Lógica Digital (1001351)

Representação Numérica e Circuitos Aritméticos

Prof. Edilson Kato
kato@ufscar.br

Prof. Maurício Figueiredo
mauricio@ufscar.br

Prof. Ricardo Menotti
menotti@ufscar.br

Prof. Roberto Inoue
rsinoue@ufscar.br

Departamento de Computação
Universidade Federal de São Carlos

Atualizado em: 10 de abril de 2019

Departamento de Computação

Relembrando...

- No sistema numérico binário, é usada a representação numérica posicional:
  - $B = b_{n-1}b_{n-2}...b_1b_0$
  - $V(B) = b_{n-1} \times 2^{n-1} + b_{n-2} \times 2^{n-2}...b_1 \times 2^1 + b_0 \times 2^0$
  - $= \sum_{i=0}^{n-1} b_i \times 2^i$

$$K = k_{n-1}k_{n-2} \cdots k_1k_0$$

$$V(K) = \sum_{i=0}^{n-1} k_i \times r^i$$

Representações Octal e Hexadecimal

Table 3.1 Numbers in different systems.

Decimal	Binary	Octal	Hexadecimal
00	00000	00	00
01	00001	01	01
02	00010	02	02
03	00011	03	03
04	00100	04	04
05	00101	05	05
06	00110	06	06
07	00111	07	07
08	01000	10	08
09	01001	11	09
10	01010	12	0A
11	01011	13	0B
12	01100	14	0C
13	01101	15	0D
14	01110	16	0E
15	01111	17	0F
16	10000	20	10
17	10001	21	11
18	10010	22	12

Representações Octal e Hexadecimal

$$101011010111_{(2)} = 5327_{(8)}$$

$$\underbrace{1\,0\,1}_{5} \qquad \underbrace{0\,1\,1}_{3} \qquad \underbrace{0\,1\,0}_{2} \qquad \underbrace{1\,1\,1}_{7}$$

Representações Octal e Hexadecimal

$$101011010111_{(2)} = 5327_{(8)}$$

$$\underbrace{1\,0\,1}_{5} \qquad \underbrace{0\,1\,1}_{3} \qquad \underbrace{0\,1\,0}_{2} \qquad \underbrace{1\,1\,1}_{7}$$

$$010111011_{(2)} = 273_{(8)}$$

$$\underbrace{0\,1\,0}_{2} \qquad \underbrace{1\,1\,1}_{7} \qquad \underbrace{0\,1\,1}_{3}$$

Representações Octal e Hexadecimal

$$1010111100100101_{(2)} = AF25_{(16)}$$

$$\underbrace{1\ 0\ 1\ 0}_{\text{A}} \qquad \underbrace{1\ 1\ 1\ 1}_{\text{F}} \qquad \underbrace{0\ 0\ 1\ 0}_{2} \qquad \underbrace{0\ 1\ 0\ 1}_{5}$$

Representações Octal e Hexadecimal

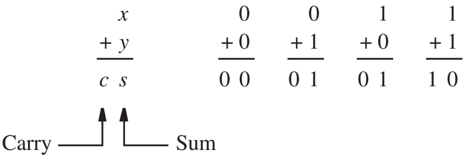
$$1010111100100101_{(2)} = AF25_{(16)}$$

$$\underbrace{1\,0\,1\,0}_{\text{A}} \quad \underbrace{1\,1\,1\,1}_{\text{F}} \quad \underbrace{0\,0\,1\,0}_{2} \quad \underbrace{0\,1\,0\,1}_{5}$$

$$001101101000_{(2)} = 368_{(16)} =$$

$$\underbrace{0\,0\,1\,1}_{3} \quad \underbrace{0\,1\,1\,0}_{6} \quad \underbrace{1\,0\,0\,0}_{8}$$

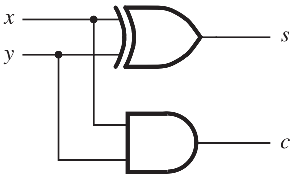
Adição de números sem sinal



(a) The four possible cases

x	y	Carry c	Sum s
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

(b) Truth table



(c) Circuit

(d) Graphical symbol

Figure 3.1 Half-adder.

