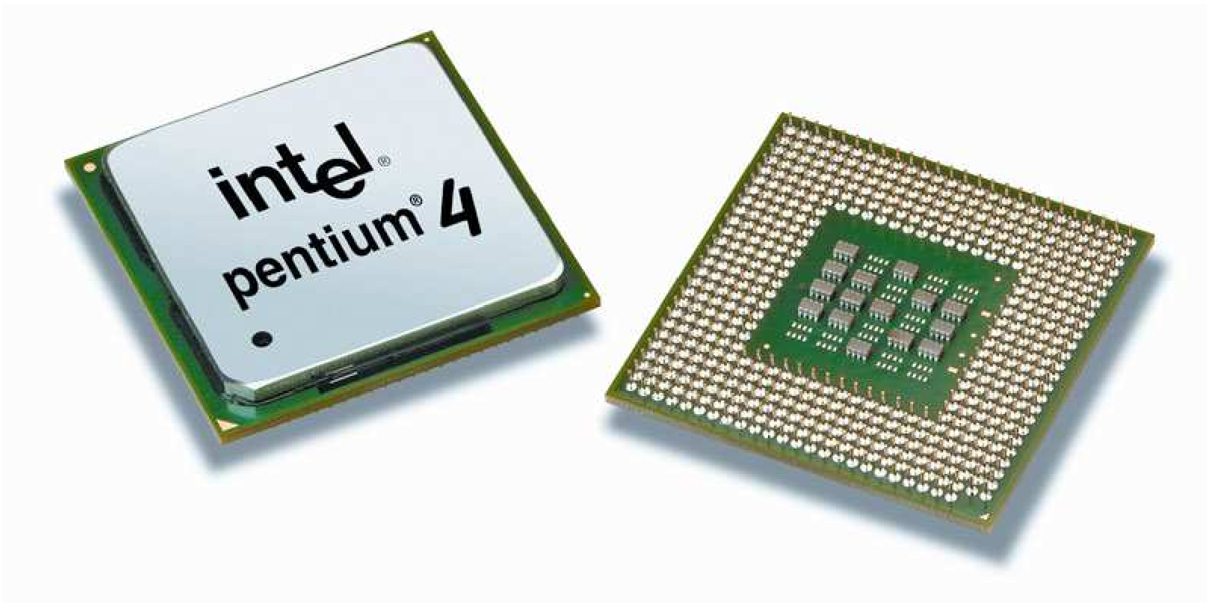
Boolean Algebra

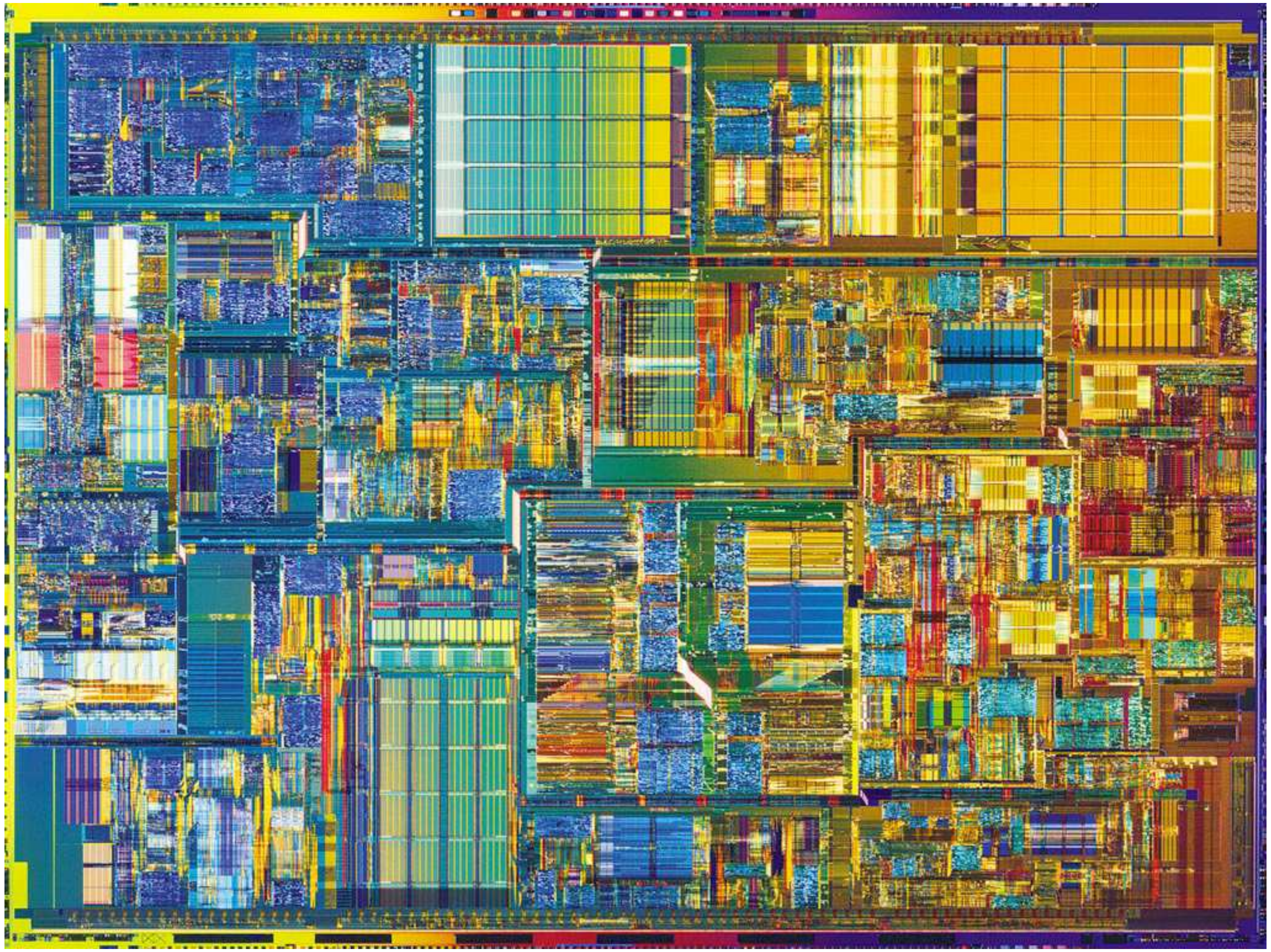
and Propositional Logic

Laura Kovács

Processors



Processors

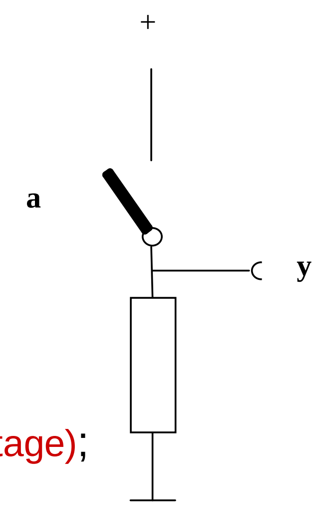


Processors are made of Switches (shalter)

- Switches, wires, resistors and capacitors
  - Pentium 4:
    - Willamette: 42 million switches
    - Northwood: 55 million switches
  - Memory chip has 256 million

Processors are made of Switches (Shalter)

- Switches, wires, resistors and capacitors
  - Pentium 4:
    - Willamette: 42 million switches
    - Northwood: 55 million switches
  - Memory chip has 256 million



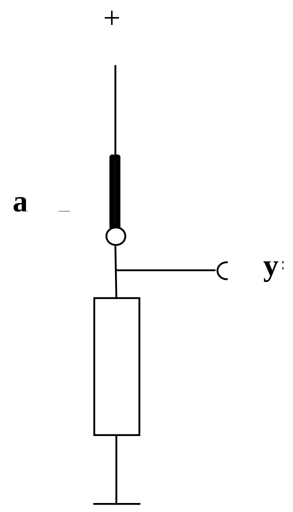
Example of a switch function:

- Switch a is open (under low voltage);

- Output y without voltage (Spannung).

Processors are made of Switches

- Switches, wires, resistors and capacitors
  - Pentium 4:
    - Willamette: 42 million switches
    - Northwood: 55 million switches
  - Memory chip has 256 million



Example of a switch function:

- Switch a is closed (under high voltage);

- Output y under voltage.

