

ระบบจัดการข้อมูลสำหรับผู้บริหารเพื่องานพิมพ์ธนบัตร

Management Information System for Bank Note Printing

ดร.กอบชัย เฉลยหาญ * วินัย ทองตัน ** พงษ์พันธ์ อักษร *** สุชาติ คุณทวีเทพ *

* คณะวิศวกรรมศาสตร์

** นักศึกษาระดับปริญญาโท บัณฑิตวิทยาลัย

และ หัวหน้าช่างไฟฟ้า โรงพิมพ์ธนบัตร ธนาคารแห่งประเทศไทย บางขุนพรหม ๑๐520

*** นักศึกษาระดับปริญญาโท บัณฑิตวิทยาลัย

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง ลาดกระบัง กรุงเทพฯ 10520

บทคัดย่อ

ระบบการตรวจสอบงานพิมพ์ธนบัตร (Note Printing) สำหรับโรงพิมพ์ธนบัตร (Note printing work) โดยการนำไมโครคอมพิวเตอร์มาช่วยในการตรวจสอบ ผ่านระบบซอฟต์แวร์และฮาร์ดแวร์ได้ถูกพัฒนาขึ้น เพื่อเป็นการเชื่อมโยงระหว่างคอมพิวเตอร์ส่วนบุคคลกับเครื่องพิมพ์ธนบัตร การพัฒนานี้เพื่อนำไปสู่การสร้างเครื่องต้นแบบ และเป็นรากฐานนำไปสู่การใช้งานในระบบจริง โดยราคาจะถูกกว่า 10 เท่าได้ทำการพัฒนาระบบซอฟต์แวร์ที่ใช้กับคอมพิวเตอร์ชนิด IBM หรือเทียบเท่าเพื่อทำการเชื่อมโยงกับระบบอิเล็กทรอนิกส์เพื่อการตรวจสอบยอดธนบัตรที่ได้ทำการพิมพ์ขึ้น อันจะเกิดประโยชน์ต่อผู้บริหารว่าต้องการจะตรวจสอบยอดเมื่อใดก็สามารถจะกระทำได้

บทนำ

ในงานพัฒนาระบบจัดการข้อมูลเพื่อช่วยการทำงาน ของผู้บริหารในงานพิมพ์ธนบัตรจำเป็นต้องสร้างเครื่องมือที่จะรับข้อมูลจากเครื่องพิมพ์ธนบัตร ทุกๆเครื่อง และนำข้อมูลที่ได้อมาประมวลผลพร้อมที่จะส่งข้อมูลถึงผู้บริหารอย่างรวดเร็วโดยผ่านเครื่องคอมพิวเตอร์ส่วนบุคคลเพื่อให้เกิดความสะดวก ในการวิจัยการทำงานของเครื่องพิมพ์ในโรงพิมพ์ธนบัตร ดังนั้นจึงเป็นจุดประสงค์ที่ต้องสร้างเครื่องจำลองการทำงาน (Simulator) เพื่อจะนำไปสร้างเครื่องรับข้อมูล (Data Gathering Panel,DGP) ที่เป็นระบบอิเล็กทรอนิกส์เพื่อการตรวจสอบยอดธนบัตร ที่ได้ทำการพิมพ์ขึ้นและข้อผิดพลาด เพื่อที่จะส่งข้อมูลไปยังเครื่องคอมพิวเตอร์ส่วนบุคคลของผู้บริหาร

RC4
HG
348
๑451

ส่วนประกอบของระบบจัดการข้อมูลที่ใช้ในงานพิมพ์ธนบัตร

(1) การทำงานของเครื่องรับข้อมูล (Data Gathering Panel,DGP)

DGP ที่เราต้องการพัฒนาขึ้นมา จะนำไปต่อกับเครื่องพิมพ์ธนบัตรแต่ละเครื่อง เพื่อรับข้อมูล โดยจะกำหนดตำแหน่ง (address) ให้กับ DGP แต่ละเครื่อง เมื่อทางเครื่องคอมพิวเตอร์ของผู้บริหาร มีความต้องการทราบข้อมูลของเครื่องพิมพ์เครื่องใดทางเครื่องคอมพิวเตอร์ของผู้บริหาร ก็จะส่งค่าตำแหน่งที่ต้องการมาทาง RS 232 Port เมื่อ DGP รับตำแหน่งที่ทางเครื่องคอมพิวเตอร์ของผู้บริหารต้องการแล้ว และตรงกับ DGP เครื่องใดทาง DGP เครื่องนั้น ก็จะส่งข้อมูลไปทางเครื่องคอมพิวเตอร์ของผู้บริหาร โดยผ่านทาง RS 232 Port ดังแสดงในรูปที่ 1

วงจร DGP จะสร้างจากไมโครคอนโทรลเลอร์ โดยใช้โรงงานนี้ใช้ชิพตระกูล MCS-51

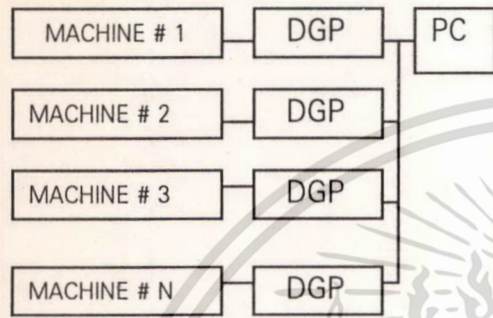
เลขหมู่.....

เลขที่เขียน..... 32237

เอกสารนี้จัดทำขึ้นเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่วัน, เดือน, ปี 11 ส.ค. 2542

เป็นส่วนประกอบของวงจร ภายใน DGP จะประกอบด้วย ส่วนของ CPU ที่ควบคุมการทำงานรับ-ส่งข้อมูลผ่านทาง Port RS 232 และการทำงานทั้งหมด ส่วนที่สองจะเป็นส่วนที่จัดเก็บข้อมูลจากเครื่องพิมพ์ ดังแสดงในรูปที่ 2.



