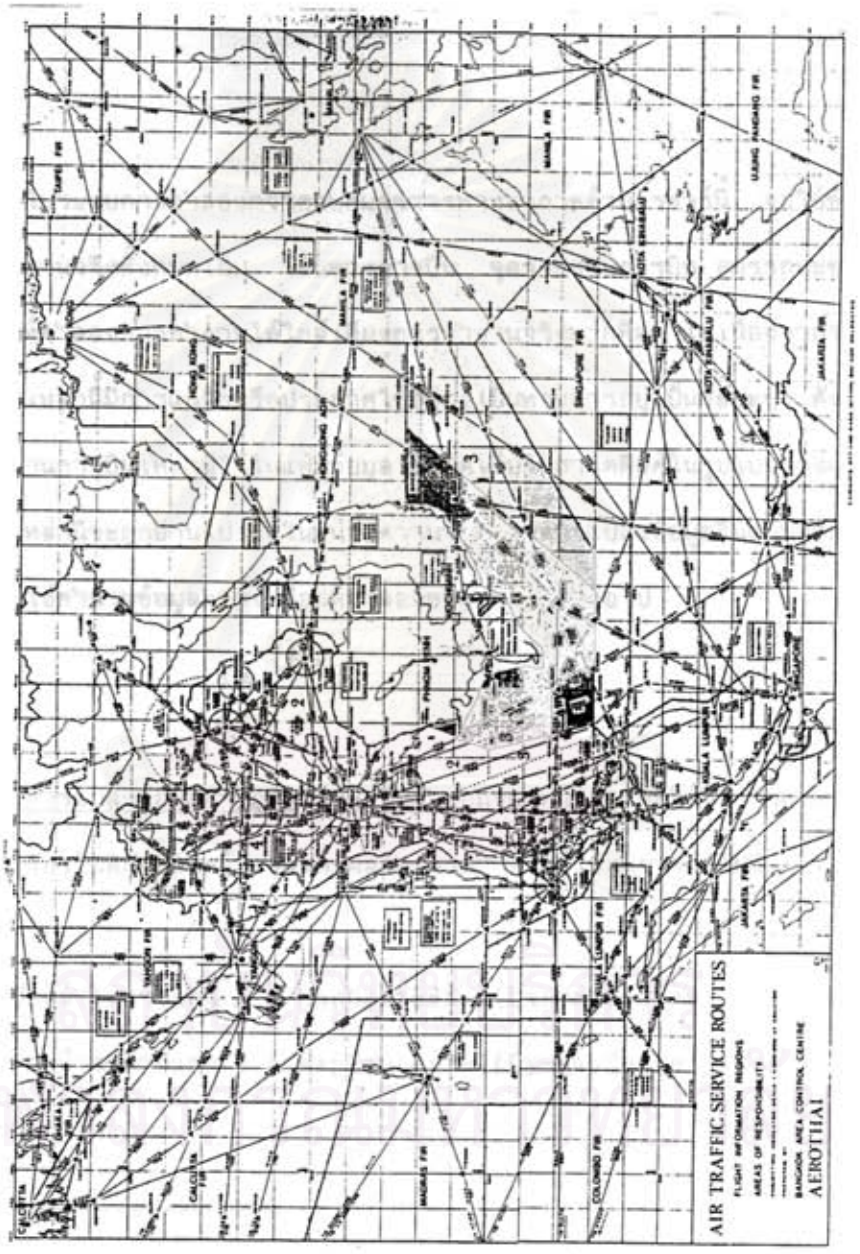
2. สามารถสร้างแบบฝึกหัดในการควบคุมจราจรทางอากาศ ให้ทำการฝึกเป็นไปในรูปแบบเดียวกันสำหรับผู้รับการฝึกแต่ละคน เพื่อที่จะประเมินผลการฝึกของผู้รับการฝึกแต่ละคนได้ (เพราะเป็นการฝึกจากแบบฝึกหัดเดียวกัน)

3. สามารถเรียกแบบฝึกหัดใดๆที่ได้สร้างไว้ เพื่อมาทำการฝึกทบทวน หรืออธิบายชี้แจงรายละเอียดในการฝึกสอนของแต่ละแบบฝึกหัดได้หลายๆครั้ง

4. สามารถเร่งลด หรือหยุดความเร็วของนาฬิการะบบ เพื่อเป็นการทดสอบความสามารถในการควบคุมจราจรหรือใช้หยุดเวลาเพื่อชี้แจงอธิบายระหว่างการฝึก

5. การฝึกจะใช้ข้อมูลจำลองโดยไม่รบกวนการทำงานกับระบบเรดาร์จริง ทั้งนี้เพราะการฝึกกับระบบที่ใช้งานจริงอาจเกิดความผิดพลาดได้ ซึ่งมีผลต่อความปลอดภัยในการเดินอากาศ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4 แผนที่เส้นทางการบินในเขตควบคุมการบินกรุงเทพฯ

บทที่ 3

การจัดเก็บแฟ้มข้อมูลที่ใช้ในการเดินอากาศ

ในระบบการจำลองการควบคุมจราจรทางอากาศด้วยเรดาร์นี้ จะใช้ข้อมูลทางด้านการบินที่ใช้งานจริงทั้งหมด เช่น เส้นทางการบิน จุดรายงานการบิน สมรรถนะของเครื่องบิน ฯลฯ เพื่อให้จำลองการทำงานได้ใกล้เคียงการทำงานจริงมากที่สุด แต่เนื่องจากจากข้อมูลทางด้านการบินเหล่านี้มีการแก้ไขหรือประกาศใช้อย่างเป็นทางการอยู่เป็นระยะๆ ดังนั้นจึงจัดเก็บข้อมูลทางด้านการบินเหล่านี้ไว้ในแฟ้มข้อมูล ซึ่งจัดเก็บในฮาร์ดดิสค์ในรูปแบบของแฟ้มตัวอักษร* แฟ้มข้อมูลเหล่านี้จะถูกอ่านไปเก็บในหน่วยความจำ โดยการแปลงข้อมูลในแบบสายอักษรให้เป็นตัวเลข เพื่อใช้คำนวณข้อมูลการเคลื่อนที่จำลองของเครื่องบินต่อไป

<u>การแปลงข้อมูลทางภูมิศาสตร์เป็นตำแหน่งบนหน้าจอ</u> จะใช้วิธีการแปลงพิกัดจากพิกัดของแผนที่ ไปเป็นพิกัดของหน้าจอแสดงผลโดยตรง** โดยใช้สมการต่อไปนี้

ตำแหน่งพิกัดหน้าจอทางแกน X = MapCenterX+((Lx-MapCenterLong)*MapScale))

ตำแหน่งพิกัดหน้าจอทางแกน Y = MapCenterY-((Ly-MapCenterLat)*MapScale))

---

* ทำให้สามารถเตรียมข้อมูลหรือแก้ไขข้อมูลได้ง่าย ด้วยการใช้โปรแกรมประมวลผลคำโดยทั่วไป และผู้ใช้ยังสามารถเรียกมาดูบนหน้าจอหรือสั่งพิมพ์ได้โดยง่าย ด้วยการเรียกใช้คำสั่งของระบบปฏิบัติการโดยตรง

** เพราะตำแหน่งพิกัดเหล่านี้ได้จากการอ่านจากแผนที่โดยตรง ซึ่งได้มีการชดเชยในส่วนของความโค้งของผิวโลกไว้แล้ว

โดยที่

Lx	เป็นตำแหน่งพิกัดทางด้านแนวนอนของจุดที่ต้องการแสดงผล
Ly	เป็นตำแหน่งพิกัดทางด้านแนวตั้งของจุดที่ต้องการแสดงผล
MapCenterX	เป็นตำแหน่งพิกัดกลางหน้าจอทางด้านแกน X
MapCenterY	เป็นตำแหน่งพิกัดกลางหน้าจอทางด้านแกน Y
MapCenterLat	เป็นพิกัดทางด้านแนวตั้งของจุดศูนย์กลางในการแสดงผล
MapCenterLong	เป็นพิกัดทางด้านแนวนอนของจุดศูนย์กลางในการแสดงผล
MapScale	เป็นตัวคูณ เพื่อปรับมาตราส่วนในการแสดงผล

โดยแฟ้มข้อมูลที่ใช้ในการทำงานของโปรแกรมทั้งหมด ประกอบไปด้วย

1. แฟ้มข้อมูล เส้นขอบ เขตควบคุมการบิน ประกอบด้วย เส้นแสดงขอบ เขตควบคุมการบิน เส้นขอบ เขตประเทศ และ เส้นแบ่ง เขตทางภูมิศาสตร์ เส้นขอบ เขตทั้งหมดนี้จะจัด เก็บในรูปแบบตำแหน่งพิกัดทางด้านแนวตั้งและแนวนอนของทั้งตำแหน่งจุด เริ่มต้นและจุดสิ้นสุดของ เส้นตรงสั้นๆจำนวนมาก* และ เมื่อต้องการแสดงผล ก็จะนำตำแหน่งพิกัดของแต่ละ เส้นตรงมาแปลง เป็นพิกัดของตำแหน่งบนหน้าจอ เพื่อแสดงผลต่อไป เส้นขอบ เขต เหล่านี้ไม่ได้ใช้ในการควบคุมจราจรทางอากาศโดยตรง เพราะว่าโดยปกติ เครื่องบินจะบินตาม เส้นทางการบิน เท่านั้น เส้นขอบ เขตเหล่านี้มี เพียง เพื่อบอกขอบ เขตแสดงความรับผิดชอบ เท่านั้น

โครงสร้างของแฟ้มข้อมูลมีดังนี้

---

* โดยในส่วนที่ เป็น เส้นโค้งจะถูกแทนด้วย เส้นตรงสั้นๆทั้งนี้ เพื่อให้ง่ายต่อการจัด เก็บข้อมูลให้ เป็นรูปแบบ เดียวกันทั้งหมด)

ชื่อจุด	ชนิดของจุด	หน่งทางละติจูด	ต้าแหน่งทางลองติจูด
A1	S	13.28333	99.91667
A2	S	13.50000	100.23333
A3	S	13.55000	101.58333
....	....	.....	.....

ตารางที่ 1 โครงสร้างแฟ้มข้อมูล เส้นแสดงขอบ เขตควบคุมการบิน

โดยที่ ชื่อจุด จะ เป็น เพียงชื่ออ้างอิงในการป้อน แก้ไขข้อมูล โดยไม่ได้นำไป แสดงผล หรือ มีความหมายพิ เศษอื่นใด

ชนิดของ เส้น สามารถการแสดงผล ให้ เป็นแบบ เส้นทึบ (S) หรือ เส้นประ (N) เพื่อ เป็นการ เน้น เส้นแต่ละ เส้น โดย เส้นขอบ เขตความรับผิดชอบ ของไทย จะแสดง เป็น เส้นทึบ ส่วน เส้นอื่นๆจะ เป็น เส้นประ

ต้าแหน่งทางละติจูด จัด เก็บข้อมูลพิกัดทางแนวตั้งของจุดนั้น

ต้าแหน่งทางลองติจูด จัด เก็บข้อมูลพิกัดทางแนวนอนของจุดนั้น

การแสดง เส้นขอบ เขตควบคุมการบิน ทำโดยการอ่านข้อมูลในแฟ้มข้อมูลบรรทัดแรก มา เป็นจุดตั้งต้นและวนรอบอ่านบรรทัดต่อไป เพื่อทำการลาก เส้นระหว่างจุดทั้งสองจากนั้นจะนำ จุดหลังมา เป็นจุดตั้งต้น เพื่ออ่านจุดปลายทางจุดใหม่มา โดยจะวนรอบ เช่นนี้จนกว่าจะจบแฟ้ม ดังนั้นจึงต้อง เรียงระ เบียนของจุดที่ใกล้กันติดต่อกันไป นอกจากในกรณีที่ เส้นขอบ เขตนั้นวน เป็น วงรอบ เช่น เกาะ หรือทะ เลสาบ จะต้องนำข้อมูลของจุด เริ่มต้นมา เป็นจุดสุดท้ายด้วย เพื่อบังคับ ให้โปรแกรมลาก เส้นมาบรรจบ เป็นวงรอบด้วย

ในการจัด เก็บแผนที่ขนาดใหญ่ที่มีความซับซ้อนมาก สามารถ เก็บข้อมูลทั้งหมดในแฟ้ม ข้อมูล เดียวกันได้ โดยจะแยก เส้นของแต่ละกลุ่มออกจากกันด้วยบรรทัดว่างที่มีอักษรตัวแรก เป็น ';' โดย เมื่อโปรแกรมอ่านพบบรรทัดว่างก็จะถือว่าจบ เส้นในชุดนั้นๆ เพื่อ เริ่มต้นวาด เส้นชุด ใหม่ต่อไป

2. แฟ้มข้อมูลจุดรายงานการบิน ซึ่งประกอบด้วยพิกัดตำแหน่งที่นักบินต้องรายงานสภาพการบินกับศูนย์ควบคุมจราจรทางอากาศมีทั้งจุดบังคับรายงานการบิน(Compulsory) และจุดไม่บังคับรายงานการบิน (Non-Compulsory) ซึ่งแต่ละจุดจะมีชื่อ เรียกโดยไม่ซ้ำกัน ดังนั้นในการจัดเก็บจึงต้องเก็บทั้งชื่อจุด ชนิดของจุด และตำแหน่งพิกัดทางภูมิศาสตร์ด้วย

แฟ้มข้อมูลจุดรายงานการบินนี้จะถูกอ่านมาเก็บในหน่วยความจำ เพื่อใช้ในการคำนวณและการแสดงผล โดยจะใช้สัญญลักษณ์สาม เหลี่ยมทึบแทนจุดบังคับรายงานการบิน และสัญญลักษณ์สาม เหลี่ยมโปร่งแทนจุดไม่บังคับรายงานการบิน ณ ตำแหน่งพิกัดทางด้านแนวตั้งและแนวนอนที่ได้กำหนดไว้ โดยใช้วิธีการค้นหาแบบ เรียงลำดับ ซึ่งมีชื่อจุดรายงานการบิน เป็นตัวอ้างอิง

ชื่อจุด	ชนิดของจุด	ตำแหน่งละติจูด	ตำแหน่งลองติจูด
ADNEP	N	8.06000	101.05833
ADPAT	C	5.93333	104.09167
ALBOS	C	14.75833	101.01667
ALGOR	C	8.00000	104.00000
.....	.....	.....	.....

ตารางที่ 2 โครงสร้างแฟ้มข้อมูลแสดงจุดรายงานการบิน

โดยที่ ชื่อจุด ชื่อจุดรายงานการบิน ตามที่ประกาศในหนังสือ AIP
ชนิดของจุด แบบบังคับรายงาน (C) และไม่บังคับรายงาน (N)
ตำแหน่งละติจูด จัดเก็บข้อมูล เส้นแวงทางภูมิศาสตร์ของจุดนั้น
ตำแหน่งลองติจูด จัดเก็บข้อมูล เส้นรุ้งทางภูมิศาสตร์ของจุดนั้น

3. แฟ้มข้อมูล เส้นทางการบิน มีจุด เริ่มต้นที่สนามบินหรือจุดรายงานการบิน ไปสิ้นสุดที่อีกสนามบินหนึ่งหรืออีกจุดรายงานการบินหนึ่ง โดยแต่ละ เส้นทางการบินจะ เป็น เส้นตรงที่ลากระหว่างจุดรายงานการบิน(หรือสนามบิน) 2 จุด เสมอ ดังนั้น เส้นทางการบินยาวๆอาจลากผ่านจุดรายงานการบินหลายๆจุดก็ได้ อีกทั้งแต่ละ เส้นทางการบินยังอาจมีความยาวไม่ เท่ากัน ในการ

จัดเก็บข้อมูลจึงจะจัดให้อยู่ในรูปแบบเดียวกันทั้งหมด ไม่ว่าจะเป็นเส้นทางการบินสั้นหรือเส้นทางบินยาวๆ เช่น เส้นทางการบิน อัลฟาวัน (A1) เริ่มจากจุดรายงานการบินอาราโต้ (ARATO) มายังลิมลา (LIMLA) โอซูกะ(OSUKA) แบงคอก(BKK) เซลก้า(SELKA) อุบล(UBL) และบุทตร้า(BUTRA) ตามลำดับ จะจัดเก็บดังนี้

ชื่อเส้นทางการบิน	ชนิดของเส้น	ชื่อจุดรายงานการบินต้นทาง	ชื่อจุดรายงานการบินปลายทาง
A1	S	ARATO	LIMLA
A1	S	LIMLA	OSUKA
A1	S	OSUKA	BKK
A1	S	BKK	SELKA
A1	S	SELKA	UBL
A1	S	UBL	BUTRA
.....	....	.....	.....

ตารางที่ 3 โครงสร้างแฟ้มข้อมูลเส้นทางการบิน

โดยที่ ชื่อเส้นทางการบิน ชื่อเส้นทางการบิน ตามที่ประกาศในหนังสือ AIP

ชนิดของเส้น สงวนไว้สำหรับการใช้งานในอนาคตปกติมีค่าเป็น 'S'

ชื่อจุดรายงานการบินต้นทาง ชื่อจุดรายงานการบินต้นทาง

ชื่อจุดรายงานการบินปลายทาง ชื่อจุดรายงานการบินปลายทางของเส้นทางการบิน

แฟ้มข้อมูลเส้นทางการบินนี้จะถูกอ่านมาเก็บในหน่วยความจำเช่นเดียวกัน เพื่อใช้ในการตรวจสอบเส้นทางการบินที่ระบุในแบบฝึกหัดการบิน และใช้ในการแสดงผลเส้นทางการบินที่หน้าจอแสดงผลเรดาร์ รวมทั้งใช้ในการคำนวณการเคลื่อนที่ของเครื่องบินตามเส้นทางการบินโดยอัตโนมัติ

เนื่องจากเส้นทางการบินที่ยาวๆก็จะจัดเก็บในลักษณะ และใช้ชื่อเส้นทางเดียวกัน ดังนั้นในการค้นหาเส้นทางการบิน จึงต้องทำการค้นหาชื่อเส้นทางการบินและจุดรายงานการบิน

เริ่มต้น แล้วทำการตรวจสอบว่าจุดรายงานปลายทางนั้นตรงกับจุดหมายปลายทางที่ต้องการหรือไม่ ถ้าไม่ใช่ก็จะต้องทำการค้นต่อไปโดยใช้ชื่อ เส้นทางการบิน เดิมและนำจุดรายงานปลายทางเดิมมา เป็นจุดรายงาน เริ่มต้นในการค้นหาใหม่โดยจะทำซ้ำ เช่นนี้จนกว่าจะพบจุดสิ้นสุดปลายทางที่ต้องการ หรือหาจนหมดแล้วยังไม่พบ

นอกจากนี้ เส้นทางการบินนี้ยังใช้ร่วมกันทั้งด้านขาไป (จุดต้นทางไปยังจุดปลายทาง) และขากลับ (จุดปลายทางกลับมายังจุดต้นทาง) โดยใช้ชื่อ เส้นทางการบิน เส้นเดียวกัน ดังนั้นในการค้นหา เส้นทางจะต้องตรวจสอบทั้งขาไปและกลับ นั่นคือการสลับจุด เริ่มต้นและจุดสิ้นสุดกันด้วย จุดรายงานการบินที่ได้มาจะนำไปค้นหาตำแหน่งพิกัดจากฐานข้อมูลจุดรายงานการบิน

4. แฟ้มข้อมูล เส้นขอบ เขตสนามบิน จะ เก็บข้อมูลของ เส้นที่แสดงขอบ เขตความรับผิดชอบของแต่ละสนามบิน ซึ่งโดยปกติแต่ละสนามบินจะมีขอบ เขตความรับผิดชอบภายในรัศมี 35 ไมล์ทะ เลโดยรอบสนามบิน แต่ เนื่องจากในบางสนามบินจะมีสภาพทางภูมิศาสตร์ที่ต่างกันไป เช่นในบางทิศทางอาจ เป็นภู เขาสูง เขตทหารหรือ เขตหวงห้ามอื่นๆ ซึ่งไม่สามารถใช้ทำการบินได้ ดังนั้นจึงจัด เก็บ เส้นแสดงขอบ เขต เหล่านี้ในแฟ้มข้อมูล โดย เส้น เหล่านี้จะมีลักษณะพื้นฐานประกอบด้วย เส้นตรง และส่วนโค้งที่มีจุดศูนย์กลางที่สนามบิน ดังนั้นในแต่ละระ เบียนจึงจะต้องสามารถ เก็บข้อมูลทั้ง เส้นตรง และส่วนโค้งได้ โดยมีโครงสร้างข้อมูลดังนี้

ชื่อสนามบิน	ชนิด	LAT1	LONG1	FIELD1	FIELD2	FIELD3
CHIANGRAI	A	19.94833	99.88167	0.0	360.0	1.0
CHIANGRAI	A	19.94833	99.88167	146.0	244.0	15.0
CHIANGRAI	L	19.82167	100.08833	19.62833	100.24333	0.0
CHIANGRAI	A	19.94833	99.88167	42.0	94.0	30.0
CHIANGRAI	L	19.60833	99.53333	19.80833	99.67667	0.0
.....	....	.....	.....	.....	.....	.....

ตารางที่ 4 โครงสร้างแฟ้มข้อมูล เส้นขอบ เขตสนามบิน

โดยที่ ชื่อสนามบิน มีไว้เพื่ออ้างอิงในการป้อนข้อมูล

ชนิด 'L' หมายถึง เส้นตรง และ 'A' หมายถึง ส่วนโค้ง

ในกรณีเส้นตรง

LAT1 พิกัดทางแนวตั้งของจุดเริ่มต้นของเส้น

LONG1 พิกัดทางแนวนอนของจุดเริ่มต้นของเส้น

FIELD1 พิกัดทางแนวตั้งของจุดสิ้นสุดของเส้น

FIELD2 พิกัดทางแนวนอนของจุดสิ้นสุดของเส้น

FIELD3 ไม่ได้ใช้

ในกรณีส่วนโค้ง

LAT1 พิกัดทางแนวตั้งของจุดศูนย์กลางของส่วนโค้ง

LONG1 พิกัดทางแนวนอนของจุดศูนย์กลางของส่วนโค้ง

FIELD1 มุมเริ่มต้นของส่วนโค้ง

FIELD2 มุมสิ้นสุดของส่วนโค้ง

FIELD3 รัศมีของส่วนโค้ง

แฟ้มข้อมูลนี้จะถูกอ่านมาและทำการแสดงผลเลย โดยไม่ได้จัดเก็บในหน่วยความจำ เนื่องจากใช้ในการแสดงผลอย่างเดียว ไม่ได้ใช้ในการคำนวณใดๆ โดยในการแสดงผลจะต้องมีการตรวจสอบชนิดของข้อมูลในระเบียนนั้นๆก่อน ซึ่งถ้าระเบียนนั้นเป็นข้อมูลของเส้นตรงจะมีข้อมูลของพิกัดจุดเริ่มต้นและจุดสิ้นสุด ซึ่งใช้ในการแสดงผลได้ แต่ถ้าระเบียนนั้นเป็นข้อมูลของส่วนโค้ง ก็จะอ่านข้อมูลของพิกัดจุดศูนย์กลางของวงกลม มุมของจุดเริ่มต้นและมุมของจุดสิ้นสุด รวมถึงรัศมีของส่วนโค้งนั้นๆมาให้ เพื่อแสดงผลส่วนโค้งของวงกลม

5. แฟ้มข้อมูลแบบฝึกหัดการบิน ประกอบด้วยข้อมูลแผนการบินของเที่ยวบินแต่ละลำ โดยที่ผู้ควบคุมการฝึกจะเป็นผู้จัดเตรียมแบบฝึกหัดการบินไว้ล่วงหน้า โดยที่ข้อมูลแต่ละบรรทัดหมายถึงข้อมูลแผนการบินของแต่ละเที่ยวบิน ข้อมูลนี้จะใช้เป็นค่าเริ่มต้นให้กับโปรแกรมจำลองข้อมูลการบิน(นักบินสมมุติ) ซึ่งมีรายละเอียดดังนี้

ชนิดของฟิลด์	รายละเอียด	หมายเหตุ
TIME	แสดงเวลาเริ่มต้นที่จะทำการบิน	
CALLSIGN	แสดงชื่อเรียกขาน ซึ่งโดยทั่วไปมักจะเป็นชื่อเที่ยวบินนั้นๆ	
ACTYPE	แสดงแบบ,ชนิดของเครื่องบินที่จะใช้ทำการบิน	
SQUAWK	แสดงรหัสประจำเที่ยวบิน	
ALTITUDE	แสดงระดับความสูงที่จะทำการบิน ณ จุดเริ่มต้น	
SPEED	แสดงความเร็วที่จะทำการบิน ณ จุดเริ่มต้น	
HEADING	แสดงทิศทางการบิน ณ ที่จุดเริ่มต้น	
STARTPOINT	แสดงจุดเริ่มต้นที่จะทำการบิน	
ROUTE	แสดงเส้นทางการบินและจุดที่ต้องเปลี่ยนเส้นทางการบิน	
TURNPOINT	แสดงจุดที่จะทำการเลี้ยวเพื่อเปลี่ยนเส้นทางบิน หรือเป็นจุดสิ้นสุดการบินแบบอัตโนมัติ (Auto Pilot)	ช่อง ROUTE และ TURN POINT สามารถมีซ้ำได้

ตารางที่ 5 โครงสร้างแฟ้มข้อมูลแบบฝึกหัดการบิน

โดยจะกำหนด เวลาที่ เที่ยวบิน เริ่มทำการบินซึ่งจะปรากฏบนจอ เรดาร์ที่ตำแหน่งของจุดรายงานการบินใดๆที่ต้องการจะ เริ่มทำการบินได้ มีการกำหนดชื่อ เรียกขาน เที่ยวบินสำหรับใช้ในการติดต่อกับนักบิน อีกทั้งรหัสประจำ เที่ยวบิน*

ในส่วนการระบุสมรรถนะการบินของ เครื่องบิน จะระบุโดยใช้แบบรุ่นหรือชนิดของเครื่องบิน เป็นตัวกำหนด เพื่อใช้ในการคำนวณตำแหน่งการ เคลื่อนที่ต่อไป อีกทั้งยังต้องกำหนดค่า เริ่มต้นให้กับ เที่ยวบินด้วยประกอบด้วยระดับ เพดานบิน ความ เร็วในการบิน และทิศทางการบิน ณ ตำแหน่งจุด เริ่มต้น รวมทั้งต้องระบุ เส้นทางการบินที่จะทำการบินตามรายทางด้วย โดยที่ เมื่อ เที่ยวบินมีการ เปลี่ยน เส้นทางการบิน จะต้องระบุจุดที่จะทำการ เปลี่ยน เส้นทาง และ เส้น

---

* ซึ่ง เป็นรหัสมาตรฐานประกอบด้วยตัวอักษร 'A' และตามด้วย เลขฐานแปดอีก 4 ตัว เช่น A3341 หรือ A2177 เป็นต้น เพื่อใช้แยกแยะ เครื่องบิน โดยที่รหัสนี้จะต้องไม่ซ้ำกันในแต่ละ เที่ยวบิน

ทางใหม่ที่จะทำการบินต่อไป เช่น

0101 THA402 B747 A2202 310 480 344 IDAGA B469 BKK

หมายถึง

เที่ยวบินที่ไทยอินเตอร์ฯ 402 เริ่มทำการบินเมื่อเวลา 01:01 ด้วยเครื่องบินแบบโบอิ้ง 747 ใช้รหัสประจำเครื่องว่า เอ2202 ใช้เพดานบิน 31,000 ฟิต ด้วยความเร็ว 480 นอต ถือเข็มทิศ 344 เริ่มจากจุดอิดาก้า (IDAGA) ตามเส้นทางการบินบราโว 469 สิ้นสุดการบินที่กรุงเทพฯ

0105 THA603 EA30 A3546 310 480 256 BUTRA A1 UBL W1 BKK

หมายถึง

เที่ยวบินที่ไทยอินเตอร์ 603 เริ่มทำการบินเมื่อเวลา 01:05 ด้วยเครื่องบินแบบแอร์บัส เอ300 ใช้รหัสเที่ยวบินว่า เอ3546 ใช้เพดานบิน 31,000 ฟิต ด้วยความเร็ว 480 นอต ถือเข็มทิศ 256 เริ่มจากจุดบุทตร้า (BUTRA) ตามเส้นทางอัลฟา 1 เมื่อถึงอุบลฯ (UBL) แล้วให้เปลี่ยนไปใช้เส้นทาง วิสกี้ 1 และสิ้นสุดการบินที่กรุงเทพ

0110 SIA62 EA31 A3341 330 490 341 RYN G592 BKK

หมายถึง

เที่ยวบินที่สิงคโปร์ 62 เริ่มทำการบินเมื่อเวลา 01:10 ด้วยเครื่องบินแบบแอร์บัส เอ310 ใช้รหัสเที่ยวบิน เอ3341 ใช้เพดานบิน 33,000 ฟิตด้วยความเร็ว 490 นอต ถือเข็มทิศ 341 เริ่มจากระยอง (RYN) ตามเส้นทางกอล์ฟ 592 สิ้นสุดการบินที่กรุงเทพ

