10301112 เทคโนโลยีสารสนเทศและการสื่อสาร

บทที่ 3: ระบบเลขฐานสองและการแทนข้อมูล

ผู้ช่วยศาสตราจารย์ ดร. ปวีณ เขื่อนแก้ว

สาขาวิชาวิทยาการคอมพิวเตอร์

คณะวิทยาศาสตร์ มหาวิทยาลัยแม่โจ้

หัวข้อ

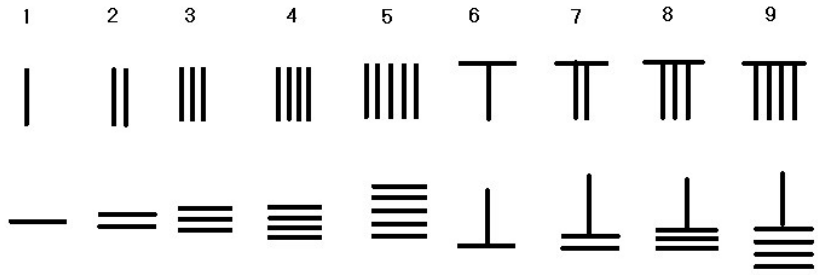
- ระบบเลขฐาน
- เลขฐานสอง
- การคำนวณด้วยเลขฐานสอง
- การแทนข้อมูลในระบบคอมพิวเตอร์ด้วยเลขฐานสอง

ระบบตัวเลข

Numeral systems

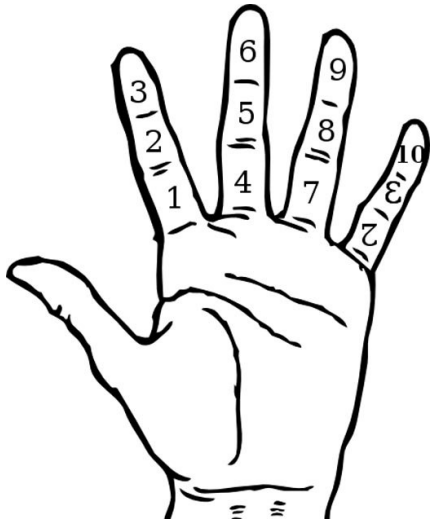
Arabic	1	3	4	5	6	10	50	90	100	500	1000	1450
Roman	I	III	IV	V	VI	X	L	XC	C	D	M	MCDL
Arabic	0	1	4	5	6	10	15	19	20	21	26	421
Mayan												
Arabic	1	2	3	4	5	10	11	20	30	40	50	59
Babylonian												

การนับเลข



เลขฐาน10

การนับเลข



×	0	1	2	3	4	5	6	7	8	9	↊	↋	10
0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	2	3	4	5	6	7	8	9	↊	↋	10
2	0	2	4	6	8	↊	10	12	14	16	18	1↊	20
3	0	3	6	9	10	13	16	19	20	23	26	29	30
4	0	4	8	10	14	18	20	24	28	30	34	38	40
5	0	5	↊	13	18	21	26	2↋	34	39	42	47	50
6	0	6	10	16	20	26	30	36	40	46	50	56	60
7	0	7	12	19	24	2↋	36	41	48	53	5↊	65	70
8	0	8	14	20	28	34	40	48	54	60	68	74	80
9	0	9	16	23	30	39	46	53	60	69	76	83	90
↊	0	↊	18	26	34	42	50	5↊	68	76	84	92	↊0
↋	0	↋	1↊	29	38	47	56	65	74	83	92	↊1	↋0
10	0	10	20	30	40	50	60	70	80	90	↊0	↋0	100

เลขฐาน12

ระบบจำนวนตัวเลข

- ระบบเลขฐานสิบ (Decimal Number System)
- ระบบเลขฐานสอง (Binary Number System)
- ระบบเลขฐานแปด (Octal Number System)
- ระบบเลขฐานสิบหก (Hexadecimal Number System)

ระบบเลขฐานสิบ (The decimal numbering system)

- คำนิยาม

  คือ ระบบเลขที่มีสัญลักษณ์แทนตัวเลขโดด 10 ตัวคือ 0, 1, 2, 3, 4, 5,6, 7, 8, 9 และถ้า N_{10} คือเลขฐานสิบใด ๆ แล้วจะได้ว่า

$$N_{10} = d_P \times 10^P + \ldots + d_2 \times 10^2 + d_1 \times 10^1 + d_0 \times 10^0 + d_{-1} \times 10^{-1} + \ldots + d_{-q} \times 10^{-q}$$

$$125.25 = 1x10^2 + 2x10^1 + 5x10^0 + 2x10^{-1} + 5x10^{-2}$$

ระบบเลขฐานสอง (The binary numbering system)

- คำนิยาม

  คือ ระบบเลขที่มีสัญลักษณ์แทนตัวเลขโดด 2 ตัวคือ 0 และ 1 และถ้า N_2 คือ เลขฐานสองใด ๆ แล้วจะได้ว่า

$$N_2 = d_p \times 2^p + \ldots + d_2 \times 2^2 + d_1 \times 2^1 + d_0 \times 2^0 + d_{-1} \times 2^{-1} + d_{-2} \times 2^{-2} + \ldots + d_{-q} \times 2^{-q}$$

$$1101.01_2 = 1x2^3 + 1x2^2 + 0x2^1 + 1x2^0 + 0x2^{-1} + 1x2^{-2}$$

ระบบเลขฐานแปด (The octal numbering system)

- คำนิยาม

ระบบเลขฐานแปดคือระบบเลขที่มีสัญลักษณ์แทนตัวเลขโดด 8 ตัวคือ 0, 1, 2, 3, 4, 5,6, 7 และถ้า N_8 คือเลขฐานแปดใด ๆ แล้วจะได้ว่า

$$N_8 = d_p \times 8^p + \ldots + d_2 \times 8^2 + d_1 \times 8^1 + d_0 \times 8^0 + d_{-1} \times 8^{-1} + \ldots + d_{-q} \times 8^{-q}$$

$$125.27_8 = 1x8^2 + 2x8^1 + 5x8^0 + 2x8^{-1} + 7x8^{-2}$$

ระบบเลขฐานสิบหก (The hexadecimal numbering system)

- คำนิยาม

  ระบบเลขฐานสิบหกคือระบบเลขที่มีสัญลักษณ์แทนตัวเลขโดด 16 ตัวคือ 0, 1, 2, 3, 4,5, 6, 7, 8, 9, A, B, C, D, E, F และถ้า N_{16} คือเลขฐานสิบหกใด ๆ แล้วจะได้ว่า

$$N_{16} = d_p \times 16^p + \ldots + d_2 \times 16^2 + d_1 \times 16^1 + d_0 \times 16^0 + d_{-1} \times 16^{-1} + \ldots + d_{-q} \times 16^{-q}$$

$$1B5.2F = 1x16^2 + Bx16^1 + 5x16^0 + 2x16^{-1} + Fx16^{-2}$$

เปรียบเทียบเลขฐานสอง ฐานสิบ ฐานแปด และสิบหก

จำนวน	ฐานสอง	ฐานสิบ	ฐานแปด	ฐานสิบหก
ศูนย์	0	0	0	0
หนึ่ง	1	1	1	1
สอง	10	2	2	2
สาม	11	3	3	3
สี่	100	4	4	4
ห้า	101	5	5	5
หก	110	6	6	6
เจ็ด	111	7	7	7
แปด	1000	8	10	8
เก้า	1001	9	11	9
สิบ	1010	10	12	A
สิบเอ็ด	1011	11	13	B
สิบสอง	1100	12	14	C
สิบสาม	1101	13	15	D
สิบสี่	1110	14	16	E
สิบห้า	1111	15	17	F

การเปลี่ยนเลขจากระบบเลขฐานสิบให้เป็นฐานสอง

- กรณีของเลขจำนวนเต็ม
  - ถ้า N เป็นเลขจำนวนเต็มฐาน 10 ใดๆ ให้นำ 2 ไปหารสั้น N ได้ผลหารเป็น N_0 และได้เศษเป็น d_0 (ค่าที่เป็นไปได้คือ 0 หรือ 1)
  - นำ 2 มาหาร N_0 ต่อไปอีก ได้ผลหารเป็น N_1 และเศษเป็น d_1
  - นำ N_1 มาหารต่อไปอีกได้ N_2 และเศษเป็น d_2
  - ดำเนินการเช่นนี้เรื่อยไป ผลที่ได้คือ $N > N_0 > N_1 > N_2 > \ldots\ldots$

    กระทั่ง 2 มีค่ามากกว่า N_p ได้ p ครั้ง
  - นั่นคือเลขฐานสองที่มีค่าเท่ากับ N ในเลขฐานสิบ คือ $d_p \ldots . d_2 d_1 d_0$

การแปลงเลขจากฐาน 10 เป็นฐาน 2

ฐาน 10		ฐาน		ได้		เศษ
1 2 3	÷	2	=	61		1
61	÷	2	=	30		1
30	÷	2	=	15		0
15	÷	2	=	7		1
7	÷	2	=	3		1
3	÷	2	=	1	→	1

$$123_{10} = 1111011_2$$

การแปลงเลขจากฐาน 2 เป็นฐาน 10

$$N = \sum d \times base^{p}$$

base=2

$$N = 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

$$N = 1 \times 64 + 1 \times 32 + 1 \times 16 + 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1$$

$$N = 64 + 32 + 16 + 8 + 0 + 2 + 1$$

$$N = 123$$

$$1111011_2 = 123_{10}$$

การแปลงเลขจากฐาน 10 เป็นฐาน 8

ฐาน 10		ฐาน		ได้		เศษ
1 2 3	÷	8	=	15		3
15	÷	8	=	1	→	7

$$123_{10} = 173_{8}$$

การแปลงเลขจากฐาน 8 เป็นฐาน 10

$$N = \sum d \times base^{p}$$

base=8

$$N = 1 \times 8^2 + 7 \times 8^1 + 3 \times 8^0$$

$$N = 1 \times 64 + 7 \times 8 + 3 \times 1$$

$$N = 64 + 56 + 3$$

$$N = 123$$

$$173_8 = 123_{10}$$

การแปลงเลขจากฐาน 10 เป็นฐาน 16

ฐาน 10		ฐาน		ได้		เศษ
1 2 3	÷	16	=	7	→	11 (B)

$$123_{10} = 7B_{16}$$

การแปลงเลขจากฐาน 8 เป็นฐาน 10

$$N = \sum d \times base^{p}$$

base=16

$$N = 7 \times 16^{1} + B \times 16^{0}$$

$$N = 7 \times 16 + 11 \times 1$$

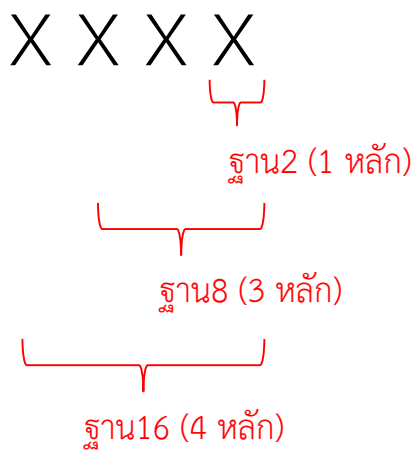
$$N = 112 + 11$$

$$N = 123$$

$$7B_{16} = 123_{10}$$

การแปลงเลขจากฐานระหว่าง 2,8,16

ฐาน2	ฐาน8	ฐาน16
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	10	8
1001	11	9
1010	12	A
1011	13	B
1100	14	C
1101	15	D
1110	16	E
1111	17	F



