



ปริญญาโทบริหารการศึกษา 2533

ภาควิชา อีเลคทรอนิคส์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหาร ลาดกระบัง

เรื่อง เครื่องรับสัญญาณเทเลเท็กซ์

ผู้จัดทำ

1. นาย ชอบคิด เกษรสมพร 30.1051
2. นาย บุญชัย ประสิทธิ์สัมฤทธิ์ 30.1129

  
.....อาจารย์ที่ปรึกษา  
(รศ. ประกิจ ตั้งติสานนท์)



เลขหมู่..... 7. 33123 65  
เลขทะเบียน..... 027956  
วัน, เดือน, ปี 180.ค. 34

027956

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## เครื่องรับสัญญาณเทเลเท็กซ์

นาย ชอบคิด เกษรสมุท 3๐.1๐51  
นาย นุญช่วย ประสิทธิ์สัมฤทธิ์ 3๐.1129  
รศ. ประกิจ ตั้งติสานนท์ อาจารย์ที่ปรึกษา  
ปีการศึกษา 2533

### บทคัดย่อ

ปฏิญญาฉบับนี้เป็นการศึกษาระบบเทเลเท็กซ์ ซึ่งเป็นรูปแบบการกระจาย  
ข่าวสารทางโทรทัศนแบบหนึ่ง โดย

1. ศึกษารูปแบบของการส่งข่าวสารระบบเทเลเท็กซ์
2. ศึกษาการออกแบบภาครับที่เรียกว่า ดีโคดีเดอร์ (DECODER)  
รายละเอียดในการออกแบบภาครับ มีดังนี้
  1. ทำการแยกสัญญาณเทเลเท็กซ์ออกจากสัญญาณภาพ
  2. ปรับแต่งสัญญาณเทเลเท็กซ์ให้เป็นสัญญาณดิจิทัล
  3. แปลงข้อมูลที่ส่งมาแบบอนุกรมให้เป็นแบบขนาน แล้วจัดเก็บลงในหน่วยความจำ
  4. ออกแบบวงจรในส่วนที่อินเทอร์เฟส (interface) กับ คอมพิวเตอร์ เพื่อ  
อ่านข้อมูลจากหน่วยความจำเข้าคอมพิวเตอร์
  5. เขียนซอฟต์แวร์ (software) ควบคุมการทำงานของคอมพิวเตอร์ ถอดรหัส  
ข้อมูล และแสดงผลทางจอมอนิเตอร์

## TELETEXT DECODER

Chobkid Kasornsamut

Boonchuay Prasitsumrit

Associate Professor

Prakit Tangtisanon Advisor

1990

### Abstract

This project is carried out for studying Teletext System ,which is one of the information broadcasting on television by...

1. To study the format of Teletext information system.

2. How to design the teletext decoder.

The detail of the decoder design is as following...

1. Separate teletext signal from video signal.

2. Adjust those signals to digital.

3. Convert the serial data to parallel data and store in memory.

4. Design computer interface circuit in order to read data from memory into computer.

5. Use software program to control computer ,decode data and display those data on monitor.

# สารบัญ

หน้า

บทที่ 1. บทนำ.....	1-1
บทที่ 2. แนวความคิดและลักษณะสัญญาณเทเลเท็กซ์.....	2-1
2.1 องค์ประกอบของเทเลเท็กซ์.....	2-1
2.2 ที่มาของเทเลเท็กซ์.....	2-1
2.3 ลักษณะของเทเลเท็กซ์.....	2-3
2.4 สัญญาณของเทเลเท็กซ์.....	2-4
2.5 รายละเอียดของ โครงสร้างเส้นข้อมูล (DATA-LINE STRUCTURE).....	2-6
หลักทั่วไปของตัวถอดรหัส (DECODER).....	2-14
2.6 ส่วนการรับข้อมูล (DATA ACQUISITION)....	2-14
2.7 ส่วนการเลือกข้อมูล (DATA SELECTION)....	2-17
2.8 ส่วนการเก็บข้อมูล (PAGE MEMORY).....	2-18
2.9 ส่วนการแสดงผล (DISPLAY CONTROL)....	2-19
บทที่ 3. การคำนวณ และการสร้างวงจร.....	3-1
3.1 วงจรยกระดับดิซี (DC) คงที่.....	3-1
3.2 วงจรแยกซิงค์และสัญญาณเทเลเท็กซ์.....	3-1
3.3 วงจรตัดสัญญาณทดสอบ (test signal)....	3-1
3.4 วงจรแปลงสัญญาณจากอนุกรมเป็นขนาน (serial to parallel) และวงจรตรวจจับรหัสเฟรมมิ่ง (Framing detector).....	3-4
3.5 วงจรเก็บข้อมูลลงหน่วยความจำ.....	3-6
3.6 วงจรป้องกันข้อมูลผิดพลาด (Correction of Hamming coded data).....	3-8
3.7 วงจรหาแม่กกาซีน แอดเดรสแถว และหน้า ของเทเลเท็กซ์.....	3-11
3.8 วงจรเลือกหน้าเทเลเท็กซ์.....	3-12
3.9 การแสดงผลเทเลเท็กซ์.....	3-17
บทที่ 4. การทดลอง และผลการทดลอง.....	4-1
บทที่ 5. วิจารณ์ และสรุป.....	5-1

กิตติกรรมประกาศ

หนังสืออ้างอิง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ

		หน้า
รูปที่ 2.2.1	การสแกนแบบสลับเส้น.....	2-2
รูปที่ 2.2.2	แสดงสัญญาณซิงค์ทางแนวตั้ง และสัญญาณซิงค์ ทางแนวนอน.....	2-2
รูปที่ 2.2.3	แสดงเส้นแบล็กคิงค์ที่อยู่ในแบล็กคิงค์ทาง แนวตั้ง.....	2-2
รูปที่ 2.2.4	แสดงตัวอย่างเทเลเท็กซ์ของ CEEFAX.....	2-3
รูปที่ 2.4.1	แสดงระดับของสัญญาณเทเลเท็กซ์ กับระดับ ของสัญญาณภาพ.....	2-5
รูปที่ 2.4.2	แสดงสัญญาณเทเลเท็กซ์ใน 1 บรรทัด.....	2-5
รูปที่ 2.4.3	แสดงสัญญาณเทเลเท็กซ์ (a) ทางด้านส่ง และ (b) ทางด้านรับ.....	2-5
รูปที่ 2.4.4	แสดงโครงสร้างมาตรฐานของเทเลเท็กซ์...	2-7
รูปที่ 2.5.3.1	แสดงพาริตีร์บิตของรหัสแฮมมิง.....	2-8
รูปที่ 2.5.3.2	แสดงการตรวจสอบพาริตีร์บิตของรหัสแฮมมิง....	2-8
รูปที่ 2.5.4.1	แสดงตัวอย่างโค๊ดของตัวอักษรและกราฟิค...	2-12
รูปที่ 2.5.5.1	แสดงมาตรฐานบรรทัดแรกของ 1 หน้า เทเลเท็กซ์.....	2-13
รูปที่ 2.5.5.2	แสดงแบบมาตรฐานของบรรทัดอื่น.....	2-13
รูปแสดง	บล็อกไดอะแกรมของเครื่องถอดรหัสเทเลเท็กซ์.....	2-14
รูปที่ 2.6.1	แสดงบล็อกไดอะแกรมของส่วนการรับข้อมูล...	2-14
รูปที่ 2.6.1.1	แสดงบล็อกไดอะแกรมของดาต้าสไลเซอร์....	2-15
รูปที่ 2.6.2.1	แสดงสัญญาณนาฬิกาที่ตรงกับข้อมูลเทเลเท็กซ์..	2-15
รูปที่ 2.6.3.1	แสดงบล็อกไดอะแกรมของดาต้าเกจ.....	2-16
รูปที่ 2.6.4.1	แสดงบล็อกไดอะแกรมของการแปลงข้อมูล อนุกรมเป็นข้อมูลขนาน.....	2-16
รูปที่ 2.6.5.1	แสดงบล็อกไดอะแกรมของการตรวจจับรหัส เฟรมมิ่ง.....	2-17
รูปที่ 2.6.6.1	แสดงบล็อกไดอะแกรมของการตรวจจับข้อมูล ผิดพลาด.....	2-17
รูปที่ 2.7.1	แสดงบล็อกไดอะแกรมของส่วนการเลือกข้อมูล.	2-18
รูปที่ 2.8.1	แสดงบล็อกไดอะแกรมของส่วนหน่วยความจำ..	2-18

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## สารบัญภาพ (ต่อ)

		หน้า
รูปที่ 2.9.1	แสดงบล็อกไดอะแกรมของส่วนควบคุมการ แสดงผล.....	2-19
รูปที่ 3.1.1	แสดงวงจรระดับดีซีคิงท์.....	3-1
รูปที่ 3.2.1	แสดงวงจรแยกซิงค์ และสัญญาณเทเลเท็กซ์..	3-2
รูปที่ 3.2.2	แสดงวงจรรวมการแยกซิงค์ และสัญญาณ สัญญาณเทเลเท็กซ์.....	3-3
รูปที่ 3.3.1	แสดงวงจรตัดสัญญาณทดสอบออกจากสัญญาณ เทเลเท็กซ์.....	3-4
	รูปแสดงแผนผังการทำงานของไอซีเบอร์ 74LS164.....	3-4
รูปที่ 3.4.1	แสดงวงจรแปลงข้อมูลอนุกรมเป็นข้อมูลขนาน..	3-5
รูปที่ 3.4.2	แสดงวงจรควบคุมเคาท์เตอร์.....	3-6
รูปที่ 3.4.3	แสดงวงจรรวมทั้งหมดของส่วนที่แปลงข้อมูล อนุกรมไปเป็นข้อมูลขนาน.....	3-7
รูปที่ 3.5.1	แสดงวงจรรวมในส่วนของการเก็บข้อมูลลง หน่วยความจำ (RAM).....	3-9
รูปที่ 3.6.1	แสดงวงจรป้องกันข้อมูลผิดพลาด.....	3-10
รูปที่ 3.7.1	แสดงการทำงานของซีพียูเตอร์ของรหัส เฟรมมิ่ง.....	3-11
รูปที่ 3.7.2	แสดงวงจรซีพียูเตอร์ของรหัสเฟรมมิ่ง.....	3-12
รูปที่ 3.7.3	แสดงวงจรตรวจจับแม่กาซีน และแอดเดรส แถว.....	3-13
รูปที่ 3.7.4	แสดงวงจรตรวจจับเลขหน้าหลักหน่วย และ เลขหน้าหลักสิบ.....	3-14
รูปที่ 3.8.1	แสดงวงจรทั้งหมดในส่วนของการเลือกหน้า เทเลเท็กซ์.....	3-16
รูปที่ 3.9.2.1	แสดงบล็อกไดอะแกรมของวงจรมินิเทอร์เฟส..	3-18
รูปที่ 3.9.2.2	แสดงวงจรมินิเทอร์เฟสที่ออกแบบไว้.....	3-20

## สารบัญตาราง

	หน้า
ตารางที่ 2.5.3.1 แสดงผลการทดสอบพาริตรี และการกลับ บิตที่ผิด.....	2-9
ตารางที่ 2.5.6.1 แสดงสัญลักษณ์ และรหัสทางด้านกราฟิก และตัวควบคุม.....	2-10
ตารางที่ 2.5.4.1 แสดงสัญลักษณ์และโค้ดของเทเลเท็กซ์...	2-11
ตารางที่ 2.5.4.2 แสดงโหมดการแสดงผลและรูปแบบของ ตัวอักษร.....	2-11
ตารางที่ 3.3.1 แสดงการทำงานของไอซีเบอร์ 74LS154.	3-2



## บทที่ 1.

## บทนำ

ระบบเทเลเท็กซ์ คือ ระบบที่สามารถส่งข้อมูลและข่าวสารโดยอาศัยคลื่นวิทยุโทรทัศน์ในขณะที่เดียวกับที่ออกอากาศรายการปกติ ข่าวสารและข้อมูลแต่ละหน้าสามารถเรียกออกมาดูได้ ตามความต้องการของผู้ชมแต่ละบุคคล ผู้ที่มีเครื่องรับโทรทัศน์ธรรมดาจะสามารถรับข้อมูลข่าวสารเหล่านี้ได้ โดยการติดตั้งแผ่นวงจรพิเศษ (ดีโคดีเตอร์) ที่เครื่องรับโทรทัศน์เท่านั้น

ข้อมูลและข่าวสารถูกรวบรวมและผลิตขึ้นจากเครื่องคอมพิวเตอร์ส่วนกลางของระบบเทเลเท็กซ์ ซึ่งภาพหรือตัวอักษรจะถูกเปลี่ยนเป็นสัญญาณวิทยุโทรทัศน์แทรกอยู่ในสัญญาณที่ส่งรายการโทรทัศน์ตามปกติ ดังนั้นไม่ว่าเครื่องรับโทรทัศน์จะอยู่ที่ใดก็ตาม ถ้าสามารถรับรายการปกติได้ผู้ชมก็สามารถรับข้อมูลและข่าวสารที่ส่งจากระบบเทเลเท็กซ์ได้เช่นเดียวกัน จึงเป็นการใช้เครือข่ายวิทยุโทรทัศน์ซึ่งปัจจุบันครอบคลุมทั่วทั้งประเทศให้ได้ใช้ประโยชน์สูงสุด กล่าวคือให้ความบันเทิงและข่าวสาร ข้อมูลที่มีประโยชน์สูงสุดในเวลาเดียวกันอย่างรวดเร็วตลอดเวลาที่ออกอากาศ

ข้อมูลและข่าวสารเหล่านี้อาจจะเป็นที่ข่าวสารทางราชการ ทางธุรกิจ การค้า ทางการศึกษาหรือข่าวสารอื่น ๆ ที่ต้องการเผยแพร่อย่างรวดเร็ว ผู้ชมจะสามารถเลือกเฉพาะข้อมูลที่ต้องการดูได้ตลอดเวลา ข้อมูลและข่าวสารส่วนหนึ่งส่งมาจากแหล่งอื่น ๆ โดยตรง เช่น ข้อมูลเที่ยวบินจากท่าอากาศยาน ข้อมูลราคาหุ้นจากตลาดหลักทรัพย์ หรือข้อมูลอัตราแลกเปลี่ยนเงินตราต่างประเทศจากธนาคาร เป็นต้น แม้แต่ข่าวสารข้อมูลจากต่างประเทศก็สามารถส่งตรงมายังระบบเทเลเท็กซ์ เพื่อเผยแพร่ไปยังผู้ชมได้ทันที ข่าวสารและข้อมูลอีกส่วนหนึ่งผลิตขึ้นโดยผู้จัดทำป้อนเข้าไปในเครื่องคอมพิวเตอร์ส่วนกลางของระบบเทเลเท็กซ์เป็นประจำ

ระบบเทเลเท็กซ์ต่างจากระบบวิดีโอเท็กซ์ซึ่งปรากฏอยู่ในปัจจุบัน กล่าวคือระบบวิดีโอเท็กซ์ ผลิตภาพหรือตัวอักษรออกจากโทรทัศน์เช่นเดียวกับการออกอากาศรายการปกติ ผู้ชมไม่สามารถเลือกดูเฉพาะข้อมูลที่ต้องการดูได้ จำเป็นต้องรอดูไปเรื่อยจนกว่าจะถึงข้อมูลที่ต้องการ ซึ่งอาจใช้เวลานานทั้งยังไม่สามารถหยุดข้อมูลที่ต้องการดูได้นานด้วย นอกจากนี้วิดีโอเท็กซ์ยังไม่สามารถส่งข้อมูลและข่าวสารในเวลาเดียวกับที่ออกอากาศตามรายการปกติได้ ส่วนระบบเทเลเท็กซ์นั้นสามารถส่งได้พร้อมกับรายการปกติและผู้ชมยังสามารถเลือกดูเฉพาะข้อมูลข่าวสารที่ต้องการเท่านั้น การทำงานคล้ายกับการเปิดดูหนังสือแต่ละหน้า สามารถหยุดดูข้อมูลได้นานเท่าที่ต้องการและเลือกดูได้เฉพาะหน้าโดยไม่ต้องเปิดดูตั้งแต่หน้าแรกเหมือนวิดีโอเท็กซ์

## บทที่ 2.

## แนวความคิดและลักษณะสัญญาณเทเลเท็กซ์

เทเลเท็กซ์ คือ ระบบที่สามารถส่งข้อมูลและข่าวสารมาตามคลื่นวิทยุโทรทัศน์พร้อมกับการออกอากาศรายการปกติ ระบบเทเลเท็กซ์ส่งข้อมูลในรูปของหนังสือพิมพ์ทางโทรทัศน์มีการแบ่งข้อมูลข่าวสารเป็นหน้า ๆ ซึ่งผู้ชมสามารถเลือกดูแต่ละหน้าตามความต้องการของตนเอง เช่น ผู้ชมคนหนึ่งอาจเลือกดูข่าวเรื่องหุ้นขณะที่ผู้ชมอีกคนเลือกดูข่าวพยากรณ์อากาศ

เทเลเท็กซ์ มีข้อได้เปรียบกว่าหนังสือพิมพ์ตรงที่ ข่าวสารสามารถแก้ไขให้ทันสมัยได้ตลอดเวลา ผู้ชมจึงได้ข้อมูลข่าวสารที่ทันต่อเหตุการณ์จริง ๆ

2.1 องค์ประกอบของระบบเทเลเท็กซ์

## 2.1.1 การรวบรวมข่าวสารข้อมูล

เนื่องจาก เทเลเท็กซ์ มีจุดประสงค์ที่จะเสนอข้อมูลข่าวสารที่ทันต่อเหตุการณ์ให้แก่ผู้ชม ดังนั้นการรวบรวมข้อมูลจึงมีความสำคัญมาก ผู้ส่งจะต้องมีเครือข่ายการรวบรวมข้อมูลที่ดี โดยวิธีการต่าง ๆ เช่น มีทีมข่าวที่ดีมีประสิทธิภาพ มีการออนไลน์ (ON LINE) กับตลาดหลักทรัพย์ เพื่อทราบการเปลี่ยนแปลงของตลาดหลักทรัพย์ตลอดเวลาที่มีการเปิดตลาด มีการออนไลน์กับธนาคาร มีการออนไลน์ กับกรมอุตุนิยมวิทยา เป็นต้น

## 2.1.2 การส่งข้อมูล

ผู้ส่งระบบเทเลเท็กซ์ จะต้องมามีเครื่องมือหลักคือ ตัวป้อนข้อมูล (DATA - ENTRY) กับตัวเข้ารหัส (ENCODER) ที่จะใช้ป้อนข้อมูลแล้วทำการเข้ารหัสข้อมูลใส่ลงในสัญญาณโทรทัศน์เพื่อออกอากาศสู่ผู้ชมต่อไป

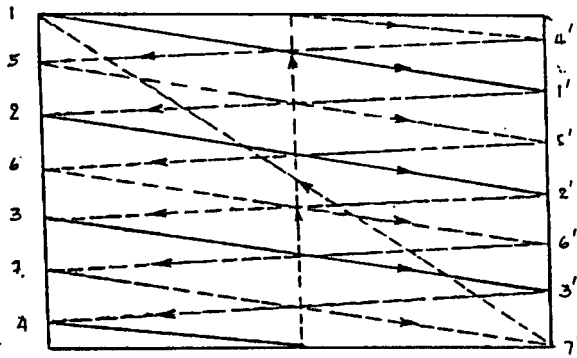
## 2.1.3 การรับข้อมูล

ผู้ชมที่ต้องการดูข้อมูลเทเลเท็กซ์ นอกจากจะต้องมีเครื่องรับโทรทัศน์แล้วยังต้องมีตัวถอดรหัส (Decoder) ที่จะใช้ถอดรหัสข้อมูลเพื่อแสดงผลออกทางจอภาพ ซึ่งตัวถอดรหัสนี้จะมีตัวควบคุม เพื่อให้ผู้ชมสามารถเลือกดูข้อมูลได้

2.2 ที่มาของเทเลเท็กซ์

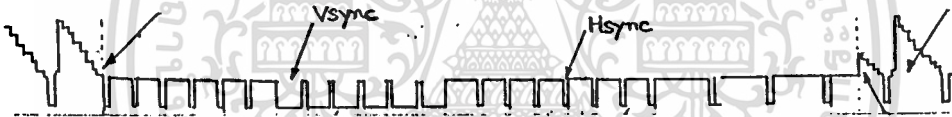
ในการปรากฏภาพบนจอโทรทัศน์ให้เราเห็นนั้น เกิดการสแกนของหลอดภาพให้เกิดภาพจำนวน 25 ภาพใน 1 วินาที และในระบบ 625 เส้น โดยวิธีสแกนนั้นจะแบ่ง 1 ภาพเป็น 2 ฟิลด์ (1 frame : 2 field) ฟิลด์ละ  $625/2 = 314.5$  เส้น ฟิลด์หนึ่งจะสแกนเฉพาะเส้นคี่ อีกฟิลด์หนึ่งก็จะสแกนเฉพาะเส้นคู่ ดังภาพ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



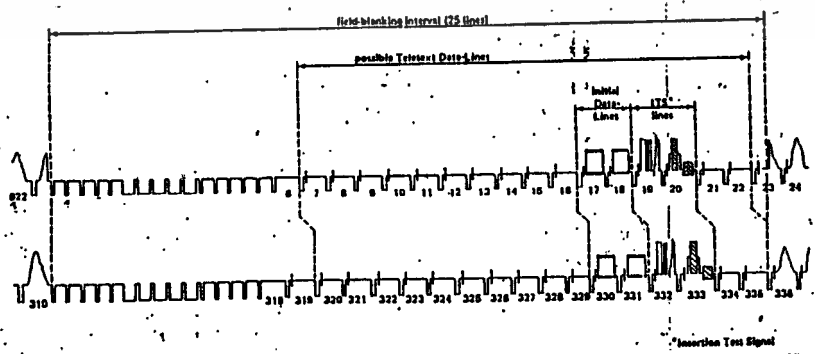
รูปที่ 2.2.1 การสแกนแบบสลับเส้น

ดังนั้นสัญญาณภาพที่สถานีส่งมาจะแบ่งเป็นเส้น ๆ โดยมีสัญญาณที่เรียกว่า สัญญาณซิงค์ทางแนวนอน (Horizontal Synchronizing : Hsync) เป็นตัวบอก และมีสัญญาณซิงค์ทางแนวตั้ง (Vertical Synchronizing : Vsync) เป็นตัวบอกการสแกนแต่ละฟิลด์



รูปที่ 2.2.2 แสดงสัญญาณซิงค์ทางแนวตั้ง และสัญญาณซิงค์ทางแนวนอน

ในช่วง สัญญาณเวอร์แบล็งค์กึ่ง (VertBlanking) หรือ ฟิลด์แบล็งค์กึ่ง (Field Blankin) นั้นจะมีเส้นว่าง ๆ ที่ไม่มีสัญญาณภาพอยู่ 25 เส้นเรียกว่า เส้นแบล็งค์กึ่ง (line blanking) เพื่อให้เป็นช่วงเวลาของการกลับไปเริ่มต้นสแกนฟิลด์ใหม่ของหลอดภาพ



รูปที่ 2.2.3 แสดงเส้นแบล็งค์กึ่งที่อยู่ในแบล็งค์กึ่งทางแนวตั้ง

นับแต่มีระบบโทรทัศน์เกิดขึ้น เหล่าวิศวกรที่เกี่ยวข้องทั้งหลายก็พยายามที่จะใช้ประโยชน์จาก เส้นแบล็งค์กึ่งคี่ เหล่านี้ พวกเขาทดลองใส่สัญญาณต่าง ๆ ลงบน เส้นแบล็งค์กึ่งคี่ เรื่อยมา ซึ่งต่อมาก็มีการใส่สัญญาณทดสอบ (test signal) ในช่วง เส้นแบล็งค์กึ่งคี่ เพื่อการทดสอบการส่งสัญญาณของสถานีและใช้กันมาจนปัจจุบัน

