Digitalteknik EITF65

Lecture 5: Boolean Algebra and Arithmetic Functions

September 11, 2018

Ring

Definition

The algebraic structure $(R, +, \cdot)$ is a ring if the following criterias are fulfilled:

Addition:

$$a + b = b + a \quad \text{(commutative)}$$
$$(a + b) + c = a + (b + c) \quad \text{(associative)}$$
$$a + 0 = a \quad \text{(0 exists)}$$
$$a + (-a) = 0 \quad \text{($-a$ exists)}$$

Multiplication:

$$(a \cdot b) \cdot c = a \cdot (b \cdot c) \quad \text{(associative)}$$
$$a \cdot 1 = 1 \cdot a = a \quad \text{(1 exists)}$$

Combination:

$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{(distributive)}$$
$$(b + c) \cdot a = b \cdot a + c \cdot a$$

Boolean rings

Definition

An element a in a ring $(R, +, \cdot)$ is idempotent if $a^2 = a$.

Definition (3.8)

A Boolean ring is a ring where all elements are idempotent.

Boolean rings (example)

Example (3.31)

The ring $(\mathbb{Z}_2, \oplus, \otimes)$ with

$\oplus$	0	1
0	0	1
1	1	0

$\otimes$	0	1
0	0	0
1	0	1

is a Boolean ring since

$$0 \otimes 0 = 0$$
$$1 \otimes 1 = 1$$

Boolean ring (example)

Example 3.32

The ring $(\mathbb{Z}_6, \oplus, \otimes)$ is not a Boolean ring.

$\oplus$	0	1	2	3	4	5
0	0	1	2	3	4	5
1	1	2	3	4	5	0
2	2	3	4	5	0	1
3	3	4	5	0	1	2
4	4	5	0	1	2	3
5	5	0	1	2	3	4

$\otimes$	0	1	2	3	4	5
0	0	0	0	0	0	0
1	0	1	2	3	4	5
2	0	2	4	0	2	4
3	0	3	0	3	0	3
4	0	4	2	0	4	2
5	0	5	4	3	2	1

Let B be the idempotent elements in $\mathbb{Z}_6$, i.e., $B = \{0, 1, 3, 4\}$. Define addition and multiplication according to

$$a + b = a \oplus b \oplus 4 \otimes (a \otimes b)$$

$$a \cdot b = a \otimes b$$

Ex 3.32 (cont'd)

Example 3.32

Then $(B, +, \cdot)$ with

$+$	0	1	3	4
0	0	1	3	4
1	1	0	4	3
3	3	4	0	1
4	4	3	1	0

$\cdot$	0	1	3	4
0	0	0	0	0
1	0	1	3	4
3	0	3	3	0
4	0	4	0	4

is a Boolean ring.

$a + a = 0$

Theorem (3.16)

If $(B, +, \cdot)$ is a Boolean ring and $a \in B$ then

$$a + a = 0$$

That is, all elements have characteristic 2.

All elements are their own additive inverses, $-a = a$.
There is no meaning to use minus in a Boolean ring.

Proof of Thm 3.16

In a Boolean ring

$$(a+b)^2 = a+b$$

But

$$\begin{aligned}(a+b)^2 &= (a+b)(a+b)\\ &= a^2+ab+ba+b^2\\ &= a+ab+ba+b\end{aligned}$$

That is

$$a+ab+ba+b = a+b$$

Theorem 3.7 (cancellation law) gives

$$ab+ba = 0$$

Let $b \leftarrow a$

$$a^2+a^2 = a+a = 0$$

Boolean rings are commutative

Theorem (3.17)

A Boolean ring $(B, +, \cdot)$ is a commutative ring. That is, if $a, b \in B$ then

$$a \cdot b = b \cdot a$$

Boolean ring (Summary)

Boolean ring

The ring $(R, +, \cdot)$ is an algebraic structure where

Addition:

$$a + b = b + a \quad \text{(commutative)}$$
$$(a + b) + c = a + (b + c) \quad \text{(associative)}$$
$$a + 0 = a \quad \text{(0 exists)}$$
$$a + (-a) = 0 \quad (-a \text{ exists})$$

Multiplication:

$$(a \cdot b) \cdot c = a \cdot (b \cdot c) \quad \text{(associative)}$$
$$a \cdot 1 = 1 \cdot a = a \quad \text{(1 exists)}$$
$$a^2 = a \quad \text{(idempotent)}$$