6. แฟ้มข้อมูลสมรรถนะของเครื่องบิน โดยที่เครื่องบินแต่ละลำจะมีคุณสมบัติ และความสามารถในการบินแตกต่างกันไป ซึ่งทั้งนักบินและผู้ควบคุมจราจรทางอากาศ จำเป็นต้องรู้ถึงสมรรถนะในการบินของเครื่องบินแต่ละชนิดเป็นอย่างดี โดยเฉพาะเครื่องบินที่ทำการบินเป็นประจำตามเส้นทางที่รับผิดชอบในการควบคุมการจราจรอยู่ ข้อมูลเหล่านี้จำเป็นอย่างยิ่งในการควบคุมการจราจร เพื่อที่จะสามารถสั่งควบคุมการจราจรได้อย่างถูกต้อง ในวิทยานิพนธ์นี้จึงได้ใช้ข้อมูลสมรรถนะของเครื่องบินที่ได้ใช้งานจริงในการจำลองการทำงาน ประกอบด้วย

ชื่อฟิลด์	รายละเอียด
PLANETYPE	แสดงชื่อรุ่น หรือชนิดของเครื่องบิน ใช้เป็นตัวอ้างอิงในการค้นหา
TURNRATE	แสดงอัตราการเลี้ยวของเครื่องบิน เป็น องศาต่อวินาที
CLIMBRATE	แสดงระดับอัตราการไต่ความสูงของเครื่องบิน ฟิตต่อนาที
DESCENDRATE	แสดงระดับอัตราการลดความสูงของเครื่องบิน ฟิตต่อนาที
SPEED	แสดงความเร็วในการบิน หน่วยเป็น knot (ไมล์ต่อชั่วโมง)
SPEEDRATELOW	แสดงอัตราการเปลี่ยนความเร็วของเครื่องบินขั้นต่ำ หน่วย knotต่อนาที
SPEEDRATEHIGH	แสดงอัตราการเปลี่ยนความเร็วของเครื่องบินขั้นสูง หน่วย knotต่อนาที

ตารางที่ 6 โครงสร้างแฟ้มข้อมูลสมรรถนะการบินของ เครื่องบิน

ตารางสมรรถนะของ เครื่องบิน จะใช้ชื่อรุ่นหรือชนิดของ เครื่องบินในการอ้างอิง เพื่อค้นหาข้อมูลซึ่งประกอบด้วยอัตราการ เลี้ยว* ส่วนความ เร็วในการบินหมายถึงความ เร็วในการการ เดินทางที่ประหยัดน้ำมัน เชื้อ เพลิงมากที่สุด มีหน่วย เป็นนอต (Knot)**

นอกจากนี้ เครื่องบินยังไม่สามารถที่จะ เร่งความ เร็วหรือลดความ เร็วได้อย่างทันทีทันใด โดยจะมีอัตราการ เปลี่ยนความ เร็วขั้นต่ำและอัตราการ เปลี่ยนความ เร็วขั้นสูง มีหน่วย เป็นนอตต่อนาที

ส่วนอัตราการไต่ความสูง และอัตราการลดระดับความสูงนั้น จะจัด เก็บ เป็นตัชนีตามประ เภทของ เครื่องบิน โดยแบ่ง เครื่องบินออก เป็นกลุ่ม เพื่อชี้ไปยังฐานข้อมูลอีกตัวหนึ่ง เรียกว่าตารางสมรรถนะ เครื่องบินแบ่งตามกลุ่ม ซึ่ง เป็นการจัด เก็บ เป็นกลุ่มตามประ เภทของ เครื่องบินและความสามารถในการ เปลี่ยนระดับการบินโดยที่ เครื่องบินแต่ละลำจะถูกจัดให้อยู่ในกลุ่มต่างๆ ซึ่งได้แบ่งออก เป็น 8 กลุ่มดังนี้

---

* โดยที่ค่าปกติจะ เป็น 3 องศาต่อวินาที เรียกว่า Rate One Turn

** 1 Knot มีค่า เท่ากับ 1 ไมล์ทะ เลต่อชั่วโมง

กลุ่มที่	อัตราการไต่ระดับที่เพดานบินต่ำกว่า 20,000 ฟิต (ฟิตต่อนาที)	อัตราการไต่ระดับที่เพดานบินสูงกว่า 20,000 ฟิต (ฟิตต่อนาที)	อัตราการลดระดับแบบปกติ (ฟิตต่อนาที)	อัตราการลดระดับแบบฉุกเฉิน (ฟิตต่อนาที)
1	4000	4000	3000	5000
2	2500	1500	3000	4000
3	2000	1500	2000	4000
4	2000	1000	1500	2500
5	1500	1500	1500	2000
6	1000	1000	1000	2000
7	1000	500	1000	1500
8	750	500	750	1000

ตารางที่ 7 แฟ้มข้อมูลสมรรถนะการบินของเครื่องบินแบ่งตามกลุ่ม

อัตราการไต่ระดับความสูงจะมีอยู่ 2 ค่าโดยแบ่งที่ระดับความสูง 20,000 ฟิต เมื่อระดับความสูงเพดานบินของเครื่องบินในขณะนั้น อยู่ต่ำกว่า 20,000 ฟิต จะใช้ค่าในช่องที่หนึ่งเป็นอัตราการไต่ความสูง และถ้าระดับเพดานบินสูงกว่า 20,000 ฟิต จะเปลี่ยนไปใช้อัตราในขั้นสูงแทน

ข้อสังเกต โดยปกติค่าอัตราการไต่ความสูงในช่องขั้นต่ำ มักจะมีค่ามากกว่าค่าในช่องขั้นสูง ทั้งนี้เนื่องจากที่ระดับ 20,000 ฟิตลงมา จะเป็นระดับชั้นบรรยากาศปกติซึ่งอากาศมักจะมีความหนาแน่นสูงกว่าที่ระดับบรรยากาศความสูง 20,000 ฟิตขึ้นไป ทำให้เครื่องบินมีอัตราการยกตัวสูงกว่า นอกจากนี้ที่บรรยากาศระดับต่ำมีความแปรปรวนสูง อันเนื่องจากอากาศมีความหนาแน่นไม่เท่ากัน (ซึ่งเป็นผลมาจากการได้รับความร้อนจากแสงอาทิตย์ มีลมพัด หรืออิทธิพลอื่นๆจากบรรยากาศ) ทำให้ไม่เป็นผลดีต่อการทำการบิน ดังนั้นเมื่อนักบินนำเครื่องบิน บินขึ้นจะประกาศให้ผู้โดยสารรัดเข็มขัดอยู่กับที่นั่ง ในระหว่างนี้ก็จะทำการไต่ระดับความสูงเพื่อให้พ้นระดับชั้น

บรรยากาศต่ำโดยเร็ว ซึ่งสังเกตได้ว่าเครื่องบินมีอาการสั่นสะเทือนมาก เมื่อเครื่องบินผ่านขึ้นเหนือระดับความสูง 20,000 ฟิตแล้วจะได้ด้วยอัตราชันน้อยลงและปลดเข็มขัดนิรภัยได้ ซึ่งการทำเช่นนี้ นอกจากจะเป็นผลดีเพราะบินพ้นระดับบรรยากาศที่มีความแปรปรวนสูงโดยรวดเร็วแล้วยังประหยัดน้ำมันเชื้อเพลิงในการทำการบินอีกด้วย (เพราะที่ระดับบรรยากาศต่ำๆที่อากาศมีความหนาแน่นสูงทำให้มีแรงเสียดทานมาก จึงสิ้นเปลืองเชื้อเพลิงมากกว่า) นอกจากนี้การบินที่บรรยากาศระดับสูงจะทำให้บินได้เรียบและนุ่มนวลกว่าอีกด้วย จากนั้นจึงค่อยๆ ทำการปรับระดับเข้าสู่ระดับการบินปกติ(ประมาณ 30,000 ฟิต) เพื่อทำการบินสู่จุดหมายปลายทางต่อไป

ส่วนอัตราการลดระดับความสูงจะมีสองค่าเช่นกันคืออัตราการลดระดับแบบปกติ และการการลดระดับความสูงอย่างรวดเร็ว ซึ่งปกติจะใช้อัตราการลดระดับแบบปกติ นอกจากมีเหตุการณ์ฉุกเฉินจึงจะใช้อัตราการลดระดับแบบฉุกเฉิน (Expedite) ทั้งนี้เนื่องจากการปรับระดับความสูงอย่างรวดเร็วจะมีผลทำให้ผู้โดยสารเครื่องบินปวดหู อันเนื่องจากการปรับความกดดันอากาศภายในแก้วหูอย่างรวดเร็วเกินไป และยังทำให้เกิดสภาวะไร้น้ำหนักบนเครื่องบิน ซึ่งอาจเกิดอันตรายได้จึงไม่นิยมที่จะลดระดับความสูงอย่างรวดเร็ว นอกจากในกรณีจำเป็นเท่านั้น

ตัวอย่าง ตารางข้อมูลสมรรถนะการบินจะมีดังนี้

B707 3 4 3 480 10 26

แสดงว่า

เครื่องบินโบอิ้ง 707 มีอัตราการเลี้ยว 3 องศาต่อวินาที

มีอัตราการไต่ความสูงที่ระดับต่ำกว่า 20,000 ฟิต = 2,000 ฟิตต่อนาที

มีอัตราการไต่ความสูงที่ระดับสูงกว่า 20,000 ฟิต = 1,000 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบปกติ = 2,000 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบฉุกเฉิน = 4,000 ฟิตต่อนาที

อัตราความเร็วประหยัด ที่ 480 ไมล์ทะเลต่อชั่วโมง

อัตราการเปลี่ยนความเร็วขั้นต่ำ 10 ไมล์ต่อนาที

อัตราการเปลี่ยนความเร็วขั้นสูง 26 ไมล์ต่อนาที

B737 3 5 3 420 10 26

แสดงว่า

เครื่องบินโบอิ้ง 737 มีอัตราการเลี้ยว 3 องศาต่อวินาที

มีอัตราการไต่ความสูงที่ระดับต่ำกว่า 20,000 ฟิต = 1,500 ฟิตต่อนาที

มีอัตราการไต่ความสูงที่ระดับสูงกว่า 20,000 ฟิต = 1,500 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบปกติ = 2,000 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบฉุกเฉิน = 4,000 ฟิตต่อนาที

อัตราความเร็วประหยัด ที่ 420 ไมล์ทะเลต่อชั่วโมง

อัตราการเปลี่ยนความเร็วขั้นต่ำ 10 ไมล์ต่อนาที

อัตราการเปลี่ยนความเร็วขั้นสูง 26 ไมล์ต่อนาที

B747 3 3 3 500 10 26

แสดงว่า

เครื่องบินโบอิ้ง 747 มีอัตราการเลี้ยว 3 องศาต่อวินาที

มีอัตราการไต่ความสูงที่ระดับต่ำกว่า 20,000 ฟิต = 2,000 ฟิตต่อนาที

มีอัตราการไต่ความสูงที่ระดับสูงกว่า 20,000 ฟิต = 1,500 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบปกติ = 2,000 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบฉุกเฉิน = 4,000 ฟิตต่อนาที

อัตราความเร็วประหยัด ที่ 500 ไมล์ทะเลต่อชั่วโมง

อัตราการเปลี่ยนความเร็วขั้นต่ำ 10 ไมล์ต่อนาที

อัตราการเปลี่ยนความเร็วขั้นสูง 26 ไมล์ต่อนาที

B74SP 3 2 3 500 10 26

แสดงว่า

เครื่องบินโบอิ้ง 747 แบบลำตัวสั้น มีอัตราการเลี้ยว 3 องศาต่อวินาที

มีอัตราการไต่ความสูงที่ระดับต่ำกว่า 20,000 ฟิต = 2,500 ฟิตต่อนาที

มีอัตราการไต่ความสูงที่ระดับสูงกว่า 20,000 ฟิต = 1,500 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบปกติ = 2,000 ฟิตต่อนาที

มีอัตราการลดระดับความสูงแบบฉุกเฉิน = 4,000 ฟิตต่อนาที

อัตราความเร็วประหยัด ที่ 500 ไมล์ทะเลต่อชั่วโมง

อัตราการเปลี่ยนความเร็วขั้นต่ำ 10 ไมล์ต่อนาที

อัตราการเปลี่ยนความเร็วขั้นสูง 26 ไมล์ต่อนาที

ข้อมูลที่เก็บในตารางสมรรถนะการบินของเครื่องบินนี้ จะมีเครื่องบินแต่ละรุ่นที่ทำการบินในเขตประเทศไทยเป็นประจำประมาณ 34 รุ่น เพื่อใช้เป็นข้อมูลในการคำนวณการบินต่อไปข้อมูลเหล่านี้มีความสำคัญมาก เพราะถ้าผู้ควบคุมจราจรทางอากาศไม่ทราบข้อมูลสมรรถนะของเครื่องบินที่กำลังทำการควบคุมอยู่ อาจสั่งการควบคุมผิดพลาดได้ เช่น ผู้ควบคุมฯ สั่งให้เครื่องบินโบอิ้ง 747 ไต่ระดับความสูงด้วยอัตราการไต่ 5000 ฟิตต่อนาที เพื่อปรับระดับการบิน ซึ่งในกรณีอย่างนี้ไม่สามารถปฏิบัติได้ เพราะเครื่องบินโบอิ้ง 747 เป็นเครื่องบินโดยสารขนาดใหญ่สามารถบรรทุกผู้โดยสารและสัมภาระได้มาก โดยจะมีอัตราการไต่ความสูงประมาณ 2500 ฟิตต่อนาที เท่านั้น ซึ่งจะทำให้เกิดการผิดพลาดจากที่ได้สั่งควบคุมการจราจรไว้

บทที่ 4

โปรแกรมจำลองข้อมูลการบิน

โปรแกรมจำลองข้อมูลการบิน(นักบินสมมุติ) ทำหน้าที่คำนวณตำแหน่งการ เคลื่อนที่ของเครื่องบินโดยอัตโนมัติทุกๆวินาที ซึ่งค่าที่คำนวณได้จะถูกส่งไปยังโปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์ต่อไป การคำนวณทั้งหมดจะใช้ข้อมูลสมรรถนะการบินของ เครื่องบินที่ใช้งานจริงทั้งหมด เพื่อให้การจำลองการบินใกล้ เคียงความ เป็นจริงมากที่สุด โดยโปรแกรมนักบินสมมุติจะอ่านข้อมูลที่ใช้ทำการฝึกมาจากแฟ้มข้อมูลแบบฝึกหัดการบิน เพื่อใช้ เป็นค่า เริ่มต้นในการจำลองการบิน นอกจากนี้ยังสามารถรับคำสั่งในการควบคุม เครื่องบินจากครูผู้ฝึกสอน เพื่อ เข้าทำการควบคุม เองได้ด้วยแทนที่จะ เป็นการบินตาม เส้นทางบินโดยอัตโนมัติ นอกจากนี้ยังจะแสดงผลการทำงานและข้อมูลการบินของ เครื่องบินแต่ละลำตลอด เวลา เพื่อให้ผู้ฝึกสอนสามารถใช้ เป็นข้อมูลในการรายงานสภาพการบินในขณะนั้นแก่ผู้ควบคุมจราจรทางอากาศได้โดยสะดวก

การทำงานของนักบิน

นักบินจะ เป็นผู้ควบคุม เครื่องบินตั้งแต่ เริ่มออก เดินทาง จนกระทั่ง เครื่องบินลงจอดยัง ณ สนามบินปลายทาง หรือบินออกนอก เขตควบคุมการบินออกไป โดยทำการบินตามทิศทางและ เส้นทางการบินที่ได้ระบุไว้ในแผนการบิน นอกจากนี้นักบินยังมีหน้าที่ในการดูแลความปลอดภัยของ เครื่องบินและผู้โดยสาร อีกทั้งมีอำนาจในการตัดสินใจสูงสุดบน เครื่องบิน เพื่อให้สามารถนำ เครื่องบิน บินสู่จุดหมายปลายทางได้อย่างปลอดภัย ซึ่ง เที่ยวบินต่างๆจะประกอบด้วย

1. <u>เที่ยวบินขาออก</u> เริ่มตั้งแต่ก่อนออก เดินทางนักบินจะไปรายงานตัวที่ห้องต้นหน ณ สนามบินที่จะออก เดินทางประมาณ 1-2 ชั่วโมง ก่อนที่จะออก เดินทางจริง เพื่อกรอกข้อมูลแผนการบิน เที่ยวบินนั้น ซึ่งจะมีข้อมูล เกี่ยวกับ เที่ยวบินนั้นทั้งหมด เช่นชื่อ เที่ยวบิน รหัส เที่ยวบิน

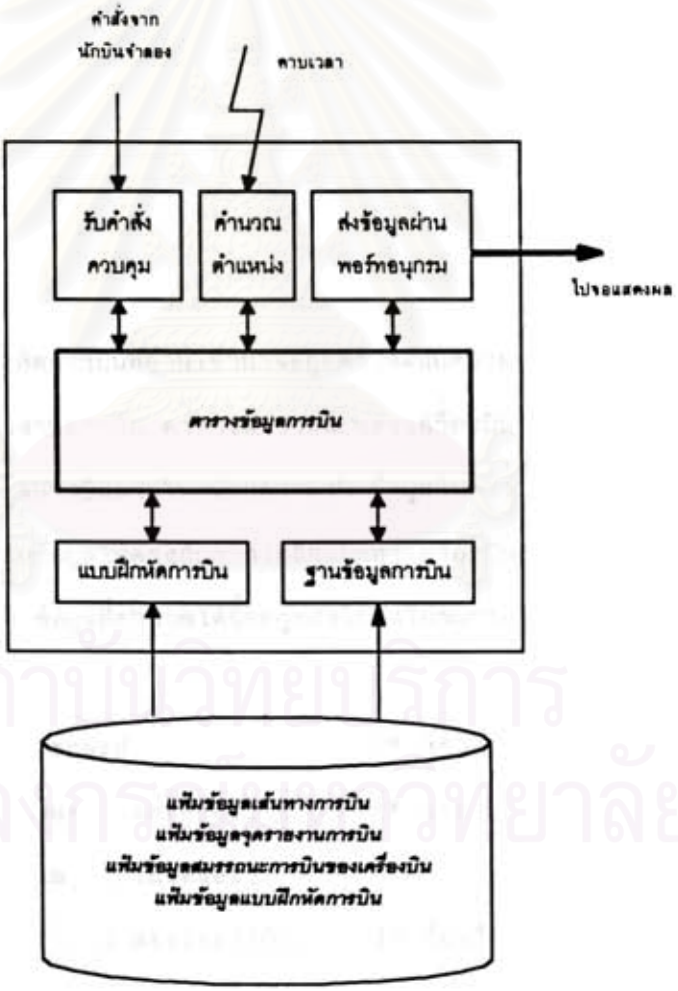
แบบหรือรุ่นของเครื่องบิน เวลาออกเดินทางโดยประมาณ ความเร็วในการบิน ระดับเพดานบิน เส้นทางการบินและสนามบินปลายทาง เป็นต้น ข้อมูลแผนการบินเหล่านี้จะถูกกระจายส่งไปยังผู้เกี่ยวข้องทั้งหมด รวมถึงผู้ควบคุมจราจรทางอากาศในเขตควบคุมการบินตามเส้นทางการบินที่เกี่ยวข้องด้วย ผ่านทางศูนย์สื่อสารการบินระหว่างประเทศ เพื่อใช้เป็นข้อมูลในการควบคุมจราจรทางอากาศ

เมื่อได้เวลาที่เที่ยวบินพร้อมที่จะเดินทางแล้ว นักบินจะเรียกไปยังหอบังคับการบินเพื่อขออนุญาตทำการบินขึ้นจากสนามบิน ซึ่งทางหอบังคับการบินจะตรวจสอบสภาพการจราจรทางอากาศในขณะนั้น รวมถึงเส้นทางการบินตลอดเส้นทางกับศูนย์ควบคุมจราจรทางอากาศ ซึ่งถ้าไม่มีข้อขัดข้องใดๆทางหอบังคับการบินจะอนุญาตให้ทำการบินขึ้นได้ นักบินจะนำเครื่องบินไปยังหัวสนามบินบริเวณทางวิ่งและทำการบินขึ้น พร้อมไต่ระดับขึ้นสู่ระดับเพดานบินตามทิศทางในเส้นทางการบินที่กำหนด ซึ่งถ้าไม่มีเหตุการณ์ใดๆ เปลี่ยนแปลง นักบินก็จะทำการบินตามเส้นทางจนบินลง ณ สนามบินปลายทาง หรือออกนอกเขตการควบคุมไป โดยจะมีการติดต่อประสานงานระหว่างศูนย์ควบคุมจราจรทางอากาศที่อยู่ใกล้เคียง เพื่อให้เที่ยวบินนั้นเดินทางได้อย่างปลอดภัยตลอดเส้นทาง

2. <u>เที่ยวบินขาเข้า</u> ก็เช่นเดียวกันกับเที่ยวบินขาออก โดยข่าวการบินจะถูกส่งมาถึงล่วงหน้าก่อนที่เที่ยวบินจะเดินทางมาถึง ผู้ควบคุมจราจรทางอากาศจะจัดการจราจรให้ตามเส้นทางการบินที่ระบุมาในข่าวการบิน(ถ้าสามารถทำได้) หรือถ้ามีปัญหาก็จะจัดหาเส้นทางการบิน และระดับความสูงที่เหมาะสมให้ จนเมื่อเที่ยวบินเข้ามาในเขตควบคุมผู้ควบคุมจราจรทางอากาศก็จะควบคุมเครื่องบินให้บินตามเส้นทางการบินที่ได้จัดเตรียมไว้

ตามปกติในขั้นตอนการบินขึ้นลงนักบินจะเป็นผู้ควบคุมเครื่องบินเอง โดยเมื่อยังอยู่ในเขตรับผิดชอบของสนามบินนักบินจะประสานงานโดยตรงกับหอควบคุมการบิน และหลังจากเมื่อเครื่องบินไต่สู่ระดับเพดานบินและเข้าเส้นทางบินเรียบร้อยแล้วจะเปลี่ยนไปใช้การบินแบบนักบินอัตโนมัติแทน ทั้งนี้เพราะว่าเที่ยวบินที่ใช้ระยะเวลาเดินทางนานเช่นการบินข้ามทวีปจะใช้เวลาบินกว่า 10 ชั่วโมงขึ้นไป และใช้เพดานบินสูงเหนือชั้นบรรยากาศระดับต่ำ(ประมาณ 30,000 ฟิตขึ้นไป ซึ่งมีความแปรปรวนต่ำและไม่ค่อยมีการรบกวนจากชั้นบรรยากาศ อีกทั้งคอมพิวเตอร์

ที่ทำหน้าที่ควบคุมการบิน มีความแม่นยำสูงขึ้นมากสามารถไว้วางใจได้ โดยระบบเตือนภัยจะทำการเตือนนักบิน เมื่อมีเหตุการณ์ผิดปกติเกิดขึ้น นักบินเพียงแต่ทำหน้าที่ตรวจสอบเส้นทางการบิน รายงานการบินกับศูนย์ควบคุมจราจรทางอากาศและคอยแก้ไขสถานการณ์เมื่อมีเหตุการณ์ผิดปกติเกิดขึ้นเท่านั้น



รูปที่ 5 โปรแกรมจำลองข้อมูลการบิน

การออกแบบโปรแกรมจำลองข้อมูลการบิน

โปรแกรมนักบินสมมุติรับข้อมูลแผนการบินเบื้องต้น มาจากแฟ้มข้อมูลแบบฝึกหัดการบิน ซึ่งจะจัดเก็บลงในตารางข้อมูลการบิน โดยใช้เวลาขึ้นบินเป็นตัวกำหนดจังหวะในการทำงาน นอกจากนี้ยังสามารถที่จะกำหนดค่าคาบเวลาเพื่อสร้างฐานเวลาสำหรับใช้ในระหว่างทำการฝึก สามารถที่จะเร่งความเร็วนาฬิกา ลดความเร็ว หรือแม้แต่กระทั่งหยุดเวลาด้วยการกำหนดคาบเวลามากน้อยตามต้องการ อีกทั้งยังสามารถรับคำสั่งจากผู้ฝึกสอนผ่านทางแป้นพิมพ์ซึ่งโปรแกรมจะนำคำสั่งนั้นมาตีความหมายและปฏิบัติตาม ถ้ามีความผิดพลาดเกิดขึ้นก็จะแสดงข้อความแจ้งสาเหตุของความผิดพลาดนั้น

ตารางข้อมูลการบิน

ข้อมูลแบบฝึกหัดการบินที่อ่านเข้ามาจะถูกตรวจสอบความถูกต้องกับข้อมูลในฐานข้อมูล เส้นทางการบิน จุดรายงานการบิน ตารางสมรรถนะของเครื่องบิน เมื่อทุกอย่างถูกต้องจะนำไปจัดเก็บในตารางข้อมูลการบินภายในหน่วยความจำ ข้อมูลในตารางข้อมูลการบินจะถูกคำนวณใหม่ทุกๆวินาที เพื่อปรับข้อมูลให้ตรงกับการเคลื่อนที่ของเครื่องบินตามจริงโดยอาศัยข้อมูลจากตารางสมรรถนะการบิน ข้อมูลที่คำนวณได้นี้จะถูกส่งไปยังโปรแกรมในส่วนแสดงผล เพื่อแสดงผลต่อไป โดยมีโครงสร้างข้อมูลดังนี้

```
PlaneRec     = record                  (* โครงสร้างข้อมูลการบิน        *)
   estimatehour : integer;             (* เวลาที่เครื่องบินเริ่มเดินทาง  *)
   estimatemin  : integer;
   callsign     : string[10];          (* ชื่อเรียกขาน                  *)
   planetype    : string[10];          (* แบบ/ชนิดของเครื่องบิน         *)
   squawk       : string[10];          (* รหัสประจำเที่ยวบิน            *)
   ident        : boolean;             (* ยังไม่ได้ใช้งาน               *)
   autoheading  : boolean;             (* ตั้งทิศทางการบินอัตโนมัติ     *)
```

```
    headingnow    : integer;        (* ทิศทางการบิน ปัจจุบัน          *)
    setheading    : integer;        (* ทิศทางการบินที่กำหนด           *)
    headingrate   : integer;        (* อัตราการเลี้ยว                 *)
    setheadingLR  : char;           (* ทิศทางการเลี้ยว (ซ้าย/ขวา)     *)
    sethold       : boolean;        (* การบินวนรอบเมื่อกำหนด          *)

    autospeed     : boolean;        (* ตั้งความเร็วแบบอัตโนมัติ       *)
    speednow      : real;           (* ความเร็วในขณะนั้น              *)
    setspeed      : integer;        (* ความเร็วที่กำหนด               *)
    speedrate     : integer;        (* อัตราการเปลี่ยนความเร็ว        *)

    autoaltitude  : boolean;        (* ตั้งระดับเพดานบินอัตโนมัติ     *)
    altitudenow   : real;           (* ระดับเพดานบินขณะนั้น           *)
    setaltitude   : real;           (* ระดับเพดานบินที่กำหนด          *)
    altituderate  : integer;        (* อัตราการไต่ความสูง             *)

    startpoint    : string;         (* จุดเริ่มต้นทำการบิน            *)
    flightroute   : string;         (* เส้นทางการบิน                  *)
    destination   : string;         (* จุดสิ้นสุดการบิน               *)

    status        : char;           (* สถานะของเครื่องบิน             *)
    elapsedtime   : integer;        (* เวลาที่ใช้ทำการบิน (วินาที)    *)
    latnow        : real;           (* ตำแหน่ง latitude เริ่มต้น      *)
    longnow       : real;           (* ตำแหน่ง longitude เริ่มต้น     *)

    next          : PlaneRecPTR;    (* ตัวชี้ไปยัง record ถัดไป       *)
end;
```

นักบินอัตโนมัติ

โปรแกรมนักบินสมมุติ สามารถจำลองการบินตามเส้นทางการบิน ได้อย่างอัตโนมัติ จึงต้องจัดเก็บข้อมูลเส้นทางการบินไว้ในอีกตารางหนึ่ง เพื่อเก็บข้อมูลที่กำหนดจุดเปลี่ยนเส้นทางทางการบิน รวมทั้งทิศทางการบินที่จะต้องทำการเลี้ยวไปยังเส้นทางการบินใหม่ เรียกว่าตารางนักบินอัตโนมัติ โดยใช้โครงสร้างข้อมูลดังนี้

```
AutoRec     = record
    squawk        : string[10];  (* รหัสเที่ยวบิน                    *)
    assignedlat   : real;        (* จุดlatitude เปลี่ยนเส้นทางบิน   *)
    assignedlong  : real;        (* จุด longitude เปลี่ยนเส้นทางบิน *)
    heading       : boolean;     (* ทิศทางบินสู่เส้นทางบินใหม่       *)
    next          : AutoRecPTR;  (* ชี้ไปยังเรคคอร์ดถัด              *)
end;
```

ข้อมูลนี้จะใช้ในการบังคับให้เครื่องบินทำการบินตามเส้นทางไปยังปลายทางที่ได้ระบุไว้โดยอัตโนมัติ โปรแกรมจะทำการตรวจสอบตำแหน่งเครื่องบินทุกๆครั้งที่มีการคำนวณตำแหน่งเครื่องบินใหม่ระหว่างตำแหน่งปัจจุบันกับตำแหน่งที่ได้คำนวณไว้ เมื่อตรวจสอบพบว่าเป็นเที่ยวบินที่รหัสเที่ยวบินและตำแหน่งของเครื่องบินตรงกับที่ได้ระบุไว้ ก็จะนำทิศทางการบินที่ได้คำนวณไว้นำไปใส่แทน เพื่อบังคับให้เครื่องบินเลี้ยวไปยังเส้นทางที่ได้ระบุไว้ โดยที่ข้อมูลนี้ได้ถูกคำนวณทิศทางไว้ล่วงหน้าในขั้นตอนการอ่านข้อมูลแบบฝึกหัดการบิน