แต่เหล่าวิศวกรก็ยังพยายามที่จะใช้ เส้นเหล่านี้เพื่อการส่งข่าวสาร ข้อมูลถึงผู้ชมให้ได้ จนในช่วงต้นทศวรรษที่ 70 เมื่อระบบ ดิจิตอลเจริญก้าวหน้าขึ้นพร้อม ๆ กัน มีผู้เสนอแนวความคิดเรื่องการส่งข่าวสารทางโทรทัศน์ในรูปแบบหนังสือพิมพ์ ที่ให้ผู้ชมเป็นผู้เลือกข้อมูลดูได้ ความคิดนี้จึงถูกพัฒนาเข้ากับการใช้ประโยชน์จาก เส้นแบล็งค์กึ่งคี่ จนเป็นผลสำเร็จ และในปี 1973 สถานีโทรทัศน์ BBC และองค์การ IBA (Independent Broadcasting Authority) ของอังกฤษก็พัฒนาระบบเทคโนโลยี ที่สมบูรณ์แบบขึ้นมีชื่อเรียกว่า "ORACLE" (Optional Reception of Announcements by Coded by Line Electronics) นับแต่นั้นมาก็แพร่หลายไปยังประเทศต่าง ๆ และมีการพัฒนาขึ้นตามลำดับ เรื่อยมา



รูปที่ 2.2.4 แสดงตัวอย่างเทคโนโลยีของ CEEFAX

### 2.3 ลักษณะของเทคโนโลยี

ในช่วง เส้นแบล็งค์กึ่งคี่ 25 เส้น (line) คือช่วง เส้น ที่ 623.5 ถึง 22.5 ของฟิลด์แรกและ เส้น 311 ถึง 335 ของฟิลด์หลังนั้น เราสามารถใส่สัญญาณใด ๆ ลงไปได้ทุกเส้น แต่เส้นที่เหมาะสมในการใส่สัญญาณเทคโนโลยีลงไป คือ เส้น 7-22 และ 320-335

สัญญาณเทคโนโลยีที่ใส่ลงไปนั้นอยู่ในรูปสัญญาณดิจิตอล (digital) ซึ่งเกิดจากการเข้ารหัสของตัวอักษรของข้อมูลที่ต้องการส่งไปยังผู้ชม และใน 1 เส้น สามารถใส่ข้อมูลดิจิตอลได้ถึง 45 ไบต์ หรือ 360 บิต ซึ่ง 45 ไบต์ จะไปปรากฏเป็นข้อมูล 1 บรรทัดบนจอภาพ และ 1 จอภาพจะมีข้อมูลทั้งสิ้น 24 บรรทัด (0-23) ซึ่งกำหนดให้เป็น 1 หน้าข้อมูล (page) ระบบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้คัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เทเลเท็กซ์ระบบหนึ่งจะแบ่งข่าวสารที่ส่งเป็นแม็กกาซีน (Magazine) ซึ่งอาจมีตั้งแต่ 1 ถึง 8 แม็กกาซีน แต่ละแม็กกาซีน จะมีจำนวนที่หน้าก็ขึ้นอยู่กับจำนวนข่าวสารที่สถานีต้องการส่งไปตั้งแต่นำ ๑๑-๑๑

การเลือกเส้นที่จะใส่ข้อมูลเทเลเท็กซ์ ลงไปนั้นขึ้นอยู่กับความสามารถของระบบเครื่องส่งและระบบเทเลเท็กซ์ ของแต่ละสถานีโทรทัศน์ เพราะถึงแม้ว่าเส้นที่ใช้ได้คือ 7-22 (32๑-335) แต่เส้นที่ดีที่สุดในการส่งข้อมูลคือ เส้น 17, 18 (33๑, 331) ฉะนั้นจึงใช้เส้นในการส่งมากเท่าใด โอกาสที่จะส่งข้อมูลผิดพลาดก็จะเพิ่มขึ้น แต่ถ้าใช้เฉพาะเส้น 17, 18 (33๑, 331) การส่งข้อมูลให้ครบทั้งหมด 1 แม็กกาซีน ก็จะใช้เวลานาน เพราะการส่งสัญญาณจะเป็น 5๑ ฟิลต์ต่อวินาทีดังที่ ดังนั้นถ้าใช้ 2 เส้น ก็จะส่งข้อมูลได้ 1๑๑ บรรทัดต่อวินาที ถ้าข้อมูลมีหลายหน้าผู้ชมก็จะต้องรอรับข้อมูลนาน การเลือกจำนวนเส้นที่จะใช้จึงขึ้นอยู่กับความเหมาะสมของจำนวนข้อมูลทั้งหมดที่จะส่งไปด้วย

สำหรับระบบเทเลเท็กซ์ระบบแรกของประเทศไทยนั้น เป็นของสถานีโทรทัศน์กองทัพบกช่อง 5 ใช้ชื่อว่า ARMTEXT มีจำนวนข่าวสาร 8 แม็กกาซีน แม็กกาซีนละ 1๑๑ หน้าคือส่งตั้งแต่หน้า 1๑๑-๑๑๑ และใช้เส้นในการส่ง 8 เส้น ต่อฟิลต์โดยใช้เส้นที่ 11-16 และ 19, 2๑ (324-329 และ 332, 333) ส่วนเส้นที่ 17, 18 (33๑, 331) ทางสถานีใช้ในการส่งสัญญาณทดสอบ

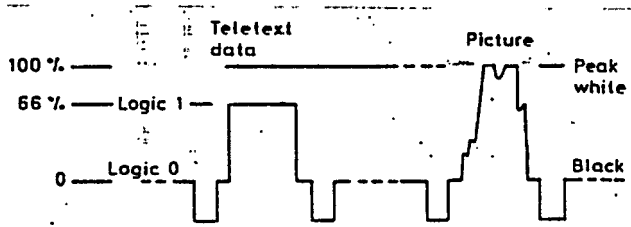
แต่ในปัจจุบันทางช่อง 5 ได้เพิ่มเส้นในการส่งข้อมูลขึ้นมาอีก 1 เส้น คือ เส้นที่ 21 (334) ทำให้การรับข้อมูลมีความเร็วเพิ่มขึ้นมาอีกระดับหนึ่ง

ทางด้านผู้ชมที่ต้องการรับข้อมูลเทเลเท็กซ์ จะต้องมีอุปกรณ์พิเศษในการรับข้อมูลถอดรหัส และเลือกข้อมูลหน้าที่ต้องการออกจอตีทัศน์ อุปกรณ์นี้เรียกว่า "ตัวถอดรหัส หรือ ดีโคดีเตอร์ (Decoder)" ในโทรทัศน์บางรุ่นจะมีดีโคดีเตอร์ติดตั้งภายในเครื่องแล้ว ผู้ชมเพียงแต่รับคลื่นช่อง 5 และกดเลือกระบบเทเลเท็กซ์ เท่านั้น

## 2.4 สัญญาณเทเลเท็กซ์

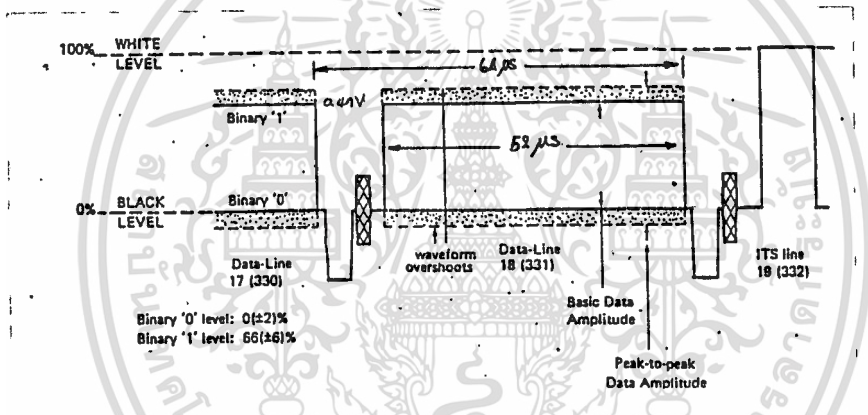
ในการนำสัญญาณเทเลเท็กซ์มาถอดรหัส เป็นข้อมูลที่นำมาแสดงบนจอโทรทัศน์ได้นั้น เราจะต้องรู้ว่าสัญญาณเทเลเท็กซ์ ที่ส่งมามีลักษณะเช่นไร ส่งข้อมูลมาในรูปแบบไหนซึ่งมีรายละเอียดดังนี้

2.4.1 สัญญาณเทเลเท็กซ์ ส่งมาในรูปแบบของสัญญาณดิจิทัล (digital signal) เป็นข้อมูลอนุกรมแบบซิงค์โครนัส (Synchronous Serial Data Signal) มีระดับลอจิก ๑ เท่ากับระดับ สีดำ (black level) หรือ ระดับต่ำสุดของสัญญาณภาพ และลอจิก 1 มีระดับเท่ากับ 66% ของระดับสีขาว (white level) หรือระดับสูงสุดของสัญญาณภาพ



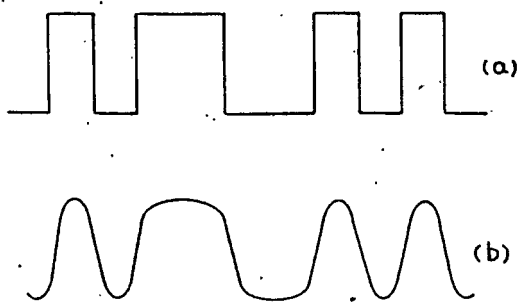
รูปที่ 2.4.1 แสดงระดับของสัญญาณเทเลเท็กซ์ กับระดับของสัญญาณภาพ

2.4.2 เส้นสแกน (Scan line) มีคาบเวลา 64 ไมโครวินาที (us) เราใช้ใส่สัญญาณเทเลเท็กซ์ 52 ไมโครวินาที ดังนั้นข้อมูลที่มี 45 ไบต์ หรือ 360 บิตต่อเส้น อัตราการส่งข้อมูลจะเป็น 6.9375 เมกกะบิตต่อวินาที (Mbit/sec) หรือ 1 บิต มีคาบเวลาเท่ากับ 144 นาโนวินาที (ns)



รูปที่ 2.4.2 แสดงสัญญาณเทเลเท็กซ์ใน 1 บรรทัด

2.4.3 รูปร่างของสัญญาณเทเลเท็กซ์ เมื่อส่งมากับสัญญาณโทรทัศน์จะเกิดความเพี้ยนทำให้มีรูปร่างเป็นสัญญาณไซน์ (sine) ในการนำออกมาใช้ต้องมีการปรับแต่งให้เป็นสัญญาณดิจิทัลที่มีคุณภาพเสียก่อน



รูปที่ 2.4.3 แสดงสัญญาณเทเลเท็กซ์ (a) ทางด้านส่ง (b) ทางด้านรับ

2.4.4 ข้อมูลใน 1 เส้น ซึ่งมี 45 ไบต์ จะแทนข้อมูล 1 บรรทัด และมีโครงสร้างเป็นมาตรฐานในระบบเทเลเท็กซ์ทุกระบบ ดังรูปที่ 2.4.4 ในหน้าที่ 2-7

## 2.5 รายละเอียดโครงสร้างของเส้นข้อมูล (DATA-LINE STRUCTURE)

2.5.1 คล็อกรันอิน (Clock Run-in) 2 ไบต์ แรกของทุกเส้นข้อมูล (DATA-LINE) จะเป็นคล็อกรันอินซึ่งมีลักษณะข้อมูลเป็น 1010101010101010 เพื่อบอกอัตราความเร็วบิตของข้อมูลเทเลเท็กซ์ เพื่อให้ตัวถอดรหัสทำการสร้างสัญญาณนาฬิกา (Clock) ที่ตรงกับข้อมูลอนุกรม (serial data) ที่ส่งมา

2.5.2 รหัสเฟรมมิ่ง (Framing Code) ไบต์ 3 ของทุกเส้นข้อมูล จะเป็นข้อมูล 11100100 เสมอเรียกว่า F.C. ใช้ในการตรวจสอบความตรงกันของไบต์ข้อมูล (byte synchronous) ของตัวถอดรหัส ว่ารับข้อมูล 1 ไบต์ ได้ตรงกับข้อมูลที่ส่งมาหรือไม่

2.5.3 แม็กกาซีน และกลุ่มบอกแถว (Magazine and Row address Group) ไบต์ที่ 4 และ 5 ของทุกเส้นข้อมูล จะเป็นข้อมูลที่บอกเลขที่ของแม็กกาซีนและหมายเลขบรรทัด ข้อมูลว่า เส้นนั้นเป็นข้อมูลบรรทัดที่เท่าไรในจอโทรทัศน์ ซึ่งในเทเลเท็กซ์ระบบหนึ่งสามารถมีได้ 1-8 แม็กกาซีน (000-111) และข้อมูล 1 หน้าจะมี 24 บรรทัด (00000-10111 ส่วน 11000-11111 ไม่ได้ใช้)

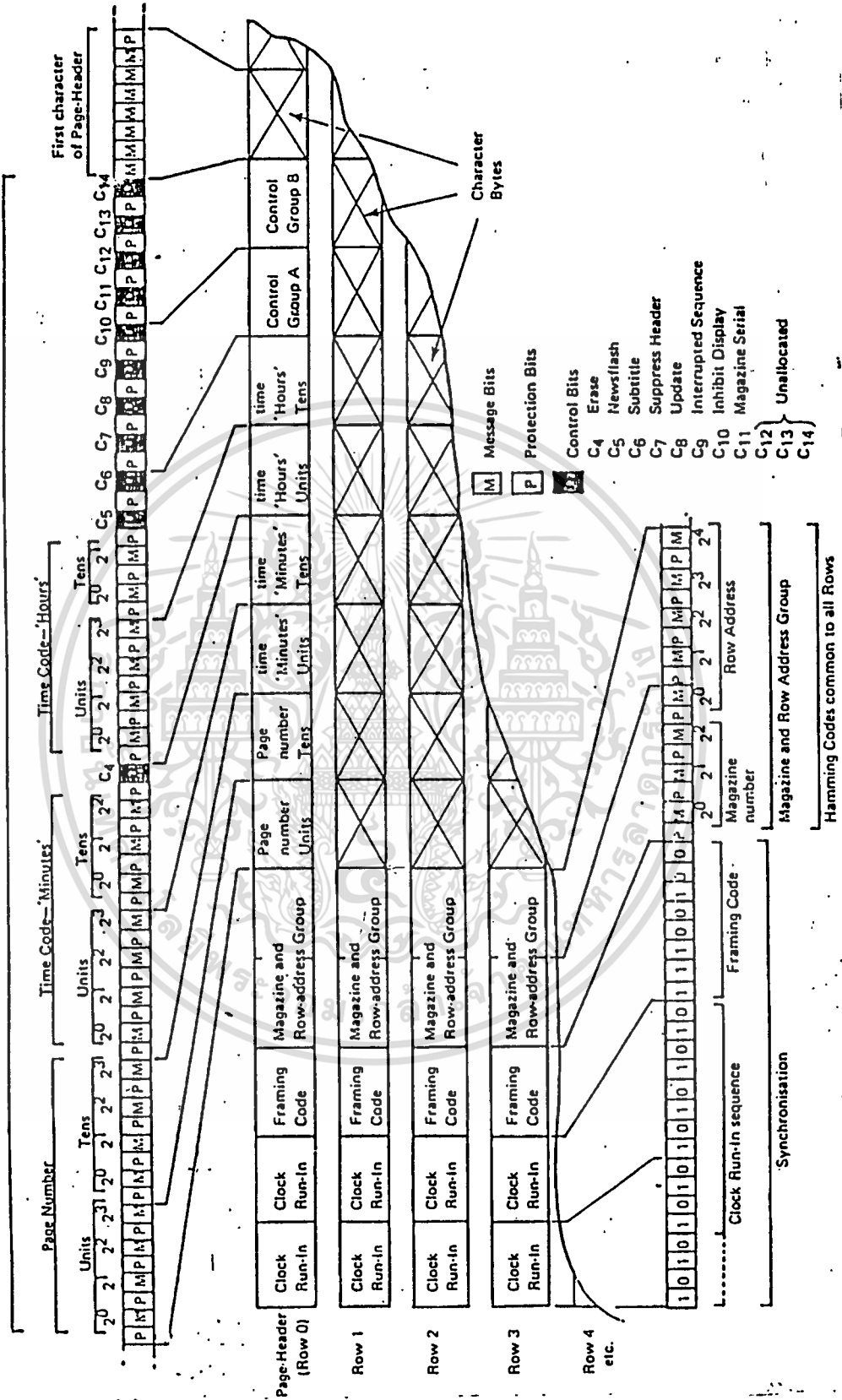
M & R-A.G. เป็นข้อมูลที่เข้ารหัสแบบรหัสแฮมมิ่ง (Hamming Code) มีรูปแบบดังนี้

P	M	P	M	P	M	P	M
---	---	---	---	---	---	---	---

รหัสแฮมมิ่ง นั้นจะแบ่งข้อมูล 1 ไบต์ เป็นบิตป้องกัน (Protection bit) 4 บิต (0,2,4,6) และบิตข้อมูล (Message bit) 4 บิต (1,3,5,7) ข้อดีของรหัสแฮมมิ่ง คือทางฝ่ายรับสามารถตรวจสอบว่าข้อมูลที่ได้รับถูกต้องหรือไม่ และถ้าข้อมูลที่ได้รับผิดพลาดทางฝ่ายรับก็สามารถแก้ไขให้ถูกต้องเองได้ แต่มีข้อเสีย คือ สามารถเข้ารหัสได้เพียง  $2^4 = 16$  ตัว เพราะใน 1 ไบต์ จะถูกใช้เป็น พาริตรีบิต (Parity bit) ถึงครึ่งหนึ่ง จึงได้บิตข้อมูลเพียง 4 บิต

รหัสแฮมมิ่ง ที่ใช้ในระบบเทเลเท็กซ์ นั้นใช้หลักการของ นาย RW HAMMING กำหนด พาริตรีบิตเป็นแบบคี่ (odd parity) คือมีบิต "1" ในไบต์นั้นเป็นจำนวนคี่ ถ้ามีข้อมูลที่ผิดพลาดเพียง 1 บิต จะสามารถตรวจสอบตำแหน่งของบิตที่ผิด และแก้ไขให้ถูกต้องได้ แต่ถ้า

Eight Hamming Codes peculiar to Page-Header

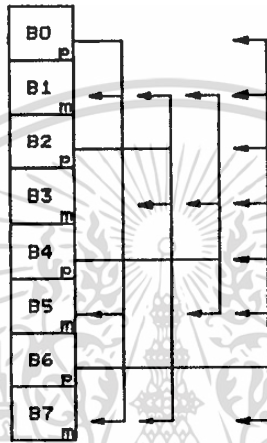


รูปที่ 2.4.4 แสดงโครงสร้างมาตรฐานของระบบเทคโนโลยี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลผิดพลาดเกินกว่า 1 บิตก็จะไม่ยอมรับข้อมูลไบนารีนั้น ซึ่งวิธีการตรวจสอบของรหัสแฮมมิงเป็นดังนี้

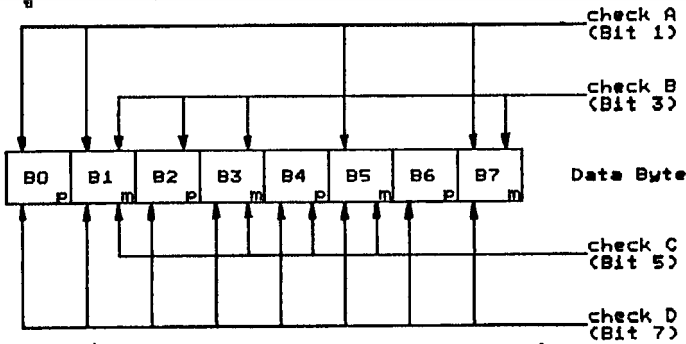
- ก. บิต 0 จะเป็น พาริตรีของบิต 1, 5, 7
  - บิต 2 จะเป็น พาริตรีของบิต 1, 3, 7
  - บิต 4 จะเป็น พาริตรีของบิต 1, 3, 5
  - บิต 6 จะเป็น พาริตรีของบิต 0, 1, 2, 3, 4, 5, 7
- พาริตรี จะเป็น "0" เมื่อจำนวน "1" ในบิตของมันเป็นจำนวนคี่



รูปที่ 2.5.3.1 แสดงพาริตรีบิตของรหัสแฮมมิง

- ข. เมื่อได้รับข้อมูล 1 ไบต์ จะมีการตรวจสอบพาริตรี 4 แบบ คือ
- A. ตรวจสอบบิต 0, 1, 5, 7
  - B. ตรวจสอบบิต 1, 2, 3, 7
  - C. ตรวจสอบบิต 1, 3, 4, 5
  - D. ตรวจสอบบิต 0-7

ถ้าตรวจสอบพบว่า มี 1 ในแต่ละบิตที่เช็ครวมเป็นจำนวนคี่ก็จะให้ผลเป็น 0 ถ้ามี 1 เป็นจำนวนคู่ จะให้ผลเป็น 1 พาริตรีแบบคี่



รูปที่ 2.5.3.2 แสดงการตรวจสอบพาริตรีของรหัสแฮมมิง



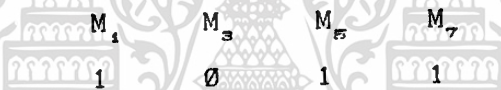
ค. นำผลการตรวจสอบไปวิเคราะห์

A	B	C	D	ACTION REQUIRED
0	0	0	0	DATA IS CORRECT
1	0	0	0	REJECT DATA TWO OR MORE ERRORS DETECTED IN BYTE.
0	1	0	0	
1	1	0	0	
0	0	1	0	
1	0	1	0	
0	1	1	0	
1	1	1	0	
0	0	0	1	INVERT BIT 6
1	0	0	1	INVERT BIT 0
0	1	0	1	INVERT BIT 2
1	1	0	1	INVERT BIT 7
0	0	1	1	INVERT BIT 4
1	0	1	1	INVERT BIT 5
0	1	1	1	INVERT BIT 3
1	1	1	1	INVERT BIT 1

0 = Parity test O.K.  
1 = Error detected

ตารางที่ 2.5.3.1 แสดงผลการทดสอบพาริตี และการกลับบิตที่ผิด

ตัวอย่างเช่น การส่งข้อมูล 1101 เมื่อใส่พาริตีแต่ละตำแหน่งจะได้



ดังนั้น  $P_0 = 0$   
 $P_2 = 1$   
 $P_4 = 1$   
 และ  $P_6 = 0$

ดังนั้นจะได้ไบต์ข้อมูลเป็นดังนี้  $P_0 M_1 P_2 M_3 P_4 M_5 P_6 M_7$   
 0 1 1 0 1 1 0 1

ซึ่งรหัสแอมมิ่ง ที่ส่งไปจะเป็น 10110110 ถ้าทางฝ่ายรับได้ข้อมูลเป็น 10010110 คือบิตที่ 5 ผิด เมื่อนำไปตรวจสอบจะได้

A = 1 บิต 0, 1, 5, 7

B = 0 บิต 1, 2, 3, 7

C = 1 บิต 1, 3, 4, 5

D = 1 ใน 8 บิต มีการผิดพลาดเป็นจำนวนคี่ บิต

เมื่อในข้อมูล 8 บิต มีการผิดพลาด แต่ บิต 1, 2, 3, 7 (ถูกต้อง) แสดงว่า บิต ที่ผิดพลาดอยู่ใน 0, 4, 5, 6 แต่ D = 1 แสดงว่า บิต ที่ผิดพลาดมีจำนวนคี่บิต เพราะถ้าผิด 2 บิต เมื่อเช็คพาริตีแบบคี่ทุกตัว จะต้องได้ผลว่าถูกต้อง (D=0) บิต ที่เป็น บิต ของ A กับ C ก็คือ

1,5 ดังนั้นจึงพบว่าบิต 5 ผิดเช่นเดียวกับการตรวจจากตาราง

2.5.4 **ไบต์ตัวอักษร (Character bytes)** ไบต์อื่น ๆ อีก 40 ไบต์ ของแต่ละเส้น จะเป็นไบต์ตัวอักษร 1 ไบต์ แทนอักษร 1 ตัว และแทนตัวอักษรได้  $2^7 = 128$  ตัว คือเป็นบิตข้อมูล 7 บิต และพาริตีบิต 1 บิต สำหรับตรวจความผิดพลาดของข้อมูลแบบพาริตีแบบครึ่งหนึ่งของอักษรต่าง ๆ ในการส่งข้อมูลเทเลเท็กซ์ เป็นดังตารางที่ 2.5.4.1 และ 2.5.4.2 ในหน้าที่ 2-11 และรูปที่ 2.5.4.1 ในหน้าที่ 2-12

