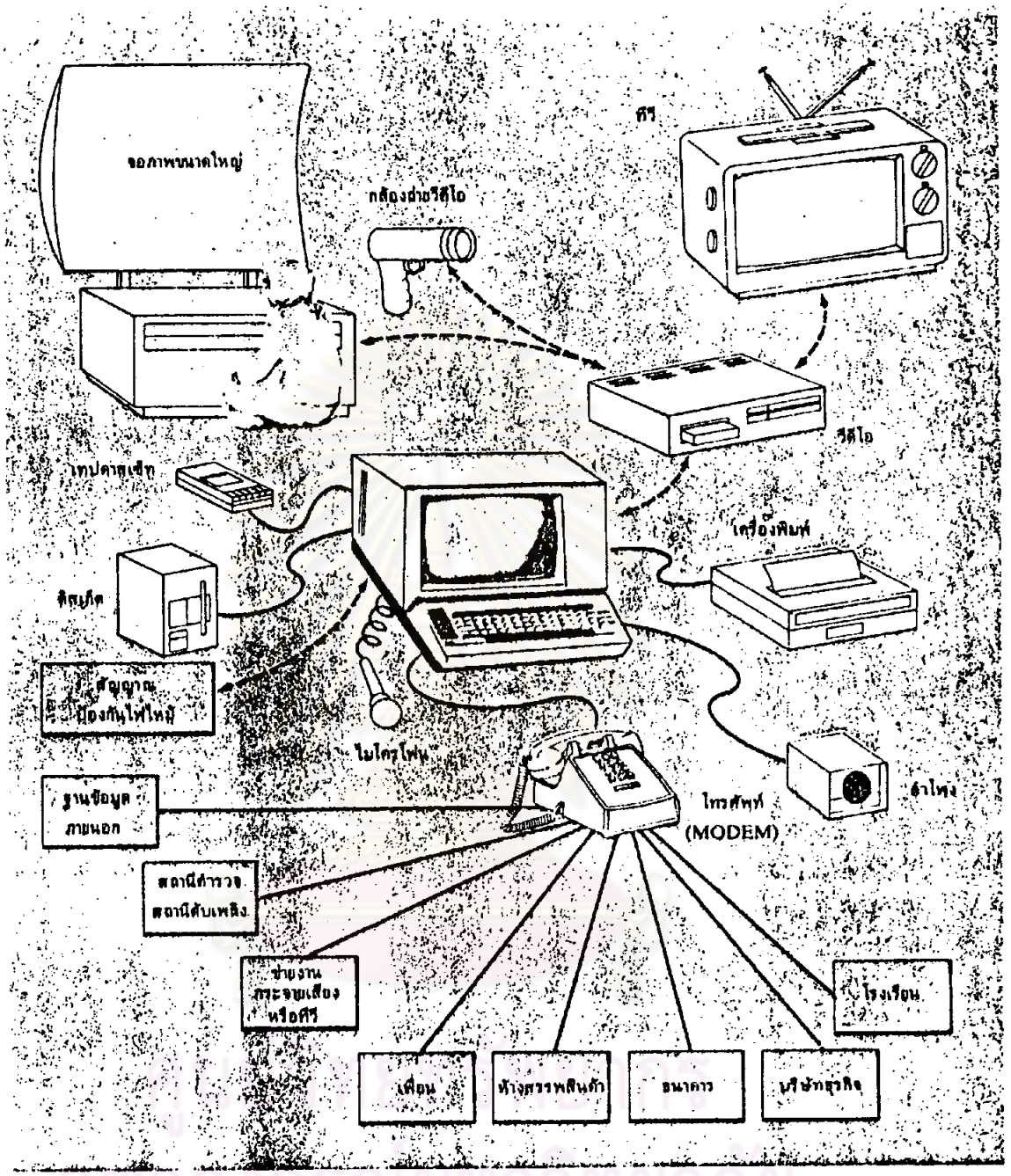
บทที่ 4
การสื่อสารข้อมูลระหว่างไมโครคอมพิวเตอร์และเครื่องควบคุม

ในปัจจุบันนี้ คอมพิวเตอร์สามารถส่งข้อมูลไปยังที่ต่างๆซึ่งไกลออกไปได้โดยไม่ยากนัก และสิ่งเหล่านี้เริ่มเข้ามามีบทบาทในชีวิตประจำวันมากขึ้น ตัวอย่างเช่นการฝากหรือถอนเงินในธนาคารต่างสาขาโดยใช้ออนไลน์คอมพิวเตอร์ การจองตั๋วเครื่องบิน ซึ่งคนที่กรุงเทพหรือฮ่องกงสามารถจองที่นั่งในเที่ยวบินเดียวกันได้ และนอกจากนี้ในต่างประเทศเริ่มมีการใช้จดหมายอิเล็ก-ทรอนิกส์ (Electronic Mail) ซึ่งผู้ใช้สามารถส่งข้อความไปยังอีกผู้หนึ่งได้รวดเร็วกว่าส่งทางจดหมายหรือเทเล็กซ์ (Telex) มาก หรือวิดีโอเท็กซ์ (Video Text) ผู้ที่ได้เป็นสมาชิกจะได้รับข่าวสารต่างๆที่ทันสมัยอยู่เสมอ เช่น อัตราการแลกเปลี่ยนเงินตราต่างประเทศ ราคาตลาดหุ้น ข่าวทั่วไป หรือแม้แต่จะใช้สอนภาษาให้กับแม่บ้าน เป็นต้น จะเห็นได้ว่าการสื่อสารข้อมูลเหล่านี้จะทวีความสำคัญยิ่งขึ้น ในอนาคตเมื่อมีการพัฒนาอุปกรณ์ต่างๆที่ใช้เป็นระบบดิจิตอลให้ดีแล้ว จะมีการรวมเอาสัญญาณต่างๆลงในสายเส้นเดียวกัน รูปที่ 4.1 เป็นรูปที่แสดงถึงการเชื่อมการสื่อสารข้อมูลหลายๆอย่างเข้าด้วยกัน เพราะฉะนั้นในอนาคตอาจจะมีสายเข้าในบ้านเพียงเส้นเดียว แล้วสามารถใช้โทรศัพท์ โทรทัศน์ระบบดิจิตอล คอมพิวเตอร์ วีดีโอเท็กซ์ ร่วมกันได้

