# การควบคุมจราจรทางอากาศแบบควบคุมพื้นที่

# AIR TRAFFIC FOR AREA CONTROL


โดย

นางสาวบุปผา วุฒิประสิทธิ์ 33100193


ปริญญานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

ภาควิชาโทรคมนาคม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์

ประจำปีการศึกษา 2536

เรื่อง การควบคุมจราจรทางอากาศแบบควบคุมพื้นที่

AIR TRAFFIC CONTROL FOR AREA CONTROL

ภาควิชา โทรคมนาคม

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ผู้จัดทำ

นางสาว บุปผา วุฒิประสิทธิ์ 33100193

----------อาจารย์ที่ปรึกษา

(อาจารย์ สมชาย จุฬะปิยะ)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

033147

เรื่อง การควบคุมจราจรทางอากาศแบบควบคุมพื้นที่

AIR TRAFFIC CONTROL FOR AREA CONTROL

โดย นางสาว บุปผา วุฒิประสิทธิ์

อาจารย์ที่ปรึกษา อาจารย์ สมยศ จุฬะปิยะ

## บทคัดย่อ

ปัจจุบันการขนส่งทางอากาศได้เจริญเติบโตขึ้นอย่างรวดเร็ว จนทำให้เกิดปัญหาด้านการจราจรคับคั่งในเขตท้องฟ้าของท่าอากาศยานสำคัญของโลก ดังนั้นจึงต้องมีการปรับปรุงระบบการควบคุมการจราจรทางอากาศ โดยการนำเอาเทคโนโลยีใหม่ๆเข้ามาใช้เพื่อความสะดวกรวดเร็วและป้องกันการเกิดอุบัติเหตุ อันเนื่องมาจากการชนกันของเครื่องบิน การควบคุมการจราจรทางอากาศในปัจจุบันใช้ระบบเรดาร์เป็นหลัก เนื่องจากสามารถบอกตำแหน่ง,ทิศทาง,ความเร็วและข้อมูลอื่นๆที่จำเป็นในการควบคุมการจราจร การฝึกหัดพนักงานควบคุมการจราจรทางอากาศจึงเป็นสิ่งจำเป็นมาก แต่เนื่องจากระบบจริงมีราคาแพงมาก และจะต้องทำงานตลอดเวลาทำให้ไม่สามารถใช้ในการฝึกได้ ดังนั้นจึงได้เสนอระบบการควบคุมการจราจรทางอากาศขึ้นมา โดยการจำลองเหตุการณ์การจราจรทางอากาศ และสร้างสถานการณ์ต่างๆ เพื่อใช้ในการฝึกเจ้าหน้าที่ควบคุมการจราจรทางอากาศ โดยให้ระบบดังกล่าวสามารถทำงานได้บนเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วๆ ไป

Nowadays,the air transportation was developed rapidly and it cause to make the problem in the Air Traffic Control (ATC) the airport.Improving air traffic control system,must be used new technology to prevent the air accident.

Radar is the main equipment used for Air Traffic control, because it can display aircraft's position direction,true-air-speed and the other important informations. Training the new air traffic controller is very essential. but A computer system that can simulate radar information and various situations is very expensive. so this thesis presents a computer system that can perform such function.The system is based on widely used personal computers and can be customized to meet the training system is requirements

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สารบัญ


| เรื่อง | หน้าที่ |
|---|---|
| บทที่ 1 | |
| - บทนำ | 1 |
| บทที่ 2 | |
| - วัตถุประสงค์ | 2 |
| บทที่ 3 การจราจรทางอากาศ | |
| 3.1 การแบ่งพื้นที่รับผิดชอบ | 3-5 |
| 3.2 การจราจรทางอากาศ | 5-10 |
| บทที่ 4 การควบคุมการจราจรทางอากาศในปัจจุบัน | |
| 4.1 การควบคุมการจราจรทางอากาศทุกวันนี้ | 11-20 |
| บทที่ 5 ระบบการหลีกเลี่ยงการชน | 21-24 |
| บทที่ 6 เจ้าหน้าที่ควบคุมการจราจรทางอากาศ | 25-27 |
| 6.1 การฝึกหัดเจ้าหน้าที่ควบคุมการจราจรทางอากาศ | 27-30 |
| บทที่ 7 การออกแบบโครงสร้างการควบคุมการจราจรทางอากาศแบบพื้นที่ | |
| 7.1 โครงสร้างระบบเมนู | 31-34 |
| 7.2 ลำดับการทำงานในแต่ละไฟล์ | 34-49 |
| บทที่ 8 องค์ประกอบทางด้านฮาร์ดแวร์และซอร์ฟแวร์ | |
| 8.1 ข้อจำกัดทางด้านเทคนิคที่ทำหน้าที่เป็นจอเรดาร์ | 50 |
| 8.2 คุณสมบัติของโปรแกรม | 51-53 |
| 8.3 การจัดเตรียมข้อมูล | 53 |
| บทที่ 9 การเขียนโปรแกรมควบคุมจราจรทางอากาศแบบควบคุมพื้นที่ | 54-108 |
| ผลการทดลอง | 109-117 |
| สรุป | 118-120 |
| แนวทางการพัฒนาต่อไป | 121-122 |



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 1
# บทนำ

จุดประสงค์หลักของการเขียนโปรแกรมฉบับนี้ขึ้นมาเพื่อใช้ในฝึกหัดควบคุมการจราจรทางอากาศสำหรับเจ้าหน้าที่ควบคุมการจราจรทางอากาศ โดยแนวทางในการเขียนโปรแกรมจะเน้นไปทางด้านให้ผู้ฝึกหัดการควบคุมการจราจรทางอากาศฝึกการตัดสินใจ และให้เกิดความเคยชิน เพื่อให้เกิดความประหม่าหรือความเครียดในการปฏิบัติงานจริงน้อยที่สุดเท่าที่จะเป็นไปได้ การเขียนโปรแกรมจะเน้นหนักไปในทางที่ทำให้ผู้ที่จะนำโปรแกรมไปใช้งานได้โดยง่ายและสะดวก ซึ่งภาษาที่ใช้ในการเขียนโปรแกรม คือ ภาษาซีและกราฟิกส์ภาษาซี เนื่องมาจากภาษาซีเป็นภาษาที่มีความคล่องตัว เป็นภาษาโครงสร้างที่มีฟังก์ชั่นการใช้งานจำนวนมาก นอกจากนี้ยังสามารถเขียนโปรแกรมให้เข้าถึงภาษาแอสแซมบลี้ได้อีกด้วย ลักษณะของโปรแกรมที่ทำการเขียนขึ้นมาจะทำงานในระบบเมนูป็อบ-อับ และพูลดาวน์(Pop-up and Pull-down Menu) มีลักษณะเป็นตัวเอดิเตอร์ของเทอร์โบซีหรือเทอร์โบปาสคาล ซึ่งเป็นระบบเมนูที่ให้เลือกคำสั่งโดยใช้ปุ่มลูกศร(Arrow Key) 4 ทิศทาง โดยที่ในจอภาพหนึ่งอาจมีได้หลายเมนูแต่จะมีเพียงหัวข้อเมนูเดียวเท่านั้นที่แอคทีพหรือกำลังถูกเลือกใช้งานอยู่ สามารถเลือกใช้เมนูอื่นๆได้โดยการกดปุ่มลูกศรซ้ายหรือขวา เมนูเก่าก็จะหายไปและจะปรากฏเมนูเก่าขึ้นมาแทน นอกจากนี้ในแต่ละเมนูสามารถเลือกเมนูย่อยได้อีก กล่าวคือ เมื่อมีการเลื่อกคีย์ลูกศรขึ้นหรือลงไปที่ตัวเลือกอันใหนก็จะเป็นการรีเวอร์สที่หัวข้อนั้น นอกจากนี้ยังได้เพิ่มเทคนิคในการสร้างแอตทริบิว(Attribute) ของหัวข้อเลือกให้เด่นชัดขึ้นมา ดังนั้นเมื่อมีการกดคีย์ลูกศรขึ้นหรือลงหนึ่งครั้งก็จะเกิดการปรับแอตริบิวของข้อเลือกสองข้อคือ การปรับแอตริบิวเดิมให้สู่สภาวะปกติ และการปรับแอตริบิวของหัวข้อเลือกใหม่ที่เลื่อนไปหาให้เด่นชัดยิ่งขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## จุดประสงค์

วัตถุประสงค์ของการทำปริญญานิพนธ์ฉบับนี้ขึ้นมา เพื่อใช้ในการฝึกหัดเจ้าหน้าที่ควบคุมการจราจรทางอากาศ ก่อนการไปปฏิบัติงานจริง อันเนื่องมาจากระบบการควบคุมการจราจรในปัจจุบัน ไม่สามารถใช้ในการฝึกสอนได้โดยตรงเพราะต้องทำงานตลอดเวลาและผิดพลาดไม่ได้เลย ดังนั้นจึงได้มีการเขียนโปรแกรมขึ้นมาโดยใช้ภาษาซีและกราฟิกส์ในการเขียน ซึ่งระบบดังกล่าวสามารถทำงานได้บนเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วๆไป โดยมีการสร้างสถานะการณ์ต่างๆขึ้นมาเพื่อให้เจ้าหน้าที่ควบคุมการจราจรฝึกหัดใช้ไหวพริบปฏิภาณในการตัดสินและให้เกิดความเคยชิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 3

## การจราจรทางอากาศ (Traffic In The Air)

เนื่องจากในปัจจุบันการคมนาคมทางอากาศได้มีการพัฒนาก้าวหน้ามากขึ้นตามลำดับ ดังจะเห็นได้จาก ปริมาณเที่ยวบินที่ได้เพิ่มขึ้นทุกๆปี ในปีที่ผ่านมามีเที่ยวบินสูงถึง 204,435 เที่ยวบิน สูงขึ้นจากปีก่อนๆ โดยเฉลี่ยถึง 31.74% (49,254 เที่ยวบิน)ดังนั้นงานให้บริการควบคุมจราจรทางอากาศก็ยิ่งมีความสำคัญมากยิ่งขึ้นตามไปด้วยเพราะเมื่อปริมาณเที่ยวบินสูงขึ้นมีผลทำให้ความหนาแน่นของเครื่องบินในแต่ละเส้นทางการบินมากขึ้นตามไปด้วยทำให้การจัดการควบคุมการจราจรทางอากาศ ให้เป็นไปตามข้อบังคับของ องค์การการบินระหว่างประเทศ ( International Civil Aviation Organization,ICAO) จึงมีความสำคัญยิ่ง ทั้งนี้เพื่อความปลอดภัย ความสะดวกรวดเร็วในการคมนาคมทางอากาศ

3.1 การควบคุมจราจรทางอากาศ แบ่งพื้นที่รับผิดชอบออกเป็น 3 ส่วน ดังนี้

1. ศูนย์ควบคุมพื้นที่(Area Control) เป็นการควบคุมอากาศยานที่ทำการบินอยู่ในเขตแถลงข่าวการบิน (Flight Information Control,FIR) ของประเทศนั้นๆ โดยมีความสูงในการบินตั้งแต่ 5,500 ฟิตขึ้นไป ซึ่งองค์การบินพลเรือนระหว่างประเทศ (The International Civil Avaition Organization หรือICAO)ได้ทำการแบ่งน่านฟ้าในโลกให้ประเทศต่างๆ ให้บริการควบคุมการบินโดยสามารถเรียกเก็บเงินค่าให้บริการได้ตามปริมาณเที่ยวบิน โดยทั่วไปในเขตแถลงข่าวการบิน ได้แก่ เขตพื้นที่ของประเทศนั้นๆเองรวมถึงบางบริเวณที่องค์การบินพลเรือนระหว่างประเทศ กำหนดให้รับผิดชอบเพิ่มด้วย (Area of Responsibility ,AOR) เช่น ในทะเล มหาสมุทร เป็นต้น โดยจะใช้เครื่องมือช่วยในการควบคุมจราจรทางอากาศ เช่น ระบบเรดาร์ เป็นต้น

2. ศูนย์ควบคุมประชิดเขต(Approch Control) เป็นการควบคุมการจราจรทางอากาศในเขตสนามบิน โดยมีพื้นที่รับผิดชอบ 35 ไมล์ทะเลโดยรอบสนามบินและมีความสูงในการบินไม่เกิน 5,500 ฟิต เป็นการจัดการจราจรสำหรับเครื่องบินที่จะบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หรือบินลงที่สนามบินนั้นๆจะให้บริการเองโดยจะติดต่อประสานงานกับศูนย์ควบคุมจราจรทางอากาศที่ควบคุมแบบพื้นที่อยู่ (Area Control) กับทางหอบังคับการ(Tower Control) โดยใช้เครื่องมือวัดเช่นเดียวกัน

3. แอโรโดรม คอนโทรล (Aerodrome Control) เป็นการควบคุมอากาศยานที่กำลังขึ้นลงในรัศมี 5 ไมล์โดยรอบสนามบินรวมถึงยานพาหนะที่อยู่บนลานจอดเครื่องบิน และทางวิ่งด้วยเป็นการควบคุมในแบบประชิดตัวมองเห็นได้ด้วยตา เพื่อให้เครื่องบินสามารถบินขึ้นลงได้อย่างปลอดภัย จากนั้นจะส่งมอบความรับผิดชอบให้กับศูนย์ควบคุมประชิดเขต ต่อไป (ในกรณีเครื่องบินขึ้น) บางทีเรียกว่า หอบังคับการ

สำหรับประเทศไทย มีหน่วยงานที่รับผิดชอบเกี่ยวกับงานควบคุมการจราจรทางอากาศอยู่ 3 หน่วยงาน ได้แก่

1. บริษัทวิทยุการบินแห่งประเทศไทยจำกัด รัฐวิสาหกิจ สังกัดกระทรวงคมนาคม ให้บริการควบคุมจราจรทางอากาศในแบบศูนย์ควบคุมแบบพื้นที่ เหนือเขตแถลงข่าวการบินกรุงเทพฯเอฟ ไอ อาร์ (Bangkok FIR) รวมถึงการควบคุมแบบประชิดเขตและแอโรโดรม ณ สนามบินนานาชาติที่หาดใหญ่ เชียงใหม่และภูเก็ต รวมไปถึงเขตความรับผิดชอบในทะเลจีนใต้กรุงเทพฯเอ โอ อาร์ (Bangkok AOR) รวมไปถึงบริการเกี่ยวเนื่องอื่นๆอีก เช่น ให้บริการเช่าใช้วิทยุโทรคมนาคมในเขตสนามบิน บริการให้เช่าใช้เครื่องเทอร์มินอล (Terminal) รับ-ส่ง ข่าวสารการบิน การซ่อมบำรุงระบบเรดาร์ที่ใช้ในงานควบคุมจราจรทางอากาศ เป็นต้น

2. การท่าอากาศยานแห่งประเทศไทย รัฐวิสาหกิจ สังกัดกระทรวงคมนาคม ให้บริการควบคุมจราจรทางอากาศ ในแบบศูนย์ควบคุมประชิดเขต ที่ทำการขึ้น-ลงที่บริเวณเขตสนามบินกรุงเทพฯ(ดอนเมือง) ซึ่งปัจจุบันได้โอนงานทางด้านการควบคุมการจราจรทางอากาศ มาให้แก่วิทยุการบินฯ ตั้งแต่วันที่ 1 กรกฎรคม 2536 คงเหลือแต่การให้บริการทางด้านภาคพื้นดิน

3. กรมการบินพาณิชย์ สังกัดกระทรวงคมนาคม ให้บริการเกี่ยวกับการควบคุมกฎระเบียบและข้อบังคับต่างๆในทางการบิน และให้บริการควบคุมการจราจรทางอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ให้แก่วิทยุการบินฯ และให้การควบคุมการจราจรทางอากาศในสนามบินอื่นๆในเขตต่างจังหวัด ซึ่งมีปริมาณเที่ยวบินไม่มากนัก และจะทยอยโอนความรับผิดชอบการควบคุมการจราจรทางอากาศให้แก่วิทยุการบินฯ ต่อไปในอนาคต

## 3.2 การจราจรทางอากาศ (Traffic In The Air)

การจราจรทางอากาศประกอบด้วยเครื่องบินโดยสาร เครื่องบินขนส่งสินค้า เครื่องบินส่วนตัว เครื่องบินฝึก เครื่องบินทหารและเครื่องร่อน ปัจจุบันในแต่ละวันมีเครื่องบินชนิดต่างๆบินขึ้นสู่ท้องฟ้าเป็นแสนๆเที่ยวบินทั่วโลก ในสภาวะปกติเครื่องบินเหล่านี้จะถูกควบคุมให้บินอยู่ภายในเส้นทางอากาศที่กำหนดไว้อย่างกวดขัน ถึงแม้ว่าเส้นทางดังกล่าวไม่สามารถมองเห็นได้อย่างชัดเจน งานควบคุมการจราจรทางอากาศ(Air Traffic Control) คือ การป้องกันไม่ให้เครื่องบินทั้งหลายเข้าใกล้จนเกิดอันตรายได้ งานนี้แยกออกได้เป็นสองส่วนใหญ่ๆ คือ

1. การกำหนดระยะห่างที่ปลอดภัยระหว่างเครื่องบินที่อยู่ใกล้เคียงกันในทิศทางตามแนวยาว(Longtitudnal) แนวดิ่ง(Vertical) และแนวด้านข้าง(Laternal)

2. การติดตามตำแหน่งจริงๆของเครื่องบินแต่ละลำในอากาศจากนาทีหนึ่งถึงอีกนาทีหนึ่งเพื่อให้เกิดความแน่ใจว่าเครื่องบินทั้งหลายยังรักษาระยะห่างของมันอยู่เสมอตามที่ถูกกำหนดไว้ให้

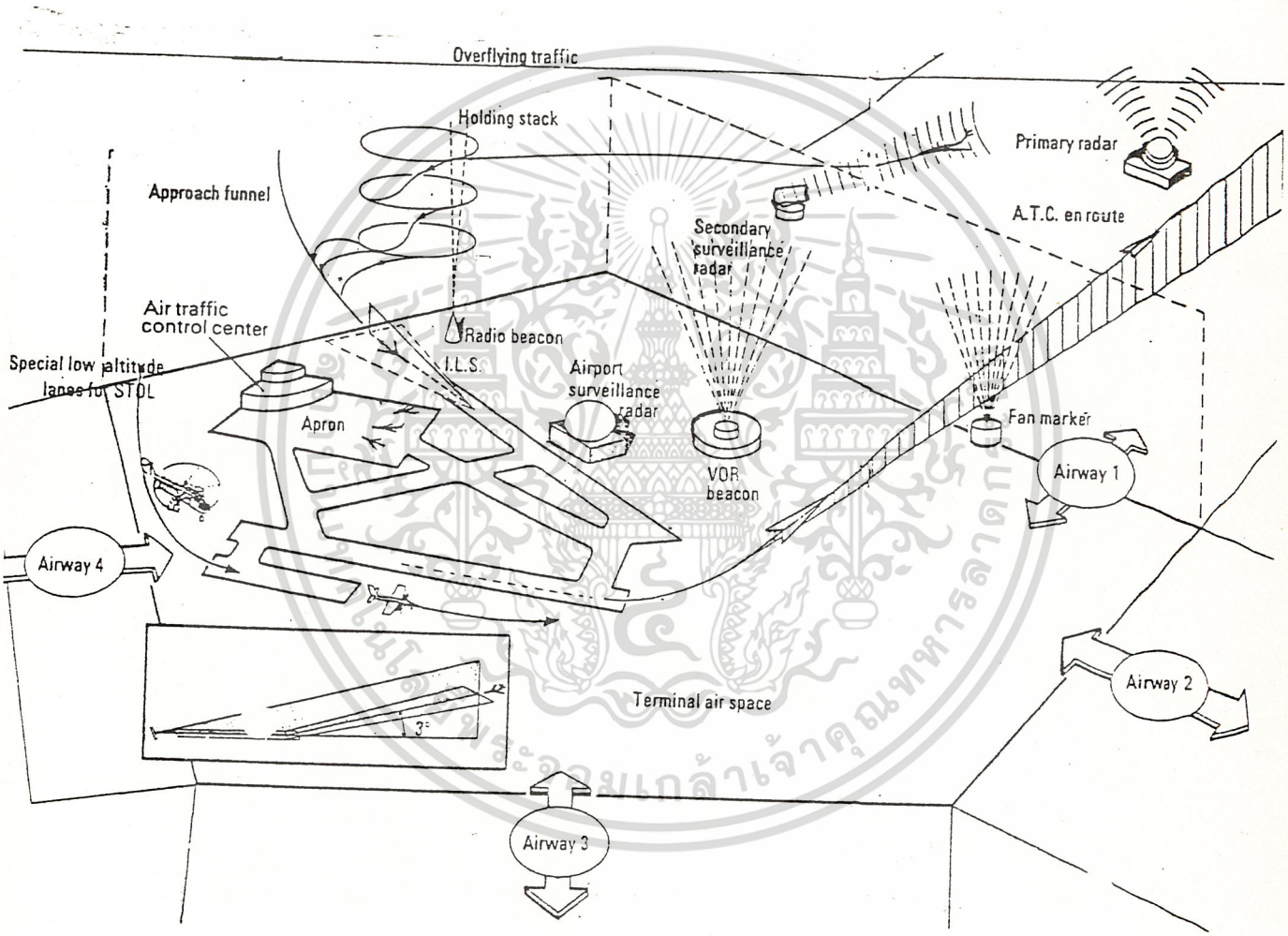
จากที่ได้กล่าวมาแล้วจะเห็นได้ว่า ปัญหาต่างๆที่เกี่ยวกับการจราจรทางอากาศจะยุ่งยากมากต้องอาศัยเทคนิคทางวิศวกรรมและการวางโครงสร้างที่สลับซับซ้อน ปัญหาที่พื้นๆอันหนึ่งในบรรดาปัญหาทั้งหลาย คือ การกำหนดตำแหน่งของเครื่องบินแต่ละลำ ณ วินาทีหนึ่งๆให้ละเอียดถูกต้องมากที่สุด ซึ่งปัญหาการกำหนดตำแหน่งให้ได้ละเอียดและแน่นอนนั้นขึ้นอยู่กับชนิดของเครื่องมือ ระยะห่างของเครื่องบินที่อยู่ห่างออกไป สภาพอากาศในเวลานั้นๆ ตลอดจนช่วงเวลาต่างๆของวัน(การเปลี่ยนแปลงของบรรยากาศทำให้การถ่ายทอดสัญญาณวิทยุเลวลงได้)บางครั้งความผิดพลาดเกิดขึ้นมากมายเป็นระยะทางถึงหลายไมล์ โดยเฉพาะการไม่รักษาเวลาและระดับเพดานบินที่ได้กำหนดไว้ก่อนการเดินทาง ทำให้เครื่องบินทุกลำบินอยู่ภายในปริมาตรแห่งความไม่แน่นอน(Volume of Uncertainly)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปัจจุบันกระแสการจราจรทางอากาศบนเส้นทางการบินสายต่างๆ คับคั่งมากขึ้นตามลำดับ ผลทำให้ปริมาตรแห่งความไม่แน่นอนอัดกันมากและเป็นอุปสรรคต่อความปลอดภัยในการขยายตัวของกระแสการจราจรทางอากาศ ดังนั้นจึงมีการปรับปรุงเทคโนโลยีหลายอย่างที่จะนำมาประยุกต์ในการควบคุมการจราจรทางอากาศ ที่มุ่งไปในการพิจารณาหาตำแหน่งของเครื่องบินในท้องฟ้าให้ถูกต้องและแม่นยำยิ่งขึ้น ทั้งนี้เพื่อลดปริมาตรความไม่แน่นอนให้เล็กลง เครื่องมือต่างๆที่ใช้ค้นหาตำแหน่งอาจติดตั้งไว้บนเครื่องบินหรือที่พื้นดินก็ได้ เช่นเดียวกับเครื่องเรดาร์และเครื่องคอมพิวเตอร์ขนาดใหญ่ๆ สำหรับคำนวณเส้นทางการบิน เครื่องบินในปัจจุบันจะมีเครื่องนำทางบินเพื่อใช้ในการวัดความเร็ว ความสูงและบอกตำแหน่งสัมพันธ์กับโลก ยิ่งไปกว่านั้นเครื่องบินบางลำจะมีเครื่องมือบอกตำแหน่งสัมพันธ์ของมันกับเครื่องบินลำอื่นๆ ซึ่งข้อมูลที่ได้กล่าวมานี้มีประโยชน์ในการป้องกันการชนกันในอากาศ แต่ในสภาวะความเป็นจริงนักบินอาจไม่ได้รับข้อมูลเกี่ยวกับเครื่องบินที่อยู่ใกล้เคียงอย่างเพียงพอ ทำให้นักบินต้องอาศัยความช่วยเหลือจากเจ้าหน้าที่ควบคุมการจราจรทางอากาศเพื่อทำหน้าที่ในการประสานงานการจราจร ตรวจสอบตำแหน่งสัมพันธ์ของเครื่องบิน และให้คำแนะนำแก่นักบินเมื่อถึงคราวต้องเปลี่ยนทิศทาง

ในการควบคุมการจราจรทางอากาศ นอกจากปัญหาทางอากาศแล้ว ยังมีปัญหาเกี่ยวกับวิธีการและระเบียบปฏิบัติบนพื้นดิน การขึ้นและการลงของเครื่องบิน ปัญหาการขนส่งผู้โดยสาร พนักงานการบิน กระเป๋าเดินทาง สินค้าและถุงไปรษณีย์จำนวนมากๆจากเครื่องบินไปยังตัวอาคาร ท่าอากาศยานหรือกลับกัน ปัญหาดังกล่าวซับซ้อนและยุ่งยากมากโดยเฉพาะกับเครื่องบินยักษ์ (Jumbo-Boeing747) ซึ่งการขึ้นลงครั้งหนึ่งๆอาจเกี่ยวข้องกับการขนส่งผู้โดยสารเข้าหรือออกเป็นจำนวนถึง 400-500 คน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




**รูปที่ 3.1 แสดงรูปท้องฟ้าเหนือท่าอากาศยานและเส้นทางการบิน**

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปแสดงบริเวณท้องฟ้า(Terminal Airspace) ของท่าอากาศยานและเส้นทางบิน(Airway) 4 เส้นทางที่ติดต่อกับมัน เส้นทางบินเหล่านี้รับทั้งการจราจรขาเข้าและการจราจรขาออกในเวลาคับคั่ง เครื่องบินที่กำลังบินเข้ามาอาจจะต้องบินวนรออยู่ในแสตก (Stack) รูปทรงกระบอกวงรีในอากาศจนกระทั่งแอบโพรค ฟูลเนว (Approach funnel) ว่างลง ถ้าทัศนวิสัยเลวหรือไม่แน่นอน การร่อนลงสู่ทางวิ่ง (Runway) จำเป็นต้องอาศัยระบบเครื่องช่วยการลง (Instrument Landing System) ดังแสดงในรูป เล็ก แถบคลื่นวิทยุโลคอลไลเซอร์ บีม (Localizer Beam) ซึ่งมีลักษณะเป็นรูปสามเหลี่ยมจุดๆ จะช่วยนำเครื่องบินให้ได้เส้นแนวทางการวิ่ง แถบคลื่นวิทยุไกลด์ สโลป บีม (Glide Slope Beam) ให้วิถีทางลง 3 องศาอย่างถูกต้อง เครื่องบินที่กำลังจะบินออกไปทั้งหมดจะอยู่ภายใต้การควบคุมของท่าอากาศยาน ซึ่งจัดการให้สิทธิการแท็กซี่และการบินขึ้น ตามด้วยการกำหนดความสูงและความเร็วก่อนที่เครื่องบินจะบินเข้าสู่เส้นทางเดินอากาศ ในขณะที่เครื่องบินกำลังบินอยู่บนเส้นทางการบิน เครื่องบินอยู่ภายใต้การควบคุมของศูนย์ควบคุมการจราจรทางอากาศตามรายทางอย่างต่อเนื่อง จุดเปลี่ยนมือการควบคุมจากศูนย์หนึ่งไปยังอีกศูนย์หนึ่ง จะมีประภาคารวิทยุ (Radio Beacons) เป็นเครื่องหมาย นอกจากนี้ท่าอากาศยานสมัยใหม่ยังมีทางวิ่งสั้นๆสำหรับเครื่องเฮลิคอปเตอร์และเครื่องบินเอส ที โอ แอล (STOL) อีกด้วย

การขนส่งทางอากาศมีบทบาทสำคัญมาก ในการพัฒนาเศรษฐกิจและสังคมของประเทศ เนื่องจากเป็นช่ายการคมนาคมที่เชื่อมโยงและครอบคลุมประเทศต่างๆ ทำให้สามารถขนส่งผู้โดยสาร สินค้าและพัสดุภัณฑ์ได้อย่างรวดเร็ว ทำให้การขนส่งทางอากาศเจริญรุดหน้าไปอย่างรวดเร็ว

ปัจจัยที่สำคัญประการหนึ่ง ที่ทำให้การขนส่งทางอากาศระหว่างประเทศเป็นไปด้วยความรวดเร็ว ปลอดภัย มีประสิทธิภาพและเป็นไปด้วยความเรียบร้อย คือ การมีบริการสื่อสารการบินหรือบริการวิทยุการบิน ในปัจจุบันเครื่องบินที่ใช้ในการขนส่งทางอากาศ ส่วนใหญ่จะเป็นเครื่องบินไอพ่นที่มีความเร็วสูงที่มีขนาดใหญ่ สามารถบรรจุผู้โดยสารได้จำนวนมาก ซึ่งมีรัศมีทำการบินได้ไกล และใช้เพดานบินสูง ทำให้การบริการสื่อสารจึงมีความจำเป็นมาก เนื่องจากการปฏิบัติงานทางการเดินอากาศเป็นงานที่เสี่ยงต่ออันตรายมาก ดังนั้นจึงต้องได้รับความช่วยเหลือจากหลายๆฝ่าย เช่น การจราจรทางอากาศ ข่าว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุตุนิยมวิทยา สนามบิน การค้นหา ช่วยเหลือและแถลงข่าวการบินซึ่งบริการสื่อสารการบินซึ่งเป็นจักรกลที่เชื่อมโยง สื่อข่าวที่จำเป็นต่อการเดินอากาศและบริการอื่นๆที่เกี่ยวข้องได้อย่างรวดเร็วและทันต่อเหตุการณ์ เพื่อความปลอดภัยของการเดินอากาศ

จากเหตุผลที่กล่าวมาข้างต้น ทำให้เห็นถึงความจำเป็นในการรักษาความปลอดภัยของงานการบิน ดังนั้นจึงต้องมีหน่วยงานที่รับผิดชอบโดยตรง นอกจากนี้ยังต้องสัมพันธ์ต่อหน่วยงานและข้อตกลงตามสากลด้วย เช่น บริษัทวิทยุการบินแห่งประเทศไทย จำกัด ประเทศไทยเป็นประเทศซึ่งเป็นภาคีสมาชิกขององค์การบินพลเรือนระหว่างประเทศ มีหน้าที่จัดบริการด้านความปลอดภัยในการขนส่งทางอากาศ รวมทั้งจะต้องจัดบริการให้แก่กิจการเดินอากาศในภูมิภาคเอเชียร์อาคเนย์ บริการเกี่ยวกับสนามบิน อุตุนิยมวิทยา การค้นหาและความช่วยเหลือ แถลงข่าวการบิน สื่อสารการบินและการจราจรทางอากาศ หน้าที่ที่บริษัทวิทยุการบินได้รับการมอบหมายจากรัฐบาลให้รับผิดชอบ คือ การให้บริการด้านการจราจรทางอากาศ(Air Traffic Service) และบริการด้านการสื่อสารการบิน (Aeronautical Communications) ภารกิจต่างๆด้านบริการการควบคุมการจราจรทางอากาศ(Air Traffic Control Service) ของบริษัทวิทยุการบินที่ได้รับความมอบหมายจากรัฐบาล เป็นผลสืบเนื่องมาจากการจัดตั้งและแถลงข่าวการบิน(Flight Information Region)ของแต่ละประเทศในการประชุมองค์การการบินพลเรือนระหว่างประเทศ ครั้งแรกของการประชุมย่านเอเชียใต้ เมืองนิวเดฮี 1984 (South Asia Regional Meeting, New Delhi,1984) งานด้านนี้มีเฉพาะเส้นทางการบินอีเอ็น (EN-route) ส่วนด้านเทอร์มินอลที่จัดการจราจรรอบๆแต่ละสนามบินนั้นเป็นงานของกรมการพาณิชย์ การท่าอากาศยานหรือทางทหาร ส่วนงานทางด้านการคมนาคมเกี่ยวกับการบิน (Aeronautical Communications) บริษัทวิทยุการบินจะรับผิดชอบอยู่ 2 แขนงคือ Aeronautical Fixed Service และ Aeronautical Mobile Service เพื่อให้กิจการขนส่งทางสายการบินต่างๆเป็นไปด้วยความปลอดภัยและรวดเร็ว ในการปฏิบัติภารกิจนั้น นอกจากจะต้องประสานกับหน่วยที่เกี่ยวข้องโดยตรง เช่น การท่าอากาศยานแห่งประเทศไทย กรมการบินพาณิชย์ กรมอุตุนิยมวิทยาแล้วยังมีการประสานงานกับหน่วยราชการทางด้านแอร์ ดีเฟ้น (Air Defence)และหน่วยบินต่างๆของทางราชการในการฝึกและปฏิบัติการ เพื่ออำนวยความสะดวก รวดเร็วและปลอดภัยในการบินทั้งทางทหารและพลเรือน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

