

# DIGITAL MULTIPLEXER

พงษ์ศักดิ์ สุสัมพันธ์ไพบูลย์.

“  
ในระบบเครือข่ายสื่อสารที่มีการส่งข้อมูลจำนวนมากผ่านสื่อสัญญาณต่างๆ ที่มีอยู่ในเครือข่าย จะมีอุปกรณ์ชนิดหนึ่งเรียกว่า Multiplexer ทำหน้าที่ในการรวบรวมสัญญาณหรือข้อมูลจากแหล่งต่างๆ ก่อนที่จะส่งไปในสื่อสัญญาณเดียวกันทางด้าน Output หรือ Aggregate โดยที่ไม่ทำให้เกิดการเปลี่ยนแปลงข้อมูลที่ฝั่งผู้รับ (Transparent) ทำให้การใช้สื่อสัญญาณที่มีอยู่เป็นไปอย่างมีประสิทธิภาพ....  
”

## 1. Multiplexer

ก่อนที่จะกล่าวถึง Digital Multiplexer จะขอกล่าวถึงวิธีการในการ Multiplex ต่างๆ ก่อน คือ Frequency Division Multiplexer, Time Division Multiplexer และ Statistical Time Division Multiplexer

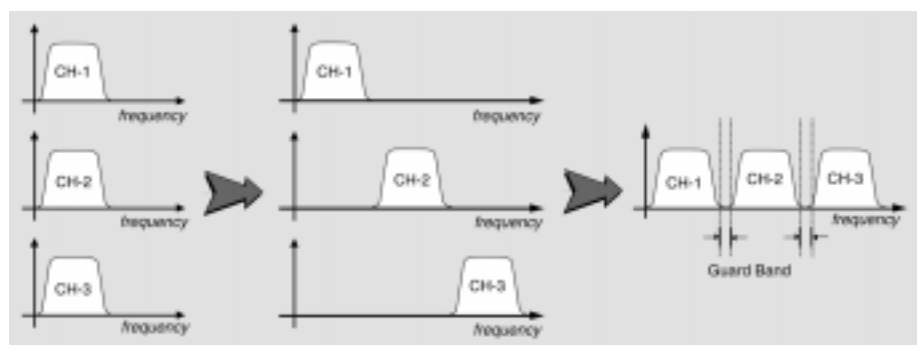
### 1.1 Frequency Division Multiplexing

Frequency Division Multiplexer (FDM) ใช้วิธีการแบ่งความถี่ของช่องสัญญาณออกเป็นความถี่ย่อยหลายๆ ช่อง และใช้ช่องความถี่ย่อยแต่ละช่องสำหรับส่งข้อมูลแต่ละ Channel แต่ละ Channel จะมีความถี่ของตัวเองเรียกว่า Carrier Frequency โดยที่ Carrier Frequency จะถูก Modulate ได้หลายวิธี ซึ่งวิธีการ Modulate และคุณสมบัติของข้อมูลในแต่ละ Channel จะเป็นตัวกำหนด Bandwidth ที่ต้องใช้ในแต่ละ Channel

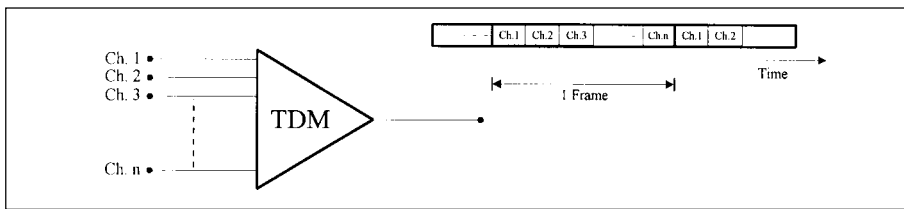
### (รูปที่ 1)

การแบ่งแถบความถี่ออกเป็นช่องความถี่ย่อย อาจเกิดปัญหาการรบกวนกันระหว่างช่องความถี่ที่อยู่ติดกันได้ จึงต้องมีแถบความถี่ระหว่างแถบความถี่ย่อย เรียกว่า Guard Band เพราะเราไม่สามารถสร้างวงจร Filter ที่สามารถแยกความถี่ 2 ช่องความถี่ที่อยู่ติดกันออกจากรันได้อย่างชัดเจน ทำให้ประสิทธิภาพในการใช้แถบความถี่เป็นไปอย่างไม่มีประสิทธิภาพ

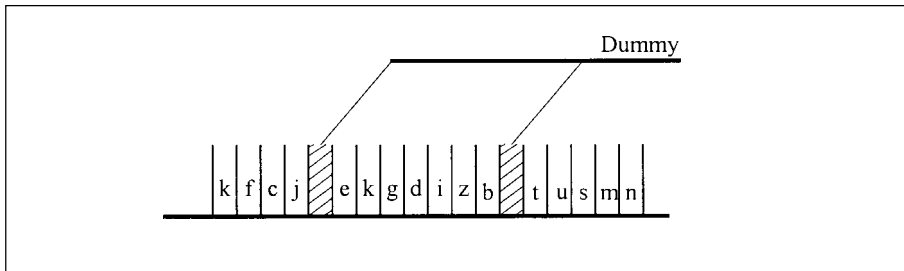
ในการสื่อสารผ่าน Optical Fiber Cable มีการ Multiplexing แบบ FDM โดยใช้ความยาวคลื่นที่แตกต่างกันของแถบแสง สำหรับแต่ละ Channel เรียกว่า Wavelength Division Multiplexing (WDM) และในกรณีที่เป็นการสื่อสารโดยผ่าน Radio Link นอกจากจะแบ่งแถบความถี่ออกเป็นความถี่ย่อยๆ แล้ว ยังมีแบ่งการใช้ Polarize ออกเป็น Vertical



รูปที่ 1 : Frequency Division Multiplexing



รูปที่ 2 : Time Division Multiplexing



รูปที่ 3 : Dummy ใน Time Slot ที่ไม่มีข้อมูลส่งของ TDM

Polarize และ Horizontal Polarize เพื่อให้สามารถใช้แถบความถี่ซึ่งมีอยู่จำกัดเป็นไปอย่างมีประสิทธิภาพ

### 1.2 Time Division Multiplexing

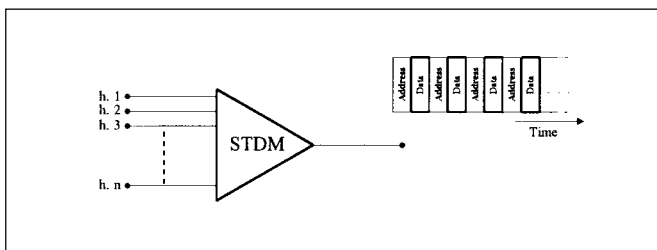
การ Multiplexing แบบ Time Division Multiplexing ทำโดยการแบ่งเวลาในการใช้ช่องทางการสื่อสารให้การส่งข้อมูลของแต่ละ Channel เรียกว่า Time Slot โดยที่แต่ละ Channel จะใช้ Bandwidth ทั้งหมดตามส่วนของเวลาในการรับส่งข้อมูล โดยวิธีการของ TDM จะทำการ Scan ทุกๆ Channel ตามลำดับ ในแต่ละรอบของการ Scan และทุกๆ Time Slot ในหนึ่งรอบการ Scan เรียกว่า Frame (รูปที่ 2)

ขณะที่ Time Slot ใดๆ ไม่มีข้อมูลที่จะส่ง จะต้องมีการใส่ Dummy เข้าไปใน Time Slot นั้น ไม่สามารถที่จะ

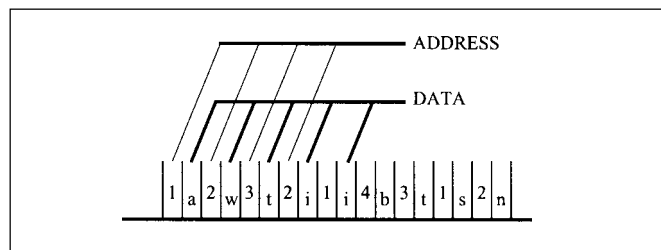
ตัด Time Slot นั้น ออกไปได้ เนื่องจากวิธีการของ TDM ไม่มีการกำหนด Address ที่ปลายทางฝั่งผู้รับ และข้อมูลอยู่ในลักษณะหมุนเวียนตลอดเวลา จึงต้องมีการใส่ Dummy เข้าไป Multiplexer ทางฝั่งรับจึงจะสามารถแยกข้อมูลได้ถูกต้อง (รูปที่ 3)

### 1.3 Statistical Time Division Multiplexing

ในการ Multiplexing แบบ TDM นั้น แต่ละ Channel จะใช้ Bandwidth ทั้งหมดที่มีอย่างต่อเนื่องตลอดเวลา แม้ว่าในขณะที่นั้นจะไม่มีข้อมูลที่จะส่งก็ตาม ทำให้ช่วงเวลานั้นเสียไปโดยไม่เกิดประโยชน์ จึงมีการพัฒนา Statistical Time Division Multiplexing (STDM) ขึ้น หรือเรียกว่า Statistical Multiplexer (Stat Mux) (รูปที่ 4)