Combination:

$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{(distributive)}$$
$$(b + c) \cdot a = b \cdot a + c \cdot a$$

Theorem:

$$a + a = 0 \quad \text{(Characteristic 2)}$$
$$a \cdot b = b \cdot a \quad \text{Commutative ring}$$

Boolean algebra

Definition (3.9)

A Boolean algebra $(B, \wedge, \vee, ')$ consists of a set B and three operations AND ($\wedge$), OR ($\vee$), and NOT ($'$) where

$$a \wedge 1 = a \qquad a \vee 0 = a$$

$$a \wedge a' = 0 \qquad a \vee a' = 1$$

$$a \wedge b = b \wedge a \qquad a \vee b = b \vee a$$

$$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c) \qquad a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$$

Boolean operations

Theorem

If $(B, +, \cdot)$ is a Boolean ring, then the Boolean operations AND, OR, and NOT can be derived as

$$a \wedge b = a \cdot b \qquad \textit{AND}$$

$$a \vee b = a + b + a \cdot b \qquad \textit{OR}$$

$$a' = 1 + a \qquad \textit{NOT}$$

$(\mathbb{Z}_2, \oplus, \otimes) \rightarrow (\mathbb{Z}_2, \wedge, \vee, ')$

Example

In the Boolean ring $(\mathbb{Z}_2, \oplus, \otimes)$ the Boolean operations are

$$a \wedge b = a \otimes b \qquad \text{AND}$$
$$a \vee b = a \oplus b \oplus a \otimes b \qquad \text{OR}$$
$$a' = 1 \oplus a \qquad \text{NOT}$$

Or

$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
1	1	1

$'$	
0	1
1	0

Ring operations

Theorem

If $(B, \wedge, \vee, ')$ is a Boolean algebra, then the operations $+$ and $\cdot$ can be derived as

$$a \cdot b = a \wedge b$$
$$a + b = (a \wedge b') \vee (a' \wedge b)$$

$(\mathbb{Z}_2, \wedge, \vee, ')\rightarrow(\mathbb{Z}_2, \oplus, \otimes)$

Example

In the Boolean algebra $(\mathbb{Z}_2, \wedge, \vee, ')$ the ring operations are

$$a \otimes b = a \wedge b \qquad \text{MULT}$$
$$a \oplus b = ab' \vee a'b \qquad \text{ADD}$$

Or

$\otimes$	0	1
0	0	0
1	0	1

$\oplus$	0	1
0	0	1
1	1	0

Often $\wedge$ and () are omitted. For example, $a \vee bc$ is equivalent to $a \vee (b \wedge c)$.

$(B, \wedge, \vee, ')$

Example

In the Boolean ring $(B, +, \cdot)$, which builds from the idempodent elements in $(\mathbb{Z}_6, \oplus, \otimes)$ and

$$a + b = a \oplus b \oplus 4 \otimes (a \otimes b)$$
$$a \cdot b = a \otimes b$$

The Boolean operations are

$$\begin{aligned} a \wedge b &= a \cdot b && = a \otimes b \\ a \vee b &= a + b + a \cdot b && = a \oplus b \oplus (5 \otimes a \otimes b) \\ a' &= 1 + a && = 1 \oplus 5 \otimes a \end{aligned}$$

$(B, \wedge, \vee, ')$

$\wedge$	0	1	3	4
0	0	0	0	0
1	0	1	3	4
3	0	3	3	0
4	0	4	0	4

$\vee$	0	1	3	4
0	0	1	3	4
1	1	1	1	1
3	3	1	3	1
4	4	1	1	4

$'$	
0	1
1	0
3	4
4	3

Principle of duality

Theorem (3.18)

For each true expression in 0, 1, $\wedge$, $\vee$, *and* $'$ *in a Boolean algebra* $(B, 0, 1, \wedge, \vee, ')$ *we also get a true expression if we exchange all*

