

CONTRIBUTION

โครงการ
เมาส์สั่งงานด้วยเสียงพูด
Speech Mouse

นายสมบัติ ทับทิมแดง B4208440

นางสาวฉลวยจิตร รุ่งเรือง B4200536

รายงานนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตรบัณฑิต

สาขาวิชาวิศวกรรมโทรคมนาคม คณะวิศวกรรมศาสตร์

มหาวิทยาลัยเทคโนโลยีสุรนารี

ปีการศึกษา 2545



ศูนย์บรรณสารและสื่อการศึกษา

มหาวิทยาลัยเทคโนโลยีสุรนารี

โครงการ	เมาส์สั่งงานด้วยเสียงพูด
ผู้ดำเนินการ	นายสมบัติ ทับทิมแดง นางสาวฉลวยจิตร รุ่งเรือง
อาจารย์ที่ปรึกษา	อ. พิระพงษ์ อุฑารสกุล
สาขาวิชา	วิศวกรรมโทรคมนาคม
ภาคการศึกษาที่	3/2545

บทคัดย่อ

ในปัจจุบันเทคโนโลยีต่างๆ ได้พัฒนาไปอย่างรวดเร็ว แต่เทคโนโลยีบางอย่างไม่ได้เอื้อประโยชน์แก่ผู้พิการ ทำให้การพัฒนาความรู้ความสามารถของผู้พิการไม่สะดวกมากนักโครงการนี้จึงมุ่งหวังที่จะช่วยเหลือผู้ที่มีความพิการทางแขนให้สามารถใช้งานเมาส์ได้ด้วยเสียงพูด ซึ่งได้นำเทคโนโลยีการรู้จำเสียงพูดมาพัฒนาเพื่อประโยชน์ในการใช้งานเมาส์ ซึ่งเมาส์คืออุปกรณ์ที่สำคัญในการใช้งานคอมพิวเตอร์ที่ขาดไม่ได้ โครงการเมาส์สั่งงานด้วยเสียงพูด เป็นการสั่งงานเมาส์ด้วยเสียงพูดผ่านไมโครโฟนโดยใช้โปรแกรมภาษา Visual Basic เป็นตัวพัฒนาและนำระบบ SAPI (Speech Application Programming Interface) เป็นตัวเชื่อมต่อระหว่างผู้ใช้งานกับระบบปฏิบัติการวินโดวส์สามารถแบ่งการทำงานของโปรแกรมได้ 3 ส่วนคือ Mouse Mode, Option Mode และ Text Mode

ในส่วนของ Mouse Mode จะเป็นการเคลื่อนที่ของเมาส์เคอร์เซอร์ ซึ่งจะเคลื่อนที่ขึ้น ลง ซ้าย ขวา และหยุดได้ตามที่สั่งงาน นอกจากนี้ยังมีการคลิกขวา คลิกซ้าย ดับเบิ้ลคลิก แดรกและครีอป ซึ่งการทำงานในโหมดนี้จะคล้ายกับการทำงานของเมาส์พื้นฐานทั่วไป ในส่วนของ Option Mode จะเป็นส่วนของการปรับความเร็วของการเคลื่อนที่ของเมาส์เคอร์เซอร์ซึ่งมี 3 ระดับคือช้า ปานกลาง และเร็ว ในส่วนของ Text Mode จะเป็นส่วนของการพิมพ์ตัวอักษรภาษาอังกฤษ และการเลื่อนหน้า ซึ่งในส่วนนี้จะเป็นการทำงานคล้ายการทำงานของคีย์บอร์ด

กิตติกรรมประกาศ

การทำโครงการนี้ ส่งผลให้คณะผู้จัดทำได้รับความรู้และประสบการณ์ต่างๆ ที่มีค่ามากมาย สำหรับโครงการนี้ สำเร็จลงได้ด้วยดีจากคำแนะนำ ความร่วมมือ ความช่วยเหลือและการสนับสนุนจากหลายฝ่ายดังนี้

1. อาจารย์ไพระพงษ์ อุซหารสกุล อาจารย์ที่ปรึกษาโครงการนี้ ท่านได้ให้ความช่วยเหลือในทุกด้าน
2. อาจารย์ประจำสาขาวิชาวิศวกรรมโทรคมนาคมทุกท่าน ที่ให้คำแนะนำที่เป็นประโยชน์แก่การทำโครงการนี้
3. คุณชนานนท์ พักจิ้น เพื่อนร่วมเรียนระดับปริญญาตรีที่ให้คำปรึกษาด้านการเขียนโปรแกรม
4. คุณรังสี น้อยเนลาและคุณพรธนิภา จันโทศรี เพื่อนร่วมเรียนระดับปริญญาตรี ที่ให้ความอนุเคราะห์คอมพิวเตอร์ในการทำโครงการ
5. เพื่อนนักศึกษาสาขาวิศวกรรมโทรคมนาคมทุกคน ที่ให้กำลังใจตลอดมา

ท้ายนี้ ขอกราบขอบพระคุณบิดา มารดา ที่ให้การเลี้ยงดูอบรมและส่งเสริมการศึกษา เป็นอย่างดีตลอดมา

สมบัติ ทับทิมแดง

ฉลวยจิตร รุ่งเรือง

สารบัญ

	หน้า
บทคัดย่อ	ก
กิตติกรรมประกาศ	ข
สารบัญ	ค
บทที่ 1 บทนำ	1-1
1.1 ความเป็นมา	1-1
1.2 วัตถุประสงค์ของโครงการ	1-2
1.3 ขอบเขตของโครงการ	1-2
1.4 ขั้นตอนการดำเนินงาน	1-3
บทที่ 2 ทฤษฎีที่เกี่ยวข้อง	2-1
2.1 เม้าส์	2-1
2.1.1 การทำงานของเม้าส์	2-2
2.1.2 ลักษณะการใช้เม้าส์	2-5
2.1.3 แนวโน้มการพัฒนาของเม้าส์	2-5
2.2 การใช้ Windows API ฟังก์ชันใน Visual Basic	2-8
2.2.1 ความรู้เบื้องต้นเกี่ยวกับ Windows API	2-8
2.2.2 การประกาศเพื่อใช้งาน API Function	2-10
2.2.3 ข้อควรรู้ในการใช้ API	2-11
2.3 Speech Technology	2-12
2.3.1 การรู้จำเสียงพูด (Speech recognition)	2-12
2.3.2 การสังเคราะห์เสียงพูด (Speech Synthesis)	2-18
2.4 โปรแกรมภาษา Visual Basic เพื่อการสั่งงานเม้าส์ด้วยเสียงพูด	2-19
2.4.1 การทำงานในส่วนของการเคลื่อนที่	2-19
2.4.2 การทำงานในส่วนของการรู้จำเสียงพูด	2-20
2.4.3 การทำงานในส่วนของการคลิกเม้าส์	2-23
2.4.4 การทำงานในส่วนของการ Text Mode	2-24

สารบัญ(ต่อ)

	หน้า
บทที่ 3 โปรแกรมเมาส์สั่งงานด้วยเสียงพูด	3-1
3.1 ขั้นตอนการ Install โปรแกรมเมาส์สั่งงานด้วยเสียงพูด	3-1
3.2 ส่วนประกอบของโปรแกรมเมาส์สั่งงานด้วยเสียงพูด	3-5
3.2.1 Mouse Mode	3-5
3.2.2 Option Mode	3-6
3.2.3 Text Mode	3-7
3.3 การทำงานของโปรแกรมเมาส์สั่งงานด้วยเสียงพูด	3-9
3.3.1 การเปิดโปรแกรมเมาส์สั่งงานด้วยเสียงพูด	3-9
3.3.2 เริ่มการใช้งานโปรแกรมเมาส์สั่งงานด้วยเสียงพูด	3-9
3.4 ซัดจำกัดและข้อควรระวังของโปรแกรมเมาส์สั่งงานด้วยเสียงพูด	3-20
บทที่ 4 บทสรุป	4-1
4.1 สรุปโดยรวมของโครงการเมาส์สั่งงานด้วยเสียงพูด	4-1
4.2 ประโยชน์ของโครงการเมาส์สั่งงานด้วยเสียงพูด	4-1
4.3 ปัญหาอุปสรรคและข้อเสนอแนะ	4-2
เอกสารอ้างอิง	จ

บทที่ 1

บทนำ

1.1 ความเป็นมา

สังคมปัจจุบันเป็นสังคมยุคสารสนเทศ เทคโนโลยีการติดต่อสื่อสารเป็นไปอย่างไร้พรมแดน แม้กระทั่งคนพิการก็สามารถดำรงชีวิตอยู่ได้ในสังคมยุคสารสนเทศนี้ด้วยการพัฒนาอุปกรณ์เครื่องมือที่ทันสมัยในการเข้าถึงเครือข่ายคอมพิวเตอร์หรือเครื่องมือสื่อสารพิเศษให้กับคนพิการแต่ละประเภท เช่น ในอดีตคนตาบอดสื่อสารกันด้วยเสียงพูดหรืออ่านหนังสือเพื่อเพิ่มพูนความรู้ด้วยหนังสือเสียงที่บันทึกเทปไว้เป็นเรื่องราว หรืออ่านหนังสือที่พิมพ์ด้วยอักษรเบรลล์ซึ่งมีความหนาและมียุทธศาสตร์เรื่องเนื้อที่ใช้ในการเก็บ ปัจจุบันเทคโนโลยีสารสนเทศก่อให้เกิดการพัฒนาเป็นทีย็บอร์ดคอมพิวเตอร์สำหรับคนตาบอดการสังเคราะห์เสียงพูดเพื่อช่วยอ่านหนังสือและการรู้จำตัวอักษรเพื่อการอ่าน ตัวอย่างของเทคโนโลยีสารสนเทศเหล่านี้ส่งผลให้คนตาบอดสามารถเรียนรู้ทุกเรื่องได้ง่าย สะดวก และรวดเร็ว รวมทั้งการป้อนข้อมูลเข้าคอมพิวเตอร์ และการแสดงผลผ่านคอมพิวเตอร์ อุปกรณ์เหล่านี้ จะช่วยให้ผู้ด้อยโอกาสได้รับสิทธิเท่าเทียมกับคนปกติในด้านการศึกษาและติดต่อสื่อสาร ซึ่งในที่สุดสังคมเราก็จะได้พลเมืองที่มีคุณภาพเพิ่มขึ้นเพื่อทำงานช่วยเหลือและพัฒนาประเทศชาติต่อไป บุคคลที่ร่างกายพิการสามารถพัฒนาคุณภาพชีวิตได้ด้วยเทคโนโลยีสารสนเทศเช่นเดียวกัน สำหรับที่คนพิการแขน ขา จะใช้คอมพิวเตอร์หรือจะสื่อสารกับบุคคลอื่น ๆ ได้ ต้องมีเทคโนโลยีเข้ามารองรับเพื่อให้สามารถใช้งานได้สะดวก

การรู้จำเสียงพูดเป็นวิธีการหนึ่งที่ช่วยให้ผู้ใช้เลือกวิธีการที่จะป้อนข้อมูลเสียงให้กับคอมพิวเตอร์โดยที่เครื่องจะรับคำสั่งจากเสียงพูดของผู้ใช้ เช่น คำสั่งการใช้งานเบื้องต้นของDOSหรือWindows ปัจจุบันซอฟต์แวร์นี้มีการพัฒนาให้สามารถรับเสียงได้โดยไม่จำกัดเพศและวัย ซึ่งเป็นประโยชน์กับผู้พิการแขนขาอย่างมากที่สามารถใช้เสียงพูดในการสั่งงานได้เลย โครงการนี้จึงเป็นการนำเทคโนโลยีนี้มาพัฒนาเพื่อประโยชน์ในการใช้งานเมาส์ ซึ่งเมาส์คืออุปกรณ์ที่สำคัญในการใช้งานคอมพิวเตอร์ที่ขาดไม่ได้

1.2 วัตถุประสงค์ของโครงการ

1. เพื่อนำโปรแกรมภาษา Visual Basic มาประยุกต์ใช้งานได้
2. เพื่อเขียนโปรแกรมประยุกต์ที่สั่งงานเมาส์ได้
3. เพื่อเขียนโปรแกรมประยุกต์ที่สั่งงานด้วยเสียงพูดได้
4. สร้างโปรแกรมประยุกต์ที่สั่งงานเมาส์ด้วยเสียงพูดได้ครบทุกการใช้งานของเมาส์ปกติอย่างสมบูรณ์
5. นำโปรแกรมประยุกต์ที่สร้างขึ้นนี้ไปใช้จริง โดยเฉพาะอย่างยิ่งเพื่อช่วยเหลือผู้พิการแขน
6. โปรแกรมที่สร้างขึ้นนี้สามารถประหยัดการใช้เมาส์ทำให้ประหยัดค่าใช้จ่ายได้ส่วนหนึ่ง

1.3 ขอบเขตของโครงการ

สำหรับการศึกษาโครงการนี้ มุ่งเน้นที่การเขียนโปรแกรมประยุกต์เพื่อสั่งงานเมาส์ผ่านเสียงพูดโดยใช้โปรแกรมภาษา Visual Basic เป็นตัวพัฒนาและนำระบบ SAPI (Speech Application Programming Interface) เป็นตัวเชื่อมต่อระหว่างผู้ใช้และระบบปฏิบัติการวินโดวส์เพื่อสั่งงานเมาส์ โดยโปรแกรมนี้สามารถสั่งเมาส์ด้วยเสียงพูดครบสมบูรณ์ทุกหน้าที่ทั้งขึ้น ลง ซ้าย ขวา คลิก คลิกขวาและดับเบิ้ลคลิก นอกจากนี้ยังมีฟังก์ชันพื้นฐานสำหรับการใช้งานอินเทอร์เน็ตได้บางส่วน เช่น Mode Text ที่สามารถพูดสั่งให้เขียนตัวอักษรภาษาอังกฤษได้ โปรแกรมนี้มุ่งหวังที่จะช่วยเหลือผู้ที่มีความพิการทางแขนได้สามารถใช้งานเมาส์ด้วยเสียงพูดได้ แต่ก็สามารถนำไปใช้ให้เป็นประโยชน์แก่ผู้ที่ต้องการความสะดวกสบายในการใช้งานเมาส์และสามารถลดค่าใช้จ่ายในการซื้อเมาส์ได้อีกด้วย

1.4 ขั้นตอนการดำเนินงาน

แบ่งขั้นตอนการดำเนินงานดังนี้

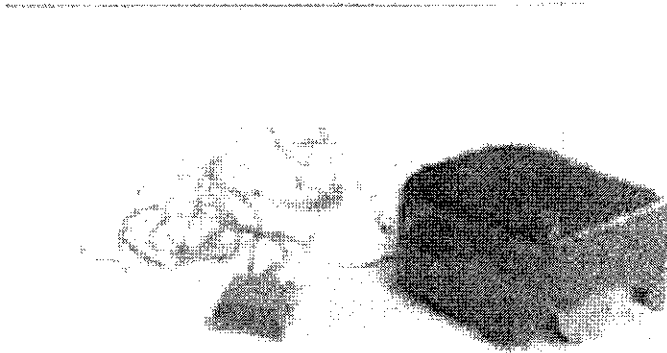
1. ศึกษาโปรแกรมภาษา Visual Basic และศึกษาการสั่งงานเมาส์ด้วยโปรแกรมภาษา Visual Basic
2. ศึกษาการรู้จำเสียงพูดบนระบบปฏิบัติการวินโดวส์
3. นำโปรแกรมภาษา Visual Basic มาประยุกต์ใช้กับการรู้จำเสียงพูด
4. สร้างโปรแกรมประยุกต์ที่สามารถสั่งงานเมาส์ด้วยเสียงพูด
5. ทดสอบการทำงานของโปรแกรมประยุกต์ที่สร้างขึ้น และปรับปรุงแก้ไข
6. สรุปผลการทำงาน

บทที่ 2

ทฤษฎีที่เกี่ยวข้อง

2.1 เมาส์ (MOUSE)

เมาส์เริ่มใช้งานกันอย่างแพร่หลายในช่วงประมาณ ปี ค.ศ.1990 ควบคู่กันมากับการเปิดตัวใช้งานระบบ ปฏิบัติการ Windows ซึ่งเป็นระบบปฏิบัติการแบบ Graphic User Interface (GUI) ทั้ง ๆ ที่เมาส์ ตัวแรกถูกประดิษฐ์คิดค้นมาตั้งแต่ช่วงประมาณปี ค.ศ. 1963 โดยนักวิจัยชื่อ Douglas C. Engelbart แห่งสถาบันวิจัยสแตนฟอร์ด (Stanford Research Institute: SRI) ซึ่งถือว่าเป็นบิดา ของเมาส์เมาส์ตัวแรกที่เขาประดิษฐ์ขึ้นจากวัสดุประเภทไม้มีปุ่ม 1 ปุ่ม และมี ล้อคู่ขนานสำหรับระบุ ตำแหน่งพิกัดบนจอคอมพิวเตอร์ดัง แสดงในรูปที่ 2.1

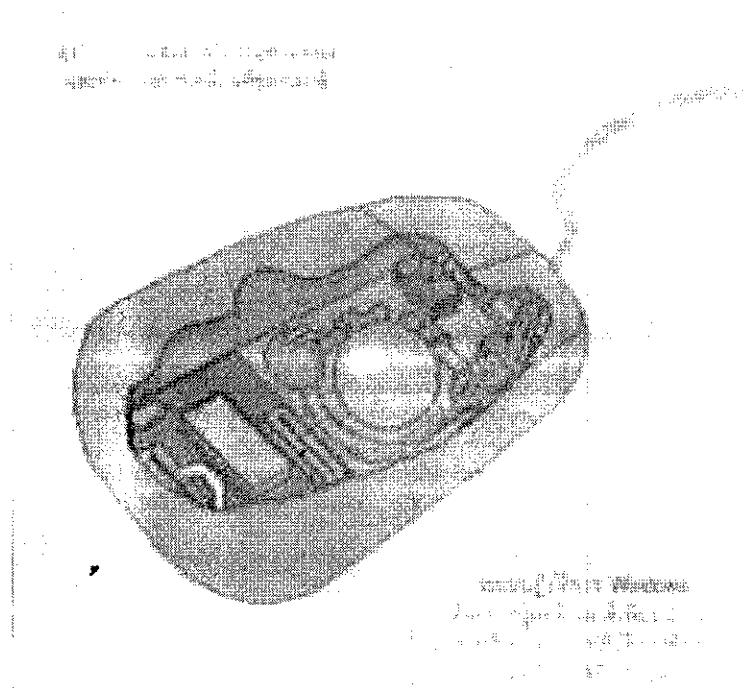


รูปที่ 2. 1 เมาส์ตัวแรกที่ผลิตโดย Douglas C. Engelbart

เมาส์ที่ใช้งานในปัจจุบันเป็นเมาส์แบบ Opto-Electronic ซึ่งรูปลักษณะภายนอกของเมาส์ก็จะมีปุ่มสำหรับคลิกตั้งแต่ 2 ปุ่มถึง 4 ปุ่ม และด้านล่างก็จะมีลูกบอลอย่างกลิ้งมากลิ้งไปได้ มีสายและหัวต่อกับพอร์ตคอมพิวเตอร์ซึ่งมีอยู่ด้วยกัน 2 แบบ คือหัวต่อแบบ PS/2 และหัวต่อแบบ DIN5 ข้อแตกต่างระหว่างหัวต่อทั้งสอง นอกจากจะเป็นรูปแบบการจัดวางขาจำนวนขาแล้ว PS/2 และ DIN5 จะใช้แรงดันไฟฟ้าและโปรโตคอลที่ต่างกันด้วยหัวต่อแบบ PS/2 จะใช้แรงดันไฟฟ้า 5 โวลต์ ใช้โปรโตคอลและการอินเตอร์เฟสของ IBM ส่วนหัวต่อแบบ PS/2 จะใช้แรงดันไฟฟ้า 12 โวลต์ ใช้โปรโตคอลและการอินเตอร์เฟสของ Microsoft