รูปที่ 1.แผนภาพแสดงระรณการจัดการข้อมูลโดยใช้เครื่องรับข้อมูล

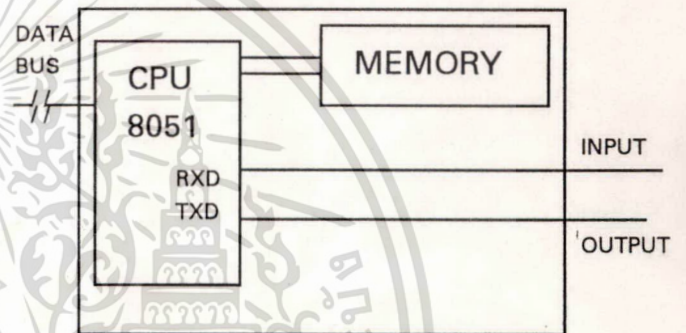
(2) เครื่องจำลองการทำงาน (Simulator)

จุดประสงค์ของการสร้าง Simulator คือเพื่อนำมาใช้แทนการทำงานของเครื่องพิมพ์ โดยการจำลองการทำงานของพิมพ์ชนบัตร ซึ่งจะนำ Simulator นี้มาติดตั้งแทนตำแหน่งของ machine แต่ละเครื่องของเครื่องพิมพ์ชนบัตร เพื่อใช้ในการพัฒนาสร้างเครื่องรับข้อมูล (DGP) และทดสอบการทำงาน ของเครื่องรับข้อมูล (DGP)

Simulator จะสร้างจากไมโครคอนโทรลเลอร์ ซึ่งในงานวิจัยนี้ได้ใช้ชิพ DS5000 ของ DALLAS SEMICONDUCTOR เพื่อควบคุมการทำงานทั้งหมด ไม่ว่าจะควบคุมการนับผลผลิตรวม (Total Product) ของจำนวนชนบัตรที่ได้ทำการพิมพ์ขึ้น ข้อผิดพลาดในการพิมพ์ (Error Code) หรือการควบคุมการส่งข้อมูลผ่านทาง Port RS232

ในการออกแบบส่วนประกอบที่ทำงานกับไมโครคอนโทรลเลอร์ สำหรับโครงการนี้ได้

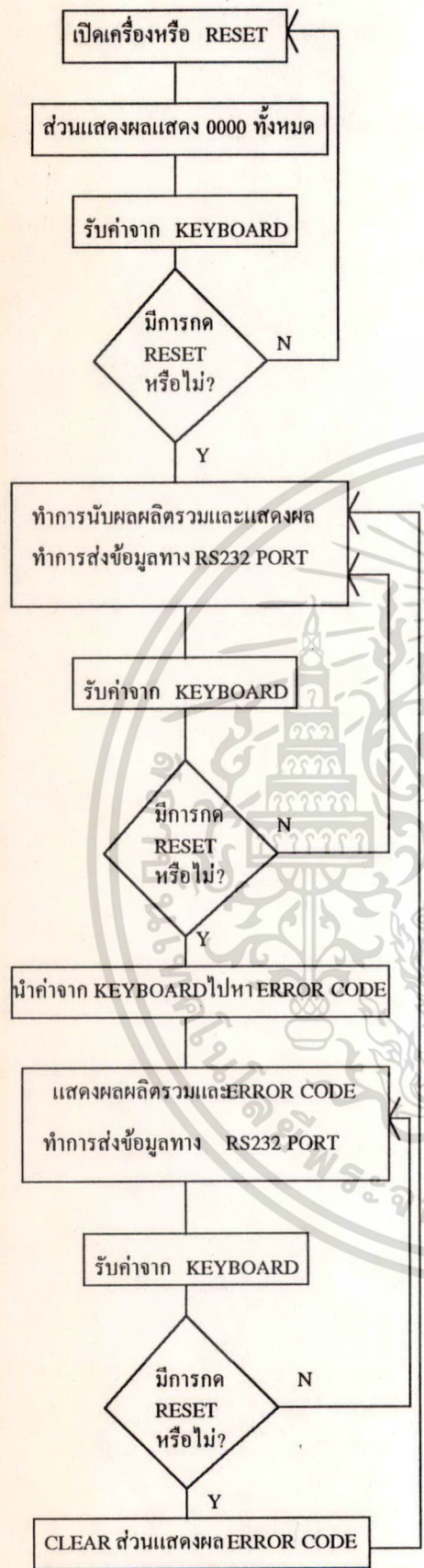
ใช้ชิพ 8255 โดยที่ชิพ 8255 จำนวน 1 ตัวจะขยายพอร์ตมาตรฐานออกเป็น 3 พอร์ต การทำงานของเครื่อง Simulator นี้จะใช้ 8255 จำนวน 2 ตัว โดยที่ตัวแรกจะใช้สำหรับ 6x3 Keyboard และส่วนแสดงผลของผลผลิตรวม (Total Product) ซึ่งเป็น 7-segment display จำนวน 4 หลัก 1 ชุด สำหรับ 8255 ตัวที่สองจะใช้สำหรับส่วนแสดงผลของ error code ซึ่งเป็น 7-segment display จำนวน 4 หลัก อีก 1 ชุด



รูปที่ 2.แผนภาพแสดงส่วนประกอบของเครื่อง DGP

(3) ขั้นตอนการทำงานของ Simulator

ชิพ DS 5000 ของ Dallas Semiconductor ที่ใช้พัฒนางานวิจัยนี้ เป็นไมโครคอนโทรลเลอร์แบบชิพเดี่ยว (Single Chip Microcontroller) คือไมโครคอมพิวเตอร์ แบบที่มีขนาดเล็กโดยบรรจุไว้ในแผงวงจรรวม (Integrated Circuit) เพียงชิพเดียวเหมาะสำหรับงานควบคุมอุปกรณ์อื่นๆ แบบอัตโนมัติ ซึ่งชิพ DS 5000 นี้เป็นชิพที่เทียบเท่ากับชิพตระกูล MCS-51 แต่พิเศษที่ภายในชิพจะมีหน่วยความจำที่สามารถเก็บ ข้อมูลและโปรแกรม (Embedded Program / Data Ram) ซึ่งสะดวกกว่าการพัฒนาด้วยชิพตระกูล MCS-51 คือไม่จำเป็นต้องใช้ EPROM



รูปที่ 3. แผนภูมิแสดงขั้นตอนการทำงานของเครื่องจำลองการทำงาน

ในงานวิจัยนี้ได้ออกแบบให้ชิพ DS 5000 ทำงานเป็นไปตามโฟลว์ชาร์ท (Flow Chart) ดังแสดงในรูปที่ 3. โดยทำการเขียนโปรแกรมควบคุม ภาษาแอสเซมบลี (AssemblyLanguage) และแปลให้เป็นภาษาเครื่อง (Machine Language) ด้วยโปรแกรม ครอส-32 (Cross-32 Assembler) ให้อยู่ในรูปแฟ้มข้อมูลแบบ Hexadecimal ก่อนจะนำเอาแฟ้มภาษาเครื่องนี้ไปเก็บใน หน่วยความจำของชิพ DS 5000 เมื่อชิพ DS 5000 ทำงานร่วมกับอุปกรณ์ภายนอก อื่นๆแล้วจะทำงานดังนี้

