SISTEMAS DIGITAIS FOLHA 2

SIST DE NUMERAÇÃO E CÓDIGOS – REPRESENTAÇÃO COMPLEMENTO PARA A BASE

---

(1). Converter os seguintes números decimais representados em sinal-amplitude para a representação de complemento para a base (10)

(a) +123 (b) −48 (c) −323 (d) −2047

(2) Repetir (1) utilizando o passo intermédio de representar para a base diminuída de 1 unidade

(3). Converter os seguintes números binários representados em sinal-amplitude para a representação de complemento para a base (2).

(a) 0001 0001 (b) 1011 0111

(c) 1000 0110 (d) 1000 1111

(4) Repetir (3) utilizando a passagem intermédia para a base diminuída de 1 unidade

---

(5) Fazer as operações aritméticas seguintes em base 10, utilizando a representação de complemento para a base

(a) 423 − 198 (b) 327 − 432

(6) O mesmo que (5), agora em base 2 e utilizando a representação de complemento para 2

(a) $(79-42)_{10} = (0100\ 1111 - 0010\ 1010)_2$

(b) $(87-99)_{10} = (0101\ 0111 - 0110\ 0011)_2$

(7) Fazer as operações seguintes utilizando o código BCD natural para representar os números decimais

(a) 79 + 151   (b) 87 + 179   (c) 98 − 43

(8) Converter os seguintes códigos de Gray para binário e vice-versa.

(a) 1010   (b) 11011   (c) 1100 0010 001

---

(9) [OPCIONAL] Fazer a multiplicação dos seguintes números binários representados em sinal-magnitude com 4 e 8 bits respectivamente.

(a) 1110 X 1101   (b) 00010101 X 00001010

---

(10) [OPCIONAL]. Converter para um número binário de 32 bits em representação complemento para base 2.

(a) $(512)_{10}$ (b) $(-1023)_{10}$ (c) $(-9\,000\,000)_{10}$

(11) [OPCIONAL]. Converter para decimal os seguintes números binários em representação de complemento para base 2

(a) 1111 1111 1111 1111 1111 1111 0000 1100

(b) 1111 1111 1111 1111 1111 1111 1111 1111

(c) 0111 1111 1111 1111 1111 1111111111 1111

SISTEMAS DIGITAIS FOLHA 2 SOLUÇÕES

1) O complemento para a base de um número decimal é definido pela equação

$$[X]_{10} = 10^n - X$$

onde n é o número de dígitos de X

1a) +123 é positivo; a representação em complemento para a base é
resultado = 123 (desaparece o sinal)

1b) $[48]_{10} = 10^2 - 48 \longrightarrow$

$$\begin{array}{r} 100 \\ -48 \\ 1 \\ \hline 52 \end{array} \longleftarrow \text{EMPRÉSTIMO}$$

resultado = 52

1c) $[323]_{10} = 10^3 - 323 \longrightarrow$

$$\begin{array}{r} 1000 \\ -323 \\ 11\ \\ \hline 677 \end{array}$$

resultado = 677

1d) $[2047]_{10} = 10^4 - 2047 \longrightarrow$

$$\begin{array}{r} 10000 \\ -2047 \\ 111\ \\ \hline 7953 \end{array}$$

resultado = 7953

2) O complemento para a base diminuída de 1 unidade de um número decimal é definido pela equação

$$[X]_9 = 10^n - 1 - X \quad , \quad \text{Logo } [X]_{10} = [X]_9 + 1$$

2a)
resultado = 123

2b) $[48]_9 = 99 - 48 \longrightarrow$

$$\begin{array}{r} 99 \\ -48 \\ \hline 51 \end{array}$$

$[48]_{10} = 51 + 1 = 52$

2c) $[323]_9 = 999 - 323 \longrightarrow$

$$\begin{array}{r} 999 \\ -\ 323 \\ \hline 676 \end{array}$$

$[323]_{10} = 676 + 1 = 677$

2d) $[2047]_9 = 9999 - 2047 \longrightarrow$

$$\begin{array}{r} 9999 \\ -\ 2047 \\ \hline 7952 \end{array}$$

$[2047]_{10} = 7952 + 1 = 7953$

3) em base 2

$[X]_2 = 2^n - X$

$[X]_1 = 2^n - 1 - X$ com n = número de bits de X

3a) $\underline{0}|001\ 0001$

↑ bit de sinal = , logo número positivo

resultado = 0001 0001

3b) $\underline{1}|\overbrace{011\ 0111}^{n=7}$

↑ bit de sinal

$[011\ 0111]_2 = 2^7 - 011\ 0111 \longrightarrow$

$$\begin{array}{r} 1\,0000000 \\ -\ 0110111 \\ 111111 \\ \hline 1001001 \end{array}$$

resultado = 11 001001

↑ extensão para 8 bits

3c) $\underline{1}|\underbrace{000\ 0110}_{n=7}$

↑ sinal

$[000\ 0110]_2 = 2^7 - 000\ 0110 \longrightarrow$

$$\begin{array}{r} 1\ 000\ 0000 \\ -\ 000\ 0110 \\ 111\ 11 \\ \hline 111\ 1010 \end{array}$$