2.1.1 การทำงานของเมาส์

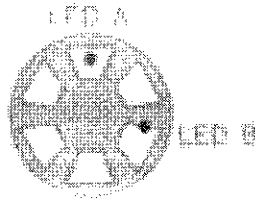
หากมีโอกาสได้ถอดเมาส์ออกเพื่อดูส่วนประกอบภายในแล้ว โดยทั่วไปแล้วก็จะเห็นลูกบอลทรงกลม ซึ่งทำมาจากลูกเหล็กแล้วหุ้มยางไว้ เพื่อให้ลูกบอลขยับมีน้ำหนักและติดหนึบกับแผ่นรองเมาส์ได้ดี นอกจากนั้นก็จะมีลูกกลิ้งทรงกระบอก 2 อัน วางในแนวตั้งฉากกัน สำหรับใช้ในการระบุทิศทางและตำแหน่งการเคลื่อนที่ในแนวซ้าย-ขวา บน-ล่าง (แกน X, Y) ที่ปลายของลูกกลิ้งทั้งสองจะมีแผ่นพลาสติกสีดำ ลักษณะเป็นแผ่นวงกลมแบนบางเจาะช่องสี่เหลี่ยมหลายช่องตามแนวรัศมีของวงกลม และมีอุปกรณ์ตรวจจับแสง(Photo Detector)ติดอยู่ อุปกรณ์ตรวจจับแสงประกอบด้วย LED (Light Emitting Diode) และอุปกรณ์รับแสง (Light Sensor) อย่างละ 2 ตัว นอกจากนั้นจะมีสวิทช์ของปุ่มคลิกเมาส์และแผ่นวงจรควบคุม(ไมโครคอนโทรลเลอร์) ซึ่งจะทำหน้าที่รับสัญญาณจากสวิทช์และอุปกรณ์รับแสงเพื่อแปลงเป็นสัญญาณข้อมูลสำหรับอินเตอร์เฟสกับคอมพิวเตอร์ดังรูปที่ 2.2



รูปที่ 2.2 องค์ประกอบภายในเมสส์

การทำงานของเมสส์ คือเมื่อมีการเลื่อนเมสส์ลูกบอลอย่างก็จะหมุนกลิ้งไปตามทิศทางของเมสส์ที่เลื่อนไป ซึ่งการหมุนของลูกบอลจะมีผลให้ลูกกลิ้งหมุนตามได้ด้วยในแนวแกน X, Y ซึ่งการหมุนของลูกกลิ้งก็จะส่งผลให้แผ่นพลาสติกกลมที่ปลายหมุนตามไปด้วยเช่นกัน การหมุนของแผ่นพลาสติกทำให้แสงจาก LED ที่อุปกรณ์รับแสงได้รับแตกต่างกันไปตามมุมมองของการหมุนตัวรับแสงก็จะส่งสัญญาณที่แตกต่างกันไปให้กับไมโครคอนโทรลเลอร์ ทำให้รู้ว่าขณะนี้มีการเคลื่อนตำแหน่งของเมสส์ ไปในทิศทางและระยะทางเท่าใด

ตัวอย่างเช่น หากมีการเจาะช่องบนแผ่นพลาสติกดำเพียง 6 ช่อง ทำมุม 60 องศาซึ่งกันและกัน ดังรูปที่ 2.2 (ในความเป็นจริงจะมีการเจาะช่องเอาไว้จำนวนมาก) จุดดำที่เห็นในรูปแทน LED ทั้ง 2 ตัว (LED A, LED B) ซึ่งอยู่ห่างกันเป็นมุม 105 องศา LED A จะอยู่ตรงกลางช่องพอดี ส่วน LED B จะอยู่บริเวณขอบของช่องที่เจาะไว้ LED แต่ละตัวจะมีอุปกรณ์รับแสงแยกกันคนละตัว เมื่อมีการหมุนพลาสติกนี้ อุปกรณ์รับแสงทั้งสองจะได้รับแสงสว่าง-มืด-สว่าง-มืด สลับกันไปตลอดเวลา ซึ่งก็อาศัยการเปลี่ยนสถานะ การรับแสงของอุปกรณ์รับแสงทั้งสองนี้ร่วมกันในการระบุว่ามีการเคลื่อนเมสส์ไปในทิศทางใด



รูปที่ 2.3 ตัวอย่างแผ่นพลาสติกที่ปลายของลูกกอล์ฟที่มีช่องอยู่เพียง 6 ช่อง

ทำมุม 60 องศาซึ่งกันและกัน

รูปที่ 2.3 เป็นการทดลองสถานการณ์เมื่อมีการหมุนแผ่นพลาสติกจาก A --> B --> C --> D --> E ซึ่งเป็นการหมุนแผ่นพลาสติกในทิศทางตามเข็มนาฬิกา ซึ่งสถานการณ์เปลี่ยนแปลงของ LED A และ B แต่ละขั้นตอนเมื่อมีการหมุนในทิศทางตามเข็มนาฬิกาและทวนเข็มนาฬิกาจะเป็นดังแสดงในตารางที่ 2. 1

ตารางที่ 2.1 แสดงสถานะของ LED เมื่อแผ่นพลาสติกหมุน

ในทิศทางตามเข็มนาฬิกาและทวนเข็มนาฬิกา

หมุนในทิศตามเข็มนาฬิกา	A --> B	B --> C	C --> D	D --> E
การเปลี่ยนสถานะของ LED A	สว่าง --> สว่าง	สว่าง --> มืด	มืด --> สว่าง	สว่าง --> สว่าง
การเปลี่ยนสถานะของ LED B	สว่าง --> มืด	มืด --> สว่าง	สว่าง --> สว่าง	สว่าง --> สว่าง
หมุนในทิศทวนเข็มนาฬิกา	E --> D	D --> C	C --> B	B --> A
การเปลี่ยนสถานะของ LED A	สว่าง --> มืด	สว่าง --> มืด	มืด --> สว่าง	สว่าง --> สว่าง
การเปลี่ยนสถานะของ LED B	สว่าง --> สว่าง	สว่าง --> สว่าง	สว่าง --> มืด	มืด --> สว่าง

ถ้าหากกำหนดให้มีการตรวจสอบสถานะของ LED A ทุกครั้งที่ LED B มีการเปลี่ยนสถานะจากสว่าง --> มีค(A --> B หรือ C --> D) จะพบว่าในการหมุนตามเข็มนาฬิกา(A --> B) LED A จะสว่างเสมอและในทิศทางทวนเข็มนาฬิกา (C -->B) LED A ก็จะมีคเสมอในลักษณะเช่นนี้เองจึงทำให้อุปกรณ์รับแสงสามารถส่งสัญญาณไปยังไมโครคอนโทรลเลอร์ เพื่อตีความได้ว่าเป็นการเคลื่อนเมาส์ไปทางขวาหรือซ้าย(แกน X) ขึ้นบนหรือลงล่าง (แกน Y) ส่วนระยะทางหรือพิคตที่เมาส์เคลื่อนที่จะทราบได้จากจำนวนครั้งที่เกิดการเปลี่ยนแปลงสถานะของ LED นั้นเอง

2.1.2 ลักษณะการใช้เมาส์

- **คลิก (Click)** คือ การกดเมาส์ปุ่มซ้ายหนึ่งครั้ง จะเสียงดังคลิกขึ้น แล้วปล่อยอย่างรวดเร็ว การคลิกมีจุดมุ่งหมายเพื่อเลือกคำสั่งหรือเลือกส่วนต่างๆ ใน โปรแกรม
- **ดับเบิลคลิก (Double Click)** คือ การกดเมาส์ปุ่มซ้ายติดกัน 2 ครั้งอย่างรวดเร็วใช้เพื่อเปิดโปรแกรมขึ้นมา
- **ลากแล้วปล่อย (Drag and Drop)** คือ การเคลื่อนตัวชี้เมาส์ไปที่ที่ส่วนใดส่วนหนึ่งของไอคอน หรือคำสั่งที่ต้องการแล้วกดเมาส์ปุ่มซ้ายค้างไว้ จากนั้นเลื่อนเมาส์ไปยังตำแหน่งที่ต้องการแล้วปล่อยเมาส์
- **คลิกเมาส์ปุ่มขวา (Right Click)** คือ การคลิกเมาส์ที่ปุ่มขวาหนึ่งครั้ง ใช้เพื่อเปิดเมนูย่อยขึ้นมา นิยมใช้ในการเปิดโปรแกรม Windows

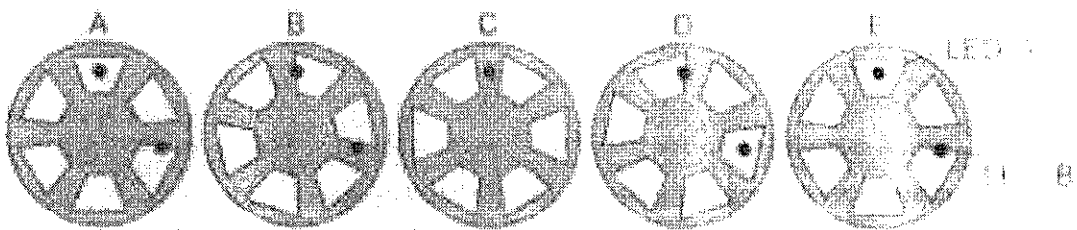
2.1.3 แนวโน้มการพัฒนาของเมาส์

คุณสมบัติของเมาส์ที่ดีและเหมาะสมที่จะเลือกใช้งาน จะต้องมีความสามารถในการเคลื่อนที่ได้อย่างว่องไวสามารถใช้งานได้อย่างคล่องตัวไม่ก่อให้เกิดความปวดเมื่อยจากการใช้งาน และไม่กินเนื้อที่บนโต๊ะทำงานมากเกินไป นักการพัฒนารูปแบบการทำงานของเมาส์จากผู้ผลิตส่วนใหญ่ก็จะเป็นไปเพื่อสนองตอบตามคุณสมบัติดังกล่าวตัวอย่างเช่น

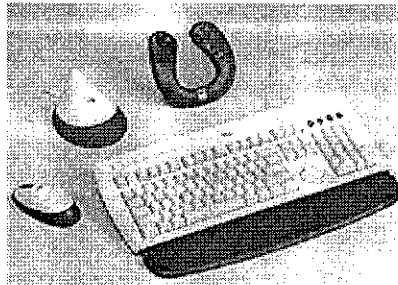
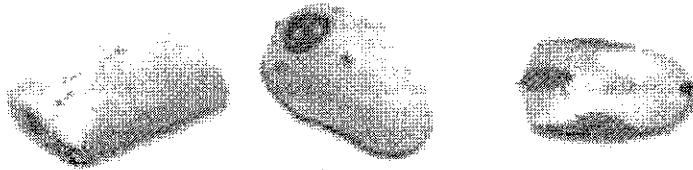
- **ออปติคอลเมาส์ (Optical Mouse)** เป็นเมาส์ที่เปลี่ยนจากการใช้ลูกบอลขยับในการตรวจสอบการเคลื่อนตำแหน่งไปตามพิคตต่างๆบนหน้าจอคอมพิวเตอร์ มาเปลี่ยนเป็นอุปกรณ์

รับแสง(Optical Sensor)โดยเมาส์ประเภทนี้ยังคงมีหลักการทำงานเหมือนกับเมาส์ที่ใช้ลูกบอลยางทุกประการ เพียงแต่ตัวลูกกลิ้งจะเชื่อมต่อกับอุปกรณ์รับแสงแทน ข้อดีของเมาส์ประเภทนี้คือทำให้ตัดปัญหาการเคลื่อนที่ที่ไม่คล่องตัวของเมาส์ อันเนื่องจากมีฝุ่นผงเกาะติดลูกบอลอย่างมากต้องถอดออกมาทำความสะอาดลูกบอลยางบ่อยๆ ดังนั้นเมาส์ประเภทนี้จึงสามารถเคลื่อนที่ได้อย่างสะดวก มีความถูกต้องของตำแหน่งที่ต้องการให้เคลื่อนที่ไปสูง และสามารถจะเคลื่อนเมาส์บนวัสดุอะไรก็ได้ ไม่จำเป็นจะต้องใช้แผ่นรองเมาส์

- **เมาส์ไร้สาย(Cordless Mouse)** การพัฒนาเมาส์ไร้สายในช่วงแรกจะอาศัยเทคโนโลยีของรังสีอินฟราเรด (Infrared Technology) แต่ต่อมาก็มีการพัฒนาไปใช้เทคโนโลยีคลื่นวิทยุ (Radio Technology) ทั้งนี้เนื่องจากการใช้เทคนิคอินฟราเรดจะมีข้อจำกัดในเรื่องการส่งรับสัญญาณ เครื่องส่งและเครื่องรับจะต้องอยู่ในทิศทางแนวเดียวกันและห้ามมีอุปกรณ์ใดๆขวางแนวการรับ-ส่งของคลื่นเด็ดขาดแต่สำหรับเมาส์ที่ใช้เทคโนโลยีความถี่คลื่นวิทยุนี้จะไม่มีปัญหาเรื่องดังกล่าวทำให้การใช้งานเมาส์มีความสะดวกมากขึ้นและใช้งานได้ในรัศมีที่กว้างขึ้น เนื่องจากคลื่นวิทยุสามารถทะลุผ่านอุปกรณกีดขวางใดๆได้ แต่จะต้องมีการติดตั้งเครื่องรับสัญญาณเพิ่มเติมที่เครื่องคอมพิวเตอร์ ข้อดีของเมาส์ประเภทนี้คือสามารถเลื่อนเมาส์ได้ในระยะที่อยู่ห่างจากคอมพิวเตอร์มากขึ้น เนื่องจากไม่มีสายเป็นตัวกำหนดขอบเขตการทำงานอีกต่อไป



รูปที่ 2.4 จำลองสถานการณ์เมื่อแผ่นพลาสติกหมุน



รูปที่ 2.5 หน้าตาเมาส์แบบต่างๆ

- **แทรคบอลเมาส์ (Trackball Mouse)** เป็นเมาส์ที่กลับเอาลูกบอลลงมายังด้านบนของเมาส์แทนที่จะเป็นด้านล่างเหมือนเมาส์ทั่วไป การใช้งานสามารถทำได้โดยการใช้นิ้วกดลูกบอลได้โดยตรงไม่จำเป็นต้องมีการเลื่อนเมาส์ไปมาบนแผ่นรองเมาส์ เพื่อหมุนลูกบอลไปตามทิศทางที่ต้องการเหมาะกับการใช้งานบนโต๊ะทำงานที่มีเนื้อที่ไม่มาก
- **เมาส์หลายมิติ(Cubic Mouse)** เป็นเมาส์ที่พัฒนาขึ้นเพื่อรองรับการทำงานหรือการเล่นเกมส์แบบเสมือนจริงโดยประยุกต์เอาหลักการทำงานของเมาส์แบบ 2 มิติ หรือเมาส์ที่ใช้งานทั่วไปแต่เพิ่มให้มีการระบุทิศทางเคลื่อนที่ในหลายมิติมากขึ้น (มีการเพิ่ม จำนวนลูกบอลสำหรับตรวจสอบแนวการเคลื่อนที่ของเมาส์ในทิศทางต่างๆมากขึ้น) เช่น 2 มิติ (4 ทิศทาง หรือ 4 องศา ความเป็นอิสระ: 4 Degree of Freedom) 3 มิติ (6 ทิศทาง หรือ 6 องศา ความเป็นอิสระ : 6 Degree of Freedom)

- เม้าส์สั่งงานด้วยเสียงพูด(Speech Mouse) สำหรับในโครงการนี้เลือกการนำเทคโนโลยีตัวใหม่เข้ามาเพื่อสั่งงานเม้าส์ ซึ่งจะให้เม้าส์ทำงานได้ทุกหน้าที่โดยการสั่งงานจากเสียงพูด ซึ่งมีข้อดีคือไม่ต้องเกี่ยวข้องกับฮาร์ดแวร์เลย ทำให้ประหยัดงบประมาณในการซื้อเม้าส์ นอกจากนี้ยังเพิ่มคุณภาพชีวิตให้แก่ผู้พิการทางแขนด้วย

อุปกรณ์ที่เป็นอินพุตนั้นเป็นส่วนของการติดต่อโดยตรงระหว่างคนกับเครื่องคอมพิวเตอร์ผ่าน BIOS แนวโน้มการพัฒนาต่อไปในอนาคตก็จะเป็นไปเพื่อให้การปฏิสัมพันธ์ระหว่างกันเป็นไปได้โดยง่าย สะดวกและปลอดภัย การติดต่อระหว่างอุปกรณ์อินพุตกับ BIOS ของเครื่องคอมพิวเตอร์จะเป็นอินเตอร์เฟซสัญญาณในรูปแบบของสัญญาณแบบดิจิทัลมากขึ้น แทนที่สัญญาณแบบแอนะล็อกและหลักการทำงานของอุปกรณ์เหล่านี้ก็จะซับซ้อนมากขึ้น เพื่อให้รองรับต่อการใช้งานแบบเสมือนจริง (Virtual Reality)

