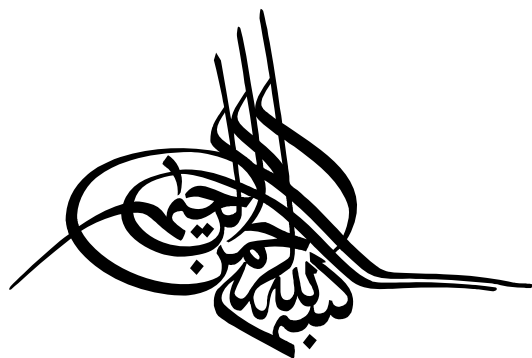




مدار منطقی

سری کتاب‌های کمک آموزشی کارشناسی ارشد
فنی مهندسی

مجموعه مهندسی کامپیوتر و مهندسی برق

مؤلف: گروه مؤلفین ماهان

سرشناسه	: گروه مؤلفین ماهان
عنوان	: مدار منطقی
مشخصات نشر	: تهران : مشاوران صعود ماهان، ۱۴۰۴
مشخصات ظاهری	: ۱۹۲ ص
فروست	: سری کتاب‌های کمک آموزشی کارشناسی ارشد
شابک	: ۹۷۸-۶۰۰-۴۵۸-۷۹۶-۹
وضعیت فهرست نویسی	: فیپای مختصر
یادداشت	: این مدرک در آدرس http://opac.njai.ir قابل دسترسی است.
شماره کتابشناسی ملی	:



نام کتاب: مدار منطقی

مؤلف: گروه مؤلفین ماهان

مدیر تولید: سمیه بیگی

ناشر: مشاوران صعود ماهان

نوبت و تاریخ چاپ: اول / ۱۴۰۴

تیراژ: ۱۰۰۰ نسخه

قیمت: ۳/۵۰۰/۰۰۰ ریال

شابک: ISBN: ۹۷۸-۶۰۰-۴۵۸-۷۹۶-۹

انتشارات مشاوران صعود ماهان: خیابان ولیعصر، بالاتر از تقاطع مطهری،
 روبروی قنادی هتل بزرگ تهران، جنب بانک ملی، پلاک ۲۰۵۰
 تلفن: ۸۸۱۰۰۱۱۳-۴

سخن ناشر

«ن والقلم و ما یسطرون»

کلمه نزد خدا بود و خدا آن را با قلم بر ما نازل کرد.

به پاس تشکر از چنین موهبت الهی، موسسه ماهان در صدد برآمده است تا در راستای انتقال دانش و مفاهیم با کمک اساتید مجرب و مجموعه کتب آموزشی خود برای شما داوطلبان ادامه تحصیل در مقطع کارشناسی ارشد گام موثری بردارد. امید است تلاش‌های خدمتگزاران شما در این موسسه پایه‌گذار گام‌های بلند فردای شما باشد.

مجموعه کتاب‌های کمک آموزشی ماهان به‌منظور استفاده داوطلبان کنکور کارشناسی ارشد سراسری و آزاد تالیف شده‌اند. در این کتاب‌ها سعی کرده‌ایم با بهره‌گیری از تجربه اساتید بزرگ و کتب معتبر داوطلبان را از مطالعه کتاب‌های متعدد در هر درس بی‌نیاز کنیم.

بدین‌وسیله از مجموعه اساتید، مولفان و همکاران محترم خانواده بزرگ ماهان که در تولید و به‌روزرسانی تالیفات ماهان نقش موثری داشته‌اند، صمیمانه تقدیر و تشکر می‌نماییم. دانشجویان عزیز و اساتید محترم می‌توانند هرگونه انتقاد و پیشنهاد درخصوص تالیفات ماهان را از طریق سایت ماهان به آدرس mahan.ac.ir با ما در میان بگذارند.

موسسه آموزش عالی آزاد ماهان

فهرست مطالب

صفحه

عنوان

۵	فصل اول - اعداد
۳۰	فصل دوم - جبر بول
۵۶	فصل سوم - خواص گیت‌ها و کامل بودن، مدارات دو سطحی
۷۲	فصل چهارم - مخاطره
۷۹	فصل پنجم - مدارات ترکیبی
۱۰۷	فصل ششم - رمزگشا و رمزگذار و مالتی پلکسر و دی مالتی پلکسر
۱۲۴	فصل هفتم - قطعات برنامه‌ریز
۱۲۹	فصل هشتم - لچ و فلیپ فلاپ
۱۴۰	فصل نهم - تحلیل مدارات ترتیبی سنکرون، میلی و مور
۱۴۸	فصل دهم - طراحی مدارات ترتیبی سنکرون
۱۶۹	فصل یازدهم - شمارنده و ثبات
۱۸۰	فصل دوازدهم - پارامترهای زمانی فلیپ فلاپ‌ها
۱۸۳	فصل سیزدهم - وریلاگ
۱۸۶	سوالات کنکور سراسری سال ۱۴۰۳

فصل اول

اعداد

اعداد

۱- مبناها

آدم‌ها با مبنای 10 سر و کار دارند که به خاطر تعداد انگشتان دو دست است. در مبنای ده (decimal) از ده نشانه یا رقم (digit) برای نمایش اعداد استفاده می‌شود. این نشانه‌ها به انگلیسی 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 می‌باشند. کامپیوترها به خاطر استفاده از ترانزیستور که دو حالت روشن و خاموش یا باز و بسته دارد از مبنای دو (binary) استفاده می‌کنند. در مبنای 2 از ارقام 0 و 1 که بیت (binary digit: bit) نام دارند استفاده می‌شود. به طور کلی در مبنای r از ارقام $0, 1, \dots, r-1$ استفاده می‌کنیم. مثلاً در مبنای 8 (octal) از ارقام 0, 1, 2, 3, 4, 5, 6, 7 استفاده می‌شود و در مبنای 16 (hexadecimal) از ارقام 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F استفاده می‌شود که A تا F به ترتیب 10 تا 15 هستند. می‌توان اعداد را از یک مبنای دیگر به مبنای 10 و بالعکس تبدیل کرد که در ادامه بررسی می‌کنیم.

۲- تبدیل به مبنای 10

برای تبدیل یک عدد از مبنای r به مبنای 10 (یا به عبارتی یافتن ارزش یا value) به هر رقم یک ارزش نسبت می‌دهیم و سپس هر رقم را در ارزش آن ضرب می‌کنیم و مقادیر به دست آمده را جمع می‌کنیم. شکل کلی عدد a در مبنای r به صورت $a = (a_{n-1}a_{n-2}\dots a_1a_0.a_{-1}a_{-2}\dots a_{-m})_r$ می‌باشد که $a_0, a_1, \dots, a_{n-1}$ ارقام صحیح و $a_{-1}, a_{-2}, \dots, a_{-m}$ ارقام اعشار هستند نقطه (point) نیز ممیز است که در فارسی با / نیز نشان می‌دهیم. ارزش رقم a_i برابر r^i می‌باشد مثلاً ارزش a_0 برابر r^0 یا 1 می‌باشد یا ارزش رقم a_{-1} برابر r^{-1} یا $\frac{1}{r}$ می‌باشد.

پس ارزش a یا عدد a در مبنای 10 برابر است با:

$$\text{val}(a) = (a)_{10} = a_{n-1} \times r^{n-1} + a_{n-2} \times r^{n-2} + \dots + a_0 \times r^0 + a_{-1} \times r^{-1} + \dots + a_{-m} \times r^{-m}$$

که می‌توان به صورت بسته سیگما نشان داد:

$$\text{val}(a) = \sum_{k=-m}^{n-1} a_k (r)^k$$

به تبدیل مبناهای زیر توجه کنید:

$$(2AF.4)_{16} = (2AF.4)_{\text{hex}} = 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 + 4 \times 16^{-1} = (687.25)_{10}$$

$$(BAG)_{20} = 11 \times 20^2 + 10 \times 20^1 + 16 \times 20^0 = (4616)_{10}$$

$$\begin{aligned} (1011001.101)_2 &= 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 2^6 + 2^4 + 2^3 + 2^0 + \frac{1}{2} + \frac{1}{8} = (89.625)_{10} \end{aligned}$$

مثال: بزرگترین عدد صحیح n رقمی در مبنای r چه شکلی است و چه مقداری دارد یعنی در مبنای 10 چه مقداری دارد؟

حل: بزرگترین عدد شامل n رقم $r-1$ است. مثلاً در مبنای 10 با 3 رقم بزرگترین عدد $999 = 10^3 - 1$ است یا در مبنای 2 بزرگترین عدد با 3 رقم $111_2 = 2^3 - 1$ می باشد. بزرگترین عدد n رقمی در مبنای r به شکل $\underbrace{((r-1)(r-1)\dots(r-1))}_n$ می باشد که مقدار این عدد برابر است با:

$$(r-1) \times r^{n-1} + (r-1) \times r^{n-2} + \dots + (r-1) \times r^0 = (r-1) \left[\underbrace{r^{n-1} + r^{n-2} + \dots + r^0}_{\text{تصادد هندسی}} \right]$$

$$= (r-1) \times \frac{r^n - 1}{r - 1} = r^n - 1$$

مثلاً بزرگترین عدد 4 رقمی مبنای 2 به شکل 1111_2 می باشد و برابر $2^4 - 1 = 15$ می باشد.

توجه

تصادد هندسی یا سری هندسی جمع تعدادی عدد است که هر عدد برابر عدد قبلی ضرب در یک مقدار ثابت به اسم قدر نسبت است. تصاعد هندسی با جمله اول a ، قدر نسبت x و تعداد n جمله به

شکل $a + ax + ax^2 + \dots + ax^{n-1}$ می باشد و برابر با $\frac{a(x^n - 1)}{x - 1}$ است.

حال اگر $|x| < 1$ و تعداد جملات یعنی n بی نهایت باشد آنگاه x^n به صفر میل می کند و حاصل سری هندسی

نامتناهی $a + ax + ax^2 + \dots$ برابر $\frac{a}{1-x}$ می شود.

$$1 + \frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \dots = \frac{1}{1 - \frac{1}{2}} = 2$$

مثال: بزرگترین اعشار n رقمی در مبنای r چه شکلی است و در مبنای 10 چه می شود؟

حل: این عدد شامل n رقم $r-1$ در سمت راست ممیز است یعنی به شکل $\underbrace{(0.(r-1)\dots(r-1))}_n$ می باشد که مقدار این

عدد $1 - r^{-n}$ است که برای اثبات، هر رقم را در ارزش آن ضرب کرده و مقادیر به دست آمده را جمع می کنیم:

$$(r-1) \times r^{-1} + (r-1) \times r^{-2} + \dots + (r-1) \times r^{-n} = (r-1) [r^{-1} + r^{-2} + \dots + r^{-n}]$$

$$= (r-1) \times \frac{r^{-1}(1 - r^{-n})}{1 - r^{-1}} = 1 - r^{-n}$$

۳- تبدیل از مبنای 10 به سایر مبناها

برای تبدیل از مبنای 10 به مبنای r ، قسمت صحیح عدد را به r تقسیم می کنیم و خارج قسمت را مجدد به r تقسیم می کنیم و به همین ترتیب خارج قسمت به دست آمده را به r تقسیم می کنیم تا در نهایت خارج قسمت صفر شود و سپس باقی مانده ها، ارقام صحیح حاصل را تشکیل می دهند طوری که آخرین باقی مانده، با ارزش ترین رقم حاصل است. قسمت اعشار عدد را در r ضرب می کنیم و رقم صحیح حاصل ضرب، با ارزش ترین رقم اعشار عدد حاصل می باشد. حال قسمت اعشار حاصل ضرب را مجدد در r ضرب می کنیم و رقم صحیح حاصل ضرب، دومین رقم اعشار عدد حاصل می باشد و به همین ترتیب قسمت اعشار را در r ضرب می کنیم و قسمت صحیح حاصل ضرب را برمی داریم تا وقتی که قسمت اعشار صفر شود یا قسمت اعشار به تکرار برسد که با مثال نشان می دهیم.

مثال: عدد $(309.2)_{10}$ را به مبنای 8 تبدیل کنید.

حل: قسمت صحیح یعنی 309 را به 8 تقسیم کرده، خارج قسمت‌ها را نیز به 8 تقسیم می‌کنیم تا اینکه خارج قسمت صفر شود.

$$\begin{array}{r} 309 \quad 8 \\ 24 \quad 38 \quad 8 \\ 69 \quad 32 \quad 4 \quad 8 \\ 64 \quad 6 \quad 0 \quad 0 \\ 5 \quad 4 \end{array}$$

پس قسمت صحیح حاصل $(465)_8$ می‌باشد (برای آنکه مطمئن شوید که درست است $(465)_8$ را به مبنای 10 تبدیل کنید که $4 \times 8^2 + 6 \times 8 + 5 = 309$ می‌شود).