Processors are made of Switches

- Switches, wires, resistors and capacitors
  - Pentium 4:
    - Willamette: 42 million switches
    - Northwood: 55 million switches
  - Memory chip has 256 million

Multi-million switches are … HARD!

We need an abstraction.

Processors are made of Switches (shalter)

- Switches, wires, resistors and capacitors
  - Pentium 4:
    - Willamette: 42 million switches
    - Northwood: 55 million switches
  - Memory chip has 256 million

Multi-million switches are … HARD!

We need an abstraction: Boolean Algebra.

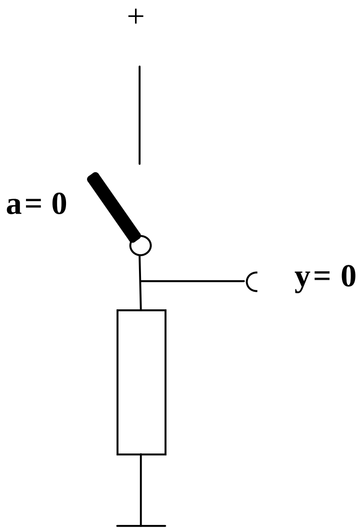
(propositional logic)

Boolean Algebra (Boole'sche Algebra)

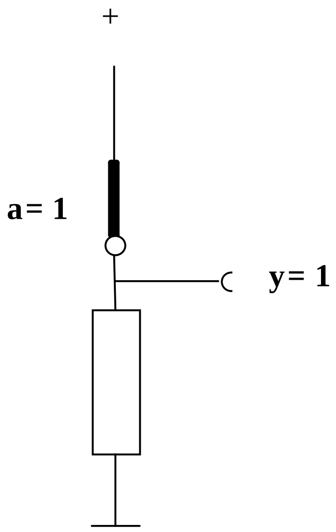
- Algebra on two-valued (binary) variables
  - G. Boole (1850) and C. Shannon (1938)
  - Straight-forward mapping to switches
    - High voltage (closed switch) ⇒ 1 (True)
    - Low voltage (opened switch) ⇒ 0 (False)

George Boole (1815 – 1864)

Boolean Algebra

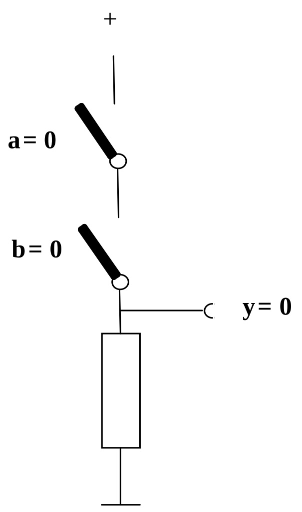


- Switch a is open
- Output y without voltage



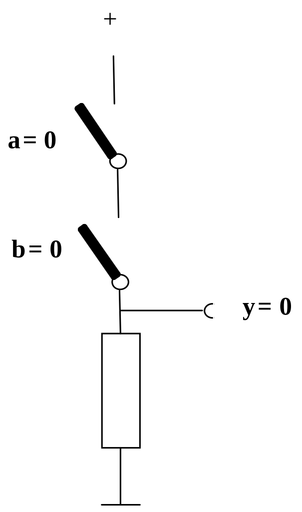
- Switch a is closed
- Output y under voltage

