

II.3. **Biến đổi thập phân thành thập lục phân.**

Để biến đổi các số thập phân thành thập lục phân, ta sử dụng phương pháp chia lặp, với cơ số 16.

Ví dụ : Biến đổi $(1776)_{10}$ thành thập lục phân.

$$1776/16 = 111 + 0 \text{ (LSB).}$$

$$111/16 = 6 + 15 \text{ hoặc F.}$$

$$6/16 = 0 + 6 \text{ (MSB).}$$

$$\text{Số thập lục phân: } (6F0)_{16}.$$

II.4. **Biến đổi thập lục phân thành nhị phân.**

Các số thập lục phân rất dễ đổi thành nhị phân. Thực ra các số thập lục phân cũng chỉ là một cách biểu diễn các số nhị phân thuận lợi hơn mà thôi (bảng 2-1). Để đổi các số thập lục phân thành nhị phân, chỉ cần thay thế một cách đơn giản từng con số thập lục phân bằng bốn bit nhị phân tương đương của nó.

Ví dụ: Đổi số thập lục phân $(DF6)_{16}$ thành nhị phân:

$$\begin{array}{ccc} \text{D} & \text{F} & 6 \\ \downarrow & \downarrow & \downarrow \\ 1101 & 1111 & 0110 \end{array}$$

$$(DF6)_{16} = (110111110110)_2.$$

II.5. **Biến đổi nhị phân thành thập lục phân.**

Để biến đổi một số nhị phân thành số thập lục phân tương đương thì chỉ cần gộp lại thành từng nhóm gồm 4 bit nhị phân, bắt đầu từ dấu chấm nhị phân.

Ví dụ: Biến đổi số nhị phân $(1111101000010000)_2$ thành thập lục phân.

$$\begin{array}{cccc} 1111 & 1010 & 0001 & 0000 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ \text{F} & \text{A} & 1 & 0 \end{array} \text{ Số thập lục phân: } (FA10)_{16}.$$

III. **Hệ BCD (Binary Code decimal).**

Giữa hệ thập phân và hệ nhị phân còn tồn tại một hệ lai: hệ BCD cho các số *hệ thập phân mã hoá bằng hệ nhị phân*, rất thích hợp cho các thiết bị đo có thêm phần hiển thị số ở đầu ra dùng các loại đèn hiển số khác nhau. Ở đây dùng bốn số hệ nhị phân (bốn bit) để mã hoá một số hệ

thập phân có giá trị nằm trong khoảng từ 0..9. Như vậy ở đây ta không dùng hết các tổ hợp có thể có của 4 bit; vì tầm quan trọng của các số BCD nên các bộ vi xử lý thường có các lệnh thao tác với chúng.

Ví dụ: $(35)_{10} = (00110101)_2$.

IV. Bảng mã ASCII.(American Standard Code for Information Interchange).

Người ta đã xây dựng bộ mã để biểu diễn cho các ký tự cũng như các con số và các ký hiệu đặc biệt khác. Các mã đó gọi là **bộ mã ký tự và số**. Bảng mã ASCII là mã 7 bit được dùng phổ biến trong các hệ máy tính hiện nay. Với mã 7 bit nên có $2^7 = 128$ tổ hợp mã. Mỗi ký tự (chữ hoa và chữ thường) cũng như các con số thập phân từ 0..9 và các ký hiệu đặc biệt khác đều được biểu diễn bằng một mã số như bảng 2-2.

Việc biến đổi thành ASCII và các mã ký tự số khác, tốt nhất là sử dụng mã tương đương trong bảng.

Ví dụ: Đổi các ký tự BILL thành mã ASCII:

Ký tự	B	I	L	L
ASCII	1000010	1001001	1001100	1001100
HEXA	42	49	4C	4C

Bảng 2-2: Mã ASCII.