รูปที่ 4 : Statistical Time Division Multiplexing



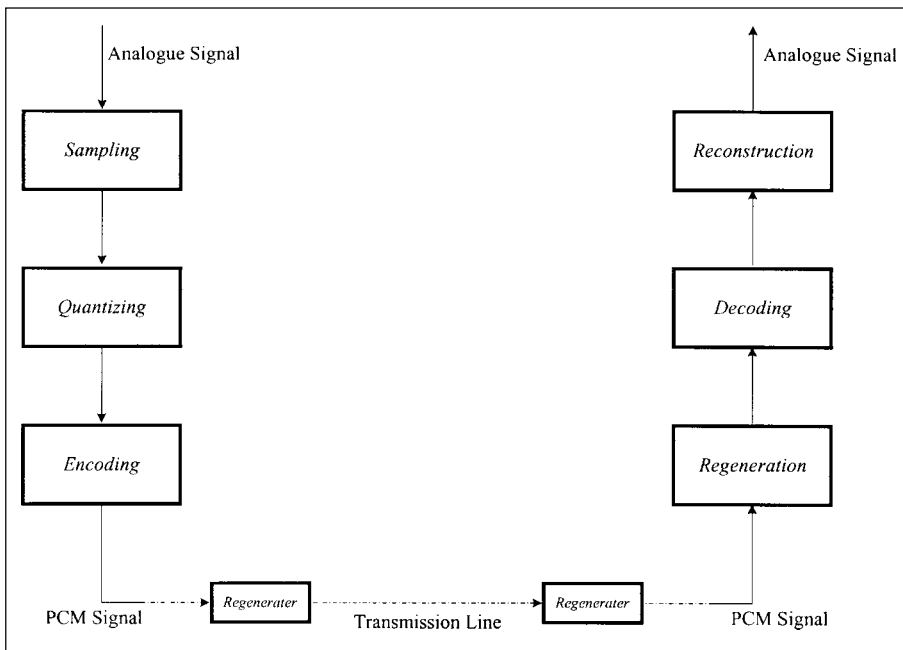
รูปที่ 5 : STDM Frame Structure

STDM จะมีการเพิ่มส่วนที่เป็น Address เข้าไปกับข้อมูลของแต่ละ Channel เพื่อส่งข้อมูลนั้นที่เวลาใดๆ ก็ได้ ทางฝั่งผู้รับจะรู้ว่าเป็นข้อมูลของ Channel ไหน ทำให้ไม่จำเป็นต้องส่งข้อมูลเป็นลำดับต่อเนื่องเหมือนใน TDM แต่จะส่งเฉพาะ Channel ที่มีข้อมูลที่จะส่งเท่านั้น แต่การเพิ่ม Address เข้าไปกับข้อมูล ทำให้ต้องเสีย Bandwidth ส่วนหนึ่งให้กับส่วนที่เป็น Address (รูปที่ 5)

การที่การส่งข้อมูลของแต่ละ Channel ไม่ถูกกำหนดเวลาที่แน่นอน ทำให้สามารถกำหนดอัตราการส่งของข้อมูลในแต่ละ Channel ได้ตั้งแต่อัตราการส่งข้อมูลเท่ากับ 0 จนถึงอัตราการส่งข้อมูลทั้งหมดของ Multiplexer (หรือเท่ากับ Bandwidth รวมทั้งหมด) ทำให้ผลรวมของอัตราการส่งข้อมูลรวมของทุก Channel อาจมากกว่า Bandwidth ของสื่อสัญญาณที่มีอยู่ได้ และด้วยเหตุนี้ทำให้จำเป็นต้องมี Buffer หรือวิธีการ Flow Control ที่เหมาะสมให้กับแต่ละ Channel เพื่อป้องกันการสูญหายของข้อมูล อันเนื่องมาจาก STDM ไม่สามารถส่งข้อมูลออกได้ทัน

STDM จะเรียกอีกชื่อหนึ่งว่า Concentrator เพราะความสามารถในการส่งข้อมูลรวมทุก Channel มากกว่า อัตราการส่งข้อมูลที่ Aggregate ได้ และรวมถึงอุปกรณ์ประเภท Flexible Multiplexer ด้วย

## 2. Digital Multiplexer



รูปที่ 6 : Pulse Code Modulation Function Blocks

ในเครือข่ายโทรศัพท์นั้นการส่งข้อมูลผ่านสื่อสัญญาณต่างๆ เช่น Coaxial Cable, Optical Fiber Cable หรือ Digital Microwave Radio จะใช้วิธีการ Multiplexing แบบ TDM เนื่องจากไม่ยุ่ง

ยากซับซ้อนและมีราคาถูกเหมือนการใช้วิธีการของ FDM โดยวิธีการแปลงสัญญาณเสียงพูดซึ่งอยู่ในรูปของสัญญาณ Analog ให้อยู่ในรูปของข้อมูล Digital ก่อนที่จะทำการ

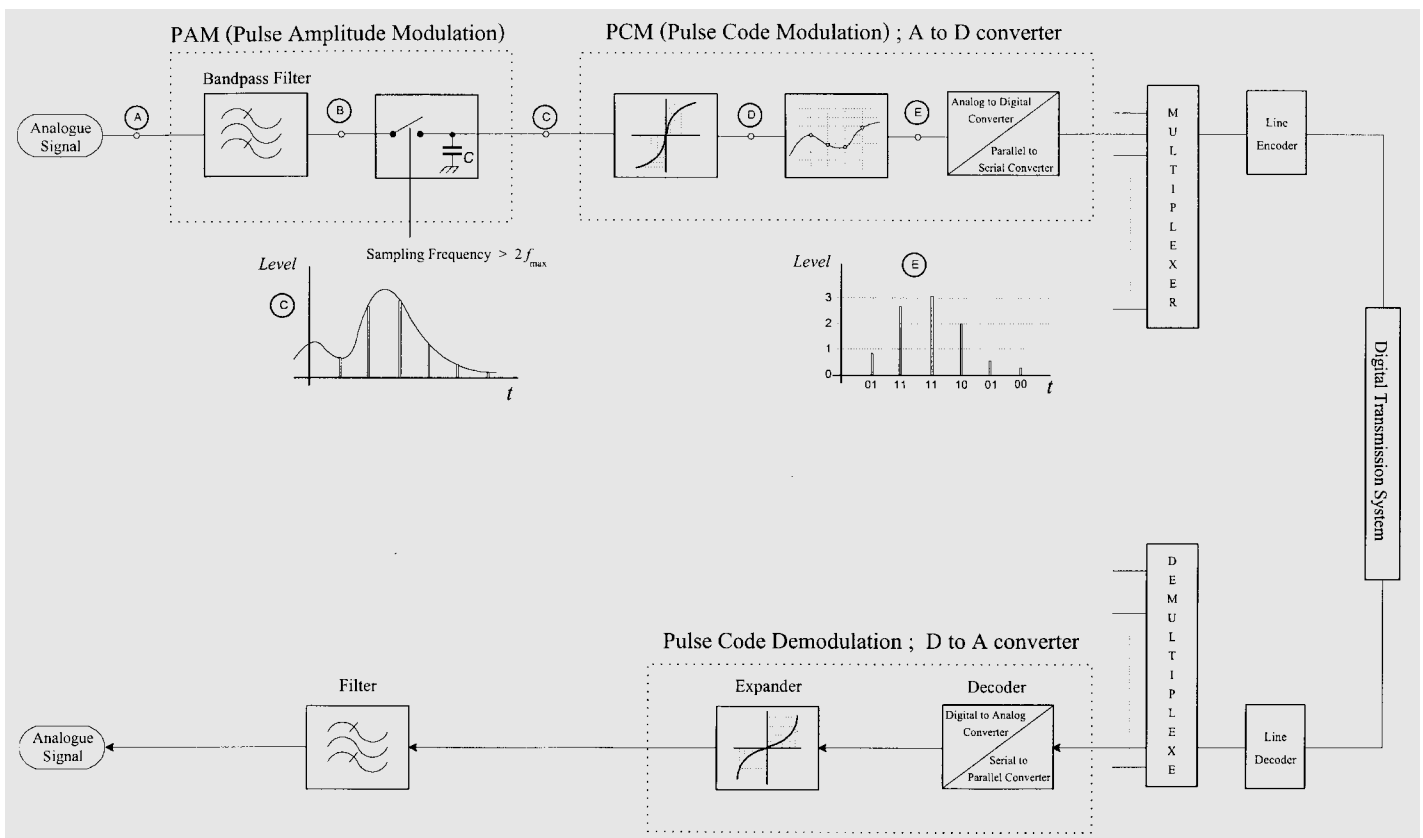
Multiplexing แบบ TDM นั้น จะใช้วิธีการของ Pulse Code Modulation (PCM)

### 2.1 Pulse Code Modulation (PCM)

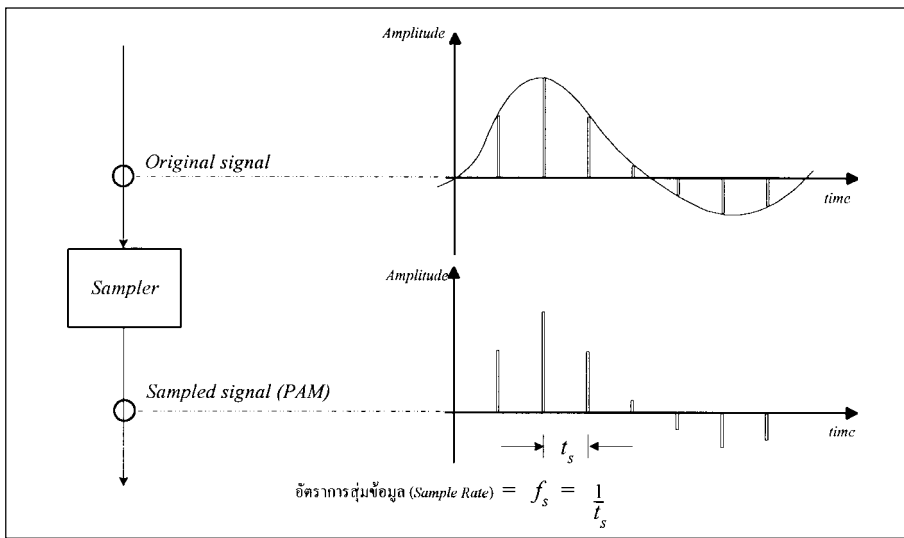
วิธีการของ Pulse Code Modulation มีขั้นตอนต่างๆ ดังรูปที่ 6 โดยขั้นตอนในขบวนการ PCM มีดังนี้คือ