2.5.5 **เพจเฮดเดอร์ (Page-Header)** เส้นที่ 0 ของข้อมูลทุก ๆ หน้าซึ่งปรากฏบนจอภาพจะเป็นข้อมูลของ

- ชื่อแม่กาซีน 1-8
- หมายเลขหน้า 00-99
- วัน
- เวลา
- บิตควบคุม

ซึ่งมีรูปแบบ (format) ในการส่ง เส้น ที่ 0 หรือ เพจเฮดเดอร์ ซึ่งแตกต่างกับเส้นข้อมูล อื่น ๆ ดังรูปที่ 2.5.5.1 และ 2.5.5.2 ในหน้าที่ 2-13

2.5.6 **รหัสกราฟิก และสี (Graphics Coding and Colour)** นอกจากข้อมูลเทเลเท็กซ์จะแสดงข้อมูลในแบบเท็กซ์ได้แล้ว เรายังสามารถส่งข้อมูลกราฟิกหรือให้สีต่าง ๆ กับอักษรบนจอโทรทัศน์ได้อีกด้วย โดยมีรหัสข้อมูลเป็นดังตารางที่ 2.5.6.1

B7	0	0	0	0	1	1	1	1
B6	0	0	1	1	0	0	1	1
B5	0	1	0	1	0	1	0	1
B4	0	1	2	3	4	5	6	7
0000	NUL	DLE			P			
0001	ALPHA RED	GRAPHICS RED			A	Q		
0010	ALPHA GREEN	GRAPHICS GREEN			B	R		
0011	ALPHA YELLOW	GRAPHICS YELLOW			C	S		
0100	ALPHA BLUE	GRAPHICS BLUE			D	T		
0101	ALPHA MAGENTA	GRAPHICS MAGENTA			E	U		
0110	ALPHA CYAN	GRAPHICS CYAN			F	V		
0111	ALPHA WHITE	GRAPHICS WHITE			G	W		
1000	FLASH	CONCEAL			H	X		
1001	STEADY	CONTIGUOUS GRAPHICS			I	Y		
1010	END BOX	SEPARATED GRAPHICS			J	Z		
1011	START BOX	ESC			K	←		
1100	NORMAL HEIGHT	BLACK BACKGROUND			L	½		
1101	DOUBLE HEIGHT	NEW BACKGROUND			M	→		
1110	'SO	HOLD GRAPHICS			N	↑		
1111	'SI	RELEASE GRAPHICS			O	#		

ตารางที่ 2.5.6.1 แสดงสัญลักษณ์และรหัสทางด้านกราฟิก และตัวควบคุม

B7	0	0	0	0	1	1	1	1
B6	0	0	1	1	0	0	1	1
B5	0	1	0	1	0	1	0	1
9	0	0	0	0				
8	0	0	0	0	Sp	0	Ⓢ	P - R
7	0	0	0	1	!	1	A	Q o q
6	0	0	1	0	"	2	B	R b r
5	0	0	1	1	£	3	C	S c s
4	0	1	0	0	\$	4	D	T d t
3	0	1	0	1	%	5	E	U e u
2	0	1	1	0	&	6	F	V f v
1	0	1	1	1	'	7	G	W g w
0	1	0	0	0		8	H	X h x
0	1	0	0	1		9	I	Y i y
0	1	0	1	0	v	:	J	Z j z
0	1	0	1	1	+	:	K	← k ↵
1	1	0	0	0	.	<	L	1 2 l ll
1	1	0	0	1	-	=	M	→ m 3 4
1	1	0	1	0	.	>	N	↑ n +
1	1	0	1	1	/	?	O	# o ■

Codes in these two columns are for control and are not displayed

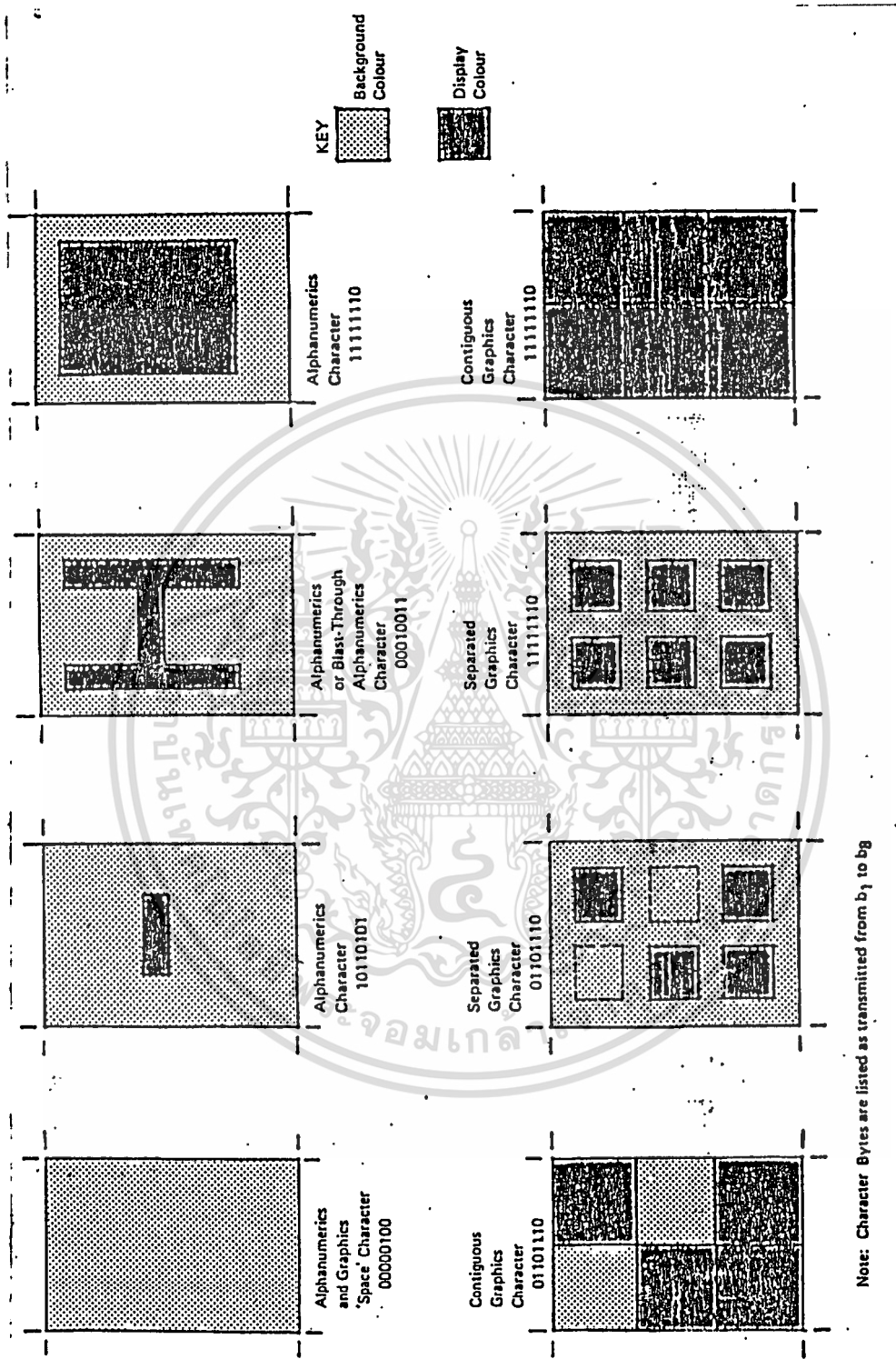
ตารางที่ 2.5.4.1 แสดงสัญลักษณ์ และ โค้ดของเทเลเท็กซ์

DISPLAY MODE	SET AT	SET AFTER	COMPLEMENTARY DISPLAY MODE	SET AT	SET AFTER	SEE SECTION	
ALPHANUMERIC	Row Start	0/4 0/1 0/2 0/3 0/5 0/6 0/7	GRAPHICS	-	1/4 1/1 1/2 1/3 1/5 1/6 1/7	3.1.1.	
CONTIGUOUS	Row Start	1/9*	SEPARATED	1/10*	1/10*	3.1.1.	
DISPLAY COLOUR	includes RED	Row Start 0/1 0/3 0/5 0/7 1/1 1/3 1/5 1/7	DISPLAY COLOUR	excludes RED	-	0/2 0/4 0/6 1/2 1/4 1/6	3.1.2.
	includes GREEN	Row Start 0/2 0/3 0/6 0/7 1/2 1/3 1/6 1/7		excludes GREEN	-	0/1 0/4 0/5 1/1 1/4 1/5	
	includes BLUE	Row Start 0/4 0/5 0/6 0/7 1/4 1/5 1/6 1/7		excludes BLUE	-	0/1 0/2 0/3 1/1 1/2 1/3	
BLACK BACKGROUND	Row Start	-	NEW BACKGROUND	1/13**	-	3.1.3.	
REVEAL	Row Start User Control***	0/4 0/1 0/2 0/3 1/1 1/2 1/3 1/4 1/5 1/5 1/7	CONCEAL	1/8	-	3.1.4.	
STEADY	Row Start	-	FLASH	-	0/8	3.1.4.	
UNBOXED	Row Start	0/10****	BOXED	0/11****	0/11****	3.1.6.	
NORMAL HEIGHT	Row Start	-	DOUBLE HEIGHT	-	0/13	3.1.6.	
RELEASE	Row Start	1/15	HOLD	1/14	-	3.1.7.	

\* these codes may take effect 'at' or 'after' their occurrence,  
 \*\* whenever this code occurs the Display Colour is adopted as the New Background colour,  
 \*\*\* the Reveal mode may be maintained throughout a page by a user control,  
 \*\*\*\* two consecutive codes are transmitted, the mode changes between them.

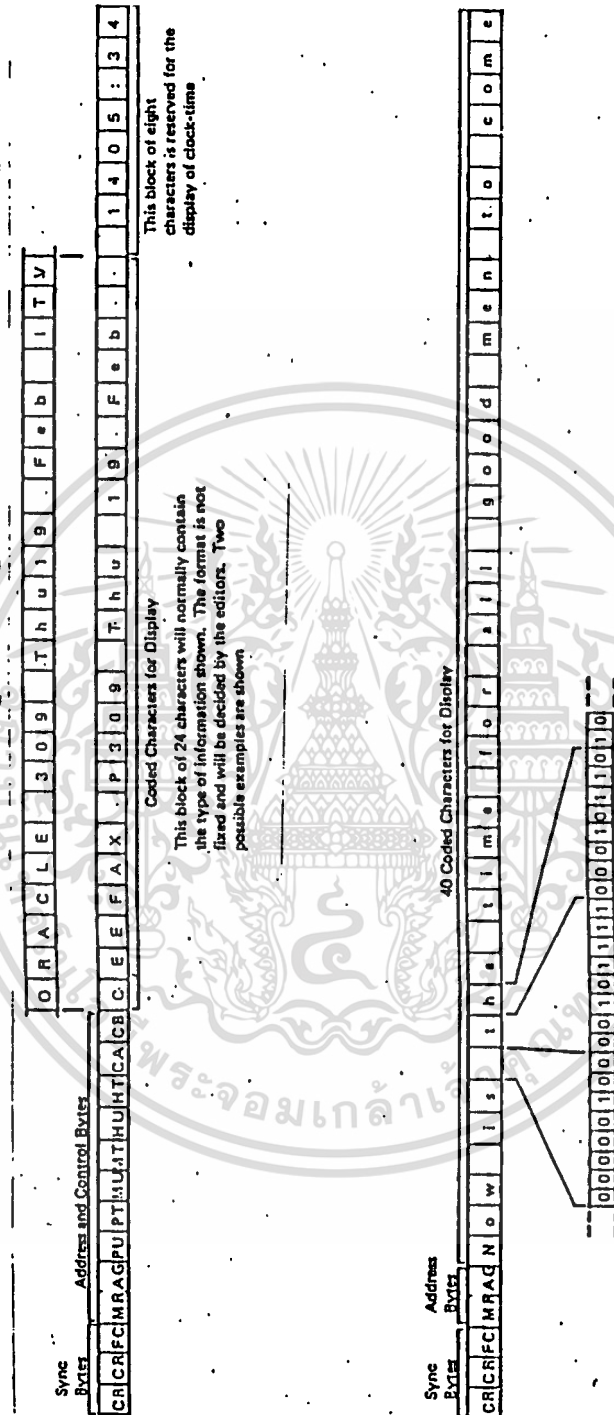
ตารางที่ 2.5.4.2 แสดงโหมดการแสดงผล และรูปแบบอักษร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าการณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.5.4.1 แสดงตัวอย่างโค้ดของตัวอักษร และกราฟิก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



**รูปที่ 2.5.5.1 แสดงแบบมาตรฐานบรรทัดแรกของ 1 หน้า텔레ทัศน์**

**รูปที่ 2.5.5.2 แสดงแบบมาตรฐานของบรรทัดอื่น**

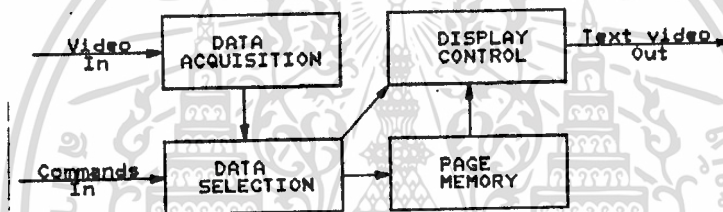
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้拿去ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## หลักทั่วไปของตัวถอดรหัส

ตัวถอดรหัส เป็นอุปกรณ์ในการรับข้อมูลเทเลเท็กซ์ คิดตั้งเข้ากับโทรทัศน์โดยมีการทำงาน ดังนี้

1. แยกสัญญาณเทเลเท็กซ์ ออกจากสัญญาณโทรทัศน์
2. เลือกรับข้อมูลตามความต้องการของผู้ชม
3. นำข้อมูลที่เลือกมาถอดรหัสและจัดเก็บ
4. แปลงข้อมูลนั้นเป็นสัญญาณภาพ เพื่อส่งไปแสดงบนจอโทรทัศน์

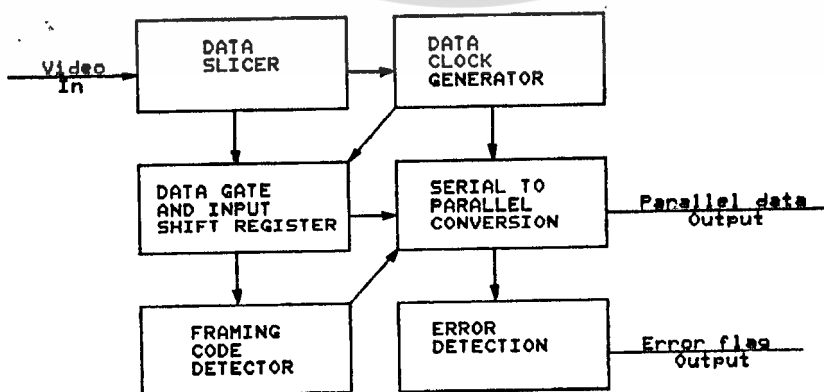
จากการทำงานของตัวถอดรหัส เราสามารถเขียนเป็นบล็อกไดอะแกรม (Block Diagram) ได้ดังนี้



รูปแสดงบล็อกไดอะแกรมของ เครื่องถอดรหัสเทเลเท็กซ์

### 2.6 ส่วนการรับข้อมูล (DATA ACQUISITION)

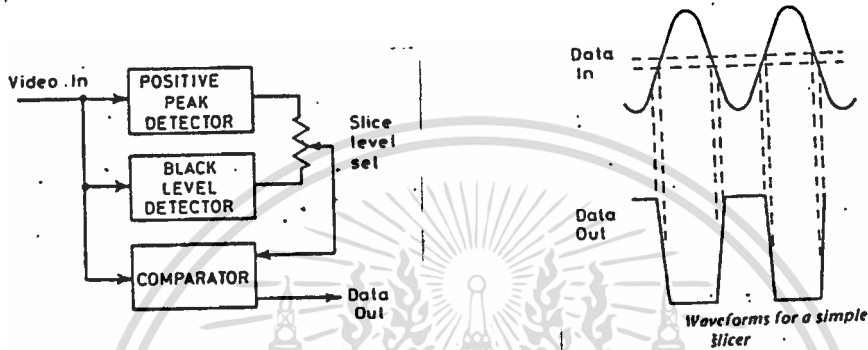
ขั้นตอนแรกของกระบวนการถอดรหัส คือ การแยกเทเลเท็กซ์ออกจากสัญญาณโทรทัศน์ โดยมีขั้นตอนที่เขียนเป็นบล็อกไดอะแกรม การทำงานได้ดังนี้



รูปที่ 2.6.1 แสดงบล็อกไดอะแกรมของส่วนการรับข้อมูล

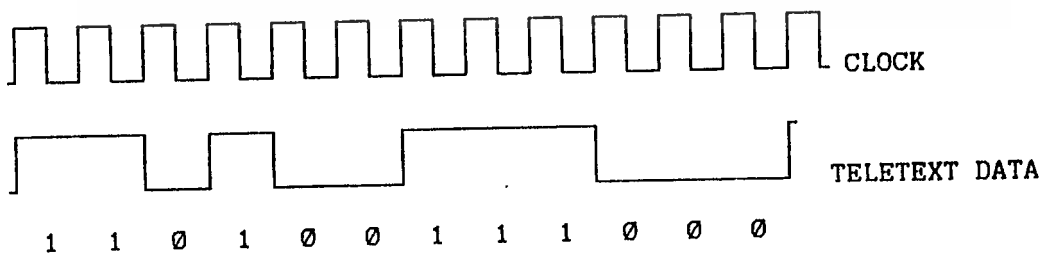
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6.1 ตาต้า สไลเซอร์ (Data Slicer) เนื่องจากสัญญาณข้อมูลเทเลเท็กซ์ที่ส่งมาถึงเครื่องรับจะมีความเพี้ยนไปเป็นรูปไซน์ (sine) และมีระดับ 0% - 66% ของ white level เท่านั้น เราจึงต้องปรับแต่งให้เป็นสัญญาณดิจิทัลที่เหมาะสม คือ เป็นคลื่นรูปสี่เหลี่ยม (Square wave) ที่มีช่วงเวลาขาขึ้น (rise time) และช่วงเวลาตก (fall time) ต่ำ และที่ระดับ logic เป็น 0-5 V. โดยมีบล็อกไดอะแกรมของวงจรเป็นดังรูป 2.6.1.1



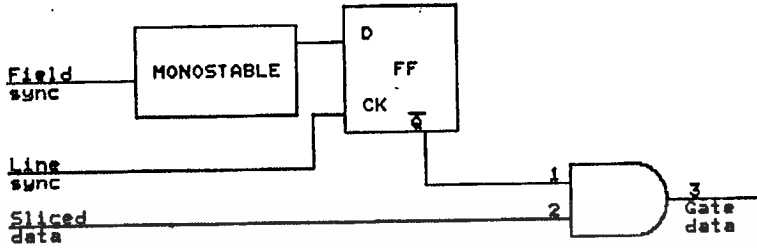
รูปที่ 2.6.1.1 แสดงบล็อกไดอะแกรมของตาต้าสไลเซอร์

2.6.2 ส่วนสร้างสัญญาณนาฬิกา (Data Clock Generator) สัญญาณข้อมูลเทเลเท็กซ์ มีลักษณะเป็นข้อมูลอนุกรมแบบซิงค์โครนัส ดังนั้นการรับข้อมูลจะต้องสร้างสัญญาณนาฬิกา ที่มีความถี่สอดคล้องกับอัตราความเร็วบิต คือ 6.9375 เมกกะเฮิร์ต (MHz) เพื่อให้จังหวะการรับข้อมูลตรงกับจังหวะการส่งข้อมูล จึงจะได้ข้อมูลที่ถูกต้อง



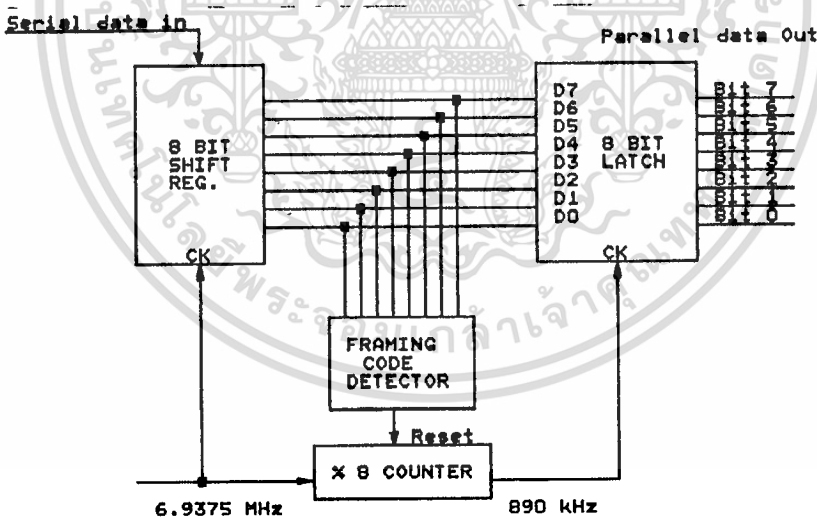
รูปที่ 2.6.2.1 แสดงสัญญาณนาฬิกาที่ตรงกับข้อมูลเทเลเท็กซ์

2.6.3 ดาต้า เกจ (Data gate) วงจรส่วนนี้จะทำหน้าที่เป็นประตูที่ยอมให้ สัญญาณข้อมูลเทเลเท็กซ์ผ่านไปได้เท่านั้น โดยสัญญาณอื่น ๆ ที่ไม่ต้องการจะถูกตัดออกไป



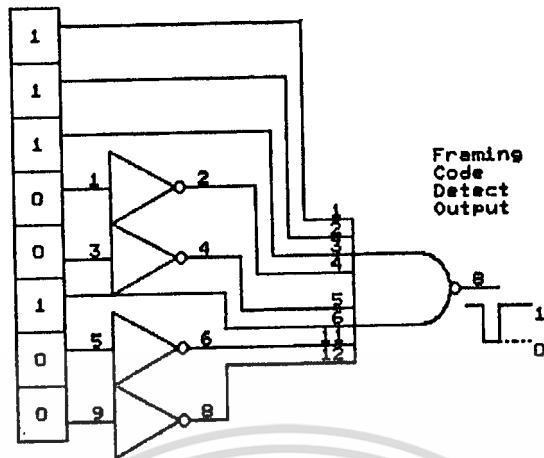
รูปที่ 2.6.3.1 แสดงบล็อกไดอะแกรมของดาต้า เกจ

2.6.4 ส่วนแปลงข้อมูลอนุกรมเป็นข้อมูลขนาน (Serial to Parallel) เป็น วงจรที่ทำหน้าที่รับข้อมูลจากดาต้า เกจ ที่อยู่ในรูปข้อมูลอนุกรมมาแปลงให้เป็น ข้อมูลขนาน คือ รับข้อมูลที่ละ 1 บิต เมื่อครบ 1 ไบต์ ก็ส่งเป็นแบบขนานไปยังส่วนต่อไป



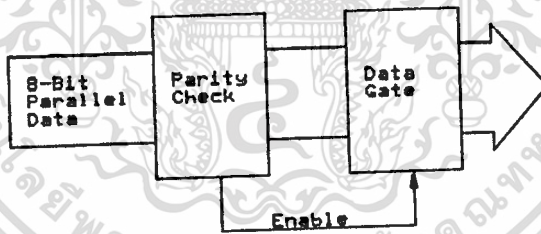
รูปที่ 2.6.4.1 แสดงบล็อกไดอะแกรมของการแปลงข้อมูลอนุกรมเป็นข้อมูลขนาน

2.6.5 การตรวจจับรหัสเฟรมมิ่ง (Framing Code Detect) เป็นส่วนที่ตรวจสอบรหัสเฟรมมิ่ง เพื่อบอกการเริ่มต้นรับข้อมูลในแต่ละเส้น และยังเป็นการตรวจสอบการตรงกัน ของไบต์ข้อมูล