$$0 \Leftrightarrow 1$$

$$\wedge \Leftrightarrow \vee$$

Derivation rules for a Boolean algebra

	$0' = 1$	$1' = 0$
	$a'' = a$	
	$a0 = 0$	$a \vee 1 = 1$
	$a1 = a$	$a \vee 0 = a$
	$aa' = 0$	$a \vee a' = 1$
(Idempotence)	$aa = a$	$a \vee a = a$
(de Morgan)	$(ab)' = a' \vee b'$	$(a \vee b)' = a'b'$
(Commutative)	$ab = ba$	$a \vee b = b \vee a$
(Associative)	$a(bc) = (ab)c$	$a \vee (b \vee c) = (a \vee b) \vee c$
(Distributive)	$a(b \vee c) = ab \vee ac$	$a \vee bc = (a \vee b)(a \vee c)$
(Absorption)	$a \vee ab = a$	$a(a \vee b) = a$
(Consensus)	$ab \vee a'c = ab \vee a'c \vee bc$	$(a \vee b)(a' \vee c) = (a \vee b)(a' \vee c)(b \vee c)$

Example of Boolean calculation

Example

Show that $(xz' \vee xy')' = x' \vee yz$.

$$
\begin{aligned}
(xz' \vee xy')' &= (xz')'(xy')' && \text{(deMorgan)} \\
&= (x' \vee z)(x' \vee y) && \text{(deMorgan)} \\
&= x'x' \vee x'y \vee x'z \vee yz && \text{(distributive)} \\
&= x' \vee x'y \vee x'z \vee yz && \text{(idempotence)} \\
&= x' \vee x'z \vee yz && \text{(absorbation)} \\
&= x' \vee yz && \text{(absorbation)}
\end{aligned}
$$

Representation of numbers

A binary vector of length n, $\mathbf{x} = (x_{n-1}, x_{n-2}, \ldots, x_1, x_0)$, can represent 2^n numbers.
One way is by the function

$$\phi(\mathbf{x}) = \sum_{i=0}^{n-1} x_i 2^i = x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \cdots + x_1 2 + x_0$$

Then,

- x_{n-1} is the most significant bit (msb).
- x_0 is the least significant bit (lsb).

Other basis

Other common basis are 8 (octal numbers) and 16 (hexa- or cedecimal)

- In octal numbers we use $\{0, 1, \ldots, 7\}$. Since $8 = 2^3$ it corresponds to three bits per digit.

$$245_{10} = (\underbrace{011}_{3}\ \underbrace{110}_{6}\ \underbrace{101}_{5})_2 = 365_8$$

- In hexadecimal numbers we use $\{0, 1, \ldots, 9, A, B, C, D, E, F\}$. Since $16 = 2^4$ it corresponds to four bits per digit.

$$245_{10} = (\underbrace{1111}_{F}\ \underbrace{0101}_{5})_2 = F5_{16}$$

Inverse of ϕ

The inverse of ϕ can be derived by using $n = 2q + r$ iteratively.

Example

Write $n = 245$ in binary form.

$$
\begin{aligned}
245 &= 2 \cdot 122 + 1 \quad \text{(lsb)} \\
122 &= 2 \cdot 61 + 0 \\
61 &= 2 \cdot 30 + 1 \\
30 &= 2 \cdot 15 + 0 \\
15 &= 2 \cdot 7 + 1 \\
7 &= 2 \cdot 3 + 1 \\
3 &= 2 \cdot 1 + 1 \\
1 &= 2 \cdot 0 + 1 \quad \text{(msb)}
\end{aligned}
$$

$\Rightarrow 245_{10} = 11110101_2$

Inverse of ϕ (explanation)

$$
\begin{aligned}
245 &= 2 \cdot 122 + 1 = 2 \cdot (2 \cdot 61) + 1 = 2^2 \cdot 61 + 1 \\
&= 2^2 \cdot (2 \cdot 30 + 1) + 1 = 2^3 \cdot 30 + 2^2 + 1 \\
&= 2^3 \cdot (2 \cdot 15) + 2^2 + 1 = 2^4 \cdot 15 + 2^2 + 1 \\
&= 2^4 \cdot (2 \cdot 7 + 1) + 2^2 + 1 = 2^5 \cdot 7 + 2^4 + 2^2 + 1 \\
&= 2^5 \cdot (2 \cdot 3 + 1) + 2^4 + 2^2 + 1 = 2^6 \cdot 3 + 2^5 + 2^4 + 2^2 + 1 \\
&= 2^6 \cdot (2 + 1) + 2^5 + 2^4 + 2^2 + 1 = 2^7 + 2^6 + 2^5 + 2^4 + 2^2 + 1 \\
&= \underbrace{1 \cdot 2^7 + 1 \cdot 2^6 + 1 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0}_{\phi(11110101)}
\end{aligned}
$$

Addition

Example

Consider the (decimal) derivation $45 + 28$

$$\begin{array}{r} {\scriptstyle 1} \\ 45 \\ +28 \\ \hline 73 \end{array}$$

Binary, this is equivalent to $45_{10} + 28_{10} = 101101_2 + 011100_2$

$$\begin{array}{r} {\scriptstyle 1\ 1\ 1\ 1} \\ 101101 \\ +011100 \\ \hline 1001001 \end{array}$$

where $1001001_2 = 73_{10}$.