การแปลงเลขจากฐานระหว่าง 2,8,16

ฐาน2	ฐาน8	ฐาน16
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	10	8
1001	11	9
1010	12	A
1011	13	B
1100	14	C
1101	15	D
1110	16	E
1111	17	F

จงแปลง FFA เป็นฐาน 8 และ 2

ฐาน 16		F	F	A
ฐาน 2		1111	1111	1010
ฐาน 8	7	7	7	2

ตัวอย่างการแสดงวิธีแปลงเลขฐาน

http://www.mathportal.org/calculators/numbers-calculators/decimal-binary-hexadecimal-converter.php

http://goo.gl/oTpWFl

การคำนวณในระบบเลขฐานสอง

การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1

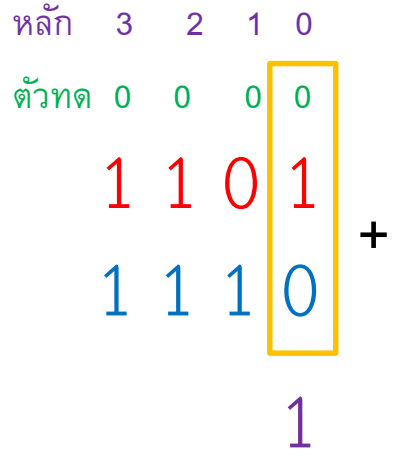
การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1



ตัวทด = 0
ตัวตั้ง = 1
ตัวบวก= 0

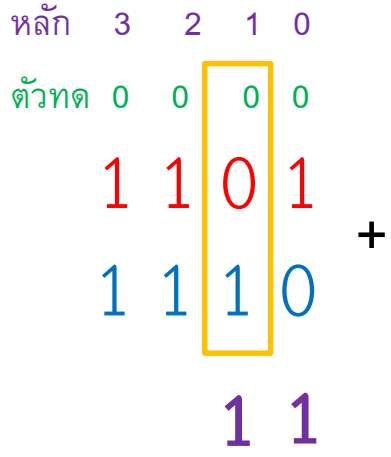
การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1



ตัวทด = 0
ตัวตั้ง = 0
ตัวบวก= 1

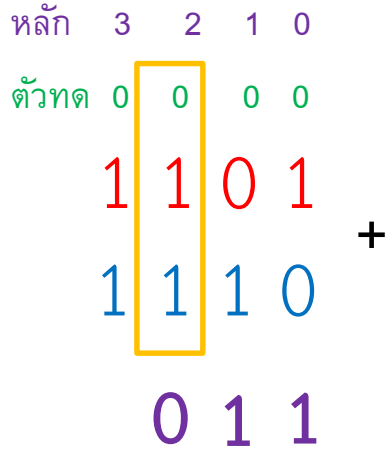
การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1



ตัวทด = 0
ตัวตั้ง = 1
ตัวบวก= 1

การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1

	หลัก	3	2	1	0	
	ตัวทด	1	0	0	0	
		1	1	0	1	+
		1	1	1	0	
ตอบ		1	1	0	1	1

ตัวทด = 1
ตัวตั้ง = 1
ตัวบวก= 1

การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1

1. 1 + 0 = 1

2. 0 + 1 = 1

3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1

1. 1 + 0 = 0 ทด 1

2. 0 + 1 = 0 ทด 1

3. 0 + 0 = 1

หลัก	5	4	3	2	1	0
ตัวทด	0	0	0	0	0	0
	1	1	0	1	1	1
+	0	1	1	0	1	1
						0

ตัวทด = 0
ตัวตั้ง = 1
ตัวบวก= 1

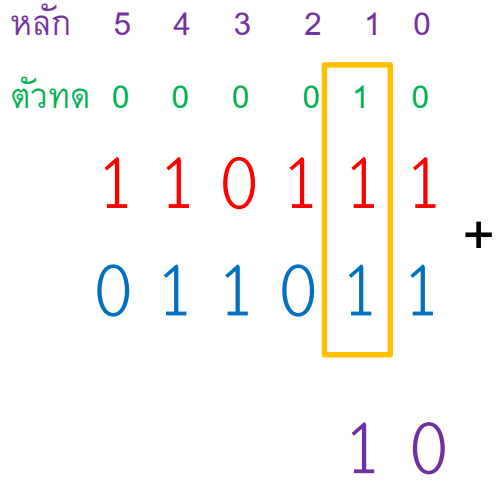
การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1



ตัวทด = 1
ตัวตั้ง = 1
ตัวบวก= 1

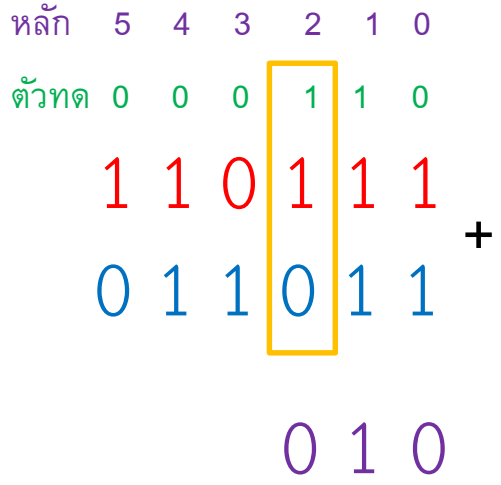
การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1



ตัวทด = 1
ตัวตั้ง = 1
ตัวบวก= 0

การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1

- ตัวอย่าง

```
  1001
  1110   +
-------
 10111
```

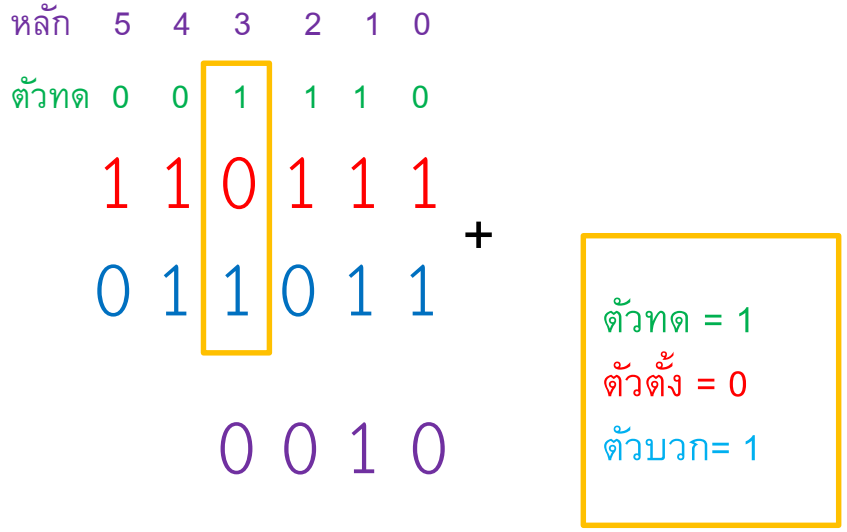
การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1



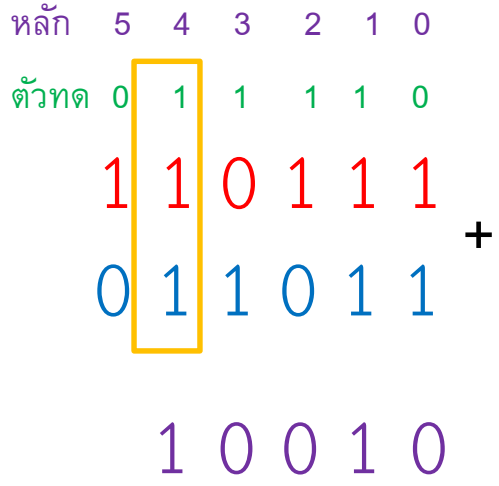
การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1



ตัวทด = 1
ตัวตั้ง = 1
ตัวบวก= 1

การบวกเลข

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1

หลัก	5	4	3	2	1	0
ตัวทด	1	1	1	1	1	0
	1	1	0	1	1	1
+	0	1	1	0	1	1
ตอบ	1	0	1	0	0	1	0

ตัวทด = 1
ตัวตั้ง = 1
ตัวบวก= 0

การบวกเลข

ทดลองบวกเลขฐานสองดังต่อไปนี้

1101 + 1111 = 1 1100

1011 + 100000 = 10 1011

111 + 11111 = 10 0110

- กฎการบวก (ไม่มีตัวทด)

0. 1 + 1 = 0 ทด 1
1. 1 + 0 = 1
2. 0 + 1 = 1
3. 0 + 0 = 0

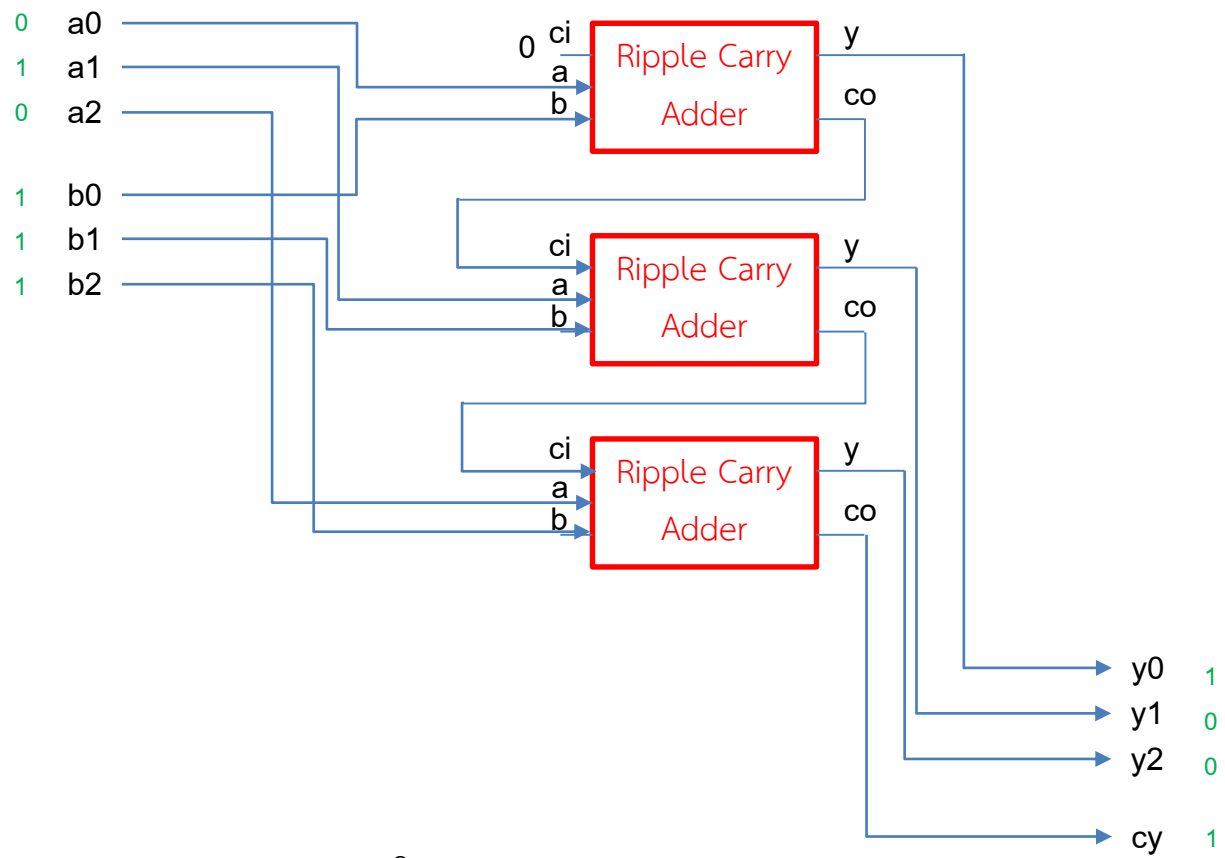
- กฎการบวก (มีตัวทด)