1. Sampling Process
2. Quantizing Process
3. Encoding Process

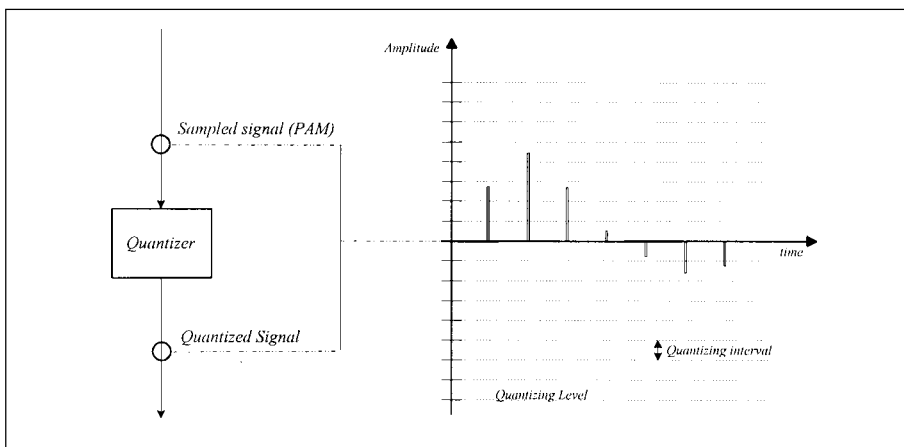
สัญญาณเสียงพูดซึ่งอยู่ในรูปของสัญญาณ Analogue จะถูกเปลี่ยนให้อยู่ในรูปของข้อมูล Digital โดยผ่านขั้นตอนทั้งสามตามลำดับ จากขั้นตอนการ Sampling เพื่อทำการสุ่มวัดค่าของสัญญาณ จะทำให้ได้สัญญาณที่เวลา ณ ขณะนั้นออกมา แล้วจึงทำการ Quantizing เพื่อดูว่าจัดแบ่งระดับของสัญญาณที่ได้ นั้น ว่ามีค่าระดับสัญญาณอยู่ในระดับใด และในขั้นตอนสุดท้ายจะทำการแทนค่าระดับของสัญญาณ ซึ่งมีการแบ่งระดับไว้แล้วนั้นด้วยค่าๆ หนึ่ง



รูปที่ 7 Pulse Code Modulation System



รูปที่ 8 : ขบวนการสุ่มตัวอย่างข้อมูล (Sampling Process)



รูปที่ 9 : ขบวนการ Quantizing Process (Quanting Process)

ด้วยวิธีการ Encoding สำหรับรายละเอียดของขั้นตอนต่างๆ มีดังนี้

### 2.1.1 Sampling

Sampling เป็นขบวนการสุ่มตัวอย่างของสัญญาณ Analogue ในอัตราสม่ำเสมอ (โดยมีช่วงเวลาในการสุ่มเท่ากับช่วงเวลาของ 1 เฟรม) เพื่อที่จะเก็บตัวอย่างขนาดสัญญาณที่เวลานั้นๆ ทำให้ได้ขบวนการของ pulse ซึ่งมีขนาด Amplitude เท่ากับหรือเป็นสัดส่วนกับสัญญาณ Analog ที่ถูก Sampling มา และการที่สัญญาณ pulse ที่ได้มีขนาด Amplitude เปลี่ยนไปตามเวลา จึงเรียกว่าเป็น Pulse Amplitude Modulation (PAM) (รูปที่ 8)

เพื่อให้สัญญาณ PAM ที่ได้

สามารถเปลี่ยนกลับเป็นสัญญาณ Analog ได้เหมือนเดิม จะต้องใช้อัตราการสุ่ม (Sample Rate) มากกว่าหรือเท่ากับ 2 เท่าของความถี่สูงสุดของสัญญาณ Analog (ทฤษฎีของ Shannon) ดังนั้นในระบบ-

โทรศัพท์ซึ่งมีช่วงความถี่เสียงขณะการสนทนาโดยเฉลี่ยระหว่าง 300 Hz ถึง 3400 Hz จึงจำเป็นต้องใช้ Sampling Rate มากกว่า 6800 Hz และตามมาตรฐานจะใช้ Sampling Rate เท่ากับ 8000 Hz ดังนั้นแต่ละสัญญาณ PAM ที่ได้จะมีช่วงเวลาห่างกัน (ซึ่งเป็นคาบเวลาของ 1 เฟรม) เท่ากับ

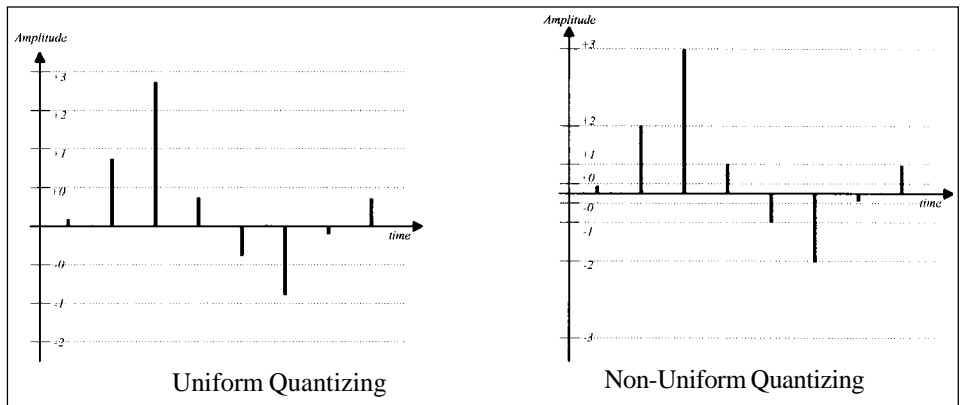
$$1/8000 \text{ Hz} = 125 \text{ Sec}$$

### 2.1.2 Quantizing

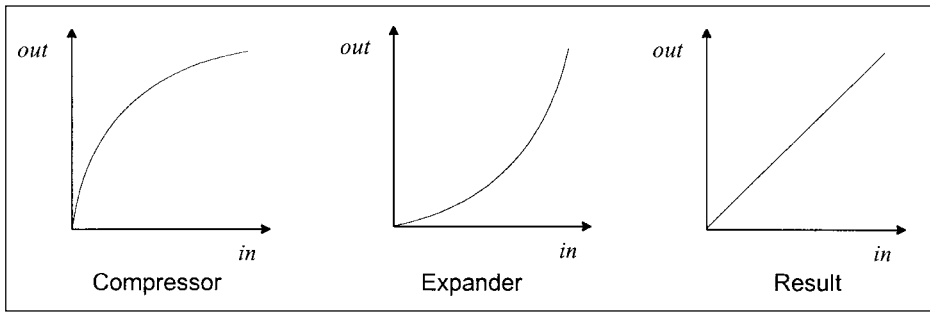
จากสัญญาณ PAM ที่ได้ในขบวนการ Sampling จะมีการกำหนดค่าให้กับสัญญาณ PAM ตามระดับ Amplitude ของสัญญาณ โดยการแบ่งระดับ Amplitude จากค่าต่ำสุดถึงค่าสูงสุด ออกเป็นค่าระดับต่างๆ เรียกว่า Quantizing Level และเรียกวิธีการนี้ว่า Quantizing (รูปที่ 9)

ในการแบ่งระดับของ Amplitude ออกเป็นระดับต่างๆ นั้น ถ้าจะให้ Quantizing Level สามารถแทนทุกค่าของระดับ Amplitude ได้ จะต้องแบ่งให้ Quantizing Level มีจำนวนเท่ากับ Infinity ซึ่งในทางปฏิบัติไม่สามารถทำเช่นนั้นได้ จึงมีการกำหนดเป็นจำนวนหนึ่ง โดยมาตรฐานของ ITU และมาตรฐานของทางอเมริกา กำหนดไว้เท่ากับ 256 Quantizing Level

การที่ Quantizing Level ไม่



รูปที่ 10 : Uniform Quantizing และ Non-uniform Quanting



รูปที่ 11 :

สามารถแทนค่าระดับของ PAM ได้ทุกค่า ทำให้มีความคลาดเคลื่อนเกิดขึ้นจากค่าความแตกต่างระหว่างระดับของสัญญาณ PAM กับระดับของ Quantizing Level ได้ และเรียกว่า Quanting Error หรือ Quantizing Noise ซึ่ง Quantizing Noise นี้ สามารถลดได้โดยการเพิ่มจำนวน Quantizing Level

โดยปกติเสียงพูดในขณะสนทนาจะมีระดับ Amplitude ต่ำ มากกว่าระดับ Amplitude สูง (เช่น ขณะตะโกน) จึงมีการกำหนด Quantizing Interval (โดยมี Quantizing Level เท่าเดิม) โดยให้ Quantizing Interval มีขนาดเล็กสำหรับช่วง Amplitude ต่ำ และมี Quantizing Interval กว้างขึ้นจำนวนน้อยสำหรับช่วง Amplitude ที่สูงขึ้น ซึ่งเป็นการทำให้ Quantizing Noise ลดลงได้ในขณะที่มีการสนทนาด้วยระดับเสียงปกติ และเรียกวิธีการกำหนด Quantizing Interval แบบนี้ว่า Non-uniform Quantizing ส่วนการกำหนด Quantizing Interval เท่าๆ กัน เราเรียกว่า Uniform Quantizing ดังรูปที่ 10