Adição de números sem sinal

$$
\begin{array}{lcccc}
\text{Generated carries} & \longrightarrow & 1\,1\,1\,0 & & \\
X = x_4 x_3 x_2 x_1 x_0 & & 0\,1\,1\,1\,1 & & (15)_{10} \\
+\,Y = y_4 y_3 y_2 y_1 y_0 & & +\,0\,1\,0\,1\,0 & & +\,(10)_{10} \\
\hline
S = s_4 s_3 s_2 s_1 s_0 & & 1\,1\,0\,0\,1 & & (25)_{10}
\end{array}
$$

(a) An example of addition

$$
\begin{array}{cccc}
\ldots & c_{i+1} & c_i & \ldots \\
\ldots & \ldots & x_i & \ldots \\
\ldots & \ldots & y_i & \ldots \\
\hline
\ldots & \ldots & s_i & \ldots
\end{array}
$$

(b) Bit position i

Figure 3.2 Addition of multibit numbers.

Adição de números sem sinal

c_i	x_i	y_i	c_{i+1}	s_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

(a) Truth table

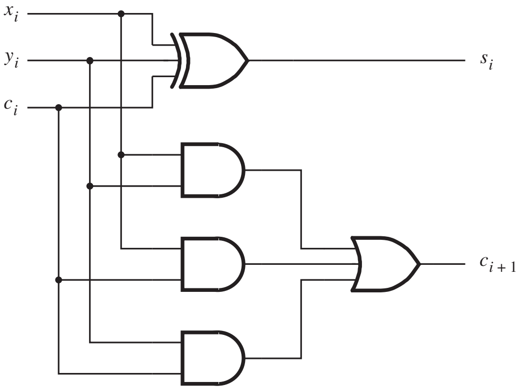
c_i \ $x_i y_i$	00	01	11	10
0		1		1
1	1		1	

$$s_i = x_i \oplus y_i \oplus c_i$$

c_i \ $x_i y_i$	00	01	11	10
0			1	
1		1	1	1

$$c_{i+1} = x_i y_i + x_i c_i + y_i c_i$$

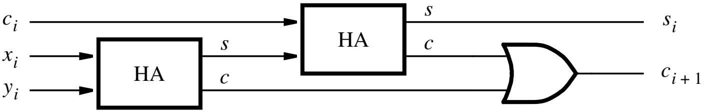
(b) Karnaugh maps



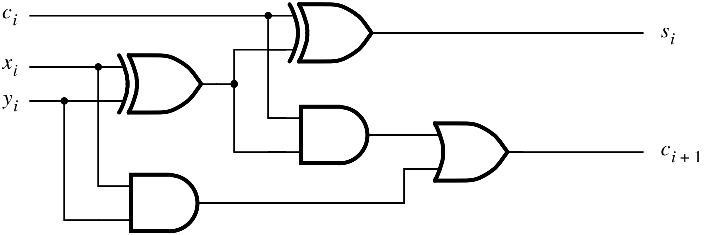
(c) Circuit

Figure 3.3 Full-adder.

Adição de números sem sinal



(a) Block diagram



(b) Detailed diagram

Figure 3.4 A decomposed implementation of the full-adder circuit.

Adição de números sem sinal

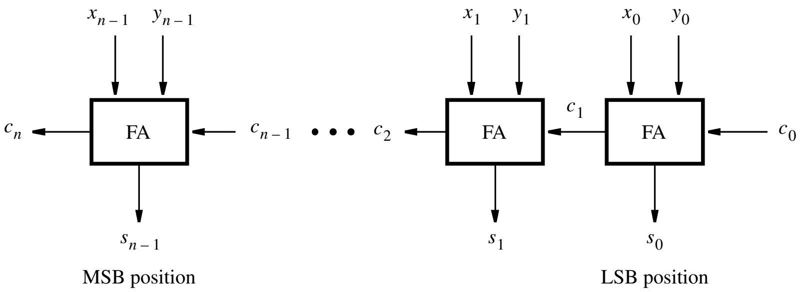


Figure 3.5 An n-bit ripple-carry adder.