2.2 การใช้ Windows API ฟังก์ชันใน Visual Basic

2.2.1 ความรู้เบื้องต้นเกี่ยวกับ Windows API

- API คืออะไร

การเขียนโปรแกรมต่างๆด้วยภาษาอะไรก็ตามซึ่งทำงานอยู่บน OS (ระบบปฏิบัติการ) อย่างเช่น MS-Windows เป็นต้น การที่จะติดต่อกับ OS ที่ว่านี้เพื่อสั่งงานใดๆ จะต้องผ่านตัวกลางที่รวบรวมเอาฟังก์ชันการทำงานต่างๆของ OS เอาไว้ ซึ่งก็คือ API นั่นเอง API ย่อมาจาก Application Programming Interface จะว่าไปแล้วหน้าที่ของ API ก็คือเป็นตัวกลางสำหรับเชื่อมโยงระหว่าง Application หรือโปรแกรมต่างๆ กับ OS นั่นเอง เคยมีผู้ยกตัวอย่างเปรียบเทียบ API กับ เครื่องคิดเลขไว้น่าสนใจคือ ปุ่มกดบนเครื่องคิดเลขก็คือตัวเชื่อมต่อ (Interface) ของเครื่องคิดเลข ส่วน API ก็คือ Interface ของ OS นั่นเอง

API จึงเป็นสิ่งสำคัญที่ทำให้ OS มีความแตกต่างกัน ยกตัวอย่างชัดๆก็คือ ซอฟต์แวร์ที่ทำงานบนเครื่อง Macintosh จะไม่สามารถทำงานบน Windows 98 ได้ ก็เพราะว่า Mac OS กับ Windows นั้นต่างก็มี API ที่แตกต่างกัน

- **การเรียกใช้ API Function**

ในทุกๆครั้งที่เขียนโปรแกรมและสั่งให้ทำงานนั้น ก็เป็นการเรียกใช้ API Function อยู่แล้ว แต่เป็นฟังก์ชันพื้นฐานต่างๆ ไป เช่น การแสดงผลบนจอภาพ การติดต่อกับคีย์บอร์ดหรือหน่วยความจำ เป็นต้น ซึ่งเวลาเขียนโปรแกรม โปรแกรมจะไปติดต่อกับAPIอัตโนมัติโดยไม่ต้องไปประกาศขอใช้งานแต่อย่างใด แต่หากว่าต้องการเขียนโปรแกรมที่ใช้ Function ที่นอกเหนือไปจากนั้น (High Level Programming) ก็คงต้องมีการประกาศการเรียกใช้งานต่างๆเพิ่มเติมเข้ามา

เกือบทุกๆฟังก์ชันของ API มักจะถูกบรรจุอยู่ใน DLL files (Dynamic Link Library) ในไดเรกทอรี C:\Windows\System (หรืออาจจะเป็นไดเรกทอรีอื่นที่เก็บ System ของ Windows ไว้) ซึ่งไฟล์เหล่านี้ก็จะบรรจุไปด้วย API Function เอาไว้และอนุญาตให้โปรแกรมภายนอกเข้ามาเรียกใช้ ซึ่งต้องเป็นโปรแกรมที่ทำงานบน Windows เท่านั้น (Windows Base Program)

ไฟล์ที่เก็บเอาฟังก์ชันของ API ส่วนใหญ่เอาไว้ก็คือไฟล์ดังต่อไปนี้

- User32.dll (user interface functions)
- kernel32.dll (operating system kernel functions)
- gdi32.dll (graphics device interface functions)
- shell32.dll (Windows shell functions)

- **ส่วนประกอบของ Windows API**

API ฟังก์ชันมีส่วนประกอบหลายส่วน แม้ว่าจะไม่ค่อยได้ใช้ทุกๆ ส่วนของ API ก็ตาม ซึ่งมีส่วนประกอบดังนี้

- Functions ส่วนนี้ก็จะประกอบไปด้วยฟังก์ชันต่างๆ ของAPI
- Structures คือส่วนที่รวบรวมเอาตัวแปรแบบต่างๆ ที่ใช้งานบ่อยๆ เอาไว้
- Named Constants ส่วนนี้จะเป็นการใช้งานตัวแปรบางค่าที่ตั้งขึ้นแทน รหัสหรือโค้ดที่จดจำยากๆ
- CallBack Functions เป็นส่วนกลับของ API Function ที่จะย้อนค่ากลับคืนมา
- Messages คือข้อความหรือข่าวสารที่ส่งไปยังวัตถุเพื่อบอกหรือสั่งให้ทำงานตามที่สั่ง

2.2.2 การประกาศเพื่อใช้งาน API Function

- การประกาศขอใช้งาน (Declaring the Function)

ก่อนที่เราจะใช้งาน API Function ได้นั้นต้องประกาศขอใช้งานก่อน เพื่อบอกให้ Visual Basic รู้ว่าจะหา API Function ได้จากที่ไหน ชื่อ Function ชื่ออะไร มีตัวแปรอะไรบ้างเป็นต้น เหล่านี้ที่นำมาเป็นข้อมูลที่ต้องบอกให้ Visual Basic ทราบ ซึ่งมีรูปแบบในการประกาศดังนี้

ส่วนการประกาศนั้น สามารถที่จะเขียนส่วนประกาศได้ 2 ที่ คือ ที่ Form และที่ Module ซึ่งหากเราประกาศไว้ที่ Form ก็จะทำให้สามารถใช้ Function ได้เฉพาะ Form ที่ประกาศไว้เท่านั้น แต่ถ้าหากประกาศไว้ที่ Module ก็จะสามารถใช้ได้กับทุกๆ Form (แบบ Public แต่ถ้าประกาศเป็นแบบ Private ก็ใช้ได้เฉพาะใน Module ที่ประกาศเท่านั้นเหมือนกับ Form)

- รูปแบบในการประกาศ (พิมพ์ต่อกันเป็นบรรทัดเดียว)

```
[{Public | Private}] Declare Function function_name Lib "DLL_filename" [Alias "function_alias"]
(argument_list ) As data_type
```

function_name

ส่วนนี้คือชื่อของ API Function เป็นชื่อที่ Visual Basic จะใช้อ้างอิง เวลาเราเขียนโค้ด ในทางทฤษฎีแล้ว สามารถตั้งชื่อเป็นอะไรก็ได้ แต่ก็ควรใช้ชื่อที่เค้าแนะนำดีกว่า

DLL_filename

ส่วนนี้คือชื่อไฟล์ที่เก็บ API Function เอาไว้ (นามสกุล DLL) ไม่ต้องใส่ Path ใส่เฉพาะชื่อไฟล์เพราะไฟล์พวกนี้อยู่ใน Folder System ของ Windows อยู่แล้วเช่น user32.dll

function_alias (Optional)

เป็นชื่อ Function ที่อยู่ใน DLL ไฟล์อีกที่ ส่วนนี้ไม่ใส่ก็ได้ เอาไว้ในกรณีที่มีไฟล์นั้นมีค่าพารามิเตอร์อยู่มากกว่า 1 พารามิเตอร์

argument_list

เป็นรายการของตัวแปรที่จะรับค่าต่างๆ ส่งผ่านไปให้กับฟังก์ชัน

data_type

ส่วนนี้คือประเภทของข้อมูลที่ Function จะคืนค่าออกมา ซึ่งจะจำกัดให้ใช้ได้เฉพาะ 4 ประเภทดังนี้

Byte An 8-bit integer.

Integer A 16-bit integer.

Long A 32-bit integer.

String A variable-length string.

ตัวอย่างการประกาศ

```
Public Declare Function ShellAbout Lib "shell32.dll" Alias "ShellAboutA" (ByVal hwnd As Long,
ByVal szApp As String, ByVal szOtherStuff As String, ByVal hIcon As Long) As Long
```

2.2.3 ข้อควรรู้ในการใช้ API

1. Save โปรแกรมที่เขียนใน VB ก่อนรัน โปรแกรม เสมอๆ
2. พารามิเตอร์ที่ยังไม่มีค่าสงวน By Val อยู่ข้างหน้า โดยเฉพาะสตริงควรตรวจสอบก่อน
3. ให้พิมพ์ข้อความ Option Explicit ในส่วนของทุกๆ โมดูลเพื่อป้องกันตัวแปร variant
4. ถ้าเกิด GPF (General Protection Fault) ขณะทดสอบ โปรแกรมอยู่ให้ Restart PC ใหม่
5. ถ้าใน VB ฟังก์ชันคล้ายๆ API ก็ควรหนีไปใช้ของ VB จะดีกว่า
6. ไม่ควรที่จะใช้ประกาศข้อมูลประเภท As Any ซึ่งจะส่งผลให้ข้อมูลผิดพลาด มีผลให้เกิด GPE
7. ข้อมูลใน VB ประเภทสตริงไม่สามารถใช้กำหนดพารามิเตอร์ใน API ได้ แต่สามารถใช้คำว่า By Val ไว้ข้างหน้าก่อน

8. ถ้าพารามิเตอร์เป็นสตริงขนาดของสตริงที่กำหนดให้ในตัวแปรต้องมากกว่าขนาดสูงสุดที่ฟังก์ชัน ให้อยู่ 1 Byte ของแอสกี 0 ที่ต่อท้ายสตริง
9. ต้องสร้างโมดูลไฟล์ไว้ใน VB
10. การประกาศฟังก์ชันตอนแรกสามารถทำได้ทั้งแบบ Public และ Private

2.3 Speech Technology

เทคโนโลยีเกี่ยวกับเสียงพูดมีดังนี้คือ การรู้จำเสียงพูด (Speech Recognition) และการสังเคราะห์เสียงพูด (Speech Synthesis)

2.3.1 การรู้จำเสียงพูด (Speech recognition)

การรู้จำเสียงพูด (Speech recognition) เป็นเทคโนโลยีที่เปลี่ยนเสียงพูด ให้กลายเป็นคำสั่ง ในรูปแบบของ ข้อความที่เครื่องคอมพิวเตอร์สามารถเข้าใจ และทำตามคำสั่งได้ อย่างเช่นการที่เราเคยโทรออกด้วยเสียงในโทรศัพท์มือถือ หรืออย่างการพิมพ์ข้อความโดยอัตโนมัติ ตามข้อความที่เราอ่านป้อนเข้าไปในคอมพิวเตอร์

เป็นเวลาหลายปีเลยทีเดียวยี่ระบบการรับทราบคำสั่งด้วยคำพูดของคอมพิวเตอร์ ไม่พัฒนาเท่าที่ควรเพียงแค่ประมาณสามปีที่ผ่านมานี้ ราคาก็ยังแพง, คุณภาพก็ไม่ดี และยุ่งยากในการใช้งาน แต่ตอนนี้ทุกอย่างเปลี่ยนแปลงไปแล้ว ด้วยเครื่องพีซีความเร็วสูง และซอฟต์แวร์รุ่นใหม่ี่พัฒนามาเพื่อ การนี้โดยเฉพาะ ได้ช่วยให้เราสามารถใช้ประโยชน์จากเทคโนโลยีนี้ได้เต็มที่ ที่สำคัญกว่านั้นมันยังสามารถนำไปใช้งานได้หลากหลาย ไม่ว่าจะเป็นการโทรศัพท์หรือแม้แต่ การส่งอีเมลล์ด้วยคำพูด เทคโนโลยีใหม่ี่ก็สามารถตอบสนองความต้องการได้

- **การทำงานของกรรู้จำเสียงพูด**

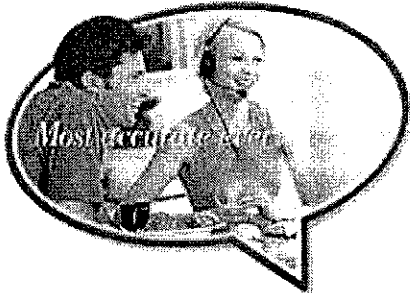
เครื่องคอมพิวเตอร์นั้นไม่ได้ใช้ภาษาเดียวกันดังนั้น การที่จะให้คอมพิวเตอร์สามารถแปลงคำพูดของเราให้กลายเป็นภาษาที่คอมพิวเตอร์สามารถเข้าใจได้นั้น จำเป็นจะต้องใช้ไมโครโฟนที่จะช่วยทำการเปลี่ยนเสียงพูด ให้กลายเป็นสัญญาณอนาล็อกเพื่อทำการถ่ายโอนเข้าไปในฮาร์ดแวร์ของเครื่องพีซีอีกครั้งหนึ่ง หลังจากนั้นตัวแปลงสัญญาณเสียงอีกตัวหนึ่ง ก็จะทำการแปลงสัญญาณจากอนาล็อกให้กลายเป็นดิจิตอลเพื่อส่งเป็นข้อมูลดิจิตอลเข้าไปในซอฟต์แวร์

- ส่วนประกอบสำคัญของเทคโนโลยีการรู้จำเสียงพูดมี 2 อย่าง คือ

1. แบบจำลองเสียง (acoustic model) ซึ่งทำการวิเคราะห์เสียงและทำการแยกเสียงออกเป็น ส่วนประกอบที่เล็กที่สุดของเสียงในภาษาพูด (dphoneme) โดยในส่วนของภาษาอังกฤษ ซึ่งเป็นภาษา สากลของโลกนั้นมีส่วนประกอบที่เล็กที่สุดของเสียงในภาษาพูด 50 phoneme

วิธีการที่คอมพิวเตอร์สามารถแยกเสียงออก ขึ้นแรกโมเดลเสียงจะทำการขจัดเสียงรบกวน และข้อมูลที่ไม่สำคัญออกไป ต่อจากนั้นก็จะใช้กระบวนการคำนวณทางคณิตศาสตร์ และทำการแปลง ข้อมูลที่ได้ให้กลายเป็นคลื่นความถี่ของเสียง หลังจากนั้นก็ทำการวิเคราะห์ข้อมูลที่ได้และเปลี่ยนคำพูด ทั้งหมดให้เป็นสัญลักษณ์ดิจิทัล ที่ใช้แทนค่าส่วนประกอบที่เล็กที่สุดของเสียงในภาษาพูด (phoneme) แต่ละตัวเหล่านั้น

ตัวอย่าง ประโยคต่อไปนี้ ซึ่งถูกแยกย่อย ออกไปเป็นส่วนประกอบที่เล็กที่สุดของเสียงใน ภาษาพูด (phoneme)

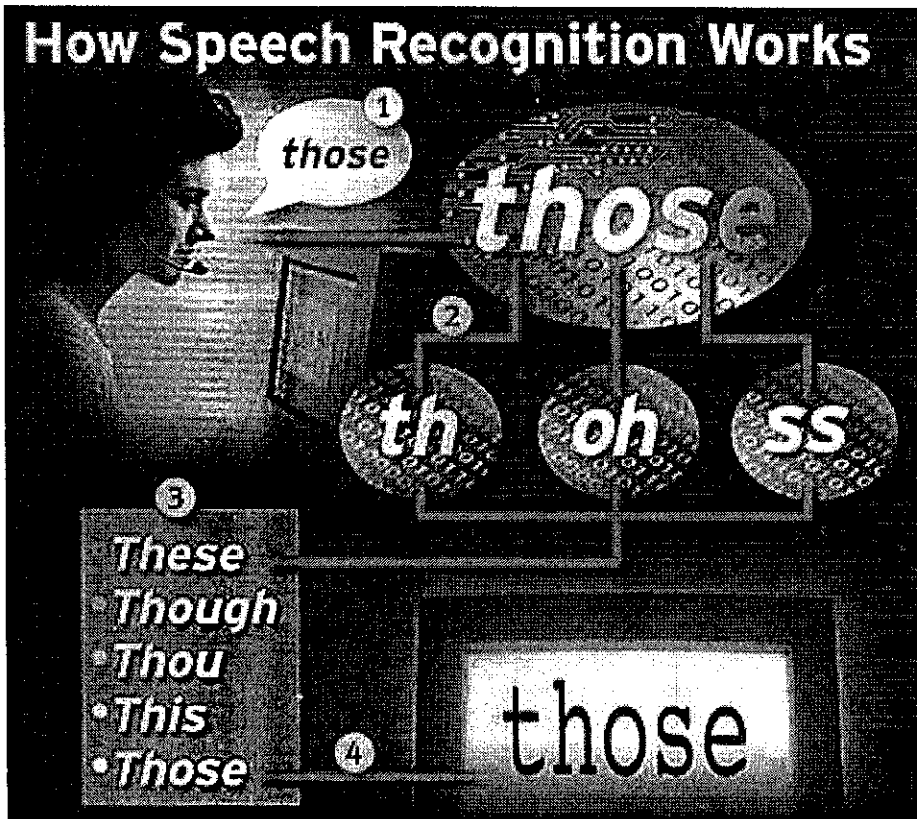


The quick brown fox jumps over the lazy dog.
 สัญลักษณ์ของส่วนประกอบที่เล็กที่สุดของเสียงในภาษาพูด (phoneme) ที่ได้ก็คือ **th Gkwith brawn faks juhmps ovR th lazy dawg**

2. แบบจำลองภาษา (Language Model) ซึ่งถือเป็นการทำงานสำคัญส่วนที่สองของโปรแกรม ซึ่งเป็นส่วนประกอบที่ทำหน้าที่ ในการแปลงข้อความให้กลายเป็นภาษาที่คอมพิวเตอร์เข้าใจ จะทำการเปรียบเทียบ การต่อเชื่อมของส่วนประกอบที่เล็กที่สุดของเสียงในภาษาพูด (phoneme) ในแต่ละคำ จากดิกชันนารีดิจิทัล ซึ่งเป็นฐานข้อมูลที่รวบรวมคำศัพท์ภาษาอังกฤษไว้มากมาย ในปัจจุบันนี้ โดยมากแล้วอุปกรณ์ชุดนี้ จะมาพร้อมกับดิกชันนารีดิจิทัลที่มีคำศัพท์ถึงกว่า 150,000 คำ โดยอุปกรณ์ แบบจำลองภาษานี้ จะทำการวิเคราะห์อย่างรวดเร็วว่าเสียงพูดที่เข้ามานั้น ตรงกับศัพท์คำไหนและ แสดงผลออกมาทางหน้าจอคอมพิวเตอร์

อย่างไรก็ตาม เนื่องจากความซับซ้อนของภาษาอังกฤษที่มีคำศัพท์จำนวนมากที่มีเสียงพ้องกัน อาทิ คำว่า there (ที่นั่น) กับ their (ของพวกเขา)ซึ่งออกเสียงเหมือนกันแต่มีความหมายต่างกันซึ่งอาจจะทำให้ซอฟต์แวร์นี้ไม่สามารถแปลความหมายได้อย่างถูกต้องเกี่ยวกับปัญหานี้จึง ได้มีการใช้เทคโนโลยี ที่เรียกว่า trigram เข้ามาช่วยโดยเทคโนโลยีนี้จะช่วยในการวิเคราะห์โครงสร้างของประโยคเข้า