0. 1 + 1 = 1 ทด 1
1. 1 + 0 = 0 ทด 1
2. 0 + 1 = 0 ทด 1
3. 0 + 0 = 1

Input			Output	
ตัวทด (ci)	ตัวตั้ง (a)	ตัวบวก (b)	คำตอบ (y)	ตัวทด (co)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

ไมโครโพนเซสเซอร์ใช้หลักการเดียวกันนี้ในการสร้างวงจรบวกเลข

Input			Output	
ตัวทด (ci)	ตัวตั้ง (a)	ตัวบวก (b)	คำตอบ (y)	ตัวทด (co)
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



วงจรบวกเลข 3 บิตแบบ Ripple Carry Adder

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก	3	2	1	0	
ตัวยืม	0	0	0	0	
	1	0	0	1	
			1	1	-
				0	

ตัวยืม = 0
ตัวตั้ง = 1
ตัวลบ= 1

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก	3	2	1	0	
ตัวยืม	0	0	0	0	
	1	0	0	1	
			1	1	-
	1	0			

ตัวยืม = 0
ตัวตั้ง = 0
ตัวลบ= 1

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

หลัก	3	2	1	0	
ตัวยืม	0	1	0	0	
	1	0	0	1	
			1	1	-
	1	1	0		

ตัวยืม = 1
ตัวตั้ง = 0
ตัวลบ= 0

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

หลัก	3	2	1	0	
ตัวยืม	1	1	0	0	
	1	0	0	1	
			1	1	-

ตอบ 0 1 1 0

ตัวยืม = 1
ตัวตั้ง = 1
ตัวลบ= 0

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก	3	2	1	0
ตัวยืม	0	0	0	0
	1	1	1	1
		1	1	1

-

0

ตัวยืม = 0
ตัวตั้ง = 1
ตัวลบ= 1

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก	3	2	1	0	
ตัวยืม	0	0	0	0	
	1	1	1	1	
		1	1	1	-
			0	0	

ตัวยืม = 0
ตัวตั้ง = 1
ตัวลบ= 1

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก	3	2	1	0
ตัวยืม	0	0	0	0
	1	1	1	1
		1	1	1
	0	0	0	

-

ตัวยืม = 0
ตัวตั้ง = 1
ตัวลบ= 1

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก	3	2	1	0
ตัวยืม	0	0	0	0
	1	1	1	1
		1	1	1

-

1 0 0 0

ตัวยืม = 0
ตัวตั้ง = 1
ตัวลบ= 0

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก 1 0

ตัวยืม 0 0

1 0

1 1 -

1

ตัวยืม = 0

ตัวตั้ง = 0

ตัวลบ= 1

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก 1 0

ตัวยืม 1 0

1 0

1 1 -

1 1

ตัวยืม = 1
ตัวตั้ง = 1
ตัวลบ= 1

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก 1 0

ตัวยืม 1 1 0

1 0

1 1 -

1 1 1

ตัวยืม = 1

ตัวตั้ง = 0

ตัวลบ= 0

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

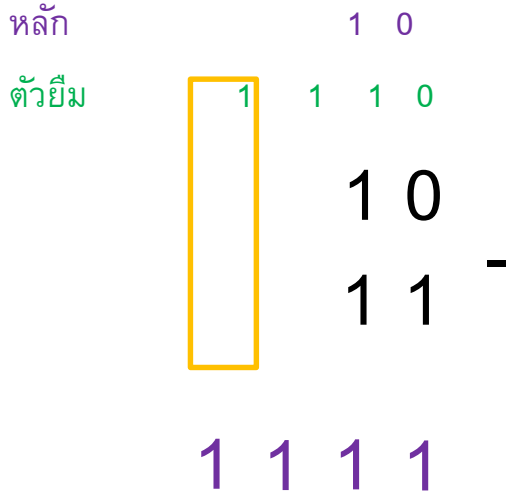
การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1



ตัวยืม = 1
ตัวตั้ง = 0
ตัวลบ= 0

การลบเลข

- กฎการลบ (ไม่มีตัวยืม)

0. 1 - 1 = 0
1. 1 - 0 = 1
2. 0 - 1 = 1 ยืม 1
3. 0 - 0 = 0

หลัก 1 0

ตัวยืม 1 1 1 0

1 0

1 1 -

∞... 1 1 1 1 1 1 1

ตัวยืม = 1
ตัวตั้ง = 0
ตัวลบ= 0

- กฎการบวก (มีตัวยืม)

0. 1 - 1 = 1 ยืม 1
1. 1 - 0 = 0
2. 0 - 1 = 0 ยืม 1
3. 0 - 0 = 1 ยืม 1

ตอบ -1

การแทนจำนวนติดลบ

ค่าติดลบจากการคำนวณเลขฐาน 2 จะทำให้เกิดเลข 1 อยู่ด้านหน้าไปจนถึงหลัก ∞

แต่ไม่โครโพรเซสเซอร์มีหน่วยความจำอย่างจำกัน ดังนั้นการแทนจำนวนในระบบคอมพิวเตอร์ต้องมีการกำหนด ขนาด หรือนิยามค่ามากที่สุด โดยใช้จำนวนหลัก (Digit) ที่ใช้ในการแทนจำนวน

1 หลัก แทนได้ 2 จำนวน คือ 0 , 1
2 หลัก แทนได้ 4 จำนวน คือ 00,01,10,11
3 หลัก แทนได้ 8 จำนวน คือ 000,001,010,011,100,101,110,111

จำนวนหลักของเลขฐานสอง เรียกว่า Binary Digit หรือ Bit

จำนวนสูงสุดที่แทนได้ = 2^{Bit}

ขนาดของจำนวนที่นิยมใช้ในระบบคอมพิวเตอร์คือ

1 bit, 4 bits, 8 bits, 16 bits, 32 bits, 64 bits, 128 bits

การแทนจำนวนติดลบ

วิธีการสร้างเลขฐานสองแบบติดลบในระบบคอมพิวเตอร์

1 ต้องนิยาม หรือกำหนดขนาดของข้อมูล หรือจำนวนบิต

2 ทำการแปลง 2's complement

2.1 ทำการกลับบิต ให้เป็นค่าตรงข้าม

2.2 นำค่าที่ได้ + 1

การแทนจำนวนติดลบ

1 ต้องนิยาม หรือกำหนดขนาดของข้อมูล หรือจำนวนบิต

2 ทำการแปลง 2's complement

2.1 ทำการกลับบิต ให้เป็นค่าตรงข้าม

2.2 นำค่าที่ได้ + 1

ตัวอย่าง จงแปลง 10_2 ให้เป็น 2's complement โดยกำหนดให้มีขนาด 8 บิต

ทำให้มีขนาด 8 บิตด้วยการเติม 0 ด้านซ้ายมือ	**0 0 0 0 0 0 1 0**	
กลับบิตให้เป็นค่าตรงข้าม	**1 1 1 1 1 1 0 1**	
ค่าที่ได้ + 1	**1**	**+**
ตอบ	**1 1 1 1 1 1 1 0**	

ทดลองตรวจคำตอบได้ง่าย ๆ เพราะการทำ 2's complement ครั้งที่ 2 คำตอบจะกลับมาเป็นค่าเดิม

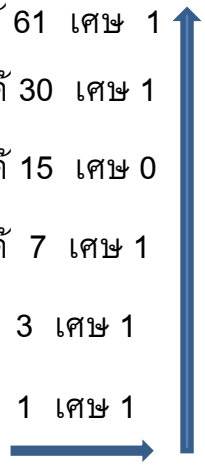
ค่าเริ่มต้นมีขนาด 8 บิตอยู่แล้ว	**1 1 1 1 1 1 1 0**	
กลับบิตให้เป็นค่าตรงข้าม	**0 0 0 0 0 0 0 1**	
ค่าที่ได้ + 1	**1**	**+**
ตอบ	**0 0 0 0 0 0 1 0**	

ได้เท่าเดิม

จงแปลงเลขฐาน10 ค่า -123 ให้อยู่ในเป็นเลขฐาน 2 ขนาด 8 บิต

1) แปลงเลข 123 เป็นเลขฐาน 2 โดยยังไม่คิดเครื่องหมาย

123 / 2 ได้ 61 เศษ 1
61 / 2 ได้ 30 เศษ 1
30 / 2 ได้ 15 เศษ 0
15 / 2 ได้ 7 เศษ 1
7 / 2 ได้ 3 เศษ 1
3 / 2 ได้ 1 เศษ 1



ได้เป็น 1 1 1 1 0 1 1

2) เนื่องจากมีค่าติดลบ จึงต้องแปลงเป็น 2's complement

เติม 0 ด้านซ้ายให้ครบ 8 บิต 0 1 1 1 1 0 1 1
กลับบิตเป็นตรงข้าม 1 0 0 0 0 1 0 0
+
บวก 1 0 0 0 0 0 0 0 1
ตอบ 1 0 0 0 0 1 0 1

จำนวนเต็มในระบบคอมพิวเตอร์มีอยู่ 2 แบบคือ

- **คิดเครื่องหมาย (Signed)**
- **ไม่คิดเครื่องหมาย (Unsigned)**

ตัวอย่างเช่นเลขฐาน 2 ขนาด 8 บิตค่า **1 0 0 0 0 1 0 1** หากเป็นแบบ**คิดเครื่องหมาย** ให้เติม 0 ด้านซ้ายจนครบทุกบิต แล้วดูบิตซ้ายสุดว่าเป็น 1 หรือไม่

1 0 0 0 0 1 0 1 กรณีนี้บิตซ้ายสุดเป็น 1 แสดงว่าเป็นเลขติดลบ ให้ทำการแปลง 2's complement อีก 1 ครั้งจะได้ค่าจริง

กลับบิต +1

1 0 0 0 0 1 0 1 -> 0 1 1 1 1 0 1 0 -> 0 1 1 1 1 0 1 1

แปลงเป็นฐาน 10 ได้เป็น $(0x2^7)+ (1x2^6) + (1x2^5) + (1x2^4) + (1x2^3) + (0x2^2) + (1x2^1) + (1x2^0)$

64 + 32 + 16 + 8 +2 + 1 = 123

ตอบ -123

จำนวนเต็มในระบบคอมพิวเตอร์มีอยู่ 2 แบบคือ

- **คิดเครื่องหมาย (Signed)**
- **ไม่คิดเครื่องหมาย (Unsigned)**

ตัวอย่างเช่นเลขฐาน 2 ขนาด 8 บิตค่า **1 0 0 0 0 1 0 1** หากเป็นแบบ**ไม่คิดเครื่องหมาย** สามารถแปลงเป็นเลขฐาน 10 ได้ทันที

แปลงเป็นฐาน 10 ได้เป็น $(1x2^7)+ (0x2^6) + (0x2^5) + (0x2^4) + (0x2^3) + (1x2^2) + (0x2^1) + (1x2^0)$

128 **+4** **+ 1 = 133**

ตอบ 133

จำนวนเต็มในระบบคอมพิวเตอร์มีอยู่ 2 แบบคือ

- **คิดเครื่องหมาย (Signed)**
- **ไม่คิดเครื่องหมาย (Unsigned)**

ตัวอย่างเช่นเลขฐาน 2 ขนาด **16** บิตค่า **1 0 0 0 0 1 0 1** หากเป็นแบบ**คิดเครื่องหมาย** แปลงเป็นเลขฐาน 10 ได้เท่าใด ?