حال قسمت اعشار یعنی 0.2 را در 8 ضرب کرده و قسمت اعشار حاصل را مجدد در 8 ضرب می‌کنیم تا وقتی که اعشار صفر شود یا تکراری شود:

$$\begin{array}{l} 0.2 \times 8 = 1.6 \\ 0.6 \times 8 = 4.8 \\ 0.8 \times 8 = 6.4 \\ 0.4 \times 8 = 3.2 \\ 0.2 \times 8 \end{array}$$

تکرار

پس $(0.2)_{10} = (0.14631463...)_{10}$ یعنی 1463 در قسمت اعشار تا بی‌نهایت تکرار می‌شود که به آن متناوب می‌گویند و گاهی برای نشان دادن تناوب، روی قسمت متناوب خط می‌کشیم:

$$(309.2)_{10} = (465.\overline{1463})_8$$

مثال: عدد $(301.375)_{10}$ را به مبنای 2 تقسیم کنید.

حل: باید 301 را به 2 تقسیم کرده، خارج قسمت‌ها را مجدد به 2 تقسیم کنیم تا خارج قسمت صفر شود سپس باقی‌مانده‌ها که 0 یا 1 هستند را یادداشت می‌کنیم، ولی این عمل زمانبر است. برای تبدیل قسمت صحیح به مبنای 2، بهتر است اعداد به شکل 2^i یعنی:

$$2^{10} = 1024, 2^9 = 512, 2^8 = 256, 2^7 = 128, 2^6 = 64, 2^5 = 32, 2^4 = 16, 2^3 = 8, 2^2 = 4, 2^1 = 2, 2^0 = 1$$

$$2^{16} = 65536, 2^{15} = 32768, 2^{14} = 16384, 2^{13} = 8192, 2^{12} = 4096, 2^{11} = 2048$$

و ... را حفظ باشید و هر عدد صحیحی را می‌توان به صورت مجموع برخی از اعداد 2^i نوشت و اگر چنین کاری کنید در واقع به مبنای 2 تبدیل کرده‌اید.

$$\text{پس } 301 = 256 + 32 + 8 + 4 + 1 \text{ یا } 301 = 2^8 + 2^5 + 2^3 + 2^2 + 2^0 \text{ بنابراین:}$$

$$301 = (100101101)_2$$

در واقع در مکان‌های $2^8, 2^5, 2^3, 2^2, 2^0$ ، یک قرار دادیم و در مکان‌های $2^7, 2^6, 2^4, 2^1$ ، صفر قرار دادیم. برای درک درستی این موضوع می‌توانید برعکس عمل کنید و $(100101101)_2$ را به مبنای 10 تبدیل کنید که $2^8 + 2^5 + 2^3 + 2^2 + 2^0 = 301$ می‌شود.

برای تبدیل قسمت اعشار، عملیات ضرب در 2 را انجام می‌دهیم:

$$0.375 \times 2 = 0.75 \Rightarrow a_{-1} = 0$$

$$0.75 \times 2 = 1.5 \Rightarrow a_{-2} = 1$$

$$0.5 \times 2 = 1.0 \Rightarrow a_{-3} = 1$$

پس $(0.375)_{10} = (0.011)_2$ که قسمت اعشار مختوم یا خاتمه‌پذیر شد، بنابراین:

$$(301.375)_{10} = (100101101.011)_2$$

مثال: کدامیک از اعداد $(0.1)_{10}$ و $(0.15)_{10}$ و $(0.775)_{10}$ و $(0.875)_{10}$ در مبنای 2 مختوم است؟

حل: باید قسمت اعشار را در 2 ضرب کنیم که صفر شود. فقط 5 ضرب در 2 صفر می شود که به این معنی است که حتماً باید رقم سمت راست اعشار 5 باشد بنابراین 0.1 متناوب می باشد:

$$0.1 \times 2 = 0.2, 0.2 \times 2 = 0.4, 0.4 \times 2 = 0.8, 0.8 \times 2 = 1.6, 0.6 \times 2 = 1.2, 0.2 \times 2 = \text{repeat}$$

0.15 را اگر در 2 ضرب کنیم 0.3 می شود که حتماً مختوم نیست چون سمت راست آن 5 نیست.

0.775 ضرب در 2 می شود 1.55 و $0.55 \times 2 = 1.1$ پس مختوم نیست. اما 0.875 مختوم است زیرا:

$$0.875 \times 2 = 1.75, 0.75 \times 2 = 1.5, 0.5 \times 2 = 1.0$$

نکته

برای آنکه عدد از مبنای 10 به مبنای 2, 4, 8, 16 مختوم باشد باید سمت راست ترین رقم آن 5 باشد و در هر مرحله نیز سمت راست آن 5 باشد.

مثال: عدد کسری $(\frac{23}{6})_{10}$ را به مبنای 2 تبدیل کنید.

حل: ابتدا قسمت صحیح و اعشار عدد را تفکیک می کنیم یعنی عدد داده شده را به فرم مخلوط تبدیل می کنیم که $\frac{23}{6} = 3 + \frac{5}{6} = 3\frac{5}{6}$ پس 3 قسمت صحیح عدد و $\frac{5}{6}$ که از یک کوچکتر است قسمت اعشار عدد می باشد، حال مثل قبل عمل می کنیم که $3 = (11)_2$ و $\frac{5}{6}$ را نیز در 2 ضرب می کنیم:

$$\begin{aligned} \frac{5}{6} \times 2 &= \frac{5}{3} = 1\frac{2}{3} \Rightarrow a_{-1} = 1 \\ \frac{2}{3} \times 2 &= \frac{4}{3} = 1\frac{1}{3} \Rightarrow a_{-2} = 1 \\ \frac{1}{3} \times 2 &= \frac{2}{3} \Rightarrow a_{-3} = 0 \\ &\vdots \end{aligned}$$

تکرار

پس $\frac{5}{6} = (0.11\overline{0})_2$ یعنی اولین رقم اعشار تکرار نمی شود ولی دو رقم بعدی تکرار می شوند که اصطلاحاً به چنین حالتی

متناوب مرکب گویند. در نتیجه: $\frac{23}{6} = (11.11\overline{0})_2$

۴- تبدیل مبنای خاص

برای آنکه عددی از مبنای مثلاً 3 به مبنای 6 تبدیل شود باید ابتدا عدد را به مبنای 10 تبدیل کنیم سپس نتیجه را به مبنای 6 تبدیل کنیم، ولی برخی مبنایها را می توان مستقیماً به هم تبدیل کرد.

مثلاً مبنای 2 به 8 و بالعکس. به طور کلی مبنایهای r و r^i را می توان مستقیماً به هم تبدیل کرد و نیازی به مبنای واسطه ندارد.

مثلاً مبنایهای 2 و $2^2 = 4$ ، مبنایهای 2 و $2^3 = 8$ ، مبنایهای 2 و $2^4 = 16$ ، مبنایهای 4 و $4^2 = 16$ ، مبنایهای 3 و $3^2 = 9$ را می توان مستقیماً به هم تبدیل کرد.

برای تبدیل از مبنای 8 به مبنای 2، هر رقم در مبنای 8 را به 3 رقم در مبنای 2 تبدیل می کنیم.

جداول زیر تبدیل ارقام مبنایهای معروف به یکدیگر را نشان می دهند:

مبنای 8	0	1	2	3	4	5	6	7
مبنای 2	000	001	010	011	100	101	110	111

مبنای ۱۶	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
مبنای ۲	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
مبنای ۴	00	01	02	03	10	11	12	13	20	21	22	23	30	31	32	33

مبنای ۹	0	1	2	3	4	5	6	7	8
مبنای ۳	00	01	02	10	11	12	20	21	22

مبنای ۴	0	1	2	3
مبنای ۲	00	01	10	11

به عنوان مثال به تبدیل مبناهای زیر توجه کنید:

$$(273.45)_8 = (010111011.100101)_2$$

$$(AB4.F8)_{16} = (1010\ 1011\ 0100.11111000)_2 = (22\ 23\ 10.33\ 20)_4$$

$$(4805)_9 = (11\ 22\ 00\ 12)_3$$

برای تبدیل یک عدد از مبنای r به r^i باید ارقام را به دسته‌های i تایی تقسیم کنیم و به ازای هر دسته، رقم معادل آن را در مبنای r^i قرار دهیم. برای این منظور، قسمت صحیح را از سمت راست و قسمت اعشار را از سمت چپ به دسته‌های i تایی تفکیک می‌کنیم و اگر آخرین دسته از i رقم کمتر داشت، صفر قرار می‌دهیم:

$$\leftarrow \rightarrow \quad (1101.1101)_2 \xrightarrow[\text{به مبنای ۸}]{\quad} (001\ 101.\ 110\ 100)_2 = (15.64)_8$$

$$(123.321)_4 \xrightarrow[\text{به مبنای ۱۶}]{\quad} (01\ 23.\ 32\ 10)_4 = (1B.E4)_{16}$$

$$(21012.2)_3 \xrightarrow[\text{به مبنای ۹}]{\quad} (02\ 10\ 12.\ 20)_3 = (235.6)_9$$

خب دوستان، امیدوارم تاکنون تبدیل مبنا را به درستی آموخته باشید. در ادامه چند مثال از تبدیل مبناهای خاص و نکاتی مربوط به مبناهای را بررسی می‌کنیم.

مثال: عدد متناوب $(0.\overline{34})_5$ را به مبنای ده تبدیل کنید.

حل: روش اول: هر رقم دارای ارزش است، ارقام را در ارزش آنها ضرب می‌کنیم و نتایج را جمع می‌کنیم:

$$(0.343434\ldots)_5 = 3 \times 5^{-1} + 4 \times 5^{-2} + 3 \times 5^{-3} + 4 \times 5^{-4} + 3 \times 5^{-5} + 4 \times 5^{-6} + \ldots$$

$$= 3(5^{-1} + 5^{-3} + 5^{-5} + \ldots) + 4(5^{-2} + 5^{-4} + 5^{-6} + \ldots) = 3\left(\frac{5^{-1}}{1-5^{-2}}\right) + 4\left(\frac{5^{-2}}{1-5^{-2}}\right)$$

$$= \frac{15}{24} + \frac{4}{24} = \frac{19}{24}$$

روش دوم: اگر عددی در مبنای r را در r^k ضرب کنیم آن عدد k واحد به سمت چپ شیفت داده می‌شود. مثلاً $(0.3434)_5$ را اگر در 5^2 ضرب کنیم برابر $(34.34)_5$ می‌شود. عدد $a = (0.\overline{34})_5$ را در 5^2 یعنی ۲۵ ضرب می‌کنیم سپس $25a$ را منهای a می‌کنیم:

$$a = (0.343434\ldots)_5$$

$$25a = (34.34.34\ldots)_5$$

$$25a - a = (34)_5 \Rightarrow 24a = 3 \times 5 + 4 \Rightarrow 24a = 19 \Rightarrow a = \frac{19}{24}$$

مثال: عدد متناوب مرکب $(0.\overline{234})_6$ را به مبنای 10 تبدیل کنید.

حل: می توان با روش اول مثال قبل این عمل را انجام داد که به عهده شما! من از روش جالب دوم استفاده می کنم، فرض کنید $a = (0.2\ 343434\ldots)_6$ پس $6a = (2.343434\ldots)_6$ پس $6a = (234.3434\ldots)_6$ بنابراین:

$$6^3 a - 6a = (234.3434\ldots)_6 - (2.3434\ldots)_6 = (234 - 2)_6 = (232)_6$$

$$\Rightarrow 216a - 6a = 2 \times 6^2 + 3 \times 6 + 2 = 92 \Rightarrow 210a = 92 \Rightarrow a = \frac{92}{210} = \frac{46}{105}$$

مثال: عدد $(x)_{10}$ چند رقم در مبنای 2 نیاز دارد؟

حل: فرض کنید این عدد نیاز به n رقم یا بیت دارد. بزرگترین عدد با n بیت برابر $2^n - 1$ است پس باید $2^n - 1 \geq x$ که بتوان x را در n بیت جا کرد: بنابراین $2^n \geq x + 1$ حال از طرفین لگاریتم در پایه 2 می گیریم پس $n \geq \log_2(x + 1)$ پس کوچکترین n برابر $\lceil \log_2(x + 1) \rceil$ است. توجه کنید اگر مثلاً $n \geq 3.5$ آنگاه کوچکترین مقدار صحیح n برابر 4 یعنی $\lceil 3.5 \rceil$ است.

توجه

علامت $\lceil \cdot \rceil$ سقف (ceiling) است. $\lceil a \rceil$ برابر کوچک ترین عدد صحیح است که از a کوچک تر نباشد. مثلاً $\lceil 3.5 \rceil = 4$ یا $\lceil -3.5 \rceil = -3$ یا $\lceil 3 \rceil = 3$.

علامت $\lfloor \cdot \rfloor$ نیز کف (Floor) است. $\lfloor a \rfloor$ برابر است با بزرگترین عدد صحیح که از a بزرگ تر نباشد. مثلاً $\lfloor 3.5 \rfloor = 3$ یا $\lfloor -3.5 \rfloor = -4$ یا $\lfloor 3 \rfloor = 3$.

توجه

لگاریتم، معکوس توان است. یعنی اگر $a^b = c$ پس $\log_a c = b$ مثلاً $\log_2 8 = 3$ چون $2^3 = 8$.