4.1 การส่งข้อมูลแบบอนุกรม และ แบบขนาน

ขนาดของเครื่องคอมพิวเตอร์ทั่วไปที่ใช้กันอยู่นี้ โดยเฉพาะไมโครคอมพิวเตอร์ จะเรียกว่าขนาด 8 บิท 16 บิท หรือ 32 บิทนั้น พิจารณาได้จากการรับส่งข้อมูลของตัวซีพียู ถ้าซีพียูสามารถส่งข้อมูลได้ครั้งละ 8 บิท จะเรียกว่าคอมพิวเตอร์ขนาด 8 บิท เช่น เครื่องแอปเปิล -ทู อะตาริ (ATARI) เป็นต้น แต่ถ้าคอมพิวเตอร์มีซีพียูที่ทำงานด้วยขนาด 16 บิท ภายในตัวเองแต่กลับส่งข้อมูลออกมาเพียง 8 บิท จะเรียกเครื่องเหล่านี้ว่า 16 บิทเทียม ได้แก่เครื่องไอบีเอ็ม พีซี (IBM PC) เป็นต้น สิ่งเหล่านี้จะเป็นตัวบอกว่าเครื่องคอมพิวเตอร์สามารถทำงานได้รวดเร็วหรือไม่ เพราะถ้ายิ่งจำนวนบิทที่ซีพียูสามารถควบคุมได้ยิ่งมาก จะทำให้การทำงานยิ่งรวดเร็วขึ้นตามลำดับ แต่เมื่อมีการส่งข้อมูลระหว่างคอมพิวเตอร์ 2 ตัว ที่อยู่ไกลกันออกไป ก็จำเป็นที่ต้องใช้สายยาวเท่ากับระยะทางที่ห่างกัน ในสภาวะเช่นนี้ราคาของสายจะมีผลมากสำหรับระบบ ดังนั้นจำเป็นต้องลดค่าใช้จ่ายลงโดยใช้สายเพียงเส้นเดียว เพราะฉะนั้นจึงเกิดแนวความคิดเกี่ยวกับการส่งข้อมูลแบบอนุกรมขึ้น 2 วิธี คือ



รูปที่ 4.1 แสดงถึงการเชื่อมการสื่อสารข้อมูลหลายอย่างเข้าด้วยกัน

-แบบอะซิงโครนัส (Asynchronous)

-แบบซิงค์โครนัส (Synchronous)

4.1.1 การส่งแบบอะซิงค์โครนัส (Asynchronous)

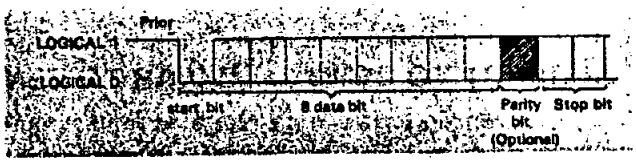
เป็นวิธีที่ใช้กันมากที่สุดในเครื่องไมโครคอมพิวเตอร์ เพราะสามารถใช้งานได้ง่าย แต่มีข้อเสียคือส่งได้ช้ากว่าแบบซิงค์โครนัส ซึ่งจะกล่าวรายละเอียดภายหลัง สาเหตุที่การส่งแบบอะซิงค์โครนัสช้ากว่าก็คือทุกๆตัวอักษรที่ส่งออกไป จะต้องมีสตาร์ทบิท (Start bit) พาริตี้บิท (Parity bit) และสต๊อบบิท (Stop bit) รวมอยู่ด้วย ก่อนที่จะอธิบายต่อไป ขอทำความรู้จักกับคำว่า บอดเรท (Baud Rate) ซึ่งมีหน่วยเป็นบิทต่อวินาที ปัจจุบันมีหลายขนาดเช่น 300 600 1200 จนถึง 19200 บิทต่อวินาที หรือต่ำกว่าก็มีแต่ไม่นิยมใช้ สำหรับการส่งข้อมูลผ่านทางสายโทรศัพท์ในเมืองไทยนี้นิยมใช้ที่ 1200 และ 2400 บิท ต่อวินาที ถ้าสูงกว่านี้อาจจะเกิดการผิดพลาดของข้อมูลได้ง่าย เมื่อมีการรบกวนเกิดขึ้น และในการส่งแต่ละครั้งจำเป็นต้องระบุความเร็วที่แน่นอนให้กับคอมพิวเตอร์ด้วย สิ่งต่อมาที่ควรจะทำความเข้าใจคือ สตาร์บิท พาริตี้บิทและสต๊อบบิท ในขณะที่คอมพิวเตอร์ยังไม่ได้เริ่มส่งข้อมูล ในสายส่งจะมีสภาวะเป็น 1 ดังรูปที่ 4.2 และเมื่อเริ่มส่งข้อมูล จะมีบิทที่บอกว่าเริ่มต้นส่งข้อมูลแล้ว มีสภาวะเป็น 0 เรียกว่าสตาร์ทบิท แล้วตามด้วยข้อมูลซึ่งมีความยาวได้ตั้งแต่ 5-8 บิท ตามด้วยพาริตี้บิท ที่ใช้สำหรับตรวจสอบความถูกต้องของข้อมูล อาจจะมีหรือไม่มีก็ได้ โดยที่พาริตี้บิทมี 2 ชนิด คือ พาริตี้บิทคี่และคู่ เช่นถ้าเลือกพาริตี้บิทคี่ คอมพิวเตอร์ทางด้านรับจะตรวจสอบความถูกต้องของข้อมูล จากจำนวนบิทที่เป็น 1 ว่ารวมกันแล้วเป็นเลขคี่ หรือไม่ ส่วนสต๊อบบิท เป็นบิทที่น้อยที่สุดใช้คั่นจนกว่าถึงข้อมูลชุดต่อไป โดยที่ไม่ทำให้เกิดการผิดพลาดของข้อมูล อาจจะมี 1 1.5 หรือ 2 บิท สิ่งเหล่านี้เป็นสิ่งจำเป็นสำหรับการส่งข้อมูลในแต่ละครั้งจะไม่กำหนดไม่ได้ ดังนั้นเมื่อรวมสิ่งต่างๆเข้ากับข้อมูลแล้วอาจจะยาวถึง 11 บิท สำหรับข้อมูลเพียง 8 บิท ทำให้อัตราการส่งข้อมูลช้าลง สำหรับการส่งแบบอะซิงค์โครนัสนี้มีใช้กันมาก ซึ่งอาจจะแบ่งออกได้เป็น 3 วิธีใหญ่ๆ คือ

1. RS-232C

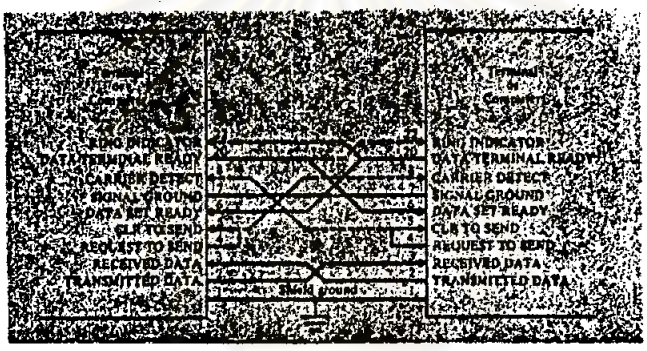
2. 20 mA Current loop

3. RS-422, RS-423 & RS-449

คำว่า RS นี้ย่อมาจากคำว่า "Recommended Standard" ซึ่งเป็นมาตรฐานที่ตั้งขึ้นโดย Electronic Industries Association หรือ EIA ซึ่งในแต่ละแบบจะมีข้อดีหรือข้อเสียแตกต่างกันไป

