

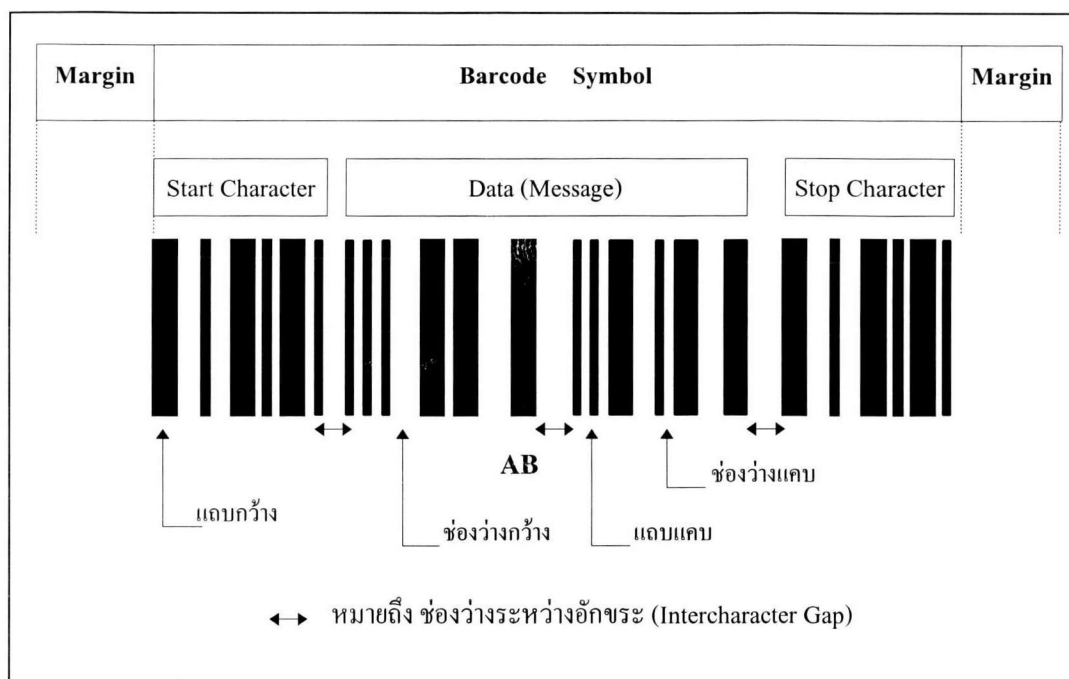
## บทที่ 4

### การพิมพ์รหัสแท่งภาษาไทย

#### รูปแบบการพิมพ์ของรหัสแท่งภาษาไทย

เมื่อกำหนดรหัสเลขฐานสองที่จะใช้แทนตัวอักษรภาษาไทย ตัวอักษรภาษาอังกฤษ ตัวเลขไทย ตัวเลขอังกฤษ และตัวอักษรพิเศษเรียบร้อยแล้ว ก็ได้กำหนดรูปแบบในการพิมพ์รหัสแท่งภาษาไทยซึ่งแสดงดังรูปที่ 4.1 ให้ประกอบไปด้วยส่วนต่างๆ ดังต่อไปนี้

1. Left Margin คือ ระยะห่างระหว่างด้านซ้ายของรหัสแท่งกับขอบกระดาษ ซึ่งจะต้องมีช่องว่างไม่ต่ำกว่า 0.5 นิ้ว (QUIET ZONE)
2. Right Margin คือ ระยะห่างระหว่างด้านขวาของรหัสแท่งกับขอบกระดาษ ซึ่งจะต้องมีช่องว่างไม่ต่ำกว่า 0.5 นิ้ว (QUIET ZONE)
3. Barcode Symbol คือ ส่วนที่เป็นแถบสีขาวและสีดำ ซึ่งมีรายละเอียด ดังนี้
  - Start Character คือ อักขระที่เป็นตัวบอกจุดเริ่มต้นของแถบรหัสแท่ง ซึ่งสำหรับรหัสแท่งภาษาไทย ใช้อักขระตัวที่ 239 (๑) เป็นตัวบอกจุดเริ่มต้นของแถบรหัสแท่งภาษาไทย
  - Data (Message) เป็นส่วนของข้อมูลที่ถูกแปลงไปเป็นแถบรหัสแท่ง โดยจะแสดงข้อมูลชุดนี้ไว้ที่ส่วนล่างของแถบ ซึ่งระยะห่างระหว่างแถบและอักขระจะต้องเว้นไว้ไม่ต่ำกว่า 0.10 นิ้ว สำหรับความกว้างระหว่างแถบกว้าง หรือความกว้างของช่องว่างต่อความกว้างของแถบแคบหรือช่องว่างแคบจะเป็นอัตราส่วน 5 : 2
  - Intercharacter Gap คือ ช่องว่างระหว่างอักขระ เมื่อสิ้นสุดอักขระแต่ละตัว
  - Stop Character คือ อักขระที่เป็นตัวกำหนดจุดสิ้นสุดของแถบรหัสแท่ง ซึ่งใช้อักขระตัวที่ 239 เป็นตัวบอกจุดสิ้นสุดของแถบรหัสแท่งภาษาไทย เช่นเดียวกันกับตัวบอกจุดเริ่มต้น