องค์ประกอบที่สำคัญที่ช่วยให้การขนส่งทางอากาศเป็นไปด้วยความรวดเร็วและประหยัดนั้น ประกอบด้วยปัจจัยหลายอย่าง เช่น อากาศยาน อุปกรณ์ที่ติดตั้งบนอากาศยาน นักบิน เครื่องช่วยในการเดินอากาศ อุปกรณ์ที่ใช้ในการควบคุมการจราจรทางอากาศ เป็นต้น สำหรับประเทศไทยนั้น สนามบินดอนเมืองเป็นท่าอากาศยานสากล (International Airport) ซึ่งถือได้ว่าเป็นศูนกลางการขนส่งทางอากาศแถบเอเชียตะวันออกเฉียงใต้ เป็นจุดรวมของสายการบินพานิชย์ต่างประเทศมากกว่า 35 บริษัท ที่มาใช้บริการของสนามบินแห่งนี้ ดังจะเห็นได้จากแผนที่การเดินอากาศว่าสนามบินดอนเมืองมีเส้นทางการบินต่างๆมากมาย ต่อเชื่อมโยงไปยังสนามบินของประเทศเพื่อนบ้านใกล้เคียงและเชื่อมโยงไปทั่วโลก เส้นทางการบินนี้นับว่ามีความสำคัญต่อการขนส่งทางอากาศมาก ทั้งนี้เนื่องจากตามเส้นทางการบินที่ถูกกำหนดนั้นมีเครื่องช่วยในการเดินอากาศชนิดต่างๆ ติดตั้งไว้ที่พื้นดิน ทำให้นักบินที่บินผ่านเส้นทางบินแต่ละเส้นทาง สามารถทราบตำบลที่อยู่ของตนเองได้ถูกต้องตลอดเวลา โดยอุปกรณ์ที่ติดตั้งอากาศยานเป็นส่วนประกอบ

การสร้างเส้นทางการบินขึ้นนี้ ก็เช่นเดียวกับการสร้างถนนให้รถยนต์วิ่งสัญจรไปมา การจราจรบนท้องถนนนั้น หากไม่กำหนดระเบียบในการจราจรแล้วอาจมีอุบัติเหตุขึ้นได้ ดังจะเห็นได้ว่าในบริเวณที่มีทางแยกหรือตามวงเวียนต่างๆมักจะมีอุบัติเหตุขึ้นเป็นประจำทั้งที่มีตำรวจจราจรคอยให้สัญญาณกับผู้ขับขี่รถยนต์ ในทำนองเดียวกับท่าอากาศยาน ถ้าปล่อยให้บินไปอย่างไม่มีระเบียบก็ย่อมต้องมีอุปเหตุขึ้นได้ ดั้งนั้นการกำหนดให้อากาศยานบินอยู่ในเส้นทางการบินตามทิศทางที่ต้องการ ก็จำเป็นต้องมีเจ้าหน้าที่ควบคุมการจราจรทางอากาศคอยให้คำแนะนำอยู่ตลอดเวลา ทั้งนี้โดยอาศัยกฎเกณฑ์ในการจราจรทางอากาศที่เป็นมาตรฐารเดียวกัน อุปกรณ์อำนวยความสะดวกในการบินขึ้น-ลง สิ่งต่างๆเหล่านี้แตกต่างไปจากการจราจรทางอากาศมากมาย ที่เป็นเช่นนี้เพราะอากาศยานมีความเร็วสูงมาก หากนักบินไม่ปฏิบัติตามคำแนะนำของเจ้าหน้าที่ควบคุมการจราจรทางอากาศ อุบัติเหตุอาจเกิดขึ้นได้ในพริบตา ทำให้สูญเสียทั้งชีวิตมนุษย์และทรัพย์สิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 4

## การควบคุมการจราจรทางอากาศในปัจจุบัน

### 4.1 การควบคุมการจราจรทางอากาศทุกวันนี้ (Air Traffic Today)

การกำหนดเส้นทางการบินที่แน่นอนขึ้น เป็นสิ่งจำเป็นมากสำหรับการควบคุมการจราจรทางอากาศ จะต้องมีการกำหนดแผนที่สำหรับการเดินอากาศขึ้นเพื่อแสดงบริเวณท้องฟ้า(Airspace) ที่อยู่ภายใต้การควบคุมที่ว่างในท้องฟ้า บริเวณรอบท่าอากาศยานจะถูกกำหนดขึ้นเป็นเทอร์มินอล แอร์สเปส (Terminal Airspace or TMA)และระหว่าง TMA ที่สำคัญๆจะถูกกำหนดเป็นเส้นทางการบิน (Airways)ที่ว่างในท้องฟ้าเหล่านั้นจะมีขอบเขตความสูงต่ำสุดและสูงสุดและภายใต้ปริมาตรอากาศอันนี้ เครื่องบินทุกลำจะอยู่ภายใต้อำนาจการควบคุมของศูนย์ควบคุมการจราจรทางอากาศ (ATC) ศูนย์เหล่านี้มีฐานบังคับการอยู่ที่ท่าอากาศยานและตามเส้นทางการบินต่างๆด้วย เขตแดนระหว่างพื้นที่ภายใต้การควบคุมของแต่ละศูนย์จะถูกกำหนดไว้อย่างชัดเจน เครื่องบินทุกลำเมื่อบินข้ามตำแหน่งดังกล่าวจะต้องรายงานโดยทางวิทยุให้ศูนย์ที่เกี่ยวข้องทราบเพื่อไม่ให้เกิดความสับสนในการเปลี่ยนมือควบคุมที่พื้นดิน

การออกบินแต่ละเที่ยว ขั้นแรกที่สุด ต้องทำแผนการบิน(Flight Plan) แผนการบินนี้ต้องคำนึงถึงสภาพอากาศที่ทำนายไว้ ความเร็วและระดับความสูงที่จะบิน เวลาคาดคะเนที่จะถึงจุดหมายปลายทาง และน้ำหนักที่บรรทุกพร้อมกับปริมาณเชื้อเพลิงสำรองเผื่อการเปลี่ยนเส้นทาง เนื่องจากสภาพอากาศเลวหรือความคับคั่งของท่าอากาศยานปลายทาง ก่อนการเดินทางนักบินจะต้องแจ้งข้อมูลเหล่านี้ให้เจ้าหน้าที่ควบคุมการจราจรทางอากาศทราบ และจะทำเช่นเดียวกับศูนย์การควบคุมการจราจรทางอากาศ(Air Traffic Control)ทั้งหมดที่อยู่ตามเส้นทาง เมื่อจะทำการออกบิน การปฏิบัติจะต้องอาศัยกฎการบินด้วยทัศนวิสัย (Visual Flight Rules) ซึ่งเรียกสั้นว่า VFR หรือกฎการบินด้วยเครื่องวัด (Instrument Flight Rules) ซึ่งเรียกสั้นว่า IFR อย่าง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ใดอย่างหนึ่ง ทั้งนี้ขึ้นอยู่กับสภาพอากาศในเวลานั้น การใช้กฎ VFR จะทำเมื่อมีความมั่นใจว่าเขาสามารถเห็นเครื่องบินลำอื่นๆ ได้ในระยะปลอดภัยที่จะไม่เกิดการชนกันขึ้น สำหรับในสภาพอากาศที่เลว ระยะการมองเห็นสั้นมาก การกำหนดตำแหน่งและความปลอดภัยขึ้นกับเครื่องมือประกอบการบินแต่อย่างเดียว ดังนั้นการปฏิบัติต้องทำตามกฎ IFR

การติดต่อระหว่างศูนย์การควบคุมการจราจรทางอากาศกับเครื่องบิน ในปัจจุบันนี้อาศัยวิทยุเป็นตัวกลางที่สำคัญ เจ้าหน้าที่ควบคุมการจราจรทางอากาศจะส่งสัญญาณโดยทางวิทยุว่าเมื่อไรเครื่องบินสามารถเคลื่อนตัวเข้าสู่ทางวิ่งได้และเมื่อไรสามารถวิ่งได้ สำหรับวิถีทางไต่ขึ้น(Climbpath) หลังจากนั้นสิ่งนี้ขึ้นอยู่กับสมรรถนะ(Performance) ของเครื่องบิน น้ำหนักบรรทุกของเครื่องบิน ความเร็วของลม วิธีการของนักบินแต่ละคนและการจำกัดขอบเขตของกำลังเครื่องยนต์ให้เป็นไปตามระเบียบของเสียงรบกวน เมื่อเครื่องบินไต่ขึ้นไปถึงระดับความสูงที่ได้ตั้งใจไว้ก็จะเลี้ยวเข้าสู่เส้นทางบินอันแรก แล้วจึงไต่ขึ้นต่อไปอีกจนถึงระดับบินด้วยความเร็วคงตัว (Cruising Height) โดยปกติในระเบียบการบิน เครื่องจะต้องผ่านตำแหน่งรายงานอันที่หนึ่ง พร้อมกับทำการแจ้งให้ศูนย์การควบคุมการจราจรทางอากาศทราบ ตำแหน่งรายงานตัวนี้เป็นประภาคารสัญญาณวิทยุจากเครื่องส่งวิทยุอย่างใดอย่างหนึ่งที่พื้นดิน ตัวอย่างของเครื่องส่งวิทยุเหล่า ได้แก่ เครื่องส่งวิทยุ (Radio Transmitter),แฟน มาร์กเกอร์ (fan marker) และ วีเอชเอฟ เรดิโอ โอมนิเร้ง บีคอน (VHF Radio Omni-Range Beacon or VOR) เป็นต้น วิถีทางบินไต่ขึ้นนี้จะต้องไม่รบกวนหรือกีดขวางการจราจรตามแนวขวางหรือการจราจรตามเส้นทางหรือการที่กำลังลง อย่างไรก็ตามผู้ควบคุมที่พื้นดินผู้ซึ่งเฝ้าดูเครื่องบินอยู่ที่จอเรดาร์แสดงตำแหน่ง (PPI Radar Scopes) จะเป็นผู้บอกให้นักบินทราบถึงสภาพแวดล้อมเพื่อให้นักบินมีความแน่ใจในช่องว่าง สำหรับเครื่องบินในการเข้าไปสู่กระแสการจราจรของเส้นทางการบิน (ซึ่งก็ปฏิบัติคล้ายๆกับการขับรถยนต์แล่นเข้าไปร่วมกับกระแสรถยนต์ความเร็วสูงบนทางด่วน) กัปตันหรือหัวหน้านักบินเป็นผู้รับผิดชอบสำหรับวิถีทางไต่ขึ้นให้เป็นไปตามที่ได้จัดตั้งไว้ การบอกตำแหน่งของตัวเองอาจทำได้โดยอาศัยการแสดงผลของแผนที่ (moving map display) ของเครื่องเดินอากาศไฮเปอร์โบลิค เดคคา ( Hyperbolic Decca Navigator) หลังจากที่ได้ออกนอกรัศมีการควบคุมของทีเอ็มเอ (TMA) แล้ว โดยเฉพาะเมื่อบินอยู่เหนือทะเลทรายหรือมหาสมุทรกว้าง การรักษาเส้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทางการบินของเครื่องบินนั้นต้องพึ่งแหล่งข้อมูลการเดินอากาศของตัวเองมากขึ้น

ตามเส้นทางการบินต่างๆ เครื่องบินที่กำลังบินสวนทิศทางกันโดยปกติจะบินในระดับต่างกัน 1000 ฟุต สำหรับการบินในระดับตั้งแต่ 29000 ฟุต ลงมา แต่ถ้าสูงกว่านี้จะแยกระดับบินให้ต่างกันทุกๆ 2000 ฟุต ระดับความสูงของเครื่องบินวัดจากค่าจริงๆ ของความดันบรรยากาศ (Barometric Pressure) โดยปกติบรรยากาศรอบๆเครื่องบินที่กำลังบินอยู่มักจะผิดไปจากความเป็นจริงเพราะการรบกวนทาง Aerodynamic อย่างไรก็ตามค่าความดันที่ใช้บอกระดับความสูงเป็นค่าที่ได้แก้ไขอย่างถูกต้องแล้ว การวัดความสูงด้วยระบบเรดาร์ยังใช้กันไม่แพร่หลายในปัจจุบัน ทั้งบนเครื่องบินและที่หอควบคุมการจราจรทางอากาศเพราะราคาแพงมาก สำหรับการบินในทิศทางเดียวกัน เครื่องบินหลายๆลำที่บินอยู่ในเส้นทางเดียวกันอาจมีความเร็วต่างๆกัน เพื่อความปลอดภัยสำหรับกรณีนี้ กฎได้วางไว้ว่า เครื่องบินที่กำลังบินตามกันในเส้นทางเดียวกันจะต้องรักษาระยะห่างแนวนอนระหว่างกันอย่างกันอย่างน้อย 100 ไมล์ ถ้าบินด้วยความเร็วต่างกันเพียง 20 mph เครื่องบินอาจจะบินตามทันกันได้ในระหว่างการบิน 5 ชั่วโมง ด้วยเหตุนี้ในการบินทิศทางหนึ่ง บางครั้งจำเป็นต้องแยกเส้นทางออกเป็นหลายๆช่องทาง ซึ่งอาจจะทำในแนวขนานด้านข้างหรือในแนวดิ่งก็ได้ ดังแสดงในรูปที่ 4 แต่ที่นิยมกันมากจะแยกเส้นทางออกไปในแนวดิ่งเป็นชั้นๆ เพื่อที่จะได้มีช่องว่างอยู่เสมอสำหรับเครื่องบินลำใดก็ตามที่กำลังบินเข้าสู่เส้นทางบินได้ในช่วงเวลาน้อยกว่าทุกๆ 10 นาที

นอกจากนี้ ยังมีการแบ่งแยกระดับความสูงระหว่างความสูงชนิดต่างๆ ด้วยเหตุผลในแง่การประหยัดน้ำมันเชื้อเพลิง เช่น เครื่องบินใบพัด (Turboprops) มีประสิทธิภาพการบินดีที่สุดที่ระดับความสูง 15000 ฟุต และ 25000 ฟุต เครื่องบินไอพ่นมีความเร็วต่ำกว่าเสียง (Subsonic Jets) ระหว่าง 30000 ฟุตและ 45000 ฟุต และเครื่องบินเร็วกว่าเสียง (Supersonic) จะบินได้ประหยัดที่สุดในระดับความสูงระหว่าง 55000 ฟุต และ 65000 ฟุต เครื่องบินโดยสารไอพ่นส่วนใหญ่มักจะเพิ่มระดับความสูงขึ้นเรื่อยๆ ขณะบินไปเพราะน้ำหนักของมันจะลดลงเรื่อยๆ เมื่อน้ำมันเชื้อเพลิงได้ใช้ไปแล้ว ดังนั้นจึงเกิดปัญหาว่า ควรจะอนุญาตให้เครื่องบินขอเปลี่ยนระดับบินได้ด้วยตัวเลขถึง 10000 ฟุต หรือจะบังคับให้สายการบินยอมรับค่าใช้จ่ายสูงขึ้นโดยรักษาระดับการบินคงที่ บางครั้งการแบ่งแยกเส้นทางออกไปทางด้านข้างโดยไม่มีการแบ่งชั้นตามแนวดิ่งและเป็นการจราจรทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เดียวก็เป็นสิ่งจำเป็น การแยกออกทางด้านข้างนั้นจะนำไปใช้ได้ก็ต่อเมื่อเครื่องบินต่างๆมีเครื่องมือบอกตำแหน่งและทิศทางที่แน่นอน การหาตำแหน่งสามารถทำได้โดยวิธีการต่างๆหลายวิธี วิธีที่รู้จักกันดี คือ ลองเร้ง โลแร้น ซี (The Long-range LORAN C) ,การสังเกตุดวงดาว(ด้วย sextant หรือด้วยเครื่องมืออัตโนมัติ) และการคำนวณหาระยะทางโดยการคูณเวลากับความเร็วของเครื่องบินหลังจากที่ได้นำความเร็วของลมมาคิดแล้ว ซึ่งวิธีหลังสุดนี้เรียกว่าเดด เรคคอนนิ่ง ( Dead Reckoning หรือ DR)

ในปัจจุบันเครื่องบินโดยสารขนาดใหญ่ส่วนมากจะมี Automatic Pilot เมื่อเข้าใกล้จุดมุ่งหมายปลายทาง นักบินสามารถตั้ง Automatic Pilot ให้นำเครื่องบินลงสู่ตำแหน่งของสัญญาณวิทยุหรือ Radio Beacon ที่ติดตั้งอยู่ตามสนามบินต่างๆได้ ระบบการส่งสัญญาณแบบที่นิยมมากที่สุด คือ VOR/DME ซึ่งเป็น Omni-Range Beacon แบบที่ก้าวหน้ามาก ส่วน DME (Distance Measuring Equipment) เป็นเครื่องมืออยู่ในเครื่องบิน ซึ่งจะมีหน้าปัดแสดงตัวเลขให้นักบินทราบถึงระยะทางระหว่างเครื่องบินและประภาคาร(Beacon) ศูนย์กลางการควบคุมการจราจรทางอากาศที่ปลายทางขณะนั้นมีสำเนาแผนการบินเริ่มอยู่เรียบร้อยแล้วและจะทำการคำนวณหาเวลามาถึงของเครื่องบินให้ทันสมัยอยู่เสมอ จากข้อมูลบอกตำแหน่งจริงๆ ของเครื่องบินที่รายงานเข้ามา ข้อมูลเกี่ยวกับเครื่องบินนี้จะถูกเขียนลงบนกระดาษอย่างเรียบร้อยพร้อมด้วยข้อมูลการมาถึงของเครื่องบินลำอื่นๆด้วย ถ้าไม่มีการเสียเวลาเกิดขึ้น เครื่องมือนี้จะผ่านมือหน่วยควบคุมตามรายทางไปยังเทอร์มินอล แอร์สเปช (Terminal Airspace) แล้วต่อโดยตรงไปยังหน่วยควบคุมประชิดเขต (Approach Control Unit) ที่นั่นเจ้าหน้าที่ควบคุมการจราจรทางอากาศจะพูดติดต่อไปยังนักบิน โดยทางวิทยุความถี่สูงมาก(Very High Frequency) ขณะที่กำลังเฝ้าดูแสงกระพริบที่ปรากฏบนจอเรดาร์ของเซอร์เวลแลน เรดาร์ อิควิบเมนท์ (Serveillance Radar Equipment หรือ SRE)สำหรับสภาพการจราจรปกติมีเครื่องบินอย่างน้อย 5-6 ลำ กำลังเรียงแถวเข้าวิถีทางนำสู่ทางวิ่ง บางลำอาจได้วิถีทางที่สั้นที่สุด ในขณะที่ลำอื่นๆต้องกระจายกันออกไปเพื่อให้มีระยะห่างกันตามแนวนอน 3 ไมล์ทะเล (Nautical Miles) ตามกฎของ ATC ขณะที่เครื่องบินกำลังบินข้ามท่าอากาศยานเมื่อเข้ามาอยู่ระยะ 10 ไมล์จากทางวิ่ง นักบินสามารถใช้ระบบการลงสู่พื้นดินด้วยเครื่องมือ(Instrument Landing System) ซึ่งมีฐานบังคับอยู่ที่ท่าอากาศยาน ระบบการลงสู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พื้นดินด้วยเครื่องมือ(ILS) นี้ประกอบด้วยลำคลื่นวิทยุเข้มข้น 2 แนวด้วยกัน ดังแสดงในรูปที่ 1 อันหนึ่งจัดให้เป็นเครื่องช่วยนำเครื่องบินลงในแนวลาด 3 องศากับทางวิ่ง และอีกอันหนึ่งส่งไปเป็นระนาบแนวดิ่งผ่าเส้นกลางของทางวิ่ง เครื่อง ILS ส่งสัญญาณนำทิศทางไปยังนักบินโดยตรง หลังจากที่เครื่องบินได้ลงสู่ทางบินเรียบร้อยแล้วและความเร็วช้าลง เครื่องบินจะเลี้ยวออกจากทางวิ่งและแท็กซี่ไปยังบริเวณที่ว่างบนลานจอดตรงกับประตูสำหรับนำผู้โดยสารและกระเป๋าเข้าสู่ท่าอากาศยาน(Terminal Building)

ทุกๆ ขั้นตอนของระบบเส้นทางการบินมีความจุที่แน่นอน การรับงานเกินตัวจะนำไปสู่ความเสียเวลาหรือนำไปสู่มาตรฐานความปลอดภัยที่ต่ำลง เช่นเดียวกับการจราจรบนท้องถนน ความคับคั่งและความเสียเวลาส่วนใหญ่เกิดขึ้น ณ บริเวณท่าอากาศยานที่อยู่ใกล้เมืองใหญ่ๆ ซึ่งมีเส้นทางต่างๆ มาพบกัน ความจุของท่าอากาศยานขึ้นอยู่กับ

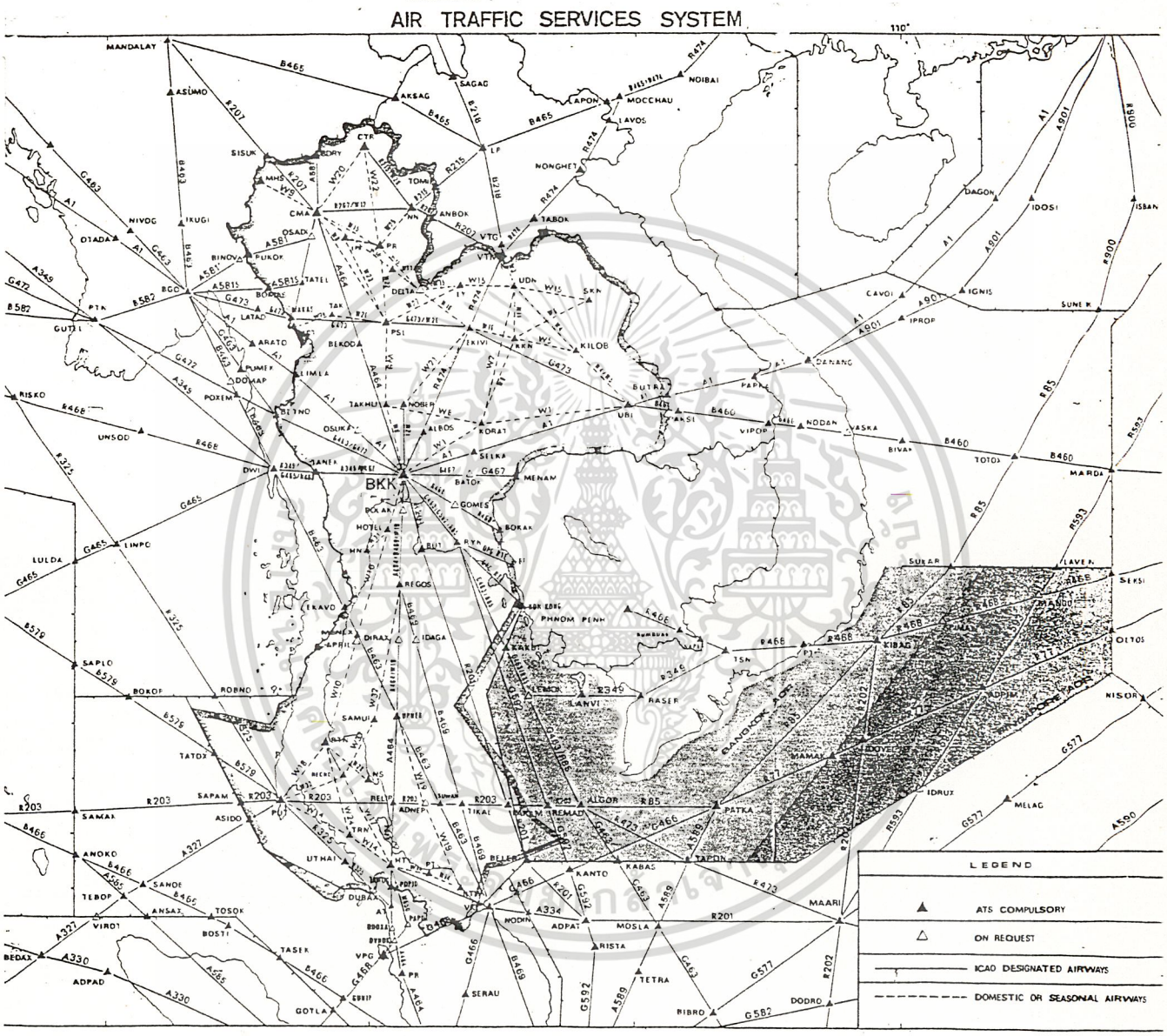
1. เวลาที่ใช้บนทางวิ่งของเครื่องบินที่มาถึง
2. ระยะเพื่อความปลอดภัยระหว่างเครื่องบินที่กำลังสู่พื้นดิน
3. การกระจายเครื่องบินที่กำลังบินเข้าสู่และออกไปจากสนามบินอย่างมีประสิทธิภาพ
4. จำนวนลานจอดเครื่องบิน

ความแตกต่างในความเร็วของเครื่องบินจากชนิดหนึ่งไปยังอีกชนิดหนึ่งจะทำให้การไหลของจราจรช้าลง ท่าอากาศยานที่มีการจราจรหนาแน่นและมีสภาพอากาศที่ทำให้ทัศนวิสัยเสียไป กฎการบินด้วยเครื่องวัด (IFR) อาจถูกบังคับใช้เป็นการถาวร เนื่องจากท่าอากาศยานเป็นคอขวดที่สำคัญที่สุดของกระแสการจราจร เมื่อการจราจรหนาแน่นเครื่องบินที่กำลังบินเข้าสู่ท่าอากาศยานจะถูกหน่วงไว้ให้บินวนเวียนอยู่บนลำเสา(Stack) ด้วยจำนวน 1,2 หรือ 3 stack ก็ได้ ทั้งนี้ขึ้นอยู่กับปริมาณของเครื่องบินที่ต้องบินรอ ลำเสาท้องฟ้าหรือ Stack นี้ปกติอยู่ห่างจากท่าอากาศยาน 10 ไมล์หรืออาจมากกว่า พื้นที่หน้าตัดของลำเสาดังกล่าวจะมีลักษณะเป็นรูปไข่ (Oval) ซึ่งมีความยาวของแกนยาวประมาณ 5 ไมล์และแกนสั้นประมาณ 3 ไมล์ เครื่องบินที่รอคอยการลงจะบินวนเวียนอยู่ด้วยความสูงคงตัวเป็นชั้นๆ แต่ละชั้นห่างกัน 1000 ฟุต เมื่อเครื่องบินล่าล่างสุดออกจาก Stack เพื่อบินเข้าสู่วิถีนำสู่ทางวิ่ง (Runway Approach Path) เครื่องบินล่าที่อยู่สูงออกไปหนึ่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชั้นจะบินลงมาแทนที่ เครื่องบินแต่ละลำภายในแสตก (Stack) ไม่สามารถแยกกันออกได้อย่างเด่นชัดด้วยแสงกระพริบบนจอเรดาร์พีพีไอ (PPI) ดังนั้นการรอคอยในแสตก ของเครื่องบินลำหนึ่งๆ ขึ้นอยู่กับการทำบัญชีของเจ้าหน้าที่ควบคุมการจราจรทางอากาศและการติดต่อทางวิทยุ การลดความเร็วของเครื่องบินลงระหว่างทางเพื่อให้เครื่องบินกระจายเวลาเข้าสู่สนามบินออกยังเป็นเรื่องที่เป็นไปไม่ได้ เพราะการทำเช่นนั้นจะทำให้เกิดความขัดแย้งต่อความเร็วที่มีประสิทธิภาพมากที่สุดของเครื่องบิน เมื่อเป็นเช่นนี้ความต้องการระบบการรอคอยในลำเสาท้องฟ้าหรือแสตกกิ้ง (Stacking)ก็ยังเป็นสิ่งจำเป็นอยู่ ดังนั้นความคับคั่งต่างๆ จึงฝากไว้ที่แสตก เหล่านี้บ่อยๆที่ความคับคั่งทำให้เกิดการเสียเวลา ซึ่งบางครั้งอาจนานถึง 5 หรือ 6 ชั่วโมงที่ต้องบินวนเวียนอยู่ในแสตก เครื่องบินลำใดน้ำมันเชื้อเพลิงใกล้หมดก็จำเป็นต้องเปลี่ยนไปลงยังท่าอากาศยานแห่งอื่น เส้นทางการบินต่างๆ จะถูกควบคุมอย่างใกล้ชิด การล่าช้ามักจะสะสมขึ้นและถ่ายทอดเรื้อรังจากจุดหนึ่งไปยังอีกจุดหนึ่ง ตามท่าอากาศยานที่สำคัญของโลก เครื่องบินต้องพบกับการเสียเวลานานถึง 6 ชั่วโมง เพราะความคับคั่งและสภาพอากาศเลว ทั้งนี้เนื่องมาจากหลายๆสาเหตุรวมกัน กล่าวคือ การเปลี่ยนในระเบียบไอ เอฟ อาร์ (IFR) การเกิดพายุและการเกิดเรดาร์ขัดข้อง แต่อย่างไรก็ตามสถานการณ์ต่างๆไม่ได้เกิดขึ้นบ่อยนัก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 4.1 แสดงลักษณะเส้นทางการบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

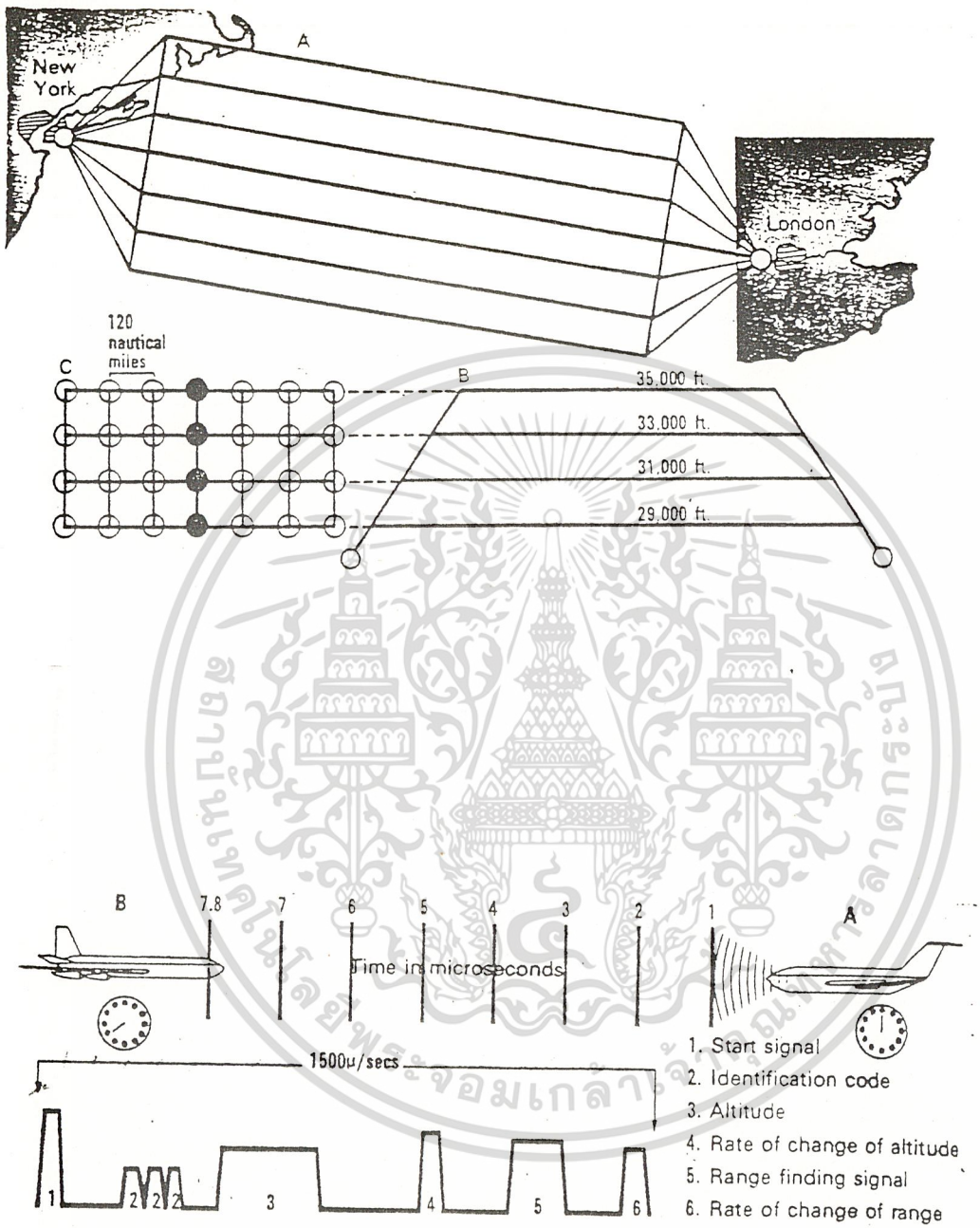
จากรูปเป็นความคับคั่งของการจราจรทางอากาศข้ามมหาสมุทรแอตแลนติกตอนเหนือระหว่างลอนดอนและนิวยอร์ก โดยที่