		Column bits(B ₇ B ₆ B ₅)								
Bits(row)		000	001	010	011	100	101	110		
R O W	111									
		0	1	2	3	4	5	6	7	
		↓	↓	↓	↓	↓	↓	↓	↓	
0	0 0 0 0	→	NUL	DLE	SP	0	@	P	\	p
1	0 0 0 1	→	SOH	DC1	!	1	A	Q	a	q
2	0 0 1 0	→	STX	DC2	“	2	B	R	b	r
3	0 0 1 1	→	ETX	DC3	#	3	C	S	c	s
4	0 1 0 0	→	EOT	DC4	\$	4	D	T	d	t
5	0 1 0 1	→	ENQ	NAK	%	5	E	U	e	u
6	0 1 1 0	→	ACK	SYN	&	6	F	V	f	v
7	0 1 1 1	→	BEL	ETB	‘	7	G	W	g	w
8	1 0 0 0	→	BS	CAN	(8	H	X	h	x

9	1	0	0	1	→		HT	EM)	9	I	Y	i	y
A	1	0	1	0	→		LF	SUB	*	:	J	Z	j	z
B	1	0	1	1	→		VT	ESC	+	;	K	[k	{
C	1	1	0	0	→		FF	FS	-	<	L	\	l	
D	1	1	0	1	→		CR	GS	,	=	M]	m	}
E	1	1	1	0	→		SO	RS	.	>	N	^	n	~
F	1	1	1	1	→		SI	US	/	?	O	_	o	DEL

Control characters:

NUL = Null; DLE = Data link escape; SOH = Start Of Heading;

DC1 = Device control 1; DC2 = Device control 2; DC3 = Device control 3.

DC4 = Device control 4; STX = Start of text; ETX = End of text;

EOT = End of transmission; ENQ = Enquiry; NAK = Negative acknowledge.

ACK = Acknowledge; SYN = Synidle; BEL = Bell.

ETB = End of transmission block; BS = Backspace; CAN = Cancel.

HT = Horizontal tab; EM = End of medium; LF = Line feed; SUB = Substitute.

VT = Vertical tab; ESC = Escape; FF = From feed; FS = File separator.

SO = Shift out; RS = Record separator; SI = Shift in; US = Unit separator.

V. Biểu diễn giá trị số trong máy tính.

V.I. Biểu diễn số nguyên.

a. Biểu diễn số nguyên không dấu:

Tất cả các số cũng như các mã ... trong máy vi tính đều được biểu diễn bằng các chữ số nhị phân. Để biểu diễn các số nguyên không dấu, người ta dùng n bit. Tương ứng với độ dài của số bit được sử dụng, ta có các khoảng giá trị xác định như sau:

Số bit	Khoảng giá trị
n bit:	0.. $2^n - 1$
8 bit	0.. 255 Byte
16 bit	0.. 65535 Word

b. Biểu diễn số nguyên có dấu:

Người ta sử dụng bit cao nhất biểu diễn dấu; bit dấu có giá trị 0 tương ứng với số nguyên dương, bit dấu có giá trị 1 biểu diễn số âm. Như vậy khoảng giá trị số được biểu diễn sẽ được tính như sau:

Số bit	Khoảng giá trị:
n bit	$2^{n-1}-1$
8 bit	-128.. 127 Short integer
16 bit	-32768.. 32767 Integer
32 bit	$-2^{31}.. 2^{31}-1$ (-2147483648.. 2147483647) Long integer

V.2. Biểu diễn số thực (số có dấu chấm (phẩy) động).

Có hai cách biểu diễn số thực trong một hệ nhị phân: số có dấu chấm cố định (fixed point number) và số có dấu chấm động (floating point number). Cách thứ nhất được dùng trong những bộ VXL (micro processor) hay những bộ vi điều khiển (micro controller) cũ. Cách thứ 2 hay được dùng hiện nay có độ chính xác cao. Đối với cách biểu diễn số thực dấu chấm động có khả năng hiệu chỉnh theo giá trị của số thực. Cách biểu diễn chung cho mọi hệ đếm như sau:

$$R = m.B^e$$

Trong đó m là phần định trị, trong hệ thập phân giá trị tuyệt đối của nó phải luôn nhỏ hơn 1. Số e là phần mũ và B là cơ số của hệ đếm.