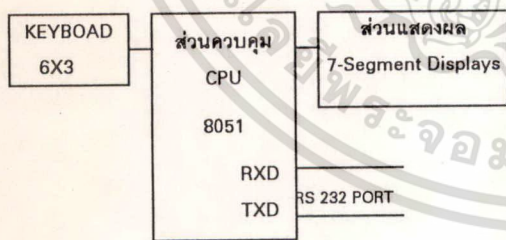
เมื่อเริ่มเปิดเครื่อง จะไม่ทำการนับผลผลิตรวม (Total Product) พื้นที่และส่วนแสดงผลผลิตรวม (Total Product) และ ส่วนแสดงผลข้อผิดพลาด (Error Code) จะแสดงผลเป็น"0000" ทั้งหมดจนกว่าจะกดแป้น Reset เมื่อทำการ Reset แล้วเครื่องจะเริ่มนับ และเครื่องจะนับไปจนกว่าจะเกิดข้อผิดพลาด (Error) ขึ้น (โดยการกดแป้นของ Keyboard) เมื่อเกิดข้อผิดพลาดขึ้นแล้วเครื่องจะหยุดนับทันที และส่วนแสดงผลข้อผิดพลาดจะแสดง Error Code ที่เกิดขึ้น และเมื่อแก้ไขข้อบกพร่องเรียบร้อยแล้วต้องการจะให้เครื่องทำงานต่อสามารถทำได้โดยกดแป้น Reset เมื่อทำการ Reset แล้วส่วนแสดงผลข้อผิดพลาดจะถูก Clear ให้เป็น"0000" และเครื่องจะทำการนับผลผลิตรวมต่อ ตลอดการทำงานของเครื่องจำลองการทำงาน (Simulator) จะส่งข้อมูลของผลผลิตรวม (Total Product) ของจำนวนรอบที่ ได้ทำการพิมพ์ขึ้น และ ข้อผิดพลาด Error Code ไปยังเครื่องคอมพิวเตอร์ส่วนบุคคลของผู้บริหารได้ตามต้องการ เมื่อเครื่องจำลองการทำงาน ได้รับหมายเลขเครื่องที่ต้องการ ที่ส่งจากเครื่องคอมพิวเตอร์ส่วนบุคคลของผู้บริหาร

(4) ซอฟต์แวร์ (Software) ในเครื่อง

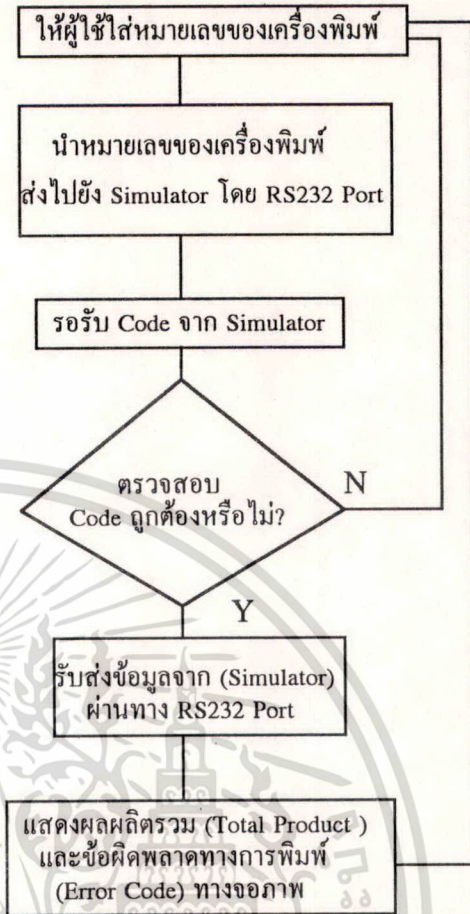
คอมพิวเตอร์ส่วนบุคคล

เพื่อให้เครื่องคอมพิวเตอร์ส่วนบุคคลทำงานร่วมกับระบบฮาร์ดแวร์ (Hardware) ต่างๆ จำเป็นต้องเขียนโปรแกรมควบคุม โดยใช้ภาษาซี เพื่อควบคุมให้เครื่องคอมพิวเตอร์ส่วนบุคคลทำงาน และรับส่งข้อมูล ตามโฟลว์ชาร์ท (Flow Chart) ในรูปที่ 5. เพื่อให้ทำงานดังนี้

เมื่อเริ่มทำงาน โปรแกรมจะให้ผู้ใช้ใส่หมายเลขของเครื่องพิมพ์ ที่ต้องการทราบข้อมูล หลังจากนั้นจะส่งหมายเลขเครื่องพิมพ์นี้ออกไปทาง RS 232 Port และรอรับข้อมูลจาก เครื่องจำลองการทำงาน (Simulator) ซึ่งทางเครื่องจำลองการทำงาน จะส่งรหัส (Code) มา ถ้ารหัสถูกต้อง ก็จะเริ่มรับส่งข้อมูล ของผลผลิตรวมและข้อผิดพลาดของการพิมพ์ โดยปกติแล้วถ้าเครื่องพิมพ์ทำงานปรกติ เมื่อเสร็จขั้นตอนทั้งหมดแล้วโปรแกรมจะกลับไปข้อผิดพลาดของการพิมพ์จะเป็น "0" แต่ถ้าเครื่องพิมพ์เกิดข้อผิดพลาด เครื่องคอมพิวเตอร์ส่วนบุคคล จะแสดงรายละเอียดของข้อผิดพลาดทางจอภาพ ให้ผู้ใช้ใส่หมายเลขเครื่องพิมพ์เพื่อเรียกดูข้อมูลครั้งต่อไป



รูปที่ 4. แผนภาพแสดงส่วนประกอบของเครื่องจำลองการทำงานของเครื่องพิมพ์ (Simulator)



รูปที่ 5. แผนภาพแสดงโฟลว์ชาร์ท (Flow Chart) การทำงานของซอฟต์แวร์ในเครื่องคอมพิวเตอร์ส่วนบุคคล ผลการวิจัย

ส่วนประกอบต่างๆของระบบจัดการข้อมูลในงานวิจัยนี้ เป็นเครื่องต้นแบบที่สร้างขึ้นมาเพื่อพัฒนาระบบจัดการข้อมูลต่อไปในอนาคต ซึ่งชุดต้นแบบประกอบตามลักษณะโครงสร้าง ในรูปที่ 6. ชุดต้นแบบที่ทดลองสร้างขึ้นมา มีลักษณะดังภาพภายในรูปที่ 7. เมื่อทำการเดินระบบ ส่วนแสดงผลทั้งหมด จะแสดงผลเป็น "0000" และจอภาพเครื่องคอมพิวเตอร์ส่วนบุคคลจะแสดงดังรูปที่ 8 ก. เมื่อทำการรีเซ็ต เครื่องจะเริ่มนับผลผลิตรวมซึ่งแสดงผลโดยส่วนแสดงผลผลผลิตรวม และส่วนแสดงผลข้อผิดพลาดจะแสดงผลเป็น "0000" และจอภาพเครื่องคอมพิวเตอร์ส่วนบุคคล จะแสดงดังในรูปที่ 8 ข.