วิธีการของ Non-uniform Quantizing ทำได้ 2 วิธีด้วยกัน คือ

### 1. Companding

โดยการใช้วิธีการบีบ (Compress) ระดับของสัญญาณ PAM ก่อนทำการ Encode (ด้วย Linear Encoder) และทางฝั่งต้นรับจะใช้วิธีการขยาย (Expand) เพื่อให้ได้สัญญาณ PAM

มีระดับสัญญาณเท่ากับต้นส่ง และการใช้วิธี Compress และ Expand เช่นนี้ เราจึงเรียกว่า “Companding Technique” ดังรูปที่ 11

### 2. Non-Linear Encoder

วิธีการนี้จะใช้วิธีการ Encoding โดยไม่จำเป็นต้องมี Compander แยกต่างหาก โดยใช้วิธีการแบบ Non-Linear เพื่อให้ได้ข้อมูลออกมาในลักษณะที่ไม่เป็นเชิงเส้นกับสัญญาณ PAM

จากวิธีการ Companding จะเห็นว่า Transfer Characteristic ระหว่าง Input และ Output ของ Compressor เป็นตัวกำหนด Signal to Quantizing Noise Ratio ให้ดีขึ้นสำหรับสัญญาณที่มี Amplitude ต่ำ โดยมีความสัมพันธ์กันในรูปของ logarithmic ตามมาตรฐาน ITU จะใช้ Compression Characteristic ในแบบของ A-Law ซึ่งมีความสัมพันธ์ระหว่าง Output (Y) และ Input (X) ดังนี้

$$y = \frac{1 + \ln Ax}{1 + \ln A} ; \frac{1}{A} \leq x \leq A$$

$$y = \frac{1 + \ln Ax}{1 + \ln A} ; \frac{1}{A} \leq x \leq A$$

#### 2.1.3 Encoding

Encoding เป็นขั้นตอนในการแปลงสัญญาณ PAM ที่ได้จากการ Sampling ให้อยู่ในรูปของข้อมูล Digital ตามค่ารหัสที่กำหนดในการ Quantizing ซึ่งตามมาตรฐานทั่วไปกำหนดเป็น 256 ระดับ (Quantizing Level) จึงต้องใช้จำ-

นวน Bit ในการแทนค่าทั้ง 256 รหัส ด้วยจำนวนบิตทั้งหมดเท่ากับ 8 บิต โดยให้บิตแรก (Most Significant Bit) เป็นตัวบอกว่า Sample นั้นมีค่าระดับแรงดันเป็นบวก (+) หรือลบ (-) เช่นถ้าให้ “1” แทนแรงดันบวก และ “0” แทนแรงดันลบ ส่วนบิตที่เหลือทุกๆ บิต จะแทนค่าตาม Quantizing Level สำหรับแรงดันซึ่งมีค่ามากกว่าและต่ำกว่าศูนย์ (“00000000” หรือ “10000000”) เราเรียกวิธีการกำหนดค่ารหัสแบบนี้ว่า “Symmetrical Binary Code”

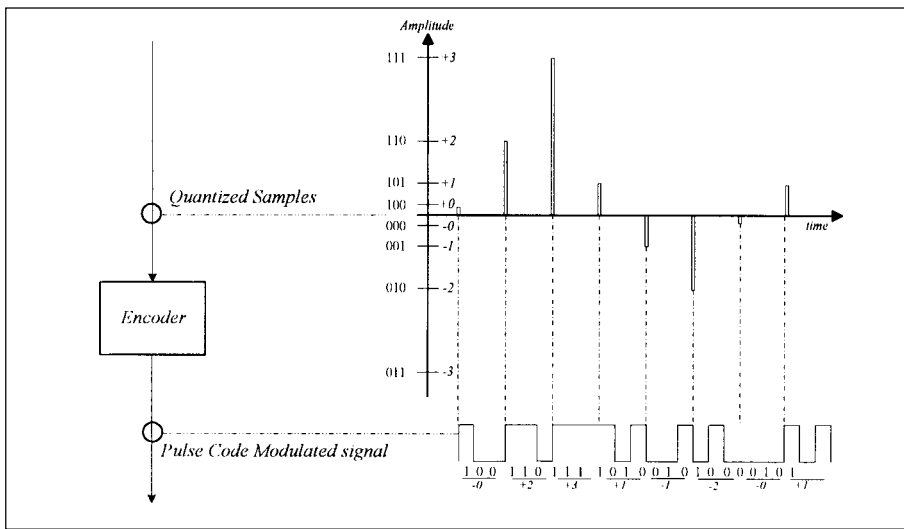
วิธีการ Encode อีกวิธีหนึ่งคือ “Ordinary Binary Code” ใช้วิธีแทนค่าสัญญาณ PAM ที่ได้ตามค่า Amplitude ของสัญญาณจากต่ำไปสูงด้วย 0000000 ถึง 1111111 ตามลำดับ โดยค่าระดับศูนย์โวลต์จะเป็น 0111111 หรือ 1000000

วิธีการ Symmetrical Binary Code จะมีข้อดีคือ ถ้าหากมี Error เกิดขึ้นกับบิตแรกจะมีผลให้การผิดพลาดทาง Amplitude น้อยกว่าวิธีการแบบ Ordinary Binary Code เพราะการผิดพลาดจาก Bit แรกซึ่งเป็น Sign Bit จะทำให้เกิดการผิดพลาดทาง Amplitude เป็นสัดส่วนกับขนาดสัญญาณ และค่าเฉลี่ยของระดับเสียงขณะสนทนาโดยทั่วไปมีระดับต่ำอยู่แล้ว การผิดพลาดทาง Amplitude จึงมีแนวโน้มต่ำด้วย (รูปที่ 12)

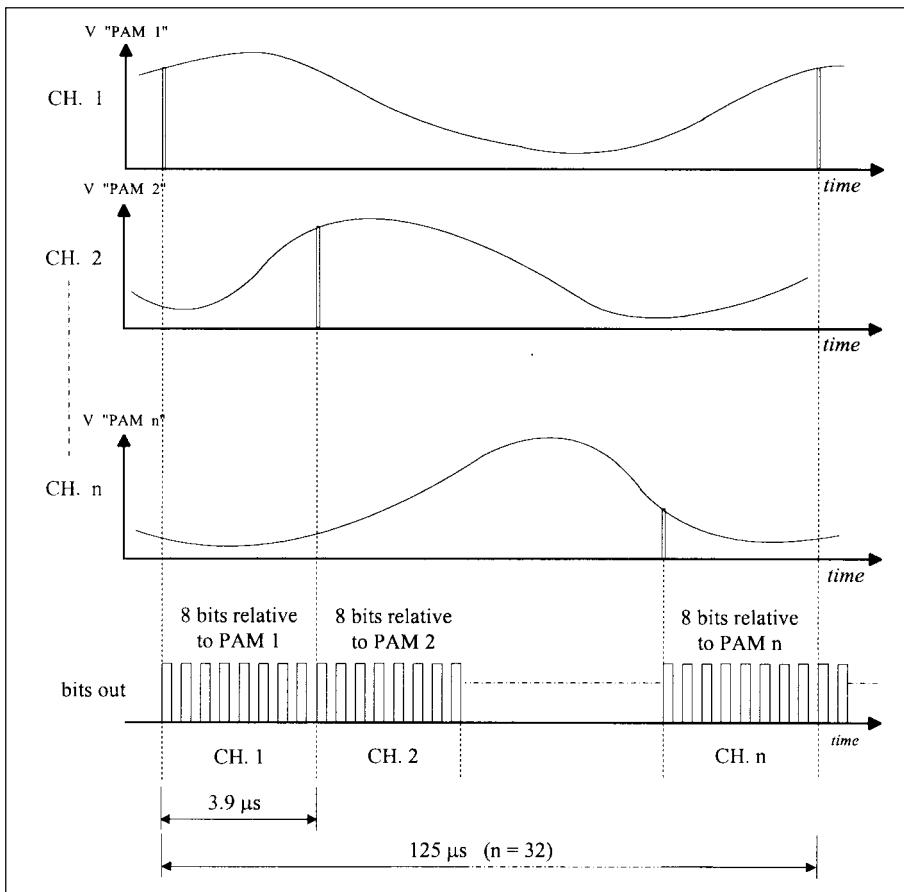
### 2.2 PCM Multiplexing and Frame Structure

จากการแปลงสัญญาณ Analogue เป็นสัญญาณ Digital ในรูปของสัญญาณ PCM เราจะทำการ Multiplex สัญญาณที่ได้ด้วยวิธีการของ Time Division Multiplexing (TDM)(รูปที่ 13)

ซึ่งตามมาตรฐานของ ITU-T กำหนดให้ทำการ Multiplex สัญญาณจำนวน 32 Channels เข้าด้วยกันโดยแบ่งออกเป็น



รูปที่ 12 : ขบวนการ Encoding (Encoding Process)



รูปที่ 13 : Time Division Multiplex (byte interleaving)

หมดเท่ากับ 8 บิต จึงได้จำนวนบิตใน 1 Frame เท่ากับ 256 บิต (8x32)