ให้เติม 0 จนครบ 16 บิต **0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 1**

บิตซ้ายสุดมีค่าเป็น 0 แสดงว่าไม่ติดลบ จึงไม่ต้องทำ 2' complement อีก และสามารถแปลงเป็นฐาน 10 ได้เลย

แปลงเป็นฐาน 10 ได้เป็น $(1x2^7)+(0x2^6)+(0x2^5)+(0x2^4)+(0x2^3)+(1x2^2)+(0x2^1)+(1x2^0)$

128 **+4** **+ 1 = 133**

ตอบ 133

ไมโครโพรเซสเซอร์ไม่จำเป็นต้องมีวงจรลบเลข หากโปรแกรมเมอร์ต้องการคำนวณการลบ สามารถทำได้โดยการนำค่าที่ต้องการลบไปหา 2's complement ก่อน จากนั้นจึงนำไปบวก

ตัวอย่างเช่นจงคำนวณ 9 - 3

หากใช้วงจรบวกเลขขนาด 8 บิต จะมีวิธีการทำดังนี้

ตัวตั้ง 9 แปลงเป็นเลขฐาน 2 ขนาด 8 บิตได้เป็น 0 0 0 0 1 0 0 1

ตัวลบ 3 แปลงเป็นเลขฐาน 2 ขนาด 8 บิตได้เป็น 0 0 0 0 0 0 1 1

แปลงตัวลบเป็น 2's complement เพื่อให้มีค่าเป็น -3 ได้เป็น 1 1 1 1 1 1 0 1

\+

นำไปบวกกับตัวตั้ง 0 0 0 0 1 0 0 1

ได้ผลบวกเป็น 1 1 1 1 1 0 1 0

ทำ 2's complement อีกครั้งเพื่อทำให้เครื่องหมายติดลบ หายไป 0 0 0 0 0 1 1 0

ตอบ 0 0 0 0 0 1 1 0 แปลงเป็น ฐาน 10 คือ 6

จะพบว่าวงจรลบเลขนั้นไม่จำเป็น มีเพียงวงจรบวกเลข (ADD) และวงจรกลับบิต (NOT) ก็เพียงพอแล้ว

เส้นจำนวนของจำนวนเต็มแบบคิดเครื่องหมาย

0	0	0
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
10	a	10
11	b	11
12	c	12
13	d	13
14	e	14
15	f	15
16	10	16
17	11	17
18	12	18
19	13	19
20	14	20
21	15	21
22	16	22
23	17	23
24	18	24
25	19	25
26	1a	26
27	1b	27
28	1c	28
29	1d	29
30	1e	30
31	1f	31
32	20	32
33	21	33
34	22	34
35	23	35
36	24	36
37	25	37
38	26	38
39	27	39
40	28	40
41	29	41
42	2a	42
43	2b	43
44	2c	44
45	2d	45
46	2e	46
47	2f	47
48	30	48
49	31	49
50	32	50
51	33	51
52	34	52
53	35	53
54	36	54
55	37	55
56	38	56
57	39	57
58	3a	58
59	3b	59
60	3c	60
61	3d	61
62	3e	62
63	3f	63

64	40	64
65	41	65
66	42	66
67	43	67
68	44	68
69	45	69
70	46	70
71	47	71
72	48	72
73	49	73
74	4a	74
75	4b	75
76	4c	76
77	4d	77
78	4e	78
79	4f	79
80	50	80
81	51	81
82	52	82
83	53	83
84	54	84
85	55	85
86	56	86
87	57	87
88	58	88
89	59	89
90	5a	90
91	5b	91
92	5c	92
93	5d	93
94	5e	94
95	5f	95
96	60	96
97	61	97
98	62	98
99	63	99
100	64	100
101	65	101
102	66	102
103	67	103
104	68	104
105	69	105
106	6a	106
107	6b	107
108	6c	108
109	6d	109
110	6e	110
111	6f	111
112	70	112
113	71	113
114	72	114
115	73	115
116	74	116
117	75	117
118	76	118
119	77	119
120	78	120
121	79	121
122	7a	122
123	7b	123
124	7c	124
125	7d	125
126	7e	126
127	7f	127

128	80	-128
129	81	-127
130	82	-126
131	83	-125
132	84	-124
133	85	-123
134	86	-122
135	87	-121
136	88	-120
137	89	-119
138	8a	-118
139	8b	-117
140	8c	-116
141	8d	-115
142	8e	-114
143	8f	-113
144	90	-112
145	91	-111
146	92	-110
147	93	-109
148	94	-108
149	95	-107
150	96	-106
151	97	-105
152	98	-104
153	99	-103
154	9a	-102
155	9b	-101
156	9c	-100
157	9d	-99
158	9e	-98
159	9f	-97
160	a0	-96
161	a1	-95
162	a2	-94
163	a3	-93
164	a4	-92
165	a5	-91
166	a6	-90
167	a7	-89
168	a8	-88
169	a9	-87
170	aa	-86
171	ab	-85
172	ac	-84
173	ad	-83
174	ae	-82
175	af	-81
176	b0	-80
177	b1	-79
178	b2	-78
179	b3	-77
180	b4	-76
181	b5	-75
182	b6	-74
183	b7	-73
184	b8	-72
185	b9	-71
186	ba	-70
187	bb	-69
188	bc	-68
189	bd	-67
190	be	-66
191	bf	-65

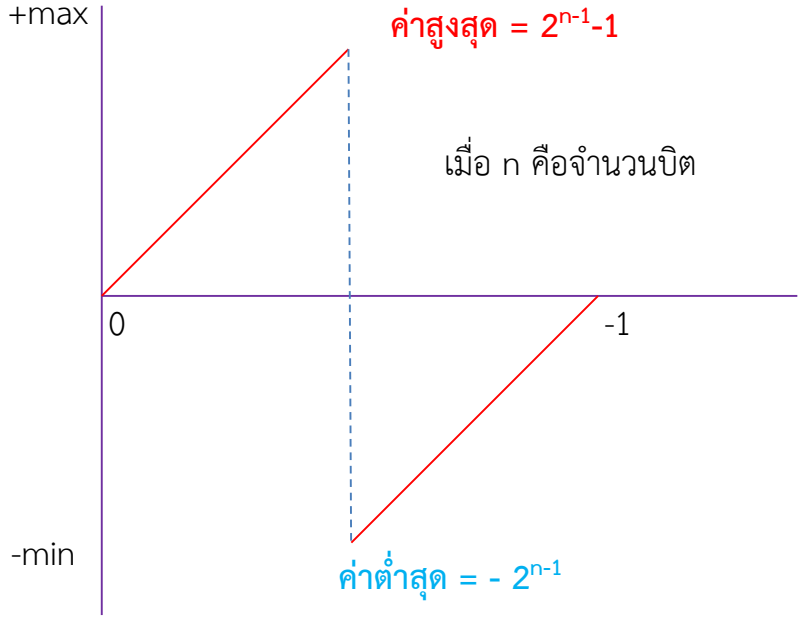
192	c0	-64
193	c1	-63
194	c2	-62
195	c3	-61
196	c4	-60
197	c5	-59
198	c6	-58
199	c7	-57
200	c8	-56
201	c9	-55
202	ca	-54
203	cb	-53
204	cc	-52
205	cd	-51
206	ce	-50
207	cf	-49
208	d0	-48
209	d1	-47
210	d2	-46
211	d3	-45
212	d4	-44
213	d5	-43
214	d6	-42
215	d7	-41
216	d8	-40
217	d9	-39
218	da	-38
219	db	-37
220	dc	-36
221	dd	-35
222	de	-34
223	df	-33
224	e0	-32
225	e1	-31
226	e2	-30
227	e3	-29
228	e4	-28
229	e5	-27
230	e6	-26
231	e7	-25
232	e8	-24
233	e9	-23
234	ea	-22
235	eb	-21
236	ec	-20
237	ed	-19
238	ee	-18
239	ef	-17
240	f0	-16
241	f1	-15
242	f2	-14
243	f3	-13
244	f4	-12
245	f5	-11
246	f6	-10
247	f7	-9
248	f8	-8
249	f9	-7
250	fa	-6
251	fb	-5
252	fc	-4
253	fd	-3
254	fe	-2
255	ff	-1

ตัวอย่างเส้นจำนวนของเลข 2's complement ขนาด 8 บิต

จำนวนตัวเลขทั้งหมด 256

ค่าสูงสุด 127

ค่าน้อยสุด -128



การป้อนข้อมูลตรวจสอบให้อยู่ในช่วง เพราะถ้าเกิดการล้น
หรือ Overflow
ข้อมูลอาจเปลี่ยนจาก 127 เป็น -128 ได้

การคูณเลข

การคูณเลขฐาน 2

1. ถ้าตัวคูณ (multiplier) มีค่าเป็น 1 ให้ copy ตัวตั้งทั้งหมดลงมา แต่ถ้าตัวคูณเป็น 0 ไม่ต้องทำอะไรทั้งสิ้น)
2. เลื่อนค่าที่ copy ลงมา ในกรณีเดียวกับการคูณเลขฐาน 10
3. รวมผลลัพธ์ทั้งหมดเข้าด้วยกัน

การคูณเลข

การคูณเลขฐาน 2

1. ถ้าตัวคูณ (multiplier) มีค่าเป็น 1 ให้ copy ตัวตั้งทั้งหมดลงมา แต่ถ้าตัวคูณเป็น 0 ไม่ต้องทำอะไรทั้งสิ้น)
2. เลื่อนค่าที่ copy ลงมา ในกรณีเดียวกับการคูณเลขฐาน 10
3. รวมผลลัพธ์ทั้งหมดเข้าด้วยกัน

ตัวอย่าง 110111x101101

```
  1 1 0 1 1 1
                x
  1 0 1 1 0 1
-------------------
    1 1 0 1 1 1
+ 1 1 0 1 1 1
-------------------
  1 0 0 0 1 0 0 1 1
+ 1 1 0 1 1 1
-------------------
  1 0 1 1 0 0 1 0 1 1
+ 1 1 0 1 1 1
-------------------
ตอบ 1 0 0 1 1 0 1 0 1 0 1 1
```