نکته

برای نمایش عدد $(x)_{10}$ در مبنای r نیاز به حداقل $\lceil \log_r(x + 1) \rceil$ یا $\lfloor \log_r x \rfloor + 1$ رقم است.

مثال: می خواهیم اعداد n رقمی را از مبنای 10 به مبنای r ببریم. در مبنای r چند رقم نیاز است؟

حل: بزرگترین عدد n رقمی در مبنای 10 برابر $x = 10^n - 1$ است و طبق نکته فوق برای نمایش این عدد در مبنای r نیاز به $\lceil \log_r(10^n - 1 + 1) \rceil = \lceil n \cdot \log_r 10 \rceil$ رقم است.

توجه کنید این تعداد رقم برای نمایش بزرگترین عدد n رقمی مبنای 10 است بنابراین برای نمایش سایر اعداد مبنای 10 نیز کافی است.

نکته

برای نمایش اعداد n رقمی مبنای t در مبنای r نیاز به $\lceil n \cdot \log_r t \rceil$ رقم در مبنای r است.

۵- جمع و تفريق در مبناهای مختلف

برای جمع دو عدد در مبنای r کافی است دو عدد را زیر هم بنویسیم طوری که یکان آنها با هم نظیر باشد، سپس از سمت راست هر دو رقم را جمع می کنیم و اگر حاصل جمع بیش تر یا مساوی r شد، r واحد از حاصل کم کنیم و یک واحد به حاصل جمع دو رقم سمت چپ آنها اضافه کنیم. به چند مثال توجه کنید.

$\begin{array}{r} 1\ 1\ 1\ 1 \\ (2\ 3\ 7\ .\ 4\ 3)_{10} \\ + (8\ 5\ .\ 7\ 9)_{10} \\ \hline (3\ 2\ 3\ .\ 2\ 2)_{10} \end{array}$	$\begin{array}{r} 1\ 1\ 1 \\ (1\ 1\ 0\ 1\ 1)_2 \\ + (1\ 0\ 1\ 1\ 0)_2 \\ \hline (1\ 1\ 0\ 0\ 0\ 1)_2 \end{array}$	$\begin{array}{r} 1\ 1 \\ (4\ 5\ 0\ 6)_8 \\ + (5\ 3\ 2\ 4)_8 \\ \hline (1\ 2\ 0\ 3\ 2)_8 \end{array}$
--	---	---

برای انجام تفریق $A - B$ در مبنای r یک روش به روش قلم و کاغذ یا قرض گرفتن معروف است که B را زیر A بنویسیم طوری که یکان آنها نظیر هم باشند و سپس ارقام B را از سمت راست از ارقام A کم کنیم و هر جا رقم B از رقم نظیرش در A بزرگتر بود از رقم سمت چپتر در A باید یک واحد قرض بگیریم که ارزش آن قرض برابر r می شود:

$$\begin{array}{r} \textcircled{13} \quad \textcircled{1} \quad \textcircled{411112} \\ 1 \textcircled{13} \textcircled{6} \quad \textcircled{3} \textcircled{13} \quad 0 \textcircled{2} 0 \textcircled{2} \textcircled{2} \\ (\cancel{2} \cancel{3} \cancel{7} . \cancel{4} \cancel{3})_{10} \quad (\cancel{1} \cancel{0} \cancel{1} \cancel{0} \cancel{0} \cancel{1})_2 \quad (\cancel{3} \cancel{3} \cancel{2} \cancel{4})_8 \\ - (\quad 8 \quad 5 \quad . \quad 7 \quad 9)_{10} \quad - (\quad 0 \quad 1 \quad 0 \quad 1 \quad 1 \quad 0)_2 \quad - (\quad 4 \quad 5 \quad 0 \quad 6)_8 \\ \hline (\quad 1 \quad 5 \quad 1 \quad . \quad 6 \quad 4)_{10} \quad (\quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1)_2 \quad (\quad 0 \quad 6 \quad 1 \quad 6)_8 \end{array}$$

روش دومی برای تفریق $A - B$ وجود دارد که A را با مکمل r عدد B جمع کنیم که با مکمل‌ها در ادامه فصل آشنا می‌شویم و این روش را بیان می‌کنیم.

۶- مکمل‌ها

در مبنای r دو نوع عمل مکمل‌گیری تعریف شده است: مکمل $r-1$ و مکمل r . کاربرد مکمل‌ها، نمایش اعداد منفی و همچنین تبدیل عمل تفریق به عمل جمع است.

مکمل $r-1$ یا مکمل کاهش یافته (reduced complement) عدد $(a)_r$ برابر است با $r^n - r^m - a$ که n و m به ترتیب تعداد ارقام صحیح و تعداد ارقام اعشار عدد a هستند. مکمل $r-1$ عدد a را به شکل $[a]_{r-1}$ یا $c_{r-1}(a)$ نشان می‌دهیم که c مخفف complement است. پس $[a]_{r-1} = r^n - r^m - a$. اگر a عدد صحیح باشد یعنی $m=0$ آنگاه $[a]_{r-1} = r^n - 1 - a$.

مکمل r یا مکمل مبنای (radix complement) عدد $(a)_r$ برابر است با $c_r(a) = [a]_r = r^n - a$. این دو فرمول قرارداد هستند. برای درک مکمل‌ها فرض کنید مثلاً $a = (237.64)_{10}$ و می‌خواهیم مکمل 9 (مکمل کاهش یافته) و مکمل 10 (مکمل مبنای) عدد a را بیابیم. عدد a دارای $n=3$ رقم صحیح و $m=2$ رقم اعشار است. پس:

$$C_9(a) = [a]_9 = 10^3 - 10^{-2} - (237.64)_{10} = 999.99 - 237.64 = (762.35)_{10}$$

$$C_{10}(a) = [a]_{10} = 10^3 - (237.64)_{10} = 1000 - 237.64 = (762.36)_{10}$$

به عنوان مثالی دیگر فرض کنید $a = (237.64)_8$ و می‌خواهیم مکمل 7 (مکمل کاهش یافته) و مکمل 8 (مکمل مبنای) عدد a را بیابیم.

$$C_7(a) = [a]_7 = 8^3 - 8^{-2} - (237.64)_{10} = (777.77)_8 - (237.64)_8 = (540.13)_8$$

$$C_8(a) = [a]_8 = 8^3 - (237.64)_{10} = (1000)_8 - (237.64)_8 = (540.14)_8$$

توجه کنید برای انجام تفریق $8^3 - 8^{-2}$ باید در مبنای 8 اعداد را بنویسیم که:

$$8^3 = (1000)_8 \quad \text{و} \quad 8^{-2} = (0.01)_8 \quad \text{بنابراین این تفریق برابر است با:}$$

$$\begin{array}{r} \textcircled{7} \textcircled{7} \textcircled{7} \quad \textcircled{7} \\ \textcircled{0} \textcircled{8} \textcircled{8} \textcircled{8} \quad \textcircled{8} \textcircled{8} \\ (\quad \cancel{1} \quad \cancel{0} \quad \cancel{0} \quad . \quad \cancel{0} \quad \cancel{0})_8 \\ - \quad (\quad 0 \quad . \quad 0 \quad 1)_8 \\ \hline (\quad 7 \quad 7 \quad 7 \quad . \quad 7 \quad 7)_8 \end{array}$$

با توجه به مثال‌های ذکر شده می‌توان به نکات زیر برای یافتن مکمل‌ها رسید که با این نکات بدون استفاده از فرمول‌ها می‌توان مکمل در مبناهای مختلف را یافت.

نکته

برای یافتن مکمل $r-1$ عدد $(a)_r$ کافی است تمام ارقام عدد a (چه صحیح چه اعشار) را از $r-1$ کم کنیم.

مثلاً برای یافتن مکمل 9 عدد $(a)_{10}$ همه ارقام a را از 9 کم می‌کنیم یا برای یافتن مکمل 7 عدد $(a)_8$ همه ارقام a را از 7 کم می‌کنیم یا برای یافتن مکمل یک عدد $(a)_2$ ، همه ارقام a را از یک کم می‌کنیم که یعنی بیت‌های a را عوض یا اصطلاحاً not یا متمم می‌کنیم. مثلاً مکمل یک عدد $a = (10010)_2$ برابر $\bar{a} = (01101)_2$ است که $\bar{a}$ یا a' مکمل یا متمم یا not عدد a است.

نکته

برای یافتن مکمل r عدد $(a)_r$ می‌توان ابتدا مکمل $r-1$ عدد a را یافت و سپس سمت راست‌ترین رقم آن را به علاوه یک کنیم. در واقع با توجه به فرمول مکمل‌ها رابطه $[a]_r = [a]_{r-1} + r^{-m}$ بین مکمل‌ها برقرار است که این رابطه به معنی گفته شده است. ولی روش سریع‌تر برای یافتن مکمل r عدد $(a)_r$ این است که از سمت راست a حرکت کنیم و صفرهای سمت راست a را در صورت وجود تغییر ندهیم. اولین رقم غیر صفر a را از r و سایر ارقام را از $r-1$ کم کنیم. مثلاً مکمل 10 عدد $(350400)_{10}$ برابر $(649600)_{10}$ است که دو صفر سمت راست تغییر نکرد، 4 از 10 کم شد و سه رقم 0 و 5 و 3 هر کدام از 9 کم شدند. به مثال‌های زیر توجه کنید و به نکات گفته شده ربط دهید:

$$(3070.600)_8 \begin{array}{l} \xrightarrow{\text{مکمل 7}} (4707.177)_8 \\ \xrightarrow{\text{مکمل 8}} (4707.200)_8 \end{array}$$

$$(101100)_2 \begin{array}{l} \xrightarrow{\text{مکمل یک}} (010011)_2 \\ \xrightarrow{\text{مکمل دو}} (010100)_2 \end{array}$$

$$(1001)_2 \begin{array}{l} \xrightarrow{\text{مکمل یک}} (0110)_2 \\ \xrightarrow{\text{مکمل دو}} (0111)_2 \end{array}$$

$$(0000)_2 \begin{array}{l} \xrightarrow{\text{مکمل یک}} (1111)_2 \\ \xrightarrow{\text{مکمل دو}} (0000)_2 \end{array}$$

توجه کنید مکمل دو عدد $(0000)_2$ در واقع $(10000)_2$ می‌شود ولی چون عدد، 4 بیتی است مکمل دوی آن را $(0000)_2$ در نظر می‌گیریم و از بیت پنجم صرف‌نظر می‌کنیم. در واقع مکمل r و $r-1$ عدد $(a)_r$ به تعداد رقم‌های عدد a ، رقم دارند.

نکته

برای یافتن مکمل دوی عدد $(a)_2$ صفرهای سمت راست عدد و اولین یک را تغییر ندهید و سایر بیت‌ها را عوض کنید.

مثلاً مکمل دوی اعداد $(0100)_2$ ، $(1000)_2$ و $(11001)_2$ به ترتیب، $(1100)_2$ و $(1000)_2$ و $(00111)_2$ می‌باشد. یکی از روش‌های انجام عمل $A-B$ در مبنای r آن است که این عمل را به جمع تبدیل کنیم که برای این منظور A را با مکمل r عدد B جمع می‌کنیم.

به مثال‌های زیر توجه کنید (این مثال‌ها را قبلاً با روش قرض دیده‌اید):

$$(237.43)_{10} - (085.79)_{10} = (237.43)_{10} + (914.21)_{10} = (1151.64)_{10} \xrightarrow{\text{حذف رقم نقلی}} (151.64)_{10}$$

توجه کنید باید هر دو عدد دارای تعداد رقم‌های برابر باشند و همچنین می‌توان ثابت کرد که اگر $A \geq B$ آنگاه یک رقم اصطلاحاً نقلی (carry) از سمت چپ خارج می‌شود که آن را باید نادیده بگیریم.

$$(101001)_2 - (010110)_2 = (101001)_2 + (101010)_2 = (1010011)_2 \xrightarrow{\text{حذف رقم نقلی}} (010011)_2$$

$$(5324)_8 - (4506)_8 = (5324)_8 + (3272)_8 = (10616)_8 \xrightarrow{\text{حذف رقم نقلی}} (0616)_8$$

لازم به ذکر است که اگر $A < B$ آنگاه قم نقلی تولید نمی‌شود ولی حاصل نیز درست نیست در واقع حاصل به شکلی منفی شده است که فعلاً بحث نمی‌کنیم.