รูปที่ 4.1 แสดงลักษณะของรหัสแท่งภาษาไทย

จากนั้นนำรหัสเลขฐานสองที่ใช้ในการแทนอักขระแต่ละตัวไปกำหนดใน Array เพื่อกำหนดรูปแบบการพิมพ์แถบของรหัสแท่งภาษาไทยในโปรแกรม โดยกำหนดเป็นค่าคงที่ array BarText[32..255] of string ซึ่งจะใช้ค่าของตัวอักขระตามรหัสแอสกีในการอ้างอิงถึงข้อมูลใน Array ดังนี้

CONST

```
BarText : array[32..255] of string =
('00010010111', '00011000111', '00011010011', '00011010110',
'00110000111', '00110010011', '00110010110', '00111000011',
'00111000110', '00111010010', '10010000111', '10010010011',
'10010010110', '10011000011', '10011000110', '10011010010',
'01000010111', '01001000111', '01001010011', '01001010110',
'01100000111', '01100010011', '01100010110', '01101000011',
'01101000110', '01101010010', '10110000011', '10110000110',
'10110010010', '10111000010', '01000011101', '01001001101',
'010010111001', '00000110111', '00001100111', '00001110011',
```

'00001110110', '00100100111', '00100110011', '00100110110',  
 '00101100011', '00101100110', '00101110010', '10000100111',  
 '10000110011', '10000110110', '10001100011', '10001100110',  
 '10001110010', '10100100011', '10100100110', '10100110010',  
 '10101100010', '00010011101', '00011001101', '00011011001',  
 '00011011100', '00110001101', '00110011001', '01001011100',  
 '01100001101', '01100011001', '01100011100', '01101001001', '#0',  
 '00110011100', '00111001001', '00111001100', '00111011000',  
 '10010001101', '10010011001', '10010011100', '10011001001',  
 '10011001100', '10011011000', '10110001001', '10110001100',  
 '10110011000', '10111001000', '01000110101', '01001100101',  
 '01001110001', '01001110100', '01100100101', '01100110001',  
 '01100110100', '01101100001', '01101100100', '01101110000',  
 '11000100101', '11000110001', '01101001100', '01101011000',  
 '11000001101', '11000011001', '#0', '#0', '#0', '#0', '#0', '#0',  
 '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0',  
 '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0', '#0',  
 '#0', '#0', '#0', '#0', {160 - Thai character start at ASCII code 161}  
 '00000011111', '00001001111', '00001011011', '00001011110',  
 '00100001111', '00100011011', '00100011110', '00101001011',  
 '00101001110', '00101011010', '10000001111', '10000011011',  
 '10000011110', '10001001011', '10001001110', '10001011010',  
 '10100001011', '10100001110', '10100011010', '10101001010',  
 '00000111101', '00001101101', '00001111001', '00001111100',  
 '00100101101', '00100111001', '00100111100', '00101101001',  
 '00101101100', '00101111000', '10000101101', '10000111001',  
 '10000111100', '10001101001', '10001101100', '10001111000',  
 '10100101001', '10100101100', '10100111000', '10101101000',  
 '00010110101', '00011100101', '00011110001', '00011110100',  
 '00110100101', '00110110001', '00110110100', '00111100001',

```
'00111100100', '00111110000', '10010100101', '10010110001',
'10010110100', '10011100001', '10011100100', '10011110000',
'10110100001', '10110100100', '#0', '#0', '#0', '#0',
'10110110000', '10111100000', '01010010101', '01011000101',
'01011010001', '01011010100', '01110000101', '01110010001',
'01110010100', '01111000001', '01111000100', '01111010000',
'11010000101', '11010010001', '11010010100', '11011000001', {238}
'11011000100', '11000000111', '11000010011', '11000010110',
'11001000011', '11001000110', '11001010010', '11100000011',
'11100000110', '11100010010', '11101000010', '#0', '#0', '#0',
'#0', '#0', '#0' );      {BarText[239] is start/stop bits. }
```