การหารเลข

- กฎการหาร
  - ดำเนินการในลักษณะเดียวกันกับการหารเลขในฐานสิบ

การหารเลข

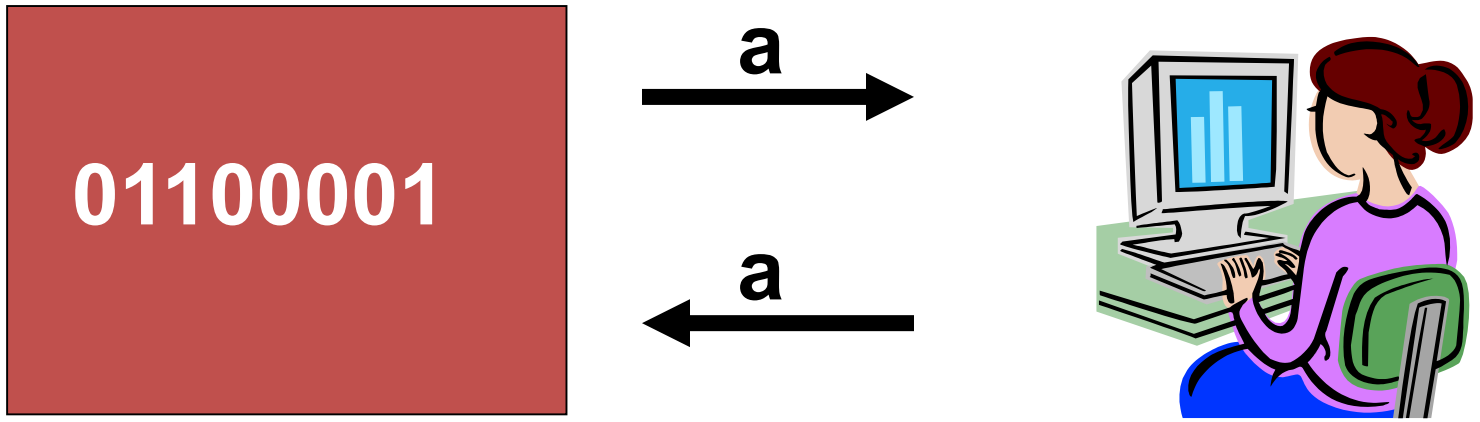
จงหาค่า 100001 ÷ 110

```
          101
      ------
110  )100001
     -
       110
      ----
        1001
      -
         110
        ----
          11
```

ตอบ 101 เศษ 11

การแทนข้อมูลตัวเลขและตัวอักษร ในเครื่องคอมพิวเตอร์

การเก็บข้อมูลไว้ในเครื่องคอมพิวเตอร์



การเก็บข้อมูลไว้ในเครื่องคอมพิวเตอร์

- ตัวเลขและตัวอักษรที่ผู้ใช้ป้อนเข้าเครื่องคอมพิวเตอร์ ไม่ได้เก็บโดยใช้สัญลักษณ์เดียวกับที่ป้อนเข้าไป
- Bit – ส่วนที่เล็กที่สุดในส่วนความจำที่สามารถเก็บข้อมูลได้
  - แต่ละ bit สามารถเก็บข้อมูลที่แทนด้วยสัญลักษณ์เพียง 2 สัญลักษณ์ คือ 0 หรือ 1
- Bit จะถูกนำมารวมกันเพื่อแทนตัวเลขหรือตัวอักษร
- Byte – กลุ่มของ bit จำนวน 8 bits

  00000000 -> 11111111

หน่วยวัดขนาดของหน่วยความจำ

มาตรฐาน เมตริก

ค่า	ตัวย่อ	หน่วยเมตริก	คำอ่าน
1000	kB	Kilobyte	กิโลไบต์
1000^2	MB	Megabyte	เมกะไบต์
1000^3	GB	Gigabyte	กิกะไบต์
1000^4	TB	Terabyte	เทราไบต์
1000^5	PB	Petabyte	เพตาไบต์
1000^6	EB	Exabyte	เอกสะไบต์
1000^7	ZB	Zettabyte	เซตะไบต์
1000^8	YB	Yottabyte	โยตะไบต์

มาตรฐาน ไออีซี

ค่า	ตัวย่อ	หน่วยเมตริก	คำอ่าน
1024	KiB	Kibibyte	คีบีไบต์
1024^2	MiB	Mebibyte	เมบิไบต์
1024^3	GiB	Gibibyte	กิบิไบต์
1024^4	TiB	Tebibyte	เทบิไบต์
1024^5	PiB	Pebibyte	เพบิไบต์
1024^6	EiB	Exbibyte	เอกบิไบต์
1024^7	ZiB	Zebibyte	เซบิไบต์
1024^8	YiB	Yobibyte	โยบิไบต์

WORD

- WORD เป็นหน่วยวัดขนาดความจำอีกประเภทหนึ่ง แต่มีขนาดไม่คงที่ แล้วแต่เครื่องคอมพิวเตอร์นั้น ๆ กำหนดขึ้น เช่น
    - 1 word = 2 bytes
    - 1 word = 4 bytes
    - 1 word = 8 bytes เป็นต้น

2. การเก็บข้อมูลในเครื่องคอมพิวเตอร์

- ข้อมูลที่เก็บไว้ในหน่วยความจำของเครื่องคอมพิวเตอร์ อาจแบ่งง่าย ๆ เป็น 3 กลุ่ม คือ
  - ตัวอักษรภาษาอังกฤษจาก A ถึง Z จำนวน 26 ตัว
  - ตัวเลขจาก 0 ถึง 9 จำนวน 10 ตัว
  - เครื่องหมายทางคณิตศาสตร์และอื่นๆ จำนวน 16 ตัว

  รวม 52 ตัว

กลุ่มข้อมูลพื้นฐาน - ข้อมูลประเภทตัวอักษร

- ข้อมูลประเภทนี้จะมีการกำหนดรหัสแทนตัวอักษรต่าง ๆ
- โดยทั่วไป รหัสแทนตัวอักษรจะใช้เนื้อที่ 1 ไบท์
- ระบบที่ใช้ในการกำหนดรหัส ให้แก่ตัวอักษรและสัญลักษณ์ ได้แก่
  - ระบบ BCD (Binary Coded Decimal)
  - ระบบเอบซีดิก (EBCDIC)
  - ระบบแอสกี (ASCII)
  - ระบบยูนิโค้ด (UNICODE)

ระบบ BCD (Binary Coded Decimal)

- เข้ารหัสข้อมูลด้วยขนาด 6 บิทต่อ 1 ไบท์
- กำหนดให้ 4 บิทแรก (1-4) เป็นบิทตัวเลข (Numerical Bit) และอีก 2 บิทที่เหลือ (5-6) เป็นโซนาบิท (Zone Bit)
- บิทที่ 7 / 6 5 / 4 3 2 1

  Parity / Zone / Numeric
- ค่าของ Check bit ขึ้นอยู่กับโครงสร้างของเครื่องคอมพิวเตอร์ว่าเป็นแบบบิทคี่ (Odd Parity Bit) หรือแบบบิทคู่ (Even Parity)

แบบบิทคี่ (Odd Parity Bit)

- ถ้ารหัสข้อมูลใน 6 บิทมีเลข 1 เป็นจำนวนคู่แล้ว Parity เป็นเลข 1

Parity	Zone	Numeric
1	11	1001
1	01	0100

- แต่ถ้ารหัสข้อมูลใน 6 บิทมีเลข 1 เป็นจำนวนคี่แล้ว Parity เป็นเลข 0

Parity	Zone	Numeric
0	01	1100
0	11	0100

แบบบิทคู่ (Even Parity)

- ถ้ารหัสข้อมูลใน 6 บิทมีเลข 1 เป็นจำนวนคู่แล้ว Parity เป็นเลข 0

Parity	Zone	Numeric
0	00	0011
0	10	1011

- แต่ถ้ารหัสข้อมูลใน 6 บิทมีเลข 1 เป็นจำนวนคี่แล้ว Parity เป็นเลข 1

Parity	Zone	Numeric
1	11	1000
1	01	0101

CDC 1604 : Punched card codes

ระบบ BCD (Binary Coded Decimal)

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		1	2	3	4	5	6	7	8	9	0	=	-			
1.	space	/	S	T	U	V	W	X	Y	Z		,	(			
2.	---	J	K	L	M	N	O	P	Q	R	-0	$	*			
3.	+	A	B	C	D	E	F	G	H	I	+0	.	)			

2 bits

4 bits

CDC 1604 : Punched card codes

ระบบ BCD (Binary Coded Decimal)

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		1	2	3	4	5	6	7	8	9	0	=	–			
1.	space	/	S	T	U	V	W	X	Y	Z		,	(			
2.	---	J	K	L	M	N	O	P	Q	R	-0	$	*			
3.	+	A	B	C	D	E	F	G	H	I	+0	.	)			

0 0 1 1 0 0 1

Odd Parity

Zone

Numeric

CDC 1604 : Punched card codes

ระบบ BCD (Binary Coded Decimal)

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		1	2	3	4	5	6	7	8	9	0	=	-			
1.	space	/	S	T	U	V	W	X	Y	Z		,	(			
2.	---	J	K	L	M	N	O	P	Q	R	-0	$	*			
3.	+	A	B	C	D	E	F	G	H	I	+0	.	)			

1 0 0 0 0 1 1

Odd Parity Zone Numeric

ระบบ BCD (Binary Coded Decimal)

CDC 1604 : Punched card codes

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		1	2	3	4	5	6	7	8	9	0	=	-			
1.	space	/	S	T	U	V	W	X	Y	Z		,	(			
2.	---	J	K	L	M	N	O	P	Q	R	-0	$	*			
3.	+	A	B	C	D	E	F	G	H	I	+0	.	)			

จงเข้ารหัสคำว่า MAEJO เป็นรหัส BCD แบบ Odd Parity

$	ZN		Z N		P Z N		HEX
M=	24	→	10 0100	→	1 100100	→	64
A=	31	→	11 0001	→	0 110001	→	31
E=	35	→	11 0101	→	1 110101	→	75
J=	21	→	10 0001	→	1 100001	→	61
O=	26	→	10 0110	→	0 100110	→	26

ตอบ 64 31 75 61 26

CDC 1604 : Punched card codes

ระบบ BCD (Binary Coded Decimal)

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		1	2	3	4	5	6	7	8	9	0	=	-			
1.	space	/	S	T	U	V	W	X	Y	Z		,	(			
2.	---	J	K	L	M	N	O	P	Q	R	-0	$	*			
3.	+	A	B	C	D	E	F	G	H	I	+0	.	)			

จงเข้ารหัสคำว่า MAEJO เป็นรหัส BCD แบบ Event Parity

$	ZN	→	Z N	→	P Z N	→	HEX
M=	24	→	10 0100	→	0 100100	→	24
A=	31	→	11 0001	→	1 110001	→	71
E=	35	→	11 0101	→	0 110101	→	35
J=	21	→	10 0001	→	0 100001	→	21
O=	26	→	10 0110	→	1 100110	→	66

ตอบ 24 71 35 21 66

ระบบ BCD (Binary Coded Decimal)

CDC 1604 : Magnetic tape BCD codes

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		1	2	3	4	5	6	7	8	9	0	#	@			tape mark
1.	space	/	S	T	U	V	W	X	Y	Z	record mark	.	%			
2.	-	J	K	L	M	N	O	P	Q	R	-0	$	*			
3.	&	A	B	C	D	E	F	G	H	I	+0	.	¤			group mark

CDC 1604 : Punched card codes

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.		1	2	3	4	5	6	7	8	9	0	=	-			
1.	space	/	S	T	U	V	W	X	Y	Z		.	(			
2.	---	J	K	L	M	N	O	P	Q	R	-0	$	*			
3.	+	A	B	C	D	E	F	G	H	I	+0	.	)			

CDC 1612 printer codes (business applications)

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.	:	1	2	3	4	5	6	7	8	9	0	=	≠	≤	!	[
1.	space	/	S	T	U	V	W	X	Y	Z	]	.	(	⌴	≡	~
2.	-	J	K	L	M	N	O	P	Q	R	%	$	*	↑	↓	>
3.	+	A	B	C	D	E	F	G	H	I	<	.	)	≥	?	;

DEC SIXBIT:

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.	space	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
1.	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
2.	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3.	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_

ECMA-1:

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.	space	HT	LF	VT	FF	CR	SO	SI	(	)	*	+	,	-	.	/
1.	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
2.	NULL	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
3.	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	ESC	DEL

ระบบเอบซีดิก (EBCDIC)

- Extended Binary Coded Decimal Interchange Code
- ใช้รหัสแทนข้อมูลด้วยขนาด 8 บิทต่อไบท์
- รหัสแทนตัวอักขระของข้อมูลได้มากขึ้นถึง 2^8=256 ตัวอักขระ
- แต่เพื่อความสะดวกในการศึกษามักเขียนเป็นเลขฐานสิบหกซึ่งมีเพียง 2 หลักเท่านั้น
- เช่น 1100 0001 แทนด้วย C1

ระบบเอบซีดิก (EBCDIC)

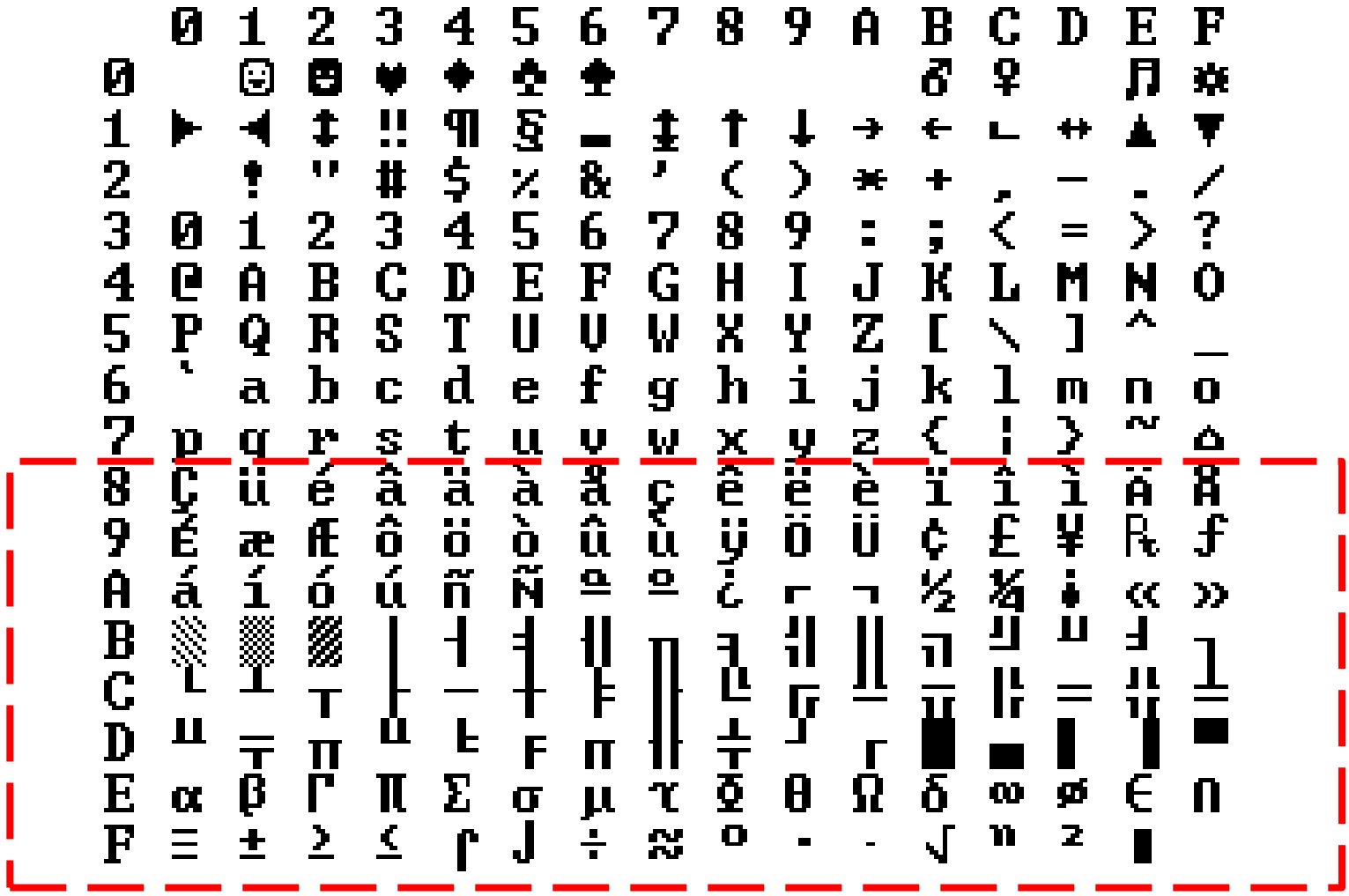
Codepage 924 - Latin 9 - EBCDIC

	-0	-1	-2	-3	-4	-5	-6	-7	-8	-9	-A	-B	-C	-D	-E	-F
0-		0001	0002	0003	009C	0005	0086	0007	0097	008D	008E	000B	000C	000D	000E	000F
1-	0010	0011	0012	0013	009D	0085	0016	0087	0018	0019	0092	008F	001C	001D	001E	001F
2-	0080	0081	0082	0083	0084	000A	0017	001B	0088	0089	008A	008B	008C	0005	0006	0007
3-	0090	0091	0016	0093	0094	0095	0096	0004	0098	0099	009A	009B	0014	0015	009E	001A
4-	0020	00A0	â 00E2	ä 00E4	à 00E0	á 00E1	ã 00E3	å 00E5	ç 00E7	ñ 00F1	Ý 00DD	. 002E	< 003C	(0028	+ 002B	! 0021
5-	& 0026	é 00E9	ê 00EA	ë 00EB	è 00E8	í 00ED	î 00EE	ï 00EF	ì 00EC	ß 00DF	] 005D	$ 0024	* 002A	) 0029	; 003B	^ 005E
6-	- 002D	/ 002F	Â 00C2	Ä 00C4	À 00C0	Á 00C1	Ã 00C3	Å 00C5	Ç 00C7	Ñ 00D1	Š 0160	, 002C	% 0025	_ 005F	> 003E	? 003F
7-	ø 00F8	É 00C9	Ê 00CA	Ë 00CB	È 00C8	Í 00CD	Î 00CE	Ï 00CF	Ì 00CC	` 0060	: 003A	# 0023	@ 0040	' 0027	= 003D	" 0022
8-	Ø 00D8	a 0061	b 0062	c 0063	d 0064	e 0065	f 0066	g 0067	h 0068	i 0069	« 00AB	» 00BB	ð 00F0	ý 00FD	þ 00FE	± 00B1
9-	° 00B0	j 006A	k 006B	l 006C	m 006D	n 006E	o 006F	p 0070	q 0071	r 0072	ª 00AA	º 00BA	æ 00E6	ž 017E	Æ 00C6	€ 20AC
A-	µ 00B5	~ 007E	s 0073	t 0074	u 0075	v 0076	w 0077	x 0078	y 0079	z 007A	¡ 00A1	¿ 00BF	Ð 00F0		Þ 00DE	® 00AE
B-	¢ 00A2	£ 00A3	¥ 00A5	· 00B7	© 00A9	§ 00A7	¶ 00B6	Œ 0152	œ 0153	Ÿ 0178	¬ 00AC	š 0161	¯ 02C9		Ž 017D	× 00D7
C-	{ 007B	A 0041	B 0042	C 0043	D 0044	E 0045	F 0046	G 0047	H 0048	I 0049	- 00AD	ô 00F4	ö 00F6	ò 00F2	ó 00F3	õ 00F5
D-	} 007D	J 004A	K 004B	L 004C	M 004D	N 004E	O 004F	P 0050	Q 0051	R 0052	¹ 00B9	û 00FB	ü 00FC	ù 00F9	ú 00FA	ÿ 00FF
E-	\ 005C	÷ 00F7	S 0053	T 0054	U 0055	V 0056	W 0057	X 0058	Y 0059	Z 005A	² 00B2	Ô 00D4	Ö 00D6	Ò 00D2	Ó 00D3	Õ 00D5
F-	0 0030	1 0031	2 0032	3 0033	4 0034	5 0035	6 0036	7 0037	8 0038	9 0039	³ 00B3	Û 00DB	Ü 00DC	Ù 00D9	Ú 00DA	009F

รหัส ASCII

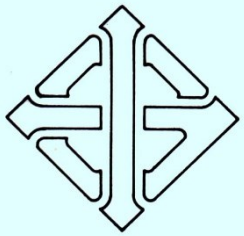
- American Standard Code for Information Interchange
- เดิม : รหัสมาตรฐานขนาด 7 บิท
- ปัจจุบัน : ถูกออกแบบให้แทนอักขระแต่ละตัวโดยใช้ 8 บิต หรือ 1 ไบท์
- อาจจะมี บิทตรวจสอบ(Parity Bit)