۷- اعداد علامت‌دار در مبنای 2

ما تاکنون با اعداد محض یا بی‌علامت سر و کار داشتیم. حال سؤال این است که چطور می‌توان در کامپیوتر اعداد منفی را نیز ذخیره کرد. برای نمایش و ذخیره اعداد مثبت و منفی چهار قرارداد یا روش مطرح شده است که ما در اینجا سه روش را بررسی می‌کنیم و روش چهارم را در درس معماری کامپیوتر بخش ممیز شناور خواهیم دید. سه روش ما عبارتند از:

* روش علامت مقدار (sign magnitude)

* روش مکمل یک (1's complement)

* روش مکمل 2 (2's complement)

در این سه روش اعداد مثبت به یک شکل نمایش داده می‌شوند و تفاوت این روش‌ها در نمایش اعداد منفی است. البته وجه اشتراک این سه روش این است که بیت سمت چپ اعداد مثبت، صفر است و بیت سمت چپ اعداد منفی، یک است. ارزش بیت‌ها در این سه روش همانند سیستم اعداد بی‌علامت مبنای 2 است یعنی از سمت راست اولین بیت دارای ارزش 2^0 (عدد را صحیح فرض کرده‌ایم)، بیت دوم دارای ارزش 2^1 و به همین ترتیب می‌باشد، فقط تفاوت در بیت سمت چپ است. اگر اعداد n بیتی باشند، ارزش بیت سمت چپ در سیستم بی‌علامت برابر 2^{n-1} است. در سیستم علامت مقدار، بیت سمت چپ ارزش ندارد و فقط نشان‌دهنده‌ی علامت است اگر این بیت 1 باشد عدد منفی و اگر 0 باشد عدد مثبت است. در سیستم مکمل یک، ارزش بیت سمت چپ $(2^{n-1}-1)$ است و در سیستم مکمل دو ارزش بیت سمت چپ 2^{n-1} است. می‌دونم احتمالاً کمی گیج شده‌اید!! برای درک بهتر فرض کنید اعداد 4 بیتی هستند. $a_3a_2a_1a_0$ در سیستم بی‌علامت ارزش a_2, a_1, a_0 و a_3 به ترتیب $2^0, 2^1, 2^2$ و 2^3 است. در سیستم علامت مقدار و مکمل یک و مکمل دو ارزش a_0, a_1, a_2 همانند بی‌علامت است فقط بیت a_3 در سیستم علامت مقدار ارزش ندارد و اگر 0 باشد عدد مثبت و اگر 1 باشد عدد منفی است. این بیت در سیستم مکمل یک دارای ارزش $(2^3-1)=7$ است و در سیستم مکمل دو دارای ارزش $2^3=8$ می‌باشد.

مثلاً عدد 4 بیتی $(1110)_2$ در سیستم بی‌علامت برابر $2^3+2^2+2^1=14$ است، در سیستم علامت مقدار برابر $-(2^3+2^1)=-6$ است، در سیستم مکمل یک برابر $-(2^3-1)+2^2+2^1=-1$ است و در سیستم مکمل دو برابر $2^3+2^2+2^1=14$ می‌باشد.

مثال: مطلوبست مقدار عدد باینری شش بیتی $(101100)_2$ در سیستم‌های بی‌علامت، علامت مقدار، مکمل یک و مکمل دو.

حل: در سیستم بی‌علامت ارزش بیت چپ برابر 2^5 است پس مقدار عدد باینری داده شده $2^5+2^3+2^2=48$ می‌باشد، در سیستم علامت مقدار، بیت سمت چپ ارزش ندارد و چون 1 است یعنی عدد منفی است و مقدار عدد داده شده $-(2^3+2^2)=-12$ می‌باشد.

در سیستم مکمل یک، ارزش بیت سمت چپ برابر $(2^5-1)=31$ است و مقدار عدد داده شده برابر $31+8+4=43$ است.

در سیستم مکمل دو ارزش بیت سمت چپ برابر $2^5=32$ است و مقدار عدد داده شده برابر $32+8+4=44$ می‌باشد. به طور کلی می‌توان عدد n بیتی $a=(a_{n-1}a_{n-2}\dots a_1a_0)$ را با فرمول‌های زیر در سیستم‌های مختلف ارزشیابی کرد یعنی به مبنای 10 تبدیل کرد که ما با $val(a)$ نشان می‌دهیم:

بی علامت $\text{val}(a) = a_{n-1} \times 2^{n-1} + a_{n-2} \times 2^{n-2} + \dots + a_1 \times 2^1 + a_0 \times 2^0$

علامت مقدار $\text{val}(a) = (-1)^{a_{n-1}} \times (a_{n-2} \times 2^{n-2} + \dots + a_1 \times 2^1 + a_0 \times 2^0)$

یک مکمل $\text{val}(a) = a_{n-1} \times -(2^{n-1} - 1) + a_{n-2} \times 2^{n-2} + \dots + a_1 \times 2^1 + a_0 \times 2^0$

دو مکمل $\text{val}(a) = a_{n-1} \times -2^{n-1} + a_{n-2} \times 2^{n-2} + \dots + a_1 \times 2^1 + a_0 \times 2^0$

با توجه به فرمول‌های داده شده می‌توان بازه نمایش یعنی مقادیر کوچک‌ترین و بزرگ‌ترین اعداد قابل نمایش با n بیت را در سیستم‌های مختلف به دست آورد.

در سیستم بی علامت کوچک‌ترین عدد $(00\dots0)_2 = 0$ و بزرگ‌ترین عدد $(111\dots1)_2 = 2^n - 1$ است. در سیستم علامت مقدار، کوچک‌ترین عدد $(111\dots1)_2 = -(2^{n-1} - 1)$ و بزرگ‌ترین عدد $(011\dots1)_2 = 2^{n-1} - 1$ می‌باشد.

در سیستم مکمل یک، کوچک‌ترین عدد $(100\dots0)_2 = -(2^{n-1} - 1)$ و بزرگ‌ترین عدد $(011\dots1)_2 = 2^{n-1} - 1$ می‌باشد.

در سیستم مکمل دو، کوچک‌ترین عدد $(100\dots0)_2 = -2^{n-1}$ و بزرگ‌ترین عدد $(011\dots1)_2 = 2^{n-1} - 1$ می‌باشد. توجه کنید

که n بیت، 2^n حالت دارد و سیستم‌های بی علامت و مکمل دو دقیقاً 2^n عدد متفاوت را ذخیره می‌کنند ولی سیستم‌های

علامت مقدار و مکمل یک، $2^n - 1$ عدد مختلف را ذخیره می‌کنند چون این دو سیستم عدد صفر را به دو شکل مختلف

نمایش می‌دهند. در سیستم علامت مقدار $(00\dots0)_2 = +0$ و $(100\dots0)_2 = -0$ هر دو نمایش صفر هستند. در سیستم مکمل

یک $(00\dots0)_2 = +0$ و $(11\dots1)_2 = -0$ هر دو نمایش صفر هستند. این را هم بدانید که در سیستم علامت مقدار برای آنکه

یک عدد قرینه شود کافی است فقط بیت سمت چپ آن در فرم دودویی عوض شود. در سیستم مکمل یک برای این منظور

باید نمایش دودویی آن عدد مکمل یک شود و در سیستم مکمل دو باید فرم دودویی آن مکمل دو شود. مثلاً با 4

بیت $(0110)_2$ در هر سه سیستم علامت مقدار و مکمل یک و مکمل دو $+6$ می‌باشد، و نمایش -6 در سیستم علامت

مقدار $(1110)_2$ ، در سیستم مکمل یک $(1001)_2$ و در سیستم مکمل دو $(1010)_2$ می‌باشد.

جدول زیر با 4 بیت، اعداد قابل نمایش در سه سیستم علامت‌دار را نشان داده است:

سیستم مکمل دو	سیستم مکمل یک	سیستم علامت مقدار	عدد دهدهی
0111	0111	0111	+7
0110	0110	0110	+6
0101	0101	0101	+5
0100	0100	0100	+4
0011	0011	0011	+3
0010	0010	0010	+2
0001	0001	0001	+1
0000	0000	0000	+0
ندارد	1111	1000	-0
1111	1110	1001	-1
1110	1101	1010	-2
1101	1100	1011	-3
1100	1011	1100	-4
1011	1010	1101	-5
1010	1001	1110	-6
1001	1000	1111	-7
1000	غیر قابل نمایش	غیر قابل نمایش	-8

عیب سیستم‌های علامت‌مقدار و مکمل یک این است که برای صفر دو نمایش مختلف دارند، هم صفر مثبت و هم صفر منفی. البته سیستم علامت مقدار عیب دیگری دارد و آن اینکه عملیات جمع و تفریق در این سیستم برای کامپیوتر زمانبر است. سیستم مکمل یک نیز عملاً جایی استفاده نمی‌شود و بهینه شده آن سیستم مکمل دو است.

نکته

پیش‌فرض اعداد علامت‌دار، سیستم مکمل دو است یعنی هر جا گفته شد اعداد علامت‌دار هستند و سیستم نمایش ذکر نشد، سیستم مکمل دو است.

۸- معرفی پرچم‌ها یا فلگ‌ها

رجیستر یا ثبات حافظه‌ی سریع و کوچکی است که اندازه آن معمولاً ۸ یا ۱۶ یا ۳۲ یا ۶۴ بیت است و داخل پردازنده قرار دارد. برخی رجیسترها عمومی هستند و برنامه‌نویس می‌تواند به آنها دسترسی داشته باشد ولی برخی خاص منظوره هستند و در حالت عادی به آنها دسترسی وجود ندارد در فصول بعدی با رجیسترها آشنا می‌شویم. در کامپیوترها رجیستری به نام FLAGS یا PSW یا STATUS وجود دارد که ۴ بیت از این رجیستر در این درس اهمیت دارند. این بیت‌ها که فلگ یا پرچم نامیده می‌شوند پس از انجام یک عمل حسابی مثل جمع یا تفریق یا عمل منطقی مقداردهی می‌شوند. این فلگ‌ها به نام‌های Z (Zero)، S (Sign)، N (Negative)، C (Carry)، V (Overflow) معروف هستند. فلگ Z وقتی ۱ می‌شود که حاصل عملیاتی مثل جمع یا تفریق صفر شود. فلگ S یا N وقتی ۱ می‌شود که حاصل عملیات منفی شود یعنی بیت سمت چپ حاصل ۱ شود. فلگ C وقتی ۱ می‌شود که نتیجه عمل دو عدد n بیتی، n+1 بیت شود یعنی از بیت سمت چپ ۱ خارج شود. فلگ V وقتی ۱ می‌شود که حاصل عملیات در بازه مجاز نباشد، البته ممکن است تصور کنید فلگ C همان V است ولی لزوماً اینطور نیست و در سیستم‌های مختلف مثل بی‌علامت و علامت‌دار، فلگ‌ها ممکن است متفاوت تعبیر شوند.

۹- جمع و تفریق در سیستم مکمل دو

قبل از اینکه به جمع و تفریق بپردازیم، لطفاً رشته‌های دودویی 11001، 1001، 11001 را که در سیستم مکمل دو فرض می‌کنیم، ارزشیابی کنید یعنی به مبنای ده ببرید. با توجه به ارزش بیت‌ها مقدار تمامی این رشته‌ها برابر ۷- می‌باشد. در سیستم مکمل ۲ اگر بیت سمت چپ را تکرار کنید، مقدار عدد عوض نمی‌شود. به این کار **گسترش علامت** (sign extend) گویند. یعنی مثلاً اگر یک رشته ۴ بیتی را بخواهیم به ۸ بیت تبدیل کنیم باید بیت سمت چپ را چهار بار تکرار کنیم. بنابراین رشته دودویی 1001 با 11111001 در سیستم مکمل دو دارای یک مقدار هستند و هر دو ۷- می‌باشند یا رشته 0111 با 00001111 هر دو ۷+ می‌باشند. برای جمع دو عدد در سیستم مکمل دو ابتدا با گسترش علامت تعداد بیت‌های آن دو عدد را یکسان می‌کنیم و سپس دو عدد را زیر هم می‌نویسیم و همانند جمع دو عدد بی‌علامت، جمع می‌کنیم.

مثال: فرض کنید سیستم مکمل دو و ۴ بیتی است یعنی اعداد ۴ بیتی هستند و حاصل نیز در ۴ بیت باید ذخیره شود. حاصل جمع‌های $(+7)+(-7)$ ، $(+4)+(-2)$ ، $(+6)+(-3)$ ، $(+4)+(+2)$ ، $(-6)+(-3)$ و $(+5)+(+4)$ را بیابید و وضعیت پرچم‌ها را مشخص کنید.

حل:

$$\begin{array}{r} 111 \\ 1001 \\ + 0111 \\ \hline 0000 \end{array}$$

$$(-7) + (+7) = (1001)_2 + (0111)_2 = (0000)_2 = 0$$

$$C=1, Z=1, S=0, V=0$$

از سمت چپ یک رقم نقلی خارج شد پس $C=1$ ، حاصل عملیات صفر است پس $Z=1$ ، بیت سمت چپ حاصل صفر است پس $S=0$ ، حاصل درست است یعنی سرریز نداریم پس $V=0$.