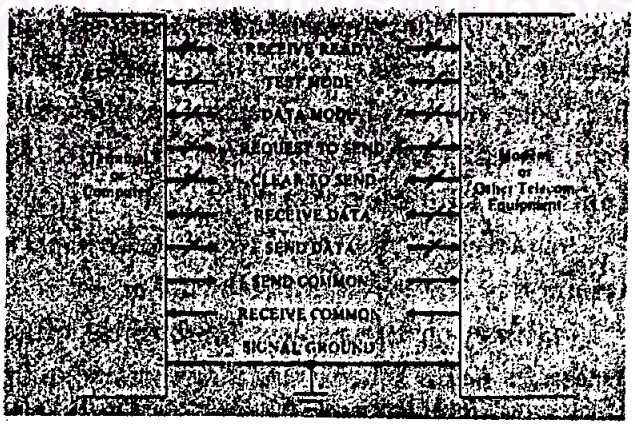


รูปที่ 4.2 แสดงภาพการส่งแบบ อาร์ เอส 232 ซี



หอสมุดกลาง สถาบันวิทยบริการ จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 4.3 แสดงภาพการต่อในลักษณะของ อาร์ เอส 232 ซี



รูปที่ 4.4 แสดงภาพการต่อในลักษณะของ อาร์ เอส 449

อาร์เอส-232ซี" (RS-232C)

เป็นระบบที่ใช้กันมากในปัจจุบัน ดังนั้นเครื่องไมโครคอมพิวเตอร์ส่วนมากจะมีส่วน RS-232C นี้ให้ใช้สำหรับสื่อสารข้อมูลกับคอมพิวเตอร์อื่นๆ เดิมทีนั้น RS-232C ถูกพัฒนาขึ้นเพื่อใช้สื่อสารข้อมูลผ่าน ทางสายโทรศัพท์ ซึ่งทำให้สามารถส่งข้อมูลไปได้ไกลยิ่งขึ้น การจะส่งข้อมูลผ่านทางสายโทรศัพท์นี้จำเป็นต้องมีอุปกรณ์ที่ใช้สำหรับแปลงสัญญาณให้เหมาะสมกับสายโทรศัพท์ อุปกรณ์นี้เรียกว่า โมเด็ม (MODEM) ซึ่งย่อมาจาก Modulator & Demodulator โมเด็มนี้จะทำหน้าที่แปลงสัญญาณดิจิตอล ซึ่งมี 0 และ 1 เป็นสัญญาณอนาลอกที่มีความถี่สูง และต่ำแทน ตัวอย่างเช่น อาจจะแปลงให้ 1 เป็นความถี่ 1025 Hz และ 0 เป็นความถี่ 1250 Hz การนำสัญญาณกลับมาทางปลายทางก็ใช้วิธีตรวจจับความถี่ ถ้าพบความถี่ 1025 Hz ก็ให้เป็น 1 เป็นต้น เนื่องจากสัญญาณอนาลอกเหมาะสมกับสายโทรศัพท์มากกว่าสัญญาณดิจิตอล สำหรับโครงสร้างของ RS-232C มีสายสัญญาณอยู่ 20 เส้น ดังรูปที่ 4.3 จากรูปเป็นการส่งข้อมูลระหว่างคอมพิวเตอร์ 2 เครื่อง มีบางเส้นที่เป็นเส้นประ (Dotted line) หมายความว่าใช้สำหรับ CARRIER DETECT ปกติไม่จำเป็นต้องใช้ และถ้าพิจารณาต่อไปจะพบว่า TRANSMITTED DATA ของข้างหนึ่งจะต่อกับ RECEIVED DATA อีกข้างหนึ่ง และยังมีอีกบางคู่ต่อสลับกันดังที่กล่าวมาแล้ว สำหรับหน้าที่ของสัญญาณแต่ละเส้นสามารถอธิบายได้ดังนี้ REQUEST TO SEND และ CLEAR TO SEND เมื่อคอมพิวเตอร์ต้องการส่งข้อมูล จะต้องตรวจดูก่อนว่าคอมพิวเตอร์อีกฝ่ายพร้อมที่จะรับหรือไม่โดยการส่งสัญญาณ REQUEST TO SEND ออกไป ถ้าอีกฝ่ายพร้อมที่จะรับข้อมูลก็จะส่งสัญญาณ CLEAR TO SEND กลับมาให้ฝ่ายส่ง เมื่อฝ่ายส่งได้รับ CLEAR TO SEND ก็จะเริ่มส่งข้อมูลให้ฝ่ายรับ ทั้งนี้เพื่อป้องการการผิดพลาด เช่น คอมพิวเตอร์ฝ่ายรับก็ต้องการส่งข้อมูลด้วย ถ้าไม่มีการตรวจสอบแล้วต่างฝ่ายต่างก็ส่งข้อมูล ข้อมูลที่ส่งออกมาจาก 2 เครื่องในเวลาเดียวกันนี้ก็จะเกิดการผิดพลาดทันที การตรวจสอบนี้จำเป็นสำหรับสายส่งประเภทฮาล์ฟ-ดิวเพล็กซ์ (Half-duplex) (สำหรับฮาล์ฟดิว-เพล็กซ์ มีรายละเอียดอยู่ใน "ประเภทของสาย") แต่ในปัจจุบัน สัญญาณ REQ TO SEND และ CLE TO SEND อาจจะไม่ต้องใช้ถ้าสายที่ใช้สำหรับรับส่งข้อมูลเป็นแบบฟูลดูเพล็กซ์ (full-duplex) (รายละเอียดของฟูลดูเพล็กซ์อยู่ใน "ประเภทของสาย") เพราะสายชนิดนี้สามารถส่งข้อมูลพร้อมกันทั้ง 2 ฝ่ายได้ จึงไม่จำเป็นต้องมีการตรวจสอบอีก

DATA SET READY ใช้ตรวจสอบว่าเครื่องคอมพิวเตอร์ฝ่ายรับพร้อมที่จะทำงานแล้วหรือยังบางครั้งอาจจะต่อสัญญาณนี้กับส่วนเปิดปิดเครื่องคอมพิวเตอร์ฝ่ายรับด้วย ดังนั้นเมื่อเครื่องคอมพิวเตอร์เปิดสวิทช์ทำงานก็จะส่งสัญญาณเพื่อบอกว่าพร้อมที่จะทำงานแล้ว

DATA TERMNINAL READY ทำหน้าที่เช่นเดียวกับ DATA SET READY แต่ใช้สำหรับตรวจสอบความพร้อมของคอมพิวเตอร์ฝ่ายส่งแทน

CARRIER DETECT & RING INDICATOR 2 สัญญาณนี้เกี่ยวข้องกับการทำงานของโทรศัพท์คือเมื่อมีการโทรศัพท์ติดต่อไปยังฝ่ายรับจะมีเสียงกริ่งโทรศัพท์ (Ring tone) ดังขึ้น สัญญาณกริ่งนี้จะไปกระตุ้นให้คอมพิวเตอร์ฝ่ายรับให้รู้ว่ามีการติดต่อเข้ามา ให้คอมพิวเตอร์ฝ่ายรับพร้อมที่จะรับข้อมูลเข้าไปด้วยส่วนสัญญาณ CARRIER DETECT เป็นการบอกให้ทราบว่าการติดต่อนี้ยังกระทำอยู่หรือไม่ ถ้าการติดต่อยุติลงด้วยเหตุผลใดก็ตาม สัญญาณ CARRIER DETECT จะขาดหายไปด้วย และการขาดหายไปของสัญญาณดังกล่าวจะทำให้คอมพิวเตอร์รู้ว่าการติดต่อได้ยุติลงแล้ว