อิทธิพลของลม

กระแสลมจะมีส่วนในการทำให้เครื่องบินบินผิดทิศทางที่ควรจะเป็น ทั้งๆที่ได้หันหัวไปในทิศที่ถูกต้องแล้วในลักษณะเดียวกับการพายเรือตัดกระแสน้ำ ดังนั้นจึงต้องเอาอิทธิพลของลมมาคำนวณร่วมด้วย เพื่อให้การจำลองการเคลื่อนที่ของเครื่องบินได้อย่างสมบูรณ์ โดยจะจัดเก็บสภาวะลมที่ระดับทุกๆชั้นความสูง 10,000 ฟิตทั้งความเร็วลมและทิศทางการพัดในตารางข้อมูล

กระแสลม ซึ่งมีโครงสร้างข้อมูลเป็นแบบแถวลำดับตั้งแต่ระดับความสูง 0-100,000 ฟิต โดยมีรายละเอียด ดังนี้

```
WindRec = record
     speed : integer;          (* ความเร็วของลม *)
     angle : integer;          (* ทิศทางการพัด  *)
end;
```

ข้อมูลกระแสลมที่จัดเก็บลงในตารางข้อมูลกระแสลม ประกอบด้วยข้อมูลความเร็วลมและทิศทางการพัดของลมที่ระดับความสูงต่างๆ ดังนั้นในการคำนวณการเคลื่อนที่ของเครื่องบินจะต้องนำอิทธิพลของลมมาคำนวณด้วยโดยขึ้นกับระดับเพดานบินในขณะนั้น

<u>การกำหนดค่าเริ่มต้นให้กับตารางข้อมูลการบิน</u>

ข้อมูลการบินที่ได้จากแฟ้มข้อมูลแบบฝึกหัดการบิน จะนำไปเป็นข้อมูลเริ่มต้นให้กับตารางข้อมูลการบินดังนี้

ชื่อฟิลด์ estimatehour	เวลาที่เครื่องบินเริ่มเดินทาง ชั่วโมง
ชื่อฟิลด์ estimatemin	เวลาที่เครื่องบินเริ่มเดินทาง นาที
ชื่อฟิลด์ callsign	ชื่อเรียกขาน
ชื่อฟิลด์ planetype	แบบ/ชนิดของเครื่องบิน
ชื่อฟิลด์ squawk	รหัสประจำเที่ยวบิน
ชื่อฟิลด์ headingnow	ทิศทางการบิน ปัจจุบัน
ชื่อฟิลด์ speednow	ความเร็วในขณะนั้น
ชื่อฟิลด์ altitudenow	ระดับเพดานบินขณะนั้น
ชื่อฟิลด์ startpoint	จุดเริ่มต้นทำการบิน
ชื่อฟิลด์ flightroute	เส้นทางการบิน
ชื่อฟิลด์ destination	จุดสิ้นสุดการบิน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ข้อมูล เส้นทางการบินที่ระบุในแผนการบินจากแบบฝึกหัดการบิน จะนำมาตรวจสอบความถูกต้องกับฐานข้อมูล เส้นทางการบินตลอดทั้ง เส้นทาง เพื่อ เป็นการยืนยันความสมบูรณ์ของแบบฝึกหัด โดยการตรวจสอบ เริ่มจากจุด เริ่มต้นทำการบิน และ เส้นทางการบิน เพื่อหาจุดสิ้นสุดใน เส้นทางการบินในช่วงแรก เมื่อตรวจสอบพบก็จะนำตำแหน่งพิกัด เริ่มต้นไปคำนวณหาทิศทางที่จะต้องทำการบิน(ในช่วงการ เดินทางนี้) เมื่อตรวจสอบว่า เส้นทางบินถูกต้องแล้วก็จะ เก็บลงในตารางนักบินอัตโนมัติ เพื่อใช้ เป็นข้อมูลในการบังคับให้ เครื่องบินบินตาม เส้นทางที่ได้กำหนดไว้โดยอัตโนมัติ โดยมีโครงสร้างข้อมูลดังนี้

ชื่อฟิลด์ squawk	รหัส เที่ยวบิน
ชื่อฟิลด์ assignedlat	ตำแหน่ง Latitude จุดรายงานการบิน
ชื่อฟิลด์ assignedlong	ตำแหน่ง Longitude จุดรายงานการบิน
ชื่อฟิลด์ heading	ทิศทางการบิน
ชื่อฟิลด์ next	Pointer เพื่อชี้ไปยัง เรคคอร์ดถัดไป

นอกจากนี้ ข้อมูลตำแหน่งพิกัดของจุด เริ่มต้นทำการบิน จะนำไป เก็บในตารางข้อมูลการบิน เพื่อ เป็นข้อมูลตำแหน่งปัจจุบันของ เที่ยวบินนั้นอีกด้วย

ชื่อฟิลด์ latnow	พิกัด เริ่มต้นทางด้านแนวตั้งของ เครื่องบิน
ชื่อฟิลด์ longnow	พิกัด เริ่มต้นทางด้านแนวนอนของ เครื่องบิน

โปรแกรมนักบินสมมุติ สามารถที่จะจำลองสภาพการทำงานของ เครื่องบินแต่ละรุ่นได้ โดยอาศัยข้อมูลสมรรถนะการบินของ เครื่องบินแต่ละรุ่น ในตารางสมรรถนะของ เครื่องบิน เพื่อใช้ในการคำนวณตำแหน่งการ เคลื่อนที่ของ เครื่องบินในแต่ละคาบ เวลา ประกอบด้วย

ชื่อฟิลด์ planetype	แสดงชื่อรุ่น หรือชนิดของ เครื่องบิน
ชื่อฟิลด์ turnrate	แสดงอัตราการ เลี้ยวของ เครื่องบิน เป็น องศาต่อวินาที
ชื่อฟิลด์ climbrate	แสดงระดับอัตราการไต่ความสูงของ เครื่องบิน
ชื่อฟิลด์ descendrate	แสดงระดับอัตราการลดความสูงของ เครื่องบิน
ชื่อฟิลด์ speed	แสดงความ เร็วในการบิน
ชื่อฟิลด์ speedrate	แสดงระดับอัตราการ เปลี่ยนความ เร็วของ เครื่องบิน

ในส่วนของข้อมูลอื่นๆ โปรแกรมจะกำหนดค่าให้โดยอัตโนมัติ ประกอบด้วย

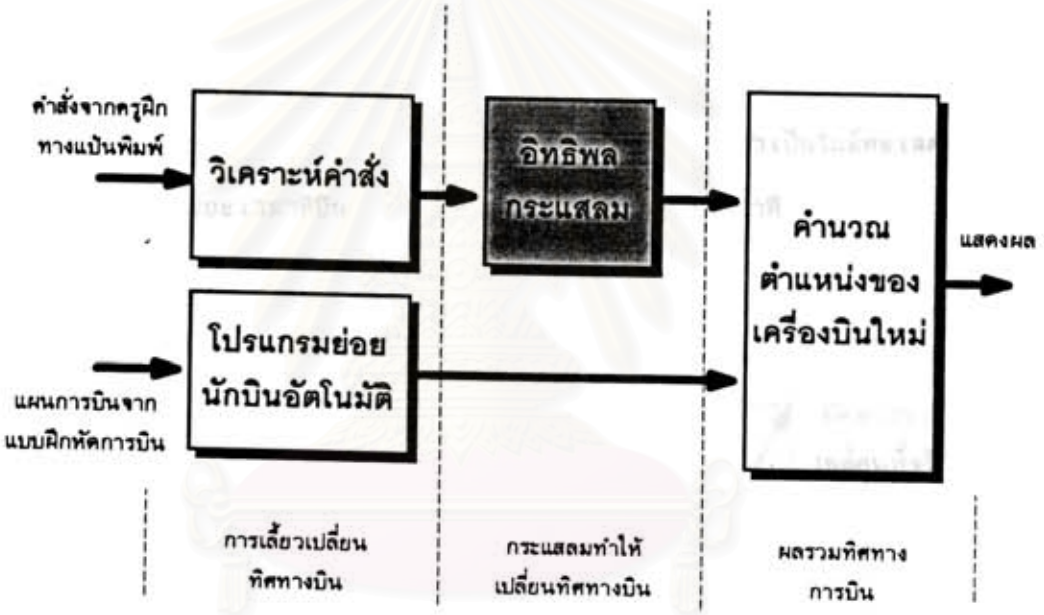
ชื่อฟิลด์ autoheading	มีค่าเป็น TRUE	
ชื่อฟิลด์ autospeed	มีค่าเป็น TRUE	;นักบินอัตโนมัติ
ชื่อฟิลด์ autoaltitude	มีค่าเป็น TRUE	
ชื่อฟิลด์ setheading	มีค่าเท่ากับ headingnow	
ชื่อฟิลด์ setspeed	มีค่าเท่ากับ speednow	
ชื่อฟิลด์ setaltitude	มีค่าเท่ากับ altitudenow	
ชื่อฟิลด์ setheadingLR	มีค่าเท่ากับ "R"	;ปกติเลี้ยวขวา
ชื่อฟิลด์ sethold	มีค่าเท่ากับ "FALSE"	;ยังไม่ HOLD
ชื่อฟิลด์ elapsedtime	มีค่าเท่ากับ 0	; เวลาที่ได้ทำการบินมา
ชื่อฟิลด์ status	มีค่าเท่ากับ "N"	;ยังไม่ขึ้นบิน
ชื่อฟิลด์ next	ให้ชี้ไปยังเรคคอร์ดถัดไป	

<u>การจำลองการทำงานของเครื่องบิน</u>

โปรแกรมนักบินสมมุติ จะทำการคำนวณตำแหน่งของเครื่องบินทุกลำ ที่อยู่ในตารางเก็บข้อมูลการบินทุกๆ 1 วินาที โดยอาศัยข้อมูลสมรรถนะการบิน เพื่อหาตำแหน่งในปัจจุบันโดยมีหลักการดังนี้

1. คำนวณตำแหน่งใหม่จากผลของกระแสลม
2. คำนวณตำแหน่งใหม่จากการเคลื่อนที่ของเครื่องบิน
3. คำนวณทิศทางการบิน
4. คำนวณระดับเพดานบิน
5. คำนวณความเร็ว

การเคลื่อนที่ของเครื่องบินเป็นผลจากแรงสองส่วน คือ แรงเนื่องจากการขับเคลื่อนของเครื่องบินเองและแรงอันเป็นผลจากกระแสลม ดังนั้นในส่วนของการคำนวณเพื่อหาตำแหน่งของเครื่องบิน จะต้องนำอิทธิพลของลมมาคำนวณด้วย ยกเว้นในกรณีที่เป็นการบินแบบใช้นักบินอัตโนมัติ ทั้งนี้เนื่องจากการบินแบบอัตโนมัติ จะมีการแก้ไขชดเชยทิศทางให้โดยอัตโนมัติทำให้เครื่องบินยังคงบินตรงตามเส้นทางอยู่



รูปที่ 6 การคำนวณตำแหน่งของเครื่องบิน

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

การคำนวณตำแหน่งของเครื่องบิน

1. ในกรณีเป็นการควบคุมด้วยครูฝึก ในการคำนวณการเคลื่อนที่ของเครื่องบินจะนำอิทธิพลของกระแสลมมาร่วมคำนวณด้วย โดยถือเสมือนว่าเป็นแรงที่กระทำต่อเครื่องบินอย่างเต็มที่มีผลทำให้เครื่องบินเคลื่อนที่ไปตามทิศทางกระแสลมด้วย ดังนั้นทิศทางการเคลื่อนที่ของ

เครื่องบินรวม คือ การรวมเวกเตอร์ของการเคลื่อนที่ในทิศทางการขับเคลื่อนของเครื่องบิน และการเคลื่อนที่อันเกิดจากกระแสลม โดยความเร็วลมและทิศทางกระแสลมจะต่างกันไปตามชั้นความสูงทุกๆระยะ 10,000 ฟิต ในการคำนวณจึงต้องใช้ความเร็วลม และทิศทางลมที่ตรงกับระดับความสูงที่เครื่องบินทำการบินอยู่ ดังนี้

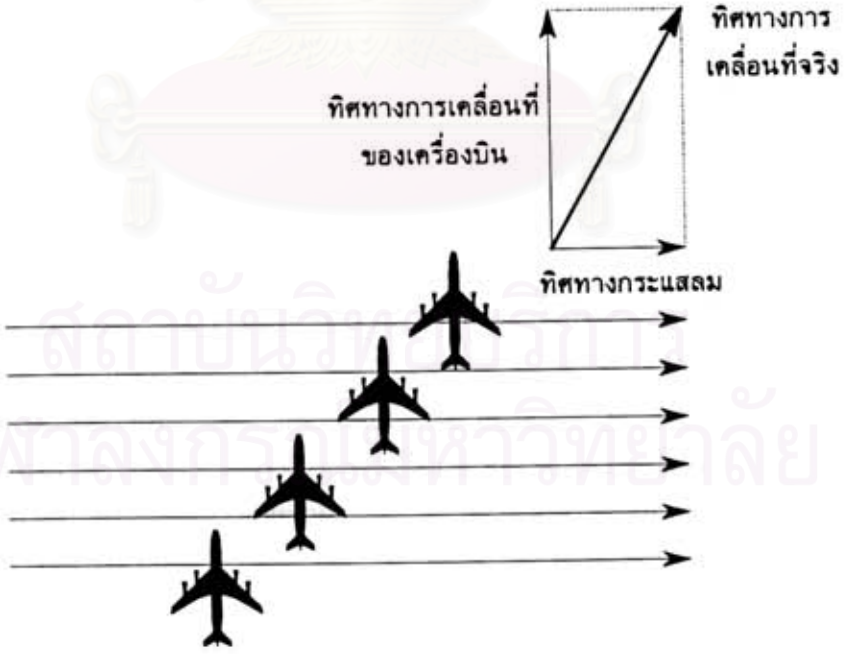
ระยะทางที่ถูกลมพัดไป = ความเร็วลมที่ระดับความสูงนั้นๆ / 3600 * ระยะเวลาที่บิน

โดยที่ ระยะทางที่ถูกลมพัดไป มีหน่วยเป็น ไมล์ทะเล

ความเร็ว มีหน่วยเป็นนอต

จึงหารด้วย 3600 เพื่อทำเป็นไมล์ทะเลต่อวินาที

ระยะเวลาที่บิน ปกติจะคำนวณทุกๆ 1 วินาที



รูปที่ 7 อิทธิพลของกระแสลมที่มีต่อเครื่องบิน

และคำนวณตำแหน่งใหม่ เมื่อคิดอิทธิพลของลมแล้วได้จากตำแหน่งเดิม บวกกับระยะทางที่ถูกลมพัดไป และทิศทางของกระแสลม ดังนี้

Latitudeใหม่ = Latitudeเดิม + ระยะทางที่ลมพัดไป/60*SIN(ทิศทางของลม)
Longitudeใหม่ = Longitudeเดิม + ระยะทางที่ลมพัดไป/60*COS(ทิศทางของลม)

โดยที่ ตำแหน่ง Latitude คือพิกัดทางแนวตั้งของเครื่องบิน มีมีหน่วยเป็นองศา
Longitude คือพิกัดทางแนวนอนของเครื่องบิน มีมีหน่วยเป็นองศา
ระยะทางที่ถูกลมพัดไป มีหน่วยเป็น ไมล์ทะเล
(จึงต้องหารด้วย 60 เพื่อแปลงเป็นองศา)
ทิศทางของลม มีหน่วยเป็นองศา
(จึงต้องแตกออกเป็นในแนวแกน X,Y)

2. <u>ในกรณีที่เป็นการควบคุมทั้งแบบควบคุมเอง และนักบินอัตโนมัติ</u> การเคลื่อนที่ของเครื่องบินอันเป็นผลมาจากการขับเคลื่อนของเครื่องบินเอง จะสามารถคำนวณได้ในลักษณะเดียวกัน จากความเร็วของเครื่องบินและระยะเวลาที่ทำการบิน ดังนี้

ระยะทางที่บิน = ความเร็วในการบิน / 3600 * ระยะเวลาที่บิน

โดยที่ ระยะทางที่บิน มีหน่วยเป็น ไมล์ทะเล
ความเร็ว มีหน่วยเป็นนอต(ไมล์ทะเลต่อชั่วโมง)
จึงต้องหารด้วย 3600 เพื่อทำเป็นไมล์ทะเลต่อวินาที
ระยะเวลาที่บิน ปกติจะคำนวณทุกๆ 1 วินาที

และสามารถคำนวณตำแหน่งใหม่ได้จากตำแหน่งเดิม บวกกับ ระยะทางและทิศทางการบิน ที่ได้ทำการบินไป ดังนี้

Latitudeใหม่ = Latitudeเดิม + ระยะทางที่บิน/60*SIN(ทิศทางการบิน)

Longitudeใหม่ = Longitudeเดิม + ระยะทางที่บิน/60*COS(ทิศทางการบิน)

โดยที่ ตำแหน่ง Latitude คือพิกัดทางแนวตั้งของเครื่องบิน มีมีหน่วยเป็นองศา

Longitude คือพิกัดทางแนวนอนของเครื่องบิน มีมีหน่วยเป็นองศา

ระยะทางที่บิน มีหน่วยเป็น ไมล์ทะเล

(จึงต้องหารด้วย 60 เพื่อแปลงเป็นองศา)

ทิศทางการบิน มีหน่วยเป็นองศา

(จึงต้องแตกออกเป็นในแนวแกน X,Y)

ตำแหน่งที่คำนวณใหม่ได้นี้ จะเป็นตำแหน่งที่ได้คำนวณทั้งผลของการเคลื่อนที่จากตัวเครื่องบินเองและผลกระทบจากอิทธิพลของลมแล้ว ค่าพิกัดตำแหน่งที่คำนวณได้ใหม่นี้คือค่าพิกัดตำแหน่งของเครื่องบินที่ทำการบินไปได้ในช่วงคาบเวลา ซึ่งจะส่งให้ส่วนแสดงผลต่อไป

<u>การปรับทิศทางการบิน</u>

การปรับทิศทางการบิน จะเกิดขึ้นจาก 2 สาเหตุ คือ

1. การปรับทิศทางการบิน อันเนื่องจากการบินแบบใช้นักบินอัตโนมัติซึ่งจะมีผลทำให้เครื่องบินบินตรงตามเส้นทางการบินที่ได้ระบุไว้ และเลี้ยวเมื่อถึงจุดเปลี่ยนเส้นทางการบินเพื่อทำการบินตามเส้นทางการบินใหม่จนกระทั่งถึงปลายทาง การบินในลักษณะนี้จะไม่นำอิทธิพลของลมไปร่วมคำนวณด้วย เพราะกระแสลมจะไม่มีผลกระทบต่อทิศทางการบินของเครื่องบิน เนื่องจากเครื่องบินจะทำการปรับทิศทางการบินของตัวเองเพื่อชดเชยกระแสลมโดยอัตโนมัติ

2. การปรับทิศทางการบินโดยครูผู้ฝึกเป็นผู้บังคับการบินแทนการบินแบบอัตโนมัติ ซึ่งครูผู้ฝึกซึ่งทำหน้าที่เป็นนักบินสมมุติ จะเป็นผู้ควบคุมการบินทั้งหมด ดังนั้นจะนำกระแสลมมาร่วมคำนวณด้วย

ในส่วนของโปรแกรมย่อยปรับทิศทางการบิน จึงได้ออกแบบ เพื่อให้ครอบคลุมทั้งสองกรณี โดยจะทำการตรวจสอบค่าต่างๆที่ได้กำหนดไว้ในตารางข้อมูลการบินในส่วนที่ เกี่ยวข้องกับทิศทางการบินทั้งหมด ดังนี้

```
for all track do
     if sethold = TRUE then
          turn right with rate of turn
     else
          if headingnow <> setheading then
               if มุมต่างระหว่าง(headingnow,setheading) น้อยกว่า
                  rate of turn then
                  headingnow <- setheading
               else
                   if setheadingLR = 'R' then
                        turn right with rate of turn
                   else
                        turn left with rate of turn
                   endif
               endif
          endif
     endif
endfor
```

1. ถ้ากำหนดให้บิน เป็นวงกลมรอคอยก็จะ เลี้ยวขวาด้วยอัตราการ เลี้ยวที่กำหนดโดยการปรับค่าทิศทางการบินใหม่ ด้วยอัตราอัตราการ เลี้ยว

2. หรือถ้าทิศทางที่กำหนด กับทิศทางปัจจุบันยังไม่ เท่ากัน ก็จะตรวจสอบต่อว่า

   2.1 ถ้าทิศที่ยังต่างกันนั้นมีค่าน้อยกว่าอัตราการ เลี้ยว แสดงว่า สามารถ เลี้ยว

ได้ในครั้งเดียวได้ทั้งหมด ก็จะนำค่าทิศที่กำหนดมาเป็นทิศทางปัจจุบันได้เลย

2.2 ถ้าต่างกันมากแสดงว่าไม่สามารถเลี้ยวครั้งเดียวหมดก็จะเลี้ยวด้วยอัตราเลี้ยวที่กำหนด ตามทิศทางการเลี้ยว(ซ้ายหรือขวา) ที่กำหนดแทน

แต่เนื่องจากโปรแกรมนักบินอัตโนมัติจะเป็นผู้กำหนดค่าทิศทางการเลี้ยวมาให้ ดังนั้นโปรแกรมนักบินสมมุติจะพยายามปรับทิศทางการบินให้สอดคล้องกับค่าที่กำหนดเสมอ

การปรับระดับเพดานบิน

ระดับเพดานบิน ได้ถูกกำหนดค่าเริ่มแรกมาจาก การอ่านแบบฝึกหัดเข้ามาเก็บไว้ในตารางการบิน แต่สามารถเปลี่ยนแปลงค่าได้จากคำสั่งของครูผู้ฝึกสอน ดังนั้นจึงได้ออกแบบโครงสร้างข้อมูลในส่วนนี้ ดังนี้

ชื่อฟิลด์ autoaltitude	ใช้อัตราเปลี่ยนเพดานบินตามมาตรฐานหรือไม่
ชื่อฟิลด์ altitudenow	ระดับเพดานบินในปัจจุบัน
ชื่อฟิลด์ setaltitude	ระดับเพดานบินที่ต้องการ
ชื่อฟิลด์ altituderate	อัตราการเปลี่ยนระดับเพดานบิน

โดยที่การปรับระดับเพดานการบิน จะใช้หลักการดังนี้

```
for all track do begin
   ค้นหาตารางแบบ/ชนิดของเครื่องบิน เพื่อหาอัตราการไต่/ลดความสูง
   if autoaltitude = TRUE then
      if altitudenow < setaltitude then
         if altitudenow < 20000 then
               altituderate = อัตราการไต่จากตารางต่ำกว่า 20000 ฟิต
         else
```

```
                altituderate = อัตราการไต่จากตารางสูงกว่า 20000 ฟิต
            endif
        else
            altituderate = อัตราการลดระดับ จากตาราง
        endif
        if ผลต่างของ(altitudenow,setaltitude) น้อยกว่า altituderate then
            altitudenow <- setaltitude
        else
            if altitudenow < setaltitude then
                altitudenow = altitudenow + altituderate/60
            else
                altitudenow = altitudenow - altituderate/60
            endif
        endif
    endif
    if altitudenow <=0 then  delete track
endfor
```

อธิบายการทำงานได้ดังนี้

1. ค้นหาตารางสมรรถนะการบินจากแบบหรือชนิดของเครื่องบิน เพื่อหาอัตราการไต่ระดับ หรือลดระดับความสูง นำไปเก็บในตารางข้อมูลการบิน

2. ถ้าเป็นการบินแบบอัตโนมัติ จะใช้ค่าอัตราการเปลี่ยนระดับเพดานบินจากตารางมาใช้ โดยแยกเป็นกรณี ดังนี้

2.1 การไต่ระดับ ต่ำกว่า 20,000 ฟิต

2.2 การไต่ระดับ สูงกว่า 20,000 ฟิต

2.3 การลดระดับเพดานบิน

3. หรือถ้า เป็นการบินแบบควบคุม เอง ก็จะใช้ค่าอัตราการ เปลี่ยนระดับที่ครูฝึก เป็นผู้กำหนดในการปรับระดับความสูง

4. ปรับระดับการบิน โดยการตรวจสอบว่า

4.1 ถ้าได้ระดับการบินแล้วหรือผลต่างระดับการบิน น้อยกว่าอัตราการ เปลี่ยนระดับ แสดงว่าสามารถปรับระดับ เพดานบินได้ทันที

4.2 หรือมิฉะนั้น จะปรับระดับ เพดานบิน

4.2.1 ถ้า เป็นการไต่ระดับ ก็จะ เพิ่มค่าระดับ เพดานบิน

4.2.2 ถ้า เป็นการลดระดับ ก็จะลดค่าระดับ เพดานบิน

5. ถ้าระดับการบินต่ำกว่าหรือ เท่ากับศูนย์ จะลบข้อมูลการบินทิ้ง เนื่องจาก เครื่องบินได้ตกสู่พื้นดินแล้ว

การปรับระดับความ เร็ว

ความ เร็วของ เครื่องบินแต่ละลำ ได้ถูกกำหนดค่าแรก เริ่ม มาจากการอ่านแบบฝึกหัดการบิน เช่น เดียวกับการกำหนดค่าระดับ เพดานบิน โดยจะถูกกำหนดให้ เป็นการปรับระดับอย่างอัตโนมัติ นอกจากครูผู้ฝึกสอนจะ เข้าทำการควบคุม เอง โดยมีโครงสร้างข้อมูลดังนี้

ชื่อฟิลด์ autospeed	มีการปรับค่าความ เร็วแบบอัตโนมัติ หรือไม่
ชื่อฟิลด์ speednow	ค่าความ เร็วในปัจจุบัน
ชื่อฟิลด์ setspeed	ค่าความ เร็วที่กำหนด
ชื่อฟิลด์ speedrate	อัตราการ เปลี่ยนความ เร็ว

โปรแกรมย่อยปรับระดับความ เร็วจะทำหน้าที่ปรับค่าความ เร็วในการบินให้ตรงกับค่าที่กำหนด โดยใช้หลักการดังนี้

```
for all track do begin
     ค้นหาตารางแบบ/ชนิดของเครื่องบิน เพื่อหาอัตราการเปลี่ยนความเร็ว
     if autospeed = TRUE then
          if speednow < 360 then
               speedrate = อัตราการเปลี่ยนความเร็วแบบต่ำ
          else
               speedrate = อัตราการเปลี่ยนความเร็วแบบสูง
          endif
     endif
     if ผลต่างของ(speednow,setspeed) < speedrate then
          speednow <- setspeed
     else
          if speednow < setspeed then
               speednow = speednow + speedrate/60
          else
               speednow = speednow - speedrate/60
          endif
     endif
     if speednow < 0 then  delete track
endfor
```

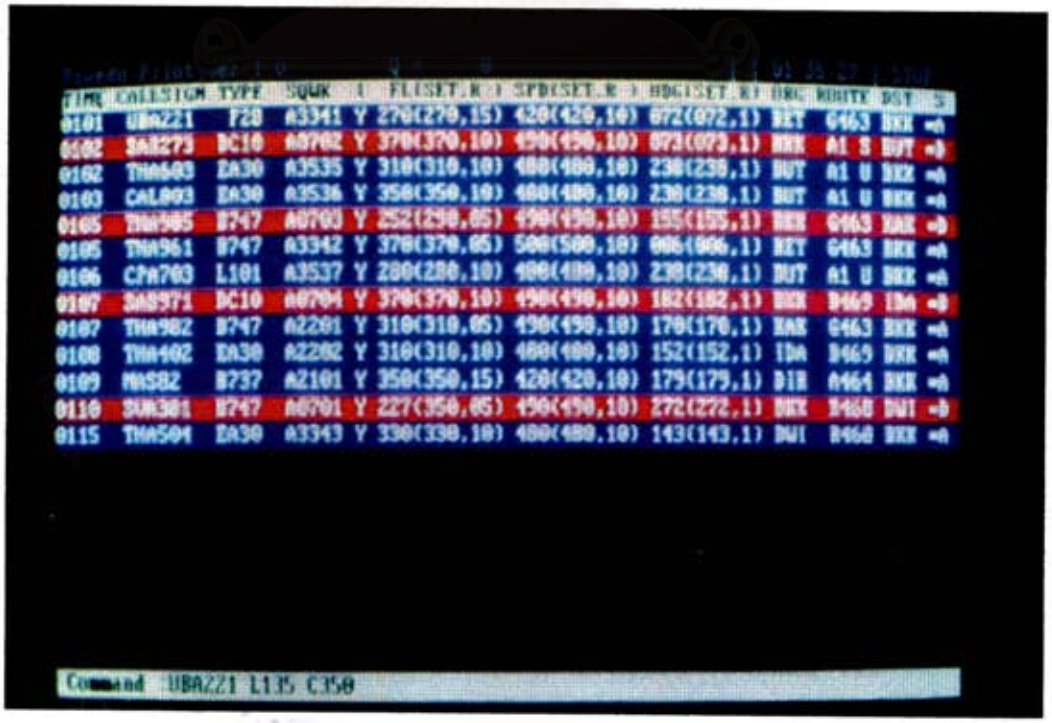
1. ค้นหาอัตราการเปลี่ยนความเร็ว จากตารางสมรรถนะเครื่องบิน
2. ถ้าเป็นการบินแบบนักบินอัตโนมัติ จะ
   2.1 ถ้าความเร็วต่ำกว่า 360 นอต ใช้อัตราการเปลี่ยนแบบต่ำ
   2.2 ถ้าความเร็วสูงกว่า 360 นอต ใช้อัตราการเปลี่ยนแบบสูง
3. ถ้าสามารถเปลี่ยนความเร็วได้ในครั้งเดียว จะเปลี่ยนความเร็วได้เลย
4. ถ้าเป็นการเร่งความเร็ว ให้เพิ่มความเร็วด้วยอัตราการเปลี่ยนความเร็ว
5. ถ้าเป็นการลดความเร็ว ให้ลดความเร็วด้วยอัตราการเปลี่ยนความเร็ว