ในสภาวะที่เกิดข้อผิดพลาด เครื่องจะหยุดนับผลผลิตรวม และส่วนแสดงผลข้อผิดพลาดจะแสดงผลตามข้อผิดพลาดที่เกิดขึ้น (ในชุดต้นแบบนี้ ข้อผิดพลาดเลือกได้โดยชุดคีย์บอร์ด) และจอภาพเครื่องคอมพิวเตอร์ส่วนบุคคลจะแสดงดังรูปที่ 8 ค. เมื่อแก้ไขข้อผิดพลาดแล้วต้องการให้เครื่องทำการนับต่อทำได้โดยการรีเซ็ต (โดยกดปุ่มรีเซ็ตจากชุดคีย์บอร์ด) ทางเครื่องคอมพิวเตอร์ส่วนบุคคล เมื่อต้องการทราบข้อมูลผลผลิตรวม หรือข้อผิดพลาด ทำได้โดยการคีย์หมายเลขเครื่อง ที่ต้องการทราบทางซอฟต์แวร์ถ้าหมายเลขเครื่องที่ต้องการทราบถูกต้องซอฟต์แวร์จะแสดงผลทั้งหมด และเวลาที่เรียกดูข้อมูล ทางจอภาพของเครื่องคอมพิวเตอร์ส่วนบุคคล

สรุปผลการวิจัย

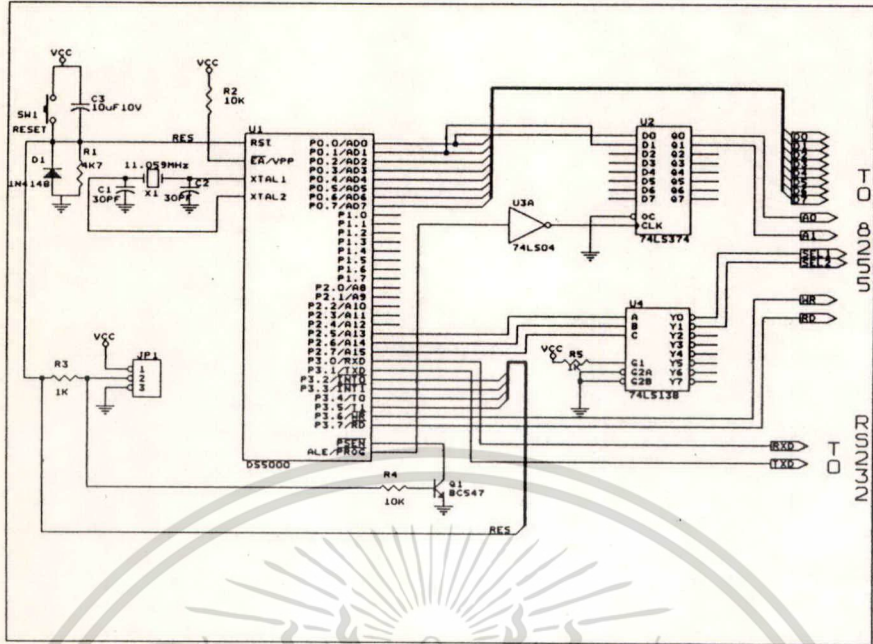
ชุดต้นแบบในงานวิจัยนี้ทำงานได้ตามที่ผู้วิจัยต้องการ ซึ่งผลการวิจัยนี้จะได้นำไปพัฒนาสร้างเครื่องรับข้อมูลจากเครื่องพิมพ์ธันบัตร ในโรงพิมพ์ธันบัตร เพื่อให้ใช้งานได้จริงในอนาคต

ข้อดีของงานวิจัยนี้คือ เป็นการดีที่สามารถนำเอาข้อมูลจากฮาร์ดแวร์หรือเครื่องที่ใช้ในการผลิตไปเก็บและประมวลผลบนเครื่องคอมพิวเตอร์ส่วนบุคคล

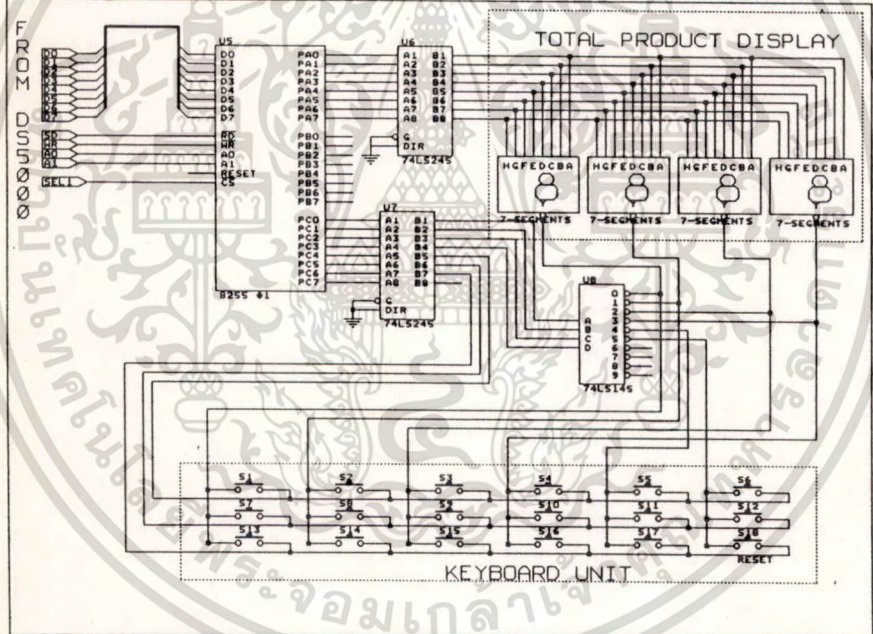
บุคคล ซึ่งจะเป็นประโยชน์ในการบริหารการผลิตมาก

เอกสารอ้างอิง

1. Dallas Semiconductor, DS 5000 User's Guide, Dallas Semiconductor
2. Intel, Microcontroller Handbook, Intel corporation, 1983
3. Intel, Embedded controller Handbook, Volume I 8 bit, Intel corporation, 1988
4. John B. Peatman, Design with Microcontroller, Mc Graw-Hill Book Company
5. PHILIPS, Microcontrollers and peripherals, PHILIPS Electronic and components and materials division, PHILIPS Co.
6. PHILIPS, Singnetics microcontroller User's Guide, PHILIPS Electronic and components and materials division, PHILIPS Co
7. สุเจตน์ จันทรัมย์, ไมโครคอนโทรลเลอร์ ชิปเดี่ยว 8051, 2535