สำหรับการติดต่อของเครื่องคอมพิวเตอร์ 2 เครื่องโดยตรงไม่ผ่านโมเด็มแบบง่ายๆทำได้ดังแสดงในรูปที่ 4.3 ส่วนระดับสัญญาณไฟฟ้าที่เป็นมาตรฐานของ RS-232C นั้นกำหนดไว้ว่าเป็น 15 โวลท์ สำหรับลอจิก 0 และ -15 โวลท์ สำหรับลอจิก 1 (บางครั้งจะเรียกว่า มาร์ค (Mark) และ สเปซ (Space) สำหรับลอจิก 1 และ 0 ตามลำดับ) แต่ในการปฏิบัติอาจจะใช้อยู่ในช่วง +/-12 ถึง +/-15 โวลท์ ก็ได้ ทั้งนี้เพื่อป้องกันการรบกวน (Noise) อันเนื่องจากการทำงานของมอเตอร์ และสิ่งที่ทำให้เกิดการสปาก (Spark) ของไฟฟ้า หรือสิ่งอื่นๆ ที่ให้ผลเช่นเดียวกัน ตัวอย่างที่ยกมานี้ตามมาตรฐานแล้วระบุไว้ว่า ระยะระหว่างเครื่อง 2 เครื่องที่ไม่เกิดการรบกวนข้อมูล คือระยะไม่เกิน 100 ฟุต ถ้าต้องการส่งไกลกว่านี้ก็จำเป็นต้องใช้โมเด็มเข้าช่วย หรือใช้ RS-422, RS-423 & RS-449 ซึ่งสามารถส่งได้ไกลกว่าแทนก็ได้

4.1.2 การส่งแบบซิงค์โครนัส (Synchronous)

การส่งข้อมูลแบบซิงค์โครนัส ใช้เวลาในการควบคุมการส่งข้อมูล ซึ่งจะมีข้อดีกว่าแบบอะซิงค์โครนัส คือ

1. เนื่องจากการรับส่งข้อมูลจะบังคับเวลาที่ใช้สำหรับข้อมูลแต่ละตัว ดังนั้นจึงไม่จำเป็นต้องมีสตาร์ทและสต๊อบบิท เพราะสามารถคำนวณได้ว่าข้อมูลแต่ละตัวจะใช้เวลาเท่าไรทำให้จำนวนบิทที่ใช้ส่งสำหรับข้อมูลแต่ละตัวน้อยลงไปด้วย ผลก็คืออัตราความเร็วในการส่งจะเพิ่มขึ้น

2. การส่งข้อมูลของซิงค์โครนัสนี้จะส่งข้อมูลเป็นชุด ในแต่ละชุดจะประกอบด้วยข้อมูลหลายๆตัวอักษรเรียกบล็อค (Block) ทำให้ประหยัดเพราะใช้ตัวควบคุมตัวเดียวกันทั้งชุด และตรวจสอบจำนวนตัวที่ใช้ในการควบคุมการส่ง ตัวเช็คข้อมูลมีผลให้ประสิทธิภาพการส่งข้อมูลสูงขึ้น

3. สามารถรับส่งข้อมูลในอัตราความเร็วสูงกว่าและส่งได้ไกลกว่าแบบอะซิงค์โครนัส ถ้าเปรียบเทียบด้านความเร็ว อะซิงค์โครนัสสามารถส่งได้เร็วที่สุดเพียง 19.2 KHz (กิโลเฮิร์ช) ในขณะที่ซิงค์โครนัสสามารถส่งได้ 500 KHz และถ้าจะส่งได้เร็วถึงขนาดนี้อะซิงค์-โครนัสต้องใช้สายโคเอ็กเชียล ซึ่งเป็นสายชนิดพิเศษมีราคาแพง หรือ RS-422 ดังที่ได้อธิบายในหัวข้อก่อนนี้ และเนื่องจากซิงค์โครนัสไม่ต้องการสตาร์ทบิท และสต๊อบบิท จึงจำเป็นจะต้องสร้างตัวที่ใช้ตรวจสอบความพร้อมในการรับส่งข้อมูล เรียกว่าคาแรคเตอร์ ซิงค์โครไนเซชัน (Character Synchronization) หรือเรียกว่าตัวซิงค์ (Synchronous)

4.2 ประเภทของสาย

นอกจากโปรโตคอลจะมีบทบาทในการสื่อสารข้อมูลแล้ว ชนิดของสายยังมีผลต่อการทำงานด้วยเช่นกัน สายที่ใช้กันจะมีอยู่ 3 ประเภทด้วยกัน

1. ซิมเบิล-ดิวเพล็กซ์ (Simple-Duplex)

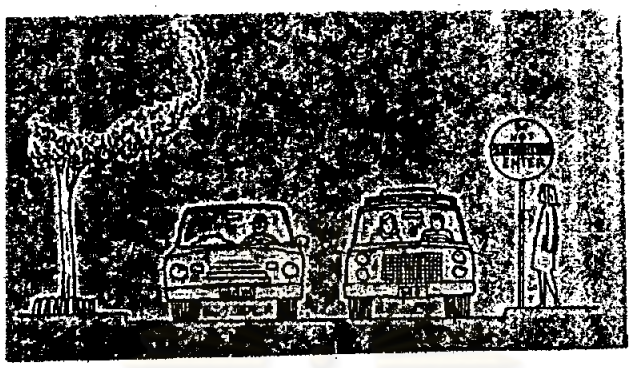
2. ฮาร์ฟ-ดิวเพล็กซ์ (Half-Duplex)

3. ฟูล-ดิวเพล็กซ์ (Full-Duplex)

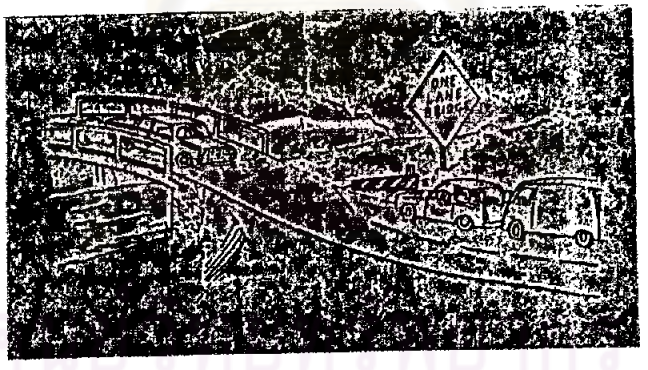
การทำงานของสายทั้ง 3 ชนิด เปรียบเหมือนกับการขับรถบนถนนคือ ซิมเบิล-ดิวเพล็กซ์จะเหมือนกับการขับรถทางเดียว รูปที่ 4.5 หมายความว่าข้อมูลสามารถส่งไปได้เพียงทิศทางใดทิศทางหนึ่งเท่านั้น เพราะฉะนั้นถ้าจะรับ/ส่งข้อมูลจะต้องใช้สายถึง 2 คู่สาย คือคู่สายหนึ่งทำหน้าที่ส่งข้อมูลเพียงอย่างเดียว อีกคู่สายหนึ่งก็จะรับข้อมูลเพียงอย่างเดียว สำหรับสายชนิดนี้ได้เลิกใช้ไปแล้ว เพราะไม่ประหยัดค่าใช้จ่าย