การแสดงผลการทำงานของโปรแกรมจำลองข้อมูลการบิน

การแสดงผลการทำงานที่หน้าจอของ โปรแกรมจำลองข้อมูลการบิน(นักบินสมมุติ) จะแสดงผลในแบบตัวอักษร เพื่อแสดงข้อมูล เครื่องบินที่กำลังจำลองการบินอยู่และข้อมูลแสดงสถานะอื่นๆ โดยข้อมูล เหล่านี้จะมีการปรับการแสดงผลทุกครั้งที่มีการ เปลี่ยนแปลง เพื่อให้ได้ข้อมูลที่มีความทันสมัยอยู่ตลอด เวลา โดยประกอบด้วย

1. เวลาของระบบ ที่กำลังจำลองการทำงานอยู่โดย เวลาระบบนี้จะถูกส่งออกไปยังจอแสดงผลด้วย เพื่อให้แสดง เวลาตัว เดียวกันทั้งในส่วนของนักบินสมมุติ และผู้รับการฝึกโดยจะปรับการแสดงผลที่หน้าจอทุกๆวินาที

2. ข้อมูลรายละ เอียดของแต่ละ เที่ยวบิน จากที่ได้ เก็บไว้ในตารางข้อมูลการบิน เพื่อที่ครูฝึกจะสามารถใช้ เป็นข้อมูลในการรายงานการบิน กับพนักงานควบคุมจราจรทางอากาศ โดยข้อมูลแต่ละบรรทัดจะแทนข้อมูลของ เครื่องบินแต่ละลำ



รูปที่ 8 หน้าจอของโปรแกรมนักบินสมมุติ

โดยที่มีรายละเอียดของข้อมูลดังนี้

ช่องข้อมูลที่ 1. เป็นเวลาที่เที่ยวบินนั้นเริ่มทำการบิน

ช่องข้อมูลที่ 2. เป็นชื่อเรียกขานของเที่ยวบิน

ช่องข้อมูลที่ 3. เป็นแบบหรือชนิดของเครื่องบิน

ช่องข้อมูลที่ 4. เป็นรหัสประจำเที่ยวบิน

ช่องข้อมูลที่ 5. เป็นระดับเพดานบินในขณะนั้น หน่วยเป็น ร้อยฟิต

ช่องข้อมูลที่ 6. เป็นระดับเพดานบินที่ต้องการ หน่วยเป็น ร้อยฟิต

ช่องข้อมูลที่ 7. เป็นอัตราการเปลี่ยนแปลงระดับเพดานบิน หน่วยเป็น ร้อยฟิตต่อนาที

ช่องข้อมูลที่ 8. เป็นความเร็วของเครื่องบินในขณะนั้น หน่วยเป็น นอต

ช่องข้อมูลที่ 9. เป็นความเร็วของเครื่องบินที่ต้องการ หน่วยเป็น นอต

ช่องข้อมูลที่ 10. เป็นอัตราการเปลี่ยนแปลงความเร็วในการบิน หน่วยเป็น นอตต่อนาที

ช่องข้อมูลที่ 11. เป็นทิศทางการบินในขณะนั้น หน่วยเป็นองศา

ช่องข้อมูลที่ 12. เป็นทิศทางการบินที่ต้องการ หน่วยเป็นองศา

ช่องข้อมูลที่ 13. เป็นอัตราการเลี้ยว หน่วยเป็นองศาต่อวินาที

ช่องข้อมูลที่ 14. เป็นจุดเริ่มต้นทำการบิน จากจุดรายงานการบินใดๆ

ช่องข้อมูลที่ 15. เป็นเส้นทางการบินที่จะทำการบิน

ช่องข้อมูลที่ 16. เป็นจุดสิ้นสุดการควบคุมการบินแบบอัตโนมัติ ณ จุดรายงานใดๆ

ช่องข้อมูลที่ 17. เป็นสถานะของเที่ยวบินนั้นๆ ประกอบด้วย

'A' เป็นเที่ยวบินขาเข้า (Arrival Flight)

'D' เป็นเที่ยวบินขาออก (Departure Flight)

'N' ยังไม่ทำการบิน เนื่องจากยังไม่ถึงเวลาทำการบิน

'*' ทำการบินแบบอัตโนมัติ (Auto-pilot)

ซึ่งข้อมูลแต่ละบรรทัด จะแสดงผลด้วยสีต่างกัน ตามลักษณะของเที่ยวบินดังนี้

เที่ยวบินขาเข้า แสดงผลด้วย ตัวอักษรสีขาว บนพื้นสีน้ำเงิน

เที่ยวบินขาออก แสดงผลด้วย ตัวอักษรสีขาว บนพื้นสีแดง

เที่ยวบินที่ยังไม่ถึงกำหนดเวลาบิน แสดงผลด้วยตัวอักษรสีดำ บนพื้นสีเทา

3. บรรทัดรับคำสั่ง เพื่อรับคำสั่งจากครูผู้ฝึกในการเข้าควบคุมการบิน แทนระบบนักบินอัตโนมัติ เช่นการเลี้ยวซ้ายหรือเลี้ยวขวา การเปลี่ยนระดับเพดานบิน เป็นต้น โดยจะต้องกดแป้นป้อนเข้า (ENTER KEY) เพื่อสั่งให้เริ่มปฏิบัติการได้

4. แสดงข้อความคำสั่งผิดพลาด ที่อาจจะมีจากการรับคำสั่งจากครูฝึก ซึ่งอาจจะเป็นทางด้านไวยากรณ์หรือค่าที่ทำการป้อนไม่ถูกต้อง โดยจะแสดงผลพร้อมทั้งกระพริบเตือนด้วย เพื่อให้ทำการแก้ไขคำสั่งใหม่

5. จำนวนข้อมูลที่ค้างในที่พักข้อมูลชั่วคราวจากการส่งข้อมูลไปยังจอแสดงผล ทั้งนี้เนื่องจากเมื่อจำลองการทำงานของเครื่องบินหลายๆลำทำให้มีข้อมูลที่ต้องรับ-ส่งมาก ซึ่งอาจมีโอกาสที่โปรแกรมไม่สามารถส่งข้อมูลได้ทันและที่พักข้อมูลเกิดการล้น (Queue Over Flow) ได้ ซึ่งโดยปกติค่านี้ควรจะมีค่าเป็นศูนย์ แสดงว่าได้ส่งข้อมูลออกไปทั้งหมดแล้ว ไม่มีข้อมูลใดๆค้างอยู่ในที่พักข้อมูลเลย

<u>การรับคำสั่งควบคุมการบิน</u>

โดยปกติแล้วโปรแกรมจะควบคุมการบินให้บินตามเส้นทางโดยอัตโนมัติ โดยที่ครูผู้ฝึกไม่จำเป็นต้องทำการควบคุมเอง เพื่อเป็นการลดภาระในการทำงานของครูผู้ฝึก เพราะครูฝึกอาจต้องควบคุมเครื่องบินพร้อมกันหลายๆลำ แต่ในบางกรณีครูฝึกอาจจะต้องทำการควบคุมการบินเอง เพื่อปฏิบัติตามคำสั่งในการควบคุมจราจรทางอากาศของผู้รับการฝึก (เช่นเดียวกับที่นักบินจริงจะต้องปฏิบัติ) ดังนั้นโปรแกรมจำลองข้อมูลการบินนี้จะทำการวิ่งวนรอบ เพื่อคอยตรวจจับการกดแป้นพิมพ์อยู่เสมอพร้อมทั้งแสดงคำสั่งที่กำลังป้อน ณ ตำแหน่งที่บรรทัดรอรับคำสั่ง โดยแป้นพิมพ์ที่สามารถใช้ป้อนคำสั่ง ได้แก่

ตัวอักษร A-Z ตัวเลข 0-9 และเครื่องหมาย "/" (สำหรับชื่อเรียกขานบางตัว)

แป้น ESC จะยกเลิกคำสั่งที่กำลังป้อนอยู่ทั้งหมด เพื่อเริ่มต้นป้อนใหม่

แป้น SPACE-BAR ใช้เป็นตัวแยก ระหว่างข้อมูลแต่ละตัว

แป้น BACK-SPACE ใช้เมื่อต้องการลบถอยหลัง ครั้งละ 1 ตัวอักษร

แป้น ENTER ใช้เมื่อจบคำสั่งนั้น โปรแกรมจะนำคำสั่งนั้นไปตรวจสอบและปฏิบัติต่อไป

<u>คำสั่งในการควบคุม</u>

คำสั่งในการควบคุม จะประกอบด้วยคำสั่งอยู่ 2 ประเภทคือ

1. <u>คำสั่งเกี่ยวกับระบบ</u> เป็นคำสั่งที่ใช้สำหรับจัดการเกี่ยวกับระบบ เพื่อเป็นการกำหนดค่าตัวแปรและสภาวะแวดล้อมให้กับโปรแกรม คำสั่งเหล่านี้ได้แก่

1.1 NEW เป็นคำสั่งที่บังคับให้โปรแกรมเริ่มต้นทำงานใหม่รวมทั้งทำการลบตารางข้อมูลการบิน เพื่อเริ่มทำการอ่านแบบฝึกหัดตัวใหม่

1.2 LOAD เป็นคำสั่งในการอ่านข้อมูลแบบฝึกหัดการบินเข้ามาเก็บในตารางข้อมูลการบิน เพื่อที่จะใช้เป็นข้อมูลที่จะทำการจำลองต่อไป คำสั่งนี้ต้องตามด้วยชื่อแฟ้มข้อมูลแบบฝึกหัดการบินที่ต้องการใช้ฝึกและได้จัดเตรียมเก็บไว้แล้ว เช่นคำสั่ง LOAD EX1 เป็นการอ่านแฟ้มข้อมูลแบบฝึกหัดการบินที่ชื่อ"EX1" เข้ามา โดยที่คำสั่งนี้จะไม่ลบข้อมูลที่มีอยู่ในตารางข้อมูลการบินเดิม ดังนั้นจึงสามารถที่จะอ่านแบบฝึกหัดการบินหลายๆตัวมารวมกัน เพื่อทำการจำลองการบินพร้อมกันได้ โดยที่ข้อมูลในแต่ละแบบฝึกหัดจะต้องไม่มีส่วนที่ขัดแย้งกันเอง เช่นต้องไม่มีมีชื่อเรียกขาน หรือรหัสประจำเที่ยวบินซ้ำกัน

1.3 STM เป็นคำสั่งในการตั้งเวลาของระบบ เพื่อที่จะเริ่มจำลองการทำงานต่อไป คำสั่งนี้ต้องตามด้วยค่าเวลาที่ต้องการเริ่มการทำงาน เช่นคำสั่ง STM 010055 หมายถึงเป็นการตั้งให้นาฬิการะบบเริ่มต้นทำงานที่ เวลา 01:00:55

1.4 ENTER เป็นคำสั่งเพื่อป้อนข้อมูลแบบทันที เพื่อที่จะป้อนแผนการบินใหม่ๆได้ทันที โดยไม่ต้องไปสร้างแบบฝึกหัดการบินก่อนทำให้ครูผู้ฝึกสามารถพลิกแพลงหรือเพิ่มเติมเที่ยวบินพิเศษขึ้นมา โดยผู้รับการฝึกไม่ได้เตรียมตัวล่วงหน้าได้ เพื่อเป็นการทดสอบการแก้ไขปัญหาเฉพาะหน้า เช่น

ENTER 0215 MAS331 B747 A0307 350 490 355 HTY A469 BKK

เป็นการป้อนเที่ยวบินพิเศษชื่อมาเลเซีย331 เครื่องบินแบบโบอิ้ง 747 เริ่มบินเวลา 02:15 ใช้รหัสเที่ยวบินว่า เอ0307 บินที่ระดับ 35,000 ฟิต ความเร็ว 490 นอต ถือเข็มทิศ 355 เริ่มบินจากหาดใหญ่ ตามเส้นทางบิน อัลฟา469 มายังดอนเมือง เป็นต้น

1.5 EXIT เป็นคำสั่งในการจบการทำงาน เพื่อกลับไปยังระบบปฏิบัติการ

2. <u>คำสั่งเกี่ยวกับการควบคุมเครื่องบิน</u> เป็นคำสั่งในการบังคับการบิน เครื่องบินแต่ละลำ โดยมีรูปแบบคำสั่งประกอบด้วย ชื่อเรียกขาน และตามด้วยรหัสคำสั่ง พร้อมค่าที่ต้องการ ดังนี้

ชื่อเรียกขาน <รหัสคำสั่ง><ค่าที่ต้องการ>

โดยที่ ชื่อเรียกขาน ได้แก่ ชื่อเรียกขานของเที่ยวบินที่ต้องการควบคุม

รหัสคำสั่ง ได้แก่ รหัสคำสั่งควบคุมที่ต้องการ

ค่าที่ต้องการ ได้แก่ ค่าที่ต้องการควบคุมตามคำสั่งนั้นๆ

คำสั่งในการควบคุมเครื่องบิน ประกอบไปด้วยคำสั่งที่จำเป็นในการบังคับการบิน ซึ่งจะเป็นการบังคับให้เครื่องบินทำการบินตามคำสั่งที่กำหนด ซึ่งจะเป็นการยกเลิกการทำงานแบบนักบินอัตโนมัติและใช้คำสั่งควบคุมนี้แทน รหัสคำสั่งและค่านี้สามารถป้อนหลายคำสั่งได้พร้อมกันในบรรทัดเดียวกัน เพื่ออำนวยความสะดวกในการสั่งงานหลายๆคำสั่งพร้อมๆกันได้ ดังมีรายละเอียดคำสั่งดังนี้

2.1 <u>คำสั่งบังคับการทิศทาง</u> เป็นคำสั่งเพื่อปรับทิศทางการบิน ให้เป็นไปตามตามทิศทางที่ต้องการโดยใช้อัตราการเลี้ยวที่ได้ระบุไว้ คำสั่งนี้จะยกเลิกการบินตามเส้นทางแบบอัตโนมัติ และเปลี่ยนไปบินตามทิศทางที่ได้ระบุไว้แทน

โดยใช้รหัสควบคุม "L" แทนการเลี้ยวไปทางซ้ายมือหรือทวนเข็มนาฬิกา

โดยใช้รหัสควบคุม "R" แทนการเลี้ยวไปทางขวามือหรือตามเข็มนาฬิกา

เช่นคำสั่ง THA603 L230 เป็นคำสั่งให้เที่ยวบินไทยอินเตอร์ 603 เลี้ยวซ้ายถือเข็ม 230

2.2 <u>คำสั่งเปลี่ยนอัตราการเลี้ยว</u> ปกติเครื่องบินทั่วๆไป (โดยเฉพาะเครื่องบินพาณิชย์จะทำการเลี้ยวด้วยอัตราปกติ มีค่าเท่ากับ 3 องศาต่อวินาที) แต่ถ้าในกรณีฉุกเฉินก็สามารถเลี้ยวด้วยอัตรามากกว่านี้ได้ การเปลี่ยนค่าอัตราเลี้ยวนี้จะใช้รหัสคำสั่งว่า "H" ตามด้วยอัตราการเลี้ยวที่ต้องการ เช่นคำสั่ง THA603 H2 หมายถึงให้เที่ยวบินที่ไทยอินเตอร์ 603 ใช้อัตราเลี้ยวเป็น 6 องศาต่อวินาที

2.3 <u>คำสั่งบังคับระดับเพดานบิน</u> เป็นคำสั่งในการบังคับเครื่องบินให้บินที่ระดับความสูงที่กำหนด (เพื่อมิให้มีเครื่องบินทำการบินในเส้นทางเดียวกันที่ระดับความสูงเดียวกัน) แต่โดยที่คำสั่งนี้มีการใช้บ่อยมาก จึงได้ย่อรหัสคำสั่งโดยให้สามารถป้อนตัวเลขความสูงได้เลย

เพื่อให้ประหยัดเวลาในการป้อนรหัสคำสั่ง เช่นคำสั่ง THA603 130 หมายถึงให้เที่ยวบินที่ไทยอินเตอร์ 603 ปรับระดับเพดานการบินไปอยู่ที่ 13,000 ฟิต โดยใช้อัตราการไต่หรือลดระดับเพดานบินตามที่กำหนด

2.4 <u>คำสั่งบังคับอัตราการไต่เพดานบิน</u> เป็นคำสั่งเปลี่ยนแปลงค่า อัตราการไต่ความสูง นอกเหนือจากที่ได้กำหนดไว้ในตารางสมรรถนะเครื่องบิน ใช้รหัสว่า "C" เช่น คำสั่ง THA603 C3000 หมายถึงให้เที่ยวบินที่ THA603 ใช้อัตราการไต่ระดับความสูง 3,000 ฟิตต่อนาที

2.5 <u>คำสั่งบังคับอัตราการลดระดับการบิน</u> เป็นคำสั่งเปลี่ยนแปลงค่า อัตราการลดระดับความสูงให้มีค่าต่างไปจากที่ได้กำหนดไว้ในตารางสมรรถนะการบินใช้รหัสว่า "D" เช่นคำสั่ง THA603 D5000 หมายถึงให้เที่ยวบินที่ไทยอินเตอร์603 ใช้อัตราการลดระดับความสูง 5,000 ฟิตต่อนาที

2.6 <u>คำสั่งบังคับความเร็ว</u> เป็นคำสั่งในการเพิ่ม-ลดความเร็วของเครื่องบินตามอัตราที่ได้กำหนดไว้ คำสั่งนี้มีผลทำให้ความเร็วผิดไปจากที่ได้ระบุไว้ในแผนการบินโดยใช้รหัสว่า "S" เช่นคำสั่ง THA603 S520 หมายถึงให้เที่ยวบินที่ไทยอินเตอร์ 603 เปลี่ยนความเร็วให้เป็น 520 นอต

2.7 <u>คำสั่งเปลี่ยนอัตราการเปลี่ยนความเร็ว</u> เป็นคำสั่งในการเปลี่ยนแปลงค่าอัตราการเปลี่ยนความเร็วจากที่ได้ระบุไว้ตามตารางสมรรถนะเครื่องบิน ทำให้เครื่องบินสามารถเพิ่ม-ลดความเร็วได้ตามอัตราที่กำหนดใช้รหัสคำสั่งว่า "X" เช่นคำสั่ง THA603 X50 หมายถึงให้เที่ยวบินที่ไทยอินเตอร์ 603 ใช้อัตราการเปลี่ยนความเร็วเป็น 50 นอต ต่อนาที

2.8 <u>คำสั่งเปลี่ยนแปลงเวลาเริ่มบิน</u> เป็นการเปลี่ยนแปลงเวลาที่เที่ยวบินนั้นจะเริ่มทำงาน ซึ่งอาจจะเป็นการเลื่อนเวลาให้เร็วขึ้น หรือถอยเวลาให้ช้าลงก็ได้ ซึ่งจะเป็นการสร้างสถานการณ์ให้แก่ผู้รับการฝึกใช้รหัสคำสั่งว่า "T" เช่นคำสั่ง THA603 T0150 หมายถึงให้เที่ยวบินที่ไทยอินเตอร์ 603 เปลี่ยนเวลาออกบินเป็นเวลา 01:50

2.9 <u>คำสั่งบังคับให้บินเป็นวงกลม</u> เป็นคำสั่งให้เครื่องบินทำการบินเป็นวงเพื่อรอคอยการจัดจราจรซึ่งอาจจะเป็นเพราะเครื่องบินเดินทางมาถึงสนามบินปลายทางแล้วแต่ทางวิ่งยังไม่ว่าง เนื่องจากยังมีเที่ยวบินอื่นอยู่ ดังนั้นเครื่องบินจะต้องบินคอยด้วยการบินเป็นวงกลมรอคอย "HOLD" จนกว่าสนามบินจะพร้อมให้ลงได้ โดยใช้รหัสคำสั่งว่า "HOLD" เช่น

คำสั่ง THA603 HOLD เป็นการสั่งให้เที่ยวบินที่ไทยอินเตอร์ 603 บินวนเป็นวงกลมเพื่อรอคอย

2.10 คำสั่งยกเลิกเที่ยวบิน เป็นคำสั่งพิเศษเพื่อยกเลิกเที่ยวบินนั้นๆออกจากตารางข้อมูลการบิน เช่นในกรณีเครื่องบินลำนั้นได้บินออกนอกเขตควบคุมไปแล้ว หรือได้ลงจอดที่สนามบินเรียบร้อยแล้วใช้รหัสคำสั่งว่า "OFF" เช่นคำสั่ง THA603 OFF เป็นการลบเที่ยวบินที่ไทยอินเตอร์ 603 ออกจากตารางข้อมูลการบิน

นอกจากนี้คำสั่งในการควบคุมเครื่องบินทั้งหมด ยังสามารถที่จะสั่งงานในคราวเดียวกันได้ เพื่อเป็นการอำนวยความสะดวกในการควบคุมเครื่องบิน เช่น ถ้าต้องการให้ให้เที่ยวบินที่ไทยอินเตอร์ 603 ลดระดับการบินไปที่ 11,000 ฟิต ลดความเร็วเหลือ 200 นอตเลี้ยวซ้ายถือเข็ม 192 เพื่อลงจอดยังสนามบินจะใช้คำสั่งว่า "THA603 110 S200 L192" <enter>

การส่งข้อมูลไปยังจอแสดงผล

ข้อมูลของเครื่องบินทั้งหมดจะถูกส่งไปแสดงผลที่หน้าจอเรดาร์โดยประกอบด้วยข้อมูลรหัสประจำเที่ยวบิน ระดับความสูงเพดานบิน ความเร็วในการบิน และ ตำแหน่งของเครื่องบิน ข้อมูลของเครื่องบินแต่ละลำนี้จะถูกส่งไปยังจอแสดงผล เพื่อทำการแสดงผลภาพเคลื่อนไหวตามตำแหน่งของเครื่องบินนั้นๆโดยที่จะส่งข้อมูลนี้ทุกๆรอบการหมุนของเรดาร์ และเพื่อให้ส่งข้อมูลได้อย่างสมบูรณ์ จึงได้เพิ่มเติมข้อมูลอื่นๆเข้าไปดังนี้

1. เวลาของระบบ เพื่อให้จอแสดงผลเรดาร์สามารถแสดงเวลาของระบบที่เป็นค่าเดียวกับนาฬิกาที่นักบินสมมุติ เพื่อให้การติดต่อประสานงานกันโดยอ้างอิงกับเวลาได้

2. รหัสตัวอักษร เพื่อบอกถึงจุดเริ่มต้นและจุดสิ้นสุดของกลุ่มข้อมูล ทำให้สามารถแยกข้อมูลของแต่ละกลุ่มออกจากกันได้ ทั้งนี้เพราะใช้การส่งข้อมูลแบบไม่ขึ้นกับเวลา โดยมีรูปแบบดังนี้

```
"<ชั่วโมง นาที> CR LF
<รหัสประจำเครื่องบิน> <ความสูง> <ความเร็ว> <Latitude> <Longitude> CR LF
<รหัสประจำเครื่องบิน> <ความสูง> <ความเร็ว> <Latitude> <Longitude> CR LF
!
```

กลุ่มของข้อมูลจะใช้ตัวอักษร "=" เป็นตัวบอกจุดเริ่มต้น ตามด้วยตัวเลขอีก 4 ตัว เพื่อแทนชั่วโมงและนาทีในขณะนั้น และจบบรรทัดด้วยรหัสจบบรรทัด "CR" และ "LF" ข้อมูลในส่วนของที่เป็นเครื่องบินจะถูกส่งตามมา โดยที่ข้อมูลของเครื่องบินแต่ละลำประกอบด้วย รหัสประจำเครื่องบิน ความสูง ความเร็ว ตำแหน่งพิกัดทางแนวตั้ง และตำแหน่งพิกัดทางแนวนอน ซึ่งข้อมูลแต่ละตัวจะถูกแยกจากกันด้วย ตัวอักษรว่างหนึ่งตัว และปิดท้ายข้อมูลของเครื่องบินแต่ละลำด้วยรหัสจบบรรทัด โดยจะส่งข้อมูลของเครื่องบินนี้ซ้ำในลักษณะนี้จนครบทุกลำและปิดท้ายด้วยตัวอักษรพิเศษ "!" เพื่อเป็นการจบข้อมูลในกลุ่มนี้ เช่น

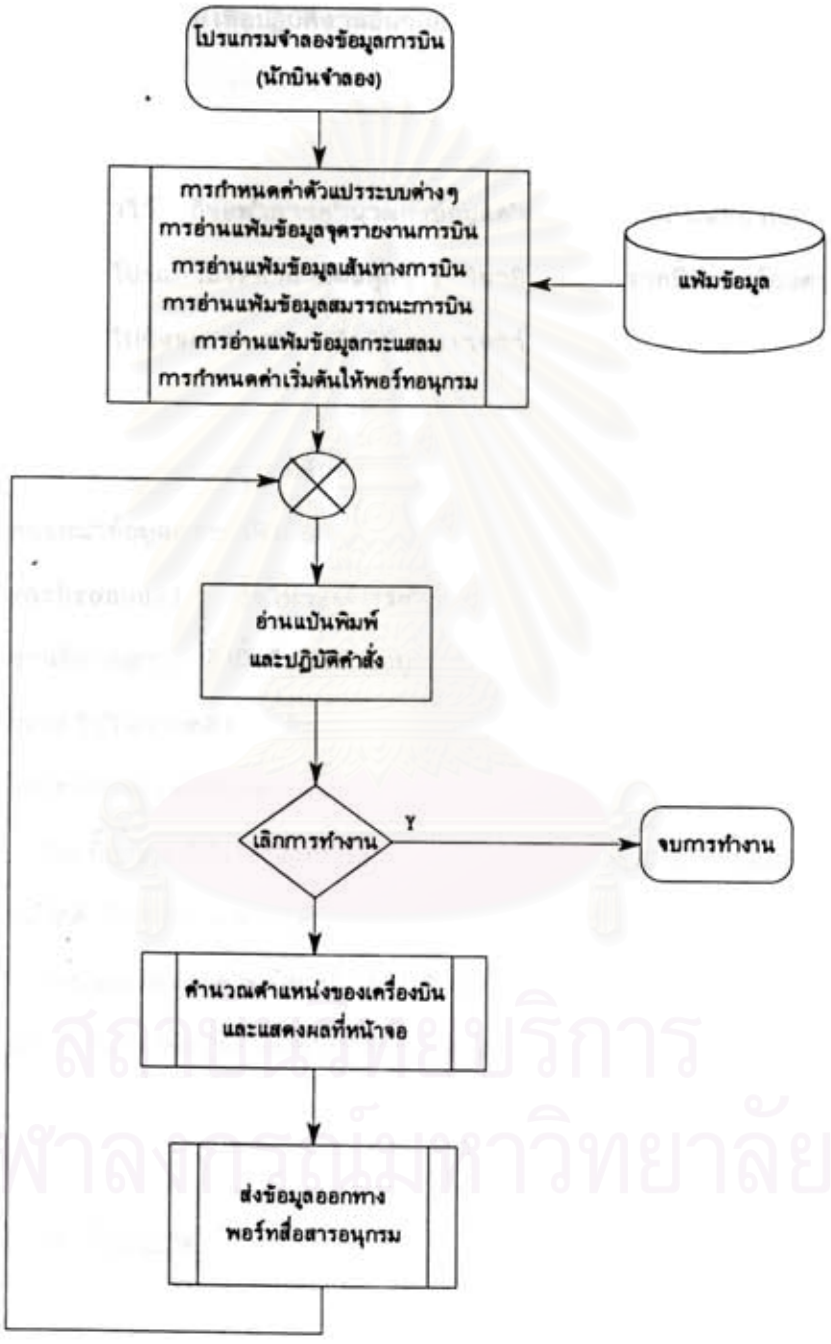
```
=0101 <CR> <LF>
A3341 330 480  5.43210 101.88620 <CR> <LF>
A0307 310 490 11.50000 100.28740 <CR> <LF>
!
```

หมายความว่า ข้อมูลชุดนี้ถูกส่งมาเมื่อเวลา 01:01 โดยมีข้อมูลดังนี้

- เครื่องบินรหัสประจำตัว A3341 ที่ระดับความสูง 33,000 ฟิต ความเร็ว 480 นอต อยู่ที่ตำแหน่งพิกัดแนวตั้ง 5.43210 ตำแหน่งพิกัดแนวนอนที่ 101.88620
- เครื่องบินรหัสประจำตัว A0307 ที่ระดับความสูง 31,000 ฟิต ความเร็ว 490 นอต อยู่ที่ตำแหน่งพิกัดแนวตั้ง 11.50000 ตำแหน่งพิกัดแนวนอนที่ 100.28740