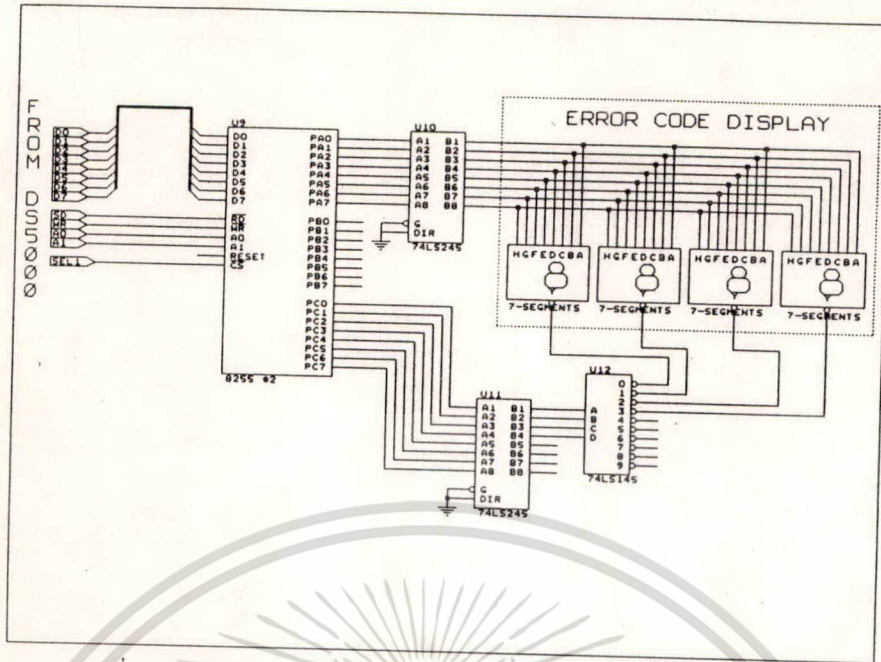
รูปที่ 6 ก. โค้ดแกรมแสดงส่วนประกอบของระบบจัดการข้อมูลในส่วนไมโครคอนโทรลเลอร์



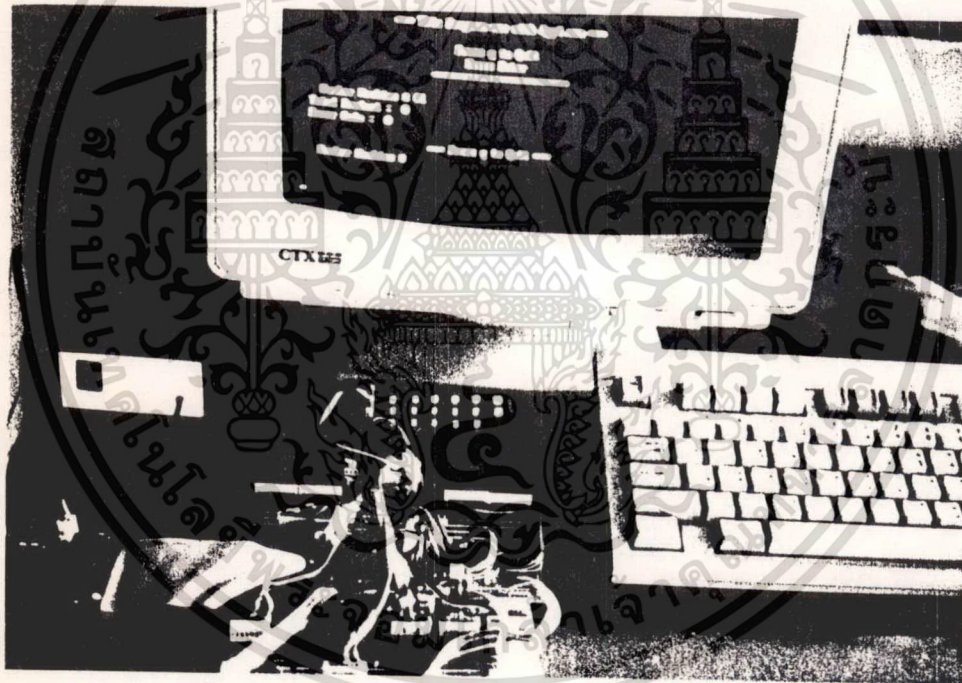
รูปที่ 6 ข. โค้ดแกรมแสดงส่วนประกอบของระบบจัดการข้อมูลในส่วนแสดงผลของผลคูณรวมและ 6X3 Keyboard โดยใช้ชิพ 8255 ตัวที่ 1 ขยายจากพอร์คของไมโครคอนโทรลเลอร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำนักหอสมุดกลาง พระจอมเกล้าลาดกระบัง



รูปที่ 6 ก. โค้ดแกรมแสดงส่วนประกอบของระบบจัดการข้อมูลในส่วนแสดงผลข้อผิดพลาด โดยใช้ชิพ 8255 ตัวที่ 2 ขยายจากพอร์ตของไมโครคอนโทรลเลอร์



รูปที่ 7. ภาพถ่ายชุดคั่นแบบระบบจัดการข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

*** This Program use with DS5000 chip ***
*** This Program use with dgp4.asm ***
Press q to Quit
Press Enter

Number Machine # 01
The current time is: 0:27:48.03
Total Product : 0
Error Code : 0

*** Press q to Quit ***
Number Machine #

รูปที่ 8 ก. แสดงผลการทดลองจากจอภาพเครื่องคอมพิวเตอร์ส่วนบุคคลเมื่อทำการเดินระบบ

*** This Program use with DS5000 chip ***
*** This Program use with dgp4.asm ***
Press q to Quit
Press Enter

Number Machine # 01
The current time is: 22:06:57.19
Total Product : 1063
Error Code : 0

*** Press q to Quit ***
Number Machine #

รูปที่ 8 ข. แสดงผลการทดลองจากจอภาพเครื่องคอมพิวเตอร์ส่วนบุคคลขณะทำงานปกติ

*** This Program use with DS5000 chip ***
*** This Program use with dgp4.asm ***
Press q to Quit
Press Enter

Number Machine # 01
The current time is: 3:08:57.22
Total Product : 1055
Error Code : 1
26b12 stop and lock at DT 1

