โครงข่ายบริการสื่อสารร่วมดิจิตอล

Integrated Service Digital Network

(ISDN)

โดย

นาย กิตติ เลิศศิริมิตร

น.ส. จารุณี ประทะดี

นาย ธนากร สิริพิเชียร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17401 PROJECT I

เรื่อง

โครงข่ายบริการสื่อสารร่วมด้วยดิจิตอล

Integrated Service Digital Network

(ISDN)

โดย

นาย กิตติ	เลิศศิริมิตร	321019
น.ส. จารุณี	ปะทะดี	321052
นาย ธนากร	สิริพิเชียร	321112

อาจารย์ที่ปรึกษา

ผศ.ดร. บุญวัฒน์ ถัดชู

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

ภาคการศึกษาที่ 1 ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ปริญญานิพนธ์ปีการศึกษา 2535

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง โครงข่ายบริการสื่อสารร่วมด้วยดิจิตอล

ผู้จัดทำ

1. นาย กิตติ เลิศศิริมิตร
2. น.ส. จารุณี ปะทะดี
3. นาย ธนากร สิริพิเชียร

.....อาจารย์ที่ปรึกษา

(ผศ.ดร. บุญวัฒน์ อัตชู)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

032755

ปริญญานิพนธ์ปีการศึกษา 2535

ภาควิชา วิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เรื่อง การออกแบบวงจรรวมเฉพาะกิจโดยใช้ AHDL

ผู้จัดทำ

1. นาย กิตติ เลิศศิริมิตร
2. น.ส. จารุณี ปะทะดี
3. นาย ธนากร สิริพิเชียร

.....อาจารย์ที่ปรึกษา

(ผศ.ดร. บุญวัฒน์ อัตชู)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้ง 032755

โครงข่ายบริการสื่อสารร่วมด้วยดิจิตอล

บทคัดย่อ

โครงงานนี้เป็นการศึกษาระบบการสื่อสารแบบ Packet Switching และ Integrate Service Digital Network (ISDN)

Packet Switching เป็นระบบการสื่อสารแบบ Store and Forward Switching (รับมาเก็บไว้ แล้วส่งต่อ) ทำงานโดยการแบ่งข้อมูลออกเป็นขนาดเล็กๆ ที่เรียกว่า Packet แล้วส่งไปยังโครงข่ายข้อมูล (network) ซึ่งเป็นโครงข่ายข้อมูลที่มีประสิทธิภาพ และความน่าเชื่อถือของข้อมูลสูงมาก มีการตรวจสอบและแก้ไขข้อผิดพลาดของข้อมูล ทำให้ใช้สายส่งได้คุ้มค่า โดยจะส่งข้อมูลเมื่อต้องการใช้จริงๆ และเมื่อไม่ต้องการส่งข้อมูล ก็จะนำสายสื่อสารไปให้กับผู้อื่นซึ่งต้องการส่งในตอนนั้นใช้แทน

โปรโตคอลที่ใช้ใน Packet Switching และเป็นโปรโตคอลซึ่งเป็นส่วนหนึ่งของการศึกษาในโครงงานนี้ก็คือ X.25

ระบบเครือข่าย Integrated Service Digital Network หรือเรียกย่อๆ ว่า ISDN คือโครงข่ายที่สามารถให้บริการสื่อสารทั้งเสียง, ข้อความ และภาพ ได้ภายในชุมสายเดียวกัน ด้วยความสามารถของ ISDN นั้น ถ้าเรามีหมายเลขโทรศัพท์หมายเลขหนึ่ง เราสามารถใช้อุปกรณ์ต่างๆ ไม่ว่าจะเป็นโทรศัพท์, เทเล็กซ์, โทรภาพ (FAX), วิดีโอเท็กซ์ หรือเครื่องคอมพิวเตอร์ได้พร้อมกันหลายๆ อย่างในเวลาเดียวกัน

โครงงานนี้เป็นการศึกษาโปรโตคอลซึ่งใช้กับระบบ ISDN ซึ่งได้ถูกกำหนดโดย CCITT และครอบคลุมในส่วนของ OSI layer 1,2,3

แม้ว่าโครงงานนี้จะศึกษาโปรโตคอล 2 ชนิด คือ โปรโตคอลของ ISDN และ X.25 แต่จะเน้นศึกษาโปรโตคอล ISDN เป็นหลัก ส่วนโปรโตคอล X.25 นั้นจะเป็นส่วนเสริมในจุดที่ ISDN สามารถรองรับการใช้งาน X.25 ได้เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Integrated Service Digital Network

(ISDN)

ABSTRACT

This Project is studying of the 2 main future telecommunication, Packet switching and ISDN.

Packet switching is a store and forward switching where its concept is based on partition data into small pieces called packet, and sending them through network. Packet switchingis a very efficient and reliable network. It lets us use the network more economical than today telephone network, by using communication line only when necessary.

Protocol used by Packet switching is X.25 which is the one of studying in this project.

The other section of the studying in this project is Integrated Service Digital Network, know as ISDN. The based concept is construct network that can serve many services (such as telephone, telex, facimile, teletex, videotext, videophone and computer) at the same time by only 1 communication line.

This project focuses on ISDN protocol that is developed by CCITT which covers OSI layer 1,2,3.

Although this project studies in 2 main protocols, ISDN is its main point. ISDN,by its own,can support X.25.Studying X.25 in this project is only added in the section that ISDN can support X.25.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบวงจรรวมเฉพาะกิจโดยใช้ AHDL

บทคัดย่อ

วงการอิเลคทรอนิคส์ในปัจจุบัน มีวงจรที่ซับซ้อนมากขึ้น และมีการใช้งานวงจรรวมกันอย่างกว้างขวาง จนอาจกล่าวได้ว่าวงจรรวมเป็นหัวใจของอิเลคทรอนิกส์ในปัจจุบัน

การออกแบบวงจรรวม เราสามารถออกแบบได้ 2 วิธีหลัก ๆ คือออกแบบ layer ต่างๆ ของวงจรรวมแล้วส่งไปให้โรงงานเจือสาร วิธีนี้จะใช้เวลานานและ เสียค่าใช้จ่ายสูงมากและต้องผลิตจำนวนมากจึงจะคุ้มทุน

ส่วนอีกวิธีหนึ่งคือ การออกแบบ โดยใช้ PLD (Programmable Logic Device) ซึ่งวิธีนี้เราจะออกแบบเพียง Schematic ของวงจรไม่ต้องออกแบบถึงระดับ Layer และสามารถ Program ลงบนตัว IC ได้เลยโดยไม่ต้องส่งโรงงานเจือสาร ทำให้ลดเวลาในการออกแบบและลดต้นทุนได้ เหมาะกับการผลิตที่ปริมาณไม่มากอีกทั้งในปัจจุบันเราสามารถออกแบบวงจรของเราโดยใช้ Hardware Description Language (HDL) ซึ่งมีลักษณะคล้ายกับภาษา C ทำให้ง่ายต่อการออกแบบเป็นอย่างมาก

โครงงานนี้เป็นการศึกษาการออกแบบวงจรรวมโดยใช้ Hardware Description Language โดยจะนำมาประยุกต์ใช้ในการออกแบบวงจร ที่เกี่ยวกับการประมวลผลภาพ (Image Processing) โดย Software ที่ใช้ คือ MAXPLUS II ของบริษัท Altera และ Hardware Description Language ที่ใช้ คือ AHDL (Altera Hardware Description Language)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ASIC Design Using AHDL

ABSTRACT

Today , Electronic field has much more complex circuit than the past and widely uses Integrated Circuit (IC). Now, Integrated Circuit plays a very important role in Electronics, so we can say that it is the heart of electronics.

There are 2 main methods to design IC. First we must design the layers of the IC and send its to fabricate at the factories. This way, it is very expensive if we fabricate in small quantity and it takes a long time to make an IC.

The other way , we use the technology of PLD (Programmable Logic Device). To implement , we only design schematics of our circuit , need not to design layers , and we can program the schematics that we have designed to the PLD by using only our own microcomputers.Its advantages are saving time to fabricate IC and cheaper if we don't use in a large quantity. Besides , we can use Hardware Description Language (HDL) to implement our designs. HDL is a language similar to high level language ,such as C. It makes us very convenient to design circuit.

This project studies about designing IC by AHDL and use the result to program to PLD. The circuit is about hardware for image processing. Software that is used is Altera's MAXPLUS II and Altera Hardware Description Language (AHDL).

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนำ

เนื่องจากโครงงานปริญญานิพนธ์ ที่ได้ศึกษาในเทอมที่ 1 คือ ระบบโครงข่ายบริการสื่อสารร่วมด้วยดิจิตอล ได้เกิดปัญหาและอุปสรรคในการศึกษาขึ้น คือไม่สามารถที่จะทำการทดลองได้จริง เนื่องจากไม่มีอุปกรณ์ในการทดลองหลายๆ อย่าง เช่น อุปกรณ์จำลองชุมสายระบบ ISDN เครื่องวิเคราะห์โปรโตคอล ทำให้ไม่สามารถที่จะทำต่อในเทอมที่ 2 ได้ จึงได้มีการเปลี่ยนแปลงหัวข้อโครงงาน โดยปรึกษากับอาจารย์ที่ปรึกษา และในที่สุดก็ได้หัวข้อปริญญานิพนธ์ใหม่ขึ้นมา นั่นคือ การออกแบบวงจรรวมเฉพาะกิจโดยใช้ AHDL (Hardware Description Language)

AHDL เป็นภาษาที่ใช้ในการบรรยายรูปแบบการทำงานของอุปกรณ์ฮาร์ดแวร์ในลักษณะข้อความ โดยในโครงงานนี้ได้ใช้ AHDL ในการสร้างวงจรเกี่ยวกับการประมวลผลภาพ 3 วงจร วงจรแรก คือ วงจร Histogram Equalization ซึ่งออกแบบโดย นาย กิตติ เลิศศิริมิตร วงจรที่สอง คือวงจร Edge Detection ออกแบบโดย น.ส. จารุณี ปะทะดี และวงจรสุดท้าย คือวงจร Median Filter ซึ่งออกแบบโดย นาย ธนากร สิริพิเชียร

สำหรับในรายงานฉบับนี้ได้แบ่งออกเป็น 2 ส่วน โดยส่วนแรกจะเป็นเรื่องระบบ ISDN ที่ได้ทำการศึกษาในเทอมที่ 1 และส่วนที่ 2 จะเป็นเรื่อง การออกแบบวงจรรวมเฉพาะกิจโดยใช้ AHDL ซึ่งได้ทำการศึกษาและสร้างขึ้นมา ในเทอมที่ 2

สุดท้ายต้องขอขอบคุณ อาจารย์ที่ปรึกษาอย่างมาก ที่ช่วยให้คำแนะนำ และจัดหาอุปกรณ์ต่างๆ ที่ใช้ในการทำโครงงานนี้ จนกระทั่งโครงงานนี้สำเร็จลุล่วงไปด้วยดี และถ้าหากรายงานฉบับนี้มีข้อผิดพลาดประการใด ทางคณะผู้จัดทำก็ขออภัยมา ณ ที่นี้ด้วย

นาย กิตติ เลิศศิริมิตร

น.ส. จารุณี ปะทะดี

นาย ธนากร สิริพิเชียร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PART 1 : ISDN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนำ

ในปัจจุบันเรามีเครือข่ายการสื่อสารอยู่หลายแบบ เช่น เครือข่ายโทรศัพท์ เทเล็กซ์ วิดิโอเทกซ์ ฯลฯ ซึ่งแต่ละเครือข่ายต่างก็ทำหน้าที่คล้ายๆ กัน คือการส่งข้อมูลเหมือนกัน จึงเกิดแนวคิดที่จะรวมเครือข่ายทั้งหลายเหล่านี้เข้าเป็นเครือข่ายเดียว ซึ่งสามารถส่งข้อมูลต่างๆ ได้ทุกชนิดทั้งภาพ เสียง และข้อมูลต่างๆ เช่น โทรศัพท์ แฟกซ์ เทเล็กซ์ โทรสาร วิดิโอเทกซ์ วิดิโอโฟน ข้อมูลคอมพิวเตอร์ ฯลฯ ด้วยความคิดนี้ทำให้เกิดเครือข่ายการสื่อสารระบบใหม่ขึ้นมา ที่เรียกว่า Integrated Service Digital Network (ISDN)

ในรายงานฉบับนี้จะแบ่งออกเป็น 2 ตอน โดยตอนแรกจะกล่าวถึงระบบเครือข่าย ISDN ซึ่งจะเน้นโปรโตคอล ที่เป็นมาตรฐานของ CCITT ตั้งแต่ชั้นที่ 1 ถึงชั้นที่ 3 และตอนที่ 2 จะกล่าวถึงเรื่องเครือข่าย Packet Switching ซึ่งใช้โปรโตคอล X.25 แต่โดยรวมแล้วจะเน้นเรื่องระบบ ISDN มากกว่า Packet Switching

สำหรับความสำเร็จของโครงงานนี้ต้องขอขอบคุณอาจารย์ที่ปรึกษา ที่ได้ให้คำแนะนำหลักการในการศึกษาโครงงานนี้ และรายงานฉบับนี้หากมีข้อผิดพลาดประการใดทางคณะผู้จัดทำก็ขออภัยมา ณ ที่นี้ด้วย

นาย กิตติ เลิศศิริมิตร
น.ส. จารุณี ปะทะดี
นาย ธนากร สิริพิเชียร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อ I

ABSTRACT II

คำนำ III

วัตถุประสงค์ของโครงงาน 1

ตอนที่ 1 ISDN Network

บทนำ ระบบเครือข่าย ISDN 2

บทที่ 1 ISDN services 12

1.1 บริการที่ต้องมี

1.2 บริการและ Attributes ของ ISDN

1.3 บริการอื่น ๆ ของ ISDN

บทที่ 2 Physical layer 27

2.1 โครงสร้างของเฟรม

2.1.1 โครงสร้างเฟรมของเบสิกแอกเซส

บทที่ 3 Data Link layer 35

3.1 LAPD Frames

3.2 การอ้างแอดเดรสของ LAPD

3.3 ลักษณะของ SAPI

3.4 ลักษณะของ TEI

บทที่ 4 Network layer 42

4.1 Layer 3 USER-NETWORK Signaling

4.2 Structure of the Protocols

4.3 Basic Circuit-Mode calls

4.4 Controlling and Invoking Supplementary

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Services

4.5 Q.931 Message Format

4.6 ADDRESSINGs

ตอนที่ 2 Packet Switching Network

บทที่ 5 Packet Switching 55

5.1 องค์ประกอบของ Packet Switching

5.1.1 Local Access Component (LAC)

5.1.2 Packet Assemblers/Disassemblers (PAD)

5.1.3 Packet Switching Nodes (PN)

5.1.4 Network Links (NL)

5.1.5 Network Management Center (NMC)

5.2 Packet Switching และ Thaipak

5.3 โปรโตคอล X.25

บทวิจารณ์และสรุป 84

ภาคผนวก

แผนงานและเป้าหมายการพัฒนาโครงข่ายโทรศัพท์เข้าสู่ ISDN 85

บรรณานุกรม 87

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาการทำงาน และการบริการของระบบเครือข่ายร่วม (ISDN)
2. ทำการศึกษารูปแบบ และโครงสร้างของโปรโตคอล X.25
3. ทำการพัฒนาซอฟต์แวร์ และ ฮาร์ดแวร์เพื่อนำมาใช้ในการเชื่อมต่อกับระบบ ISDN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

ระบบเครือข่าย ISDN

เครือข่ายดิจิตอลในการให้บริการรวม (ISDN)

แนวความคิดพื้นฐานของเครือข่ายดิจิตอลในการให้บริการรวม หรือที่เรียกทับศัพท์กันว่า ISDN นี้ เกิดขึ้นมาจากการที่ในปัจจุบันมีเครือข่ายที่ให้บริการด้านการสื่อสารอยู่หลายชนิด ซึ่งแต่ละชนิดได้ถูกออกแบบสำหรับงานเฉพาะอย่างเท่านั้น เมื่อนำไปให้บริการพิเศษแบบอื่นจึงไม่สามารถใช้ประโยชน์เครือข่ายนั้นได้เต็มที่

ตัวอย่างที่เห็นได้ชัดในปัจจุบันคือ การใช้เครือข่ายโทรศัพท์ในการส่งข้อมูลคอมพิวเตอร์ ซึ่งไม่สามารถส่งข้อมูลได้ด้วยความเร็วสูง เนื่องจากตัวโมเด็มที่ใช้ต้องทำงานในย่านความถี่เสียงเท่านั้น ระบบ ISDN เป็นเครือข่ายเดียวที่ให้บริการด้านการสื่อสารทุกรูปแบบ โดยมีระบบสื่อสารรวมสำหรับการใช้งานทุกชนิด มีการเข้าถึงเครือข่ายที่เป็นมาตรฐานเพื่อให้อุปกรณ์สำหรับการสื่อสาร หรือเครื่องคอมพิวเตอร์ที่ออกแบบมาสำหรับ ISDN ต่างชนิดติดต่อกันได้และที่สำคัญอุปกรณ์ดังกล่าว ที่มีอยู่ก่อนการเกิดขึ้นของ ISDN ต้องสามารถติดต่อกับ ISDN ได้ด้วย การเกิดขึ้นของ ISDN จึงมีประโยชน์อย่างเด่นชัดมากในเรื่องของประสิทธิภาพต่อราคา เนื่องจากมีเพียงเครือข่ายเดียวที่ให้บริการด้านการสื่อสารได้ทุกอย่างแทนที่จะมีหลายๆ เครือข่าย เพื่อให้บริการหลายๆ แบบ นอกจากนี้อัตราเร็วในการส่งข้อมูลในระบบ ISDN สามารถทำได้สูงกว่าเครือข่ายในปัจจุบันมาก

เครือข่ายในปัจจุบัน

เครือข่าย (Network) ไม่ว่าจะเป็นแบบใดก็ตาม ต้องมีองค์ประกอบพื้นฐาน 3 ประการ ดังนี้คือ

1. อุปกรณ์ชุมสาย (Switching equipments) เพื่อทำหน้าที่สลับสายหรือข้อมูล
2. ระบบสื่อสัญญาณ (Transmission system) เพื่อทำหน้าที่เป็นตัวกลางให้สัญญาณผ่านไปได้
3. อุปกรณ์ปลายทาง (Terminal equipments) เพื่อทำหน้าที่ส่งหรือรับหรือทั้งส่งทั้งรับข้อมูล

คำว่า "บริการ" ในที่นี้หมายถึงบริการทางด้านการสื่อสาร ซึ่งบริการชนิดหนึ่งๆ อาจทำได้บนเครือข่ายหลายๆ ประเภทก็ได้หรือเครือข่ายชนิดหนึ่งอาจให้บริการการสื่อสารได้มากกว่า 1 ชนิดก็ได้

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ไม่ว่ากรณีใดๆ ทั้งสิ้น ไม่อนุญาตให้ทำซ้ำ ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ดังที่กล่าวมาแล้วว่า ISDN เป็นเครือข่ายเดียวที่ให้บริการการสื่อสารทุกแบบ จึงต้องขอกล่าวก่อนว่าในปัจจุบันมีเครือข่ายและการให้บริการอะไรบ้าง ดังนี้

1. เครือข่ายโทรศัพท์ (Telephone network)

2. เครือข่ายเทเล็กซ์ (Telex network)

3. เครือข่ายข้อมูลสาธารณะแบบเซอร์กิตสวิตช์ (Public Circuit Switched Data Network หรือ PCSN)

4. เครือข่ายข้อมูลสาธารณะแบบแพกเก็ตสวิตช์ (Public Packet Switched Data Network หรือ PPSN)

การให้บริการบนเครือข่าย

การให้บริการบนเครือข่ายแบบต่างๆ ที่จะกล่าวต่อไปนี้ บางเครือข่ายสามารถให้บริการได้มากกว่า 1 ชนิด เช่น เครือข่ายโทรศัพท์ นอกจากจะให้บริการทางด้านการติดต่อด้วยเสียง ยังสามารถให้บริการ การติดต่อระหว่างคอมพิวเตอร์ เครื่อง ATM หรือโทรสารก็ได้ บริการต่างๆ ที่จะกล่าวถึงมีดังนี้

บริการโทรศัพท์

ให้บริการด้านเสียงพูดซึ่งเป็นแบบอนาลอก แม้ว่าในปัจจุบันชุมสายโทรศัพท์ส่วนมากจะเปลี่ยนเป็นระบบดิจิตอลกันมากแล้วก็ตาม แต่ในแง่ของสายที่ต่อจากบ้านผู้เช่าไปยังชุมสาย ยังคงเป็นระบบอนาลอกอยู่

การให้บริการเทเล็กซ์ระหว่างชาติ (International Telex Service)

บริการเทเล็กซ์มองง่ายๆ ก็คือการส่งข้อมูลพิมพ์ดีดจากที่หนึ่งไปพิมพ์ออกอีกที่หนึ่ง บริการชนิดนี้ทำกันอยู่บนเครือข่ายเทเล็กซ์ ซึ่งต้องมีชุมสายเทเล็กซ์ สายงานและเครื่องเทเล็กซ์ อัตราเร็วต่ำมากประมาณ 50 บิตต่อวินาที ใช้รหัส Baudot ในการส่ง

การให้บริการเทเลเท็กซ์ (Teletex Service)

การให้บริการเทเลเท็กซ์ต่างกับเทเล็กซ์คือ แทนที่จะเป็นเครื่องพิมพ์ดีดก็กลายเป็นเครื่องคอมพิวเตอร์ที่ทำหน้าที่เวิร์ดโปรเซสเซอร์ จึงสามารถเก็บข้อมูลที่พิมพ์และสามารถเรียกขึ้นมาแก้ไขใหม่จากดิสก์ได้ ความเร็วจะเร็วกว่าเทเล็กซ์ประมาณ 10 เท่า

บริการโทรสาร (Facsimile or Telefax Service)

บริการนี้คือการถ่ายเอกสารแล้วส่งให้กันในระยะไกล บริการแบบนี้มีประโยชน์ที่เห็นได้ชัดในด้าน การส่งรูปภาพ ลายเซ็น ลายมือ ซึ่งเทเล็กซ์กับเทเลเท็กซ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ทำไม่ได้ บริการโทรสารทำบนเครือข่ายโทรศัพท์ ตัวส่งจะทำการสแกนแผ่นกระดาษที่ส่ง จากนั้นส่งสัญญาณนี้ลงไปในเครือข่ายโทรศัพท์ให้ฝ่ายรับ ซึ่งทำการแปลงสัญญาณดังกล่าวไปพิมพ์ในแบบดอตแมตริกซ์

บริการเท็กซ์แฟก (Texfax Service)

ลักษณะการให้บริการแบบนี้คล้ายๆ โทรสารกับเทเลเท็กซ์รวมกัน คือทางด้านรูปภาพจะถูกเข้ารหัสและส่งในหลักการของโทรสาร ส่วนข้อความจะส่งในหลักการของเทเลเท็กซ์ การที่แยกกันแบบนี้ก็เพื่อเตรียมไว้ใช้ในระบบ ISDN นั่นเอง

บริการวิดีโอเท็กซ์ (Videotex Service)

บริการแบบนี้เป็นแบบโต้ตอบกัน ระหว่างเครื่องคอมพิวเตอร์ซึ่งปกติมักผ่านเครือข่ายโทรศัพท์ ด้านเครื่องของผู้ใช้อาจเป็น เครื่องคอมพิวเตอร์ส่วนบุคคล หรือเป็นโทรทัศน์ พร้อมด้วยคีย์บอร์ดชนิดพิเศษและ โมเด็มเพื่อติดต่อขอข้อมูลกับฐานข้อมูล โดยมีคอมพิวเตอร์ส่วนกลางสำหรับเป็นตัวควบคุม การให้บริการจะเป็นแบบมีเมนูมาให้เลือก ถ้าผู้ที่ติดต่อเข้ามาพอใจรายการอะไรก็สามารถเลือกโดยผ่านคีย์บอร์ด ตัวอย่างของข้อมูลมีดังนี้

- สอบถามรวมทั้งจองตั๋วเครื่องบินด้วยตนเอง
- สอบถามสภาพอากาศ
- ราคาหุ้น รายการเงินในธนาคารของตนเอง

บริการวิดีโอเท็กซ์แบบออกอากาศ (Broadcast videotex Service)

ในประเทศไทยเรียกบริการแบบนี้ว่า เทเลเท็กซ์ (Teletext) รูปแบบการให้บริการแตกต่างจากวิดีโอเท็กซ์ธรรมดา ตรงที่มีการเอาข้อมูลจากฐานข้อมูลมาทำการมอดูเลชั่นแล้วส่งออกอากาศ การส่งข้อมูลจะส่งเป็นหน้าๆ ออกมาในช่วงแบลงค์กิ้งของเส้นราสเตอร์จอภาพของทีวีธรรมดา ผู้ใช้บริการจึงไม่สามารถโต้ตอบกับฐานข้อมูลได้ ต้องรอดูจนกว่าจะมีการเปลี่ยนหน้าต่อไป ในประเทศไทยสามารถรับได้โดยตรงจาก ทีวีสีช่อง 9 (อสมท)

เครือข่าย ISDN

ISDN นั้นตามคำจำกัดความซึ่งให้โดย คณะกรรมการที่ปรึกษาโทรเลขและโทรศัพท์ระหว่างประเทศ (CCITT) คือ โครงข่ายที่วิวัฒนาการมาจากโครงข่ายโทรศัพท์ระบบดิจิตอล (IDN) ที่สามารถส่งทอดสัญญาณดิจิตอลได้สมบูรณ์ โดยไม่มีการเปลี่ยนแปลง และมี

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ต่อเชื่อมมาตรฐานชุดหนึ่ง เพื่อให้บริการได้หลายรูปแบบ

จากคำจำกัดความนี้ เราอาจจะมองว่าเป็นคำจำกัดความที่ค่อนข้างกว้าง ซึ่งถ้าเรามามองในทางปฏิบัติจริงๆ แล้ว เราอาจแบ่ง ISDN ออกได้โดยใช้ความสามารถของโครงข่ายเป็นเกณฑ์ ได้เป็น Narrow band ISDN (ISDN-N) ซึ่งเป็น ISDN ในยุคเริ่มแรก และ Broad band ISDN (ISDN-B) เป็นโครงข่าย ที่เพิ่มขีดความสามารถขึ้นจาก ISDN-N และ Universal ISDN (ISDN-U) เป็นโครงข่าย ISDN ที่สมบูรณ์

การพัฒนาเครือข่าย ISDN

ในการพัฒนาระบบ ISDN นั้น เราสามารถแบ่งได้เป็น 3 ระยะ คือ

ระยะแรก คือ *Narrow band ISDN (ISDN-N)* เป็นการพัฒนาระบบ ISDN เพื่อใช้ในการสื่อสารต่าง ๆ ทั้งทางเสียง ข้อมูลคอมพิวเตอร์ โทรสาร แฟกซ์

ระยะที่ 2 คือ *Broad band ISDN (ISDN-B)* เป็นโครงข่ายที่พยายามขีดความสามารถของโครงข่าย ISDN ให้สามารถส่งทอดสัญญาณภาพในระดับภาพทีวีได้ ในการนี้ จำเป็นต้องวางใยแก้วนำแสงจากชุมสายไปถึงบ้านผู้ใช้บริการ

ระยะที่ 3 คือ *Universal ISDN (ISDN-U)* เป็นโครงข่ายที่เกิดจากการประสานกันโดยสมบูรณ์ทั้งทางด้านเข่ายสาย ชุมสาย และด้านสื่อส่งสัญญาณ

โครงข่าย ISDN-U นี้ มีคุณสมบัติสำคัญ 5 ประการ คือ

1. **ด้านอุปกรณ์เชื่อมต่อมาตรฐาน** อุปกรณ์เชื่อมต่อมาตรฐาน ซึ่งมีอยู่หลายแบบใน ISDN-N และ ISDN-B จะพัฒนาจนถึงระดับที่มีเพียงแบบเดียว ซึ่งนำเอาอุปกรณ์ปลายทางแบบใดก็ได้มาต่อใช้ได้ เช่น เดียวกับปลั๊กไฟฟ้า
2. **ด้านรูปแบบโครงข่าย** จะเป็นรูปแบบโครงข่ายซึ่งสามารถปรับเปลี่ยนได้โดยอิสระแทนที่จะต้องเป็นรูปแบบที่ตายตัว
3. **ด้านโครงสร้างช่องสัญญาณ** จะเป็นรูปแบบอิสระที่ผู้ใช้สามารถกำหนดและเรียกใช้ตามความต้องการใช้บริการในแต่ละครั้งได้
4. **ด้านเครื่องชุมสาย** จะมีเพียงชุมสายเดียวที่สามารถให้บริการทุกรูปแบบตามความต้องการของผู้ใช้บริการในแต่ละครั้งได้อย่างมีประสิทธิภาพ
5. **ด้านบริการ** ผู้ใช้บริการจะสามารถกำหนดและ เรียกใช้บริการได้ตามความต้องการในแต่ละครั้งของการเรียกใช้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คุณลักษณะทางการส่งสัญญาณภายในเครือข่าย ISDN

การติดต่อระหว่างชุมสายในปัจจุบันนั้นส่วนมากจะใช้ระบบ digital และใช้หลักการที่เรียกว่า Pulse Code Modulation (PCM) ระบบ PCM นั้นจะมีการแบ่งสายสัญญาณออกเป็นแชนแนล มาตรฐานของระบบ PCM ตามมาตรฐานของยุโรปจะมี 32 แชนแนล ส่วนของอเมริกา และญี่ปุ่น มี 24 แชนแนล และเนื่องจากสัญญาณที่ส่งนั้นเป็นดิจิตอล ทำให้ผู้ออกแบบระบบสามารถส่งสัญญาณเสียงพูด ข้อมูลดิจิตอล และสัญญาณที่ใช้ในการติดต่อ (signalling) ลงในแชนแนลต่าง ๆ ตามต้องการ โดยทั่วไปจะมีแชนแนลสำหรับส่งสัญญาณการติดต่อ (signalling)จะมีเพียง 1 แชนแนล นอกนั้นจะเป็นแชนแนลสำหรับส่งเสียงพูด และข้อมูลดิจิตอลต่าง ๆ

โปรโตคอลที่ใช้ติดต่อระหว่างชุมสาย ISDN มีชื่อเรียกว่า CCITT No. 7 common channel signalling

แชนแนลในระบบ ISDN

CCITT ได้กำหนดแชนแนลหลักๆ ในระบบ ISDN ดังนี้คือ

1. แชนแนล "A" เป็นแชนแนลที่ใช้กับสัญญาณอนาลอก ซึ่งมีแบนด์วิดธ์ขนาด 3.1 กิโลเฮิรตซ์

2. แชนแนล "B" เป็นแชนแนลพื้นฐานในการส่งข้อมูลดิจิตอล ทำงานรับส่งด้วยอัตราเร็ว 64 กิโลบิต/วินาที ข้อมูลที่ส่งในแชนแนล B อาจได้รับการสวิตช์ แบบเซอร์กิตสวิตช์หรือแพคเก็ตสวิตช์ก็ได้

3. แชนแนล "D" ในเบื้องต้นได้กำหนดไว้ว่าแชนแนล D ทำหน้าเป็นช่องสัญญาณสำหรับใช้ในการติดต่อ ซึ่งอยู่ในรูปแพคเก็ต แต่ในบางครั้งอาจใช้ส่งข้อมูลดิจิตอลความเร็วต่ำก็ได้ อัตราเร็วของแชนแนล D กำหนดไว้ 2 แบบ คือ 16 กิโลบิต/วินาที และ 64 กิโลบิต/วินาที ขึ้นอยู่กับผู้ใช้ว่าต้องการติดต่อกับ ISDN ด้วยข้อมูลจำนวนมากหรือน้อย

ในแชนแนล D มีการใช้โปรโตคอลที่เรียกว่า LAP-D (Link Access Protocol on the D Channel) ซึ่งคล้ายคลึงกับ CCITT X.25 แต่มีการเพิ่มข้อมูลสำหรับการอ้างแอดเดรส เพื่อใช้ในการติดต่อแบบจุดต่อหลายๆ จุดได้

4. แชนแนล "E" เป็นอีกแชนแนลหนึ่งที่มีอัตราเร็ว 64 กิโลบิต/วินาที ใช้ในการส่งสัญญาณเพื่อการติดต่อกันคล้ายๆ กับแชนแนล D แต่ในชั้นที่ 2 ของมันมีพื้นฐานมาจาก CCITT No.7 ส่วนในชั้นที่ 3 เป็นกระบวนความในการควบคุม ซึ่งเหมือนกับในแชนแนล D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขนาด 64 กิโลบิต/วินาที แชนแนล E ใช้ในระยะเริ่มต้นเพื่อเชื่อมโยง PABX กับ ระบบ ISDN โดยผ่านแชนแนลขนาด 2.048 เมกะบิต/วินาที

5. แชนแนล "H" เป็นแชนแนลที่ใช้ส่งข้อมูลด้วยอัตราเร็วสูงมาก ใช้กับ ISDN ที่เรียกว่าแถบความถี่กว้างและบรอดแบนด์ ISDN แชนแนล H มีหลายแบบคือ H1 H2 H3 และ H4 H1 มีอัตราเร็วในการส่งข้อมูลขนาด 384 กิโลบิต/วินาที แชนแนล H1 ยังประกอบด้วย H1(1) ที่ทำงานด้วยอัตราเร็ว 1536 กิโลบิต/วินาที และ H1(2) ทำงานด้วยอัตราเร็ว 1920 กิโลบิต/วินาที

แชนแนล H2 มีอัตราเร็ว 30 เมกะบิต/วินาที แชนแนล H3 70 เมกะบิต/วินาที และแชนแนล H4 อัตราเร็ว 140 เมกะบิต/วินาที

โครงสร้างช่องสัญญาณ

ISDN ในระยะเริ่มต้นนี้จะเป็นแบบย่านความถี่แคบ หรือ narrow band ISDN ซึ่งประกอบด้วยแชนแนลที่มีอัตราเร็ว 2 แบบ คือ

1. Basic Access (BA) ใช้ติดต่อกับผู้ใช้ธรรมดา
2. Primary Rate Access (PRA) ใช้ติดต่อกับผู้ใช้ที่ต้องการข้อมูลจำนวนมากด้วยความเร็วสูง เช่น PABX, LAN และ DATABASE

Basic Access

เบสิกแอกเซส ประกอบด้วยแชนแนล B จำนวน 2 วงจร และแชนแนล D จำนวน 1 วงจร มัลติเพล็กซ์เข้าด้วยกัน ดังนั้นอัตราเร็วในการถ่ายทอดข้อมูลคือ 2B+D ซึ่งถ้าแชนแนล D เป็น 16 กิโลบิท/วินาที

เบสิกแอกเซส = 2*64+16 = 144 กิโลบิท/วินาที

ถ้าแชนแนล D = 64 กิโลบิท/วินาที

เบสิกแอกเซส = 2*64+64 = 192 กิโลบิท/วินาที

ในการส่งข้อมูล ผู้ใช้สามารถส่งข้อมูลจากอุปกรณ์ที่มีความเร็วเท่าใดก็ได้ แต่ความเร็วรวมของทุกอุปกรณ์จะต้องไม่เกิน 144 กิโลบิท/วินาที หรือ 192 กิโลบิท/วินาที

Primary Rate Access

ไพมารีเรทแอกเซส ประกอบด้วยแชนแนล B จำนวน 30 วงจร สำหรับส่ง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ข้อมูลดิจิตอลหรือเสียง และแชนแนล D อีก 1 วงจร นอกจากนี้ยังมีการเพิ่มแชนแนลขนาด 64 กิโลบิต/วินาที เพื่อใช้ในการซิงโครไนท์ไทม์มิ่งและการควบคุม ทำให้ผลรวมของอัตราเร็วทั้งหมดเป็น 2.048 เมกะบิต/วินาที

การอินเตอร์เฟสกับผู้ใช้

เพื่อให้ผู้ผลิตสามารถผลิตอุปกรณ์ที่ได้มาตรฐาน CCITT จึงได้มีการกำหนดจุดอ้างอิงต่างๆ เอาไว้ คือ R S T U และ V นอกจากนี้ยังได้กำหนดหน้าที่ของอุปกรณ์ต่างๆ ที่ใช้ในระบบและอุปกรณ์เชื่อมต่อ โดยเรียกว่า Functional/Protocol Blocks ซึ่งมีดังนี้ คือ

1. TE1 (Terminal Equipment 1) คือ อุปกรณ์ที่ออกแบบมาเพื่อใช้กับระบบ ISDN เช่น โทรศัพท์ระบบดิจิตอล สัญญาณที่ออกจากระบบดังกล่าวเป็นสัญญาณดิจิตอล และได้ถูกออกแบบให้สอดคล้องกับ มาตรฐานของเบสิกแอกเซส จุดอ้างอิงที่ TE1 ต่ออยู่เรียกว่าจุด S

2. TE2 (Terminal Equipment 2) เป็นอุปกรณ์ที่ไม่ได้ออกแบบมาเพื่อใช้ กับ ISDN (non-ISDN Terminal) ซึ่งก็คืออุปกรณ์ที่เราใช้กันในปัจจุบันนี้ อุปกรณ์ดังกล่าวทำงานแตกต่างกันในแง่การส่งข้อมูล โปรโตคอลและรูปแบบของการส่ง เช่น V.24 terminal, X.25 packet terminal จุด R เป็นจุดอ้างอิงที่ใช้ในการอินเตอร์เฟสระหว่าง TE2 กับ TA

3. TA (Terminal Adapter) เป็นอุปกรณ์อินเตอร์เฟสระหว่าง TE2 กับ ระบบ ISDN หน้าที่หลักของ TA มีดังนี้คือ
   - 3.1 เปลี่ยนแปลงอัตราเร็วในการส่งข้อมูลของ TE2 ที่จุด R ให้ได้เอาท์พุทที่จุด S เป็นสัญญาณดิจิตอลขนาด 64 กิโลบิต/วินาที ถ้า TE2 ใช้อุปกรณ์อะนาลอกสัญญาณที่จุด R จะต้องมีการแปลงให้เป็นดิจิตอลในตัว TA เพื่อให้ได้เอาท์พุทเป็นดิจิตอลที่จุด S
   - 3.2 เปลี่ยนแปลงโปรโตคอลที่ใช้ในการติดต่อเพื่อส่งข้อมูลของ TE2 ปัจจุบัน CCITT ได้กำหนดว่ามี TA 3 แบบ เพื่อใช้กับ X.21 และ X.21 bis circuit terminal , X.25 packet terminal และ อนุกรม V

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

terminal

TA จึงต้องแปลงโปรโตคอลที่ใช้กับอุปกรณ์ทั้ง 3 ให้เป็นโปรโตคอลที่ใช้กับแชนแนล D ที่เรียกว่า LAP-D

4. NT1 (Network Termination 1) เป็นอุปกรณ์ที่ต่อเชื่อมกับปลายสายของผู้ใช้ เพื่อการเชื่อมต่อทางไฟฟ้า อุปกรณ์นี้ติดตั้งให้โดยบริษัทที่ให้บริการ ISDN จุดอ้างอิงที่ NT1 ต่อเชื่อมกับ NT2 คือ จุด T

5. NT2 (Network Termination 2) เป็นอุปกรณ์ที่ทำหน้าที่สวิตชิ่งและทำการรวบรวมข้อมูลของผู้ใช้ ตัวอย่างของ NT2 คือ PABX หรือ LAN

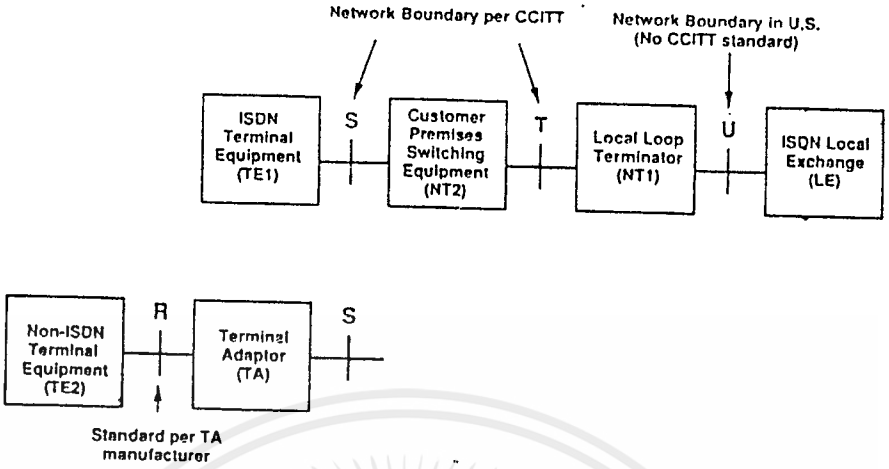
NT2 มีพอร์ทเพื่อต่อเทอร์มินอลได้สูงสุด 8 พอร์ท โดยมีรูปแบบของการต่อเป็น แบทัส

ที่จุดอ้างอิง S ถ้าหากมีอุปกรณ์เพียงตัวเดียวเท่านั้นที่สามารถต่อกับ NT2 ได้โดยตรงจุดอ้างอิง S และ T จะเป็นจุดเดียวกัน ในกรณีนี้จึงไม่มีการใช้ NT2 คงมีเพียง NT1 เท่านั้น NT1 หรือ NT2 นี้ติดตั้งอยู่ในบริเวณบ้านหรือสำนักงานของผู้ใช้

6. LT (Line Termination) เป็นอุปกรณ์ที่ติดตั้งอยู่ในชุมสาย ISDN ใช้สำหรับการต่อสายของผู้ใช้ที่ชุมสาย จุดอ้างอิงอยู่ระหว่าง LT กับ NT1 คือจุด U

7. ET (Exchange Terminal) เป็นอุปกรณ์ที่ติดตั้งอยู่ในชุมสาย ISDN สำหรับทำหน้าที่ส่งสัญญาณเพื่อการเริ่มต้นการติดต่อระหว่างชุมสาย ISDN

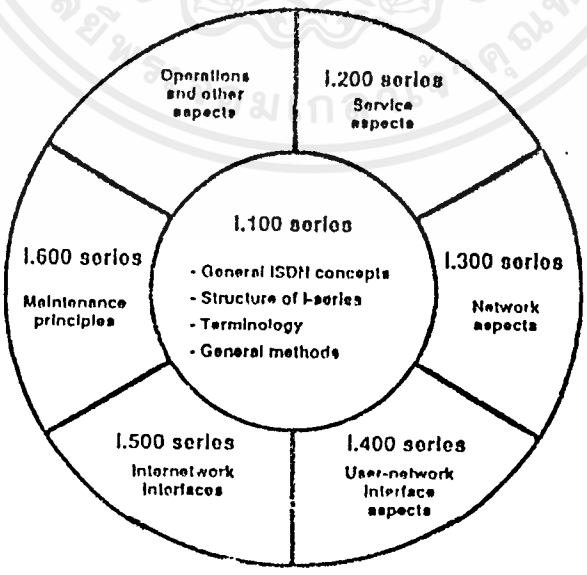
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้
032755



รูปที่ 1 แสดงจุดอ้างอิงต่าง ๆ ใน ISDN

โครงสร้างทั่วไปของข้อกำหนดมาตรฐานระบบ ISDN

CCITT ได้วางข้อกำหนดเพื่อให้โครงข่าย ISDN เป็นโครงข่ายที่สามารถรับการติดต่อสื่อสารถึงกันได้ในโครงข่ายของผู้ใช้บริการ และในระหว่างโครงข่ายดิจิตอลด้วยกัน (INTERWORK INTERFACES) รูปแบบโครงสร้าง CCITT ได้กำหนดไว้ใน I-series ซึ่งผลการประชุมครั้งที่ 9 MELBOURNE 1988 ได้กำหนดโครงสร้างไว้ดังรูปที่ 2



รูปที่ 2 โครงสร้างทั่วไปของข้อกำหนดมาตรฐานระบบ ISDN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

CCITT ได้กำหนดหลักเกณฑ์ใหญ่ๆ ไว้ดังนี้

1. เพื่อวางมาตรฐานในด้านการให้บริการ (Standardization of services) แก่ผู้ใช้บริการ
2. เพื่อวางมาตรฐานในด้านการเชื่อมโยงระหว่างโครงข่ายกับโครงข่ายของผู้ใช้ user-network interface กล่าวคือ เพื่อให้อุปกรณ์ปลายทาง terminal equipment สามารถพกพาไปใช้งานที่ใดก็ได้
3. เพื่อวางมาตรฐานในด้านขีดความสามารถของโครงข่าย ให้สามารถมีการติดต่อระหว่าง user-to-network และ network-to-network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

ISDN Services

ISDN มีบริการต่างๆ มากมาย ในส่วนนี้จะกล่าวถึงบริการต่างๆ ของเครือข่าย ISDN นอกจากนี้การที่มีบริการมากมายหลายอย่าง ทำให้อุปกรณ์ที่ต่อเชื่อมกับเครือข่าย ISDN ต้องมีวิธีการบอกเครือข่าย ถึงบริการต่างๆ ที่ต้องการ ซึ่งก็จะกล่าวถึงในส่วนนี้ด้วย

1.1 บริการที่ต้องมี

เนื่องจาก ISDN จะต้องมีบริการต่างๆ มากมาย ดังนั้นความสามารถของ network ที่จะให้บริการ ISDN ได้ จะต้องมีดังนี้

- สามารถจัดการข้อมูล voice, audio, interactive หรือ bulk data (ข้อมูลเป็นกลุ่ม), ข้อมูล fax, compressed video, และ full-motion video ได้
- สามารถส่งข้อมูลได้ทั้งแบบ continuous traffic (ต่อเนื่อง) และ bursty traffic (ไม่ต่อเนื่อง) ได้
- จัดสรร bandwidth ให้เมื่อต้องการใช้ได้
- อนุญาตให้มี fast call establishment และ termination (คือการต่อเชื่อมคู่สายให้อย่างรวดเร็ว)
- จัดการเกี่ยวกับ ความเร็วในการส่ง และ call hold times ได้หลายแบบ
- ประกันความผิดพลาด,ล่าช้า ของข้อมูลให้อยู่ในอัตราที่ต่ำ
- สามารถจัดการ Security ในระดับต่าง ๆ

ตาราง 1.1 และ 1.2 เป็นรายการความต้องการการบริการของอุปกรณ์ต่าง ๆ ที่ใช้ในที่พักอาศัย (residences) และธุรกิจ (business)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Service	Required Bandwidth	ISDN Channel Type			Switching Facility		
		B	H	D	Circuit Switched	Packet Switched	Channel Switched
Telephone	8, 16, 32, 64 kbps	•			•		
Alarm systems	10-100 bps			•		•	
Utility metering	0.1-1 kbps			•	•	•	
Energy management	0.1-1 kbps			•	•	•	
Interactive information svcs.	4.8-64 kbps	•		•		•	
Electronic mail	4.8-64 kbps	•		•		•	
Broadcast video	96 Mbps		•				•
Switched video	96 Mbps		•				•
Interactive video	96 Mbps		•		•		•

ตาราง 1.1 ความต้องการบริการของที่พักอาศัย

Service	Required Bandwidth	ISDN Channel Type			Switching Facility		
		B	H	D	Circuit Switched	Packet Switched	Channel Switched
Telephone	8, 16, 32, 64 kbps	•			•		
Interactive data communications	4.8-64 kbps	•		•		•	
Electronic mail	4.8-64 kbps	•		•		•	
Bulk data transfer	4.8-64 kbps	•			•		
Facsimile/graphics	4.8-64 kbps	•			•		
Slow scan TV	56-64 kbps	•			•		
Videoconferencing	1.544 Mbps		•				•

ตาราง 1.2 ความต้องการบริการของธุรกิจ

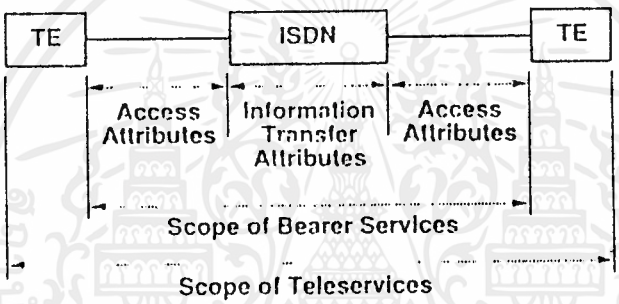
1.2 บริการ และ Attributes ของ ISDN

การบริการ ของ ISDN สามารถแบ่งย่อย ออกเป็น 2 ประเภทตามขอบเขตของการบริการ (แสดงในรูป 1.1) คือ

- *Bearer Services* ให้บริการผู้ใช้ในการส่งข้อมูลจากอุปกรณ์หนึ่งไปยังอีกอุปกรณ์หนึ่ง ข้อมูลจะถูกส่งไปโดยจะเกี่ยวข้องกับการทำงานของ Layer ต่างๆ ในระดับล่าง (เช่น OSI Layer 1 ถึง 3 เป็นต้น)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- *Teleservices* เป็น value-added service (การบริการที่เพิ่มเติมขึ้นจาก bearer services) ที่จัดหาให้โดย network กล่าวคือ Teleservices จะให้บริการ user แบบ end-to-end communication และจะเกี่ยวข้องกับ layer ระดับล่าง (คือ bearer service) และ Layer ระดับที่สูงขึ้น (คือ value-added service)



รูปที่ 1.1 Bearer และ Teleservices

Bearer Services

เราสามารถกำหนดคุณลักษณะต่างๆ ของ Bearer Services ได้โดยอาศัย attributes ซึ่งประกอบไปด้วย access attributes,information transfer attributes และ general attributes

- Access attributes จะอธิบายถึงวิธีการที่ user จะ access network
- Information transfer attributes กล่าวถึงการส่งข้อมูลผ่าน network
- ส่วน General attributes จะอธิบายถึงคุณลักษณะอื่น ๆ ของ bearer service เช่น supplementary services (บริการเสริม),quality-of-service parameters และ internetworking

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Information Transfer Attributes

ตาราง 1.3 แสดง Information Transfer Attributes ทั้ง 7 ตัว แต่ละตัวอธิบายถึงการเชื่อมต่อระหว่าง user และ ISDN

Attribute	Possible values of attribute									
Information transfer attributes										
1. Information transfer mode	Circuit								Packet	
2. Information transfer rate	Bit rate (kbps)								Throughput (packets/sec)	
	64	2 x 64	384	1536	1920	Other values (FFS)			(FFS)	
3. Information transfer capability	Unrestricted Digital info.		Speech	3.1 kHz audio	7 kHz audio	15 kHz audio	Video	Other value (FFS)		
4. Structure	8 kHz integrity		Service data unit integrity			TSSI		RDTD	Unstructured	
5. Establishment of communication	Demand			Reserved					Permanent	
6. Symmetry	Unidirectional			Bidirectional symmetric					Bidirectional asymmetric	
7. Communication configuration	Point to point			Multipoint					Broadcast (FFS)	

ตาราง 1.3 Bearer Service Information Transfer Attributes

Information Transfer modes บอกถึง mode ในการส่งข้อมูลว่าจะเป็น *Circuit Mode* หรือ *Packet mode*

Information Tranfer rate คือ throughput ในการติดต่อสื่อสาร ซึ่งกำหนดเป็น bits per second สำหรับ circuit-mode และ packets per second สำหรับ packet mode ใน circuit mode เราสามารถกำหนด information transfer rate ได้ คือ 64,2*64,384,1536 สำหรับค่าอื่นๆ และ ขณะปัจจุบัน ทาง ยุโรปและ ทั่วโลก ยังไม่มีการกำหนดออกมา จนต้องมีการศึกษาเพิ่มเติมต่อไป (for further study - FFS)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Information transfer capability บอกถึงชนิดของข้อมูลที่ส่ง

- *Unrestricted digital information* เป็นข้อมูลแบบ bit stream ซึ่งทุก ๆ บิทจะอยู่ในรูป Octet และเป็นค่า default ของ packet mode

- *Speech และ 3.1-kHz audio* เป็นคำพูด และเสียง (audio) ซึ่งทั้ง 2 อย่างใช้ แชนแนลที่มีความสามารถส่งในระดับส่งเสียงได้จำนวน 1 แชนแนล (single vioce-grade channel) ที่ต้องแยกออกเป็น 2 ประเภท เพราะ วิธีการในการ compression และ encoding ต่างกัน

- *Seven และ 15-kHz audio* ให้คุณภาพของเสียงเหมือนกับสถานีส่งวิทยุแบบ mono และ stero ตามลำดับ

- *Video* เป็นการส่งสัญญาณภาพ

Speech จะเป็นค่า default สำหรับ information transfer capability สำหรับ circuit-mode service