<u>การทำงานของโปรแกรมหลัก</u>

โปรแกรมหลัก ทำหน้าที่ในการกำหนดค่าเริ่มต้นให้กับโปรแกรมใช้เป็นการเริ่มต้นในการทำงาน จากนั้นจึงจะวิ่งวนเพื่อรอรับคำสั่งงานจากผู้ใช้ต่อไป แต่เนื่องจากโปรแกรมจะต้องต้องทำงานแบบประมวลผลทันที (Real Time) ทำให้ระบบไม่สามารถหยุดรอการทำงานที่จุดใดจุดหนึ่งได้ จึงมีจุดน่าสนใจอยู่ตรงที่ "ในขณะที่ระบบรอการกดคีย์จากผู้ใช้จะต้องทำงานอื่นๆไปด้วย เพื่อไม่ให้โปรแกรมหยุดรอเหตุการณ์อย่างใดอย่างหนึ่ง" ดังนั้นในการอ่านค่าแป้นพิมพ์จะต้องตรวจสอบสถานะของแป้นพิมพ์เสียก่อน ถ้ามีการกดแป้นใดแป้นหนึ่งจึงจะอ่านค่าเพื่อนำไปประมวลผลต่อไป ดังนั้นในวงรอบการทำงานหลักจะประกอบด้วย



รูปที่ 9 ผังการทำงานของโปรแกรมจำลองข้อมูลการบิน

1. การตรวจสอบการกดแป้นพิมพ์และปฏิบัติตามคำสั่งงาน โดยจะต้องมีการตรวจสอบสถานะการกดแป้นก่อนเสมอ ถ้ามีการกดจึงจะเรียกโปรแกรมย่อยที่รับค่าจากแป้นพิมพ์มาทำงาน มิฉะนั้นจะผ่านเลยไปเพื่อปฏิบัติงานอื่นๆก่อน

2. ตรวจสอบคาบเวลาเพื่อการคำนวณข้อมูลตำแหน่งการบิน และปรับการแสดงผล ถ้าถึงกำหนดที่ได้ตั้งค่าไว้ ก็จะทำการคำนวณค่าข้อมูลตำแหน่งการบิน และปรับการแสดงผลให้สอดคล้องกัน ซึ่งปกติโปรแกรมจะคำนวณผลทุกๆ 1 วินาที นอกจากนี้ยังจะต้องตรวจสอบคาบเวลาในการส่งข้อมูลไปยังจอแสดงผล เพื่อใช้แทนเรดาร์สวีพ

3. ส่งข้อมูลที่เก็บอยู่ในที่พักข้อมูลชั่วคราวออกทางพอร์ทอนุกรม โดยโปรแกรมย่อยในส่วนนี้ จะทยอยนำข้อมูลการบินที่เก็บอยู่ในที่พักข้อมูลชั่วคราวออกทางพอร์ทอนุกรมแบบไม่เข้าจังหวะ (Asynchronous) ซึ่งในรอบการทำงานปกติโปรแกรมจะพยายามส่งข้อมูลอยู่ตลอดเวลาถ้าไม่มีงานที่สำคัญกว่า ทั้งนี้เนื่องจากข้อมูลที่จะจัดส่งจะถูกจัดเก็บลงในที่พักข้อมูลชั่วคราวก่อนที่จะถูกส่งออกไปในภายหลัง ดังนั้นปริมาณความจุของที่พักข้อมูลจะต้องมีมากเพียงพอที่จะพักข้อมูลได้ทั้งหมดก่อนที่จะทยอยส่งออกจนหมด ซึ่งขึ้นอยู่กับความเร็วในการส่งผ่านข้อมูล และปริมาณข้อมูล มิฉะนั้นจะทำให้เกิดปัญหาข้อมูลบางส่วนสูญหายไปเพราะที่พักข้อมูลเต็ม ดังนั้นในรอบการทำงานปกติ โปรแกรมหลักจะต้องเรียกโปรแกรมย่อยที่ทำหน้าที่ส่งข้อมูลให้บ่อยครั้งที่สุดเท่าที่จะทำได้ โดยจะแสดงจำนวนข้อมูลที่ยังค้างในที่พักข้อมูลที่บรรทัดแสดงสถานะ ซึ่งปกติควรจะมีค่าเป็นศูนย์ นั่นคือไม่มีข้อมูลค้างอยู่ในที่พักข้อมูลเลย

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

<u>การส่งข้อมูลผ่านพอร์ทอนุกรม</u>

ข้อมูลการบินที่โปรแกรมนักบินสมมุติได้คำนวณขึ้นมา จะถูกส่งไปยังจอแสดงผลผ่านทางพอร์ทสื่อสารแบบอนุกรมทุกๆคาบเวลาเรดาร์สวีพ โดยปริมาณข้อมูลที่จะส่งนั้นขึ้นอยู่กับจำนวนเครื่องบินที่กำลังทำการบินอยู่ในขณะนั้น ดังนั้นเมื่อมีจำนวนเที่ยวบินเพิ่มมากขึ้นจะทำให้มีปริมาณข้อมูลที่จะต้องส่งจำนวนมากด้วยซึ่งจะมีผลทำให้โปรแกรมต้องใช้เวลานานในการส่งข้อมูลทั้งหมด

การส่งข้อมูลผ่านพอร์ทอนุกรม จะสามารถทำได้ด้วยโปรแกรม เริ่มจากการกำหนดค่า เริ่มต้นให้แก่อุปกรณ์รับส่งข้อมูล (UART)* แล้วก็จะ เริ่มทำการตรวจสอบค่าสถานะว่าขณะนี้พร้อมที่จะรับข้อมูล เพื่อส่งออกได้หรือไม่ เพราะอาจ เป็นได้ว่ายังส่งข้อมูลตัวที่แล้วไม่หมดสิ้นหรือฝ่ายตรงข้ามยังไม่พร้อม เมื่อตรวจสอบพบว่า เป็นสถานะ เป็นว่างโปรแกรมก็จะสามารถส่งข้อมูลไปที่พักข้อมูลภายในตัวอุปกรณ์รับส่งข้อมูล เอง ซึ่งจะถูกนำส่งออกทางช่องสื่อสารอนุกรมโดยอัตโนมัติ

ข้อจำกัดและการแก้ไขในการส่งข้อมูลผ่านพอร์ทอนุกรม

โปรแกรมนักบินสมมุติจะทำงานในแบบประมวลผลทันที(Real time Processing) ทำให้โปรแกรมไม่สามารถรอคอยให้การส่งข้อมูล เสร็จสิ้นก่อนได้ เพราะว่าถ้ามีข้อมูลที่ต้องจัดส่งมากอาจต้องใช้ เวลาในการส่งข้อมูลนาน เกินไป ซึ่งจะทำให้ไม่สามารถกลับไปทำงานอื่นๆได้ทันจึงมีผลทำให้ดู เหมือนว่าโปรแกรมหยุดนิ่งไป เฉยๆ

ในการแก้ไขปัญหานี้จึงกำหนดให้มีที่พักข้อมูลอยู่ในหน่วยความจำ เพื่อทำการ เก็บข้อมูลที่จะส่งไว้ชั่วคราวก่อน ทำให้โปรแกรมหลักไม่ต้องรอคอยให้ข้อมูลทั้งหมดถูกส่งออกไปทั้งหมดหลังจากนั้นก็จะค่อยๆทยอยส่งข้อมูลในที่พักข้อมูลออกไปคราวละ 1 ตัวอักษร เมื่อโปรแกรมหลัก "ว่าง" จากการทำงานอื่นๆแล้ว ทั้งนี้ เนื่องจากข้อมูลทั้งหมดไม่จำ เป็นต้องถูกส่งออกไปในคราวเดียวกันแต่จะสามารถทยอยส่งได้ เพียงแต่ข้อมูลทั้งหมดจะต้องถูกส่งไปหมดก่อนที่จะครบรอบเรดาร์สวีพ

---

* Universal Asynchronous Receiver Transmitter (UART) เป็นตัวทำหน้าที่ในการติดต่อสื่อสารแบบอนุกรมที่ใช้ใน เครื่องคอมพิว เตอร์ส่วนบุคคลทั่วๆไป

สรุปผลการทำงาน

โปรแกรมนี้จะทำงานในลักษณะประมวลผลทันที (real time) โดยจะสามารถรับข้อมูล เข้าในรูปแบบต่างๆ รวมทั้งคำนวณหาตำแหน่งของ เครื่องบินทุกๆลำด้วยอัตราค่าที่กำหนด ซึ่งค่านี้จะถูกส่งออกไปแสดงผลยังจอแสดงผล เรดาร์ของผู้รับการฝึก เพื่อใช้ในการฝึกการควบคุมจราจรทางอากาศ ระบบนี้สามารถทำงานได้ เป็นอย่างดี ด้วยการออกแบบการรับข้อมูล เข้าให้เหมาะสม ทำให้รับข้อมูลมาประมวลผลได้อย่างทันการณ์ โดยการใช้ที่พักข้อมูลชั่วคราวในการเก็บข้อมูลไว้ก่อนในขณะที่ระบบยังไม่พร้อมที่จะประมวลผล เพื่อรอการประมวลผล เมื่อพร้อมต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 5

โปรแกรมแสดงผลข้อมูลการบินด้วยเรดาร์

โปรแกรมแสดงผลข้อมูลการบินด้วยเรดาร์นี้ เป็นส่วนที่แสดงผลข้อมูลของเครื่องบินที่กำลังทำการบินอยู่ให้ผู้รับการฝึก(ซึ่งทำหน้าที่พนักงานควบคุมจราจรทางอากาศ) ได้รับทราบในรูปแบบการแสดงผลกราฟิกเช่นเดียวกับจอเรดาร์ที่ใช้งานจริง* โดยรับข้อมูลมาจากโปรแกรมจำลองข้อมูลการบินผ่านทางช่องสื่อสารแบบอนุกรม นอกจากนี้ยังสามารถรับคำสั่งในการปรับการแสดงผล หรือคำสั่งอื่นๆจากผู้รับการฝึกได้อีกด้วย ทำให้ผู้รับการฝึกมีความคุ้นเคยในการทำงานกับระบบเรดาร์โดยไม่ต้องไปทำการฝึกกับระบบจริง

หน้าจอที่ใช้ในการแสดงผลจะแบ่งออกเป็นหลายๆส่วน ดังนี้

1. หน้าต่างแสดงชื่อโปรแกรม เพื่อชื่อโปรแกรมและรุ่นของโปรแกรมรวมถึงวันที่ที่แก้ไขครั้งสุดท้าย

2. หน้าต่างแสดงนาฬิการะบบ เพื่อแสดงนาฬิกาของระบบที่ใช้ในการจำลองเหตุการณ์ตามแบบฝึกหัดการบินที่กำลังฝึกบินอยู่ ซึ่งเป็นเวลาตัวเดียวกันกับที่จอของนักบินสมมุติในรูปแบบชั่วโมงและนาทีโดยทำการปรับเวลาในการแสดงผลทุกๆนาที

3. หน้าต่างแสดงสถานะของระบบ เพื่อรายงานค่าสถานะของระบบ รวมทั้งค่าต่างๆที่ป้อนโดยผู้ใช้งานด้วย ประกอบไปด้วย

---

* ซึ่งในปัจจุบันใช้ระบบแสดงผลเรดาร์รุ่นดีดีเอสแปดศูนย์ (DDS-80) ของบริษัทเซเลเนีย (SELENIA) จากประเทศอิตาลี

3.1 ระยะในการแสดงผลของหน้าจอ จากรัศมีในการแสดงผล วัดจากศูนย์กลางการแสดงผล หน่วยเป็นไมล์ทะเล

3.2 แผนที่ต่างๆที่ได้แสดงผลอยู่ในปัจจุบัน โดยผู้ใช้งานสามารถเลือกการแสดงผลเฉพาะบางแผนที่ที่ต้องการได้ ซึ่งประกอบไปด้วย

3.2.1 แผนที่ 1 แสดงผลเส้นขอบเขตควบคุมการบิน

3.2.2 แผนที่ 2 แสดงผลจุดรายงานการบิน

3.2.3 แผนที่ 3 แสดงผลเส้นทางการบิน

3.2.4 แผนที่ 4 แสดงเส้นตารางพิกัดของแผนที่

3.2.5 แผนที่ 5 แสดงวงแหวนบอกระยะทาง มีจุดศูนย์กลางในการแสดงผลเรดาร์ที่ระยะต่างๆ เพื่อใช้ในการประมาณระยะห่างของเครื่องบินที่ปรากฏบนจอถึงจุดศูนย์กลางได้อย่างรวดเร็ว

4. หน้าต่างป้อนข้อมูลล่วงหน้า เป็นหน้าจอที่แสดงชื่อเรียกขาน และรหัสประจำเที่ยวบิน ซึ่งมีประโยชน์สำหรับผู้รับการฝึกในการแยกแยะเครื่องบินที่ตนเองกำลังควบคุมอยู่โดยสามารถกำหนดชื่อเรียกขานให้กับเครื่องบินที่ตนเองควบคุมได้ด้วยการป้อนข้อมูลนี้ไว้ล่วงหน้า

5. หน้าต่างรับคำสั่งจากผู้ใช้ เป็นหน้าจอที่ผู้รับการฝึกใช้ในการป้อนคำสั่งต่างๆ ได้ เช่นคำสั่งปรับมาตราส่วนการแสดงผล คำสั่งวัดระยะทางระหว่างสองจุดใดๆ เป็นต้น โดยที่คำสั่งนี้จะใช้รหัสคำสั่งเดียวกับที่ใช้ในระบบเรดาร์ที่ใช้งานจริง

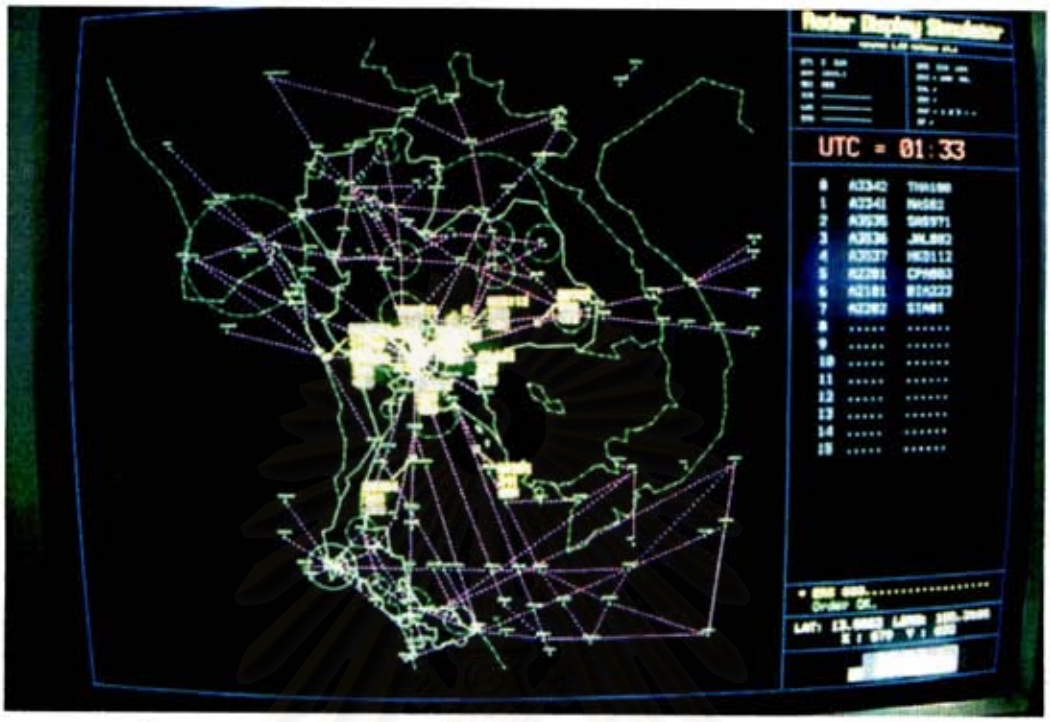
6. หน้าต่างแสดงตำแหน่งพิกัดของตัวชี้ จะแสดงตำแหน่งพิกัดของตัวชี้ตลอดจะแสดงในรูปแบบพิกัดแนวตั้งและแนวนอน รวมทั้งพิกัด X,Y เพื่อใช้ในการอ้างอิงกับหน้าจอแสดงผล

7. หน้าต่างแสดงผลการบิน เพื่อแสดงข้อมูลการบินด้วยเรดาร์ ประกอบด้วย

7.1. แผนที่เส้นขอบเขตควบคุมการบิน ได้แก่ เส้นขอบเขตควบคุมการบินและเขตควบคุมการบินเพิ่มเติม ซึ่งเป็นพื้นที่ที่อยู่ในขอบเขตความรับผิดชอบในการควบคุมจราจรทางอากาศ นอกจากนี้ยังแสดงเส้นแสดงขอบเขตทางภูมิศาสตร์ได้ด้วย เช่น เกาะ เส้นขอบเขตประเทศ อีกด้วย

7.2. แผนที่แสดงเส้นทางการบิน

7.3. แผนที่แสดงจุดรายงานการบิน



รูปที่ 10 หน้าจอของโปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์

7.4. แผนที่แสดงขอบ เขตควบคุมการบินของแต่ละสนามบิน ซึ่งโดยปกติทั่วไป จะมีรัศมีโดยรอบ 35 ไมล์ทะ เล

7.5. แผนที่แสดง เส้นพิกัดของแผนที่

7.6. แผนที่แสดงวงแหวนบอกรัศมี

7.7. สัญญลักษณ์แสดงแทน เครื่องบิน สัญญลักษณ์ที่แสดงแทน เครื่องบินนี้ จะปรากฏบนจอ เพื่อแทนตำแหน่งของ เครื่องบิน นอกจากนี้ยังสามารถแสดงข้อมูลอื่นๆ ที่ เกี่ยวข้อง กับ เครื่องบินลำนั้นได้อีกด้วย ซึ่งประกอบไปด้วย

7.7.1 สัญญลักษณ์แสดงตำแหน่งของ เครื่องบิน โดยจะแสดง เป็นสี่ เหลี่ยม เล็กๆ เพื่อแทนตำแหน่งของ เครื่องบิน

7.7.2 สัญญลักษณ์แสดงตำแหน่งของ เครื่องบินในรอบที่ผ่านมา จะแสดงตำแหน่งพิกัด 4 ตำแหน่งสุดท้ายที่ เครื่องบินได้บินผ่านมา ซึ่งจะมีประโยชน์ในการบอกทิศทางในการบินของ เครื่องบิน

7.7.3 สัญญลักษณ์แสดงรหัสพื้นที่ควบคุมการบิน ซึ่งจะ เป็นตัวอักษรที่ เป็นชื่อใช้แทน เขตควบคุม (Sector) ที่พนักงาน ATC กำลังรับผิดชอบอยู่ เช่นถ้าตัวอักษรนี้ เป็น

'N' หมายความว่า เครื่องบินลำนี้ถูกควบคุมโดย เขตควบคุม 'N' (North) หรือ ถ้าเป็น 'E' หมายความว่า เครื่องบินลำนี้ถูกควบคุมโดย เขตควบคุม 'E' (East) เป็นต้น

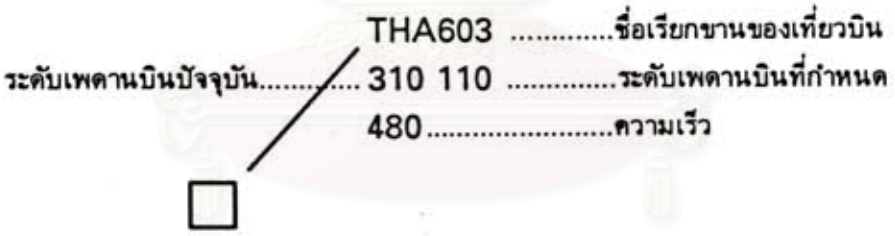
7.7.4 เส้นสำหรับชี้ข้อมูล เครื่องบิน เป็นเส้นที่ใช้ชี้ เชื่อมโยงข้อมูลรายละ เอียดของ เครื่องบินลำนั้นกับสัญญลักษณ์ เครื่องบิน ซึ่งสามารถที่จะ เลื่อนตำแหน่ง เส้นนี้ได้ 8 ทิศรอบตัว เพื่อมิให้บังการแสดงผลของ เครื่องบินลำที่อยู่ใกล้ เคียง

7.7.5 ข้อมูลรายละ เอียดของ เครื่องบิน โดยจะแสดงข้อมูลนี้ดังนี้

บรรทัดบนสุด แสดงชื่อ เรียกขานของ เครื่องบิน (หรือรหัสประจำ เที่ยวบิน ถ้ายังไม่มีการป้อนข้อมูล)

บรรทัดที่สอง แสดงระดับ เพดานบินของ เครื่องบินลำนั้น โดยมีหน่วย เป็นร้อยฟุต และอาจตามด้วยระดับ เพดานบินที่ต้องการ (Assigned Altitude)

บรรทัดที่สาม แสดงความ เร็วของ เครื่องบิน หน่วย เป็นนอต



รูปที่ 11 สัญญลักษณ์ของ เครื่องบินที่ปรากฏบนจอ เรดาร์

7.8. สัญญลักษณ์แสดงตัวชี้บอกตำแหน่ง และ เส้นวัดระยะทาง ในจอ เรดาร์จะมีตัวชี้บอกตำแหน่ง (Cursor) สามารถบังคับการ เคลื่อนที่ได้จาก เมาส์หรือแทรคบอล นอกจากนี้ยังใช้สำหรับวัดระยะทางและมุมระหว่างจุด 2 จุดได้

คำสั่งสำหรับจอแสดงผล

โปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์นี้ ผู้ใช้สามารถป้อนคำสั่งต่างๆได้ผ่านทางหน้าต่างรับคำสั่ง และถ้ามีข้อผิดพลาดก็จะรายงานข้อผิดพลาดนั้นให้ทราบ เพื่อให้ผู้ใช้ทำการป้อนใหม่ให้ถูกต้อง โดยมีรายละ เอียดการใช้งานดังนี้

1. รูปแบบการป้อนคำสั่ง แต่ละคำสั่งจะประกอบด้วย 3 ตัวอักษร เสมอ และอาจตามด้วยข้อมูลอื่นๆอีก รวมความยาวทั้งหมดได้ไม่ เกิน 20 ตัวอักษร โดยที่ตัวอักษรที่สามารถใช้ป้อนข้อมูลได้ ได้แก่

แป้น ตัวอักษร A-Z, 0-9 และ SPACEBAR

แป้น BACK-SPACE ใช้สำหรับลบข้อมูลในที่พักข้อมูลชั่วคราว ถอยหลัง 1 ตัวอักษร

แป้น ESC เป็นการยก เลิกข้อมูลในที่พักข้อมูลชั่วคราวที่ได้ป้อนไปทั้งหมด

แป้น RRTURN เป็นการสั่งให้ปฏิบัติตามคำสั่งที่ได้ป้อนไว้

นอกจากนี้ ในบางคำสั่ง ยังสามารถที่จะใช้ปุ่มพิ เศษ (Hot-Key) เพื่อให้สามารถสั่งงานได้อย่างรวด เร็ว โดยที่ไม่ต้องพิมพ์คำสั่ง เองทั้งหมด

2. รายการคำสั่งสำหรับจอแสดงผล เรดาร์ แบ่งออกได้ เป็นกลุ่มๆตามลักษณะการทำงานได้ดังนี้

2.1 คำสั่งปรับการแสดงผล เป็นคำสั่งที่ใช้ในการปรับสภาพการแสดงผล ที่หน้าจอ ตามความต้องการของผู้รับการฝึก ได้แก่

2.1.1 MAP เป็นการ เลือกการแสดงผลแผนที่แบบต่างๆโดยสามารถที่จะ เลือกแสดงหรือไม่แสดงผลในแต่ละแผนที่ได้ มีรูปแบบคำสั่งดังนี้

MAP <n>

โดยที่ <n> เป็นตัว เลขหนึ่งตัว ที่ระบุแผนที่ ที่ต้องการให้แสดงหรือไม่แสดง คำสั่งนี้ เป็นคำสั่งแบบ "toggle"

ผู้ใช้จะสามารถสังเกตว่ามีแผนที่ใดแสดงอยู่บ้างได้ จาก Status Window ดังนี้

MAP 1 2 3 4 5

on on on off off

หมายความว่า แผนที่ 1,2,3 กำลังแสดงผลอยู่ และ แผนที่ 4,5 ไม่ได้แสดงผลอยู่ โดยที่ในโปรแกรมนี้ได้กำหนดให้

แผนที่ 1 = เส้นขอบเขตความรับผิดชอบในการควบคุมการบิน และเส้นแสดงความรับผิดชอบของแต่ละสนามบิน

แผนที่ 2 = แสดงจุดรายงานการบิน

แผนที่ 3 = แสดงเส้นทางการบิน

แผนที่ 4 = แสดงเส้นพิกัดทางแนวตั้ง และแนวนอน

แผนที่ 5 = แสดงวงแหวนบอกระยะทาง

2.1.2 ERS เป็นคำสั่งในการปรับมาตราส่วนในการแสดงผลบนหน้าจอ มีรูปแบบดังนี้

ERS <รัศมีที่ต้องการหน่วยเป็นไมล์ทะเล>

เช่นถ้าป้อนคำสั่ง ERS 300 เป็นการปรับมาตราส่วนรัศมีในการแสดงผลเป็น 300 ไมล์ทะเล รอบจุดศูนย์กลาง โดยที่ระยะสูงสุดที่สามารถปรับการแสดงผลได้คือ 999 ไมล์ทะเล และระยะใกล้สุดคือ 5 ไมล์ทะเล

2.1.3 OCR เป็นคำสั่งในการเปลี่ยนจุดศูนย์กลางในการแสดงผลให้ให้เป็นตำแหน่งใดก็ได้ในหน้าจอ เพื่อเป็นการเลือกแสดงผลที่จุดต่างๆได้ เช่นในแบบฝึกหัดการบินขึ้นหรือลงที่สนามบินเชียงใหม่ จะสามารถเลื่อนจุดในการแสดงผลไปที่สนามบินเชียงใหม่ได้ คำสั่งนี้มีแป้นพิเศษ คือปุ่ม F4 การใช้งานจะต้องใช้ควบคู่กับตัวชี้โดยจะต้องเลื่อนตำแหน่งของตัวชี้ให้ไปที่ตำแหน่งที่ต้องการให้เป็นจุดศูนย์กลางแล้วกดปุ่ม "F4"

2.1.4 CEN เป็นคำสั่งในการย้ายจุดศูนย์กลางการแสดงผล กลับมาที่จุดศูนย์กลางที่ได้กำหนดไว้ล่วงหน้าแล้วในโปรแกรม ซึ่งในวิทยานิพนธ์นี้ได้กำหนดไว้ที่สนามบินดอนเมือง คำสั่งนี้มีปุ่มพิเศษที่ตำแหน่ง "F5"

2.2 คำสั่งในการวัดระยะทางและมุม เป็นคำสั่งในการการวัดระยะทางและทิศทางระหว่างสองจุดใดๆบนหน้าจอ โดยที่จะต้องระบุจุดตั้งต้นและจุดสิ้นสุด ซึ่งโปรแกรมจะทำการคำนวณระยะทางระหว่างจุดทั้งสอง รวมถึงทิศทางของมุมและมุมกลับ (Reverse angle) ด้วย ซึ่งจะมีประโยชน์ในการสั่งการควบคุมทิศทางเครื่องบิน มีปุ่มพิเศษดังนี้คือ

ปุ่ม "F1" = CRO ใช้ในการกำหนดจุดเริ่มต้น

ปุ่ม "F2" = CRE ใช้ในการกำหนดจุดสิ้นสุด

ปุ่ม "F3" = CRD ใช้ในการลบเส้นวัดระยะทาง

โดยมีรูปแบบการใช้งานดังนี้

2.2.1 CRO (Cursor Origin) เป็นการระบุจุดเริ่มต้น ในการวัดระยะทาง โดยการเลื่อนตัวชี้ไปยังจุดที่ต้องการ แล้วกดปุ่ม "CRO" (F1) ซึ่งระบบจะตอบรับรู้ด้วยวงกลมสีขาวเล็กๆ (ถ้าต้องการระบุตำแหน่งใหม่ ก็สามารถทำได้ ด้วยวิธีการเดิม)

2.2.2 CRE (Cursor End) เป็นการกำหนดจุดสิ้นสุดที่ต้องการวัด โปรแกรมจะลากเส้นสีขาวระหว่างทั้งสองจุดนั้น พร้อมกับคำนวณระยะทางและมุมทั้งทิศทางตรงและทิศทางกลับ โดยจะบอกระยะทางในบรรทัดแรก และมุมพร้อมมุมกลับในบรรทัดที่สอง (โดยที่ระยะทางจะมีหน่วยเป็นไมล์ทะเล มุมตรงจะวัดจากจุดเริ่มต้นมายังจุดสิ้นสุด และมุมกลับจะจะวัดจากจุดสิ้นสุดมายังจุดเริ่มต้น) มีหน่วยเป็นองศา

2.2.3 CRD (Cursor Delete) เป็นการลบเส้นวัดระยะทาง เมื่อไม่ต้องการใช้อีกต่อไป ดังนั้นเมื่อต้องการวัดใหม่จะต้องกำหนดจุดตั้งต้นใหม่