ประกอบด้วยอีกทางหนึ่งทำให้เกิดความแม่นยำมากขึ้นในการทำงานของโปรแกรม อาทิ หากมีประโยคที่พูดว่า Let's go there (ไปที่นั่นกันเถอะ) โปรแกรมนี้จะสามารถรับทราบได้ทันทีว่า เป็นคำว่า there ไม่ใช่ their เพราะว่าจากโครงสร้างของประโยคนั้น จะต้องใช้คำว่า there ไม่ใช่ their ที่พิเศษกว่านั้น ก็คือ ซอฟต์แวร์นี้ยังสามารถที่จะปรับตัวเข้ากับเสียงของบุคคลใดบุคคลหนึ่งเป็นการเฉพาะตัวด้วย เพราะเสียงพูดของแต่ละคนอาจแตกต่างกันไปบ้าง โดยโปรแกรมของอุปกรณ์นี้จะสามารถวิเคราะห์ได้ทั้งเสียง, การเน้นคำ, รูปแบบการพูด และจดจำไว้เพื่อวิเคราะห์ออกมาได้อย่างถูกต้องด้วย นอกจากนี้ หากคุณชอบใช้ศัพท์ใหม่ๆ ที่ในดิกชันนารีจิดิตอลไม่มี โปรแกรมนี้ยังมี ส่วนเสริมพิเศษ สำหรับใส่คำศัพท์ใหม่ๆ เพื่อเก็บไว้ใช้งานในครั้งต่อไปด้วย



รูปที่ 2.6 แสดงแบบจำลองการทำงานของการรู้จำเสียงพูด(Speech Recognition)

❶ โปรแกรม Speech Recognition จะเปลี่ยนเสียงพูดให้กลายเป็นสัญญาณคลื่นอนาล็อก โดยผ่านไมโครโฟน จากนั้นสัญญาณเสียงจะถูกเปลี่ยนรูปเป็นสัญญาณดิจิทัล โดยผ่านซาว์นการ์ดเพื่อให้คอมพิวเตอร์เข้าใจในรูปแบบของสัญญาณดิจิทัล

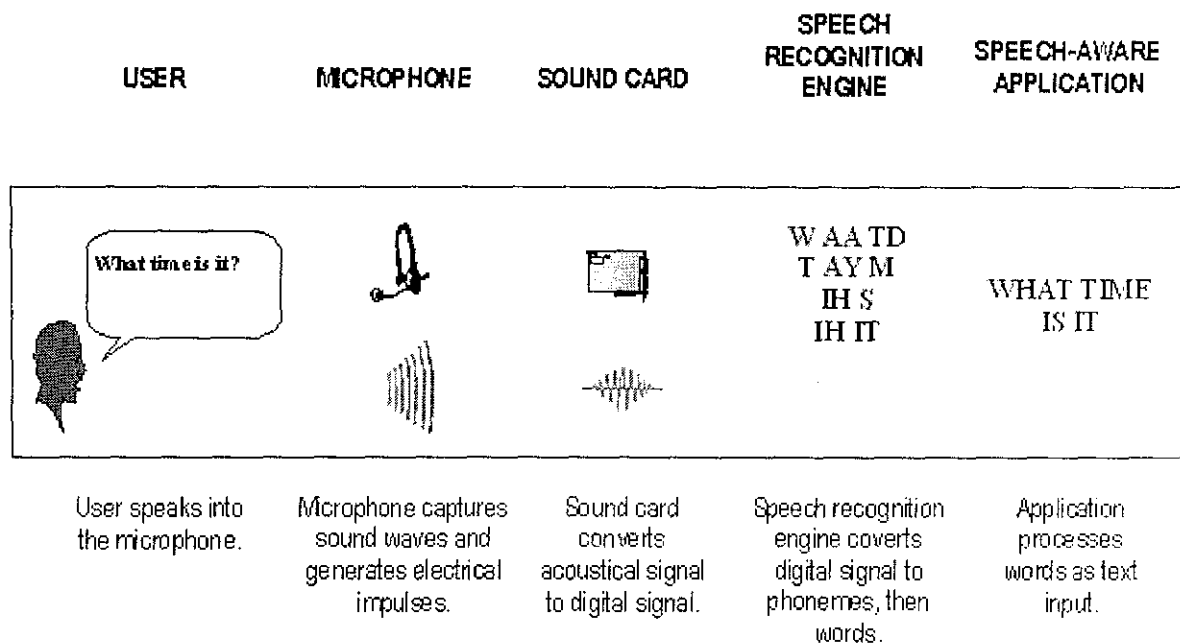
❷ ด้วยซอฟต์แวร์แบบจำลองเสียง(Acoustic) model จะวิเคราะห์สัญญาณที่เสียงที่อยู่ในรูปแบบดิจิทัลนั้น เพื่อแตกออกเป็นส่วนประกอบที่เล็กที่สุดของเสียงในภาษาพูด (phoneme)

❸ ถัดจากนั้นด้วยซอฟต์แวร์ แบบจำลองภาษา(Language model) จะวิเคราะห์ส่วนประกอบที่เล็กที่สุดของเสียงในภาษาพูด (phoneme) โดยเปรียบเทียบคำจากคิกชันนารีดิจิทัล

❹ สุดท้าย ซอฟต์แวร์จะตัดสินใจว่าคำที่ได้จากการเปรียบเทียบนั้นคือคำใด แล้วแสดงผลออกมาทางจอภาพ

สามารถแสดงได้ด้วยแผนภูมิดังต่อไปนี้

ผิดพลาด!

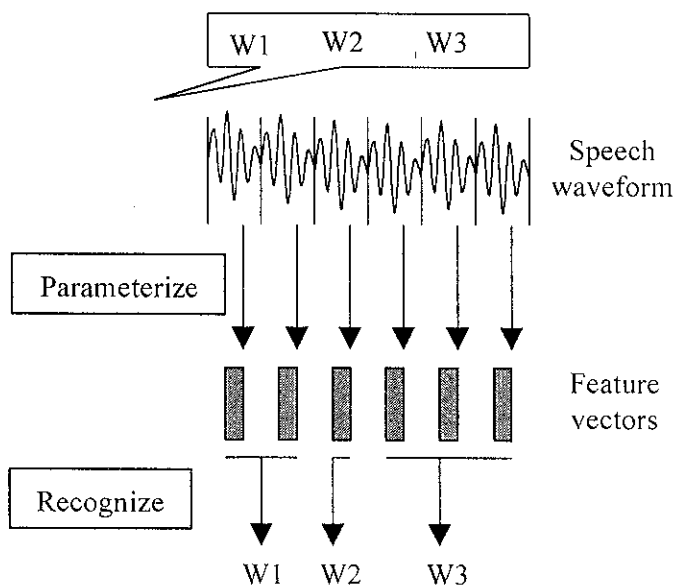


รูปที่ 2.7 แสดงกระบวนการในการรู้จำเสียง

จากรูปที่ 2.7 กระบวนการที่ Speech Recognition Engine ใช้ในการวิเคราะห์เสียงเพื่อแตกออกเป็น phoneme นั้นมีดังนี้คือ นำสัญญาณดิจิทัลที่มาจาก sound card มาทำ Fast Fourier Transform ทุกๆ 0.1 วินาที เพื่อลดขนาดของข้อมูลและง่ายต่อการนำไปวิเคราะห์ในขั้นตอนต่อไป นั่นก็คือ การใช้ทฤษฎี Hidden Markov Model ซึ่งมีรายละเอียดของ Hidden Markov Model ดังนี้

- **แบบจำลองฮิดเดินมาร์คอฟ**

Young (1997) ได้กล่าวว่า คำพูดเสียงเป็นสิ่งที่แสดงถึงคำหนึ่งคำหรือหลายๆคำที่มีลักษณะการเปลี่ยนแปลงของสัญญาณความถี่ (spectral envelop) ที่ซ้ำ มนุษย์ได้ยินสัญญาณความถี่ นี้และเปลี่ยนมันเป็นคำและสามารถเข้าใจถึงความหมายได้ จุดประสงค์ของการประมวลเสียงคำพูดคือการลอกเลียนระบบดังกล่าวของมนุษย์เพื่อที่จะสร้างเครื่องมือสำหรับพูดคุยกับมนุษย์เองได้



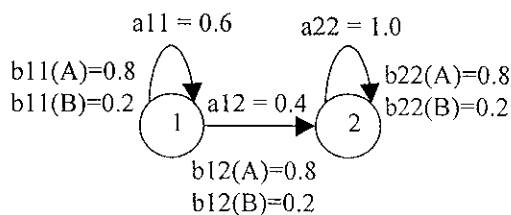
รูปที่ 2.8 กระบวนการของการรู้จำเสียงแบบอัตโนมัติ

ก้าวแรกของการเริ่มต้นระบบนี้คือการแปลงคำพูดที่ได้ยินหรือสัญญาณเสียงให้เป็นเครื่องหมายที่เหมาะสม เครื่องหมายที่เหมาะสมในที่นี้คือลักษณะพิเศษที่สามารถหาได้จากสัญญาณเสียงเช่น ความถี่มูลหลัก(Fundamental frequency) สัญญาณความถี่ (spectral envelop) เป็นต้น วิธีการหนึ่งที่น่านำมาใช้และให้ผลดีที่สุดในขณะนี้ในการแปลงคำพูดให้เป็นเครื่องหมายคือการใช้วิธีการสถิติของแบบจำลองฮิดเดินมาร์คอฟ

- ความหมายของ แบบจำลองฮิดเดินมาร์คอฟ

แบบจำลองฮิดเดินมาร์คอฟ สามารถโมเดลคำหรือหน่วยเสียงก็ได้ แต่เหตุผลของการที่จะ โมเดล ว่าเป็นคำหรือหน่วยเสียง จะขึ้นอยู่กับการใช้งานของระบบการรู้จำเสียงเช่น ถ้าระบบรู้จำเสียงมีข้อจำกัดคือให้รู้จำคำศัพท์ที่น้อยและรู้จำคำพูดที่ไม่ติดต่อกัน การโมเดลเสียงให้เป็นคำเสียงจะดีกว่าการ โมเดลหน่วยเสียงเนื่องจากการ โมเดลเป็นคำจะให้ความแม่นยำมากกว่าและวิธีการของระบบรู้จำคำ เสียงจะง่ายกว่าการ โมเดลหน่วยเสียง การโมเดลหน่วยเสียงจะต้องการ โมเดลแต่ละหน่วยเสียงและจะ ต้องรู้จำหน่วยเสียงนั้นๆก่อน เมื่อรู้จำหน่วยเสียงได้แล้วก็จะต้องมีวิธีการสร้างคำจากหน่วยเสียงที่ได้รู้ จำมา ถ้ามีการรู้จำหน่วยเสียงหนึ่งผิดไปก็จะทำให้คำคำนั้นรู้จำผิดได้ แต่สำหรับการโมเดลคำเสียง ความแม่นยำในการรู้จำจะขึ้น โดยตรงกับ โมเดลคำเสียงที่ได้ฝึกฝนมาและไม่ต้องการวิธีสร้างคำจาก หน่วยเสียง แต่ถ้าสำหรับระบบที่มีคำศัพท์มาก(มากกว่า 1000 คำ)และรู้จำเสียงแบบพูดติดต่อกัน ระบบ รู้จำที่มี โมเดลหน่วยเสียงจะมีประสิทธิภาพมากกว่าเพราะจำนวน โมเดลของหน่วยเสียงที่ต้องการจะมี น้อยกว่าโมเดลของคำเสียงและทำให้ระบบรู้จำเสียงใช้เวลาในการค้นหาคำเสียงที่มีความน่าจะเป็นมาก ที่สุดมากกว่า อีกประการหนึ่งที่สำคัญก็คือการรู้จำคำพูดแบบติดต่อกัน มีความเป็นไปได้สูงที่คำหนึ่ง คำจะเปลี่ยนไปเมื่อใช้กับคำๆอื่น ถ้าระบบมีการฝึกฝนที่ไม่ดีก็จะทำให้ระบบรู้จำมีความแม่นยำต่ำ แต่ สำหรับหน่วยเสียงจะมีการเปลี่ยนแปลงน้อยกว่าเมื่อใช้กับคำอื่นจึงทำให้มีความแม่นยำมากกว่า

ตามทฤษฎีแบบจำลองฮิดเดินมาร์คอฟ คือ การเก็บรวบรวมของสถานะหลายๆ สถานะที่ถูก เชื่อมโยงโดยการเปลี่ยนสถานะในที่นี้คือ ลักษณะของสัญญาณเสียงหรือเครื่องหมายที่หมายถึง สัญญาณเสียงนั้น ณ เวลาหนึ่งๆ ส่วนการเปลี่ยนสถานะคือ การเลื่อนหรือเปลี่ยนแปลงของสัญญาณ เสียงจากเวลาหนึ่งไปหาอีกเวลาหนึ่ง แบบจำลองฮิดเดินมาร์คอฟ มีค่าความน่าจะเป็นอยู่สองชนิดคือ ค่าความน่าจะเป็นที่การเปลี่ยนสถานะหนึ่งที่จะเกิดขึ้น (transition probability) และ ค่าความน่าจะเป็น ของผลลัพธ์ (output symbol) เมื่อมีการเปลี่ยนสถานะหนึ่งเกิดขึ้น ผลลัพธ์ที่ว่ามันจะนำไปใช้ในการตัด สินใจว่าเสียงที่ได้ยินนั้นเป็นเสียงอะไร (output probability) รูปที่ 2.9 แสดงถึงตัวอย่างของแบบจำลอง ฮิดเดินมาร์คอฟที่มี 2 เอ้าท์พุทคือ A และ B และ 2 สถานะ คือ 1 และ 2



รูปที่ 2.9 ตัวอย่าง แบบจำลองฮิดเดินมาร์คอฟ ที่มี 2 สถานะ และมี 2 เครื่องหมายผลลัพธ์

กำหนดให้

$a_{ij} = P(X_{t+1} = j | X_t = i)$ คือ a ความน่าจะเป็นของสถานะถัดไปจะเป็น j เมื่อสถานะปัจจุบันเป็น i

$b_{ij}(k) = P(Y_t = k | X_t = i, X_{t+1} = j)$ คือ ความน่าจะเป็นของผลลัพธ์เป็น k เมื่อมีการเปลี่ยนสถานะจากสถานะ-i ไปหาสถานะ-j

รูปที่ 2.9 เป็นตัวอย่างโมเดลแบบง่าย ๆ ที่ผ่านการฝึกฝนหรือมีการกำหนดค่าความน่าจะเป็นของ a และ b มาแล้ว ค่า $a_{12}=0.4$ หมายความว่า เมื่อสถานะปัจจุบันเป็น 1 สถานะต่อไปจะเป็นสถานะ 2 มีความน่าจะเป็นอยู่ 0.4 ส่วน $b_{12}(A)=0.8$ คือ เมื่อมีการเปลี่ยนสถานะจากสถานะ 1 ไปหาสถานะ 2 ความน่าจะเป็นที่จะมีผลลัพธ์เป็น A คือ 0.8 (เมื่อมีการเปลี่ยนสถานะจะมีการสร้างผลลัพธ์หนึ่งออกมาเท่านั้น)

คุณสมบัติของ a และ b มีดังนี้

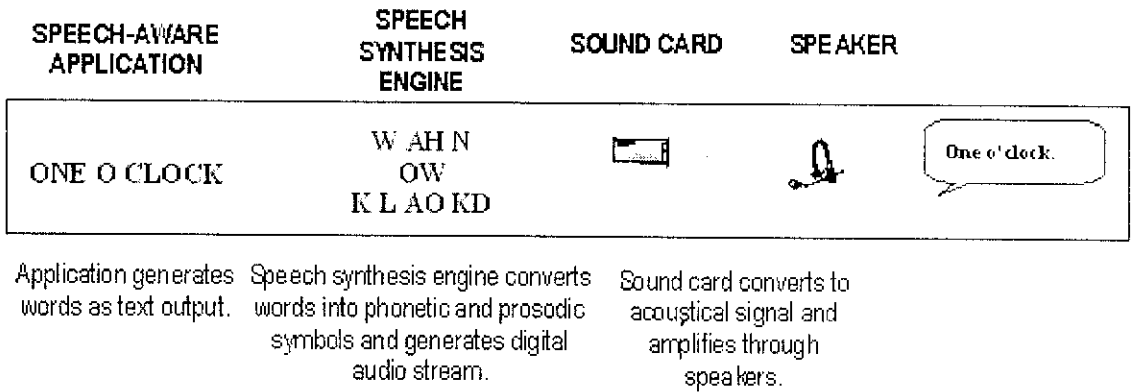
- $a_{ij} \geq 0, b_{ij}(k) \geq 0, \forall i, j, k$
- $\sum_j a_{ij} = 1$
- $\sum_k b_{ij}(k) = 1$

สำหรับแบบจำลองฮิดเดินมาร์คอฟลำดับที่ 1 จะต้องมีสมมุติฐานอยู่สองประการคือ

1. ความน่าจะเป็นของสถานะถัดไปขึ้นอยู่กับสถานะปัจจุบันเท่านั้น
2. ความน่าจะเป็นของเครื่องหมายผลลัพธ์ (symbol output) ถัดไปขึ้นอยู่กับสถานะปัจจุบันเท่านั้น ไม่ใช้การเปลี่ยนสถานะในอดีต

2.3.2 การสังเคราะห์เสียงพูด (Speech Synthesis)

การสังเคราะห์เสียงพูด (Speech Synthesis) หรือ text – to – speech คือกระบวนการในการแปลง text เป็นเสียงพูด โดยการแยกย่อย text เป็นส่วนเล็กๆ จากนั้นนำมาวิเคราะห์ ซึ่งในการวิเคราะห์ text ที่เป็นแบบพิเศษ เช่น ตัวเลข, ค่าเงิน, การออกเสียงสูง – ต่ำ และ การเน้นวรรคตอน ซึ่งจะมีการเพิ่มฟังก์ชันเพื่ออักษรแบบพิเศษ จากนั้นจะสร้าง digital audio เพื่อเล่นเสียงที่ประมวลได้ รูปที่ 2.10 แสดงกระบวนการสังเคราะห์เสียงพูด



รูปที่ 2.10 แสดง กระบวนการสังเคราะห์เสียงพูด

2.4 โปรแกรมภาษา Visual Basic เพื่อการสั่งงานเมาส์ด้วยเสียงพูด

2.4.1 การทำงานในส่วนของการเคลื่อนที่

การทำงานในส่วนนี้จะใช้ ความสามารถของ DLL ของวินโดวส์ประกอบการเขียนโปรแกรมรองรับเพื่อให้เมาส์สามารถเคลื่อนที่ได้ ซึ่งมีฟังก์ชันที่ใช้ดังนี้คือ

```
Public Declare Function SetCursorPos Lib "user32" (ByVal x As Long, ByVal y As Long) As Long
```