Structure บอกถึงหน่วยของการส่งซึ่งมีการประกันความถูกต้อง (guarantee)

- *Service data unit integrity* ประกันว่าข้อมูล หรือ packet ที่ส่งไปจะอยู่ในรูปแบบเดิมเมื่อไปถึงผู้รับ เป็นค่า default สำหรับ packet mode

- *Eight-kHz integrity* บอกว่าบิททุกบิทที่ส่งไปทุก ๆ 125 us (8 kHz) จะไปถึงผู้รับภายในเวลา 125 us เป็นค่า default ของ circuit-mode structure

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- *Time slot sequence integrity (TSSI)* ใช้ในกรณีที่มี channels ติดต่อระหว่าง users หลาย channels โดยกล่าวว่าข้อมูลที่ส่งจะถึงฝ่ายรับในลำดับเดิมไม่เปลี่ยนแปลง

- *Restricted differential time delay (RDTD)* ใช้ร่วมกับ 8-kHz intergrity เพื่อกำหนดว่าข้อมูลจะถูกส่งไปยังฝ่ายรับภายในระยะเวลา 50 ms หลังจากที่ฝ่ายส่งเริ่มทำการส่ง

- *Unstructured* หมายถึงไม่มีการกำหนด structure ใดๆ

Establishment of communication บอกว่าจะให้บริการเมื่อใด

- *Demand* หมายถึงจะให้บริการก็ต่อเมื่อ user ต้องการใช้เท่านั้น คล้ายกับระบบโทรศัพท์แบบ dial-up ในปัจจุบัน และเป็นค่า default สำหรับ attribute นี้

- *Reserved* หมายถึง user สามารถใช้ service และสามารถกำหนดเวลาที่จะใช้ service ภายหลังได้ ซึ่งสามารถกำหนดได้ว่าเมื่อถึง เวลาที่กำหนด แล้ว ถ้า channels ไม่ว่างจะให้ยกเลิกการใช้ , เลื่อน การใช้ออกไปจนกระทั่ง channel ว่าง หรือ อาจให้คนอื่น ๆ ที่ใช้ channel อยู่เลิกใช้ แล้วนำ channel ที่ว่างนี้มาใช้แทนก็ได้

- *Permanent* เหมือนกับ leased line ที่ใช้กันอยู่ในปัจจุบัน

Symmetry บอกถึงการไหลของข้อมูล

- *Unidirectional* หมายถึงการส่งทางเดียว

- *Bidirectional symmetric* หมายถึงการส่งข้อมูล 2 ทิศทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยที่อัตราการส่งข้อมูลเท่ากันทั้ง 2 ฝ่าย

- *Bidirectional asymmetric* หมายถึงการส่งข้อมูล 2 ทิศทาง โดยที่อัตราการส่งข้อมูลไม่เท่ากัน

Communication Configuration บอกถึง configuration ในการติดต่อกัน ระหว่าง users ว่าจะเป็นแบบ Point-to-point, Multipoint หรือ Boardcast

Access Attributes

Access attributes (ตาราง 1.4) บอกถึงข้อมูลเกี่ยวกับ channel type และ protocols ที่ user ใช้ access network เป็นการบอกว่า network จะส่ง user's information อย่างไร

Attribute	Possible values of attributes						
Access attributes							
8. Access channel and rate	D (16 kbps)	D (64 kbps)	B	H_0	H_{11}	H_{12}	Others (FFS)
9.1— 9.3 Signaling access protocol layers 1-3	I.430/I.431	LAPD	I.451	X.25 LAPB	X.25 PLP		
	Rate adaption: I.461, I.462, I.463, I.465						
	Others (FFS)						
9.4— 9.6 Information access protocol layers 1-3	I.430/I.431	HDLC	LAPD	X.25 LAPB	X.25 PLP		
	G.711 PCM	G.721 ADPCM	I.451 (FFS)	T.70 Telex/FAX			
	Rate adaption: I.460, I.461, I.462, I.463, I.465						
	Others (FFS)						

ตาราง 1.4 Beare Service Access Attributes

Access channel and rate เป็นการบอกถึง channel ที่ user ต้องการใช้

Signaling access protocols บอกถึง protocols ที่ใช้สำหรับ user-network signaling

เอกสารนี้เป็นเอกสาร[illegible]นั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Information access protocols คือ protocols ที่ใช้สำหรับ แลกเปลี่ยนข้อมูลระหว่าง users

General Attributes and Supplementary Services

General Attributes แสดงอยู่ในตาราง 1.5 Attributes ส่วนใหญ่ในหมวดนี้ ยังไม่ถูกกำหนดออกมา ยังอยู่ในระหว่างการศึกษาอยู่แต่สำหรับ Supplementary services ซึ่งเป็นส่วนหนึ่งของ general attributes นั้นบางส่วนได้กำหนดออกมาแล้ว

Attribute	Possible values of attributes			
General attributes				
10. Supplementary services provided	Number Identification	Call Offering	Call Completion	Multiparty
	Community of Interest	Charging	Additional Information Transfer	
11. Quality of service	Under Study			
12 Interworking possibilities				
13 Operational and commercial aspects				

ตาราง 1.5 Bearer Service General Attributes

Supplementary services อนุญาตให้ network สามารถบริการผู้ใช้ได้มากกว่าที่จะเพียงแค่ส่งข้อมูลแต่ละบิทไปยังผู้รับ บริการเหล่านี้จะอยู่บนพื้นฐานของข้อมูล (information) ของ network ที่รู้มาก่อนแล้ว เช่น ชื่อผู้เรียก และไม่เกี่ยวกับการเปลี่ยนแปลง หรือแก้ไขข้อมูลของ user ที่ส่งโดย network

Supplementary services ที่กำหนดโดย CCITT ซึ่งอยู่ใน Recommendation I.250 มีดังนี้

Number Identification เป็น services ที่มีพื้นฐานมาจาก การแสดงหมายเลข ISDN ของฝ่ายผู้เรียก เช่น การแสดงหมายเลขผู้เรียกไปยังผู้รับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Call Offering เป็น services ที่มีผลต่อการติดต่อ และเส้นทาง ของการเรียก เช่น อนุญาตให้ผู้ใช้สามารถถ่ายการเรียกไปยังบุคคลที่ 3 ได้

Call Completion มีผลต่อความสำเร็จในการติดต่อหนึ่ง ๆ เช่น อาจมี การเตือนผู้ใช้ซึ่งกำลังใช้ channel อยู่นั้นว่ามีการเรียกเข้ามา

Multiparty เป็น services ที่อนุญาตให้มีการติดต่อระหว่าง user มากกว่า 1 คู่ เช่นอนุญาตให้ user หลาย ๆ คนสามารถสนทนาได้พร้อม ๆ กัน

Community of Interest อนุญาตให้เราสามารถมี เครือข่ายส่วนตัว ใน ISDN ซึ่งเป็นเครือข่ายสาธารณะ เช่นอาจตั้งกลุ่มของ user ขึ้นมา โดยภายในกลุ่ม นั้น user สามารถติดต่อกันเองได้ แต่จะมี users บางคนเท่านั้นที่สามารถติดต่อ กับ user อื่น ๆ ภายนอกกลุ่มได้

Charging ให้ข้อมูลเกี่ยวกับค่าบริการในการใช้ network และอนุญาตให้คิดค่าบริการกับบุคคลอื่น ซึ่งไม่ใช่ผู้เรียกได้

Additional Information Transfer อนุญาตให้มีการส่งข้อมูลที่เพิ่ม เติมไปจากข้อมูลทั่วไป เช่น สามารถจำกัดจำนวนข้อมูลที่ส่งในการเรียกแต่ละครั้ง

Teleservices

Teleservices เป็น value-added services (VAS) ที่จัดบริการเพิ่มเติมมาจาก Bearer Service ที่อธิบายข้างต้น โดยการบริการจะเกี่ยวข้องกับ Layer ที่สูงขึ้นมา คือ OSI Layer ที่ 4 ถึง 7

Teleservices สามารถให้บริการโดย User ใน network หรือตัว network เองก็ได้ โดยที่ attributes ต่าง ๆ ของแต่ละ Layer คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- ชนิดของ user information อันได้แก่ คำพูด, เสียง, อักษร (teletex),Group 4 facsimile, mixed-mode text facsimile, videotex, video และ interactive text(telex)

- Protocol ของ Layer 4 ซึ่งรวมทั้ง CCITT และ ISO transport layer protocals

- Protocol ของ Layer 5 ซึ่งรวมทั้ง CCITT และ ISO session layer protocals

- Protocol ของ Layer 6 ซึ่งรวมทั้ง CCITT และ ISO presentation layer protocals ยกตัวอย่างเช่น document interchange protocols, resolution และ graphic mode

- Protocol ของ Layer 7 ซึ่งรวมทั้ง CCITT และ ISO application layer protocols ยกตัวอย่างเช่น Message Handling System (MHS) และ directory services

1.3 บริการอื่น ๆ ของ ISDN

Rate Adaption

ในกรณีที่ Terminal Equipment ของเรามีความเร็วแตกต่างไปจากความเร็วของเครือข่าย ISDN เช่น 64 kbps สำหรับ channel B เราจะต้องมีการแปลงอัตราเร็วให้เหมาะสม

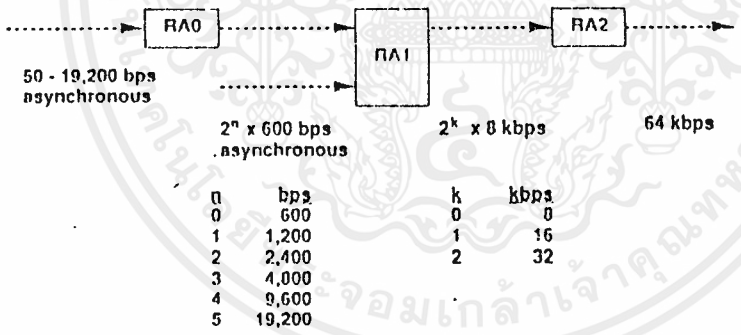
CCITT Recommendation I.460 อธิบายถึงหลักการในการเปลี่ยนอัตราเร็วและการ Multiplex ส่วน Recommendation X.30 (I.461) จะกล่าวถึงการรองรับการใช้งาน X.21 และ X.21 bis data terminal equipment บนเครือข่าย ISDN และที่สำคัญ X.21 และ X.21 bis เป็น physical layer ของ

เอกสารนี้เป็นการที่ได้รับอนุญาตให้ใช้เพื่อการศึกษาเท่านั้น ไม่อนุญาตให้ใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

synchronous public data networks และX.25 ด้วย

Recommendation X.31 (I.462) อธิบายถึงการใช้งาน X.25 DTEs บน ISDN

Recommendation V.110 (I.463) อธิบายถึงการใช้งาน synchronous และ asynchronous terminals บน ISDN การแปลงอัตราเร็ว (Rate Adaption) สามารถทำได้ 3 ขั้นตอนดังรูปที่ 1.2 ทั้งนี้จะขึ้นอยู่กับความเร็วของข้อมูลที่เข้ามา ในขั้นแรก (RA0) จะรับข้อมูล asynchronous ที่อัตราเร็วระหว่าง 50 ถึง 19,200 bps และแปลงให้อยู่ในรูปของ synchronous โดยการเพิ่มบิทพิเศษเข้าไป Synchronous bit ก็จะถูกแปลงต่อในขั้นตอนที่ 2 (RA1) ให้เป็นข้อมูลที่มีความเร็วขนาด 8,16 หรือ 32 kbps ซึ่งก็จะถูกแปลงต่อไปเป็น 64 kbps อีกต่อหนึ่ง Recommendation V.110 นี้ได้รับความนิยมมากในยุโรป และญี่ปุ่น



n	bps
0	600
1	1,200
2	2,400
3	4,800
4	9,600
5	19,200

k	kbps
0	8
1	16
2	32

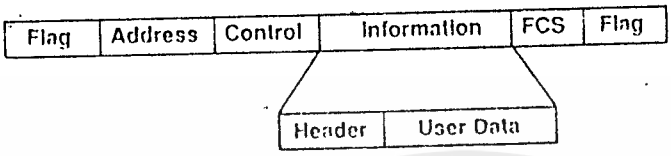
รูปที่ 1.2 Recommendation V.110 (I.463) rate adaption stages

Recommendation V.120 (I.465) กล่าวถึงการแปลงความเร็ว และการ Multiplex สำหรับการส่งข้อมูลแบบ asynchronous ที่อัตราเร็วไม่เกิน 19.2 kbps

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ การส่งข้อมูลแบบ synchronous ที่มีพื้นฐานบน HDLC protocol ซึ่งมีอัตราเร็วไม่เกิน 56 kbps

V.120 มีพื้นฐานมาจาก HDLC และใช้มาตรฐานของ bit-oriented protocol โดยมีรูปแบบของเฟรมดังรูป 1.3



รูปที่ 1.3 Recommendation V.120 (I.465) rate adaption frame format

Fields ต่าง ๆ ใน V.120 frame มีหน้าที่ ดังนี้

- Flag bit pattern (01111110) ใช้ในการบอกจุดเริ่มต้น และสิ้นสุดของเฟรม

- Address field ทำให้มี logical link identifier และทำให้ V.120 สามารถ multiplex users ที่มีความเร็วต่ำ หลาย ๆ เครื่องเข้าไปใน channel ขนาด 64 kbps เพียง channel เดียว

- Information field ประกอบด้วย Header และ User data Header จะเก็บข้อมูลต่าง ๆ เช่น การบอก errors, ควบคุมสถานะของ physical interface และบอกว่า user data นั้น ประกอบขึ้นอย่างไร ส่วน User data ก็คือ ข้อมูลของ user
- Frame Check Sequence (FCS) เป็นข้อมูลที่ใช้ตรวจจับข้อผิดพลาด

V.120 นี้นิยมใช้มากในสหรัฐอเมริกา ทั้ง V.110 และ V.120 นั้น ออกแบบมาเพื่อ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

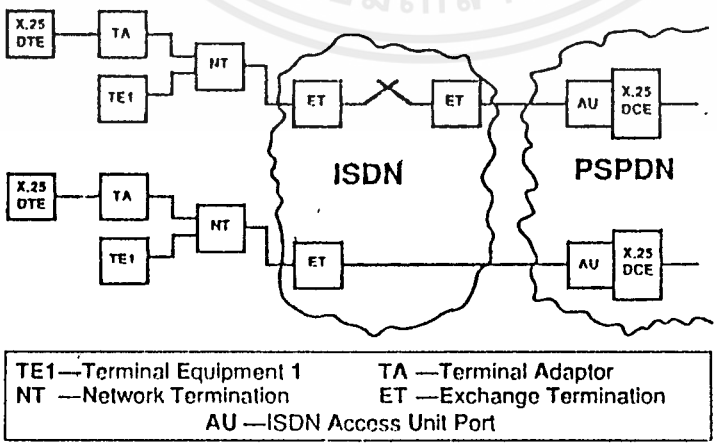
ใช้กับ Circuit-mode applications

Packet-mode Services

Services ที่สำคัญอีกอย่างหนึ่งของ ISDN ก็คือ ความสามารถในการส่งข้อมูลแบบ Packet-mode โดยใช้ protocol X.25 ซึ่งทำงานอยู่บนเครือข่าย Packet Switched Public Data Networks (PSPDNs) ได้ ในเครือข่าย ISDN สิ่งที่ทำหน้าที่นี้ก็คือ Terminal Adapter (TA) โดย TA จะใช้ protocol X.25 เชื่อมต่อกับ DTE ส่วนที่เชื่อมต่อกับ network ก็จะใช้ protocol ของ ISDN Recommendation X.31 (I.462) จะกำหนดการทำงานของ TA ซึ่งอนุญาตให้ X.25 DTE สามารถติดต่อกับเครือข่าย ISDN ได้

มี 2 วิธีในการที่ ISDN จะสนับสนุนการใช้งาน PSPDN คือ PSPDN เป็นส่วนหนึ่งของ ISDN (หรือเรียกว่า Case A) หรือ ISDN และ PSPDN แยกกันโดยอิสระ (Case B)

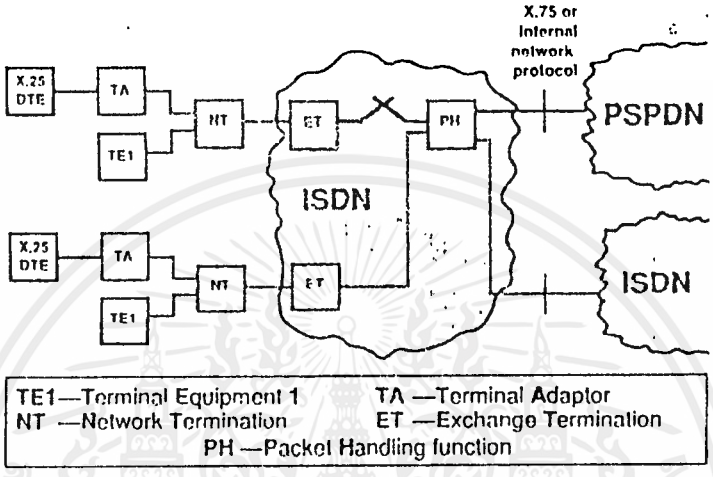
ใน Case A หรือที่เรียกว่า minimum service integration scenario (รูป 1.4) ข้อมูล packet จะส่งผ่าน exchange termination (ET) ไปยัง access unit (Au) port อย่าง transparently ซึ่งอาจเป็นการติดต่อระหว่าง 2 ETs แล้วจึงค่อยต่อไปยัง AU หรือต่อจาก ET ไปยัง AU เลยก็ได้ ในกรณีนี้ channel D จะใช้ส่ง signal เท่านั้น ไม่สามารถส่งข้อมูลได้



รูปที่ 1.4 Recommendation X.31 Case A (access to PSPDN services) Packet-mode service.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Case B หรือที่เรียกว่า maximum service integration scenario (รูปที่ 1.5) ในกรณีนี้จะเป็นการแยก PSPDN ออกจาก ISDN โดยจะมี packet handling (PH) ดังรูป 1.5 ทางด้านที่คล้ายกับเป็น X.25 DCE ในการเชื่อมต่อระหว่าง network ซึ่งอาจเป็น PSPDN หรือ ISDN ใช้ X.75



รูปที่ 1.5 Recommendation X.31 Case B (access to the ISDN Virtual circuit service) Packet-mode service.

Broadband Services (B-ISDN)

Boardband Services เป็นบริการที่มีความเร็วสูงกว่า Primary rate interface สำหรับ applications ของ B-ISDN เช่น video telephony, EQTV, HDTV เป็นต้น

B-ISDN Services สามารถแบ่งได้กว้าง ๆ ออกเป็น communication services และ distribution services

Communication Serivces จะประกอบไปด้วยบริการของโทรศัพท์ในปัจจุบัน บวกด้วย conversational, messaging และ retrieval services

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์ ใช้เพื่อการศึกษาเท่านั้น ห้ามมิให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Conversational Services ประกอบไปด้วยการส่งข้อมูลแบบ 2-way, real-time, end-to-end ระหว่าง users

Message services คือการที่ users ติดต่อกันโดยผ่านทาง mailboxes

Retrieval services จะอนุญาตให้usersสามารถดึงข้อมูลที่เก็บไว้ใน database ได้

Distribution services มี 2 แบบ คือ อยู่ในความควบคุมของ users (Distribution with user control) และไม่อยู่ในความควบคุมของ users (Distribution without user control)

Distribution services without user control หรือ pure broadcast services คือการส่งข้อมูลจากแหล่งข้อมูลหนึ่ง ๆ ที่ซึ่ง user ไม่สามารถควบคุมการเริ่มต้น หรือ หยุดการส่งได้ตัวอย่างการบริการ เช่น high-quality pay-per-view television, high-speed data financial reporting

ส่วน *Distribution services with user control* ก็มีลักษณะคล้ายกัน เพียงแต่ว่า users สามารถดูส่วนใด ส่วนหนึ่ง ของข้อมูลที่ส่งมาได้ สามารถควบคุมลำดับและ เลือกดูในส่วนต่าง ๆ ได้ ตัวอย่างการบริการ เช่น broadcast videography

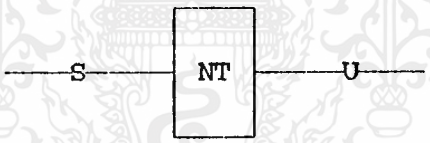
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

PHYSICAL LAYER

ใน physical layer ของระบบ ISDN จะอ้างอิงตามมาตรฐานของ CCITT Recommendation I.430 และ I.431 โดยที่ I.430 ใช้กับ Basic Access และ I.431 ใช้กับ Primary rate Access แต่ในที่นี้จะศึกษาแต่เพียง I.430 ของ Basic Access เท่านั้น

เมื่อพิจารณาการเชื่อมต่อแบบ Basic Access อาจกล่าวได้ว่า Network Termination เป็นอุปกรณ์ที่ต้องใช้งานคู่กับ Line Terminator (ที่ติดตั้งที่ชุมสาย) เสมออุปกรณ์ Network terminator จะถูกติดตั้งที่บ้านผู้เช่า โดยมีรูปร่างเป็นแบบ wall mounted box ซึ่งจะมีฟังก์ชั่นการทำงานที่จะต่อร่วม ช่องสัญญาณ B1+B2+D0 ระหว่างจุดปลายของผู้เช่ากับจุดปลายของโครงข่าย ISDN เข้าด้วยกัน



รูปที่ 2.1 แสดงการเชื่อมต่อแบบ Basic Access

พิจารณาจุดอินเตอร์เฟสทางด้าน S

จะมีคุณลักษณะที่ใช้ในการต่อร่วมกับอุปกรณ์ของผู้เช่าดังนี้

- จะใช้สัญญาณที่มีรหัสที่คล้ายคลึงกับ AMI-CODE
- เป็นระบบสายส่งแบบ 4-wire ที่สามารถต่อร่วมกับอุปกรณ์ของผู้เช่าได้สูงสุด 8 อุปกรณ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

อุปกรณ์ NT จะมีแหล่งจ่ายไฟเป็นของตนเองที่จะใช้เลี้ยงเครื่องโทรศัพท์แบบดิจิตอล โดยการต่อร่วมที่จุดอินเตอร์เฟส S ในการทำงานช่วงฉุกเฉิน เช่น แหล่งจ่ายไฟเกิดเหตุขัดข้อง เครื่องชุมสายแบบดิจิตอลจะต้องส่งไฟมาเลี้ยงอุปกรณ์โทรศัพท์แทน เพื่อเป็นการรับประกันว่าอย่างน้อยที่สุดอุปกรณ์ของผู้เช่าก็ยังมีเครื่องโทรศัพท์ที่ยังใช้งานได้อยู่ 1 เครื่อง

ทั้งอุปกรณ์ Line Termination ที่ติดตั้งที่ชุมสายกับ อุปกรณ์ Network Termination จะต้องสามารถใช้งานร่วมกันอย่างไม่มีปัญหา แม้ว่าอุปกรณ์ทั้ง 2 จะถูกผลิตมาจากคนละบริษัทหรือ เทคโนโลยีในการผลิตต่างกัน ก็ต้องสามารถใช้งานร่วมกันได้

เพื่อให้แน่ใจว่าอุปกรณ์ทั้ง 2 จะต่อร่วมกันได้เสมอ หน่วยงานที่ให้บริการ ISDN จะต้องกำหนดพารามิเตอร์ต่างๆ ลงในรายละเอียด เพื่อที่ผู้จำหน่ายอุปกรณ์หรือผู้ผลิตอุปกรณ์จะใช้เป็นแนวทาง

การต่ออุปกรณ์ TE ที่จุดอ้างอิง S

CCITT ได้กำหนดชนิดของการต่ออุปกรณ์ TE ที่จุดอ้างอิง S ออกเป็น 2 ลักษณะคือ

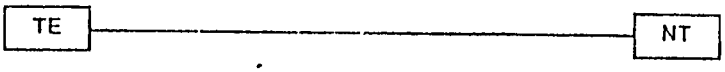
1. แบบจุดต่อจุด (point to point) สายสัญญาณที่จุดอ้างอิง S ต่อเข้ากับอุปกรณ์ TE (Terminal Equipment) 1 เครื่อง

2. แบบระบบบัส (passive bus) สายสัญญาณที่จุดอ้างอิง S ต่อเข้ากับอุปกรณ์ TE ได้ถึง 8 เครื่อง

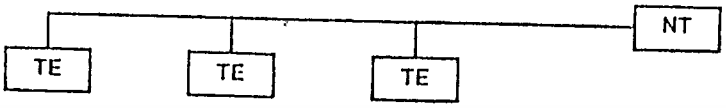
การต่อแบบจุดต่อจุด ความยาวของสายเคเบิลที่ใช้ต่อระหว่าง ISDN Terminal กับ subcriber socket ต้องไม่ทำให้เกิดการลดทอนของสัญญาณเกินกว่า 6 dB (maximam attenuation) ที่บริเวณแถบความถี่สเปคตรัมที่ 96 กิโลเฮิร์ต (ช่วงแถบพลังงานส่วนใหญ่จะอยู่แถวบริเวณครึ่งหนึ่งของความถี่ 192 กิโลเฮิร์ต) ดังนั้นความยาวของสายเคเบิลจะมีค่าประมาณ 750 ถึง 1500 เมตร โดยประมาณ

การต่อแบบระบบบัส (passive bus) สาย I.410 bus สามารถต่อเข้ากับอุปกรณ์ ISDN terminal ได้ถึง 8 เครื่อง แต่ I.410 bus จะมีช่องสัญญาณ (channel) เพียง 3 ช่องสัญญาณ คือ B channel ที่มีความเร็ว 64 กิโลบิทต่อวินาที 2 ช่องสัญญาณ และ D channel ที่มีความเร็ว 16 กิโลบิทต่อวินาที อีก 1 ช่องสัญญาณ นอกจากช่องสัญญาณทั้ง 3 แล้ว I.410 bus ยังมีการใส่ Synchronization bits และ Control bits เพิ่มเติมเข้าไปอีก จนกระทั่งได้ความเร็วของสัญญาณของ I.410 bus เป็น 192 กิโลบิทต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.2 Point-to-point BRI configuration (ได้ถึง 3000 ฟุต)



รูปที่ 2.3 Point-to-Multipoint BRI configuration โดยใช้ short passive bus (ได้ถึง 500 ฟุต)

รูปที่ 2.4 Point-to-multipoint BRI configuration โดยใช้ extended passive bus (ได้ถึง 3300 ฟุต)

อุปกรณ์ ISDN terminal ไม่สามารถที่จะเข้าถึง B channel ได้โดยตรง การ call จะต้องเริ่มต้นบน D channel ก่อนเสมอ เลขหมายของผู้รับปลายทางและเลขหมายประจำตัวของ ISDN terminal จะถูกส่งโดยการคีย์แป้น key set ก่อนที่อุปกรณ์ ISDN terminal จะส่ง information ดังกล่าว อุปกรณ์จะต้องทำการตรวจสอบ D channel ก่อนว่าว่างหรือไม่ ถ้าว่างหลังจากนี้ก็จะส่ง information (เลขหมายของผู้รับปลายทางและเลขหมายประจำตัวเครื่อง) ลงในช่องสัญญาณ D channel ด้วยความเร็ว 16 กิโลบิทต่อวินาที แต่ละบิทจะถูกส่งในจังหวะที่แน่นอนเพื่อให้พอดีกับตำแหน่งใน D channel ถ้าข่าวสาร "CALL message" ที่ส่งออกมีการกำหนดความจำเป็นที่ต้องใช้ B channel ในการติดต่อ สัญญาณตอบกลับบน D channel จะกำหนดให้ว่าจะใช้ B channel อันไหนได้ หลังจากนั้นอุปกรณ์ ISDN terminal จึงจะต่อเข้ากับ B channel นั้น

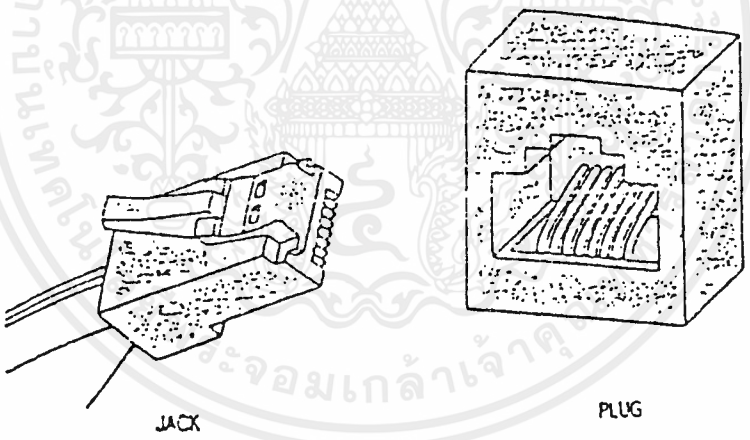
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ถ้าการ CALL คือการเรียกใช้โทรศัพท์ และ B channel ที่กำหนดให้ใช้ได้คือ B1 ดังนั้นเครื่องโทรศัพท์ (hand set) จะถูกต่อเข้ากับ B1 channel สัญญาณเสียง 8 บิทก็จะถูกรับส่งด้วยความเร็ว 8,000 ครั้งต่อวินาที

I.410 bus จะเป็นระบบบัสแบบ 4 wire bus มีความเร็วของสัญญาณ 192 กิโลบิทต่อวินาที และวิธีการรับส่งสัญญาณจะใช้วิธีการเข้ารหัสแบบ AMI code (Alternate Mark Inversion)

นอกจากนี้ยังมีการตรวจสอบ Collision check อย่างอัตโนมัติอีกด้วย เพื่อป้องกัน D channal ถูกใช้งานมากกว่า 1 termal ในเวลาเดียวกัน

Connector ที่ใช้ที่จุดอ้างอิง S คือ Plug กับ Socket ตามมาตรฐานของ ISO/DIS 8877 international standard project ส่วนสายเคเบิลที่ใช้สำหรับการติดตั้งเป็นสายเคเบิลธรรมดา เบอร์ 278 series ซึ่งมีตัวนำที่มีขนาดเส้นผ่าศูนย์กลาง 4 มิลลิเมตร อยู่ 4 คู่สาย

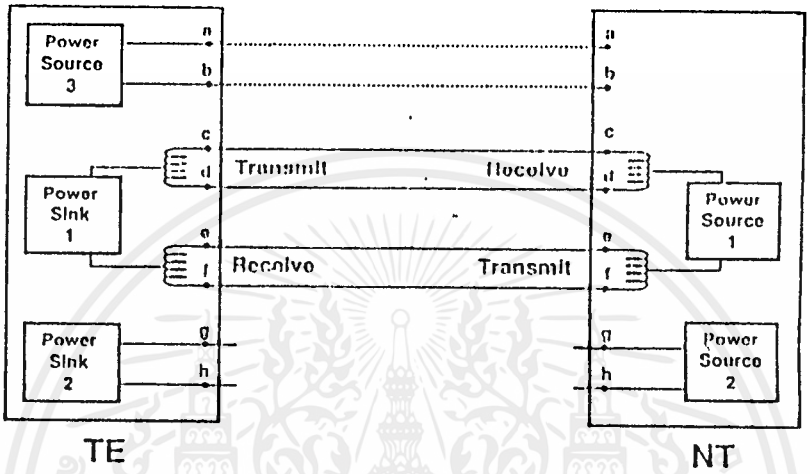


รูปที่ 2.5 S interface connectors

เรามีอุปกรณ์ ISDN terminal หลายชนิดที่สามารถต่อเข้ากับสายไฟ 220 Volt ได้โดยตรงโดยที่ไม่จำเป็นต้องใช้ไฟ supply ที่ส่งมาจาก NT แต่อย่างไรก็ดี สำหรับ ISDN terminal ที่มีฟังก์ชั่นทางด้านการเรียกใช้โทรศัพท์ด้วย จะใช้ไฟ supply ที่ส่งมาจาก NT ซึ่งหมายความว่า ในกรณีแหล่งจ่ายไฟ 220 volt (ไฟบ้าน) เกิดขัดข้องก็ยังคงแน่ใจว่ายังสามารถที่จะใช้ฟังก์ชั่นที่เกี่ยวกับโทรศัพท์ได้ตลอดเวลา ไฟ supply จะถูกจ่ายมายัง ISDN terminal ด้วย 4 wire passive bus โดยใช้เทคนิคที่เรียกว่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Phantom method ดังคู่สายที่แสดงในรูปต่อไปนี้ อุปกรณ์ NT จะถูกต่อเข้ากับไฟ 220 volt แต่อย่างไรก็ดี มันจะมีแบตเตอรี่ backup อยู่ภายในด้วย



รูปที่ 2.6 Supply แบบ Phantom method

พิจารณาจุดอินเตอร์เฟสทางด้าน U

จะมีคุณลักษณะที่ใช้ในการต่อร่วมกับโครงข่าย ISDN โดยเข้าสู่ระบบสื่อสัญญาณไปยังเครื่องชุมสาย ดังนี้

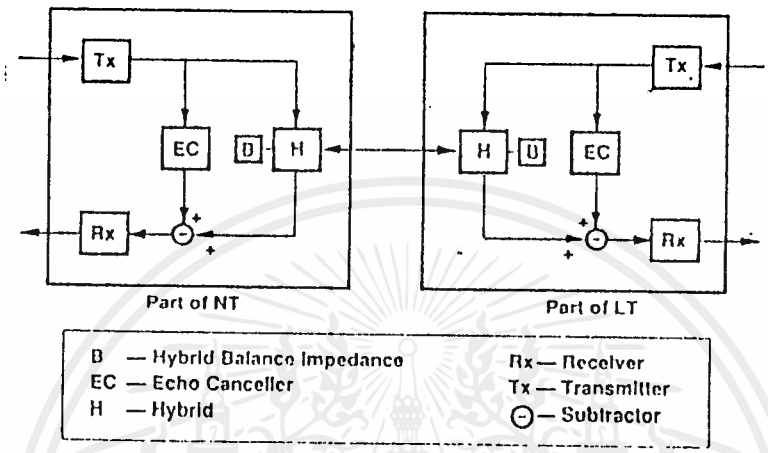
- จะใช้สัญญาณที่มีรหัสเป็น 4B/3T code หรืออาจจะเป็น Differential Biphase code

- มีการป้องกันสัญญาณสะท้อนกลับทาง Transmitter ที่จะไปรบกวนสัญญาณทางด้าน Receiver เมื่อต้องการแปลงจาก 4-wire ไปเป็น 2-wire ซึ่งวิธีการป้องกันดังกล่าว เราเรียกว่า Echo Cancellation Method

การทำงานของชุมสายจะใช้สัญญาณที่มีสายส่งเป็นแบบ 4-wire แต่ในเขตเดียวกันเครื่องปลายผู้เช่าจะใช้สายสัญญาณเป็นแบบ 2-wire CCITT ไม่ได้ออกข้อ

เอกสารนี้เป็น... ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ก็หลายวิธีการที่จะแก้ไขปัญหานี้ ดังนั้นในทางปฏิบัติ จึงเป็นหน้าที่ของผู้ใช้เองว่าจะเลือกใช้วิธีการใด จึงจะเหมาะสมในการแก้ปัญหานี้ วิธีหนึ่งที่นิยมใช้ในทางปฏิบัติและได้ผลดี ก็คือ Adaptive Digital Hybrid Method ดังแสดงในรูป 2.7



รูปที่ 2.7 The Echo Cancellation

2.1 โครงสร้างทดงเฟรม

บุนสาย ISDN รับข้อมูลดิจิตอลของผู้ใช้ โดยดูจากโครงสร้างเฟรมของเบสิกแอกเซสและไพรมารี่เรตแอกเซส จากนั้นจึงจ่ายข้อมูลให้กับระบบสื่อสัญญาณในเครือข่าย ISDN ข้อมูลที่ปรากฏตามจุดต่างๆ ของการอินเตอร์เฟส (จุด S หรือ T) จึงมีโครงสร้างของเฟรมที่แตกต่างกัน ซึ่งได้แก่โครงสร้างโฟรมสำหรับเบสิกแอกเซสและโครงสร้างเฟรมของไพมารี่ เรตแอกเซส

2.1.1 โครงสร้างเฟรมของเบสิกแอกเซส

ที่จุดอ้างอิง S (หรือ T) 1 เฟรมประกอบด้วยข้อมูล 48 บิต ใน เวลา 250

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไมโครวินาที ดังนั้นอัตราเร็วของ 1 เฟรม จึงเท่ากับ 192 กิโลบิต/วินาที ดังรูปที่ 2.8 จากบิตทั้งหมด 48 บิต มี 32 บิตที่เป็นแชนแนล B (อย่างละ 16 บิต) และแชนแนล D อีก 4 บิต ที่เหลืออีก 12 บิตมีชื่อและหน้าที่ดังนี้คือ

1. บิต F หรือ เฟรมไทม์มิ่งบิต เป็นบิตที่บอกการเริ่มต้นของเฟรม

2. บิต L หรือ DC Balancing Bit ใช้ในการกำจัดองค์ประกอบทางด้านไฟตรงให้หมดไป บิต L จะเป็น "0" ถ้าหากจำนวนบิต "0" ที่นับต่อจากบิต L ตัวสุดท้ายเป็นจำนวนคี่ ดังนั้นผลรวมของบิต "0" ทั้งหมด จึงเป็นคู่ องค์ประกอบทางด้านไฟตรงจึงไม่มี อีกกรณีหนึ่งถ้าหากจำนวนบิต "0" ที่นับต่อจากบิต L ตัวสุดท้ายเป็นจำนวนคู่ บิต L จะมีค่าเป็น "1" ซึ่งสัญญาณในสายเป็นศูนย์โวลต์ ดังนั้นองค์ประกอบของไฟตรงจึงไม่มีเช่นกัน ตำแหน่งของบิต L ดูได้จากรูปที่ 2.8

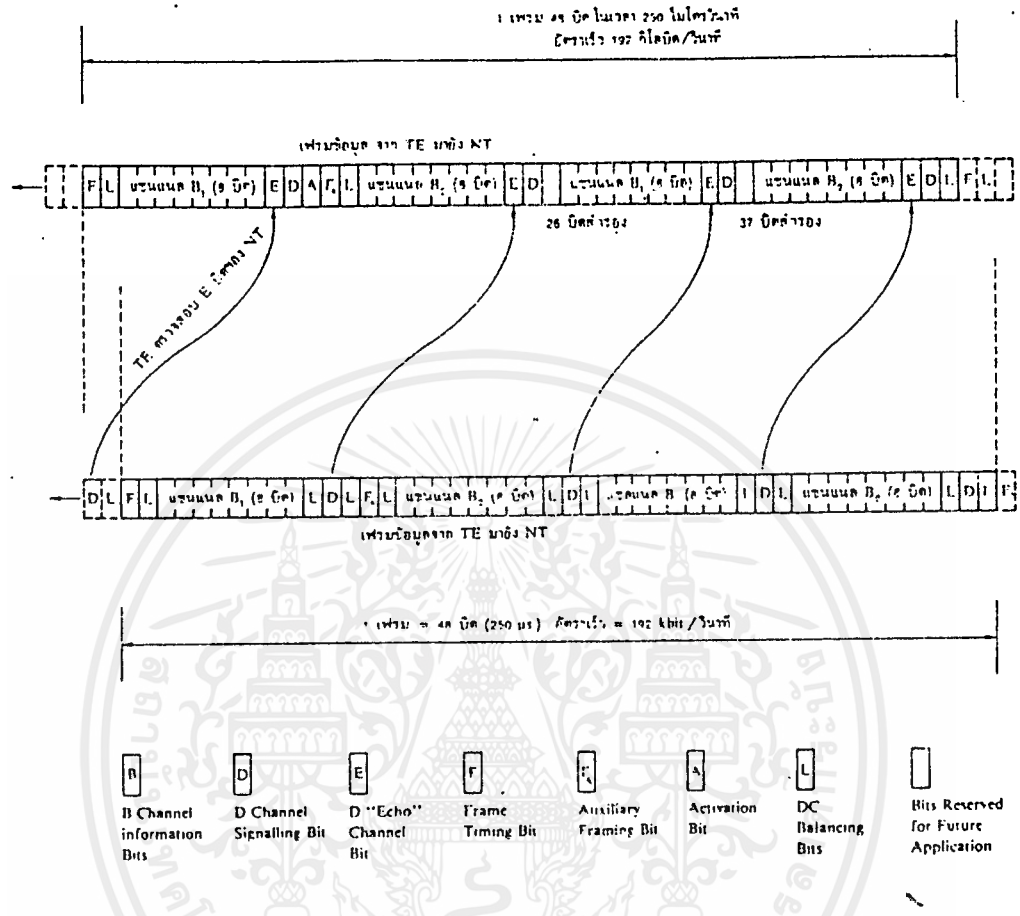
3. บิต E หรือ D'Echo channel bit มีทั้งหมด 4 บิต เป็นบิตที่สะท้อนการทำงานของบิต D ตัวสุดท้ายที่ส่งสัญญาณให้ NT ดังรูปที่ จะเห็นว่าสัญญาณการทำงานของบิต D ของเฟรมที่ส่งจาก TE ไปยัง NT ไปปรากฏอยู่ในบิต E ของเฟรมที่ส่งจาก NT ไปยัง TE มีบิต E เฉพาะในเฟรมที่ส่งจาก NT มายัง TE โดยอยู่ในตำแหน่งที่ 11 24 35 และ 46

4. บิต A หรือ Activation บิต เป็นบิตที่ถูกควบคุมจากชุมสาย เพื่อกำหนดให้เทอร์มินอลทำงานในโหมด power on ในขณะติดต่อกับชุมสายและโหมด power down ในขณะที่เทอร์มินอลไม่ได้ทำงาน บิต A อยู่ในตำแหน่งที่ 13 ของเฟรมที่ส่งมาจากNT

5. บิต FA เป็นบิตช่วยของเฟรมมิ่งบิต (บิต F)

6. มีบิตอีก 2 บิตที่สำรองไว้ใช้ในอนาคต ที่ตำแหน่ง 26 และ 37 ผลรวมของอัตราเร็วที่สามารถใช้งานได้ คำนวณจากบิตที่ใช้สำหรับแชนแนล B จำนวน 32 บิต และแชนแนล D อีก 4 บิต รวม 36 บิต หารด้วยเวลาที่ใช้ในแต่ละเฟรมคือ 250 ไมโครวินาที จะได้ 144 กิโลบิต/วินาที ซึ่งเท่ากับ 2B + D

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.8 แสดงโครงสร้างของเฟรมที่ส่งจาก NT ไปยัง TE และ TE ไปยัง NT

สาเหตุที่เฟรมของเมสิก แตกเซสที่ส่งจาก NT มายัง TE กับ TE มายัง NT แตกต่างกัน เนื่องมาจาก NT 1 ตัวต้องคอยควบคุม และบริหาร TE หลาย ๆ ตัว ในหน้าที่การทำงานของ TE ขึ้นอยู่กับผู้ใช้ จึงเกิดความไม่สมมาตรกัน ทำให้เฟรมสำหรับส่งจาก TE ไป NT เลื่อนช้ากว่าออกไป 2 บิต ดังรูปที่ 2.8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

DATA LINK LAYER

ใน Layer 2 ของ D-channel protocol ได้ออกแบบ protocol ที่ทำหน้าที่บน Layer นี้คือ LAPD ซึ่งย่อมาจาก Link Access Procedure on D Channel

จุดประสงค์ของ Layer 2 หรือ protocol LAPD นี้คือ การเคลื่อนย้ายข่าวสารโดยต้องปราศจากข้อผิดพลาดในการสื่อสารซึ่งหน้าที่หลักของ LAPD จะประกอบไปด้วยดังนี้

- Framing: การจัดการเรื่องสัญญาณเริ่มต้นและจบของการส่ง และการจำกัดข้อมูลของผู้ใช้
- Addressing: การบ่งชี้อุปกรณ์บนสายสื่อสารว่าเป็นตัวส่งหรือตัวรับ frame
- Sequencing: การรักษาขั้นตอนตามลำดับของการส่ง frame ข้อมูล
- Acknowledgment: การยอมรับ frame ข้อมูล
- Time outs: การจัดการในเรื่องเวลาของการตอบรับต่างๆ
- Error control: การตรวจสอบข้อผิดพลาดต่างๆ เช่น การส่งข้อมูลที่ผิดพลาด รูปแบบของข้อมูลที่ส่งมาผิดพลาด และการแก้ไขหรือกู้ข้อมูลที่ผิดพลาดกลับคืนมา
- Flow control: การจัดการในด้าน mechanism ให้มีการส่งข้อมูลที่รวดเร็ว

ลักษณะของ protocol LAPD นี้จะคล้ายคลึงกับ X.25 (LAPB) ซึ่งทั้ง 2 ชนิดนี้ก็เป็น subset ของ ISO High-level Data Link Control (HDLC) protocol แต่ทั้ง 2 ชนิดนี้ก็มีความแตกต่างกันอยู่บ้าง เช่น การอ้าง address ของ LAPD เป็นต้น

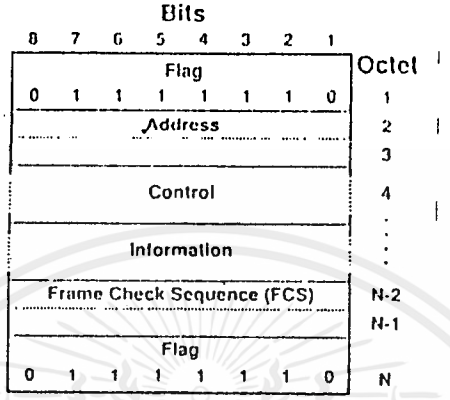
protocol LAPD นี้จะใช้ในการเชื่อมต่อระหว่าง user(TE) และ network (NT or LE) หรือที่จุดต่ออ้างอิง S/or T บน D-channel โดยสามารถใช้ได้กับการต่อแบบขนาน , synchronous ,full-duplex ระหว่าง การเชื่อมต่อแบบ point-to-point หรือ point-to-multipoint

หลักทั่วๆ ไปของ LAPD ได้ถูกอธิบายไว้ใน CCITT Recommendation Q.920 (I.440) และที่แสดงขั้นตอนการทำงานโดยละเอียดไว้ใน CCITT Recommendation Q.921 (I.441)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 LAPD Frames

ส่วนประกอบของ เฟรมในโปรโตคอล LAPD มีรูปแบบเป็นฟิลด์ ดังนี้



รูปที่ 3.1 แสดง LAPD frame format

1. Flag : เป็นชุดของบิตฐาน 2 ในรูปแบบ 01111110 ใช้ในจุดเริ่มต้นและลงท้ายของเฟรม

2. Address : เป็นตัวบ่งชี้อุปกรณ์ของผู้ใช้และบริการ ที่ต้องการส่งหรือรับจะมีความยาว 2 octets หรือ 16 บิท เสมอ

3. Control : เป็นตัวบ่งชี้ชนิดของเฟรม ซึ่งอาจจะมีความยาวได้ทั้ง 1 หรือ 2 octets ขึ้นอยู่กับชนิดของเฟรม

4. Information : ฟิลด์นี้จะมีความยาวเท่าใดก็ได้ และจะมีข่าวสารใดๆ ก็ได้แต่ต้องอยู่ในรูปแบบของ octet และฟิลด์นี้อาจจะไม่มีในเฟรมก็ได้

5. Frame check sequence (FCS) : มีทั้งหมด 16 บิทใช้ในการตรวจสอบความผิดพลาดของบิทในเฟรม โดยใช้วิธีการของ Cyclic Redundancy Check (CRC)

เนื่องจากตัวข่าวสารที่นำมาส่ง เองอาจมีรูปแบบของบิทเหมือนของแฟลก (01111110) ก็ได้ ฉะนั้นในการส่งจะต้องแน่ใจว่าข่าวสารที่ส่งมาตามสายนั้น จะไม่มีรูปแบบดังกล่าว ซึ่งสามารถกระทำได้ โดยเทคนิคที่เรียกว่า การแทรกเลข 0 (Zero

เอกสารนี้เป็นส่วนหนึ่งของวิทยานิพนธ์ ที่ได้รับอนุญาตให้นำขึ้นเผยแพร่ในเครือข่ายอินเทอร์เน็ตเพื่อการศึกษาเท่านั้น ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

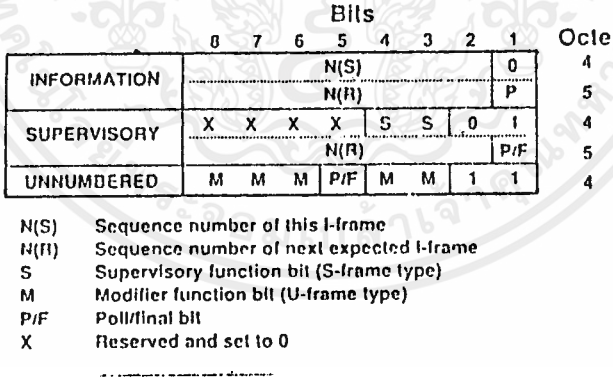
Insertion) โดยตัวควบคุมการสื่อสารของทางด้านส่งจะทำการตรวจสอบบิทที่อยู่ระหว่างแฟลกถ้าหากมี 1 เข้ามาติดต่อกัน 5 ตัวก็จะเพิ่ม 0 เข้าไปหลังจากบิท 1 บิทที่ 5 เพื่อป้องกันไม่ให้มี 1 มาติดต่อกัน 6 ตัว ซึ่งเป็นรูปแบบของแฟลกที่จะทำให้เกิดความผิดพลาด

ยกตัวอย่างวิธีการแทรกเลข 0 เมื่อคอมพิวเตอร์ส่งข่าวสารที่มีรูปแบบ 111111 เข้ามาข่าวสารที่ปรากฏบนช่วงต่อจะเป็น 1111101 และเช่นเดียวกันหากข่าวสารมีค่าเป็น 1111101 ตัวควบคุมการสื่อสารจะเพิ่ม 0 อีก 1 ตัวเป็น 11111001 แล้วส่งออกไป

สำหรับทางด้านรับนั้น ถ้าตัวควบคุมการสื่อสาร รับข่าวสารมามี 1 รวม 6 ตัว ก็จะแปลเป็นแฟลก แต่ถ้าได้รับ 1 ติดต่อกัน 5 ตัวตามด้วย 0 ก็จะตัด 0 ตัวนั้นออกไป ทำให้ได้ข้อมูลชุดเดียวกับที่ออกจากทางด้านส่ง ยกตัวอย่างหากข้อมูลที่เข้ามาคือ 1111101 ก็จะได้เอาท์พุทออกมาคือ 111111 หรือหากอินพุทคือ 11111001 เอาท์พุทที่ออกคือ 1111101

ฟิลด์ Control และชนิดของ LAPD Frame

Control Field จะบ่งบอกชนิดของเฟรมที่จะส่งออกไป ซึ่งแบ่งออกโดยทั่วๆ ไปได้ 3 แบบดังรูปที่ 3.2



รูปที่ 3.2 แสดง LAPD Control field format

1. Information (I) frames : บรรจุข่าวสารระดับ 2 หรือ ข้อมูลจาก layer ที่สูงกว่า รวมทั้งข้อมูลของผู้ใช้ มีความยาว 2 octet บรรจุ N(S) เป็นตัวบอกจำนวนลำดับของเฟรม I และ N(R) เป็นตัวบอกจำนวนลำดับของเฟรม I ที่คาดว่าจะได้รับต่อไปจากการส่งขนส่าย

เอกสารนี้ใช้เพื่อการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2. Supervisory (S) frames : เฟรมนี้ใช้สำหรับการควบคุมการไหลของข้อมูล โดยการสร้างการตอบรับการรับเฟรม I มีความยาว 2 octet บรรจุตัวบอกจำนวนลำดับของเฟรม I ที่คาดว่าจะได้รับต่อไปคือ N(R) และบิท SS จะบ่งบอกถึง ชนิดของเฟรม I

3. Unnumbered (U)frames : ใช้ควบคุมสถานะของการเชื่อมทาง Logic และการแลกเปลี่ยนข้อมูลที่ไม่มีหมายเลขหรือไม่มีลำดับ โดยการใช้ parameter และบ่งบอกเงื่อนไขความผิดพลาด มีความยาว 1 octet และ บิท MMMMM ใช้สำหรับบอกชนิดของเฟรม U

ตารางต่อไปนี้แสดงแบบของเฟรม I,S,U ทั้งหมดที่ใช้ใน LAPD และมีคำอธิบายหน้าที่ต่างๆ อย่างคร่าวๆ และจะเห็นว่ารูปแบบของเฟรมอาจจะเป็น Command, Responses หรือบางทีอาจเป็นได้ทั้ง 2 อย่าง