ฮาร์ฟ-ดิวเพล็กซ์ เปรียบได้กับการขับรถทางเดียวเหมือนกัน แต่กำหนดเวลาในการเปลี่ยนทิศทางด้วย เช่น ในคู่สายจะใช้ส่งเป็นเวลา 1 วินาที และสลับกับการรับอีก 1 วินาที เช่นกัน จะสลับไปมาเช่นนี้ตลอดไป ทำให้สามารถประหยัดจำนวนคู่สายจาก 2 คู่สายเหลือเพียง 1 คู่สายได้ ดังรูปที่ 4.6

ฟูล-ดิวเพล็กซ์ เป็นสายที่ปรับปรุงและพัฒนาข้อจำกัดให้ดีขึ้น คือสามารถส่งข้อมูลพร้อมกันทั้ง 2 ข้างได้ เหมือนกับรันเวย์ (Run way) ที่สามารถวิ่งสวนไปมาบนถนนเดียวกัน



รูปที่ 4.5 แสดงภาพเปรียบเทียบการส่งแบบSIMPLEX



รูปที่ 4.6 แสดงภาพเปรียบเทียบการส่งแบบ HALF-DULPLEX

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4.3 วิธีการเชื่อมต่อระหว่างเครื่องควบคุมและไมโครคอมพิวเตอร์

ตามที่ได้กล่าวมาข้างต้นแล้วว่า การติดต่อสื่อสารข้อมูลสามารถกระทำได้หลายวิธี เช่น RS-232C ,RS-422 และอื่นๆ โดยแต่ละวิธีมีข้อดีและข้อเสียแตกต่างกันไป สำหรับในหัวข้อนี้จะกล่าวถึงการเชื่อมต่อระหว่างเครื่องควบคุมและไมโครคอมพิวเตอร์ผ่านระบบบัส RS-232C การที่เลือกระบบนี้ในการติดต่อส่งผ่านข้อมูลเนื่องมาจาก เป็นระบบที่ใช้โดยทั่วไปเป็นมาตราฐาน ใช้งานได้ง่าย และอุปกรณ์ที่ใช้สามารถหาได้ นอกจากนั้นอัตราการรับส่งข้อมูลก็อยู่ในช่วงที่ต้องการ คือ มีอัตราการรับส่งได้ตั้งแต่ 300 จนถึง 19200 บิทต่อวินาที ความยาวของสายส่งประมาณไม่เกิน 100 ฟุต และเป็นระบบที่เครื่องคอมพิวเตอร์ส่วนใหญ่มีอยู่แล้ว (Built in) เป็นมาตรฐาน

4.3.1 ข้อกำหนดของระบบสื่อสารข้อมูล

เพื่อให้ระบบเป็นมาตรฐาน สามารถใช้ได้กับอุปกรณ์ประกอบอื่นๆ จึงได้กำหนดมาตรฐานของระบบไว้ดังนี้

-ความเร็วในการรับส่งข้อมูล 2400 บิทต่อวินาที

-ข้อมูลจะส่งเป็นชนิด 8 บิทข้อมูล

-มี Start bit 1 บิท

-มี Stop bit 1 บิท

-มี Parity bit เป็นชนิด Even

-การรับส่งข้อมูลเป็นชนิด 2 ทิศทาง ใช้สายส่ง 3 เส้น คือ Tx,Rxและ Ground

4.3.2 ระบบการสื่อสารข้อมูลบนเครื่องควบคุม

สำหรับเครื่องควบคุมการส่งข้อมูลในระบบ RS-232C จะใช้ชิพ ไอ.ซี เบอร์ 8251 ซึ่งเป็นไอ.ซี ที่สามารถใช้ส่งในลักษณะของRS-232Cได้ โดยมีขาสัญญาณต่างๆครบถ้วน เนื่องจากขารับและขาส่งของ 8251 มีระดับสัญญาณ 0 และ 5 โวลต์เท่านั้น ดังนั้นจึงต้องใช้ ไอ.ซี.เบอร์ 1488 และ 1489 เพื่อเปลี่ยนระดับแรงดันให้เป็น +/-12 โวลต์ สำหรับวงจรดูได้จากรูปที่ 3.6 ในบทที่ 3 การสั่งให้ไอ.ซี ทำได้โดยการใช้โปรแกรมซึ่งสามารถโปรแกรมจำนวน Start bit, Stop bit,Parity bit และจำนวนบิทข้อมูลได้ แต่สำหรับค่า Baud rate จะต้องใช้สัญญาณนาฬิกาป้อนเข้าไป โดยกำหนดความถี่ของสัญญาณนาฬิกา เพื่อคำนวณให้ได้ค่า Baud rate ตามที่ต้องการ

4.3.3 ระบบการสื่อสารข้อมูลบนเครื่องไมโครคอมพิวเตอร์

โดยทั่วไปแล้วเครื่องไมโครคอมพิวเตอร์สามารถที่จะโปรแกรมค่าต่างๆ ในระบบ RS-232C ได้อยู่แล้ว การนำมาเชื่อมต่อกับเครื่องควบคุมก็สามารถทำได้โดยกำหนดค่าต่างๆให้ตรงกันตามข้อกำหนด ก็สามารถติดต่อสื่อสารกันได้ ไม่ว่าจะเป็นเครื่องคอมพิวเตอร์ชนิดใด ที่จะกล่าวถึงรายละเอียดในหัวข้อนี้จะขอกล่าวถึง การโปรแกรมไมโครคอมพิวเตอร์ ยี่ห้อ Basis Medfly ซึ่งเป็นเครื่องที่คล้ายกับไมโครคอมพิวเตอร์ Apple II[14] ให้สามารถรับส่งข้อมูลกับเครื่องควบคุมได้ บนเครื่องไมโครคอมพิวเตอร์ Medfly มีระบบ RS-232C อยู่แล้วโดยใช้ ไอ.ซี. เบอร์ 6551 เป็นตัวควบคุม ไอ.ซี.เบอร์นี้เป็น ACIA (Asynchronous Communication Interface Adapter) โดยสามารถโปรแกรม อัตราการส่งข้อมูลได้ตั้งแต่ 50-19200 บิทต่อวินาที กำหนดความยาวของข้อมูลได้ตั้งแต่ 5,6,7 และ 8 บิท สำหรับ Stop bit จะโปรแกรมได้ตั้งแต่ 1, 1 1/2 และ 2 บิท เป็นต้น นอกจากนั้นยังสามารถกำหนด Parity bit ให้เป็นชนิด Even หรือ Odd ได้ ที่กล่าวมานี้สามารถทำได้โดยการใช้โปรแกรม สำหรับตำแหน่งและ รีจีสเตอร์ต่างๆ ของ ไอ.ซี.เบอร์นี้บนเครื่อง Medfly ดูได้จากตารางที่ 4.1