Có hai chuẩn định dạng dấu chấm động quan trọng là: chuẩn MSBIN của Microsoft và chuẩn IEEE. Cả hai chuẩn này đều dùng hệ đếm nhị phân.

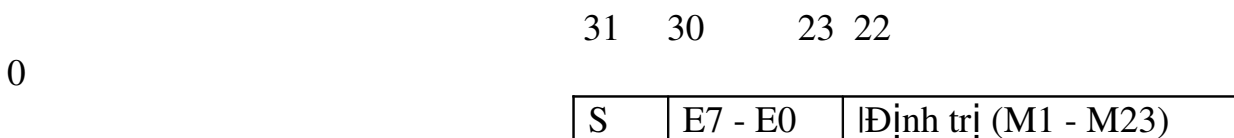
Thường dùng là theo tiêu chuẩn biểu diễn số thực của IEEE 754-1985 (Institute of Electric & Electronic Engineers), là chuẩn được mọi hãng chấp nhận và được dùng trong bộ xử lý toán học của Intel. Bit dấu nằm tại vị trí cao nhất; kích thước phần mũ và khuôn dạng phần định trị thay đổi theo từng loại số thực.

Giá trị số thực IEEE được tính như sau:

$$R = (-1)^S * (1 + M_1 * 2^{-1} + \dots + M_n * 2^{-n}) * 2^{E_7 \dots E_0 - 127}$$

Chú ý: giá trị đầu tiên M_0 luôn mặc định là 1.

- Dùng 32 bit để biểu diễn số thực, được số thực ngắn: $-3,4.10^{38} < R < 3,4.10^{38}$



- Dùng 64 bit để biểu diễn số thực, được số thực dài: $-1,7.10^{308} < R < 1,7.10^{308}$

0

63 62 52 51

S	E10 - E0	Định trị (M1 - M52)
---	----------	---------------------

Ví dụ tính số thực:

0100 0010 1000 1100 1110 1001 1111 1100

	<p>Phần định trị: $2^{-4}+2^{-5}+2^{-8}+2^{-9}+2^{-10}+2^{-12}+2^{-15}+$ $+2^{-16}+2^{-17}+2^{-18}+2^{-19}+2^{-20}+2^{-21} = 0,1008906.$ Giá trị ngầm định là: 1,1008906.</p> <p>Phần mũ: $2^8+2^2+2^0 = 133$ Giá trị thực (bit cao nhất là bit dấu): $133-128=6.$</p> <p>Dấu: 0 = số dương</p>
--	--

Giá trị số thực là: $R = 1,1008906.2^6 = 70,457.$

Phương pháp đổi số thực sang số dấu phẩy động 32 bit:

- Đổi số thập phân thành số nhị phân.
- Biểu diễn số nhị phân dưới dạng $1, xxxBy$ (B: cơ số 2).
- Bit cao nhất 31: lấy giá trị 0 với số dương, 1 với số âm.
- Phần mũ y đổi sang mã excess -127 của y, được xác định bằng cách: $y + (7F)_{16}.$
- Phần xxx là phần định trị, được đưa vào từ bit 22..0.

Ví dụ: Biểu diễn số thực $(9,75)_{10}$ dưới dạng dấu phẩy động.

Ta đổi sang dạng nhị phân: $(9,75)_{10} = (1001.11)_2 = 1,00111B3.$

Bit dấu: bit 31 = 0.

Mã excess - 127 của 3 là: $7F + 3 = (82)_{16} = 82H = (10000010)_2.$ Được đưa vào các bit tiếp theo: từ bit 30 đến bit 23.

Bit 22 luôn mặc định là 0.