Frame Type	Frame Name		C	R	Function
I	I	Information	•		Transfer Layer 3 data
S	RR	Receive Ready	•	•	Acknowledges receipt of previously transmitted I-frames, clears a busy condition signaled by an RNR, and indicates willingness to receive more I-frames
	RNR	Receive Not Ready	•	•	Used for flow control, indicating a temporary inability by the receiver to receive I-frames (a busy condition)
	REJ	Reject	•	•	Requests retransmission of I-frames and clears the RNR busy condition; sent by receiver when an out-of-sequence I-frame is received
U	SABME	Set Asynchronous Balanced Mode Extended	•		Start the logical link for acknowledged information transfer using modulo 128 sequencing (i.e., frames can take on sequence numbers in the range 0 to 127).
	DISC	Disconnect	•		Terminate logical link for acknowledged information transfer
	UI	Unnumbered Information	•		Transfer Layer 3 or link management information using unacknowledged information transfer (i.e., no sequencing numbers)
	UA	Unnumbered Acknowledgement		•	Acknowledges SABME and DISC
	DM	Disconnected Mode		•	Indicates that the logical link has some sort of error and that the station sending the DM cannot continue with information transfer
	FRMR	Frame Reject		•	Reports an error condition that cannot be cleared by the retransmission of an identical frame that caused the error
	XID	Exchange Identification	•	•	Used for automatic data link layer parameter negotiation

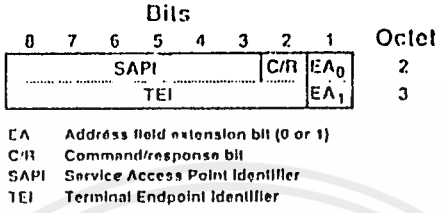
C: Command R: Response

ตารางที่ 3.1 แสดงแบบต่างๆ ของเฟรม LAPD

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การอ้าง Address ของ LAPD

ลักษณะที่สำคัญอย่างหนึ่งของ LAPD ก็คือ โครงสร้างของฟิลด์ Address ดังแสดงในรูปที่ 3.3



รูปที่ 3.3 แสดง LAPD Address field format

การอ้างแอดเดรสเรียกว่า Data Link Control Identifier ซึ่งจะมีความยาว 13 บิท และประกอบด้วยฟิลด์ย่อย ที่เรียกว่า Subfields อีก 2 Subfields คือ Terminal Endpoint Identifier (TEI) และ Service Access Point Identifier (SAPI)

บิท Command/Response (C/R) ในฟิลด์ของ Address ใช้ในขอกแบบของเฟรมเหมือนดังตารางที่ 3.1 โดยทางด้านผู้ใช้ (User) จะตั้งค่าบิทนี้เป็น 0 หมายถึง Command และบิท 1 หมายถึง Response ซึ่งจะตรงกันข้ามกับทางด้านโครงข่าย โดยจะตั้งบิทนี้เป็น 0 หมายถึง Response และบิท 1 หมายถึง Command

สำหรับบิทลำดับแรกสุด เรียกว่าบิท Address field extension โดยที่ octet แรกของ Address field บิทที่ 1 จะตั้งให้เป็น 0 และบิทที่ 1 ของ octet ที่ 2 จะตั้งให้เป็น 1

3.3 ลักษณะของ SAPI

SAPI มีความยาว 6 บิท อยู่ใน octet แรกของฟิลด์ Address เพราะฉะนั้นมันจึงมีได้ถึง 64 แบบ แต่ CCITT ก็ได้กำหนดไว้เพียง 4 แบบดังตารางที่ 3.2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

SAPI Value	Related Level 3 Entity
0	Call Control procedures
1	Reserved for packet-mode communications using I.451 call control procedures
16	Packet communication using X.25 PLP
32-47	(reserved for national use)
63	OAM,Level 2 management fuctions
others	(reserved for future standardization)

ตารางที่ 3.2 แสดง Service Access Point Identifier (SAPI)

3.A ลักษณะของ TEI

TEI มีความยาว 7 บิท อยู่ใน octet ที่ 2 ของฟิลด์ Address ดังนั้นจึงกำหนดได้ถึง 127 แบบ ซึ่งก็มากเกินพอสำหรับการอ้างอุปกรณ์ ISDN

TEI มี 3 แบบดังนี้

- Nonautomatic TEI :

ได้ถูกกำหนดให้กับอุปกรณ์ terminal ที่ไม่สามารถใช้โปรโตคอล LAPD ในการร้องขอ TEI จากโครงข่ายได้ นั่นยังรวมถึงอุปกรณ์ที่ TEI ถูกโปรแกรมไปยัง ROM หรือที่ๆ ผู้ใช้จะเลือก TEI เป็นตัวข้ามไปหรือตัวสวิตช์

- Automatic TEI :

ถูกกำหนดให้กับอุปกรณ์ terminal ที่สามารถใช้โปรโตคอล LAPD ได้อย่างอัตโนมัติ

- Group address (broadcast) :

TEI แบบนี้จะอ้างแอดเดรสไปยัง ISDN terminal ทุกตัวที่เชื่อมต่ออยู่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

TEI Value	User Type
0-63	Non-automatic TEI assignment user equipment
64-126	Automatic TEI assignment user equipment
127	Group assignment (broadcast) TEI

ตารางที่ 3.3 แสดง Teminal Endpoint Identifier (TEI) Values

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

NETWORK LAYER

ใน ISDN ระดับ Network layer นั้นประกอบด้วย recommendation ต่าง ๆ หลายอัน Recommendation Q.930 (I.450) อธิบายหลักการทั่วไปของ user-networking signal, Recommendation Q.931 (I.451) อธิบายเกี่ยวกับ user-network messages สำหรับ basic call control และ recommenddation Q.932 (I.452) อธิบาย messages สำหรับ supplementary services Protocols เหล่านี้ประกอบขึ้นเป็น Digital Subscriber Signaling system No.1 (DSS1) Network layer

4.1 Layer 3 USER-NETWORK Signaling

คำว่า *user-network* นั้นบอกเราถึงขั้นตอนที่ใช้ในการเชื่อมต่อระหว่าง ISDN terminal ของ user กับ serving ISDN ISDN terminal ในที่นี้ก็ คือ terminal ที่มีความสามารถในการใช้ procedure ต่าง ๆ ที่จะกล่าวต่อไปนี้ได้ อาจเป็น PBX หรือ concentrator หรือ computer ก็ได้

User-network interface procedures นั้นเชื่อมต่อระหว่าง ISDN user และ serving network เท่านั้น ต่างกับ protocol ที่ใช้ใน switching node คือ Signaling System No. 7 และ user-network protocol นี้จะไม่ขยายข้าม network ข้อมูลที่จะส่งผ่าน network นั้นจะต้องถูกอ่านโดยใช้ ISDN protocol และ mapped ไปยัง SS7 protocol ส่วนทางด้านรับอาจมีการ mapp ย้อนกลับที่ ISDN exchange ของจุดหมายทางด้านรับ

Basic call control ตามความหมายนั้นหมายถึง การ established , maintained และ disconnected communication อย่างง่าย ๆ ซึ่งจากความหมายนั้น คือการบอกว่ามี transmission path เพียง path เดียวที่ถูกสร้างขึ้นจาก user ข้าม network ไปยังอีก user Basic call control ได้กล่าวไว้ใน reccomendation Q.931 (I.451)

Supplementary Services เป็นคำที่มีความหมายรวม ๆ ของ features

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ในเชิงพาณิชย์ หรือในเชิงการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ต่าง ๆ ที่ทำให้ users สามารถควบคุมการ,จัดการ transmission path ของ network ได้มากขึ้น เช่น call forwarding,call waiting ฯลฯ

Signaling procedures เป็นขบวนการที่ใช้โดย user's terminal และ network ในการส่งสัญญาณระหว่างกัน

สำหรับ signaling procedures ที่ใช้ติดต่อระหว่าง users กับ user's terminal นั้นอยู่นอกเหนือขอบเขตที่จะกล่าว แต่จะเป็นหน้าที่ที่ผู้ผลิตแต่ละรายจะสร้างสรรค์วิธีการของตนขึ้นมาเอง เช่น การบอกว่ามีโทรศัพท์มา อาจใช้กระดิ่งธรรมดา หรืออาจใช้เสียงพูดที่สังเคราะห์ขึ้นโดยเครื่องก็ได้

4.2 Structure of the Protocols

ISDN Signaling protocols ใช้หลักในการติดต่อโดยการส่งข้อมูล digital ที่เรียกว่า Message ระหว่าง user's terminal กับ network แต่ละ message ที่ส่งไปนั้นจะประกอบไปด้วย informaition elements หลายตัว

ISDN มี signaling messages อยู่ทั้งหมด 33 messages โดยกำหนดใน Q.931 25 messages และอีก 8 messages กำหนดไว้ใน Q.932 แต่ละ message นั้นจะบอกถึงข้อมูลที่จำเป็นในการ call หรือ service ในขณะนั้น แต่ละ message จะมีจุดประสงค์ในการทำงานต่างกันไปตามชื่อของมัน เช่น SETUP,CONNECT เป็นต้น

แต่ละ message จะประกอบไปด้วย header และ informaiton elements Headers ประกอบด้วย 3 ส่วนคือ Protocol Discriminator บอกว่า message นั้นเป็นของ protocol อะไร,Call Reference Value (CRV) ใช้กำหนด specific call,Message type บอกว่าเป็น messages ชนิดไหนใน 33 ชนิด

Information elements จะอยู่ในลำดับหลังจาก header และชื่อของมันจะบอกถึงจุดประสงค์ของมัน เช่น Called party number,Packet size ฯลฯ

Information element หนึ่ง ๆ จะประกอบไปด้วย fields หลาย ๆ fields และเช่นเดียวกัน ชื่อของแต่ละ field จะบอกชนิดของข้อมูลที่มันเก็บอยู่ เช่น Information transfer capability field ใช้กำหนดว่าการ call นั้นเป็นการส่ง speech,digital information หรือ video

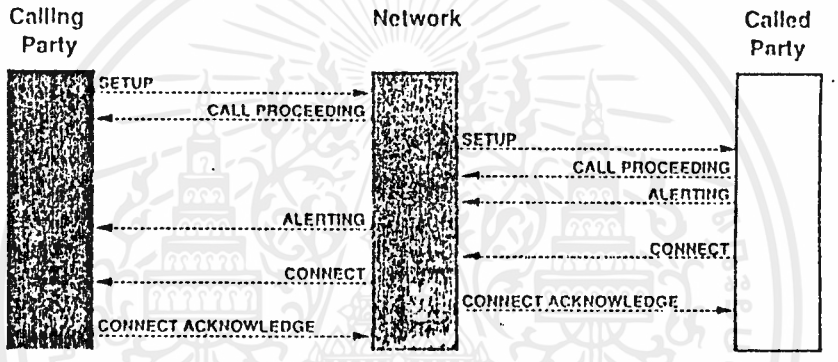
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

A.3 Basic Circuit-Mode calls

Basic circiut mode call คือการติดต่อระหว่าง user's terminal และ network ที่ให้ user ครอบครอง channel ตลอดเวลา

Conneting the Call

รูปการส่ง messages เพื่อทำการติดต่อแบบ circuit-mode แสดงดังรูปที่ 4.1



รูปที่ 4.1 Call establishment message exchange

ฝ่ายส่งจะส่ง message SETUP ไปยัง network ใน SETUP message นั้นจะบอกถึงข้อมูลที่จำเป็นในการติดต่อ

ใน SETUP message มี information elements หลายตัวที่สำคัญ Bearer capability information element ใช้กำหนด attributes ของ service ที่ user ต้องการจาก network เช่น บอกว่าเป็นการส่ง packet mode หรือ circuit mode, ส่งเสียงพูดหรือ ข้อมูลหรือ video ฯลฯ

Called party number information element ให้ข้อมูลเกี่ยวกับหมายเลขของฝ่ายรับ ประกอบด้วย 3 fields คือ Type of number บอกชนิดของหมายเลขว่าเป็นระหว่างประเทศ,ภายในประเทศ หรือ numbering plan ที่ใช้กำหนด network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Numbering plan identification กำหนดว่าหมายเลขนั้นเป็นของ numbering plan ใด และ number digits field คือหมายเลขที่ใช้แทน address ของผู้รับ

Information elements อื่น ๆ ใน SETUP message คือ Channel identification information element ใช้กำหนด bearer channel ว่าจะใช้ channel ใดในการติดต่อ

ใน SETUP message นั้น user สามารถส่ง address ของผู้เรียกไปยังผู้รับได้ใน 2 modes คือ *enbloc* คือส่งไปทั้งหมดภายใน message เดียว และ *overlap* คือส่ง address ไปในหลาย ๆ messages ในที่นี้สมมติว่าเป็นแบบ enbloc

เมื่อ network รับ enbloc SETUP message แล้วก็จะตรวจสอบว่า message นั้น valid,user มีอำนาจในการใช้ bearer capability ฯลฯ เมื่อพบว่าถูกต้องก็จะส่ง CALL PROCEEDING เพื่อบอกว่าการ call นั้นถูกต้องและกำลังดำเนินการอยู่

หลังจากนั้น network ก็จะส่ง SETUP message ไปยังฝ่ายรับซึ่ง SETUP message นี้ก็จะคล้ายกับที่ได้รับมาจากฝ่ายส่ง

จากนั้นในบางกรณีฝ่ายรับจะส่ง CALL PROCEEDING message กลับไปให้ network ซึ่งใช้มากกับอุปกรณ์ multiplex เช่น PBX เพื่อบอกว่าได้รับสัญญาณแล้วและกำลังดำเนินการอยู่ ปกติแล้ว network จะมี timer ในการจับเวลาการตอบรับไม่ให้เกินเวลาที่กำหนด ถ้าเกิน network จะสงสัยว่าจะมีข้อผิดพลาดในการส่งข้อมูล CALL PROCEEDING นั้นจำเป็นในกรณีที่อุปกรณ์จะต้องตอบรับก่อนที่เวลาของ timer จะหมดลง

โดยทั่วไปแล้วฝ่ายรับจะส่งสัญญาน ALERTING message ในการตอบรับ SETUP message เพื่อบอกว่า user's terminal ได้รับสัญญาณและได้แจ้งกับ user แล้ว

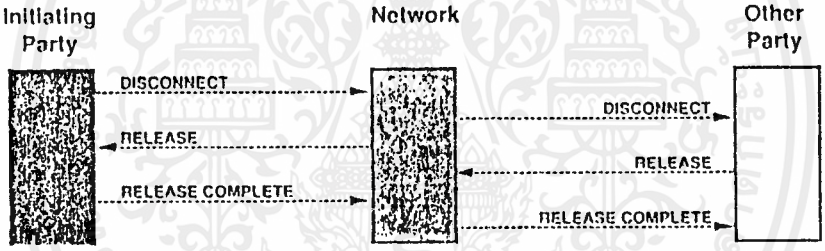
หลังจากนั้น network ก็จะสร้างสัญญาน ALERTING message แล้วส่งไปยังฝ่ายเรียก เมื่อฝ่ายรับรับการ call นั้นแล้วก็จะส่ง CONNECT message ไปให้ network เมื่อ network ได้รับแล้วก็จะสลับ Bearer channel ให้ แล้วส่ง CONNECT ACKNOWNLEDGE message ไปให้ฝ่ายรับ และส่ง CONNECT ไปให้ฝ่ายส่ง ฝ่ายส่งก็จะตอบด้วย CONNECT ACKNOWLEDGE message ซึ่งขณะนี้ทั้งฝ่ายรับและส่งก็จะพร้อมที่จะติดต่อกัน ซึ่งในภาษา ISDN จะเรียกว่า data transfer phase

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Disconnecting the call

การยกเลิกการติดต่อนั้นแสดงไว้ในรูปที่ 4.2 เริ่มจากฝ่ายใดฝ่ายหนึ่งส่ง DISCONNECT message ไปให้ network เมื่อ network ได้รับแล้วก็จะตอบกลับด้วย RELEASE และส่ง DISCONNECT ไปให้อีกฝ่ายหนึ่ง เมื่อฝ่ายแรกที่ส่งได้รับ RELEASE แล้วก็จะส่ง RELEASE COMPLETE message ไปให้ network ซึ่งขณะนี้ network ก็จะทำการปล่อย B-channel

ส่วนอีกฝ่ายหนึ่งเมื่อได้รับ DISCONNECT message แล้วก็จะตอบรับด้วย RELEASE message network ก็จะปล่อย B-channel แล้วส่ง RELEASE COMPLETE ไปให้ และที่จุดนี้ทรัพยากรทั้งหมดก็จะถูกปลดปล่อยให้ผู้อื่นสามารถเข้ามาใช้ได้



รูปที่ 4.2 Call termination message exchange

BASIC X.25 Calls

X.25 terminals นั้นจะเชื่อมต่อกับสิ่งที่เรียกว่า virtual circuit service ซึ่งอาจเป็น Packet Switched Data Network (PSPDN) หรือ packet handler ของ ISDN ก็ได้ Virtual circuit services จะมี protocol ที่ช่วยในการควบคุมการ call และการถ่ายเทข้อมูล หน้าที่ของ ISDN คือการเชื่อมต่อการ access ของ X.25 terminal กับ virtual circuit

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ISDN มี 3 วิธีในการทำการเชื่อมต่อการ access (access connections) คือ

- "Dial-up" access connection บน B-channel ใช้ basic circuit-mode call ในการเชื่อมไปถึง remote packet handler (ปกติก็คือ PSPDN)

- Packet-mode access connections บน B-channel เพื่อเชื่อมไปยัง packet handler ของ ISDN

- Packet-mode access connections บน D-channel เพื่อเชื่อมไปยัง packet handler ของ ISDN

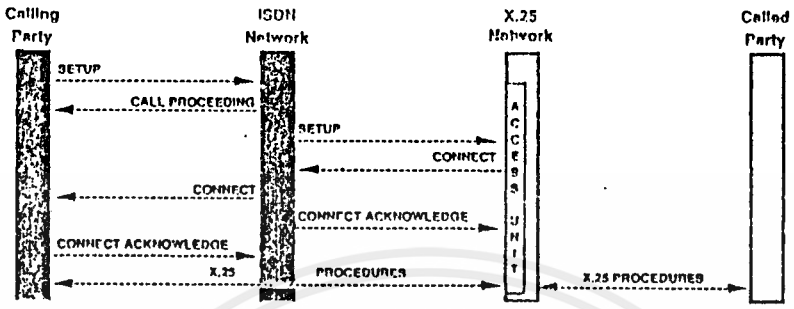
Circuit-Mode access to Remote Packet Handlers

ในแบบนี้ฝ่ายส่งนั้นจะต้องการใช้ bearer channel ในการเชื่อมต่อการ access ไปยัง PSPDN ส่วนฝ่ายรับที่จะรับ ISDN SETUP message จะเป็น access protocol ใน PSPDN ซึ่งเรียกว่า access unit (AU) (ในรูปที่ 3) ซึ่ง AU นั้นอาจไม่มีการส่ง CALL PROCEEDING หรือ ALERTING message ตามปกติ

ถ้า AU รับ X.25 call จากฝั่งของ PSPDN ซึ่งจะส่งไปให้ user ใน ISDN AU จะทำหน้าที่เหมือนเป็นผู้เรียกและสร้าง ISDN SETUP message ส่งไปยัง ISDN network

ในวิธีการนี้จะใช้ X.25 data link layer procedures (LAPB) แทน protocol ของ ISDN (LAPD)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 4.3 การแลกเปลี่ยน Message เพื่อสร้าง circuit-mode access connection ไปยัง remote packet handler

B-channel Access to ISDN virtual circuit Service

การเชื่อมต่อ B-channel ไปยัง ISDN packet handler เป็นวิธีการที่ทำให้ users สามารถเชื่อมต่อ B-channel ตรงไปยัง X.25 virtual circuit service ได้

Signaling Procedure จะคล้ายกับแบบ circuit-mode แต่มีบางอย่างที่แตกต่างคือไม่อนุญาตให้มีการส่งแบบ overlap และ network จะไม่มีการส่ง CALL PRODCEEDING message มันจะส่ง CONNECT message ไปทันทีที่ได้รับสัญญาน SETUP message

วิธีการนี้จะใช้ X.25 data-link layer procedures เช่นเดียวกับแบบ circuit-mode

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

D-channel Access to the ISDN virtual circuit service

D-channel Access to the ISDN virtual circuit service เป็นอีกวิธีหนึ่งในการเชื่อม user ไปยัง packet handler ของ ISDN โดยตรง โดยการใช้วิธีนี้ user เพียงแต่ใส่ X.25 packet เข้าไปใน LAPD Information frame, กำหนดให้ Service Access Point Identifier (SAPI) เป็น 16 แล้วส่งเฟรมไปยัง channel D

4.4 Controlling and Invoking Supplementary Services

ISDN ทำให้เราสามารถควบคุมและสั่งการ supplementary services ได้ 3 วิธีคือ keypad protocol, Feature key Management protocol และ Functional protocol

สำหรับ 2 วิธีแรกนั้นเราสามารถจัดให้เป็นลักษณะของ stimulus protocols Stimulus protocols คือ protocol ที่ user's terminal ไม่ต้องใช้ความรู้ใด ๆ ในการควบคุมและสั่งการ supplementary services ส่วน functional protocols นั้นมีขั้นตอนการทำงานคล้ายกับ basic call control protocols ซึ่ง user's terminal จะต้องรู้ถึง message และขั้นตอนในการส่ง /รับ message

Keypad Protocol

คือการที่ user ที่เป็นมนุษย์ทำการกดปุ่ม (0-9,#,*) บนหน้าปัดเพื่อทำการควบคุมและสั่งการ supplementary service ซึ่งข้อมูลนี้จะถูกส่งไปใน keypad facility information element ของ SETUP หรือ INFORMATION message เพื่อส่งไปให้ network อีกทีหนึ่ง

Feature Key Management Protocol

Feature Key Management Protocol เป็น protocol ที่คล้ายกับ keypad

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

protocol แต่ต่างกันตรงที่ปุ่มที่ใช้เป็นปุ่มพิเศษนอกเหนือจากปุ่มโทรศัพท์ธรรมดา

เมื่อมีการกดปุ่มแล้วจะมีการสร้างสัญญาน feature identifier number ซึ่งจะถูกส่งไปยัง Feature activation information element ของ SETUP หรือ INFORMATION message ซึ่งจะส่งไปให้ network เพื่อควบคุมและสั่งการ supplementary services

Functional Protocol

Functional Protocol นั้นสามารถแบ่งการทำงานได้เป็น 2 แบบคือ แบบ seperate messages และแบบ common information element procedure

ในแบบ seperate message นั้นจะใช้ message ต่าง ๆ กันในการกำหนดการทำงานที่ต้องการ มี message ทั้งหมด 6 messages และในจำนวนนี้ 3 messages ใช้ในการเรียก และอีก 3 messages ใช้ในการรับ (retrieve) ฝ่ายเรียกจะส่ง HOLD message ไปส่วนอีกฝ่ายจะตอบ HOLD ACKNOWLEDGE กลับมา HOLD REJECT จะใช้ในกรณีไม่สามารถทำงานที่ต้องการได้ เช่นเดียวกัน การ retrieve call จะส่ง RETRIEVE message ไปและจะรับ RETRIEVE ACKNOWLEDGE กลับมา ส่วน RETRIEVE REJECT ใช้ในกรณีเช่นเดียวกับ HOLD REJECT

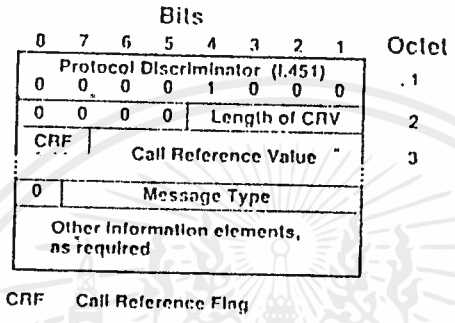
ส่วนใน common information element procedure จะใช้ information element คือ Facility information element ในการส่งสัญญานสำหรับ supplementary services และ Facility information element นี้จะอยู่ใน FACILITY message , หรือ REGISTER message หรือใน contorl message อื่น ๆ ส่วนขั้นตอนการทำงานของ information elements นี้ยังไม่ถูกกำหนดแต่จะออกมาในปี 1992 CCITT recommendation

4.5 Q.931 Message Format

ในหนึ่ง message นั้นประกอบด้วย information element ต่าง ๆ และในทุก message จะมีรูปแบบที่เหมือน ๆ กัน (รูปที่ 4.4) ประกอบด้วย information element ต่าง ๆ คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Protocol discriminator
- Call Reference Value
- Message type
- Information Elements อื่น ๆ



รูปที่ 4.4 รูปแบบของ Q.931 (I.451) messages

Protocol Discriminator

Protocol discriminator เป็นส่วนแรกของ message ทำหน้าที่บอกว่าส่วนที่ตามมานั้นเป็น message ใน layer 3 ซึ่งก็คือ Q.931 และ X.25

รูปแบบของ protocol discriminator เป็นดังตารางที่ 1 สังเกตุว่า บิท 5 และ 6 ที่ไม่ใช่ 00 จะเป็น protocol X.25

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

BITS 8 7 6 5 4 3 2 1	PURPOSE
0 0 0 0 0 0 0 0 through 0 0 0 0 0 1 1 1	Assigned in I.451; not available for use in the message protocol discriminator.
0 0 0 0 1 0 0 0	I.451 user-network call control messages.
0 0 0 1 0 0 0 0 through 0 0 1 1 1 1 1 1	Reserved for other Network Layer or Layer 3 protocols, including X.25.
0 1 0 0 0 0 0 0 through 0 1 0 0 1 1 1 1	Reserved for national use.
0 1 0 1 0 0 0 0 through 1 1 1 1 1 1 1 1	Reserved for other Network Layer or Layer 3 protocols, including X.25.
	All other values reserved.

ตารางที่ 4.1 Protocol Discriminator Codings

Call Reference Value (CRV)

Call Reference Value เป็นหมายเลขที่ใช้โดย user และ network ในการอ้างถึง call ที่กำลังติดต่ออยู่ (active call)

ใน Octet แรกของ call reference value จะบอกถึงจำนวน CRV ที่จะตามมา Call Reference Flag เป็นบิทที่ใช้ในการบอกว่าใครเป็นฝ่ายกำหนด call reference ขึ้นมา

Call Reference Value ที่มีค่าเป็น 0 เราจะเรียกว่า global call reference value เป็นการอ้างถึง calls ทั้งหมดที่ติดต่อกันอยู่

Message Type

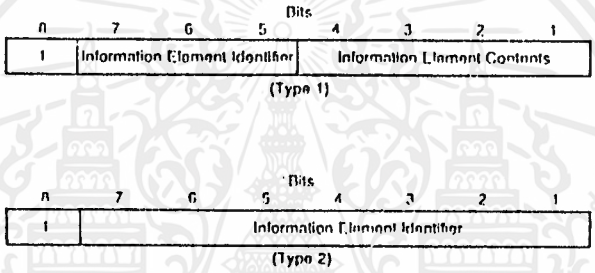
Message type information element ใช้บอกชนิดของ message

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

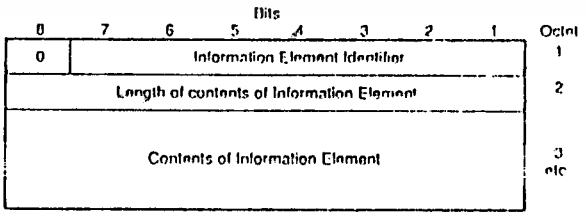
Other Information elements

ใน message หนึ่งๆ นอกจาก information element ข้างต้นแล้ว ยังประกอบด้วย inofrmation element อื่น ๆ อีก หลายตัวเพื่อที่จะทำงานที่แตกต่างกันไป

ใน message นั้นไม่ได้มีส่วนที่กำหนดความยาวเกาไว้ ดังนั้น information element จะมีส่วนกำหนดความยาวของตัวมันเองทั้งในแบบที่มีเพียง octet เดียวดังรูปที่ 4.5 หรือแบบมีหลาย octet ดังรูปที่ 4.6 สำหรับกรณีหลังนั้นจะมี field ที่ใช้บอกความยาวของ information element นั้น ๆ



รูปที่ 4.5 รูปแบบของ Single-octet information element



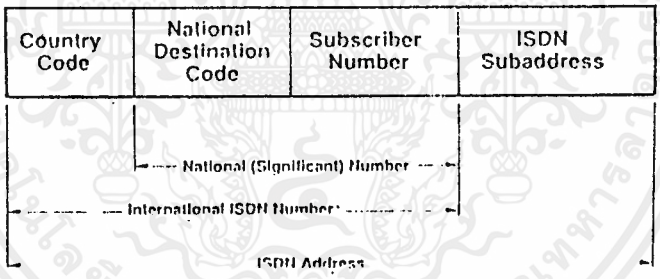
รูปที่ 4.6 รูปแบบของ Variable length information element

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.6 ADDRESSING

การกำหนด address ใน ISDN เป็นดังรูปที่ 4.7 โดยที่ address ของ ISDN นั้นจะมีความยาวไม่คงที่ International ISDN number ซึ่งมีความยาวได้ถึง 17 ตัวเลข ประกอบด้วย 2 fields คือ country code และ national (significant) number หรือ N(S)N Country code นั้นใช้กำหนดประเทศ ส่วน N(S)N กำหนด user แต่ละคนซึ่งใน field นี้จะประกอบด้วย National Destination Code และ Subscriber Number fields

Subaddresses เป็นหมายเลขเพิ่มเติมจาก ISDN number เพื่อใช้กำหนด address ภายใน private network Subaddress สามารถมีความยาวได้ถึง 40 ตัวเลข



รูปที่ 4.7 รูปแบบของ ISDN number (address)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

Packet Switching

เป็นกระบวนการสวิตชิงแบบ Store and Forward Switching (รับมาเก็บแล้วส่งต่อทันที โดยไม่มีการเก็บข้อมูลลงดิสก์ก่อน แต่เก็บชั่วคราวในหน่วยความจำของเครื่องแทน ซึ่งใช้เวลาเรียกใช้ หรือเก็บข้อมูลเร็วมาก)

เพื่อจะทำให้ "รับข้อมูลแล้วส่งทันที โดยไม่มีการเก็บข้อมูล" หรือใช้เวลาในการเก็บข้อมูลให้น้อยที่สุด ดังนั้นขนาดของข้อมูลจึงจำเป็นต้องมีขนาดเล็ก ซึ่งการทำให้ข้อมูลมีขนาดเล็ก ทำได้โดยการแบ่งข้อมูลที่จะส่งผ่านแพ็กเกจสวิตชิงออกเป็นชิ้น ๆ แต่ละชิ้นเรียกว่า "Packet" โดยแต่ละแพ็กเกจจะมีขนาด 64 ไบต์ หรือ 128 ไบต์ (สำหรับ Thaipak ขนาดแพ็กเกจจะมีขนาด 64 ไบต์) โดยภายหลังจากการส่งแพ็กเกจเหล่านั้นไปยังเน็ตเวิร์คปลายทาง แพ็กเกจเหล่านั้นจะถูก assemble หรือประกอบโครงสร้าง และรายละเอียดของข้อมูลให้เหมือนเดิมอีกที อุปกรณ์ที่เกี่ยวข้องกับการแบ่งแพ็กเกจ หรือต่อเชื่อมแพ็กเกจให้คืนโครงสร้างเหมือนเดิมนี้ เรียกว่า" PAD (Packet Assembler/ Disassembler) "

สำหรับประโยชน์ของแพ็กเกจสวิตชิงที่ดีกว่าการใช้โมเด็ม แล้วส่งข้อมูลบนคู่สายโทรศัพท์แบบเช่าก็คือความน่าเชื่อถือของการส่งข้อมูล ทั้งนี้เพราะเป็นระบบที่ได้มีการออกแบบกระบวนการตรวจสอบความถูกต้องของข้อมูลไว้แล้ว (built in) การประหยัดค่าใช้จ่าย นอกจากนี้ในกรณีที่ผู้ใช้ต้องการสร้างเครือข่ายของตนเอง (เช่น บริษัทที่มีเครือข่ายสาขามาก ๆ หลายพื้นที่) แต่ไม่ต้องการเข้าไปเกี่ยวข้องกับฮาร์ดแวร์ ซอฟต์แวร์ต่าง ๆ ตลอดจนการดูแลระบบ การซ่อมบำรุงเน็ตเวิร์คให้ทำงานอยู่ตลอดเวลา แพ็ก-เกจสวิตชิงจะเป็นตัวเลือกที่ดีที่สุดสำหรับองค์กรชนิดนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

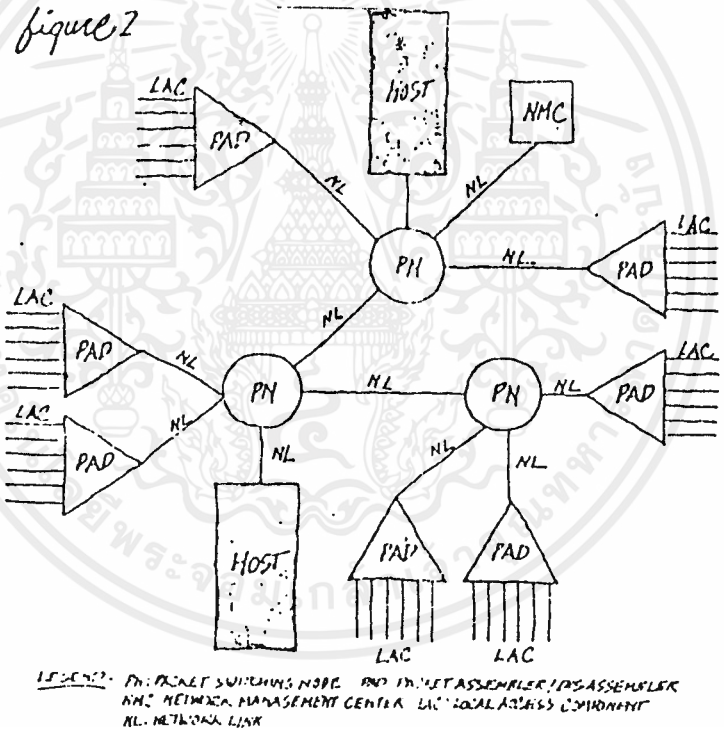
ตารางเปรียบเทียบข้อดีข้อเสียของการใช้คู่สายเช่า (leased line) กับแพ็กเกจสวิตชิง

ข้อพิจารณา	แพ็กเกจสวิตชิง	คู่สายโทรศัพท์แบบเช่า
ค่าใช้จ่าย	แปรตามจำนวนข้อมูลที่ใช้ และเวลา นาทีละ 3.5 บาท+64 ไบต์ละ 0.35 บาท	เหมือนค่าโทรศัพท์ทางไกล ซึ่งแปรตามระยะทาง และเวลา นาทีละ 42 บาท
ระบบตรวจสอบเพื่อความถูกต้องของข้อมูล	มีภายในกระบวนการแล้ว (Built in)	ผู้ใช้ต้องสร้างเอง
เวลาหน่วง	มีจนสามารถสังเกตเห็นได้	น้อยมากจนแทบไม่มี
การตรวจซ้ำเพื่อทดสอบความถูกต้องของข้อมูล	มีภายในกระบวนการ	ผู้ใช้ต้องสร้างเอง
ความเร็วในการส่ง	9,600 บิตต่อวินาที	38,400 บิตต่อวินาที
การดูแลเน็ตเวิร์ค	หน่วยงานที่ให้บริการดูแลให้	ผู้ใช้ดูแลระบบเอง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1 องค์ประกอบของ Packet Switching

1. ส่วนประกอบในการเข้าถึงแบบโลคัล (Local Access Component หรือ LAC)
2. Packet Assemblers/Disassemblers (PAD)
3. Packet Switching Node (PN)
4. Network Links (NL)
5. Network Management Center (NMC)



รูปแสดง แสดงความสัมพันธ์ระหว่างองค์ประกอบของ PSDN

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1.1 Local Access Component (LAC)

การส่งข้อมูลผ่าน PSDN (Packet Switching Data Network) เริ่มต้นข้อมูลจะเคลื่อนจากผู้ใช้ไปที่ส่วน PAD(PacketAssemblers/Disassemblers) หรือส่วน Packet Switching Nodes ซึ่งอยู่ภายใน PAD

LAC ประกอบด้วยอุปกรณ์ 3 ส่วนคือ

- ข้อมูลจากเทอร์มินัล (End User data Terminal)
- สายสัญญาณ (Physical Line)
- อุปกรณ์ส่งข้อมูล (Modem)

ปัจจุบัน LAC ที่ใช้ใน Packet Switching มี 3 ชนิดคือ

- switch analog line เช่น Dial-up
- leased analog channels เช่น สาย private
- leased digital channels เช่น DDS circuit

การใช้งานแต่ละแบบขึ้นอยู่กับความเร็วในการปฏิบัติงาน

3.1.2 Packet Assemblers / Disassemblers (PAD)

เป็นส่วนที่ให้ user สามารถที่จะติดต่อกับโครงข่ายได้ หน้าที่การทำงานเบื้องต้นของ PAD คือสร้างความ Compatible ระหว่างอุปกรณ์ที่ใช้งานหลายชนิดให้เข้ากันได้กับระบบโครงข่าย อุปกรณ์เทอร์มินัล (Terminal Device) เปลี่ยนแปลงไปตามผู้ผลิต , ลักษณะการติดต่อสื่อสารที่ใช้ protocol, ความเร็ว และรหัสต่าง ๆ ที่ใช้งาน

อย่างไรก็ตาม เอาท์พุทที่ได้จะต้องเป็นมาตรฐาน ก่อนที่จะถูกส่งไปที่ Packet Switching Node ส่วน PAD จะจัดการข้อมูลจากอุปกรณ์ต่าง ๆ โดยวิธีการที่ เรียกว่า Packetized ข้อมูลที่ได้ออกมา (ในมาตรฐานและรูปแบบอย่างเดียวกัน) จะถูกส่งไปที่ PN (ส่วน PAD จะต่อกับโนดผ่านทาง access port หรือ gateways)และอีกหน้าที่หนึ่งจะทำงานโดยการนำข้อมูลจากโนดมาทำการ Depacketize หรือ Disassemble ก่อนที่จะส่งไปที่ เทอร์มินัลปลายทาง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หน้าที่อื่น ๆ ของส่วน PAD

- ที่รวมของสายสัญญาณเชื่อมต่อ
- เรียกการ SETUP และ CLEAR
- แปลง PROTOCOL
- แปลงรหัส (CODE)
- protocol emulation
- Local switching
- Local call recording (billing) functions

PAD ถูกออกแบบให้สร้างผลลัพธ์ต่ำกว่า switching node (ในช่วง 10-100 Packet ต่อวินาที ประมาณ 10,000-100,000 บิท/นาที) บางที PAD ยังทำหน้าที่ร่วมกับ switching node

5.1.3 Packet Switching Nodes (PN)

หัวใจของ PSDN คือ ส่วน Packet Switching Nodes (PN) หน้าที่หลักที่สำคัญที่สุดคือ ทำการส่งแต่ละ packet ไปยังปลายทางที่ถูกต้อง

หน้าที่อื่น ๆ ของ PN คือ

- Call record journalling (billing)
- การตรวจสอบเกตในโครงข่าย (Diagnostic)
- สนับสนุนการติดต่อโดยตรงกับ Host
- เชื่อมต่อกับโครงข่ายภายนอก

PN จะให้ผลลัพธ์ที่สูงมาก (ประมาณ 70-3,000 packet/sec)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.1.4 Network Links (NL)

ส่วนของการเชื่อมต่อระหว่าง Packet Switching Nodes กับส่วนอื่น ๆ เรา เรียกว่า Network Links เทคโนโลยีการส่งข้อมูลต่างๆ จะใช้ NL ต่างกันไป โดย Network Links จะมี

- วงจร Analog
- วงจร Digital
- ระบบ Microwave
- ระบบ Satellite

5.1.5 Network Management Center (NMC)

ส่วนนี้บางทีเรียกว่า "Network Control Center" ใช้เพื่อการควบคุม และแสดงผล Packet switched Network

หน้าที่ที่สำคัญที่สุดของส่วนนี้คือการเก็บ และรักษาฐานข้อมูลของโครงข่าย โดยฐานข้อมูล (Database) นี้เป็น master software และโครงสร้างระบบในแต่ละ Node ในฐานข้อมูล ส่วน NMC จะเก็บตารางการติดต่อ และข้อมูลของผู้ใช้ ในกรณีที่เกิดปัญหาข้อมูลจากฐานข้อมูลจะถูกโหลดผ่าน NL ไปที่โนดที่เกิดความผิดพลาด เพื่อแก้ปัญหา โดยการควบคุมของส่วน NMC นี้ ทำให้โนดต่าง ๆ ใน Network สามารถทำงานโดยที่ไม่ต้องเอาใจใส่มากได้ ทำให้ลดค่าใช้จ่ายในส่วนนี้

หน้าที่อื่น ๆ ของ NMC ได้แก่

- ตรวจสอบความปลอดภัยในการติดต่อ / เรียกการ SETUP
- รวบรวมสถิติการทำงานของโครงข่าย
- รับสัญญาณจากอุปกรณ์ของโครงข่ายที่เกิดทำหน้าที่ผิดพลาด
- เก็บ และรวบรวมข้อมูล billing จากโนดของโครงข่าย
- ทำการทดสอบการทำ Diagnostic
- แก้ไขการทำงานของโครงข่ายแบบระยะไกล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนอุปกรณ์เทอร์มินัล หรือ workstation ของ NMC จะมีส่วนอินเตอร์เฟสระหว่างผู้ปฏิบัติงาน และคอมพิวเตอร์ควบคุมโครงข่าย ส่วนเวอร์กสเตชั่นผู้ปฏิบัติงานสามารถทำงานได้หลายอย่าง

- สร้างโครงสร้างการอินเตอร์เฟสของ end user ในโครงข่าย
- เซ็ต software หรือโครงสร้างเริ่มต้น
- การทดสอบโครงข่าย
- รับข้อมูลที่เป็นปัญหาจากโครงข่าย
- รับผลลัพธ์จากการ Diagnostic

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.2 Packet Switching และ Thaipak

THAIPAK เป็นแพ็กเกจสวิตชิง เน็ต เวิร์คที่อิงสเปค เน็ต เวิร์คตาม "คำแนะนำการใช้" ขององค์การ CCITT ที่ X.25 (X.25 ไม่ใช่ เน็ต เวิร์ค แต่ เป็นคำแนะนำมาตรฐานที่ทำให้เครื่องคอมพิว เตอร์สามารถสื่อสารไปมาบนแพ็ก เกจสวิตชิง เน็ต เวิร์คได้อย่างถูกต้อง หรือกล่าวได้ว่า X.25 คือโปรโตคอลที่ทำให้เครื่อง PC ธรรมดาที่บริษัทสามารถ เชื่อมต่อ เข้ากับแพ็ก เกจสวิตชิง เน็ต เวิร์คได้)

วิธีการต่อ เข้า เน็ต เวิร์คทำได้ 2 วิธีคือ

1. เชื่อมต่อโดยตรง (PVC : Permanent Virtual Circuits) โดยการใช้คู่สายโทรศัพท์เช่า (leased line) วิธีการนี้จะทำให้สามารถใช้ความเร็วในการส่งได้สูงคือ 9600 บิตต่อวินาที โดย เวลาใช้จะต้องรันซอฟต์แวร์และใช้ร่วมกับการ์ดอะแดปเตอร์ ซึ่งจะทำให้เป็น PAD ตัวหนึ่งเลย จากนั้นต่อ เข้ากับ leased line

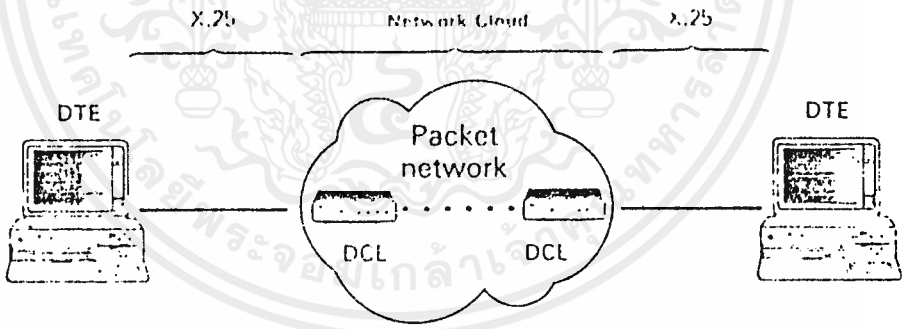
2. เชื่อมต่อโดยการหมุนโทรศัพท์เข้ากสท. (SVC : Switched Virtual Circuits) โดยทาง กสท. จะมีหมายเลขโทรศัพท์อยู่ชุดหนึ่งที่อนุญาตให้เราหมุน เข้าหาได้ ข้อดีของการ เชื่อมต่อโดยการหมุน เมื่อต้องการก็คือ ประหยัด เงินไม่ต้องทิ้งคู่สายโทรศัพท์ไว้ เป็นสาย เชื่อมต่อตลอดเวลา และสามารถ เรียก เข้าหา กสท. ที่ใด เมื่อไรก็ได้แต่ข้อ เสียคือ จะ เป็นการสื่อสารแบบทิศทาง เดียว เสียมากกว่า คือ เราจะใช้โทรไปหาคนอื่นได้ แต่คนอื่นๆ ไม่สามารถโทร เข้าหา เราได้ เหมือนกับ เป็นโฟนพอยน์นอกจากนี้ยังสามารถใช้งานที่ความ เร็วสูง ๆ คือ 9600 บิตต่อวินาที โดยความ เร็วที่อนุญาตให้ใช้คือ 1200 และ 300 บิตต่อวินาที

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

5.3 โปรโตคอล X.25

โปรโตคอล X.25 เป็นแพ็กเกจสวิตชิงที่ใช้การทำงานในลักษณะ "รับมาแล้วส่งต่อ" สร้างขึ้นมาโดยองค์การ CCITT โดย X.25 เป็นคำแนะนำมาตรฐานที่ทำให้เครื่องคอมพิวเตอร์สามารถสื่อสารไปมาบนแพ็กเกจสวิตชิงเน็ตเวิร์คได้อย่างถูกต้องรู้เรื่องกันหรือ อาจกล่าวว่า X.25 คือโปรโตคอลที่ทำให้เครื่อง PC ธรรมดาสามารถเชื่อมต่อเข้ากับแพ็กเกจสวิตชิงเน็ตเวิร์คได้ นอกจากนี้โปรโตคอล X.25 ยังกำหนดถึงข้อผิดพลาด และ การตรวจค้น

ใน packet network interface ซึ่งเป็นการแลกเปลี่ยนข้อมูลกันระหว่าง user device (DTE) และ network node (DCE) จะใช้โปรโตคอล X.25 เป็นข้อกำหนดในการเชื่อมต่อ ดังรูป



DTE: Data Terminal Equipment (User Equipment)
DCE: Data Circuit-Terminating Equipment (Network Node or Packet Exchange)

DTE : Data Terminal Equipment (User Equipment)

DCE : Data Circuit-Terminating Equipment (Network Node or Packet Exchange)

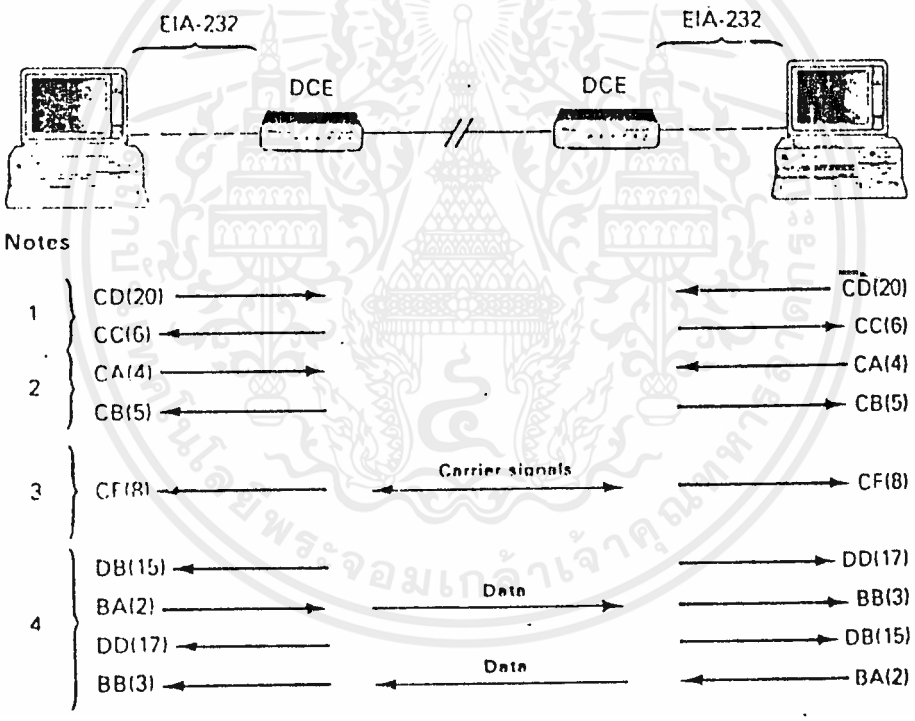
รูปแสดง X.25 Networkk Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

DTE และ DCE

DTE (Data Terminal Equipment) และ DCE (Data Circuit-terminating Equipment) เป็นอุปกรณ์ที่มีความสำคัญในการสื่อสารมาก อุปกรณ์ประเภท DTE จะตั้งอยู่ที่ปลายทางของระบบสื่อสาร ส่วน DCE จะเป็นตัวทำหน้าที่เชื่อมต่อ DTE เข้ากับระบบสื่อสารโดยจะปรับสถานะของสัญญาณให้เหมาะสมกับโครงข่ายสื่อสารนั้น ๆ

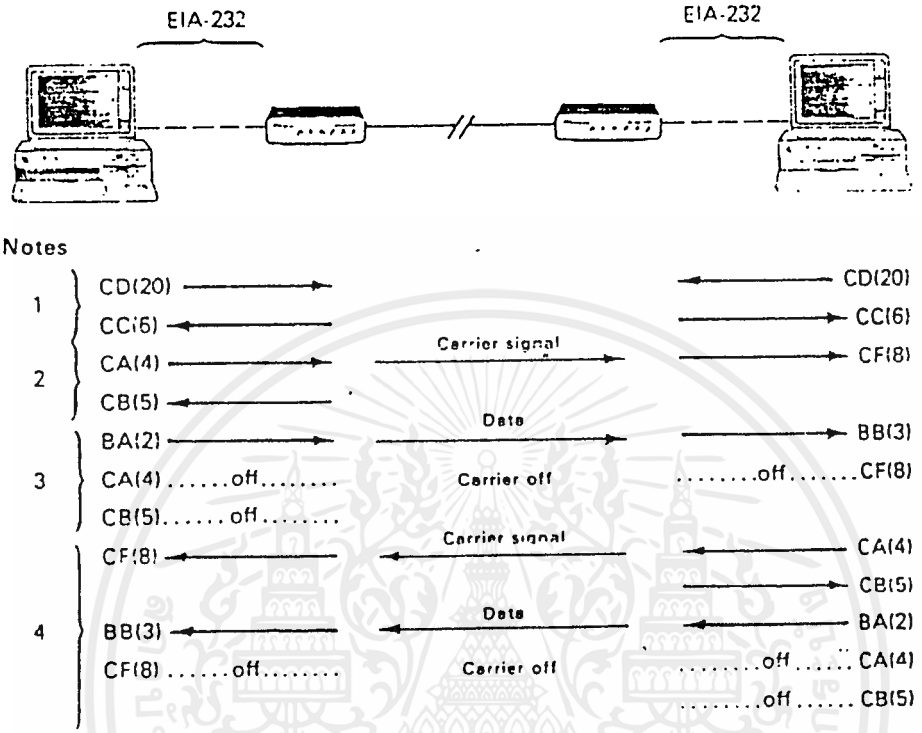
ตัวอย่างของ DTE เช่นคอมพิวเตอร์ และตัวอย่างของ DCE ก็คือโมเด็ม ซึ่งทำหน้าที่ปรับสถานะของข้อมูลให้เหมาะสมที่จะส่งผ่านไปในระบบโทรศัพท์ การเชื่อมต่อ DCE เข้ากับ DTE นั้น มาตรฐานที่กำหนดคือ RS232-C



Notes
1: Initial Power Up
2: Initiate Controls for Transmission
3: Synchronize DCEs
4: FDX Operations

รูปแสดง Full Duplex Operations Across DTE/DCE Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดง Half Duplex Operations Across DTE/DCE Interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากโมเดล OSI แบบ 7 ชั้นคือ