6551 Addresses

ADDRESS	WRITE	READ
$C098	Transmit Data Register	Receive Data Register
$C099	Program Reset*	Status Register
$C09A	Command Register	Command Register
$C09B	Control Register	Control Register

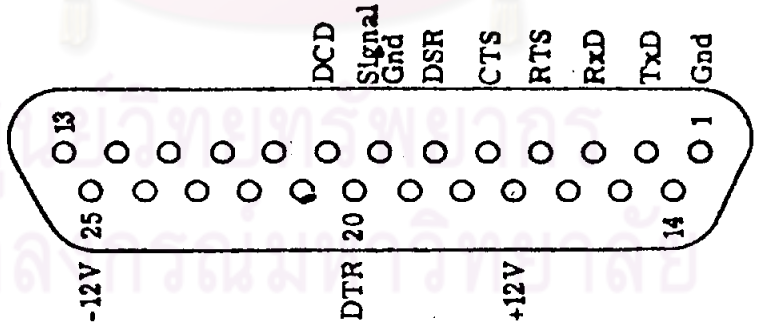
ตารางที่ 4.1 ตารางแสดงตำแหน่งของรีจีสเตอร์ต่างๆของ ไอ.ซี.เบอร์6551

RS232 Parameter Addresses

FUNCTION	DOS BASIC	UCSD II.1.1. (6502 addressing)	UCSD IV.0	CP/M (Z-80 addressing)
6551 Control Register (baudrate, word length, stopbits)	$06F9	$FFCE	$0271	$F280
6551 Command Register (Parity, RTS, DTR)	$0779	$FFCF	$0270	$F282
Input:	IN#9*	REMIN: #7:	REMIN: #7:	UR1:
Output	PR#9	REMOUT: #8:	REMOUT: #8:	UL1: UP1:

* POKE 41153,10 enables the use of the PR#9 and IN#9 commands with DOS.

ตารางที่ 4.2 ตารางแสดงตำแหน่งของการเรียกใช้ไอ.ซี.เบอร์ 6551 ในกรณีที่ใช้โปรแกรมควบคุมต่างกัน



รูปที่ 4.7 แสดงหัวต่อชนิด DB-25 ที่ใช้ต่อ RS-232C ระหว่างเครื่องควบคุมและไมโครคอมพิวเตอร์

เนื่องจากไมโครคอมพิวเตอร์ Medfly สามารถใช้โปรแกรมควบคุมได้หลายระบบ เช่น UCSD,CP/M และอื่นๆเป็นต้น ดังนั้นในแต่ละระบบการอ้างถึงตำแหน่งของรีจีสเตอร์ต่างๆจึงแตกต่างกันไป ตารางที่ 4.2 จะแสดงถึงตำแหน่งของ Parameter Address ของ RS-232C ในกรณีที่ใช้โปรแกรมควบคุมต่างกัน

สำหรับรูปที่4.7 จะแสดงหัวต่อชนิด DB-25 ที่ใช้บนเครื่องไมโครคอมพิวเตอร์ Medfly

4.3.4 โปรโตคอลที่ใช้สำหรับการสื่อสารข้อมูล

ตามที่ได้กล่าวมาแล้วข้างต้นจะเห็นว่า โดยทั่วไปแล้วการส่งข้อมูลในลักษณะเช่นนี้ มักจะต้องมีรหัสซึ่งทราบกันว่าเป็นตัวกำหนดการติดต่อระหว่างกัน เครื่องซึ่งสามารถติดต่อกันได้จะต้องทราบรหัสของกันและกัน ซึ่งรหัสตัวนี้เราจะเรียกว่า โปรโตคอล สำหรับโปรโตคอลที่ใช้ในเครื่องควบคุมนี้ สามารถดูได้จากตารางที่ 4.3

FUNCTION	CODE	COMMENT
ENQUIRE FOR WEIGHT	W	ขอดูข้อมูลจากการชั่งน้ำหนักครั้งที่ผ่านมา
ENQUIRE FOR STATUS	S	ขอดูสถานะของตัวควบคุม
SEND FORMULAR	F	บอกเครื่องควบคุมว่าจะส่งสูตร
SEND OFFSET	O	บอกเครื่องควบคุมว่าจะส่งออฟเซ็ท
SEND START COMMAND	T	บอกเครื่องให้เริ่มทำงาน
SEND STOP COMMAND	P	บอกเครื่องให้หยุดทำงาน
SEND ALARM ACK	K	บอกเครื่องว่ารับทราบสัญญาณเตือนแล้ว
ACK	A	ตอบรับการรับ-ส่ง ข้อมูล
NO-ACK	N	ไม่ตอบรับหรือไม่พร้อม

ตารางที่ 4.3 ตารางแสดงระหัสที่ใช้ติดต่อระหว่างไมโครคอมพิวเตอร์และเครื่องควบคุม

จากตารางจะเห็นว่าในกรณีที่เครื่องควบคุม หรือไมโครคอมพิวเตอร์ต้องการข้อมูลอะไร ก็จะส่งรหัสตัวนั้นออกมา เช่นเมื่อไมโครคอมพิวเตอร์ต้องการส่งค่า Offset ก็จะส่ง 'o' ไปก่อน เครื่องควบคุมก็จะรู้ว่าเครื่องไมโครคอมพิวเตอร์ต้องการส่งค่า Offset มาให้ ถ้าพร้อมก็จะตอบรับด้วย 'A' (Acknolage) หลังจากนั้นไมโครคอมพิวเตอร์ก็จะส่งค่า Offset ไปให้ แล้วปิดท้ายข้อมูลด้วย Line feed เครื่องควบคุมก็จะรู้ว่าข้อมูลครบแล้ว ถ้าทุกอย่างเรียบร้อยก็จะตอบกลับด้วย 'A' ถ้ามีข้อผิดพลาดก็จะตอบกลับด้วย'N' ดังนี้เป็นต้น

4.3.5 รูปแบบของการรับส่งข้อมูล

รูปแบบของการรับส่งข้อมูลสามารถแบ่งออกได้เป็น 4 วิธีด้วยกันคือ

-การรับส่งข้อมูลชนิด 1 Byte และตอบกลับด้วย 1 Byte เช่นกัน เช่น Start,Stop และ Alarm Acknowledge

-การรับส่งข้อมูลชนิดอนุกรมเป็นชุด เช่น formular, Offset

-การส่งข้อมูล 1 Byte เพื่อขอข้อมูลเป็นชุด เช่น Enquire for Status,Enquire for Weight