รูปที่ 2.6.5.1 แสดงบล็อกโดยแกรมของการตรวจรับรหัสเฟรมมิ่ง

2.6.6 การตรวจจับข้อมูลผิดพลาด (Error Detection) เป็นวงจรลอจิกที่ตรวจสอบข้อมูลที่ได้รับว่ามีความผิดพลาดเกิดขึ้นหรือไม่ ถ้าข้อมูลผิดพลาดก็จะไม่ยอมรับข้อมูลไว้ โดยใช้วิธีตรวจพาริตรีแบบคี่ของข้อมูล



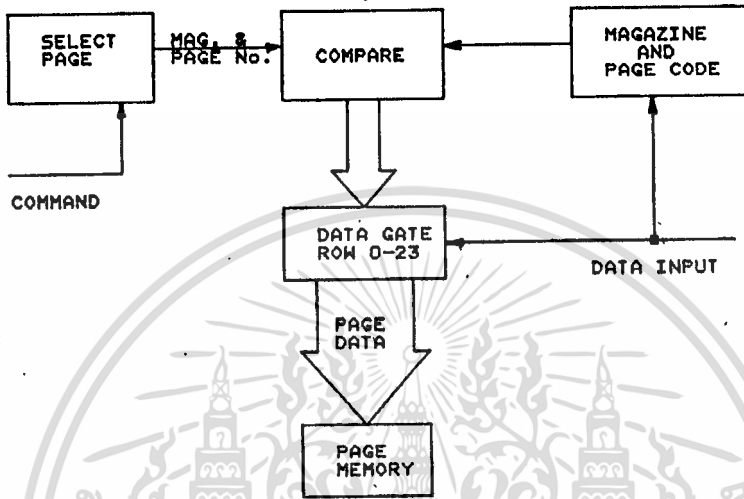
รูปที่ 2.6.6.1 แสดงบล็อกโดยแกรมของการตรวจจับข้อมูลผิดพลาด

**2.7 ส่วนการเลือกข้อมูล (DATA SELECTION)**

เป็นส่วนที่มีหน้าที่รับคำสั่งจากผู้ขมาที่ต้องการดูข้อมูลจากแม่กกาซินใด หน้าที่เท่าไร ส่วนการเลือกข้อมูลนี้จะค้นหาข้อมูลตำแหน่งนั้น โดยตรวจจากข้อมูลไบต์ ที่ 4 และ 5 ของข้อมูลแต่ละเส้น ที่รับมาจากส่วนการรับข้อมูล (Data Acquisition) ซึ่งจะบอกหมายเลขแม่กกาซินและเลขบรรทัด เมื่อพบว่าเป็นบรรทัดที่ 0 ของแม่กกาซินที่ต้องการแล้ว ก็จะตรวจสอบต่อไปว่าเป็นบรรทัดที่ 0 ของหน้าที่ต้องการหรือไม่ โดยตรวจจากไบต์ ที่ 6 และ 7 ของบรรทัดที่ 0 ซึ่งบอกเลขหน้า เมื่อได้ตำแหน่งที่ต้องการก็จะรับข้อมูลตั้งแต่เส้นที่ 0 จนถึงเส้นสุดท้ายของหน้า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

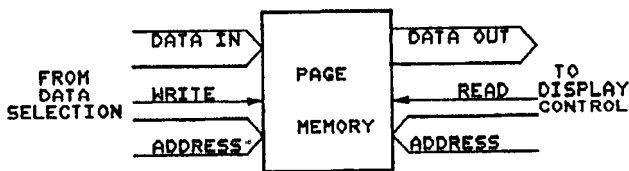
(23) ส่งไปเก็บไว้ในหน่วยความจำ (Page Memory) ถ้าไม่มีคำสั่งใหม่ส่วนการเลือกข้อมูลนี้ก็จะรับข้อมูลตำแหน่งเดิมอีกครั้ง เมื่อทางสถานีโทรทัศน์ส่งข้อมูลวนมาอีกรอบ (สถานีจะส่งข้อมูลเป็นอนุกรมจากหน้าที่ 100 ถึง 899 แล้ววนส่ง 100 ใหม่) ถ้าสถานีมีการแก้ไขข้อมูลหน้านั้น ผู้ชมก็จะได้รับข้อมูลใหม่ของหน้านั้นด้วย ซึ่งเรียกว่าการอัปเดต (UPDATE) ข้อมูล



รูปที่ 2.7.1 แสดงบล็อกไดอะแกรมของส่วนการเลือกข้อมูล

**2.8 ส่วนการเก็บข้อมูล (PAGE MEMORY)**

ข้อมูลที่ถูกเลือกทั้ง 24 บรรทัดจะถูกนำมาเก็บไว้ในหน่วยความจำเรียกว่า เพจ-เมมโมรี่ (Page Memory) เพื่อนำไปถอดรหัสต่อไป ดังนั้นหน่วยความจำที่ใช้จะต้องเก็บข้อมูลทั้งหมดหนึ่งหน้าได้ คือมีความจุไม่น้อยกว่า 24 บรรทัด x 40 ไบต์ = 960 ไบต์ หรือ 7 x 960 = 6720 บิต รวมทั้งหน่วยความจำส่วนที่เก็บตำแหน่งของแม็กกาซีน และหน้าที่ผู้ชมเลือกด้วย



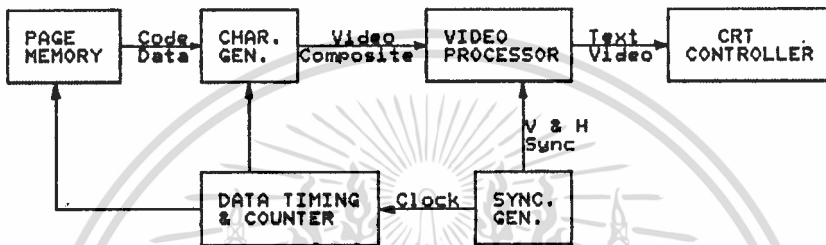
รูปที่ 2.8.1 แสดงบล็อกไดอะแกรมของส่วนหน่วยความจำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## 2.9 ส่วนการแสดงผล (DISPLAY CONTROL)

เป็นวงจรส่วนที่นำข้อมูลเทเลทีกซ์ ไปแสดงบนจอโทรทัศน์โดยมีกระบวนการ ดังนี้

- ข้อมูลจากหน่วยความจำมาแปลงเป็นสัญญาณภาพ และสัญญาณสี ตรงตามความหมายของรหัสข้อมูล
- ควบคุมการสแกนจอภาพให้สามารถนำสัญญาณภาพไปปรากฏบนจอโทรทัศน์ได้ถูกต้อง โดยการสร้างสัญญาณซิงค์ขึ้นมากำหนดการสแกนใหม่



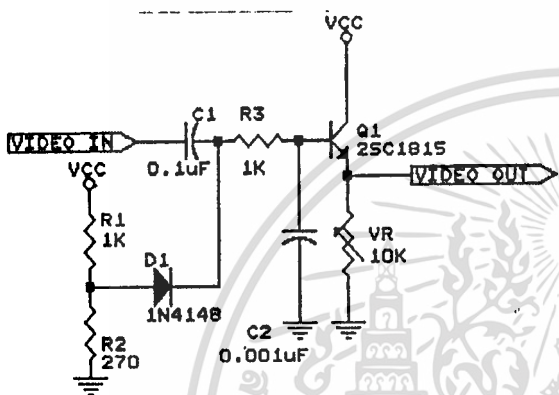
รูปที่ 2.9.1 แสดงบล็อกไดอะแกรมของส่วนควบคุมการแสดงผล

## บทที่ 3.

## การคำนวณ และการสร้าง

วงจรต่าง ๆ ที่ได้ออกแบบไว้ทั้งในส่วนของฮาร์ดแวร์ (อุปกรณ์ภายนอก) ส่วนที่เป็นวงจรเชื่อมต่อกับคอมพิวเตอร์ (อินเทอร์เฟซ) และส่วนของโปรแกรมมีดังนี้

## 3.1 วงจรยกระดับติชิตี (DC) คงที่



รูปที่ 3.1.1 แสดงวงจรยกระดับติชิตีคงที่

สัญญาณภาพที่ส่งมาจากสถานีจะมีระดับติชิตีเปลี่ยนแปลงตลอดเวลา เราจึงทำการตัดติชิตีออก แล้วยกระดับติชิตีให้คงที่ด้วยวงจรดังรูปที่ 3.1.1 ซึ่งในรูปแบบนี้จะมีเฟลเฟอร์ (buffer) อยู่ด้วยเพื่อป้องกันไม่ให้สัญญาณมีความเพี้ยนเมื่อมีการนำสัญญาณมาใช้กับวงจรทั้งหมด

## 3.2 วงจรแยกเชิงคี่และสัญญาณเทเลเท็กซ์

เนื่องจากความถี่ของสัญญาณเชิงคี่ทางแนวนอนและแนวตั้งมีความถี่ต่างกัน โดยสัญญาณเชิงคี่ทางแนวนอนมีความถี่ = 15,625 Hz ส่วนสัญญาณเชิงคี่ทางแนวตั้งมีความถี่ = 50 Hz เราจึงสามารถแยกสัญญาณทั้ง 2 ออกจากกันได้ ซึ่งสัญญาณเชิงคี่ทางแนวนอนแยกได้โดยใช้วงจรดิฟเฟอเรนเชียลดิฟเฟอเรนเชียล (differentiator) และสัญญาณเชิงคี่ทางแนวตั้งแยกได้โดยใช้วงจรอินทิเกรตเตอร์ (Integrator) แล้วนำสัญญาณทั้ง 2 ที่แยกได้มาทริกโมโนสเตเบิล ซึ่งกำหนดคาบไว้แล้ว ในส่วนของสัญญาณเชิงคี่ทางแนวตั้งนั้นนำไปทริกโมโนสเตเบิลที่กำหนดคาบไว้เป็นช่วงของสัญญาณแบบสังคี่ทางแนวตั้ง เพื่อนำสัญญาณนี้ไปตัดสัญญาณภาพออกจากสัญญาณเทเลเท็กซ์ ดังรูปที่ 3.2.1 ในหน้า 3-2

## 3.3 วงจรตัดสัญญาณทดสอบ

เราสามารถแยกสัญญาณทดสอบออกจาก สัญญาณเทเลเท็กซ์ได้โดยใช้ ไอซีดีดี-เตอร์ 74LS154 ซึ่งมีการทำงานดังตารางที่ 3.3.1

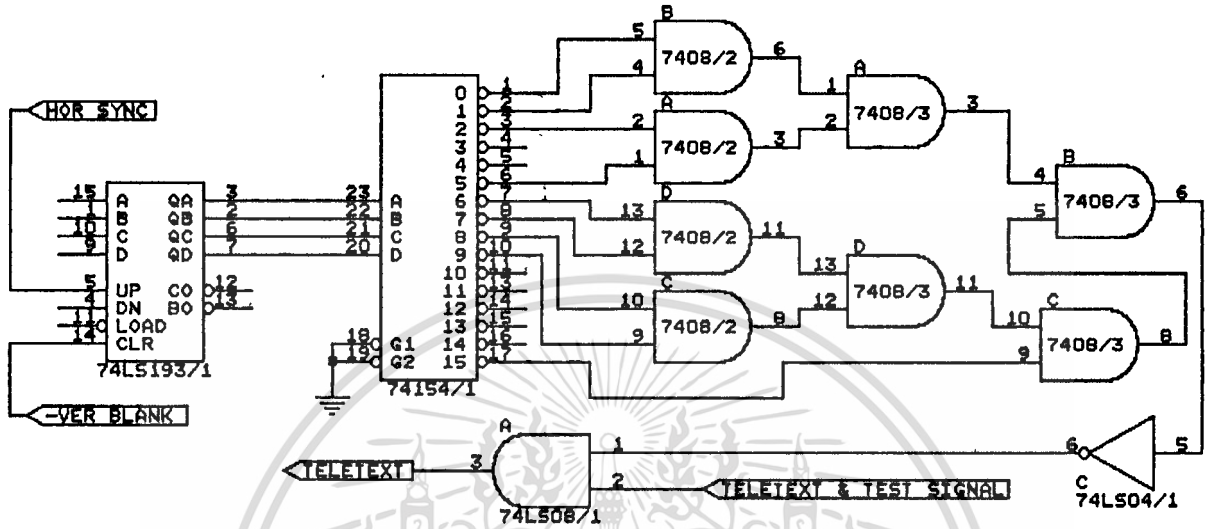
เอาต์พุตของ 154 จะขึ้นอยู่กับอินพุตที่เป็นเลขฐานสอง ดังนั้นเราจึงนำเอาสัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





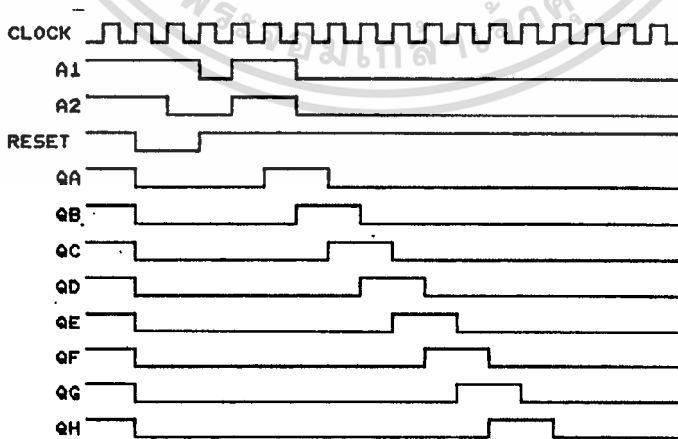
ซึ่งค้ทางแนวนอน มาเป็นตัวทริกเค้ทเตอร์ (74LS193) เพื่อให้ท้การนับ (สัญญาณซึ่งค้แนวนอน) ไปเรื่อย ๆ เฉพาะในช่วงของสัญญาณแบล้กซึ่งค้ทางแนวดั้งเท้านั้น เพราะท้เราน้เอาสัญญาณแบล้กซึ่งค้ทางแนวดั้งมาเข้าข้ Clear ของ 193 ต้วย ดังรูปที่ 3.3.1



รูปที่ 3.3.1 แสดงวงจรถัดสัญญาณทดสอบออกจากสัญญาณเทเลเท็กซ์

### 3.4 วงจรถแปลงสัญญาณจากอนุกรมเป็นขนาน (serial to parallel) และ วงจรถรวจจับรหัสเฟรมมิ่ง (Framing detector)

เราสามารถแปลงข้อมูลอนุกรมเป็นข้อมูลขนานได้โดยใช้ ไอซีเบอร์ 74LS164 ซึ่งมีแผนผังเวลาการทำงานดังรูป

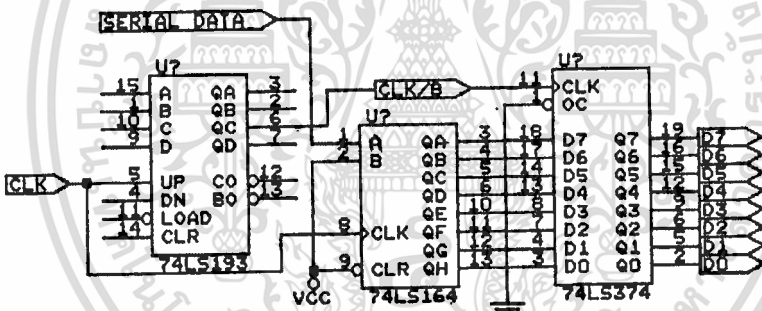


รูปแสดงแผนผังการทำงานของไอซีเบอร์ 74LS164


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปข้อมูลอนุกรมจะเข้ามาทางขา A<sub>1</sub> และจะถูกแปลงให้เป็นข้อมูลแบบขนาน โดยการเลื่อนข้อมูลไปตามขาเอาต์พุต Q<sub>A</sub>-Q<sub>H</sub> การเลื่อนข้อมูลจะทำทุก ๆ ขาขึ้นของสัญญาณนาฬิกาที่ป้อนให้กับไอซี เมื่อข้อมูลอนุกรมบิตแรกถูกเลื่อนให้ไปปรากฏที่ขาเอาต์พุต Q<sub>H</sub> ข้อมูลอนุกรมบิตสุดท้ายจะถูกเลื่อนให้ไปปรากฏที่ Q<sub>A</sub> พอดี ณ.เวลานี้ การแปลงข้อมูลจากขนานเป็นอนุกรมจะเสร็จสิ้น ได้ข้อมูลแบบขนาน 1 ไบต์ ณ.ช่วงเวลานี้ เราจะนำข้อมูล ณ.ช่วงเวลานี้ไปเก็บไว้ในหน่วยความจำ ซึ่งหมายถึงค่าแอดเดรสที่ป้อนไปยังหน่วยความจำต้องมีการเปลี่ยนแปลง ณ.ช่วงเวลานี้พอดี

จากลักษณะการทำงานของ 74LS164 เราจึงต้องใช้สัญญาณนาฬิกาที่มีความถี่ตรงกับอัตราเร็วของบิตของสัญญาณเทเลเท็กซ์ คือ 6.9375 MHz เพื่อให้การเลื่อนข้อมูลเป็นไปตามลำดับ ไม่มีการกระโดดของข้อมูล และนำเอาสัญญาณนาฬิกามาผ่านเคาท์เตอร์เบอร์ 74LS193 เพื่อหาร 8 และใช้ควบคุมไอซี 74LS374 ซึ่งเป็นตัวแลทช์ (latch) ข้อมูล เพื่อส่งต่อไปยังหน่วยความจำต่อไป ดังรูปที่ 3.4.1

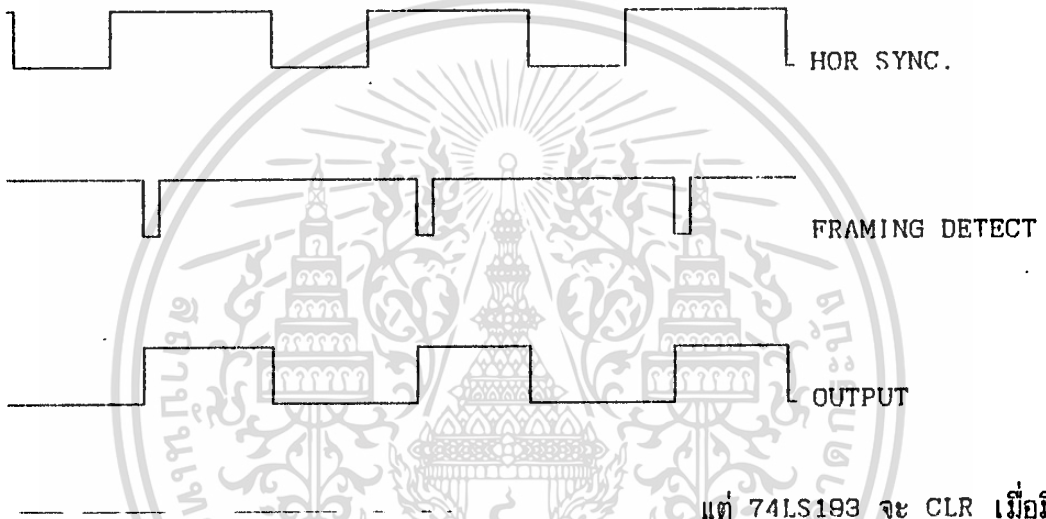


รูปที่ 3.4.1 แสดงวงจรแปลงข้อมูลอนุกรมเป็นข้อมูลขนาน

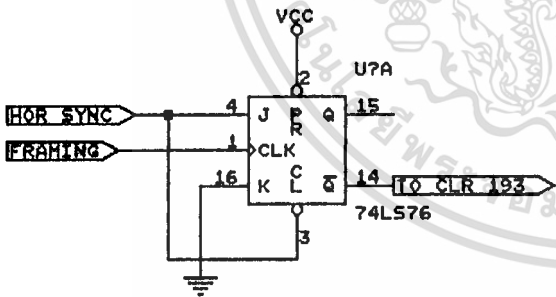
สำหรับบางจรรยาจับเฟรมมิ่ง (Framing detector) นั้น ข้อมูลของรหัสเฟรมมิ่งคือ 11100100 เป็นรหัสมาตรฐานเราจึงนำบิต 0 ทั้งหมดมาผ่านน็ทเกต (not gate) เพื่อแปลงสัญญาณให้เป็นบิต 1 แล้วนำสัญญาณรวมทั้งหมดมาผ่านแนนเกต (nand gate) เพื่อว่าเมื่อมีรหัสเฟรมมิ่งเข้ามาก็จะได้ลักษณะของเอาต์พุต เป็นดังรูป  และวงจรถูกเป็นดังรูป 2.6.5.1 ในหน้า 2-18

สัญญาณเทเลเท็กซ์ จะมีเป็นช่วง ๆ เท่านั้น แต่สัญญาณคล็อกจะมีอยู่ตลอดเวลา เพราะฉะนั้น เมื่อถึงช่วงที่ไม่มีสัญญาณเทเลเท็กซ์ จะทำให้ RAM เก็บข้อมูลที่ไมทราบเข้าไปด้วย ดังนั้นเพื่อหลีกเลี่ยงปัญหาที่เกิดขึ้น เราจึงต้องทำให้เคาท์เตอร์ (74LS193) ที่ใช้ในการหาร 8 นั้น

รีเซ็ต (Reset) ตัวเอง เมื่อมีสัญญาณซิงค์แนวนอนเข้ามาและเซ็ท (set) ตัวเองเมื่อตรวจจับรหัสเฟรมมิ่งได้ แล้วนำสัญญาณที่ได้นี้ไปเข้าขาเคลียร์ (CLR) ของ 74LS193 วงจรที่จะทำการรีเซ็ต เมื่อมีสัญญาณซิงค์แนวนอนเข้ามา และเซ็ทเมื่อมีเฟรมมิ่งเข้ามานั้นเราจะใช้ไอซีเบอร์ 74LS76 ซึ่งเป็นเจเค-ฟลิปฟลอป (JK FLIP-FLOP) โดยเอาขาเจ (J) และเคลียร์ (CLR) ต่อกับสัญญาณซิงค์แนวนอน และขาคล็อก (CLK) ต่อกับตัวตรวจจับเฟรมมิ่ง โดยเอาขา K ต่อลงกราวด์ (GND) และขาเซ็ท (set) ต่อขึ้นไฟบวก (Vcc) ซึ่งจะมีลักษณะการทำงานดังแผนภาพข้างล่างนี้



แต่ 74LS193 จะ CLR เมื่อมี "H" เข้ามาและทำงานเมื่อมี "L" เข้ามา เพราะฉะนั้นเราจะต้องเอาขา -Q ต่อเข้ากับขาเคลียร์ของ 74LS193 ดังรูปที่ 3.4.2 และวงจรรวมทั้งหมดของภาคนี้เป็นดังรูปที่ 3.4.3 ในหน้าที่ 3-7



รูปที่ 3.4.2 วงจรควบคุมเคาท์เตอร์

### 3.5 วงจรเก็บข้อมูลลงหน่วยความจำ

ในส่วนนี้เราใช้หน่วยความจำ คือ สแตติกแรม (SRAM) ขนาด 8K ซึ่งมีแอดเดรสตั้งแต่  $A_0 - A_{12}$  ทำให้ต้องใช้เคาท์เตอร์ (74LS193) 3ตัว และเจเค-ฟลิปฟลอป 1 ตัว มาต่อเป็นวงจรนับ เพื่อกำหนดแอดเดรสในการเก็บข้อมูลเข้าสู่หน่วยความจำ โดยที่ใช้สัญญาณ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่นอนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



คล็อกที่หาร 8 จากวงจรอนุกรมเป็นขนาน มาเป็นตัวควบคุมการทำงานของวงจร นั่นคือเมื่อสัญญาณคล็อกหาร 8 มาข้อมูลที่แปลงจากอนุกรมเป็นขนานจะเข้าสู่แลตช์ (74LS374) ขณะเดียวกันก็จะเข้าสู่หน่วยความจำ และคล็อกหาร 8 นี้ก็จะทำให้แอดเดรสเพิ่มขึ้นอีก 1 บิต เพื่อเป็นการรื้อข้อมูลในไบต์ถัดไป