- ชั้นที่ 1 PHYSICAL LAYER
- ชั้นที่ 2 DATA LINK LAYER
- ชั้นที่ 3 NETWORKK LAYER
- ชั้นที่ 4 TRANSPORT LAYER
- ชั้นที่ 5 SESSION LAYER
- ชั้นที่ 6 PRESENTATION LAYER
- ชั้นที่ 7 APPLICATION LAYER

สำหรับโปรโตคอล X.25 จะครอบคลุมเพียง 3 ชั้นเท่านั้น คือ

- Physical Layer
- Data Link Layer

และ - Network Layer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Physical Layer

ฟังก์ชันของชั้นที่ 1 โดยพื้นฐานจะเป็นการทำงานเกี่ยวกับการส่งบิทของข้อมูล (0 หรือ 1) ไปตามช่ายการสื่อสาร ซึ่งในชั้นนี้จะกำหนดกฎในการส่ง 1 บิทของข้อมูลจากทางด้านส่งไปยังอีกด้านหนึ่งให้รับได้เป็น 1 บิท เหมือนที่ส่งออกมา และพื้นฐานต่าง ๆ ที่จะพิจารณาในชั้นนี้เช่น แรงดันไฟฟ้า กฎเกณฑ์สำหรับการเริ่มต้นในการติดต่อ การเลิกการติดต่อ เมื่อการส่งเสร็จสิ้นสมบูรณ์แล้ว รวมทั้งมาตรฐานการเชื่อมต่อ (Connector) ว่าจะเป็น RS-232C RS 449 หรือ X.21 และพิจารณาเกี่ยวกับลักษณะทางกายภาพที่สัญญาณไฟฟ้าและฟังก์ชันที่ทำหน้าที่ในการสร้าง บำรุงรักษา การเลิก ภายในช่วงต่อระหว่าง DTE และ DCE

ฟังก์ชันหลักของ Physical Layer

- การส่งข้อมูลระหว่างการอินเตอร์เฟสกันระหว่าง DTE (Computer Terminal) และ DCE (เช่น Modem / Connector / Cable)
- สัญญาณการควบคุมระหว่าง Devices
- Provide สัญญาณนาฬิกาให้การไหลของข้อมูล synchronous และ bit rate
- Provide สำหรับ electrical ground
- provide การเชื่อมต่อ (connectors) ระหว่าง machines เช่น pin , sockets และ plugs

ในชั้น physical มีคุณสมบัติในการ interface อยู่ 4 ประการคือ

1. Electrical
2. Functional
3. Mechanical
4. Procedural

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Electrical attributes

แสดงระดับ voltage (หรือกระแส) , เวลาของสัญญาณ Electrical , ลักษณะ Electrical อื่น ๆ ทั้งหมด (Capacitance , signal rise time)

Functional attributes

อธิบายถึงฟังก์ชันซึ่งมีอยู่ใน physical interface protocol จำนวนมาก แบ่งได้ดังนี้คือ ฟังก์ชันการควบคุม , function timing , data และ ground

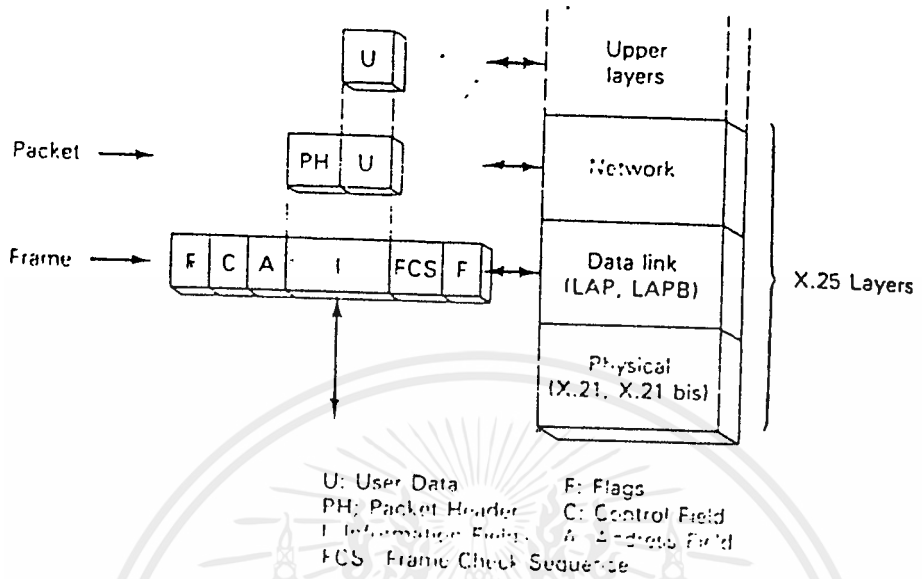
Mechanical attributes

อธิบายถึงขนาดของ connector และจำนวนสายที่จะใช้ในการ interface ซึ่งโดยทั่ว ๆ ไป ข้อมูลสัญญาณ และสายควบคุมจะอยู่ด้วยกัน

Procedural attributes

อธิบายถึงหน้าที่ของ connector ว่าควรทำอะไรบ้างและลำดับของเหตุการณ์ถูกร้องขอ ซึ่งจะก่อให้เกิดผลกับการส่งข้อมูลระหว่างการ interface

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



U : User Data

PH : Packet Header

C : Control Field

A : Address Field

I : Information Field

FCS : Frame Check Sequence

F : Flags

รูปแสดง X.25 Layer และ Protocol Data Units

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โปรโตคอล X.25 และ Physical Layer

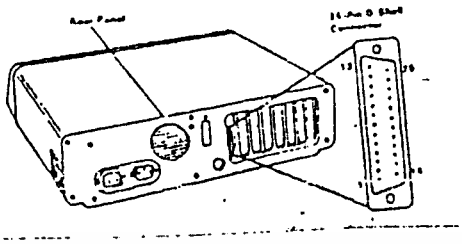
สำหรับการพิจารณาโปรโตคอล X.25 ในระดับ Physical นั้น ก็จะเป็นข้อกำหนดของการ interface เกี่ยวกับด้าน electrical ,mechanical และรูปแบบในการติดต่อสื่อสาร (Communication) และมาตรฐานของ Connector ที่ใช้คือ X.21 หรือ EIA-232C (เป็นมาตรฐานที่กำหนดโดย Electronic Industries Association (EIA)บางครั้งเรียก RS232C) ซึ่งใช้ในการ support PSDNs (Packet Switching Data Networks)ในแถบอเมริกาเหนือ X.21 ยังไม่เป็นที่นิยมกันเท่าไร ดังนั้น CCITT จึงกำหนดมาตรฐาน connectorในระดับ physical ขึ้นใหม่เรียกว่า X.21bis หรือในอเมริกาเรียกว่า EIA-232C หรือ RS232C

ในการ interface EIA-232C นั้นจะใช้ร่วมกับอุปกรณ์ทางการสื่อสารข้อมูลนั่นคืออุปกรณ์ใดที่ใช้ในการ interface ถ้า compatible กับ EIA-232C แล้วจะ compatible กับ X.25 PSDN ในชั้นที่ 1 (Physical Layer) ด้วย

RS232-C

สิ่งที่ RS232-C ระบุอยู่ได้แก่สัญญาณทางไฟฟ้าต่าง ๆ ตำแหน่งของสัญญาณคุณสมบัติทางไฟฟ้า และหน้าที่ของสัญญาณต่าง ๆ ในการสื่อสาร แต่ RS232-C ไม่ได้กำหนดคุณสมบัติทางกายภาพหรือลักษณะของขั้วที่ใช้ในการเชื่อมต่อ แต่เนื่องจากเราใช้คอนเนคเตอร์รูปตัว D หรือที่เรียกกันว่า D-connector ขนาด 25 ขาเป็นขั้วต่อสำหรับ RS232 มาตลอด จึงเกิดความเข้าใจผิดว่า D-connector เป็น RS232-C ความจริงแล้วคุณสมบัติทางกายภาพของ RS232-C จะกำหนดเพียงว่าถ้าหากเป็นอุปกรณ์ DTE แล้ว ขั้วต่อจะต้องเป็นแบบ Male และถ้าเป็น DCE แล้วจะต้องเป็นแบบ Female เท่านั้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดง ลักษณะของ D-Connector แบบ DB25 และ DB9

ขาสัญญาณของ RS232-C

ในส่วนของสัญญาณทางไฟฟ้านั้น RS232-C จะกำหนดหน้าที่และคุณสมบัติทางไฟฟ้าของขาต่าง ๆ เอาไว้และเรียกขาสัญญาณเหล่านั้นว่าวงจร (Circuit) ทั้งนี้เพราะเมื่อเชื่อมอุปกรณ์เข้าด้วยกันโดยใช้มาตรฐาน RS232-C แล้ว ขาสัญญาณแต่ละขาจะถูกเชื่อมเข้าด้วยกันเสมือนเป็นวงจรทั้งหมด ในแต่ละวงจรจะประกอบด้วยสวิตช์ที่ถูกควบคุมโดยอุปกรณ์ไม่ว่าจะเป็น DTE หรือ DCE และอุปกรณ์ตรวจจับสถานะของวงจรว่า ON คือมีกระแสไฟฟ้าหรือมีระดับแรงดัน หรือ OFF คือไม่มีกระแสไฟฟ้าหรือไม่มีระดับแรงดันหรือระดับแรงดันเท่ากับแรงดันอ้างอิง (ระดับ ground)

ขาสัญญาณทั้งหมดสามารถแบ่งออกเป็นกลุ่ม ๆ ดังนี้

- กลุ่ม Electrical Ground
- กลุ่ม Data Transfer
- กลุ่ม Control Signal

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม Electrical Ground

วงจร AA หรือ Protective Ground จะทำหน้าที่ป้องกันสายส่งสัญญาณจากสัญญาณรบกวนต่าง ๆ โดยทั่วไปแล้วนิยมเชื่อมเข้ากับตัวถังที่เป็นโลหะของเครื่องคอมพิวเตอร์ หรือเชื่อมเข้ากับตัวคอนเนคเตอร์ซึ่งคอนเนคเตอร์จะไปสัมผัสกับตัวถังอีกทีหนึ่ง ถ้าหากจะให้ได้ผลสมบูรณ์ควรจะเชื่อมตัวถังของเครื่องเข้ากับสายดินด้วย วิธีนี้นอกจากจะทำให้สามารถป้องกันสัญญาณรบกวนจากภายนอกที่จะรั่วไหลเข้าไปในสายส่ง และป้องกันสัญญาณรบกวนจากเครื่องคอมพิวเตอร์ไม่ให้ไปรบกวนเครื่องใช้ไฟฟ้าประเภทวิทยุ โทรทัศน์ได้

วงจร AB หรือ Signal Ground เส้นนี้มีไว้เป็นระดับแรงดันอ้างอิงสำหรับสัญญาณต่าง ๆ ใน RS232-C อีกที หรือที่ทางไฟฟ้านิยมเรียกว่าระดับกราวด์นั่นเอง อย่างไรก็ตามวงจรนี้ไม่มีความเกี่ยวข้องกับวงจร AA ถึงแม้ว่าชื่อจะคล้าย ๆ กัน วงจร AA นั้นเป็นวงจรที่เกี่ยวข้องกับระบบไฟฟ้ากำลังซึ่งอยู่ก่อนภาคจ่ายไฟ ส่วนวงจรนี้เป็นวงจรที่เกี่ยวข้องกับสัญญาณไฟฟ้า ซึ่งเป็นวงจรที่อยู่หลังจากภาคจ่ายไฟ เพราะฉะนั้นจึงห้ามเอาวงจรนี้ไปเชื่อมเข้ากับวงจร AA โดยเด็ดขาด เพราะอาจจะทำให้เกิดการลัดวงจรของระบบจ่ายไฟอาจจะทำให้ภาคจ่ายไฟฟ้าของคอมพิวเตอร์เสียหายได้ วงจรนี้อยู่ที่ขา 7 ของคอนเน็คเตอร์

กลุ่ม Data Transfer

วงจร BA หรือ Transmit Signal เป็นวงจรที่ทำหน้าที่ส่งข้อมูลอุปกรณ์ออกจากอุปกรณ์ DTE ไปยังอุปกรณ์ DCE การทำงานของวงจรนี้จะขึ้นอยู่กับวงจร CA, CB, CC และ CD ตำแหน่งของวงจรนี้อยู่ที่ขา 2

วงจร BB หรือ Receive Signal ทำหน้าที่รับข้อมูลจาก DCE ที่จะเข้ามาสู่ DTE วงจรนี้มีการทำงานอิสระไม่ได้ขึ้นกับวงจรอื่น ๆ เหมือนวงจร BA ตำแหน่งของวงจรนี้อยู่ที่ขา 3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

กลุ่ม Control Signal

วงจร CA หรือ Request to Send เป็นสัญญาณที่ DTE ใช้บอก DCE ว่ามีข้อมูลที่ต้องการจะส่งออกจาก DTE เมื่อใดก็ตามที่วงจรนี้เปลี่ยนสถานะจาก OFF ไปเป็น ON อุปกรณ์ DTE จะต้องเตรียมตัวที่จะส่งข้อมูล วงจรนี้อยู่ที่ขา 4

วงจร CB หรือ Clear to Send เป็นสัญญาณที่ DCE ใช้บอก DTE ว่าพร้อมที่จะรับข้อมูลจาก DTE แล้ว สัญญาณนี้มักจะทำงานหลังจากที่ DCE ตรวจพบสถานะ ON ของวงจร CA หรือสัญญาณ Request to Send จาก DTE หลังจากที่ทำการเตรียมพร้อมเสร็จก็จะส่งสัญญาณ Clear to Send หรือ ON วงจร CB เพื่อบอก DTE ว่าพร้อมที่จะรับข้อมูลแล้ว ตำแหน่งของวงจรนี้อยู่ที่ขา 5

วงจร CC หรือ Data Set Ready เป็นวงจรที่ DCE ใช้บอก DTE ว่าอุปกรณ์ DCE อยู่ในสถานะเตรียมพร้อมที่จะรับส่งข้อมูลได้แล้ว นั่นคืออุปกรณ์ DCE สามารถติดต่อกับ DCE ของปลายทางได้ และพร้อมที่จะรับส่งข้อมูลแล้ว ตำแหน่งบนคอนเน็คเตอร์คือขาที่ 6

วงจร CD หรือ Data Terminal Ready เป็นวงจรที่ DTE บอก DCE ว่าอุปกรณ์ DTE อยู่ในสถานะเตรียมพร้อมที่จะรับส่งข้อมูลได้แล้วทั้ง CC และ CD จะอยู่ในสถานะ ON เมื่ออุปกรณ์นั้น ๆ พร้อม เช่น ถ้าหาก DCE พร้อมก็จะทำการ ON วงจร CC และถ้าหาก DTE พร้อมก็จะทำการ ON วงจร CD เป็นต้น ตำแหน่งของวงจรนี้อยู่ที่ขา 20 บน DB25 หรือขาที่ 9 บน DB9

วงจร CE หรือ Ring Indicator เป็นวงจรที่ DCE ใช้บอก DTE ว่าสามารถตรวจพบสัญญาณกระดิ่งจากสายสื่อสาร

วงจร CF หรือ Receive Line Signal Detect นิยมเรียกกันอีกชื่อว่า Carrier Detect หรือสัญญาณ CD ตามที่ปรากฏในหน้าปัดของโมเด็ม (ไม่ใช่วงจร CD แต่เป็นลักษณะเดียว) เป็นวงจรที่ DCE บอกกับ DTE ว่าสามารถรับสัญญาณ Carrier จาก DCE อีกตัวหนึ่งได้ ตำแหน่งของวงจรอยู่ที่ขา 8

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

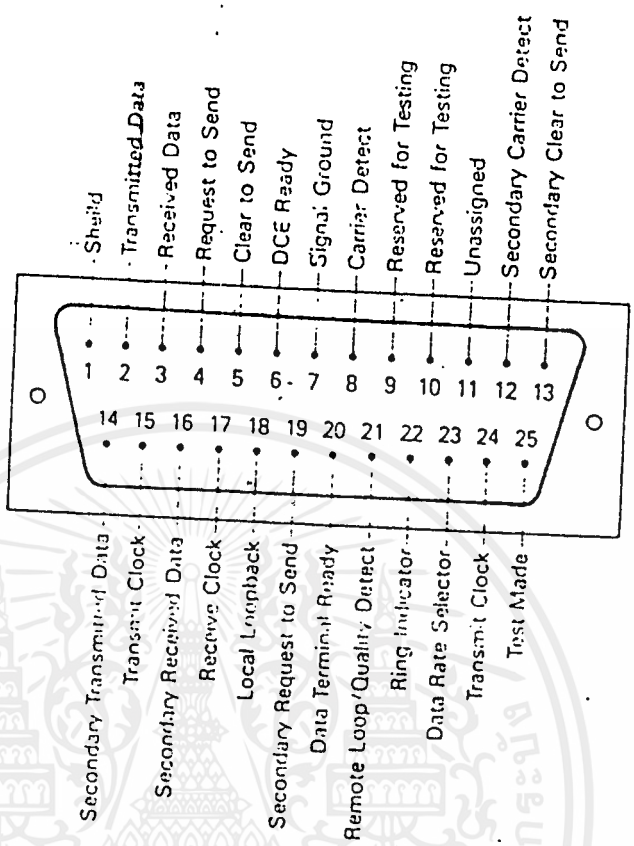
วงจร CG หรือ Signal Quality Detector เป็นวงจรที่กำหนดขึ้นมาเพื่อควบคุมคุณภาพของระบบสื่อสาร เมื่อใดก็ตามที่สัญญาณนี้แอกทีฟจะหมายความว่าข้อมูลได้รับนั้นเชื่อถือได้ว่าไม่มีการผิดพลาด

วงจร CH และ CI หรือ Data Signal Rate Selector เป็นวงจรที่ใช้เลือกอัตราเร็วในการส่งข้อมูลระหว่าง DTE กับ DCE หน้าที่ของวงจรนี้ค่อนข้างจะคาบเกี่ยวกับหน้าที่หน้าที่ของวงจรในกลุ่ม Clocking Signal

กลุ่ม Clocking Signal

ในกลุ่มนี้จะเป็นวงจรที่ทำหน้าที่เกี่ยวกับการกำหนดอัตราเร็วในการแลกเปลี่ยนข้อมูลระหว่าง DTE และ DCE รวมทั้งสัญญาณนาฬิกาที่ทำหน้าที่ควบคุมจังหวะการแลกเปลี่ยนข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



Pin	Circuit	Source	Description
1	AA	—	Shield
2	BA	DTE	Transmitted Data
3	BB	DCE	Received Data
4	CA	DTE	Request to Send
5	CB	DCE	Clear to Send
6	CC	DCE	DCE Ready
7	AB	—	Signal Ground
8	CF	DCE	Carrier Detect
9	—	—	Reserved for Data Set Testing
10	—	—	Reserved for Data Set Testing
11	—	—	Unassigned
12	SCF	DCE	Secondary Carrier Detect
13	SCB	DCE	Secondary Clear to Send
14	SBA	DTE	Secondary Transmitted Data
15	DB	DCE	Transmission Signal Element Timing
16	SBB	DCE	Secondary Received Data
17	D	DCE	Receiver Signal Element Timing
18	LL	—	Local Loopback
19	SCA	DTE	Secondary Request to Send
20	CD	DTE	Data Terminal Ready
21	RL/CG	DCE	Remote Loopback/Signal Quality Detector
22	CE	DTE	Ring Indicator
23	CH	DTE	Data Signal Rate Selector
23	CI	DCE	Data Signal Rate Selector
24	DA	DTE	Transmit Signal Element Timing
25	TM	—	Test Mode

รูปแสดง ขาต่าง ๆ แบบ D-Connector (DB25)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การเชื่อมต่ออุปกรณ์ด้วย RS232-C

ในการเชื่อมต่ออุปกรณ์ 2 ชิ้นเข้าด้วยกันโดยใช้ RS232-C ทำได้หลายวิธีแต่ก่อนอื่นต้องพิจารณาว่าอุปกรณ์เหล่านั้นเป็นอุปกรณ์ประเภทใด เป็น DTE หรือ DCE ซึ่งเป็นจุดที่สำคัญมาก เพราะถ้าพิจารณาผิดแล้วก็จะต่อผิดไปด้วย

ตัวอย่างเช่น

- การเชื่อมต่อระหว่างคอมพิวเตอร์กับโมเด็ม จะเป็นการเชื่อมต่อระหว่าง DTE กับ DCE
- การเชื่อมต่อระหว่างคอมพิวเตอร์ 2 เครื่อง จะเป็นการเชื่อมต่อระหว่าง DTE กับ DTE (จะใช้การเชื่อมต่อแบบไม่เป็นมาตรฐาน)

วิธีการเชื่อมต่อ DTE เข้ากับ DCE ทำได้ 3 วิธีดังนี้คือ

- การเชื่อมตามมาตรฐาน คือเชื่อมทุก ๆ วงจรเข้าด้วยกัน วิธีนี้จะทำให้ได้คุณสมบัติของ RS232-C ครบถ้วนทุกประการ แต่จะไม่เป็นการประหยัด เพราะต้องใช้สายอย่างน้อย 9 เส้น ถ้าเป็นระยะทางไกล ๆ แล้วค่าสายสัญญาณอาจจะแพงเกินไป จึงมีการคิดวิธีการต่อแบบประหยัดขึ้นดังวิธีต่อไป

- การต่อแบบใช้สายเพียง 3 เส้นต่อระหว่างอุปกรณ์ 2 ชิ้น ซึ่งสามารถใช้งานได้กับอุปกรณ์บางตัวเท่านั้น ทั้งนี้เพราะอุปกรณ์บางตัวมีการตรวจสอบวงจรอื่น ๆ ด้วย เมื่อมีการตรวจพบว่าวงจรอื่น ๆ ไม่มีการตอบสนองในทางที่ถูกต้องแล้วอาจจะไม่สามารถทำการรับส่งข้อมูลได้ จึงมีการพัฒนาการเชื่อมต่อแบบประหยัดขึ้นมาโดยทำการ Loop Back ซึ่งคือวิธีต่อไป

- การเชื่อมต่อแบบ Loop Back คือการเชื่อมวงจรของอุปกรณ์ตัวเดียวกันบางวงจรเข้าด้วยกันเพื่อหลอกอุปกรณ์ว่ามีสัญญาณหรือสถานะที่สามารถทำการรับส่งข้อมูลเกิดขึ้น การใช้วิธีนี้จะได้ผลก็ต่อเมื่อเรามีซอฟต์แวร์คอยตรวจสอบความถูกต้องของข้อมูลอีกทีหนึ่ง ซึ่งก็ได้แก่โปรโตคอลการรับส่งไฟล์ต่าง ๆ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

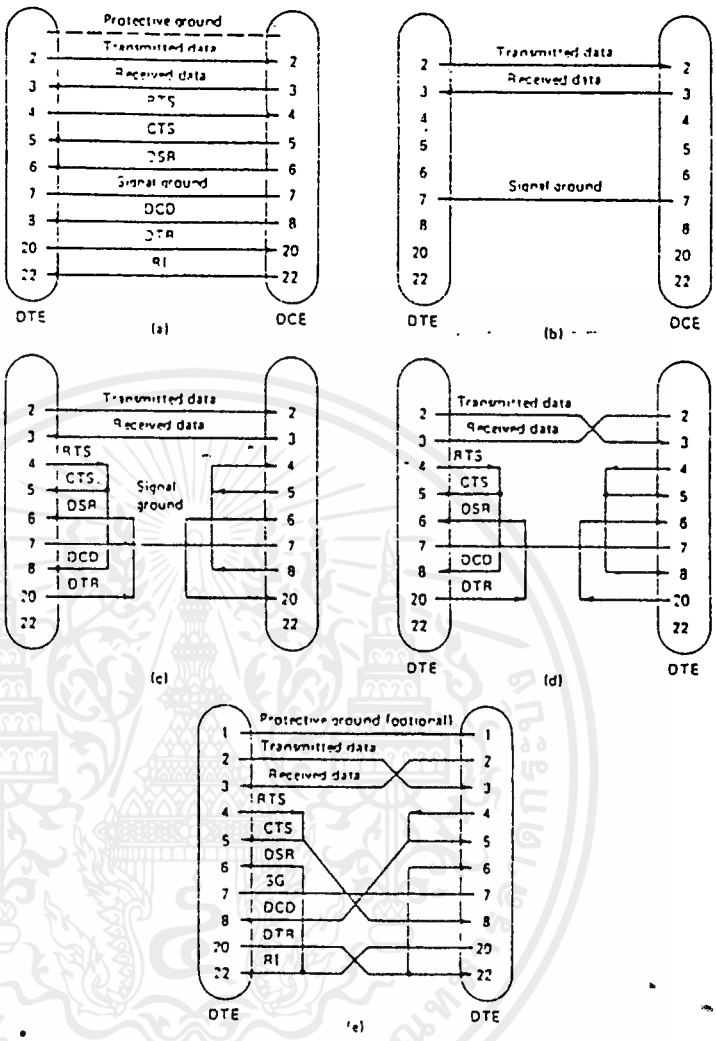
การเชื่อมต่อระหว่าง DTE สองตัวนั้นจะมีขั้นตอนพิเศษเพิ่มขึ้นมาอีกเล็กน้อย คือจะต้องมีการสลับสัญญาณของทั้ง 2 ฝั่งเป็นบางตัวเพื่อให้หลอก DTE ว่าอุปกรณ์ตรงข้ามเป็น DCE ซึ่งสามารถทำได้ 2 วิธีคือ

- ทำ Loop Back

หรือ - ทำ Double Cross

จุดสำคัญในการเชื่อมต่อ DTE สองตัวเข้าด้วยกันก็คือการสลับวงจร BA และ BB ของทั้ง 2 ฝั่งวิธีนี้เรียกว่า Null Modem

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปแสดง การเชื่อมต่อ RS232-C ระหว่างอุปกรณ์ 2 ชุด ในแบบที่เป็นมาตรฐานและไม่เป็นมาตรฐาน รูป a ถึง c เป็นการเชื่อมต่อ DTE เข้ากับ DCE ส่วนรูป d และ e เป็นการเชื่อมอุปกรณ์สองชุดเข้าด้วยกัน

รูป a เป็นการเชื่อมต่อแบบปกติตามมาตรฐาน

รูป b เป็นการเชื่อมต่อแบบประหยัดคือใช้สายเพียง 3 เส้นเท่านั้น

รูป c เป็นการเชื่อมต่อแบบประหยัดและมีการทำ Loop Back

รูป d เป็นการเชื่อมต่อแบบ Null Modem พร้อมกับทำ Loop Back

รูป e เป็นการเชื่อมต่อแบบ Null Modem พร้อมกับทำ Double Cross

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Data Link Layer

หน้าที่หลักของชั้นนี้คือ เมื่อมีการส่งข้อมูลจากผู้ส่งไปยังผู้รับ จะมีการตรวจสอบว่าข้อมูลที่ผู้รับได้รับนั้นถูกต้องตามที่ทางด้านส่งส่งไปหรือไม่ และเมื่อเกิด ERROR ขึ้นสามารถแก้ไขได้เร็ว โดยจะมีการจัดวงจรการสื่อสารพื้นฐานที่สร้างขึ้นในชั้นที่ 1 (physical layer) และทำการเปลี่ยนให้ไปคงอยู่ในวงจรหรือช่วงต่อที่ไม่มีข้อผิดพลาดในการส่งช่วงต่อที่ไม่มีข้อผิดพลาดของการส่งนี้จะเชื่อมต่ออย่างใกล้ชิดกับชั้นที่ 3 คือ Network Layer ชั้นช่วงต่อข้อมูลนี้มีการทำงานโดยการแบ่งข้อมูลอินพุทออกมาเป็นเฟรมของข้อมูล แล้วส่งเฟรมเหล่านี้ไป

ในชั้นที่ 1 จะทำการรับ และส่งข้อมูลเป็นอนุกรมทางบิทโดยไม่สนใจถึงความหมายหรือโครงสร้างของมันเอง หน้าที่การแบ่งเป็นเฟรม การกำหนดขอบเขตของเฟรม ตลอดจนการตรวจสอบข้อผิดพลาดระหว่างการส่งเป็นหน้าที่ของชั้นที่ 2

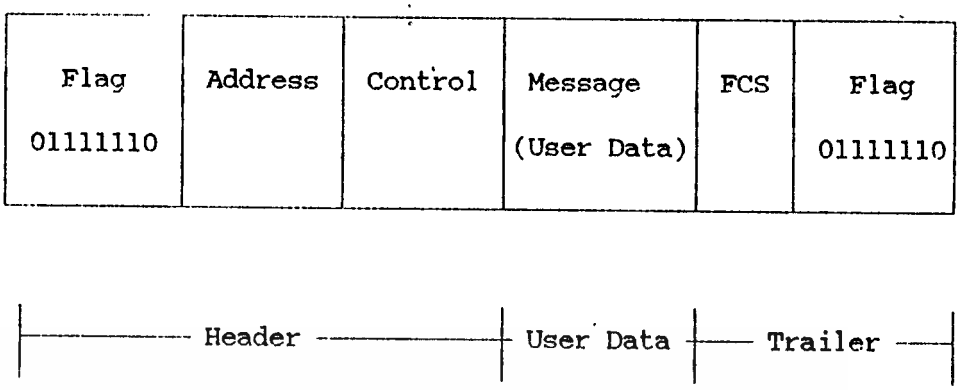
หน้าที่ของชั้นนี้ยังรวมถึงการแก้ปัญหาเกี่ยวกับการสูญเสีย การถูกทำลายหรือการซ้ำกันของเฟรมเพื่อให้ชั้นที่ 3 ได้ข่าวสารที่ไม่มีความผิดพลาดไปใช้งาน

X.25 and the Data Link Layer

โปรโตคอล X.25 จะ assume Data Link Layer เป็น LAPB (Link Access Procedure Balance) ซึ่งเป็น subset ของ HDLC ในชั้นนี้จะจัดการเกี่ยวกับการส่งข้อมูล โดยจะทำการส่ง data units เป็นเฟรม ซึ่งเรียกว่า LAP-B frames ในการส่งเฟรมข้อมูลนั้นจะเป็นการส่งจาก DTE ไปยัง DCE

รูปแบบของ LAP-B frame แสดงดังรูป โดยจะมีการแบ่งเฟรมออกเป็นส่วน Header,User Data ,และ Trailer ในส่วนของ Header และ Trailerจะแบ่งออกเป็น field ย่อย ๆ คือ flag , address , control , และ frame check sequence แต่ละ field จะประกอบไปด้วย bit pattern ซึ่งผู้ส่งเป็นผู้สร้างขึ้นมา และทางด้านผู้รับก็จะทำการอ่านข้อมูลจากเฟรมข้อมูลที่รับมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



FCS : Frame Check Sequence

รูป . เฟรม ข้อมูลของ X.25

รายละเอียด เฟรมข้อมูลของ X.25

- Flag ใช้ในการบอกจุด เริ่มต้นและสิ้นสุดของข้อมูล โดยแสดงด้วยชุดของบิท ในรูปแบบ 01111110 หรือบางที เรียกว่าแฟลก เริ่มต้นและสิ้นสุด (Begining and ending flag)

แฟลก เริ่มต้นอ้างถึงตำแหน่งของส่วนของ เฟรมที่ เป็นแอด เดรสและควบคุมรวมทั้งการเริ่มต้น วิธีการตรวจสอบ ข้อผิดพลาด

สำหรับแฟลกสิ้นสุด เป็นตัวบอกการจบสิ้นของวิธีการตรวจสอบข้อผิดพลาดในกรณีที่มีเฟรมติดต่อกันหลาย เฟรม เฟรมนี้อาจจะ เป็นแฟลก เริ่มต้นของ เฟรมต่อไป

- Address field ใช้สำหรับการบ่งชี้อุปกรณ์ปลายตัวหนึ่ง สำหรับระบบสายแบบจุดต่อจุด บางครั้งอาจใช้สำหรับการแยกแยะ command สำหรับการตอบรับ หรือ เพื่อการกำหนดอุปกรณ์ เฉพาะหลังอุปกรณ์ปลาย สำหรับอุปกรณ์ปลายที่มีหลาย ๆ ฟังก์ชัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Control field เป็นส่วนที่แสดงถึงชนิดของเฟรมที่จะส่ง เช่น เป็นเฟรมของข่าวสาร (Information) เฟรมชี้แนะ (Supervisory) หรือ เฟรมที่ไม่ถูกกำหนดหมายเลข (Unnumbered)

เฟรมข่าวสาร (Information) ใช้สำหรับการส่งหรือรับข่าวสาร บอกหมายเลขเฟรมกรณีที่มีเฟรมหลายเฟรมติดต่อกัน

เฟรมชี้แนะ (Supervisory) ใช้สำหรับการส่งข่าวสาร หรือแอ็คโนว เล็ด เมนท์ เพื่อ

- แสดงถึงเฟรมที่คาดว่าจะได้รับต่อไป
- แสดงความผิดพลาดในการส่งที่ตรวจตักได้
- ตอบรับว่าเฟรมทุกเฟรมที่ส่งมาได้รับอย่างถูกต้อง
- หยุดการส่ง
- เรียกการส่งซ้ำสำหรับเฟรมที่กำหนดบางเฟรม

เฟรมที่ไม่กำหนดหมายเลข (Unnumbered) ใช้สำหรับวัตถุประสงค์อื่น ๆ เช่น เพื่อจัดหาคอมมานด์ "disconnect" ที่ให้อุปกรณ์ปลายสามารถแจ้งการเลิกใช้งานได้

เฟรมที่ใช้สำหรับการควบคุมอาจสูญหายหรือผิดพลาดไป เช่นเดียวกับเฟรมข่าวสาร ดังนั้นเฟรมเหล่านี้จึงต้องมีการตอบรับ (Acknowledgement) เช่นกัน

- Message field หรือ ฟิลด์ข่าวสาร เป็นฟิลด์ที่มีความยาวเปลี่ยนแปลงได้ ข่าวสารก็คือข่าวสารของผู้ใช้หรือที่ผู้ใช้เรียกขอ ฟิลด์นี้อาจรวมถึงตัวแยกรูปแบบ จำนวนกลุ่มของทางเดินสัญญาณทางตรรก จำนวนทางเดินทางตรรก ตัวแยกชนิดของ packet แอดเดรส DTE ที่เป็นข่าวสารภายใน (DTE ที่เรียก หรือ DTE ที่ถูกเรียก) และข่าวสารที่จะส่ง

จะสังเกตเห็นว่าในการกำหนดลำดับจะมีอยู่ 2 ชนิดคือ

- ในฟิลด์การควบคุมจะเรียงลำดับหมายเลขของเฟรมต่าง ๆ ที่ต่อเนื่องกัน
- ในฟิลด์ของข่าวสาร จะเรียงลำดับหมายเลขของแพ็กเกจแต่ละแพ็กเกจ เมื่อระบบแบ่งข่าวสารออกเป็นแพ็กเกจจำนวนหลายแพ็กเกจ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Frame check sequence field (FCS) หรือฟิลด์สำหรับการตรวจสอบเฟรม จะทำหน้าที่ในการจัดหาการคำนวณเกี่ยวกับการตรวจสอบความผิดพลาด CRC (Cyclic Redundancy checking) ขนาด 16 บิท เมื่อได้รับข่าวสารมาแล้วทางด้านรับจะนำข่าวสารมาคำนวณหาค่า CRC ใหม่อีกครั้งแล้วนำไปเปรียบเทียบกับค่าที่ได้รับมา เพื่อหาว่ามีข้อผิดพลาดเกิดขึ้นในการส่งหรือไม่ ถ้าค่าของ CRC ไม่เหมือนกันแสดงว่ามีข้อผิดพลาดเกิดขึ้น จะมีการส่งสัญญาณร้องขอไปยังทางด้านส่งให้ส่งข้อมูลหรือข่าวสารนั้นมาใหม่

ฟังก์ชันหลักของ LAP-B คือ

- Link management
- Error control
- Flow control
- Failure control

Link management

ใน LAP-B สามารถใช้เฟรมชนิดต่างกันได้หลายชนิด บางเฟรมอาจจะประกอบไปด้วยข้อมูล และ application ต่าง ๆ ของ end-user ส่วนเฟรมอื่นๆจะถูกส่งผ่านการ link โดยจะมีฟังก์ชันที่ทำขึ้นเป็นตัวกำหนดว่าจะยอมรับเฟรมข้อมูลนั้นหรือไม่ จะยกเลิกการติดต่อที่มีอยู่ขณะนั้นหรือไม่

Error control

จะมีการตรวจเช็คข้อผิดพลาด(ERROR) โดยการใช้ FCS (Frame Check Sequence) ซึ่งเป็นส่วนที่ทำในการตรวจสอบโดยการนำข้อมูลที่จะส่งไปผ่านขบวนการทางคณิตศาสตร์ ก็จะได้ส่วน FCS ออกมา หลังจากนั้นข้อมูลจะถูกส่งออกไปตามสายสื่อสารพร้อมกับค่า FCS ทางด้านรับจะนำข้อมูลที่รับมานั้นไปผ่านกระบวนการทางคณิตศาสตร์เดียวกันกับที่ทางด้านส่งใช้เพื่อสร้าง FCS ขึ้นมาใหม่ เป็น FCS ที่ได้จากการคำนวณของทางด้านรับ สำหรับ FCS ที่ส่งมาจากทางด้านส่งนั้น เมื่อทางด้านรับได้รับมาก็จะนำมาเปรียบเทียบกับ FCS ที่คำนวณได้ ถ้าหากว่า FCS ทั้งสองตรงกันก็แสดงว่าข้อมูลที่ส่งมาถูกต้อง แต่ถ้าไม่ตรงกันก็แสดงว่าเกิดความผิดพลาดขึ้น จะต้องมีการแก้ไขให้ถูกต้องต่อไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Network Layer

ฟังก์ชันของชั้นนี้เกี่ยวข้องกับการทำงานภายในของโครงข่าย เช่น การกำหนดแกดเดรสของอุปกรณ์ต่อแกดจนเกี่ยวกับเส้นทางเดิน อาจกล่าวได้ว่าเป็นชั้นที่จัดเตรียมการบริการการส่งข้อมูลผ่านโครงข่ายไปยังอุปกรณ์ปลายที่จุดปลายทาง ชั้นโครงข่าย นี้เป็นชั้นที่ควบคุมการทำงานรวมของชั้นที่ 1, 2,และ 3 ซึ่งบางครั้งเรียกโครงข่ายนี้ว่า Packet Switching Network

ข่าวสารการควบคุมที่สร้างโดยชั้นนี้ รวมถึงการต่อสายและการร้องขอการสิ้นสุด (Termination request) รวมทั้งข่าวสารการยืนยันต่าง ๆ ซึ่งต้องสร้างในชั้นนี้และส่งไปยังชั้น Data Link เพื่อนำไปประกอบเป็นเฟรมแล้วส่งต่อไปยังชั้น Physical เพื่อการส่งออกไปตามสายสื่อสาร

สรุปหน้าที่ของ X.25

- การเรียกการสิ้นสุด (Termination request)
- การกำหนดหมายเลขทางเดิน (ในทางตรรก) ที่ทางด้านเริ่มต้นการทำงานเรียกมา
- แบ่งข่าวสารออกเป็น packet หลาย ๆ packet หรือหลายเฟรม
- ป้องกันการ overload โดยการควบคุมการไหลของ packet
- กำหนดทางเดินที่ packet จะถูกส่งออกไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทวิจารณ์และสรุป

ได้ทำการศึกษาระบบ ISDN รวมทั้งรูปแบบและการทำงานของโปรโตคอล X.25 ครบตามวัตถุประสงค์ที่ได้ตั้งไว้ แต่ในส่วนของการพัฒนาซอฟต์แวร์ และ ฮาร์ดแวร์ ที่จะนำมาใช้กับระบบ ISDN ไม่สามารถทำการสร้างขึ้นมาได้ เนื่องจากขาดอุปกรณ์หลายอย่างคือ อุปกรณ์จำลองเครือข่ายชุมสาย ISDN และ อุปกรณ์วิเคราะห์โปรโตคอล ซึ่งเป็นเครื่องมือที่มีความสำคัญในการทำโครงงานชิ้นนี้ ทำให้ไม่สามารถทำการทดลองการส่งข้อมูลจริงๆ ได้ ซึ่งก็ทำให้โครงงานส่วนนี้ไม่มีชิ้นงานออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาคผนวก

แผนงานและเป้าหมายในการพัฒนาโครงข่ายโทรศัพท์เข้าสู่ระบบ ISDN

ขั้นตอนที่ 1 ระหว่างปี 2527-2535 (รวมระยะเวลา 9 ปี)

ดำเนินการแนะนำบริการ ISDN-N (Narrow band ISDN) โดยนำมาทดลองใช้ในเชิงพาณิชย์ในปริมาณจำกัด ในเขตกรุงเทพมหานครและจังหวัดที่สำคัญ โดยมีวัตถุประสงค์คือ

- ศึกษาปฏิกิริยาของผู้ใช้บริการ ทั้งในความสะดวกในการใช้และคุณค่าของบริการเมื่อคำนึงถึงระดับราคา
- ศึกษาหาข้อมูลทางเทคนิคและประสบการณ์ เพื่อนำไปกำหนดเป็นมาตรฐานของประเทศ
- สาธิตบริการต่อประชาชนทั่วไป เป็นการแนะนำบริการ และกระตุ้นให้เกิดความต้องการการใช้บริการ

ขั้นตอนที่ 2 ระหว่างปี 2536-40 (ช่วงนี้จะพอดีกับโครงการโทรศัพท์ 2 ล้านหมายเลขของทีพีเสร็จพอดี)

ดำเนินการขยายโครงข่าย ISDN-N เพื่อให้บริการแก่ธุรกิจขนาดใหญ่และขนาดกลางทั่วประเทศ ทั้งนี้หากการประเมินผลของการแนะนำบริการชี้ให้เห็นถึงความต้องการของธุรกิจขนาดเล็ก และบ้านพักอาศัย ก็จะมีการดำเนินการปรับแปนให้สอดคล้องกับความต้องการของผู้ใช้บริการระดับนี้ด้วย โดยมุ่งที่เขตกรุงเทพ ฯ และจังหวัดสำคัญ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขั้นตอนที่ 3 ระหว่างปี 2541-45 (รวมเวลา 5 ปี)

1. เริ่มดำเนินการขยายบริการ ISDN-N ให้กับธุรกิจและบ้านพักอาศัย
2. เริ่มแนะนำบริการ ISDN-B โดยนำมาทดลองใช้เชิงพาณิชย์ในปริมาณจำกัดและดำเนินการประเมินผล
3. ปรับรูปแบบ ISDN-N สู่รูปแบบของ Intelligence Network
4. ยกเลิกโครงข่ายโทรศัพท์แบบอนาล็อคโดยสมบูรณ์

ขั้นตอนที่ 4 ระหว่างปี 2546-50 (รวมเวลา 5 ปี)

เริ่มเปิดให้บริการ ISDN-U และดำเนินการผนวกประสานโครงข่ายถ่ายทอดสัญญาณโทรทัศน์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. เซมิคอนดักเตอร์ ฉบับ 89 , บริษัท ซี เอ็ดยูเคชั่น จำกัด
2. คอมพิวเตอร์รีวิว ฉบับ 91 , บริษัท แมนการุฟ จำกัด
3. คอมพิวเตอร์รีวิว ฉบับ 93 , บริษัท แมนการุฟ จำกัด
4. เอกสารประกอบ จาก บริษัท ดาต้าแมท จำกัด
5. เอกสารประกอบ จาก องค์การโทรศัพท์แห่งประเทศไทย
6. ISDN Gary C.Kessler : McGraw - Hill , Inc
7. Data Network Concepts, Theory and Practice
   Uyless D.Black : Prentice - Hall International edition

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

PART 2: ASIC Design using AHDL

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

การออกแบบวงจรรวมเฉพาะกิจโดยใช้ AHDL

Application Specific Integrated Circuit Design Using AHDL

(ASIC Design Using AHDL)

โดย

นาย กิตติ เลิศศิริมิตร

น.ส. จารุณี ประทะดี

นาย ธนากร สิริพิเชียร

ปริญญานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิศวกรรมศาสตร์บัณฑิต

สาขาวิศวกรรมคอมพิวเตอร์

สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

17402 PROJECT II

เรื่อง

การออกแบบวงจรรวมเฉพาะกิจโดยใช้ AHDL

Application Specific Integrated Circuit Design Using AHDL

(ASIC Design Using AHDL)

โดย

นาย กิตติ เลิศศิริมิตร 321019

น.ส. จารุณี ปะทะดี 321052

นาย ณนาท สิริพิเชียร 321112

อาจารย์ที่ปรึกษา

ผศ.ดร. บุญสม ศัตรู

ภาควิชาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

สถาบันเทคโนโลยีพระจอมเกล้า เจ้าคุณทหารลาดกระบัง

ภาคการศึกษาที่ 2 ปีการศึกษา 2535

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

คำนำ

AHDL (Hardware Description Language (AHDL) เป็นภาษาที่ใช้ในการบรรยายรูปแบบการทำงานของคุณลักษณะฮาร์ดแวร์ในลักษณะข้อความ โดยในโครงงานนี้ได้ใช้ AHDL ในการสร้างวงจรเกี่ยวกับการประมวลผลภาพ 3 วงจร วงจรแรกคือวงจร Histogram Equalization วงจรที่ 2 คือวงจร Edge Detection และวงจรสุดท้ายคือวงจร Median Filter ซึ่งแต่ละวงจรได้แยกส่วนกันในการออกแบบ

ในรายงานช่วงแรกจะกล่าวถึงรายละเอียดต่างๆ ของ AHDL และช่วงถัดมาจะแสดงรายละเอียดของวงจรทั้ง 3 วงจร ตั้งแต่ ทฤษฎี ขั้นตอนการออกแบบ จนกระทั่งผลที่ได้จากการ simulate ของวงจรออกมา โดยวงจรแรก คือวงจร Histogram Equalization ซึ่งออกแบบโดย นาย กิตติ เลิศศิริมิตร วงจรที่สอง คือวงจร Edge Detection ออกแบบโดย น.ส. จารุณี ปะทะดี และวงจรสุดท้าย คือวงจร Median Filter ซึ่งออกแบบโดย นาย ธนากร สิริพิเชียร

สุดท้ายต้องขอขอบคุณ อาจารย์ที่ปรึกษาอย่างมาก ที่ช่วยให้คำแนะนำ และจัดหาอุปกรณ์ต่างๆ ที่ใช้ในการทำโครงงานนี้ จนกระทั่งโครงงานนี้สำเร็จลุล่วงไปด้วยดี และถ้าหากรายงานฉบับนี้มีข้อผิดพลาดประการใด ทางคณะผู้จัดทำก็ขออภัยมา ณ ที่นี้ด้วย

นาย กิตติ เลิศศิริมิตร

น.ส. จารุณี ปะทะดี

นาย ธนากร สิริพิเชียร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

บทคัดย่อ I

ABSTRACT II

คำนำ III

วัตถุประสงค์ของโครงงาน 1

ตอนที่ 1 AHDL

บทนำ การออกแบบวงจรรวมเฉพาะกิจ 2

บทที่ 1 MAX+PLUS II & AHDL 4

1.1 ลักษณะของ MAX+PLUS II

1.2 ลักษณะของ AHDL

1.3 ประโยชน์ของ AHDL

1.4 Text Design Files (TDFs)

1.5 ไฟล์ใน Project Hierarchy

บทที่ 2 ส่วนประกอบพื้นฐานของ AHDL 11

2.1 รูปแบบของ Text Design Files

2.2 Reserved Keywords

2.3 Symbolic names

2.4 Ports

2.5 Numbers

2.6 Boolean Expressions

2.7 TTL MacroFunctions

2.8 Primitives

บทที่ 3 โครงสร้าง AHDL 20

3.1 Subdesign Section

3.2 Logic Section

3.3 Variable Section

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 Design Section

3.5 Function Prototype Statement

3.6 Title Statement

3.7 Constant Statement

3.8 Include Statement

3.9 Option Statement

บทที่ 4 State Machine 31

4.1 ขั้นตอนในการสร้าง State Machine

บทที่ 5 Report file 36

5.1 Project Information

5.2 Device-Specific Information

5.3 Report file ของ Project count9

บทที่ 6 ตัวอย่างวงจร 44

6.1 วงจร Bus Register

6.2 วงจร Counter

6.3 วงจร Decoder

6.4 วงจร Multiplexer

ตอนที่ 2 การออกแบบวงจร โดยใช้ AHDL

บทที่ 7 วงจร Histogram Equalization 49

7.1 การปรับปรุงคุณภาพของภาพ

7.2 การปรับปรุงภาพโดยใช้ Histogram Modification

บทที่ 8 การออกแบบวงจร Histogram Equalization โดยใช้ AHDL . 63

8.1 การทำงานของวงจร

8.2 ขั้นตอนในการหา Histogram equalization โดยการใช้วงจรรวมที่ออกแบบ

8.3 ขั้นตอนการออกแบบ

8.4 สรุป

8.5 เปรียบเทียบเวลาที่ใช้โดย Software

List program และ ผลการ simulate .....