1. A แสดงการแบ่งเส้นทางการบินออกไปตามแนวนอนโดยมีระห่างด้านข้าง2 องศา (ประมาณ 120 ไมล์ทะเล) ช่องทางที่ดีที่สุดของวัน (ตรงกลาง) คำนวณจากระยะทางบินที่สั้นที่สุด เมื่อค่านิงสภาพอากาศที่ได้ทำนายไว้ ช่องทางบินอื่นๆจะกระจายออกรอบๆเส้นทางที่ดีที่สุด

2. B แสดงระดับชั้นสูงของชั้นบินที่นิยมกันในแง่การประหยัดน้ำมันเชื้อเพลิงซึ่งโดยทั่วๆไปมีอยู่ 3 หรือ 4 ระดับเท่านั้น

3. C แสดงให้เห็นถึงกลุ่มของช่องทางบินข้ามมหาสมุทรแอตแลนติกตอนเหนือโดยมีช่องทางที่ดีที่สุดอยู่ตรงกลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 5.1 แสดงรูประบบการหลีกเลี่ยงการชน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงระบบหลีกเลี่ยงการชนในอากาศโดยวิธีไทม์ ฟรีเควนซี(Time Frequency) ซึ่งมีนาฬิกาชนิดเที่ยงตรงยอดเยี่ยมที่มีอยู่ในเครื่องบินแต่ละลำ นาฬิกาทั้งสองลำจะถูกตั้งไว้ให้ตรงกัน ทุกๆ 3 วินาที เครื่องบินลำ A จะส่งพัลส์วิทยุเพื่อนำข้อมูลออกไปแสดงตนบอกตำแหน่ง ความสูงและเวลาที่พัลส์ส่งออก หลังจากนั้น 7.8 ไมโครวินาที เครื่องบินลำ B จะทราบได้ว่า เครื่องบินลำ A ที่ส่งพัลส์อยู่ในระยะปลอดภัยหรืออันตราย ถ้าอันตรายลำ B ก็จะหลบหลีกตามความเหมาะสม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 5

## ระบบการหลีกเลี่ยงการชน(Collision Avoidance Systems )

ปัญหาความปลอดภัยในการจราจรทางอากาศเป็นเรื่องที่มีความสำคัญมากที่ไม่อาจละเลยได้ ไม่ว่าจะสิ้นเปลืองเท่าไรก็ตาม ทั้งๆที่ระบบการควบคุมการจราจรทางอากาศจากพื้นดินสามารถเชื่อถือได้ เครื่องบินก็อาจเกิดการชนกันได้ เช่น ในปี พ.ศ.2503 มีอุบัติเหตุเกิดเครื่องบินชนกันในเขตท้องฟ้า TMA ของมหานครนิวยอร์กซึ่งเกิดขึ้นเนื่องจากเมฆหมอกหนาทึบ อย่างไรก็ตามในยามที่ท้องฟ้าแจ่มใสก็เคยมีอุบัติเหตุเกิดเครื่องบินชนกัน เช่น ที่ แกรน แคนยอน (Grand Canyon) เมื่อเดือนมิถุนายนของปี พ.ศ.2499 เกิดการชนกันในเส้นทางการบิน ซึ่งจะเห็นว่าโอกาสการเกิดการชนกันของเครื่องบินจะเพิ่มขึ้นตามปริมาณของการเพิ่มการจราจรทางอากาศด้วยอัตราเป็นอนุกรมเรขาคณิตส่วนสาเหตุอื่นที่มีส่วนด้วย ได้แก่ การมีความเร็วสูงขึ้นของเครื่องบินในปัจจุบัน การบินในทุกสภาพอากาศ ทัศนวิสัยจากห้องนักบินเลวมาก เครื่องมือต่างๆที่มีอยู่จำนวนมากในห้องนักบินและความคับคั่งอันเนื่องมาจากสายการบินทั้งหลายต่างก็ต้องการเส้นทางการบินที่ดีที่สุด ระบบหลีกเลี่ยงการชน(Collision Avoidance Systems) เป็นสิ่งที่ช่วยเสริมความปลอดภัยสำหรับการควบคุมการจราจรทางอากาศในอาณาเขตของ Terminal ที่คับคั่งมาก อีกทั้งยังเป็นเครื่องเฝ้าดู (Monitor) ท้องฟ้าที่สำคัญอันหนึ่งสำหรับนักบินที่บินอยู่นอกเขตเรดาร์ควบคุมจากพื้นดิน ปัญหาป้องกันเครื่องบินชนกันในอากาศเป็นเรื่องที่ยากลำบากมาก เพราะเครื่องบินมีความเร็วสูง อีกประการหนึ่งคือ เครื่องบินอาจจะบินอยู่ในระนาบแนวนอนที่ใกล้ชิดกันมากทั้งข้างบนและข้างล่าง เมื่อรู้ตัวว่าจะเกิดการชนกันขึ้นเครื่องบินไม่สามารถเปลี่ยนระดับการบินได้อย่างกระทันหัน เพื่อหลบหลีกอุบัติเหตุที่ใกล้จะมาถึงได้เสมอไป ซึ่งข้อกำหนดสำหรับการหลบหลีกการชนกันระหว่างเครื่องบินทั้งสองลำต้องการรู้ข้อมูลที่แน่นอนซึ่งไม่ได้มีอยู่ในระบบการเดินอากาศและมักจะมีการเปลี่ยนแปลงทุกๆนาทีด้วย ซึ่งจะเห็นได้ว่างานดังกล่าว เป็นงานที่มีความยุ่งยากมากในบรรดาการควบคุมการขนส่งทั้งหมด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

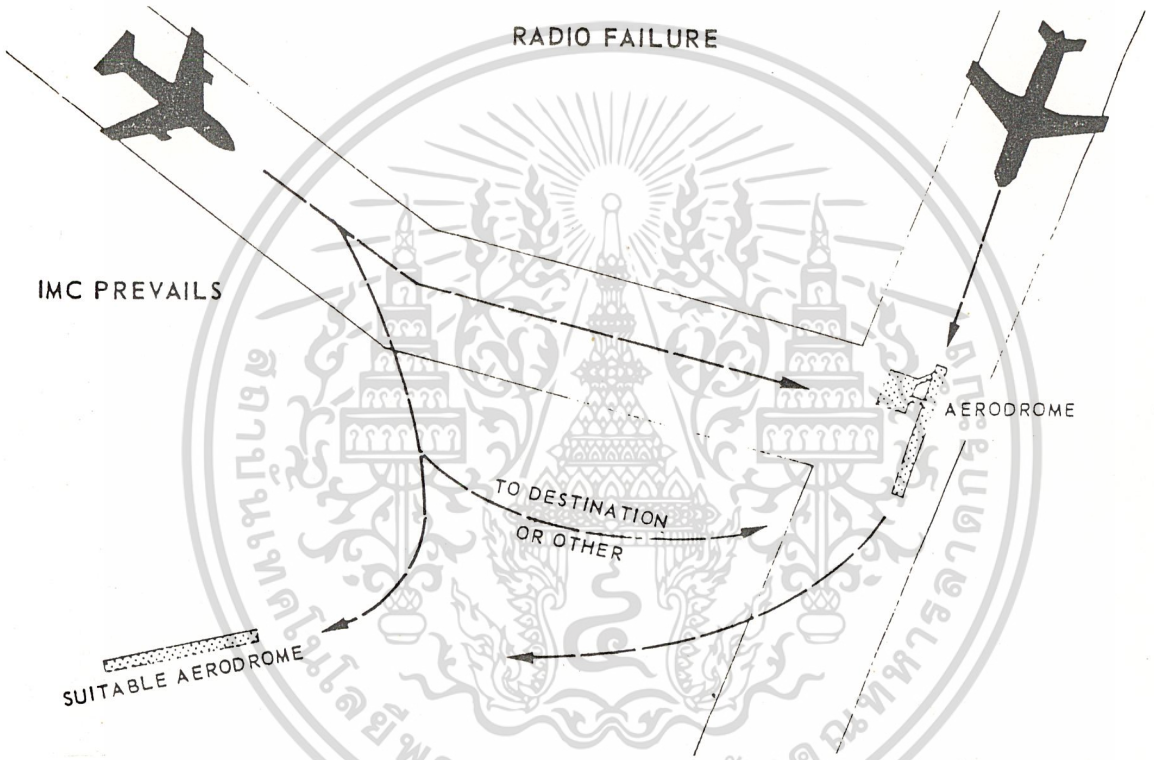
การค้นคว้าในเรื่องการหลีกเลี่ยงการชนกัน แบ่งปัญหาออกเป็น 5 เรื่องด้วยกัน คือ

1. การเตือนการชนกัน
2. การประเมินค่าของเหตุการณ์ที่จะเกิดขึ้น
3. การเลือกวิธีหลบหลีกที่ดีที่สุด
4. การให้เวลาเพื่อการหลบหลีก
5. การรับประกันไม่ให้เกิดการหลบหลีกที่ขัดแย้งกันระหว่างเครื่องบินทั้งสองลำที่กำลังจะชนกัน

เครื่องมือที่ง่ายที่สุดสำหรับงานนี้ คือ พรอกซิมิตี้ วอมมิ่ง อินดิเคเตอร์ (The Proximity Warming Indicator) ซึ่งใช้ค้นหาเครื่องบินที่อยู่ใกล้เคียงโดยทางอิเลคทรอนิกส์และคอยเตือนนักบินเมื่อมีสิ่งรุกล้ำเข้ามาภายในระยะที่กำหนดไว้ทุกทิศทางซึ่งความจริงแล้วเครื่องมือชนิดนี้ให้ประโยชน์น้อยมากเพราะนักบินไม่สามารถที่จะรู้ได้ว่าเครื่องบินจะเกิดการชนกันทิศทางใด การป้องกันการชนกันโดยการเผื่อปริมาตรไว้อย่างกว้างๆจะทำให้เสียเนื้อที่บนท้องฟ้ามากขึ้นด้วย ดังนั้นปัญหาทางเทคนิค คือ หาข้อมูลที่แน่นอนเกี่ยวกับเหตุการณ์ที่เกิดขึ้นให้มากขึ้น ลดระยะการเตือนภัย และการลดการเตือนภัยอย่างผิดๆให้น้อยลง ความล้มเหลวในการบอกภัยอันตรายทำให้ระบบการหลีกเลี่ยงการชนไร้ความหมาย และการเตือนภัยอย่างผิดๆ การเตือนภัยอย่างผิดๆมากจนเกินไปอาจทำให้เกิดอันตรายขัดแย้งกับลำอื่นๆที่บินอยู่อย่างปกติ อีกประการหนึ่งถ้าระบบการหลีกเลี่ยงการชนต้องการปริมาตรแห่งความไม่แน่นอนกว้างมาก สิ่งนี้ก็จะขัดแย้งกับมาตรฐานระยะห่างระหว่างเครื่องบินของระบบการควบคุมการจราจรและนำไปสู่การลดปริมาณการไหลของการจราจร

เนื่องจากระบบการหลีกเลี่ยงการชนกันในอากาศต้องทำงานในเมฆ วิธีทางเรดาร์และวิทยุเท่านั้นที่ได้รับการทดลองปฏิบัติดู เรดาร์แบบสะท้อนกลับทั่วๆไปไม่เหมาะสมเพราะความละเอียดเชิงมุมสำหรับการพิจารณาหาความแตกต่างในความสูงเลวมากประมาณ 1/3 องศา ปัจจุบันระบบที่กำลังดำเนินการอยู่ คือ การร่วมมือระหว่างเครื่องบินด้วยกัน ความถี่ของคลื่นวิทยุ 1545 เมกะเฮิตร์เป็นที่นิยมใช้กันมากที่สุด เครื่องบินแต่ละลำจะมีชื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 5.1 แสดงลักษณะการลงจอดของเครื่องบินเมื่อวิทยุใช้การไม่ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นรหัสตัวเลข ระบบนี้ใช้หลักการความถี่ของเวลา เครื่องบินทุกลำจะมีนาฬิกาปรมาณู (Atomic Clock)ที่ละเอียดถูกต้องถึง $1/10^{12}$ และทุกอันตั้งเวลาไว้ตรงกับเวลามาตรฐาน ระบบการหลีกเลี่ยงการชน (CAS)จะส่งพัลส์สอบถามทุกๆ 3 วินาที และเครื่องบินที่อยู่ใกล้เคียงทุกลำจะตอบกลับมาด้วยรหัสประจำตัวพร้อมกับความสูง ระยะห่างระหว่างกันวัดได้จากช่วงเวลาระหว่างสัญญาณที่ส่งออกไปและที่ตอบมาถึง เนื่องจากคลื่นวิทยุเดินทางด้วยความเร็ว 1000 ฟุตต่อหนึ่งไมโครวินาที (คือ 1/1 ล้านของนาที) ความละเอียดในการตั้งนาฬิกาให้เที่ยงตรงจึงเป็นเรื่องวิกฤติมาก นาฬิกาซีเซี่ยม(Cesium) นี้มีเสถียรภาพมากถึง 1 ส่วนใน $10^{12}$ ต่อวัน แต่สำหรับที่ใช้ในเครื่องบินมีราคาประมาณเรือนละ 2 แสนบาท ซึ่งเป็นราวร้อยละ 30 ของระบบ CAS ทั้งหมด

การประเมินหาปริมาตรอันตรายอย่างต่อเนื่องทำโดยอาศัยระยะสัมพันธ์ อัตราการเปลี่ยนระยะสัมพันธ์และอัตราการเปลี่ยนระดับความสูง นอกจากนี้ยังมีปัญหาอื่นๆอีก คือ คำสั่งของระบบการหลีกเลี่ยงการชน (CAS) อาจจะไปขัดแย้งกับคำสั่งของเจ้าหน้าที่ควบคุมการจราจรทางอากาศ และจากการหักหลบฉุกเฉิน เครื่องบินต้องหักหลบเข้าสู่เส้นทางที่ปลอดภัยอีกอันหนึ่งโดยไม่ให้เครื่องบินลำอื่นๆในบริเวณนั้นได้รับการรบกวน ถ้าเครื่องบินอยู่ใกล้พื้นดินมาก คำสั่งหลบของระบบการหลีกเลี่ยงการชนอาจจะนำไปสู่ความหายนะได้ หรือถ้าเครื่องบินต้องบินขึ้นด้วยความชัน คำสั่งเช่นนี้อาจทำให้เครื่องบินเสียการทรงตัวได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 6

## เจ้าหน้าที่ควบคุมการจราจรทางอากาศ

การเป็นเจ้าหน้าที่ควบคุมการจราจรทางอากาศจะต้องอาศัยความรู้ความเข้ามากเป็นพิเศษ ต้องมีประการณ์การทำงานที่เพียงพอจึงอาจกล่าวได้ว่า ต้องอาศัยความรับผิดชอบมากกว่านักบิน เนื่องมาจากบินรับผิดชอบเฉพาะอากาศยานที่ตนบินอยู่เท่านั้นในขณะที่เจ้าหน้าที่ควบคุมการจราจรทางอากาศต้องรับผิดชอบต่ออากาศยานทุกลำที่บินอยู่ในเขตรับผิดชอบ งานควบคุมการจราจรทางอากาศเป็นงานที่ทำให้เกิดความเครียด ดังนั้นจึงมีการจัดให้เจ้าหน้าที่ควบคุมการจราจรทางอากาศปฏิบัติงานในระยะเวลาสั้นๆ การปฏิบัติหน้าที่ของเจ้าหน้าที่ควบคุมการจราจรทางอากาศ ด้านประชิดเขตควบคุม (Approach Control) มีวัตถุประสงค์เช่นเดียวกับเจ้าหน้าที่ควบคุมการจราจรทางอากาศในด้านอื่นๆ กล่าวคือ

1. ป้องกันการชนกันระหว่างอากาศยานกับอากาศยานและระหว่างอากาศยานกับสิ่งกีดขวางอื่นๆ

2. เพื่อให้การจราจรทางอากาศดำเนินไปอย่างรวดเร็วและมีระเบียบ

การที่จะให้การเดินอากาศของอากาศยานเป็นไปด้วยความปลอดภัย มีความรวดเร็วและเป็นระเบียบตามวัตถุประสงค์ดังกล่าวข้างต้น เจ้าหน้าที่ควบคุมการจราจรทางอากาศจะต้องมีความรู้ความสามารถ ตลอดจนมีความเข้าใจในกฎเกณฑ์ที่เป็นไปตามมาตรฐานของ องค์การบินพลเรือนระหว่างประเทศ (ICAO)โดยเฉพาะอย่างยิ่งภาษาที่ใช้ในการติดต่อทางวิทยุโทรศัพท์ซึ่งมีอยู่ด้วยกัน 4 ภาษา คือ ภาษาอังกฤษ,ภาษาฝรั่งเศส, ภาษาเสปญ และภาษารัสเซีย ส่วนประเทศไทยนั้นใช้ภาษาอังกฤษในการติดต่อระหว่างเจ้าหน้าที่ควบคุมการจราจรทางอากาศ ส่วนวิธีการติดต่อกันนี้ทางองค์การได้กำหนดขึ้นและเสนอแนะให้ประเทศสมาชิกได้ถือปฏิบัติเหมือนๆกัน แต่ถึงกระนั้นก็ยังมีอุบัติเหตุเกิดขึ้นอันเนื่องมากการใช้คำพูดผิดระหว่างเจ้าหน้าที่ควบคุมการจราจรทางอากาศกับนักบิน อุปกรณ์ที่สำคัญที่ให้ประโยชน์กับนักบินและเจ้าหน้าที่ควบคุมการจราจรทางอากาศมาก คือ เรดาร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Radar)ที่ใช้ในกิจการควบคุมการจราจรทางอากาศได้แพร่หลายไปทั่วโลก ระบบที่ทันสมัย ได้แก่ เรดาร์(Radar)ระบบ Syntheic ซึ่งใช้คอมพิวเตอร์(Computer)ประกอบการทำงาน สามารถที่จะให้ข้อมูลต่างๆแก่เจ้าหน้าที่ควบคุมการจราจรทางอากาศ เช่น

- เป้าหมายและชื่อเรียกขานของอากาศยาน
- ตำบลและทิศทางที่อากาศยานเคลื่อนไป
- ระดับความสูงที่อากาศยานกำลังบินอยู่

จากข้อมูลที่เจ้าหน้าที่ควบคุมการจราจรทางอากาศ สามารถมองเห็นได้จากจอเรดาร์นี้จะนำมาพิจารณา และดำเนินการในการควบคุมการจราจรทางอากาศให้เป็นไปโดยปลอดภัย เช่น มีการจัดระยะต่อระหว่างอากาศยานกับอากาศยานหรืออากาศยานกับสิ่งกีดขวางและมีการแนะนำทราฟฟิค(Traffic)อื่นๆ ให้กับนักบินที่เกี่ยวข้องทราบภายในเขตพื้นที่รับผิด-ชอบ ตลอดจนให้คำแนะนำเกี่ยวกับสภาพทางอุตุนิยมวิทยาที่สามารถตรวจพบได้ด้วยเรดาร์การที่เจ้าหน้าที่ควบคุมการจราจรทางอากาศ จะได้ข้อมูลดังกล่าวโดยถูกต้องและครบถ้วนจำเป็นต้องอาศัยอุปกรณ์ที่ติดตั้งบนอากาศยานซึ่งมีความสัมพันธ์กัน ได้แก่ ทรานสปอนเดอร์ (Transponder) จากอุปกรณ์ดังกล่าวที่กล่าวมานี้ เจ้าหน้าที่ควบคุมการจราจรทางอากาศด้านประชิดเขตบริการควบคุมการจราจรทางอากาศ (Approach Control Service) จะให้คำแนะนำและควบคุมอากาศยานทุกเครื่องที่กำลังบินเข้าหาเครื่องช่วยการเดินอากาศ หลังจากนั้นจะแนะนำให้อากาศยานร่อนลงสู่สนามบิน ตามวิธีการร่อนลงสู่สนามบินที่ได้กำหนดและประกาศให้นักบินทราบทั่วกัน ซึ่งเครื่องช่วยในการเดินอากาศเหล่านี้สามารถที่จะให้ข้อมูลต่างๆแก่นักบินได้ เช่น ให้ทิศทางบินเข้าสู่สนามบินด้วยความปลอดภัย โดยมีเจ้าหน้าที่ควบคุมการจราจรทางอากาศยานเฝ้าติดตามมองดูในจอเรดาร์อย่างใกล้ชิด หากมีสิ่งมีผิดปกติเกิดขึ้น เจ้าหน้าที่ควบคุมการจราจรทางอากาศจะแจ้งให้นักบินทราบได้ สำหรับการให้คำแนะนำต่ออากาศยานที่กำลังบินออกจากสนามบินเช่นกัน เจ้าหน้าที่ควบคุมการควบคุมการจราจรทางอากาศด้วยการกำหนดทิศทาง ระยะสูงเพื่อที่จะให้อากาศยานนั้นๆสามารถบินเข้าสู่ทางบินตามความประสงค์ได้อย่างรวดเร็ว โดยอาศัยเรดาร์เป็นอุปกรณ์ในการนำทางโดยตลอด เมื่ออากาศยานนั้นบินเข้าสู่ทางบินเรียบร้อยแล้ว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นักบิน ก็จะอาศัยเครื่องช่วยบินในการเดินอากาศที่มีอยู่ทำการบินออกไปได้โดยถูกต้องและปลอดภัย ทั้งนี้จะมีเจ้าหน้าที่ควบคุมการจราจรทางอากาศเป็นผู้ให้คำแนะนำตลอดเส้นทางการบินด้วยความปลอดภัย รวดเร็ว และมีระเบียบตรงตามจุดมุ่งหมายที่วางไว้

## 6.1 การฝึกหัดเจ้าหน้าที่ควบคุมการจราจรทางอากาศ

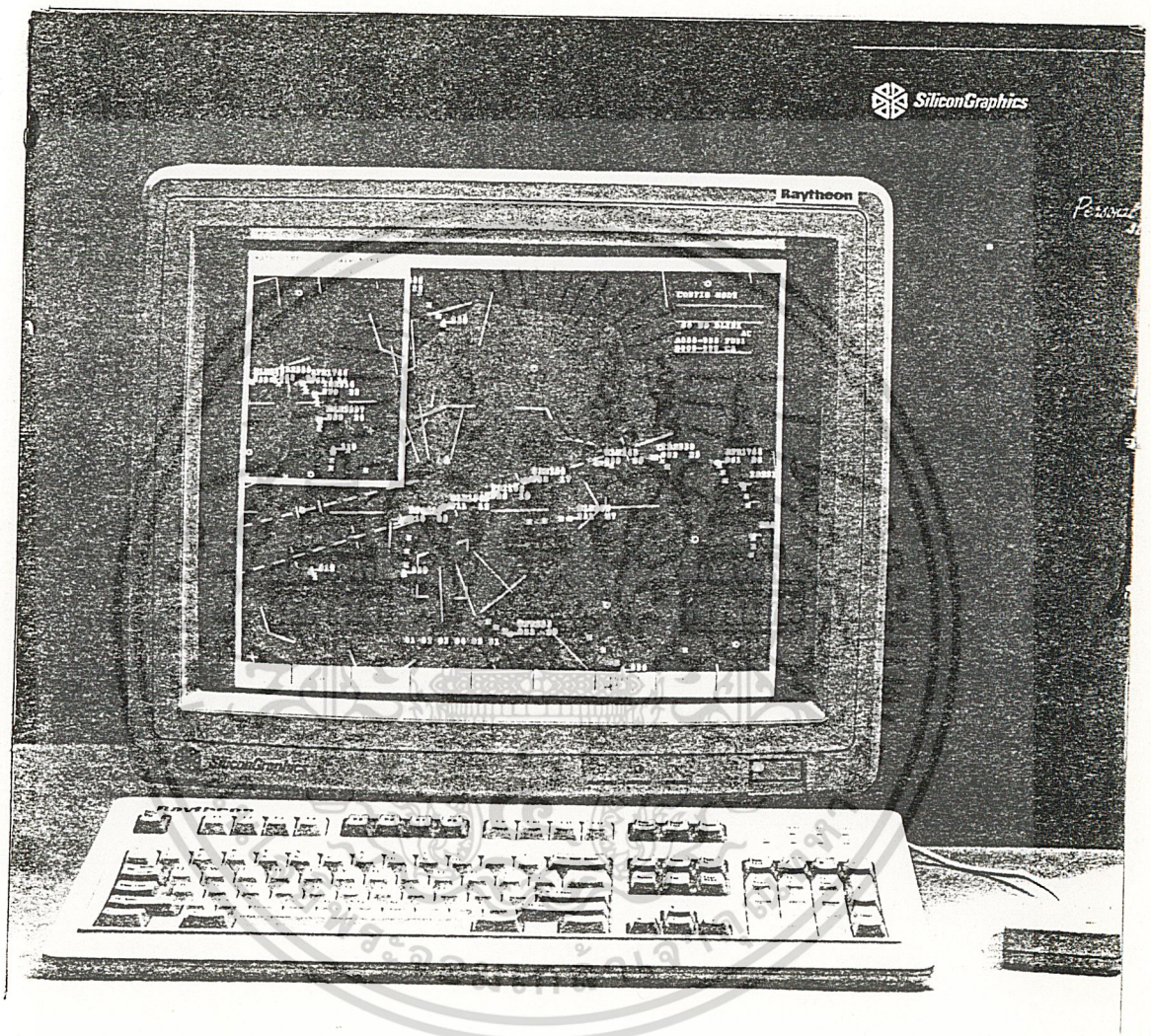
พนักงานควบคุมงานจราจรทางอากาศ จะต้องมีความสามารถในการควบคุมจราจรทางอากาศเป็นอย่างดี ต้องผ่านการฝึกอย่างหนัก กว่าจะขึ้นมารับผิดชอบในตำแหน่งนี้ได้ซึ่งจะได้รับการฝึกอย่างหนักทั้งทางร่างกายและสภาพจิตใจ นอกเหนือไปจากความรู้ความสามารถทั้งทางด้านทฤษฎีและปฏิบัติ ทั้งนี้เนื่องจากงานควบคุมจราจรทางอากาศจะต้องทำงานตลอด 24 ชั่วโมงโดยแบ่งเป็นกะ ดังนั้นพนักงานทุกคนจึงต้องมีสุขภาพดีตลอดเวลา ทั้งทางด้านร่างกายและจิตใจ

การทำงานในฐานะพนักงานควบคุมจราจรทางอากาศ จึงมีความสำคัญเป็นอย่างยิ่ง ความผิดพลาดในการทำงานหมายถึงความสูญเสียครั้งยิ่งใหญ่ ในปัจจุบันพนักงานควบคุมการจราจรทางอากาศยังมีไม่เพียงพอในการรับหน้าที่ในปัจจุบัน อันเนื่องมาจากภาวะการเจริญเติบโตของประเทศค่อนข้างสูง ปริมาณเที่ยวบินเพิ่มขึ้นทุกปี ดังนั้นจึงต้องมีการฝึกพนักงานควบคุมจราจรทางอากาศขึ้นใหม่เพื่อทำงานทดแทนอยู่ตลอดเวลา ซึ่งในแต่ละรุ่นมีผู้ที่ผ่านการฝึกเพียงไม่กี่คน อีกทั้งยังต้องไปฝึกหาประสบการณ์ในการทำงานกับระบบจริงอีกเป็นเวลานาน กว่าจะขึ้นมารับหน้าที่เต็มตัวได้ ในปัจจุบันพนักงานที่ได้รับการฝึกการควบคุมการจราจรทางอากาศ จะต้องผ่านหลักสูตรการควบคุมจราจรทางอากาศขั้นพื้นฐานจากศูนย์ฝึกการบินพลเรือนในประเทศไทย นอกจากนี้จะเข้ารับการตรวจสุขภาพทั้งทางร่างกายและจิตใจที่สถาบันเวชศาสตร์การบิน หลังจากนี้จะเข้าปฏิบัติงานในตำแหน่งพนักงานข่าวการบินเป็นลำดับแรก มีหน้าที่เตรียมข่าวการบินให้กับพนักงานควบคุมการจราจรทางอากาศ เพื่อให้คุ้นเคยกับสภาพการทำงานเป็นกะและสภาพการทำงานของการควบคุมจราจรทางอากาศ หลังจากได้ทำงานระยะหนึ่งก็จะคัดเลือกผู้ที่มีความชำนาญเข้ารับการฝึกขั้นสูงต่อไปหรือส่งไปฝึกยังต่างประเทศ ซึ่งมีความพร้อมทั้งทางด้านบุคลากรและเครื่องมือมากกว่าซึ่งกว่าจะขึ้นมาทำงานจริงได้จะต้องใช้เวลาไม่ต่ำกว่า 10 ปี ปัญหาในการผลิต

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บุคลากรด้านนี้เรื่องหนึ่ง คือ การขาดแคลนเครื่องมือฝึก โดยเฉพาะเครื่อง เรดาร์ ซิมมูเลชั่น ซึ่งเป็นการจำลองการทำงานของระบบเรดาร์ควบคุมการจราจรทางอากาศที่ทำงานคล้ายระบบจริง เพื่อฝึกหัดให้พนักงานควบคุมการจราจรทางอากาศมีความชำนาญในการทำงานกับระบบเรดาร์ แต่เนื่องจากระบบนี้มีราคาค่อนข้างสูง จึงยังมิได้มีใช้ในปัจจุบัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 6.1 แสดงภาพระบบหน้าจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 6.2 แสดงลักษณะภาพภายในห้องควบคุมจราจรทางอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 7
# การออกแบบโครงสร้างเรดาร์ ซิมมูเลชั่น

## 7.1 โครงสร้างระบบเมนู

โครงสร้างการออกแบบระบบการทำงานของเมนู สามารถแสดงได้ดังรูปที่ 6.1 ซึ่งสามารถแบ่งเป็นข้อเมนูหลัก(Main Menu) 4 หัวข้อดังนี้ คือ ไฟล์ (File),เรดาร์สโคป(Radar Scope),รายละเอียด(Detail)และควิต(Quit) ซึ่งสามารถอธิบายได้ดังนี้

ไฟล์เมนู (Menu File)

เป็นเมนูป๊อบ-อัปที่ประกอบไปด้วยเมนูย่อยอีก 2 เมนู คือ

1. เมนูเปิดแผนการบิน (Open Flight Plan Menu) เมื่อเลื่อนเคอร์เซอร์มาทับเมนูเปิแผนการบินแล้วเกิดกดคีย์รีเทอร์นหรือการกดคีย์พิเศษจะเกิดแถบสว่างขึ้นแล้วปรากฏรายละเอียดเกี่ยวกับพารามิเตอร์ที่ใช้ในการออกบินของเครื่องบิน เช่น เส้นทางการบิน ความเร็ว ระดับเพดานบิน ฯลฯ

2. เมนูปิดแผนการบิน (Close Flight Plan Menu)เมื่อมีการเลือกเมนูนี้จะเป็นการปิดแผนการบิน

ไฟล์เรดาร์สโคป (Radarscope)

ประกอบไปด้วยเมนูย่อย 4 เมนู คือ

1. เมนูแผนที่ (Map menu) เมื่อมีการเลือกเมนูย่อยนี้จะปรากฏพล็อตจุดแผนที่ประเทศไทยและเพื่อนบ้าน ซึ่งสามารถเลือกได้ว่าต้องการให้แสดงแผนที่ถึงจุดใหนได้ การแสดงทำได้โดยการเปิดแฟ้มเก็บข้อมูล