วงจรในส่วนนี้เราได้เอาแอดเดรสบัส และดาตาบัสทั้งหมดของหน่วยความจำต่อกับ 74LS245 ซึ่งเป็นบัลทรานซีฟเวอร์ที่มีเอาต์พุต 3 สถานะ. (Bus transceivers-with 3-state o/ps) เพื่อใช้ในการตัดระบบทั้งหมดออกจากหน่วยความจำ ให้คอมพิวเตอรืเข้าควบคุมการทำงานแทน โดยเราจะมีการตั้งค่าแอดเดรสสุดท้ายของเทเลเท็กซ์ ซึ่งแอดเดรสสุดท้ายคือ 1004 เรานำเอาแอดเดรสนี้มาแปลงเป็นเลขฐานสอง จะทำให้ทราบว่ามีค่า  $A_9, A_8, A_7, A_6, A_5, A_4$  และ  $A_2$  ที่มีค่า "1" เรานำเอาแอดเดรสดังกล่าวมาเข้า แนนเกจ (nand gate) ซึ่งจะให้เอาต์พุตเป็น "0" เมื่ออินพุตที่เข้ามาเป็น "1" ทั้งหมด ดังนั้นเราจึงนำสัญญาณของแนนเกจนี้ไปบอกคอมพิวเตอรืว่ามีการเก็บข้อมูลครบ 1 หน้าแล้วให้คอมพิวเตอรืส่งสัญญาณตอบกลับมา บอกไอซี 74LS245 เพื่อตัดระบบทั้งหมดออกจาก หน่วยความจำเหลือแต่การควบคุมจากคอมพิวเตอรืเพียงอย่างเดียว ดังรูปที่ 3.5.1 ในหน้าที่ 3-9

### 3.6 วงจรป้องกันข้อมูลผิดพลาด (Correction of Hamming coded data)

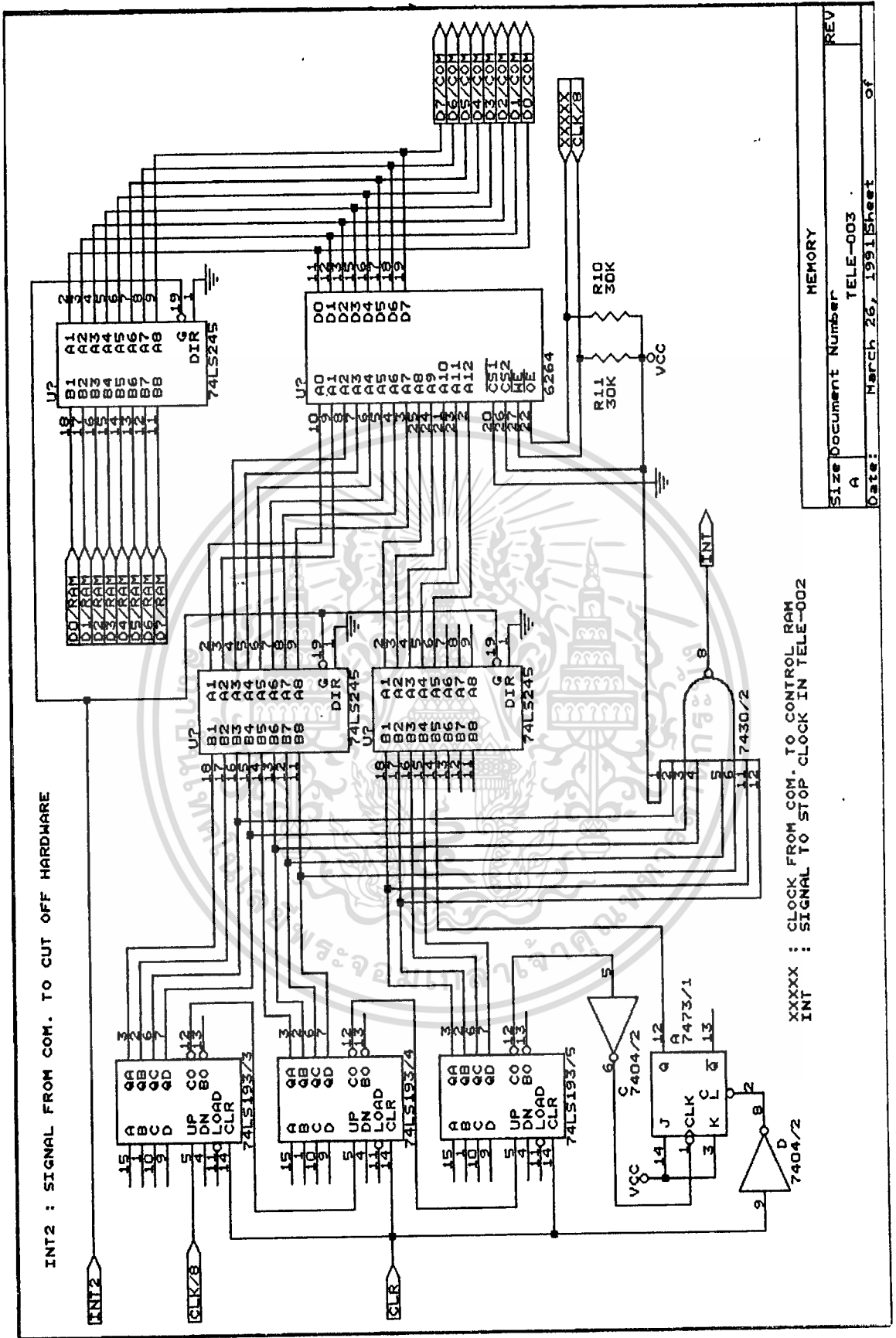
ในสัญญาณเทเลเท็กซ์ข้อมูลในแถวแรก (page Header) จะส่งมาเป็นข้อมูลแบบรหัสแฮมมิง ข้อมูลในส่วนนี้จะเป็นส่วนที่สำคัญสำหรับเทเลเท็กซ์ในแต่ละหน้า มีตั้งแต่แม็กกาซีนและแอดเดรสแถว (Magazine and Row address) เลขหน้าหลักหน่วย (Page units) เลขหน้าหลักสิบ (Page tens) และอื่น ๆ ดังนั้นเพื่อป้องกันข้อมูลที่ส่งมามีความผิดพลาดจึงต้องมีวงจรที่จะนำบิตข้อมูลที่ผิดมาแก้ไขให้ถูกต้องได้เองโดยอัตโนมัติ ซึ่งวงจรดังกล่าวเป็นดังรูปที่ 3.6.1 ในหน้าที่ 3-10

โดยในบล็อก A จะทำการตรวจสอบบิต 1,3,4,5

โดยในบล็อก B จะทำการตรวจสอบบิต 1,2,3,7

โดยในบล็อก C จะทำการตรวจสอบบิต 0,1,5,7

ทั้งหมดจะให้เอาต์พุตเป็น "0" เมื่อข้อมูลที่เข้ามาถูก และเป็น "1" เมื่อข้อมูลที่เข้ามาผิดจากนั้นจะเอาเอาต์พุตทั้ง 3 ต่อกับวงจรรีอ (not) และแอนด์เกจ (and gate) เพื่อไปควบคุมในการกลับบิตข้อมูลที่ผิดให้ถูกต้องไป โดยจะกลับบิตก็ต่อเมื่อมี "1" มารอที่อินพุตของวงจรในบล็อก D

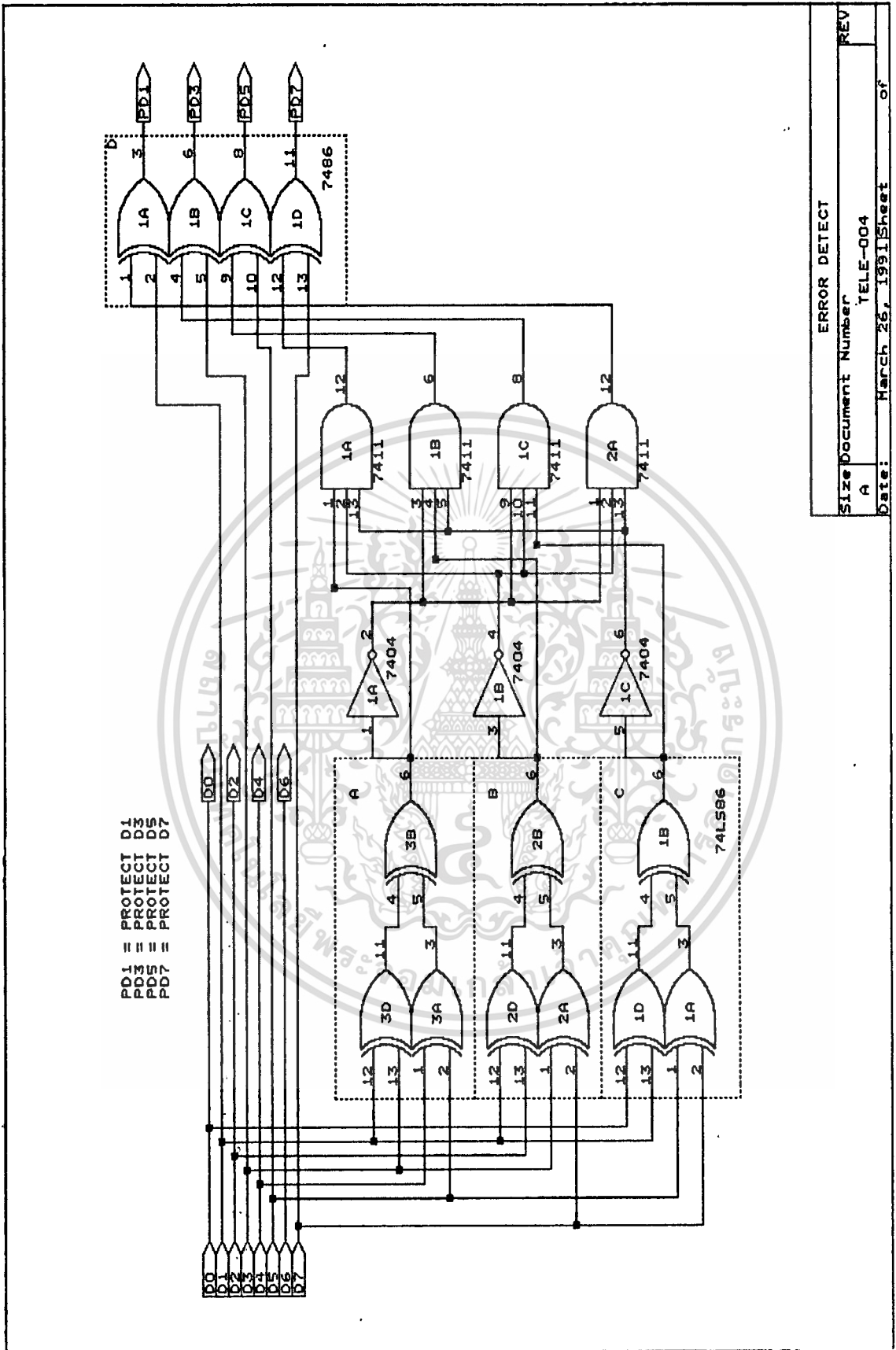


XXXXX : CLOCK FROM COM. TO CONTROL RAM  
 INT : SIGNAL TO STOP CLOCK IN TELE-002

รูปที่ 3.5.1 แสดงวงจรรวมทั้งหมดในส่วนของการเก็บข้อมูลลงหน่วยความจำ (RAM)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีโดยทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Size	Document Number	TELE-003	REV
A			
Date:	March 26, 1991		Sheet of



ERROR DETECT	
Size Document Number	REV
A	TELE-004
Date:	March 26, 1991
Sheet	of

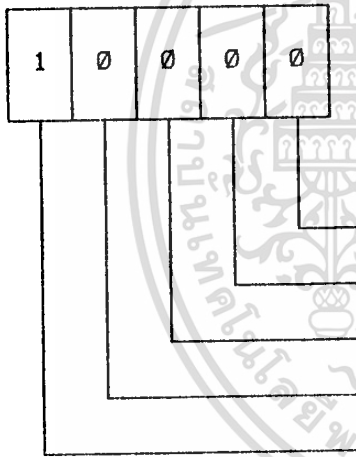
รูปที่ 3.6.1 แสดงวงจรป้องกันข้อมูลผิดพลาด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

**3.7 วงจรหาแม็กกาซีน แอดเดรสแถว และหน้าของเทเลเท็กซ์**

ข้อมูลแอดเดรสของแถว (row address data) ซึ่งเป็นข้อมูล 2 byte ต่อจากรหัสเฟรมมิ่ง เราสามารถตรวจจับข้อมูลแอดเดรสของแถว โดยนำเอาสัญญาณรหัสเฟรมมิ่งที่ตรวจจับได้เข้ารีจิสเตอร์ ซึ่งจะมีการทำงานดังนี้

เมื่อเราจับสัญญาณรหัสเฟรมมิ่งได้ สัญญาณนี้จะเข้ารีจิสเตอร์ ทำให้มีสถานะเป็น 1 ในสภาวะแรก และจะมีสถานะ 0 ในสภาวะอื่น ชิพรีจิสเตอร์ตัวนี้จะใช้สัญญาณคล็อก 8 ในส่วนของเปลี่ยนข้อมูลจากอนุกรมเป็นขนาน เป็นตัวควบคุมแต่ละสัญญาณคล็อก 1 ลูก สถานะ "1" จะเลื่อนไปยังสภาวะอื่น นั่นคือเมื่อสถานะ "1" ถูกเลื่อนไปสภาวะที่ 2 ในช่วงนี้จะเป็นช่วงของแม็กกาซีนเข้ามาในวงจรแปลงอนุกรมไปเป็นขนาน และเมื่อสถานะ "1" เลื่อนไปสู่สภาวะที่ 3 ก็จะเป็นช่วงของเลขหน้าหลักหน่วยเข้ามาในวงจรแปลงอนุกรมไปเป็นขนานเช่นกัน จะเป็นลักษณะเช่นนี้ไปเรื่อย ๆ เราจะนำเอาสถานะ "1" ที่มีการเลื่อนไปนี้ใช้ควบคุมข้อมูลแอดเดรส 4 บิต ดังรูปที่ 3.7.1



- เป็น 1 เมื่อเลขหน้าหลักสิบเข้ามา (ไบต์ที่ 7)
- เป็น 1 เมื่อเลขหน้าหลักหน่วยเข้ามา (ไบต์ที่ 6)
- เป็น 1 เมื่อแอดเดรสแถวเข้ามา (ไบต์ที่ 5)
- เป็น 1 เมื่อแม็กกาซีนเข้ามา (ไบต์ที่ 4)
- เป็น 1 เมื่อตรวจจับรหัสเฟรมมิ่งได้ (ไบต์ที่ 3)

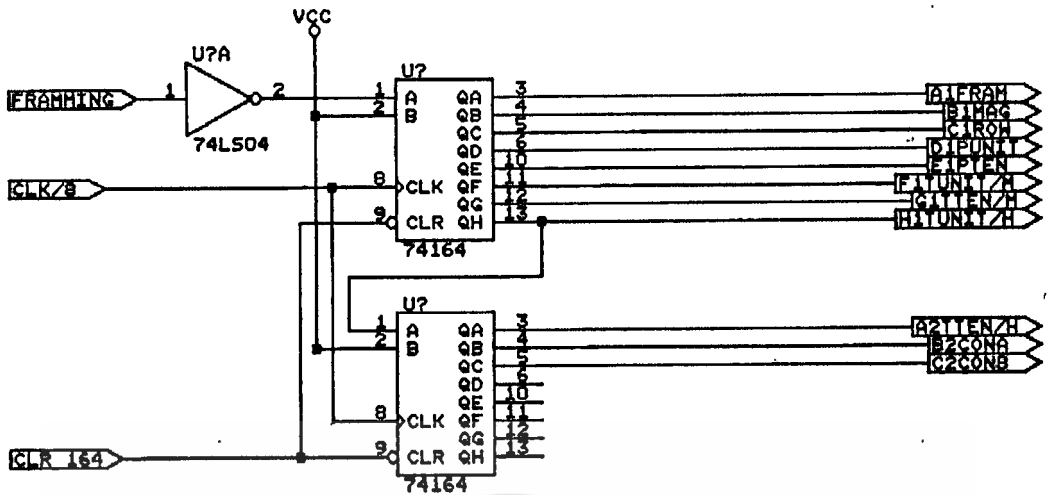
รูปที่ 3.7.1 แสดงการทำงานของชิพรีจิสเตอร์ของรหัสเฟรมมิ่ง

จากแผนภาพและความต้องการสัญญาณดังกล่าวเราจึงสร้างวงจรในส่วนนี้ได้ดังรูปที่ 3.7.2 ในหน้าที่ 3-12 โดยเราจะต้องให้สถานะตอนแรกของไอซี 74LS164 เป็น "0" ทั้งหมดซึ่งเรานำเอาสัญญาณจากโมโนสเตเบิลมาเคลียร์ (จากวงจร 3.8.1) ก่อน

เพราะฉะนั้นวงจรในส่วนที่ใช้ตรวจจับแม็กกาซีน แอดเดรสแถว เลขหน้าหลักหน่วย เลขหน้าหลักสิบ และอื่น ๆ จะมีลักษณะของวงจรคล้ายกัน เพียงแต่เปลี่ยนตัวควบคุมดังรูปที่

3.7.3 ในหน้าที่ 3-13 และรูปที่ 3.7.4 ในหน้าที่ 3-14

จะเห็นว่าที่ขาคล็อก (CLK) ของดี-ฟลิปฟล็อป (D-Flip-Flop) ในส่วนของวงจร



รูปที่ 3.7.2 แสดงวงจรชิฟต์รีจิสเตอร์ของรหัสเฟรมมิ่ง

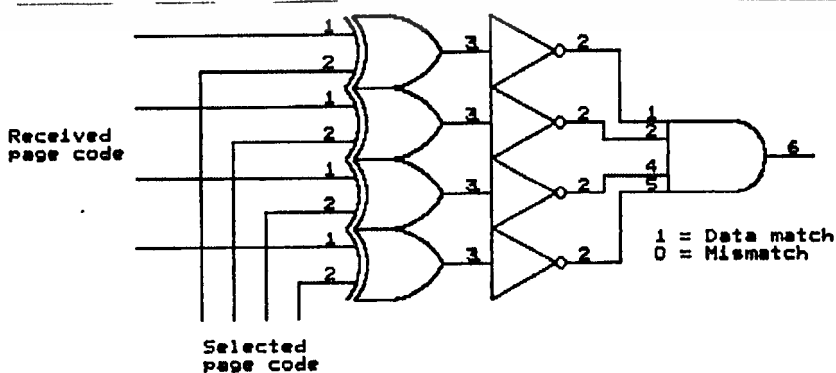
ตรวจจับแม็กกาซีน แอดเดรสแถว เลขหน้าหลักหน่วย เลขหน้าหลักสิบ จะใช้คนละตัวเพื่อให้ อินพุทของฟลิปฟล็อปมาปรากฏที่เอาต์พุทที่เวลาต่างกัน แล้วนำสัญญาณเอาต์พุทนี้ไปเปรียบเทียบกับ แม็กกาซีน เลขหน้าหลักหน่วย เลขหน้าหลักสิบที่ส่งมาจากคอมพิวเตอรืต่อไป

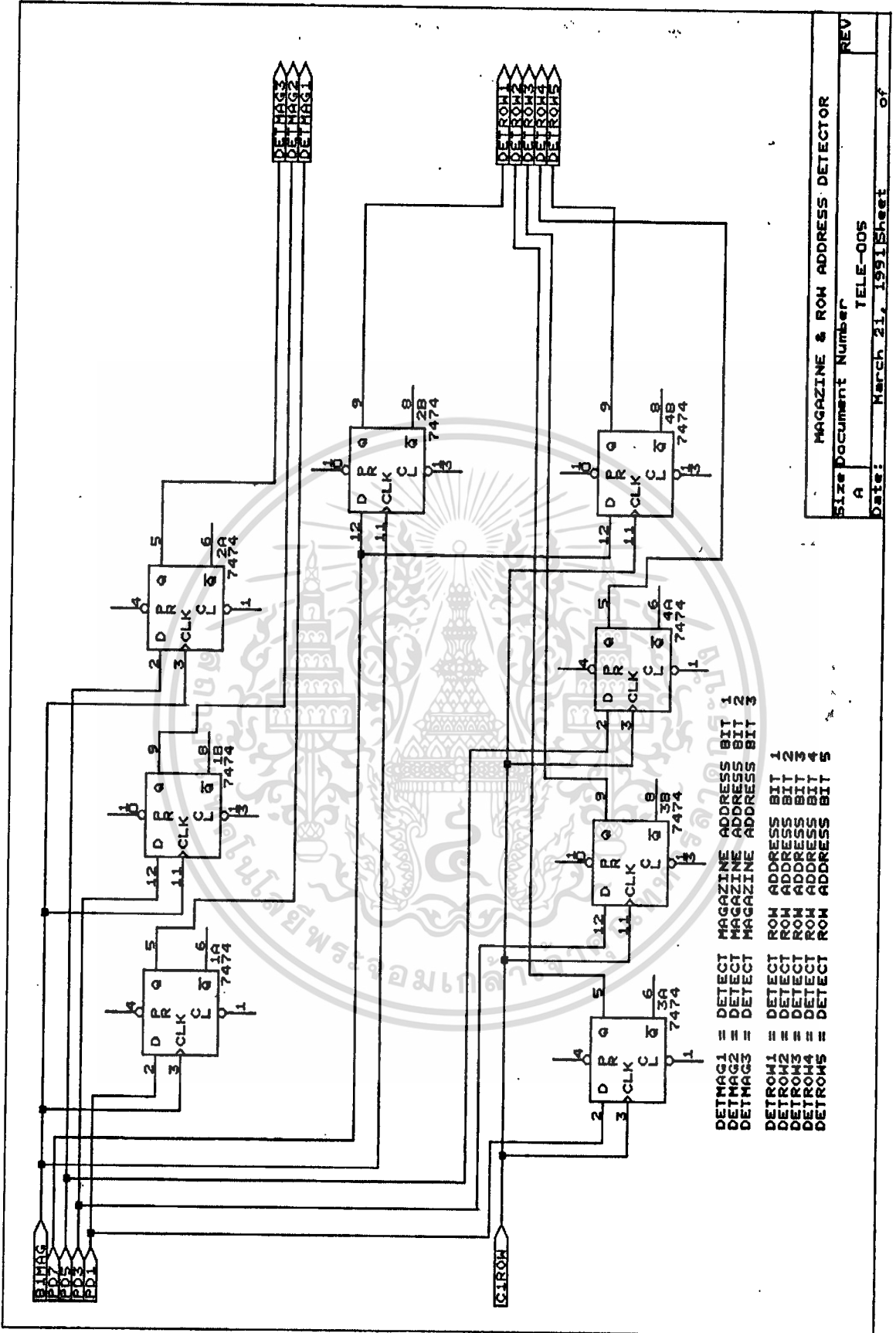
### 3.8 วงจรเลือกหน้าเทเลเท็กซ์

เมื่อเราเลือกหน้าใดหน้าหนึ่งของเทเลเท็กซ์ จะมีสัญญาณส่งมาจากคอมพิวเตอรืดัง นี้คือ

แม็กกาซีน 3 บิต เลขหน้าหลักหน่วย 4 บิต และเลขหน้าหลักสิบอีก 4 บิต

เราจึงนำเอาสัญญาณเหล่านี้มาเปรียบเทียบกับ แม็กกาซีน เลขหน้าหลักหน่วยและ เลขหน้าหลักสิบ จากสัญญาณเทเลเท็กซ์ที่ผ่านเข้ามาทางวงจร ตรวจจับแม็กกาซีนและแอดเดรส แถว และวงจรตรวจจับหน้า ซึ่งมีหลักการง่าย ๆ ดังวงจรข้างล่างนี้