①

$$(-3) + (+6) = (1101)_2 + (0110)_2 = (0011)_2 = +3$$

$$\begin{array}{r} 1\ 1\ 0\ 1 \\ +\ 0\ 1\ 1\ 0 \\ \hline 0\ 0\ 1\ 1 \end{array}$$

$$C=1, Z=0, S=0, V=0$$

①

$$(-2) + (-4) = (1110)_2 + (1100)_2 = (1010)_2 = -6$$

$$\begin{array}{r} 1\ 1\ 1\ 0 \\ +\ 1\ 1\ 0\ 0 \\ \hline 1\ 0\ 1\ 0 \end{array}$$

$$C=1, Z=0, S=1, V=0$$

$$(+2) + (+4) = (0010)_2 + (0100)_2 = (0110)_2 = +6$$

$$\begin{array}{r} 0\ 0\ 1\ 0 \\ +\ 0\ 1\ 0\ 0 \\ \hline 0\ 1\ 1\ 0 \end{array}$$

$$C=0, Z=0, S=0, V=0$$

$$(-3) + (-6) = (1101)_2 + (1010)_2 = (0111)_2 = +7$$

$$\begin{array}{r} 1\ 1\ 0\ 1 \\ +\ 1\ 0\ 1\ 0 \\ \hline 0\ 1\ 1\ 1 \end{array}$$

$$C=1, Z=0, S=0, V=1$$

حاصل جمع 3- با 6- باید 9- شود ولی 9- با 4 بیت قابل نمایش نیست و به خاطر همین حاصل اشتباه شده و سرریز رخ داده است و پرچم V فعال می شود.

①

$$0\ 1\ 0\ 0\ (+4) + (+5) = (0100)_2 + (0101)_2 = (1001) = -7$$

$$\begin{array}{r} 0\ 1\ 0\ 0 \\ +\ 0\ 1\ 0\ 1 \\ \hline 1\ 0\ 0\ 1 \end{array}$$

$$C=0, Z=0, S=1, V=1$$

حاصل جمع 4+ با 5+ باید 9+ شود ولی 9+ با 4 بیت قابل نمایش نیست و به خاطر همین حاصل اشتباه شده و سرریز رخ داده است و پرچم V فعال می شود.

نکته

شرط لازم و کافی برای وقوع سرریز در جمع دو عدد سیستم مکمل دو آن است که حاصل جمع دو عدد مثبت، منفی شود یا حاصل جمع دو عدد منفی، مثبت شود. توجه کنید که جمع دو عدد با علامت های مختلف هیچگاه سرریز نمی شود.

نکته

شرط لازم و کافی برای وقوع سرریز در جمع دو عدد سیستم مکمل دو آن است که رقم نقلی که به بیت چپ وارد می شود با رقم نقلی که از آن خارج می شود (یعنی همان فلگ C) متفاوت باشد. در جمع 4+ با 5+، به بیت چپ رقم نقلی 1 وارد شد و رقم نقلی 0 (C=0) خارج شد و سرریز شد. در جمع 3- با 6- به بیت چپ رقم نقلی 0 وارد شد ولی 1 (C=1) خارج شد که نشان دهنده سرریز است. در واقع هرگاه دو رقم نقلی آخر یکسان نباشند به این معنی است که حاصل جمع دو عدد هم علامت، علامتی متفاوت از آن دو عدد دارد.

برای انجام تفریق $A-B$ در سیستم مکمل دو، A را با قرینه B یعنی B جمع می‌کنیم. B یعنی مکمل دوی B که یعنی مکمل یک B به علاوه 1 پس:

$$A-B = A + (-B) = A + \bar{B} + 1$$

البته $A-B$ را می‌توان به شیوه قلم و کاغذ و قرض گرفتن نیز انجام داد یعنی B را زیر A بنویسیم و بیت‌های B را از بیت‌های A کم کنیم ولی در کامپیوترها تفریق به جمع تبدیل می‌شود که با یک سخت‌افزار جمع‌کننده هم جمع و هم تفریق انجام شود.

مثال: تفریق $(-3) - (-5)$ را در سیستم مکمل دو با 4 بیت انجام دهید و وضعیت پرچم‌ها را مشخص کنید.
حل:

$$\begin{array}{r} 11 \\ 1011 \\ + 0010 \\ \hline 1110 \\ C=0, Z=0, S=1, V=0 \end{array}$$

$$(-5) - (-3) = (1011)_2 - (1101)_2 = (1011)_2 + (0010)_2 + 1 = (1110)_2 = -2$$

نکته

در عمل $A-B$ اگر A و B هم علامت باشند امکان وقوع سرریز وجود ندارد و حاصل حتماً در بازه‌ی مجاز است. سرریز وقتی پیش می‌آید که A و B مختلف‌العلامه باشند و علامت حاصل با علامت B یکسان باشد.

مثال: تفریق $(+4) - (-5)$ را در سیستم مکمل دو با 4 بیت انجام دهید و وضعیت فلگ‌ها را مشخص کنید.
حل:

$$\begin{array}{r} 11 \\ 1011 \\ + 1011 \\ \hline 0111 \\ C=1, Z=0, S=0, V=1 \end{array}$$

$$(-5) - (+4) = (1011)_2 - (0100)_2 = 1011 + 1011 + 1 = (0111)_2 = +7$$

نکته

در عمل $A-B$ در سیستم مکمل دو، با کمک فلگ‌ها می‌توان A و B را مقایسه کرد و متوجه شد که A بزرگ‌تر، کوچک‌تر یا مساوی B هست یا خیر.

شرط	وضعیت فلگ‌ها
$A \geq B$	$S = V$
$A < B$	$S \neq V$
$A = B$	$Z = 1$
$A \neq B$	$Z = 0$
$A > B$	$S = V$ and $Z \neq 0$
$A \leq B$	$S \neq V$ or $Z = 1$

در جدول مقابل حداقل شروط لازم و کافی برای هر حالت آمده است. مثلاً وقتی $A \geq B$ ما انتظار داریم که حاصل $A-B$ مثبت شود یعنی $S=0$ ولی به شرطی چنین می‌شود که سرریز رخ ندهد یعنی $V=0$. حال اگر سرریز رخ دهد یعنی $V=1$ آنگاه حاصل منفی می‌شود یعنی $S=1$ ، پس $A \geq B$ اگر و فقط اگر $S=V$. سایر حالات را می‌توانید به راحتی بررسی کنید.

۱۰- جمع و تفریق سیستم مکمل یک

در سیستم مکمل یک نیز همانند مکمل دو اگر بیت سمت چپ را تکرار کنیم، مقدار عدد عوض نمی‌شود. مثلاً در این سیستم 1001، 11001 و 111001 همگی 6- هستند. در این سیستم برای جمع دو عدد باید تعداد بیت آن دو عدد یکسان باشد و آن دو عدد را زیر هم بنویسیم و همانند سیستم مکمل دو جمع کنیم فقط باید نتیجه را با فلگ C مجدد جمع کنیم!

به عنوان مثال با 4 بیت حاصل جمع $(+3) + (-5)$ و $(-3) + (+7)$ را می‌یابیم:

$$\begin{array}{r} \textcircled{1} \\ 1010 \\ + 0011 \\ \hline 1101 \end{array}$$

$(-5) + (+3) = (1010)_2 + (0011)_2 = (1101)_2 = -2$

$$\begin{array}{r} \textcircled{1} \\ (0111)_2 \\ + (1100)_2 \\ \hline (0100)_2 \end{array}$$

$(+7) + (-3) = (0111)_2 + (1100)_2 = (0100)_2 = +4$

C=1
0/00

برای انجام عمل تفریق $A - B$ کافی است A را با قرینه B جمع کنیم که قرینه B در سیستم مکمل یک یعنی مکمل یک B پس:

$$A - B = A + (-B) = A + \bar{B}$$

مثلاً $(+7) - (+3) = (0111)_2 - (0011)_2 = (0111)_2 + (1100)_2 = (0100)_2 = +4$

$$(12+7) - (12+3) = (1100)_2 + (0111)_2 = (1001)_2 = -5$$

۱۱- جمع و تفریق سیستم بی علامت

جمع دو عدد بی علامت n بیتی حداکثر $n+1$ بیت می شود پس اگر حاصل جمع دو عدد n بیتی را در n بیت ذخیره کنیم آنگاه فلگ C همان فلگ سرریز است یعنی اگر $C=1$ سرریز رخ داده است.

مثلاً با 4 بیت حاصل جمع $12+7$ باید 19 شود ولی 19 در 4 بیت قابل ذخیره نیست و سرریز رخ می دهد:

$$12+7 = (1100)_2 + (0111)_2 = \overset{\text{C}}{\downarrow} 10011 = (0011)_2 = 3$$

فلگ C یک شده است پس سرریز می باشد. البته همانطور که گفته شد اگر خروجی در 5 بیت ذخیره شود حاصل $(10011)_2$ می شود که برابر 19 است.

برای انجام عمل تفریق $A - B$ یک روش استفاده از قرض گرفتن است یعنی B را زیر A بنویسیم و نظیر به نظیر تفریق کنیم و هر جا لازم بود قرض بگیریم. در این روش فلگ C رقم نقلی نیست بلکه رقم قرضی است و وقتی 1 می شود که $A < B$ یعنی نیاز به قرض باشد و اگر $A \geq B$ آنگاه $C=0$.

ولی روش پیش فرض برای انجام $A - B$ ، تبدیل به جمع است که همانند سیستم مکمل دو، A را با مکمل دوی عدد B جمع می کنیم یعنی $A - B = A + \bar{B} + 1$ ، در این روش اگر $A \geq B$ آنگاه رقم نقلی $C=1$ می شود و اگر $A < B$ آنگاه $C=0$. به عنوان مثال با 4 بیت تفریق 3-9 را با دو روش انجام می دهیم با روش قرض $(0110)_2 - (0011)_2 = (1001)_2 = -5$ می شود که $C=0$ یعنی نیازی به قرض نیست:

$$\begin{array}{r} \textcircled{0} \textcircled{1} \textcircled{2} \\ 1001 \\ - 0011 \\ \hline 0110 \end{array}$$

با روش جمع $9-3 = (1001)_2 - (0011)_2 = (1001)_2 + (1100)_2 + 1 = (0110)_2 = 6$ می باشد:

$$\begin{array}{r} \textcircled{1} \\ 1001 \\ + 1100 \\ \hline 0110 \end{array}$$

۱۲- جمع و تفریق سیستم علامت مقدار

جمع و تفریق در این سیستم نیاز به ملاحظات زیادی دارد و نمی‌توان همانند سیستم‌های قبلی دو عدد را به صورت معمولی باهم جمع یا تفریق کرد. بلکه باید علامت دو عدد بررسی شوند و با توجه به علامت دو عدد و عمل بین دو عدد، ارزش دو عدد را (علامت‌ها را حذف کنیم) با هم جمع و تفریق کنیم و در نهایت علامت حاصل را تنظیم کنیم.

مثلاً با 4 بیت می‌خواهیم حاصل جمع $(0011)_2 + (1101)_2$ را بیابیم. عدد اول منفی و دومی مثبت است. پس باید ارزش این دو عدد را از هم کم کنیم و سپس علامت عدد بزرگ‌تر را قرار دهیم. عدد اول 5- و دومی 3+ است که ارزش عدد اول 5 و ارزش عدد دوم 3 است و ارزش حاصل 2 می‌شود و علامت عدد بزرگ‌تر منفی است پس حاصل $(1010)_2$ می‌باشد. البته انجام این عملیات برای کامپیوتر کمی زمانبر است و از این نظر این سیستم بدتر از بقیه سیستم‌هاست. الگوریتم جمع و تفریق در این سیستم در درس معماری آموزش داده می‌شود.

۱۳- کدهای BCD (Binary Coded Decimal)

کدهای BCD ارقام مبنای ده یعنی 0 تا 9 را به صورت باینری نمایش می‌دهند. این کدها حداقل 4 بیتی هستند چون 0 تا 9 ده حالت است و با سه بیت نمی‌توان ده حالت را نشان داد (سه بیت 8 حالت دارد). برخی از کدهای BCD معروف در جدول زیر آمده‌اند:

digit	NBCD	Excess3	8421	2421
0	0000	0011	0000	0000
1	0001	0100	0111	0001
2	0010	0101	0110	0010
3	0011	0110	0101	0011
4	0100	0111	0100	0100
5	0101	1000	1011	1011
6	0110	1001	1010	1100
7	0111	1010	1001	1101
8	1000	1011	1000	1110
9	1001	1100	1111	1111

کد NBCD (Natural BCD) یا BCD طبیعی یک کد وزن‌دار (weighted) است یعنی بیت‌های آن دارای ارزش هستند. ارزش بیت سمت چپ 8، بیت دوم 4، بیت سوم 2 و بیت آخر 1 است. به این کد 8421 نیز گفته می‌شود. پس رشته‌ی 0111 در این کد $7 = 4 + 2 + 1$ می‌باشد. توجه کنید کدهای BCD با باینری متفاوت هستند و فقط ارقام 0 تا 9 را نمایش می‌دهند. مثلاً برای نمایش 68 به صورت NBCD، باید 6 و 8 را جداگانه نمایش دهیم که 01101000 می‌شود ولی همین عدد به صورت باینری 1000100 می‌باشد. کد excess3 یا افزونی 3 یک کد بی‌وزن است یعنی بیت‌های آن دارای ارزش نیستند. این کد از افزودن 3 به NBCD به دست می‌آید. مثلاً رشته 0011 اگر NBCD بود 3 را نشان می‌دهد ولی در کد افزونی سه این رشته صفر را نشان می‌دهد یا 1100 در کد افزونی 3 رقم 9 را نمایش می‌دهد. علت ساخت کد افزونی سه این است که این کد اصطلاحاً خودمکمل است.

