'z' → ASCII $7A_{16}$ → Unicode 7A

UTF-8 ~~0~~111 1010 → 0 111 1010

7 data bits

7 A

UTF-16 7A is in the BMP

00 7A

Big Endian: 00 7A

Little Endian: 7A 00

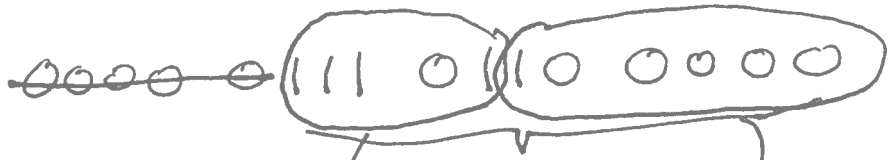
UTF-32 00 00 00 7A

Big Endian: 00 00 00 7A ....

Little Endian: 7A 00 00 00 ....

Unicode 0760_{16} (Thaana)

UTF-8



110 11101 10 10 0000

1101 1101 1010 0000

D D A 0

DD A0

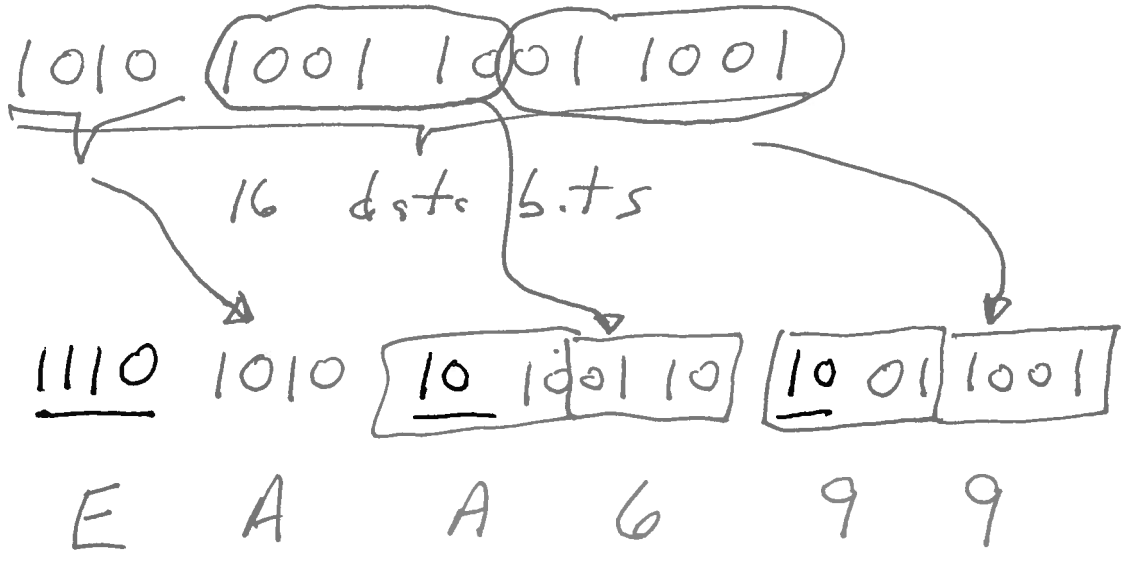
UTF-16

0760 → Big Endian 07 60
Little Endian 60 07

UTF-32 00000760

Unicode $A999_{16}$ (Javanese)

UTF-8



UTF-16 A999 → A9 99 B.E.

→ 99 A9 L.E.

UTF-32 00 00 A9 99

Unicode 12345_{16}

UTF-8 ~~0000~~1 0010 0011 0100 0101

17 data bits

encode in (21) bits in 4 bytes

~~11110 100 10 10 0011 10~~

need to start over!

Unicode 12345_{16} → UTF-32 00012345

UTF-8 ~~000~~1 0010 0011 0100 0101 17 bits

0 00|01 0010|0011 01|00 0101 21 bits

1111|0 000|10 01|0010|10 00|1101|10 00|0101

F 0 9 2 8 D 8 5

UTF-16

1. 12345
   - 10000
   02345

0|0000 00|10 00|11|0100|0101

0 0 8 3 4 5

2. D800 + 008 = D808

   DC00 + 345 = DF45

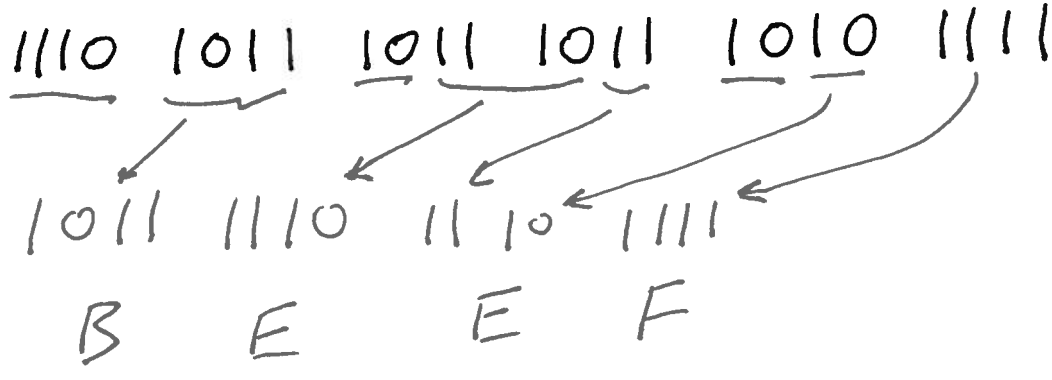
3. D808 DF45

4.

n	D8	08
n+1	08	D8
n+2	DF	45
n+3	45	DF
	↑	↑
	Big Endian	Little Endian

UTF-8: EB BB AF

in hex



UTF-8 : C0 81

in hex

1100 0000 1000 0001

0 0000 000001 Unicode 1 !

this is an error!

1 should have been encoded in 1 UTF-8 byte

namely 01

Overlong encoding is an error!

UTF-8 : D7 B3 ED BA AD 44
in hex

1101 0111 1011 0011 1110 1011 1010 1010 1101 0100 0100

1010
V

1101 1011

1110 1101 1011 1010 1010 1101

self-synchronizing

UTF-16: 01 11 D8 02 DD 19

in hex

Big Endian

0111 | D802 DD19

↓

BMP

D802 − D800 = 0002

DD19 − DC00 = 0119

top 10 bits: 00 0000 0010

bottom 10 bits: 01 0001 1001

00 0000 0010 01 0001 1001

0 0 9 1 9

00919 + 10000 = 10919

BOM :
UTF-16
Big Endian
or Little
Endian?

FF FE ....

If ~~Big~~ Endian, FFFE but that's
not legal character
it's a noncharacter

If Little Endian, FE FF That's the BOM!

So the file is Little Endian.