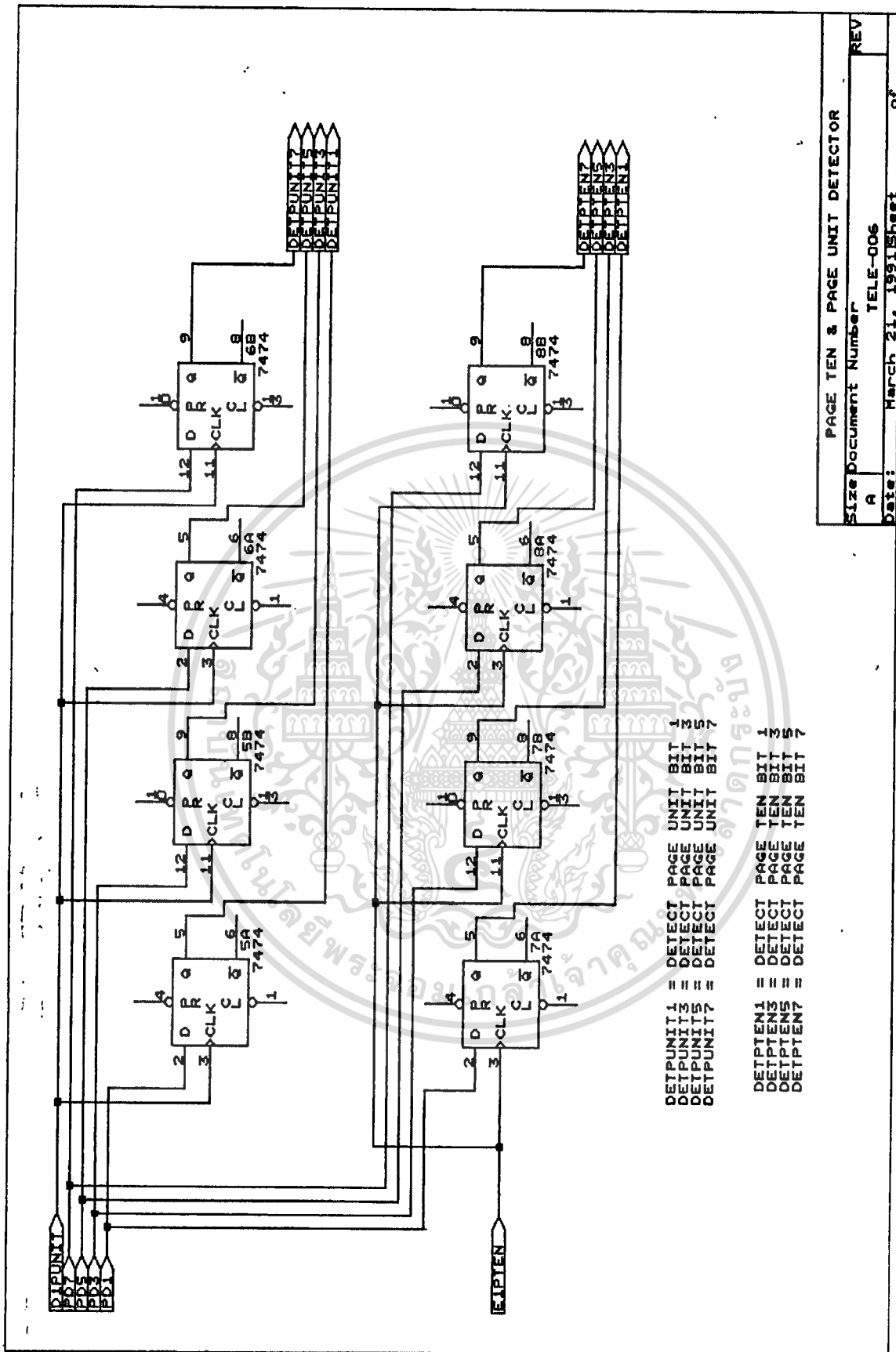




MAGAZINE & ROM ADDRESS DETECTOR	
Size Document Number	REV
A	A
Date: March 21, 1991	Sheet of

รูปที่ 3.7.3 แสดงวงจรตรวจจับแม็กกาซีน และแอดเดรสแถว

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



PAGE TEN & PAGE UNIT DETECTOR	
Size Document Number	REV
A	A
Date: March 21, 1991	Sheet
	of

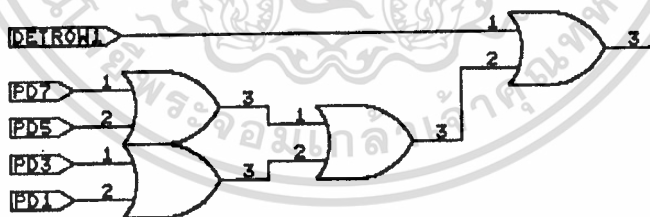
รูปที่ 3.7.4 แสดงวงจรตรวจจับเลขหน้าหลักหน่วย และเลขหน้าหลักสิบ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

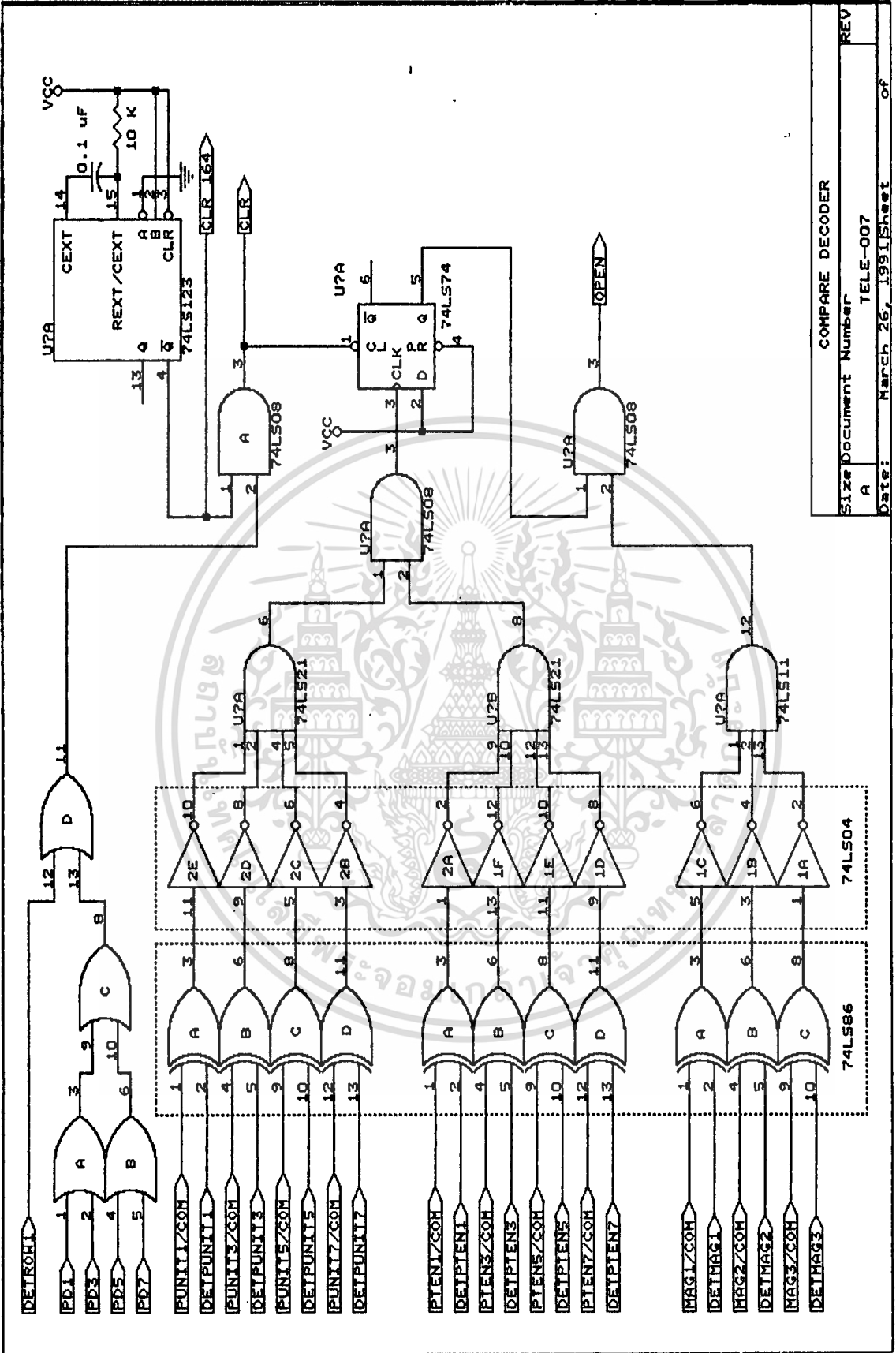
คือถ้าค่าของสัญญาณบอกหน้า เทเลเท็กซ์ตรงกับสัญญาณหน้าจากคอมมิวเตอรฺจะทำให้  
เอาท์พุทมีค่าเป็น "1"

จากหลักการเดียวกันนี้จึงสร้างวงจรการเลือกหน้าของเทเลเท็กซ์ได้ดังรูปที่ 3.8.1  
ในส่วนของเลขหน้าหลักหน่วยเมื่อสัญญาณจากเทเลเท็กซ์และสัญญาณหน้าเช่นเดียวกัน  
กับ ส่วนของเลขหน้าในหลักสิบ เมื่อสัญญาณทั้ง 2 ตรงกัน เอาท์พุทของแอนด์เกจ B ก็จะมี  
สถานะเป็น "1" จากนั้นนำเอาท์พุททั้งสองมาผ่านแอนด์เกจอีกครั้งเพื่อให้ได้เอาท์พุทเป็น "1"  
เมื่อสัญญาณเลขหน้าหลักหน่วยและหลักสิบของสัญญาณเทเลเท็กซ์ตรงกับสัญญาณหน้าจากคอมมิวเตอรฺ  
ขณะเดียวกันแล้ว เอาท์พุทนี้ไปเข้าขาคาส์ล็อกของดี-ฟลิปฟล็อป เพื่อทำการเก็บค่า "1" ไว้  
สัญญาณแม็กกาซีนจากเทเลเท็กซ์ที่ตรงกับสัญญาณแม็กกาซีนจากคอมมิวเตอรฺแล้วก็จะได้เอาท์พุทเป็น  
"1" เรานำสัญญาณนี้ไปแอนด์กับสัญญาณที่ออกมาจากดี-ฟลิปฟล็อป (Q) เพื่อนำเอาท์พุทนี้ไปเป็นตัว  
เปิดเกจในส่วนของวงจรอนุกรมไปเป็นวงจรขนานเพื่อให้วงจรเริ่มทำงานเมื่อเราสามารถตรวจ  
จับแม็กกาซีนและ เลขหน้าได้ตรงกับที่เราต้องการแล้ว เท่านั้น

แต่ในขณะที่เราต้องการ เปลี่ยนหน้าที่จะดู เราจะต้องเอาสัญญาณมาเคลียร์ดี-ฟลิป  
ฟล็อป ก่อนเพื่อให้เอาท์พุทเป็น "0" ซึ่งสัญญาณนี้ได้มาจากการตรวจจับสัญญาณแถวที่ 0 เนื่อง  
จากสัญญาณนี้จะมาก่อนสัญญาณบอกเลขหน้าซึ่งลักษณะวงจร เป็นดังนี้



แต่เพื่อให้สถานะของฟลิปฟล็อปเป็น "0" ขณะเปิดเครื่องเราจึงนำเอาโมโนสเต-  
เบิล มาเป็นตัวทริกขาเคลียร์ของฟลิปฟล็อป ซึ่งมีวงจรรวมทั้งหมดเป็นดังรูปที่ 3.8.1 ในหน้า  
ที่ 3-16



รูปที่ 3.8.1 แสดงวงจรรวมทั้งหมดในส่วนของการเลือกหน้าของเทเลเท็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่าการณีโดยทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.9 การแสดงผลเทเลเท็กซ์

สำหรับโครงการตัวถอดรหัสสัญญาณเทเลเท็กซ์ (Teletext Decoder) นี้ออกแบบให้ตัวถอดรหัส (Decoder) ติดต่อกับผู้ใช้ผ่านทางคอมพิวเตอร์ (Personal Computer : PC User Interface) โดยกำหนดให้คอมพิวเตอร์ ทำงานตามขั้นตอนดังนี้

1. ให้ผู้ใช้กำหนดหน้าข้อมูลเทเลเท็กซ์ที่ต้องการ
2. ส่งหน้าข้อมูลเทเลเท็กซ์ที่เลือกไปยังฮาร์ดแวร์ (Hardware)
3. ตรวจสอบฮาร์ดแวร์ว่าเก็บข้อมูลเสร็จแล้วหรือยัง
4. เมื่อฮาร์ดแวร์เก็บข้อมูลลงหน่วยความจำเสร็จ ให้คอมพิวเตอร์เป็นผู้ควบคุมหน่วยความจำที่เก็บข้อมูลเทเลเท็กซ์ แทนฮาร์ดแวร์แล้วทำการอ่านข้อมูลมาเก็บไว้ที่หน่วยความจำภายในคอมพิวเตอร์
5. ทำการถอดรหัสข้อมูลและแสดงผลออกมามีเตอร์เพื่อให้คอมพิวเตอร์ ทำงานตามขั้นตอนที่ต้องการได้จำเป็นต้องมี
  - 5.1 ฮาร์ดแวร์ส่วนที่ติดต่อ (Interfac) กับวงจรภายนอกคอมพิวเตอร์
  - 5.2 ซอร์ฟแวร์ (Software) เป็นโปรแกรมควบคุมการทำงานของคอมพิวเตอร์

#### 3.9.1 ฮาร์ดแวร์

การรับ-ส่งข้อมูลระหว่างคอมพิวเตอร์ กับอุปกรณ์ภายนอก กระทำผ่านไอโอพอร์ท (IO PORT) ซึ่งสำหรับไอบีเอ็มพีซี (IBM PC) หรือคอมแพททิเบิล (COMPATIBLE) จะมีไอโอพอร์ทให้ใช้ถึง  $2^{10} = 1024$  พอร์ต บางพอร์ต ถูกใช้ในการติดต่อกับอุปกรณ์มาตรฐานต่าง ๆ เช่น จอมอนิเตอร์ (monitor) คีย์บอร์ด (keyboard) พรินเตอร์ (printer) ฯลฯ เราเลือกพอร์ตที่ว่างมาใช้งาน พอร์ตที่เลือก คือ พอร์ต 2B0-2B3

ขั้นตอนการทำงานของคอมพิวเตอร์ในการติดต่อไอโอพอร์ท เป็นดังนี้

การส่งข้อมูลไปยังไอโอพอร์ท (ไอโอไรท์ : IO WRITE)

1. ส่งแอดเดรสของพอร์ตที่ต้องการติดต่อ
2. ส่งสัญญาณไอโอดับเบิลยู (IOW)
3. ส่งข้อมูลไปยังบัสข้อมูล

การรับข้อมูลจากไอโอพอร์ท (ไอโอรีด : IO READ)

1. ส่งแอดเดรสของพอร์ตที่ต้องการติดต่อ
2. ส่งสัญญาณไอโออาร์ (IOR)

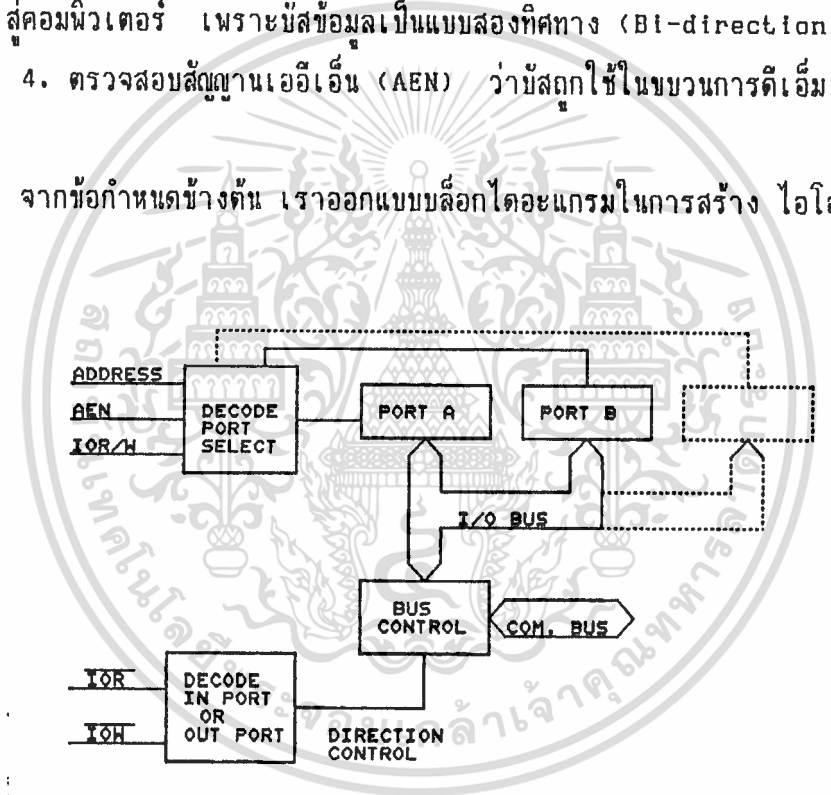
เอกสารนี้เป็นเอกสารที่อ่านข้อมูลจากบัสข้อมูลเพื่อการศึกษาเท่านั้น ไม่นิยนาให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

### 3.9.2 การออกแบบวงจรอินเทอร์เฟส

การนำอุปกรณ์ใด ๆ มาอินเทอร์เฟสกับไอโอพอร์ต มีข้อกำหนดดังนี้

1. ต้องตรวจสอบว่าแอดเดรส ที่ตรวจสอบเป็นแอดเดรสพอร์ท (Port address) หรือแอดเดรสหน่วยความจำ (memory address) เพราะมีแอดเดรสของวงจรต่าง ๆ ทั้งบนบอร์ดหลัก (main board) และอุปกรณ์ภายนอกถูกใช้ร่วมกัน
2. ต้องมีการเชื่อมต่อกับบัสข้อมูลเฉพาะเมื่อพอร์ทนั้น ๆ ถูกติดต่อกับเท่านั้น เพราะบัสข้อมูลก็ใช้ร่วมกัน
3. ต้องแยกทิศทางของสัญญาณข้อมูลว่าเป็นจากคอมพิวเตอร์ สู่ไอโอพอร์ต หรือ ไอโอพอร์ต สู่คอมพิวเตอร์ เพราะบัสข้อมูลเป็นแบบสองทิศทาง (Bi-direction)
4. ตรวจสอบสัญญาณเออีเอ็น (AEN) ว่าบัสถูกใช้ในขบวนการดีเอ็มเอ (DMA) หรือไม่

จากข้อกำหนดข้างต้น เราออกแบบบล็อกไดอะแกรมในการสร้าง ไอโอพอร์ตได้ดังนี้



รูปที่ 3.9.2.1 แสดงบล็อกไดอะแกรมของวงจรอินเทอร์เฟส

#### หน้าที่ของวงจรอินเทอร์เฟสที่ต้องการ

1. ส่งข้อมูลไปยังวงจรภายนอก
  - 1.1 ส่งหมายเลขแม็กกาซีน (Magazine number) 3 บิต
  - 1.2 ส่งเลขหน้า (Page number) 8 บิต
  - 1.3 ส่งสัญญาณควบคุมหน่วยความจำ 1 บิต

2. รับข้อมูลจากวงจรภายนอก	10	บิต
2.1 รับสัญญาณข้อมูลพร้อม	2	บิต
2.2 อ่านข้อมูลจากหน่วยความจำภายนอก	8	บิต

ส่วนของวงจรอินเทอร์เฟสออกแบบไว้ดังรูปที่ 3.9.2.2 ในหน้าที่ 3-20

การรับส่งข้อมูลเราใช้พอร์ท 4 พอร์ท คือ พอร์ท 2B0-2B3 แล้วแยกเป็นพอร์ทส่งและพอร์ทรับ ถ้าเปรียบเทียบแล้วเราจะได้ประตูทางเข้า 2 ทาง (ส่งข้อมูลเข้าคอมพิวเตอร์) คือ ไอซีเบอร์ 74LS245(2) และ (3) และได้ประตูทางออก 4 ทางคือ ไอซีเบอร์ 74LS374 (1)-(4) โดยมีเอาท์พุท  $Y_n$  จากไอซีเบอร์ 74LS138 เป็นสัญญาณประตูแต่ละบาน ส่วน 74LS245(1) จะเป็นตัวกำหนดทิศทางข้อมูล

เนื่องจากการติดต่อไอโอพอร์ท 2B0-2B4 จะใช้แอดเดรส  $A_0-A_9$  พอร์ทที่นำมาใช้ คือพอร์ท 2B0-2B3 ซึ่งเขียนเป็นเลขฐานสองได้ดังนี้

$A_9$	$A_8$	$A_7$	$A_6$	$A_5$	$A_4$	$A_3$	$A_2$	$A_1$	$A_0$
1	0	1	0	1	1	0	0	0	0
1	0	1	0	1	1	0	0	0	1
1	0	1	0	1	1	0	0	1	0

เรานำ  $A_0-A_9$  มาถอดรหัสด้วย 7404 และ 7430 ถ้า  $A_9-A_7$  มีแอดเดรสตรงกับพอร์ทที่จะได้อาท์พุทเป็น "1"

เพื่อแยกไอโอรีด (IO READ) และไรท์ (write) เราจึงใช้ 74138 ซึ่งเป็นไอซีดีโค็ด 2 ตัว ตัวหนึ่งถอดรหัสที่อ่าน อีกตัวหนึ่งถอดรหัสพอร์ทที่เขียน

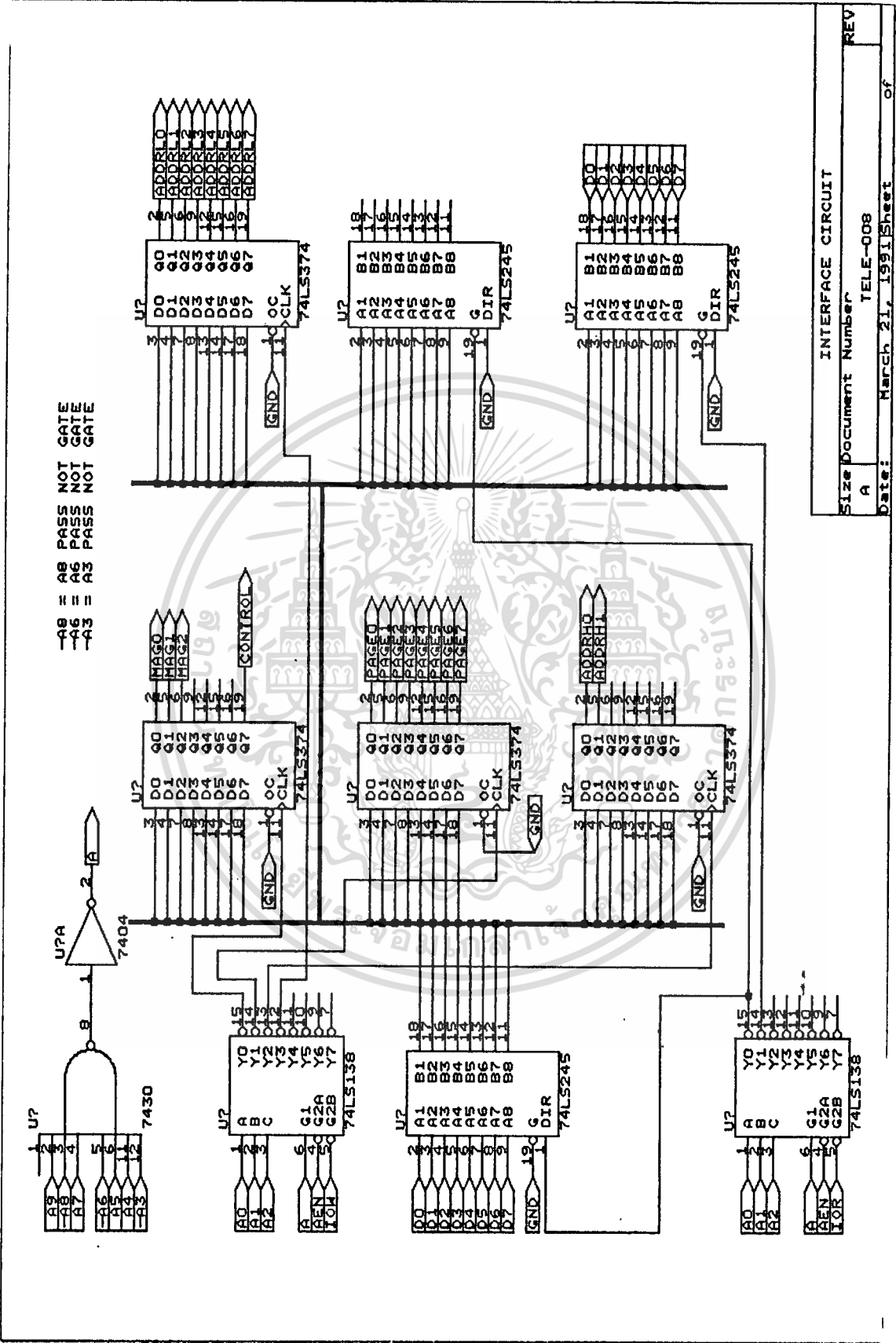
ซึ่ง 74138 จะให้อาท์พุท เมื่อ

- แอดเดรสของพอร์ทตรงกับ 2B0-2B1
- เออีเอ็น (AEN) สู่ภาวะแอ็คทีฟ (ACTIVE) คือเป็น "0"
- ไอโอดับเบิ้ลยู (IOW) สำหรับ 74138(1) หรือ
- ไอโออาร์ (IOR) สำหรับ 74138(2) แอ็คทีฟ (ACTIVE)

