

سیستم های عدد نویسی

محاسبات کامپیوتری در مبنای دو انجام می شود. به طور معمول از سیستم عددی هگزادسیمال برای نمایش اعداد باینری استفاده می شود.

[سیستم های عدد نویسی](#)

[سیستم عددی اعشاری](#)

[سیستم عددی دودویی](#)

[سیستم عددی هگز](#)

سیستم های عدد نویسی

در کارهای روزمره از سیستم عددی اعشاری یا مبنای ۱۰ استفاده می شود. این سیستم برای کامپیوتر مناسب نیست و برای سادگی سخت افزار، کلیه اطلاعات به شکل بیت های روشن و خاموش رمز می شوند. بنابراین سیستم عددی باینری که تنها شامل ارقام صفر و یک است برای این منظور بسیار مناسب است. عدد ۱ (on) مشخص کننده +5 ولت و عدد صفر (off) مشخص کننده 0.5 ولت است.

برای تعیین مبنای عدد یک حرف کوچک در انتهای آن قرار می گیرد. مثال 45h به معنی عدد 45 در مبنای شانزده است. و 11010011b یعنی این عدد در مبنای ۲ است. این روشی است که اسمبلر اعداد را در برنامه های اسمبلی تشخیص می دهد.

سیستم عددی اعشاری (Decimal)

اعداد اعشاری یا مبنای ۱۰ از ۱۰ رقم (۰ تا ۹) تشکیل شده اند. هر رقم به توانی از ۱۰ مرتبط است که نشان دهنده ارزش مکانی رقم در عدد است.

$$234 = 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0 \\ = 200 + 30 + 4$$

سیستم عددی دودویی (binary)

سیستم باینری بر اساس تنها دو وضعیت است: روشن (1) یا خاموش (0)، بنابراین در مبنای ۲ است. یک رقم باینری یک بیت نامیده می شود (در واقع کلمه Bit مخفف Binary Digit است).

تبدیل باینری به اعشاری

مقدار یک عدد باینری بر اساس بیت های ۱ و ارزش مکانی آنها بستگی می آید. ارزش مکانی هر بیت توانی از ۲ است. برای محاسبه مقدار اعشاری یک عدد باینری، کافی است هر رقم از راست به چپ در ارزش مکانی اش ضرب شده سپس کلیه اعداد با هم جمع شوند.

مثال ۱. تبدیل عدد 11001 به مبنای ۱۰.

$$\text{Binary: } 11001 \\ \text{Decimal: } 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ = 16 + 8 + 0 + 0 + 1 \\ = 25$$

مثال ۲. تبدیل عدد باینری 10010000 به مبنای ۱۰.

$$\text{Binary: } 10010000 \\ \text{Decimal: } 1 \times 2^7 + 0 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 \\ = 128 + 0 + 0 + 16 + 0 + 0 + 0 + 0 \\ = 144$$

کاراکتر ^ نشان دهنده عمل توان است.

تبدیل اعشاری به باینری

چندین روش برای تبدیل اعداد اعشاری به باینری وجود دارد. یک روش متداول تقسیم های متوالی بر ۲ است. به این ترتیب که عدد اعشاری بر ۲ تقسیم می شود، باقیمانده بعنوان رقم باینری نگهداشته و خارج قسمت مجدد بر ۲ تقسیم می شود این عمل تا زمانی که خارج قسمت صفر شود ادامه پیدا می کند.

مثال. تبدیل عدد 43 به مبنای ۲

عدد	خارج قسمت	باقیمانده
$43 \div 2$	21	1
$21 \div 2$	10	1
$10 \div 2$	5	0
$5 \div 2$	2	1
$2 \div 2$	1	0
$1 \div 2$	0	1

با قرار دادن باقیمانده های تقسیم از پایین به بالا عدد باینری 101011 بدست می آید.

جمع اعداد باینری

جمع باینری ساده به صورت زیر محاسبه می شود:

$$\begin{aligned} 0 + 0 &= 0 \\ 0 + 1 &= 1 \\ 1 + 1 &= 10 \\ 1 + 1 + 1 &= 11 \end{aligned}$$

برای جمع و عدد باینری کافی است بیت به بیت از سمت راست به چپ عمل جمع انجام شود. رقم نقلی حاصل از هر ستون در جمع ستون بعدی اعمال می شود.

$$\begin{array}{rcccccc} & & & 1 & 1 & & \\ + & 1 & 1 & 0 & 1 & 1 & \\ \hline & 1 & 1 & 1 & 0 & 0 & \end{array}$$

سیستم عددی هگز (Hexadecimal)

هگزادسیمال (یا به طور خلاصه هگز) روش فشرده تری را برای نمایش اعداد باینری ارائه می دهد به همین دلیل توسط اسمبلر و دیباگر برای مختصر نوشتن اعداد باینری بکار می رود.