ตัวอย่างเช่น ตัวอักษร A มีค่าของตัวอักษรตามรหัส ASCII เป็น 65 ดังนั้นโปรแกรมจะสามารถค้นหารหัสแท่งของตัวอักษร A ได้ โดยการใช้ BarText[65] ซึ่งคือ 00000110111 อย่างนี้เป็นต้น

เมื่อทราบข้อมูลที่ต้องการพิมพ์รหัสแท่งแล้ว ก็จะนำข้อมูลแต่ละอักขระนั้นไปทำการแปลงให้เป็นข้อมูลของรหัสเลขฐานสองเสียก่อนตามตารางที่กำหนดใน Array BarText โดยจะใส่รหัสเริ่มต้นของรหัสแท่งภาษาไทยก่อนเป็นตัวแรก จากนั้นตามด้วยรหัสเลขฐานสองของอักขระแต่ละตัวที่ต้องการพิมพ์รหัสแท่ง และสิ้นสุดการพิมพ์ข้อมูลแต่ละครั้งด้วยรหัสสิ้นสุดของรหัสแท่งภาษาไทย สำหรับรหัสเริ่มต้นและรหัสสิ้นสุดของรหัสแท่งภาษาไทย คือ BarText[239] ซึ่งมีรหัสเลขฐานสอง เป็น 11011000100 และเนื่องจากรหัสแท่งแต่ละชุดจะเริ่มต้นหรือสิ้นสุดด้วยแถบดำ ดังนั้นจึงต้องมีการพิมพ์แถบขาวหรือช่องว่างระหว่างรหัสแท่งแต่ละชุด ซึ่งกำหนดได้โดยการเพิ่มบิต 0 ลงไปเป็นตัวสุดท้ายในรหัสแท่งแต่ละชุด

### ตัวอย่างการพิมพ์รหัสแท่งภาษาไทย

ถ้าต้องการพิมพ์ข้อความ 1ศ2345 ในรูปแบบของรหัสแท่งภาษาไทย สามารถทำได้ดังนี้

1. ใส่รหัสเริ่มต้นการพิมพ์รหัสแท่งก่อน ซึ่งได้แก่ 11011000100
2. นำอักขระแต่ละตัวไปหารหัสเลขฐานสองใน Array BarText โดยใช้ค่าของตัวอักษรตามรหัส ASCII เป็นตัวอ้างอิงในการค้นหาข้อมูล ดังขั้นตอนต่อไปนี้

ค่าของตัวเลข 1 ในรหัส ASCII คือ 49 ดังนั้น จะสามารถหารหัสเลขฐานสองของตัวเลข 1 ในตารางได้ โดยการกำหนดตัวอ้างอิงในการค้นหาเป็น BarText[49] ซึ่งก็คือ 01001000111

ค่าของตัวอักษร ศ ในรหัส ASCII คือ 200 ดังนั้น จะสามารถหารหัสเลขฐานสองของตัวอักษร ศ ในตารางได้ โดยการกำหนดตัวอ้างอิงในการค้นหาเป็น BarText[200] ซึ่งก็คือ 10101101000

ค่าของตัวเลข 2 ในรหัส ASCII คือ 50 ดังนั้น จะสามารถหารหัสเลขฐานสองของตัวเลข 2 ในตารางได้ โดยการกำหนดตัวอ้างอิงในการค้นหาเป็น BarText[50] ซึ่งก็คือ 01001010011

ค่าของตัวเลข 3 ในรหัส ASCII คือ 51 ดังนั้น จะสามารถหารหัสเลขฐานสองของตัวเลข 3 ในตารางได้ โดยการกำหนดตัวอ้างอิงในการค้นหาเป็น BarText[51] ซึ่งก็คือ 01001010110

ค่าของตัวเลข 4 ในรหัส ASCII คือ 52 ดังนั้น จะสามารถหารหัสเลขฐานสองของตัวเลข 4 ในตารางได้ โดยการกำหนดตัวอ้างอิงในการค้นหาเป็น BarText[52] ซึ่งก็คือ 01100000111