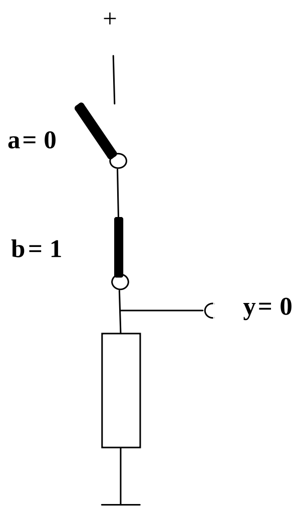
Boolean Algebra



- Switches a and b are open
- Output y without voltage

Boolean Algebra

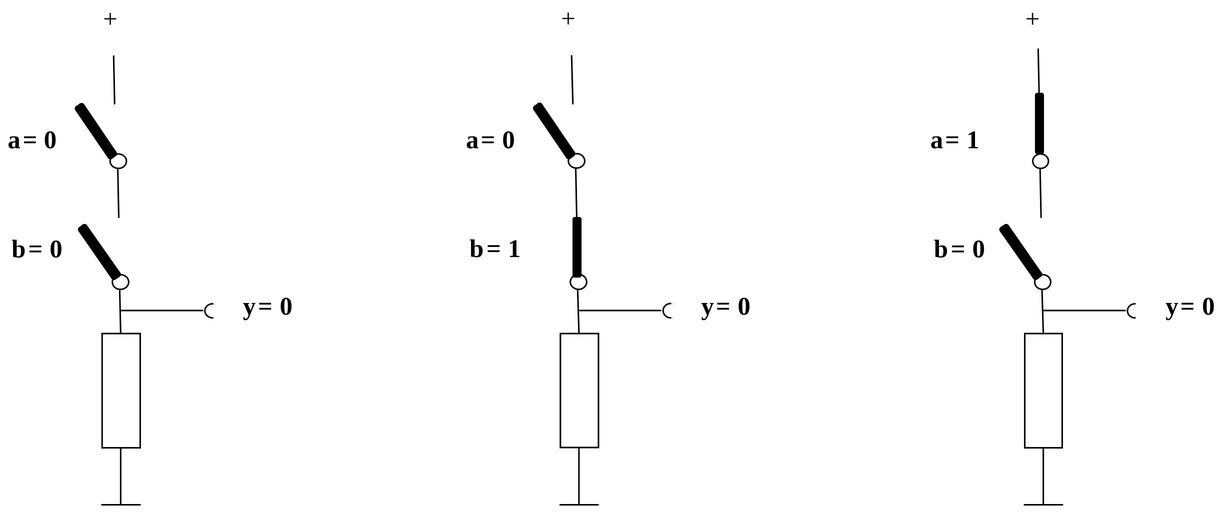




- Switches a and b are open
- Output y without voltage

- Switch a is open, b is closed
- Output y without voltage

Boolean Algebra

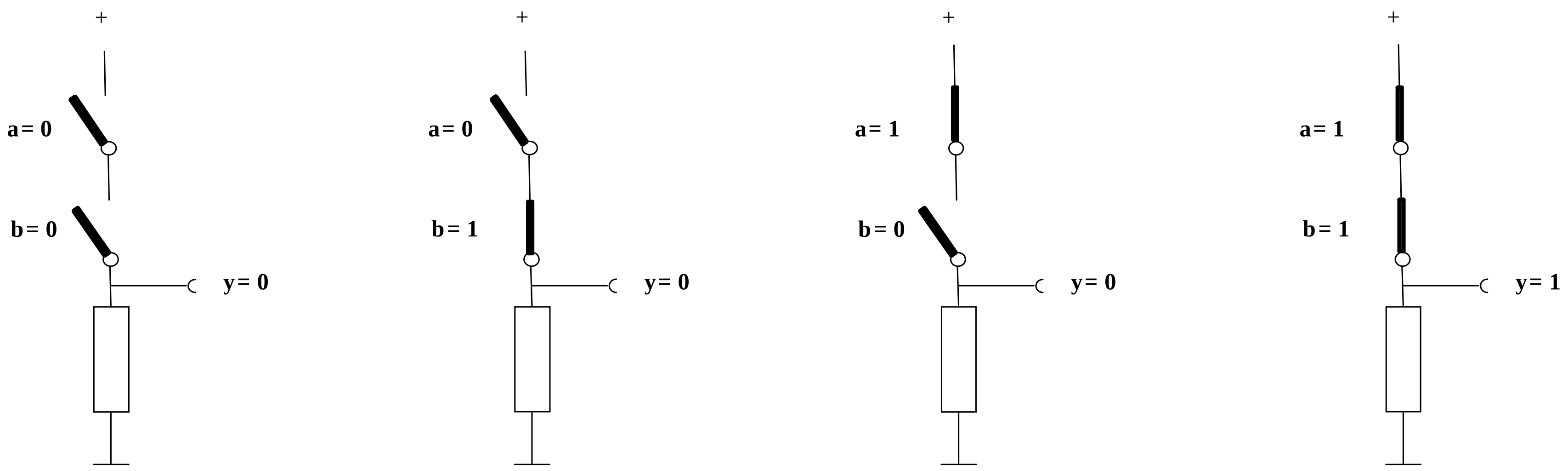


- Switches a and b are open
- Output y without voltage

- Switch a is open, b is closed
- Output y without voltage

- Switch a is closed, b is open
- Output y without voltage

Boolean Algebra



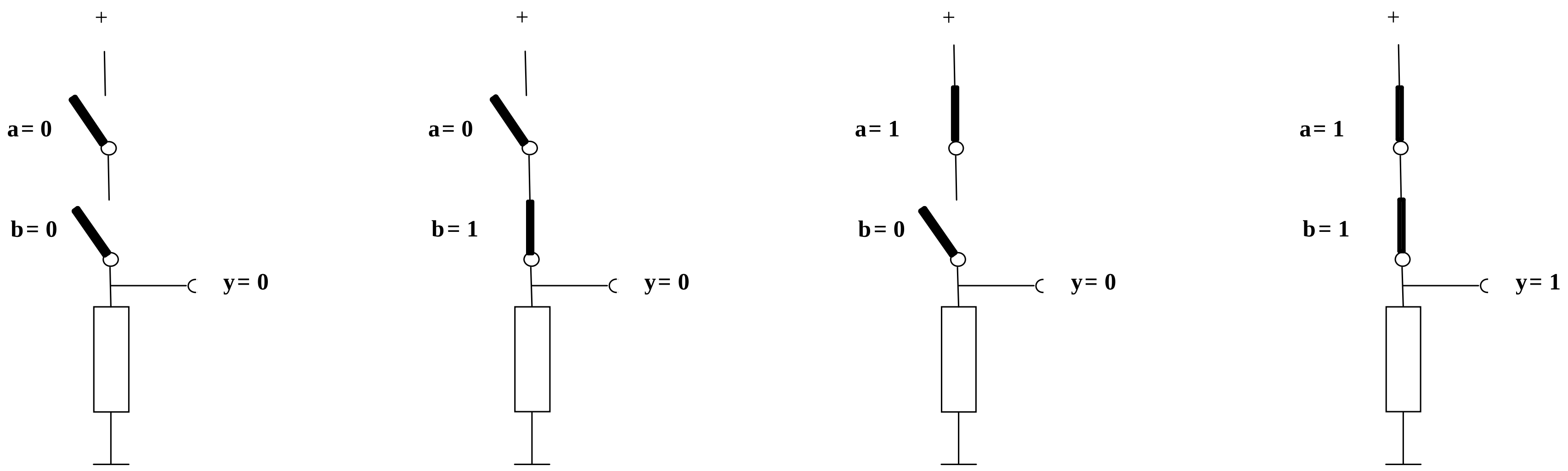
- Switches a and b are open
- Output y without voltage

- Switch a is open, b is closed
- Output y without voltage

- Switch a is closed, b is open
- Output y without voltage