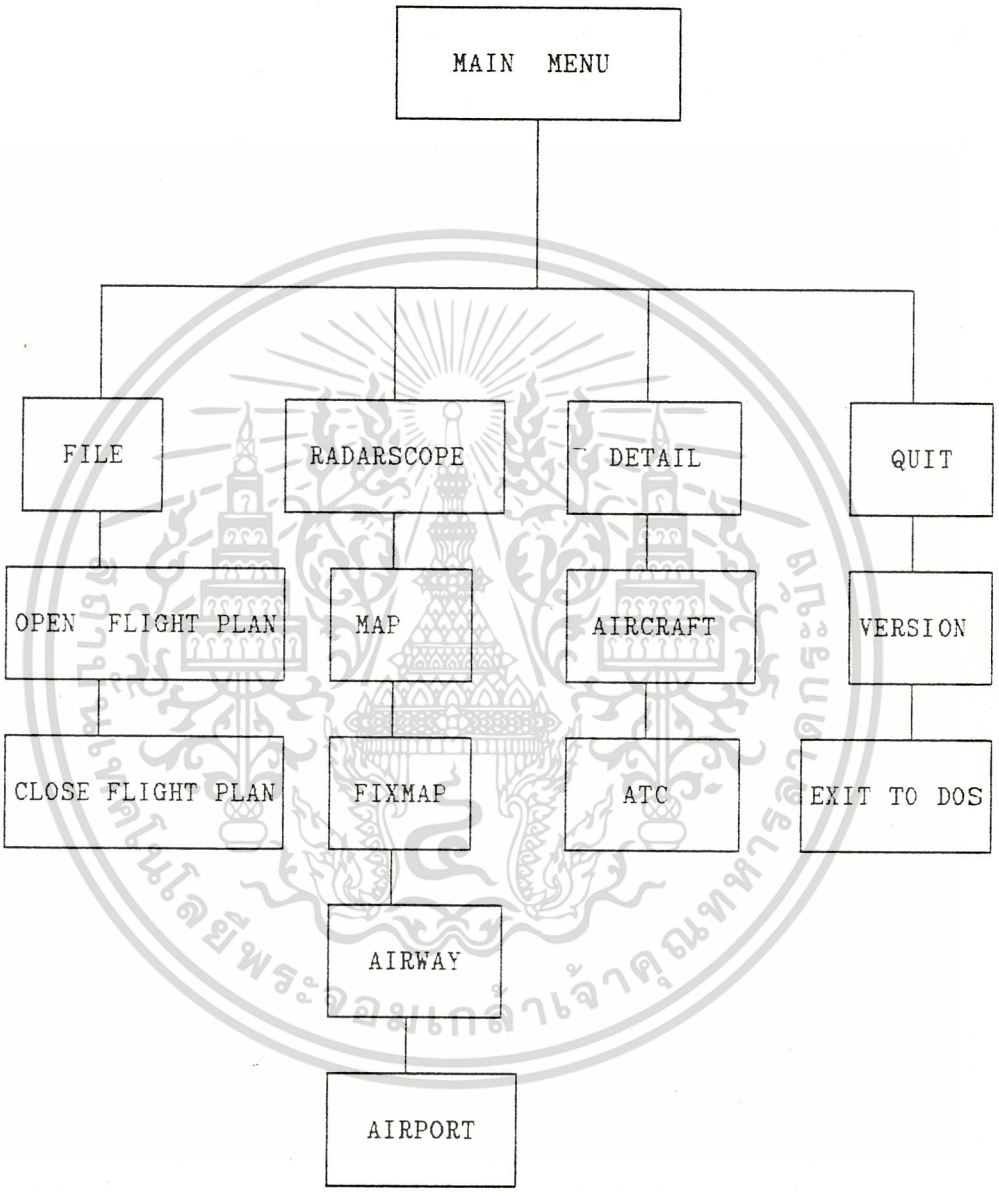
2. เมนูจุดรายงาน (Fixmap Menu) ประกอบไปด้วยส่วนที่แสดงชื่อจุดรายงานในบริเวณเขตการรับผิดชอบของศูนย์ควบคุมการจราจรทางอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3. เมนูเส้นทางการบิน (Airway Menu) ประกอบไปด้วยส่วนที่แสดงถึงเส้นทางการบินเหนือเขตควบคุมการรับผิดชอบของศูนย์ควบคุมการจราจรทางอากาศ

4. เมนูท่าอากาศยาน (Airport Menu) จะแสดงบริเวณอาณาเขตของท่าอากาศ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 7.1 แสดงบล็อกไดอะแกรมของโครงสร้างระบบเมนู

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไฟล์รายรายละเอียด ( Detail)

ประกอบไปด้วยเมนูย่อย 2 เมนู คือ

1. เมนูแอคราฟ (Aircraft Menu) เป็นเมนูที่แสดงรายละเอียดเกี่ยวกับสมรรถนะของเครื่องบินลำต่างๆ อาทิ ความเร็วที่ใช้ในการบิน น้ำหนัก ความกว้างของลำตัว ความยาวของลำตัว ปริมาณความจุผู้โดยสาร ฯลฯ

2. เมนูเอทีซี (ATC)

ไฟล์ออก (Quit)

ประกอบไปด้วยเมนูย่อย 2 เมนู คือ

1. เมนูเวอร์ชั่น (Version Menu) เมื่อมีการกดแถบสว่างไปที่เมนูนี้จะปรากฏรายละเอียดเกี่ยวกับผู้จัดทำโปรแกรม

2. เมนูกลับไปสู่ดอส (Exit to Dos) เป็นเมนูที่แสดงการกลับไปสู่ดอส เมื่อมีการกดคีย์ดังกล่าว จะปรากฏข้อความ "Are you sure exit to dos Y/N ?" ถ้าต้องการออกไปสู่ดอสก็กดตัวอักษร

6.2 ลำดับการทำงานในแต่ละไฟล์

สามารถแสดงการทำงานได้ตามโฟล์ชาร์ต (Flow Chart)

6.2.1 โปรแกรมหลัก (Main Program)

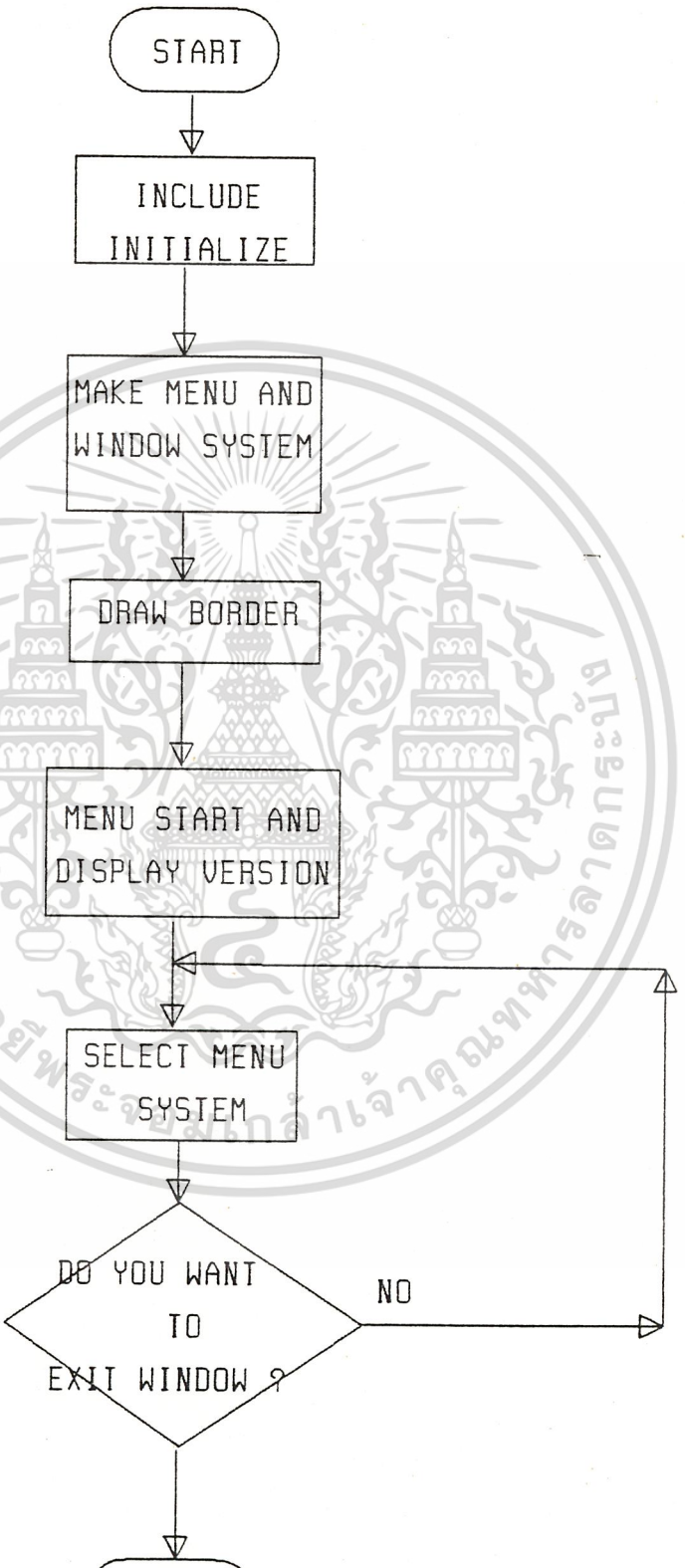
โปรแกรมหลักเรดาร์ ชิมมูเลเตอร์ จะเป็นโปรแกรมหลักที่ใช้ภาษาซีในการเขียน ซึ่งในส่วนของโปรแกรมหลักดังกล่าว สามารถแสดงได้โดยโฟล์ชาร์ต ดังนี้ คือ

1. การทำการอินนิชเชียล (Initialize) ค่าตัวแปรต่างๆที่ต้องการใช้งานไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. เข้าสู่ขั้นตอนของการสร้างระบบวินโดว์ (Windows)และเมนู จากนั้นทำการสร้างกรอบของตัวโปรแกรมที่มีหน้าต่างที่ใช้แสดงข้อความที่ให้ผู้ใช้ปฏิบัติตาม หลังจากนั้นจะเข้าสู่ขั้นตอนการเลือกหัวข้อเมนูที่ต้องการจะให้ทำงาน โดยที่มี 5 หัวข้อ คือ เมนูไฟล์ เมนูเรดาร์สโคบ เมนูรายละเอียดและเมนูออก (Quit) ซึ่งโครงสร้างการทำงานของแต่ละโปรแกรมนั้นจะได้กล่าวถึงในส่วนถัดไป และโปรแกรมก็จะทำงานอยู่ตลอดจนกว่าจะมีการเลือกหัวข้อเมนูออก แล้วเลือกการกลับสู่ดอส(Exit to dos) เมื่อต้องการเลิกการทำงาน ซึ่งจะมีคำถามย้ำเพื่อความแน่ใจอีกครั้ง ถ้าหากยังไม่ต้องการออกจากโปรแกรมก็ให้ตอบ 'ไม่' ('n')แต่ถ้าแน่ใจว่าจะเลิกการทำงานจากโปรแกรมแล้วให้ตอบ'ตกลง' ('y')หลังจากนั้นโปรแกรมก็จะทำการเลิกการจองหน่วยความจำ(Unallocate Memory) เพื่อเป็นการคืนหน่วยความจำที่ได้จองไว้คืนให้กับระบบปฏิบัติการ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

รูปที่ 7.2 แสดงโฟล์ชาร์ตการทำงานของโปรแกรมเรดาร์ชิมมูเลชั่น

7.2.2 โปรแกรมที่ใช้ในการควบคุมระบบการเลือกเมนู

หน้าที่หลักของโปรแกรมในส่วนนี้คือ จะทำการควบคุมการใช้งานในการเลือกหัวข้อ เมนูต่างๆ โดยได้รวมเอา ฟังก์ชัน(Function) ต่างๆที่เกี่ยวกับ วินโดว์(Window) เข้าไว้ด้วย ในส่วนต้นของโปรแกรมนั้นจะมีการกำหนด อาเรย์(Array)ของ ชา(char) เพื่อเก็บไว้เป็นหัวข้อ เมนู(Menu)ย่อยต่างๆซึ่งเป็นลักษณะของ บ็อกเมนู(Box Menu) ส่วนหัวข้อหลักที่เป็น เมนูหลัก(Main menu) นั้นจะกำหนดเป็น อาเรย์(Array)ชื่อ Menuo

เมนู ฟังก์ชัน (Menu Function)

- menu_start(MENU s[],int fore,int back)

ฟังก์ชัน นี้จะทำการสร้างและแสดงหัวข้อ เมนูหลัก(Main menu) ออกมาในแบบ สไลด์บาร์เมนู (slide bar menu)

- first_active(MENU s[],int fore,int back)

ฟังก์ชัน นี้จะทำการตรวจสอบคีย์ว่ามีการกดคีย์ F10 หรือไม่ ถ้ามีก็จะทำให้หัวข้อเมนูแอคทีฟ (Menu active) เป็น รีเวิร์ส วีดีโอReverse video

- menu_choice(MENU s[],int fore,int back)

ฟังก์ชัน นี้จะเป็นหัวใจสำคัญของระบบเมนูทั้งหมด กล่าวคือ จะทำหน้าที่ในการตรวจสอบการกดคีย์ที่ใช้ควบคุมการเลือกหัวข้อ เมนูต่างๆ เช่น คีย์ลูกศรซ้าย-ขวา หรือ คีย์ที่เป็นตัวอักษรตัวแรกของแต่ละหัวข้อเมนู แล้วให้ แอคทีฟ(active) ตามหัวข้อเมนู ที่เลื่อนคีย์ลูกศรไป หลังจากนั้นให้กดคีย์ Enter หรือกดคีย์ลูกศรลง (down arrow) เพื่อเป็นการเลือกหัวข้อ เมนู(menu)นั้นๆ

- select_menu()

ฟังก์ชัน นี้จะเป็นการเลือกไปทำงานตามหัวข้อเลือกต่างๆ

- border()

ฟังก์ชัน นี้จะเป็นการสร้างกรอบหน้าต่างที่หน้าจอ โดยได้แบ่งเป็น 1 ช่องหน้าต่าง ซึ่งแสดงถึงข้อความที่แจ้งให้ผู้ใช้ได้ทราบและทำตามข้อความที่แสดง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สกรีนฟังก์ชัน(Screen function)

ฟังก์ชัน ในส่วนนี้จะเป็นการใช้งานเกี่ยวกับการแสดงผลทางจอภาพต่างๆ เช่น cls(), active(), active_off(), c_scroll(), colorstr(), colorchr(), getvpage(), getvmode(), video_mode(), write_string(),write_char() , save_video(), display_menu(),draw_border(),

เคอร์เซอร์ ฟังก์ชัน(Cursor function)

ฟังก์ชัน ในส่วนนี้จะเป็นการทำงานเกี่ยวกับการจัดการกับ เคอร์เซอร์ต่างๆ เช่น goto_xy(), cursor(), hide_cursor(), getvcols(), find_cursor()

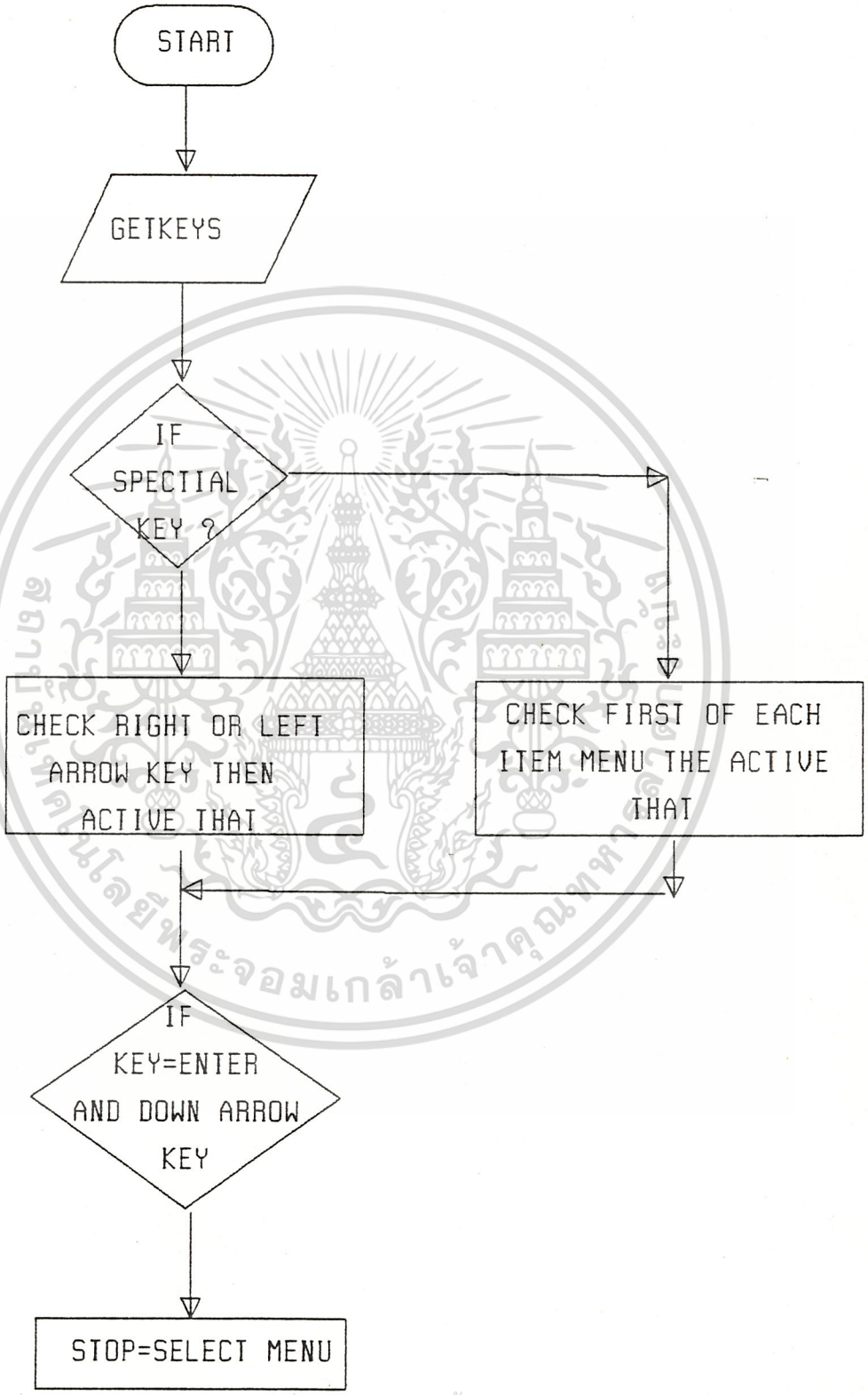
ฟังก์ชันคีย์ (key function)

เป็น ฟังก์ชันที่ทำหน้าที่เกี่ยวกับคีย์ควบคุมต่างๆ เช่น ตรวจสอบการกดคีย์ หรือตรวจสอบคีย์ที่ใช้ในการควบคุมการเลือกในแต่ละหัวข้อ เมนู ซึ่งมี ฟังก์ชัน ต่างๆดังนี้คือ get_key(), get_resp(), get_resp_sub(), get_resp_sub1()

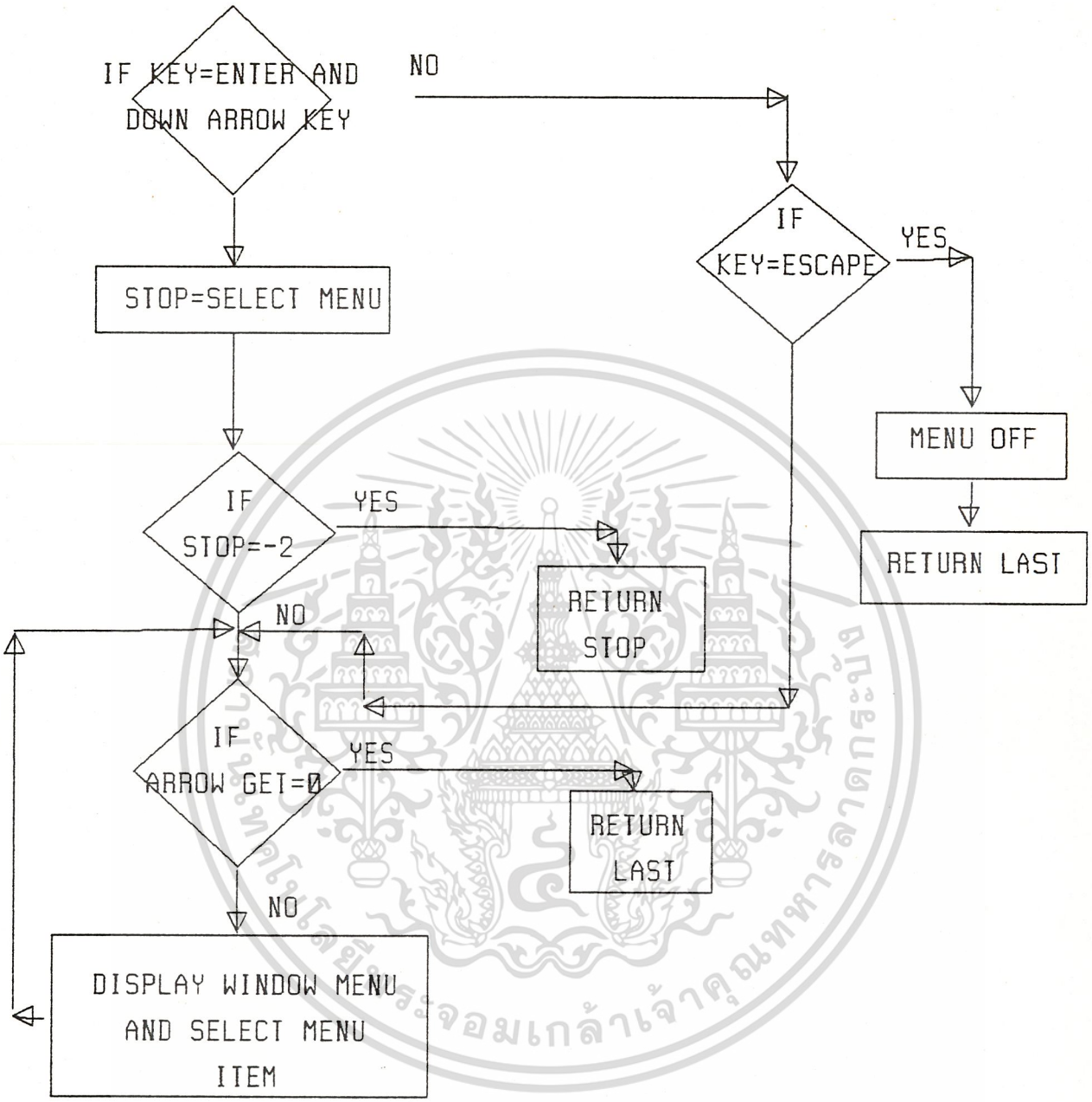
ฟังก์ชันวินโดว์ (Window function)

ฟังก์ชัน ที่อยู่ในส่วนนี้ จะเป็นฟังก์ชันที่ทำหน้าที่เกี่ยวกับการใช้งาน วินโดว์ต่างๆ เช่น การเรียกวินโดว์มาใช้งาน หรือการสร้างวินโดว์ หรืออาจจะเป็นการเขียนตัวอักษร หรือข้อความต่างๆ ภายในวินโดว์ ซึ่งมีฟังก์ชันต่างๆ ดังนี้คือ window_(), make_window(), deactive_win(), display_header(), draw_border1(), window_puts(), window_putchar(), window-xy(), window_gets(), window_getchar(), window_cls(), window_cleol(), window_upline(), window_downline(), window_bksp(), save_video_win(), restore_video_win(), get_special()

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 7.3 แสดงโฟล์ชาร์ตของฟังก์ชั่นการเลือกเมนู
(Menu Choice)

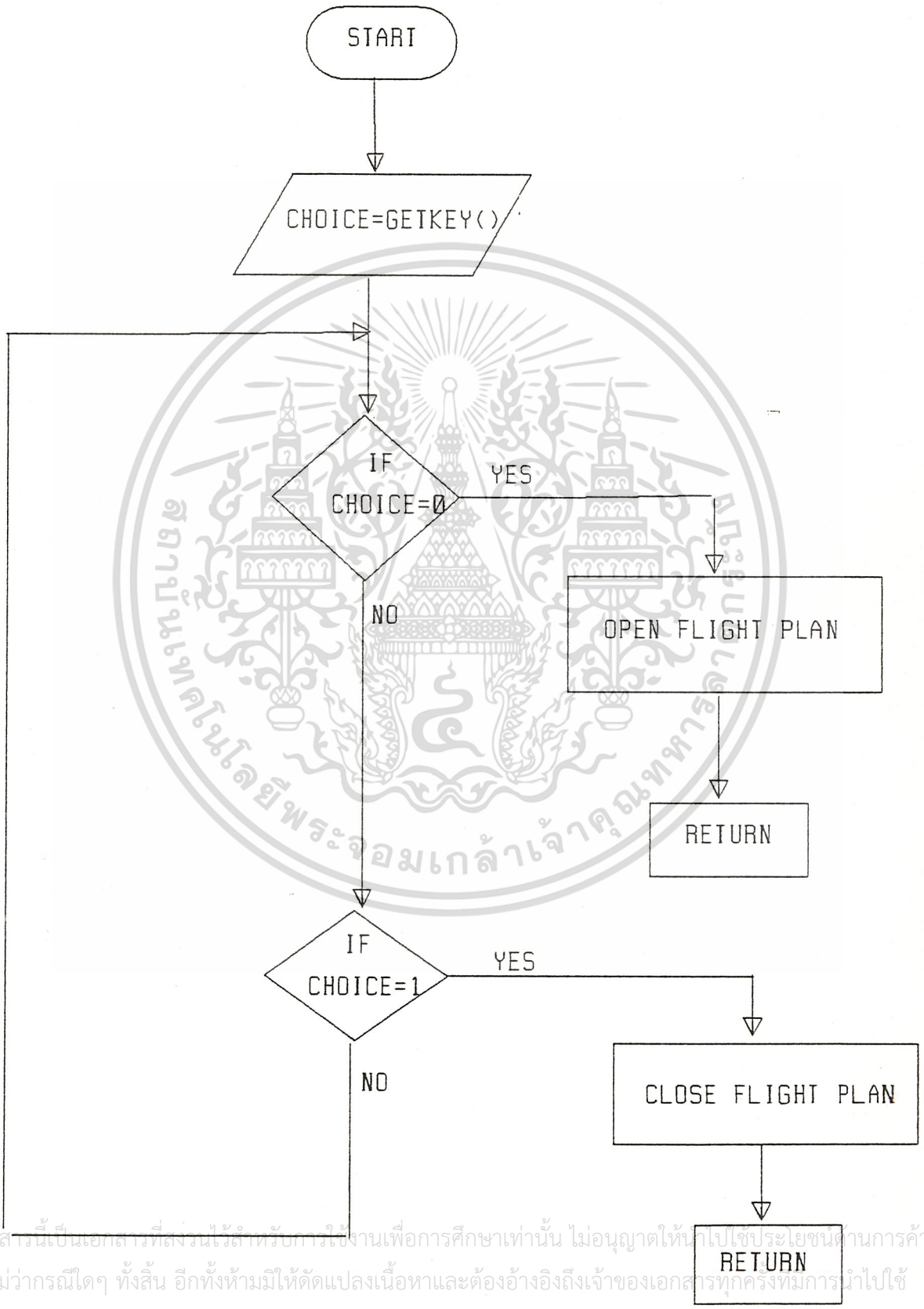
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.3 โครงสร้างเมนูรายละเอียด (Detail)

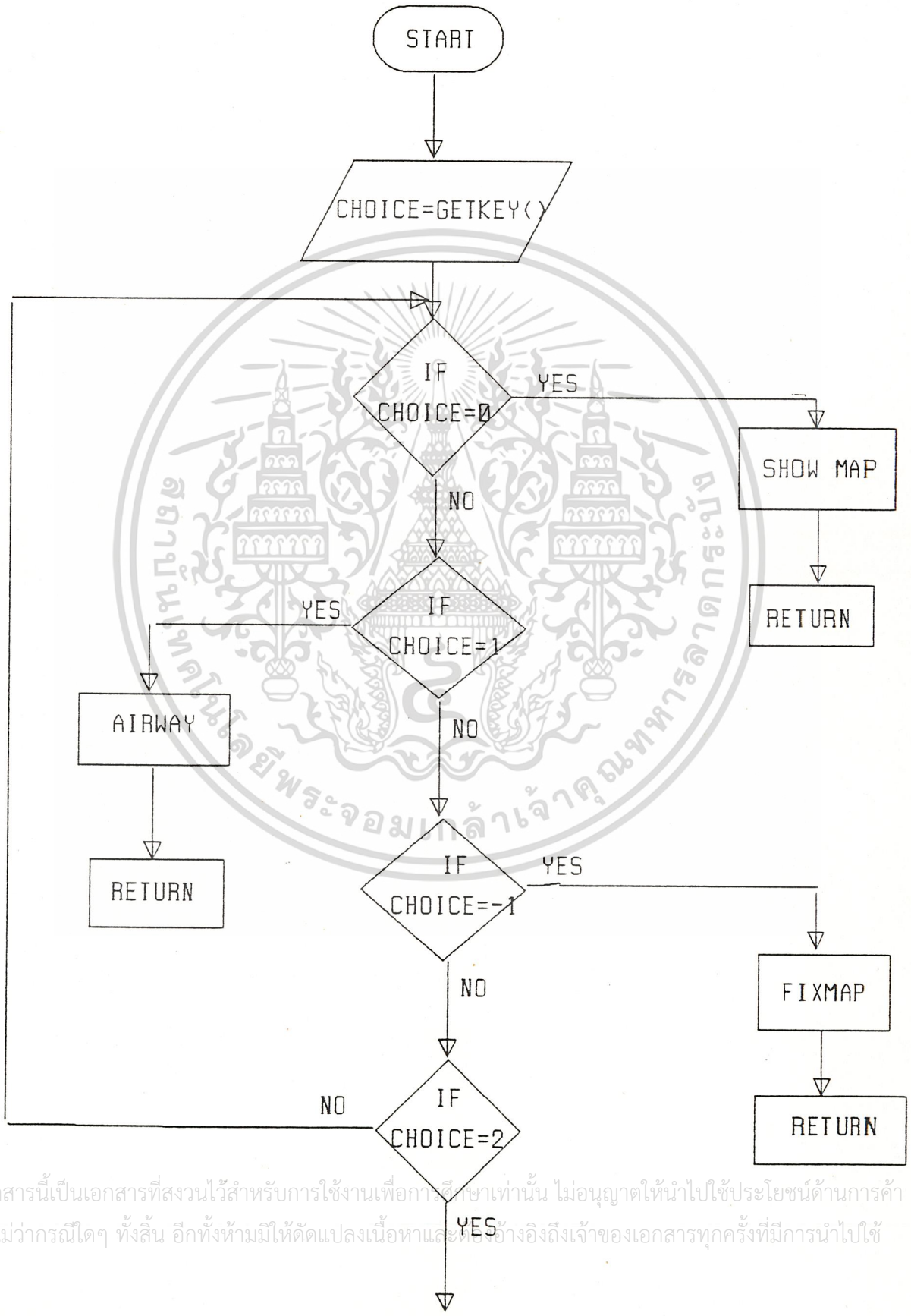
เป็นการแสดงรายละเอียดที่แสดงถึงสมรรถภาพของเครื่องบินแต่ละลำ ตัวอย่าง

| | | |
|---|---|---|
| นามเรียกขาน | : | A3341 |
| ประเภท | : | Turboprop |
| ความจุผู้โดยสาร | : | 250 ที่นั่ง |
| ความเร็ว | : | 270 Knots |
| ระดับเพดานบิน | : | 420 ฟิต |
| ความกว้างของปีก | : | 50 ฟิต |
| ความยาวของลำตัว | : | 200 ฟิต |

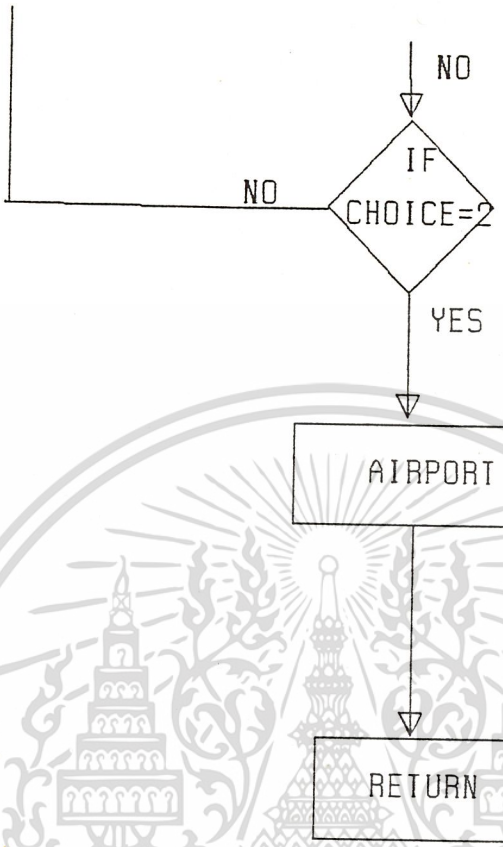
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