ค่าของตัวเลข 5 ในรหัส ASCII คือ 53 ดังนั้น จะสามารถหารหัสเลขฐานสองของตัวเลข 5 ในตารางได้ โดยการกำหนดตัวอ้างอิงในการค้นหาเป็น BarText[53] ซึ่งก็คือ 01100010011

3. ใส่รหัสสิ้นสุดการพิมพ์รหัสแท่ง ซึ่งได้แก่ 11011000100 เป็นการสิ้นสุดการแปลงข้อมูลที่ต้องการพิมพ์รหัสแท่งเป็นเลขฐานสอง

ดังนั้น เมื่อต้องการพิมพ์ข้อความ 1ศ2345 โปรแกรมจะทำการแปลงข้อความนี้เป็นรหัสเลขฐานสองและเพิ่มบิต 1 ลงไปเมื่อสิ้นสุดรหัสเลขฐานสองของอักขระแต่ละตัว ดังนี้  
11011000100 1 01001000111 1 10101101000 1 01001010011 1 01001010110 1  
01100000111 1 01100010011 1 11011000100

4. ในการพิมพ์รหัสแท่งมีข้อกำหนดว่าจะต้องพิมพ์แถบดำก่อน และจะต้องพิมพ์สลับกับแถบขาวหรือช่องว่าง แล้วสิ้นสุดการพิมพ์รหัสแท่งของอักขระแต่ละตัวด้วยแถบดำเสมอ ดังนั้น รหัสเลขฐานสองตัวแรกหรือที่ตำแหน่งคือของอักขระแต่ละตัวจะหมายถึงแถบดำ ซึ่งถ้าเป็นบิต 1 จะหมายถึงแถบดำกว้าง แต่ถ้าเป็นบิต 0 จะหมายถึงแถบดำแคบ และรหัสเลขฐานสองตัวที่สองหรือที่ตำแหน่งคู่จะหมายถึงแถบขาวหรือช่องว่าง ซึ่งถ้าเป็นบิต 1 จะหมายถึงแถบขาวกว้าง แต่ถ้าเป็น

บิต 0 จะเป็นแถบขาวแคบ สลับกันไประหว่างแถบดำและแถบขาวในการแทนรหัสเลขฐานสองของอักขระแต่ละตัว

เมื่อต้องการส่งพิมพ์รหัสแท่งจะต้องมีการแปลงรหัสเลขฐานสองนี้ให้อยู่ในรูปของบิต 1 แทนการพิมพ์แถบดำ และบิต 0 แทนการพิมพ์แถบขาวก่อน แต่เนื่องจากแต่ละบิตถือเป็นหนึ่งโมดูล (Module) ดังนั้นการพิมพ์รหัสแท่งของแถบแคบจะมีการพิมพ์แถบเพียง 2 โมดูล แต่สำหรับแถบกว้างจะมีการพิมพ์แถบเป็น 5 โมดูล เนื่องจากการอัตราส่วนในการพิมพ์แถบกว้างต่อแถบแคบเป็น 5 : 2 ดังนั้นเมื่อได้รหัสเลขฐานสองที่ต้องการจะพิมพ์ตามรูปแบบของรหัสแท่งภาษาไทยแล้ว จะต้องพิจารณาว่าตำแหน่งของรหัสเลขฐานสองนั้นเป็นตำแหน่งของแถบดำหรือแถบขาวและเป็นแถบกว้างหรือแถบแคบ ซึ่งถ้าเป็นแถบดำกว้างจะต้องแปลงให้เป็น 11111 แต่ถ้าเป็นแถบดำแคบจะต้องแปลงให้เป็น 11 หรือถ้าเป็นแถบขาวกว้างจะต้องแปลงให้เป็น 00000 แต่ถ้าเป็นแถบขาวแคบจะต้องแปลงให้เป็น 00 ดังตัวอย่างต่อไปนี้