کد خودمکمل (selfcomplement) به کدی گویند که اگر شکل دودویی آنها را مکمل یک (یعنی NOT) کنیم، شکل دهدهی آن مکمل 9 می‌شوند (یعنی از 9 کم می‌شوند). در کد افزونی سه، 0011 صفر است حال آنکه 1100 (مکمل یک 0011) نشان‌دهنده 9 می‌باشد که 9 مکمل 9 صفر است. یا 0100 رقم 1 را نشان می‌دهد و 1011 که مکمل یک 0100 می‌باشد رقم 8 که مکمل 9 رقم 1 می‌باشد را نشان می‌دهد. پس در کد خودمکمل اگر رشته دودویی ارقام 0, 1, 2, 3, 4 را

مکمل یک کنیم رشته دودویی ارقام 5,6,7,8,9 به دست می آید. کدهای خودمکمل برای انجام عمل تفریق مناسب هستند که در مدارات تفریق کننده خواهیم دید.

در جدول فوق به جز NBCD سایر کدها خودمکمل هستند.

کد 8421 یک کد وزن دار است که ارزش بیت ها از سمت چپ به ترتیب 8, 4, 2, 1- و 1- است بنابراین رشته 0111 در این کد رقم $1=1-2-4$ را نشان می دهد یا رشته ی 1111 در این کد رقم $9=1-2-4+8$ را نشان می دهد. کد 2421 نیز وزن دار است که البته در این کد برخی ارقام را به شکل های مختلف می توان ساخت. مثلاً رقم 2 را می توان به صورت 0010 یا 1000 نمایش داد ولی کدی که ما در جدول آورده ایم طوری ساخته ایم که خودمکمل نیز باشد.

مثال: حاصل جمع $01011001 + 00111000$ را در کد NBCD بیابید.

حل: جمع و تفریق در کدهای BCD مثل باینری نیست که دو عدد را زیر هم بنویسیم، بلکه همانند مبنای ده باید عمل کنیم. در واقع 00111000 یعنی عدد 38، 01011001 یعنی 59 و حاصل جمع 38 با 59 می شود 97 که به صورت NBCD به شکل 10010111 می باشد پس:

$$(00111000)_{NBCD} + (01011001)_{NBCD} = (10010111)_{NBCD}$$

که رسیدن به چنین حاصلی نیاز به جمع کننده و سخت افزار خاصی دارد که در فصل های مربوطه خواهیم دید. همانطور که گفته شد کدهای BCD حداقل 4 بیتی هستند و می توانند بیش از 4 بیتی نیز باشند. مثلاً در جدول زیر چند کد BCD که 4 بیتی نیستند نشان داده شده است.

digit	Biquinary	$\frac{1}{10}$ code	$\frac{2}{5}$ code	$\frac{2}{7}$ code
0	0100001	1000000000	01100	1010000
1	0100010	0100000000	10001	1001000
2	0100100	0010000000	10010	1000100
3	0101000	0001000000	00011	1000010
4	0110000	0000100000	10100	1000001
5	1000001	0000010000	00101	0110000
6	1000010	0000001000	0110	0101000
7	1000100	0000000100	11000	0100100
8	1001000	0000000010	01001	0100010
9	1010000	0000000001	01010	0100001

کد دوپنجی (biquinary) هفت بیتی است. برای ارقام 0 تا 4 دو بیت سمت چپ 01 است و برای ارقام 5 تا 9 دو بیت سمت چپ 10 می باشد. در پنج بیت سمت راست نیز فقط یک بیت 1 می باشد که به ترتیب 1 بودن آن بیت از سمت راست نشان دهنده ارقام 0 تا 4 یا 5 تا 9 می باشد. کد $\frac{1}{10}$ یک کد ده بیتی است که فقط یک بیت آن 1 است. کد $\frac{2}{5}$ یک کد 5 بیتی است که دو بیت آن 1 است در این کد به جز رقم صفر، سایر ارقام از وزن 07421 پیروی می کنند یعنی از سمت چپ ارزش بیت ها به ترتیب 0, 2, 4, 7, 1 می باشد. مثلاً رشته 00110 در این کد رقم $6=2+4$ می باشد یا رشته 11000 در این کد رقم $7=0+7$ می باشد. کد $\frac{2}{7}$ یک کد 7 بیتی است که دو بیت آن 1 است. تمام ارقام 0 تا 9 از وزن 0501234 پیروی می کنند یعنی این کد وزن دار است. مثلاً 0101000 در این کد رقم $6=1+5$ را نشان می دهد. کاربرد اصلی این نوع کدها که از 4 بیت بیشتر هستند، خطایابی است یعنی اگر خطایی رخ دهد و بیتی عوض شود، رشته دودویی نامعتبر می شود و می توان خطا را متوجه شد.

در پایان بحث کدهای BCD، بد نیست بدانیم که کد NBCD معروفترین کد BCD است و فقط مشکل این است که خودمکمل نیست. اما چرا این مسئله مهم است؟ در مبنای 2 برای انجام عمل $A - B$ باید A را با مکمل دوی B جمع کنیم. در مبنای 10 برای انجام این عمل باید A را با مکمل ده B جمع کنیم یعنی باید A را با مکمل نه B به علاوه 1 جمع کنیم:

$$A - B = A + (10' \text{ complement } B) = A + (9' \text{ s comp } B) + 1$$

مثلاً برای انجام تفریق 3-8 داریم:

$$8 - 3 = 8 + (9' \text{ s comp of } 3) + 1 = 8 + 6 + 1 = 15 \xrightarrow[\text{carry}]{\text{reject}} 5$$

پس باید مکمل 9 را بتوان سریع پیدا کرد که عمل تفریق به سرعت انجام شود. اگر کدی خودمکمل باشد مکمل 9 کردن آن بسیار ساده است و فقط کافی است همه بیت‌های B عوض (NOT) شوند. ولی کد NBCD خودمکمل نیست و برای آنکه B مکمل 9 شود دو روش وجود دارد:

روش ۱- ابتدا B را با 6 جمع کنید و سپس بیت‌های حاصل را NOT (مکمل یک) کنید.

روش ۲- ابتدا B را مکمل یک (NOT) کنید و سپس حاصل را با ده جمع کنید و از رقم نقلی صرف‌نظر کنید.

مثلاً می‌خواهیم $3 = 0011$ را مکمل 9 کنیم. روش ۱ می‌گوید ابتدا با 6 جمع کنیم که 1001 حاصل می‌شود، حال حاصل را مکمل یک می‌کنیم که 0110 یعنی 6 حاصل می‌شود که درست است. روش ۲ می‌گوید ابتدا مکمل یک کنیم که 1100 حاصل می‌شود و حال نتیجه را با ده جمع کنیم که 10110 به دست می‌آید و با حذف رقم نقلی 0110 به دست می‌آید.

۱۴- کد گری (gray)

در کد گری هر دو رشته متوالی فقط در یک بیت با هم تفاوت دارند. برای درک بهتر این جمله، کد باینری 3 بیتی و کد گری 3 بیتی را می‌آوریم:

value	binary	gray
0	000	000
1	001	001
2	010	011
3	011	010
4	100	110
5	101	111
6	110	101
7	111	100

در کد باینری اعداد متوالی لزوماً در یک بیت تفاوت ندارند. مثلاً $3 = 011$ با $4 = 100$ در سه بیت تفاوت دارند که اصطلاحاً گفته می‌شود فاصله‌شان سه است.

0 1 1

↕↕↕

1 0 0

یا $5 = 101$ و $6 = 110$ در دو بیت تفاوت دارند یعنی فاصله‌شان دو است.

1 0 1

↕↕

1 1 0

ولی در کد گری هر دو عدد متوالی فاصله‌شان یک است. مثلاً در کد گری $3=010$ و $4=110$ که فاصله‌شان یک است:

010

↕

110

کد گری بسیار متنوع است، معروف‌ترین آن، کد گری انعکاسی است که در جدول فوق آورده‌ایم. انعکاسی بودن این کد به خاطر نحوه ساخت آن است. برای ساخت گری $n+1$ بیتی از روی گری n بیتی کافی است، اعداد کد n بیتی را زیر هم بنویسید و سپس آنها را مجدد در ادامه بنویسید ولی به صورت تصویر آینه‌ای، سپس پشت نیمه اول رشته‌ها صفر قرار دهید و پشت نیمه دوم، یک قرار دهید (گیج شدیم!) در شکل مقابل گری 2 بیتی را از روی یک بیتی و گری سه بیتی را از روی دو بیتی ساخته‌ایم.

1 bitgray		2 bitgray		3 bitgray
0	reflex →	0	→	00
1	→	1	→	01
		1	→	11
		0	→	10
			→	10
			→	11
			→	01
			→	00

البته روش مستقیمی برای تبدیل کد باینری به کد گری (انعکاسی) وجود دارد.

برای بیان روش باید با عملی به اسم XOR (eXclusive OR) آشنا شویم. این عمل را با علامت $\oplus$ نشان می‌دهیم که دو بیت دریافت می‌کند و اگر مساوی باشند 0 می‌دهد و اگر مخالف باشد 1 تولید می‌شود، یعنی:

$$1 \oplus 0 = 1, 0 \oplus 1 = 1, 1 \oplus 1 = 0, 0 \oplus 0 = 0$$

برای تبدیل کد باینری n بیتی به گری n بیتی کافی است از سمت چپ حرکت کرده، بیت سمت چپ را عیناً در گری قرار دهیم و سپس برای یافتن بیت i ام در گری کافی است بیت i ام در باینری را با بیت سمت چپ آن در باینری XOR کنیم:

$$(b_{n-1} \ b_{n-2} \ b_{n-3} \ \dots \ b_1 \ b_0)_{\text{binary}}$$

$$\downarrow \quad \downarrow \quad \downarrow \quad \dots \quad \downarrow$$

$$= (g_{n-1} \ g_{n-2} \ g_{n-3} \ \dots \ g_1 \ g_0)_{\text{gray}}$$

$$g_{n-1} = b_{n-1}, g_{n-2} = b_{n-1} \oplus b_{n-2}, g_{n-3} = b_{n-2} \oplus b_{n-3}, \dots, g_0 = b_1 \oplus b_0$$

مثال: عدد 91 را با هشت بیت در کد گری نمایش دهید.

حل: ابتدا 91 را باینری می‌کنیم که $91 = 64 + 16 + 8 + 2 + 1$ پس: $91 = (01011011)_{\text{binary}}$ حال طبق فرمول گفته شده به گری تبدیل می‌کنیم:

$$b_7 = 0, b_6 = 1, b_5 = 0, b_4 = 1, b_3 = 1, b_2 = 0, b_1 = 1, b_0 = 1$$

$$g_7 = b_7 = 0, g_6 = b_7 \oplus b_6 = 1, g_5 = b_6 \oplus b_5 = 1, g_4 = b_5 \oplus b_4 = 1,$$

$$g_3 = b_4 \oplus b_3 = 0, g_2 = b_3 \oplus b_2 = 1, g_1 = b_2 \oplus b_1 = 1, g_0 = b_1 \oplus b_0 = 0$$

$$\Rightarrow 91 = (01110110)_{\text{gray}}$$

فرمولی نیز برای تبدیل گری به باینری وجود دارد به شکل زیر است:

$$(g_{n-1} \ g_{n-2} \ g_{n-3} \ \dots \ g_1 \ g_0)_{\text{gray}}$$

$$= (b_{n-1} \ b_{n-2} \ b_{n-3} \ \dots \ b_1 \ b_0)_{\text{binary}}$$

$$b_{n-1} = g_{n-1}, \ b_{n-2} = b_{n-1} \oplus g_{n-2}, \ b_{n-3} = b_{n-2} \oplus g_{n-3}, \dots, \ b_0 = b_1 \oplus g_0$$

البته این فرمول‌ها را می‌توان از فرمول‌های قبلی به دست آورد. در فصل بعدی خواهیم دید که XOR خواصی دارد از جمله اینکه اگر $a = b \oplus c$ آنگاه $b = a \oplus c$ و $c = a \oplus b$ ، داشتیم $g_{n-2} = b_{n-1} \oplus b_{n-2}$ بنابراین $b_{n-2} = b_{n-1} \oplus g_{n-2}$ و به همین ترتیب. به عنوان مثال با این فرمول‌ها اگر $(01110110)_{\text{gray}}$ را به باینری تبدیل کنید $(01011011)_{\text{binary}}$ حاصل می‌شود.

	gray
0	001
1	101
2	111
3	110
4	100
5	000
6	010
7	011

همانطور که گفتیم کد گری تنوع زیادی دارد و هر کدی که هر دو عدد متوالی آن فاصله یک داشته باشند کد گری است. مثلاً جدول مقابل یک گری 3 بیتی را نشان می‌دهد.