-รอรับข้อมูลทางเดียว เช่น รับข้อมูลที่ได้จากการทำงานของเครื่องควบคุม

รูปที่ 4.8 จะแสดงถึงการรับและส่งข้อมูลในทั้ง 4 กรณีที่กล่าวข้างต้น

นอกจากนั้นแล้วถ้าจะแบ่งตามลักษณะวิธีใช้ จะสามารถแบ่งออกได้เป็น 4 ชนิดด้วยกันคือ

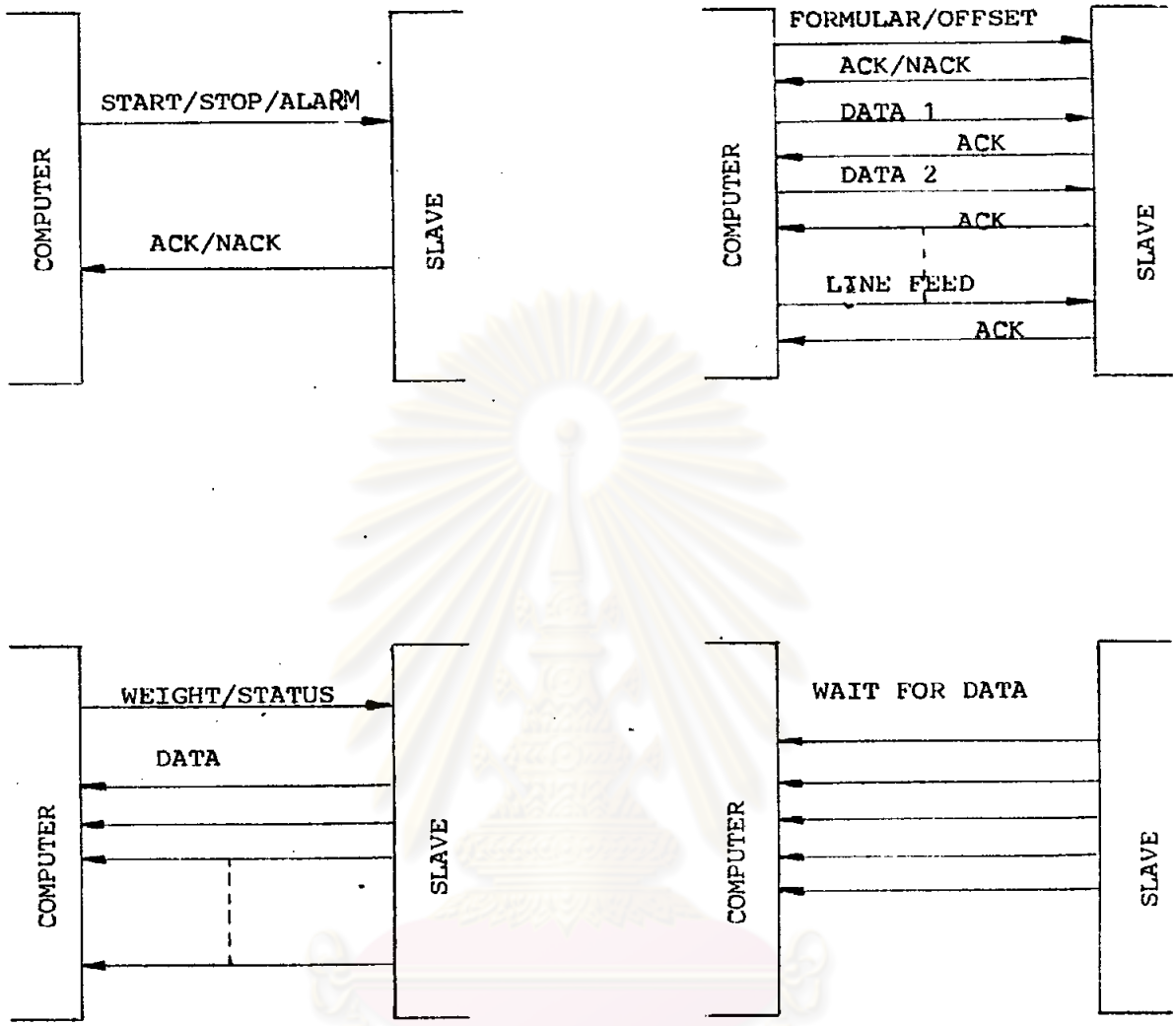
-ชนิดคำสั่ง เช่น Start,Stop

-ชนิดคำตอบรับ เช่น ACK,NACK

-ชนิดคำร้องขอที่จะรับข้อมูล เช่น Enquire for Status,Enquire for weight

-ชนิดคำร้องขอที่จะส่งข้อมูล เช่น Send Formular,Send Offset เป็นต้น

จุดสำคัญอีกจุดหนึ่งที่จะต้องทราบก็คือ การทำงานของระบบ RS-232C บนเครื่องควบคุมจะใช้วิธีการอินเตอร์รัพ ส่วนการทำงานบนไมโครคอมพิวเตอร์จะเป็นการกวาดดู ดังนั้นเครื่องควบคุมจะสามารถตอบรับคำสั่งได้ตลอดเวลา ในขณะที่ไมโครคอมพิวเตอร์จะรับได้เฉพาะเมื่อมีการตรวจสอบ



รูปที่ 4.8 แสดงรูปแบบการติดต่อระหว่างคอมพิวเตอร์และเครื่องควบคุม

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

4.3.6 โปรแกรมรับส่งข้อมูลบนเครื่องไมโครคอมพิวเตอร์

โปรแกรมนี้จะถูกเขียนขึ้นด้วยภาษาเครื่อง(Machine)ของ CPU เบอร์ 6502[12] โดยโปรแกรมนี้จะใช้ในการควบคุมการทำงานของไอ.ซี.เบอร์ 6551 ลักษณะของการทำงานของโปรแกรมจะเป็นทางเข้า (Entry Point) ซึ่งมีอยู่ด้วยกัน 5 ทางคือ

-INIT 6551 เป็นการโปรแกรมใช้ 6551 ทำงานตามเงื่อนไขที่กำหนด

-Tx เป็นการส่งข้อมูล 1 ตัวจาก buffer ที่กำหนด

-Rx เป็นการรับข้อมูล 1 ตัวจาก 6551 แล้วนำไปเก็บยัง buffer ที่กำหนด

-Rdata เป็นการรับข้อมูลจาก 6551 แล้วนำไปเก็บยังตำแหน่งที่กำหนดต่อกันไปเรื่อยๆจนกว่าจะครบชุดข้อมูล

-Tdata ทำนองเดียวกับ Rdata แต่เป็นการส่งข้อมูล

4.4 โปรแกรมควบคุมการทำงานบนเครื่องไมโครคอมพิวเตอร์

โปรแกรมควบคุมจะแบ่งออกเป็น 2 ส่วนด้วยกันคือ ส่วนหนึ่งจะเป็นภาษาเครื่องและอีกส่วนหนึ่งจะเป็นภาษาเบสิก โปรแกรมภาษาเครื่องจะใช้สำหรับโปรแกรม 6551 ให้ทำงานตามที่กำหนด และส่วนหนึ่งจะใช้ในการส่งและรับข้อมูลทาง RS-232C โดยทั้งสองโปรแกรมนี้สามารถส่งผ่านข้อมูลกันได้ ตารางที่ 4.4 จะแสดงถึงตำแหน่งของ การเรียกใช้โปรแกรมภาษาเครื่อง