รหัส ASCII



20=SPACE

ส่วนขยาย

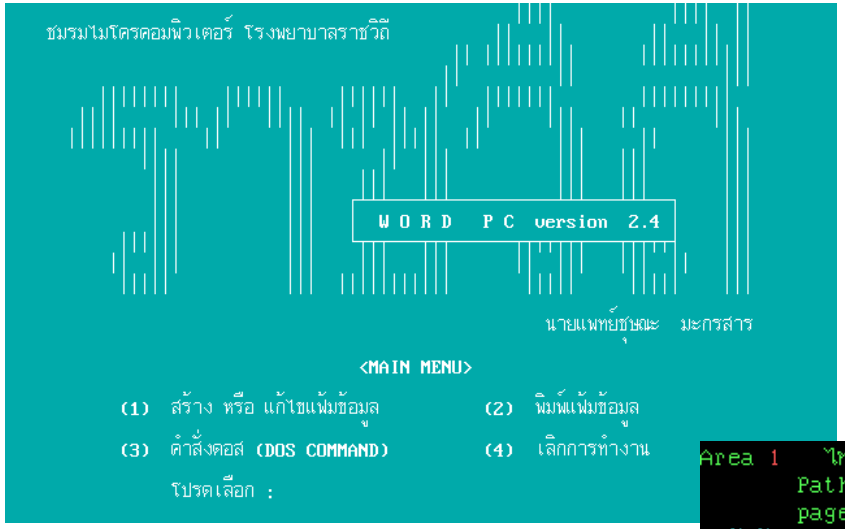


มาตรฐานผลิตภัณฑ์อุตสาหกรรม มอก. 620-2533

รหัสสำหรับอักขระไทยที่ใช้กับคอมพิวเตอร์

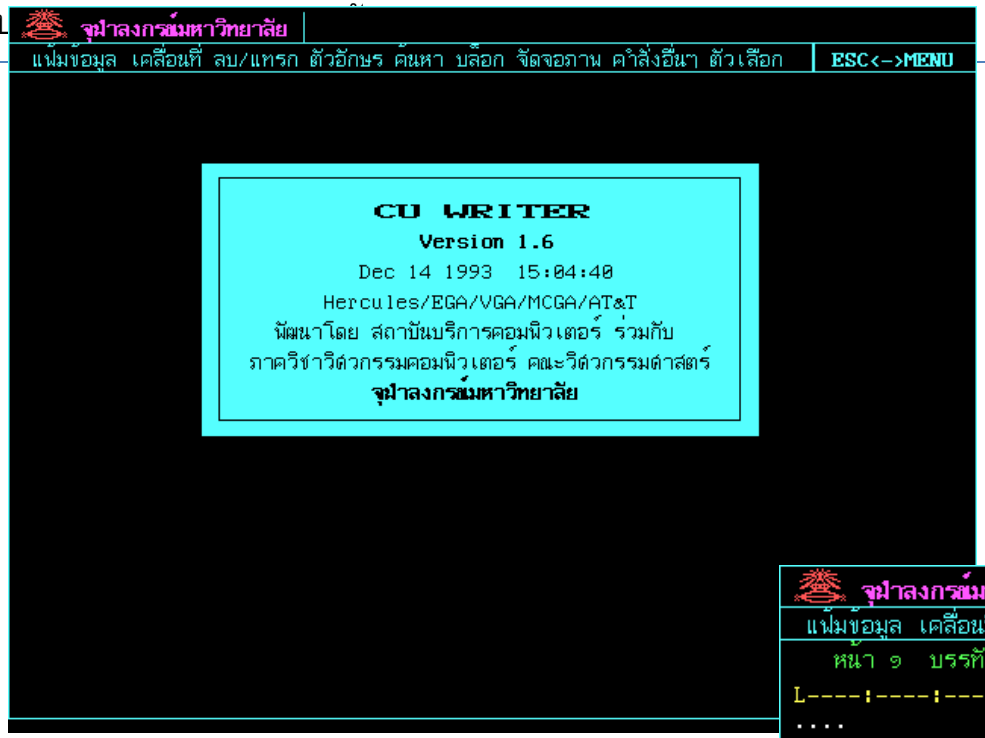
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0												ฐ	ภ	ะ	เ	๐
1	1											ก	ฑ	ม	ั	แ	๑
2	2											ข	ฒ	ย	า	โ	๒
3	3											ฃ	ณ	ร	ำ	ใ	๓
4	4											ค	ด	ฤ	ิ	ไ	๔
5	5											ฅ	ต	ล	ี	ๅ	๕
6	6											ฆ	ถ	ฦ	ึ	ๆ	๖
7	7											ง	ท	ว	ื	็	๗
8	8											จ	ธ	ศ	ุ	่	๘
9	9											ฉ	น	ษ	ู	้	๙
A	10											ช	บ	ส		๊	ฯ
B	11											ซ	ป	ห		๋	฿
C	12											ฌ	ผ	ฬ		์	
D	13											ญ	ฝ	อ		๎	
E	14											ฎ	พ	ฮ		๏	
F	15											ฏ	ฟ	ฯ	฿	๐	

ก เริ่มต้นที่ตำแหน่ง A1

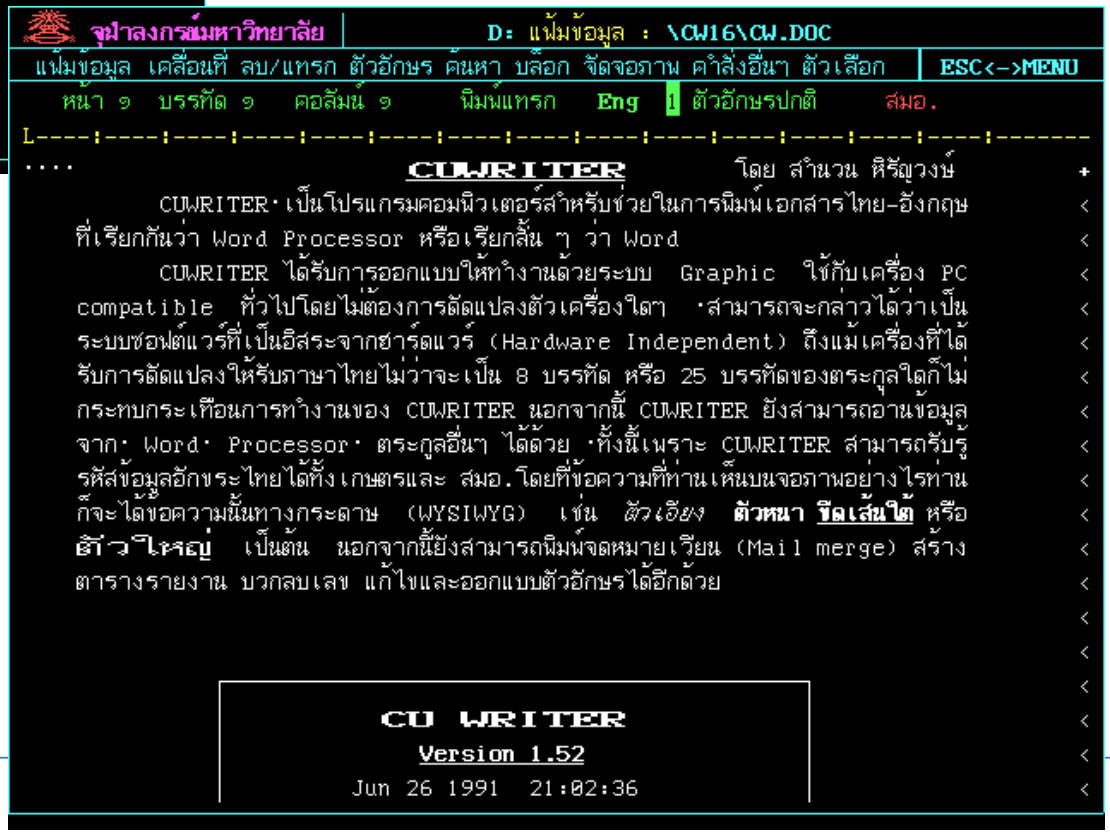


ราชวิถีเวิร์ดพีซี พัฒนาขึ้นโดยชมรมไมโครคอมพิวเตอร์ โรงพยาบาลราชวิถี นำโดยนายแพทย์ชุษณะ มะกรสาร เมื่อ พ.ศ. 2526 และออกเผยแพร่สู่ผู้ใช้ครั้งแรก เมื่อ เมษายน พ.ศ. 2529

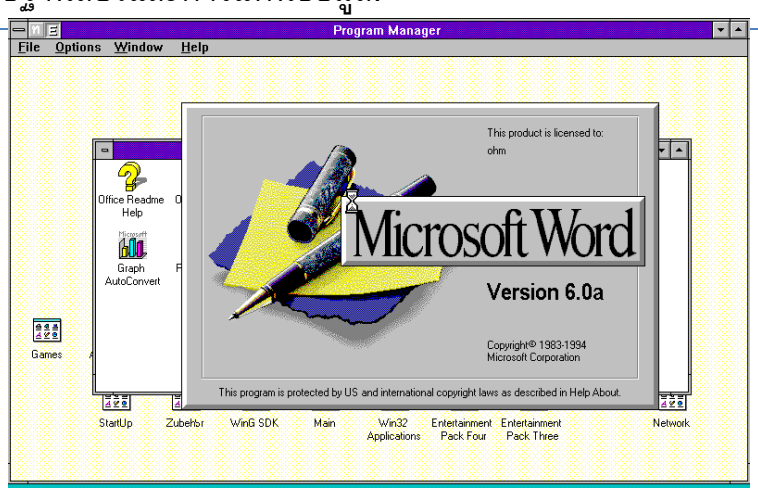
- เปลี่ยนภาษาแป้นพิมพ์ด้วยปุ่ม ESC



ซียูไรเตอร์ (อังกฤษ: CU Writer) หรือเรียกอย่างไม่เป็นทางการว่า จุฬาเวิร์ด, เวิร์ดจุฬา หรือ CW สร้างขึ้นโดยความร่วมมือจาก สถาบันบริการคอมพิวเตอร์ กับภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย นำโดยนายสำนวน หิรัญวงษ์ เมื่อต้นปี พ.ศ. 2532 พัฒนาอย่างต่อเนื่องจนถึง พ.ศ. 2536