Full adder (FA)

Consider bit-level i in an addition of $\mathbf{x}$ and $\mathbf{y}$. Then we add the carry from the previous bit-level c_i with x_i and y_i. The sum is given in two bits, the answer at this level s_i and the carry to the next level c_{i+1}.

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

A component that realizes this is called a Full Adder (FA).

Full adder (FA)

The truth table for a FA is

x_i	y_i	c_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

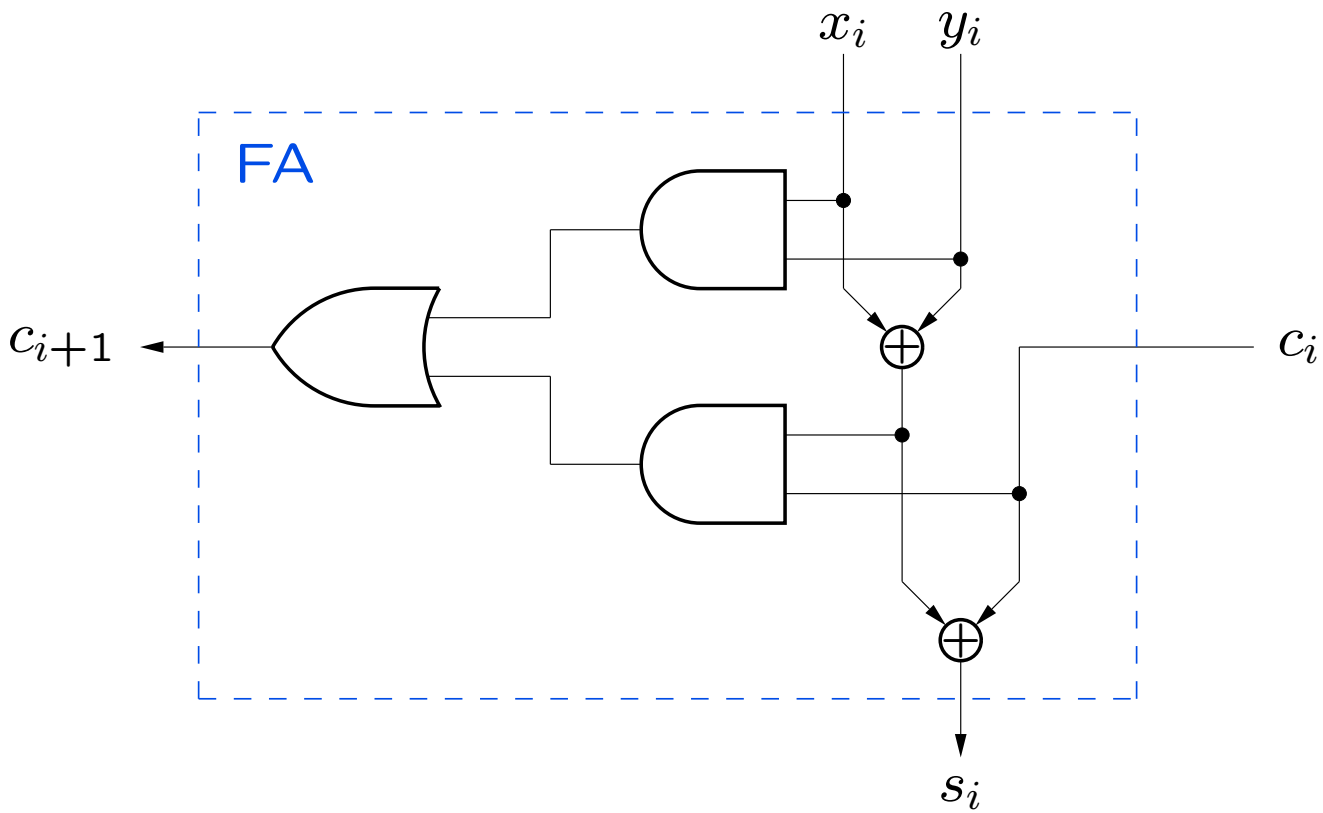
The functions s_i and c_{i+1}

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

$$c_{i+1} = x_i y_i \vee (x_i \oplus y_i) c_i$$

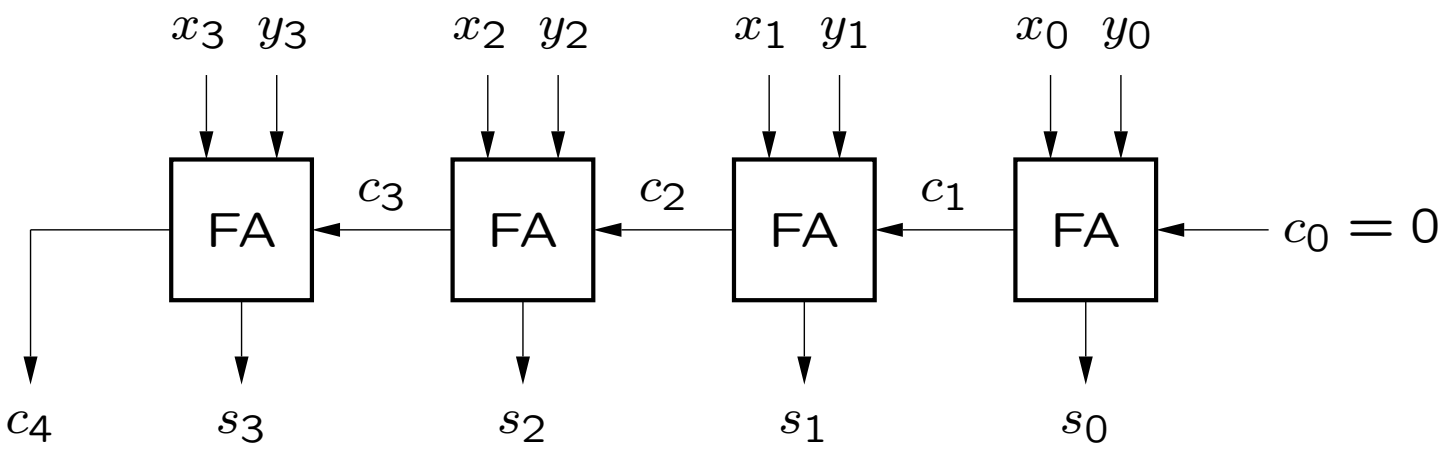
FA realization

The Full Adder (FA) can be realized as



4 bit Adder

To add two 4-bit numbers use a cascade of four FAs.



Overflow (positive numbers)