จะเห็นได้ว่าในเวลา 125 mSec มีจำนวนบิตทั้งหมดเท่ากับ 256 บิต เราจึงคำนวณหาอัตราบิต (Bits per Second; Bps) เท่ากับ

$$\frac{256 \text{ bits}}{125 \text{ mS}} = 2.048 \times 10^6 \text{ bit/sec} = 2.048 \text{ Mbit/sec}$$

ในแต่ละ Time Slot มีอัตราบิตเท่ากับ 64 Kbps

การแบ่งเฟรมออกเป็น 32 Time Slot ตาม ITU-T G.704 มีรูปแบบดังรูปที่ 14

### 2.3 Line Coding

สัญญาณ PCM ที่ได้จะประกอบด้วยกลุ่มข้อมูลของ “1” และ “0” ซึ่งจะถูกส่งผ่านไปนีสื่อสัญญาณต่างๆ และเพื่อให้ข้อมูลนี้สามารถส่งไปถึงปลายทางได้อย่างถูกต้อง จำเป็นต้องมีการเปลี่ยนรูปของสัญญาณ PCM ที่ได้ให้มีรูปแบบที่เหมาะสมก่อน จึงจะส่งไปในสื่อสัญญาณต่างๆ ได้ ซึ่งรูปแบบแต่ละรูปแบบเราเรียกว่า Line Code (หรือการเข้ารหัสทางสายส่ง) โดยมี Code Conversion ทำหน้าที่ในการแปลงสัญญาณ PCM ให้อยู่ในรูปแบบที่ต้องการ และมีหลักการในการกำหนด Line Code ดังนี้

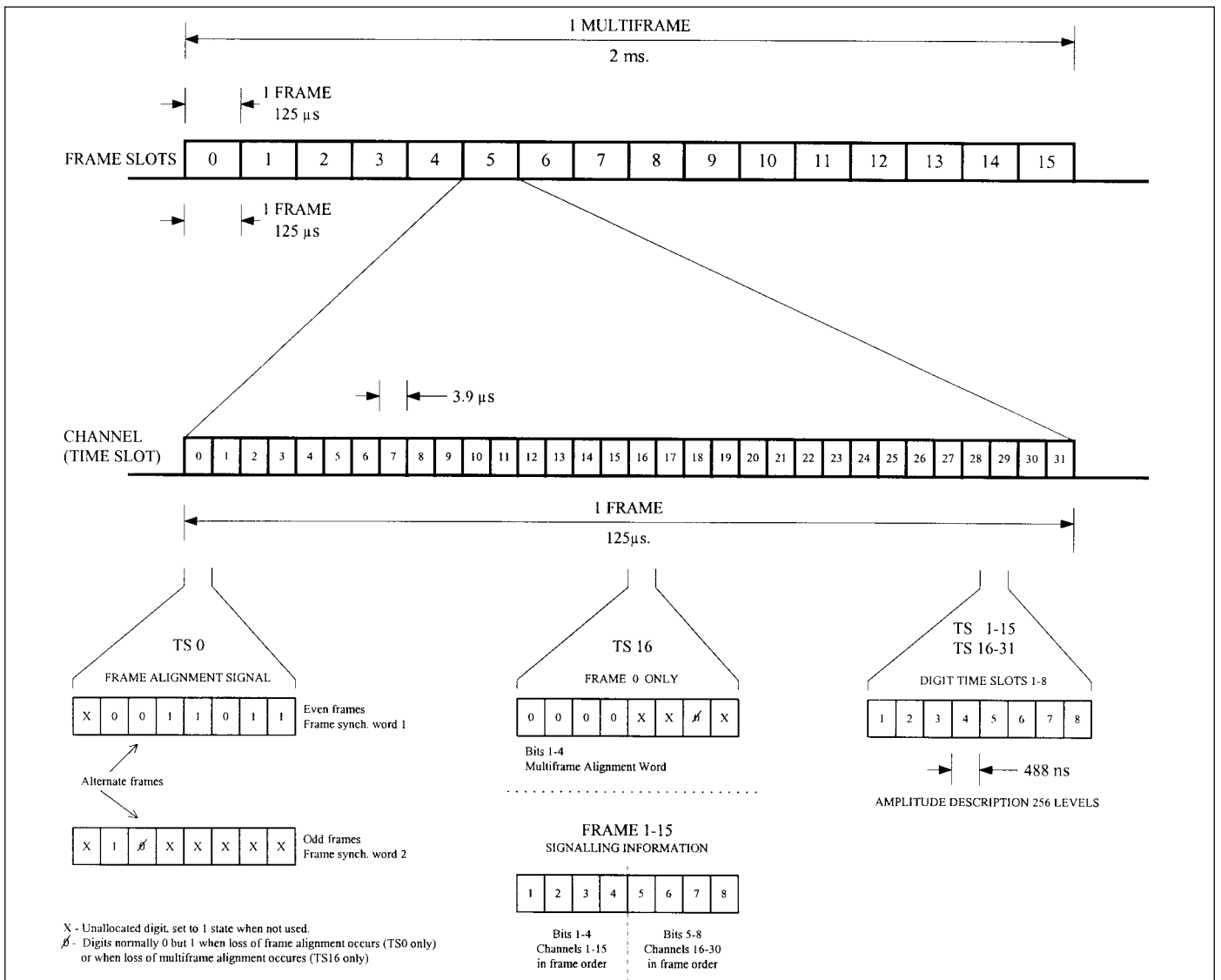
1. สัญญาณนั้นจะต้องไม่มีส่วนประกอบของสัญญาณไฟตรง (DC) ตลอดเวลาหรือเป็นช่วงเวลานาน เพราะสัญญาณไฟตรงไม่สามารถผ่านอุปกรณ์พวก Capacitor หรือ Transformer ซึ่งมีไว้อยู่ในวงจรส่วนต่างๆ ได้

2. พลังงานที่ความถี่ต่ำจะต้องมีค่าน้อย เนื่องจากจะทำให้ต้องใช้อุปกรณ์ที่มีขนาดใหญ่ ในส่วนของวงจร Equalization

3. ต้องสามารถดึงสัญญาณ

30 Channel สำหรับการส่งข้อมูล (Speech Channel)  
 1 Channel สำหรับ Multi-frame Alignment Signal, Service bits และ Signaling  
 1 Channel สำหรับ Frame Alignment Signal และ Service bits

ดังนั้นใน 1 Frame จึงประกอบด้วยจำนวน Channel ทั้งหมด 32 Channels หรือเรียกว่า 32 Time Slot โดยคาบเวลาของทั้ง 32 Time Slot เท่ากับ 125 mSec หรือกล่าวได้ว่า แต่ละ Time Slot มีคาบเวลาเท่ากับ 3.9 mSec และใน 1 Time Slot ประกอบด้วยจำนวนบิตทั้ง-



รูปที่ 14 :

Clock กลับคืนได้จากสัญญาณนั้น โดยให้สัญญาณนั้นมีการเปลี่ยนแปลงโดยผ่านระดับแรงดันศูนย์โวลต์

4. สัญญาณ Code นั้น จะต้องมียกเว้นเพียง Code เดียวที่แทนสัญญาณนั้นเพื่อไม่ให้เกิดความคลุมเครือที่จะทำให้การ Decode แปลงค่าผิดได้

5. ต้องใช้ Bandwidth ในการส่งสัญญาณนั้นน้อยที่สุด

6. จะต้องมีการตรวจจับหรือแก้ไขข้อผิดพลาดจากการส่งได้ (Error Detection and Correction)

สัญญาณ Pulse ที่ได้จากขั้นตอนการ Encode จะอยู่ในรูปของ Unipolar คือ "0" แทนการไม่มี Pulse และ "1"

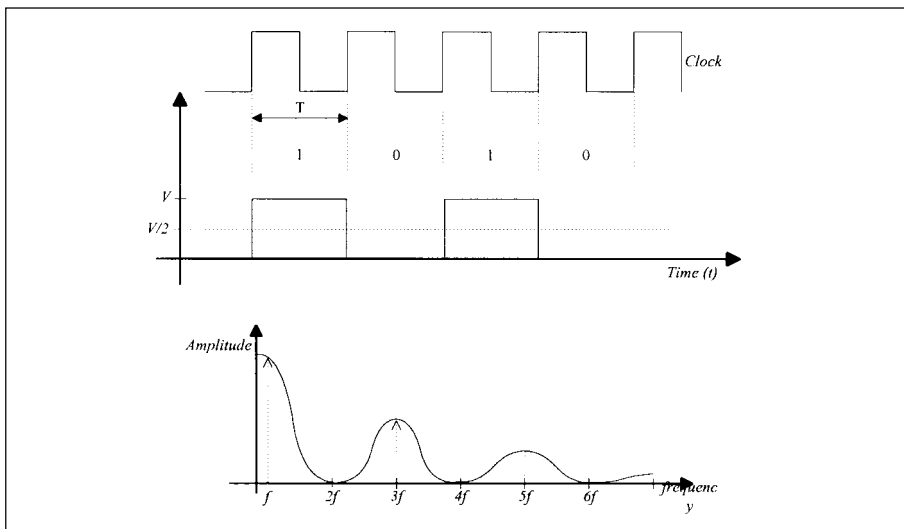
แทนการมี Pulse ซึ่งสัญญาณในรูปของ Unipolar จะมีส่วนประกอบของสัญญาณไฟตรง (DC) อยู่มาก และแปรผันตามความหนาแน่นของ Pulse ทำให้เกิดปัญหาการรบกวนระหว่างระบบจ่ายไฟเลี้ยง (DC Power Feeding) ของ Repeater ได้ และหากพิจารณาในด้านของการกระจายพลังงาน (Power Spectrum) จะเห็นว่าพลังงานส่วนใหญ่ของ Unipolar อยู่ที่ความถี่ศูนย์ (สัญญาณไฟ DC) และกระจายไปจนถึงความถี่สองเท่าของสัญญาณ ทำให้ต้องใช้ Bandwidth ในการส่งสัญญาณมากกว่าความถี่ของสัญญาณ Pulse นั้น (รูปที่ 15)