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9 วงจร Edge Detection 88
9.1 Algorithm
9.2 รายละเอียดของวงจรแต่ละ Module
9.3 รายละเอียดของ Module 1-4
9.4 รายละเอียดของ Module 5
9.5 ข้อสรุปเกี่ยวกับวงจร
List program และ ผลการ simulate .....
บทที่ 10 การออกแบบวงจร Median Filter 11
10.1 ลักษณะของวงจร Median Filter
10.2 Algorithm ของวงจร Median Filter
10.3 โครงสร้างของวงจร
10.4 ลักษณะภายในของวงจร Median Filter
List program และ ผลการ simulate .....
บทวิจารณ์และสรุป 134
บรรณานุกรม 135

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

วัตถุประสงค์ของโครงงาน

1. เพื่อศึกษาการออกแบบวงจรรวมเฉพาะกิจ
2. เพื่อศึกษาการใช้ Hardware Description Language โดยในที่นี้จะใช้ AHDL
3. เพื่อศึกษาวงจร Digital Image Processing
4. ออกแบบวงจร Digital Image Processing เพื่อให้ได้การทำงานของวงจร ที่มีความรวดเร็วกว่าการใช้ซอฟต์แวร์ ทำงาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทนำ

การออกแบบวงจรรวมเฉพาะกิจ

ปัจจุบันเทคโนโลยีทางอิเลคทรอนิคได้มีการพัฒนาไปเป็นอย่างมาก ในการผลิตอุปกรณ์ฮาร์ดแวร์ทางอุตสาหกรรม อุปกรณ์ที่ผลิตขึ้นมีขนาดเล็กลงและมีความซับซ้อนภายใน ที่ประกอบด้วยวงจรขนาดเล็กจำนวนมากมาย ดังนั้นในการออกแบบวงจรรวมขนาดใหญ่มาก [VLSI:Very Large Integrated Circuit] จำเป็นต้องอาศัยวิธีการออกแบบวงจรที่มีประสิทธิภาพ และใช้เครื่องมือ หรืออุปกรณ์ที่ทันสมัย เช่น คอมพิวเตอร์ช่วยในการออกแบบวงจร เพื่อให้ได้ความสะดวก ถูกต้อง ประหยัดเวลา และลดค่าใช้จ่ายให้มากที่สุด ซึ่งในปัจจุบันเทคโนโลยีทางด้านคอมพิวเตอร์นี้ก้าวไกลเป็นอย่างมาก เช่น คอมพิวเตอร์สามารถที่จะคำนวนได้อย่างรวดเร็วกว่าแต่ก่อนมาก รวมทั้งซอฟท์แวร์ (Software) ที่มีให้เลือกใช้ได้อย่างมากมายหลายด้าน ทั้งด้านวิศวกรรม ด้านพาณิชย์ ด้านอุตสาหกรรม ด้านการศึกษา เป็นต้น

ในอดีต การออกแบบวงจรดิจิตอล (Digital) นักออกแบบวงจรส่วนมากมักจะ เริ่มต้นด้วยการออกแบบวงจรในระดับเกท (gate-level) ก่อน ซึ่งเป็นลักษณะของการออกแบบจากล่างขึ้นบน (Bottom-Up design) การออกแบบในลักษณะนี้เหมาะสำหรับวงจรที่มีความซับซ้อนไม่มาก และสามารถทำได้คนเพียงคนเดียวในการออกแบบทั้งหมดได้

แต่ในปัจจุบัน เรากำลังเกี่ยวข้องกับการออกแบบวงจรรวมเฉพาะกิจ [ASIC design: Application Specific Integrated Circuit] หรือการออกแบบวงจรที่มีขนาดใหญ่มากและมีความซับซ้อนเกินกว่าที่คนเพียงคนเดียวจะสามารถจัดการออกแบบได้

อีกทั้งคงจะเป็นการเสียเวลาและไม่เกิดประโยชน์อันใด ถ้ากระบวนการในการออกแบบยังคงดำเนินไป โดยไม่รู้ว่าในแต่ละขั้นตอน หรือในแต่ละส่วนของการออกแบบสามารถทำงานได้ตรงตามที่คาดหวังไว้หรือไม่ เนื่องจากในปัจจุบันชิพวงจรรวมที่มีขนาดใหญ่มากสามารถจุทรานซิสเตอร์ได้มากถึงจำนวนแสนหรือล้านตัว ซึ่งถ้าหากมีความผิดพลาดเกิดขึ้นนั่นย่อมหมายถึงค่าใช้จ่าย และเวลาที่ต้องสูญเสียไป ดังนั้นการออกแบบในแต่ละขั้นตอนเราจะต้องแน่ใจว่าแต่ละส่วนสามารถทำงานได้ถูกต้อง และยังคงทำงานได้ถูกต้องเมื่อนำมาประกอบกันเป็นวงจรรวมขนาดใหญ่

ในช่วงฤดูร้อนของปี 1981 สถาบันการวิเคราะห์สำหรับการป้องกัน (The

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Institute for Defence Analysis) ในสหรัฐอเมริกา ได้จัดตั้งคณะทำงานขึ้นคณะหนึ่งเพื่อ ทำการพัฒนาภาษา ที่ใช้ในการบรรยาย หรืออธิบายรูปแบบการทำงาน และความสัมพันธ์ของอุปกรณ์ฮาร์ดแวร์แบบใหม่ขึ้น ผลการทำงานของคณะทำงานชุดนี้ ได้ก่อให้เกิดภาษาการบรรยายฮาร์ดแวร์ขึ้นเรียกว่า VHDL ซึ่งย่อจาก (VHSIC Hardware Description Language) โดย VHSIC เป็นชื่อย่อของแผนกหนึ่งของสถาบันที่ทำงานเกี่ยวข้องกับวงจรรวมที่มีความเร็วสูงมาก (Very High Speed Integrated Circuit)

ต่อมาในปี 1985 IEEE ได้ทำการผลักดันให้ VHDL กลายเป็นภาษาที่เป็นมาตรฐาน และมีการยอมรับกันอย่างกว้างขวางในวงการอุตสาหกรรมคอมพิวเตอร์

Altera Hardware Description Language (AHDL) คือภาษาที่ใช้อธิบายหรือบรรยายลักษณะพฤติกรรมการทำงานของวงจรและความสัมพันธ์ของอุปกรณ์ฮาร์ดแวร์ในลักษณะของข้อความ เป็นภาษาที่นำมาใช้ในการออกแบบอุปกรณ์ฮาร์แวร์ต่างๆ ทั้งในระดับเกทจนถึงระบบดิจิตอลที่ซับซ้อน เป็นภาษาแบบ high level language ที่มีลักษณะเป็นแบบ modular

AHDL มีลักษณะคล้ายคลึงกับ VHDL (VHSIC Hardware Description Language) ซึ่งเป็นภาษาที่เหมาะสม กับการนำมาใช้ในการเลียนแบบ การทำงานของอุปกรณ์ที่ออกแบบ ทำให้ประหยัดเวลาและลดค่าใช้จ่ายในการออกแบบ นักออกแบบสามารถกำหนดรูปแบบพฤติกรรมการทำงานได้ ทั้งของวงจรดิจิตอลทั่วๆ ไป และในระบบที่แตกต่างออกไป เช่น พฤติกรรมการทำงานของระบบเรดาร์ เป็นต้น

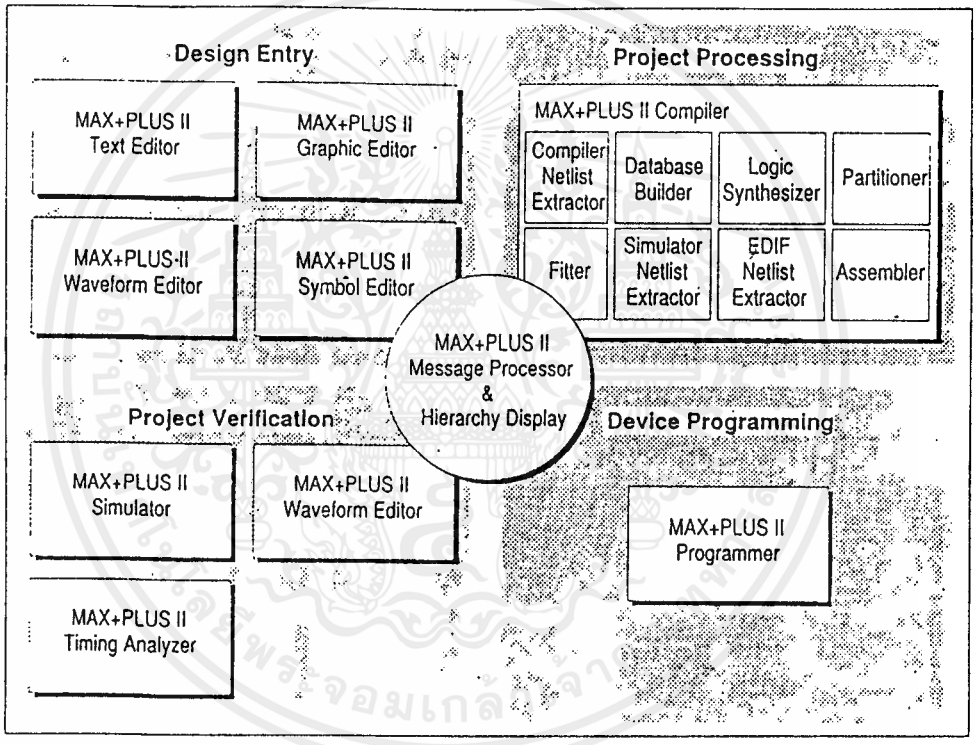
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 1

MAX+PLUS II & AHDL

1.1 ลักษณะของ MAX+PLUS II

MAX+PLUS II เป็น Package software ที่รวบรวม Tools ต่างๆที่ใช้ในการออกแบบวงจร logic โดยสามารถแบ่งเป็นกลุ่ม ได้ดังรูปต่อไปนี้



รูปที่ 1.1 แสดงการแบ่งระบบของ MAX+PLUS II

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Design Entry

เป็นส่วนเริ่มต้นที่ใช้สำหรับออกแบบวงจรโดยมีได้ 4 รูปแบบคือ

1. Text Editor

ใช้สำหรับออกแบบวงจรแบบใช้ AHDL ซึ่งมีลักษณะเป็น Text file

2. Graphic Editor

ใช้สำหรับออกแบบวงจร logic โดยใช้รูปอุปกรณ์ต่างๆ เช่น And gate, Or gate หรือ D-flipflop เป็นต้น

3. Waveform Editor

ใช้สำหรับวาด Waveform ของวงจร

4. Symbol Editor

ใช้สำหรับกำหนด Symbol ของวงจร

- Project Processing

เป็นส่วนที่ใช้ในการประมวลผลวงจรที่เราได้ออกแบบ จาก Design Entry โดยใช้ Compiler เป็นตัวกระทำ

- Project Verification

เป็นส่วนที่ใช้สำหรับวิเคราะห์ผลของวงจรที่เสร็จจากการ compile มีอยู่ 3 ชนิด คือ

1. Simulator

2. Waveform Editor

3. Timing Analyzer

- Device Programming

เป็นขั้นตอนสุดท้ายของ MAX+PLUS II คือการโปรแกรมลงอุปกรณ์ EPLD (Erasable Programmable Logic Devices) ซึ่งมี Programmer เป็นเป็นตัวกระทำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

1.2 ลักษณะของ AHDL

Altera Hardware Description Language (AHDL) เป็นภาษาที่เป็น High-level modular language ซึ่งรวมอยู่ในระบบ MAX+PLUS II คุณสามารถใช้ MAX+PLUS II Text Editor หรือ Text Editor ของคุณเพื่อสร้าง AHDL Text Design files (TDFs) นอกจากนี้คุณยังสามารถ compile และ simulate AHDL TDFs และใช้มันในการโปรแกรม Altera EPLDs ได้

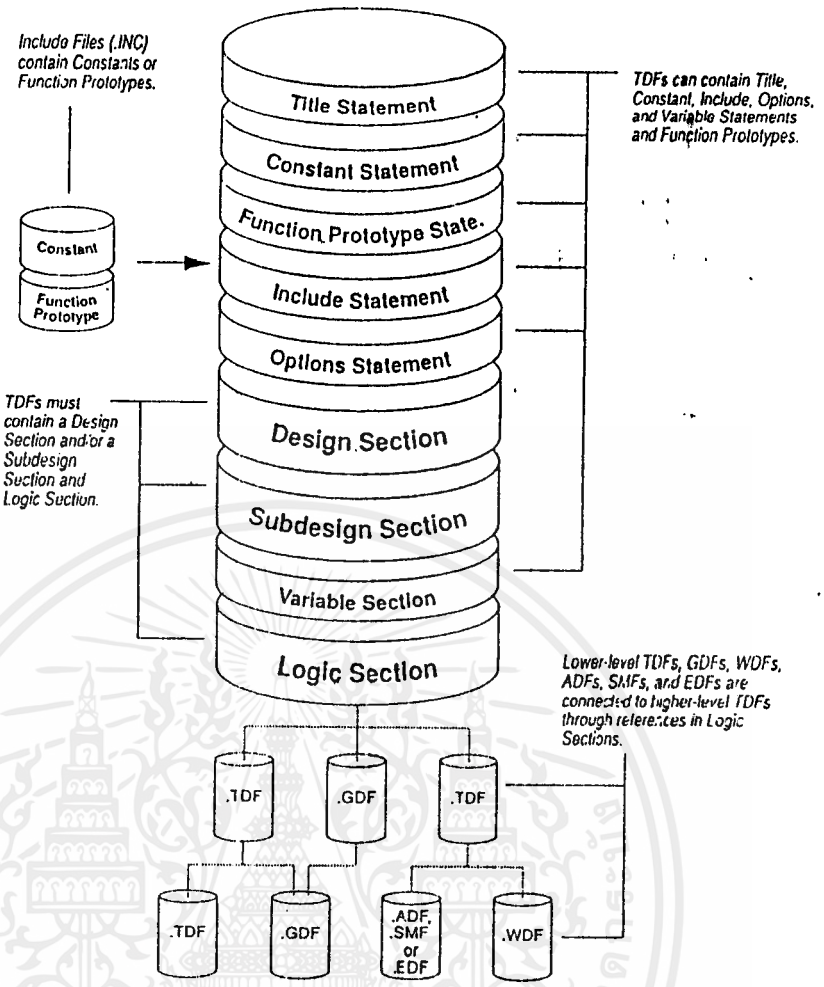
1.3 ประโยชน์ของ AHDL

- สนับสนุนการใช้ state machines, truth tables, boolean equations และ group operation
- สามารถใช้ text, graphic, waveform รวม ๆ กันในการออกแบบ
- ค่าคงที่ และ prototypes ที่มักใช้เสมอ ๆ สามารถเก็บไว้ใน Include files (นามสกุล .inc) และสามารถดึงเข้ามาใช้ใน TDF ได้ นอกจากนี้ MAX+PLUS II ยังเตรียม prototypes สำหรับหน้าที่การทำงานพื้นฐานทุกอย่างไว้ใน MAX+PLUS II TTL Macro Function Library
- อุปกรณ์ต่าง ๆ สามารถกำหนดโดยผู้ใช้ หรือกำหนดโดยอัตโนมัติได้

1.4 Text Design Files

Text Design File (นามสกุล .TDF) เป็น ASCII Text file ที่สามารถสร้างโดยใช้ MAX+PLUS II Text Editor หรือ Text Editor มาตรฐานใดๆ ก็ได้ รูปที่ 1.2 แสดงให้เห็นถึง TDF รวมทั้ง ส่วน (section) และคำสั่ง (statement) ต่างๆ ของมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 1.2 แสดงโครงสร้างของ Text Design File

Text Design File Sections

ส่วน (section) และคำสั่ง (statement) ต่อไปนี้เป็นส่วนประกอบของ TDF แต่ละ TDF ต้องมี

Design section หรือ

Subdesign section และ Logic section

โดยอาจมีเพียงข้อเดียว หรือทั้ง 2 ข้อก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ส่วนต่าง ๆ ของ Text design file

- (optional) Title statement : เป็นคำอธิบายสำหรับ Report file (นามสกุล .RPT) ซึ่งสร้างโดย MAX+PLUS II compiler

- (optional) Constant statement : กำหนดชื่อ(ที่เป็นอักขระ) ซึ่งใช้แทนค่าตงที่

- (optional) Function Prototype statement : ประกาศ ports ของ MacroFunction หรือ Primitive และบอกลำดับของ ports นั้น ๆ ซึ่งจะต้องถูกประกาศใน in-line reference

- (optional) Include statement : กำหนด Include file ที่ใช้แทนคำสั่ง include ใน TDF

- (required) Design section : ใช้กำหนดอุปกรณ์ (device), clique, chip, pin, กำหนด microcell และ logic options

- (required) Subdesign section : ประกาศ input, output และ bidirectional ports ในการออกแบบ

- (optional) Variable section : ประกาศตัวแปรที่ใช้แทนข้อมูลภายใน

- (required) Logic section : ใช้เพื่อนิยาม logical operations ในการออกแบบ

หมายเหตุ Optinal หมายถึง เป็นส่วนที่จะมีในไฟล์ .TDF หรือไม่ก็ได้
Required หมายถึง ส่วนที่จะต้องมีในทุก ๆ .TDF ไฟล์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

AHDL เป็นภาษาที่มีลักษณะเป็น concurrent language กล่าวคือ พฤติกรรมต่าง ๆ ที่ถูกกำหนดไว้ใน logic section จะถูกประมวลผลพร้อม ๆ กัน สมการต่าง ๆ ที่ให้ค่าหลาย ๆ ค่าแก่ AHDL node เดียวกัน หรือตัวแปรเดียวกันจะถูกเชื่อมต่อกันแบบ logical (Or ถ้าเป็น active high, And ถ้าเป็น active low)

TDF จะต้องประกอบด้วย Design section หรือ Subdesign section/ Logic section หรือทั้ง 2 อย่าง นอกจากนี้ส่วนอื่น ๆ อาจมีหรือไม่ก็ได้

Design section จะบอก architectural description ของการออกแบบ และจะอยู่ถัดจาก Title, Include, Constant, Function Prototype และ Options statements

ส่วนท้ายของ TDF จะเป็น Subdesign section, Variable section และ Logic section ซึ่งจะใช้ประกอบกันเพื่ออธิบายถึง behavioral description ของการออกแบบ

1.5 ไฟล์ใน Project Hierarchy

โปรแกรม MAX+PLUS II นั้น อนุญาตให้เราสามารถสร้างวงจรขึ้นมาวงจรหนึ่งได้หลายวิธี อาจสร้างโดย AHDL หรือ Graphic Editor หรือ WaveForm Editor ก็ได้

ในโครงงานหนึ่ง ๆ อาจประกอบด้วยไฟล์หลาย ๆ ไฟล์ โดยแต่ละไฟล์นั้นอาจถูกสร้างขึ้นโดยวิธีต่าง ๆ กัน เช่น อาจสร้างโดย AHDL หรือ Graphic Editor หรือ WaveForm Editor ฯลฯ นอกจากนี้ ในโครงงานของเรา เราอาจมีไฟล์อยู่ในระดับต่าง ๆ กัน โดยไฟล์ที่อยู่ในระดับสูง จะเรียกใช้งานไฟล์ที่อยู่ในระดับต่ำลงไปได้ เราเรียกไฟล์ต่าง ๆ ที่ประกอบกันเป็นโครงงานนั้น ๆ ว่า ลำดับชั้นของโครงงาน (Project Hierarchy)

ไฟล์ซึ่งอยู่ในลำดับชั้นของโครงงาน (Project hierarchy) (เช่น MacroFuctions) ที่สามารถมีได้คือ TDFs,Graphics Design Files (GDFs), Waveform Design Files (WDFs), Altera Design Files (ADFs), State Machine Files (SMFs) หรือ EDIF Input Files (นามสกุล .EDF) แต่ละ file (MacroFunction) จะถูกเชื่อมต่อผ่านทาง input, output ports ของมัน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ไปยังการออกแบบที่สูงขึ้นไป

Function Prototype Statement (คือคำสั่งที่ใช้ในการประกาศ Input, Output ของ MacroFunction) จะต้องรวมอยู่ใน TDF ปัจจุบันเพื่อใช้ในการเรียก MacroFunction ในระดับที่ต่ำกว่า คุณสามารถใส่ Function Prototype Statement นี้ไว้ใน Include Files แล้วเรียกใช้มันโดย Include statement ก็ได้

Include Files

Include file เป็น ASCII Text file ที่มีนามสกุล .INC เมื่อพบ Include statement ใด ๆ ใน TDF ก็จะนำเอาข้อความใน Include file เข้าไปแทนที่ ข้อความใน Include file สามารถเป็นอะไรก็ได้ แต่ทางบริษัท Altera แนะนำให้เป็นเพียง Constant หรือ Function Prototype statements

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 2

ส่วนประกอบพื้นฐานของ AHDL

2.1 รูปแบบของ Text Design File

AHDL file เป็น ASCII text file ที่มีนามสกุล .TDF สามารถสร้างขึ้นจาก MAX+PLUS II Text Editor หรือจาก text editor มาตรฐานใด ๆ ก็ได้ แต่ TDF file จะต้องมีข้อกำหนดทางประการดังนี้

~ แต่ละบรรทัดยาวได้ไม่เกิน 255 อักขระ

~ ในส่วนที่เป็นช่องว่าง (space) เราสามารถขึ้นบรรทัดใหม่ได้โดยไม่มีผลต่อความหมายของโปรแกรม AHDL

~ Keywords, ชื่อ, และตัวเลข จะต้องถูกคั่นด้วยสัญลักษณ์ หรือ operators ที่เหมาะสม

~ ตัวอักษรใหญ่ หรือเล็กใน AHDL นั้น มีความหมายเหมือนกัน

~ คำอธิบาย (Comment) จะต้องถูกคลุมด้วย %

2.2 Reserved Keywords

Reserved keywords นั้นใช้ในการ

~ เริ่ม, จบ, และเปลี่ยนคำสั่ง

~ กำหนดทางเลือกต่าง ๆ

~ กำหนดค่าคงที่

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตารางแสดง Reserved Keywords

Reserved Keywords		
AND	IF	OUTPUT
ANY	INCLUDE	PTERM_ALLOC
BEGIN	INPUT	RETURNS
BIDIR	IS	STATES
BITO	LSB	SECURITY
BITS	MACHINE	SUBDESIGN
BURIED	MACRO	THEN
CASE	MSB	TITLE
CLIQUE	NAND	TABLE
CONSTANT	NODE	TURBO
DEFAULTS	NOR	VARIABLE
DESIGN	NOT	VCC
DEVICE	OF	WHEN
ELSE	OFF	WITH
ELSIF	ON	X
END	OPTIONS	XNOR
FUNCTION	OR	XOR
GND	OTHERS	

2.3 Symbolic names

เป็นชื่อที่ผู้ใช้กำหนดขึ้นมาเอง โดยจะกำหนดขึ้นมาเพื่อใช้แทน

~ Internal และ External nodes

~ ค่าคงที่

~ ใช้แทน ตัวแปร, บิท, สถานะ, และ ชื่อ ของ state machine

Symbolic names มี 2 ประเภทคือ แบบ unquoted และ quoted

Unquoted Symbolic Names

ได้แก่ชื่อที่ประกอบด้วยอักษร A-Z,a-z และ underscore(_) ซึ่งชื่อที่ตั้งขึ้นจะต้องไม่ซ้ำกับ keywords สามารถมีตัวเลขได้ แต่ต้องไม่เป็นตัวเลขทั้งหมด

Quoted Symbolic Names

ได้แก่ symbolic names ที่ถูกคลุมด้วยเครื่องหมายอัญประกาศ (')

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.4 Ports

Ports เป็นตัวแปรที่ต่อเชื่อมกับ inputs หรือ outputs ของ primitive หรือ MacroFunctions

ตัวอย่างเช่น เรากำหนดตัวแปร cap เป็นชนิด DFF (D flip flop) เราสามารถอ้างถึง ports คือ cap.d, cap.clk, cap.clrn, cap.q ได้

Groups

เราสามารถรวมเอา names หรือ ports เข้าเป็นกลุ่มเดียวกันได้ เรียกว่า group

ตัวอย่างของ group เช่น a[4..1] ประกอบด้วยสมาชิก 5 ตัว เราอาจเขียน group ข้างต้นได้อีกแบบเป็น (a4, a3, a2, a1) และการใช้ a[] หมายถึง a[4..1]

2.5 Numbers

Numbers คือค่าตัวเลข อาจเป็นได้ทั้งฐาน 2,8,10,16 เช่น B"10001" คือ เลขฐาน 2 ที่มีค่า 10001

ไวยากรณ์ของการกำหนดจำนวนเป็นเลขฐานต่าง ๆ

เลขฐาน 10		< series of digits 0 ถึง 9 >
เลขฐาน 2		B"<series of 0's,1's,X's>"
		(X คือ don't care)
เลขฐาน 8		O"<series of digits 0 ถึง 7>" หรือ
		Q"<series of digits 0 ถึง 7>"
เลขฐาน 16		X"<series of digits 0 ถึง 9,A ถึง F>"
	หรือ	H"<series of digits 0 ถึง 9,A ถึง F>"

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.6 Boolean Expressions

Boolean expressions ประกอบด้วย operands (symbolic names, ports, groups, หรือ ค่าคงที่) ซึ่งถูกคั่นด้วย logical หรือ arithmetic operators หรือ comparators

Logical operators

! หรือ NOT	ใช้แทน	one's complement (prefix inverter)
& หรือ AND	ใช้แทน	AND
!& หรือ NAND	ใช้แทน	AND inverter
# หรือ OR	ใช้แทน	OR
!# หรือ NOR	ใช้แทน	OR inverter
$ หรือ XOR	ใช้แทน	Exclusive or
!$ หรือ XNOR	ใช้แทน	Exclusive NOR

Arithmetic Operators

+,- (unary)	ใช้แทน	ค่าบวกและค่าลบ
+,-	ใช้แทน	การบวกและการลบ

Comparators

==	ใช้แทน	เท่ากับ
!=	ใช้แทน	ไม่เท่ากับ
<	ใช้แทน	น้อยกว่า
>	ใช้แทน	มากกว่า
<=	ใช้แทน	น้อยกว่า หรือเท่ากับ
>=	ใช้แทน	มากกว่า หรือเท่ากับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Priority

ลำดับของการทำ operators ต่าง ๆ นั้น จะทำตามลำดับความสำคัญ (priority) ดังนี้

Priority	Operator
1	- (unary)
1	! (not)
2	+ (การบวก)
2	- (การลบ)
3	==
3	!=
3	<
3	<=
3	>
3	>=
4	& (AND)
4	!& (NAND)
5	$ (XOR)
5	!$ (XNOR)
6	# (OR)
6	!# (NOR)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

2.7 TTL MacroFunctions

MAX+PLUS II TTL,bus,EPLD-optimized MacroFunctions เป็นวงจรสำเร็จรูปที่สามารถนำมาใช้ร่วมกับวงจรตรรกะพื้นฐานเพื่อการออกแบบวงจรตรรกะต่าง ๆ ได้

MacroFunctions ของ MAX+PLUS II มีมากมาย เช่น Adder, Arithmetic Logic Unit, Counter, Decoder, Encoder, Frequency Divider, Converter, Comparater, Multiplier, Multiplexer, Register ฯลฯ

2.8 Primitives

MAX+PLUS II มีฟังก์ชันพื้นฐานต่าง ๆ เพื่อการออกแบบวงจร ฟังก์ชันพื้นฐานของ MAX+PLUS II มีดังนี้

Buffer Primitives

ช่วยให้คุณสามารถควบคุมขบวนการสังเคราะห์ตรรกะ (logic synthesis process) ได้ โดยทั่วไปคุณไม่จำเป็นต้องใช้ Buffer เหล่านี้ จะใช้กรณีที่วงจรของคุณซับซ้อนมาก ๆ จน Complier ไม่สามารถจัดการได้ ซึ่งในกรณีนี้คุณอาจต้องเพิ่ม Buffer ตัวใดตัวหนึ่งเข้าไปในวงจร เพื่อเป็นแนวทางให้ Logic Synthesizer สร้างผลลัพธ์บางอย่างออกมา

Buffer primitives มีดังนี้

~ EXP = Expander Buffer

~ MCELL = Macrocell Buffer

~ GLOBAL = Global Buffer

~ SCLK = Synchronous Clock Buffer

~ SOFT = Soft Buffer

~ TRI = Tri-State Buffer

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Flipflops & Latch

MAX+PLUS II มี Flipflop และ latch พื้นฐานแทบทุกชนิด เช่น LATCH, DFF,DFFE,JKFF,JKFFE,SRFF,SRFFE,TFF,TFFE

ตารางแสดง Filpflops และ Latch

Primitive	AHDL Function Prototype
LATCH	FUNCTION LATCH (d, ena) RETURNS (q);
DFF	FUNCTION DFF (d, clk, clrn, prn) RETURNS (q);
DFFE	FUNCTION DFFE (d, clk, clrn, prn, ena) RETURNS (q);
JKFF	FUNCTION JKFF (j, k, clk, clrn, prn) RETURNS (q);
JKFFE	FUNCTION JKFFE (j, k, clk, clrn, prn, ena) RETURNS (q);
SRFF	FUNCTION SRFF (s, r, clk, clrn, prn) RETURNS (q);
SRFFE	FUNCTION SRFFE (s, r, clk, clrn, prn, ena) RETURNS (q);
TFF	FUNCTION TFF (t, clk, clrn, prn) RETURNS (q);
TFFE	FUNCTION TFFE (t, clk, clrn, prn, ena) RETURNS (q);

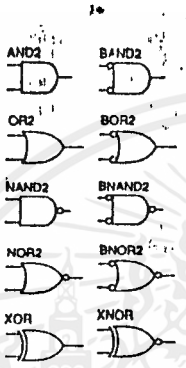
Notes:

clk	= Register Clock Input
clrn	= Clear Input
d, j, k, r, s, t	= Input from Logic Array
ena	= Latch Enable Input
prn	= Preset Input
q	= Output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Logic Primitives

Logic primitives ของ MAX+PLUS II มีมากมายหลายชนิดดังแสดงในตาราง

Sample Symbols	Description
	Name: AND2, AND3, AND4, AND6, AND8, AND12 OR2, OR3, OR4, OR6, OR8, OR12 NAND2, NAND3, NAND4, NAND6, NAND8, NAND12 NOR2, NOR3, NOR4, NOR6, NOR8, NOR12 BAND2, BAND3, BAND4, BAND6, BAND8, BAND12 BOR2, BOR3, BOR4, BOR6, BOR8, BOR12 BNAND2, BNAND3, BNAND4, BNAND6, BNAND8, BNAND12 BNAND2, BNAND3, BNAND4, BNAND6, BNAND8, BNAND12 BNOR2, BNOR3, BNOR4, BNOR6, BNOR8, BNOR12 Description: Output: OUT = logical <*name*> of inputs Input: IN1, IN2, ...IN12 = 2, 3, 4, 6, 8, 12 inputs
	Name: NOT Description: Output: OUT = inverse of input Input: IN1 = 1 input
	Name: GND Description: Assigns a node to GND
	Name: VCC Description: Assigns a node to VCC

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Input & Output Primitives

เราสามารถกำหนด Input และ Output ของวงจรเราได้โดยใช้ Input & Output primitives

ตารางแสดง Input และ Output Primitives

Pin Port	Description	Pin Port Declaration
INPUT	Input pin	<pin_name> : INPUT
OUTPUT	Output pin	<pin_name> : OUTPUT
BIDIR	Bidirectional pin	<pin_name> : BIDIR

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

โครงสร้างของ AHDL

จากที่กล่าวมาแล้วว่า ในการออกแบบ AHDL นั้น ส่วนประกอบที่ต้องมีอย่างน้อยที่สุด คือ Subdesign Section และ Logic Section ส่วนอื่น ๆ อาจมีหรือไม่ก็ได้ และในระดับบนสุดของลำดับชั้นในการออกแบบ (hierarchy) สามารถมี เพียง Design section เพียงส่วนเดียวได้

ตัวอย่าง วงจรที่ออกแบบโดยใช้ AHDL

```
%*****************************************
   KEYBOARD is a keyboard encoder
   for a 16-key calculator style
   keyboard.  It generates a 4-bit
   key code and a strobe signal
   whenever a key is pressed.  2-key
   rollover is provided.
   The keyboard should have SPST
   normally-open switches arranged
   in a 4-by-4 matrix.
*****************************************%

DESIGN IS "Jumbo Keyboard"
     DEVICE IS EPM5032;

INCLUDE "74151.inc";
INCLUDE "74154.inc";
INCLUDE "4count.inc";
```

Comment ถูกปิดด้วย %

Design Section

include statement

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FUNCTION opencol (in) RETURNS (out);
FUNCTION debounce (clk, key_pressed)
     RETURNS (pulse);

SUBDESIGN  keyboard
(
     clk        :  INPUT;  % 50-KHz clock %
     col[3..0]  :  INPUT;  % keyboard columns %
     row[3..0]  :  OUTPUT; % keyboard rows %
     d[3..0]    :  OUTPUT; % key code %
     strobe     :  OUTPUT; % key code is valid %
)
VARIABLE
     keypressed : NODE;
     mux        : 74151;
     decoder    : 74154;
     counter    : 4count;

BEGIN
     % Drive keyboard rows with a decoder and %
     % open-collector outputs.  %
     row0 = opencol (decoder.o0n);
     row1 = opencol (decoder.o1n);

     row2 = opencol (decoder.o2n);
     row3 = opencol (decoder.o3n);
     decoder.(b,a) = counter.(qd,qc);
```

Function Prototype Statement

Subdesign Section

Variable Section

Logic Section

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
% Sense keyboard columns with a multiplexer. %
mux.d[3..0] = col[3..0];
mux.(b,a) = counter.(qb,qa);
keypressed = !mux.y;    % Pressed keys pull %
                        % columns low.  %

% Scan keyboard until a key is pressed. %
% Drive key's code onto d[] outputs. %
counter.clk = clk;
counter.cin = !keypressed;
d[] = counter.(qd,qc,qb,qa);

% Generate strobe when key has settled. %
strobe = debounce (clk, keypressed);
END;
```

3.1 *Subdesign Section*

ใช้ในการประกาศ input,output และ bidirectional ports

ตัวอย่างของ Subdesign section :

```
SUBDESIGN  top
(
     foo, bar, clk    : INPUT = VCC;
     a0, a1, a2       : OUTPUT;
     B[7..0]          : BIDIR;
)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการประกาศว่า ชื่อของ subdesign คือ top และ foo,bar,clk เป็นสัญญาน input; a0,a1,a2 เป็นสัญญาน output ส่วน B[7..0] เป็น bidirectional bus(เป็น bus ขนาด 8 บิท ที่สามารถ input หรือ output ข้อมูลก็ได้)

3.2 *Logic Section*

เป็นส่วนที่ใช้กำหนดการกระทำเกี่ยวกับตรรกะของการออกแบบLogic section นี้ถือได้ว่าเป็น body ของ Subdesign section เป็นส่วนที่จำเป็นต้องมีเสมอ

ใน Logic section ประกอบด้วยคำสั่งต่าง ๆ มากมายมาประกอบกัน ได้แก่

~ Boolean equations

~ Truth Table Statement

~ Case Statement

~ If statement

~ Default Statement

~ In-line MacroFunction หรือ primitive reference

เราจะใช้คำสงวน คือ BEGIN และ END; เพื่อปิดหัวท้ายของ Logic Section นอกจากนี้ มีสิ่งที่ต้องระวังคือ AHDL เป็น concerrent language กล่าวคือ คำสั่งทุกคำสั่งใน Logic section จะถูกประมวลผลพร้อม ๆ กัน (แทนที่จะถูกประมวลผลตามลำดับก่อนหลัง) และสมการหลาย ๆ สมการ ที่ให้ค่ากับจุด ๆ หนึ่ง (node) หรือตัวแปรมากกว่า 1 ค่า ค่าเหล่านั้นจะถูก logically OR กัน

Boolean Equation

ใช้กำหนดการเชื่อมสายต่าง ๆ ในการออกแบบ

ตัวอย่างของ Boolean Equation

```
a[] = ( (c[] & B"001101") + e[6..1] ) # (p,q,r,s,t,v);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Truth Table Statement

เป็นอีกวิธีในการกำหนด พฤติกรรมทางตรรกะ ใน Truth table แต่ละบรรทัด จะประกอบด้วยกลุ่มของ input ที่จะให้ค่า output ที่ตรงกันออกมา

ตัวอย่าง

```
TABLE
     a0,  f[4..1].q      =>  f[4..1].d, control;

     0,   H"0"           =>  H"1",      1;
     0,   H"4"           =>  H"2",      0;
     1,   B"0XXX"        =>  H"4",      0;
     X,   H"F"           =>  H"5",      1;
END TABLE;
```

บรรทัดแรก เป็นการบอกว่าสัญญาน input คือ a0 และ f[4..1].q ส่วนสัญญาน output คือ f[4..1].d และ control

ส่วนบรรทัดถัดมาเป็น การบอกว่าเมื่อมี input ค่าเท่านี้แล้ว จะให้ค่า output เป็นเท่าไร เช่น เมื่อค่า a0 = 0 และ f[4..1].q = H"0" แล้ว จะให้ค่า f[4..1].d เป็น H"1" และ control เป็น 1

Case Statement

เป็นการบอกทางเลือกที่จะทำซึ่งขึ้นอยู่กับค่าของ ตัวแปร, group, expression ที่ ตามหลัง คำสงวน CASE

ตัวอย่าง

```
CASE f[].q  IS
     WHEN  H"00"   =>
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            addr[] = 0;
            s = a & b;
        WHEN  H"01"   =>
            count[].d = count[].q + 1;
        WHEN  OTHERS  =>
            f[].d =  f[].q
    END CASE;
```

เป็นการบอกว่าเมื่อตัวแปร f[].q มีค่าต่าง ๆ กันจะมีการทำงานที่แตกต่างกัน เช่น ถ้า f[].q = H"01" จะทำคำสั่ง count[].d = count[].q + 1;

ส่วน WHEN OTHERS เป็นการกำหนดว่าค่าทางเลือกที่ไม่ตรงกับทางเลือกอื่นทางบนเลย ให้มาที่ WHEN OTHERS

IF Statement

ใช้กำหนดทางเลือก

ตัวอย่าง

```
IF (a[] == b[])  THEN
    c[8..1] = H"77";
    addr [3..1] = f [3..1].q;
    f[].d = addr[] +1;
ELSEIF (g3 $ g4) THEN
    f[].d = addr[];
ELSE d =VCC;
END IF;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Defaults Statement

คำสั่งนี้จะทำให้คุณสามารถกำหนดค่า default ให้กับตัวแปรที่ใช้ในคำสั่ง Truth table, IF และ Case ได้

ตัวอย่าง

```
DEFAULTS
    a = VCC;
END DEFAULTS;

IF x & y THEN
    a = GND;        % a is active low %
END IF;
```

จากตัวอย่าง เป็นการกำหนดให้ตัวแปร a มีค่าปกติเป็น VCC โดยทั่วไปถ้าเราไม่กำหนดค่า default ให้กับตัวแปรแล้ว โปรแกรมจะกำหนดให้ค่าปกติของตัวแปรเป็น GND (ground)

In-line MacroFunction หรือ Primitive Reference

In-line MacroFunction หรือ primitive reference เป็น Boolean equation ที่มีการอ้างถึง MacroFunction หรือ primitive

ตัวอย่าง เรากำหนด Function prototype ที่มี input คือ green และ yellow และ output คือ blue,red และ cycle

```
FUNCTION moonbeam (green, yellow)
    RETURNS (blue, red, cycle);
```

~ รายละเอียดของ Function Prototype จะกล่าวถึงภายหลัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จาก Function Prototype เราสามารถเขียน in-line MacroFunction เพื่อกล่าวถึง moonbeam ได้ดังต่อไปนี้

```
(sun3, , sun1) = moonbeam (color4, color3)
```

จากตัวอย่างนี้ input ของ moonbeam คือ green และ yellow จะถูกเชื่อมด้วยตัวแปรที่ชื่อ color4 และ color3 ตามลำดับ ส่วน output ของมันคือ blue และ cycle จะถูกเชื่อมกับ sun3 และ sun1 ตามลำดับ ส่วน red นั้นไม่ได้ถูกเชื่อมกับตัวแปรใด และเครื่องหมายจุลภาค (,) ที่เพิ่มขึ้นมาทางฝั่งซ้ายของสมการนั้น มีเพื่อให้จำนวนสมาชิกใน group เท่ากับ output ของ moonbeam

3.3 *Variable Section*

เป็นส่วนที่ใช้ประกาศค่าตัวแปรซึ่งจะถูกใช้ใน Logic section ตัวแปรของ AHDL มีลักษณะคล้ายกับตัวแปรของภาษาระดับสูงอื่น ๆ มันถูกใช้ในการกำหนดตรรกะภายใน (buried logic, internal logic)

ตัวอย่าง

```
VARIABLE
    pen, pencil, eraser  : NODE;
    temp                 : HALFADD;
```

เป็นการประกาศว่า pen,pencil,และ eraser เป็นตัวแปรชนิด NODE และ temp เป็นตัวแปรที่เป็น MacroFunction ที่ชื่อ HALFADD

ชนิดของตัวแปรใน Variable section ที่สามารถเป็นได้คือ NODE, <primitive>,<macrofunction>, หรือ <statemachine>

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.4 *Design Section*

ในการโปรแกรมข้อมูลลง EPLDs นั้น เราสามารถกำหนดได้ว่าจะให้ pin ใดของ EPLD แทน INPUT ,OUTPUT ของวงจรตรรกะของเรา ในการกำหนดนี้เราสามารถทำได้ในส่วนของ Design section

Design section เป็นส่วนที่มีหรือไม่ก็ได้ ถ้าไม่มี โปรแกรมจะทำการเลือก EPLDs ที่เหมาะสม และจัดวงจรของเราลง EPLDs ให้โดยอัตโนมัติ

ตัวอย่าง

```
DESIGN IS cnt4
BEGIN
     DEVICE IS EMP5016
     BEGIN
          in_1 @ 7, in_2 @ 8, in_3 @ 9   : INPUT;
          out @ 11                       : OUTPUT;
          !my_sub|sub_nd @ mc10          : BURIED;
     END;
END;
```

3.5 *Function Prototype Statement*

Function prototype statement นั้นคล้ายกับการประกาศชื่อฟังก์ชันในภาษาระดับสูงอื่นๆ (แต่ใน AHDL จะใช้คำว่า MacroFunciton แทนคำว่า function) โดยเราจะบอก Input,Output ของ MacroFunction นั้น ๆ ส่วนการทำงานภายในของมันจะถูกกำหนดไว้ก่อนแล้วในไฟล์อีกไฟล์หนึ่ง

ตัวอย่าง

```
FUNCTION  moonbeam (green, yellow)
     RETURNS (blue, red, cycle);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เป็นการบอกว่า MacroFunction นี้ชื่อ moonbeam มี input คือ green และ yellow และมี output คือ blue,red,cycle

3.6 *Title Statement*

คำสั่งนี้ไม่มีผลต่อการทำงานของโปรแกรม มีไว้เพื่อบอกชื่อในการออกแบบ โดย complier ของ MAXPLUS จะนำข้อความใน Title statement ไปใส่ไว้ในตอนต้นของ report file (.rpt)

ตัวอย่าง

```
TITLE "Octopus Design";
```

3.7 *Constant Statement*

Constant statement ช่วยให้คุณสามารถแทนค่าคงที่ด้วยชื่อที่มีความหมาย

ตัวอย่าง

```
CONSTANT  UPPER_LIMIT = B"110";
```

3.8 *Inlucde Statement*

Include statement อนุญาตให้คุณสามารถดึง text file จากไฟล์อื่นเข้ามาในไฟล์ของคุณได้

ตัวอย่าง

```
INCLUDE "const.inc";
```

การทำงานเมื่อ Complier พบคำสั่งนี้ก็จะทำการดึงเข้าข้อมูลซึ่งอยู่ในไฟล์ชื่อ const.inc เข้ามาแทนที่คำสั่ง Include นี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

โดยทั่วไปแล้วข้อมูลในไฟล์ .inc ควรจะมีเพียงแค่ Function Prototype และ Constant Statement

3.9 *Option Statement*

หน้าที่ของ Option statement คือ

- ใช้ set Turbo และ Security Bits ของ Device subsection ที่อยู่ถัดจากมัน

  Turbo bit และ Security bit เป็นการกำหนดคุณสมบัติของ EPLDs
- กำหนดว่า bit ที่ต่ำที่สุดของ group จะเป็น most significant bit (MSB) หรือ least significant bit (LSB)
- กำหนดว่า product terms ควรจะถูก allocated หรือไม่

ตัวอย่าง

```
OPTIONS   SECURITY  =  ON, TURBO = OFF, BITO = MSB;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 4

State Machines

AHDL นั้นอนุญาตให้เราสามารถสร้าง State Machine ได้อย่างง่ายดายเพียงแค่กำหนด State ว่ามี states อะไรบ้าง และกำหนด state transition

ตัวอย่าง State Machine

```
%****************************
*                            *
  Stepper Motor Controller
*                            *
****************************%

TITLE "Super Stepper";

SUBDESIGN  stepper
(
    sysclk    : INPUT;  % system clock %
    reset     : INPUT;  % system reset %
    ccw       : INPUT;  % rotate counter clockwise %
    cw        : INPUT;  % rotate clockwise %
    power_on  : INPUT;  % apply power to stepper %
                        % controller %
    phase[3..0] : OUTPUT; % drives lines for  %
                          % the stepper motor %
)
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
VARIABLE

    % Declare state machine for stepper controller. %
    % Assign each state a "rotating bit" pattern,   %
    % which drives phase lines of stepper motor.    %
    step : MAHCINE      <------ State Machine Declaration
        OF BITS ( q[3..0] )    <----- Bits declaration
        WITH STATES (s0 = B"1000",  <-- State declaration
                     s1 = B"0100",       and assignment to
                     s2 = B"0010",       binary numbers
                     s3 = B"0001" );

BEGIN
    % Connect state bits to output pins  %
    % if power_on is GND, all output lines will be GND %
    phase[] = q[] & power_on;

    % Connect state machine clock and reset lines %
    step.clk   = sysclk;  <---  State control equations
    step.reset = reset;

    % Define state transitions %
    CASE ( step ) IS
        WHEN s0 =>
            IF ( ccw & power_on ) THEN
                step = s1;          % Turn counter %
                                    % clockwise %
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            ELSEIF ( cw & power_on ) THEN
                step = s3;          % Turn clockwise %
            END IF;                 % else hold %
        WHEN s1 =>
            IF ( ccw & power_on ) THEN
                step = s2;
            ELSEIF ( cw & power_on ) THEN
                step = s0;
            END IF;
        WHEN s2 =>
            IF ( ccw & power_on ) THEN
                step = s3;
            ELSEIF ( cw & power_on ) THEN
                step = s1;
            END IF;
        WHEN s3 =>
            IF ( ccw & power_on ) THEN
                step = s0;
            ELSEIF ( cw & power_on ) THEN
                step = s2;
            END IF;
    END CASE;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

4.1 ขั้นตอนในการสร้าง State Machine

ในการสร้าง State machine นั้น มี 3 ขั้นตอนหลัก ๆ คือ

1 ประกาศ State machine ไว้ในส่วน Variable section

2 กำหนด control equations ในส่วน Logic section

3 กำหนด State transition ในส่วน Logic section

State Machine declaration

ในการสร้าง state machine นั้น เราต้องประกาศ state machine ไว้เสียก่อนใน Variable section

ตัวอย่าง การประกาศ state machine

```
VARIABLE
    SS : MACHINE OF BITS  ( q1,q2,q3 )
         WITH STATES ( s1 = B"000",
                       s2 = B"010",
                       S3 = H"7" );
```

Control Equations

Control equations เป็นสมการบูลลีนที่ถูกกำหนดใน Logic section เพื่อใช้ set up State mahcine และใช้ reset สัญญาน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง

```
ss.clk = clk1;
ss.reset = a & b;
ss.ena = clklena;
```

State Transitions

State Transitions คือการกำหนดว่า เมื่อค่า State ของ State Machine เป็นค่าหนึ่ง และ Input เป็นค่าหนึ่ง State machine นั้น ๆ จะให้ค่า output เป็นเท่าไร และจะเปลี่ยนค่า state ไปเป็น state ใด

เราสามารถใช้คำสั่งใด ๆ กำหนด state transitions ก็ได้ แต่คำสั่งที่แนะนำให้ใช้ คือ Case และ Truth table

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 5

Report File

Report file เป็น ASCII file ที่มี extension คือ .RPT เป็น file ที่ได้มาหลังจากการ compile เสร็จเรียบร้อยแล้ว มีข้อมูลต่างๆ ที่เกี่ยวกับวงจร เช่น input, output, buried logic เป็นต้น

Report file จะแบ่งออกเป็น 2 ส่วนดังนี้

1. Project Information
2. Device-specific information

โดยที่ Project Information จะอยู่ในส่วนแรกของ Report file และ Device-specific จะเป็นข้อมูลที่ใช้สำหรับ chip แต่ละตัวใน project

5.1 Project Information

บรรทัดแรกของ Report file จะมีข้อความอันใดอันหนึ่งดังนี้

- Project compiled without errors
- Project compiled with errors

ซึ่งข้อความข้างต้นนี้เป็นข้อความที่ใช้บอกความผิดพลาดของ Project ถ้าไม่มีความผิดพลาดเราจะสามารถโปรแกรม Project นี้ลงใน EPLD ได้ ถ้ามี errors เราต้องแก้ไขให้ถูกต้องและทำการ compile ใหม่

บรรทัดถัดมาจากข้อความข้างต้น จะเป็นข้อมูลที่มีการสรุปส่วนต่างๆ ดังนี้

- Device Summary

เป็นข้อมูลที่สรุปเกี่ยวกับอุปกรณ์ที่ถูกใช้ใน Project

- Auto Global Signals Section

ส่วนนี้จะเป็นการแจกชื่อของ node ที่ใช้สำหรับ global Clock,Clear Preset,Output ซึ่งถูกกำหนดโดย Logic Synthesis

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Multiple Pin Connections Section

ส่วนนี้จะแสดง pin ที่ถูกเชื่อมต่อระหว่างกันใน Project

- Pin/MC/Chip Assignments Section

ส่วนนี้จะแสดงรายการของการกำหนด pin, macrocell และ chip ที่ได้ออกแบบไว้ใน file วงจรของเรา

- Clique Assignments Section

จะแสดงรายการของ clique และ node ที่ได้ถูกกำหนดไว้

- Probes Section

จะแสดงรายชื่อของ probe ต่างๆ และ primitive หรือ macrocell

- State Machine Assignments Section

ส่วนนี้จะแสดง state machine ที่ถูกกำหนดโดย Logic Synthesizer สำหรับ Waveform Design Files หรือ AHDL Text Design File

- File Hierarchy Section

จะแสดงลำดับชั้นต่างๆ ของชื่อ node ของ primitive หรือ macrocell แต่ละตัว ที่ถูกใช้ในลำดับชั้นต่างๆ ของ Project

5.2 Device-Specific Information

บรรทัดแรกของส่วนนี้ จะมีข้อความอันใดอันหนึ่งดังนี้

Device <chip name> compiled without errors

Device <chip name> contains errors--see ERROR SUMMARY

ส่วนถัดมาจากบรรทัดข้างต้นจะประกอบไปด้วยส่วนอื่นๆ อีกหลายส่วนดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- Error Summary Section

เป็นส่วนที่เก็บข้อมูลเกี่ยวกับข้อผิดพลาดของ Project นี้ ถ้า Project มีความผิดพลาดเกิดขึ้น

- Resource Usage Section

เป็นส่วนที่แสดงรายละเอียดเกี่ยวกับการใช้ ทรัพยากรของ EPLD ที่ถูก Project นำมาใช้

- Input, Output & Buried Logic Sections

เป็นส่วนที่แสดงรายละเอียดต่างๆ ของ input, output และ buried logic

- Macrocell Interconnections Section

เป็นส่วนที่แสดงการเชื่อมต่อกันของ macrocell ทุกตัว ใน EPLD

- Equations Section

เป็นส่วนที่แสดงการลดรูปของสมการ ที่ใช้ใน logic ทุกตัวในอุปกรณ์ โดยที่ logic จะถูกแสดงในรูปของ ชื่อ pin และ ชื่อ node ภายใน

5.3 Report file ของ Project count9

count9 นี้คือวงจรนับ 9 ที่สร้างขึ้นจาก AHDL Text Editor โดยใช้หลักการของ state machine ในการออกแบบ ซึ่ง Report file ที่ได้หลังจากการ compile มีดังนี้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
Project Information                    C:\MAX2WORK\MEDI\COUNT9.RPT

MAX+plus II Compiler Report File
Version 1.14  10/17/91

***** Project compiled without errors

Untitled

** DEVICE SUMMARY **

Chip/                  Input  Output  Bidir
POF     Device  Pins    Pins    Pins    MCs  EXPs    % Utilized

COUNT9  EPM5016  3       4       0      4     0          25 %

Project Information                    C:\MAX2WORK\MEDI\COUNT9.RPT

** STATE MACHINE ASSIGNMENTS **

count: MACHINE
OF BITS (out3,out2,out1,out0 )
WITH STATES (
        s0 = B"0000",
        s1 = B"0001",
        s2 = B"0010",
        s3 = B"0011",
        s4 = B"0100",
        s5 = B"0101",
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        s6 = B"0110",
        s7 = B"0111",
        s8 = B"1000",
        s9 = B"1001"
);
```

COUNT9 C:\MAX2WORK\MEDI\COUNT9.RPT

***** Logic for device COUNT9 compiled without errors

Device: EPM5016

Turbo: ON

Security: OFF

```
                    EPM5016
                    - - - - -
          GND -|1       20|- clock
          GND -|2       19|- inc
     RESERVED -|3       18|- q0
     RESERVED -|4       17|- q1
          VCC -|5       16|- VCC
          GND -|6       15|- GND
     RESERVED -|7       14|- q2
     RESERVED -|8       13|- q3
          GND -|9       12|- reset
          GND -|10      11|- GND
                    - - - - -
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COUNT9 C:\MAX2WORK\MEDI\COUNT9.RPT

** RESOURCE USAGE **

Logic Array Block	Macrocells	I/O Pins	Expanders
A: MC1 - MC16	4/16(25%)	4/ 8(50%)	0/32(0%)

Total dedicated input pins used:	3/ 8 (37%)
Total I/O pins used:	4/ 8 (50%)
Total macrocells used:	4/ 16 (25%)
Total expanders used:	0/ 32 (0%)
Total input pins required:	3
Total output pins required:	4
Total bidirectional pins required:	0
Total macrocells required:	4
Total expanders in database:	0
Synthesized macrocells:	0/ 16 (0%)

COUNT9 C:\MAX2WORK\MEDI\COUNT9.RPT

** INPUTS **

				Expanders		Fan-In		
Pin	MCell	LAB	Primitive	Total	Shared	INP	FBK	Name
20	-	-	INPUT	0	0	0	0	clock
19	-	-	INPUT	0	0	0	0	inc
12	-	-	INPUT	0	0	0	0	reset

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

COUNT9 C:\MAX2WORK\MEDI\COUNT9.RPT

** OUTPUTS **

Pin	MCell	LAB	Primitive	Expanders Total	Expanders Shared	Fan-In INP	Fan-In FBK	Name
18	15	A	DFF	0	0	3	0	q0
17	13	A	DFF	0	0	3	2	q1
14	11	A	DFF	0	0	3	3	q2
13	9	A	DFF	0	0	3	3	q3