Suppose that we use n bit representation. Then the numbers are in the interval

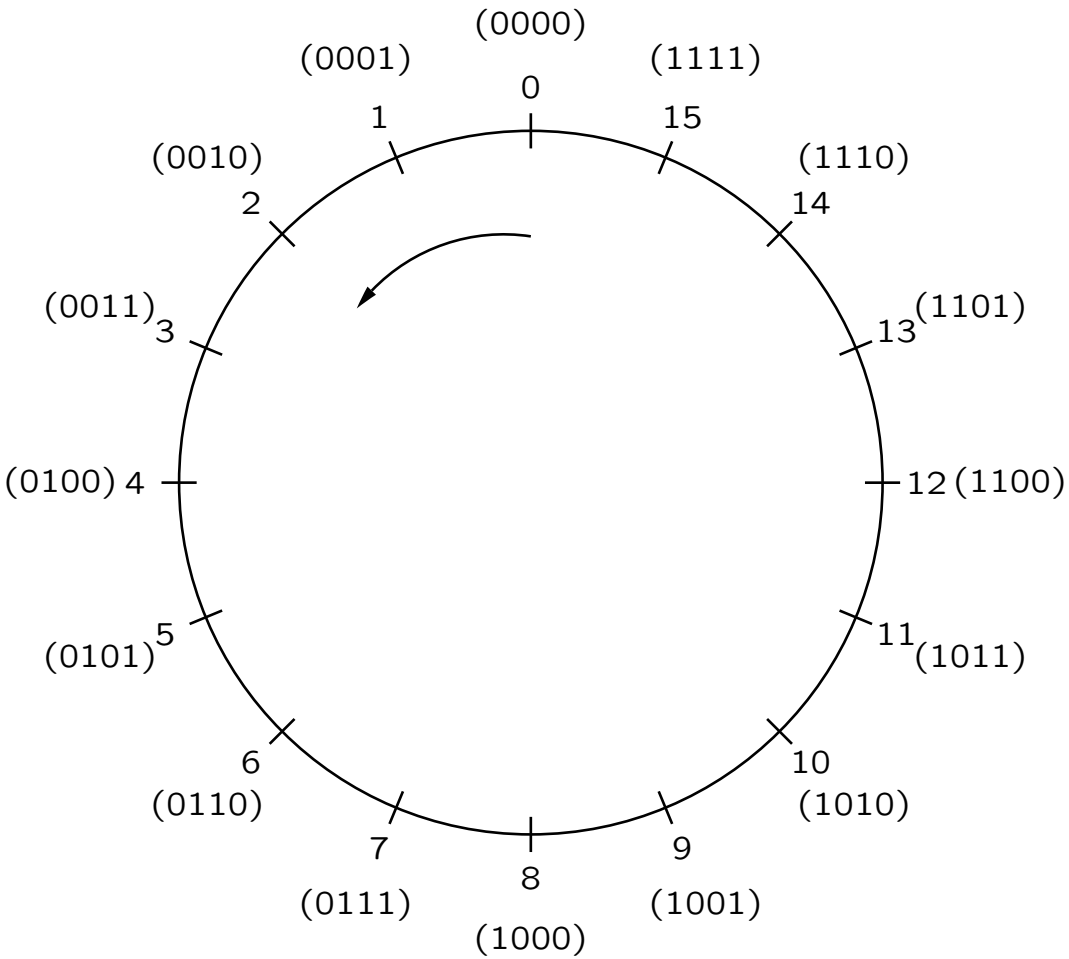
$$x, y, s \in \mathbb{Z}_{2^n} = \{0, 1, \ldots, 2^n - 1\}$$