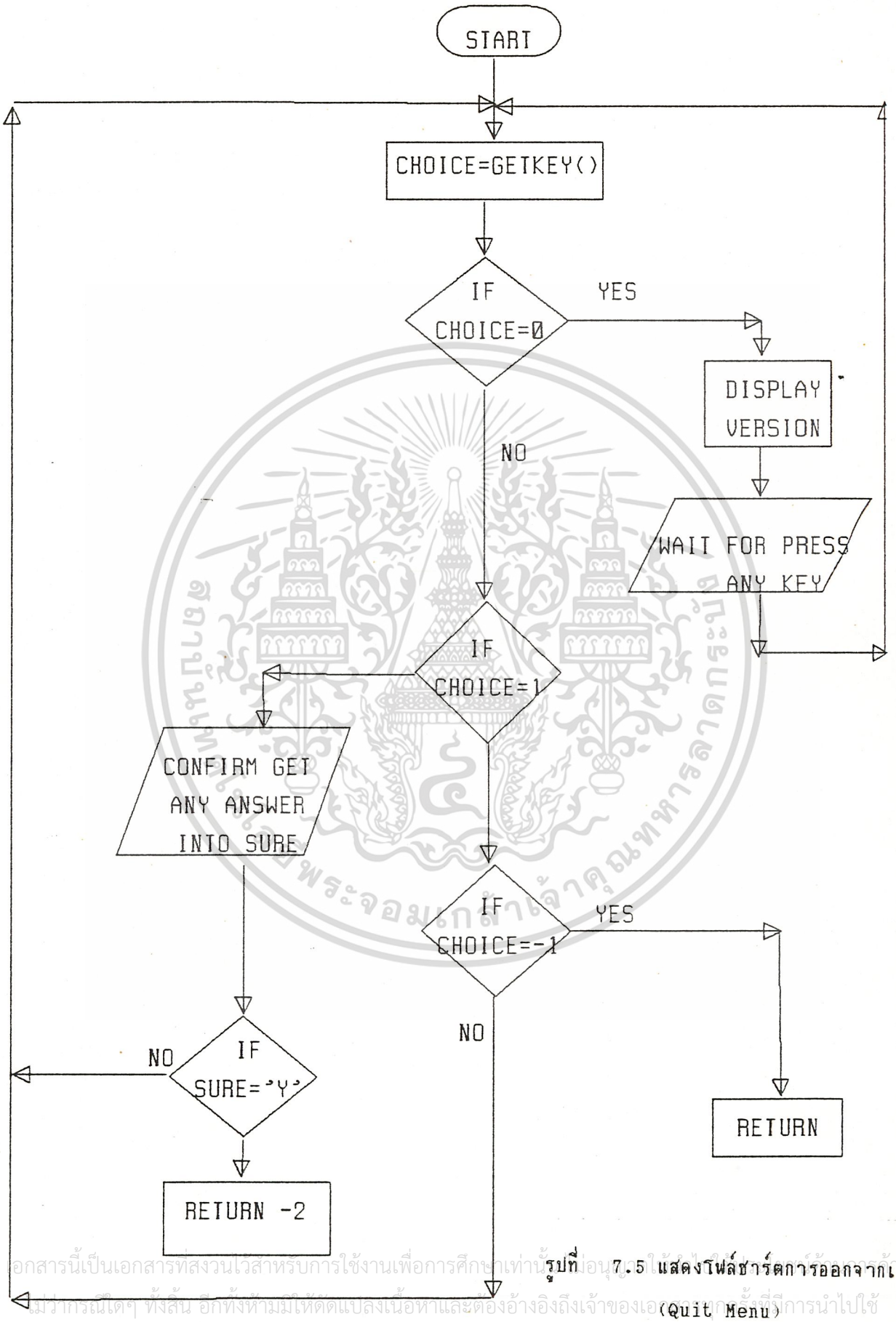

รูปที่ 7.4 แสดงโฟล์ชาร์ตการเลือกระบบเมนูย่อย

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.4 แสดงโครงสร้างการออกจากโปรแกรม (Quit Menu)

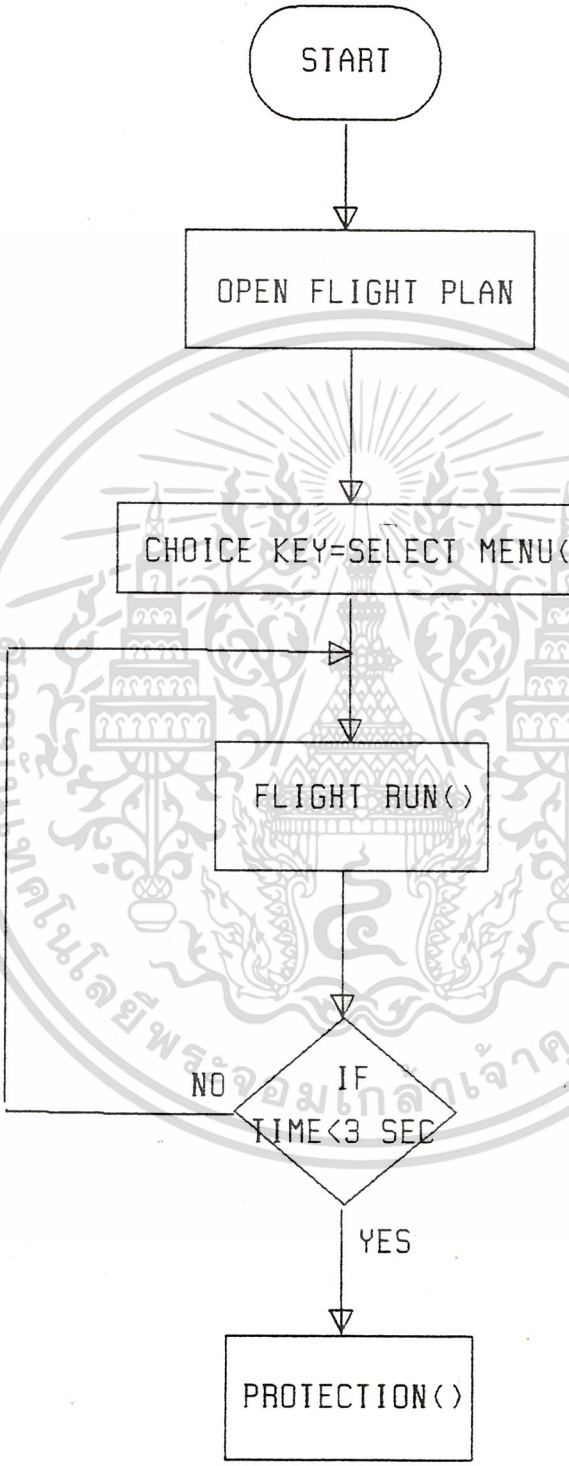
หลังจากที่ได้เรียกตัวโปรแกรมและใช้งานเสร็จแล้วเมื่อต้องการกลับไปสู่ดอส (Exit dos) กด f10 และเลื่อนแถบสว่างไปยัง เมนูควิท(Quit Menu) กดเอนเทอร์ (Enter) จากนั้นจะมีหน้าต่างขึ้นมาและถามว่า "Do you want to exit to dos Y /N ?"กด yes เมื่อต้องการกลับสู่ดอส

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



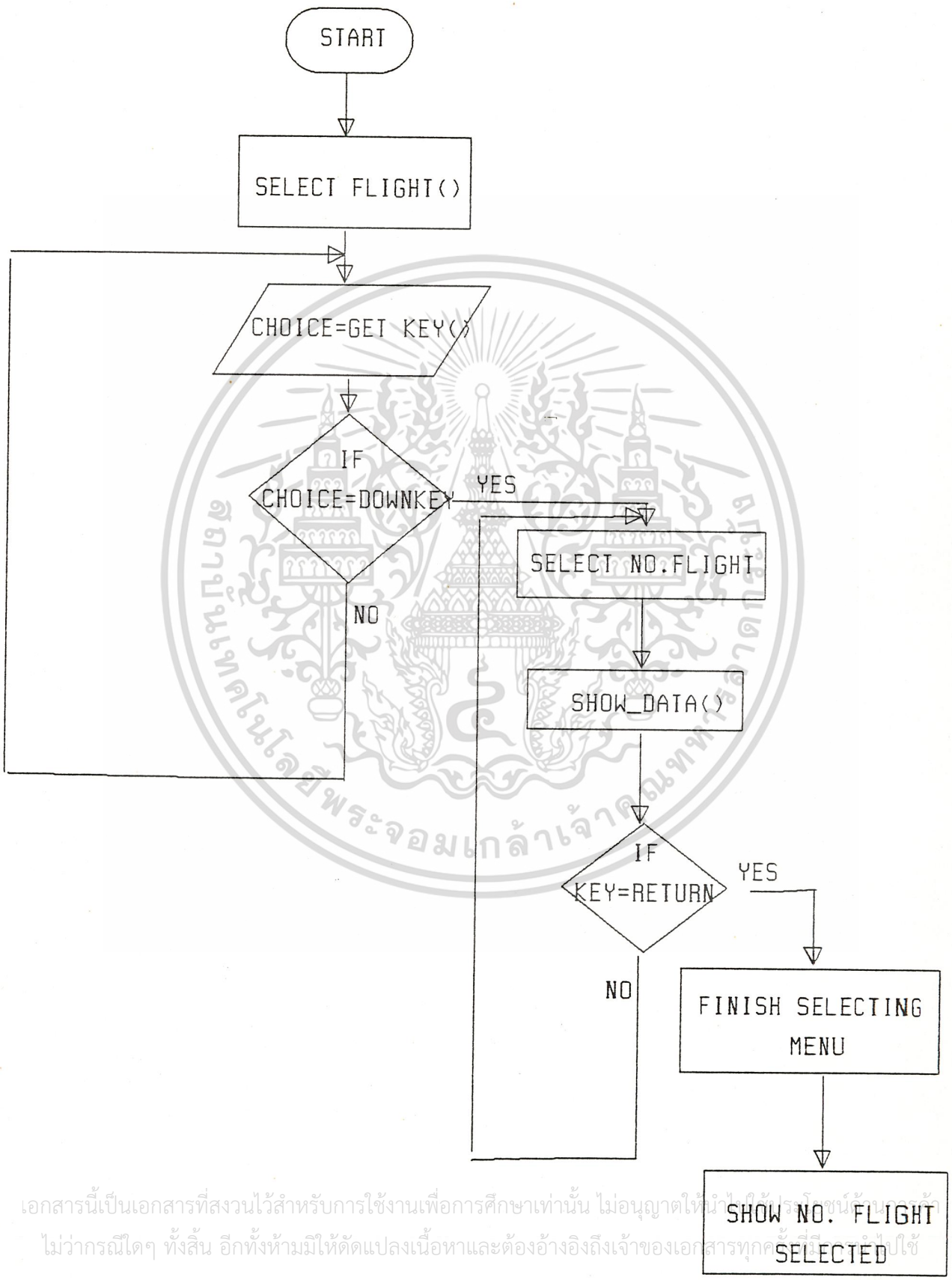

รูปที่ 7.5 แสดงโฟล์ชาร์ตการออกจากเมนู (Quit Menu)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

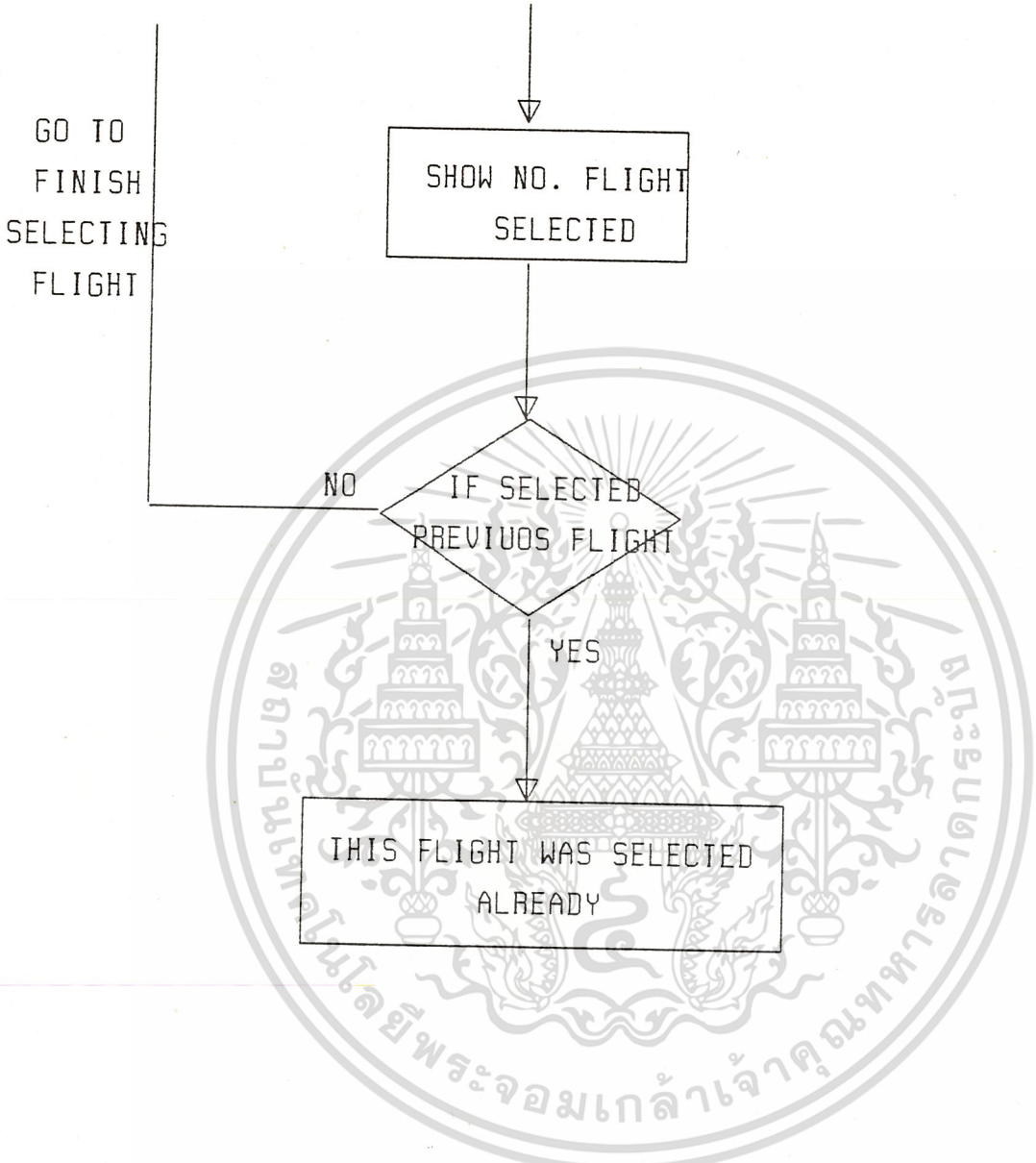



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

7.2.5 โครงสร้างโปรแกรมรันไฟท์ รัน (Flight Run)




เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้




รูปที่ 7.6แสดงโฟล์ชาร์ตไฟล์รัน (Flight run)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ 8

## องค์ประกอบทางด้านฮาร์ดแวร์และซอร์ฟแวร์

### 8.1 ข้อจำกัดทางด้านเทคนิคที่ทำหน้าที่เป็นจอแสดงผลเรดาร์ (Display)

1. หน่วยประมวลผลกลาง(Main Board)
   - ใช้หน่วยประมวลผลกลาง 80486 เอ เอ็ม ไอ (AMI)
   - หน่วยความจำ 4 เมกะบิท (mega Bit)
   - หน่วยแสดงผลแบบ วี จี เอ (VGA) เพื่อแสดงผลเท็คโหมด (Text Mode)
   - จอแสดงผล VGA ซึ่งสามารถรายละเอียดทางด้านกราฟิกส์ได้ 600*800 จำนวน 16 สี
2. หน่วยความจำภายนอก (External Storage)
   - ฟลอบปี้ดิส (Floppy Disk) ขนาด 1.44 เมกะบิท
   - ฮาร์ดดิส (Hard Disk) ขนาด 345 เมกะบิทเพื่อใช้ในการเก็บข้อมูลและตัวโปรแกรม
3. จอภาพ (Monitor)
   - จอภาพกราฟิกส์รายละเอียดสูง สามารถแสดงผลทางด้านกราฟิกส์ได้ 600*800 จุด จำนวน 16 สี
4. แป้นพิมพ์ (Keyboard)
   - ภาษาอังกฤษ,ตัวเลข,สัญลักษณ์พิเศษ,นิวเมอริก คีย์แพด (Numeric Keypad),ฟังชั่น คีย์ (Function Key) และแป้นควบคุมเคอร์เซอร์ 4 ทิศทาง รวม 101 คีย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 8.2 คุณสมบัติของโปรแกรม (Software Feature)

1. จอแสดงผลเรดาร์จำลอง (Radar Display) มีคุณสมบัติดังต่อไปนี้

1.1 คุณสมบัติเบื้องต้น (Basic Feature)

- สามารถแสดงผลเป็นแบบกราฟิกส์ (Synthetic Track) บนหน้าจอแบบความละเอียด 600*800 จุด
- สามารถแสดงเส้นขอบเขตทางภูมิศาสตร์(Linemap), เส้นทางการบิน (Airway), จุดรายงาน (Fixmap) ได้ตามแผนที่การบินที่ใช้งานได้ตามปัจจุบัน โดยข้อมูลต่างๆเหล่านี้จะเก็บเป็นแฟ้มเป็นข้อมูลจุดและเป็นตัวอักษร โดยแบ่งเป็น

1.1.1 ไลน์แมบ (Linemap)เส้นขอบเขตประเทศไทยและประเทศเพื่อนบ้านใกล้เคียง

1.1.2 จุดรายงาน (Fixmap)

1.1.3 เส้นทางการบิน (Airway)

1.1.4 รูปร่างและเส้นขอบเขตของสนามบิน (Airportmap)

1.2 จำลองการแสดงผลการเคลื่อนที่แบบอิสระ เพื่อให้การเคลื่อนที่นุ่มนวล สมจริง

1.3 สามารถระบุนามเรียกขาน (Call Sign) ให้กับแทรก(Track) ที่ต้องการได้ (Track Identify)

1.4 สามารถแสดงรายละเอียดของโหมดโค๊ด (Mode Code) และความสูงของเครื่องบินในแบบบล็อกเต็มรูปแบบ(Full Data Block) ซึ่งประกอบด้วย

แถวที่ 1

เวลาที่บินออกจากท่าอากาศยานต้นทาง : มี 4 คอลัมน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แถวที่ 2

นามเรียกขานของเครื่องบิน : มีอักษรไม่เกิน 8 ตัว

แถวที่ 3

ชื่อเครื่องบิน

แถวที่ 6

โหมด ซี ลีเวล (Mode C Level) : ตัวเลขแสดงความสูง (100 ฟิต)

แถวที่ 5

ความเร็ว : แสดงความเร็วเป็นน็อต (Knots) ด้วยตัวเลข 3 ตัว

แถวที่ 6

ทิศทาง : แสดงทิศทางของหัวเครื่องบินเทียบ 0 องศาเสมอ

แถวที่ 7

จุดเริ่มต้น : แสดงท่าอากาศยานเริ่มต้นที่เครื่องบินออกบิน

แถวที่ 8

จุดปลายทาง : แสดงปลายทางที่เครื่องมาถึง

1.4 การป้อนคำสั่งควบคุมระบบ (User Command)

- สามารถป้อนคำสั่งเพื่อเลือกแผนการบิน (Flight Plan) โดยผ่านทางคอมมานไลน์ (Command Line)ได้
- สามารถเปิดแฟ้มข้อมูลเพื่อเลือกแสดงแผนที่ (Map Selection) ที่บริเวณต่างๆได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สามารถเปลี่ยนแปลงค่าความเร็ว ทิศทางของหัวเครื่องบิน (Heading)ได้
- สามารถเลือกแผนการบินได้สูงสุดถึง 20 แผนการบิน
- การสร้างสถานการณ์ต่างๆ เช่น การออกบินของเครื่องบินในขณะที่ทัศนวิสัยไม่ดี มีฝนตกหรือมีลมแรง เพื่อทดสอบความพร้อมและการแก้ปัญหาเฉพาะหน้าของผู้รับการฝึก
- ผู้ควบคุมการทำงานสามารถป้อนคำสั่งเพื่อเปลี่ยนแปลงสถานะของเที่ยวบินได้ตลอดเวลา ด้วยการใช้กดเคอร์เซอร์ คีย์รีเทอร์น (Return Key ) และคีย์เอสเคบ (Escape) เช่น

1.4.1 การเลี้ยว

1.4.2 การเปลี่ยนความเร็วของเครื่องบิน

1.4.3 การเปลี่ยนระดับความสูง

1.4.4 การยกเลิกเที่ยวบิน

8.3 การจัดเตรียมข้อมูล

ในส่วนของการเขียนโปรแกรม ได้ใช้ข้อมูลที่ใช้งานในความเป็นจริงที่ใช้ในสายการบินปัจจุบัน ได้ทำการจัดเก็บเป็นแฟ้มข้อมูลไว้ ซึ่งประกอบด้วยแฟ้มข้อมูลเกี่ยวกับตำแหน่งทางภูมิศาสตร์ เช่น จุดรายงาน เส้นขอบเขต เส้นทางการบิน ท่าอากาศยาน และแฟ้มข้อมูลแสดงสมรรถนะของเครื่องบิน เช่น นามเรียกขาน ชนิดของเครื่องบิน ความจุจำนวนผู้โดยสาร ความเร็ว ระดับเพดานบิน อัตราการไต่ที่ระดับความสูง น้ำหนักของตัวเครื่องบิน ความยาวและความกว้างของเครื่องบิน

1. แฟ้มจุดแสดงแผนที่
2. แฟ้มจุดเส้นทางการบิน
3. แฟ้มข้อมูลจุดรายงาน
4. แฟ้มข้อมูลท่าอากาศยาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# บทที่ ๑

# การเขียนโปรแกรมการควบคุมจราจรทางอากาศแบบควบคุมพื้นที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
#include <time.h>
#include <stdio.h>
#include <dos.h>
#include <stdlib.h>
#include <graphics.h>
#include <conio.h>
#include <math.h>
#include <graphics.h>
#include <bios.h>
#include <alloc.h>
#include <string.h>
#include "p:\window\menu.h"
#include "p:\window\display.h"
#include "p:\pro\show.c"

#define  NO_ALL_FLIGHT 20
#define  NUM_AIRPOINT  40
#define  MES_LINE1 "     RADAR  SIMULATION       "
#define  MES_LINE2 "       VERSION  1.00         "
#define  MES_LINE3 "           BY                "
#define  MES_LINE4 "MISS  BUPPHA   VUTTIPRASITH  "
#define  MES_LINE5 "         ADVISOR             "
#define  MES_LINE6 " MR. SOMYOT JUNNAPIYA        "
#define  MES_LINE7 "<<  F10 TO MAIN MENU  >>"
#define  UP_ARROW 0x4800
#define  DW_ARROW 0x5000
#define  RETURN   0x1C0D
#define  ESC      0x011B
#define  BACKSPACE_KEY 8

int   kbhit(void);
void  cls(void);

void  map_(void);
void  fix_map(void);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void  airway(void);
void  airport(void);
void  aircraft(void);
void  atc_(void);
int   menu_quit(void);
void  menu_file(void);
void  menu_rad_scp(void);
void  menu_detail(void);
void  open_f(void);
void  close_f(void);
void  disp_version(void);
void   window_cls(int);
/* void   menu_quit(int); */
void  close_graphics_mode(void);
void  airway();
void  fix_map();
void  airport();
void  aircraft();
void  atc_();
void  read_airpoint(void);
void  show_map(void);
void  show_airway(void);
void  show_fixmap(void);
void  show_city(void);
void  show_compulsory_reporting(void);
void  show_cross_animation(int y);
/* void  clearviewport(); */
void  show_aircraft_1(void);
void  show_aircraft_2(void);
void  show_aircraft_3(void);
void  show_aircraft_4(void);
void  show_aircraft_5(void);

void  set_flight(void);
void  show_data(int flight,int y);
void  flight_run(void);
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int   each_flight_run(int flight_order,long now_time);
long  time_2_long(int t);
void  check_status(int flight);
float r(float a,float b);
int   current_position(int Flight,long run_time,float *px,float *py);
int   find_airpoint(char *name);
void  show_time(int hr,int min);
void  display_flight_data(void);
void  warn(char message[50]);
int   midx(void);
int   midy(void);

/********************** FUNCTION PROTOTYPE ******************************/

void  menu_start(MENU s[],int fore, int back);
void  first_active(MENU s[],int fore, int back);
void  menu_on(MENU m[],int fore,int back);
void  menu_off(MENU m[],int fore,int back);
void  menu_end(void);
int   menu_choice(MENU m[],int fore,int back);
int   select_menu(void);

void  write_head(int x,int y,char *h,int f,int attr);
void  clsmess(void);
void  cls();
void  active_off(char *m,int row,int col,int fore, int back);
void  c_scroll(int row,int col,int wide,int deep,int num,int f,int color);
void  colorstr(char *s,int fcolor, int bcolor);
void  colorchr(int c,int fcolor, int bcolor);
void  scroll(int row,int col,int wide,int deep,int num, int function);

int   getvpage(void);
int   getvmode(void);
int   video_mode(void);
void  write_string(int x,int y,char *p,int attrib);
void  write_char(int x,int y,char ch,int attrib);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void  save_video(int num);
void  restore_video(int num);
int   pulldown(int num);
int   make_menu(int num,char *menu[],char *keys,int count,int x,int y,int border);
void  display_menu(int num);
void  draw_border(int num);
void  goto_xy(int x,int y);
void  cursor(int row,int col);
void  hide_cursor(void);
int   getvcols(void);
void  find_cursor(int *row,int *col);
int   get_key(void);
int   get_resp(int num);
int   is_in(char *s,char c);
void  window_(int num);
int   make_window(int num,char *header,int startx,int starty,int endx,int endy,int border);
void  deactivate_win(int num);
void  display_header(int num);
void  draw_border1(int num);
int   window_puts(int num,char *str,int attr);
int   window_putchar(int num,char ch,int at);
int   window_xy(int num,int x,int y);
void  window_gets(int num,char *s);
int   window_getche(int num);
void  save_video_win(int num);
void  restore_video_win(int num);
int   get_special(void);
void  read_point(void);
float get_float(FILE *fp);


struct airpoint_struct
        { char air_name[10];
          float py;
          float px;
        };
struct flight_struct {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                int  time;
                char call_sign[10];
                char flight[10];
                int  level;
                int  speed;
                int  heading;
                char from[10];
                char airway[10];
                char destination[10];
                long  cur_t;
                float cur_x;
                float cur_y;
        };
struct flight_struct flight[NO_ALL_FLIGHT];
struct airpoint_struct airpoint[NUM_AIRPOINT];
int   status[NO_ALL_FLIGHT];
FILE  *fp;
int   num_line=0;
char  *cptr;
int   last,menucount,menulen,*nptr,pcol,prow;
int   arrow_get,arrow_item,arrow_item1;
int   key_geted,key_recive,key_recive1;
int   arrow_choice,len_deso;
char  far *vid_mem;
int   flight_select[NO_ALL_FLIGHT];
int   Num_flight;                                   /* number of flight that
select to run */
float speed_fact=0.014455;                          /* 1 Degree=110 Km,1
Degree=69.182 Nautical miles */

        /* 1 Knot = 1 Naultical mile/hr = 0.014455 Degree/hr */
char  *file[]={
                " Open flight plan  ",
                " Close flight plan ",
                };
char *rad_scp[]={
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    " Map      ",
                    " Fixmap   ",
                    " Airway   ",
                    " Airport  ",
                    };
char *detail[]={
                    " Aircraft     ",
                    " ATC          ",
                    };
char *quit[]={
                    " Version    ",
                    " Exit to dos ",
                    };
 MENU   menu0[] = {
        OPTION(" File "," Open flight plan and Close flight plan",'f')
        OPTION(" Radarscope "," Map Fixmap Airway and Airport ",'r')
        OPTION(" Detail "," Aircraft and ATC",'d')
        OPTION(" Quit ","display Version or Exit to dos",'q')
        OPTIONEND};
 /* start of menu file */
void  menu_file()
{
        int choice;
        arrow_choice=0;
        key_geted=0;
        pulldown(0);
        while((choice=get_resp(0)) != -1) {
                switch(choice) {
                        case 0:
                                open_f();
                                break;
                        case 1:
                                close_f();
                                break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        restore_video(0);
}
 /* start of menu radarscope */
void  menu_rad_scp()
{
        int choice;
        arrow_choice=0;key_geted=0;
        pulldown(2);
        while((choice=get_resp(2)) != -1) {
                switch(choice) {
                        case 0:
                                map_();
                                break;
                        case 1:
                                fix_map();
                                break;
                        case 2:
                                airway();
                                break;
                        case 3:
                                airport();
                                break;
                        }
                }
        restore_video(2);
}
 /* start of detail menu */
void  menu_detail()
{
        int choice;
        arrow_choice=0;
        key_geted=0;
        pulldown(1);
        while((choice=get_resp(1)) != -1) {
                switch(choice) {
                        case 0:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                display_flight_data();
                                break;
                        case 1:
                                atc_();
                                break;
                                }
                }
        restore_video(1);
}
 /* start of map menu */
void map_()
{   int driver =DETECT ,mode;

        initgraph(&driver,&mode,"c:\\lang\\bc\\bgi");
        setcolor(15);
        setbkcolor(1);
        show_map();
        set_flight();
        flight_run();
        closegraph();

        make_menu(6, rad_scp,"mafp",4,2,17,BORDER);
        cls();
        cursor(2,5);
        menu_start(menu0,0,7);
        first_active(menu0,0,7);
        menu_choice(menu0,0,7);
}
 /* start of airway menu */
void  airway()
{
    int driver =DETECT ,mode;
    initgraph(&driver,&mode,"c:\\lang\\bc\\bgi");
    setcolor(15);
    setbkcolor(1);
    show_airway();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void  show_aircraft_4(void);

void  show_aircraft_5(void);
    getch();
    closegraph();
          make_menu(6, rad_scp,"mafp",4,2,17,BORDER);
          cls();
          cursor(2,5);
          menu_start(menu0,0,7);
          first_active(menu0,0,7);
          menu_choice(menu0,0,7);
}
 /*  start of fixmap menu */
void  fix_map()
{
    int driver =DETECT ,mode;
    initgraph(&driver,&mode,"c:\\lang\\bc\\bgi");
    setcolor(15);
    setbkcolor(1);
    show_compulsory_reporting();
    getch();
    closegraph();
          make_menu(6, rad_scp,"mafp",4,2,17,BORDER);
          cls();
          cursor(2,5);
          menu_start(menu0,0,7);
          first_active(menu0,0,7);
          menu_choice(menu0,0,7);
}
 /* start of airport menu */
void airport()
{
   int driver =DETECT ,mode;
   initgraph(&driver,&mode,"c:\\lang\\bc\\bgi");
   setcolor(15);
   setbkcolor(1);
   show_city();
   getch();
   closegraph();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void  scroll(int row,int col,int wide,int deep,int num, int function);

        make_menu(1, detail,"at",2,2,31,BORDER);
int   getvpage(void);
        cls();
        cursor(2,5);
        menu_start(menu0,0,7);
        first_active(menu0,0,7);
        menu_choice(menu0,0,7);
}
void  atc_()
{
}
struct  menu_frame {
        int startx, endx, starty,endy;
        unsigned char *p;
        char **menu;
        char *keys;
        int border ,count;
        int active;
        }frame[MAX_FRAME],i;
struct  window_frame {
        int startx, endx, starty,endy;
        int curx,cury;
        unsigned char *p;
        char *header;
        int border;
        int active;
        }frame_win[MAX_FRAME];