สำหรับ ส่งเมาส์เคอร์เซอร์ไปตำแหน่งที่ต้องการ

ตัวอย่าง

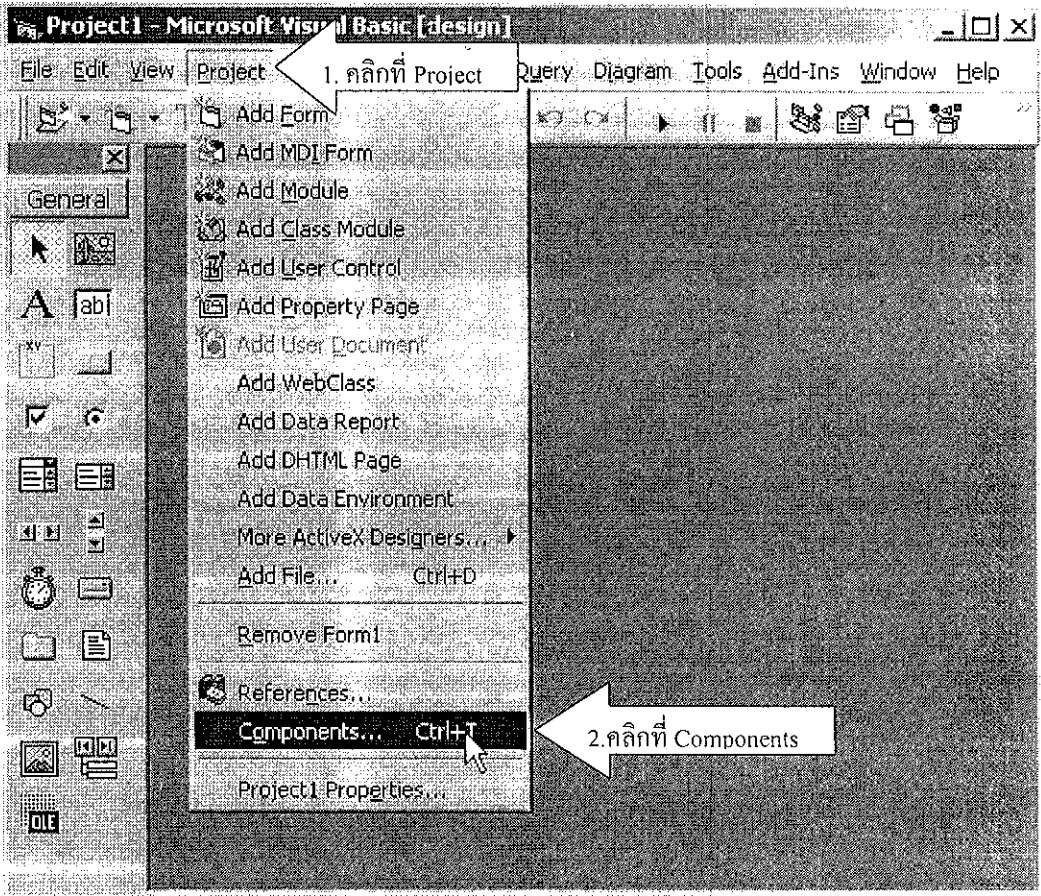
คำสั่ง -> SetCursorPos (100,100)

ผล -> เมาส์เคอร์เซอร์เคลื่อนไปที่พิกัด (100,100)

2.4.2 การทำงานในส่วนของการรู้จำเสียง

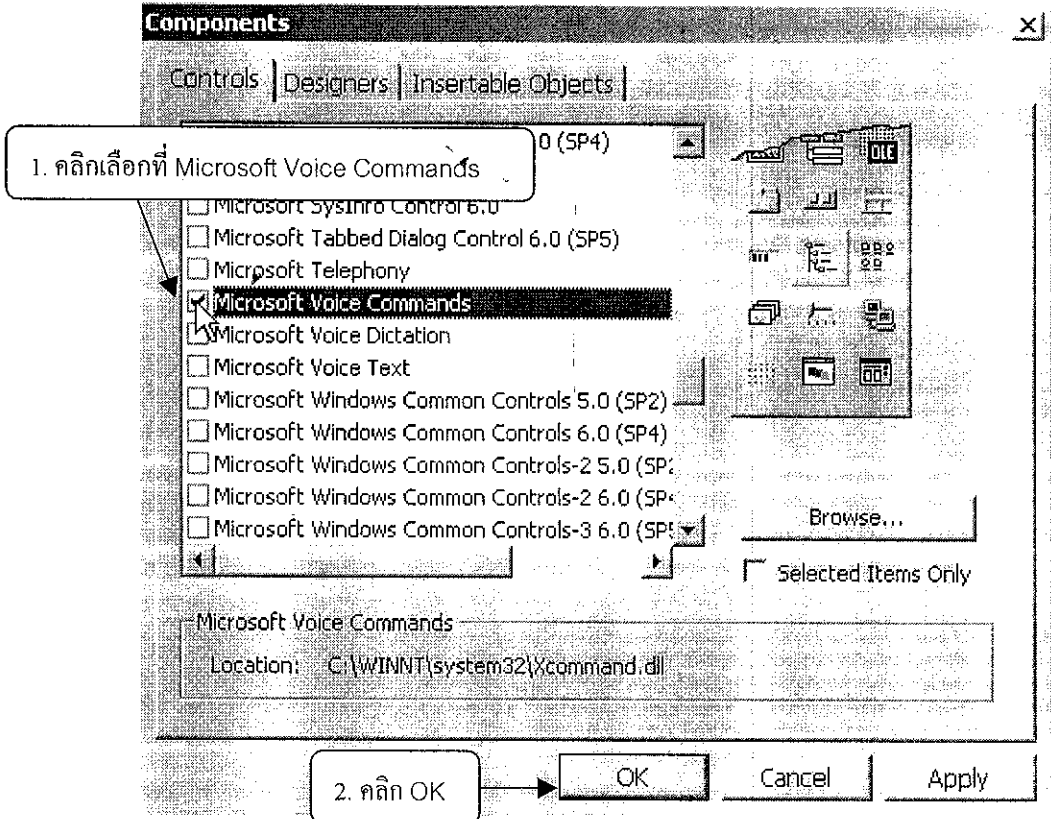
การทำงานในส่วนนี้จำเป็นต้องมี การติดตั้งโปรแกรม Microsoft Speech SDK 4 เมื่อมีการติดตั้งเสร็จแล้ว ให้ทำการเพิ่มคอมโพเนนท์ Microsoft Voice Command ซึ่งมีขั้นตอนดังนี้

1. เมื่อเปิดโปรแกรม Visual Basic เสร็จเรียบร้อยแล้ว คลิกที่เมนู Project และเลือกที่ Components ดังรูปที่ 2.11



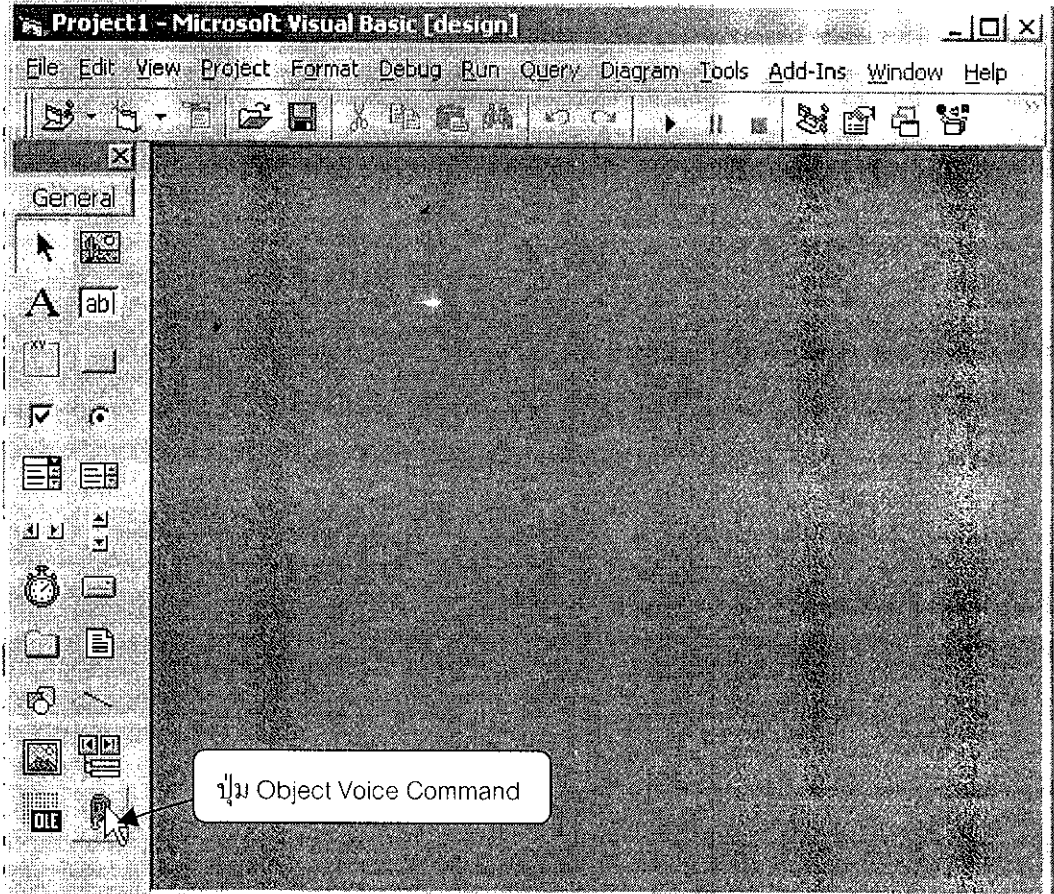
รูปที่ 2.11 แสดงการเพิ่มคอมโพเนนท์

2. จากนั้นคลิกเลือกที่ Microsoft Voice Commands ดังรูปที่ 2.12



รูปที่ 2.12 แสดงการเลือก Microsoft Voice Commands เพื่อเพิ่มคอมโพเนนท์

3. เมื่อเสร็จสิ้นแล้ว จะได้ Object Voice Command เพิ่มมาดังรูปที่ 2.13



รูปที่ 2.13 แสดงคอมโพเนนต์ Microsoft Voice Command หลังจากเพิ่มเสร็จแล้ว

จากนั้นสามารถเขียนโค้ดใน Visual Basic ได้ดังนี้

- Private Sub Vcommand1_CommandRecognize(ByVal ID As Long, ByVal CmdName As String, ByVal flags As Long, ByVal action As String, ByVal NumLists As Long, ByVal ListValues As String, ByVal command As String)
เพื่อให้โปรแกรมรู้จำเสียงพูด เมื่อพูดคำสั่งใดออกไป โปรแกรมจะไปทำงานตามSub
นั้นๆ ซึ่งในการทำงานในคำสั่งนี้ จำเป็นต้องมีการ initial คำดังนี้

```
Vcommand1.Initialized = 1
gMymenu = Vcommand1.MenuCreate(App.EXENAME, "state1", 4)
Vcommand1.Enabled = 1
```
- Vcommand1.Activate gMymenu
เพื่อให้โปรแกรมเริ่มฟังเสียงพูด
- Vcommand1.Deactivate gMymenu
เพื่อให้โปรแกรมหยุดฟังเสียงพูด
- Vcommand1.Remove gMymenu, 1
เพื่อให้โปรแกรมลบคำสั่งที่ต้องการรู้จำเสียงพูดออก
- Vcommand1.AddCommand gMymenu, 1, Moption(i), "when you say"--Moption(i).
"listen list", 0, ""
เพื่อให้โปรแกรมเพิ่มคำสั่งที่ต้องการรู้จำเสียงพูด

2.4.3 การทำงานในส่วนของการคลิกเมาส์

การทำงานในส่วนนี้จะใช้ความสามารถของของวินโดวส์ API ประกอบกับการเขียนโปรแกรมมารองรับเพื่อทำให้สามารถคลิกเมาส์ได้ ซึ่งมีฟังก์ชันที่ใช้ดังนี้คือ

- Public Declare Sub mouse_event Lib "user32.dll" (ByVal dwFlags As Long, ByVal dx As Long, ByVal dy As Long, ByVal cButtons As Long, ByVal dwExtraInfo As Long)
เป็นการประกาศเพื่อให้รู้จักเหตุการณ์คลิกเมาส์

ตัวอย่างการคลิกขวา

คำสั่ง `Public Const MOUSEEVENTF_RIGHTDOWN = &H8`
 `Public Const MOUSEEVENTF_RIGHTUP = &H10`
 อธิบายผล เป็นเหตุการณ์การคลิกขวา

ตัวอย่างการคลิกซ้าย

คำสั่ง `Public Const MOUSEEVENTF_LEFTDOWN = &H2`
 `Public Const MOUSEEVENTF_LEFTUP = &H4`
 อธิบายผล เป็นเหตุการณ์การคลิกซ้าย

ตัวอย่างการดับเบิลคลิก

คำสั่ง `Public Const MOUSEEVENTF_LEFTDOWN = &H2`
 `Public Const MOUSEEVENTF_LEFTUP = &H4`
 `Public Const MOUSEEVENTF_RIGHTDOWN = &H8`
 `Public Const MOUSEEVENTF_RIGHTUP = &H10`
 อธิบายผล เป็นเหตุการณ์การดับเบิลคลิก

2.4.4 การทำงานในส่วนของ Text Mode

- `SendKeys (String)`
 เป็นการส่งค่าตัวอักษรเพื่อพิมพ์ออกที่ตำแหน่งที่ต้องการ
- `SendKeys("{Enter}")`
 เป็นตัวอย่างการให้โปรแกรมเสมือนกด Enter ที่คีย์บอร์ด

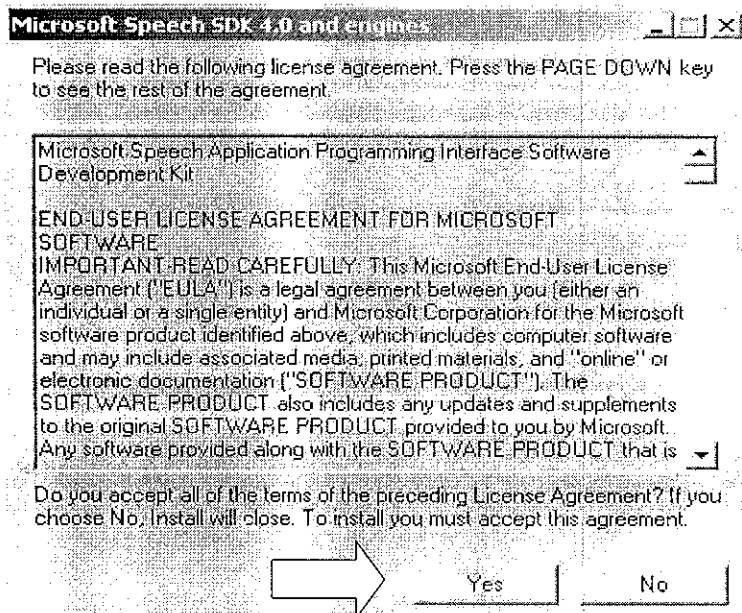
บทที่ 3

โปรแกรมเมตาส่งงานด้วยเสียงพูด

3.1 ขั้นตอนการ Install

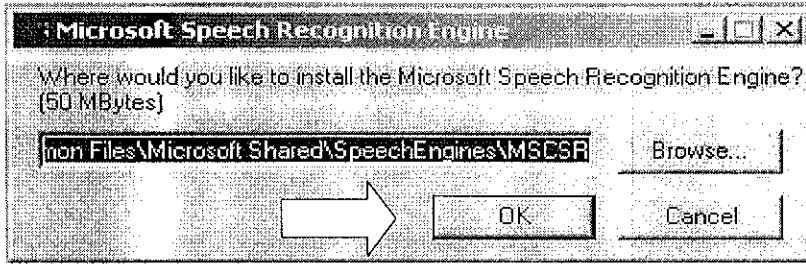
ในการใช้งานโปรแกรมเมตาส่งงานด้วยเสียงพูด ขั้นแรกจะต้องติดตั้ง Speech Application Programming Interface (SAPI) ของ Microsoft ซึ่งสามารถ Download ได้จากที่ <http://microsoft.com/downloads/details.aspx?FamilyID=7d13964c-06fd-4bf9-b49c-814faa6a86ea&DisplayLang=en> เมื่อ Download เสร็จแล้วให้ทำการติดตั้งดังต่อไปนี้

1. ให้ Double click ที่ file ที่ชื่อ SAPI4SDKSUITE.exe เพื่อทำการติดตั้งจะปรากฏผลดังรูปที่ 3.1 และกด Yes เพื่อยอมรับข้อตกลงของ Microsoft



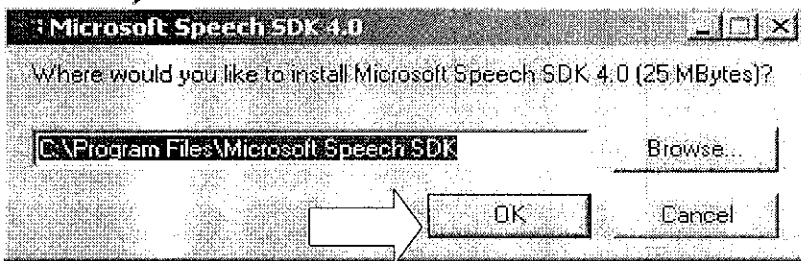
รูปที่ 3.1 แสดงหน้าต่างเพื่อทำการติดตั้งโปรแกรม Microsoft Speech SDK 4.0 และ Engines

2. เมื่อตอบ Yes แล้วรอสักครู่จนปรากฏไดอะล็อกบ็อกดังรูปที่ 3.2 และตอบ OK เพื่อทำการติดตั้ง Speech Engine ในไดเรกทอรีที่กำหนดไว้



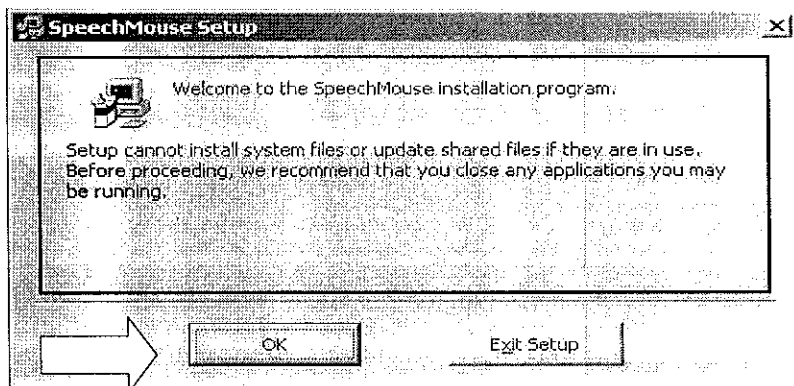
รูปที่ 3.2 แสดง ไดอะล็อกบ็อกซ์ที่ทำการติดตั้ง Speech Engine