รูปที่ 12 เส้นสำหรับวัดระยะทางและมุม

หมายเหตุ

เมื่อต้องการเปรียบเทียบจุดเริ่มต้นกับจุดปลายทางหลายๆจุด จะสามารถเปลี่ยนจุดปลายทางได้ใหม่ด้วยวิธีการเดิมโดยไม่ต้องกำหนดจุดเริ่มต้นใหม่ในทำนองเดียวกัน เมื่อต้องการวัดเปรียบเทียบระหว่างจุดเริ่มต้นหลายๆจุดก็สามารถทำได้ด้วยการย้ายจุดตั้งต้นได้ใหม่ โดยไม่ต้องกำหนดจุดปลายทางใหม่

2.3 คำสั่งเกี่ยวกับการป้อนและลบข้อมูลของเครื่องบิน เป็นคำสั่งในการป้อนและลบข้อมูลของเครื่องบิน ประกอบด้วยคำสั่งดังนี้

2.3.1 IDT เป็นคำสั่งที่กำหนดชื่อเรียกขาน (Callsign) ให้กับเครื่องบินที่ต้องการ โดยจะแสดงผลแทนรหัสประจำเที่ยวบินนั้น เพื่อต้องการให้สื่อความหมายมากขึ้นและจดจำได้ง่ายกว่า มีรูปแบบในการใช้งานดังนี้

IDT <รหัสประจำเที่ยวบิน> <ชื่อเรียกขาน>

เช่นคำสั่ง IDT A3341 THA27 เป็นการกำหนดให้เครื่องบินที่มีรหัสประจำเที่ยวบิน "A3341" มีชื่อเรียกขานว่า "THA271" เป็นต้น

หมายเหตุ

ชื่อเรียกขานนี้ ผู้ใช้จะสามารถกำหนดได้เอง โปรแกรมจะเก็บข้อมูลนี้ไว้ และแสดงผลที่หน้าต่างรอคอย เรียงตามลำดับการป้อนข้อมูล ซึ่งจะมีข้อมูลรหัสเที่ยวบินและชื่อเรียกขานพร้อมเลขลำดับที่ป้อนข้อมูล

2.3.2 DRP (Drop) เป็นคำสั่งที่ใช้ลบชื่อเรียกขาน ที่ได้ป้อนเก็บไว้ โดยการลบจะต้องระบุหมายเลขลำดับที่ต้องการลบตามที่ปรากฏในหน้าต่างรอคอย ดังนี้

DRP <ลำดับที่ปรากฏในหน้าต่างรอคอย>

เช่นคำสั่ง DRP 3 จะเป็นการลบข้อมูลชื่อเรียกขานที่ได้ป้อนไว้ ด้วยคำสั่ง IDT ในลำดับที่ 3 ในจอหน้าต่างรอคอย

หมายเหตุ

ถ้าชื่อเรียกขานที่ลบนี้ ตรงกับเครื่องบินที่ได้ทำการบินเข้ามาแล้ว โปรแกรมจะแสดงชื่อของเที่ยวบิน นั้นด้วยรหัสประจำเที่ยวบินเช่นเดิม ซึ่งเป็นข้อดีในการแยกแยะว่าเครื่องบินบินลำนี้ ยังไม่มีผู้ควบคุมจราจรทางอากาศให้ (เพราะถ้ามีผู้ควบคุมฯก็จะมีชื่อเรียกขานปรากฏ ดังนั้นการใช้คำสั่ง "DROP" จึงใช้ในความหมายการยกเลิกการควบคุมเครื่องบินลำนั้นด้วย)

2.4 คำสั่งเกี่ยวกับการแสดงผลของเครื่องบิน เป็นคำสั่งที่ปรับการแสดงผลสัญญลักษณ์ที่ใช้แทนเครื่องบินตามความต้องการของผู้รับการฝึก คำสั่งในกลุ่มนี้จะต้องนำตัวชี้ไปทับกับสัญญลักษณ์ของเครื่องบินที่ต้องการแล้วจึงป้อนคำสั่งดังนี้

2.4.1 EAA เป็นการกำหนดระดับเพดานบินที่กำหนดให้กับเครื่องบินที่กำลังทำการควบคุมอยู่ เพื่อเป็นการเตือนความจำว่าได้สั่งให้เครื่องบินทำการปรับระดับความสูงแล้ว ดังนี้

EAA <ระดับเพดานบินที่ต้องการ>

ดังนั้นเมื่อผู้ควบคุมจราจรทางอากาศสั่งให้ปรับระดับเพดานบินใหม่ เขาก็จะป้อนคำสั่ง "EAA" ทำให้ค่าความสูงนี้ไปปรากฏที่บรรทัดที่สองตามหลังจากเลขระบุเพดานบิน ดังนั้นเมื่อสังเกตจากบรรทัดนี้ก็จะทราบว่าเครื่องบินลำนี้กำลังปรับระดับไปสู่ความสูงที่ต้องการหรือไม่ เช่น

EAA 150

หมายถึง ต้องการให้เครื่องบินลำนี้ปรับระดับการบินมาอยู่ที่ระดับ 15,000 ฟิต ซึ่งโปรแกรมจะแสดงผลในส่วนข้อมูลเครื่องบินในบรรทัดที่สองดังนี้ (สมมุติว่าเดิมบินอยู่ที่ระดับ 31,000 ฟิต)

310 150

หมายความว่าได้มีการสั่งการ(ทางวิทยุ) ให้เครื่องบินลำนี้ทำการลดระดับเพดานบินจากความสูง 31,000 ฟิตลงมาเหลือ 15,000 ฟิต ซึ่งเมื่อเวลาผ่านไปสักระยะ(แล้วแต่ความสามารถในการปรับระดับการบินของเครื่องบินแต่ละชนิด) ตัวเลขนี้จะต้องเปลี่ยนเป็น

150 150

แสดงว่า เครื่องบินได้ปรับระดับการบินลงมาที่ความสูง 15,000 ฟิต เรียบร้อยแล้ว

หมายเหตุ

คำสั่งนี้เป็นการป้อนข้อมูล เพื่อเตือนความจำของผู้ควบคุมจราจรทางอากาศเท่านั้น ไม่มีผลในการบังคับหรือสั่งการให้เครื่องบินทำการบินที่ระดับความสูงที่กำหนดแต่อย่างใด เพราะเมื่อผู้ควบคุมจราจรทางอากาศ ได้สั่งการควบคุมเครื่องบินหลายๆลำอาจเกิดความสับสนว่า ได้สั่งการให้เครื่องลำไหนไปที่ระดับความสูงใดแล้ว

2.4.2 ROT เป็นคำสั่งในการหมุนทิศทางข้อมูลของเครื่องบินที่ปรากฏบนหน้าจอที่ชี้โดยเส้นชี้ เพื่อมิให้เป็นการบังการแสดงผลข้อมูลอื่นๆบนหน้าจอ โดยจะสามารถหมุนได้ 8 ทิศทางรอบตัว (ทิศทางละ 45 องศา) ดังนั้นในการใช้งานจะต้องเลื่อนตัวชี้ไปทับกับสัญญลักษณ์ของเครื่องบินที่ต้องการ และกดปุ่มพิเศษ "F6" โดยการหมุนจะหมุนตามเข็มนาฬิกาครั้งละ 45 องศา และสามารถกดซ้ำได้จนกว่าจะอยู่ตามทิศทางที่ต้องการ

โครงสร้างทางด้านฮาร์ดแวร์

จอแสดงผลข้อมูลการบินด้วยเรดาร์นี้ จะประกอบด้วยเครื่องคอมพิวเตอร์ที่มีคุณสมบัติพิเศษทางด้านกราฟิก ในลักษณะเดียวกับจอเรดาร์ที่ใช้งานจริง ซึ่งประกอบด้วยอุปกรณ์หลักที่จำเป็นในการใช้งานอยู่ 5 ส่วนคือ

1. เครื่องคอมพิวเตอร์เพื่อใช้ในการประมวลผล โดยใช้เครื่องคอมพิวเตอร์ส่วนบุคคลซึ่งมีราคาถูกแต่มีความสามารถในการประมวลผลสูง โดยมีอุปกรณ์มาตรฐานที่ติดตั้งมาพร้อมกันด้วย เช่น การ์ดแสดงผลแบบกราฟิก จอแสดงผล ฮาร์ดดิสค์ความจุสูง ช่องทางสื่อสารอนุกรม เป็นต้น นอกจากนี้ยังติดตั้งอุปกรณ์ทางด้านฮาร์ดแวร์เพื่อเพิ่มประสิทธิภาพในบางส่วนที่ต้องการความสามารถพิเศษนอกเหนือไปจากอุปกรณ์มาตรฐานที่มาพร้อมกับเครื่อง ซึ่งเป็นข้อดีของเครื่องคอมพิวเตอร์ส่วนบุคคลที่มีช่องต่อขยาย (Expansion Slot) ทำให้ขยายระบบเพื่อเพิ่มความสามารถตามความต้องการได้โดยง่าย

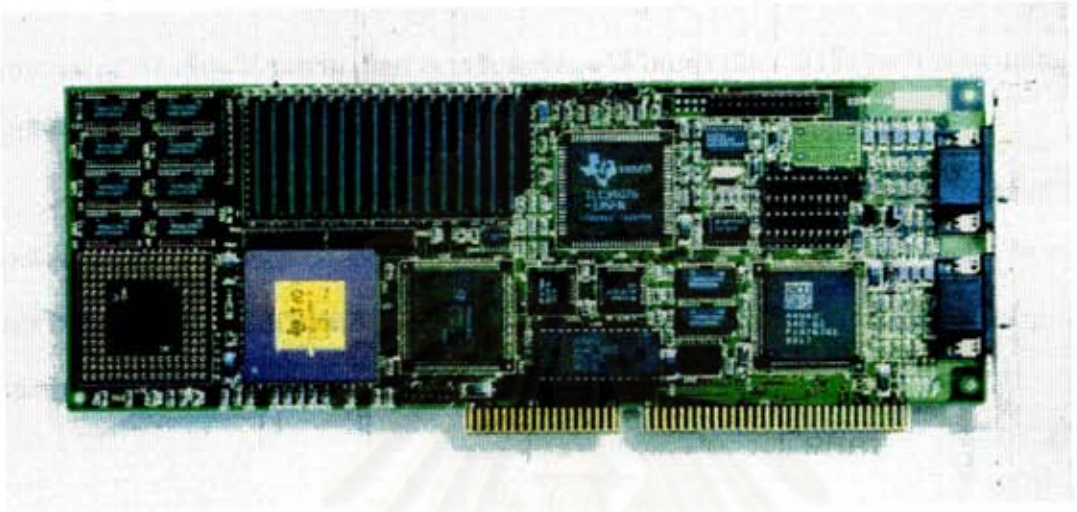
จากแนวความคิดในการจำลองการทำงานของจอ เรดาร์ โดยใช้ เครื่องคอมพิว เตอร์ส่วนบุคคลที่มีใช้กันอย่างแพร่หลายทั่วไปทดแทนการใช้ เครื่องสถานีงาน (Workstation) ขนาดใหญ่ ซึ่งถึงแม้จะมีข้อดีใน เรื่องความสามารถในการประมวลผลสูงกว่าก็ตามแต่ก็มีราคาแพงกว่า ในบทวิทยานิพนธ์นี้ จึงได้ เลือกใช้ เครื่องคอมพิว เตอร์ส่วนบุคคลที่ใช้หน่วยประมวลผลกลาง เบอร์ 486DX-33 หรือ เทียบ เท่า

2. <u>กราฟิกการ์ด</u> ใน เครื่องคอมพิว เตอร์ส่วนบุคคลทั่วไปจะติดตั้งกราฟิกการ์ดแบบ "VGA" เป็นมาตรฐานมากับ เครื่องซึ่งสามารถแสดงผลทางด้านกราฟิกได้ด้วยความละ เอียด เพียง 640x480 จุดภาพ แสดงสีได้ เพียง 16 สี ซึ่งยังมีจำนวนจุดภาพไม่ เพียงพอที่จะนำมาใช้พัฒนาระบบจำลองข้อมูลการบินด้วย เรดาร์นี้ได้ ถึงแม้จะได้มีการพัฒนา เป็น "Super VGA" ซึ่งมีความละ เอียดมากขึ้น เป็น 800x600 และ 1024x768 หรือแม้แต่ 1280x1024 จุดภาพก็ตาม แต่ก็ยังไม่มีมาตรฐานที่แน่นอน อีกทั้งคุณภาพในการแสดงผลยังไม่ดี เท่าที่ควร มีการสั่นกระพริบและสีผิด เพี้ยน ดังนั้นในบทวิทยานิพนธ์นี้ จึงได้ เลือกใช้การ์ดแสดงผลขยาย เพิ่ม เติมโดยใช้ตามมาตรฐาน "TIGA" * ซึ่งมีหน่วยประมวลผลทางด้านกราฟิกในตัว เอง ทำให้สามารถแสดงผลได้อย่างรวด เร็ว

การแสดงผลในแบบกราฟิก จำ เป็นต้องมีการคำนวณ เป็นอย่างมาก เพื่อหาตำแหน่งที่จะ เป็นจุดสว่าง มืด หรือสีต่างๆ ไม่ว่าจะ เป็น เส้นตรง เส้นโค้ง วงกลม ตัวอักษร หรือรูปภาพต่างๆ ทำให้ตัวประมวลผลกลางต้องคำนวณในการแสดงผลอย่างมากและ เหลือ เวลาในการประมวลผล

---

* Texas Instrument Graphic Architecture (TIGA) เป็นมาตรฐานทางด้านกราฟิกจากบริษัท Texas Instrument ซึ่งใช้ตัวประมวลผลช่วยทางด้านกราฟิกตระกูล TMS320xx ทำให้มีความ เร็วในการแสดงภาพกราฟิกสูงมาก โดยมีฟังก์ชั่นมาตรฐานในการ เรียกใช้งานได้โดยตรง จากคลังโปรแกรมพื้นฐานทางด้านกราฟิก เช่น การลาก เส้นตรง การวาดวงกลม การกำหนดหน้าต่างแสดงผล หรือแบบตัวอักษรต่างๆ เป็นต้น ทำให้สะดวกในการใช้งาน เป็นอย่างมาก



รูปที่ 13 กราฟิกการ์ดที่มีหน่วยประมวลผลช่วยทางด้านกราฟิก

อื่นๆน้อยลง ดังนั้น เพื่อ เป็นการ เพิ่มประสิทธิภาพในการทำงาน จึงออกแบบฮาร์ดแวร์พิ เศษให้มี ตัวประมวลผลช่วยในการทำงานทางด้านกราฟิก (Graphic Coprocessor) โดยจะรับคำสั่ง ในการแสดงภาพกราฟิกจากตัวประมวลผลหลักแล้วทำการคำนวณ เพื่อทำการแสดงผลโดยไม่ต้อง พึ่งพาการคำนวณจากตัวประมวลผลหลักอีกต่อไป ทำให้หน่วยประมวลผลหลัก มี เวลาว่างทำงาน ในส่วนอื่นๆมากขึ้น

3. จอแสดงผลความละ เอียดสูง ทำหน้าที่ในการแสดงผลบนหน้าจอโดยรับสัญญาณ มาจากกราฟิกการ์ด เพื่อนำออกแสดงผล ในบทวิทยานิพนธ์นี้ได้ใช้จอแบบราส เตอร์ ในลักษณะ เดียวกับที่ใช้ใน เครื่องรับโทรทัศน์และ เครื่องคอมพิว เตอร์ทั่วไป โดยใช้คู่กับกราฟิกการ์ดซึ่ง สามารถแสดงผลได้ที่ความละ เอียดสูงถึง 1600x1200 จุดภาพ (ได้ปรับ เพื่อใช้ความละ เอียด 1280x1024 จุดภาพ เพื่อสามารถใช้กับจอภาพโดยทั่วไปได้) จอภาพนี้จะใช้จอขนาด 17 นิ้ว ซึ่งมีพื้นที่แสดงผลมากกว่าจอขนาด 14 นิ้ว สามารถแสดงผลด้วยความละ เอียดได้อย่างชัด เจน

4. การสื่อสารแบบอนุกรม ทำหน้าที่ในการรับ-ส่งข้อมูลต่างๆที่ใช้ในการฝึก จากเครื่องคอมพิวเตอร์ของครูผู้ฝึกไปยังเครื่องคอมพิวเตอร์ที่ทำหน้าที่แสดงผลของผู้รับการฝึก โดยในบทวิทยานิพนธ์นี้ ใช้ช่องสื่อสารแบบอนุกรมที่เป็นมาตรฐานมากับเครื่องคอมพิวเตอร์ส่วนบุคคล เพราะสามารถเขียนโปรแกรมเพื่อทำการรับส่งข้อมูลได้โดยตรงมีความเร็วในการส่งผ่านข้อมูลสูงถึง 19200 บิทต่อวินาที โดยใช้ช่องสื่อสารแบบอนุกรมช่องที่หนึ่ง (COM1:)

5. อุปกรณ์ชี้ (Mouse หรือ Track-ball) สำหรับผู้รับการฝึก ใช้ในบังคับการเคลื่อนที่ของตัวชี้โดยอาศัยเมาส์ของเครื่องคอมพิวเตอร์ส่วนบุคคลมาใช้งาน เพราะมีความสะดวกและสามารถเขียนโปรแกรมเพื่อเรียกใช้งานได้โดยตรง โดยต่อเข้ากับพอร์ทแบบอนุกรมช่องที่สอง (COM2:)

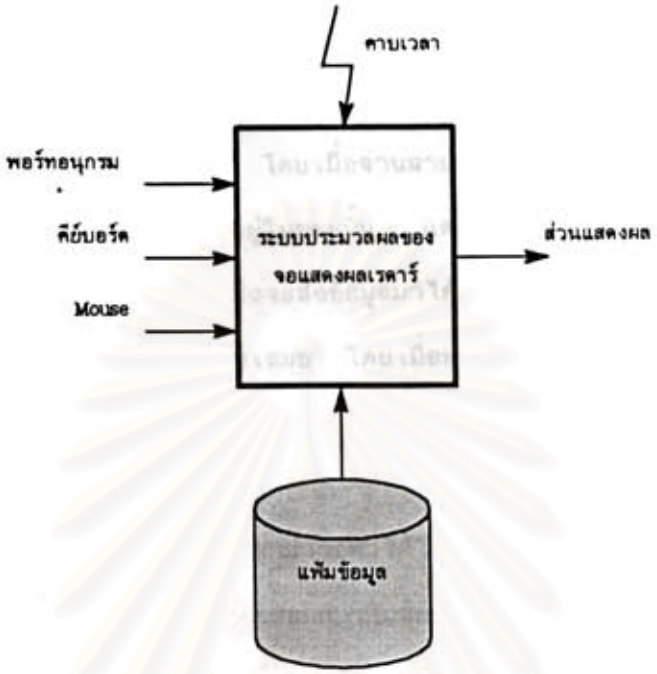
การออกแบบระบบซอฟต์แวร์

การออกแบบโครงสร้างระบบได้ทำการพัฒนาแยกเป็นส่วนต่างๆแยกกัน (Module) เพื่อที่จะพัฒนาโปรแกรมและทดสอบแยกกันได้ ทำให้โปรแกรมในแต่ละส่วนมีขนาดไม่ใหญ่จนเกินไป อีกทั้งสามารถหาข้อผิดพลาดได้ง่าย และจะนำมาประกอบรวมเข้าด้วยกันในภายหลังโดยในส่วนของจอแสดงผลเรดาร์จะพัฒนาด้วยภาษา C * เนื่องจากมีประสิทธิภาพสูงและหาเครื่องมือในการพัฒนาระบบได้โดยง่าย

---

* ในวิทยานิพนธ์นี้ใช้ C Compiler ของบริษัทไมโครซอฟต์ เวอร์ชั่น 6.0 ที่ใช้บนบนระบบปฏิบัติการ เอ็ม เอสดอสของบริษัทไมโครซอฟต์ โดยใช้ร่วมกับชุดพัฒนาทางด้านกราฟิกของบริษัท เท็กซัสอินสตรูเมนท์ในรูปของคลังโปรแกรมทางด้านกราฟิก ซึ่งจะมีฟังก์ชั่นทางด้านกราฟิกให้เรียกใช้งานได้

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 14 โปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์

โครงสร้างโปรแกรมในส่วนการแสดงผลข้อมูลการบินด้วย เรดาร์

โปรแกรมจะทำงานในลักษณะประมวลผลทันที (Real Time Processing) นั่นคือ จะวิ่งวนรอบต่อ เนื่อง เพื่อรอรับข้อมูลและนำมาประมวลผลตามชนิดของข้อมูลนั้นๆ โดยที่จะต้อง สามารถตอบสนองการรับข้อมูล เข้ามาได้อย่างทันท่วงที โดยไม่ เกิดการสูญหายของข้อมูลใดๆ นอกจากนี้ในระหว่างที่กำลังรอรับข้อมูลชนิดหนึ่ง แต่ข้อมูลนั้นยังไม่สมบูรณ์หรือ เสร็จสิ้นครบถ้วน ระบบก็จะต้องไม่หยุดชะงักการทำงานอื่นๆ เพื่อรอข้อมูลนั้นอย่าง เดียว จะต้องทำงานอื่นๆไปด้วย ดังนั้นการ เขียนโปรแกรมในลักษณะนี้ จึงต้องออกแบบระบบให้สามารถตรวจสอบอุปกรณ์ เหล่านี้ อยู่ตลอด เวลา เมื่อมีข้อมูล เข้ามาก็จะนำข้อมูล เหล่านั้นไปประมวลผลตามชนิดของข้อมูลนั้นๆ โดย มีอุปกรณ์รับข้อมูลดังนี้

1. ช่องสื่อสารแบบอนุกรม ทำหน้าที่ในการรับข้อมูลจากพอร์ทอนุกรม ที่ถูกส่งมาจากโปรแกรมจำลองข้อมูลการบิน(นักบินสมมุติ) ซึ่งประกอบด้วยข้อมูลสองส่วนคือ นาฬิกาของระบบ และข้อมูลของเครื่องบิน โดยข้อมูลทั้งสองนี้จะถูกส่งมาทุกๆคาบเวลา (ซึ่งเปรียบได้กับรอบการหมุนของสายอากาศของเรดาร์) โดยเมื่อจานสายอากาศเรดาร์หมุนไปครบหนึ่งรอบจะได้ข้อมูลของเครื่องบินที่กำลังทำการบินอยู่ในขณะนั้น แต่เนื่องจากในส่วนของการรับข้อมูลจะไม่สามารถทราบกำหนดเวลาที่ทางด้านส่งจะส่งข้อมูลมาได้แน่นอน ดังนั้นโปรแกรมในส่วนของการรับข้อมูลจะต้องพร้อมในการรับข้อมูลเสมอ โดยเมื่อทางด้านส่งส่งข้อมูลมา ทางด้านรับจะต้องรับข้อมูลได้ทันทีโดยไม่ผิดพลาดหรือขาดหายไป ดังนั้นโปรแกรมในส่วนการรับข้อมูลจะมีความสำคัญมาก

เพื่อให้การออกแบบโปรแกรมสามารถทำได้โดยง่าย จึงได้พัฒนาโปรแกรมย่อยแบบฝังตัวในหน่วยความจำ เพื่อทำหน้าที่ตรวจสอบการรับข้อมูลจากพอร์ทแบบอนุกรมช่องที่หนึ่ง แล้วจัดเก็บข้อมูลลงในที่พักข้อมูลชั่วคราว และรอให้โปรแกรมหลักมาอ่านข้อมูลต่อไป

2. คาบเวลา เป็นข้อมูลที่ทำการกำหนดคาบเวลา ในการดำเนินการภายในของโปรแกรมที่ต้องทำงานตามกำหนดคาบเวลา ซึ่งประกอบด้วย

2.1 คาบเวลาการกระพริบ เป็นการกำหนดอัตราการกระพริบของ ตัวชี้ เนื่องจากฮาร์ดแวร์ไม่สนับสนุนการกระพริบในโหมดกราฟิก ดังนั้นจึงต้องเขียนโปรแกรมเพื่อทำการกระพริบเอง ด้วยการแสดงผลและลบออกทุกๆคาบเวลาการกระพริบทำให้มองดูเสมือนเกิดการกระพริบ แต่ด้วยข้อจำกัดทางด้านฮาร์ดแวร์นี้จึงกำหนดให้อัตราการกระพริบทุกๆ 1 1 วินาที ดังนั้นทุกๆวินาทีโปรแกรมจะต้องทำการแสดงและลบการแสดงผลตัวชี้ด้วย

2.2 คาบเวลาการวาดภาพหน้าจอ เพื่อให้การแสดงผลที่หน้าจอเป็นไปโดยนุ่มนวล ไม่เกิดการกระพริบ โดยต้องทำการปรับการแสดงผลเรียงตามตำแหน่งของเครื่องบินบนหน้าจอไล่เรียงกันไป เพราะถ้าทำการแสดงผลที่หน้าจอพร้อมกันทั้งหมดทำให้โปรแกรมต้องคำนวณเป็นอย่างมากทำให้ภาพทั้งหมดเกิดการกระพริบ เพื่อหลีกเลี่ยงปัญหานี้จึงแบ่งการแสดงผลเครื่องบินออกเป็นกลุ่มย่อย แล้วทยอยนำออกแสดงผลตามลำดับการรับข้อมูลจากช่องสื่อสารอนุกรม ทำให้ภาพการเคลื่อนที่ของเครื่องบินบนจอภาพนุ่มนวลสมจริง

2.3 คาบเวลาการแสดงผลนาฬิกา โดยที่การแสดงผลนาฬิกาของระบบ ให้สอดคล้องกับนาฬิกาที่ปรากฏในจอของผู้ควบคุมการฝึก จึงจะรับข้อมูลนาฬิกานี้มาจากผู้ควบคุมการฝึก โดยจะทำการปรับการแสดงผลทุกๆนาที

หมายเหตุ

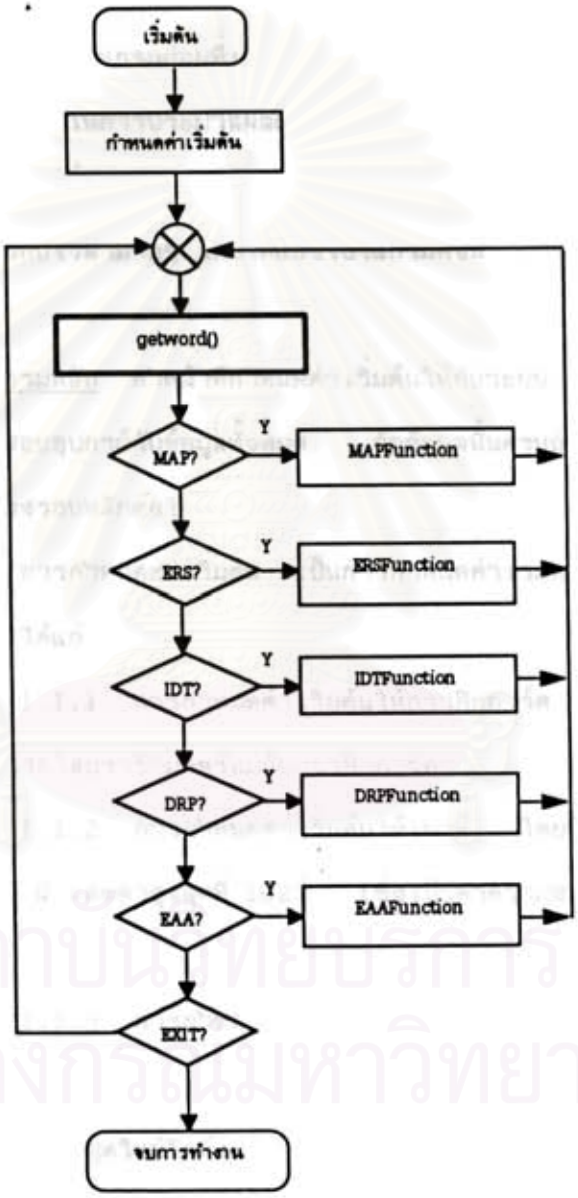
โปรแกรมในกลุ่มนี้ ถึงแม้จะมีส่วนขึ้นกับเวลา แต่ไม่ถึงกับเป็นจุดวิกฤติ เนื่องจากถึงแม้โปรแกรมจะวนมาดำเนินการประมวลผลไม่ทัน ก็จะมีผลเพียงแค่ การแสดงผลบนหน้าจอช้าลงเล็กน้อยเท่านั้น ไม่ทำให้โปรแกรมทำงานผิดพลาดแต่อย่างใด

3. แป้นพิมพ์ เป็นส่วนที่รับคำสั่งต่างๆ มาจากผู้ใช้(ผู้รับการฝึก) โดยผ่านทางจอหน้าต่างรับคำสั่งจากผู้ใช้โดยจะมีตัวชี้ตำแหน่ง เพื่อแสดงตำแหน่งการป้อนข้อมูล แต่ในระหว่างที่กำลังรอรับการกดปุ่มจากผู้ใช้ระบบยังต้องสามารถทำงานอื่นได้อยู่ ดังนั้นจึงจำเป็นต้องมีที่พักข้อมูลชั่วคราว เพื่อพักข้อมูลที่ผู้ใช้ได้ป้อนไว้แต่ยังไม่เสร็จสิ้น เก็บไว้ก่อน ดังนั้นในระหว่างที่รอการป้อนข้อมูลจากผู้ใช้งานโดย เก็บสถานะการป้อนข้อมูลในปัจจุบันไว้ก่อน เมื่อมีการกดปุ่มป้อนข้อมูลใหม่ก็จะนำสถานะ เดิมมาตรวจสอบและจัดเก็บข้อมูลลงในที่พักข้อมูลตามลำดับ จนกว่าจะมีการสั่งให้นำไปปฏิบัติงานด้วยการกดแป้นรับข้อมูล เข้า หรือการแก้ไขการป้อนข้อมูล