کد گری را می‌توان برای BCD نیز به کار برد یعنی فقط 0 تا 9 را نمایش داد که به آن BCD gray گویند. جدول مقابل دو کد BCD gray را نشان می‌دهد که اولی انعکاسی است و دومی excess BCD gray می‌باشد. علت اینکه به کد گری ستون آخر افزونی سه گفته می‌شود این است که 0010 را که در کد انعکاسی به رقم 3 نسبت داده‌ایم در این کد به رقم 0 نسبت داده‌ایم.

digit	reflexive gray	ex3 gray
0	0000	0010
1	0001	0110
2	0011	0111
3	0010	0101
4	0110	0100
5	0111	1100
6	0101	1101
7	0100	1111
8	1100	1110
9	1101	1010

۱۵- کدهای تشخیص و تصحیح خطا

فرض کنید کد باینری 3 بیتی داریم که 0 تا 7 را به صورت 000, 001, 010, 011, 100, 101, 110, 111 نمایش می‌دهد. حال می‌خواهیم مثلاً عدد 3 یعنی رشته 011 را برای یک گیرنده ارسال کنیم یا از حافظه بخوانیم. اگر نویز یا خطا ایجاد شود و مثلاً بیت سمت راست عوض شود گیرنده رشته 010 را دریافت می‌کند ولی متوجه نمی‌شود که این رشته خطاست چون 010 نیز معتبر است و گیرنده فکر می‌کند هدف ما واقعاً ارسال 010 بوده است. حال فرض کنید ما برای گیرنده، اعداد

را با کد NBCD ارسال می‌کنیم که شامل رشته‌های 0000 تا 1001 می‌باشد و می‌خواهیم رشته 0111 را ارسال کنیم. اگر بیت سمت چپ خطا شود گیرنده 1111 دریافت می‌کند و متوجه می‌شود غلط است چون این رشته در کد NBCD نامعتبر است. به طور کلی برای آنکه قابلیت تشخیص خطا وجود داشته باشد باید هم رشته‌های معتبر و هم رشته‌های نامعتبر وجود داشته باشند یعنی مثلاً با n بیت که 2^n رشته دودویی وجود دارد ما تمام این 2^n رشته را استفاده نکنیم و تعدادی را نامعتبر در نظر بگیریم که اگر خطا باعث شد رشته‌های نامعتبر ایجاد شود، خطا را تشخیص دهیم. اکنون سؤال این است که اگر ما فرض کنیم فقط یک بیت ممکن است خطا شود و نه بیشتر، و بخواهیم این خطا را تشخیص دهیم، کد باید چه شرطی داشته باشد. در کد NBCD لزوماً نمی‌توان یک بیت خطا را تشخیص داد مثلاً اگر رشته‌ی 0011 را بخواهیم ارسال کنیم و بیت سمت راست خطا شود آنگاه گیرنده رشته 0010 را دریافت می‌کند که خطا قابل تشخیص نیست چون 0010 یک رشته معتبر است.

فاصله بین دو عدد تعداد اختلاف بیت‌های آن دو عدد است. مثلاً دو عدد 0110 و 1010 دارای فاصله 2 هستند. فاصله در یک کد برابر حداقل فاصله بین اعداد آن کد است. تمام کدهایی که تاکنون معرفی کرده‌ایم فاصله یک دارند و در چنین کدهایی در حالت کلی نمی‌توان یک خطا را تشخیص داد مثلاً دو عدد 0110 و 1110 دارای فاصله یک هستند و اگر بیت سمت چپ عوض شود به هم تبدیل می‌شوند. برای آنکه بتوان یک خطا را تشخیص داد باید فاصله کد حداقل 2 باشد یعنی هر دو عدد در کد حداقل در 2 بیت تفاوت داشته باشند.

برای آنکه بتوان فاصله کد را از یک به دو افزایش داد کافی است یک بیت موسوم به توان (parity) به اعداد آن کد اضافه کنیم. بیت توازن دو نوع زوج (even) و فرد (odd) است. بیت توازن زوج طوری به هر عدد اضافه می‌شود که تعداد یک‌ها در هر عدد زوج شود. مثلاً در جدول مقابل به داده‌های باینری سه بیتی $b_2b_1b_0$ بیت توازن زوج p_e اضافه کرده‌ایم. حال به جای آنکه سه بیت $b_2b_1b_0$ را برای گیرنده ارسال کنیم 4 بیت $b_2b_1b_0p_e$ را ارسال می‌کنیم و اگر یک بیت خطا شود آنگاه گیرنده تشخیص می‌دهد که خطاست چون با عوض شدن یک بیت تعداد یک‌ها فرد می‌شود که نامعتبر است. اگر توجه کنید فاصله کد $b_2b_1b_0p_e$ برابر دو است و یک خطا را می‌توان تشخیص داد.

b_2	b_1	b_0	p_e
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

ولی با افزودن بیت توازن و تولید کدی با فاصله دو فقط می‌توان خطا در یک بیت را تشخیص داد ولی نمی‌توان فهمید کدام بیت اشتباه شده است و در واقع نمی‌توان خطا را تصحیح کرد. مثلاً اگر گیرنده‌ای رشته 0111 را دریافت کند متوجه می‌شود خطاست چون تعداد 1ها فرد است ولی متوجه نمی‌شود رشته درست ارسالی چه بوده است (رشته ارسالی می‌تواند هر یک از رشته‌های 1111 یا 0011 یا 0101 یا 0110 باشد) برای آنکه کدی قابلیت تصحیح خطا داشته باشد یعنی بتوان فهمید کدام بیت اشتباه شده است باید کدی با فاصله 3 بسازیم. یعنی هر دو عدد در کد باید حداقل در 3 بیت تفاوت داشته باشند (چرا؟) برای ساخت کد با فاصله 3 باید بیشتر از یک بیت به هر عدد اضافه کنیم. ثابت خواهیم کرد اگر اعداد 4 بیتی باشند باید 3 بیت اضافه کنیم و 7 بیت برای گیرنده ارسال کنیم و اگر اعداد 8 بیتی باشند باید 4 بیت اضافه کنیم.

معروفترین کد که قابلیت تصحیح یک بیت خطا را دارد کد همینگ است. یعنی اگر در یک بیت خطا به وجود آید، گیرنده متوجه می‌شود و آن بیت را عوض می‌کند. فرض کنید اعداد اولیه ما 4 بیتی هستند، 3 بیت به این داده‌ها اضافه می‌کنیم و 7 بیت برای گیرنده ارسال می‌کنیم. این 7 بیت را شماره‌گذاری می‌کنیم به صورت

1	2	3	4	5	6	7

داده اولیه را $b_3b_5b_6b_7$ می‌نامیم و 3 بیت اضافه شده را $c_1c_2c_4$ نام‌گذاری می‌کنیم و 7 بیت ارسالی $c_1c_2b_3c_4b_5b_6b_7$ می‌باشد در واقع در کد همینگ بیت‌های اضافه شده در مکان‌های 4, 2, 1 (مکان‌هایی که شماره آنها توانی از 2 است) قرار می‌گیرند. بیت‌های اضافه شده c_4, c_2, c_1 با فرمول‌های زیر به دست می‌آیند.

$$c_1 = b_3 \oplus b_5 \oplus b_7 = \text{XOR}(3, 5, 7)$$

$$c_2 = b_3 \oplus b_6 \oplus b_7 = \text{XOR}(3, 6, 7)$$

$$c_4 = b_5 \oplus b_6 \oplus b_7 = \text{XOR}(5, 6, 7)$$

عمل XOR روی چند بیت به این صورت عمل می‌کند که اگر تعداد 1ها فرد باشد خروجی 1 می‌شود.

$$\text{مثلاً } 1 \oplus 0 \oplus 1 = 0 \text{ و } 1 \oplus 1 \oplus 1 = 1.$$

برای حفظ فرمول‌های فوق توجه کنید که اگر شماره بیت‌های داده اصلی یعنی 3, 5, 6, 7 را به صورت جمع 1, 2, 4 بنویسید یعنی $3 = 2 + 1$, $5 = 4 + 1$, $6 = 4 + 2$, $7 = 4 + 2 + 1$ و حال شماره‌های 7, 5, 3 که عامل 1 دارند سازنده c_1 هستند. 3, 6, 7 که عامل 2 دارند سازنده c_2 هستند و 5, 6, 7 که عامل 4 دارند سازنده c_4 هستند.

برای درک این فرمول‌ها فرض کنید می‌خواهیم داده $b_3b_5b_6b_7 = 0101$ را برای گیرنده ارسال کنیم، c_4, c_2, c_1 را محاسبه کرده به داده اضافه می‌کنیم:

$$c_1 = \text{XOR}(0, 1, 1) = 0, \quad c_2 = \text{XOR}(0, 0, 1) = 1, \quad c_4 = \text{XOR}(1, 0, 1) = 0$$

پس داده ارسالی به گیرنده $c_1c_2b_3c_4b_5b_6b_7 = 0100101$ می‌باشد. حال گیرنده با کمک فرمول‌های زیر در صورت وجود خطا، محل خطا را می‌فهمد و آن را درست می‌کند:

$$d_1 = \text{XOR}(1, 3, 5, 7), \quad d_2 = \text{XOR}(2, 3, 6, 7), \quad d_4 = \text{XOR}(4, 5, 6, 7)$$

این فرمول‌ها همان فرمول‌های فرستنده است فقط خود بیت حاصل یعنی c ها نیز به داخل XOR اضافه شده است. گیرنده $d_4d_2d_1$ را به عنوان یک عدد 3 بیتی می‌بیند که شماره خطا را مشخص می‌کند. اگر $d_4d_2d_1 = 000$ یعنی خطایی وجود ندارد اگر $d_4d_2d_1 = 001$ یعنی بیت شماره 1 یعنی c_1 خطا شده است. اگر $d_4d_2d_1 = 010$ یعنی بیت شماره 2 یا c_2 اشتباه است. اگر $d_4d_2d_1 = 011$ یعنی بیت شماره 3 یا b_3 اشتباه است و باید عوض شود و به همین ترتیب اگر $d_4d_2d_1 = 111$ یعنی بیت شماره 7 اشتباه است و باید عوض شود.

به عنوان مثال فرض کنید فرستنده داده 7 بیتی 0100101 را ارسال کرده ولی گیرنده 0100111 دریافت کرده است. با توجه به فرمول‌های d_4, d_2, d_1 داریم:

$$d_1 = \text{XOR}(0, 0, 1, 1) = 0, \quad d_2 = \text{XOR}(1, 0, 1, 1) = 1, \quad d_4 = \text{XOR}(0, 1, 1, 1) = 1$$

یعنی $d_4d_2d_1 = 110$ که یعنی بیت 6 خطاست و باید عوض شود که اگر این بیت را عوض کنیم همان داده ارسالی فرستنده به دست می‌آید.

حال بیا برای داده‌های 8 بیتی کد همینگ بسازیم. باید 4 بیت اضافه کنیم و 12 بیت ارسال کنیم. 4 بیت اضافه شده را در مکان‌های 8, 4, 2, 1 قرار داده و با فرمول‌های زیر به دست می‌آوریم:

$$c_1 = \text{XOR}(3, 5, 7, 9, 11), \quad c_2 = \text{XOR}(3, 6, 7, 10, 11)$$

$$c_4 = \text{XOR}(5, 6, 7, 12), \quad c_8 = \text{XOR}(9, 10, 11, 12)$$

1	2	3	4	5	6	7	8	9	10	11	12
C_1	C_2		C_4				C_8				

مثلاً برای ارسال داده 8 بیتی $b_3b_5b_6b_7b_9b_{10}b_{11}b_{12} = 10001101$ مقادیر اضافه شده عبارتند از:

$$c_1 = \text{XOR}(1, 0, 0, 1, 0) = 0, \quad c_2 = \text{XOR}(1, 0, 0, 1, 0) = 0$$

$$c_4 = \text{XOR}(0, 0, 0, 1) = 1, \quad c_8 = \text{XOR}(1, 1, 0, 1) = 1$$

پس دوازده بیت ارسالی 001100011101 می باشد. حال گیرنده با فرمول های زیر در صورتی که در یک بیت خطا ایجاد شود می فهمد و تصحیح می کند:

$$d_1 = \text{XOR}(1, 3, 5, 7, 9, 11), \quad d_2 = \text{XOR}(2, 3, 6, 7, 10, 11)$$

$$d_4 = \text{XOR}(4, 5, 6, 7, 12), \quad d_8 = \text{XOR}(8, 9, 10, 11, 12)$$

عدد $d_8d_4d_2d_1$ اگر صفر باشد یعنی عدم وجود خطا و اگر 0001 تا 1100 باشد نشان می دهد کدام یک از بیت های 1 تا 12 خطاست.

به عنوان مثال فرض کنید ما 12 بیت 001100011101 را برای گیرنده ارسال کرده ایم و گیرنده 001110011101 را دریافت کرده است. با توجه به فرمول های d_8, d_4, d_2, d_1 :

$$\left. \begin{aligned} d_1 &= \text{XOR}(0, 1, 1, 0, 1, 0) = 1 \\ d_2 &= \text{XOR}(0, 1, 0, 0, 1, 0) = 0 \\ d_4 &= \text{XOR}(1, 1, 0, 0, 1) = 1 \\ d_8 &= \text{XOR}(1, 1, 1, 0, 1) = 0 \end{aligned} \right\} \Rightarrow d_8d_4d_2d_1 = 0101$$

پس بیت 5 خطاست که باید 0 باشد که به همان داده ارسالی توسط فرستنده می رسیم.