*** Press q to Quit ***
Number Machine #

รูปที่ 8 ค. แสดงผลการทดลองจากจอภาพเครื่องคอมพิวเตอร์ส่วนบุคคลขณะเกิดข้อผิดพลาด

พลาด

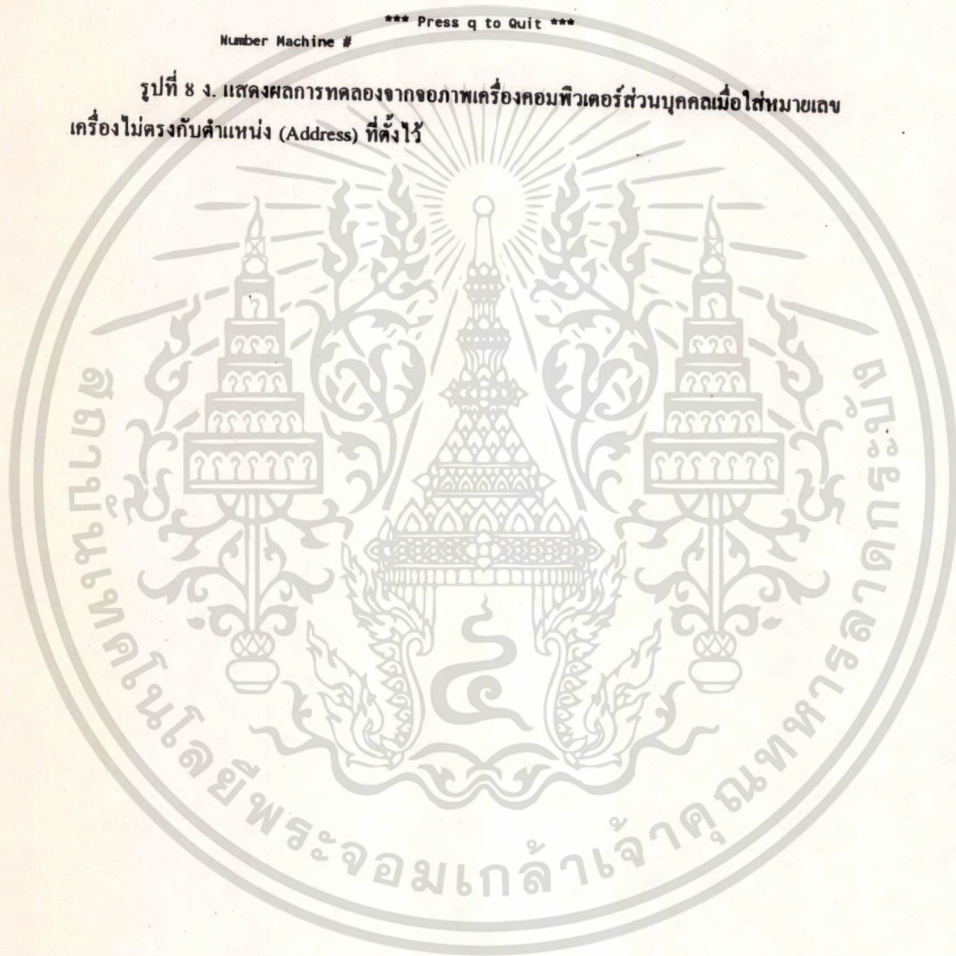
*** This Program use with DS5000 chip ***
*** This Program use with dgp4.asm ***

Press q to Quit
Press Enter

Number Machine # 00
The current time is: 3:17:38.96
Not Found Machine...

*** Press q to Quit ***
Number Machine #

รูปที่ 8 ง. แสดงผลการทดลองจากจอภาพเครื่องคอมพิวเตอร์ส่วนบุคคลเมื่อใส่หมายเลข
เครื่องไม่ตรงกับตำแหน่ง (Address) ที่ตั้งไว้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก ก.โปรแกรมภาษาแอสเซมบลี
ที่ใช้ควบคุมการทำงานของชิพ DS 5000

```

CPU "8051.TBL"          ERROR: EQU  24H
HOF "INT8"             FULT: EQU  26H
ORG 0000H             RSER: EQU  28H

B: EQU 0F0H ;B REGISTER      TMUL: EQU  2AH
ACC: EQU 0E0H ;ACCUMULATOR   RESET: EQU  2BH
SBUF: EQU 99H ;SEND BUFFER
SCON: EQU 98H ;SERIAL CONTROL *****
TH1: EQU 8DH ;TIMER 1 HIGH *****
TH0: EQU 8CH ;TIMER 0 HIGH    MOV  DPTR,#2
TL1: EQU 8BH ;TIMER 1 LOW     MOV  A,#0F0H
TL0: EQU 8AH ;TIMER 0 LOW     ORL  A,R5
TMOD: EQU 89H ;TIMER MODE     MOVX @DPTR,A
TCON: EQU 88H ;TIMER CONTROL  NOP
PCON: EQU 87H ;POWER CONTROL  NOP
; REGISTER                 NOP
DPH: EQU 83H ;DATA POINTER HIGH MOVX A,@DPTR
DPL: EQU 82H ;DATA POINTER LOW SWAP A
P0: EQU 80H ;PORT 0          ORL  A,#0F0H
P3: EQU 0B0H
P3.0: EQU 0B0H
CY: EQU 0D7H ;CARRY FLAG *****
AC: EQU 0D6H ;AUXILIARY-CARRY MMUL: MACRO
; FLAG                     MOV  TMUL,#00H
P: EQU 0D0H ;PARITY FLAG      MOV  R6,#0FFH
TI: EQU 099H ;TRANSMIT INTERRUPT CLR  C
; FLAG                     INC6: INC  R6
RI: EQU 098H ;RECEIVE INTERRUPT RRC  A
; FLAG                     JNC  INC6
TR1: EQU 08EH
TA: EQU 0C7H
IP: EQU 0B8H
PORTA: EQU 00H
PORTB: EQU 01H
PORTC: EQU 02H
BCDN: EQU 18H
BUFFER: EQU 20H
TEN: EQU 21H *****
HUN: EQU 22H             BCD: MACRO
SAND: EQU 23H           MOV  A,TEN

```


SCAN1: LCALL DISPLA

LCALL ST1

KEYSCAN

JZ SCAN0 ;NOT PRESS

;KEY PRESS

MOV R6,A

KEYSCAN

XRL A,R6

JNZ SCAN0

MOV A,R6

MMUL

MOV A,TMUL

CJNE A,#12H,CON

LJMP ESCAN1

CON: LCALL FILL

ESCAN1: LJMP SCAN0

ESCAN: RET

.....

FILL: MOV RESET,#00H

MOV R1,#ERROR

MOV A,TMUL

MOV DPTR,#ERTAB1

MOVC A,@A+DPTR

MOV @R1,A

INC R1

MOV A,TMUL

MOV DPTR,#ERTAB2

MOVC A,@A+DPTR

MOV @R1,A

INC R1

MOV A,TMUL

MOV DPTR,#FTAB1

MOVC A,@A+DPTR

MOV @R1,A

INC R1

MOV A,TMUL

MOV DPTR,#FTAB2

MOVC A,@A+DPTR

MOV @R1,A

MOV A,TMUL

MOV R5ER,A

.....