Cuối cùng số thực $(9,75)_{10}$ được biểu diễn dưới dạng dấu phẩy động 32 bit như sau:

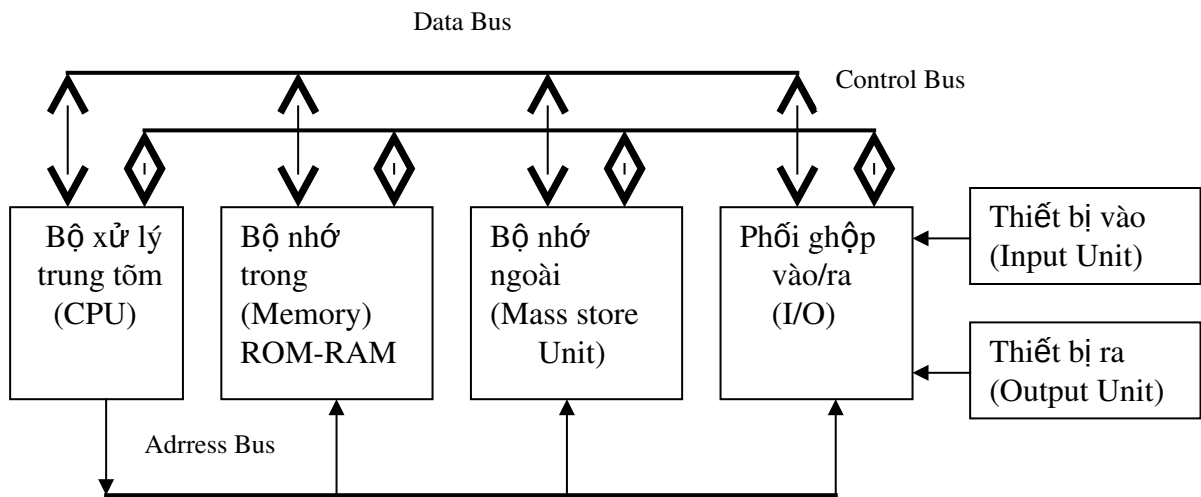
$\underline{0100\ 0001\ 0001\ 1100\ 0000\ 0000\ 0000\ 0000}$
 bit 31|30 23|22 0|1

CHƯƠNG III. CÁC KHỐI CƠ BẢN CỦA MÁY TÍNH

I. Giới thiệu sơ lược cấu trúc của máy vi tính.

So với từ khi ra đời, cấu trúc cơ sở của các máy vi tính ngày nay không thay đổi mấy. Mọi máy tính số đều có thể coi như được hình thành từ sáu phần chính (như hình 3-1):

Hình 3-1: Giới thiệu sơ đồ khối tổng quát của máy tính số



Trong sơ đồ này, các khối chức năng chính của máy tính số gồm:

- Khối xử lý trung tâm (central processing unit, CPU),
- Bộ nhớ trong (memory), như RAM, ROM
- Bộ nhớ ngoài, như các loại ổ đĩa, băng từ
- Khối phối ghép với các thiết bị ngoại vi (vào/ra)
- Các bộ phận đầu vào, như bàn phím, chuột, máy quét
- Các bộ phận đầu ra, như màn hình, máy in

Bốn khối chức năng đầu liên hệ với nhau thông qua tập các đường dây để truyền tín hiệu, gọi chung là *bus hệ thống*. Bus hệ thống bao gồm 3 bus thành phần; ứng với các tín hiệu xác lập địa chỉ từ CPU đến các đơn vị thành phần ta có bus địa chỉ; với các dữ liệu được liên hệ giữa các khối

qua bus dữ liệu (data bus); các tín hiệu điều khiển bao gồm các lệnh, các đáp ứng, các trạng thái của các khối được xác lập qua bus điều khiển.

Sự khác biệt quan trọng nhất của các hệ máy tính là kích thước và tốc độ, các máy tính nhỏ hơn và nhanh, mạnh hơn theo từng năm. Sự phát triển không ngừng của các thế hệ máy tính nhờ vào hai yếu tố quan trọng, đó là sự phát triển của công nghệ chế tạo IC và công nghệ chế tạo bộ nhớ.