3. เมื่อขั้นตอนที่ 2 เสร็จสิ้นแล้ว จะปรากฏไดอะล็อกบ็อกซ์ดังรูปที่ 3.3 เพื่อทำการติดตั้ง Microsoft Speech SDK 4.0 แล้วกดที่ปุ่ม OK เป็นการจบขั้นตอนการติดตั้ง

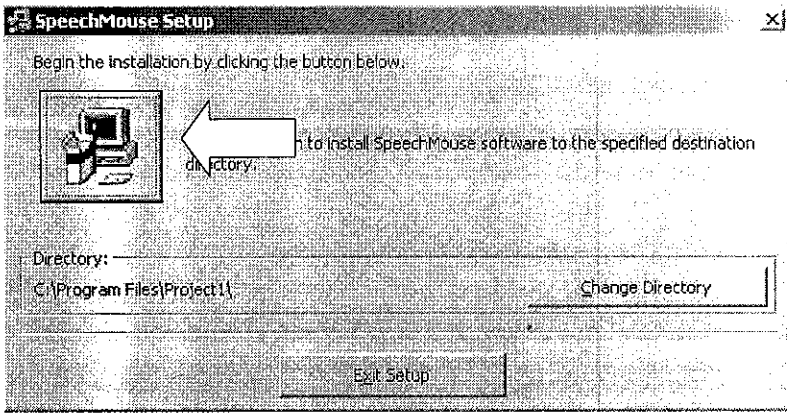


รูปที่ 3.3 แสดงการติดตั้ง Microsoft Speech SDK 4.0

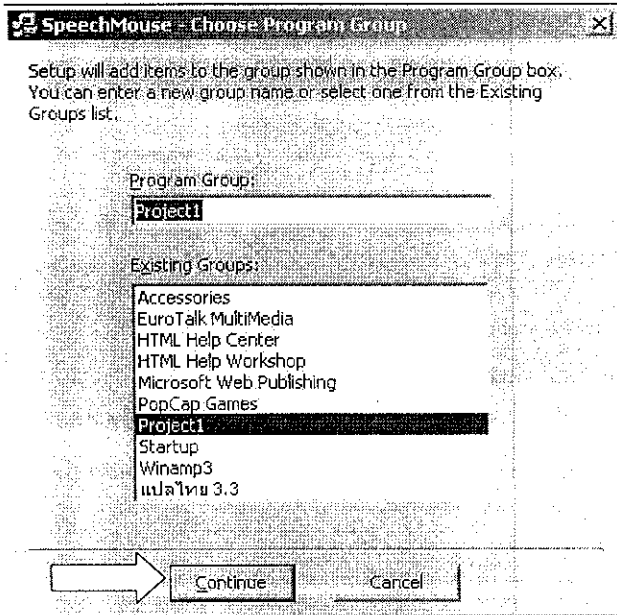
4. เมื่อเสร็จสิ้นขั้นตอนที่ 3 แล้ว ต่อไปเราจะทำการติดตั้ง โปรแกรมเมาส์สั่งงานด้วยเสียงพูด โดยดับเบิลคลิกที่ Setup จะปรากฏไดอะล็อกบ็อกซ์ดังรูปที่ 3.4 จนถึงรูปที่ 3.8



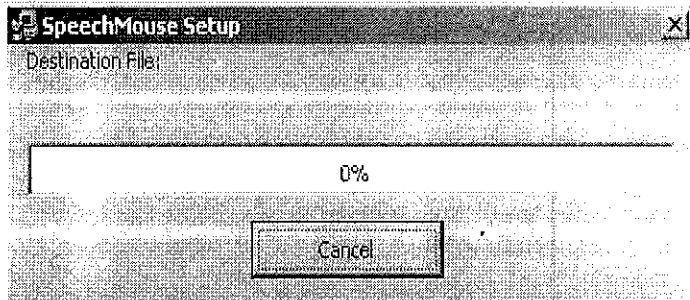
รูปที่ 3.4



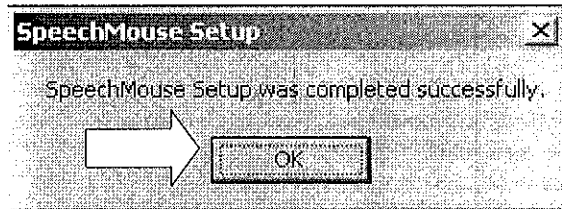
รูปที่ 3.5



รูปที่ 3.6



รูปที่ 3.7



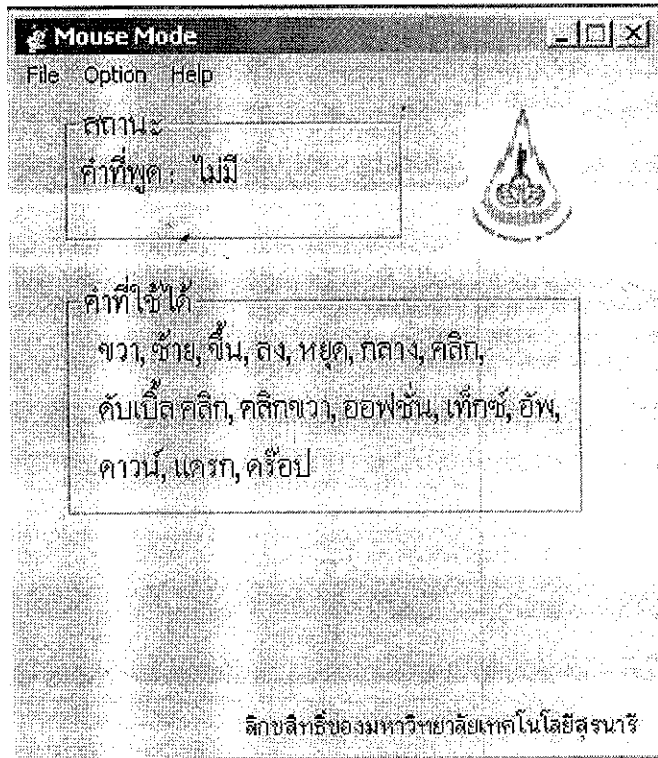
รูปที่ 3.8

5. เมื่อเราทำการติดตั้ง โปรแกรมเสร็จสิ้นแล้ว ต่อไปเราจะทำการเปิดโปรแกรมโดยเราสามารถเปิดโปรแกรมได้โดย double click ที่ Icon หรือที่ start menu -> programs -> startup เพื่อทำการ run โปรแกรม และจะมีรูปแบบขอโปรแกรมดังรูปที่ 3.9

3.2 ส่วนประกอบของโปรแกรม

ในส่วนของโปรแกรมประยุกต์จะประกอบด้วย 3 ส่วนใหญ่ๆ ดังนี้

3.2.1 Mouse Mode ดังแสดงในรูปที่ 3.9 ซึ่งใน Mode นี้ประกอบไปด้วย



รูปที่ 3.9 แสดงสถานะเมื่อโปรแกรมทำงานใน Mouse Mode

1. การเคลื่อนที่ ซึ่งประกอบไปด้วยคำสั่งดังนี้

ขึ้น คือ เม้าส์เคอร์เซอร์เลื่อนขึ้น

ลง คือ เม้าส์เคอร์เซอร์เลื่อนลง

ขวา คือ เม้าส์เคอร์เซอร์เลื่อนไปทางขวา

ซ้าย คือ เม้าส์เคอร์เซอร์เลื่อนไปทางซ้าย

กลาง คือ เม้าส์เคอร์เซอร์เลื่อนไปหยุดที่จุดกึ่งกลางหน้าจอ

2. คลิก ซึ่งประกอบไปด้วยคำสั่งดังนี้

ดับเบิลคลิก คือ คลิกซ้าย 2 ครั้ง

คลิกขวา คือ คลิกขวา 1 ครั้ง

แดรก คือ คลิกซ้ายค้างไว้

ดรออป คือ การปล่อยแดรก

3. การเชื่อมต่อไปยัง Option Mode และ Text Mode ซึ่งประกอบไปด้วยคำสั่งดังนี้

ออฟชั่น คือ เชื่อมต่อ ไปยัง Option Mode

เท็กซ์ คือ เชื่อมต่อ ไปยัง Text Mode

4. การเรียกคืนหน้าต่างโปรแกรม คำสั่งที่ใช้คือ

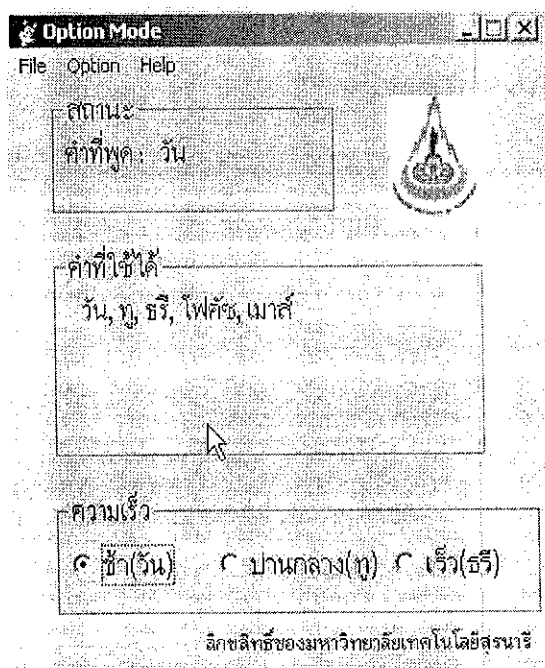
โฟกัส คือ เรียกคืนหน้าต่างโปรแกรม

5. การเลื่อนหน้าต่าง คำสั่งที่ใช้คือ

อัป คือ การกด Page Up บนคีย์บอร์ด

ดาวน์ คือ การกด Page Down บนคีย์บอร์ด

3.2.2 Option Mode ดังแสดงในรูปที่ 3.10 ใน Mode นี้ประกอบไปด้วย



รูปที่ 3.10 แสดงสถานะเมื่อโปรแกรมทำงานใน Option Mode

1. การปรับความเร็วของเมาส์ซึ่งมี 3 ระดับคือ

วัน คือ ช้า

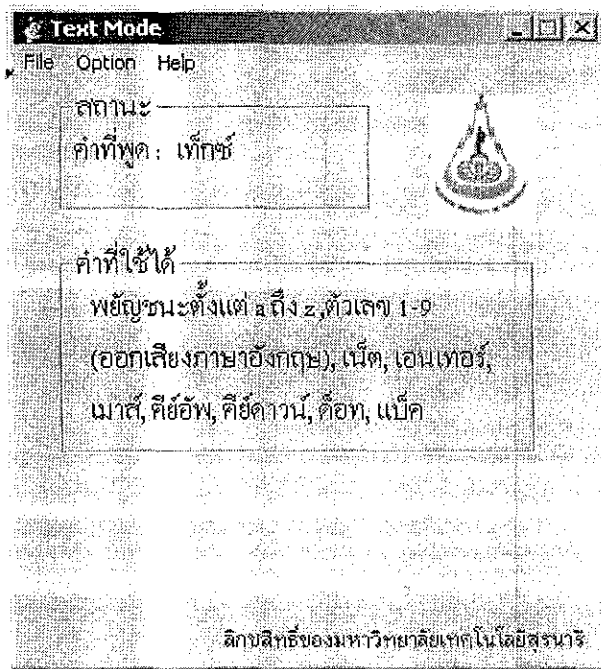
ทู คือ ปานกลาง

ทรี คือ เร็ว

2. การกลับสู่ Mouse Mode ด้วยคำสั่งดังนี้

เมาส์ คือ การสั่งให้โปรแกรมกับไปทำงานที่ Mouse Mode

3.2.3 Text Mode ดังแสดงในรูปที่ 3.11 ใน Mode นี้ประกอบไปด้วย



รูปที่ 3.11 แสดงสถานะเมื่อ โปรแกรมทำงานใน Text Mode

1. การสั่งพิมพ์ตัวอักษรภาษาอังกฤษ ได้แก่ ตัวเลขตั้งแต่ 0-9 และตัวอักษรภาษาอังกฤษ a ถึง z โดยออกเสียงเป็นภาษาอังกฤษ

2. การใช้เป็นลูกศรมีคำสั่งดังนี้

คีย์อัฟ คือ การกดแป้นลูกศรขึ้น

คีย์ดาวน์ คือ การกดแป้นลูกศรลง

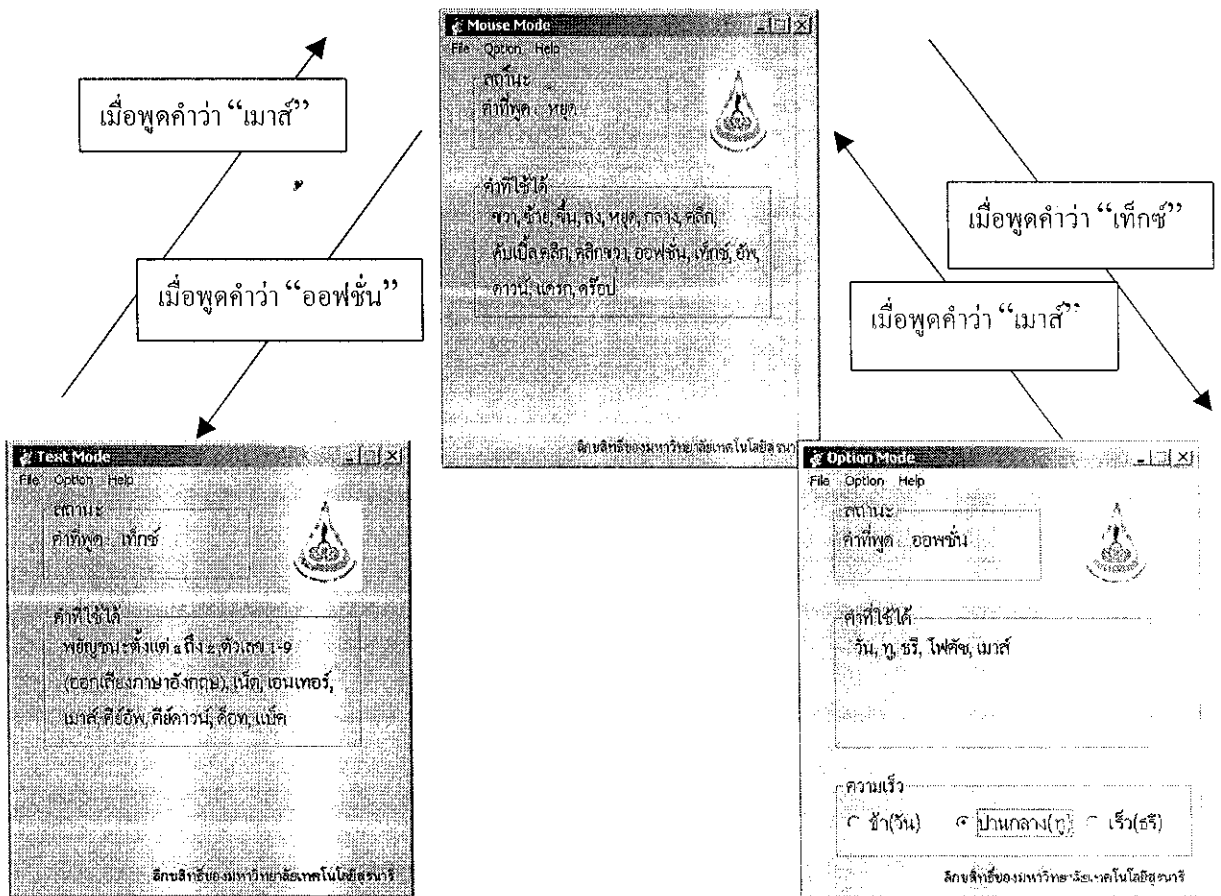
3. การให้คำสั่งพิเศษมีคำสั่งดังนี้

แบ็ค คือ การลบตัวอักษร 1 ตัว

คือท คือ การพิมพ์จุด 1 ตัว

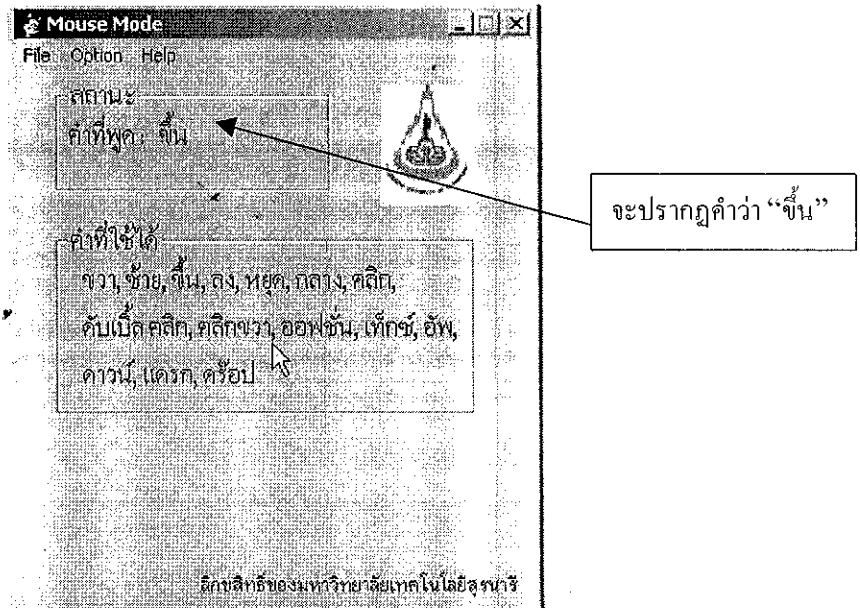
เน็ต คือ การพิมพ์ http://www.