        /*************************************
        *          MENU   FUNCTION          *
        *************************************/

void  menu_start(MENU s[],int fore, int back)
{
        int  i,r,c,wide;
        wide = getvcols()-1;
        find_cursor(&prow,&pcol);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
float get_float(FILE *fp);


        c_scroll(prow,pcol-4,wide,1,0,6,back << 4);
struct airpoint_struct
        last = 0;
        nptr = (int *) calloc(wide,sizeof(int));
        cptr = (char *) calloc(wide,sizeof(char));
        for(menulen = i = 0; s[i].prompt;i++) {
           find_cursor(&r,&c);
           nptr[i] = c;
           cptr[i] = s[i].letter;
           colorstr(s[i].prompt,fore,back << 4);
        if (i != 4)
                colorstr(MARGIN,fore,back << 4);
        menulen += s[i].plen;
        }
        menucount = 1;
        menulen += (strlen(MARGIN) * (menucount-1));
}
void  first_active(MENU s[],int fore, int back)
{
        union scan {
        int c;
        char ch[2];
}sc;
        do {
                hide_cursor();
                sc.c = get_key();
        }while(sc.ch[1] != F10);
        switch(last) {
                case 0:
                        pcol = 5;
                        break;
                case 1:
                        pcol =17;
                        break;
                case 2:
                        pcol = 35;
                        break;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
/* 1 Knot = 1 Naultical mile/hr = 0.014455 Degree/hr */

char   *file[]={
                        case 3:
"  Open flight plan  ",          pcol =49;
                                 break;
                        default:
                                 break;
                        }
  cursor(prow,pcol);
  colorstr(s[last].prompt,back,A_INTENSE);
  len_deso = (80-strlen(s[last].deso)) / 2;
  cursor(prow + 23,len_deso);
  colorstr(s[last].deso,fore,A_NORM);
  }
  void menu_on(MENU m[],int fore,int back)
  {
          active(m[last].prompt,prow,nptr[last],back,A_INTENSE);
          active(m[last].deso,prow +23,len_deso,fore,back);
  }
  void menu_off(MENU m[],int fore,int back)
  {
          active_off(m[last].prompt,prow,nptr[last],fore,back);
          c_scroll(prow +23,len_deso,m[last].dlen,1,1,6,A_INTENSE);
  }
  void menu_end(void)
  {
          free(nptr);
          free(cptr);
  }
  int menu_choice(MENU m[],int fore,int back)
  {
          int stop = 0;
          union scan {
                  int c;
                  char ch[2];
                  }sc;
          while(TRUE) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    arrow_get=0;
    hide_cursor();
    sc.c = get_key();
if(sc.ch[0] == 0) {
        switch (sc.ch[1]) {
            case RARROW:
                    menu_off(m,fore,back);
                    if(last +1 >= menucount)
                    {
                    if(last == 3)
                    last = 0;
                    else
                    last++;
                    }
                    else
                    last = 0;
                    menu_on(m,fore,back);
                    break;
                case LARROW:
                    menu_off(m,fore,back);
                    if(last - 1 < 0)
                            last = menucount + 2;
                    else
                            last--;
                    menu_on(m,fore,back);
                    break;
                default:
                    break;
        }
}else
switch(tolower(sc.ch[0])){
        case 'f':
                menu_off(m,fore,back);
                last=0;
                menu_on(m,fore,back);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                break;
        case 'r':
                menu_off(m,fore,back);
                last=1;
                menu_on(m,fore,back);
                break;
        case 'd':
                menu_off(m,fore,back);
                last=2;
                menu_on(m,fore,back);
                break;
        case 'q':
                menu_off(m,fore,back);
                last=3;
                menu_on(m,fore,back);
                break;
        default:
                break;
        }
        if (sc.ch[0] == ENTER || sc.ch[1] == DARROW)
        {
                stop = select_menu();
                if(stop == -2) return stop;
        }
        else if(sc.ch[0] == ESCAPE){
                menu_off(m,fore,back);
                clsmess();
                hide_cursor();
                break;
        }
do{
        switch(arrow_get) {
                case LARROW:
                        menu_off(m,fore,back);
                        if(last - 1 < 0)
                                last = 3;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    else
                        last--;
                    menu_on(m,fore,back);
                    stop = select_menu();
                    if(stop == -2)return stop;
                    break;
                case RARROW:
                    menu_off(m,fore,back);
                    if(last + 1 >= menucount)
                    {
                    if(last == 3)
                        last = 0;
                    else
                        last++;
                    }
                    else
                        last = 0;
                    menu_on(m,fore,back);
                    stop = select_menu();
                    if(stop == -2)return stop;
                    break;
                case 10:
                    menu_off(m,fore,back);
                    return last;
                }
            }while(arrow_get != 0);
        }
        return last;
}
int select_menu()
{
        int sure = 0;
            switch(last) {
                case 0:
                    menu_file();
                    clsmess();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                    break;
                case 1:
                    menu_rad_scp();
                    clsmess();
                    break;
                case 2:
                    menu_detail();
                    clsmess();
                    break;
                case 3:
                    sure = menu_quit();
                    clsmess();
                    return sure;
                }
        return 0;
}
void write_head(x,y,h,f,attr)
int x,y,f,attr;
char *h;
  {
    int i;
    for(i=0;h[i];i++,x++) {
        cursor(y,x);
        colorchr(h[i],f,attr);
    }
  }
void clsmess()
  {
    cursor(23,2);
    printf("                ");
  }
void cls()
  {
    union REGS r;
    r.h.ah=6;
    r.h.al=0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    r.h.ch=0;
    r.h.cl=0;
    r.h.dh=24;
    r.h.dl=79;
    r.h.bh=7;
    int86(0x10,&r,&r);
  }
void active_off(char *m,int row,int col,int fore, int back)
  {
    cursor(row,col);
    colorstr(m,fore,back << 4);
  }
void active(char *m,int row,int col,int fore, int back)
  {
    cursor(row,col);
    colorstr(m,fore,back);
  }
void c_scroll(int row,int col,int wide,int deep,int num,int f,int color)
  {
    union REGS ireg;
    row--;
    col--;
    ireg.h.ah=f;
    ireg.h.al=num;
    ireg.h.ch=row;
    ireg.h.cl=col;
    ireg.h.dh=row+deep - 1;
    ireg.h.dl=col+wide;
    ireg.h.bh=color;
    int86(0x10,&ireg,&ireg);
  }
void colorstr(char *s,int fcolor, int bcolor)
  {
    int col,row,wide;
    wide = getvcols();
    find_cursor(&row,&col);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
while(*s) {
  switch(*s) {
        case '\n':
            s++;
            row++;
            col = 1;
            break;
        case '\b':
            s++;
            col-- ;
            break;
        case '\r':
            s++;
            col =1;
            break;
        case '\t':
            s++;
            col += TAB;
            break;
        case '\f':
            s++;
            break;
        case '\a':
            col++;
            putchar(BELL);
            continue;
        case '\\':
            col++;
            break;
        default:
             col++;
             break;
}
cursor(row,col);
colorchr(*s++,fcolor,bcolor);
if(col>wide){
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            row++;
            col=1;
      }
      }
      cursor(row,col);
   }
void colorchr(int c,int fcolor, int bcolor)
   {
      union REGS ireg;
      ireg.h.ah = 0x09;
      ireg.x.cx = 1;
      ireg.h.bh = getvpage();
      ireg.h.al = (char) c;
      ireg.h.bl = (char) (fcolor | bcolor);
      int86(0x10,&ireg,&ireg);
   }
void scroll(int row,int col,int wide,int deep,int num, int function)
   {
      union REGS ireg;
      ireg.h.ah = function;
      ireg.h.al = num;
      ireg.h.ch = row - 1;
      ireg.h.cl = col - 1;
      ireg.h.dh = row + deep;
      ireg.h.dl = col + wide;
      ireg.h.bh = 0;
      int86(0x10,&ireg,&ireg);
   }
int getvpage(void)
   {
      union REGS ireg;
      ireg.h.ah= 0x0f;
      int86(0x10,&ireg,&ireg);
      return(ireg.h.bh);
   }
int getvmode(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
    union REGS ireg;
    ireg.h.ah = 0x0f;
    int86(0x10,&ireg,&ireg);
    return(ireg.h.al);
}
int video_mode()
{
    union REGS r;
    r.h.ah = 15;
    return int86(0x10,&r,&r) & 255;
}
void write_string(x,y,p,attrib)
int x,y;
char *p;
int attrib;
{
    register int i;
    char far *v;
    v=vid_mem;
    v +=(x*160) + y*2;
    for(i=y;*p;i++) {
            *v++ = *p++;
            *v++ = attrib;
    }
}
void write_char(x,y,ch,attrib)
int x,y;
char ch;
int attrib;
{
    char far *v;
    v=vid_mem;
    v+=(x*160)+y*2;
    *v++=ch;
    *v=attrib;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  }
void save_video(num)
int num;
  {
    register int i,j;
          unsigned char *buf_ptr;
    char far *v,far *t;
    buf_ptr = frame[num].p;
    v=vid_mem;
    for(i=frame[num].starty;i<frame[num].endy;i++)
     for(j=frame[num].startx;j<frame[num].endx+1;j++) {
          t=v+(j*160)+i*2;
          *buf_ptr++ = *t++;
          *buf_ptr++ = *t;
          *(t-1) =' ';
    }
  }
void restore_video(num)
int num;
  {
    register int i,j;
    char far *v,far *t;
          unsigned char *buf_ptr;
    buf_ptr = frame[num].p;
    v=vid_mem;
    t=v;
    for(i=frame[num].starty;i<frame[num].endy;i++)
      for(j=frame[num].startx;j<frame[num].endx+1;j++) {
      v=t;
      v+=(j*160) + i*2;
      *v++ =*buf_ptr++;
      *v = *buf_ptr++;
      }
      frame[num].active = 0;
  }
int pulldown(num)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int num;
  {
    int vmode;
    vmode = video_mode();
    if((vmode!=2)&& (vmode!=3)&&(vmode!=7))
      { printf("video most be in 80 column text mode ");
            exit(1);
      }
            if(vmode==7) vid_mem =(char far *)0xB0000000l;
    else vid_mem =(char far *)0xB8000000l;
    if(!frame[num].active)
      { save_video(num);
            frame[num].active = 1;
      }
   if(frame[num].border) draw_border(num);
     display_menu(num);
  return 0;
  }
int make_menu(num,menu,keys,count,x,y,border)
int num;
char *menu[];
char *keys;
int count;
int x,y;
int border;
  {
    register int i,len;
    int endx,endy;
    unsigned char *p;
    if(num>MAX_FRAME)
      { printf("Too many menus \n");
            return 0 ;
      }
    if((x>24) || (x<0) || (y>79) || (y<0))
      { printf("range error");
            return 0 ;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    }
  len = 0;
  for(i=0;i<count;i++)
    if(strlen(menu[i])>len)
        len = strlen(menu[i]);
    endy = len + 2 + y;
    endx = count + 1 + x;
    if((endx+1>24)||(endy+1>79))
        { printf("menu won't fit");
          return 0;
        }
    p = (unsigned char *) malloc(2 * (endx-x+1) * (endy-y+1));
    if(!p)
                exit(1);
    frame[num].startx = x; frame[num].endx = endx;
    frame[num].starty = y; frame[num].endy = endy;
    frame[num].p=p;
    frame[num].menu =(char**)menu;
    frame[num].border = border;
    frame[num].keys = keys;
    frame[num].count = count;
    frame[num].active = 0;
    return 1;
}
void display_menu(num)
int num;
{
  int x,y;
  register int i,x_;
  char **m;
  x_ = frame[num].startx+1;
  m  = frame[num].menu;
  for(i=0;i<frame[num].count;i++,x_++)
  write_string(x_,frame[num].starty+1,m[i],A_INVERSE);
  x=frame[num].startx+1;
  y=frame[num].starty+1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    goto_xy(x,y);
    write_string(x,y,frame[num].menu[0],A_INTENSE);
    hide_cursor();
  }
void draw_border(num)
int num;
  {
    register int i;
    char far *v,far *t;
    v = vid_mem;
    t = v;
    for(i=frame[num].startx+1;i<frame[num].endx;i++)
      { v +=(i*160) + frame[num].starty*2;
            *v++ = C_VD;
            *v = A_INVERSE;
            v = t;
            v +=(i*160) + (frame[num].endy-1)*2;
            *v++ = C_VD;
            *v = A_INVERSE;
            v = t;
    }
    for(i=frame[num].starty+1; i<frame[num].endy;i++)
      { v += (frame[num].startx*160) + i*2;
            *v++ = C_HD;
            *v  = A_INVERSE;
            v = t;
            v += (frame[num].endx*160) +i*2;
            *v++ = C_HD;
            *v = A_INVERSE;
            v = t;
      }
    write_char(frame[num].startx,frame[num].starty,C_ULD,A_INVERSE);
    write_char(frame[num].startx,frame[num].endy-1,C_URD,A_INVERSE);
    write_char(frame[num].endx,frame[num].starty,C_LLD,A_INVERSE);
    write_char(frame[num].endx,frame[num].endy-1,C_LRD,A_INVERSE);
  }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void goto_xy(x,y)
int x,y;
  {
     union REGS r;
     r.h.ah=2;
     r.h.dl=y;
     r.h.dh=x;
     r.h.bh=0;
      int86(0x10,&r,&r);
  }
void cursor(int row,int col)
  {
     union REGS ireg;
     ireg.h.ah= 0x02;
     ireg.h.bh= getvpage();
     ireg.h.dh= row-1;
     ireg.h.dl= col-1;
     int86(0x10,&ireg,&ireg);
  }
void hide_cursor(void)
  {
     cursor(26,1);
  }
int getvcols(void)
  {
     union REGS ireg;
     ireg.h.ah = 0x0f;
     int86(0x10,&ireg,&ireg);
     if(ireg.h.al < 2)
          return 40;
     if(ireg.h.al > 3 && ireg.h.al != 7)
          return 0;
     else
          return 80;
  }
void find_cursor(int *row,int *col)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
    union REGS ireg;
    ireg.h.ah = 0x03;
    ireg.h.bh = getvpage();
    int86(0x10,&ireg,&ireg);
    *row = (int) ireg.h.dh + 1;
    *col = (int) ireg.h.dl + 1;
}
int get_key()
{
    union REGS r;
    r.h.ah=0;
    return int86(0x16,&r,&r);
}
int get_resp(num)
int num;
{
    union inkey {
            char ch[2];
            int i;
            }c;
    int key_choice;
    int x,y;
    arrow_get = 0;
    x = frame[num].startx+1;
    y = frame[num].starty+1;
    do{ c.i =get_key();
          if(c.ch[0])
            { if(c.ch[0] == ESCAPE) return -1;
              if(c.ch[0] == ' ') break;
              if(c.ch[0] == ENTER)
                { key_geted = arrow_choice;
                    return arrow_choice;
                }
              key_choice = is_in(frame[num].keys,tolower(c.ch[0]));
              if(key_choice==0)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    return -10;
  goto_xy(x+key_geted,y);
  write_string(x+key_geted,y,frame[num].menu[key_geted],A_INVERSE);
  key_geted = key_choice-1;
  if(key_geted)
    { if(key_geted==frame[num].count) key_geted = 0;
        if(key_geted<0) key_geted=frame[num].count-1;
        goto_xy(x+key_geted,y);
        write_string(x+key_geted,y,frame[num].menu[key_geted],A_INTENSE);
        hide_cursor();
    } else
          { goto_xy(x+key_geted,y);
            write_string(x+key_geted,y,frame[num].menu[key_geted],A_INTENSE);
            hide_cursor();
          }
  arrow_choice=key_geted;
  return key_geted;
} else
    { goto_xy(x+arrow_choice,y);
        write_string(x+arrow_choice,y,frame[num].menu[arrow_choice],A_INVERSE);
        switch(c.ch[1])
          { case UARROW:
              arrow_choice--;
              break;
            case DARROW:
                  arrow_choice++;
                  break;
            case LARROW:
                  arrow_get = LARROW;
                  return -1;
            case RARROW:
                  arrow_get = RARROW;
                  return -1;
          }
        if(arrow_choice==frame[num].count) arrow_choice=0;
        if(arrow_choice<0) arrow_choice=frame[num].count-1;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                key_geted = arrow_choice;
                goto_xy(x+arrow_choice,y);
                write_string(x+arrow_choice,y,frame[num].menu[arrow_choice],A_INTENSE);
                hide_cursor();
            }
        }while(TRUE);
    return 0;
  }
int get_resp_sub(num)
int num;
{
        union inkey {
                char ch[2];
                int i;
        } c;
        int key_choice;
        int x,y;
        arrow_get = 0;
        x = frame[num].startx+1;
        y = frame[num].starty+1;
        do {
                c.i = get_key();
                if(c.ch[0]) {
                        if(c.ch[0] == ESCAPE) return -1;
                        if(c.ch[0] == ' ') break;
                        if(c.ch[0] == ENTER)
                        {
                        key_recive=arrow_item;
                        return arrow_item;
                }
        key_choice = is_in(frame[num].keys,tolower(c.ch[0]));
        if(key_choice==0) return -10;
        goto_xy(x+key_recive,y);
        write_string(x+key_recive,y,frame[num].menu[key_recive],A_INVERSE);
        key_recive=key_choice-1;
        if(key_recive) {
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            if(key_recive==frame[num].count) key_recive=0;
            if(key_recive<0) key_recive=frame[num].count-1;
            goto_xy(x+key_recive,y);
            write_string(x+key_recive,y,frame[num].menu[key_recive],A_INTENSE);
            hide_cursor();
} else {
            goto_xy(x+key_recive,y);
            write_string(x+key_recive,y,frame[num].menu[key_recive],A_INTENSE);
            hide_cursor();
}
arrow_item=key_recive;
return key_recive;
} else {
            goto_xy(x+arrow_item,y);
            write_string(x+arrow_item,y,frame[num].menu[arrow_item],A_INVERSE);

            switch(c.ch[1]) {
                        case UARROW:
                                    arrow_item--;
                                    break;
                        case DARROW:
                                    arrow_item++;
                                    break;
                        }
if(arrow_item==frame[num].count) arrow_item=0;
if(arrow_item<0) arrow_item=frame[num].count-1;
key_recive = arrow_item;
goto_xy(x+arrow_item,y);
write_string(x+arrow_item,y,frame[num].menu[arrow_item],A_INTENSE);
            hide_cursor();
      }
    }while(TRUE);
    return 0;
}
int get_resp_sub1(num)
int num;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
  union inkey
   {
        char ch[2];
        int i;
   } c;
  int key_choice;
  int x,y;

  arrow_get = 0;
  x = frame[num].startx+1;
  y = frame[num].starty+1;
  do { c.i = get_key();
        if(c.ch[0])
          { if(c.ch[0] == ESCAPE) return -1;
            if(c.ch[0] == ' ') break;
            if(c.ch[0] == ENTER)
              {
                key_recive1=arrow_item1;
                return arrow_item1;
              }
            key_choice = is_in(frame[num].keys,tolower(c.ch[0]));
            if(key_choice==0) return -10;
            goto_xy(x+key_recive1,y);
            write_string(x+key_recive1,y,frame[num].menu[key_recive1],A_INVERSE);
            key_recive1 = key_choice-1;
            if(key_recive1)
              { if(key_recive1==frame[num].count) key_recive1=0;
                  if(key_recive1<0) key_recive1=frame[num].count-1;
                  goto_xy(x+key_recive1,y);
                  write_string(x+key_recive1,y,frame[num].menu[key_recive1],A_INTENSE);
                  hide_cursor();
              } else
                   { goto_xy(x+key_recive,y);
                     write_string(x+key_recive1,y,frame[num].menu[key_recive1],A_INTENSE);
                     hide_cursor();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                }
            arrow_item1=key_recive1;
            return key_recive1;
          } else
            { goto_xy(x+arrow_item1,y);
                write_string(x+arrow_item1,y,frame[num].menu[arrow_item1],A_INVERSE);
                switch(c.ch[1])
                 { case UARROW:
                        arrow_item1--;
                        break;
                   case DARROW:
                        arrow_item1++;
                        break;
                 }
                if(arrow_item1==frame[num].count) arrow_item1=0;
                if(arrow_item1<0) arrow_item1=frame[num].count-1;
                key_recive1=arrow_item1;
                goto_xy(x+arrow_item1,y);
                write_string(x+arrow_item1,y,frame[num].menu[arrow_item1],A_INTENSE);
                hide_cursor();
            }
      } while(TRUE);
  return 0;
  }
int is_in(s,c)
char *s,c;
  {
    register int i;
    for(i=0;*s;i++)
    if(*s++==c) return i+1;
    return 0;
  }
void window_(num)
int num;
  {
    int vmode;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    int x,y;
    vmode = video_mode();
    if((vmode!=2) && (vmode !=3) && (vmode !=7))
     { printf("video must be in 80 column text mode");
          exit(1);
     }
    if(vmode==7) vid_mem = (char far *) 0x0000000;
    else vid_mem = (char far *) 0xb8000000l;

    if(!frame_win[num].active)
     {
          save_video_win(num);
          frame_win[num].active =1;
     }
    if(frame_win[num].border) draw_border1(num);
    display_header(num);
    x = frame_win[num].startx + frame_win[num].curx +1;
    y = frame_win[num].starty + frame_win[num].cury + 1;
    goto_xy(x,y);
  }
int make_window(num,header,startx,starty,endx,endy,border)
int num;
char *header;
int startx,starty;
int endx,endy;
int border;
  { unsigned char *p;
    if(num>MAX_FRAME)
     { printf("Too many windows \n ");
          return 0;
     }
    if((startx>24) || (startx<0) || (starty>78) || (starty<0))
     { printf("range error");
          return 0;
     }
    if((endx>24) || (endy>79))
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    { printf("window won't fit");
        return 0;
    }
  p = (unsigned char *) malloc(2*(endx-startx+1)*(endy-starty+1));
  if(!p) exit(1);
  frame_win[num].startx = startx; frame_win[num].endx = endx;
  frame_win[num].starty = starty; frame_win[num].endy = endy;
  frame_win[num].p = p;
  frame_win[num].header = header;
  frame_win[num].border = border;
  frame_win[num].active = 0;
  frame_win[num].curx = 0; frame_win[num].cury = 0;
  return 1;
}
void deactivate_win(num)
int num;
{
  frame_win[num].curx = 0;
  frame_win[num].cury = 0;
  restore_video_win(num);
}
void display_header(num)
int num;
{
  register  int y, len;
  y = frame_win[num].starty;
  len = strlen(frame_win[num].header);
  len = (frame_win[num].endy - y - len) / 2;
  if(len<0) return;
  y = y + len;
  write_string(frame_win[num].startx,y,frame_win[num].header,A_INVERSE);
}
void draw_border1(num)
int num;
{ register int i;
  char far *v,far  *t;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    v=vid_mem;
    t=v;
    for(i=frame_win[num].startx+1;i<frame_win[num].endx;i++)
     { v += (i*160) + frame_win[num].starty*2;
          *v++ = C_VD;
          *v = A_INVERSE;
          v=t;
          v +=(i*160) + frame_win[num].endy*2;
          *v++ = C_VD;
          *v = A_INVERSE;
          v=t;
     }
    for(i=frame_win[num].starty+1; i<frame_win[num].endy;i++)
     { v += (frame_win[num].startx*160)+i*2;
          *v++ = C_HD;
          *v = A_INVERSE;
          v = t;
          v += (frame_win[num].endx*160) + i*2;
          *v++ = C_HD;
          *v = A_INVERSE;
          v = t;
     }
    write_char(frame_win[num].startx,frame_win[num].starty,C_ULD,A_INVERSE);
    write_char(frame_win[num].startx,frame_win[num].endy,C_URD,A_INVERSE);
    write_char(frame_win[num].endx,frame_win[num].starty,C_LLD,A_INVERSE);
    write_char(frame_win[num].endx,frame_win[num].endy,C_LRD,A_INVERSE);
  }
int window_puts(num,str,attr)
int num,attr;
char *str;
  { if(!frame_win[num].active) return 0;
    for(;*str;str++)
      window_putchar(num,*str,attr);
    return 1;
  }
int window_putchar(num,ch,at)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int num,at;
char ch;
  { register int x,y;
    char far *v;
    if(!frame_win[num].active) return 0;
    x = frame_win[num].curx + frame_win[num].startx + 1;
    y = frame_win[num].cury + frame_win[num].starty + 1;
    v = vid_mem;
    v += (x*160) + y*2;
    if(y>=frame_win[num].endy)
      return 1;
    if(x>=frame_win[num].endx)
      return 1;
    if(ch=='\n')
      { x++;
          y = frame_win[num].startx+1;
          v = vid_mem;
          v += (x*160) + y*2;
          frame_win[num].curx++;
          frame_win[num].cury = 0;
      } else
            { frame_win[num].cury++;
              *v++ = ch;
              *v++ = at;
            }
    window_xy(num,frame_win[num].curx,frame_win[num].cury);
    return 1;
  }
int window_xy(num,x,y)
int num,x,y;
  {
    if(x<0 || x+frame_win[num].startx>=frame_win[num].endx-1)
    return 0;
    if(y<0 || y+frame_win[num].starty>=frame_win[num].endy-1)
    return 0;
    frame_win[num].curx = x;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้นำไปดัดแปลง ทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    frame_win[num].cury = y;
    goto_xy(frame_win[num].startx+x+1,frame_win[num].starty+y+1);
    return 1;
  }
void window_gets(num,s)
int num;
char *s;
  { char ch,*temp;
    temp = s;
    for(;;)
      { ch = window_getche(num);
          switch(ch)
            { case ESCAPE:
                  *s = -1;
                  return;
              case ENTER:
                  *s='\0';
                  return;
              case BKSP :
                  if(s>temp)
                    { s--; frame_win[num].cury--;
                      if(frame_win[num].cury<0) frame_win[num].cury = 0;
                      window_xy(num,frame_win[num].startx+
frame_win[num].curx+1,frame_win[num].starty+frame_win[num].cury+1);
                    }
                  break;
              default :
                  *s = ch;
                  s++;
            }
      }
  }
int window_getche(num)
int num;
  { union inkey {
      char ch[2];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        int i;
        } c;
    if(!frame_win[num].active) return 0;
    window_xy(num,frame_win[num].curx,frame_win[num].cury);
    c.i = bioskey(0);
    if(c.ch[0])
      { switch(c.ch[0])
          { case ENTER:
              break;
            case BKSP:
              break;
            default:
              if(frame_win[num].cury+frame_win[num].starty < frame_win[num].endy-1)
                {
write_char(frame_win[num].startx+frame_win[num].curx+1,frame_win[num].starty+frame_win[num].cury+1,c.ch[0]
,A_INVERSE);
                  frame_win[num].cury++;
                }
          }
        if(frame_win[num].curx < 0)frame_win[num].curx = 0;
        if(frame_win[num].curx+frame_win[num].startx >frame_win[num].endx-2)
        frame_win[num].curx--;
        window_xy(num,frame_win[num].curx,frame_win[num].cury);
      }
    return c.i;
  }
void window_cls(num)
int num;
  { register int i,j;
    char far *v,far *t;
    v=vid_mem;
    t=v;
    for(i=frame_win[num].starty+1;i<frame_win[num].endy;i++)
    for(j=frame_win[num].startx+1;j<frame_win[num].endx;j++){
    v=t;
    v+=(j*160) + i*2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    *v++ = ' ';
    *v = A_INVERSE;
    }
    frame_win[num].curx = 0;
    frame_win[num].cury = 0;
  }
void window_cleol(num)
int num;
  {
    register int i,x,y;

    x=frame_win[num].curx;
    y=frame_win[num].cury;
    window_xy(num,frame_win[num].curx,frame_win[num].cury);
    for(i=frame_win[num].cury;i<frame_win[num].endy-1;i++)
    window_putchar(num, ' ',A_INVERSE);
    window_xy(num,x,y);
  }
int window_upline(num)
int num;
  {
    if(frame_win[num].curx>0) {
          frame_win[num].curx--;
          window_xy(num,frame_win[num].curx,frame_win[num].cury);
    }
    return 1;
  }
int window_downline(num)
int num;
  {
    if(frame_win[num].curx<frame_win[num].endx-frame_win[num].startx-1)
    {
          frame_win[num].curx++;
          window_xy(num,frame_win[num].curx,frame_win[num].cury);
          return 1;
    }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    return 0;
  }
void window_bksp(num)
int num;
  {
    if(frame_win[num].cury>0) {
    frame_win[num].cury--;
    window_xy(num,frame_win[num].curx,frame_win[num].cury);
    window_putchar(num, ' ',1);
    frame_win[num].cury--;
    window_xy(num,frame_win[num].curx,frame_win[num].cury);
    }
  }
void save_video_win(num)
int num;
  {
    register int i,j;
          unsigned char *buf_ptr;
    char far *v,far *t;
    buf_ptr = frame_win[num].p;
    v = vid_mem;
    for(i=frame_win[num].starty; i<frame_win[num].endy+1;i++)
    for(j=frame_win[num].startx;j<frame_win[num].endx+1;j++) {
          t = v + (j*160) + i*2;
          *buf_ptr++ = *t++;
          *buf_ptr++ = *t;
          *(t-1) = ' ';
          }
  }
void restore_video_win(num)
int num;
  {
    register int i,j;
    char far *v,far *t;
    unsigned char *buf_ptr;
    buf_ptr = frame_win[num].p;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    v = vid_mem; t= v;
    for(i=frame_win[num].starty;i<frame_win[num].endy+1;i++)
    for(j=frame_win[num].startx;j<frame_win[num].endx+1;j++)
      { v=t;
            v+=(j*160) +i*2;
            *v++ = *buf_ptr++;
            *v = *buf_ptr++;
      }
    frame_win[num].active = 0;
  }
int get_special()
  {
  union inkey { char ch[2];
                int i;
                } c;
    while(!bioskey(1));
    c.i=bioskey(0);
    return c.ch[1];
  }
extern char *cptr;
extern int  last,menucount,menulen,mouse_here,*hptr,pcol,prow;
extern int  arrow_get,arrow_item,arrow_item1;
extern int  key_geted,key_recive,key_recive1;
extern int arrow_choice,len_deso;