เพื่อหลีกเลี่ยงปัญหาดังกล่าว

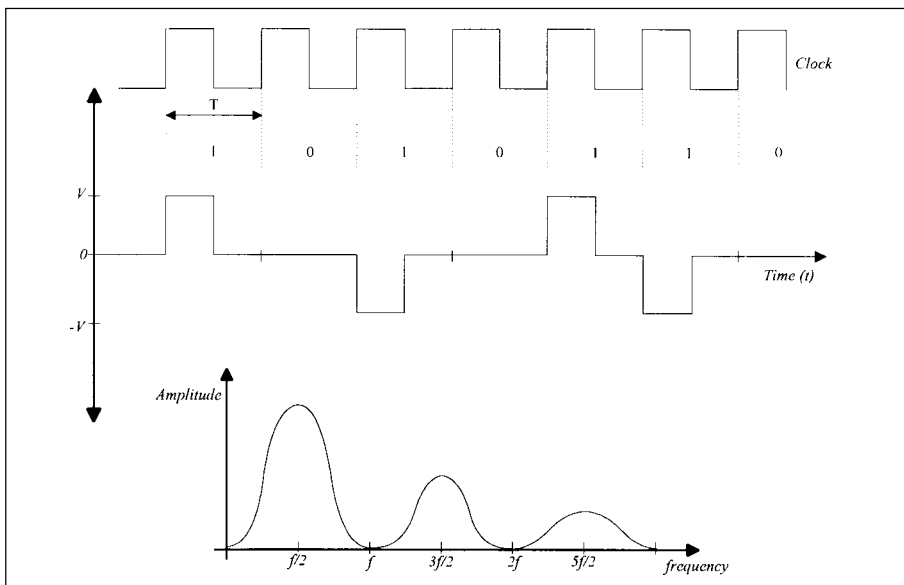
จึงมีการแปลงสัญญาณ Unipolar ให้อยู่ในรูปของ Bipolar ก่อนโดยสัญญาณแบบ Bipolar ซึ่งมีทั้ง Pulse บวกและ Pulse ลบสลับกันไป ทำให้สัญญาณไฟตรง (DC) ที่เกิดขึ้นบนสายมีค่าต่ำจนสามารถตัดทิ้งได้ ดังนั้นจึงไม่มีปัญหาการรบกวนระหว่าง DC Power feeding สำหรับ Repeater และจาก Power Spectrum จะเห็นว่าพลังงานส่วนใหญ่ของ Bipolar อยู่ที่ความถี่เท่ากับครึ่งหนึ่งของ Pulse repetition frequency ความต้องการ Bandwidth จึงมีเพียงครึ่งหนึ่งของ Pulse repetition frequency

วิธีการเข้ารหัสแบบ Unipolar

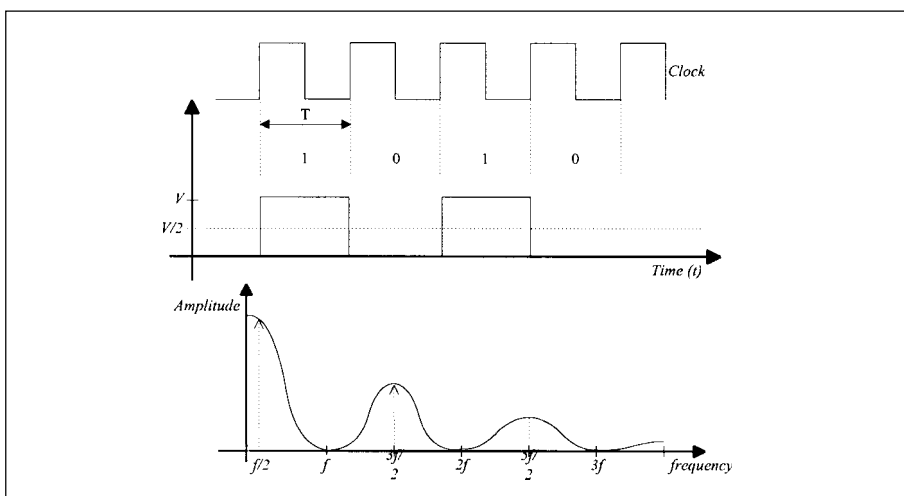
เช่น NRZ (Non-Return to Zero), RZ



รูปที่ 15 : Uni-polar Coding



รูปที่ 16 : Bi-polar Coding



รูปที่ 17 : The NRZ code and its spectrum

(Return to Zero) และ ADI ส่วนวิธีการเข้ารหัสแบบ Bipolar มีด้วยกันหลายวิธี เช่น AMI, HDB3, CMI มีรายละเอียดต่อไป

### 2.3.1 NRZ Code (100% Unipolar)

NRZ (Non-Return to Zero) เป็นการเข้ารหัสซึ่งมีความกว้างของ Pulse เท่ากับความกว้างของคาบเวลาของสัญญาณ Clock ดังรูปที่ 17

จากรูปแบบการเข้ารหัสแบบ NRZ พบว่ามีส่วนประกอบของสัญญาณไฟตรง (DC) อยู่ด้วย และจาก Frequency Spectrum พบว่าที่ความถี่เท่ากับความถี่ของสัญญาณ Clock ไม่มีค่าสัญญาณใดๆ อยู่เลย แสดงว่าเมื่อส่งสัญญาณนี้ออกไปทางฝั่งรับจะไม่สามารถดึงสัญญาณ Clock กลับมาได้เลย ดังนั้นวิธีการแบบ NRZ จึงไม่เหมาะสมที่จะใช้ในการส่งข้อมูลผ่านสื่อสัญญาณต่างๆ

### 2.3.2 RZ Code (50% Unipolar)

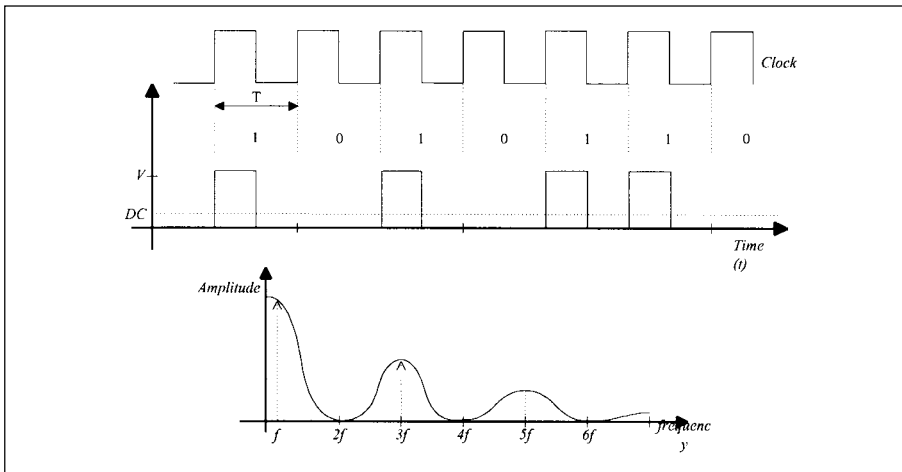
วิธีการเข้ารหัสแบบ RZ (Return to Zero) จะคล้ายกับการเข้ารหัสแบบ NRZ แต่มีความกว้างของ Pulse เพียงครึ่งหนึ่ง และยังคงมีส่วนประกอบของสัญญาณไฟตรง (DC) อยู่ แต่จาก Frequency Spectrum พบว่าที่ความถี่เท่ากับความถี่ของสัญญาณ Clock จะมีระดับของสัญญาณอยู่ค่าหนึ่ง ดังนั้นเราจึงสามารถดึงสัญญาณ Clock กลับมาได้จากการเข้ารหัสแบบ RZ แต่การเข้ารหัสแบบ RZ ไม่สามารถตรวจจับความผิดพลาดที่อาจเกิดขึ้นได้ระหว่างการส่ง จึงไม่นำมาใช้ในการส่งข้อมูลผ่านสื่อสัญญาณต่างๆ เช่นกัน (รูปที่ 18)

### 2.3.3 ADI Code (100% Unipolar)

ADI (Alternate Digit Inversion) วิธีการของ ADI จะให้มีการกลับขั้วของสัญญาณทุก 2 บิตดังรูปที่ 19 ซึ่งวิธีการของ ADI เป็นวิธีที่สามารถดึงสัญญาณ Clock กลับมาได้เช่นเดียวกัน

### 2.3.4 AMI Code (Alternate Mark





รูปที่ 18 : The RZ code and its spectrum

สัญญาณ Pulse ลบให้เป็น Pulse บวก และจะได้สัญญาณมีรูปแบบเดียวกับ RZ

วิธีการของ AMI สามารถแก้ไขข้อผิดพลาดได้ เช่น หากเกิดการรบกวนขึ้นจนทำให้ข้อมูลซึ่งแทน “0” ถูกเปลี่ยนเป็น “1” ทางด้านรับจะทราบได้ทันทีที่เกิดข้อผิดพลาดขึ้น เนื่องจากจะไม่มี Pulse ที่เหมือนกันต่อเนื่องกันทำให้ทางด้านรับสามารถแก้ไขข้อผิดพลาดได้ แต่ในกรณีที่มีข้อมูล “0” ต่อเนื่องกันยาวมากๆ ก็อาจมีปัญหาในการ Synchronize ระหว่างกลุ่มข้อมูลได้ (รูปที่ 20)