การทำงานของวงจรเป็นดังนี้

1. เมื่อคอมพิวเตอร์ส่งข้อมูลไปยังพอร์ท 2B0  $y_0$  ของ 74138(1) แอ็คทีฟจะมี

พัลส์ (Pulse) ลงไปยังขาคล็อก (Clk) ของ 74374(1) ทำให้ข้อมูลบนบัสไปปรากฏยังเอาท์พุท  $Q_0-Q_7$  ขณะที่ตัวอื่นจะไม่มีคล็อกไปทริก เช่นเดียวกันในกรณีของการส่งข้อมูลไปยังไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมีเหตุดบแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



REV	
Size	Document Number
A	TELE-008
Date:	March 21, 1991
	Sheet of

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษานี้เท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 รูปที่ 3.9.2.2 แสดงวงจรอินเทอร์เฟซที่ออกแบบไว้  
 ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมีเหตุดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

พอร์ต 2B1, 2B2 และ 2B3 ในขณะที่ส่งข้อมูล (2) และ (3) จะอยู่ในสถานะไตรสเตรท (Tristate) ไม่มีสัญญาณจากภายนอกเข้าสู่ค่าตัวบัส

2. เมื่อคอมพิวเตอร์อ่านข้อมูลจากพอร์ต 2B0 ของ 74138(2) เท่านั้นที่แอสซิมบลีทำให้ 74245(2) ยอมให้ข้อมูลผ่านลงบัสเช่นเดียวกันในกรณีอ่านพอร์ต 2B1 และ 74245 (1) จะเปลี่ยนทิศทางของบัสเป็นจากพอร์ต สู่คอมพิวเตอร์ในการอ่านพอร์ตใด ๆ

### 3.9.3 ซอร์ฟแวร์ (Software)

โปรแกรมสำหรับควบคุมการทำงานของคอมพิวเตอร์ เขียนขึ้นด้วยภาษาซี (C) เพราะภาษาซีเป็นภาษาสูงที่เขียนโปรแกรมติดต่อกับฮาร์ดแวร์ได้สะดวก

สำหรับขั้นตอนการทำงานของโปรแกรมโดยรวมเขียนไฟล์ชาร์ทได้ตั้งหน้า 3-22 จากขั้นตอนการทำงาน เราแบ่งโปรแกรมเป็นส่วน ๆ ตามหน้าที่ไว้ 3 ส่วน คือ

- 3.9.3.1 ส่วนที่ติดต่อกับผู้ใช้
- 3.9.3.2 ส่วนควบคุมวงจรรีจิสเตอร์เฟส
- 3.9.3.3 ส่วนแสดงผลข้อมูลเทเลเท็กซ์
- 3.9.3.1 ส่วนที่ติดต่อกับผู้ใช้

มีหน้าที่หลัก คือ รับคำสั่งเลือกหน้าข้อมูลเทเลเท็กซ์ที่ผู้ใช้ต้องการ เพื่อให้โปรแกรมส่วนอื่นจัดการต่อไป

#### 3.9.3.2 ส่วนควบคุมวงจรรีจิสเตอร์เฟส

โปรแกรมในส่วนนี้จะคอยควบคุมการรับ-ส่งข้อมูล ระหว่างคอมพิวเตอร์ กับวงจรรายนอก โดยมีขั้นตอนการทำงานตามลำดับดังนี้

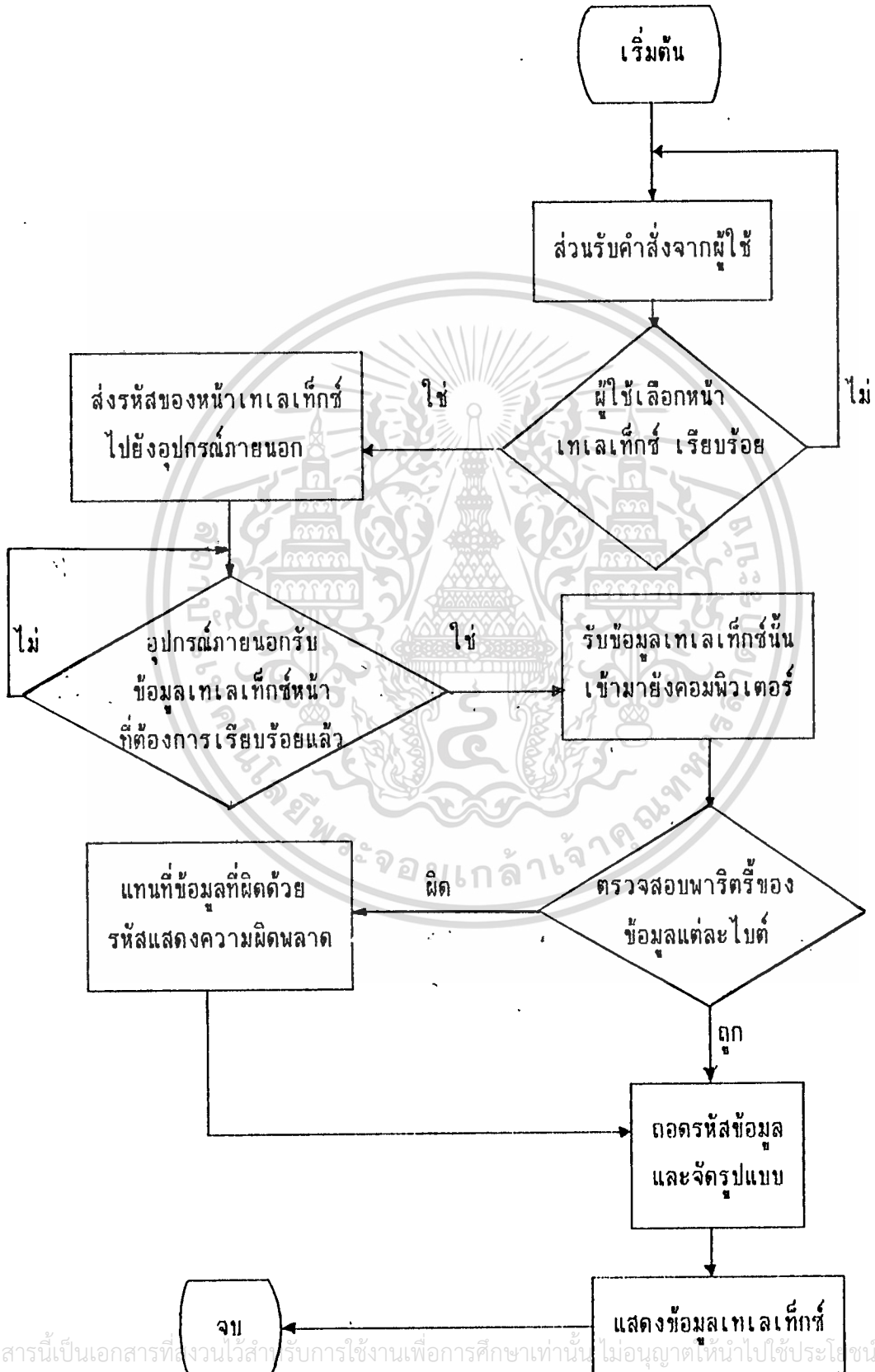
ก. ส่งรหัสหน้าข้อมูลที่เลือกไปที่วงจรรายนอก โดย

- ส่งแอดเดรสของแม็กกาซีนไปที่พอร์ต 2b0
- ส่งแอดเดรสของหน้าไปที่พอร์ต 2b1

ข. ตรวจสอบว่าวงจรรายนอกรับข้อมูลเทเลเท็กซ์เรียบร้อยแล้วหรือยัง โดยกำหนดให้วงจรรายนอกส่งรหัสมายังพอร์ต 2b0 เมื่อรับข้อมูลเรียบร้อยแล้ว ซึ่งโปรแกรมจะคอยตรวจสอบที่พอร์ต 2b0 ว่ามีรหัส FCH เข้ามาหรือไม่ (รหัส FCH = 11111100 คือการทริกขา B<sub>0</sub>, B<sub>1</sub> ของ 74245(1) ให้เป็น 0 นั้นเอง)

ค. เมื่อพบว่าเก็บข้อมูลเรียบร้อยแล้ว จะส่งรหัสไปที่พอร์ต 2b0 เพื่อเป็นสัญญาณไปบอกวงจรรายนอกให้ตัดการควบคุมหน่วยความจำ (RAM) แล้ววงจรรีจิสเตอร์เฟสจะไปควบคุมหน่วยความจำแทน (รหัสที่ส่งคือ 80H = 10000000 คือการให้ Q<sub>7</sub> ของ 74974 เป็น "1" ซึ่งวงจรรายนอกจะตรวจสอบที่ขานี้)

ขั้นตอนการทำงานของโปรแกรม



ง. อ่านข้อมูลจากหน่วยความจำภายใน 1004 ไบต์ โดย

- ส่งแอดเดรสของหน่วยความจำภายนอก ( $A_0-A_{12}$ ) ผ่านพอร์ท 2b2 และ 2b3

- อ่านข้อมูลที่แอดเดรสนั้นของหน่วยความจำภายนอกโดยผ่านพอร์ท 2b1

### 3.9.3.3 ส่วนแสดงผลข้อมูลเทเลเท็กซ์

เนื่องจากข้อมูลเทเลเท็กซ์แต่ละไบต์ (8 บิต) จะเป็นรหัสข้อมูลจริง ๆ 7 บิต (อีก 1 บิตที่เหลือเป็นพาริตีบิต) ซึ่งแทนข้อมูลได้  $2^7 = 128$  แบบ แต่ความต้องการนั้นมีทั้งรหัสตัวอักษร และรหัสรูปภาพ ซึ่งเกินกว่า 128 แบบ

เทเลเท็กซ์จึงกำหนดรหัสควบคุมขึ้นมาเพื่อแยกการแสดงผลให้ได้มากกว่า 128 แบบ ซึ่งรหัสควบคุมจะแบ่งเป็น 2 พวกคือ รหัสควบคุมรูปภาพ (10H-1FH) และรหัสควบคุมอักษร (00H-0FH) (ดูตารางที่ 2.5.6.1 และ 2.5.4.1 ประกอบด้วย) โดยมีวิธีแยกแยะข้อมูลดังนี้ คือ

แต่ละแถวข้อมูลจะถือว่ารหัสทุกตัวเป็นรหัสตัวอักษร (ตามตารางที่ 2.5.4.1) จนกว่าจะเจอรหัสควบคุมรูปภาพ (11H-17H) จึงจะถือว่าข้อมูลหลังจากนั้นเป็นรหัสรูปภาพ (ตามตารางที่ 2.5.6.1) จนกระทั่งหมดแถวนั้น หรือพบรหัสควบคุมอักษร (11H-17H) จึงกลับมาแสดงอักษรอีกครั้ง

ข้อมูลในแต่ละแถวจะเรียงกันตามลำดับซึ่งตรงกับตำแหน่งที่มันจะแสดงบนจอ ดังนั้น ข้อมูลตัวใดซึ่งเป็นรหัสควบคุมตำแหน่งจะเป็นช่องว่างไม่มีการแสดงผล

จากรูปแบบของข้อมูลเราจึงกำหนดให้โปรแกรมแสดงผล (Program Display) มีขั้นตอนการทำงานดังนี้

1. นำข้อมูลที่อ่านจากหน่วยความจำ (RAM) ภายนอกมาเก็บลงตามตำแหน่งในอาร์เรย์ (array) ของข้อมูลขนาด  $24 \times 40$  (24 แถว แถวละ 40 ตัวอักษร)

2. ตรวจสอบพาริตีของข้อมูลทุกตัว แล้วตัดพาริตีทิ้งโดยทำให้บิตที่ 8 ของข้อมูลเป็น 0 ข้อมูลที่พบว่า เป็นพาริตีผิดจะถูกแทนด้วยรหัสแสดงความผิดพลาด ซึ่งกำหนดให้แสดงบนจอเป็น 'e'

3. แสดงผลข้อมูลในโหมดกราฟิกของคอมพิวเตอร์ เพื่อให้แสดงได้ทั้งอักษร และรูปภาพ โดย

3.1 อ่านข้อมูลเข้ามาทีละแถว

3.2 อ่านข้อมูลในแถวนั้นมาทีละตัว

3.3 ถ้าเป็นรหัสตัวอักษร (20H-7FH) ให้นำมาแสดงตัวอักษรบนจอภาพที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับใช้ภายในเท่านั้น ห้ามเผยแพร่โดยไม่ได้รับอนุญาต  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตำแหน่งแถวและหลักนั้น ๆ

3.4 ถ้าเป็นรหัสควบคุมให้แสดงตัวว่าง (space) ที่ตำแหน่งนั้นและถ้าเป็นรหัสควบคุมรูปภาพ ให้ถือว่าข้อมูลที่ตามมาเป็นรหัสของรูปภาพแล้วแสดงผลบนจอ จนกว่าจะขึ้นบรรทัดใหม่ หรือพบรหัสควบคุมอักษร จึงเปลี่ยนไปแสดงผลแบบอักษร

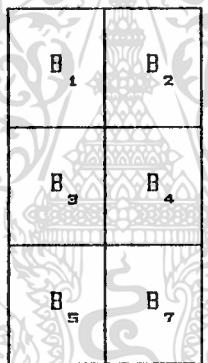
เขียนเป็นโฟลว์ชาร์ท (Flow chart) ได้ดังรูปหน้าที่ 3-25

อนึ่งเนื่องจากโปรแกรมใช้กับคอมพิวเตอร์ที่ใช้จอโมโนโครม เพื่อความสะดวกและเหมาะสมจึงสนใจรหัสควบคุมเฉพาะว่าเป็นอักษรหรือรูปภาพเท่านั้น ไม่สนใจการกำหนดสีของตัวควบคุม ซึ่งข้อมูลเทเลเท็กซ์สามารถแสดงได้ถึง 7 สี ทั้งสีพื้นและสีของตัวอักษรและรูปภาพ

การแสดงผลรูปภาพ

รหัสรูปภาพของเทเลเท็กซ์แต่ละตัวจะมีลักษณะสัมพันธ์กับรหัสของมันแบบบิตแมป

(bit map) ดังนี้



รหัสรูปภาพแต่ละตัวจะประกอบด้วยบล็อก 6 บล็อก แต่ละบล็อกขึ้นอยู่กับแต่ละบิตของรหัส ถ้าบิตนั้น ๆ เป็น 1 บล็อกนั้นจะแสดงสี ถ้าเป็น 0 จะไม่แสดง เช่น

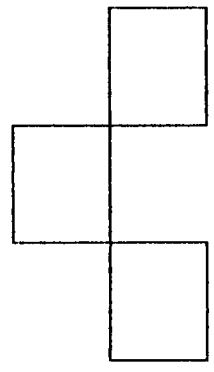
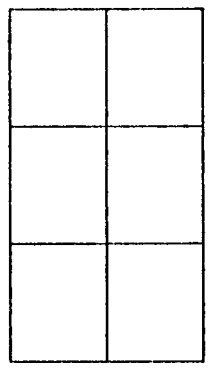
รหัส      1   1   0   0   1   1   0

$B_7$   $B_6$   $B_5$  -  $B_4$   $B_3$   $B_2$   $B_1$

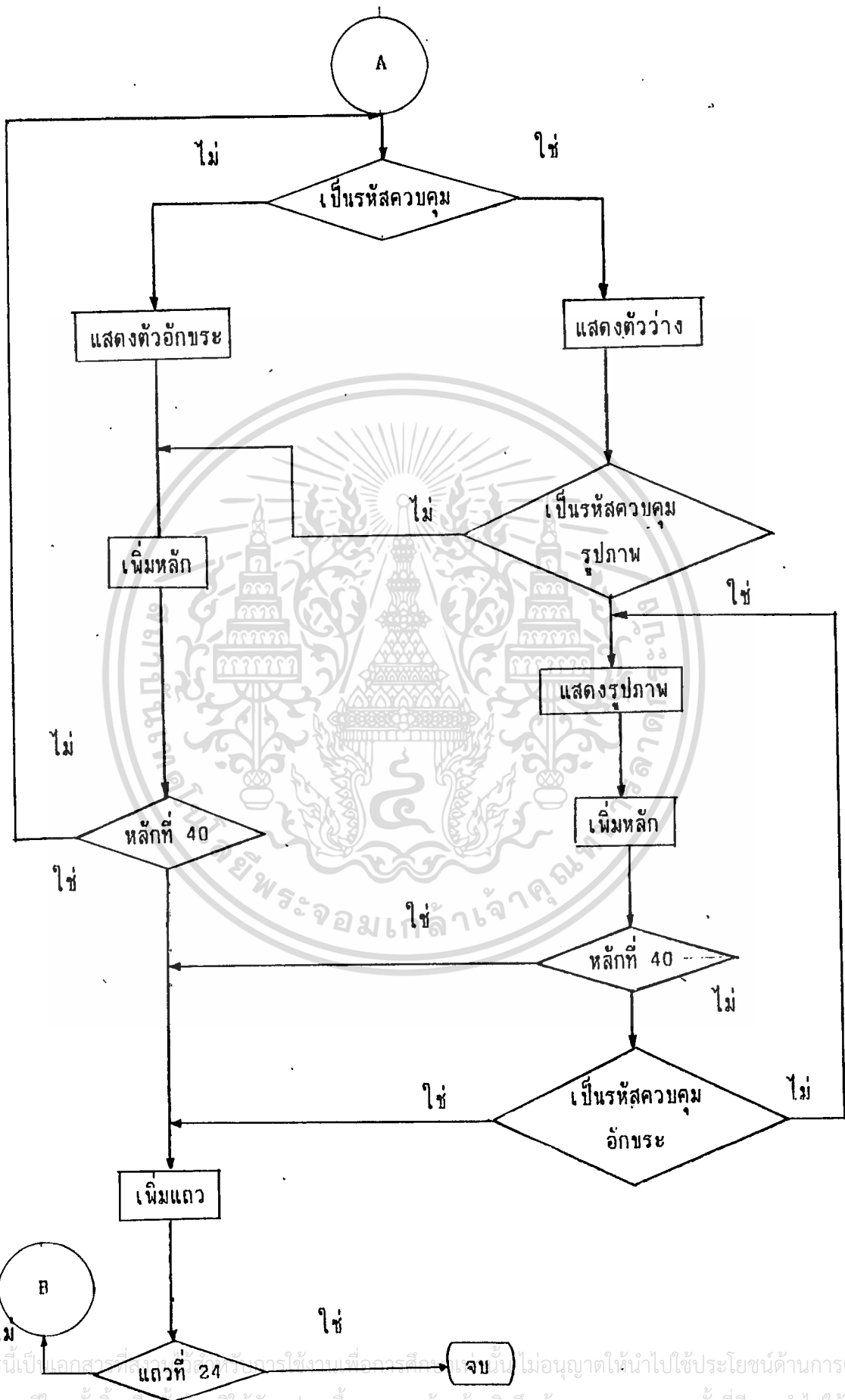
ทำบิตแมปจะได้

0	1
1	0
0	1

ซึ่งแสดงผลบนจอเป็น







```

/*      TELETEXT DECODER PROGRAM      */

#ifdef __TINY__
#error BGIDEMO will not run in the tiny model.
#endif

#include <stdio.h>
#include <conio.h>
#include <bios.h>
#include <alloc.h>
#include <dos.h>
#include <graphics.h>

#define port0 0x2b0    /*out mag*/
#define port1 0x2b1    /*out page*/
#define port2 0x2b2    /*out address h*/
#define port3 0x2b3    /*out address l*/
#define ERRC 0x40
#define control1 0x80
#define control2 0xfc
#define texnumber 1004
#define WID 8
#define HIG 8
#define X0 0
#define Y0 0
#define X1 4
#define X2 8
#define Y1 4
#define Y2 8
#define Y3 12

int    GraphDriver;    /* The Graphics device driver      */
int    GraphMode; /* The Graphics mode value*/
double AspectRatio; /* Aspect ratio of a pixel on the screen*/
int    MaxX, MaxY; /* The maximum resolution of the screen */

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

int    MaxColors; /* The maximum # of colors available*/
int    ErrorCode; /* Reports any graphics errors*/

char *texbuffer;
char *scrbuff;
char texrow[25][41];
static unsigned long vidiouseg=0xb0000001;
int item0,item1=1;
char *vdo_buffer,*vdo_buffer2;
char *helpinfo[]={
    "          HELP INFORMATION          ",
    "  To select any command,use arrow and enter",
    "  keys or press first letter of comand key.",
    "  To get TELETEXT information,select",
    "  select PAGE SELECT and input page number",
    "  that you want.",
    "  Press ESC or select QUIT to end program."};
void tex_display(void);
int parity_chck(char tex);
void tex_in_row(void);
int in_tex(char mag,char pageten,char pageunit);
void quit_func(void);
void option_func(void);
void help_func(void);
int page_func(void);
void func_(int func);
void clr_hili(int item);
void set_hili(int item);
int select_func(void);
void wr_func(void);
void title_(void);
void save_box(int row1,int col1,int row2,int col2,char *vdo_buf);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

void restore_box(int row1,int col1,int row2,int col2,char *vdo_buf);
void vdo_wr(int row,int col,int att,char *string);
void wr_box(int row1,int col1,int row2,int col2,char attr);
void vdo_att(int row1,int col1,int col2,int att);
int get_key(void);
void Initialize(void);
void gputbox(char font,int xpos,int ypos)
main()
{
    clrscr();
    title_();
    getch();
    wr_func();
    set_hili(1);
    while(1==1)
    {
        func_(select_func());
    }
}
/*-----*/
int get_key(void)
{
    union REGS r;
    r.h.ah=0;
    return int86(0x16,&r,&r);
}
/*-----*/
void vdo_att(int row1,int col1,int col2,int att)
{
    register int i;
    char far *offset1;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

offset1=(char far *) (vidioseg +col1+col1+1+row1*160);
for(i=col1;i<=col2;i++)
{
*offset1=(char)att;
offset1=offset1+2;
}
}
/*-----*/

void wr_box(int row1,int col1,int row2,int col2,char attr)
{
int i,j;
vdo_wr(row1,col1,attr,"");
vdo_wr(row1,col2,attr,"");
vdo_wr(row2,col1,attr,"");
vdo_wr(row2,col2,attr,"");
for(i=col1+1;i<col2;i++)
{
vdo_wr(row1,i,attr,"");
vdo_wr(row2,i,attr,"");
}
for(i=row1+1;i<row2;i++)
{
vdo_wr(i,col1,attr,"");
vdo_wr(i,col2,attr,"");
for(j=col1+1;j<col2;j++)
vdo_wr(i,j,7," ");
}
}
/*-----*/

void vdo_wr(int row,int col,int att,char *string)

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
char far *offset;
offset=(char far *) ( vidioseg +col+col+row*160);
while (*string)
{ *offset++ =*string++;
*offset++ =(char)att; }
}
/*-----*/
void restore_box(int row1,int col1,int row2,int col2,char *vdo_buf)
{
register int i,j;
char far *offset;
for(j=row1;j<=row2;j++)
{
offset=(char far *) ( vidioseg+col1*2+j*160);
for(i=col1;i<=col2;i++)
{
*offset++=*vdo_buf++;
*offset++=*vdo_buf++;
}
}
}
/*-----*/
void save_box(int row1,int col1,int row2,int col2,char *vdo_buf)
{
register int i,j;
char far *offset;
for(j=row1;j<=row2;j++)
{
offset=(char far *) ( vidioseg +col1+col1+j*160);
for(i=col1;i<=col2;i++)