اکنون می خواهیم به این سؤال جواب دهیم که به داده های n بیتی باید چند بیت اضافه کنیم که بتوان یک خطا را تصحیح کرد. در حالت داده های 4 بیتی ما 3 بیت اضافه کردیم و 7 بیت ارسال کردیم و این 3 بیت اضافه شده برای گیرنده 8 حالت درست کرد که حالت 000 یعنی عدم وجود خطا و 7 حالت دیگر نشان داد کدام خانه خطاست. یا برای داده های 8 بیتی 4 بیت اضافه کردیم و 12 بیت ارسال کردیم که 4 بیت اضافه شده 16 حالت دارد و حالت 0000 یعنی عدم وجود خطا و از 15 حالت دیگر 12 حالت آن یعنی از 0001 تا 1100 نشان می دهد کدام بیت ارسالی اشتباه شده است. به طور کلی برای داده های n بیتی ما k بیت اضافه می کنیم و $n+k$ بیت ارسال می کنیم. k بیت اضافه شده برای گیرنده 2^k حالت درست می کند که یک حالت آن نشان دهنده عدم وجود خطاست و $2^k - 1$ حالت دیگر باید نشان دهد کدام یک از $n+k$ بیت ارسالی خطا شده است که یعنی باید $2^k - 1 \geq n+k$ و از این نامساوی می توان با داشتن n ، حداقل مقدار k را به دست آورد.

مثال: به داده های 27 بیتی حداقل چند بیت اضافه کنیم که بتوان یک بیت خطا را تشخیص داد و تصحیح کرد؟

حل: $n = 27$ و به دنبال حداقل k هستیم، با توجه به $2^k - 1 \geq 27 + k$ پس $2^k - k \geq 28$ که $k = 6$ جواب است. پس باید حداقل 6 بیت به 27 اضافه کنیم.

مثال: به داده های n بیتی 5 بیت اضافه کرده ایم که یک خطا را تصحیح کنیم. مقادیر n را بیابید.

حل: $k = 5$ و با توجه به رابطه $2^5 - 1 \geq n + 5$ نتیجه می شود که $n \leq 26$. پس n حداکثر 26 بیت می باشد. برای یافتن حداقل n کافی است بررسی کنیم به ازای $k = 4$ ، حداکثر n چند می باشد:

$2^4 - 1 \geq n + 4$ که $n \leq 11$ پس به ازای $n = 12$ تا $n = 26$ بیت باید حداقل 5 بیت اضافه کنیم که یک بیت خطا را تصحیح کنیم.

ما تاکنون فرض کردیم احتمال وجود خطا در یک بیت وجود دارد و نه بیشتر. ولی اگر احتمال وقوع خطا در یک یا دو بیت وجود داشته باشد آنگاه به داده های n بیتی باید حداقل چند بیت اضافه کنیم که بتوان خطا در یک بیت یا دو بیت را تصحیح کند؟ فرض کنید به n بیت k بیت اضافه کرده ایم و $n+k$ بیت ارسال کرده ایم. k بیت اضافه شده برای گیرنده 2^k حالت ایجاد می کند که یک حالت آن عدم وجود خطاست و $2^k - 1$ حالت دیگر باید بتواند محل خطا در هر یک از $n+k$

بیت یا در هر دو بیت از $n+k$ بیت را نشان دهند که تعداد حالات دو بیت از $n+k$ بیت برابر ترکیب $\binom{n+k}{2}$ می باشد پس باید:

$$2^k - 1 \geq n+k + \binom{n+k}{2}$$

یادآوری می کنیم که ترکیب m از n یعنی $\binom{n}{m}$ برابر $\frac{n!}{m!(n-m)!}$ می باشد و به طور خاص $\binom{n}{2}$ برابر $\frac{n(n-1)}{2}$ می باشد.

مثال: به داده های 4 بیتی، حداقل چند بیت اضافه کنیم که بتوان خطا در یک یا دو بیت را تصحیح کرد؟

حل: $n=4$ و به دنبال حداقل k هستیم:

$$2^k - 1 \geq 4+k + \binom{4+k}{2}$$

برای یافتن k باید آزمایش و خطا کنیم که به $k=6$ می رسیم:

$$2^6 - 1 \geq 4+6 + \binom{4+6}{2} \Rightarrow 63 \geq 10+45$$

در پایان خوب است بدانید که اگر فاصله در کدی برابر d باشد (یعنی هر دو عدد در کد حداقل در d بیت تفاوت داشته باشند) آنگاه می توان حداکثر $d-1$ خطا را تشخیص داد (ولی نمی توان تصحیح کرد) و می توان حداکثر $\left\lfloor \frac{d-1}{2} \right\rfloor$ خطا را تصحیح کرد.

مثلاً اگر کدی با فاصله 5 بسازیم آنگاه تا 4 بیت خطا را می توان تشخیص داد. با فاصله 5 تا حداکثر $\left\lfloor \frac{5-1}{2} \right\rfloor = 2$ بیت خطا را می توان تصحیح کرد.

خلاصه فصل

۱- در مبنای r از ارقام $0, 1, \dots, r-1$ استفاده می شود. برای نشان دادن ارقامی که از دو رقم تشکیل شده اند یعنی $10, 11, 12, \dots$ از حروف $A, B, C, \dots$ استفاده می شود.

۲- برای تبدیل یک عدد به مبنای 10 کافی است هر رقم را در ارزش آن رقم ضرب کنیم و مقادیر به دست آمده را جمع کنیم. در عدد $(a_{n-1}a_{n-2}\dots a_1a_0.a_{-1}a_{-2}\dots a_{-m})_r$ ارزش رقم a_i برابر r^i می باشد.

۳- برای تبدیل یک عدد از مبنای 10 به مبنای r کافی است قسمت صحیح را به r تقسیم و خارج قسمت های به دست آمده را مجدد به r تقسیم کنیم تا وقتی به خارج قسمت صفر برسیم و سپس از آخرین باقی مانده، باقی مانده ها را کنار هم می نویسیم (آخرین باقی مانده رقم با ارزش حاصل می باشد). قسمت اعشار را نیز در r ضرب می کنیم و قسمت صحیح حاصل ضرب را به عنوان اولین رقم بعد از ممیز در جواب یادداشت می کنیم و سپس قسمت اعشار حاصل ضرب را در r ضرب می کنیم و قسمت صحیح حاصل ضرب را یادداشت می کنیم تا وقتی که قسمت اعشار صفر شود (مختوم) یا به یک تکرار برسد (متناوب).

۴- مبنای r و r^x را می توان مستقیم به هم تبدیل کرد. برای تبدیل از مبنای r^x به r به جای هر رقم باید x رقم در مبنای r قرار دهیم. برای تبدیل از مبنای r به r^x به جای هر x رقم (قسمت صحیح از راست و قسمت اعشار از چپ به دسته های x تایی تفکیک شوند) باید یک رقم در مبنای r^x قرار دهیم.

۵- برای نمایش عدد x در مبنای r نیاز به $\lceil \log_r(x+1) \rceil = \lfloor \log_r x \rfloor + 1$ رقم است.

۶- برای نمایش اعداد n رقمی مبنای t در مبنای r نیاز به $\lceil n \cdot \log_r t \rceil$ رقم در مبنای r است.

۷- در مبنای r برای عدد a دو مکمل تعریف شده است. مکمل $r-1$ یا مکمل کاهش یافته که

برابر $a[r-1]_r = r^n - r^m - a$ می باشد و مکمل r یا مکمل مبنای که برابر $[a]_r = r^n - a$ می باشد. برای یافتن مکمل $r-1$

می‌توان سریع‌تر عمل کرد و همه ارقام را از $r-1$ کم کرد. برای یافتن مکمل r ، صفرهای سمت راست عدد تغییر نمی‌کنند. اولین رقم غیرصفر از r و سایر ارقام از $r-1$ کم می‌شوند.

۸- برای نمایش اعداد علامت‌دار در مبنای ۲ روش‌های مختلفی هست از جمله، علامت مقدار، مکمل یک و مکمل دو. در این سه روش ارزش بیت‌ها مثل سیستم بی‌علامت است یعنی بیت سمت راست (عدد صحیح است) دارای ارزش 2^0 و بیت بعدی 2^1 ، ... می‌باشد. فقط بیت سمت چپ با n بیت در سیستم بی‌علامت دارای ارزش 2^{n-1} است و در سیستم‌ها مکمل دو دارای ارزش 2^{n-1} است و در سیستم مکمل یک دارای ارزش $(2^{n-1}-1)$ است و در سیستم علامت مقدار این بیت دارای ارزش نیست و فقط علامت را نشان می‌دهد.

۹- در کامپیوترها وقتی عملی مثل جمع یا تفریق انجام می‌شود، فلگ‌های S, Z, V, C مقداردهی می‌شوند. اگر حاصل عملیات صفر باشد $Z=1$ ، اگر حاصل منفی باشد یعنی بیت چپ حاصل یک باشد $S=1$ ، اگر حاصل در بازه مجاز نباشد $V=1$ ، اگر از بیت سمت چپ رقم نقلی خارج شود $C=1$.

۱۰- در سیستم مکمل ۲ در عمل جمع $A+B$ شرط لازم و کافی برای وقوع سرریز آن است که A و B هم علامت باشند ولی علامت حاصل متفاوت باشد، یا رقم نقلی وارد شده به بیت سمت چپ با رقم نقلی خروجی از آن بیت مساوی نباشد.

۱۱- در سیستم مکمل ۲ در عمل تفریق $A-B$ که می‌توان با جمع $A+\bar{B}+1$ تبدیل کرد، شرط لازم و کافی برای $A \geq B$ آن است که $s \oplus v = 0$ و برای $A < B$ آن است که $s \oplus v = 1$ و برای $A=B$ آن است که $Z=1$ و برای $A \neq B$ آن است که $Z=0$.

۱۲- در سیستم مکمل یک در عمل جمع $A+B$ رقم نقلی خروجی باید با حاصل مجدد جمع شود. برای انجام تفریق $A-B$ می‌توان به جمع تبدیل کرد که $A+\bar{B}$ می‌شود. نکات مربوط به فلگ‌ها در این سیستم مشابه سیستم مکمل دو است.

۱۳- در سیستم بی‌علامت اگر دو عدد n بیتی با هم جمع شوند و حاصل در n بیت ذخیره شود آنگاه اگر $C=1$ یعنی سرریز رخ داده است. برای انجام تفریق $A-B$ می‌توان به جمع $A+\bar{B}+1$ تبدیل کرد که شرط لازم و کافی برای $A \geq B$ آن است که $C=1$ و برای $A < B$ آن است که $C=0$. البته اگر عمل $A-B$ را با قرض انجام دهیم شرط لازم و کافی برای $A \geq B$ آن است که $C=0$ و برای $A < B$ آن است که $C=1$.

۱۴- کدهای BCD ارقام ۰ تا ۹ را به صورت دودویی نمایش می‌دهند. کدهای 2421 ، 8421 و $excess3$ خود مکمل هستند یعنی اگر شکل دودویی آنها را مکمل یک کنیم شکل‌دهدهی آنها مکمل ۹ می‌شود. کد NBCD یا 8421 خودمکمل نیست و برای یافتن مکمل ۹ در این کد یک روش این است که ابتدا با ۶ جمع کنیم و حاصل را مکمل یک کنیم. یک روش این است که ابتدا مکمل یک کنیم و حاصل را با ده جمع کنیم و از رقم نقلی صرف‌نظر کنیم. اصولاً کد وزنداری که جمع وزن‌های آن ۹ نیست نمی‌تواند خود مکمل باشد.

۱۵- کد گری دارای این خاصیت است که هر دو عدد متوالی در این کد فقط در یک بیت تفاوت دارند. معروف‌ترین گری، گری انعکاسی است. برای تبدیل یک کد باینری به گری کافی است بیت i ام در باینری را با بیت سمت چپ آن در باینری XOR کنیم و بیت i ام در گری به دست می‌آید.

۱۶- برای آنکه در یک کد بتوان وقوع خطا را تشخیص داد باید فاصله بین اعداد کد حداقل ۲ باشد. کدی که فاصله دو دارد قابلیت تشخیص یک بیت خطا را دارد. کدی که فاصله d دارد قابلیت تشخیص حداکثر $d-1$ بیت خطا را دارد.

۱۷- برای آنکه در یک کد بتوان وقوع خطا در یک بیت را تشخیص داد و آن را تصحیح کرد فاصله کد باید حداقل ۳ باشد. کدی که فاصله ۳ دارد قابلیت تصحیح یک بیت خطا را دارد. کدی که فاصله d دارد قابلیت تصحیح حداکثر $\left\lfloor \frac{d-1}{2} \right\rfloor$ بیت خطا را دارد.

۱۸- برای آنکه در یک کد n بیتی که فاصله یک دارد قابلیت تصحیح یک خطا را ایجاد کنیم باید k بیت اضافه کنیم که حداقل k از رابطه $2^k - 1 \geq n + k$ به دست می‌آید.

فصل دوم

جبر بول