RRUN: MOV R5,#06

RSCAN0: DEC R5

CJNE R5,#0FFH,RSCAN1

LJMP RRUN

RSCAN1: LCALL DISPLA

LCALL ST1

KEYSCAN

JZ RSCAN0 ;NOT PRESS

;KEY PRESS

MOV R6,A

KEYSCAN

XRL A,R6

JNZ RSCAN0

MOV A,R6

MMUL

MOV A,TMUL

XRL A,#12H

JNZ RSCAN0

MOV TMUL,#00

MOV RESET,#0CH

MOV R1,#7

MOV R0,#ERROR

REFILL: MOV @R0,#00

INC R0

DJNZ R1,REFILL

RET

.....

DISPLA: MOV R0,#BUFFER

.....

MOV R1,#04H

MOV R2,#00H

LOOP1: MOV A,@R0

LCALL SHOW

INC R0

DJNZ R1,LOOP1

MOV R1,#04H

MOV R2,#00

LOOPE: MOV A,@R0

LCALL SHOWE

INC R0

DJNZ R1,LOOPE

RET

.....

SHOW: ANL A,#0FH

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

.....
MOV DPTR,#TABLE
MOVC A,@A+DPTR
MOV DPTR,#PORTA
MOVX @DPTR,A ;SEND SEGMENT
MOV A,R2
MOV DPTR,#PORTC
MOVX @DPTR,A ;SEND COMMON
DELAY
MOV A,#00
MOV DPTR,#PORTA
MOVX @DPTR,A ;OFF SEGMENT
MOV A,#06H
MOV DPTR,#PORTC
MOVX @DPTR,A ;OFF COMMON
INC R2
RET
.....
SHOWE: ANL A,#0FH
.....
MOV DPTR,#TABLE
MOVC A,@A+DPTR
MOV DPH,#80H
MOV DPL,#PORTA
MOVX @DPTR,A ;SEND SEGMENT
MOV A,R2
MOV DPL,#PORTC
MOVX @DPTR,A ;SEND COMMON
DELAY
MOV A,#00
MOV DPL,#PORTA
MOVX @DPTR,A ;OFF SEGMENT
MOV A,#06H
MOV DPL,#PORTC
MOVX @DPTR,A ;OFF COMMON
INC R2
RET
.....
ST1: MOV P3,#0FFH
DELAY
JB P3.0,PRS
LCALL RECI
XRL A,#01H
.....
JNZ EN1
MOV A,#63H
LCALL SEND
.....
MOV R0,#BUFFER
LCALL RECI
MOV A,@R0
LCALL SEND
INC R0
LCALL RECI
MOV A,@R0
LCALL SEND
INC R0
LCALL RECI
MOV A,@R0
LCALL SEND
INC R0
LCALL RECI
MOV A,@R0
LCALL SEND
INC R0
LCALL RECI
MOV A,@R0
LCALL SEND
INC R0
LCALL RECI
MOV A,@R0
LCALL SEND
INC R0
LJMP EN
.....
EN1: MOV A,#00H
LCALL SEND
EN: RET
.....
RECI: JNB RI,RECI
CLR RI
MOV A,SBUF
RET
.....
SEND: MOV R2,0FH
DLY1: DJNZ R2,DLY1
RESM2: JNB TI,RESM2
CLR TI
MOV SBUF,A
PRS: RET
.....
TABLE: DFB 3FH,06H,5BH,4FH,66H,6DH,
DFB 7DH,07H,7FH,6FH,77H,7CH,39H,

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

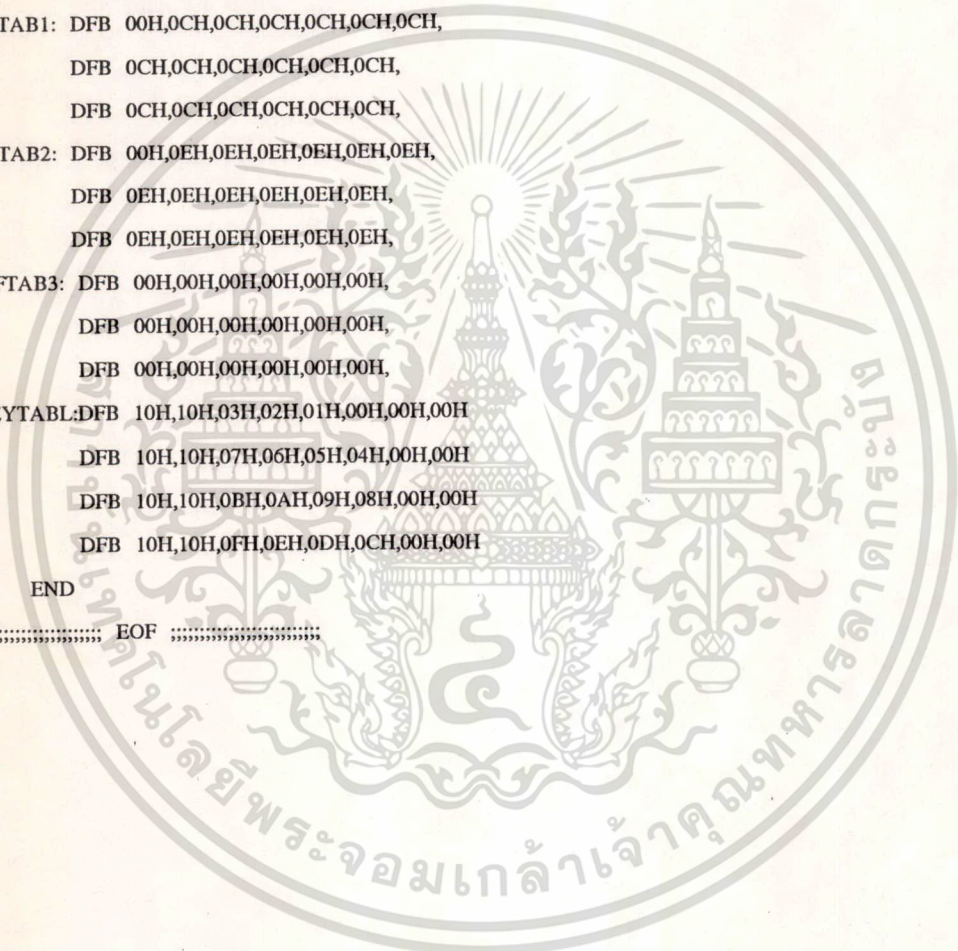
DFB 5EH,79H
ERTAB1: DFB 00H,01H,02H,03H,04H,05H,06H,
          DFB 07H,08H,09H,00H,01H,02H,
          DFB 03H,04H,05H,06H,07H,08H,
ERTAB2: DFB 00H,00H,00H,00H,00H,00H,00H,
          DFB 00H,00H,00H,01H,01H,01H,
          DFB 01H,01H,01H,01H,01H,01H,
ERTAB3: DFB 00H,00H,00H,00H,00H,00H,00H,
          DFB 00H,00H,00H,00H,00H,00H,
          DFB 00H,00H,00H,00H,00H,00H,
FTAB1:  DFB 00H,0CH,0CH,0CH,0CH,0CH,0CH,
          DFB 0CH,0CH,0CH,0CH,0CH,0CH,
          DFB 0CH,0CH,0CH,0CH,0CH,0CH,
FTAB2:  DFB 00H,0EH,0EH,0EH,0EH,0EH,0EH,
          DFB 0EH,0EH,0EH,0EH,0EH,0EH,
          DFB 0EH,0EH,0EH,0EH,0EH,0EH,
FTAB3:  DFB 00H,00H,00H,00H,00H,00H,
          DFB 00H,00H,00H,00H,00H,00H,
          DFB 00H,00H,00H,00H,00H,00H,
KEYTABL:DFB 10H,10H,03H,02H,01H,00H,00H,00H
          DFB 10H,10H,07H,06H,05H,04H,00H,00H
          DFB 10H,10H,0BH,0AH,09H,08H,00H,00H
          DFB 10H,10H,0FH,0EH,0DH,0CH,00H,00H

```

END

..... EOF

□



ภาคผนวก ข. โปรแกรมภาษาซี
ที่ใช้ควบคุมในเครื่องคอมพิวเตอร์