اعداد هگز مبنای ۱۶ را استفاده می کنند و از ۱۶ رقم (0-15) تشکیل شده اند. برای نمایش ارقام بعد از ۹ از حروف A تا F استفاده می شود. به عبارت دیگر ۱۶ رقم هگز شامل ۰، ۱، ۲، ۳، ۴، ۵، ۶، ۷، ۸، ۹، A، B، C، D، E، F است که حروف A-F ارقام ۱۰-۱۵ را نشان می دهند (A=10، B=11، C=12، D=13، E=14 و F=15).

هر رقم هگز معادل چهار بیت باینری است. یک عدد چهار بیتی یک نیبل (Nibble) نام دارد. پس هر رقم هگز معادل یک نیبل است. دو نیبل یک بایت (Byte) را می سازد بنابراین هر بایت می تواند دو رقم هگز را نشان بدهد. مقدار یک بایت می تواند از 00000000 تا 11111111 باینری، 00 تا FF در هگز و 0 تا 255 در دسیمال باشد.

تبدیل هگز به اعشاری

ارزش هر رقم هگز با توانی از ۱۶ مشخص می شود. برای تبدیل اعداد از مبنای ۱۶ به ۱۰ هر رقم عدد در ارزش مکانی اش ضرب می شود.

مثال. تبدیل عدد 3BA4h به مبنای ۱۰.

Hex : 3BA4

Decimal: $3 \times 16^3 + 11 \times 16^2 + 10 \times 16^1 + 4 \times 16^0$

$$= 3 \times 4096 + 11 \times 256 + 10 \times 16 + 4 \times 1$$

$$= 15268$$

تبدیل اعشاری به هگز

برای تبدیل دسیمال به هگز مانند باینری تقسیم های متوالی بر ۱۶ انجام می شود.

مثال. تبدیل عدد 589 به هگز

عدد	خارج قسمت	باقیمانده
$589 \div 16$	36	13
$36 \div 16$	2	4
$2 \div 16$	0	2

با قرار دادن باقیمانده های تقسیم از پایین به بالا عدد باینری 24D بدست می آید.

تبدیل هگز به باینری

تبدیل هگز به باینری ساده است. کافی است هر رقم هگز به یک عدد چهار رقمی باینری تبدیل شود.

مثال. تبدیل عدد 160794h به باینری.

Hex: 1 6 0 7 9 4

Binary: 0001 0110 0000 0111 1001 0100

توجه کنید که صفرهای ابتدای چهار بیت اهمیت دارند. اگر این صفرها برای ارقام میانی قرار نگیرند حاصل اشتباه است.

تبدیل باینری به هگز

تبدیل از باینری به هگز هم ساده است. ابتدا عدد باینری از راست به چپ به گروه های چهاربیتی تقسیم شده (اگر آخرین گروه سمت چپ کمتر از چهار بیت بود صفر اضافه می شود)، سپس هر بخش به یک رقم هگز تبدیل می شود.

مثال. تبدیل عدد 0001 0110 0000 0111 1001 0100b به هگز

Binary: 0001 0110 0000 0111 1001 0100

Hex: 1 6 0 7 9 4

= 160794h

جمع اعداد در هگزادسیمال

چند جمع ساده در مبنای ۱۶.

$$7 + 3 = A$$

$$6 + 7 = D$$

$$F + 1 = 10$$

$$10 + 30 = 40$$

$$F + F = 1E$$

$$38 + 18 = 50$$

$$FF + 1 = 100$$

جمع دو عدد هگز

	1		1	
	7	E	C	6
+	3	4	0	A
	B	2	D	0

$$6 + A = 6 + 10 = 16 \Rightarrow 10h$$

$$C + 0 + 1 = 12 + 0 + 1 = 13 \Rightarrow Dh$$

$$E + 4 = 14 + 4 = 18 \Rightarrow 12h$$

$$7 + 3 + 1 = 11 \Rightarrow Bh$$