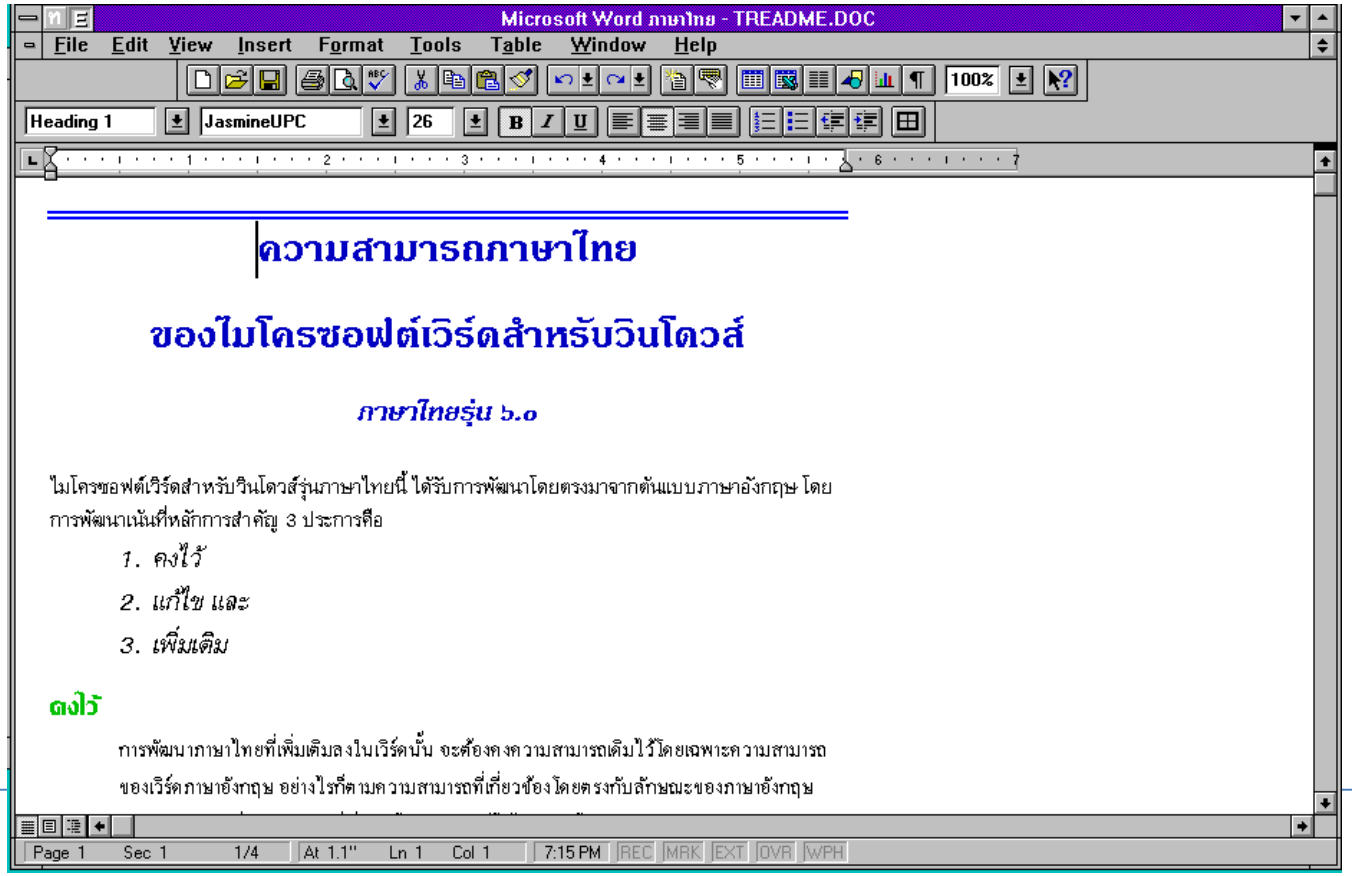


- เปลี่ยนภาษาแป้นพิมพ์ด้วยปุ่ม F10

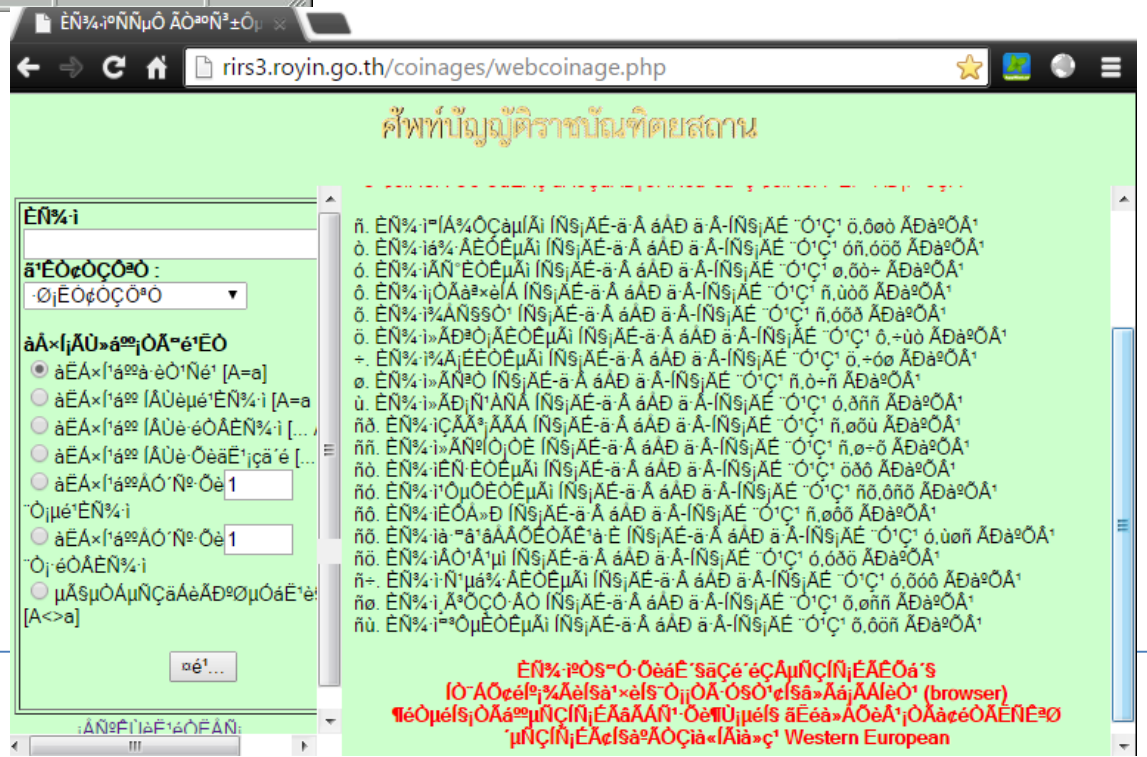


Word for Windows ผลิตโดยบริษัท Microsoft รุ่น 1.0 และ 2.0 วางตลาดในปี 1991 และ รุ่น 3.0 ในปี 1995

•เปลี่ยนภาษาแป้นพิมพ์ด้วยปุ่ม ~

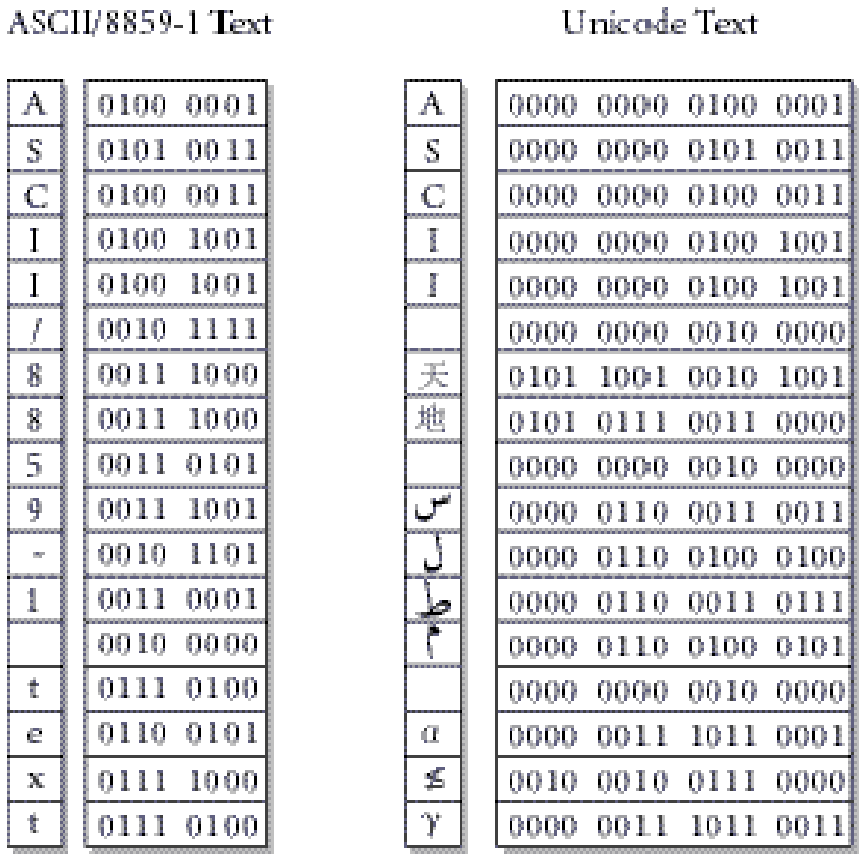


ระบบเลขฐานสองและการแทนข้อมูล:



รหัส UNICODE

- รหัส UNICODE จะเป็นรหัสที่สามารถแสดงตัวอักขระได้มาก เพราะได้ถูกออกแบบให้แทนอักขระแต่ละตัวโดยใช้ 16 บิต หรือ 2 ไบท์

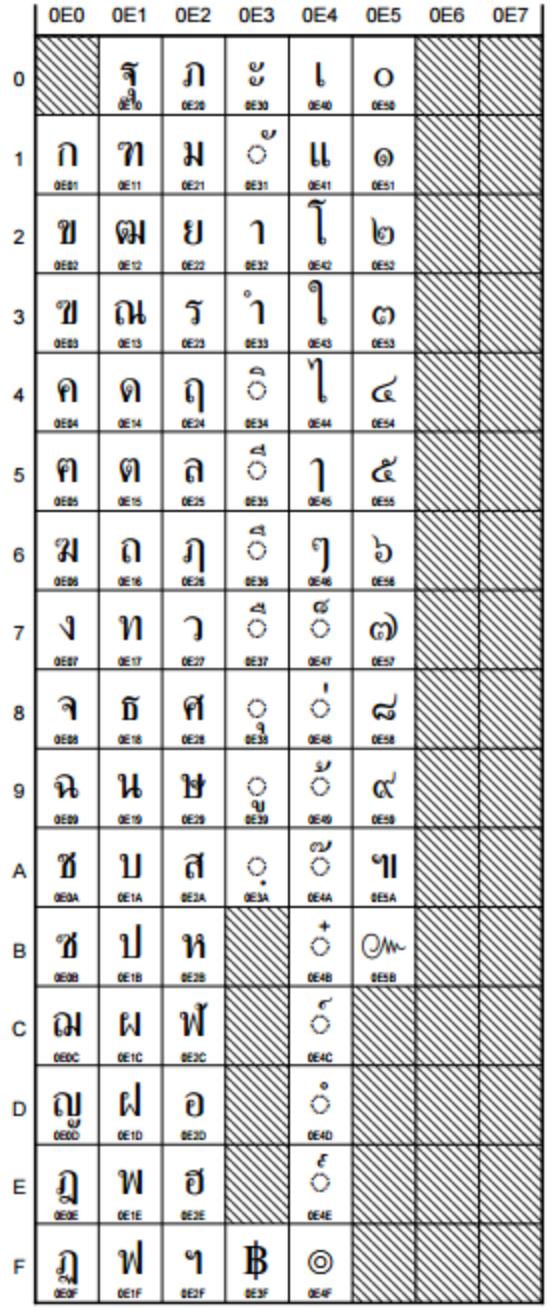


ASCII/8859-1 Text	
A	0100 0001
S	0101 0011
C	0100 0011
I	0100 1001
I	0100 1001
/	0010 1111
8	0011 1000
8	0011 1000
5	0011 0101
9	0011 1001
-	0010 1101
1	0011 0001
	0010 0000
t	0111 0100
e	0110 0101
x	0111 1000
t	0111 0100

Unicode Text	
A	0000 0000 0100 0001
S	0000 0000 0101 0011
C	0000 0000 0100 0011
I	0000 0000 0100 1001
I	0000 0000 0100 1001
	0000 0000 0010 0000
天	0101 1001 0010 1001
地	0101 0111 0011 0000
	0000 0000 0010 0000
س	0000 0110 0011 0011
ل	0000 0110 0100 0100
ط	0000 0110 0011 0111
م	0000 0110 0100 0101
	0000 0000 0010 0000
α	0000 0011 1011 0001
≤	0010 0010 0111 0000
Υ	0000 0011 1011 0011

เปรียบเทียบรหัสของตัวอักขระใน ASCII และ UNICODE

รหัส UNICODE



	0E0	0E1	0E2	0E3	0E4	0E5	0E6	0E7
0		ฐ 0E10	ภ 0E20	ะ 0E30	เ 0E40	๐ 0E50		
1	ก 0E01	ฑ 0E11	ม 0E21	◌ั 0E31	แ 0E41	๑ 0E51		
2	ข 0E02	ฒ 0E12	ย 0E22	า 0E32	โ 0E42	๒ 0E52		
3	ฃ 0E03	ณ 0E13	ร 0E23	◌ำ 0E33	ใ 0E43	๓ 0E53		
4	ค 0E04	ด 0E14	ฤ 0E24	◌ิ 0E34	ไ 0E44	๔ 0E54		
5	ฅ 0E05	ต 0E15	ล 0E25	◌ี 0E35	ๅ 0E45	๕ 0E55		
6	ฆ 0E06	ถ 0E16	ฦ 0E26	◌ึ 0E36	ๆ 0E46	๖ 0E56		
7	ง 0E07	ท 0E17	ว 0E27	◌ื 0E37	◌็ 0E47	๗ 0E57		
8	จ 0E08	ธ 0E18	ศ 0E28	◌ุ 0E38	◌่ 0E48	๘ 0E58		
9	ฉ 0E09	น 0E19	ษ 0E29	◌ู 0E39	◌้ 0E49	๙ 0E59		
A	ช 0E0A	บ 0E1A	ส 0E2A	◌ฺ 0E3A	◌๊ 0E4A	๚ 0E5A		
B	ซ 0E0B	ป 0E1B	ห 0E2B		◌๋ 0E4B	๛ 0E5B		
C	ฌ 0E0C	ผ 0E1C	ฬ 0E2C		◌์ 0E4C			
D	ญ 0E0D	ฝ 0E1D	อ 0E2D		◌ 0E4D			
E	ฎ 0E0E	พ 0E1E	ฮ 0E2E		◌๎ 0E4E			
F	ฏ 0E0F	ฟ 0E1F	ฯ 0E2F	฿ 0E3F	◌๏ 0E4F			

สรุป

- ระบบเลขฐาน
- เลขฐานสอง
- การคำนวณด้วยเลขฐานสอง
- การแทนข้อมูลในระบบคอมพิวเตอร์ด้วยเลขฐานสอง