If the sum of two numbers is too large we have that $c_n = 1$. Therefore, the result is split in two parts:

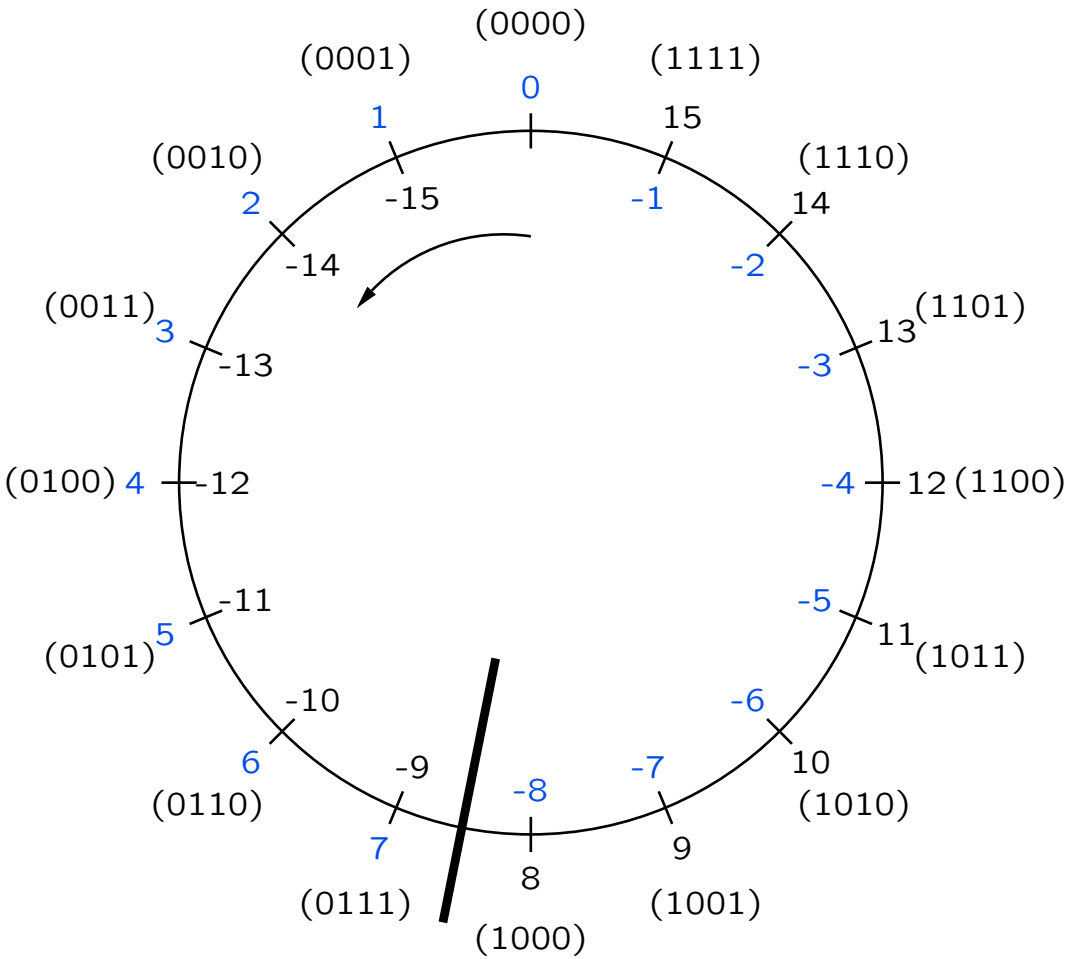
$$\mathbf{s} = R_{2^n}(\mathbf{x} + \mathbf{y})$$
$$ov = c_n$$