### 2.3.5 HDB3 (High Density Bipolar 3)

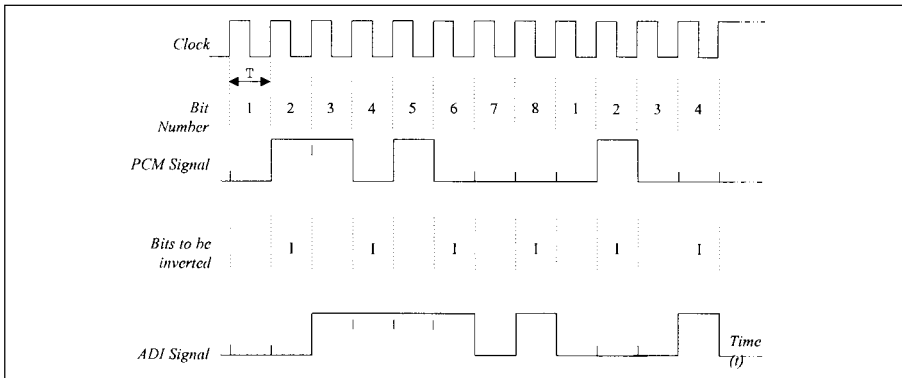
วิธีการ HDB3 เป็นวิธีที่ใช้เพื่อจำกัดจำนวนบิตที่แทนค่า “0” ที่ต่อเนื่องกันเป็นแถบยาวให้เหลือเป็นแถบต่อเนื่องกันไม่เกิน 3 บิต เพื่อให้การดึงสัญญาณ Clock กลับคืนมาทางฝั่งด้านรับเป็นไปอย่างมีประสิทธิภาพ และ HDB3 เป็นมาตรฐานตาม ITU-T G.703 ใช้งานกับระบบสื่อสารสัญญาณ 2, 8 และ 34 Mbps

วิธีการแบบ HDB3 จะไม่ทำให้เกิดส่วนประกอบของสัญญาณไฟตรง (DC) ขึ้นใน Frequency Spectrum และมีหลักการเข้ารหัสแบบ HDB3 ดังนี้ (รูปที่ 21)

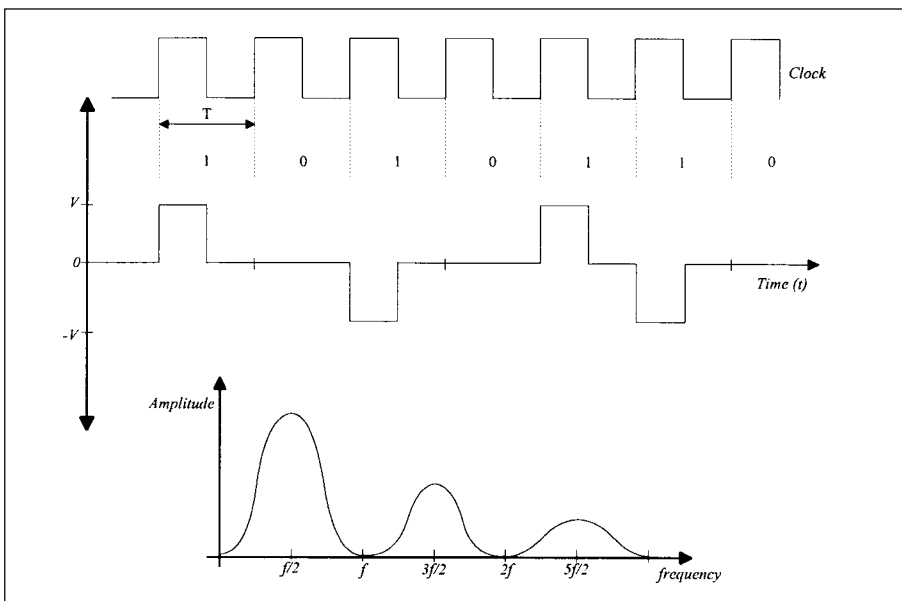
1. ถ้ามีบิต “0” ต่อเนื่องกันไม่เกิน 3 บิต จะเข้ารหัสโดยแทน “1” ด้วย Pulse บวกและ Pulse ลบสลับกันไปเช่นเดียวกับวิธีการของ AMI

2. ถ้ามีบิต “0” ต่อเนื่องกันเกิน 3 บิต จะแทนบิตที่ 4 ด้วย Pulse ของ “1” ซึ่งมีชั่วของ Pulse เหมือนกับ Pules ก่อนหน้านี้ และเรียก Pules ที่สร้างขึ้นใหม่นี้ว่า “Violation Pulse”

3. Violation Pulse ที่อยู่ติดกันจะต้องมีชั่วสลับกัน แต่ถ้าชั่วของ Violation Pulse ถูกบังคับให้มีชั่วเหมือนกับ Violation Pulse ก่อนหน้านี้ ให้กำหนดให้ “0”



รูปที่ 19 : Code conversion by ADI

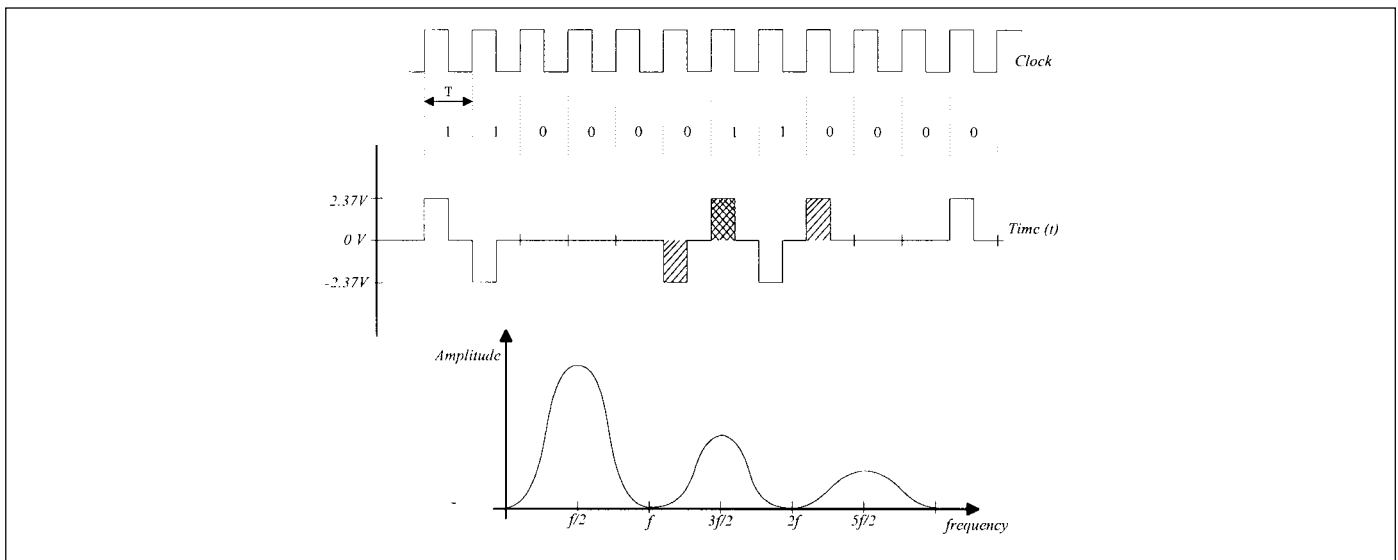


รูปที่ 20 : The AMI code and its spectrum

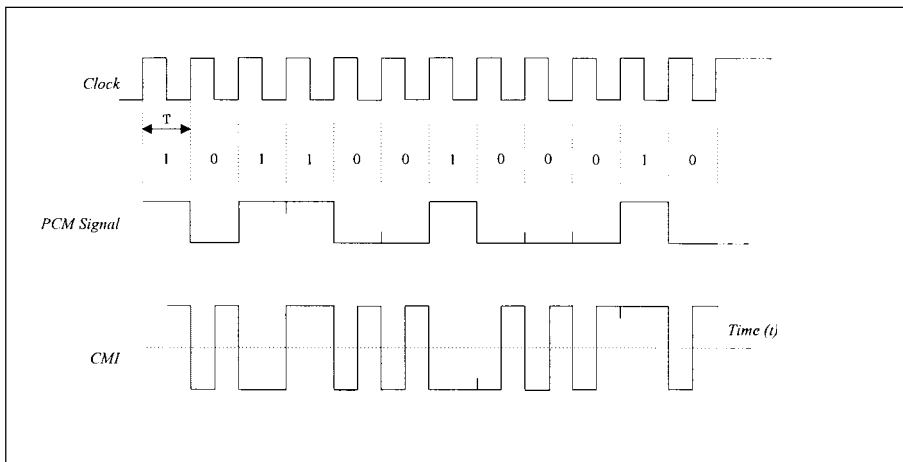
#### Inversion)

วิธีการของ AMI จะแทนภาวะ “1” ด้วย Pulse บวกและ Pulse ลบ สลับกันไปดังรูป ทำให้ไม่มีส่วนประกอบของสัญญาณไฟตรง (DC) ปรากฏขึ้นใน

Frequency Spectrum และถึงแม้ว่าไม่มี ความถี่ของสัญญาณ Clock ปรากฏ เป็น ความถี่พื้นฐานบน Frequency Spectrum ก็ตาม แต่ก็สามารถดึงสัญญาณ Clock กลับคืนมาทางฝั่งรับมาได้ โดยการ Rectify