Adição de números sem sinal

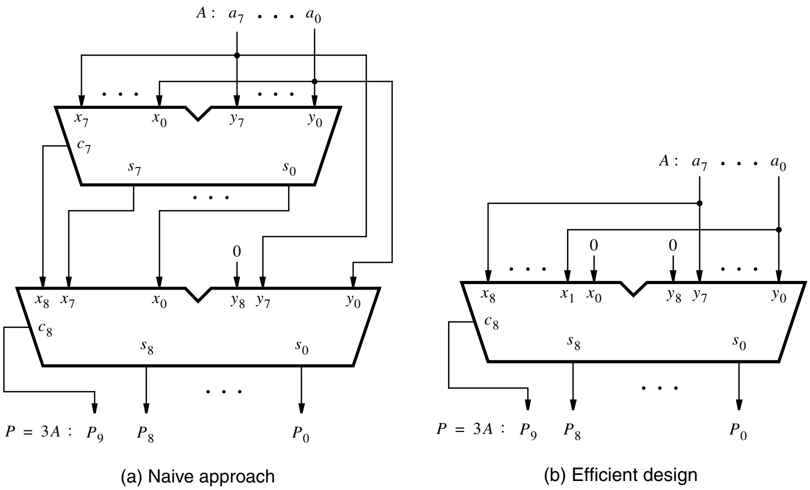


Figure 3.6 Circuit that multiplies an eight-bit unsigned number by 3.

Números com sinal

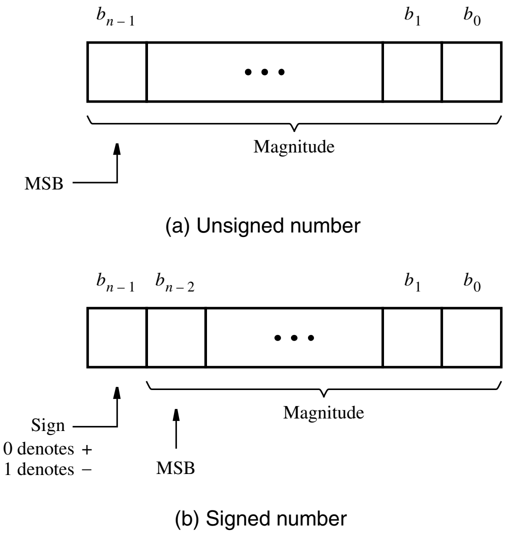


Figure 3.7 Formats for representation of integers.

Números com sinal

- Sinal/Magnitude
  - $+5 = 0101$ e $-5 = 1101$

Números com sinal

- Sinal/Magnitude
  - $+5 = 0101$ e $-5 = 1101$
- Complemento de 1
  - $K = (2^n - 1) - P$
  - $+5 = 0101$ e $-5 = 1010$

Números com sinal

- Sinal/Magnitude
  - $+5 = 0101$ e $-5 = 1101$
- Complemento de 1
  - $K = (2^n - 1) - P$
  - $+5 = 0101$ e $-5 = 1010$
- Complemento de 2
  - $K = (2^n - P)$
  - $+5 = 0101$ e $-5 = 1011$
  - `0101` `10110100` `00000001` `1000`
  - `101`**`1`** `01001`**`100`** `1111111`**`1`** **`1000!`**

Números com sinal

Table 3.2 Interpretation of four-bit signed integers.