where *ov* is the *overflow* function.

Modulo 16


(0000)
0
(0001)
1
(1111)
15
(0010)
2
(1110)
14
(0011)
3
(1101)
13
(0100)
4
12
(1100)
(0101)
5
11
(1011)
6
(0110)
10
(1010)
7
(0111)
8
9
(1001)
(1000)

Representation of negative numbers



2-complement

In 2-complement representation the binary vector

$$\mathbf{x} = (x_{n-1}, x_{n-2}, \ldots, x_1, x_0)$$

then gives the value (since derived mod 2^n)

$$\begin{aligned}\phi(\mathbf{x}) &= (x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \cdots + x_1 2 + x_0) - x_{n-1}2^n \\ &= -x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \cdots + x_1 2 + x_0 \\ &= -x_{n-1}2^{n-1} + \sum_{i=0}^{n-2} x_i 2^i\end{aligned}$$

i.e. the most significant bit (msb) has a negative value. The numbers are now in the interval

$$\phi(\mathbf{x}) \in \{-(2^{n-1}), \ldots, 0, \ldots, 2^{n-1} - 1\}$$

2-complement

Since we work in $(\mathbb{Z}_{2^n}, \oplus)$ we have

$$-x \equiv 2^n - x = (2^n - 1 - x) + 1 \quad \text{modulo } 2^n$$

Then, since

$$2^n - 1 = \underbrace{11\ldots1}_{n \;\; 1:s}$$

and $1 - x_i = x_i'$ we get

$$2^n - 1 - x = x'_{n-1}x'_{n-2}\ldots x'_0$$

That is,

$$-\mathbf{x} = (x'_{n-1}x'_{n-2}\ldots x'_0) + 1$$

Subtraction

Example

Consider the subtraction $45 - 28$.

$$\begin{aligned} 45 - 28 &= 45 + (-28) \\ &= 0101101 + (-0011100) \\ &= 0101101 + [0011100]' + 1 \\ &= 0101101 + 1100011 + 1 \\ &= 0010001 = 17 \end{aligned}$$

It can be derived as

$$\begin{array}{r} {\scriptstyle 1\,1\;\;1\,1\,1\,1}1 \\ 0101101 \\ +1100011 \\ \hline 0010001 \end{array}$$

Overflow (2-complement)