นอกจากนี้โปรแกรมยังจะต้องตรวจสอบการกดปุ่มพิเศษ (ฟังก์ชั่นคีย์ F1-F10) โดยโปรแกรมจะปฏิบัติงานทันทีตามหน้าที่การทำงานที่ได้กำหนดไว้ได้ทันที

4. เมาส์ เป็นส่วนการรับข้อมูลจากผู้ใช้ เพื่อบังคับการเคลื่อนที่ของตัวชี้ในการวัดระยะทางบนหน้าจอแสดงผลด้วยเมาส์ โดยมีโปรแกรมย่อยทำหน้าที่ตรวจสอบการเคลื่อนที่หรือการกดปุ่มเมาส์ เพื่อรอการเรียกใช้จากโปรแกรมหลักต่อไป

5. แฟ้มข้อมูลต่างๆ ใช้สำหรับเก็บข้อมูลที่จำเป็นต่างๆโดยมีข้อดีที่สามารถเปลี่ยนแปลง แก้ไขข้อมูลได้ง่ายโดยไม่ต้องแก้ไขตัวโปรแกรม เช่น แฟ้มข้อมูลต่างๆ ซึ่งจัดเก็บอยู่ในฮาร์ดดิสก์ โดยจะอ่านข้อมูลมาเก็บในหน่วยความจำ เพื่อให้สามารถเข้าถึงข้อมูลได้อย่างรวดเร็วขึ้น



รูปที่ 15 ผังการทำงานของโปรแกรมแสดงผลข้อมูลการบินด้วย เรดาร์

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ผังการทำงานของโปรแกรม

โครงสร้างของโปรแกรมจะแบ่งงานออกเป็นส่วนเล็กๆ เพื่อที่จะทำการประมวลผลให้เสร็จสิ้นในเวลาอันรวดเร็วโดยแบ่งตามประเภทของชนิดของข้อมูลนั้นๆ เมื่อมีข้อมูลชนิดนั้นเข้ามา ก็จะทำการเรียกโปรแกรมย่อยที่เหมาะสมมาประมวลผล แล้วกลับไปวนรอรับข้อมูลในรอบต่อไป โดยที่ระยะเวลาในการประมวลผลแต่ละรอบจะต้องสั้นที่สุด เพื่อที่ได้จะไม่ไปกระทบกระเทือนการทำงานในส่วนอื่นๆ นอกจากนี้ยังได้มีการออกแบบโปรแกรมย่อย เพื่อจัดการในส่วนการทำงานที่ต้องขึ้นกับเวลามากๆ โดยได้แบ่งโปรแกรมดังนี้

1. <u>โปรแกรมหลัก</u> ทำหน้าที่กำหนดค่าเริ่มต้นให้กับระบบ จากนั้นจะเข้าสู่วงรอบหลักเพื่อทำหน้าที่ตรวจสอบอุปกรณ์รับข้อมูลทั้งหมด ถ้าข้อมูลนั้นครบถ้วนก็จะนำไปประมวลผลตามข้อมูลนั้นแล้วกลับสู่วงรอบหลักต่อไป จนกว่าจะออกจากโปรแกรม ประกอบด้วย

1.1 การกำหนดค่าเริ่มต้น เป็นการกำหนดค่าเริ่มต้นให้กับตัวแปรต่างๆ ในตอนเริ่มต้นการทำงาน ได้แก่

1.1.1 การกำหนดค่าเริ่มต้นให้กราฟิกการ์ด เป็นฟังก์ชั่นสำเร็จรูปซึ่งสามารถเรียกใช้ได้จากไลบรารีที่มาพร้อมกับกราฟิกการ์ด

1.1.2 การกำหนดค่าเริ่มต้นให้เมาส์ โดยให้มีค่าเริ่มต้นที่ตำแหน่งกลางหน้าจอมีค่าต่ำสุดที่ 0 และค่าสูงสุดที่ 1023 (ซึ่งเป็นค่าความละเอียดสูงสุดของหน้าจอทางด้านแกน Y)

1.1.3 การกำหนดค่าเริ่มต้นให้กับพอร์ทอนุกรม ได้แก่การกำหนดค่าความเร็วในการรับ-ส่งและรูปแบบในการส่งข้อมูล นอกจากนี้ยังลบข้อมูลที่อยู่ในที่พักข้อมูลชั่วคราว เพื่อเตรียมรับข้อมูลชุดใหม่อีกด้วย

1.1.4 การกำหนดค่าตัวแปรเริ่มต้นอื่นๆ ได้แก่

- รัศมีการแสดงผลที่หน้าจอในตอนเริ่มต้นการทำงานให้มีรัศมีเท่ากับ 300 ไมล์ทะเล

- กำหนดตำแหน่งศูนย์กลางการแสดงผลที่หน้าจอ โดยกำหนดให้จุดศูนย์กลางอยู่ที่สนามบินดอนเมือง

- การป้อนค่าตั้งต้นให้กับนาฬิกาของระบบด้วยค่าว่าง เพื่อรอรับข้อมูลนาฬิกาจากโปรแกรมนักบินจำลอง

- ลบข้อมูลต่างๆในตาราง เก็บข้อมูลการบิน เพื่อเริ่มต้นการรับข้อมูลใหม่

- กำหนดค่าตัวบ่งชี้ในการแสดงผล ให้แสดงแผนที่ 1,2 และ 3 โดยอัตโนมัติ เมื่อเริ่มต้นการทำงาน

1.2 <u>การอ่านแฟ้มข้อมูล</u> เป็นการอ่านข้อมูลจากแฟ้มข้อมูลต่างๆ เพื่อจัดเก็บในฐานข้อมูลของระบบ เพื่อใช้งานต่อไป

1.3 <u>วงรอบหลัก</u> ทำหน้าที่วนรอบเพื่อรอคำสั่งจากผู้ใช้จากแป้นพิมพ์ โดยจะมีโปรแกรมย่อยทำหน้าที่วิเคราะห์คำสั่งและตรวจสอบไวยากรณ์ข้อมูลที่ป้อนเข้ามา ซึ่งถ้าข้อมูลทุกอย่างถูกต้อง ก็จะเรียกโปรแกรมย่อยตามฟังก์ชั่นการทำงานนั้นๆมาทำงานและแสดงข้อความว่าปฏิบัติคำสั่งจบสิ้นแล้ว แต่ถ้าคำสั่งนั้นไม่สามารถปฏิบัติงานได้ไม่ว่าจะด้วยผิดหลักไวยากรณ์หรือป้อนข้อมูลไม่ถูกต้อง ก็จะแสดงข้อความแจ้งสาเหตุของความผิดพลาด หลังจากนั้นก็จะกลับไปยังวงรอบหลักเพื่ออ่านคำสั่งต่อไป

ในวงรอบนี้จะเรียกฟังก์ชั่นที่สำคัญตัวหนึ่งคือ <u>getword()</u> ซึ่งทำหน้าที่รอรับคำสั่งจากผู้ใช้งานผ่านทางแป้นพิมพ์ เก็บลงในที่พักข้อมูลชั่วคราวจนกว่าจะกดแป้นรับข้อมูลเข้า ก่อนจะกระโดดไปทำงานตามคำสั่งนั้นๆ แต่เนื่องจากผู้ใช้ไม่ได้ป้อนข้อมูลอยู่ตลอดเวลาดังนั้นในเวลาส่วนที่รอการกดคีย์จากผู้ใช้จะต้องไปทำงานอื่นๆด้วย จากข้อจำกัดนี้ฟังก์ชั่น getword() จึงไม่สามารถเรียกใช้ฟังก์ชั่นในการอ่านคีย์บอร์ดของไลบรารีภาษา C โดยตรงได้ จึงพัฒนาฟังก์ชั่นในส่วนรับข้อมูลจากแป้นพิมพ์ใหม่เพื่อใช้งานแทนฟังก์ชั่นเดิม โดยมีรายละเอียดดังนี้

1.3.1 <u>การอ่านค่าจากคีย์บอร์ด</u> จะเรียกใช้ฟังก์ชั่น wait_key() แทนฟังก์ชั่นการอ่านค่าคีย์บอร์ดที่มีในภาษาซีเดิม ซึ่งฟังก์ชั่นนี้จะทำการตรวจสอบการกดคีย์ก่อน ถ้าไม่มีการกดคีย์ก็จะข้ามไปทำงานอื่นๆก่อน และวนกลับมาตรวจสอบการกดคีย์ใหม่ในรอบต่อไป ดังนี้

```
wait_key()
{
    loop forever
        if( key_pressed )
            if( special_key )
                case function   : do_function
                case key_cursor : do_cursor
            else
                return character_from_key_board
            endif
        else
            if( mouse_move) read_mouse_movement
            if( serial_input) read_serial_data
            if( one_second_passed )
                flash_cursor
                re-draw track from top to bottom
            endif
        endif
    loop again
}
```

ฟังก์ชั่นนี้จะวนรอบเพื่อทำการตรวจสอบการกดคีย์ ถ้ามีการกดปุ่มก็จะตรวจสอบว่าเป็นปุ่มพิเศษหรือไม่ ถ้าใช่ก็จะไปทำงานตามหน้าที่การทำงานที่ได้กำหนดไว้ ถ้าไม่ใช่แสดงว่าเป็นตัวอักษรปกติก็จะส่งค่านี้กลับไปยังโปรแกรมที่เรียกมา แต่ถ้าไม่มีการกดปุ่มใดๆ เลยก็จะกลับไปทำงานของระบบอื่นๆ ได้แก่

- ตรวจสอบการเลื่อนเมาส์ ซึ่งถ้ามีการเลื่อนก็จะนำค่าระยะของการเลื่อนไปรวมเพื่อคำนวณหาตำแหน่งปัจจุบัน และแสดงผลตัวชี้ที่ตำแหน่งใหม่

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

- ตรวจสอบการรับข้อมูลจากพอร์ทอนุกรม ซึ่งถ้ามีข้อมูลเข้ามาก็จะอ่านข้อมูลนี้จากที่พักข้อมูลชั่วคราว เพื่อรวบรวมข้อมูลจนกว่าจะครบทั้งชุดจึงนำข้อมูลนี้ไปทำการปรับปรุงตารางเก็บข้อมูลการบิน ก่อนที่จะนำไปแสดงผลต่อไปในภายหลัง

- ตรวจสอบคาบเวลา เมื่อครบ 1 วินาทีก็จะทำการแสดงผลนาฬิกาที่ได้รับมาจากนักบินจำลอง อีกทั้งทำการแสดงผลเครื่องบินบนหน้าจอ เพื่อให้การแสดงผลบนหน้าจอเป็นไปด้วยความนุ่มนวล โดยจะค่อยๆแสดงผลเรียงลำดับไล่กันไปจนแสดงผลครบทุกตัวภายในรอบเรดาร์สวีพ นอกจากนี้ยังจะแสดงผลตัวชี้ในมีลักษณะกระพริบด้วย

2. <u>โปรแกรมย่อยทำหน้าที่แสดงผลหน้าจอ</u> โดยที่หน้าจอแสดงผลได้แบ่งการแสดงผลออกเป็นหลายๆส่วนทำหน้าที่แสดงผลในแต่ละหน้าต่างแยกจากกัน ซึ่งมีข้อดีที่จะเลือกแสดงผลในส่วนที่ต้องการได้ โดยไม่จำเป็นต้องปรับการแสดงผลทั้งหมด(ทั้งๆที่ต้องการปรับการแสดงผลเพียงแค่ส่วนเดียว) ทั้งนี้โดยอาศัยการออกแบบที่ได้เก็บข้อมูลทุกอย่างไว้ในตารางหรือตัวแปรไว้หมด ดังนั้นการปรับการแสดงผลจึงสามารถทำได้ตลอดเวลา โดยสามารถตรวจสอบได้จากค่าสถานะที่ได้เก็บไว้นั่นเอง การแสดงผลจะแบ่งออกเป็น

2.1 การแสดงผลชื่อโปรแกรม เป็นหน้าจอเล็กๆแสดงชื่อโปรแกรมและวันที่ที่พัฒนาโปรแกรม ซึ่งหน้าต่างนี้จะเป็นข้อมูลที่ไม่มีการเปลี่ยนแปลงจากการทำงานเลยทำให้สามารถสั่งแสดงผลได้โดยตรง

2.2 การแสดงผลสถานะของระบบ เป็นหน้าจอที่แสดงสถานะของระบบ ซึ่งจะต้องปรับการแสดงผลทุกครั้งที่มีการเปลี่ยนแปลงค่าตัวแปรระบบ เช่น การปรับระยะของรัศมีการแสดงผล เป็นต้น

2.3 การแสดงผลนาฬิกาของระบบ เป็นหน้าจอที่แสดงเวลาของระบบซึ่งจะมีการปรับค่าเวลาทุกเรดาร์สวีพ เพื่อให้ข้อมูลนาฬิกาตรงกับทางด้านนักบินจำลองเสมอ

2.4 การแสดงผลการป้อนข้อมูลการบินล่วงหน้า เป็นหน้าจอแสดงตารางการป้อนชื่อเที่ยวบินที่กำลังทำการควบคุมอยู่จากคำสั่ง"IDT" ซึ่งเป็นคำสั่งในการป้อนข้อมูลเข้าและคำสั่ง "DRP" ซึ่งเป็นคำสั่งลบข้อมูลทิ้ง ดังนั้นทุกๆครั้งที่มีการเรียกใช้สองคำสั่งนี้จะต้องปรับการแสดงผลที่หน้าจอนี้ด้วย

2.5 การแสดงผลการรับคำสั่งจากผู้ใช้งาน เป็นหน้าจอที่แสดงคำสั่งที่กำลังป้อนโดยผู้ใช้งานผ่านทางแป้นพิมพ์ ซึ่งจะป้อนเข้ามาทีละหนึ่งตัวอักษรโดยจะแสดงให้เห็นผ่านทางหน้าจอนี้ ดังนั้นเมื่อมีการป้อนตัวอักษรจะต้องปรับการแสดงผลที่หน้าจอนี้ด้วย

2.6 การแสดงผลตำแหน่งพิกัดตัวชี้ เป็นหน้าจอที่แสดงตำแหน่งตัวชี้จากการเลื่อนเมาส์ ซึ่งจะต้องปรับการแสดงผลที่หน้าจอนี้ด้วย โดยจะทำการแปลงการเคลื่อนที่ออกมาเป็นตำแหน่งพิกัดของหน้าจอ เพื่อแสดงผลต่อไป

2.7 การแสดงผลข้อมูลการบิน เป็นส่วนที่สำคัญในการแสดงผลข้อมูลการบิน โดยจะแสดงผลในลักษณะ เช่น เดียวกับจอ เรดาร์จริงประกอบด้วย

2.7.1 การแสดงผลแผนที่ต่างๆ จากแฟ้มข้อมูลการบินต่างๆดังนี้

2.7.1.1 แผนที่ เส้นขอบ เขตการควบคุมการบิน ซึ่งจัด เก็บข้อมูลจุดที่อยู่ติดกัน เรียงติดกัน ดังนั้นโปรแกรมจะอ่านข้อมูลขึ้นมาสองระ เบียนแรกแล้วลาก เส้นเชื่อมระหว่างสองจุดนั้น จากนั้นจะวนรอบอ่านข้อมูลจุดถัดไปขึ้นมาแล้วลาก เส้นต่อกับ เส้นแรก เช่นนี้ไป เรื่อยจนกว่าจะจบฐานข้อมูล แต่มีการพิ เศษโดย เมื่อพบระ เบียนที่มีอักษรแรก เป็น ';' จะถือว่าจบชุด เส้นขอบ เขตแรก และไม่นำระ เบียนนี้ไปประมวลผล แต่อ่านระ เบียนถัดไปแทน โดยถือว่า เป็นจุด เริ่มต้นของ เส้นใหม่ ดังนั้นระ เบียนพิ เศษนี้จึงสามารถใช้ เป็นตัวคั่น หรือใช้อธิบายแฟ้มข้อมูลได้

2.7.1.2 แผนที่แสดงจุดรายงานการบิน ซึ่งจัด เก็บข้อมูลจุดรายงานการบิน เป็นอิสระไม่ เกี่ยวข้องซึ่งกันและกัน การแสดงผลจึงทำได้โดยการอ่านข้อมูลจากฐานข้อมูลพร้อมกับแสดงผลที่ตำแหน่งพิกัดตามที่ได้ระบุไว้

2.7.1.3 แผนที่แสดง เส้นทางการบิน เป็น เส้นที่ เชื่อมโยงระหว่างจุดรายงานการบิน ประกอบกัน เป็น เส้นทางการบิน แต่ เนื่องจากแต่ละ เส้นทางมีความซับซ้อนไม่ เท่ากันมีทั้งสั้นและยาว เพื่อให้การจัด เก็บมีรูปแบบ เดียวกันจึงได้จัด เก็บแยก เป็นช่วงๆ โดยที่แต่ละระ เบียนหมายถึง เส้นทางการบินในหนึ่งช่วงย่อย ดังนั้นในการแสดงผลจึงแสดงทีละช่วงย่อยได้ จนหมดข้อมูลในตาราง โดยที่จะต้องนำจุดรายงานการบินต้นทางและจุดปลายทางไปทำการค้นหาตำแหน่งพิกัดทางแนวตั้งและแนวนอน ก่อนที่จะนำไปแสดงผลต่อไป

2.7.1.4 แผนที่แสดง เส้นขอบ เขตสนามบิน เพื่อแสดงผล เส้นขอบ เขตความรับผิดชอบของแต่ละสนามบิน โดยประกอบด้วย เส้นตรงและ เส้นโค้งต่างๆ ซึ่ง

ได้ระบุไว้ในฐานข้อมูลทั้งหมดแล้ว โปรแกรมจะอ่านข้อมูลทีละหนึ่งระเบียนและทำการแสดงผลตามข้อมูลในระเบียนนั้นๆ

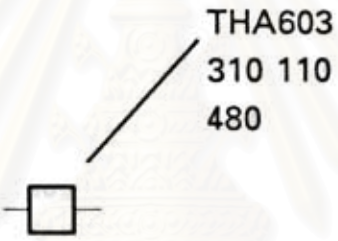
2.7.2 การแสดงผลข้อมูลสัญญลักษณ์แทนตำแหน่งเครื่องบินซึ่งเป็นข้อมูลที่มีการเปลี่ยนแปลงเกือบตลอดเวลา เพื่อให้การแสดงผลทำได้อย่างรวดเร็วจึงใช้วิธีการ "Exclusive OR" ไปบนภาพเดิมและเมื่อต้องการลบภาพก็จะใช้การ "Exclusive OR" ลงไปที่ภาพเดิม ซึ่งจะทำให้ได้ภาพเดิมกลับคืนมา

การใช้วิธีการ "XOR" จะทำให้ภาพทั้งหมดไม่ถูกกระทบกระเทือนซึ่งทำให้การแสดงผลนุ่มนวลกว่า โดยไม่เกิดการกระพริบ

2.7.2.1 การจัดเก็บข้อมูลเครื่องบิน จะเก็บลงในตารางข้อมูลการบิน "tracklist" ซึ่งประกอบด้วยข้อมูลต่างๆของเครื่องบินทั้งหมด ดังนี้

1. squawk เก็บรหัสเครื่องบิน
2. speed เก็บความเร็วเครื่องบิน
3. altitude เก็บระดับเพดานบิน
4. latitude เก็บตำแหน่งทางด้านแนวตั้ง
5. longitude เก็บตำแหน่งทางด้านแนวนอน
6. callsign เป็นข้อมูลที่ผู้ใช้ป้อนเพื่อใช้เป็นชื่อเรียกขานแทนรหัสเครื่องบิน
7. eaa เป็นข้อมูลที่ผู้ใช้ป้อนเพื่อใช้กำหนดเพดานบิน
8. leadangle เป็นข้อมูลที่ผู้ใช้ป้อนเพื่อใช้แสดงทิศทางการแสดงข้อมูล
9. miss เพื่อใช้เก็บค่าการไม่ได้รับการ update
10. lat1 เพื่อใช้เก็บค่าตำแหน่งเดิมครั้งที่ 1
11. lng1
12. lat2 เพื่อใช้เก็บค่าตำแหน่งเดิมครั้งที่ 2
13. lng2
14. lat3 เพื่อใช้เก็บค่าตำแหน่งเดิมครั้งที่ 3
15. lng3
16. lat4 เพื่อใช้เก็บค่าตำแหน่งเดิมครั้งที่ 4
17. lng4

ข้อมูล เหล่านี้ จะใช้ในการแสดงผลบนหน้าจอตามรูปแบบ ที่ใช้ในจอแสดงผลที่ใช้งานจริง รวมถึงข้อมูลที่ เป็นการป้อนจากผู้ใช้ด้วย ซึ่งข้อมูล เหล่านี้จะถูกปรับค่าทุกครั้งที่ได้รับข้อมูลชุดใหม่ โดยมีข้อมูลพิ เศษสำหรับ เก็บค่าการปรับปรุงข้อมูล เรียกว่า "miss" ซึ่งค่านี้จะถูกปรับให้ เป็นศูนย์ทุกครั้งที่ได้รับข้อมูลชุดใหม่และ เพิ่มค่าอีกหนึ่งถ้าไม่ได้รับการปรับข้อมูลในรอบ เรดาร์สวีพนั้น ซึ่งหมายความว่าในรอบที่ผ่านมา เรดาร์ไม่สามารถจับภาพได้และแสดงผลบนหน้าจอ เพิ่มด้วยขีด เส้นกลางหนึ่ง เส้น ถ้าในรอบต่อไปยังไม่ได้รับอีกก็จะมีขีดอีกหนึ่งรวม เป็นสอง เส้น และถ้าในรอบต่อไปไม่ได้รับอีก เป็นครั้งที่สาม จะถือว่า เครื่องบินลำนี้ได้หายไปจากจอ เรดาร์แล้ว เรียกว่า "missed track"



รูปที่ 16 แสดงสัญญลักษณ์ เครื่องบินที่ไม่ได้รับข้อมูลใหม่

การแสดงผลจะอาศัยข้อมูลที่ เก็บไว้ในตาราง เหล่านี้ในการแสดงผล แต่ เนื่องจากในระหว่างที่กำลังแสดงผลอยู่จะต้อง เตรียมที่พักข้อมูลชั่วคราวไว้ เผื่อข้อมูลใหม่ที่อาจจะถูกส่ง เข้ามาได้อีก โดยที่ข้อมูลชุด เดิมก็ยังมีความจำ เป็นที่จะต้อง เก็บไว้ เพื่อสำหรับลบภาพออกจากจอด้วย ดังนั้นจึงต้องมีอีกตารางข้อมูลหนึ่ง ซึ่งมีโครงสร้าง เช่น เดียวกัน เรียกว่า "oldtracklist" เพื่อใช้ เก็บข้อมูลที่ได้แสดงผลไปแล้ว

ทั้งสองตารางนี้ มีความสำคัญต่อระบบ เป็นอย่างมากและจะต้องมีข้อมูลที่สอดคล้องกันเสมอ โดยตารางตัวแรกจะรับข้อมูลจากที่พักข้อมูลชั่วคราว ซึ่งได้รับมาจากพอร์ทอนุกรมมา เก็บไว้ เพื่อรอการแสดงผล และ เมื่อแสดงผล เรียบร้อยแล้วก็จะสำ เนาข้อมูลนี้ไป เก็บไว้ยังตารางที่สอง เพื่อไว้ใช้ในการลบภาพที่ เพิ่งแสดงออกไป เมื่อต้องการปรับการแสดงผล โปรแกรมจะวนรอบในลักษณะนี้ เสมอทุกๆครั้งที่มีข้อมูล เข้ามา

2.7.2.2 การแสดงผล เครื่องบิน จะแสดง เป็น สัญญลักษณ์ตามที่ปรากฏในจอแสดงผล เรดาร์ที่ใช้งานจริง ซึ่งจะประกอบด้วย

- สัญญลักษณ์แสดงตำแหน่ง เครื่องบิน จะแสดงด้วยสี่ เหลี่ยม เล็กสีขาวตามตำแหน่งพิกัดทางด้านแนวตั้ง และแนวนอน

- เส้นโยงบอกรายละ เอียด เป็น เส้นชี้จากสัญญลักษณ์แสดงตำแหน่ง เพื่อแสดงข้อมูลรายละ เอียด ซึ่ง เส้นนี้จะ เลือกแสดงได้ 8 ทิศทางรอบตัว

- ชื่อรหัส เครื่องบินหรือชื่อ เรียกขาน เที่ยวบิน

- ระดับ เพดานบิน และระดับ เพดานบินที่กำหนด

- ความ เร็ว เครื่องบิน

- ตำแหน่งของ เครื่องบินในครั้งที่ผ่านมา เป็นจุดแสดงตำแหน่งของ เครื่องบินใน 4 ตำแหน่งที่ผ่านมา เพื่อ เป็นการบอกทิศทางการ เคลื่อนที่ของ เครื่องบิน

- เส้นแสดงการไม่ได้รับข้อมูลโดยข้อมูลจาก "miss" เพื่อ เตือนว่าข้อมูลนี้ไม่ได้รับมาแล้วกี่ครั้ง โดยจะแสดงทีละ เส้น, สอง เส้น และลบข้อมูลนี้ออกจากตารางข้อมูลการบิน ถ้าไม่ได้รับข้อมูล เลยภายในสามคาบ เวลาของ เรดาร์สวีพ

2.8 การแสดงผลหน้าจอทั้งหมด เป็นการลบภาพหน้าจอ เพื่อทำการแสดงผลใหม่ทั้งหมด อัน เนื่องมาจากมีการปรับค่าพารามิ เตอร์ที่มีผลกระทบในการแสดงผลทั้งหมด จึงต้องมีการแสดงผลใหม่ทั้งหมด (ด้วยค่าพารามิ เตอร์ใหม่) เช่น การปรับระยะการแสดงผลทำให้ต้องคำนวณระยะการแสดงผลบนหน้าจอใหม่ทั้งหมด ซึ่งจะใช้ เวลาในการคำนวณค่อนข้างมาก

3. <u>โปรแกรมย่อยทำหน้าที่จัดการ เมาส์</u> โดย เรียกใช้ฟังก์ชั่นของ เมาส์ไดร์ เวอร์โดยผ่านการขัดจังหวะทางด้านซอฟต์แวร์ เบอร์ 33H ประกอบไปด้วย

3.1 การกำหนดค่า เริ่มต้นให้ เมาส์ เพื่อตรวจสอบการติดตั้ง เมาส์โดย เรียกใช้ฟังก์ชั่น 0 ซึ่งจะส่งค่าจำนวนปุ่มกดของ เมาส์มาให้

3.2 การอ่านค่าตำแหน่งของ เมาส์ โดยการ เรียกใช้ฟังก์ชั่น 11 โดยจะได้

- ค่าพิกัดของเมาส์ที่เคลื่อนที่ไปในแนวแกน x ใน Register cx
- ค่าพิกัดของเมาส์ที่เคลื่อนที่ไปในแนวแกน y ใน Register dx

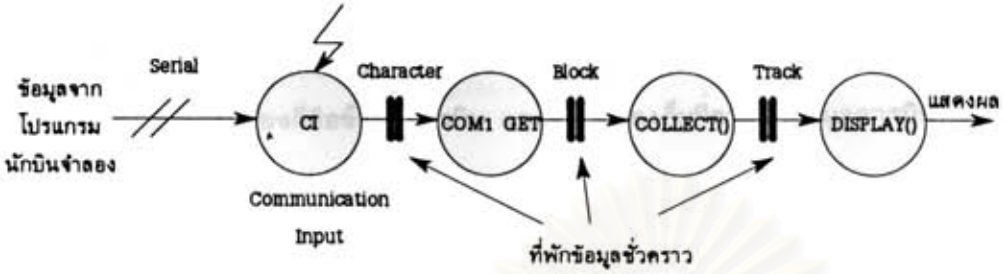
ทั้งสองค่านี้จะนำไปคำนวณเป็นพิกัดตำแหน่งบนหน้าจอต่อไป ซึ่งถ้าทั้งสองค่านี้ไม่เป็นศูนย์ แสดงว่าเมาส์มีการเคลื่อนที่ไปในแนวทิศใดทิศหนึ่งหรือทั้งสองทิศพร้อมๆกัน ก็จะนำค่านี้ไปรวมกับค่าตำแหน่งเดิมของเมาส์ที่เก็บไว้ และปรับการแสดงผลตัวชี้ที่หน้าจอด้วย

3.3 การตรวจสอบสถานะของปุ่มกดของเมาส์ โดยการเรียกใช้ฟังก์ชั่นที่ 3 โดยจะได้ผลดังนี้

โดยที่ บิทที่ 0 ถ้าเป็นหนึ่ง หมายความว่า ปุ่มซ้ายมือถูกกด
ถ้าเป็นศูนย์ หมายความว่า ปุ่มซ้ายมือไม่ได้กด
บิทที่ 1 ถ้าเป็นหนึ่ง หมายความว่า ปุ่มขวามือถูกกด
ถ้าเป็นศูนย์ หมายความว่า ปุ่มขวามือไม่ได้กด
บิทที่ 2 ถ้าเป็นหนึ่ง หมายความว่า ปุ่มกลางมือถูกกด
ถ้าเป็นศูนย์ หมายความว่า ปุ่มกลางมือไม่ได้กด