resultado = 1111 1010

↑ extensão para 8 bits

3d) $1|\underbrace{000\ 1111}_{n=7}$

$[000\ 1111]_2 = 2^7 - 000\ 1111 \longrightarrow$

$$
\begin{array}{r}
1000\ 0000 \\
-\ 000\ 1111 \\
\hline
111\ 0001
\end{array}
$$

resultado = 1111 0001

extensão para 8 bits

4a) resultado = 0001 0001

4b) 1|011 0111

$[011\ 0111]_1 = 2^7 - 1 - 011\ 0111 \longrightarrow$

$$
\begin{array}{r}
111\ 1111 \\
-\ 011\ 0111 \\
\hline
\end{array}
$$

NOTA: A operação é equivalente a inverter os bits! $\longrightarrow$ 100 1000

$$
\begin{array}{r}
[011\ 0111]_2 = 100\ 1000 \\
+1 \\
\hline
1100\ 1001
\end{array}
$$

extensão para 8 bits

4c) 1|000 0110

$$
\begin{array}{r}
[000\ 0110]_1 = 111\ 1001 \\
+1 \\
\hline
\end{array}
$$

$[000\ 0110]_2 = 111\ 1010$

resultado = 1111 1010

extensão para 8 bits

5a) Forma habitual:

$$\begin{array}{r} 423 \\ -198 \\ {\scriptstyle 11} \\ \hline 225 \end{array}$$

Complemento para base 10:

$[198]_{10} = 10^3 - 198 = 802$

$$\begin{array}{r} 423 \\ +802 \\ \hline 1\,225 \end{array}$$

ignorar (overflow)

5b) Forma habitual

$$\begin{array}{r} 327 \\ -432 \\ \hline -105 \end{array}$$

Complemento para base 10

$[432]_{10} = 10^3 - 432 = 568$

$$\begin{array}{r} 327 \\ +568 \\ \hline 895 \end{array}$$

$895 > \frac{10^3}{2} (=500)$

logo é um número negativo;

Para se saber o seu valor absoluto é necessário fazer o complemento para a base

$[895]_{10} = 10^3 - 895 = 105$

6a) Forma habitual

$$\begin{array}{r} 0100\ 1111 \\ -0010\ 1010 \\ {\scriptstyle 1} \\ \hline 0010\ 0101 \end{array}$$

Complemento para base 2

$$\begin{array}{r} [0010\ 1010]_2 = 1101\ 0101 \\ +1 \\ \hline 1101\ 0110 \end{array}$$

$$\begin{array}{r} {\scriptstyle 1\ \ 11\ \ 11} \\ 0100\ 1111 \\ +1101\ 0110 \\ \hline 1\,0010\ 0101 \end{array}$$

Ignorar (overflow)

6b) Forma habitual

$$
\begin{array}{r}
0110\ 0011 \\
-\ 0101\ 0111 \\
\hline
-\ 0000\ 1100
\end{array}
$$

Complemento para 2

$$[0110\ 0011]_2 = 1001\ 1100$$

$$
\begin{array}{r}
1001\ 1100 \\
+1 \\
\hline
1001\ 1101
\end{array}
$$

$$
\begin{array}{r}
0101\ 0111 \\
+\ 1001\ 1101 \\
\hline
1111\ 0100
\end{array}
$$

nº negativo; para se saber o valor absoluto calcular o complemento para a base

$$[1111\ 0100]_2 = 0000\ 1011$$

$$
\begin{array}{r}
0000\ 1011 \\
+1 \\
\hline
0000\ 1100
\end{array}
$$

7a)

$$(79)_{10} = (\overbrace{0111}^{7}\ \overbrace{1001}^{9})_{BCD}$$

$$(101)_{10} = (\overbrace{0001}^{1}\ \overbrace{0000}^{0}\ \overbrace{0001}^{1})_{BCD}$$

$$79 + 101 = 180$$

$$(180)_{10} = (0001\ 1000\ 0000)_{BCD}$$

7b)

$$(87)_{10} = (1000\ 0111)_{BCD}$$

$$(179)_{10} = (0001\ 0111\ 1001)_{BCD}$$

$$87 + 179 = 266$$

$$(266)_{10} = (0010\ 0110\ 0110)_{BCD}$$

1)

algoritmo binário → gray

① Manter MSB

② Da esq para a dir somar os 2 bits adjacentes (esquecer o transporte)

(a) 1111 (b) 10110 (c) 10100011001

algoritmo gray → binário

① manter MSB

② Da esq para a dir somar o bit gray seguinte c/ o bit binário anterior (esquecer transporte)

(a) 1111 ← gray
1010 ← bin

(b) 10110
11011

(c) 10100011001
11000010001

2a) $(-6)_{10} \times (-5)_{10} = (+30)_{10}$

bit sinal: 1+1 = 1|0
bit de sinal

magnitude:

```
      110
    x 101
    -----
      110
   + 000
    -----
     0110
  + 110
    -----
    11110 (= 30)
```

resultado: 011110

NOTA: numa representação em 4 bits há OVERFLOW porque o resultado tem 6 bits

(4b) $(+21)_{10} \times (+10)_{10} = (+210)_{10}$

bit sinal: 0+0 = 0

magnitude:

```
        10101
       x 1010
       ------
        00000
     + 10101
       ------
       101010
      00000
   + 10101
   ----------
     11010010 (=210)
```

$\underbrace{1101}_{D}\ \underbrace{0010}_{2}\ (=210)_{10}$

13 × 16 + 2 = 210