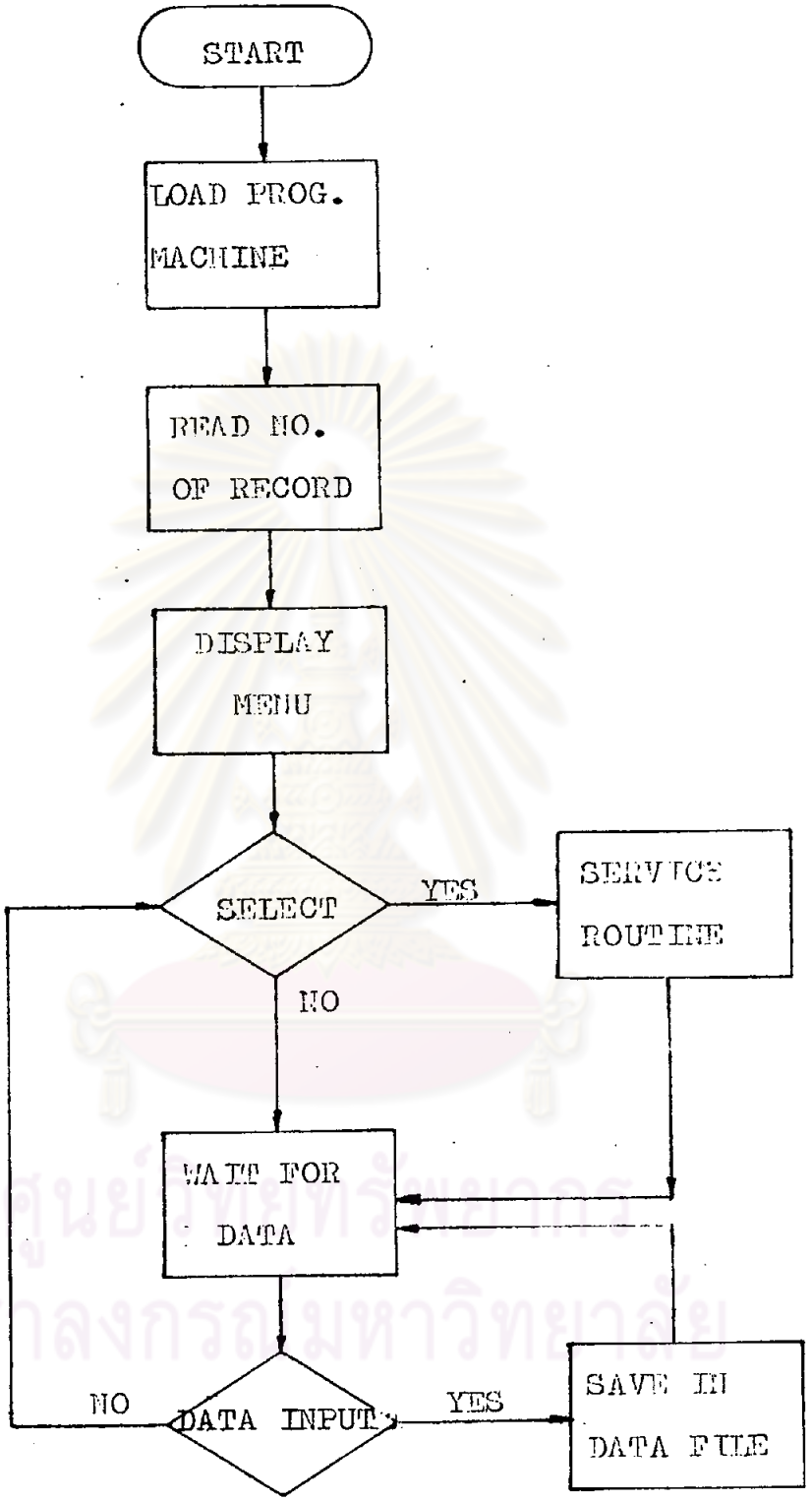
ส่วนโปรแกรมในภาษาเบสิกจะเป็นโปรแกรมที่ใช้สำหรับตอบสนองกับผู้ใช้ เก็บข้อมูล ประมวลผล กำหนดเงื่อนไขต่างๆ และรูปแบบในการทำงาน รูปที่ 4.9 แสดงถึงผังทำงานโดยย่อของโปรแกรม

สำหรับวิธีการเก็บข้อมูลและสูตรในโปรแกรมจะมีวิธีการต่างกันคือ การเก็บข้อมูลจะใช้การสร้างแฟ้มข้อมูลเป็นแบบแรนดอม(Random) โดยจะบันทึกจำนวนข้อมูลไว้ที่ตำแหน่ง 0 และเริ่มบันทึกข้อมูลจากตำแหน่งที่ 1 ต่อไปตามลำดับ ทุกครั้งที่มีการบันทึกข้อมูล จะมีการเพิ่มจำนวนข้อมูลในตำแหน่งที่ 0 ให้เท่ากับจำนวนข้อมูลเสมอ ข้อดีคือ ทำให้เวลาที่ใช้ในการเก็บข้อมูลไม่ขึ้นกับความยาวของข้อมูลเดิม ส่วนวิธีการเก็บสูตรจะเก็บในลักษณะของการสร้างแฟ้มข้อมูล แบบเรียงตามลำดับโดยเก็บเบอร์สูตรและสูตรต่อกันไปเรื่อยๆ ข้อดีของวิธีนี้คือ ใช้เนื้อที่ในหน่วยความจำน้อย ทำให้สามารถเก็บสูตรได้มากขึ้น

```
                1000 * RS232B VERSION
                1010 *--------------------------------
                1020 * RS-232C COMMUNICATION DRIVER ROUTINE
                1030 *--------------------------------
                1040 *
                1050 * DEF BUFFER AND CONSTANTS
C098:           1060 REG    .EQ $C098    ;UART 6551 REG ADDR
C099:           1070 STATUS .EQ $C099
C09A:           1080 CMDREG .EQ $C09A
C09B:           1090 CTRLRG .EQ $C09B
0308:           1100 PAR    .EQ $308     ;PARITY CHECK SUM BYTE
0309:           1110 ERCNT  .EQ $309     ;ERROR COUNT
0310:           1120 TXD    .EQ $310     ;TX DATA BUF
0311:           1130 RXD    .EQ $311     ;RX DATA BUF
0312:           1140 ERR    .EQ $312     ;ERROR BYTE
0313:           1150 WAIT   .EQ $313     ;TIME DELAY COUNT
8100:           1160 RBUF   .EQ $8100    ;RECEIVE DATA BUFFER
8100:           1170 TBUF   .EQ $8100    ;TX DATA BUFFER
                1180 *--------------------------------
                1190 * ENTRY POINT
                1200 *--------------------------------
                1210        .OR $8000
8000: 4C 0F 80  1220 J1     JMP INIT
8003: 4C 1F 80  1230 J2     JMP TX
8006: 4C 2F 80  1240 J3     JMP RX
8009: 4C 47 80  1250 J4     JMP RDATA
800C: 4C 71 80  1260 J5     JMP TDATA
```

ตารางที่ 4.4 แสดงตำแหน่งการเรียกใช้ภาษาเครื่อง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย



รูปที่ 4.9 แสดงผังทำงานโดยย่อของโปรแกรมควบคุม