3.2.4 สรุปการทำงานโดยรวมของโปรแกรมเมาส์สั่งงานด้วยเสียงพูดดังรูปที่ 3.12



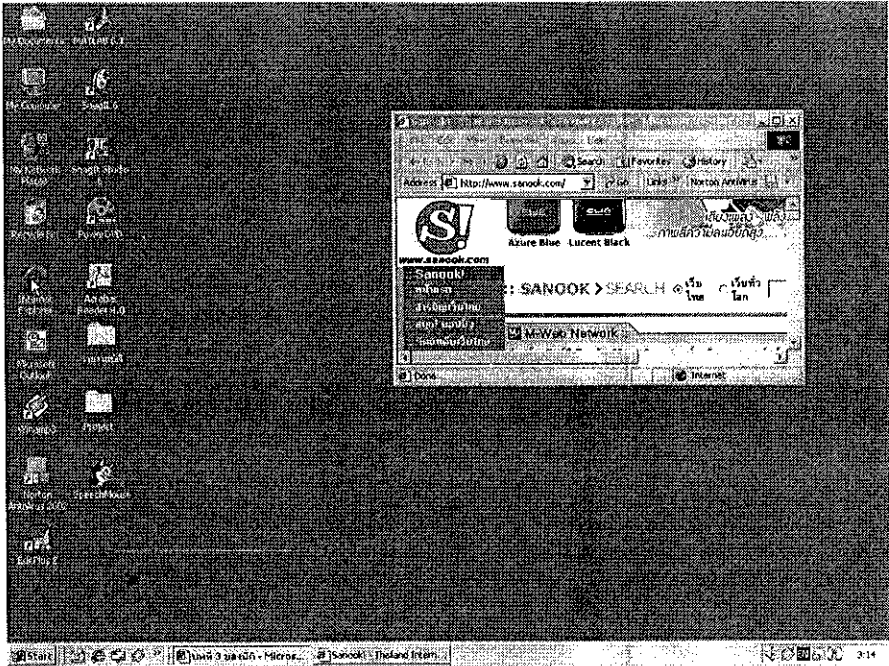
รูปที่ 3.12 แสดงการทำงานโดยรวมของโปรแกรมเมาส์สั่งงานด้วยเสียงพูด

2. ในการสั่งงานให้เมาส์เคลื่อนที่ได้นั้นจะต้องออกเสียงให้ตรงกับคำสั่งที่เขียนแสดงไว้ในบล็อกด้านล่างของโปรแกรม ซึ่งคำสั่งที่ใช้ในการสั่งงานให้เมาส์เคลื่อนที่ได้แก่ ขึ้น, ลง, ซ้าย, ขวา ดังเช่นแสดงตัวอย่างในรูปที่ 3.16 ในการเคลื่อนที่ของเมาส์นั้นจะมีความเร็วในการเคลื่อนที่ในตอนเริ่มเปิดโปรแกรมที่ 70 pixel/sec และสามารถใช้คำสั่ง “กลาง” เพื่อให้เมาส์มาอยู่ที่ตำแหน่งกลางหน้าจอ

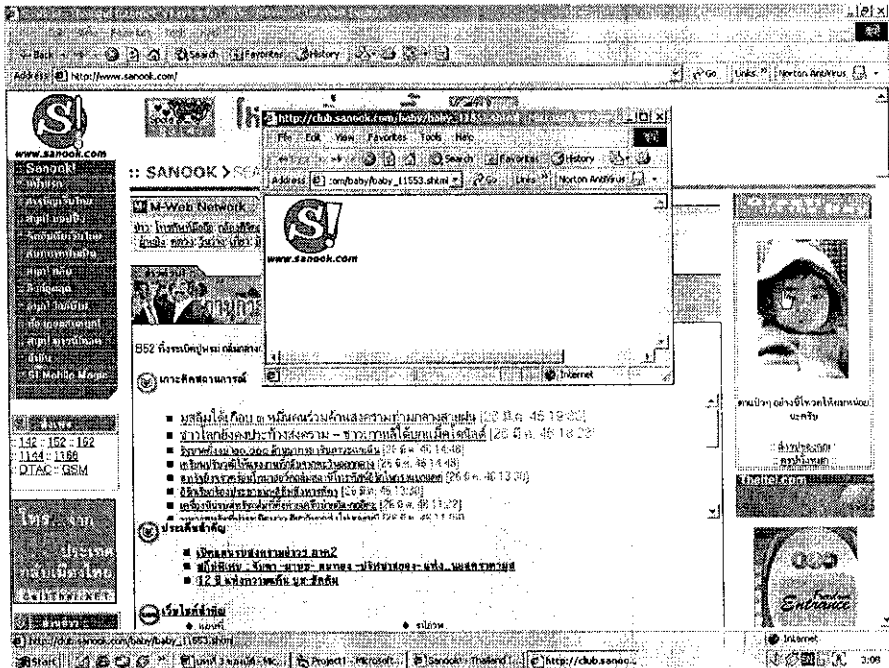


รูปที่ 3.15 แสดงตัวอย่างการสั่งงานให้เมาส์เคลื่อนที่ขึ้น

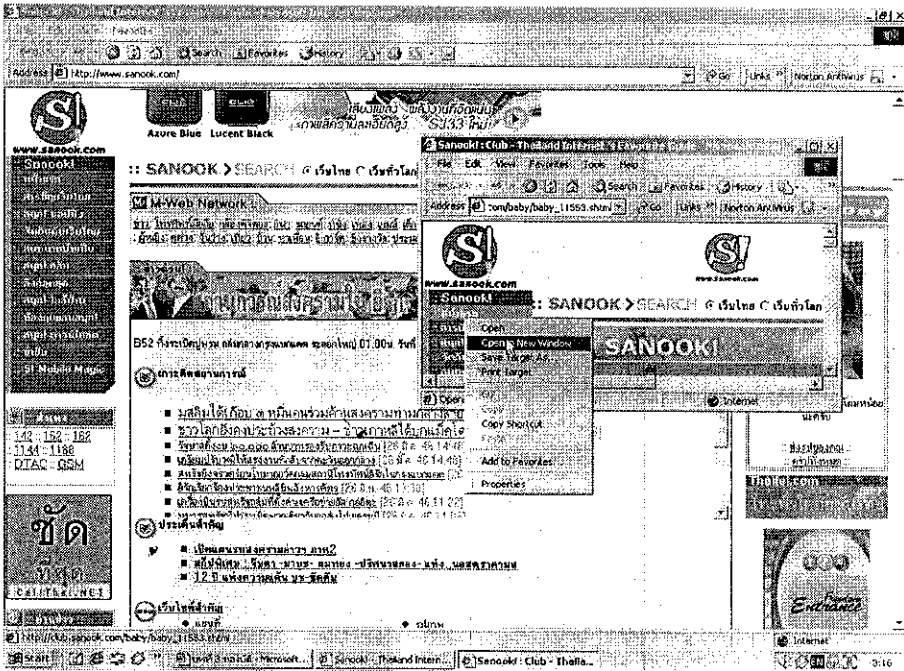
3. เมื่อเมาส์เคลื่อนที่มายังตำแหน่งที่ต้องการแล้ว สามารถสั่งให้เมาส์ดับเบิลคลิกเพื่อเปิดโปรแกรม ได้โดยออกเสียงคำว่า “ดับเบิลคลิก” จะได้ผลดังรูปที่ 3.17(a) และยังสามารถสั่งให้เมาส์คลิกเพื่อลิงค์ไปยัง web อื่น ๆ โดยการออกเสียงว่า “คลิก” จะได้ผลดังรูปที่ 3.17(b) นอกจากนั้นสามารถสั่งให้เมาส์คลิกขวาโดยการออกเสียงคำว่า “คลิกขวา” ดังแสดงในรูปที่ 3.17(c)



รูปที่ 3.16a) การดับเบิลคลิกที่ Internet Explorer

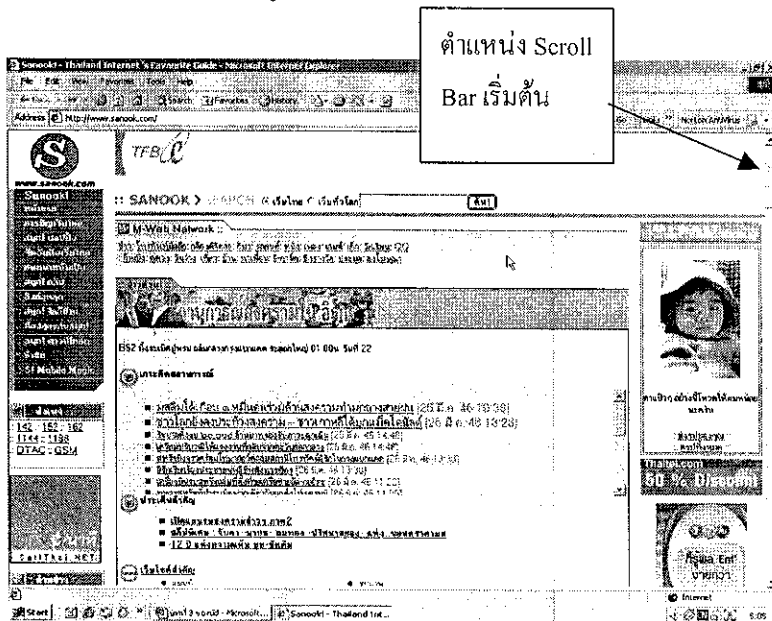


รูปที่ 3.16b) การคลิกเพื่อลิงก์ไปยัง web อื่นๆ

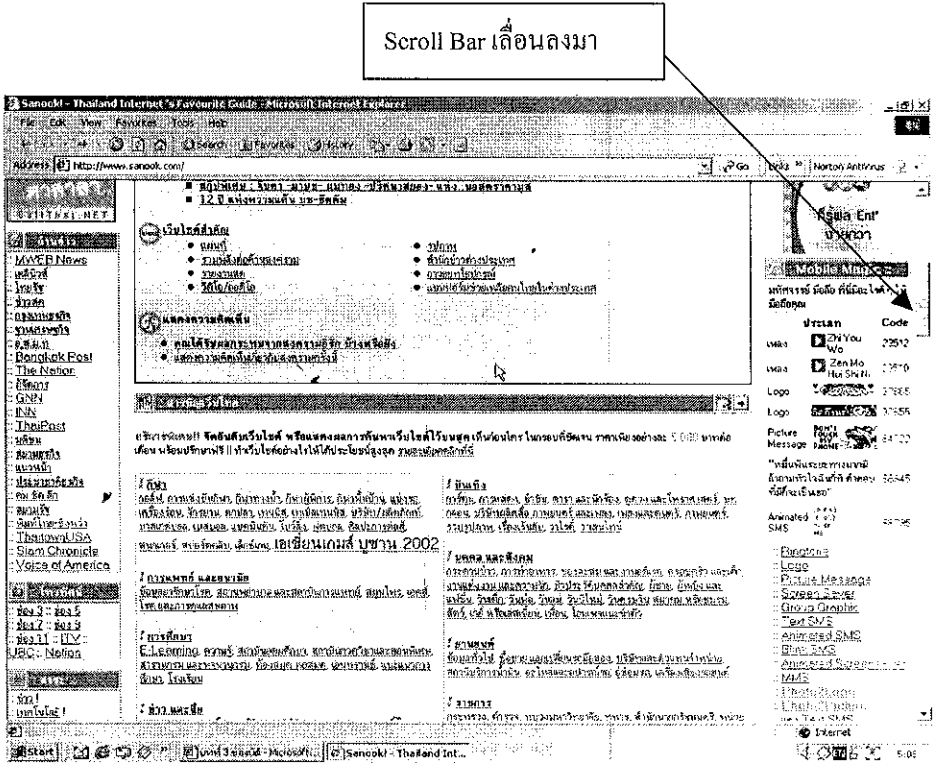


รูปที่ 3.16c) การคลิกขวา

4. การใช้คำสั่ง “อัป” เป็นการสั่งงานเสมือนให้ Keyboard กดปุ่ม Page Up และ คำสั่ง “ดาวน์” เป็นการสั่งงานเสมือนให้ Keyboard กดปุ่ม Page Down ดังแสดงเป็นตัวอย่างในรูปที่ 3.17 เป็นการเลื่อน Scroll Bar ลงเพื่อที่จะสามารถดู Web ได้ทั้งหมด

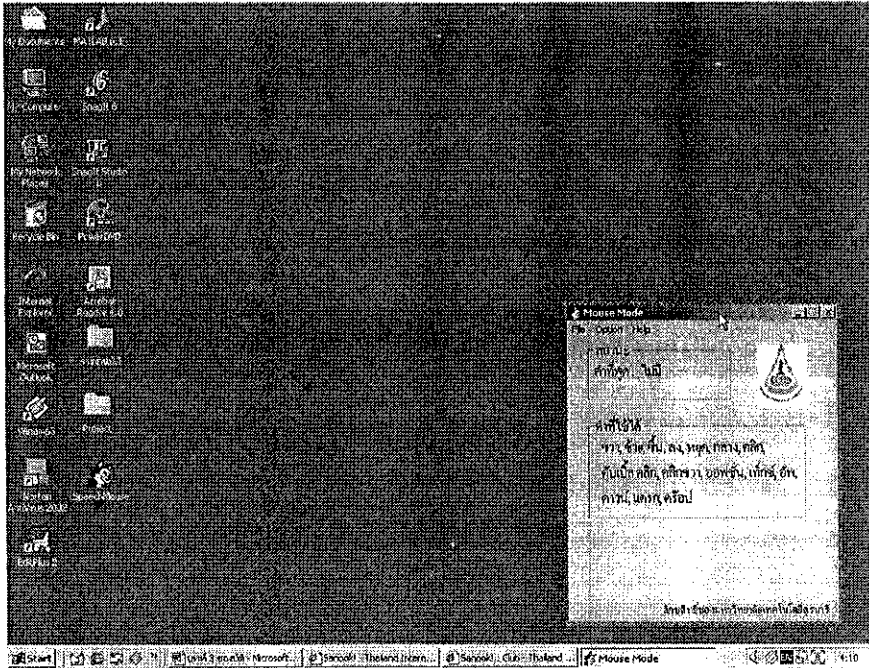


รูปที่ 3.17 แสดงภาพเริ่มต้นก่อนใช้คำสั่ง “ดาวน์โหลด”

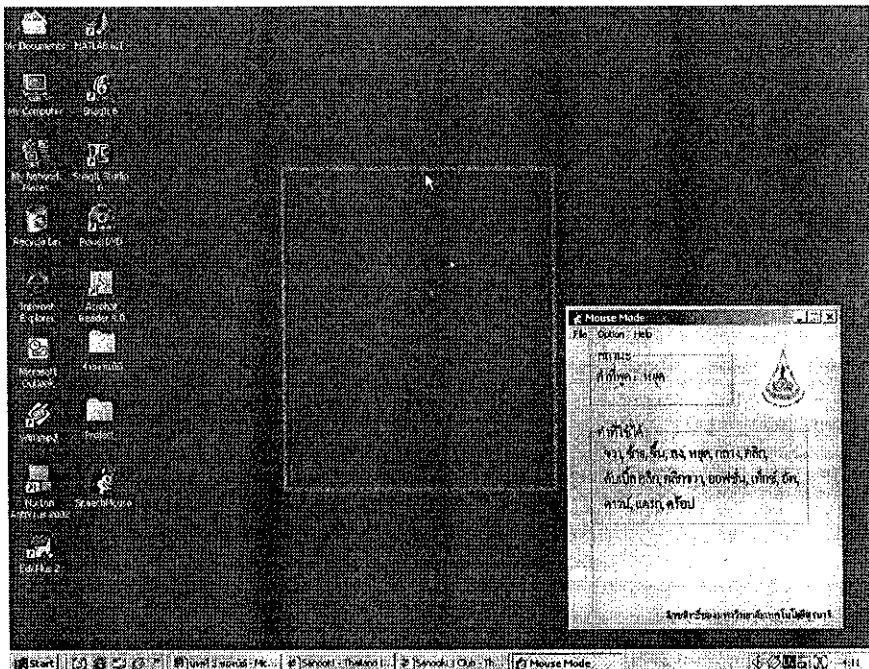


รูปที่ 3.18 แสดงผลการใช้คำสั่ง “ดาวน์โหลด”

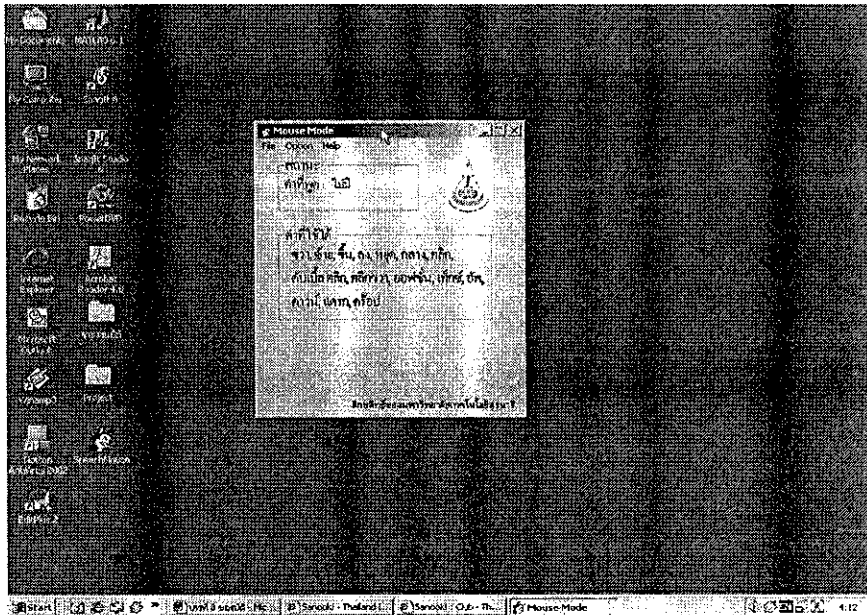
6. การใช้คำสั่ง “แทรก” เป็นการสั่งให้เมาส์คลิกขวาค้างไว้ และคำสั่ง “ครีอป” เป็นการสั่งให้เมาส์ปล่อยจากการคลิกขวา ดังที่แสดงในรูปที่ 3.18 เป็นการตั้งงานให้เมาส์เคลื่อนย้ายหน้าต่างของโปรแกรมเมาส์ตั้งงานด้วยเสียงพูดย้ายไปยังที่ตำแหน่งต่างๆ



รูปที่ 3.19(a) แสดงการใช้คำสั่ง “แดรก”

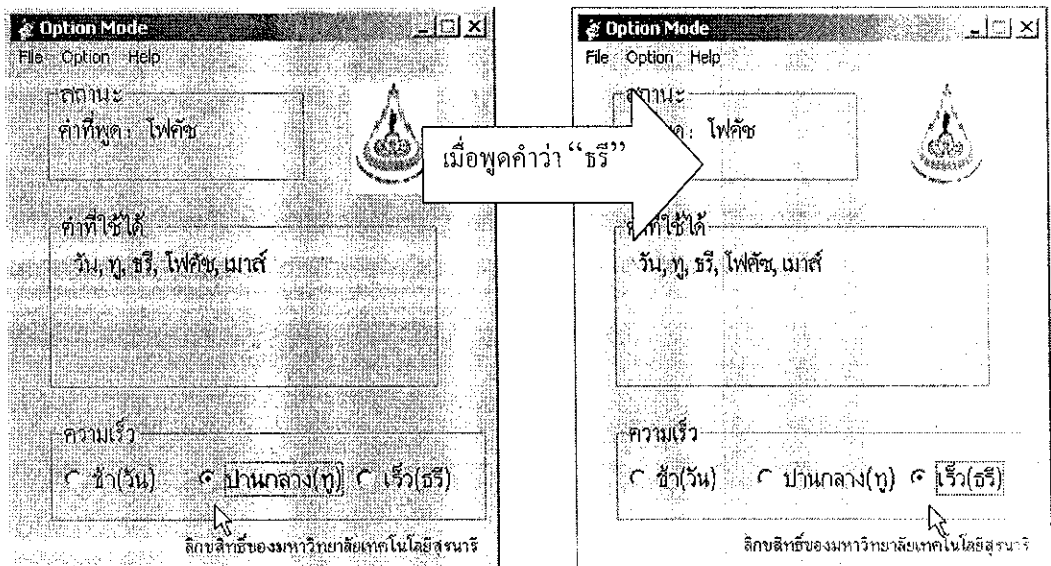


รูปที่ 3.19(b) แสดงการเคลื่อนย้ายของหน้าต่างโปรแกรม



รูปที่ 3.19(c) แสดงการใช้คำสั่ง “ตรีอ”