    /* start of quit menu */
int menu_quit()
{
        int choice;
        char ch;

        arrow_choice=0;key_geted=0;
        pulldown(3);
        while((choice=get_resp(3)) != -1) {
                switch(choice) {
                case 0:
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                        disp_version();
                        break;
                case 1:
                        window_(3);
                        window_cls(3);
                        window_xy(3,0,1);
                        window_puts(3,"Are you sure exit to dos [y/n] ? ",A_INVERSE);
                        window_xy(3,0,36);
                        ch = window_getche(3);
                        if(tolower(ch)=='y') {
                                deactivate_win(3);
                                restore_video(3);
                                exit(0);                                /* return -2 */
                        }else { deactivate_win(3);hide_cursor(); }
                        break;
                }
        }
    restore_video(3);
    return 0;
}
void  disp_version()
{
    window_(0); window_cls(0);
    window_xy(0,1,(40-strlen(MES_LINE1))/2);
    window_puts(0,MES_LINE1,A_INVERSE);
    window_xy(0,2,(40-strlen(MES_LINE2))/2);
    window_puts(0,MES_LINE2,A_INVERSE);
    window_xy(0,3,(40-strlen(MES_LINE3))/2);
    window_puts(0,MES_LINE3,A_INVERSE);
    window_xy(0,4,(40-strlen(MES_LINE4))/2);
    window_puts(0,MES_LINE4,A_INVERSE);
    window_xy(0,5,(40-strlen(MES_LINE5))/2);
    window_puts(0,MES_LINE5,A_INVERSE);
    window_xy(0,6,(40-strlen(MES_LINE6))/2);
    window_puts(0,MES_LINE6,A_INVERSE);
    window_xy(0,8,(40-strlen(MES_LINE7))/2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            window_puts(0,MES_LINE7,A_INVERSE);
            hide_cursor();
            get_key();
            deactivate_win(0);
}

void read_airpoint(void)
{ FILE *file;
  int i=0;

  file=fopen("airpoint.dat","rt");
  while(i<NUM_AIRPOINT&& !feof(file))
            { fscanf(file,"%s %f %f",&airpoint[i].air_name,
                        &airpoint[i].py,&airpoint[i].px);
              i++;
            }
}

void show_map(void)
{
   int i=0;
   float x0,y0,x1,y1;
            FILE *file;
            file=fopen("point.dat","rt");
            if(!file){
             printf("can' t open file point.dat");
             exit(1);
             }
            setcolor(15);
            while(i<2000 && !feof(file))
              { fscanf(file,"%f %f %f %f",&x0,&y0,&x1,&y1);
                        line(x0,y0,x1,y1);
              }
}

void close_graphics_mode(void)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
{
        closegraph();
}
void set_flight(void)
{
        void far   *image;
        static int no_flight=0;
        char num_flight[4],num_flight_select[4];
        int  i,x,y,end,flight=1,no_flight_select=0;

        for(i=0;i<NO_ALL_FLIGHT;i++)
          { flight_select[i]=0;
                status[i]=0;
          }
        y=getmaxy()-120;

        image = farmalloc(imagesize(0,y,getmaxx(),getmaxy()-20));
        if(image==NULL)
          return;
        getimage(0,y,getmaxx(),getmaxy()-20,image);
        setfillstyle(SOLID_FILL,LIGHTCYAN);
        setcolor(MAGENTA);
        bar(0,y,getmaxx(),getmaxy()-20);
        y=y+10;
        outtextxy(20,y,"SELECT FLIGHT PLAN");
        outtextxy(20,y+15,"SELECT NUMBER OF FLIGHT : ");
        outtextxy(20,y+30,"NUMBER FLIGHT SELECTED : ");
        outtextxy(20,y+45,"TIME  CALL-SIGN  FLIGHT  LEVEL  SPEED  HEADING  FROM   AIRWAY
DISTINATION");
        outtextxy(20,y+60,"0101  A3341      F28     270    420    115      BETNO  G463    BKK");

        setfillstyle(SOLID_FILL,MAGENTA);
        setcolor(LIGHTCYAN);
        x=20+textwidth("SELECT NUMBER OF FLIGHT : ");
        y=y+15;
        bar(x-2,y-1,x+32,y+9);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
sprintf(num_flight,"%d",no_flight);
outtextxy(x,y,num_flight);
for(end=0;end==0;)
 { switch(bioskey(0))
        { case UP_ARROW : if(no_flight == 0) no_flight = NO_ALL_FLIGHT;
                          else no_flight -= 1;
                          bar(x-2,y-1,x+32,y+9);
                          sprintf(num_flight,"%d",no_flight);
                          outtextxy(x,y,num_flight);
                          break;
          case DW_ARROW : if(no_flight == NO_ALL_FLIGHT) no_flight = 0;
                          else no_flight += 1;
                          bar(x-2,y-1,x+32,y+9);
                          sprintf(num_flight,"%d",no_flight);
                          outtextxy(x,y,num_flight);
                          break;
          case RETURN   : setfillstyle(SOLID_FILL,LIGHTCYAN);
                          setcolor(MAGENTA);
                          bar(x-2,y-1,x+32,y+9);
                          sprintf(num_flight,"%d",no_flight);
                          outtextxy(x,y,num_flight);
                          end=1;
                          break;
          case ESC      : putimage(0,getmaxy()-120,image,COPY_PUT);
                          farfree(image);
                          return;
        }
 }
setfillstyle(SOLID_FILL,MAGENTA);
setcolor(LIGHTCYAN);
y=y+15;
x=20+textwidth("NUMBER FLIGHT SELECTED  : ");
sprintf(num_flight_select,"%d",no_flight_select);
bar(x-2,y-1,x+32,y+9);
outtextxy(x,y,num_flight_select);
show_data(1,y+30);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for(end=0;end==0;)
 { switch(bioskey(0))
        { case UP_ARROW : if(flight == 1) flight = NO_ALL_FLIGHT;
                          else flight -= 1;
                          show_data(flight,y+30);
                          if(status[flight]==1)
                            outtextxy(10,y+30,"*");
                          break;
          case DW_ARROW : if(flight == NO_ALL_FLIGHT) flight = 1;
                          else flight += 1;
                          show_data(flight,y+30);
                          if(status[flight]==1)
                            outtextxy(10,y+30,"*");
                          break;
          case RETURN   : if(no_flight_select==no_flight) warn("FINISH SELECTING
FLIGHT");
                          else if(status[flight]==1) warn("THIS FLIGHT WAS SELECTED
ALREADY");
                          else {
                           outtextxy(10,y+30,"*");
                           status[flight]=1;
                           flight_select[no_flight_select]=flight-1;
                           no_flight_select += 1;
                           sprintf(num_flight_select,"%d",no_flight_select);
                           bar(x-2,y-1,x+32,y+9);
                           outtextxy(x,y,num_flight_select);
                           if(no_flight_select==no_flight) {
                                   setfillstyle(SOLID_FILL,LIGHTCYAN);
                                   setcolor(MAGENTA);
                                   bar(x-2,y-1,x+32,y+9);
                                   outtextxy(x,y,num_flight_select);
                                   warn("FINISH SELECTING FLIGHT");
                           }
                          }
                          break;
          case ESC      : putimage(0,getmaxy()-120,image,COPY_PUT);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                                        farfree(image);
                                        end=1;
                                        Num_flight = no_flight_select;
                                        return;
                        }
                }
}
void warn(char message[50])
 {
   void far *image;
   int      half_size;

   half_size  = textwidth(message)/2+30;
   image = farmalloc(imagesize(midx()-half_size,300,midx()+half_size,330));
   if(image)
   {
     getimage(midx()-half_size,300,midx()+half_size,330,image);
     setfillstyle(SOLID_FILL,RED);
     setcolor(WHITE);
     bar(midx()-half_size,300,midx()+half_size,330);
     outtextxy(midx()-half_size+30,310,message);
     getch();
     putimage(midx()-half_size,300,image,COPY_PUT);
     farfree(image);
   }
   else  printf("\007");
 }

int midx(void)
{
 return (getmaxx()/2);
}

int midy(void)
{
 return (getmaxy()/2);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
}

void open_f(void)
{ int n;

  fp=fopen("flight.dat","rb");
  rewind(fp);
  for(n=0;n<NO_ALL_FLIGHT && !feof(fp);n++)
        { /* fscanf(fp,"%d\t%s\t%s\t%d\t%d\t%d\t%s\t%s\t%s\n", */
          fscanf(fp,"%d%s%s%d%d%d%s%s%s\n",
                &(flight[n].time),flight[n].call_sign,flight[n].flight,
                &(flight[n].level),&(flight[n].speed),&(flight[n].heading),
                flight[n].from,flight[n].airway,flight[n].destination);
        }
  fclose(fp);
}
void close_f(void)
{ int n;

  fp=fopen("flight.dat","wb");
  for(n=0;n<NO_ALL_FLIGHT;n++)
        { fprintf(fp,"%d\t%s\t%s\t%d\t%d\t%d\t%s\t%s\t%s\n",
                flight[n].time,flight[n].call_sign,flight[n].flight,
                flight[n].level,flight[n].speed,flight[n].heading,
                flight[n].from,flight[n].airway,flight[n].destination);
        }
  fclose(fp);
}

void display_flight_data()
{
  int   driver =DETECT ,mode;
  int   n,x,y;
  char  str[4];
  initgraph(&driver,&mode,"c:\\lang\\bc\\bgi");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
if(!fp)
        warn("NO FILE LOADED");
else {
        setfillstyle(SOLID_FILL,LIGHTCYAN);
        setcolor(LIGHTMAGENTA);
        bar(0,0,getmaxx(),getmaxy());
        rectangle(10,30,getmaxx()-10,getmaxy()-10);
        rectangle(15,35,getmaxx()-15,getmaxy()-15);
        setfillstyle(SOLID_FILL,LIGHTMAGENTA);
        setcolor(LIGHTCYAN);
        bar(midx()-100,25,midx()+100,40);
        outtextxy(midx()-96,28,"DISPLAY FLIGHT PLAN DATA");
        setfillstyle(SOLID_FILL,LIGHTCYAN);
        setcolor(LIGHTMAGENTA);

        outtextxy(30,50,"TIME  CALL-SIGN  FLIGHT  LEVEL  SPEED  HEADING  FROM  AIRWAY
DESTINATION");
   y=70;
   for(n=0;n<NO_ALL_FLIGHT;n++)
    {
        x=30;
        sprintf(str,"%d",flight[n].time);
        outtextxy(x,y,str);
        x+=textwidth("TIME   ");
        sprintf(str,"%s",flight[n].call_sign);
        outtextxy(x,y,str);
        x+=textwidth("CALL-SIGN   ");
        sprintf(str,"%s",flight[n].flight);
        outtextxy(x,y,str);
        x+=textwidth("FLIGHT   ");
        sprintf(str,"%d",flight[n].level);
        outtextxy(x,y,str);
        x+=textwidth("LEVEL   ");
        sprintf(str,"%d",flight[n].speed);
        outtextxy(x,y,str);
        x+=textwidth("SPEED   ");
```

เอกสารนี้สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น ห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            sprintf(str,"%d",flight[n].heading);
            outtextxy(x,y,str);
            x+=textwidth("HEADING  ");
            sprintf(str,"%s",flight[n].from);
            outtextxy(x,y,str);
            x+=textwidth("FROM  ");
            sprintf(str,"%s",flight[n].airway);
            outtextxy(x,y,str);
            x+=textwidth("AIRWAY  ");
            sprintf(str,"%s",flight[n].destination);
            outtextxy(x,y,str);
            y=y+12;
      }
     getch();
   }
   closegraph();
   make_menu(6, rad_scp,"mafp",4,2,17,BORDER);
   cls();
   cursor(2,5);
   menu_start(menu0,0,7);
   first_active(menu0,0,7);
   menu_choice(menu0,0,7);
}

void flight_run(void)
{ int  i,n,run=1,f;
   clock_t now_time=0,start_time;

   start_time=clock()-3600;
/*  now_time=3600; */
   for(n=0; n<Num_flight; n++)
            { f=flight_select[n];
              flight[f].cur_t = time_2_long(flight[f].time);
              i = find_airpoint(flight[f].from);
              flight[f].cur_x = airpoint[i].px;
              flight[f].cur_y = airpoint[i].py;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        }
    while(run)
        { for(n=0,run=0; n<Num_flight; n++)
            { now_time=clock()-start_time;
              run += each_flight_run(flight_select[n],now_time);        /* status for not complete return
true */
              show_time(now_time/3600,(now_time%3600)/60);
              if(kbhit())  if(getch()=='q')  return;
            }
        }
    warn("Already flight to destination complete");
}

int each_flight_run(int flight_order,long now_time)                    /* return 1 for not complete */
{ long flight_time;
  float *p_x=&(flight[flight_order].cur_x);
  float *p_y=&(flight[flight_order].cur_y);

  flight_time=time_2_long(flight[flight_order].time);
  if(now_time-flight_time>=0)
        { if(current_position(flight_order,now_time,p_x,p_y))
            { check_status(flight_order);
              if(*p_x>=94 && *p_x<113.4 && *p_y<=21 && *p_y>8.2)
                  putpixel(33.*(*p_x-94.),(21.-*p_y)*37.5,YELLOW);
            } else return 0;
    }
  return 1;
}

long time_2_long(int t)
{ return ((t/100)*3600 + (t%100)*60);
}

void check_status(int Flight)
{ float x11,x21,y11,y21,Speed,rt;
  /* x12,x22,y12,y22,m1,m2,x,y; */
```

บทคัดย่อนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่อนุญาตให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
int n;

for(n=0; n<Num_flight; n++)
        { if(flight[Flight].level==flight[flight_select[n]].level && flight_select[n]!=Flight)
                { x11 = flight[Flight].cur_x;
                  y11 = flight[Flight].cur_y;

                  /* j = find_airpoint(flight[Flight].destination);
                        x12 = airpoint[j].px;
                        y12 = airpoint[j].py;

                        j = find_airpoint(flight[n].destination);
                        x22 = airpoint[j].px;
                        y22 = airpoint[j].py;
                  */

        x21 = flight[flight_select[n]].cur_x;
                  y21 = flight[flight_select[n]].cur_y;
                  Speed = flight[Flight].speed+flight[n].speed;
                  rt = r(x21-x11,y21-y11);
                  if(rt<(Speed*speed_fact*180/3600.0))
                        warn("CLASH  CLASH  CLASH!!!");
/*
                  if(x12==x11 || x22==x21)
                        return 0;
                  m1 = (y12-y11)/(x12-x11);
                  m2 = (y22-y21)/(x22-x21);
                  if(m1!=m2)
                        { x = ((y21-y11)+m1*x11-m2*x21) / (m1-m2);
                          y = m1*(x-x11)+y11;
                          status = (r(x-x11,y-y11)<=r(x12-x11,y12-y11));
                          status &= (r(x-x21,y-y21)<=r(x22-x21,y22-y21));
                        }
*/
                }
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
  /*
  flight[flight].speed
  flight[flight].from
  flight[flight].destination
  */
}

float r(float a,float b)
{  return sqrt(a*a + b*b); }

int current_position(int Flight,long cur_time,float *px,float *py)
{ float distance,factor;                    /* run_time in unit minuts */
  long run_time;
  int j;

  j = find_airpoint(flight[Flight].destination);
  run_time = cur_time- flight[Flight].cur_t;
  distance = sqrt(pow(*px-airpoint[j].px,2)+pow(*py-airpoint[j].py,2));
  factor   = (flight[Flight].speed * speed_fact * run_time/3600.0)/distance;
  if(factor<1)
          { *px=((airpoint[j].px- *px)*factor+ *px);
            *py=((airpoint[j].py- *py)*factor+ *py);
            flight[Flight].cur_t = cur_time;
            return 1;
          }
  return 0;
}

int find_airpoint(char *name)
{ int i,found;
  char msg[40];

  for(i=-1,found=0; i<NUM_AIRPOINT && !found; i++)
          found = !strcmp(airpoint[i+1].air_name,name);
  if(!found)
          { strcpy(msg,"No data of fixpoint ");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                strcat(msg,name);
                warn(msg);
                exit(0);
              }
    return i;
}
void show_time(int hr,int min)
{  int old_color;
    static char st_time[10],old_str[10];

    old_color = getcolor();
    itoa(hr+100,st_time,10);
    itoa(min+100,st_time+3,10);
    st_time[0]=' ';
    st_time[3]=':';
    if(strcmp(st_time,old_str)==0)
              return;
    strcpy(old_str,st_time);
    setfillstyle(SOLID_FILL,CYAN);
    setcolor(MAGENTA);
    bar(getmaxx()-50,0,getmaxx(),textheight(" "));
    outtextxy(getmaxx()-50,0,st_time);
    setcolor(old_color);
}

void main()
{   int  back, chk=0,fore,mode;

          read_airpoint();
          make_window(0,"PROJECT",8,19,18,60,BORDER);
          make_window(3,"CONFIRM",10,20,12,59,BORDER);
          make_menu(0, file,"oc",2,2,2,BORDER);
          make_menu(2, rad_scp,"mafp",4,2,17,BORDER);
          make_menu(1, detail,"at",2,2,31,BORDER);
          make_menu(3, quit,"ve",2,2,49,BORDER);
          cls();
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
cursor(2,5);
mode = getvmode();
if (mode == 1 || mode == 3) {
        fore = 0;
        back = 7;
} else {
        fore = 0;
        back = 7;
}
menu_start(menu0,fore,back);
disp_version();
do {
        first_active(menu0,fore,back);
        chk = menu_choice(menu0,fore,back);

            if(chk == -2) {
            menu_end();
            cls();cursor(1,1);
            exit(0);
            };
        }while(TRUE);
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

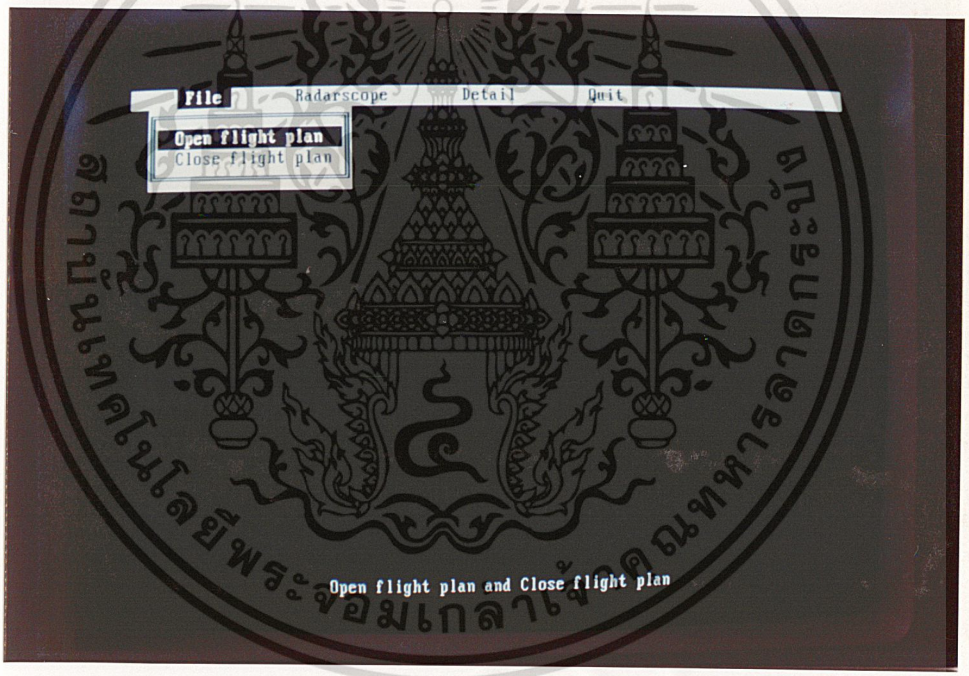
## ผลการทดลอง

จากรูปเป็นส่วนที่แสดงถึงลักษณะของระบบเมนูเรียกว่า ระบบเมนูเมนูป็อป-อัพ พูลดาวน์ซึ่งมีลักษณะเหมือนหน้าต่างของเทอร์โบซีหรือปาสคาล แต่ที่ปรากฏอยู่ในรูปได้ใช้ภาษาซีในการเขียน แบ่งออกเป็นเมนูไฟล์ (File Menu) เมนูเรดาร์สโคบ (Radarscope Menu) เมนูรายละเอียด (Detail Menu)และเมนูการออก (Quit Menu) นอกจากส่วนที่เป็นระบบเมนูแล้วยังแสดงถึงลักษณะของหน้าต่างแสดงรายละเอียดเกี่ยวกับชื่อผู้จัดทำและอาจารย์ที่ปรึกษาซึ่งอยู่บริเวณตรงกลางของหน้าจอคอมพิวเตอร์จากที่ได้กล่าวมาแล้ว เป็นผลมาจากการรันโปรแกรมที่เขียนขึ้นมาเพื่อเป็นส่วนหนึ่งในการจำลองระบบการควบคุมการจราจรทางอากาศแบบพื้นที่

รูปที่ 9.1 แสดงส่วนที่เป็นเมนูป็อบ-อัพ และหน้าต่างรายละเอียด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

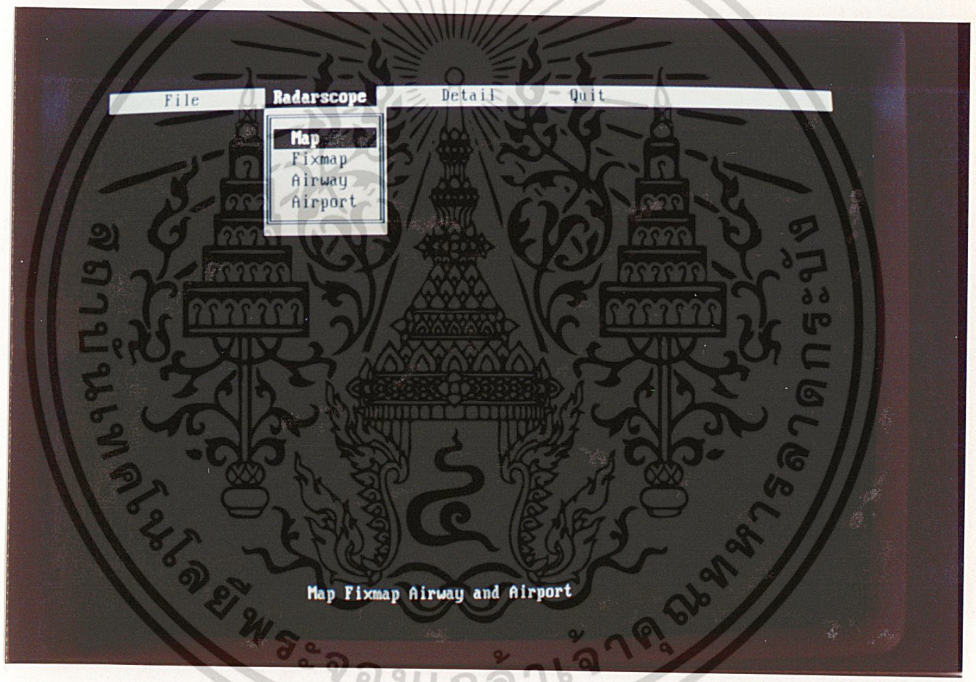
จากรูปที่ 9.2 แสดงถึงลักษณะของระบบเมนูย่อย (Submenu) ที่แยกออกมาจากเมนูป็อปอัพชื่อว่าเมนูไฟล์ ประกอบไปด้วยเมนูย่อย 2 เมนู คือ เมนูย่อยเปิดแผนการบิน (Open Flight Plan) และเมนูย่อยปิดแผนการบิน (Close Fight Plan) โดยที่ในเมนูเปิดแผนการบินจะมีข้อมูลเกี่ยวกับการออกบินของเครื่องบิน เช่น เวลา (Time)ที่ใช้ในการออกบิน นามเรียกขาน (Call-Sign) ชื่อเครื่องบิน (Flight) ความเร็ว (Speed) ระดับเพดานบิน (Ceiling) ทิศที่หัวเครื่องบินทำมุมกับทิศเหนือ (Heading) จุดรายงานที่เครื่องบินออกบิน เส้นทางการบิน (Airway) จุดหมายปลายทาง (Destination) ซึ่งจากที่กล่าวมาทั้งหมดถูกเก็บรวบรวมเป็นแฟ้มข้อมูลชื่อ Airpoint เมื่อต้องการให้เครื่องบินออกบินจะต้องทำการเปิดแฟ้มข้อมูลนี้ก่อน ทำได้โดยการกด F10 หรือการกด Hot Keys จากนั้นทำการเลื่อนแถบสว่างมาที่เมนูย่อยชื่อว่า Open Fight Plan แล้วทำการกด Enter ก็จะเป็นการเปิดแฟ้มข้อมูล




รูปที่ 9.2 แสดงส่วนเมนูย่อย 2เมนูในเมนูไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปเป็นส่วนที่แสดงถึงลักษณะของเมนูป๊อปอัพ ซึ่งประกอบไปด้วยเมนูย่อย 4 เมนู คือเมนูย่อยแผนที่ (Map Submenu) เมนูย่อยจุดรายงาน (Fixmap Submenu)เมนูย่อยเส้นทางการบิน (Airway Submenu)และเมนูย่อยท่าอากาศยาน (Airport Submenu) ซึ่งการเข้าสู่ระบบเมนูย่อยเหล่านี้ทำได้ดังที่ได้กล่าวมาแล้ว




รูปที่ 9.3 แสดงเมนูย่อย map,fixmap,airway และ airport

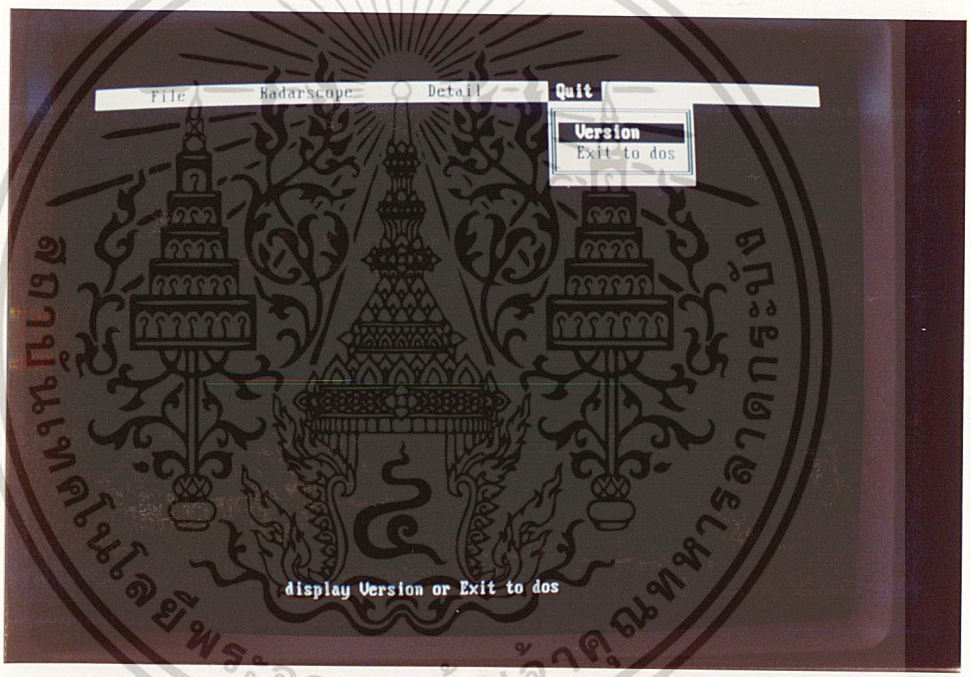
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 9.4 แสดงถึงลักษณะเมนูรายละเอียด (Detail Menu) ซึ่งประกอบไปด้วย เมนูย่อยแอคราฟ และเมนูย่อย ATC เมื่อทำการเลือกแถบสว่างไปที่เมนูแอคราฟแล้วกด Enter จะเป็นการโหลดข้อมูลเกี่ยวกับสมรรถภาพของเครื่องบิน

รูปที่ 9.4 แสดงเมนูย่อยแอคราฟและเมนูย่อย ATC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปเป็นการแสดงถึงลักษณะของเมนูย่อยเวอร์ชั่น (Version Submenu) และเมนูย่อยการกลับไปสู่ดอส (Exit to Dos) การเข้าสู่เมนูย่อยเวอร์ชั่นโดยการเลื่อนแถบสว่างมาที่เมนูดังกล่าวแล้วกด Enter จะเป็นการแสดงรายละเอียดเกี่ยวกับการจัดทำโปรแกรม แต่ถ้าทำการเลื่อนแถบสว่างไปที่ Exit to Dos แล้วกด Enter จะมีข้อความขึ้นมาเพื่อถามว่าต้องการกลับสู่ดอสจริงหรือไม่ ถ้าจริงกด "Y" ถ้าไม่ต้องการก็กด "N"




รูปที่ 9.5 แสดงเมนูย่อยเวอร์ชั่นและเมนูย่อยการกลับสู่ดอส

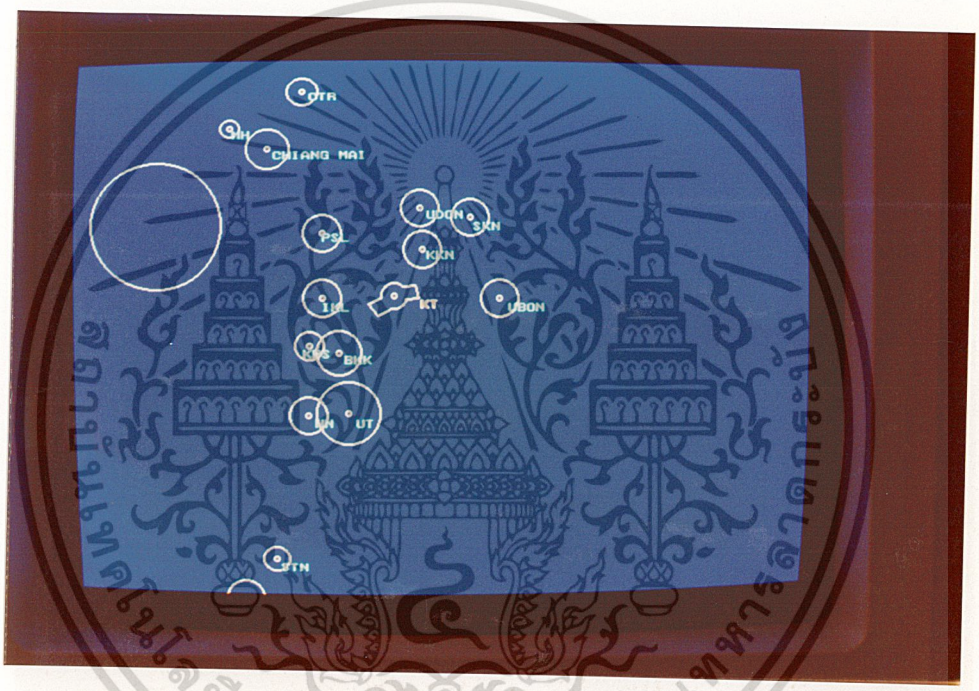
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปเมื่อทำการเลื่อนคีย์ลูกศร 4 ทิศทาง (ส่วนที่เป็นคีย์ล่าง) แล้วเลื่อนแถบสว่างไปที่เมนูย่อยจากนั้นกด Enter จะปรากฏลักษณะแผนที่ที่ Display อยู่บนหน้าจอคอมพิวเตอร์ ซึ่งแผนที่ดังกล่าวได้จากการเขียนโปรแกรมโดยการเก็บรายละเอียดเกี่ยวกับจุดที่ใช้ในการวาดแผนที่ รายละเอียดดังกล่าวได้ทำการเก็บไว้เป็นแฟ้มข้อมูลชื่อ Point.dat เมื่อมีการเลือกเมนูย่อยแผนที่ (กด Enter)จะปรากฏรูปแผนที่ขึ้นโดยอัตโนมัติ นอกจากนี้ยังมีลักษณะบาร์อยู่บริเวณด้านล่าง ซึ่งส่วนดังกล่าวเป็นส่วนที่ใช้ในการเลือกแผนการบินของเครื่องบินลำต่างๆ กล่าวคือ สามารถเลือกได้ว่าจะให้เครื่องบินลำไหนออกบินได้ โดยการกดคีย์ลูกศรลง (Down Key) ซึ่งสามารถเลือกให้เครื่องบินออกบินกี่ลำก็ได้ โดยมีลิมิตสูงสุด 20 ลำ

รูปที่ 9.6 แสดงรูปแผนที่และส่วนที่ใช้ในการเลือกเที่ยวบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

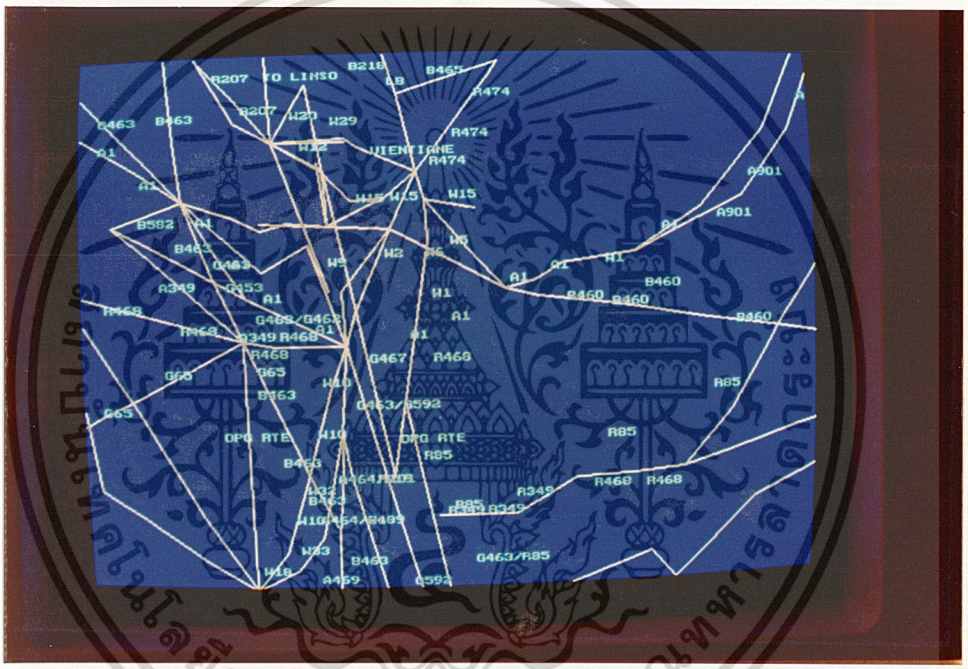
จากรูปเป็นส่วนที่แสดงถึงลักษณะของบริเวณท่าอากาศยาน ซึ่งข้อมูลที่ใช้ในการแสดงท่าอากาศยานจะถูกเก็บไว้ในแฟ้มข้อมูลสามารถเรียกให้แสดงบริเวณที่ตั้งได้โดยการเลื่อนแถบสว่างมาที่ Airport แล้วทำการกด Enter




รูปที่ 9.7 แสดงลักษณะที่ตั้งของเมืองท่าอากาศยาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