COUNT9 C:\MAX2WORK\MEDI\COUNT9.RPT

** MACROCELL INTERCONNECTIONS **

```
          M M M M
          C C C C
          1 1 1 9
          5 3 1 |
MCELL     | | | | | Name
MC15 -> @ @ @ @ | q0
MC13 -> - @ @ @ | q1
MC11 -> - - @ @ | q2
MC9  -> - @ @ @ | q3

Pin
20   -> @ @ @ @ | clock
19   -> @ @ @ @ | inc
12   -> @ @ @ @ | reset
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
** EQUATIONS **

clock    : INPUT;
inc      : INPUT;
reset    : INPUT;

q3       =  out3;
 out3    = DFF(  _EQ001 $  GND,  _EQ002, !reset,  VCC );
  _EQ001 =  out0 &  out1 &  out2 & !out3
         # !out0 & !out1 & !out2 &  out3;
  _EQ002 =  clock &  inc;

q2       =  out2;
 out2    = DFF(  _EQ003 $  out2,  _EQ004, !reset,  VCC );
  _EQ003 =  out0 &  out1 & !out3;
  _EQ004 =  clock &  inc;

q1       =  out1;
 out1    = DFF(  _EQ005 $  out1,  _EQ006, !reset,  VCC );
  _EQ005 =  out0 & !out3;
  _EQ006 =  clock &  inc;

q0       =  out0;
 out0    = DFF(  VCC $  out0,  _EQ007, !reset,  VCC );
  _EQ007 =  clock &  inc;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 6

ตัวอย่างวงจร

6.1 วงจร Bus Register

```
SUBDESIGN bus_reg
(
     reg_clk,oe      :    INPUT;
     io_bus[7..0]    :    BIDIR;
)
VARIABLE
     t[7..0]    :    TRI;
     d[7..0]    :    DFF;
BEGIN
%  load 8 bit register from the "io_bus[]" bus %
          d[].clk  =  reg_clk;
          d[].d    =  io_bus[];

%  connect control signals and bus signals to tri-state %
          t[].in    =    d[].q;
          t[].oe    =    oe;
               io_bus[]  =    t[].out;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.2 วงจร Counter

```
SUBDESIGN  counter
(
     clock,ld,en,clr,d[3..0]        :    INPUT;
     q[3..0]                        :    OUTPUT;
)
VARIABLE
     count[3..0]          :    DFF;
BEGIN
  count[].clk  = clock;----% connect clock to
                           CLK port of register group %
  count[].clrn = !clr;-----% connect active-high CLR to
                           active-high CLRN port.NOT(!)
                           is an inverter %
  IF  ld  THEN-------------% if statement prioritizes
     count[].d  =  d[];          counter operation%
  ELSEIF en
     count[].d  =  count[].q + 1;
  ELSE
     count[].d  =  count[].q;
  END IF;
  q[]    =    count[].q;     --------% connect Q port to
                                      output signals%
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.3 วงจร Decoder

```
SUBDESIGN  7SEG
(
     i[3..0]              :     INPUT;
     a,b,c,d,e,f,g,       :     OUTPUT;
)

BEGIN

DEFAULTS  -------------------% Defaults  Statement%
     a = VCC;
     b = VCC;
     c = VCC;
     d = VCC;
     e = VCC;
     f = VCC;
     g = VCC;
END  DEFAULTS;

TABLE     ---------------% Truth Table Statement %

     i[ ]=>a,b,c,d,e,f,g;
     0 =>1,1,1,1,1,1,0;
     1 =>0,1,1,0,0,0,0;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        2 =>1,1,0,1,1,0,1;
        3 =>1,1,1,1,0,0,1;
        4 =>0,1,1,0,0,1,1;
        5 =>1,0,1,1,0,1,1;
        6 =>1,0,1,1,1,1,1;
        7 =>1,1,1,0,0,0,0;
        8 =>1,1,1,1,1,1,1;
        9 =>1,1,1,1,0,1,1;

    END TABLE;

    END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

6.4 วงจร Multiplexer

```
% Quad 4-to-1 Multiplexer using a 74153 TTl MacroFunction %

    FUNCTION  74153 (a,b,1gn,1c[3..0],2gn,2c[3..0])
    SUBDESIGN   mux
    (
        s[1..0],d[15..0]     ;     INPUT;
        y[3..0]              ;     OUTPUT;
    )
    VARIABLE
        mux[1..0]    :74153    ----- % Declares two buried
                                      74153 MacroFunction %
    BEGIN
        mux1.(b,a)    =    s[];
        mux0.(b,a)    =    s[];
       (mux1.2c[],mux1.1c[],mux0.2c[],mux0.1c[])   =   d[];
        y[]     =   (mux1.(2y,1y),mux0.(2y,1y));
    END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 7

วงจร Histogram Equalization

Histogram Equalization เป็นขบวนการหนึ่งซึ่งใช้หลักการทางคณิตศาสตร์ ในการปรับปรุงคุณภาพของภาพที่เก็บไว้ในเครื่องคอมพิวเตอร์ ในการใช้ Histogram Equalization เพื่อปรับปรุงคุณภาพของภาพนั้นจะต้องมีการคำนวนซึ่งใช้เวลานาน ดังนั้นจึงได้เกิดแนวความคิดในการสร้างวงจร Hardware ขึ้นเพื่อทำการคำนวนแทนคำนวนโดยใช้ software เพื่อความรวดเร็วในการทำงาน

ในโครงงานนี้จะเป็นการออกแบบวงจรรวมเพื่อทำการคำนวน Histogram Equalization โดยในการออกแบบนั้นออกแบบโดยใช้ Hardware description Language ของบริษัท Atera คือ Altera Hardware Description Language (AHDL)

7.1 การปรับปรุงคุณภาพของภาพ (Image Enchancement)

หลักการของการปรับปรุงคุณภาพของภาพ คือ ขบวนการที่ใช้กับภาพ image ที่กำหนด เพื่อให้เกิดผลลัพธ์เหมาะสมมากกว่าภาพต้นแบบ ในการนำไปใช้ในงานเฉพาะอย่าง สาเหตุที่ต้องบอกว่านำไปใช้งานเฉพาะอย่างนั้น เนื่องจากขบวนการในการปรับปรุงคุณภาพของภาพนั้นมีหลายวิธี และการนำวิธีการอย่างใดอย่างหนึ่งไปใช้นั้น ขึ้นอยู่กับลักษณะของงานนั้น ๆ เป็นอย่างมาก ตัวอย่างเช่น วิธีการในการปรับปรุงภาพ X-ray นั้นไม่สามารถนำมาใช้ได้กับการปรับปรุงภาพที่ส่งมาจากดาวอังคารได้

เราสามารถแบ่งวิธีการปรับปรุงคุณภาพของภาพ ได้ออกเป็น 2 พวกใหญ่ ๆ คือ

- *Spatial-domain Methods*
- *Frequency-domain Methods*

Spatial-domain Methods

Spatial domain หมายถึงการรวมกันของจุดซึ่งประกอบเป็นภาพ และ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

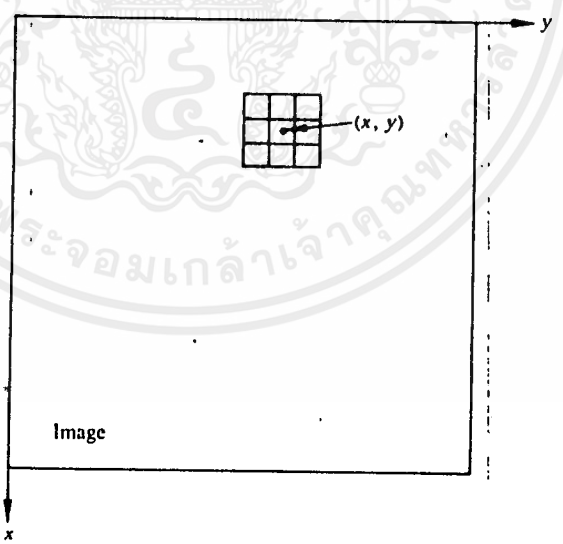
Spatial-domain Methods คือขบวนการที่ปฏิบัติโดยตรงกับจุด สำหรับ function ที่ใน spatial domain สามารถแสดงได้คือ

$$g(x,y) = T[f(x,y)] \qquad \text{-------(1)}$$

โดยที่ f(x,y) เป็นภาพต้นแบบ, g(x,y) เป็นภาพผลลัพธ์จากขบวนการ ส่วน T เป็น operator ที่กระทำกับ f ซึ่ง T จะถูกกำหนดโดยค่าของจุดรอบ ๆ (x,y)

หลักการที่จะกำหนดจุดรอบ ๆ (x,y) นั้นเราใช้ภาพขนาดย่อยที่เป็นสี่เหลี่ยม และมี (x,y) เป็นจุดศูนย์กลาง ดังรูปที่ ๗.1 ซึ่งมีภาพขนาด 3x3 มี (x,y) เป็นจุดศูนย์กลางของภาพ ซึ่งเราเรียกภาพขนาด 3x3 นี้ว่า *Mask* (หรือบางทีก็เรียก *window, templates* หรือ *filter*)

เราจะเลื่อนจุดศูนย์กลางของ mask คือ (x,y) ไปทีละจุด เริ่มจากมุมบนซ้ายสุด และกระทำ operator ที่แต่ละ (x,y) เพื่อให้ได้ค่า g ที่ตำแหน่ง (x,y) นั้น



รูปที่ ๗.1 ภาพจุดรอบข้างของ (x,y) ขนาด 3x3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Frequency-Domain Methods

พื้นฐานของ frequency-domain methods นั้นมาจากทฤษฎี convolution ให้ g(x,y) เป็นภาพที่ได้จากการ convolution ของภาพ f(x,y) กับ operator ที่ค่าไม่ขึ้นกับตำแหน่ง คือ h(x,y) เราจะเขียนสมการได้ดังนี้

$$g(x,y) = h(x,y) * f(x,y) \qquad \text{------(2)}$$

* แทนการ convolution

และจากทฤษฎี convolution เราจะได้สมการต่อไปนี้เป็นจริงด้วย คือ

$$G(u,v) = H(u,v)\, F(u,v) \qquad \text{------(3)}$$

โดยที่ G,H,F เป็น Fourier transforms ของ g,h,f ตามลำดับ และ H(u,v) นั้นบางครั้งเราเรียกว่า *Transfer Function*

ขบวนการปรับปรุงภาพหลายวิธีสามารถแทนได้ด้วย สมการ (3) ในการปรับปรุงภาพนั้น f(x,y) เป็นภาพต้นแบบ ส่วนขั้นตอนในการหาผลลัพธ์ทำได้โดยการหา F(u,v) และเลือก H(u,v) ผลลัพธ์คือ

$$g(x,y) = \mathcal{F}^{-1}[H(u,v)F(u,v)] \qquad \text{------(4)}$$

ซึ่งผลลัพธ์นี้จะแสดงลักษณะบางอย่างของ f(x,y) ออกมา เช่น เราสามารถเน้นขอบของ f(x,y) ได้โดยการเลือก H(u,v) ที่ส่วนประกอบที่มีความถี่สูงของ F(u,v)

7.2 การปรับปรุงภาพโดยใช้ Histogram Modification

Histogram ของระดับสีดำ-ขาว (gray-level) เป็นตัวบอกถึงลักษณะของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ภาพโดยรวม ซึ่งการปรับปรุงภาพโดยใช้ Histogram modification เป็นการปรับปรุงภาพโดยใช้ frequency-domain methods แบบหนึ่ง วิธีการคือทำการเปลี่ยนแปลงค่า histogram ของภาพเพื่อให้ได้ผลลัพธ์ตามต้องการ

หลักการพื้นฐาน

กำหนดให้ r เป็นระดับสีดำ-ขาว ซึ่งค่าของ r อยู่ระหว่าง 0 ถึง 1 ค่า 0 แทนระดับสีดำ และค่าหนึ่งแทนระดับสีขาว

$$0 <= r <= 1 \qquad \text{------(5)}$$

สำหรับค่า r ซึ่งมีค่าระหว่าง 0 ถึง 1 เราจะหาค่า s โดยใช้สมการ

$$s = T(r) \qquad \text{------(6)}$$

ซึ่งให้ค่า s สำหรับทุก ๆ ค่าของ r และเราตั้งสมมุติฐานว่าสมการ (6) นั้นเป็นไปตามเงื่อนไขดังนี้

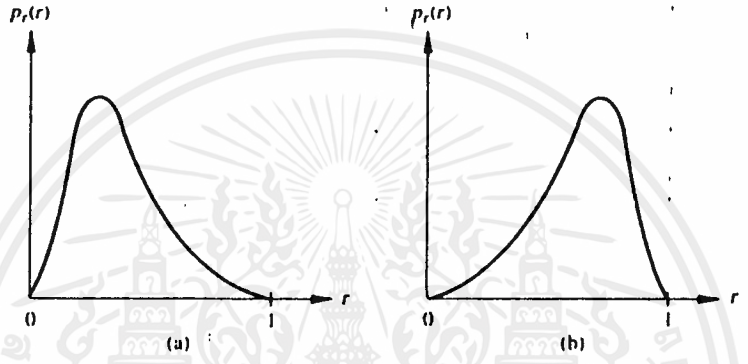
ก) T(r) เป็น single-values และ monotonically increasing ในช่วง 0 <= r <= 1 อาจกล่าวง่าย ๆ ได้ว่า เมื่อให้ค่า r ค่าหนึ่ง T(r) จะต้องให้ผลลัพธ์ค่าเดียว และเมื่อค่าของ r เพิ่มขึ้น ค่าของ T(r) ก็จะต้องเพิ่มขึ้นตามไปด้วย

ข) 0 <= T(r) <= 1 สำหรับ 0 <= r <= 1

เงื่อนไข ก) มีเพื่อรักษาระดับสีดำ-ขาว ให้เพิ่มจากดำไปจนถึงขาว ส่วนเงื่อนไข ข) มีเพื่อประกันว่าค่าของภาพที่แปลงออกมาแล้วจะอยู่ในขอบเขตของระดับสี 0 ถึง 1 รูปที่ 7.2 แสดงภาพของ Transformation function

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการระบายคุณลักษณะของภาพโดยใช้ probability density function ดูได้จากรูปที่ 7.3 จากรูปจะสังเกตได้ว่าภาพ a) นั้นค่อนข้างมืด เนื่องจากค่าของกราฟเบนไปทางซ้าย แสดงว่าจุดส่วนมากมีระดับใกล้เคียงค่า 0 ส่วนภาพ b) นั้นจะค่อนข้างสว่าง



รูปที่ 7.3 แสดง Gray-level probability density function a) ภาพค่อนข้างมืด b) ภาพสว่าง

จากทฤษฎีของ probability ถ้าเรารู้ค่าของ $p_r(r)$, T(r) และ T(r) เป็นไปตามเงื่อนไข ก) แล้วเราสามารถพิสูจน์ได้ว่า

$$p_s(s) = [\, p_r(r)\ (dr/ds)\ ]_{r=T^{-1}(s)} \quad --(8)$$

สำหรับวิธี Histogram Equalization นั้น ก็คือหลักการปรับปรุงภาพโดยการควบคุมค่า probability density function ของระดับสีดำ-เทา โดยใช้ transformation function T(r)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Histogram Equalization

พิจารณาสมการ

$$S = T(r) = \int_0^r p_r(w)dw \quad ; 0 <= r <= 1 \text{ --(9)}$$

โดยที่ w เป็น dummy variable ของการ integration ด้านทางขวาสุดของสมการ (9) เรียกว่า cumulative distribution function (CDF) ของ r และเงื่อนไข ก) และ ข) ที่ได้กล่าวไปก็ยังคงใช้กับสมการนี้ กล่าวคือ CDF จะเพิ่มขึ้นอย่าง monotonically จาก 0 ไปยัง 1 ตามค่าของ r

จากสมการ (9) ค่า derivative ของ s เทียบกับ r คือ

$$ds/dr = p_r(r) \text{ .} \qquad \text{----(10)}$$

แทนค่า dr/ds เข้าไปในสมการ (8) ได้

$$p_s(s) = [\, p_r(r)\ 1/p_r(r)\ ]_{r=T^{-1}(s)}$$

$$= [1]_{r=T^{-1}(s)}$$

$$= 1 \quad , 0 <= s <= 1 \qquad \text{----(11)}$$

เราก็ได้ว่า $p_s(s)$ จะมีค่าเป็น uniform density ในช่วงของค่าตัวแปร s ที่กำหนดให้

เราอาจกล่าวได้ว่าการใช้ transformation function ที่เป็น cumulative distribution จะให้ค่าผลลัพธ์ s ที่มีค่า probability density ($p_s(s)$) เป็น uniform density ซึ่งในการปรับปรุงภาพนั้น จะหมายถึงการเพิ่มช่วงของเฉดของจุดซึ่งจะได้แสดงผลที่เห็นต่อไป และสิ่งนี้จะมีผลต่อสิ่งที่ปรากฏในภาพนั้นมาก

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่าง ตัวอย่างนี้แสดงให้เห็นถึงนำเอาสมการที่ (8) และ (9) มาใช้ สมมุติว่าระดับความดำ-ขาว r มีค่า probability density function แสดงในรูปที่ .4 (a) ซึ่งจะเขียนเป็นสมการได้ดังนี้

$$p(r) = \begin{cases} -2r+2 & , \ 0 <= r <= 1 \\ 0 & , \ elsewhere \end{cases}$$

แทนค่าของสมการในสมการ (9) ซึ่งจะให้ค่า transformation function

$$s = T(r) = \int_0^r (-2w+2)dw$$

$$= -r^2+2r$$

แม้ว่าเราจะใช้เพียง T(r) สำหรับ histogram equalization แต่เพื่อต้องการแสดงว่าค่าของ density $p_s(s)$ เป็นค่า uniform จึงจำเป็นต้องหาค่า $T^{-1}(s)$ ซึ่งในทางปฏิบัติแล้วขั้นตอนนี้ไม่จำเป็นเพราะสมการ (11) นั้นไม่ขึ้นกับค่าของ inverse transformation

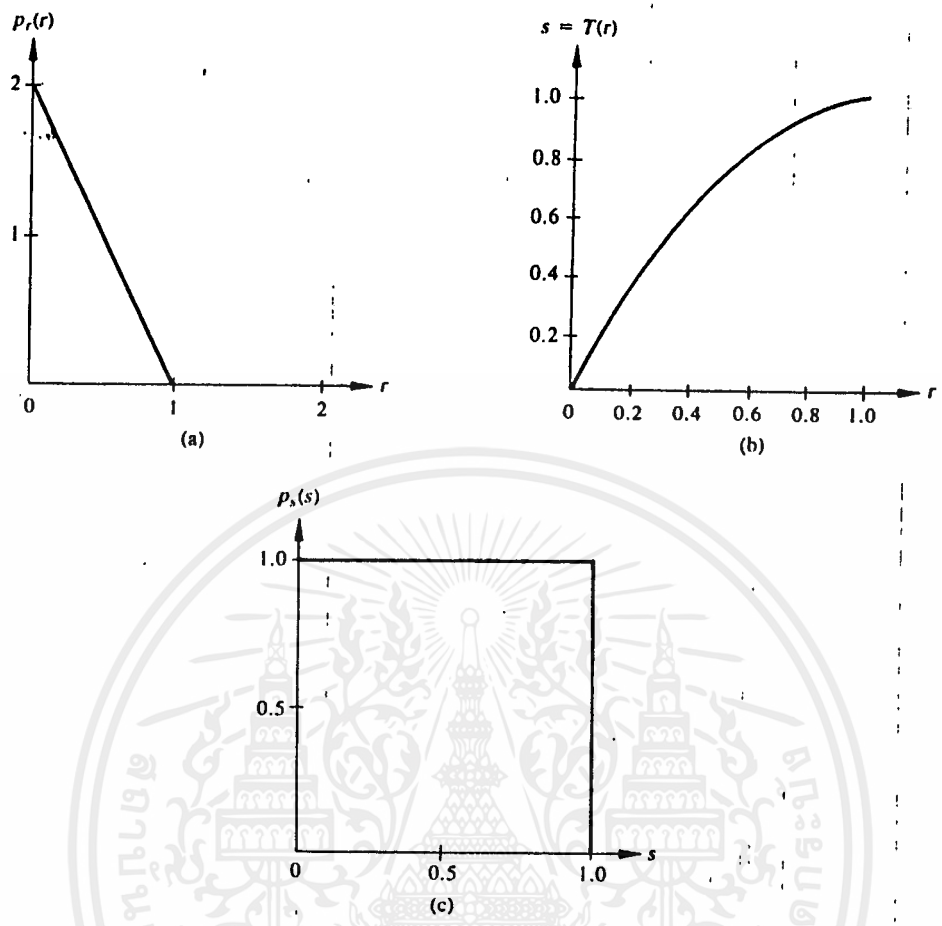
หาค่าของ r ในรูปของ s จะได้

$$r = T^{-1}(s) = 1 - \sqrt{1-s}$$

เนื่องจากค่า r อยู่ระหว่าง [0,1] ดังนั้นค่าที่ใช้ได้ คือ

$$r = T^{-1}(s) = 1 - \sqrt{1-s}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.4 แสดง unform-density method รูป a) เป็น probability density function เริ่มแรก b) Transformation function c) uniform density ที่ได้

Probability density function ของ s หาได้โดยแทนค่าข้างต้นลงในสมการ (8)

$$p_S(s) = [\ p_r(r)\ dr/ds\ ]_{r=T^{-1}(s)}$$

$$= [\ (-2r+2)\ dr/ds\ ]_{r=1-\sqrt{1-s}}$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$= [\ 2\pi 1\text{-}s\ d(1\text{-}\pi 1\text{-}s)/ds\ ]$$

$$= 1\ ,\quad 0 <= s <= 1$$

ซึ่งเป็นค่า uniform density ในช่วงที่ต้องการ และ Tranformation function T(r) แสดงในรูป 7.4(b) และ $P_s(s)$ แสดงในรูป 7.4(c)

เพื่อให้สามารถใช้ประโยชน์ได้ใน digital image processing หลักการข้างต้นนี้ต้องดัดแปลงให้สามารถใช้กับรูปแบบ discrete ได้ สำหรับระดับดำ-ขาวที่เป็นค่า discrete เราจะได้ค่า probability คือ

$$p_r(r_k) = n_k/n \quad ,\ 0 <= r <= 1 \qquad \text{-------(12)}$$
$$k = 0,1,\ldots,L\text{-}1$$

โดย L เป็นจำนวนของระดับ, $p_r(r_k)$ เป็นค่า probability ของระดับดำ-ขาวที่ kth, n_k เป็นจำนวนครั้งที่ระดับนี้ปรากฏอยู่ในภาพ, และ n จำนวนของจุดทั้งหมดในภาพ การ plot ค่าของ $p_r(r_k)$ เทียบกับ r_k เราเรียกว่า *histogram* และวิธีการที่เราใช้เพื่อให้ได้ uniform histogram เราเรียกว่า *histogram equalization* หรือ *histogram linearization*

สมการ (9) เราเขียนให้อยู่ในรูปของ discrete ได้เป็น

$$s_k = T(r_k) = \sum_{j=0}^{k} n_j/n$$
$$\text{-------(13)}$$
$$= \sum_{j=0}^{k} p_r(r_j) \quad ,\ 0 <= r_k <= 1$$
$$k = 0,1,\ldots,L\text{-}1$$

ค่า inverse transformation คือ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

$$r_k = T^{-1}(s_k) \qquad , 0 <= s_k <= 1$$

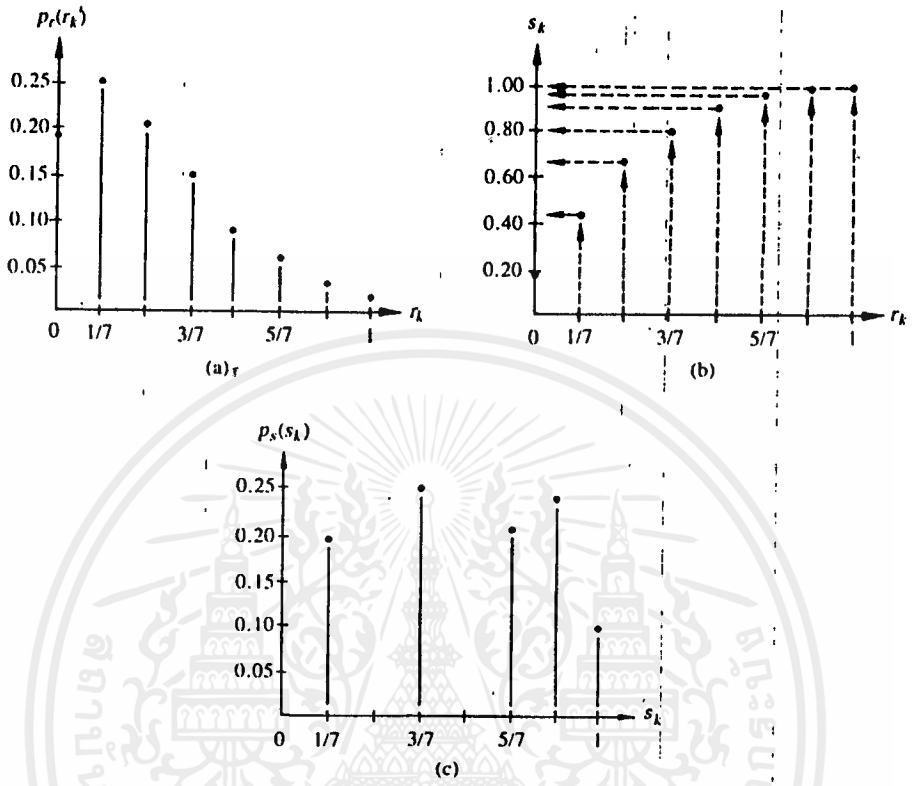
โดยที่ $T(r_k)$ และ $T^{-1}(s_k)$ เป็นไปตามเงื่อนไข ก) และ ข) ที่ได้กล่าวมาแล้ว และสังเกตได้ว่า Transformation function $T(r_k)$ สามารถคำนวนได้โดยตรงจากภาพต้นแบบโดยใช้สมการ (13)

ตัวอย่าง สมมุติว่ามีภาพขนาด 64x64, 8 ระดับ ที่มี distribution แสดงในตารางที่ 1 และค่า histogram levels แสดงในรูปที่ 7.5

ตาราง 1

r_k	n_k	$p_r(r_k) = n_k/n$
r_k=0	790	0.19
r_k=1/7	1023	0.25
r_k=2/7	850	0.21
r_k=3/7	656	0.16
r_k=4/7	329	0.08
r_k=5/7	245	0.06
r_k=6/7	122	0.03
r_k=1	81	0.02

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 7.5 แสดงวิธี histogram equalization a) histogram เริ่มต้น b) Transformation function c) Equalized histogram

Transformation function หาได้โดยสมการ (13) เช่น

$$s_0 = T(r_0) = \sum_{j=0}^{0} p_r(r_j)$$

$$= p_r(r_0)$$

$$= 0.19$$

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สำหรับ s_1

$$s_1 = T(r_1) = \sum_{j=0}^{1} p_r(r_j)$$

$$= p_r(r_0)+p_r(r_1)$$

$$= 0.44$$

และ

$s_2 = 0.65$ $s_5 = 0.95$

$s_3 = 0.81$ $s_6 = 0.98$

$s_4 = 0.89$ $s_7 = 1.00$

Transformation function แสดงได้ในรูป 7.5(b)

เนื่องจากมีเพียง 8 ระดับเท่านั้น (คือ 0,1/7,...,1) ดังนั้นค่าที่แปลงออกมาแล้วจึงต้องกำหนดให้ใกล้เคียงกับค่าระดับที่ใกล้เคียงที่สุด ซึ่งเราจะได้

$s_0 \sim 1/7$ $s_4 \sim 6/7$

$s_1 \sim 3/7$ $s_5 \sim 1$

$s_2 \sim 5/7$ $s_6 \sim 1$

$s_3 \sim 6/7$ $s_7 \sim 1$

สังเกตได้ว่ามีเพียง 5 ระดับที่แตกต่างกันเท่านั้นสำหรับ histogram-equalized เราจะกำหนดสัญลักษณ์ใหม่ดังนี้

$s_0 = 1/7$ $s_3 = 6/7$

$s_1 = 3/7$ $s_4 = 1$

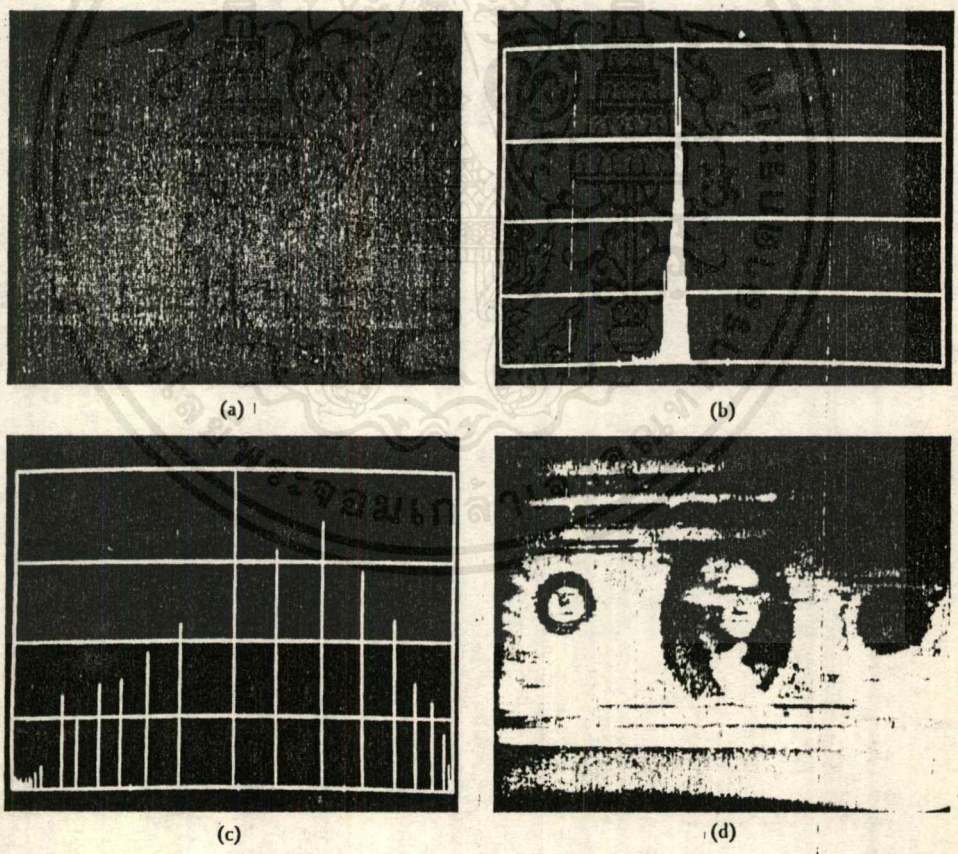
$s_2 = 5/7$

โดยที่ r_3 ถูก mapped เข้าที่ $s_0=1/7$ ซึ่งมี 790 จุดสำหรับค่าใหม่นี้ และ 1023 จุด สำหรับค่า $s_1=3/7$ และ 850 จุดที่ค่า $s_2=5/7$ สำหรับระดับ r_3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และ r_4 นั้นถูก mapped ไปยัง s_3 = 6/7 ซึ่งมีจำนวนจุดสำหรับระดับใหม่นี้ 656 + 329 = 985 จุด และเช่นเดียวกัน 245+122+81 = 448 จุดที่มีค่า s_4 = 1 เมื่อเราหารค่าเหล่านี้ด้วย n=4096 (64x64) ก็จะได้ histogram ดังรูป 7.5(c) Histogram นั้นเป็นค่าประมาณของ probability density function โดยทั่วไปแล้วค่า density ที่เป็นค่าเรียบเท่ากันนั้นเป็นไปได้ยากเมื่อระดับเป็น discrete

ตัวอย่าง เป็นการแสดงการนำเอา histogram equalization ไปใช้งานจริง พิจารณารูปที่ 7.6(a) ซึ่งเป็นภาพธนบัตร dollar ซึ่งภาพเลือนมาก รูปที่ 7.6(b) แสดงค่าของ histogram ของภาพซึ่งมีค่าอยู่ในช่วงแคบ สำหรับ histogram ที่ผ่านขบวนการ histogram equalization แล้วเป็นดังรูป 7.6(c) และภาพที่ได้เป็นดังรูป 6.6(d) เราจะสังเกตุได้ว่ามีการปรับปรุงภาพให้ชัดขึ้นมาก



รูปที่ 7.6 แสดง Histogram equalization a) ภาพต้นแบบ b) histogram ต้นแบบ c) Histogram ที่ถูก equalized แล้ว d) ภาพมีความคมชัดขึ้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 8

การออกแบบวงจร Histogram Equalization โดยใช้ AHDL

โครงงานนี้จะเป็นการนำเอา Altera Hardware Description Language มาใช้เพื่อออกแบบวงจรรวม โดยวงจรรวมที่จะออกแบบนั้นเป็นวงจรที่ใช้ในงาน image processing คือวงจรที่ช่วยในการคำนวน Histogram Equalization

Histogram Equalization นั้นเป็นกระบวนการในการปรับปรุงภาพให้ชัด (Image enhancement) โดยใช้หลักการของการกระจายค่าของระดับสีที่อยู่ติดกันให้กระจายออกไป เป็นผลให้ภาพชัดขึ้น

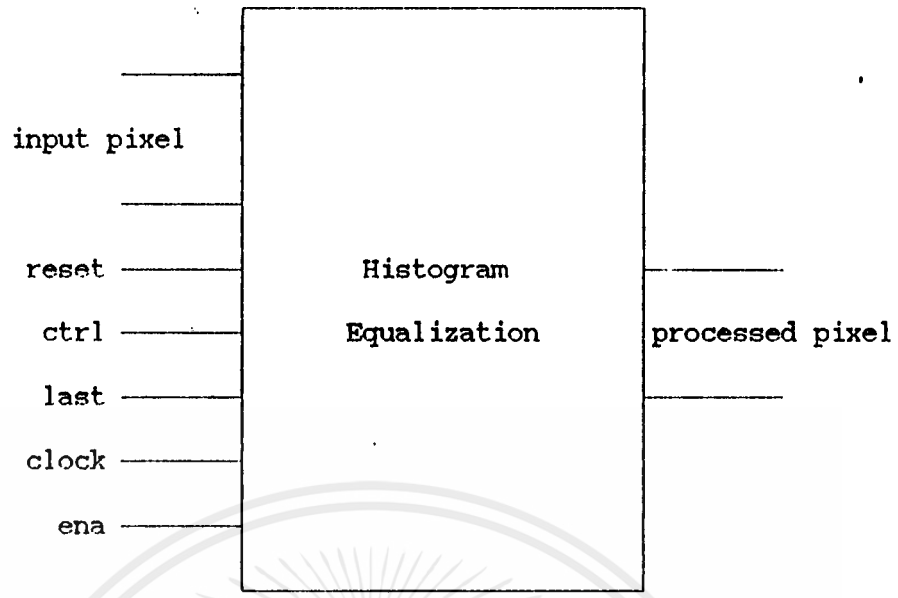
Algorithm โดยคร่าว ๆ ของ histogram equalization นั้นจะต้องหา histogram ของภาพทั้งหมดก่อน เมื่อหาได้แล้วก็ทำการคำนวนหา transfer function แล้วเก็บค่าเอาไว้ ขั้นต่อมาก็ทำการ input ค่าจุดของภาพเข้ามาแล้วเทียบดูกับ transfer function ดูว่าได้ค่าเท่าไร ซึ่งค่านี้ก็คือจุดที่ทำการ Equalized แล้วนั้นเอง

8.1 การทำงานของวงจร

วงจรรวมที่ออกแบบจะออกแบบเพื่อหา histogram equalization ของภาพ grey-scale 16 ระดับ และขนาดภาพ 64x64 จุด

รูปที่ 8.1 แสดงขา input และ output ต่าง ๆ ของวงจรรวม

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.1 แสดงขา Input และ Output ของวงจร Histogram Equalization

การทำงานของวงจรจะแบ่งเป็น 2 mode คือ

- mode 0 เป็นการบอกวงจรว่าจะทำการใส่ค่า gray-level ของภาพเพื่อให้วงจรทำการหาค่า histogram
- mode 1 เป็นการบอกวงจรว่าจะใส่ค่า gray-level ของภาพ เพื่อให้วงจรทำการ map ค่าของจุดที่ใส่เข้ามา โดยเทียบกับ transfer function ที่ได้คำนวนไว้แล้ว ผลลัพธ์จะเป็นจุดที่ถูก equalized แล้ว

เราสามารถกำหนด mode ได้โดยการกำหนดขา ctrl โดย

- ctrl=0 เป็น mode 0 และ
- ctrl=1 เป็น mode 1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.1 ขั้นตอนในการหา Histogram equalization โดยใช้วงจรรวมที่ออกแบบ

เราสามารถแบ่งขั้นตอนใหญ่ ๆ ได้เป็น 3 ขั้นตอน คือ

ขั้นที่ 1 เริ่มแรกเราจะต้องใส่ค่าระดับของจุดเข้าไปทั้งหมดก่อน (โดยเลือก mode ให้เป็น mode0) เพื่อที่วงจรจะนำค่าระดับจุดเหล่านี้เข้าไปหา histogram

ขั้นที่ 2 หลังจากใส่ค่าจุดเข้าไปจนหมดแล้ว จึงให้ค่า last = 1 เพื่อบอก วงจรว่าได้ใส่จุดครบแล้ว และให้วงจรทำการหาค่า transfer function ของภาพ และทำการเก็บค่า transfer function ไว้ใน register ภายในวงจร

ขั้นที่ 3 ใส่ค่าจุดเข้าไปเพื่อหาค่า equalization โดยเมื่อเราใส่ค่าจุดเข้าไปแล้ว วงจรจะทำการเทียบค่าระดับของจุด กับค่า transfer function ที่เก็บไว้ แล้วก็จะให้ค่า output ออกมา

เราสามารถสรุปขั้นตอนการหา Histogram equalization โดยใช้วงจรได้เป็นตารางดังนี้

ขาวงจร	ค่าระดับของขาต่าง ๆ		
	ขั้นที่ 1	ขั้นที่ 2	ขั้นที่ 3
ctrl	0	X	1
last	0	1	0
ena	1	1	1

<u>หมายเหตุ</u> X หมายถึงระดับ 0 หรือ 1 ก็ได้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขาต่าง ๆ ของวงจรรวม

Inputpixel : คือขา input ซึ่งมีขนาด 4 บิท (แสดง gray-level 16 ระดับ)

Processed pixels : คือขา output ขนาด 4 บิท

ctrl : คือขาสำหรับเลือก mode
ถ้าเท่ากับ 0 หมายถึง mode การหา histogram
1 หมายถึง mode การหา equalized pixels

last : มีค่าเป็น 1 เมื่อส่งจุดเข้ามาเพื่อหา histogram ครบแล้ว โดยการทำงานของขา last จะไม่ขึ้นกับ mode

ena : เป็น 1 เมื่อต้องการให้วงจรทำงาน

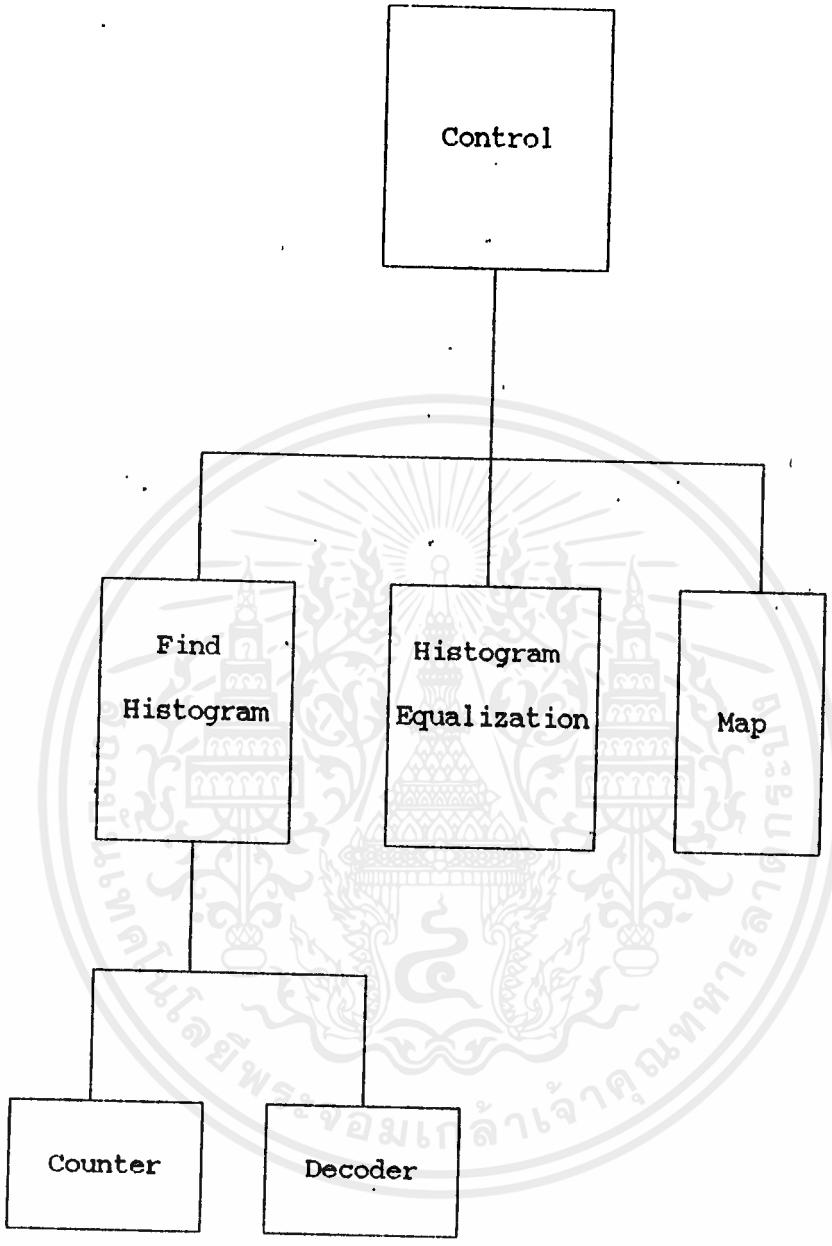
clock : เป็น Clock ของวงจรรวม

reset : ขา reset ของวงจร เพื่อ reset ค่า registers ต่าง ๆ ภายในวงจร

8.3 ขั้นตอนการออกแบบ

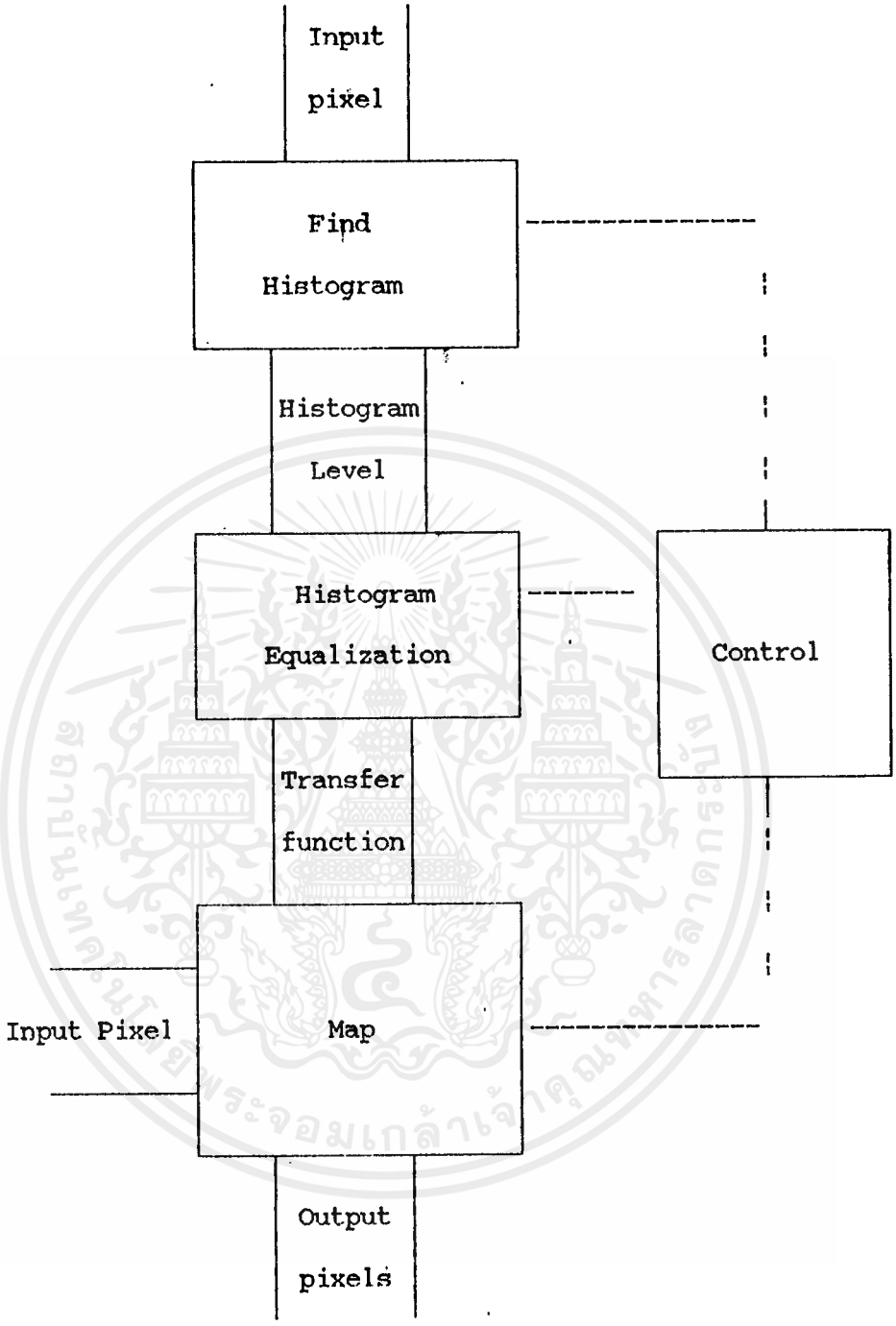
จากข้อกำหนดต่างๆ ของวงจรรวม เราจะทำการออกแบบวงจร รูปที่ 8.2 แสดงโครงสร้างของวงจร และรูปที่ 8.3 แสดงการส่งผ่านข้อมูลภายในวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ ๘.2 โครงสร้างของวงจรหา Histogram Equalization

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 8.3 แสดงการส่งผ่านข้อมูลภายในวงจร

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 8.2 ซึ่งแสดงถึงโครงสร้างภายในของวงจรรวม เราสามารถแบ่ง module ออกเป็น module ย่อย ๆ ได้ 6 module โดยแบ่งเป็น 3 ระดับ

ระดับบนสุดจะเป็น module ที่ใช้ในการควบคุม module นี้ใช้ในการควบคุมการทำงานของ module ระดับล่างอื่น ๆ

ระดับถัดมา ประกอบด้วย module 3 module คือ

- Find histogram ใช้ในการหา histogram สำหรับ module Find Histogram นี้ จะมีการเรียกใช้ module ในระดับล่างลงไปอีก คือ module Counter และ Decoder

- Histogram equalization จะนำค่า histogram ที่ได้จาก module Find histogram มาทำการหา Transfer function

- Map เป็น module ที่รับค่า input เข้ามาแล้วทำการเทียบดูว่าตรงกับค่า transfer function อะไร แล้วจึงให้ผลลัพธ์ออกมา

การทำงานของวงจรอาจแสดงได้โดยรูปที่ 8.3 โดยเริ่มแรกเราจะให้ input pixel เข้ามาในวงจร Find histogram เพื่อหา histogram ของภาพก่อน ซึ่งผลลัพธ์ที่ได้จาก module Find histogram ก็คือค่า histogram ของภาพ ซึ่งจะให้เป็น input ของ module Histogram equalization ซึ่งจะนำไปหา transfer function ค่า transfer function จะถูกส่งไปยัง module Map และ module map จะเก็บค่า transfer function นี้ไว้ใน register ภายใน และเมื่อเราป้อนค่า input pixel เข้าไปอีกที module map จะนำค่าระดับของ pixel เข้าไปเทียบกับ transfer function เพื่อให้ output ออกมา

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ตัวอย่างการหา Histogram equalization โดยใช้ Software

```
#include <stdio.h>
main()
{
     void plotframe(int,int,int *,int *);
     void plothis(int *,float);
     void transfer(long int *,unsigned char *,unsigned int);
     unsigned long int shadelevel[16],i,numpel;
     int  shade,sizex,sizey;
     int scale;
     char name[40],reply;
     unsigned char s[16];
     FILE *img,*img2;

     clrscr();
     printf("Enter file to Histrogram eq (X:XXXXXX.XXX) : ");
     gets(name);
     if((img=fopen(name,"rb"))==NULL)
     {
          fprintf(stderr,"cannot open input file. \n");
          return(1);
     }
     cputs("What size of your image ?\n\r");
     cputs("X: ");scanf("%d",&sizex);
     cputs("Y: ");scanf("%d",&sizey);
     numpel= (long)(sizex) * (long)(sizey);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
for(i=0;i<16;i++)
     shadelevel[i]=0;

/*  Find Histogram */

i=0;
while( i< numpel ) {
     shade=fgetc(img);
     shadelevel[shade]++;
     i++;
}
fclose(img);

/* Equalization */
 transfer(shadelevel,s,numpel);

printf("Do you want to save ?");
reply=getche();
if ( reply!='y' && reply!='Y') exit(1);
if((img=fopen(name,"rb"))==NULL)
{
     fprintf(stderr,"cannot open input file. \n");
     return(1);
}
printf("\nWhat is the name of your eq file ? \n");
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        scanf("%s",name);
        if((img2=fopen(name,"wb"))==NULL)
        {
             fprintf(stderr,"cannot open input file. \n");
             return(1);
        }
        i=0;
        while(i<numpel) {
             scale=fgetc(img);
             fputc(s[scale],img2);
             i++;
        }
        fclose(img);
        fclose(img2);
}

void transfer(long int shadelevel[],unsigned char s[],
              unsigned int numpel)
{
        int i,j;
        float prob[16];
        float t[16],tt[16];
        for (i=0;i<16;i++) {
             t[i]=0;tt[i]=0;
        }
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    for(i=0;i<16;i++) {
        prob[i]=(float)shadelevel[i]/(numpel-1);
    }
    for(i=0;i<16;i++)
        for(j=0;j<=i;j++)
            t[i]+=prob[j];
    for(i=0;i<16;i++)
        tt[i]=t[i]*15;


    for(i=0;i<16;i++) {
        s[i]=(int)tt[i];
        if( ( (int)(tt[i]*1000) % 1000) > 500 ) s[i]++;
        printf(" %d",s[i]);
    }
    printf("\n");
}
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

8.4 สรุป

IC ที่วงจรต้องการใช้คือ EMP5192 จำนวน 3 ตัว

เวลาที่ใช้ น้อยกว่า 1.7 ms

8.5 เปรียบเทียบเวลาที่ใช้โดย Software

จำนวนคำสั่งที่ใช้โดย software 301,405 คำสั่ง

ประมาณว่าแต่ละคำสั่งใช้ 5 clock

ดังนั้น clock ทั้งหมดที่ใช้คือ 5 x 301,405 คำสั่ง

ถ้าเครื่องมีความเร็ว 100 MHz

ใช้เวลาทั้งหมด 15 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SUBDESIGN  14count
(
     inc,clock,reset : INPUT;
     q[13..0]        : OUTPUT;
)
VARIABLE
     count[13..0]: jkff;
BEGIN

     DEFAULTS
          count[].clrn = VCC;
     END DEFAULTS;

     count[].clk = clock & inc;
     count[].clrn = reset;
     q[] = count[].q;
     count0.(j,k) = VCC;
     count1.(j,k) = count0.q;
     count2.(j,k) = count0.q & count1.q;
     count3.(j,k) = count0.q & count1.q & count2.q;
     count4.(j,k) = count0.q & count1.q & count2.q & count3.q;
     count5.(j,k) = count0.q & count1.q & count2.q & count3.q &
                    count4.q;
     count6.(j,k) = count0.q & count1.q & count2.q & count3.q &
                    count4.q & count5.q;
     count7.(j,k) = count0.q & count1.q & count2.q & count3.q &
                    count4.q & count5.q & count6.q;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
count8.(j,k) = count0.q & count1.q & count2.q & count3.q &
               count4.q & count5.q & count6.q & count7.q;
count9.(j,k) = count0.q & count1.q & count2.q & count3.q &
               count4.q & count5.q & count6.q & count7.q &
               count8.q;
count10.(j,k) = count0.q & count1.q & count2.q & count3.q &
               count4.q & count5.q & count6.q & count7.q &
               count8.q & count9.q;
count11.(j,k) = count0.q & count1.q & count2.q & count3.q &
               count4.q & count5.q & count6.q & count7.q &
               count8.q & count9.q & count10.q;
count12.(j,k) = count0.q & count1.q & count2.q & count3.q &
               count4.q & count5.q & count6.q & count7.q &
               count8.q & count9.q & count10.q & count11.q;
count13.(j,k) = count0.q & count1.q & count2.q & count3.q &
               count4.q & count5.q & count6.q & count7.q &
               count8.q & count9.q & count10.q & count11.q
               count12.q;