The addition $\mathbf{x} + \mathbf{y} = \mathbf{s}$

should give overflow if

- **x** and **y** are positive and **s** is negative

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

- **x** and **y** are negative and **s** is positive

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

should not give overflow if

- **x** positive, **y** negative and **s** is positive

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

- **x** positive, **y** negative and **s** is negative

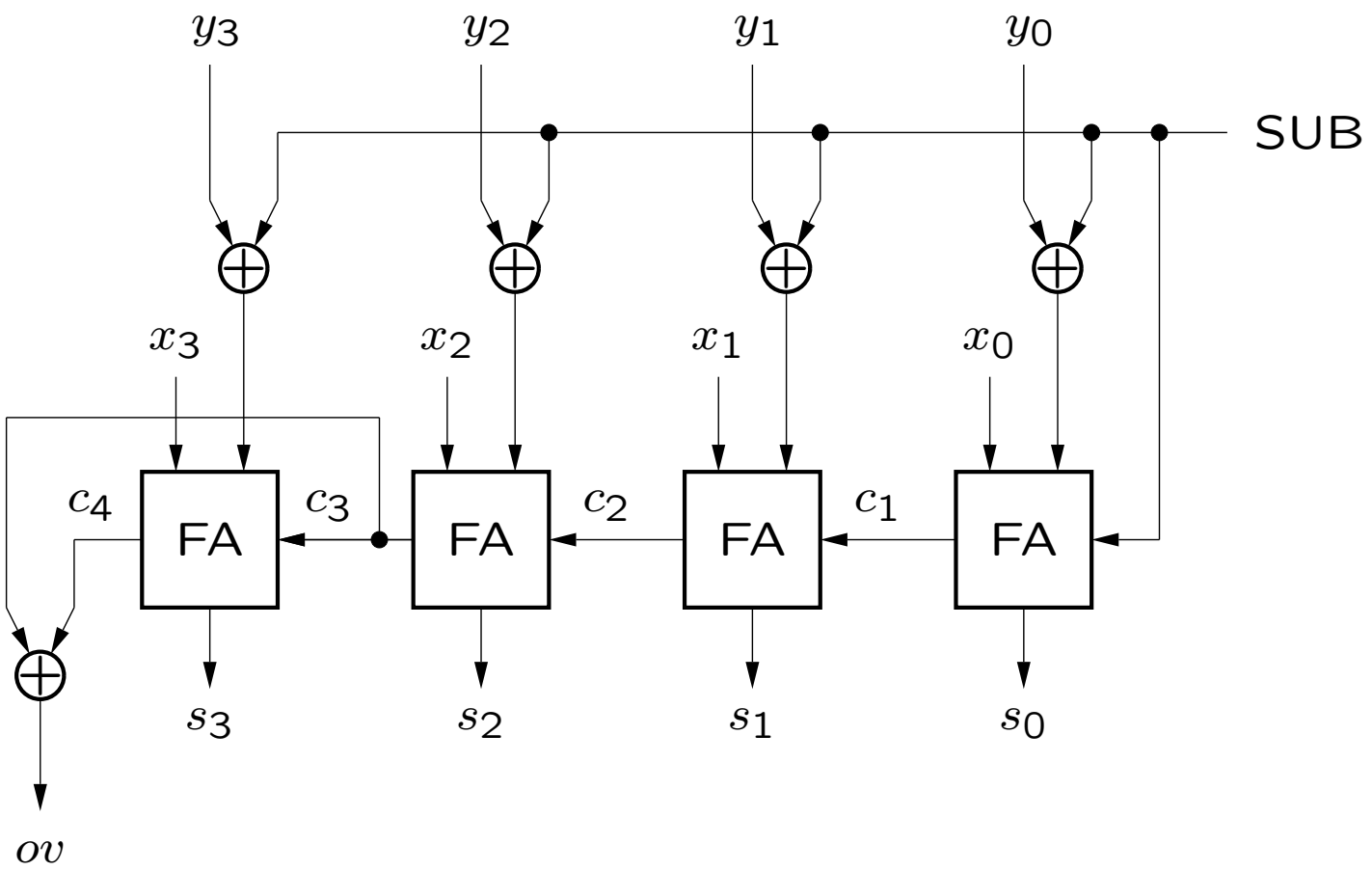
$$\begin{array}{r} {\scriptstyle 0\ 0}\ldots \\ 0\ldots \\ +1\ldots \\ \hline 1\ldots \end{array}$$

Hence,

$$ov = c_n \oplus c_{n-1}$$

Addition/Subtraction

Both addition and subtraction can be performed by the following circuit.



Multiplication

The product of two binary numbers can be performed in the same way as for decimal numbers.

Example

Multiply the 3 bit numbers $\mathbf{x} = x_2x_1x_0$ and $\mathbf{y} = y_2y_1y_0$:

			x_2	x_1	x_0
			y_2	y_1	y_0
			x_2y_0	x_1y_0	x_0y_0
		x_2y_1	x_1y_1	x_0y_1	
	x_2y_2	x_1y_2	x_0y_2		
z_5	z_4	z_3	z_2	z_1	z_0

Multiplication (circuit)