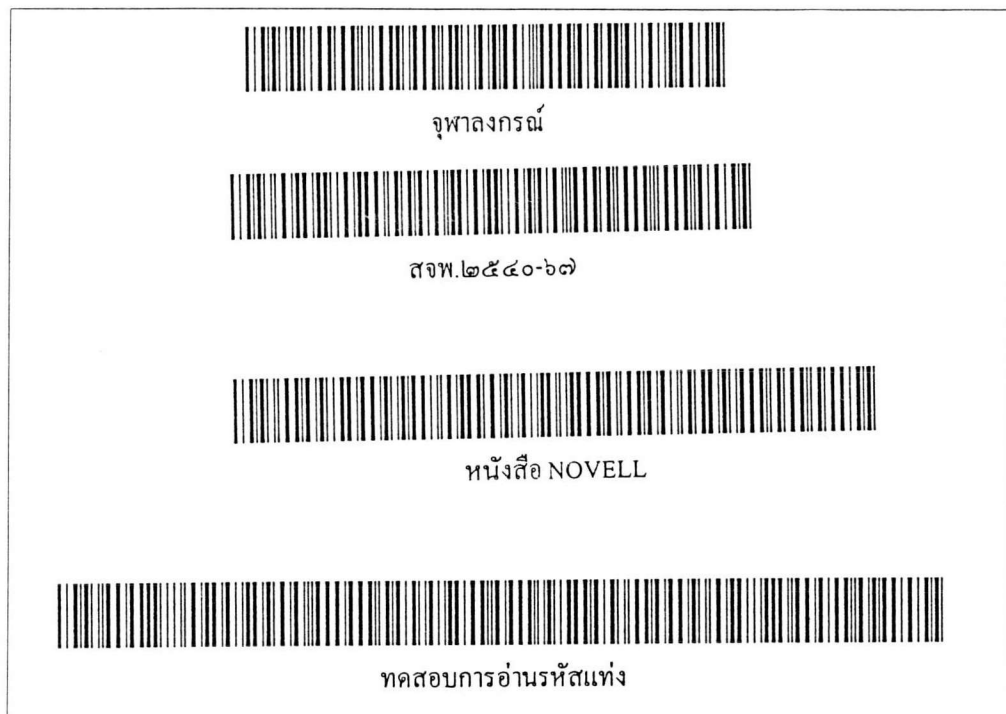
เมื่อได้รหัสเลขฐานสองชุดแรก คือ 11011000100 1 จะสามารถแปลงให้อยู่ในรูปแบบของรหัสเลขฐานสองเพื่อการพิมพ์เป็นรหัสแท่งภาษาไทยดังนี้

1 ตำแหน่งแรก	หมายถึง แถบดำกว้าง	จะถูกแปลงเป็น 11111
1 ตำแหน่งที่สอง	หมายถึง แถบขาวกว้าง	จะถูกแปลงเป็น 00000
0 ตำแหน่งที่สาม	หมายถึง แถบดำแคบ	จะถูกแปลงเป็น 11
1 ตำแหน่งที่สี่	หมายถึง แถบขาวกว้าง	จะถูกแปลงเป็น 00000
1 ตำแหน่งที่ห้า	หมายถึง แถบดำกว้าง	จะถูกแปลงเป็น 11111
0 ตำแหน่งที่หก	หมายถึง แถบขาวแคบ	จะถูกแปลงเป็น 00
0 ตำแหน่งที่เจ็ด	หมายถึง แถบดำแคบ	จะถูกแปลงเป็น 11
0 ตำแหน่งที่แปด	หมายถึง แถบขาวแคบ	จะถูกแปลงเป็น 00
1 ตำแหน่งที่เก้า	หมายถึง แถบดำกว้าง	จะถูกแปลงเป็น 11111
0 ตำแหน่งที่สิบ	หมายถึง แถบขาวแคบ	จะถูกแปลงเป็น 00
0 ตำแหน่งที่สิบเอ็ด	หมายถึง แถบดำแคบ	จะถูกแปลงเป็น 11
1 ตำแหน่งที่สิบสอง	เป็นช่องว่างระหว่างอักขระ	จะถูกแปลงเป็น 11111

ในการพิมพ์รหัสแท่งโปรแกรมจะทำการพิมพ์แต่ละแถบของแถบดำหรือแถบขาวตามจำนวนของรหัสเลขฐานสองนี้และจะได้ขนาดของแถบตามที่ออกแบบไว้

ดังนั้นจะสามารถสรุปรูปแบบการพิมพ์รหัสแท่งได้ดังนี้

รหัสเริ่มต้น + ช่องว่าง + รหัสอักษร 1 + ช่องว่าง + รหัสอักษร n + ช่องว่าง + รหัสสิ้นสุด  
 เช่น 11011000100 + 1 + 00000011111 + 1 + 11000010011 + 1 + 11011000100



รูปที่ 4.2 ตัวอย่างขนาดและลักษณะของรหัสแท่งภาษาไทย