END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SUBDESIGN  4decoder
(
      in[3..0]   : INPUT;
      out[15..0]: OUTPUT;
)
BEGIN
      IF (in[]==0) THEN
            out0 = VCC;
      ELSIF (in[]==1) THEN
            out1 = VCC;
      ELSIF (in[]==2) THEN
            out2 = VCC;
      ELSIF (in[]==3) THEN
            out3 = VCC;
      ELSIF (in[]==4) THEN
            out4 = VCC;
      ELSIF (in[]==5) THEN
            out5 = VCC;
      ELSIF (in[]==6) THEN
            out6 = VCC;
      ELSIF (in[]==7) THEN
            out7 = VCC;
      ELSIF (in[]==8) THEN
            out8 = VCC;
      ELSIF (in[]==9) THEN
            out9 = VCC;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        ELSIF (in[]==10) THEN
              out10 = VCC;
        ELSIF (in[]==11) THEN
              out11 = VCC;
        ELSIF (in[]==12) THEN
              out12 = VCC;
        ELSIF (in[]==13) THEN
              out13 = VCC;
        ELSIF (in[]==14) THEN
              out14 = VCC;
        ELSE
              out15 = VCC;
        END IF;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FUNCTION heqmap ( level[3..0],trig,reset,r[3..0],strobe )
     RETURNS ( y[3..0] );

SUBDESIGN ctrl1ip
(
     a[3..0]                          : INPUT;
     ctrl,last,clock,ena,reset        : INPUT;
     e[3..0]                          : OUTPUT;
)
VARIABLE
     d[3..0]    : DFF;
     count      : MACHINE WITH
          STATES (s0,s1,s2,s3 );
     htgeq      : heqmap;

BEGIN
     count.clk = clock;
     IF ena == 1  THEN   %  ENA  is used to enable IC %

     %  If set CTRL=0 and LAST=0 %
     %  It will be in FIND HISTOGRAM MODE %

          IF ctrl==0 & last==0 THEN
               htgeq.level[] = a[];
               htgeq.trig    = clock;
          END IF;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
%  If last == 1 ,then  %
%last pixel entered  and save Transfer function %

        CASE (count)  IS
             WHEN  s0 =>
                  IF (last==1 )  THEN
                       count = s1;
                  END IF;
             WHEN  s1 =>
                  count = s2;
             WHEN  s2 =>
                 count = s3;
             WHEN  s3 =>
                  htgeq.strobe = VCC;
                  count = s0;
             END CASE;

%  If  last==0 and ctrl = 1 ,then  %
%  find the transfered pixels correspond to input pixels %

        IF ( last==0 & ctrl==1 ) THEN
             htgeq.r[] = a[];
             d[].clk = !clock;
        END IF;

END IF;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        d[] = htgeq.y[];
        e[] = d[];
        d[].clrn = reset;
        htgeq.reset = reset;
END;
```


เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FUNCTION  14count (inc,clock,reset)
     RETURNS (q[13..0]);
FUNCTION  4decoder (in[3..0])
     RETURNS (out[15..0]);

SUBDESIGN  heqmap
(
     level[3..0],trig,reset                    : INPUT;
     r[3..0],strobe                  : INPUT;
     y[3..0]                                   : OUTPUT;
)

VARIABLE
     numpix[15..0]                             : 14count;
     af[13..0],ae[13..0],ad[13..0],ac[13..0]   : NODE;
     ab[13..0],aa[13..0],a9[13..0],a8[13..0]   : NODE;
     a7[13..0],a6[13..0],a5[13..0],a4[13..0]   : NODE;
     a3[13..0],a2[13..0],a1[13..0],a0[13..0]   : NODE;
     uf[14..0],ue[14..0],ud[14..0],uc[14..0]   : NODE;
     ub[14..0],ua[14..0],u9[14..0],u8[14..0]   : NODE;
     u7[14..0],u6[14..0],u5[14..0],u4[14..0]   : NODE;
     u3[14..0],u2[14..0],u1[14..0],u0[14..0]   : NODE;
     regaf[3..0],regae[3..0],regad[3..0],regac[3..0]: DFF;
     regab[3..0],regaa[3..0],rega9[3..0],rega8[3..0]: DFF;
     rega7[3..0],rega6[3..0],rega5[3..0],rega4[3..0]: DFF;
     rega3[3..0],rega2[3..0],rega1[3..0],rega0[3..0]: DFF;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BEGIN

% Find  Histogram  %

      numpix[].inc = 4decoder(level[]);
      numpix[].clock = !trig;
      numpix[].reset = reset;
      (af[],ae[],ad[],ac[]) = numpix[15..12].q[];
      (ab[],aa[],a9[],a8[]) = numpix[11..8].q[];
      (a7[],a6[],a5[],a4[]) = numpix[7..4].q[];
      (a3[],a2[],a1[],a0[]) = numpix[3..0].q[];

% Find Transfer function  %

      u0[] = (0,a0[]);
      u1[] = u0[] + (0,a1[]);
      u2[] = u1[] + (0,a2[]);
      u3[] = u2[] + (0,a3[]);
      u4[] = u3[] + (0,a4[]);
      u5[] = u4[] + (0,a5[]);
      u6[] = u5[] + (0,a6[]);
      u7[] = u6[] + (0,a7[]);
      u8[] = u7[] + (0,a8[]);
      u9[] = u8[] + (0,a9[]);
      ua[] = u9[] + (0,aa[]);
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ub[] = ua[] + (0,ab[]);
uc[] = ub[] + (0,ac[]);
ud[] = uc[] + (0,ad[]);
ue[] = ud[] + (0,ae[]);
uf[] = ue[] + (0,af[]);

% Save  Histogram Level in Registers %

rega0[]  = u0[13..10] # u0[14];
rega1[]  = u1[13..10] # u1[14];
rega2[]  = u2[13..10] # u2[14];
rega3[]  = u3[13..10] # u3[14];
rega4[]  = u4[13..10] # u4[14];
rega5[]  = u5[13..10] # u5[14];
rega6[]  = u6[13..10] # u6[14];
rega7[]  = u7[13..10] # u7[14];
rega8[]  = u8[13..10] # u8[14];
rega9[]  = u9[13..10] # u9[14];
regaa[]  = ua[13..10] # ua[14];
regab[]  = ub[13..10] # ub[14];
regac[]  = uc[13..10] # uc[14];
regad[]  = ud[13..10] # ud[14];
regae[]  = ue[13..10] # ue[14];
regaf[]  = uf[13..10] # uf[14];
```

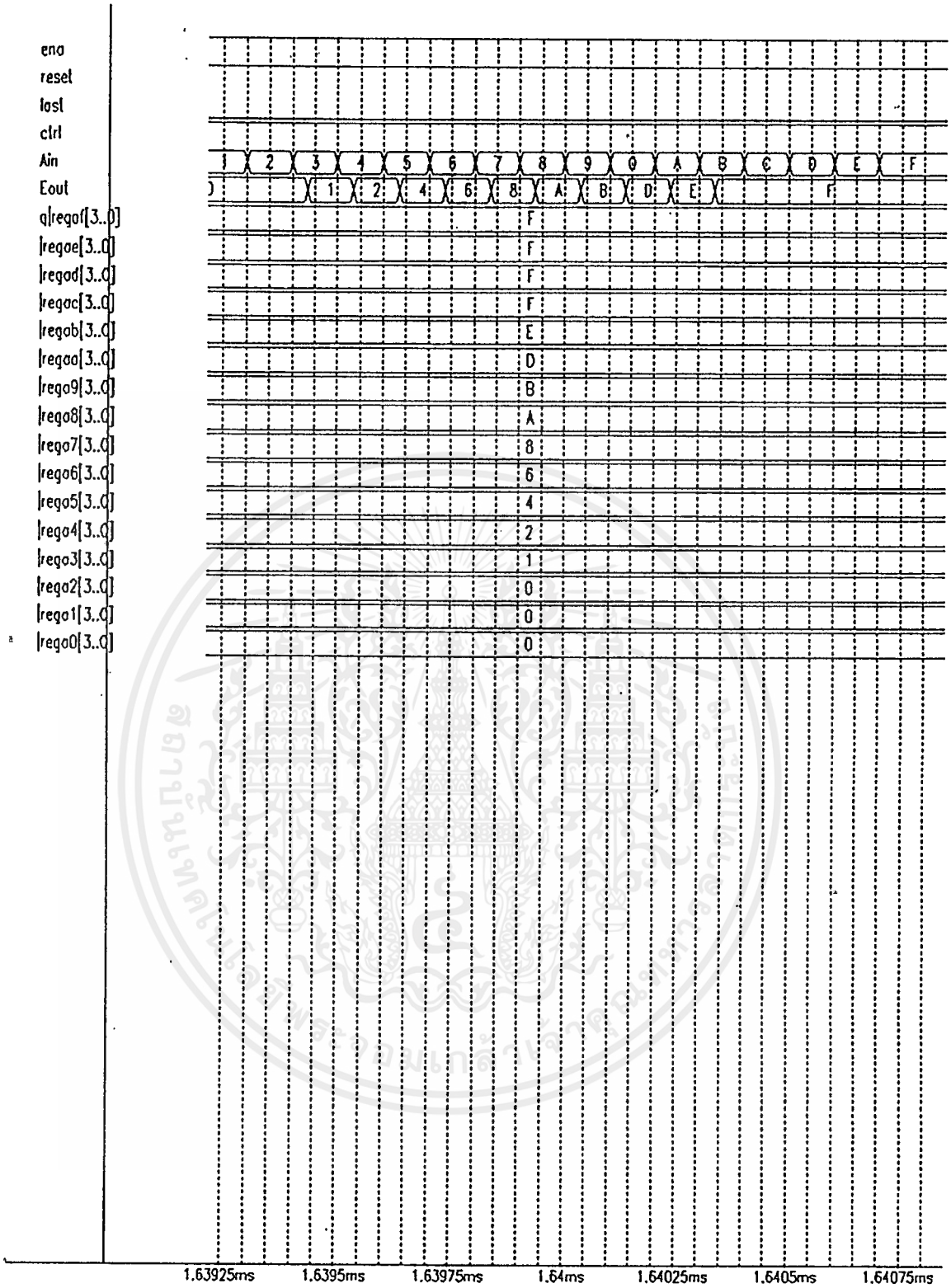
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
IF r[] == 0 THEN
     y[] = rega0[];
ELSIF  r[] == 1 THEN
     y[] = rega1[];
ELSIF  r[] == 2 THEN
     y[] = rega2[];
ELSIF  r[] == 3 THEN
     y[] = rega3[];
ELSIF  r[] == 4 THEN
     y[] = rega4[];
ELSIF  r[] == 5 THEN
     y[] = rega5[];
ELSIF  r[] == 6 THEN
     y[] = rega6[];
ELSIF  r[] == 7 THEN
     y[] = rega7[];
ELSIF  r[] == 8 THEN
     y[] = rega8[];
ELSIF  r[] == 9 THEN
     y[] = rega9[];
ELSIF  r[] == x"a" THEN
     y[] = regaa[];
ELSIF  r[] == x"b" THEN
     y[] = regab[];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ELSIF  r[] == x"c" THEN
         y[] = regac[];
    ELSIF  r[] == x"d" THEN
         y[] = regad[];
    ELSIF  r[] == x"e" THEN
         y[] = regae[];
    ELSE
         y[] = regaf[];
    END IF;

    (rega0[],rega1[],rega2[],rega3[]).clk = strobe;
    (rega4[],rega5[],rega6[],rega7[]).clk = strobe;
    (rega8[],rega9[],regaa[],regab[]).clk = strobe;
    (regac[],regad[],regae[],regaf[]).clk = strobe;
    (rega0[],rega1[],rega2[],rega3[]).clrn = reset;
    (rega4[],rega5[],rega6[],rega7[]).clrn = reset;
    (rega8[],rega9[],regaa[],regab[]).clrn = reset;
    (regac[],regad[],regae[],regaf[]).clrn = reset;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 9

วงจร Edge Detection

9.1 Algorithm

Algorithm ที่ใช้สำหรับการทำให้ edge หรือขอบของภาพให้มีความคมชัดหรือ smooth ขึ้นนั้นจะใช้ algorithm ของ Kasvand นั่นคือ EIKV1 ซึ่ง algorithm นี้มีวิธีรูปแบบดังนี้คือ

$$S_{max} = MAXSUM(L^0, L^1, L^2, L^3)$$

จากสมการข้างบนจะเป็นการนำค่าสูงสุดออกไปเป็นค่าของ output วิธีการทำงานมีดังนี้คือจะมีการรับค่าของ input เข้ามา แล้วนำค่า input ที่รับเข้ามานั้นไปคูณกับค่าคงที่ที่เก็บในรูปของ matrix 5X5 ซึ่งมีทั้งหมด 4 ชุดด้วยกันคือ L^0, L^1, L^2 และ L^3

การรับ input เข้ามาก็จะรับเข้ามาในรูปของ matrix 5x5 เช่นกัน และนำค่าคงที่นั้นไปคูณกับค่าของ input ที่รับเข้ามา วิธีการคูณจะเป็นการคูณแบบนำค่าที่มีตำแหน่งตรงกันมาคูณกัน (ทำทีละชุด) หลังจากนั้นนำค่าผลลัพธ์ที่ได้จากการคูณมาบวกกันแล้วเก็บค่าผลบวกไว้ เมื่อทำครบทั้ง 4 ชุดแล้วจะนำผลบวกที่ได้ของแต่ละชุดซึ่งทั้งหมดมี 4 ชุดนั้นมาทำการเปรียบเทียบกัน ค่าที่มากที่สุดก็คือ output

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ค่าคงที่ที่นำมาใช้ใน algorithm นี้คือ

-1	-1	-1	-1	-1
0	0	0	0	0
2	2	2	2	2
0	0	0	0	0
-1	-1	-1	-1	-1

L^0

-1	-1	2	0	0
-1	-1	0	2	0
-1	0	2	0	-1
0	2	0	-1	-1
2	0	-1	-1	0

L^1

-1	0	2	0	-1
-1	0	2	0	-1
-1	0	2	0	-1
-1	0	2	0	-1
-1	0	2	0	-1

L^2

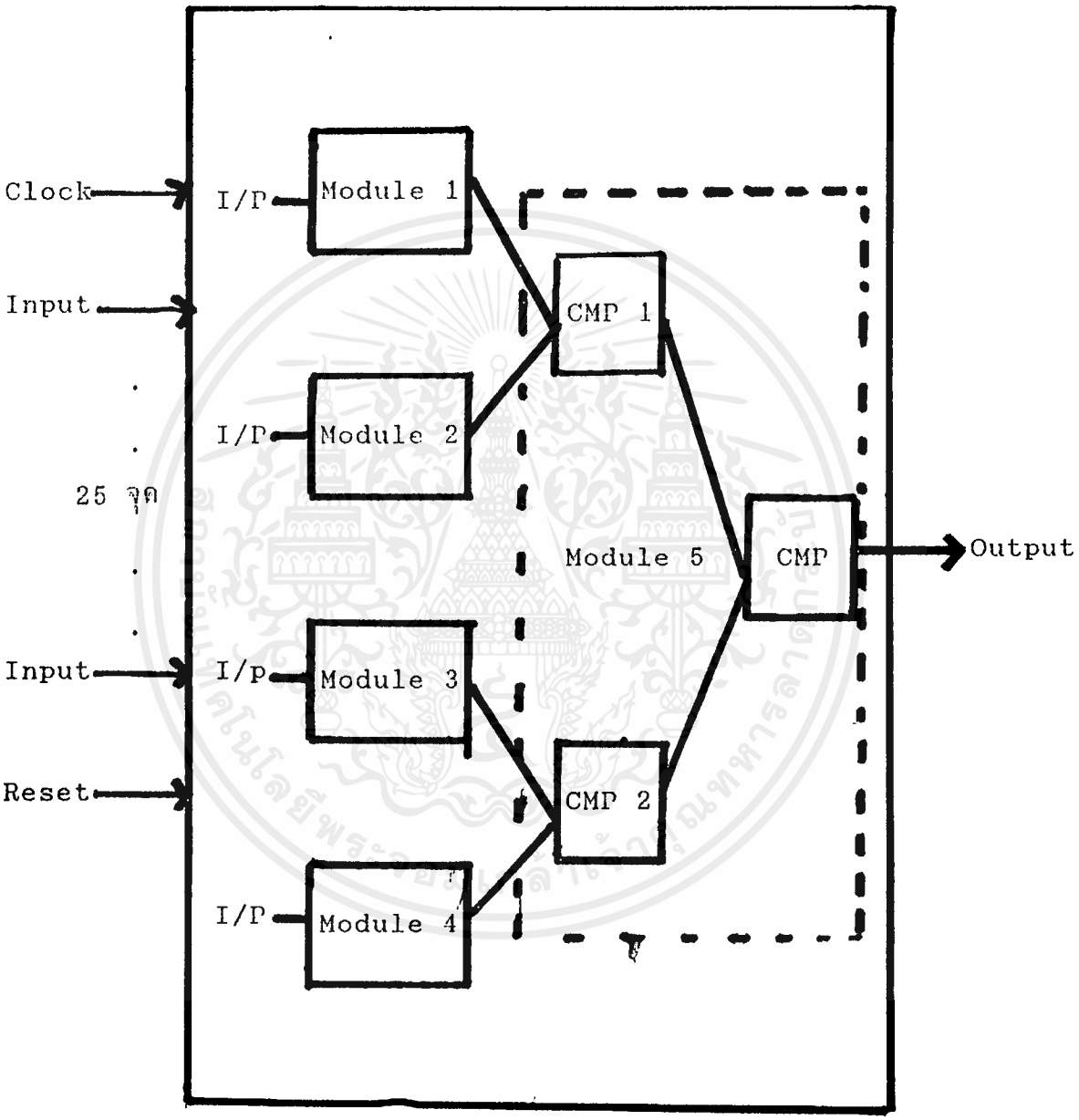
2	-1	0	-1	0
0	2	0	-1	-1
-1	0	2	0	-1
-1	-1	0	2	0
0	-1	-1	0	2

L^3

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

๑.๒ รายละเอียดของวงจรแต่ละ module

ในส่วนของวงจร EDGE DETECTION นั้นจะมีการออกแบบวงจรโดยแบ่งออกเป็น module ย่อย ๆ ดังแสดงในรูป



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะเห็นได้ว่ามีการแบ่งการทำงานของวงจรออกเป็น Module ย่อย ๆ ได้ 5 Module คือ Module ที่ 1 ถึง Module ที่ 5 ซึ่งแต่ละ Module มีการทำงานดังนี้คือ

๑.3 <u>รายละเอียดของ Module 1 ถึง 4 มีดังนี้</u>

1). นำค่า input ทั้ง 25 ค่า ของวงจรต่อเข้ากับขา input ของ D-Flip Flop (DFF) และนำขา clock และ ขา reset ของวงจรต่อเข้าขา clk และขา clrn ของ D-Flip Flop (DFF) ซึ่ง input แต่ละจุดจะมีขนาดเท่ากับ 4 bits ซึ่งจะถูกกำหนดไว้ในส่วนของ SUBDESIGN ส่วนขา input ของ DFF นั้นจะมีขนาดเท่ากับ 8 bits และตัวแปรจะถูกกำหนดในส่วนของ VARIABLE โดยตำแหน่งบิทของ DFF ที่ไม่ถูกนำมาใช้จะถูกกำหนดค่าให้เท่ากับ ground (มีค่า = 0) และในการต่อ input ของวงจรเข้ากับ input ของ DFF นั้นจะต้องใช้จำนวนบิทของข้อมูลที่มีขนาดเท่ากัน ไม่เช่นนั้นจะผิดหลักไวยากรณ์ ตัวอย่างเช่น

```
SUBDESIGN
(
        a0[3..0]      :    INPUT    % Declare circuit input
                                      มีขนาดเท่ากับ 4 bits %
)
VARIABLE
        p0[7..0]      :    DFF      %กำหนดตัวแปร p0[7..0] เป็น
                                      DFF ซึ่งมีขนาดเท่ากับ 8 bits%

BEGIN
        p0[7..1].d    =    GND;
        p0[3..0]      =    a0[];
```

ในส่วน BEGIN เป็นการนำค่า input ของวงจรต่อเข้ากับ input ของ DFF ซึ่งจำนวนบิทของ DFF ที่จะใช้นั้นจะใช้เพียง 4 บิทเท่ากับจำนวนบิทของ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

input ของวงจร ดังนั้นบิทที่เหลือของ DFF ก็จะถูกกำหนดให้มีค่าเท่ากับ ground

2.หลังจากต่อ input ของวงจรให้กับ DFF เรียบร้อยแล้ว ก็จะนำค่า output ของ DFF มาใช้งานในการคำนวณคือนำค่า output ไปคูณกับค่าคงที่ที่ algorithm กำหนดมาให้ซึ่งค่าคงที่นี้กำหนดในรูปของ matric 5x5 แต่กรรมวิธีในการคูณนั้นจะเป็นการคูณโดยการนำค่า output ของ DFF ไปคูณกับค่าที่ตำแหน่งที่ตรงกัน (ไม่ใช่การคูณแบบ matric) ตัวอย่างเช่น นำจุดหรือของภาพมาจำนวนเท่ากับ 25 จุด เพื่อนำมาคูณกับค่าคงที่ที่กำหนดโดย algorithm EIKV2

ค่าที่นำมาจาก ภาพ

a0[]	a1[]	a2[]	a3[]	a4[]
b0[]	b1[]	b2[]	b3[]	b4[]
c0[]	c1[]	c2[]	c3[]	c4[]
d0[]	d1[]	d2[]	d3[]	d4[]
e0[]	e1[]	e2[]	e3[]	e4[]

ค่าคงที่ที่กำหนดโดย algorithm EIKV2

-1	-1	-1	-1	-1
0	0	0	0	0
2	2	2	2	2
0	0	0	0	0
-1	-1	-1	-1	-1

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

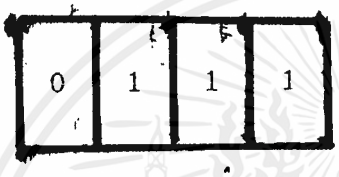
จะได้ผลบวกจากการคูณดังนี้คือ

(2(c0[]+c1[]+c2[]+c3[]+c4[])) ((a0[]+a1[]+a2[]+a3[]+a4[]) + (c0[]+c1[]+c2[]+c3[]+c4[]))

หมายเหตุ

ในกรณีที่นำค่า input ไปคูณกับค่าคงที่ที่มีค่าเท่ากับ 2 นั้นจะทำโดยใช้หลักในการ SHIFT LEFT บิทของ input นั้น และกำหนดให้บิทที่ 0 มีค่าเท่ากับ ground

ตัวอย่างเช่น



x 2

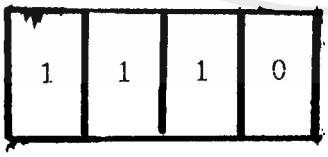
ค่า 0111 ฐานสองแปลงเป็นเลขฐานสิบได้ดังนี้

$$(0x2^3)+(1x2^2)+(1x2^1)+(1x2^0) = 7$$

ดังนั้น

$$7X2 = 14$$

เมื่อทำการ Shift-Left จะได้



ค่า 1110 ฐานสองแปลงเป็นเลขฐานสิบได้ดังนี้

$$(1x2^3)+(1x2^2)+(1x2^1)+(0x2^0) = 14$$

จะเห็นว่าให้ผลลัพธ์เท่ากัน

3). ค่าผลบวกจากการคูณจะนำไปเก็บไว้ใน ตัวแปร NODE โดยจะมีการนำค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

นี้ไปใช้ในการเปรียบเทียบกันทีหลัง (หลังจากได้ผลลัพธ์จากการคูณครบทั้ง 4 ค่าแล้ว)

4).Module 2 ถึง 3 มีหลักการทำงานเหมือน Module 1

๑.๔ <u>รายละเอียดของ Module 5</u>

นำค่า Node ที่เก็บผลบวกจากการคูณมาทำการเปรียบเทียบกัน โดยวิธีการเปรียบเทียบจะทำการเปรียบเทียบทีละ 2 Node แล้วเก็บค่าผลลัพธ์ไว้ในตัวแปร Node อีกค่าหนึ่งซึ่งกำหนดขึ้นมาใหม่ ซึ่งในวงจรนี้กำหนดเป็นตัวแปร m และ n

หลังจากนั้นก็จะนำค่าที่เก็บในตัวแปร Node m และ n มาทำการเปรียบเทียบกันอีกที โดยค่าที่มากที่สุดก็จะเป็นค่า Output ของวงจร

หลังจากได้ค่า Output ของวงจรแล้วเราจะนำค่านั้นไปแทนที่จุดที่อยู่รอบ ๆ จุดกึ่งกลางของภาพ ดังนั้นเราจำเป็นจะต้องหาจุดกึ่งกลางภาพให้ได้ก่อน

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

9.5 สรุป

1). Module ONE.TDF จะใช้เวลาในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 1 จะใช้เวลาเท่ากับ 500 nanosecond (ns) ซึ่งมีค่าเท่ากับ 0.0005 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุด ก็จะใช้เวลารวมทั้งหมดเท่ากับ (64 x 64) x 0.0005 = 2.0480 millisecond (ms) และ IC หรือ CHIP ที่ใช้กับวงจรนี้ก็คือ EPM5192

2). Module TWW.TDF จะใช้เวลาในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 2 จะใช้เวลาเท่ากับ 600 nanosecond (ns) ซึ่งมีค่าเท่ากับ 0.0006 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ (64 x 64) x 0.0006 = 2.4576 millisecond (ms) และ IC หรือ CHIP ที่ใช้กับวงจรนี้ก็คือ EPM5192

3). Module TREE.TDF จะใช้เวลาในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 3 จะใช้เวลาเท่ากับ 400 nanosecond (ns) ซึ่งมีค่าเท่ากับ 0.0004 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ (64 x 64) x 0.0004 = 1.6384 millisecond (ms) และ IC หรือ CHIP ที่ใช้กับวงจรนี้ก็คือ EPM7096

4). Module FOUR.TDF จะใช้เวลาในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 4 จะใช้เวลาเท่ากับ 600 nanosecond (ns) ซึ่งมีค่าเท่ากับ 0.0006 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ (64 x 64) x 0.0006 = 2.4576 millisecond (ms) และ IC หรือ CHIP ที่ใช้กับวงจรนี้ก็คือ EPM5192

5). Module MAX.TDF จะใช้เวลาในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 5 จะใช้เวลาเท่ากับ 80 nanosecond (ns) ซึ่งมีค่า

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เท่ากับ 0.00008 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ

(64 x 64) x 0.00008 = 0.32768 millisecond (ms) และ IC หรือ CHIP
ใช้กับวงจรนี้ก็คือ EPM5064

ดังนั้นเวลารวมทั้งหมดที่ใช้ในการ Simulate จนกระทั่งได้ output ของวงจรคื

(2.0480 + 2.4576 + 1.6384 + 2.4576 + 0.32768) ms = 8.92928 ms

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

List Program ของวงจร Module ONE.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SUBDESIGN  one
(
a0[3..0],a1[3..0]          :     INPUT;
a2[3..0],a3[3..0]          :     INPUT;
a4[3..0]                   :     INPUT;
c0[3..0],c1[3..0]          :     INPUT;
c2[3..0],c3[3..0]          :     INPUT;
c4[3..0]                   :     INPUT;
e0[3..0],e1[3..0]          :     INPUT;
e2[3..0],e3[3..0]          :     INPUT;
e4[3..0]                   :     INPUT;
clock,reset                :     INPUT;
o0[7..0]                   :     OUTPUT;
)
VARIABLE
p0[7..0],p1[7..0]          :     DFF;
p2[7..0],p3[7..0]          :     DFF;
p4[7..0]                   :     DFF;
x0[7..0],x1[7..0]          :     DFF;
x2[7..0],x3[7..0]          :     DFF;
x4[7..0]                   :     DFF;
z0[7..0],z1[7..0]          :     DFF;
z2[7..0],z3[7..0]          :     DFF;
z4[7..0]                   :     DFF;
(aa[7..0],ab[7..0],ac[7..0],ad[7..0],ae[7..0])     :     NODE
(ca[7..0],cb[7..0],cc[7..0],cd[7..0],ce[7..0])     :     NODE
(ea[7..0],eb[7..0],ec[7..0],ed[7..0],ee[7..0])     :     NODE
(u0[7..0],u1[7..0])                                :     NODE
u2[7..0]                                           :     NODE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
BEGIN
(p0[],p1[],p2[],p3[],p4[]).clk      =     clock;
(x0[],x1[],x2[],x3[],x4[]).clk      =     clock;
(z0[],z1[],z2[],z3[],z4[]).clk      =     clock;
(p0[],p1[],p2[],p3[],p4[]).clrn     =     reset;
(x0[],x1[],x2[],x3[],x4[]).clrn     =     reset;
(z0[],z1[],z2[],z3[],z4[]).clrn     =     reset;
p0[7..4].d      =    GND;
p1[7..4].d      =    GND;
p2[7..4].d      =    GND;
p3[7..4].d      =    GND;
p4[7..4].d      =    GND;
p0[3..0].d      =    a0[];
p1[3..0].d      =    a1[];
p2[3..0].d      =    a2[];
p3[3..0].d      =    a3[];
p4[3..0].d      =    a4[];
aa[]            =    p0[].q;
ab[]            =    p1[].q;
ac[]            =    p2[].q;
ad[]            =    p3[].q;
ae[]            =    p4[].q;

      u0[]    =      (aa[]+ab[]+ac[]+ad[]+ae[]);
x0[7..5].d      =    GND;
x0[0].d         =    GND;
x0[4..1].d      =    c0[];
x1[7..5].d      =    GND;
x1[0].d         =    GND;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
x1[4..1].d      =    c1[];
x2[7..5].d      =    GND;
x2[0].d         =    GND;
x2[4..1].d      =    c2[];
x3[7..5].d      =    GND;
x3[0].d         =    GND;
x3[4..1].d      =    c3[];
x4[7..5].d      =    GND;
x4[0].d         =    GND;
x4[4..1].d      =    c4[];
ca[]            =    x0[].q;
cb[]            =    x1[].q;
cc[]            =    x2[].q;
cd[]            =    x3[].q;
ce[]            =    x4[].q;
   u1[]     =   (ca[]+cb[]+cc[]+cd[]+ce[]);
z0[7..4].d      =    GND;
z1[7..4].d      =    GND;
z2[7..4].d      =    GND;
z3[7..4].d      =    GND;
z4[7..4].d      =    GND;
z0[3..0].d      =    e0[];
z1[3..0].d      =    e1[];
z2[3..0].d      =    e2[];
z3[3..0].d      =    e3[];
z4[3..0].d      =    e4[];
ea[]            =    z0[].q;
eb[]            =    z1[].q;
ec[]            =    z2[].q;
```

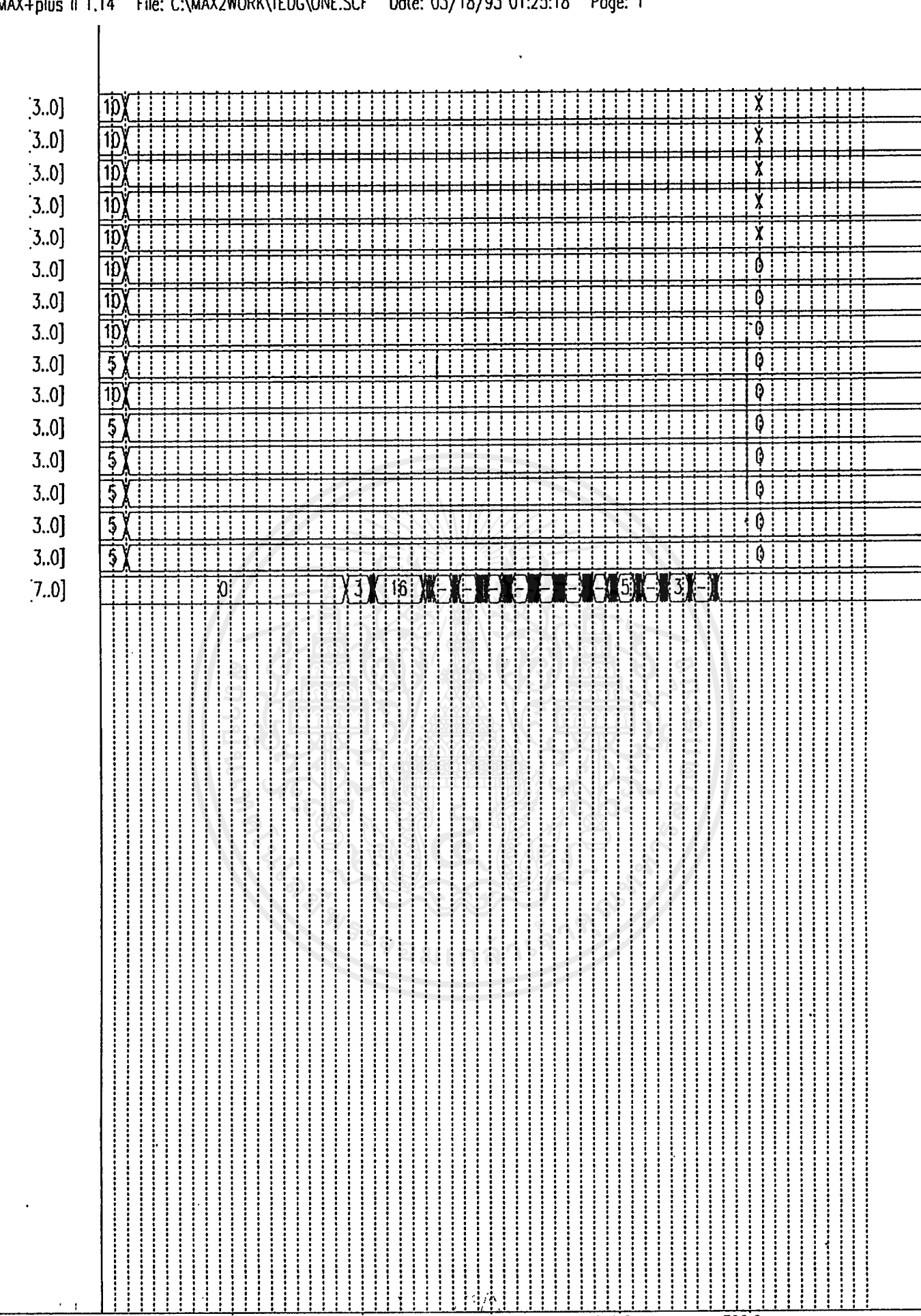
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
ed[]            =    z3[].q;
ee[]            =    z4[].q;
     u2[]       =    (ea[]+eb[]+ec[]+ed[]+ee[]);
IF(u1[]>(u0[]+u2[])) THEN
     o0[]       =    u1[]-(u0[]+u2[]);
ELSE
     o0[]       =    GND;
END IF;
END;
```

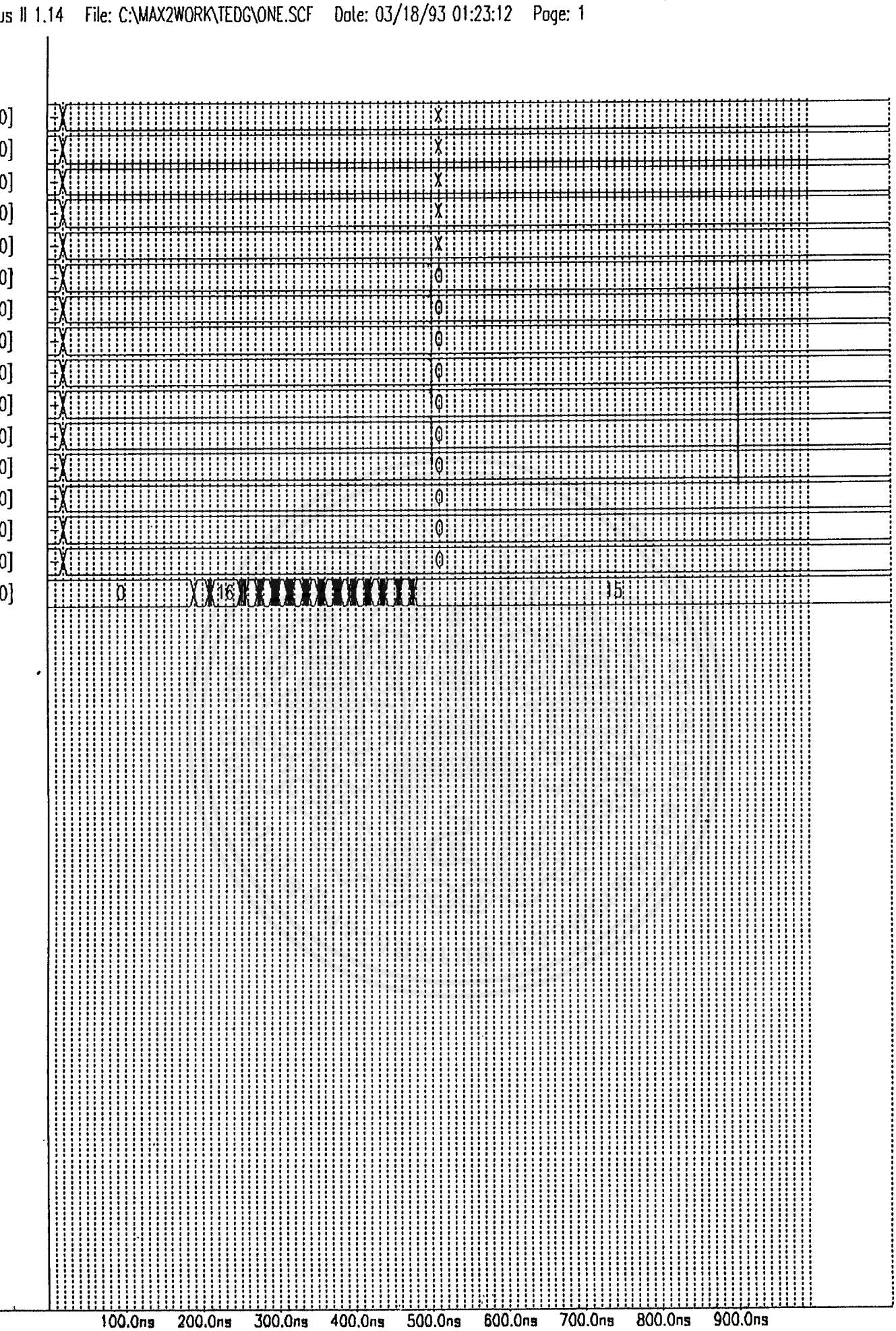
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการ Simulate ของ Module ONE.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการ Simulate ของวงจร Module ที่ 1 คือวงจร ONE.TDF จะเห็นว่าเวลาที่ใช้ในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 1 จะใช้เวลาเท่ากับ 500 ns ซึ่งมีค่าเท่ากับ 0.0005 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ

(64 x 64) x 0.0005 = 2.0480 millisecond (ms)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

List Program ของวงจร Module TWW.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SUBDESIGN tww
(
a0[3..0],a1[3..0]          :INPUT;
a2[3..0],b0[3..0]          :INPUT;
b1[3..0],b3[3..0]          :INPUT;
c0[3..0],c2[3..0]          :INPUT;
c4[3..0],d1[3..0]          :INPUT;
d3[3..0],d4[3..0]          :INPUT;
e0[3..0],e2[3..0]          :INPUT;
e3[3..0],clock,reset       :INPUT;
o1[7..0]                   :OUTPUT;
)
VARIABLE
(p0[7..0],p1[7..0])        :DFF;
(p2[7..0],f1[7..0])        :DFF;
(f3[7..0],f4[7..0])        :DFF;
(t0[7..0],t1[7..0])        :DFF;
(t3[7..0],x0[7..0])        :DFF;
(x2[7..0],x4[7..0])        :DFF;
(z0[7..0],z2[7..0])        :DFF;
z3[7..0]                   :DFF;
(aa[7..0],ab[7..0],ac[7..0])  :NODE;
(db[7..0],dd[7..0],de[7..0])  :NODE;
(ba[7..0],bb[7..0],bd[7..0])  :NODE;
(ca[7..0],cc[7..0],ce[7..0])  :NODE;
(ea[7..0],ec[7..0],ed[7..0])  :NODE;
u0[7..0],u1[7..0]             :NODE;
u2[7..0]                      :NODE;
BEGIN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(p0[],p1[],p2[],f1[],f3[],f4[]).clk     =    clock;
(t0[],t1[],t3[],x0[],x2[],x4[]).clk     =    clock;
(z0[],z2[],z3[]).clk                    =    clock;
(p0[],p1[],p2[],f1[],f3[],f4[]).clrn    =    reset;
(t0[],t1[],t3[],x0[],x2[],x4[]).clrn    =    reset;
(z0[],z2[],z3[]).clrn                   =    reset;
(p1[7..4],p2[7..4])       =    GND;
 p0[7..4].d               =    GND;
 p0[3..0].d               =    a0[];
 p1[7..4].d               =    GND;
 p1[3..0].d               =    a1[];
(t0[7..4],t1[7..4]).d     =    GND;
 t0[3..0].d               =    b0[];
 t1[3..0].d               =    b1[];
 x0[7..4].d               =    GND;
 x0[3..0].d               =    c0[];
 aa[]          =    p0[].q;
 ab[]          =    p1[].q;
 ba[]          =    t0[].q;
 bb[]          =    t1[].q;
 ca[]          =    x0[].q;
    u0[]       =    (aa[]+ab[]+ba[]+bb[]+ca[]);
 x4[7..4].d    =    GND;
 x4[3..0].d    =    c4[];
(f3[7..4],f4[7..4]).d     =    GND;
 f3[3..0].d    =    d3[];
 f4[3..0].d    =    d4[];
(z2[7..4],z3[7..4]).d     =    GND;
 z2[3..0].d    =    e2[];
```

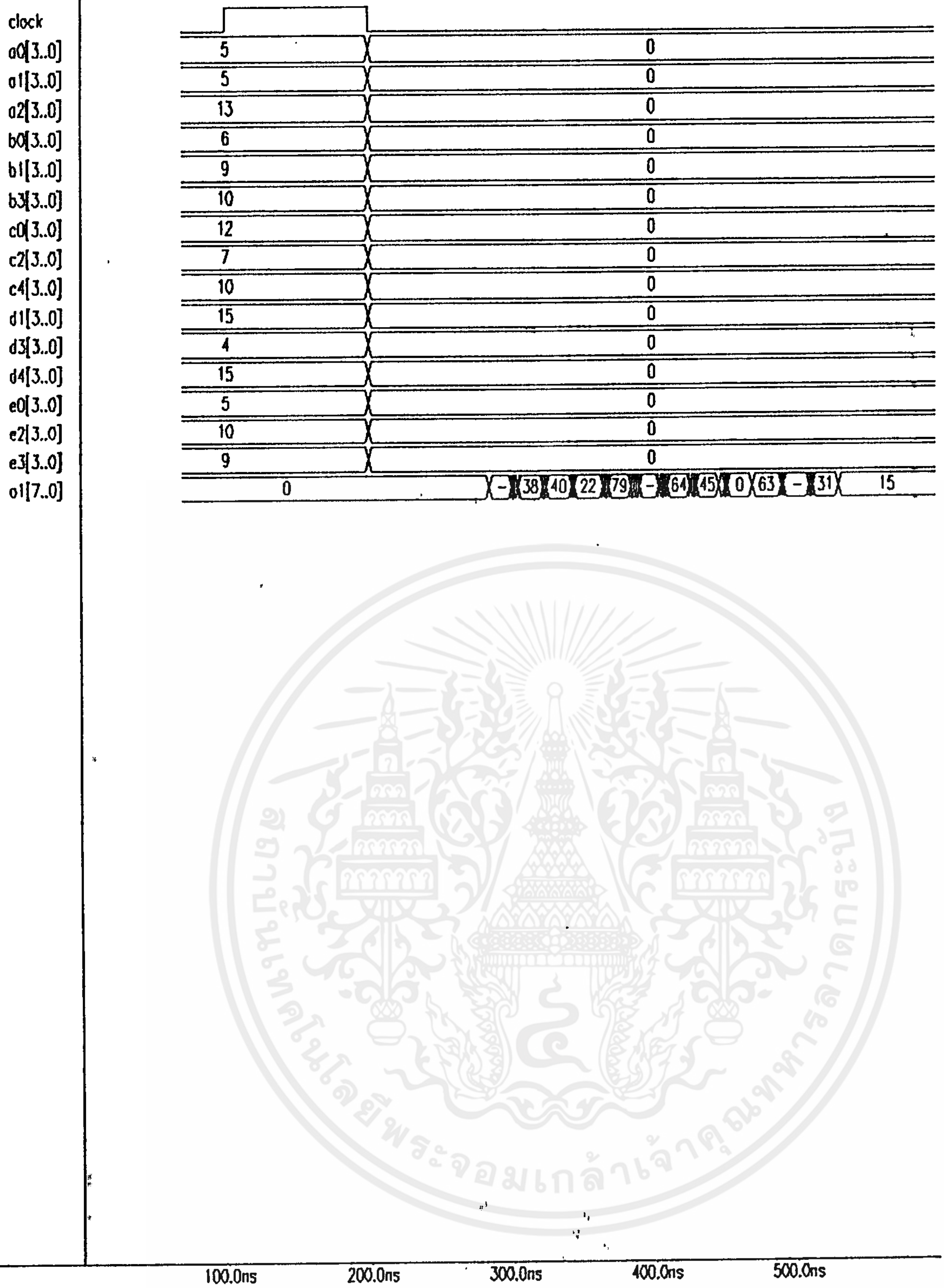
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
z3[3..0].d     =     e3[];
ce[]           =     x4[].q;
dd[]           =     f3[].q;
de[]           =     f4[].q;
ec[]           =     z2[].q;
ed[]           =     z3[].q;
    u1[]       =     (ce[]+dd[]+de[]+ec[]+ed[]);
(z0[7..5],z0[0]).d       =     GND;
z0[4..1].d               =     e0[];
(f1[7..5],f1[0]).d       =     GND;
f1[4..1].d               =     d1[];
(x2[7..5],x2[0]).d       =     GND;
x2[4..1].d               =     c2[];
(t3[7..5],t3[0]).d       =     GND;
t3[4..1].d               =     b3[];
(p2[7..5],p2[0]).d       =     GND;
p2[4..1].d               =     a2[];
ea[]           =     z0[].q;
db[]           =     f1[].q;
cc[]           =     x2[].q;
bd[]           =     t3[].q;
ac[]           =     p2[].q;
    u2[]       =     (ea[]+db[]+cc[]+bd[]+ac[]);
IF (u2[]>(u0[]+u1[]))THEN
    o1[]       =      u2[]-(u0[]+u1[]);
ELSE
    o1[]       =     GND;
END IF;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการ Simulate ของ Module TWW.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการ Simulate ของวงจร Module ที่ 2 คือวงจร TWW.TDF จะเห็นว่าเวลาที่ใช้ในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 2 จะใช้เวลาเท่ากับ 600 ns ซึ่งมีค่าเท่ากับ 0.0006 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ

(64 x 64) x 0.0006 = 2.4576 millisecond (ms)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

List Program ของวงจร Module TREE.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SUBDESIGN   tree
(
a0[3..0],a2[3..0]   :    INPUT;
a4[3..0]            :    INPUT;
b0[3..0],b2[3..0]   :    INPUT;
b4[3..0]            :    INPUT;
c0[3..0],c2[3..0]   :    INPUT;
c4[3..0]            :    INPUT;
d0[3..0],d2[3..0]   :    INPUT;
d4[3..0]            :    INPUT;
e0[3..0],e2[3..0]   :    INPUT;
e4[3..0]            :    INPUT;
clock,reset         :    INPUT;
o2[7..0]            :    OUTPUT;
)
VARIABLE
p0[7..0],p2[7..0]   :    DFF;
p4[7..0],f0[7..0]   :    DFF;
f2[7..0],f4[7..0]   :    DFF;
t0[7..0],t2[7..0]   :    DFF;
t4[7..0],z0[7..0]   :    DFF;
z2[7..0],z4[7..0]   :    DFF;
x0[7..0],x2[7..0]   :    DFF;
x4[7..0]            :    DFF;
(aa[7..0],ac[7..0],ae[7..0])  :    NODE;
(ba[7..0],bc[7..0],be[7..0])  :    NODE;
(ca[7..0],cc[7..0],ce[7..0])  :    NODE;
(da[7..0],dc[7..0],de[7..0])  :    NODE;
(ea[7..0],ec[7..0],ee[7..0])  :    NODE;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
u0[7..0],u1[7..0]                    :     NODE;
u2[7..0]                             :     NODE;
BEGIN
(p0[],p2[],p4[],f0[],f2[],f4[]).clk      =     clock;
(t0[],t2[],t4[],x0[],x2[],x4[]).clk      =     clock;
(z0[],z2[],z4[]).clk                     =     clock;
(p0[],p2[],p4[],f0[],f2[],f4[]).clrn     =     reset;
(t0[],t2[],t4[],x0[],x2[],x4[]).clrn     =     reset;
(z0[],z2[],z4[]).clrn                    =     reset;
(p0[7..4],t0[7..4]).d     =     GND;
(x0[7..4],f0[7..4]).d     =     GND;
 z0[7..4].d               =     GND;
p0[3..0].d                =     a0[];
t0[3..0].d                =     b0[];
x0[3..0].d                =     c0[];
f0[3..0].d                =     d0[];
z0[3..0].d                =     e0[];
aa[]      =     p0[].q;
ba[]      =     t0[].q;
ca[]      =     x0[].q;
da[]      =     f0[].q;
ea[]      =     z0[].q;

     u0[]      =     (aa[]+ba[]+ca[]+da[]+ea[]);
(p4[7..4],t4[7..4]).d     =     GND;
(x4[7..4],f4[7..4]).d     =     GND;
 z4[7..4].d               =     GND;
p4[3..0].d      =     a4[];
t4[3..0].d      =     b4[];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
x4[3..0].d      =     c4[];
f4[3..0].d      =     d4[];
z4[3..0].d      =     e4[];
ae[]            =     p4[].q;
be[]            =     t4[].q;
ce[]            =     x4[].q;
de[]            =     f4[].q;
ee[]            =     z4[].q;
     u1[]       =     (ae[]+be[]+ce[]+de[]+de[]);
(p2[7..5],p2[0]).d  =     GND;
(t2[7..5],t2[0]).d  =     GND;
(x2[7..5],x2[0]).d  =     GND;
(f2[7..5],f2[0]).d  =     GND;
(z2[7..5],z2[0]).d  =     GND;
p2[4..1].d          =     a2[];
t2[4..1].d          =     b2[];
x2[4..1].d          =     c2[];
f2[4..1].d          =     d2[];
z2[4..1].d          =     e2[];
ac[]                =     p2[].q;
bc[]                =     t2[].q;
cc[]                =     x2[].q;
dc[]                =     f2[].q;
ec[]                =     z2[].q;
     u2[]       =     (ac[]+bc[]+cc[]+dc[]+ec[]);
IF (u2[]>(u0[]+u1[]))THEN
     o2[]       =     u2[]-(u0[]+u1[]);
ELSE
     o2[]       =     GND;
```

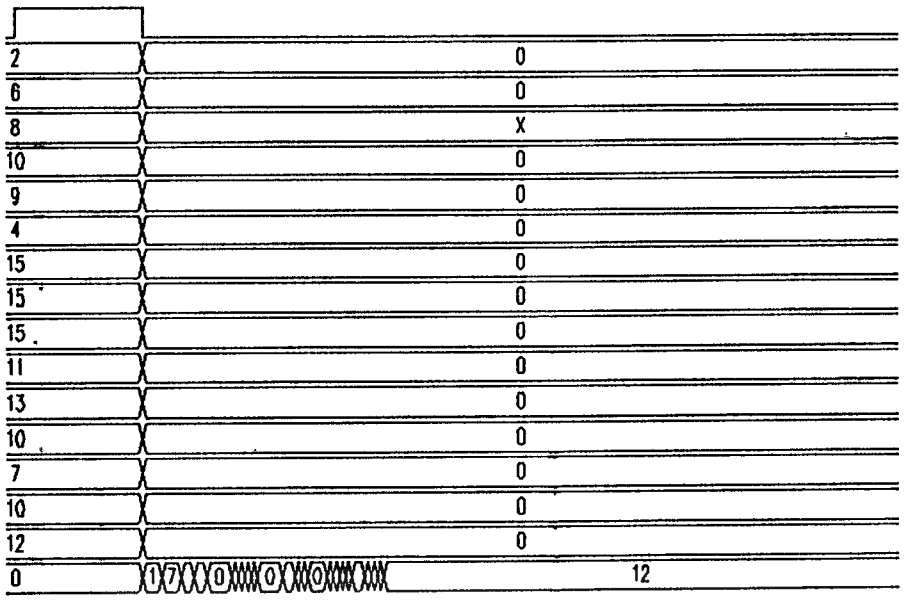
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