6. จากที่ได้กล่าวมาแล้วในข้อที่ 1 ถึง 5 นั้นเป็นการสั่งงานใน Mouse Mode เมื่อเราต้องการที่จะปรับความเร็วของเมาส์ได้โดยการสั่งงานด้วยคำสั่ง “ออฟชั่น” จะปรากฏตัวโปรแกรมดังรูปที่ 3.19 (a) จะเห็นได้ว่าเราสามารถปรับความเร็วของเมาส์ได้ 3 ระดับคือ ช้า, ปานกลาง, และเร็ว เช่นเมื่อออกเสียงคำว่า “ตรี” จะทำให้ความเร็วของเมาส์เปลี่ยนไป ดังแสดงในรูปที่ 3.19(b)



รูปที่ 3.20 แสดงการเปลี่ยนความเร็วของเมาส์

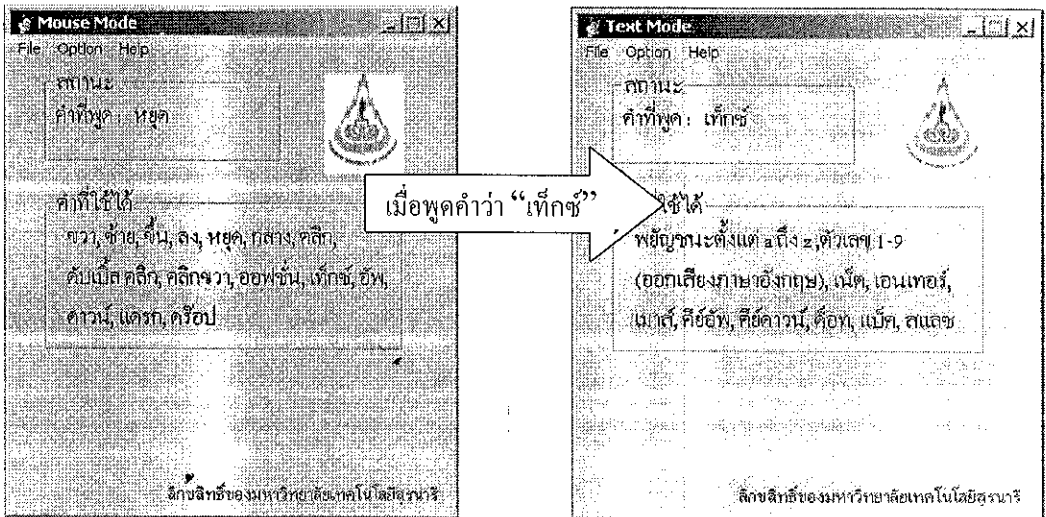
- ที่ความเร็วของเมาส์ที่ระดับ “ช้า” เมาส์จะเคลื่อนที่ด้วยความเร็วโดยประมาณ 23 pixel/sec
- ที่ความเร็วของเมาส์ที่ระดับ “ปานกลาง” เมาส์จะเคลื่อนที่ด้วยความเร็วโดยประมาณ 70 pixel/sec
- ที่ความเร็วของเมาส์ที่ระดับ “เร็ว” เมาส์จะเคลื่อนที่ด้วยความเร็วโดยประมาณ 140 pixel/sec
- เมื่อต้องการที่จะกลับไปทำงานที่ Mouse Mode ให้สั่งงานด้วยการออกเสียงว่า “เมาส์”

7. การใช้คำสั่ง “โฟกัส” เป็นการสั่งงานตัวโปรแกรมให้ Minimize หรือ Maximize โดยที่คำสั่งนี้สามารถใช้ได้ในทุกๆ โหมดการทำงานของโปรแกรมดังแสดงในรูปที่ 3.20



รูปที่ 3.21 แสดงการใช้คำสั่ง “โฟกัส”

7. ต่อไปจะเป็นการทำงานของโปรแกรมเมาส์สั่งงานด้วยเสียงพูดใน Text Mode ซึ่งการจะเข้าไปสู่การทำงานใน Text Mode จะต้องใช้คำสั่ง “ที็กซ์” เมื่อพูดคำว่า “ที็กซ์” แล้วตัวโปรแกรมจะมีการเปลี่ยนแปลงดังแสดงในรูปที่ 3.21



รูปที่ 3.22 แสดงการเปลี่ยนสถานะของโปรแกรม

ในส่วนของการทำงานใน Text Mode นั้นมีคำสั่งที่สามารถสั่งได้คือ พยัญชนะภาษาอังกฤษ a ถึง z และตัวเลข 1-9 โดยที่ออกเสียงสั่งงานเป็นภาษาอังกฤษเช่น “เอส”, “ยู”, “ที” ดังนั้นผลลัพธ์ที่แสดงออกมาคือ sut ถ้าพิมพ์ผิดสามารถลบได้โดยใช้คำสั่ง “แบ็ค” และสามารถใส่ลูกศรขึ้นและลงได้โดยใช้คำสั่ง “คีย์อัฟ”, “คีย์คิวรี่” ตามลำดับ

ใน Text Mode นี้จะมีคำสั่งพิเศษจะช่วยอำนวยความสะดวกในการใช้อินเตอร์เน็ต ดัง

นี้

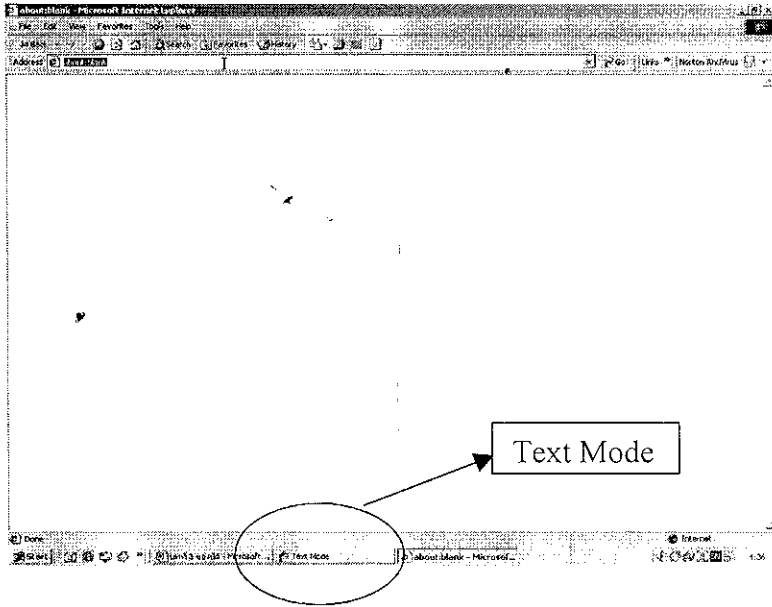
1. คำสั่ง “เน็ต” จะเป็นการพิมพ์ “http://www.”
2. คำสั่ง “ดอท” จะเป็นการพิมพ์ “.”
3. คำสั่ง “สแลช” จะเป็นการพิมพ์ “/”
4. คำสั่ง “เอนเทอร์” จะเป็นการสั่งให้กด Enter

เมื่อต้องการที่จะกลับไปสู่การทำงานใน Mouse Mode ให้พูดคำว่า “เมาส์”

ตัวอย่างการใช้งานโปรแกรมเมาส์สั่งงานด้วยเสียงพูดในการทำงานที่ Text Mode

1. เมื่อคลิกเข้ามาใน Internet Explorer สั่งคลิกที่ Address Bar จากนั้นเลือกการทำงาน

ที่ Text Mode



รูปที่ 3.23 แสดงผลการคลิกที่ Address Bar และการเลือก Text Mode

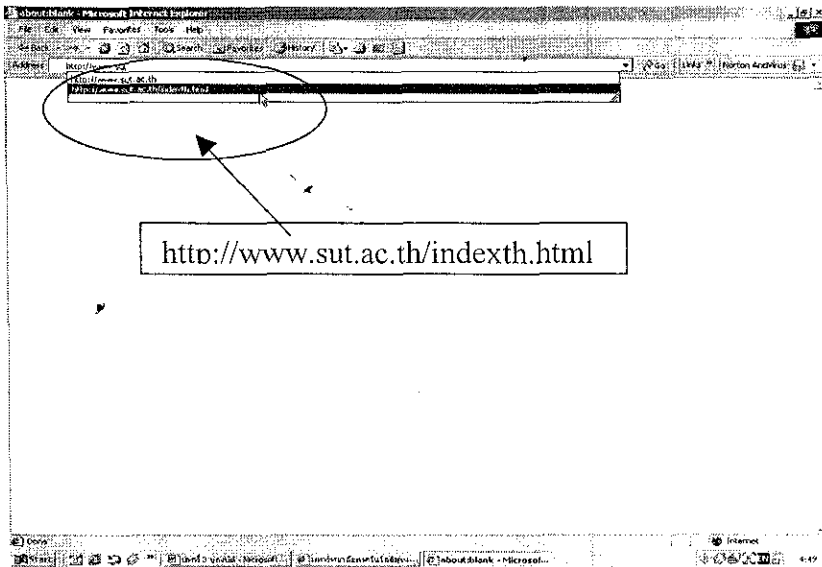
2. พูดย่อคำว่า “เน็ต” ที่ Address Bar จะแสดงตัวอักษร http://www.



รูปที่3.24 แสดงผลการใช้คำสั่ง “เน็ต” ในText Mode

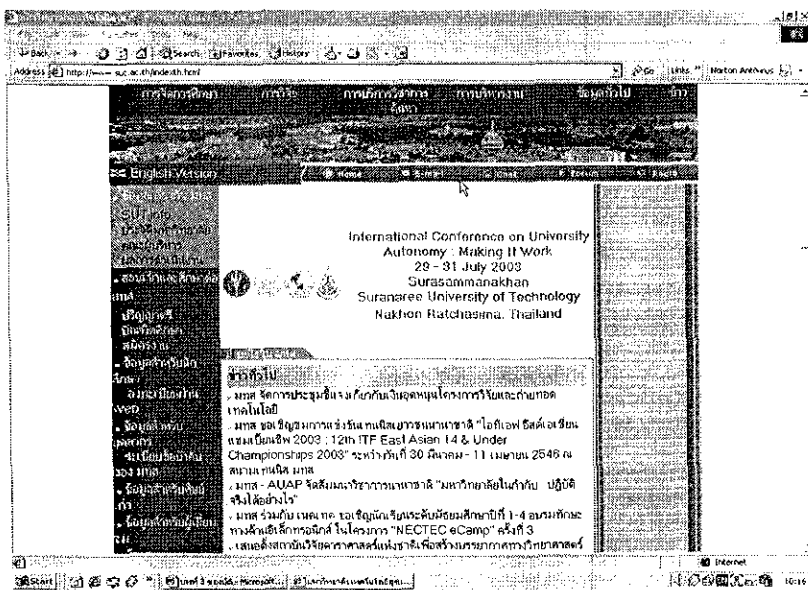
3. พูดคำว่า “เน็ต” ต่อจากนั้นสามารถใช้คำสั่ง “คียคาวนั” เพื่อเลือก address

http://www.sut.ac.th/indexth.html



รูปที่3.25 แสดงผลการใช้คำสั่ง “คียคาวนั”

4. พูดคำว่า “เอ็นเทอร์” เพื่อเป็นการตกลง



รูปที่ 3.26 แสดงผลการใช้คำสั่ง “เอ็นเทอร์”

3.4 ซีดจำกัดและข้อควรระวังของโปรแกรม

1. สามารถใช้คำสั่งได้เฉพาะในส่วนที่กำหนดไว้เท่านั้น
2. ในส่วนของเท็กซ์โหมด จะมีปัญหาในการออกเสียงตัวอักษรภาษาอังกฤษที่เสียงใกล้เคียงกันเช่น ตัว H และ ตัว S ซึ่งในบางครั้งอาจเกิดความผิดพลาดได้
3. ความแม่นยำในการใช้งานเมาส์
4. ระหว่างใช้งานควรอยู่ในที่ที่ไม่มีเสียงรบกวนเพราะเสียงรบกวนจะทำให้โปรแกรมทำงานผิดพลาดได้
5. เสียงของสุภาพสตรีใช้งานได้ไม่ตื้นัก เนื่องจากเป็นเสียงแหลมที่มีความถี่สูง ซึ่งที่ความถี่สูงสัญญาณรบกวนรบกวนจะสูงตามด้วย

บทที่ 4

บทสรุป

4.1 สรุปโดยรวมของโครงการเมาส์สั่งงานด้วยเสียงพูด

โครงการเมาส์สั่งงานด้วยเสียงพูด มีวัตถุประสงค์เพื่อต้องการนำโปรแกรมภาษา Visual Basic มาประยุกต์ใช้งานเพื่อสั่งงานเมาส์ด้วยเสียงพูด และสามารถช่วยเหลือผู้พิการ โดยเฉพาะผู้พิการทางแขนได้ หลังจากติดตั้งโปรแกรมนี้เสร็จเรียบร้อยแล้ว สามารถใช้งานโปรแกรมนี้ได้จากการ double click ที่ Icon หรือที่ start menu -> programs -> SUT -> Speech Mouse แบ่งการทำงานของโปรแกรม ออกได้ 3 ส่วนคือ Mouse Mode, Option Mode และ Text Mode

ในส่วนของ Mouse Mode จะเป็นการเคลื่อนที่ของเมาส์เคอร์เซอร์ ซึ่งจะเคลื่อนที่ขึ้น ลง ซ้าย ขวา และหยุดได้ตามที่สั่งงาน นอกจากนี้ยังมีการคลิกขวา คลิกซ้ายและดับเบิลคลิก ซึ่งการทำงานในโหมดนี้จะคล้ายกับการทำงานของเมาส์พื้นฐานทั่วไป ในส่วนของ Option Mode จะเป็นส่วนของการปรับความเร็วของการเคลื่อนที่ของเมาส์เคอร์เซอร์ซึ่งมี 3 ระดับคือช้า ปานกลาง และเร็ว ในส่วนของ Text Mode จะเป็นส่วนของการพิมพ์ตัวอักษรภาษาอังกฤษ และการเลื่อนหน้า ซึ่งในส่วนนี้จะเป็นการทำงานคล้ายการทำงานของคีย์บอร์ด

4.2 ประโยชน์ของโครงการเมาส์สั่งงานด้วยเสียงพูด

จากการทำโครงการทั้งหมดสามารถสรุปผลได้ดังนี้

1. สามารถนำโปรแกรมภาษา Visual Basic มาประยุกต์ใช้งานได้
2. ในส่วนของโปรแกรมประยุกต์ที่สร้างขึ้นสามารถสั่งงานเมาส์ด้วยเสียงพูดได้ครบทุกการใช้งานของเมาส์ปกติอย่างสมบูรณ์
3. สามารถนำโปรแกรมประยุกต์ที่สร้างขึ้นนี้ไปใช้จริง โดยเฉพาะอย่างยิ่งเพื่อช่วยเหลือผู้พิการแขน
4. โปรแกรมที่สร้างขึ้นนี้สามารถประหยัดการใช้เมาส์ทำให้ประหยัดค่าใช้จ่ายได้ส่วนหนึ่ง

4.3 ปัญหาอุปสรรคและข้อเสนอแนะ

1. ในส่วนของโปรแกรม Microsoft SAPI 4 ซึ่งสนับสนุนการใช้งานในภาษาอังกฤษ ทำให้ในส่วนของโปรแกรมไม่สามารถใช้งานในภาษาไทยได้ทั้งหมด ซึ่งจะได้มีการพัฒนาต่อไป
2. การใช้งานโปรแกรมนั้นต้องใช้คำสั่งที่มีการออกเสียงที่แตกต่างกัน ดังนั้นในบางคำสั่งจึงไม่สื่อความหมายกับการใช้งานได้ตรงกันนัก

เอกสารอ้างอิง

BrianLong(2545).**SpeechSynthesis&SpeechRecognition** .ในระหว่างวันที่25 ก.พ. – 10 มี.ค. 2546,
จาก <http://www.blong.com/Conferences/DCon2002/Speech/Speech.htm>

Microsoft Corporation (2545). **Microsoft Speech Technologies**; ในระหว่างวันที่ 15 ม.ค. – 9 ก.พ.
2546, จากhttp://www.microsoft.com/speech/evaluation/techover/#_rec_over

Power Plus Marketing Services Co.,Ltd.(2544). **How it work Speech Recognition**; ในระหว่างวันที่
20ก.พ. – 5 มี.ค. 2546, จาก <http://www.ready2e.com/scoops/sc119/speech-recognition2.asp>

การใช้งาน **Visual Basic**. ในระหว่างวันที่ 5ก.พ. – 20มี.ค.2546, จาก
www.rilp.ac.th/~nukit/document/vbles02.doc

การใช้งาน **Window API**. ในระหว่างวันที่ 5ม.ค. – 2 มี.ค.2546, จาก
<http://www.siamnets.com/kasem/chapter14.htm>

ฐนียา สัตยพานิช (2544). ระบบรู้จำเสียงภาษาไทยชนิดปรับตัวเข้ากับผู้ใช้แบบเรียลไทม์; ในระหว่าง
วันที่ 9 – 10 เม.ย., จาก<http://master.cpe.ku.ac.th/~g4465009/seminar/proposal.doc>

ธนาคม ตาพวัฒน์(2543). **Browser สำหรับคนพิการที่ไม่สามารถใช้ Keyboard** ได้; ในระหว่างวันที่
25 ก.พ. – 10 มี.ค. 2546, จาก<http://www.geocities.com/kom2002/report.html>

ประพนธ์ อัครวานุวัฒน์ (2545). **Windows API** .ในระหว่างวันที่ 10 ม.ค. – 2 มี.ค. 2546 ,
จาก http://micro.se-ed.com/content/mc209/MC209_97.asp#3

ส.สมไชย. (2545). การใช้งาน **Application Programming Interface (API)**; ในระหว่างวันที่ 10 ม.ค. –
2 มี.ค. 2546, จากhttp://www25.brinkster.com/vbthaiteam/tut_api.html