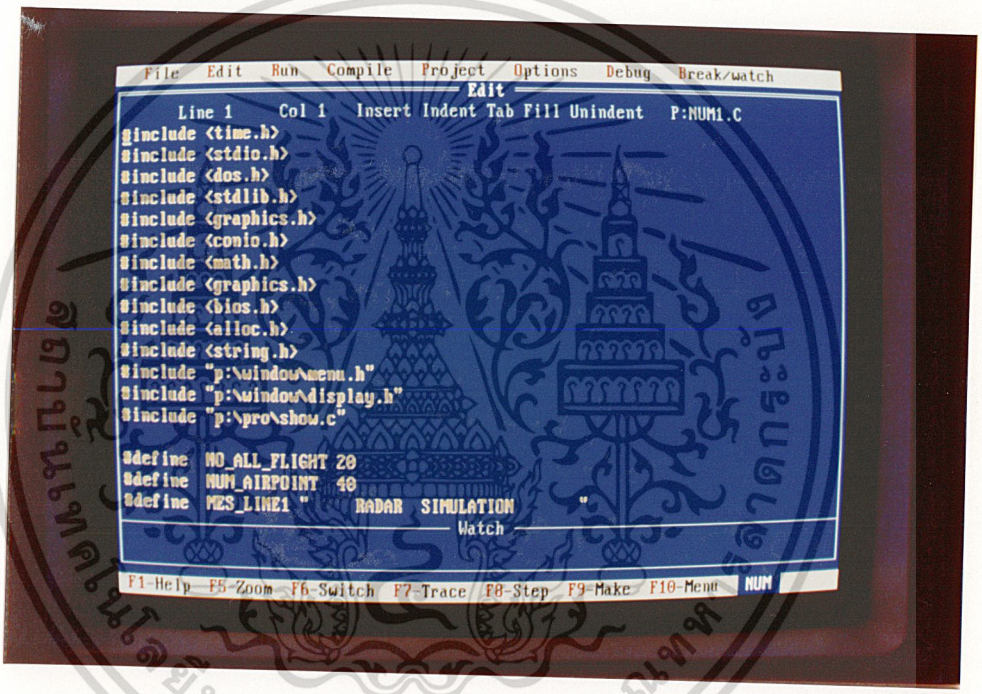
จากรูปเป็นส่วนที่แสดงถึงลักษณะของจุดรายงาน กล่าวคือ เป็นจุดที่เครื่องบินบินผ่านตลอดเส้นทางการบิน ข้อมูลเกี่ยวกับจุดรายงานถูกเก็บเป็นแฟ้มข้อมูล จุดรายงานแต่ละจุดได้มาจากการคำนวณบริเวณที่ตั้งจริงเทียบกับจุดพิกเซลบนหน้าจอคอมพิวเตอร์




รูปที่ 9.6 แสดงถึงลักษณะของเส้นทางการบิน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงลักษณะหน้าจอคอมพิวเตอร์ที่ใช้ในการเขียนโปรแกรมการควบคุมการจราจรทางอากาศแบบควบคุมพื้นที่




รูปที่ 9.8 แสดงลักษณะหน้าจอคอมพิวเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# สรุป

จากการเขียนโปรแกรมเรดาร์ ซิมมูเลเตอร์ เพื่อใช้ในการฝึกเจ้าหน้าที่ควบคุมการจราจรทางอากาศให้มีความพร้อมความชำนาญก่อนการไปปฏิบัติงานจริงๆ โดยระบบโปรแกรมที่เขียนขึ้นมาสามารถทำงานได้บนเครื่องคอมพิวเตอร์ส่วนบุคคลทั่วๆไป ภาษาที่ใช้ในการเขียนโปรแกรมใช้ภาษาซีและกราฟิกส์ภาษาซีในการเขียน ซึ่งเป็นภาษาซีเป็นภาษาซีมีโครงสร้างฟังก์ชั่นการทำงานจำนวนมากให้เลือกใช้ได้ จากโปรแกรมได้แบ่งส่วนที่ใช้ในการเขียนโปรแกรมออกเป็น 2 ส่วนด้วยกัน คือ

1. ส่วนของการเขียนโครงสร้างระบบเมนู ซึ่งแบ่งออกเป็น

1.1 ส่วนของเมนูป็อบอัพ (Pop Menu) แยกได้เป็นเมนูไฟล์ (File Menu) เมนูเรดาร์สโคป (Radar Scope) เมนูรายละเอียด (Detail Menu) และเมนูออก (Quit Menu)

1.2 ส่วนของเมนูพูลดาวน์ (Pull Down) ซึ่งแบ่งออกได้เป็นเมนูย่อยเปิดแผนการบิน (Open Flight Plan) เมนูย่อยปิดแผนการบิน (Close Flight Plan) เมนูย่อยแผนที่ (Map Menu) เมนูย่อยจุดรายงาน (Fixmap Menu) เมนูย่อยเส้นทางการบิน (Airway Menu) เมนูย่อยท่าอากาศยาน (Airport Menu) เมนูย่อยแอคราฟ (Aircraft Menu) เมนูย่อยเอทีซี (ATC Menu) เมนูย่อยเวอร์ชั่น (Version Menu) และเมนูย่อยการกลับไปสู่คอส (Exit to Dos) ซึ่งแต่ละเมนูย่อยนั้นสามารถอธิบายได้ดังนี้ คือ

- เมนูย่อยเปิดแผนการบิน จะทำการเก็บข้อมูลที่ใช้ในการออกบินของเครื่องบิน ซึ่งประกอบไปด้วย รายละเอียดเกี่ยวกับเวลาที่จะออกบิน (Time) นามเรียกขานของ (Call Sign) สควอกโค้ด (Squak Code) ความเร็ว (Speed) ระดับเพดานบิน (level) ทิศทางหัวเครื่องบินเมื่อเทียบกับทิศเหนือโดยกำหนดให้ทิศเหนือเป็นทิศที่ 0 องศา (Heading) ท่าอากาศยานที่ออกบิน (From) ท่าอากาศยานที่บินไปถึง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

(Distination)และเส้นทางการบิน (Airway) ซึ่งข้อมูลที่กล่าวมานั้นได้ทำการเก็บรวบรวมไว้เป็นแฟ้มข้อมูลโดยสามารถเรียกใช้ได้เมื่อต้องการใช้และสามารถปิดการใช้ได้เมื่อต้องการเลิก ข้อมูลดังกล่าวจัดเป็นแฟ้มข้อมูลแบบโกลบอล (Global)

- เมนูย่อยปิดแผนการบิน เป็นส่วนที่ใช้ในการปิดแผนการบิน เมื่อไรต้องการใช้แผนการบินนั้นแล้ว
- เมนูย่อยแสดงแผนที่ จะแสดงให้เห็นถึงการแสดงการพล็อตขอบเขตของแผนที่ โดยสามารถสั่งให้มีการพล็อตแผนที่ได้ช้า เร็ว ตามที่ต้องการได้
- เมนูย่อยแสดงจุดรายงาน (Fixmap) จะแสดงให้เห็นถึงจุดรายงานต่างๆ พร้อมกับการแสดงชื่อของจุดรายงานทั้งหลาย
- เมนูย่อยเส้นทางการบิน (Airway) แสดงให้เห็นถึงเส้นทางการบินพร้อมชื่อของเส้นทางการบิน
- เมนูย่อยท่าอากาศยาน แสดงถึงท่าอากาศยาน
- เมนูย่อยแอคราฟ (Aircraft) แสดงให้เห็นถึงการแสดงรายละเอียดของสมรรถภาพของเครื่องบินลำต่างๆ
- เมนูย่อยเวอร์ชั่น (Version) แสดงรายละเอียดเกี่ยวกับผู้จัดทำโปรแกรมเรดาร์ ซิมมูเลเตอร์ อาจารย์ที่ปรึกษาโปรเจค
- เมนูการออกจากโปรแกรมเพื่อกลับไปสู่ดอส (Exit to Dos)

ในส่วนของการแสดงให้เครื่องบินมีการเคลื่อนที่นั้นก่อนได้โดยการเลือกแผนการบินก่อนมีการไปที่ ฟังก์ชั่นการเลือกแผนการบิน (Flight_Select Function) แล้วสั่งให้เครื่องบินบินโดยใช้ฟังก์ชั่นให้เครื่องบินแต่ละลำบิน (Each_Flight_Run Function) ในตลอดเส้นทางการบินนั้นจะมีการเช็คฟังก์ชั่นการชน (Clash Function) โดยมีการกำหนดว่าเครื่องบินจะไม่เกิดการชนกันถ้าระยะห่างระหว่างเครื่องบินแต่ละลำมากกว่า 3 วินาที แต่ถ้าเวลาที่เช็คได้นั้นน้อยกว่า 3 วินาที จะต้องเข้าไปสู่ฟังก์ชั่นการป้องกันการชน (Protect Function) ซึ่งตรงจุดนี้สามารถป้องกันการชนกันของเครื่องบินได้โดยป้อนข้อมูลใหม่ให้กับเครื่องบินที่จะเกิดการชน ซึ่งข้อมูลดังกล่าวจะเป็นพวก ความเร็ว ทิศทาง ฯลฯ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากการเขียนโปรแกรมดังกล่าว ก็ได้เป็นไปตามจุดประสงค์ของความเป็นไปได้ของการทำโปรเจคตามที่อาจารย์ที่ปรึกษาได้ตั้งขอบเขตไว้ แต่อย่างไรก็ตามการเขียนโปรแกรมดังกล่าวยังจะต้องได้รับการปรับปรุงในส่วนต่างๆอีกมาก เช่น ในเรื่องของการย่อยการแสดงผล การสามารถตั้งการแสดงผลกลับมาสู่ศูนย์กลางของแผนที่ได้ การแสดงผลของระบบเวลา การแสดงตำแหน่งรวมถึงโคออร์ดิเนตของเคอร์เซอร์ บนจอทั้งในแบบละเอียด ลองเอียดได้ เพื่อให้สามารถใช้งานได้ดียิ่งขึ้น นอกจากนี้ในการทำโปรเจคในเรื่องเรดาร์ ซิมมูเลเตอร์ จะมีปัญหาในเรื่องของการค้นหาข้อมูล รายละเอียดซึ่งข้อมูลดังกล่าวไม่สามารถหาได้ภายในสถาบันใช้ในการเขียนโปรแกรม

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## แนวทางการพัฒนาต่อไป

สำหรับแนวทางในการพัฒนาโปรแกรมการควบคุมการจราจรทางอากาศ จะทำการสร้างโปรแกรมขึ้นมาอีกหนึ่งส่วน เพื่อจะใช้สำหรับผู้รับการฝึกการควบคุมจราจรทางอากาศ (ATC Tranee) โดยใช้เครื่องคอมพิวเตอร์สองเครื่อง ให้เครื่องหนึ่งสำหรับผู้รับการฝึกโดยรับข้อมูลจากเครื่องคอมพิวเตอร์อีกเครื่องหนึ่ง ที่ผู้ควบคุมการฝึก (ATC Traner) ทำหน้าที่เป็นนักบินจำลอง (Pseudo Pilot) เพื่อใช้ในการควบคุมเครื่องบินที่ปรากฏอยู่บนหน้าจอในรูปแบบที่ต้องการเช่น การเลือกแผนการบินซึ่งประกอบไปด้วยรายละเอียดเกี่ยวกับ นามเรียกขาน ความเร็ว ระดับเพดานบิน เส้นทางการบิน ฯลฯ ผู้รับการฝึกจะต้องฝึกควบคุมการจราจรทางอากาศ ผู้รับการฝึกจะทำการควบคุมผ่านระบบอินเตอร์คอม (Intercom)สั่งการไปยังครูผู้ฝึก ซึ่งทำหน้าที่เป็นนักบินจำลอง เพื่อให้เครื่องบินแต่ละลำสามารถไปถึงจุดหมายปลายทางได้อย่างปลอดภัย

รูปที่ 1 แสดงโครงสร้างระบบเรดาร์ ชิมมูเลเตอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ในส่วนของระบบ ฮาร์ดแวร์ จะประกอบด้วยเครื่องคอมพิวเตอร์ที่มีความสามารถทางด้านกราฟฟิกและมีจอภาพที่ละเอียดสูง ใช้สำหรับผู้รับการฝึกการจราจรทางอากาศและอีกเครื่องหนึ่งใช้เป็นส่วนของผู้ควบคุมหรือนักบินจำลอง (Instructore or pseudo pilot) โดยจะทำการเชื่อมโยงข้อมูลกันทางช่องสื่อสารแบบอนุกรม

- ให้สามารถย่อ,ขยายการแสดงผลได้

- สามารถป้อนข้อมูลเที่ยวบินเมื่อต้องการออกบินได้ตามต้องการ

- สามารถเร่งความเร็วในการทำงานได้ เพื่อดูผลการทำงานอย่างรวดเร็ว

- สามารถลดความเร็วในการคำนวณลงได้ 2 เท่า เพื่อให้สามารถสังเกตุรายละเอียดได้ดียิ่งขึ้น

- สามารถหยุดการทำงานลงชั่วคราวได้ (Pause Time)ให้สามารถเริ่มการทำงานได้ต่อจากที่หยุดไว้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# หนังสืออ้างอิง


[1] เมธี เสรีอรุโณ "ระบบจำลองเรดาร์ควบคุมจราจรทางอากาศแบบควบคุมพื้นที่ " การประชุมใหญ่ทางวิชาการประจำปี 2536 วิศวกรรมสถานแห่งประเทศไทย ในพระบรมราชูปถัมภ์ 27-30 พฤศจิกายน 2536 หน้าที่ 735-747

[2] สหพัฒน์ รุ่งตะวันเรืองศรี "เรียนรู้คอมพิวเตอร์กราฟิกส์ 2 มิติด้วยภาษาซี " กรุงเทพฯ

[3] มนตรี พจนารถลาวัณย์ " การเขียนโปรแกรมคอมพิวเตอร์ด้วยภาษาเทอร์โบซี "

[4] " เฮอร์เบิร์ต ชิลด์ " การประยุกต์ใช้งานภาษาซี " เรียบเรียงโดย
3.1 ศิวัฒน์ ศิวะบวร
3.2 พรชัย จักรธำรงค์
3.3 จิรศักดิ์ ชัยวิริยะกุล

[5] ธันวา ศรีประมง " การโปรแกรมภาษาซี สำหรับวิศวกรรม "สำนักพิมพ์วิทยาลัยมหานคร

[6] ครรชิต มาลัยวงศ์ และ วิชิต ปุณวัตร์ " เทคนิคการออกแบบโปรแกรม "

[7] นระ คมนากุล " การควบคุมการจราจรทางอากาศ " สำนักพิมพ์สถาบันเทคโนโลยีพระจอมเกล้าธนบุรี

[8] BEN EZZELL Graphics Programming in Turbo C 2.0 , November 1988

[9] MALLARD Team " Air Traffic Control "



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

# กิตติกรรมประกาศ

ในการทำปริญญานิพนธ์ฉบับนี้ ขอขอบคุณ อาจารย์ สมยศ จุณะปิยะ ที่ได้ให้คำแนะนำตลอดจนความช่วยเหลือต่างๆ จนทำให้ปริญญานิพนธ์ฉบับนี้สามารถสำเร็จลุล่วงไปด้วยดี นอกจากนี้ต้องขอขอบคุณ

1. คุณ เมธี เสรีอรุโณ เจ้าหน้าที่วิเคราะห์และพัฒนาระบบงานคอมพิวเตอร์ บริษัทวิทยุการบินแห่งประเทศไทยจำกัดม กรุงเทพฯ ที่ได้ให้ข้อมูลที่เป็นประโยชน์ คำแนะนำ และข้อเสนอแนะต่างๆ
2. คุณ สุทัศธร อินทรโชติ
3. คุณ กรรณิการ์ ศรีสันติโรจน์
4. คุณ วารีรัตน์ ปลื้มรุ่งโรจน์
5. คุณ สัญชัย วรรณโพธิ์กลาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

แสดงรายชื่อของจุดรายงาน

| NAME OF FIXMAP | | LATTITUDE | LONGTITUDE |
|---|---|---|---|
| LIMLA | C | 15.81667 | 98.60000 |
| OSUKA | N | 14.80000 | 99.75000 |
| BKK | C | 13.99500 | 100.65333 |
| SELKA | C | 14.40000 | 102.00000 |
| UBL | C | 15.25333 | 104.86500 |
| BUTRA | C | 15.41667 | 105.60000 |
| PUT | C | 8.11167 | 98.31000 |
| ASIDO | C | 7.77167 | 97.71667 |
| HTY | C | 6.93000 | 100.39500 |
| VKB | C | 6.16500 | 102.29333 |
| TANEX | C | 14.05000 | 98.97500 |
| CMA | C | 18.76500 | 98.96500 |
| BEKOD | C | 16.37167 | 99.84833 |
| POLAK | N | 13.34667 | 100.62833 |
| REGOS | C | 12.00000 | 100.58500 |
| DIRAX | N | 11.00000 | 100.55000 |
| UPNEB | C | 9.70167 | 100.49667 |
| RELIP | C | 8.07333 | 100.44167 |
| TAMOS | C | 6.53333 | 100.40500 |
| PUKOK | C | 18.00000 | 97.71667 |
| OSADI | N | 18.32333 | 98.87500 |
| BDRY | C | 19.80000 | 98.95000 |
| BOMAS | C | 17.38333 | 98.10000 |
| TATEL | C | 17.48333 | 98.76667 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| PATKA | C | 8.00000 | 106.50000 |
| TAPON | C | 7.00000 | 105.96667 |
| PAK | C | 15.13333 | 105.78333 |
| EKAVO | C | 11.62500 | 99.51000 |
| MENEK | C | 11.14000 | 99.76500 |
| SUWAN | C | 8.05333 | 101.34167 |
| IDAGA | N | 11.00000 | 100.90000 |
| TIKAL | C | 8.04333 | 101.75000 |
| NOLIK | C | 6.25833 | 102.26000 |
| TOTOX | C | 8.95000 | 97.03333 |
| BETNO | C | 15.09667 | 98.21167 |
| RYN | C | 12.77833 | 101.68167 |
| ALGOR | C | 8.00000 | 104.00000 |
| KABAS | C | 7.00000 | 104.65833 |
| MOSLA | C | 5.82167 | 105.42833 |
| BIBRO | C | 4.24833 | 106.45000 |
| KANTO | C | 6.82667 | 103.80500 |
| BATOK | N | 13.97500 | 101.90000 |
| MENAM | C | 13.95500 | 102.79500 |
| MAKAS | C | 16.82833 | 98.50000 |
| PSL | C | 16.76833 | 100.29500 |
| EKIVI | C | 16.65500 | 101.92833 |
| REMAD | C | 8.00000 | 103.38833 |
| PARAM | C | 8.02000 | 102.64500 |
| BELER | C | 7.00000 | 103.00000 |
| ADPAT | C | 5.93333 | 104.09167 |
| KIBAG | C | 11.00000 | 109.51500 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| DOVEL | C | 9.21333 | 109.26667 |
| SAPAM | C | 8.08667 | 97.54500 |
| ADNEP | N | 8.06000 | 101.05833 |
| SISUK | C | 19.80000 | 98.04833 |
| NAN | C | 18.80833 | 100.78167 |
| ANBOK | C | 18.60833 | 101.21667 |
| CTR | C | 19.94833 | 99.88167 |
| TOMIP | C | 19.18333 | 101.26667 |
| ROBNO | C | 10.00000 | 97.08333 |
| UTHAI | C | 7.00000 | 99.50000 |
| DOBAX | C | 6.44167 | 100.07833 |
| TANEK | C | 14.05000 | 98.97500 |
| GOMES | C | 13.45000 | 101.63000 |
| BOKAK | C | 12.95833 | 102.50000 |
| PIMAN | C | 11.35500 | 110.75500 |
| MANGU | N | 11.86333 | 112.53500 |
| SEKSI | C | 12.26833 | 114.00000 |
| ALBOS | C | 14.75833 | 101.01667 |
| VTN | C | 18.00000 | 102.53333 |
| MAMAK | C | 8.93833 | 108.66000 |
| ADPIM | C | 10.16167 | 111.45000 |
| OLTOS | C | 11.24333 | 114.00000 |
| KAKET | C | 10.85000 | 102.60000 |
| LEMOK | N | 10.00000 | 103.03667 |
| SUKAR | C | 12.36667 | 110.89833 |
| LANVI | C | 10.00000 | 104.00500 |
| RASER | C | 10.00000 | 105.09333 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| TSN | C | 10.82333 | 106.66000 |
| MAARI | C | 5.93833 | 108.81333 |
| IDRUX | N | 8.37667 | 110.32667 |
| LAVEN | C | 12.36667 | 112.91000 |
| MARDA | C | 14.12167 | 114.00000 |
| KRT | C | 14.92667 | 102.12833 |
| UDN | C | 17.38333 | 102.77833 |
| KILOB | C | 16.21833 | 103.92500 |
| KKN | C | 16.47000 | 102.79167 |
| SKN | C | 17.26667 | 104.13333 |
| MST | C | 16.69833 | 98.54167 |
| TL | C | 15.26667 | 100.30000 |
| MHS | C | 19.31833 | 97.91000 |
| APRIL | N | 11.00000 | 99.72167 |
| STN | C | 9.13667 | 99.15167 |
| LPN | C | 18.28333 | 99.51667 |
| PR | C | 18.12667 | 100.16667 |
| TRN | C | 7.50833 | 99.63000 |
| PT | C | 6.78667 | 101.14833 |
| NTW | C | 6.52667 | 101.74500 |
| DELTA | N | 17.34167 | 100.93833 |
| LY | C | 17.45000 | 100.93833 |
| UTTAR | C | 17.71667 | 100.45500 |
| NOBER | C | 15.27833 | 100.65333 |
| TK | C | 16.88667 | 99.28000 |
| NS | C | 8.46167 | 99.96167 |
| HHN | C | 12.63500 | 99.95500 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| HOTEL | C | 13.00000 | 100.33333 |
| SM | C | 9.55000 | 100.06667 |
| RECNO | C | 8.57167 | 99.47667 |
| POPID | C | 6.48333 | 100.54000 |
| BUT | C | 12.69730 | 101.03150 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## ชื่อแสดงเส้นทางการบินของการควบคุมการจราจรทางอากาศ

| ชื่อเส้นทาง | | จาก | ถึง |
|---|---|---|---|
| A1 | S | LIMLA | OSUKA |
| A1 | S | OSUKA | BKK |
| A1 | S | BKK | SELKA |
| A1 | S | SELKA | UBL |
| A1 | S | UBL | BURTA |
| A1 | S | BURTA | PAPRA |
| A1 | S | PAPRA | DANANG |
| A1 | S | DANANG | CAVOI |
| A1 | S | CAVOI | DAGON |
| A1 | S | DAGON | CH |
| A576 | S | MEDAN | DAKIL |
| A576 | S | DAKIL | SALAK |
| A576 | S | SALAK | KASTA |
| A576 | S | KASTA | WESTPOINT |
| A576 | S | WESTPOINT | SIN |
| A901 | S | DANANG | IPROP |
| A901 | S | IPROP | IGNIS |
| A901 | S | IGNIS | IDOS |
| A901 | S | IDOS | CH |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| A327 | S | PUT | ASIDO |
| A334 | S | HTY | VKB |
| A334 | S | VKB | ADPAT |
| A349 | S | BKK | TAN |
| A464 | S | CMA | BEKOD |
| A464 | S | BEKOD | BKK |
| A464 | S | BKK | POLAK |
| A464 | S | POLAK | REGOS |
| A464 | S | REGOS | DIRAX |
| A464 | S | DIRAX | UPNEB |
| A464 | S | UPNEB | RELIP |
| A464 | S | RELIP | HTY |
| A464 | S | HTY | TAMOS |
| A464 | S | TAMOS | ALORSETAR |
| A464 | S | ALORSETAR | VBA |
| A464 | S | VBA | MALACCA |
| A464 | S | MALACCA | PASLA |
| A464 | S | PASLA | PONTIAN |
| A464AA | S | PONTIAN | SIN |
| A581 | S | PUKOK | OSADI |
| A581 | S | OSADI | CMA |
| A581 | S | CMA | BDRY |
| A581S | S | BOMAS | TATEL |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| A581S | S | TATEL | CMA |
| A583 | S | CH | NABOK |
| A583 | S | NABOK | ISBOK |
| A583 | S | ISBOK | REPAN |
| A583 | S | REPAN | ANPIK |
| A589 | S | PATKA | TAPON |
| A589 | S | TAPON | MOSLA |
| A589 | S | MOSLA | TETRA |
| A589 | S | TETRA | REBAK |
| A589 | S | REBAK | MERSING |
| B460 | S | UBL | PAK |
| B460 | S | PAK | VIPOP |
| B460 | S | VIPOP | NODAN |
| B460 | S | NODAN | VASKA |
| B460 | S | VASKA | BIVAK |
| B460 | S | BIVAK | TOTOX |
| B460 | S | TOTOX | MARDA |
| B460 | S | MARDA | BOPDO |
| B460 | S | BOPDO | ISBOK |
| B460 | S | ISBOK | LUBANG |
| B463 | S | EKAVO | MENEX |
| B463 | S | MENEK | UPNEB |
| B463 | S | UPNEB | SUWAN |
| B463 | S | SUWAN | VKB |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| B469 | S | BKK | POLAK |
| B469 | S | POLAK | REGOS |
| B469 | S | REGOS | IDAGA |
| B469 | S | IDAGA | TIKAL |
| B469 | S | TIKAL | NOLIK |
| B469 | S | NOLIK | VKB |
| B469 | S | VKB | PEKAN |
| B469 | S | PEKAN | PU |
| B469 | S | PU | MERSING |
| B469 | S | MERSING | SIN |
| | | | |
| B579 | S | PUT | TATOX |
| | | | |
| G463 | S | BETNO | BKK |
| G463 | S | BKK | RYN |
| G463 | S | RYN | KAKET |
| G463 | S | KAKET | LEMOX |
| G463 | S | LEMOX | ALGOR |
| G463 | S | ALGOR | KABAS |
| G463 | S | KABAS | MOSLA |
| G463 | S | MOSLA | BIBRO |
| | | | |
| G466 | S | PATKA | KANTO |
| G466 | S | KANTO | VKB |
| G466 | S | VKB | SERAU |
| G466 | S | SERAU | VBA |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| G468 | S | VKB | PENANG |
| G468 | S | PENANG | GUNIP |
| G468 | S | GUNIP | GOLTA |
| G468 | S | GOLTA | MEDAN |
| G472 | S | BETNO | BKK |
| G473 | S | MAKAS | PSL |
| G473 | S | PSL | EKIVI |
| G473 | S | EKIVI | UBL |
| G577 | S | MERSING | VIRUT |
| G577 | S | VIRUT | SINFIR1 |
| G577 | S | SINFIR1 | MEVAS |
| G577 | S | MEVAS | BIBRO |
| G577 | S | BIBRO | MAARI |
| G577 | S | MAARI | LUBANG |
| G579 | S | VBA | GEMAS |
| G579 | S | GEMAS | VJR |
| G579 | S | VJR | SIN |
| G580 | S | SIN | HOSBA |
| G580 | S | HOSBA | TOMAN |
| G582 | S | VBA | PEKAN |
| G582 | S | PEKAN | SINFIR3 |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| G582 | S | SINFIR3 | NOBUN |
| G582 | S | NOBUN | MEVAS |
| G582 | S | MEVAS | DODRO |
| G592 | S | BKK | RYN |
| G592 | S | RYN | REMAD |
| G592 | S | REMAD | KANTO |
| G592 | S | KANTO | ADPAT |
| G592 | S | ADPAT | RISTA |
| G592 | S | RISTA | NOBU |
| G592 | S | NOBUN | SINFIR2 |
| G592 | S | SINFIR2 | REBAK |
| G592 | S | REBAK | SIN |
| G647 | S | BKK | BATOK |
| G647 | S | BATOK | MENAM |
| R593 | S | MAARI | IDRUX |
| R593 | S | IDRUX | ADPIM |
| R593 | S | ADPIM | MANGU |
| R593 | S | MANGU | LAVEN |
| R593 | S | LAVEN | MARDA |
| R593 | S | MARDA | NOBOK |
| R201 | S | BKK | BUT |
| R201 | S | BUT | PARAM |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| R201 | S | PARAM | BELER |
| R201 | S | BELER | ADPAT |
| R201 | S | ADPAT | MAARI |
| | | | |
| R202 | S | KIBAG | DOVEL |
| R202 | S | DOVEL | MAARI |
| R202 | S | MAARI | DODRO |
| | | | |
| R203 | S | SAPAM | PUT |
| R203 | S | PUT | RELIP |
| R203 | S | RELIP | ADNEP |
| R203 | S | ADNEP | SUWAN |
| R203 | S | SUWAN | TIKAL |
| R203 | S | TIKAL | PARAM |
| R203 | S | PARAM | REMAD |
| R203 | S | REMAD | ALGOR |
| | | | |
| R207 | S | SISUK | CMA |
| R207 | S | CMA | NAN |
| R207 | S | NAN | ANBOK |
| | | | |
| R215 | S | CTR | NAN |
| R215 | S | NAN | TOMIP |
| | | | |
| R325 | S | ROBNO | PUT |
| R325 | S | PUT | UTHAI |
| R325 | S | UTHAI | DUBAK |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| R461 | S | MEDAN | PUKER |
| R461 | S | PUKER | VBA |
| | | | |
| R468 | S | TANEK | BKK |
| R468 | S | BKK | GOMES |
| R468 | S | GOMES | BOKAK |
| R468 | S | BOKAK | PNH |
| R468 | S | PNH | SAPEN |
| R468 | S | SAPEN | TSN |
| R468 | S | TSN | KIBAG |
| R468 | S | KIBAG | PIMAN |
| R468 | S | PIMAN | MANGU |
| R468 | S | MANGU | SEKSI |
| R468 | S | SEKSI | LUBANG |
| | | | |
| R473 | S | ALGOR | TAPON |
| R473 | S | TAPON | MAARI |
| | | | |
| R474 | S | BKK | ALBOS |
| R474 | S | ALBOS | EKIVI |
| R474 | S | EKIVI | VTN |
| | | | |
| R77 | S | PATKA | MAMAK |
| R77 | S | MAMAK | DOVEL |
| R77 | S | DOVEL | ADPIM |
| R77 | S | ADPIM | OLTOS |
| R77 | S | OLTOS | LUBANG |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| R85 | S | BKK | RYN |
| R85 | S | RYN | KAKET |
| R85 | S | KAKET | LEMOK |
| R85 | S | LEMOK | ALGOR |
| R85 | S | ALGOR | PATKA |
| R85 | S | PATKA | KIBAG |
| R85 | S | KIBAG | SUKAR |
| R85 | S | SUKAR | TOTOX |
| R85 | S | TOTOX | SUNEK |
| R85 | S | SUNEK | ISBAN |
| R85 | S | ISBAN | CH |
| R349 | S | LEMOK | LANVI |
| R349 | S | LANVI | RASER |
| R349 | S | RASER | TSN |
| R593 | S | MAARI | IDRUX |
| R593 | S | IDRUX | ADPIM |
| R593 | S | ADPIM | MANGU |
| R593 | S | MANGU | LAVEN |
| R593 | S | LAVEN | MARDA |
| R593 | S | MARDA | NOBOK |
| W1 | S | BKK | KRT |
| W1 | S | KRT | UBL |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| W2 | S | KRT | UDN |
| W3 | S | VTN | UDN |
| W4 | S | UDN | KILOB |
| W4 | S | KILOB | UBL |
| W5 | S | KKN | KILOB |
| W5 | S | KILOB | UBL |
| W6 | S | KRT | KKN |
| W6 | S | KKN | SKN |
| W7 | S | CMA | MST |
| W8 | S | TL | KRT |
| W9 | S | MHS | CMA |
| W9 | S | CMA | PSL |
| W9 | S | PSL | TL |
| W9 | S | TL | BKK |
| W10 | S | BKK | MENEK |
| W10 | S | MENEK | APRIL |
| W10 | S | APRIL | STN |
| W11 | S | KKN | UDN |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| W12 | S | CMA | NAN |
| W13 | S | LPN | PR |
| W14 | S | PUT | TRN |
| W14 | S | TRN | HTY |
| W14 | S | HTY | PT |
| W14 | S | PT | NTW |
| W15 | S | CMA | PR |
| W15 | S | PR | DELTA |
| W15 | S | DELTA | LY |
| W15 | S | LY | UDN |
| W15 | S | UDN | SKN |
| W16 | S | CMA | UTTAR |
| W16 | S | UTTAR | EKIVI |
| W16 | S | EKIVI | KKN |
| W17 | S | STN | HTY |
| W18 | S | PUT | STN |
| W19 | S | BKK | POLAK |
| W19 | S | POLAK | REGOS |
| W19 | S | REGOS | DIRAX |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| W19 | S | DIRAX | UPNEB |
| W19 | S | UPNEB | ADNEP |
| W19 | S | ADNEP | NTW |
| W20 | S | CMA | CTR |
| W21 | S | BKK | NOBER |
| W21 | S | NOBER | EKIVI |
| W22 | S | PSL | PR |
| W22 | S | PR | CTR |
| W23 | S | PSL | LPN |
| W24 | S | STN | TRN |
| W25 | S | PR | NAN |
| W26 | S | MST | TK |
| W26 | S | TK | PSL |
| W26 | S | PSL | EKIVI |
| W27 | S | PSL | LY |
| W28 | S | STN | NS |
| W29 | S | CTR | NAN |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

| | | | |
|---|---|---|---|
| W31 | S | HHN | HOTEL |
| W32 | S | SM | REGOS |
| W33 | S | PUT | RECNO |
| W650 | S | HTY | POPID |

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้