รูปที่ 21 : The HDB3 code and its spectrum



รูปที่ 22 : CMI Code

ตารางที่ 1

ระดับที่การ Multiplex	จำนวน Channel	อัตราการส่งข้อมูล	Line Code
1	30	2.048 Mbps	HDB3
2	120	8.448 Mbps	HDB3
3	480	34.368 Mbps	HDB3
4	1920	139.264 Mbps	CMI
5	7680	565.992 Mbps	CMI

หมายเหตุ 1 Channel = 64 Kbps

ตัวแรก ในแถวของ “0” ที่ต่อเนื่องกัน เป็นเสมือน “1” เพื่อให้ Violation Pulse ที่สร้างขึ้นใหม่มีชั่วตรงข้ามกันกับ Violation Pulse ก่อนหน้านี้

### 2.3.6 CMI (Code Mark Inversion)

เป็นวิธีการตามมาตรฐานของ

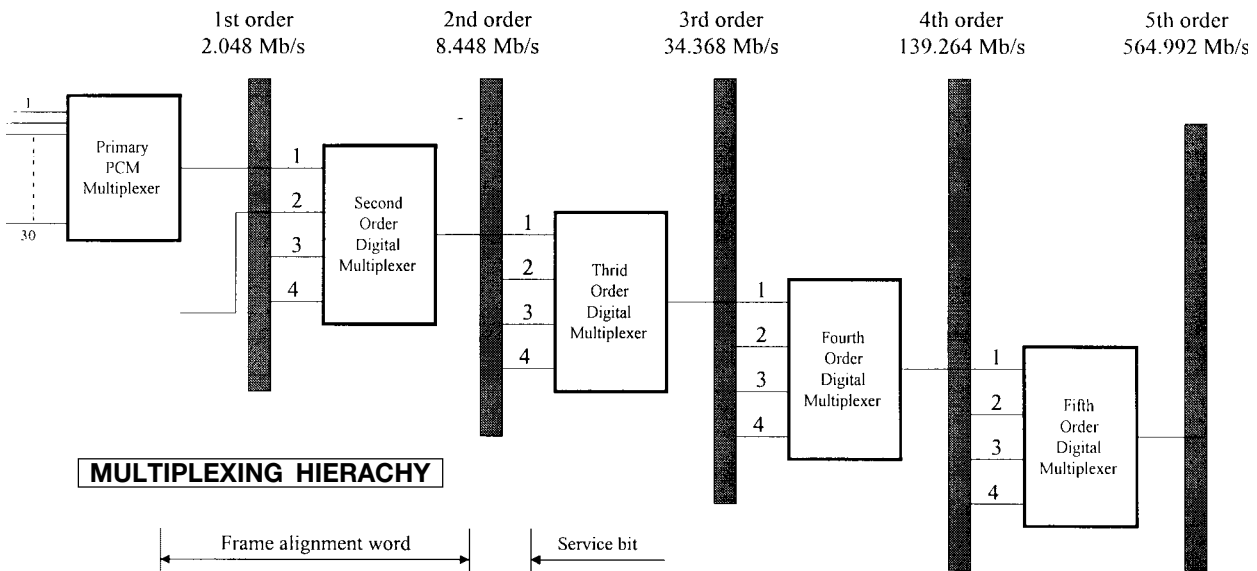
ITU-T G.703 เช่นเดียวกับ HDB3 แต่ใช้สำหรับการ Multiplex ที่ความเร็ว 139.264 Mbps โดยการแทน “1” ด้วย pulse บวกและลบสลับกันไป โดย pulse นั้นมีความกว้างของ pulse เท่ากับคาบเวลาของสัญญาณ Clock และแทน “0” ด้วย pulse ที่มีความกว้างของ pulse

เหมือนกับสัญญาณ Clock (Duration 50%) ดังรูปที่ 22

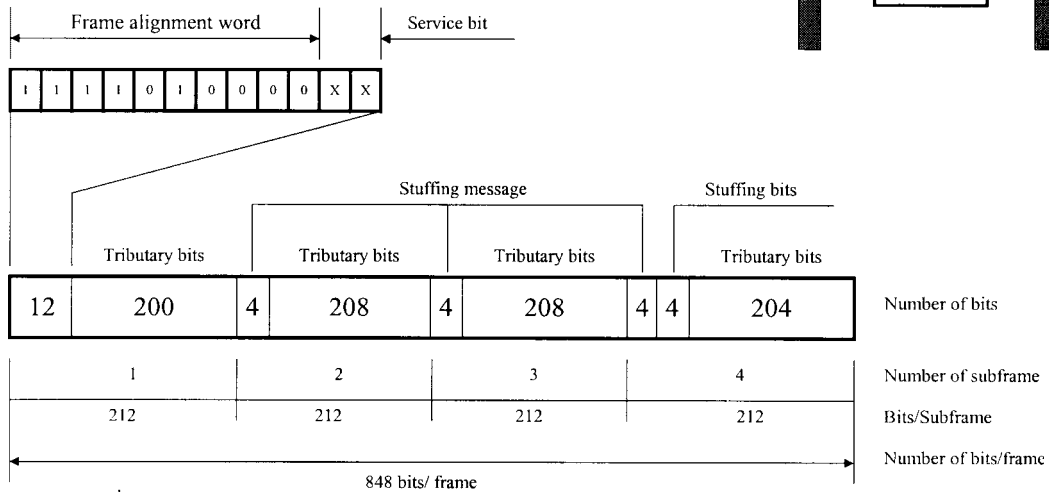
### 3. Digital Multiplexing Hierarchy

ตามมาตรฐาน ITU-T ได้กำหนดลำดับชั้นในการ Multiplexing หรือเรียกว่า Digital Multiplexing Hierarchy ไว้ตามลำดับตามจำนวน Channel ที่ต้องการ Multiplexing โดยระบบ PCM ซึ่ง Multiplex สัญญาณทั้ง 30 Channel เข้าด้วยกัน ถือเป็นระดับการ Multiplex แรกสุด (2.048 Mbps) และการ Multiplex ในระดับที่ 2 จะทำการ Multiplex สัญญาณในระดับที่หนึ่งจำนวน 4 วงจรเข้าด้วยกัน ทำให้จำนวนของสัญญาณเพิ่มขึ้นเป็น 4 เท่าหรือเท่ากับ 120 Channels และในระดับการ Multiplex ที่สูงขึ้นก็จะมีรูปแบบเดียวกันนี้ โดยดูได้จากตารางที่ 1

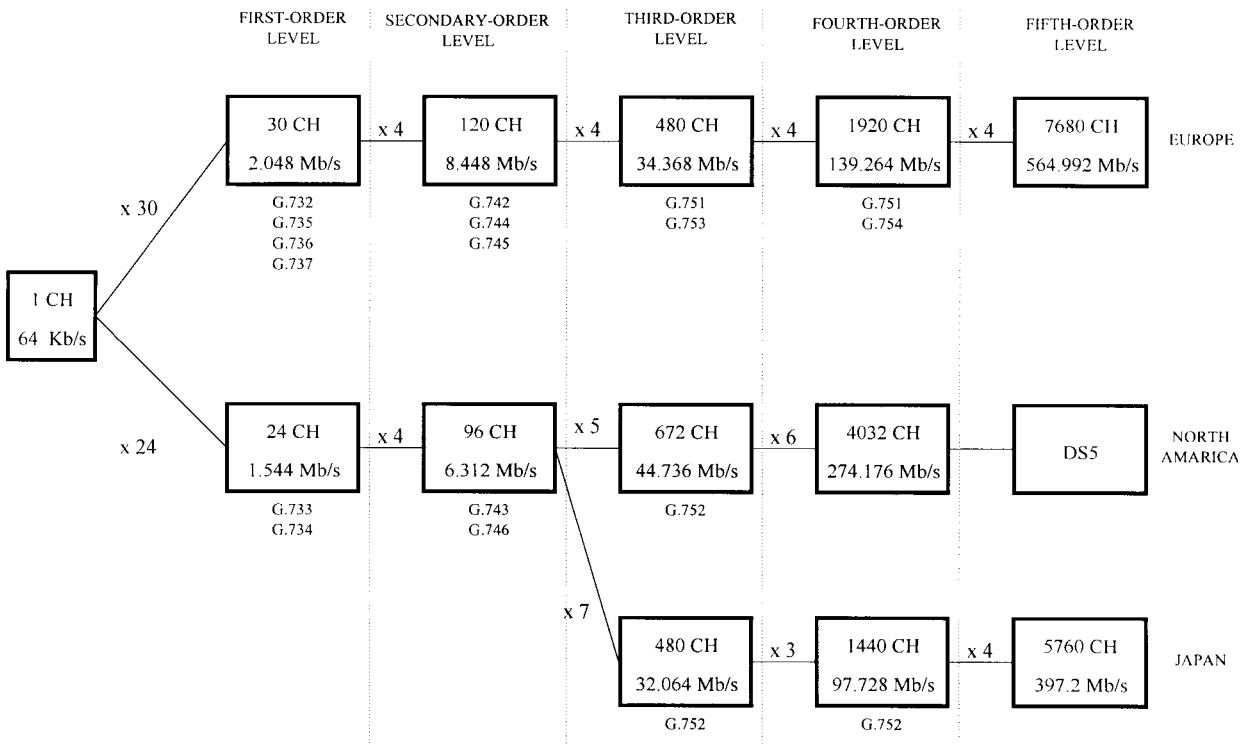
GEW



**MULTIPLEXING HIERACHY**



**The 8 Mbps frame structure (positive stuffing/justification)**



**ASYNCHRONOUS MULTIPLEXING HIERACHY**