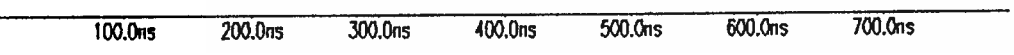
```
END IF;

END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการ Simulate ของ Module TREE.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้





เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการ Simulate ของวงจร Module ที่ 3 คือวงจร TREE.TDF จะพบว่า เวลาที่ใช้ในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 3 จะใช้เวลาเท่ากับ 400 ns ซึ่งมีค่าเท่ากับ 0.0004 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ

(64 x 64) x 0.0004 = 1.6384 millisecond (ms)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

List Program ของวงจร Module FOUR.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
SUBDESIGN  four
(
a0[3..0],a2[3..0]    :     INPUT;
a3[3..0],b1[3..0]    :     INPUT;
b3[3..0],b4[3..0]    :     INPUT;
c0[3..0],c2[3..0]    :     INPUT;
c4[3..0],d0[3..0]    :     INPUT;
d1[3..0],d3[3..0]    :     INPUT;
e1[3..0],e2[3..0]    :     INPUT;
e4[3..0]             :     INPUT;
clock,reset          :     INPUT;
o3[7..0]             :     OUTPUT;
)
VARIABLE
p0[7..0],p2[7..0]    :     DFF;
p3[7..0],f0[7..0]    :     DFF;
f1[7..0],f3[7..0]    :     DFF;
t1[7..0],t3[7..0]    :     DFF;
t4[7..0],z1[7..0]    :     DFF;
z2[7..0],z4[7..0]    :     DFF;
x0[7..0],x2[7..0]    :     DFF;
x4[7..0]             :     DFF;
(aa[7..0],ac[7..0],ad[7..0])  :     NODE;
(da[7..0],db[7..0],dd[7..0])  :     NODE;
(bb[7..0],bd[7..0],be[7..0])  :     NODE;
(eb[7..0],ec[7..0],ed[7..0])  :     NODE;
(ca[7..0],cc[7..0],ce[7..0])  :     NODE;
u0[7..0],u1[7..0]             :     NODE;
u2[7..0]                      :     NODE;
BEGIN
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
(p0[],p2[],p3[],f0[],f1[],f3[]).clk     =     clock;
(t1[],t3[],t4[],z1[],z2[],z4[]).clk     =     clock;
(x0[],x2[],x4[]).clk                    =     clock;


(p0[7..5],p0[0]).d          =     GND;
(t1[7..5],t1[0]).d          =     GND;
(x2[7..5],x2[0]).d          =     GND;
(f3[7..5],f3[0]).d          =     GND;
(z4[7..5],z4[0]).d          =     GND;
p0[4..1].d      =     a0[];
t1[4..1].d      =     b1[];
x2[4..1].d      =     c2[];
f3[4..1].d      =     d3[];
z4[4..1].d      =     e4[];
aa[]            =     p0[].q;
bb[]            =     t1[].q;
cc[]            =     x2[].q;
dd[]            =     f3[].q;
ee[]            =     z4[].q;
     u2[]       =     (aa[]+bb[]+cc[]+dd[]+ee[]);
(p2[7..4],t3[7..4]).d       =     GND;
(x4[7..4],f0[7..4]).d       =     GND;
z1[7..4].d                  =     GND;
p2[3..0].d      =     a2[];
t3[3..0].d      =     b3[];
x4[3..0].d      =     c4[];
f0[3..0].d      =     d0[];
z1[3..0].d      =     e1[];
eb[]            =     z1[].q;
ac[]            =     p2[].q;
```

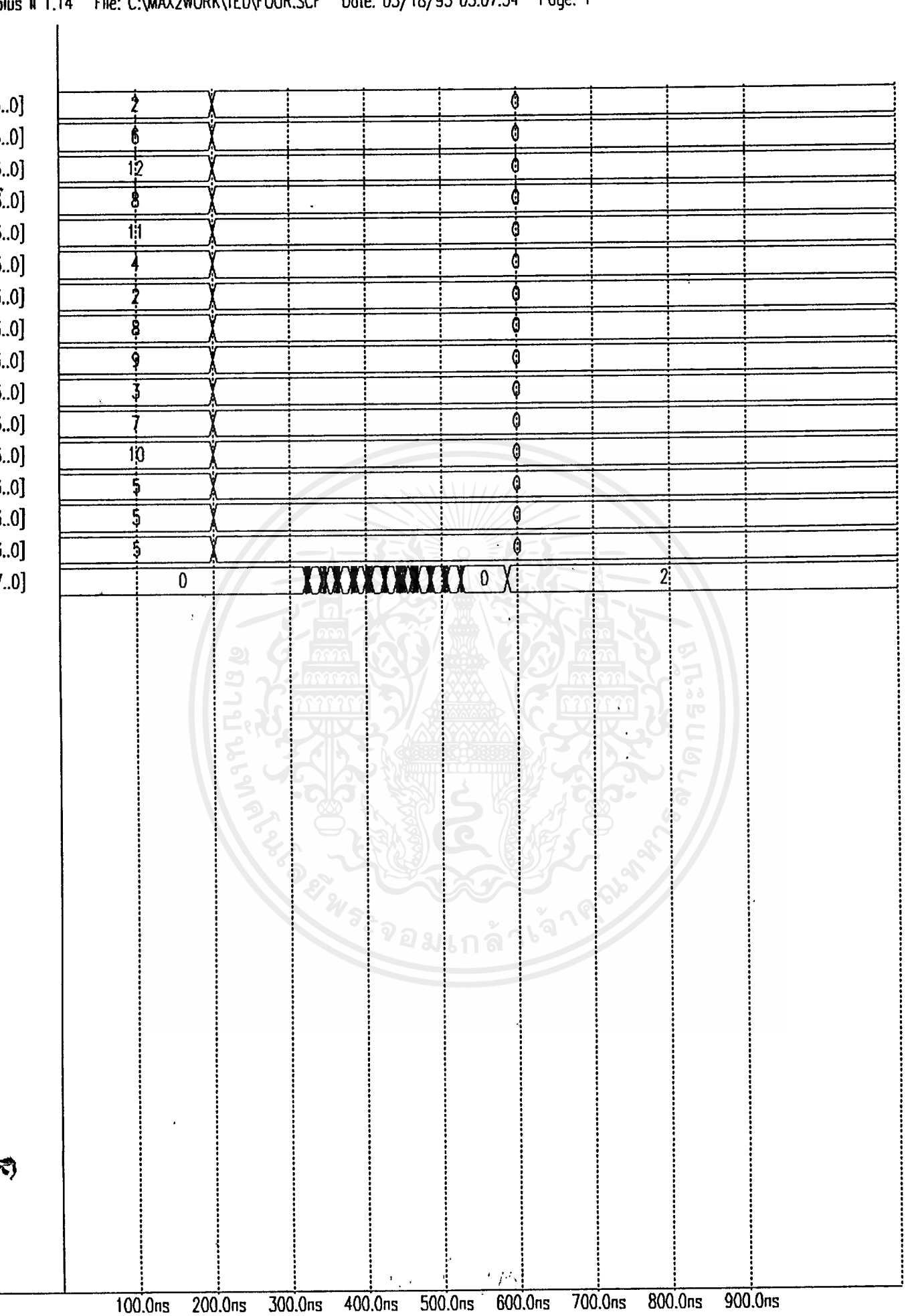
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
bd[]            =    t3[].q;
ce[]            =    x4[].q;
da[]            =    f0[].q;
     u1[]       =    (ac[]+bd[]+ce[]+da[]+eb[]);
(p3[7..4],t4[7..4])       =    GND;
(x0[7..4],f1[7..4])       =    GND;
z2[7..4]                  =    GND;
p3[3..0]        =    a3[];
t4[3..0]        =    b4[];
x0[3..0]        =    c0[];
f1[3..0]        =    d1[];
z2[3..0]        =    e2[];

     u0[]       =    (p3[]+t4[]+x0[]+f1[]+z2[]);
IF (u2[]>(u0[]+u1[]))THEN
     o3[]       =    u2[]-(u0[]+u1[]);
 ELSE
     o3[]       =    GND;
 END IF;
 END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการ Simulate ของ Module FOUR.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการ Simulate ของวงจร Module ที่ 4 คือวงจร FOUR.TDF จะพบว่า
เวลาที่ใช้ในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 4 จะใช้เวลา
เท่ากับ 600 ns ซึ่งมีค่าเท่ากับ 0.0006 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ

(64 x 64) x 0.0006 = 2.4576 millisecond (ms)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

List Program ของวงจร Module MAX.TDF

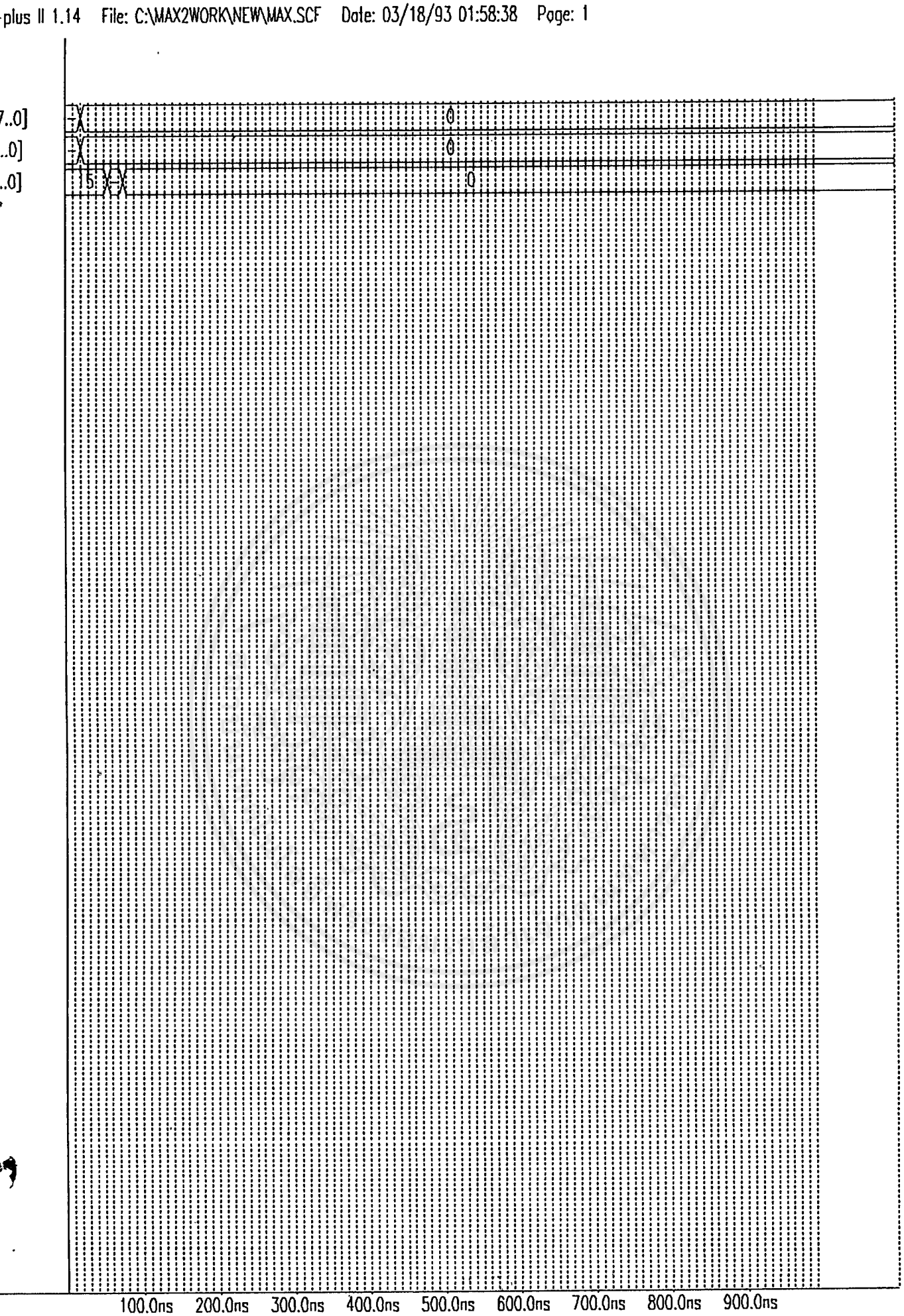
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
FUNCTION  8MCOMPB(a[7..0],b[7..0])
          RETURNS(altb,aeqb,agtb,aeb[7..0]);
SUBDESIGN   max
(
m[7..0],n[7..0]      :     INPUT;
y[7..0]              :     OUTPUT;
)
VARIABLE
w0    :     NODE;
maxw  :     8MCOMPB;
BEGIN
maxw.a[]        =     m[];
maxw.b[]        =     n[];
w0              =     maxw.agtb;
IF(w0==VCC)THEN
          y[]   =     m[];
ELSE
          y[]   =     n[];
END IF;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการ Simulate ของ Module MAX.TDF

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากผลการ Simulate ของวงจร Module ที่ 5 คือวงจร MAX.TDF จะพบว่าเวลาที่ใช้ในการ Simulate จนกระทั่งได้ค่า output ของ Module ที่ 5 จะใช้เวลาเท่ากับ 80 ns ซึ่งมีค่าเท่ากับ 0.00008 millisecond (ms)

ดังนั้นถ้าภาพมีขนาดเท่ากับ 64 x 64 จุดก็จะใช้เวลารวมทั้งหมดเท่ากับ

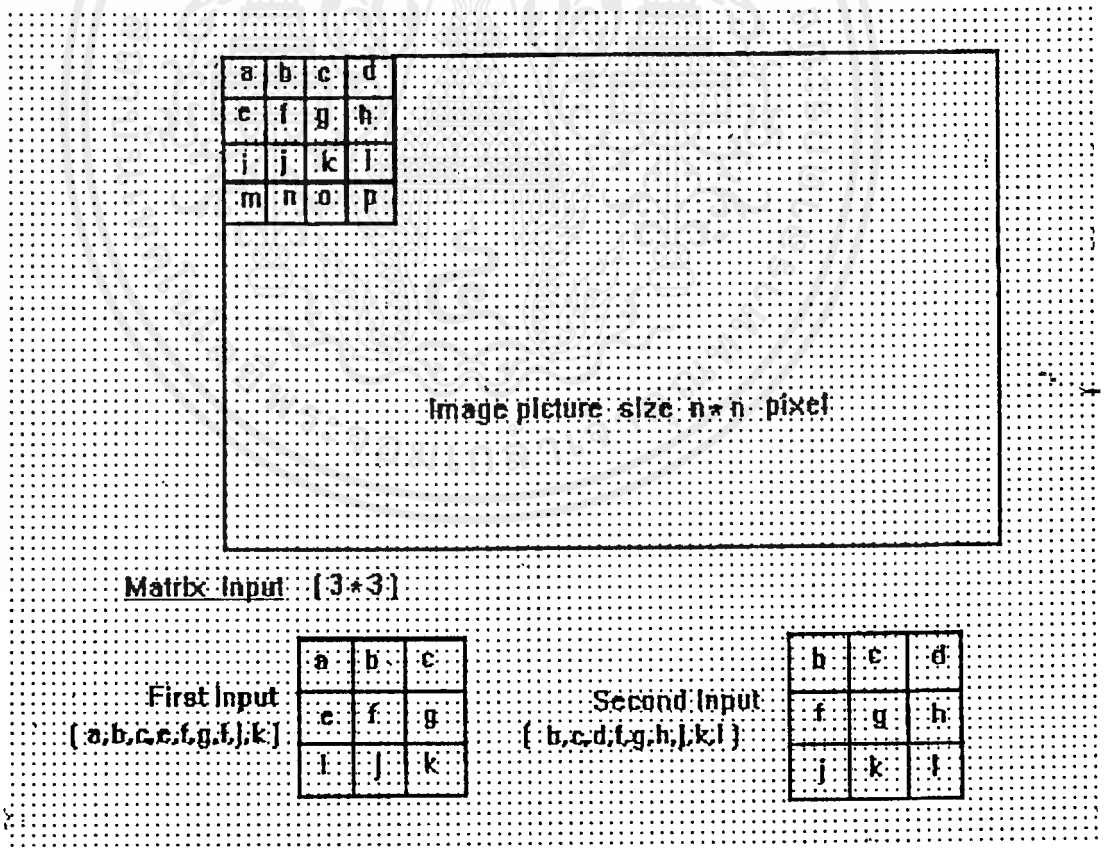
(64 x 64) x 0.00008 = 0.32768 millisecond (ms)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 10

การออกแบบวงจร Median Filter

Median Filter หรือ 2-dimension non-linear filter Circuit เป็นวงจรที่ใช้กรองสัญญาณรบกวน (noise) ที่ติดมากับภาพ (Image) ที่เราต้องการเพื่อที่จะทำให้ได้ภาพที่มีสัญญาณรบกวนที่น้อยที่สุด เนื่องจากในภาพทั่วๆ ไป จะมีสัญญาณรบกวนที่ติดมาในภาพเสมอ โดยที่สัญญาณรบกวนนี้อาจทำให้จุดต่างๆ ในภาพมีความแตกต่างกว่าจุดของภาพปกติ ซึ่งอาจจะทำให้จุดมีความเข้ม (gray level) มากขึ้น หรือมีความเข้มลดลง ในการแก้ปัญหาเรื่องนี้แบบหนึ่งก็คือ พยายามหาทางลดความแตกต่างของความเข้มของจุดในภาพ ให้มีค่าใกล้เคียงกันมากที่สุด โดยใช้วิธีการ นำเข้ากลุ่มของจุดในภาพ Image ทีละกลุ่มไปเรื่อยๆ จนกระทั่งหมดทั้งภาพ ในวงจรนี้จะนำเข้าทีละ 9 จุดและมีรูปแบบเป็นเมตริกซ์ 3x3 ดังภาพประกอบที่ 10.1



รูปที่ 10.1 แสดงการนำเข้ากลุ่มเมตริกซ์ของภาพ Image

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปที่ 10.1 เมื่อนำเข้ากลุ่ม เมตริกซ์ 3x3 ชุดแรกมาแล้ว ก็จะทำการเปลี่ยนแปลง ค่าของจุดกลาง ของกลุ่ม เมตริกซ์นี้ (จุด f) ให้เป็นค่ากลาง (median) โดยค่า median ได้มาจากการวิเคราะห์จากกลุ่มจุดทั้ง 9 จุดใน เมตริกซ์นี้

ดังนั้นจุดกลางของกลุ่ม เมตริกซ์ (จุด f) ก็จะถูกแทนด้วยค่า median ซึ่งก็เป็นการทำให้ลดความแตกต่างของจุดในภาพได้จุดหนึ่ง ต่อไปก็จะนำเข้ากลุ่ม เมตริกซ์กลุ่มถัดไปโดย เลื่อนจากตำแหน่งเดิมมาทางขวา 1 จุด และก็หาค่า median ของกลุ่ม เมตริกซ์นั้น ๆ เพื่อนำมาแทนในจุดกลางของ เมตริกซ์ และก็ทำแบบนี้ไปเรื่อยๆ จนกว่าจะจบแถวในแนวนอน เมื่อจบแถวในแนวนอนแล้ว ก็จะ เลื่อนลงมาข้างล่างอีกทีละจุด แล้วทำตามขั้นตอนข้างบนไปเรื่อยๆ จนครบทั้งภาพก็จะสำเร็จขั้นตอนในการลดสัญญาณรบกวนของภาพ 1 ภาพ ดังตัวอย่างในรูปที่ 10.2 ซึ่งแสดงตัวอย่างการหา median filter ของภาพ Image ขนาด 8x16 ซึ่งรูปจะแสดงให้ดูภาพก่อนและหลังการหา median filter

Example : median fliter (size 8 * 16)

[before]

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	3	0	0	2	2	1	0	0	3	8	0	0	0
0	0	1	f	f	f	a	a	e	0	0	2	2	1	1	0
0	1	1	f	f	f	a	b	c	2	2	5	6	0	0	0
0	3	0	e	e	f	b	b	b	f	6	7	8	0	1	1
0	4	1	f	f	f	e	e	e	e	8	6	6	0	0	0
0	3	f	e	a	8	b	b	b	a	a	a	f	0	6	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

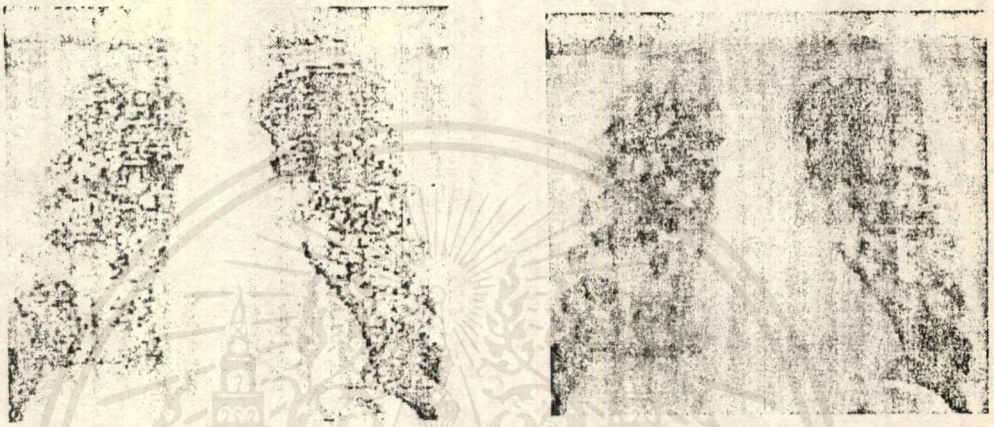
[after]

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	2	0	0	0	0	3	1	0	0	0
0	0	1	3	f	a	a	a	2	1	2	2	2	0	0	0
0	0	1	e	f	f	b	b	b	2	2	5	6	0	0	0
0	1	3	e	f	f	e	b	b	c	6	6	6	0	0	1
0	3	4	e	e	e	b	b	b	b	a	8	6	1	0	0
0	0	3	a	a	a	b	b	b	a	8	6	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

รูปที่ 10.2 แสดงตัวอย่างการหา median filter

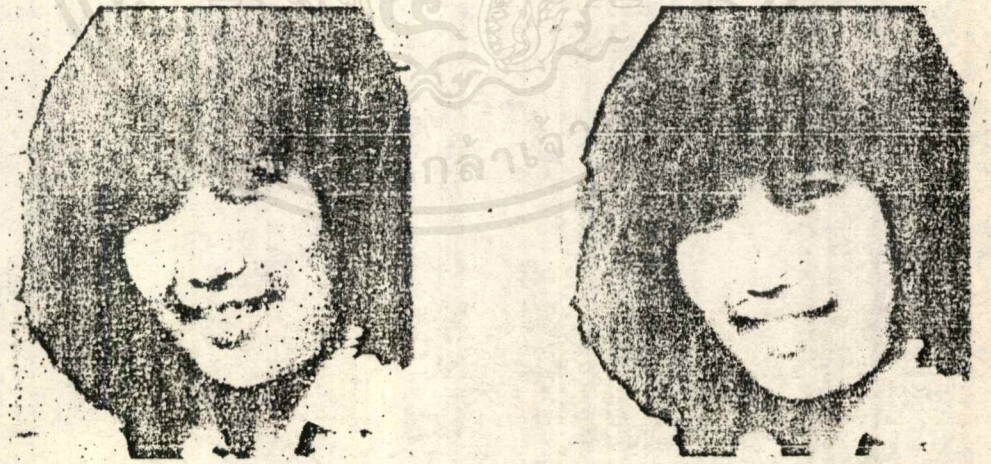
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูปจะเห็นว่าถ้ามีสัญญาณรบกวนเกิดขึ้นในภาพเช่น จุดที่มีความแตกต่างกับจุดรอบข้างอย่างมาก จุดสัญญาณรบกวนนี้ จะถูกแทนด้วยค่า median ของเมตริกซ์ ที่สัมพันธ์กันระหว่าง จุดสัญญาณรบกวนและจุดรอบข้างของสัญญาณรบกวนนั้น ทำให้จุดที่มีสัญญาณรบกวนหายไป หรือสัญญาณรบกวนอาจลดน้อยลง ซึ่งจะเห็นว่าโดยรวมทั้งหมดจะมีความกลมกลืนกันทั้งภาพ หรือดังแสดงตัวอย่างกับรูปภาพจริงๆ ในรูปที่10.3 และ10.4



Input image　　Output image

รูปที่10.3 แสดงตัวอย่างการหา median filter ของภาพ X-rays



Input image
(Salt and pepper noise was added.)

Output image

รูปที่10.4 แสดงตัวอย่างการหา median filter ของภาพที่เติมสัญญาณรบกวนเข้าไป

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.1 ลักษณะของวงจร Median Filter

วงจรนี้ input จะเป็น image data ซึ่งจะเป็นภาพ image ขนาดใดก็ได้ และมีระดับความแตกต่างของสี (gray level) 16 ระดับ หลักการของวงจรก็คือด้านรับข้อมูลเข้าของวงจร (input) จะรับจุดของภาพเข้ามา เป็นเมตริกซ์ขนาด 3*3 แล้วทำการวิเคราะห์หาค่า median ของเมตริกซ์ทั้ง 9 จุดนี้ และทางด้านออกของวงจร (output) ก็คือค่า median ของเมตริกซ์ทั้ง 9 จุด เมื่อทำเมตริกซ์แรกเสร็จเรียบร้อยแล้ว ก็ให้ค่าเมตริกซ์ถัดต่อไป ทำอย่างนี้ไปจนครบทั้งภาพ

10.2 algorithm ของวงจร Median Filter

1. ให้ input เป็นค่าเลขฐาน 2 จำนวน 4 บิท (gray level 16 ระดับ) เข้ามาเป็นเมตริกซ์ 3*3 ซึ่งจะเรียกว่า window

2. หา histogram ของ window นี้โดยวงจรหา histogram และส่งค่าที่ได้ออกมา

3. นำค่าที่ได้จากวงจร histogram มาเข้าวงจรหาค่า median ค่าที่ได้ออกมาคือ output ของวงจร

4. จะสิ้นสุดเมื่อใส่ window จนกระทั่งครบทั้งภาพ

10.3 โครงสร้างของวงจร

โครงสร้างโดยรวมของวงจรมีลักษณะดังรูปที่ 10.5

ที่ด้าน input ของวงจรจะมีขา glevel[3..0] ซึ่งมีขนาด 4 ขา แต่ละขาใช้แทนเลขฐาน 2 จำนวน 1 บิท ดังนั้น glevel[3..0] จึงใช้รับค่าจุดของภาพที่เข้ามาเป็นระดับความเข้มได้ 16 ระดับ (4 บิท) และจุดของภาพที่เข้ามานี้ จะเข้ามาทีละจุดจนครบ 9 จุด (1 เมตริกซ์)

ขา trig คือขา clock synchronous input ทำหน้าที่ ให้ clock เข้าไปพร้อมกับ glevel[3..0] เพื่อกระตุ้นให้วงจรทำงาน คือรับค่า glevel[3..0] เข้ามาทำการประมวลผล

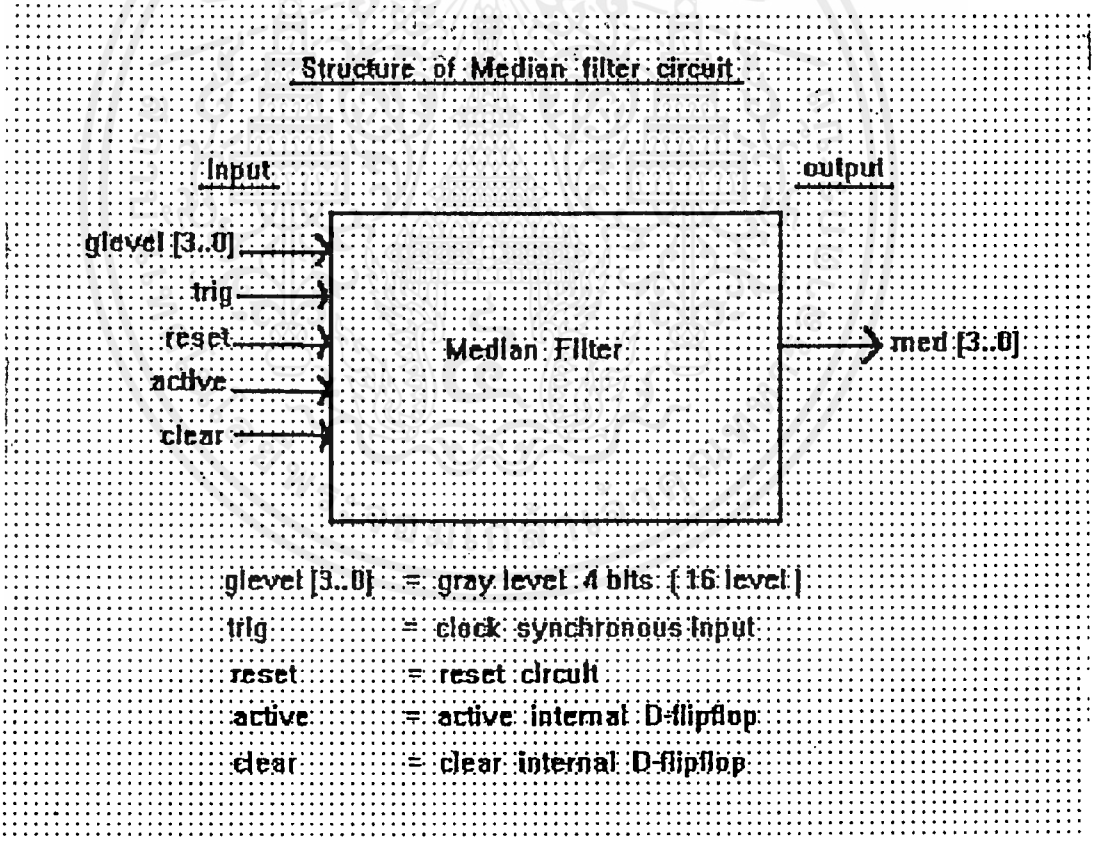
ขา reset เป็นขาที่ทำหน้าที่ ลบล้างข้อมูลทุกอย่างในวงจร จะลบข้อมูลเมื่อป้อนค่า 1 เข้าไป โดยปกติขณะที่กำลังป้อน glevel[3..0] เข้าไป ขา reset นี้จะต้องเท่ากับ 0 ตลอด

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ขา active คือขาที่ใช้กระตุ้นการทำงานของ d-flipflop ภายในวงจร เมื่อป้อนค่า glevel[3..0] เข้าไปจนครบ 9 ค่า แล้ว ก็จะให้ clock แก่ขานี้ 1 ลูก เพื่อให้วงจรทำการประมวลผลหาค่า median ออกมา

ขา clear เป็นขาที่ใช้สำหรับ clear วงจร โดยจะลบข้อมูลใน d-flipflop ทุกตัวของวงจร โดยจะต้องป้อนเข้าไปพร้อมกับขา reset เสมอ แต่ขานี้จะทำงานหรือทำการลบข้อมูล เมื่อป้อนค่า 0 เข้าไป ถ้าเป็นขณะกำลังป้อนค่า glevel[3..0] ขานี้จะต้องมีค่าเป็น 1 ตลอด

ส่วนด้าน output มีขา med[3..0] ซึ่งเป็นค่า median ของเมตริกซ์ 3x3 ที่ป้อนเข้าไป โดยขา med[3..0] นี้ ก็จะมีขนาด 4 บิทเช่นเดียวกับ glevel[3..]

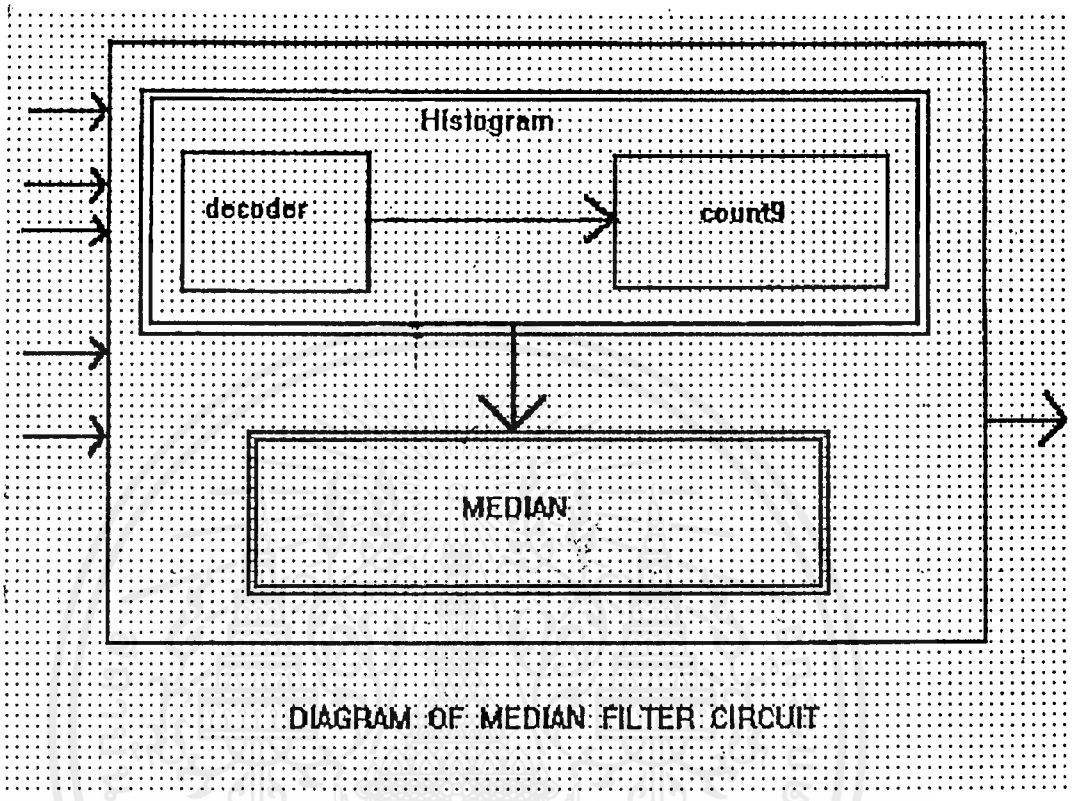


รูปที่ 10.5 แสดงโครงสร้างของวงจร Median Filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

10.4 ลักษณะภายในของวงจร Median Filter

ลักษณะภายในของวงจรสามารถแสดงเป็น block diagram ได้ดังนี้



รูปที่10.6 แสดง diagram ของวงจร Median Filter

จาก diagram ของวงจรจะเห็นว่าวงจรแบ่งออกเป็น 2 ส่วนใหญ่ๆ คือ

1. Histogram จะทำหน้าที่หา Histogram ของเมตริกซ์ ทั้ง 9 จุด โดยมีขั้นตอนคือ เมื่อรับค่า glevel เข้ามา ก็จะนำมาเข้าสู่วงจร decoder เพื่อแยกแยะหาค่าความเข้มของ glevel ว่าอยู่ในระดับเท่าไร เมื่อแยกเสร็จแล้ว ก็จะนำไปเก็บในตัว d-flipflop ชื่อ count9 ซึ่งมีอยู่จำนวน 16 ตัว

เมื่อรับค่าเข้ามาจนครบ 9 จุดแล้ว จะได้ Histogram ของจุด 9 จุด ที่เรียงลำดับจากมากไปน้อย เพื่อส่งต่อให้วงจรหา median ต่อไป

2. Median เป็นวงจรที่ใช้หาค่า median ของเมตริกซ์จุดทั้ง 9 จุด ที่ได้รับจากวงจร Histogram ค่า median ที่ได้ก็จะเป็น output ของวงจร median filter โดยส่งออกไปทางขา med[3..0]

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ผลการทำงานของวงจร

หลังจากคอมไพล์เสร็จ จะได้ report file ออกมา ซึ่งเนื้อหาภายในจะเป็นรายละเอียดต่างๆ ของอุปกรณ์ที่ใช้ในการสร้างวงจร และการเชื่อมต่อกันในวงจร โดยจะมี list ของอุปกรณ์ที่ใช้ใน วงจร ดังรูปที่ 10.7

```
MAX+plus II Compiler Report File
Version 1.14  10/17/91

***** Project compiled without errors

Untitled

** DEVICE SUMMARY **
```

Chip/ POF	Device	Input Pins	Output Pins	Bidir Pins
MEDI	EPM5130J	33	10	0
MEDI1	EPM5128	14	25	0
TOTAL:		47	35	0

รูปที่ 10.7 แสดงอุปกรณ์ EPLD ที่ใช้ในวงจร Median filter

และจากผลการ simulate วงจรจะเห็นว่าเวลาที่ใช้ในการหาค่า median 1 ค่าใช้เวลา 10.0 microsecond (us) ดังนั้นถ้าภาพ Image มีขนาด 64x64 จุด ก็จะใช้เวลารวมทั้งหมดคือ (64x64) x 10 us = 40.96 millisecond (ms)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        % subdesign section %

SUBDESIGN count9
(
  clock,reset,inc        : INPUT;
  q[3..0]                : OUTPUT;
)

        % variable section %

VARIABLE
  count : MACHINE OF BITS (out[3..0])   % define state machines s0-s9 %
        WITH STATES (s0 = 0,
                     s1 = 1,
                     s2 = 2,
                     s3 = 3,
                     s4 = 4,
                     s5 = 5,
                     s6 = 6,
                     s7 = 7,
                     s8 = 8,
                     s9 = 9);

        % logic section %

BEGIN
        % define input and output of circuit %

    q[] = out[];
    count.clk = clock&inc;
    count.reset = reset;

        % find state machine of count %

    CASE (count) IS

        WHEN s0 =>
            count=s1;
        WHEN s1 =>
            count=s2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
            WHEN s2 =>
                count=s3;
            WHEN s3 =>
                count=s4;
            WHEN s4 =>
                count=s5;
            WHEN s5 =>
                count=s6;
            WHEN s6 =>
                count=s7;
            WHEN s7 =>
                count=s8;
            WHEN s8 =>
                count=s9;
            WHEN s9 =>
                count=s0;

        END CASE;

    END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        % subdesign section %

SUBDESIGN  decoder
(
    in[3..0]            : INPUT;
    out[15..0]          : OUTPUT;
)
        % logic section %

BEGIN
        IF (in[] == 0) THEN
                out0 = VCC;
        ELSIF (in[] == 1) THEN
                out1 = VCC;
        ELSIF (in[] == 2) THEN
                out2 = VCC;
        ELSIF (in[] == 3) THEN
                out3 = VCC;
        ELSIF (in[] == 4) THEN
                out4 = VCC;
        ELSIF (in[] == 5) THEN
                out5 = VCC;
        ELSIF (in[] == 6) THEN
                out6 = VCC;
        ELSIF (in[] == 7) THEN
                out7 = VCC;
        ELSIF (in[] == 8) THEN
                out8 = VCC;
        ELSIF (in[] == 9) THEN
                out9 = VCC;
        ELSIF (in[] == 10) THEN
                out10 = VCC;
        ELSIF (in[] == 11) THEN
                out11 = VCC;
        ELSIF (in[] == 12) THEN
                out12 = VCC;
        ELSIF (in[] == 13) THEN
                out13 = VCC;
        ELSIF (in[] == 14) THEN
                out14 = VCC;
        ELSE
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
                out15 = VCC;
        END IF;
END;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
        % call function  count9 and decoder %

FUNCTION  count9 (inc,clock,reset)
        RETURNS (q[3..0]);
FUNCTION  decoder (in[3..0])
    RETURNS (out[15..0]);

        % subdesign section %

SUBDESIGN  medi
(
    glevel[3..0],trig,reset,active,clear    :INPUT;
    med[3..0]                               :OUTPUT;  % output = median valu
e %

)

        % variable section %

VARIABLE

    hist[15..0]             :count9;  % define hist[] is function count9
 %
    h0[3..0],h1[3..0]       :NODE;    % define node %
    h2[3..0],h3[3..0]       :NODE;
    h4[3..0],h5[3..0]       :NODE;
    h6[3..0],h7[3..0]       :NODE;
    h8[3..0],h9[3..0]       :NODE;
    ha[3..0],hb[3..0]       :NODE;
    hc[3..0],hd[3..0]       :NODE;
    he[3..0],hf[3..0]       :NODE;
    a0[3..0],a1[3..0]       :NODE;
    a2[3..0],a3[3..0]       :NODE;
    a4[3..0],a5[3..0]       :NODE;
    a6[3..0],a7[3..0]       :NODE;
    a8[3..0],a9[3..0]       :NODE;
    aa[3..0],ab[3..0]       :NODE;
    ac[3..0],ad[3..0]       :NODE;
    ae[3..0],af[3..0]       :NODE;
    d0[3..0],d1[3..0]       :DFF;     % define D-flipflop %
    d2[3..0],d3[3..0]       :DFF;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งไม่อนุญาตให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
d4[3..0],d5[3..0]          :DFF;
d6[3..0],d7[3..0]          :DFF;
d8[3..0],d9[3..0]          :DFF;
da[3..0],db[3..0]          :DFF;
dc[3..0],dd[3..0]          :DFF;
de[3..0],df[3..0]          :DFF;

     % logic section %

BEGIN

     % define input of circuit %

hist[].inc = decoder(glevel[]);
hist[].clock = trig;
hist[].reset = reset;

     % define output from  histogram %

(h3[],h2[],h1[],h0[])   = hist[3..0].q[];
(h7[],h6[],h5[],h4[])   = hist[7..4].q[];
(hb[],ha[],h9[],h8[])   = hist[11..8].q[];
(hf[],he[],hd[],hc[])   = hist[15..12].q[];

     % find sumation of histogram %

a0[] = b"0000" # h0[];
a1[] = a0[] + h1[];
a2[] = a1[] + h2[];
a3[] = a2[] + h3[];
a4[] = a3[] + h4[];
a5[] = a4[] + h5[];
a6[] = a5[] + h6[];
a7[] = a6[] + h7[];
a8[] = a7[] + h8[];
a9[] = a8[] + h9[];
aa[] = a9[] + ha[];
ab[] = aa[] + hb[];
ac[] = ab[] + hc[];
ad[] = ac[] + hd[];
ae[] = ad[] + he[];
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
af[] = ae[] + hf[];

    % define clock and clear of D-flipflop %

(d0[],d1[],d2[],d3[]).clk   = active;
(d4[],d5[],d6[],d7[]).clk   = active;
(d8[],d9[],da[],db[]).clk   = active;
(dc[],dd[],de[],df[]).clk   = active;
(d0[],d1[],d2[],d3[]).clrn   = clear;
(d4[],d5[],d6[],d7[]).clrn   = clear;
(d8[],d9[],da[],db[]).clrn   = clear;
(dc[],dd[],de[],df[]).clrn   = clear;

    % save value of sumation to D-flipflop %

d0[] = a0[];
d1[] = a1[];
d2[] = a2[];
d3[] = a3[];
d4[] = a4[];
d5[] = a5[];
d6[] = a6[];
d7[] = a7[];
d8[] = a8[];
d9[] = a9[];
da[] = aa[];
db[] = ab[];
dc[] = ac[];
dd[] = ad[];
de[] = ae[];
df[] = af[];

      % find median value of circuit %

IF (d0[] > 4) THEN
    med[] = 0;
ELSIF (d1[] > 4) THEN
       med[] = 1;
ELSIF (d2[] > 4) THEN
       med[] = 2;
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

```
    ELSIF (d3[] > 4) THEN
            med[] = 3;
    ELSIF (d4[] > 4) THEN
            med[] = 4;
    ELSIF (d5[] > 4) THEN
            med[] = 5;
    ELSIF (d6[] > 4) THEN
            med[] = 6;
    ELSIF (d7[] > 4) THEN
            med[] = 7;
    ELSIF (d8[] > 4) THEN
            med[] = 8;
    ELSIF (d9[] > 4) THEN
            med[] = 9;
    ELSIF (da[] > 4) THEN
            med[] = 10;
    ELSIF (db[] > 4) THEN
            med[] = 11;
    ELSIF (dc[] > 4) THEN
            med[] = 12;
    ELSIF (dd[] > 4) THEN
            med[] = 13;
    ELSIF (de[] > 4) THEN
            med[] = 14;
    ELSE
            med[] = 15;
    END IF;

END;
```

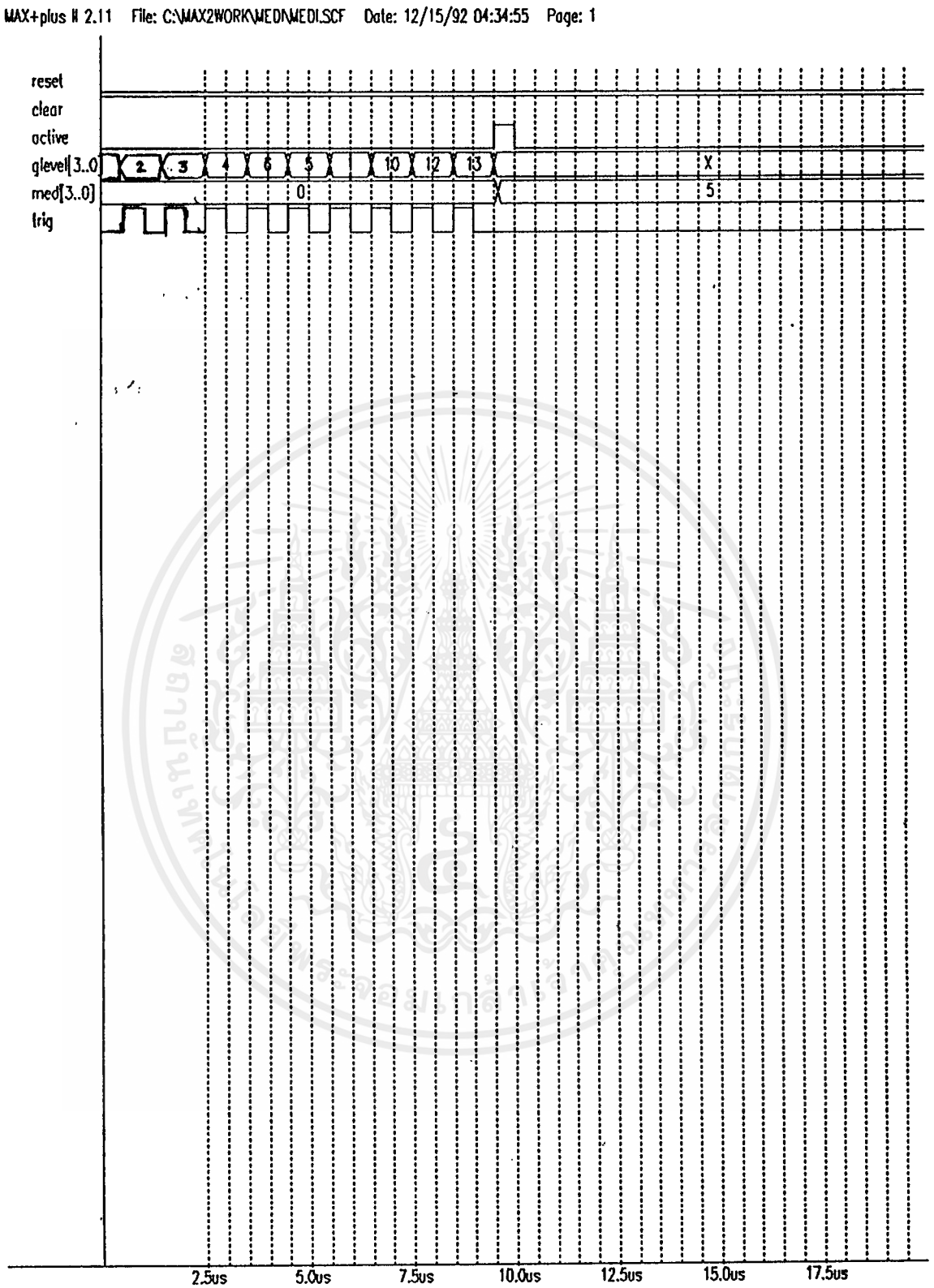
เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากรูป Wave form ของการ Simulate วงจร median filter

1. ป้อนค่า trig เข้าไปโดยใช้เวลาลูกละ 1 usec จะป้อนเข้าไป 9 ลูกซึ่งเท่ากับจำนวนของ จุด image 9 จุด
2. ป้อนค่า reset ให้เป็น 0
3. ป้อนค่า clear ให้เป็น 1
4. ป้อนค่า glevel[] เข้าไป 9 ค่า ซึ่งเท่ากับป้อนค่า matrix ของจุด 9 จุด เข้าไป
5. ให้ค่า active เป็น 1 หลังจากป้อนค่า glevel[] ครบ 9 จุดแล้ว

และหลังจาก simulate เสร็จเรียบร้อย จะได้ผลลัพธ์ซึ่งก็คือ ค่า median ออกมาเป็น output ของวงจร median filter

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทวิจารณ์และสรุป

AHDL เป็นภาษาที่ใช้ในการบรรยายหรืออธิบายรูปแบบการทำงานของวงจรที่ง่ายแก่การเข้าใจและศึกษา เป็นภาษาที่ช่วย อำนวยความสะดวกในเรื่องการออกแบบโปรแกรมทำให้การออกแบบโปรแกรมง่ายขึ้น สะดวกขึ้น อีกทั้งยังประหยัดเวลา สามารถที่จะทดสอบวงจรได้ทันทีโดยการ simulate

อุปสรรคที่สำคัญในการทำโครงงานนี้คือ เรื่องเวลาที่ต้องใช้ในการ compile โปรแกรม คือถ้าวงจรนั้นมีความซับซ้อน โปรแกรมจะใช้เวลาในการ compile นานมาก เป็นชั่วโมง หรือบางทีก็ compile ไม่ได้เลย โดยแจ้งข่าวสารออกมาว่า วงจรนั้นซับซ้อนมากเกินไป

การแก้ปัญหาในเรื่องนี้สามารถแก้ปัญหาได้เพียงระดับหนึ่งเท่านั้น คือ ได้ทำการเปลี่ยนเครื่องคอมพิวเตอร์ให้มีสมรรถนะที่สูงขึ้น จากเดิมที่ใช้เครื่องระดับ 386 ก็ได้เปลี่ยนเป็นเครื่องระดับ 486 ซึ่งก็ทำให้มีความเร็วขึ้นมาอีกนิดหน่อยแต่ก็ยังใช้เวลาเกือบชั่วโมง

สำหรับการทำงานของวงจรที่ได้ออกแบบทั้ง 3 วงจร สามารถทำงานได้ตรงตามความต้องการ และเมื่อลองเปรียบเทียบเรื่องเวลา (อยู่ในบทที่ 6) ระหว่างวงจรที่ออกแบบกับการใช้ซอฟต์แวร์ภาษา c ทำงานตาม algorithm ของวงจร จะเห็นว่าวงจรที่ออกแบบโดยใช้ AHDL มีความเร็วในการทำงานเร็วกว่า ซอฟต์แวร์ภาษา C อยู่มากซึ่งก็เป็นไปตามวัตถุประสงค์ ที่ต้องการออกแบบวงจรเพื่อให้มีการทำงานที่รวดเร็วขึ้นกว่าการทำงานของซอฟต์แวร์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บรรณานุกรม

1. A D&T Roundtable, "Behavioral Description Language", IEEE Design & Test of computers, Feb. 1990 .
2. MAX+PLUS II, "Text Editor & AHDL", version 1.0, ALTERA Corporation, April 1991.
3. MAX+PLUS II, "User Guide", version 1.0, ALTERA Corporation, April 1991.
4. MAX+PLUS II, "Compiler", version 1.0, ALTERA Corporation, April 1991.
5. MAX+PLUS II, "Simulator,Timing Analyzer & Waveform Editor", version 1.0, ALTERA Corporation, April 1991.
6. Rafael C. Gonzalez, " Digital Image Processing ", Second Edition, Paul Wintz Addison-Wesley publishing company.
7. SPIDER Subroutine Package for Image Data Enhancement and Recognition Users' Manual, JSD joint system development corp.

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหาและต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้