$b_3b_2b_1b_0$	Sign and magnitude	1's complement	2's complement
0111	+7	+7	+7
0110	+6	+6	+6
0101	+5	+5	+5
0100	+4	+4	+4
0011	+3	+3	+3
0010	+2	+2	+2
0001	+1	+1	+1
0000	+0	+0	+0
1000	−0	−7	−8
1001	−1	−6	−7
1010	−2	−5	−6
1011	−3	−4	−5
1100	−4	−3	−4
1101	−5	−2	−3
1110	−6	−1	−2
1111	−7	−0	−1

Números com sinal

$$
\begin{array}{rr}
 & (+5) \\
+ & (+2) \\
\hline
 & (+7)
\end{array}
\qquad
\begin{array}{rr}
 & 0101 \\
+ & 0010 \\
\hline
 & 0111
\end{array}
\qquad\qquad
\begin{array}{rr}
 & (-5) \\
+ & (+2) \\
\hline
 & (-3)
\end{array}
\qquad
\begin{array}{rr}
 & 1011 \\
+ & 0010 \\
\hline
 & 1101
\end{array}
$$

$$
\begin{array}{rr}
 & (+5) \\
+ & (-2) \\
\hline
 & (+3)
\end{array}
\qquad
\begin{array}{rr}
 & 0101 \\
+ & 1110 \\
\hline
 & 1\ 0011
\end{array}
\qquad\qquad
\begin{array}{rr}
 & (-5) \\
+ & (-2) \\
\hline
 & (-7)
\end{array}
\qquad
\begin{array}{rr}
 & 1011 \\
+ & 1110 \\
\hline
 & 1\ 1001
\end{array}
$$

ignore　　　　ignore

Figure 3.9 Examples of 2's complement addition.

Números com sinal

(+ 5) − (+ 2) (+ 3)	0 1 0 1 − 0 0 1 0	⇒	0 1 0 1 + 1 1 1 0 1 0 0 1 1 ignore
(−5) − (+ 2) (−7)	1 0 1 1 − 0 0 1 0	⇒	1 0 1 1 + 1 1 1 0 1 1 0 0 1 ignore
(+ 5) − (−2) (+ 7)	0 1 0 1 − 1 1 1 0	⇒	0 1 0 1 + 0 0 1 0 0 1 1 1
(−5) − (−2) (−3)	1 0 1 1 − 1 1 1 0	⇒	1 0 1 1 + 0 0 1 0 1 1 0 1

Figure 3.10 Examples of 2's complement subtraction.

Números com sinal

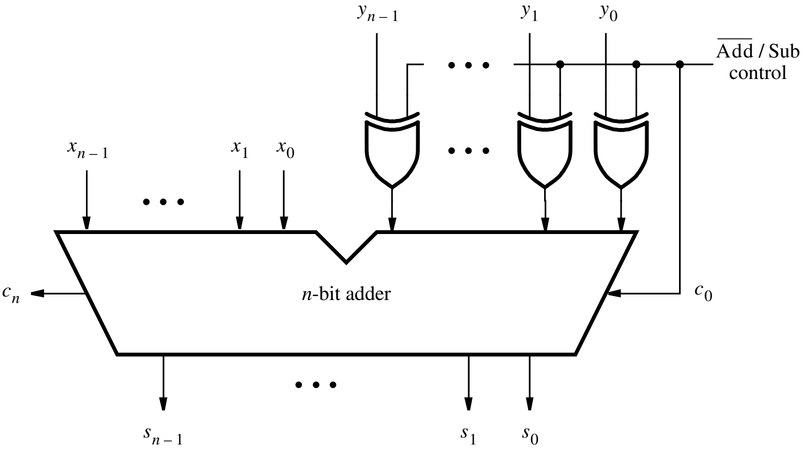


Figure 3.12 Adder/subtractor unit.

Números com sinal

$$
\begin{array}{r}
(+7) \\
+\ (+2) \\
\hline
(+9)
\end{array}
\qquad
\begin{array}{r}
0\,1\,1\,1 \\
+\ 0\,0\,1\,0 \\
\hline
1\,0\,0\,1
\end{array}
\qquad\qquad
\begin{array}{r}
(-7) \\
+\ (+2) \\
\hline
(-5)
\end{array}
\qquad
\begin{array}{r}
1\,0\,0\,1 \\
+\ 0\,0\,1\,0 \\
\hline
1\,0\,1\,1
\end{array}
$$