```
#include<dos.h>
#include<stdio.h>
#define com1 0
#define RS232 0x14
#define SETUP 0x0a3 /* 0XA3 = 2400bps 0x0c3 = 4800bps */
#define MASK 0x7f
#define MCODE1 99
#define DTARDY 0x100
#define WRITECH 1
#define READCH 2
#define STATUS 3
void sport(int),port_init(),setup();
unsigned char check_stat();
unsigned int rport();
/* main program */
main()
{
char buf,dta;
unsigned int u,tt,h,s,er,tot,r,ch;
struct time t;

buf = '0';
printf(" \n");
printf(" \n");
printf(" ***** \n");
printf(" \n");
printf(" *** This Program use with DS5000 chip *** \n");
printf(" *** This Program use with dgp4.asm *** \n");
port_init();

printf(" \n");

printf(" Press q to Quit \n");
printf(" Press Enter \n");
printf(" ***** \n");

while((buf = getchar()) != 'q')
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
    r = 0;
    printf(" Number Machine # ");
    scanf("%d",&ch);
    gettime(&t);
    printf(" The current time is: %2d:%02d:%02d.%02d\n",
        t.ti_hour, t.ti_min, t.ti_sec, t.ti_hund);
    sport(ch);
    r = rport();
    if(r == MCODE1)
    {
        tot = er = u = tt = h = s = 0;
        sport(256);
        u = rport();
        sport(256);
        tt = rport();
        sport(256);
        h = rport();
        sport(256);
        s = rport();
        tot = s*1000 + h*100 + tt*10 + u;
        printf(" Total Product : ");
        printf("%d \n",tot);
        sport(256);
        er = rport();
        printf(" Error Code : ");
        printf("%d \n",er);
    }
    switch (er) {
        case 1: printf(" 26b12 stop and lock at DT 1 \n"); break;
        case 2: printf(" 27b6 stop and lock at DT 2 \n"); break;
        case 3: printf(" b927 stop and lock by feeder (check protection bar) \n"); break;
        case 4: printf(" 35b32 cover over frontray opened \n"); break;
        case 5: printf(" 11u18 photoelectric doublesheet control \n"); break;
        case 6: printf(" d938 thermal overload feeder \n"); break;
        case 7: printf(" 75u15 timing point for overshoot or frontray control missed \n"); break;
        case 8: printf(" 11u9 frontlay control left side \n"); break;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 9: printf(" 11u10 fronrlay control right side \n"); break;
case 10: printf(" 11u13 overshoot control at feeder left side \n"); break;
case 11: printf(" 11u13 overshoot control at feeder right side \n"); break;
case 12: printf(" b925 mechanical doublesheet control feeder \n"); break;
case 13: printf(" c906 pile limitation top at feeder \n"); break;
case 14: printf(" 38b22 finger protection bar at plate cylinder 1,recto printing \n"); break;
case 15: printf(" 38b24 finger protection bar at plate cylinder 2,recto printing \n"); break;
case 16: printf(" 38b26 finger protection bar at plate cylinder 3,recto printing \n"); break;
case 17: printf(" 38b28 finger protection bar at plate cylinder 4,recto printing \n"); break;
default:break;

```

```

}

buf = '0';
}
else printf(" Not Found Machine... \n");
printf(" \n");
printf(" *** Press q to Quit *** \n");
}
printf(" Source CA\DGP_COM1.C Used With CONTMP.asm ... 05/06/37 \n");
return 0;
}

/*****

void sport(c)
int c;

{
union REGS r;
r.x.dx = com1;
r.h.al = c;
r.h.ah = 1;
int86(0x14,&r,&r);
if(r.h.ah & 128)
{
printf("send error detected inserial port \n");
printf(" press any key \n");
getch();
}
}

```

```

}
/*#####*/
unsigned int rport()
{
    union REGS r;
    while (!(check_stat() & 25008)) /* 256 , 25008 */
        if(kbhit()){
            printf("read status error in serial port press any key\n");
            getch();
            getch();
            exit(1); }
    r.x.dx = com1;
    r.h.ah = 2;
    int86(0x14,&r,&r);
    if(r.h.ah & 128){
        printf("read error detected in serial port press any key\n");
        getch();
    }
    else
        printf("...Success \n");
    return r.h.al;
}
/*#####*/
unsigned char check_stat()
{
    union REGS r;
    r.x.dx = com1;
    r.h.ah = 3;
    int86(0x14,&r,&r);
    return r.x.ax;
}
/*#####*/
void port_init()
{
    union REGS r;
    r.x.dx = com1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

r.h.ah = 0;
r.h.al = SETUP;
int86(0x14,&r,&r);
}
/*##### SETUP #####*/
void setup()
{
  union REGS regs;
  printf("      ....Setup the serial port.... \n ");
  printf("*** 2400 bps/no Parity/1 Stop bit/8 Bit data *** \n ");
  printf(" \n ");
  regs.h.ah = 0;
  regs.x.dx = com1;
  regs.h.al = SETUP;
  int86(RS232,&regs,&regs);
}
/*#####*/

```