```



```

delay(500);
/*wr_fn_selectpage()*/
wr_box(7,65,9,77,0xf);
vdo_wr(8,66,7,"SELECT PAGE");
delay(500);
/*wr_fn_option()*/
wr_box(12,65,14,77,0xf);
vdo_wr(13,66,7,"OPTION");
delay(500);
/*wr_fn_quit()*/
wr_box(17,65,19,77,0xf);
vdo_wr(18,66,7,"QUIT");
}
/*-----*/
int select_func(void)
{
    union scan_key{int c;
                char ch[2];}key;
    loop2:
        key.c=get_key();
        if(key.ch[0]!=0)
        {
            switch(key.ch[1])
            {
                case 80:item0=item1;
            }
        }
    if(item1<4)
        {item1=item1+1;}
        else
        {item1=1;}

        clr_hili(item0);
        set_hili(item1);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        break;
    case 72:item0=item1;
if(item1>1)
    {item1=item1-1;}
    else
    {item1=4;}

    clr_hili(item0);
    set_hili(item1);
break;

    default;;
    }
goto loop2;
}
else
{
    item0=item1;
    switch(key.ch[0])
    {
case 'h':clr_hili(item0);set_hili(1);item1=1;return(1);/*help_func()
        ;break;*/
case 's':clr_hili(item0);set_hili(2);item1=2;return(2);/*page_func()
        ;break;*/
case 'o':clr_hili(item0);set_hili(3);item1=3;return(3);/*option_func
        ();break;*/
case 'q':clr_hili(item0);set_hili(4);item1=4;return(4);/*quit_func()
        ;break;*/
case '\r':return(item1);/*func_(item1);break;*/
case 0x1b:exit(0);
default;;
    }
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
/*-----*/
void set_hili(int item)
{
    switch(item)
    {
        case 1:vdo_att(3,66,76,0x70);break;
        case 2:vdo_att(8,66,76,0x70);break;
        case 3:vdo_att(13,66,76,0x70);break;
        case 4:vdo_att(18,66,76,0x70);break;
    default:;
    }
}
/*-----*/
void clr_hili(int item)
{
    switch(item)
    {
        case 1:vdo_att(3,66,76,7);break;
        case 2:vdo_att(8,66,76,7);break;
        case 3:vdo_att(13,66,76,7);break;
        case 4:vdo_att(18,66,76,7);break;
    default:;
    }
}
/*-----*/
void func_(int func)
{
    switch(func)
    {
        case 1:help_func();break;

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

case 2:if(page_func()==0)
    {
        tex_in_row();
        tex_display();
        title_();
        wr_func();
    }
    break;
case 3:option_func();break;
case 4:quit_func();break;
default;;
}
}
/*-----*/
int page_func(void)
{
    char mag,page,pageten,pageunit;
    int inrdy;
    vdo_buffer=(char *)calloc((6*21*2),sizeof(char));
    save_box(18,40,23,60,vdo_buffer);
    wr_box(18,40,23,60,0xf);
    vdo_wr(19,41,7,"Enter Page Number");
    vdo_wr(20,41,7,"    100-899");
    vdo_wr(21,41,7,"Your select is");
    /*-----get page from user-----*/
    loop:
    vdo_att(21,56,58,0x70);
    gotoxy(57,22);
    mag=getche();
    if(mag=='\r')
    {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

    restore_box(18,40,23,60,vdo_buffer);
    free(vdo_buffer);
    return(1);
}

pageten=getche();
pageunit=getche();
vdo_att(21,56,58,0x70);
/*-----check over page number-----*/
if(mag<0x31||mag>0x38||pageten<0x30||pageten>0x39||pageunit<0x30||
    pageunit>0x39)
{
    vdo_buffer2=(char *)calloc((4*30*2),sizeof(char));
    save_box(17,35,20,63,vdo_buffer2);
    wr_box(17,35,20,63,0x78);
    vdo_wr(18,36,0x70,"Page Number only 100-899  ");
    vdo_wr(19,36,0x70,"Press anykey to select new ");
    getch();
    restore_box(17,35,20,63,vdo_buffer2);
    free(vdo_buffer2);
    gotoxy(57,22);
    puts("  ");
    goto loop;
}

/*-----get data from decoder-----*/
vdo_wr(22,41,0xf,"ENTER to continue");
if(getch()=='\r')
{
    vdo_buffer2=(char *)calloc((4*24*2),sizeof(char));
    save_box(12,35,15,58,vdo_buffer2);
    wr_box(12,35,15,58,0xf);
    vdo_wr(13,36,0x87,"  please wait...  ");

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

inrdy=in_tex(mag,pageten,pageunit);
/*----check get data ready---*/
if(inrdy==0)
{
vdo_wr(13,36,0x87,      " Get data complete ");
vdo_wr(14,36,0x70,      "  press anykey  ");
getch();

    restore_box(12,35,15,58,vdo_buffer2);
    restore_box(18,40,23,60,vdo_buffer);
    free(vdo_buffer2);
    free(vdo_buffer);
return(0);
}
else
{
vdo_wr(13,36,0x87,      " Get data time out ");
vdo_wr(14,36,0x70,      "  press anykey  ");
getch();
    restore_box(12,35,15,58,vdo_buffer2);
    restore_box(18,40,23,60,vdo_buffer);
    free(vdo_buffer2);
    free(vdo_buffer);
return(1);
}
}
else
{
    restore_box(18,40,23,60,vdo_buffer);
    free(vdo_buffer);
return(1);
}
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
/*-----*/
void help_func(void)
{
    vdo_buffer=(char *)calloc((9*48*2),sizeof(char));
    save_box(12,9,20,56,vdo_buffer);
    wr_box(12,9,20,56,7);
    vdo_wr(13,10,1,helpinfo[0]);
    vdo_wr(14,10,7,helpinfo[1]);
    vdo_wr(15,10,7,helpinfo[2]);
    vdo_wr(16,10,7,helpinfo[3]);
    vdo_wr(17,10,7,helpinfo[4]);
    vdo_wr(18,10,7,helpinfo[5]);
    vdo_wr(19,10,7,helpinfo[6]);
    getch();
    restore_box(12,9,20,56,vdo_buffer);
    free(vdo_buffer);
}
/*-----*/
void option_func(void)
{
    vdo_buffer=(char *)calloc((7*12*2),sizeof(char));
    save_box(14,63,18,74,vdo_buffer);
    wr_box(14,63,18,74,0xf);
    vdo_wr(15,64,0x70,"LAST DATA ");
    vdo_wr(16,64,0x70,"SAVE DATA ");
    vdo_wr(17,64,0x70,"LOAD DATA ");
    getch();
    restore_box(14,63,18,74,vdo_buffer);
    free(vdo_buffer);
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

/*-----*/
void quit_func(void)
{
    vdo_buffer=(char *)calloc((6*17*2),sizeof(char));
    save_box(14,50,19,66,vdo_buffer);
    wr_box(14,50,19,66,0xf);
    vdo_wr(16,51,7,"Please confirm");
    vdo_wr(17,51,0xf," Y to Quit  ");
    if(getch()=='y')
        exit();
    restore_box(14,50,19,66,vdo_buffer);
    free(vdo_buffer);
}
/*-----in/out data-----*/
int in_tex(char mag,char pageten,char pageunit)
{
    char page;
    char *texbuff;
    int i,addrh=0,addrl=0,wait;
    mag=mag-0x30;
    pageten=pageten-0x30;
    pageunit=pageunit-0x30;
    page=pageten*0x10;
    page=page|pageunit;
    outportb(port0,mag);
    outportb(port1,page);
    wait=0;
    /*---check decoder ready---*/
    while(inportb(port0)!=control2)
    {
        wait++;
    }
}

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

delay(500);
    if(wait==960)return (1);
}
/*-----get data from RAM-----*/
outportb(port0,0x80|mag);
texbuffer=(char *)malloc(texnumber);
texbuff=texbuffer;
delay(1000);
for(i=0;i<texnumber;i++)
{
    outportb(port2,addrh);
outportb(port3,addrl);
delay(20);
    *texbuff++=inportb(port1);
    if(addrl==0xff)
    {
        addrh++;
        addrl=0;
    }
    else
        addrl++;
}
delay(100);
outportb(port0,0);
outportb(port1,0);
outportb(port2,0);
outportb(port3,0);
}
/*-----format data in row-----*/
void tex_in_row(void)
{

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

char *texbuff;
int i,n,m;
    texbuff=texbuffer;
    for(i=0;i<1004;i++)
    {
if(parity_chck(*texbuff)==0)
    *texbuff=ERRC;
else
    *texbuff=*texbuff&0x7f;
texbuff++;
    }
for(n=0;n<8;n++)
{
    texrow[0][n]=0x20;
}
texbuff=texbuffer+6;
for(n=8;n<40;n++)
{
    texrow[0][n]=*texbuff;
    texbuff++;
}
for(m=1;m<24;m++)
{
    texbuff=texbuff+2;
    for(n=0;n<40;n++)
    {
        texrow[m][n]=*texbuff;
        texbuff++;
    }
}
free(texbuffer);

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

}
/*-----*/
int parity_chk(char tex)
{
    struct bit_chk
    {
        unsigned int b0:1;
        unsigned int b1:1;
        unsigned int b2:1;
        unsigned int b3:1;
        unsigned int b4:1;
        unsigned int b5:1;
        unsigned int b6:1;
        unsigned int b7:1;
    };
    union data_chk
    {
        struct bit_chk check0;
        unsigned char check1;
    }data;
    data.check1=tex;
    data.check0.b0=data.check0.b0^data.check0.b1;
    data.check0.b2=data.check0.b2^data.check0.b3;
    data.check0.b4=data.check0.b4^data.check0.b5;
    data.check0.b6=data.check0.b6^data.check0.b7;
    data.check0.b0=data.check0.b0^data.check0.b2;
    data.check0.b4=data.check0.b4^data.check0.b6;
    data.check0.b0=data.check0.b0^data.check0.b4;
    if(data.check0.b0)
        return(1);
    else

```

```

    return(0);
}
/*-----*/
void tex_display(void)
{
    int xpos=0,ypos=0,i,j,grp=2;
    unsigned char cha[2];
    Initialize();
    restorecrtmode();
    setgraphmode(GraphMode);
    settextstyle(0,0,1);
    cha[1]=NULL;
    for(i=0;i<8;i++)
    {
        tex=texrow+i;
        for(j=0;j<40;j++)
        {
            cha[0]=*tex;
            if(cha[0]<0x20)
            {
                if(cha[0]>0x10)
                    grp=1;
                if((cha[0]<8)&&(cha[0]>0))
                    grp=2;
                cha[0]=0x20;
                outtextxy(xpos,ypos,&cha[0]);
            }
        }
        if(grp==1)/*---graphics character---*/
        {
            if(((cha[0]>0x1f)&&(cha[0]<0x40))||((cha[0]>0x5f)&&(cha[0]<0x80)))
            {

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

        gputbox(cha[0],xpos,ypos);
    }
    else
        outtextxy(xpos,ypos,&cha[0]);
}
else /*----text character----*/
    outtextxy(xpos,ypos,&cha[0]);
    xpos=xpos+WID;
    tex++;
}
    xpos=0;
ypos=ypos+Y3;
}
    getch();
    closegraph();
}
/*-----init to graphics mode-----*/
void Initialize(void)
{
    int xasp, yasp; /* Used to read the aspect ratio*/
    GraphDriver = DETECT; /* Request auto-detection*/
    initgraph( &GraphDriver, &GraphMode, "" );
    ErrorCode = graphresult(); /* Read result of initialization*/
    if( ErrorCode != grOk ){ /* Error occured during init*/
        printf(" Graphics System Error: %s\n", grapherrormsg( ErrorCode ) );
        exit( 1 );
    }
    MaxX = getmaxx();
    MaxY = getmaxy(); /* Read size of screen*/
}
/*-----bit map data-----*/

```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
void gputbox(char font,int xpos,int ypos)
```

```
{
    struct bit
    {
        unsigned int b0:1;
        unsigned int b1:1;
        unsigned int b2:1;
        unsigned int b3:1;
        unsigned int b4:1;
        unsigned int b5:1;
        unsigned int b6:1;
        unsigned int b7:1;
    };
    union dbyte
    {
        struct bit aa;
        unsigned char bb;
    }gbyte;
    int x,y;
    gbyte.bb=font;
    if(gbyte.aa.b6)
    {
        for(y=ypos+Y2;y<ypos+Y3;y++)
        {
            for(x=xpos+X1;x<xpos+X2;x++)
            {
                putpixel(x,y,1);
            }
        }
    }
    if(gbyte.aa.b3)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```

{
  for(y=ypos+Y1;y<ypos+Y2;y++)
  {
    for(x=xpos+X1;x<xpos+X2;x++)
      {
        putpixel(x,y,1);
      }
  }
}

if(gbyte.aa.b1)
{
  for(y=ypos+Y0;y<ypos+Y1;y++)
  {
    for(x=xpos+X1;x<xpos+X2;x++)
      {
        putpixel(x,y,1);
      }
  }
}

if(gbyte.aa.b4)
{
  for(y=ypos+Y2;y<ypos+Y3;y++)
  {
    for(x=xpos+X0;x<xpos+X1;x++)
      {
        putpixel(x,y,1);
      }
  }
}

if(gbyte.aa.b2)
{

```

```

for(y=ypos+Y1;y<ypos+Y2;y++)
{
for(x=xpos+X0;x<xpos+X1;x++)
{
    putpixel(x,y,1);
}
}
}
if(gbyte.aa.b0)
{
for(y=ypos+Y0;y<ypos+Y1;y++)
{
for(x=xpos+X0;x<xpos+X1;x++)
{
    putpixel(x,y,1);
}
}
}
}
/*-----*/

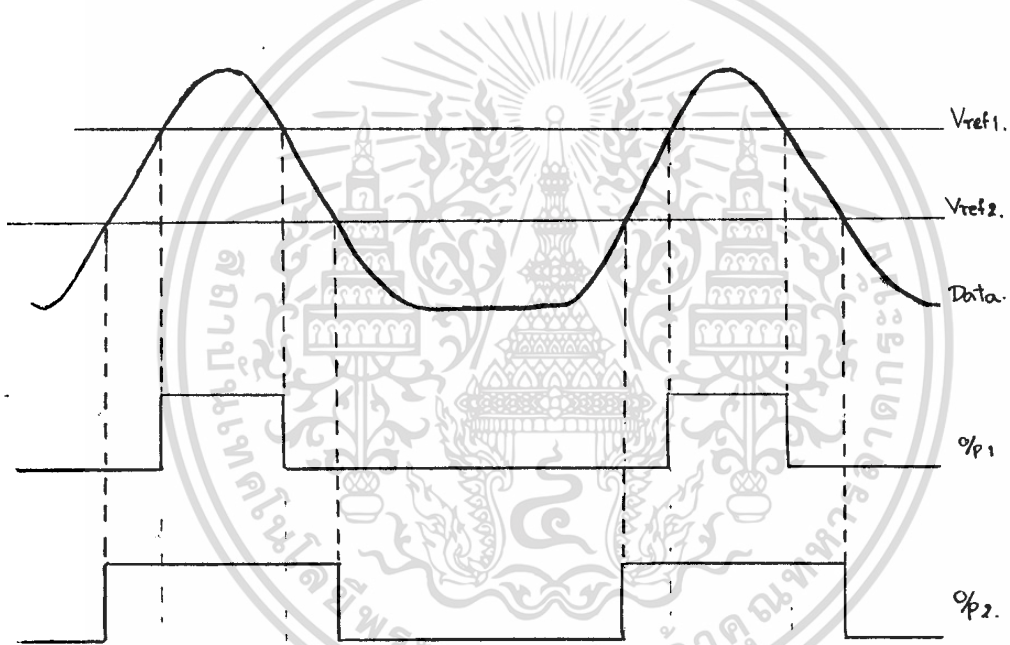
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
 ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 4.

## การทดลอง และผลการทดลอง

การทดลองจำเป็นต้องใช้สัญญาณเทเลเท็กซ์จากช่อง 5. ซึ่งออกอากาศเป็นเวลาจากการทดลอง สามารถตัดสัญญาณเทเลเท็กซ์ออกจากสัญญาณภาพได้แล้วโดยใช้วงจร 3.2.2 ในหน้า 3-3 และวงจร 3.3.1 ในหน้า 3-4 ซึ่งเป็นวงจรตีคิงที่ แยกซิงค์และตัดสัญญาณทดสอบออกจากสัญญาณเทเลเท็กซ์ แต่สัญญาณเทเลเท็กซ์ที่เป็นข้อมูลดิจิทัลนั้น เราได้มาจากการใช้วงจรคอมพาราเตอร์เป็นตัวกำหนดค่า "1" และ "0" ของข้อมูล แต่จากการทดลองพบว่าสัญญาณของเทเลเท็กซ์เองมีการเคลื่อนที่ขึ้นลงตลอดเวลา แต่แรงดันอ้างอิงของคอมพาราเตอร์นั้นเป็นค่าคงที่ ทำให้สัญญาณเทเลเท็กซ์ที่เป็นดิจิทัลนั้นอาจจะมั่วค่าที่ผิดพลาดได้ ดังรูป



และเมื่อนำเอาสัญญาณเทเลเท็กซ์ที่เป็นข้อมูลแบบอนุกรมนีมาแปลงเป็นข้อมูลแบบขนาน ก็ยังไม่สามารถที่จะทำได้ เนื่องจากวงจรในส่วนนี้จะทำงานได้โดยอาศัยสัญญาณนาฬิกาที่จะต้องมีความถี่ 6.9375 MHz ซึ่งเป็นอัตราความเร็วบิตของข้อมูลเทเลเท็กซ์ แต่การผลิตสัญญาณนาฬิกาที่มีความถี่ 6.9375 MHz นี้ยังไม่สามารถที่จะทำได้ เนื่องจากไม่มีคริสตอลที่มีความถี่เป็นจำนวนเท่าของความถี่นี้ ทำให้ได้ข้อมูลแบบขนานที่ไม่ตรงกับสัญญาณที่ส่งมา คือ อาจจะมีการกระโดดของข้อมูลบางบิตไป และจากเหตุผลอื่นนี้ทำให้วงจรทั้งหมดหลังจากวงจรตัดสัญญาณทดสอบออก ไม่สามารถที่จะทำงานได้ เนื่องจากต้องใช้สัญญาณนาฬิกาตัวเดียวกันนี้ ควบคุมการทำงานทั้งหมด ดังนั้นวงจรต่าง ๆ ดังกล่าวจึงเป็นเพียงวงจรที่คิดขึ้นจากความเข้าใจใน

ระบบเทเลเท็กซ์เท่านั้น ซึ่งทำให้ยังไม่สามารถที่จะทำการทดลองได้

ในส่วนของวงจรรีโมทเฟสนั้น วงจรสามารถส่งข้อมูลได้ โดยได้ทดลองส่งข้อมูล (ตัวอักษร) และดูรหัสที่วงจรโดยการต่อ LED ที่เอาท์พุท ตัวอักษรที่กดก็จะให้ค่ารหัสแอสกีออกมาที่เอาท์พุทนี้

ในส่วนของซอร์ฟแวร์ ได้ทำเมนูขึ้นเพื่อความสวยงามและเป็นการบอกรหัสใช้เครื่อง ซึ่งในส่วนนี้ได้ทำการทดลองแล้ว



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า  
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

## บทที่ 5.

## วิจารณ์และสรุปผล

สัญญาณเทเลเท็กซ์ที่ใช้ทำการทดลอง จะออกอากาศเป็นเพียงบางช่วงเท่านั้น คือ ออกพร้อมกับรายการปกติของทางสถานีช่อง 5. และสัญญาณภาพของโทรทัศน์ก็ได้มาจากโทรทัศน์ที่ใช้เสอากาศในตัวเป็นตัวรับคลื่น จึงทำให้ได้สัญญาณที่ไม่คมชัด มีสัญญาณรบกวนมากมาย

เราไม่สามารถใช้เครื่องเล่นวิดีโอบันทึกรายการของทางสถานีช่อง 5. เพื่อที่จะนำสัญญาณดังกล่าวมาเป็นอินพุทของวงจรได้ เนื่องจากเครื่องเล่นวิดีโอมีแบนด์วิทประมาณ 3 MHz แต่สัญญาณเทเลเท็กซ์นั้นมีอัตราความเร็วบิตเท่ากับ 6.9375 MHz จึงทำให้สัญญาณที่ออกมาจากเครื่องเล่นวิดีโอมีความผิดพลาดขึ้น นอกจากจะใช้เครื่องเล่นวิดีโอที่มีแบนด์วิทมากกว่าปกติ

ในส่วนของการทำสัญญาณเทเลเท็กซ์ที่มาถึงเครื่องรับให้เป็นดิจิทัลนั้น เราใช้วงจรคอมพิวเตอร์ดังที่กล่าวมาแล้ว ซึ่งจะทำได้ข้อมูลบิตของเทเลเท็กซ์ที่มีความผิดพลาดได้ ซึ่งแนวทางแก้ไขอาจจะใช้วงจรเอจีซี (AGC) เพื่อให้วงจรหาค่าเฉลี่ยให้ และนำค่าเฉลี่ยดังกล่าวไปเป็นแรงดันอ้างอิงของคอมพิวเตอร์ ซึ่งค่าดังกล่าวก็จะขึ้นลงตามสัญญาณเทเลเท็กซ์

เนื่องจากยังไม่สามารถสร้างสัญญาณนาฬิกาที่มีความถี่ตรงกับอัตราความเร็วบิตของข้อมูลได้ จึงทำให้ไม่สามารถเก็บข้อมูลที่ถูกต้องเข้าหน่วยความจำได้

การจับสัญญาณเทเลเท็กซ์ไม่สามารถจะทำได้ เพราะว่ามีสโคปจับสัญญาณได้ไม่เพียงพอที่จะมองสัญญาณได้ เนื่องจากสัญญาณจากโทรทัศน์มีการเปลี่ยนแปลงตลอดเวลา อีกทั้งสัญญาณที่ได้มาก็มีสิ่งรบกวนค่อนข้างมาก จึงทำให้ไม่ทราบว่าสัญญาณที่ได้นั้นถูกต้องหรือไม่

## กิตติกรรมประกาศ

ขอขอบพระคุณ รศ. ประกิจ ตั้งติสานนท์ ที่ให้คำแนะนำ อุปกรณ์ และสถานที่ในการวิจัย

เอกสารและคำแนะนำเกี่ยวกับระบบтелеเท็กซ์จากสถานีโทรทัศน์ช่อง 5.

คำแนะนำเกี่ยวกับระบบтелеเท็กซ์จากสถานีโทรทัศน์ช่อง 9.

อาจารย์ โกศล ชานขยัน ที่ให้คำแนะนำ



## เอกสารอ้างอิง

1. B.W.Kernigham and D.M.Ritchie , "The C Programming Language" , Bell Laboratories , 1985
2. IBM , "IBM Technical Reference" , 1981
3. Jan Gecsei , "The Architecture of Videotex Systems", Prentice-Hall, 1983
4. Mitchell W., Stephen P. and Donald M , "C Primer Plus" , Howard W. Sams & Co., Inc. ,1985
5. Paul Hurly , Matthias Laucht , Denis Hlynka , "The Videotex and Teletext Handbook :Home and office Communications using Micro-computer and terminals." , Harper & Row , 1985
6. ทีมงานสัมภาษณ์พาไปดู , "เทคโนโลยีเครือข่ายสื่อสารข้อมูลทางอากาศ" , เซมิคอนดักเตอร์อิเล็กทรอนิกส์ , ฉบับที่ 104 , มกราคม-กุมภาพันธ์ , พ.ศ. 2534 , บริษัทซีเอ็ดยูเคชั่น
7. บุญเลิศ เขี่ยมทัศนาศ , ยืน กุวารารวม , สมนึก ศิริโต , "โปรแกรมคอมพิวเตอร์ภาษาซี" , บริษัท ซีเอ็ดยูเคชั่น จำกัด , 2521
8. ประกิจ ตั้งติสานนท์ , "ทฤษฎีโทรทัศน์ หลักการและเครื่องรับโทรทัศน์" , คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าลาดกระบัง , 2527