$$
c_4 = 0 \quad c_3 = 1 \qquad\qquad c_4 = 0 \quad c_3 = 0
$$

$$
\begin{array}{r}
(+7) \\
+\ (-2) \\
\hline
(+5)
\end{array}
\qquad
\begin{array}{r}
0\,1\,1\,1 \\
+\ 1\,1\,1\,0 \\
\hline
1\,0\,1\,0\,1
\end{array}
\qquad\qquad
\begin{array}{r}
(-7) \\
+\ (-2) \\
\hline
(-9)
\end{array}
\qquad
\begin{array}{r}
1\,0\,0\,1 \\
+\ 1\,1\,1\,0 \\
\hline
1\,0\,1\,1\,1
\end{array}
$$

$$
c_4 = 1 \quad c_3 = 1 \qquad\qquad c_4 = 1 \quad c_3 = 0
$$

Figure 3.13 Examples for determination of overflow.

Números com sinal

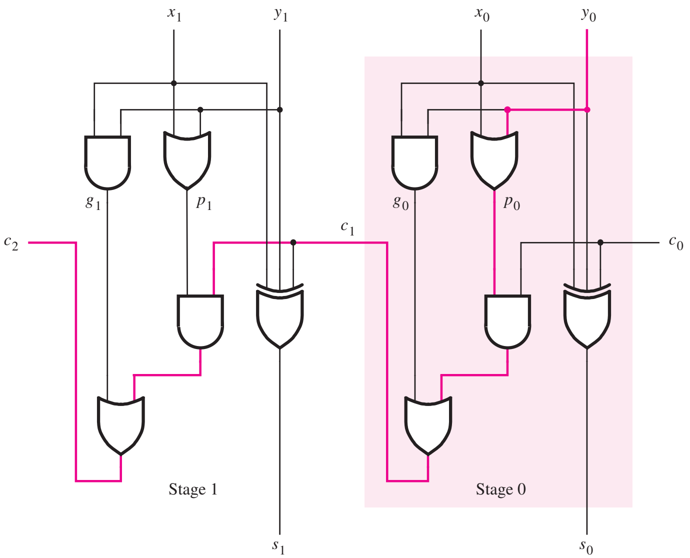


Figure 3.14 A ripple-carry adder based on expression 3.3.

Números com sinal

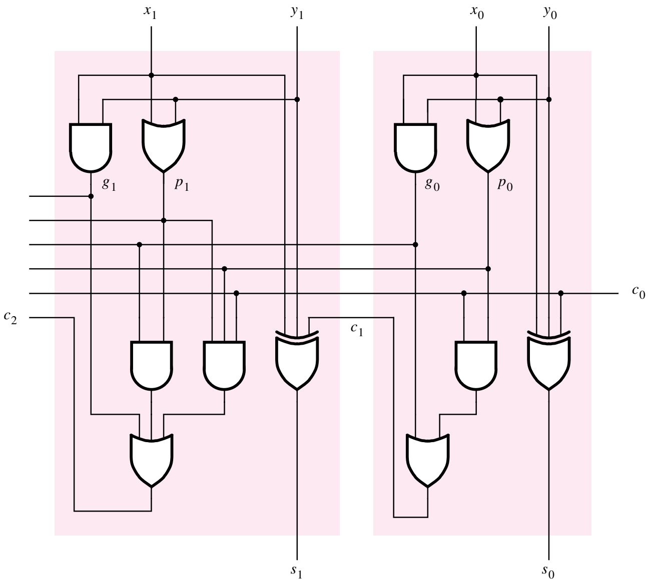


Figure 3.15 The first two stages of a carry-lookahead adder.

Bibliografia

- Brown, S. & Vranesic, Z. - Fundamentals of Digital Logic with Verilog Design, 3rd Ed., Mc Graw Hill, 2009

Lógica Digital (1001351)

Representação Numérica e Circuitos Aritméticos

Prof. Edilson Kato
kato@ufscar.br

Prof. Maurício Figueiredo
mauricio@ufscar.br

Prof. Ricardo Menotti
menotti@ufscar.br

Prof. Roberto Inoue
rsinoue@ufscar.br

Departamento de Computação
Universidade Federal de São Carlos

Atualizado em: 10 de abril de 2019

Departamento de Computação