4. <u>โปรแกรมย่อยจัดการข้อมูลจากพอร์ทอนุกรม</u> การรับข้อมูลจากพอร์ทอนุกรมจะมีขั้นตอนที่ซับซ้อนเล็กน้อย เนื่องจากข้อมูลจากนักบินจำลองถูกส่งมาเป็นระยะๆโดยไม่สามารถกำหนดจังหวะเวลาที่แน่นอนได้ ดังนั้นทางด้านฝ่ายรับจึงต้องเตรียมพร้อมในการรับเสมอแต่ในบางครั้งก็ไม่สามารถยืนยันได้แน่นอนว่าระบบจะพร้อมในการรับข้อมูลอยู่ตลอดเวลา ดังนั้นจึงต้องหาวิธีการเพื่อให้โปรแกรมพร้อมในการรับข้อมูลอยู่เสมอ ซึ่งวิธีการที่เหมาะสมได้แก่การใช้โปรแกรมแบบฝังตัว "CI" ทำหน้าที่ตรวจจับข้อมูลที่รับมาจากพอร์ทอนุกรม

"CI" เป็นโปรแกรมแบบฝังตัวที่ได้พัฒนาขึ้นมาเองด้วยภาษา Pascal ซึ่งทำหน้าที่ดักจับข้อมูลที่รับมาทางพอร์ทอนุกรมผ่านทางการขัดจังหวะทางด้านฮาร์ดแวร์ เพื่อเก็บลงในที่พักข้อมูลชั่วคราวแบบวนรอบ (Circular Queue) โดยเรียงตามลำดับการมาถึงของข้อมูล เพื่อรอการอ่านข้อมูลจากโปรแกรมหลักอีกทีหนึ่ง ด้วยวิธีการนี้จะทำให้สามารถมั่นใจได้ว่าโปรแกรมจะสามารถรับข้อมูลได้อย่างครบถ้วน



รูปที่ 17 การรับข้อมูลจาก"นักบินสมมุติ"มาประมวลผล

โปรแกรม "CI" จะฝังตัวเองลงในหน่วยความจำ และรอเวลาที่จะกลับมาทำงาน โดยจะถูกกระตุ้นผ่านทางการขัดจังหวะทางด้านฮาร์ดแวร์เบอร์ 12 เมื่อมีข้อมูลเข้ามาทางพอร์ทอนุกรม โปรแกรมหลักจะอ่านข้อมูลออกมาจากที่พักข้อมูลชั่วคราวผ่านทางฟังก์ชั่นของ "CI" เมื่อโปรแกรมหลักว่างจากการทำงานปกติ จนกระทั่งได้รับรหัสจบกลุ่มข้อมูลก็จะเริ่มประมวลผลข้อมูลและแสดงผลต่อไป

ถ้าอ่านข้อมูลในที่พักข้อมูลหมดแล้วแต่ยังไม่ได้รับรหัสจบข้อมูลแสดงว่าข้อมูลยังถูกส่งมาไม่หมด โปรแกรมก็จะกลับไปทำงานอื่นก่อนแล้วค่อยวนกลับมารับข้อมูลในส่วนที่เหลือในช่วงเวลาต่อไปจนกว่าจะได้รับข้อมูลทั้งชุดแล้ว

การเรียกใช้ฟังก์ชั่นของ "CI" มีดังนี้

1. init เป็นการกำหนดค่าเริ่มต้นให้กับพอร์ทอนุกรมและลบข้อมูลทั้งหมดที่ค้างในที่พักข้อมูล เพื่อเริ่มต้นการทำงานใหม่

2. length เป็นการส่งกลับจำนวนข้อมูลที่ยังคงค้างอยู่ในที่พักข้อมูล เพื่อตรวจสอบว่ามีข้อมูลค้างอยู่ในที่พักข้อมูลเท่าใด

3. get เป็นการอ่านข้อมูลที่อยู่ในที่พักข้อมูล โดยเรียงตามลำดับก่อนหลังของข้อมูล ดังนั้นเมื่อต้องการอ่านข้อมูล จะต้องตรวจสอบว่ามีข้อมูลในที่พักข้อมูลหรือไม่

ข้อมูลการบินที่อ่านได้มาจะนำเก็บไว้ในที่พักข้อมูลชั่วคราวก่อน จากนั้นโปรแกรมย่อย "collect" จะทำหน้าที่ตรวจสอบความสมบูรณ์ของข้อมูล โดยทำการค้นหาจุดเริ่มต้นและจุดสิ้นสุดของข้อมูล ซึ่งมีตัวอักษรพิเศษ "" เป็นจุดเริ่มต้นและสิ้นสุดด้วยตัวอักษร "!" ข้อมูลที่อ่านได้ระหว่างตัวอักษรทั้งสองก็คือข้อมูลการบิน และจะนำจัดเก็บที่ตารางข้อมูลการบิน ก่อนที่จะนำไปปรับการแสดงผลต่อไป

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บทที่ 6

สรุปผลและ เสนอแนะ

ระบบนี้ได้ถูกออกแบบมา เพื่อช่วยปรับปรุงวิธีการฝึกสอนผู้ควบคุมจราจรทางอากาศให้สามารถทำการฝึกกับระบบควบคุมจราจรทางอากาศที่มี เรดาร์ได้ด้วย นอก เหนือไปจากการอำนวยความสะดวกในการฝึกแบบไม่ใช้ เรดาร์ โปรแกรมจำลอง เรดาร์นี้ได้ถูกพัฒนาขึ้นมาใช้งานภายในประ เทศโดยใช้ เครื่องคอมพิว เตอร์ส่วนบุคคลซึ่งมีราคาถูกแต่มีประสิทธิภาพสูง อีกทั้งสามารถ เลือก เครื่องมือที่ใช้ในการพัฒนาโปรแกรมได้ง่าย

แต่ เนื่องจาก เนื้อหาส่วนใหญ่ เกี่ยวข้องกับการควบคุมจราจรทางอากาศ ดังนั้นการการออกแบบระบบจึงแตกต่างจากระบบงานทางคอมพิว เตอร์ทั่วๆไป จำ เป็นต้องศึกษาข้อมูลและขั้นตอนในการทำงานต่างๆทั้งหมดอย่างระมัดระวัง เพื่อให้ระบบนี้สามารถทำงานได้ตรงตามที่ต้องการ โดยได้รับความร่วมมือ เป็นอย่างดีจาก งานฝึกอบรม ศูนย์ควบคุมจราจรทางอากาศทั้งทางด้านข้อมูล เอกสาร และข้อแนะนำต่างๆรวมไปถึงการทดสอบโปรแกรมในลักษณะ เช่น เดียวกับในการใช้งานจริง

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

วิธีการทดสอบการทำงาน

ในการทดสอบนี้ได้ใช้ เครื่องคอมพิว เตอร์ส่วนบุคคลยี่ห้อ AEG Olypia ซึ่งใช้หน่วยประมวลผลกลาง เบอร์ 486DX-33 สำหรับแสดงผลข้อมูลการบินด้วย เรดาร์สำหรับผู้รับการฝึกโดยติดตั้งกราฟิกการ์ดความละ เอียดสูงยี่ห้อ Artist WinSprint/400 และจอภาพขนาด 21 นิ้วยี่ห้อ Philips รุ่น Brillance 2110 ส่วน เครื่องของครูผู้ฝึกจะใช้ เครื่องคอมพิว เตอร์ส่วนบุคคลยี่ห้อ IBM PS/1 ซึ่งใช้หน่วยประมวลผลกลาง เบอร์ 386SX-20 และใช้จอภาพสีแบบ VGA 14 นิ้ว สำหรับทำหน้าที่จำลองข้อมูลการบิน (นักบินสมมุติ) โดยที่ เครื่องคอมพิว เตอร์ทั้ง

สองจะตั้งอยู่แยกจากกันคนละห้องไม่สามารถมองเห็นกันและกันได้ แต่เชื่อมโยงข้อมูลกันด้วยสายข้อมูล (RS-232) ผ่านทางพอร์ทอนุกรม โดยที่ทั้งครูผู้ฝึก(ที่ทำหน้าที่เป็นนักบินสมมุติ) และผู้รับการฝึก(ที่ทำหน้าที่เป็นผู้ควบคุมจราจรทางอากาศ) จะติดต่อพูดคุยผ่านทางระบบสื่อสารทางสาย (Intercom) โดยมีหูฟังและปากพูด (Headset) เช่นเดียวกับระบบวิทยุสื่อสารที่ใช้งานจริงที่ศูนย์ควบคุมจราจรทางอากาศ เพื่อให้สามารถจำลองการฝึกได้ใกล้เคียงมากที่สุด

ผู้รับการฝึก จะนั่งประจำที่ตำแหน่งจอแสดงผลเรดาร์ เพื่อเฝ้าดูการเคลื่อนที่ของเครื่องบินตามที่ครูผู้ฝึกได้กำหนดในแบบฝึกหัดการบิน และทำการควบคุมการจราจรทางอากาศตามหลักทฤษฎีทางวิชาการโดยการสั่งไปยังนักบินสมมุติ(ครูผู้ฝึก) ให้ทำการบินตามต้องการ ครูผู้ฝึกจะป้อนคำสั่งควบคุมการเคลื่อนที่ของเครื่องบินผ่านทางแป้นพิมพ์ตามคำสั่งการของผู้รับการฝึก ซึ่งโปรแกรมนักบินสมมุติจะนำคำสั่งการนี้ ไปคำนวณข้อมูลการบินให้โดยอัตโนมัติและส่งผลไปยังจอแสดงผลต่อไป สำหรับเที่ยวบินที่ไม่ได้สั่งควบคุมเป็นพิเศษ โปรแกรมก็จะคำนวณการบินตามเส้นทางบินที่ได้กำหนดไว้ในแบบฝึกหัดการบินโดยอัตโนมัติ

ด้วยวิธีการนี้ทำให้สามารถลดภาระในการทำงานของนักบินสมมุติไปได้อย่างมาก โดยครูฝึกสามารถควบคุมเครื่องบินได้หลายๆลำพร้อมกัน เพราะเที่ยวบินส่วนใหญ่จะบินตามเส้นทางบินอยู่แล้ว มีเพียงบางเครื่องเท่านั้นที่มีปัญหาและต้องบินออกนอกเส้นทางเพื่อจัดการจราจรให้เป็นพิเศษ

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ผลการทดสอบการใช้งาน

ระบบนี้ได้รับการทดสอบการทำงานโดยได้ทดลองใช้ร่วมกับการฝึกภาคปฏิบัติ ในหลักสูตรการควบคุมจราจรทางอากาศด้วยเรดาร์ ณ งานฝึกอบรม ศูนย์ควบคุมจราจรทางอากาศ บริษัทวิทยุการบินแห่งประเทศไทยจำกัด ทุ่งมหาเมฆ กรุงเทพฯ โดยครูผู้ฝึกจะจัดเตรียมแบบฝึกหัดการบินไว้ล่วงหน้า ดังตัวอย่างแบบฝึกหัดนี้

0101 UBA221 F28 A3341 270 420 115 BETNO G463 BKK

0102 THA603 EA30 A3535 310 480 256 BUTRA A1 UBL W1 KRT W1 BKK

0105 THA961 B747 A3342 370 500 115 BETNO G463 BKK

0106 CPA703 L101 A3537 280 480 256 BUTRA A1 UBL W1 KRT W1 BKK

0106 SAB273 DC10 A0702 370 490 073 BKK A1 SELKA A1 UBL A1 BUTRA

0107 THA982 B747 A2201 310 490 334 KAKET G463 RYN G463 BKK

0107 THA100 EA31 A0706 390 480 341 BKK A464 CMA

0107 CCA973 B767 A3310 330 470 161 CMA A464 BKK

0109 MAS82 B737 A2101 350 420 002 DIRAX A464 REG A464 BKK

0110 THA690 B737 A0710 330 420 024 BKK R474 VTN

0110 VIH222 B737 A3311 240 420 204 VTN R474 BKK

0115 THA504 EA30 A3343 330 480 092 DWI R468 BKK

แบบฝึกหัดนี้ เริ่มฝึกด้วย เวลา 01:00 มี เที่ยวบินที่ต้องทำการควบคุม 12 ลำโดยจะมีจุดที่ ผู้รับการฝึกจะต้องสั่งการ เพื่อจัดการจราจรหลายจุด โดยมีรายละ เอียด เกี่ยวกับ เที่ยวบินดังนี้ *

เที่ยวบินที่ยู เนี่ยนแอร์ UBA221 บินนำหน้า เที่ยวบินที่ไทยอิน เตอร์ THA961 ใน เส้นทางบิน กอล์ฟ463 และมีจุดหมายปลายทางที่ดอน เมือง เหมือนกันแต่มีระยะ เวลาห่างกัน 4 นาทีบิน แต่ เครื่องไทยอิน เตอร์ฯใช้ความ เร็วในการบิน 500 นอตซึ่งมากกว่ายู เนี่ยนแอร์ฯที่บินอยู่ข้างหน้าที่มีความ เร็ว เพียง 420 นอต นอกจากนี้ยังบินที่ระดับ เพดานบินสูงกว่าอีกด้วย ทำให้ เครื่องไทยอิน เตอร์ฯสามารถบินไล่ทันและลดระดับตัดหน้า เครื่องยู เนี่ยนแอร์ฯ เพื่อลงที่ดอน เมือง

เที่ยวบินที่ไทยอิน เตอร์ THA603 บินนำหน้า เครื่องคา เซ่ย์ CPA703 ใน เส้นทางบินอัลฟา 1 และวิสกี้ 1 โดยบินตามกันมามีระยะห่างกัน 4 นาทีบิน โดย เครื่องไทยอิน เตอร์ฯบิน

---

* ดูแผนที่ เส้นทางบินในหน้า 19 ประกอบ

ที่ระดับ เพดานบินสูงกว่าแต่มีจุดปลายทางที่ดอน เมือง เหมือนกัน ทำให้ต้องลดระดับตัดหน้า เครื่อง คา เช่ย์ฯ นอกจากนี้ยังมี เที่ยวบินที่ซาบีน่า SAB273 บินขึ้นจากดอน เมืองใน เส้นทาง เดียวกัน และได่ระดับตัดหน้าทั้งสอง เที่ยวบิน

เที่ยวบินที่ไทยอิน เตอร์ THA100 ออกจากสนามบินดอน เมือง เพื่อไป เชียงใหม่ใน เวลา เดียวกับ เที่ยวบินที่ไซน่าแอร์ CCA973 ออกจากสนามบิน เชียงใหม่ เพื่อมาลงยังดอน เมือง ซึ่งบินใน เส้นทางบิน อัลฟา 464 เหมือนกัน ทำให้มีการต้องได่ระดับและลดระดับตัดหน้าซึ่งกัน และกัน

เที่ยวบินที่ไทยอิน เตอร์ THA690 ออกจากสนามบินดอน เมือง เพื่อไป เวียงจันทน์ ใน เวลา เดียวกับ เที่ยวบินที่วิหค VIH222 ออกจากสนามบิน เวียงจันทน์ เพื่อมาลงยังดอน เมือง ซึ่งบินใน เส้นทางบิน โรมิโอ 474 เหมือนกัน ทำให้ต้องมีการได่ระดับและลดระดับตัดหน้าซึ่งกัน และกัน

นอกจากนี้ ยังมี เที่ยวบินที่ทำการบินมาลงยังสนามบินดอน เมืองอีก 3 เที่ยวบิน รวม เป็น 9 เที่ยวบินและบินขึ้นจากดอน เมืองอีก จำนวน 3 เที่ยวบิน โดยที่ เที่ยวบินทั้งหมดจะทำการบินขึ้นภายใน เวลา 01:15 รวม 12 เที่ยวบิน ซึ่งผู้ควบคุมจราจรทางอากาศจะต้องระมัดระวังในการจัดลำดับการจราจร ในการบินขึ้นและลงให้ถูกต้อง เพื่อให้ เที่ยวบินทั้งหมดสามารถบินตามกำหนดการได้มากที่สุด แบบฝึกหัดนี้ใช้ เวลาในการฝึกประมาณ 1 ชั่วโมง

<u>สรุปผลการทดสอบ</u>

<u>ในส่วนของโปรแกรมนักบินสมมุติ</u> โปรแกรมสามารถอ่านข้อมูลแบบฝึกหัดการบินและทำการจำลองข้อมูลการบินได้อย่างถูกต้อง ทั้งในการบินแบบอัตโนมัติและแบบควบคุมโดยครูผู้ฝึก โดยทำการแสดงผลข้อมูลการบินทั้งหมดที่หน้าจอด้วย เพื่อใช้ เป็นข้อมูลในการรายงานการบิน

<u>ในส่วนของโปรแกรมแสดงผล เรดาร์</u> โปรแกรมสามารถแสดงผลข้อมูลการบินในรูปแบบ เดียวกับระบบแสดงผล เรดาร์ที่ใช้งานจริงได้อย่างถูกต้อง และมีฟังก์ชั่นการทำงานพื้นฐาน เช่น เดียวกันโดยข้อมูลการบินนี้จะรับมาจากโปรแกรมนักบินสมมุติผ่านทางพอร์ทอนุกรมมาตรฐาน

ในส่วนของฐานข้อมูลการบิน สามารถจัด เตรียม และ เก็บในแฟ้มข้อมูลแบบตัวอักษร ได้ทั้งหมด ทำให้สามารถ เพิ่ม เติมหรือแก้ไขได้โดยง่ายโดยใช้โปรแกรมประมวลผลคำทั่วไป ซึ่งสะดวกในการใช้งาน

ข้อ เสนอแนะในการพัฒนาระบบต่อไป

จากการที่ได้นำโปรแกรมนี้ไปทดลองใช้งานได้พบข้อจำกัดในการทำงานบางประการ ซึ่งสามารถที่จะแก้ไขได้ โดยมีข้อ เสนอแนะดังนี้

1. โปรแกรมนักบินสมมุตินี้ เมื่อทำงานบน เครื่องที่ใช้ซีพียู 80386SX-20 จะทำงานได้ค่อนข้างช้า โดยจะสามารถมีจำนวน เครื่องบินในแต่ละแบบฝึกหัดได้ประมาณ 12 ลำ เท่านั้น เมื่อทดลองใช้ เครื่องที่มีความสามารถสูงขึ้น เช่น เครื่องที่ใช้ซีพียู 80486DX-33 จะมีผลทำให้โปรแกรมทำงานได้ เร็วขึ้น ส่วนโปรแกรมแสดงผล เรดาร์ที่ทำงานบน เครื่องที่ใช้ซีพียู486DX-33 สามารถทำงานได้รวด เร็ว เป็นที่น่าพอใจ ดังนั้น เครื่องที่ เหมาะสมในการใช้งานควรจะ เป็น เครื่องที่ใช้ซีพียู เบอร์ 486DX-33 ขึ้นไป จะทำให้การทำงานโดยรวม เร็วขึ้น

2. การส่งผ่านข้อมูล ซึ่งใช้พอร์ทอนุกรมสามารถส่งข้อมูลได้ในปริมาณจำกัด โดย เฉพาะแบบฝึกหัดการบินขนาดใหญ่ที่มีจำนวน เที่ยวบินมาก ทำให้ไม่สามารถส่งข้อมูลออกได้ทัน นอกจากนี้ยัง เป็นการรับส่งข้อมูลแบบจุดต่อจุด โดยสามารถมีนักบินสมมุติหนึ่งคนและผู้รับการฝึกหนึ่งคน เท่านั้น การแก้ไขสามารถใช้ เทคโนโลยีระบบ เครือข่ายท้องถิ่น (LAN) ช่วยในการส่งผ่านข้อมูลทำให้สามารถ เพิ่มจำนวนนักบินสมมุติและผู้รับการฝึกได้หลายคนพร้อมกัน เพื่อใช้จำลองการฝึกในแบบฝึกหัดการบินขนาดใหญ่ได้ โดยระบบ เครือข่ายท้องถิ่นจะมีความ เร็วในการส่งผ่านข้อมูลสูงถึง 10 ล้านบิทต่อวินาที ซึ่งทำให้สามารถส่งผ่านข้อมูลได้อย่างรวด เร็ว

การใช้ระบบ เครือข่ายนี้สามารถทำได้โดยการแก้ไขในส่วนโปรแกรมย่อยรับส่งข้อมูลใหม่ เท่านั้นโดยไม่ต้องแก้ไขโปรแกรมในส่วนอื่นๆ เลยทั้งนี้จากข้อดีของการออกแบบ เป็นส่วนย่อย

3. การป้อนข้อมูลของนักบินสมมุติ อาจใช้วิธีการอื่นๆซึ่งอำนวยความสะดวกในการป้อนข้อมูลได้มากกว่า เช่นการใช้ เมาส์ชี้ เพื่อ เลือกรายการที่ต้องการ ทั้งนี้ เนื่องจากผู้ทำหน้าที่ เป็นนักบินสมมุติบางคนยังใช้แป้นพิมพ์ไม่คล่องทำให้ไม่สะดวกในการควบคุม เครื่องบิน

ข้อจำกัด เหล่านี้มาจากปัญหาในทางปฏิบัติ อัน เป็นผลจากข้อจำกัดทางด้านฮาร์ดแวร์ ทั้งนี้ เนื่องจากต้องการให้ระบบ สามารถใช้งานได้กับ เครื่องคอมพิว เตอร์ทั่วไป และง่ายในการออกแบบระบบ ซึ่งข้อจำกัด เหล่านี้สามารถที่จะแก้ไขได้ ด้วยการ เลือกใช้อุปกรณ์ฮาร์ดแวร์ ที่เหมาะสม และแก้ไขซอฟต์แวร์ เฉพาะส่วนที่ เกี่ยวข้อง เท่านั้น

สรุปผลการทำวิทยานิพนธ์

โปรแกรมจำลองข้อมูลการบินด้วย เรดาร์ (Radar Simulator) นี้ เป็นโปรแกรมต้นแบบที่สามารถทำงานได้ตรงตามวัตถุประสงค์ที่ได้ออกแบบไว้ และสามารถนำไปใช้ในการฝึกหัดผู้ควบคุมจราจรทางอากาศได้ เป็นอย่างดี ซึ่งสามารถอำนวยความสะดวกในการฝึกได้ เป็นอย่างมาก โดยโปรแกรมจะคำนวณการ เคลื่อนที่ของ เครื่องบินและรายงานผลการคำนวณอยู่ตลอด เวลาแทนที่จะต้องคำนวณข้อมูลในการฝึกต่างๆ เอง นอกจากนี้ยังสามารถปรับปรุง หรือดัดแปลงแบบฝึกหัดการบินตามต้องการได้ง่าย ด้วยการแก้ไขแฟ้มข้อมูลการบินที่ เกี่ยวข้อง

ในส่วนของโปรแกรมที่ใช้ในการแสดงผล เรดาร์ ได้ใช้ระบบกราฟิกความละ เอียดสูง ซึ่งมีหน่วยประมวลผลทางด้านกราฟิกในตัว ทำให้สามารถจำลองการแสดงผลได้ เช่น เดียวกับจอ เรดาร์ที่ใช้งานจริงในปัจจุบันได้อย่างรวด เร็วโดยแสดงสัญญลักษณ์แทนตำแหน่งของ เครื่องบิน ซึ่ง เคลื่อนที่ตามตำแหน่งที่สมนัยกับ เครื่องบิน นอกจากนี้ยังสามารถใช้คำสั่งในการปรับการแสดงผล เช่นการขยายระยะการแสดงผล การ เลือกแสดงแผนที่ ฯลฯ ได้ด้วย โดยรับข้อมูลการบินมาจากนักบินสมมุติผ่านทางสายสื่อสารแบบอนุกรม

โปรแกรมทั้งหมดนี้ได้พัฒนาขึ้นใช้งานด้วยภาษา C และ PASCAL โดยใช้การแปลแบบ "LARGE MODEL" ด้วยการระบุในขั้นตอนการแปล เป็นออปเจกต์โปรแกรม ทั้งนี้ เนื่องจากโปรแกรมนี้ เป็นโปรแกรมขนาดใหญ่ * และใช้หน่วยความจำในการ เก็บข้อมูลจำนวนมาก

---

* ตัวโปรแกรมมีขนาดมากกว่า 64 กิโลไบต์

การพัฒนาระบบขึ้นมาเองมีข้อดีตรงที่ สามารถพัฒนาให้ตรงกับความต้องการได้ดีกว่า อีกทั้งยังสามารถดัดแปลง แก้ไข เพิ่มเติมความสามารถได้ง่าย ไม่ว่าจะเป็นทางด้านซอฟต์แวร์ หรือเลือกซื้อฮาร์ดแวร์ที่เหมาะสม อีกทั้งการเลือกใช้เครื่องคอมพิวเตอร์ส่วนบุคคลทำให้มีราคาย่อมเยากว่าการใช้เครื่องแบบสถานีงาน (Work Station) ขนาดใหญ่ที่มีราคาแพง

ระบบนี้นอกจากใช้ทำการฝึกแล้ว ยังสามารถนำออกแสดงตามนิทรรศการต่างๆ เพื่อเป็นการประชาสัมพันธ์ เผยแพร่อธิบายและแสดงการทำงานการควบคุมจราจรทางอากาศให้แก่ผู้สนใจ ได้ดีกว่าการอธิบายด้วยรูป หรือคำพูด เนื่องจากสามารถแสดงการทำงานให้ชมได้ทันที โดยได้นำออกแสดงระหว่างการประชุมสุดยอดทางด้านการบินในเขตเอเชียแปซิฟิก (Pacific Asia Coast/Regional Air Navigation) จัดประชุมทุกๆ 10 ปี จัดโดย ICAO ซึ่งประเทศไทยเป็นเจ้าภาพในปี 2536 นอกจากนี้ยังได้นำไปออกแสดงในงานนิทรรศการแสดงผลิตภัณฑ์ทางด้านการบินในภูมิภาคเอเชีย (Asia Aerospace'94) ที่ประเทศสิงคโปร์ และงานวันสื่อสารแห่งชาติที่ศูนย์ประชุมแห่งชาติสิริกิติ์ เป็นต้น อีกทั้งยังประหยัดเงินตราในการจัดซื้อเครื่องนี้สำเร็จรูปจากต่างประเทศอีกด้วย

ระบบนี้เป็นประโยชน์อย่างมากในการฝึกหัดผู้ควบคุมจราจรทางอากาศ ซึ่งจะส่งผลให้มีจำนวนบุคคลากรในการควบคุมจราจรทางอากาศมากขึ้น เพื่อให้สามารถรองรับการเจริญเติบโตของการคมนาคมทางอากาศของประเทศไทยในอนาคตได้เป็นอย่างดี

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

บรรณานุกรม

ภาษาไทย

ปริมาณการจราจรทางอากาศ ใน รายงานบริษัทประจำปี 2535. หน้า 45-46. กรุงเทพมหานคร: บริษัทวิทยุการบินแห่งประเทศไทยจำกัด, 2535

เมธี เสรีอรุโณ. ระบบจำลองเรดาร์ควบคุมจราจรทางอากาศ. ประชุมใหญ่วิชาการทางวิศวกรรมประจำปี 2536 หน้า 735-747 กรุงเทพมหานคร: วิศวกรรมสถานแห่งประเทศไทยในพระบรมราชูปถัมภ์, 2536

ภาษาอังกฤษ

Department of Aviation. Aeronautical Information Publication THAILAND. Bangkok, 1986

Henry W. Cole. Understanding Radar. London:Collins Professional and tecnical books, 1985

Herbert Schildt, The Art of C : Elegant Programming Solutions, California:Osborne/McGraw-Hill, 1991

J.C.F.Ritchie, Aircraft Navigation Aids, Macdonald and Co. (Publishers) Ltd, London: 1968

Kris Jamsa, Graphics Programming with Microsoft C and Microsoft Quick C, USA:Microsoft Press, 1990

SELENIA. CDS-80 User's Manual, 1990

Stephen K. O'brien, Turbo Pascal 6 the complete reference, California: Osborne/McGraw-Hill, 1991

Texas Instruments., TIGA Interface User's giude, 1991

Training section, Air traffic control center, Aeronautical radio of thailand ltd. Aircraft's Performance. Bangkok: 1990

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย

ประวัติผู้เขียน

นายเมธี เสรีอรุโณ เกิดเมื่อวันที่ 30 สิงหาคม พ.ศ.2507 ที่อำเภอเมือง จังหวัดสมุทรปราการ สำเร็จการศึกษาปริญญาตรีวิศวกรรมศาสตร์บัณฑิต สาขาวิศวกรรมสื่อสาร ภาควิชาวิศวกรรมไฟฟ้า คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าพระนครเหนือ ในปีการศึกษา 2530 และเข้าศึกษาต่อในหลักสูตรวิทยาศาสตร์มหาบัณฑิต ที่จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีพ.ศ.2534 ปัจจุบันทำงานเป็นเจ้าหน้าที่วิเคราะห์และพัฒนาระบบงานคอมพิวเตอร์ กองวิเคราะห์และพัฒนาระบบงานคอมพิวเตอร์ บริษัทวิทยุการบินแห่งประเทศไทยจำกัด รัฐวิสาหกิจสังกัดกระทรวงคมนาคม ซอยงามดูพลี ทุ่งมหาเมฆ สาธร กรุงเทพมหานคร

สถาบันวิทยบริการ
จุฬาลงกรณ์มหาวิทยาลัย