-Switches a and b are closed
- Output y without voltage

Boolean Algebra → Serial Switches



- Switches a and b are open
- Output y without voltage

- Switch a is open, b is closed
- Output y without voltage

- Switch a is closed, b is open
- Output y without voltage

-Switches a and b are closed
- Output y without voltage

Output y under voltage if and only if switches a <u>and</u> b are closed.

Boolean Algebra

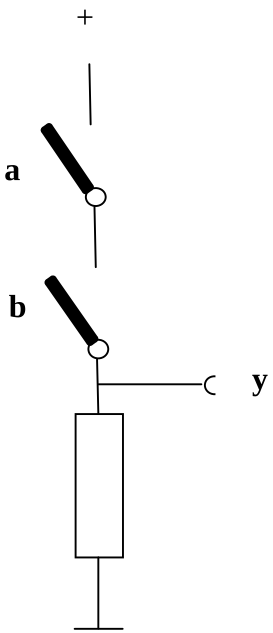
- Algebra on two-valued (binary) variables ← boolean variables (atoms)
- Boolean operators → to form boolean functions (formulas):
  - Conjunction (AND): $\wedge$

$$a \wedge b = 1 \quad \text{if and only if} \quad a=1 \text{ and } b=1$$

Boolean Algebra → Serial Switches

Conjunction (AND): ∧

$y = a \wedge b$



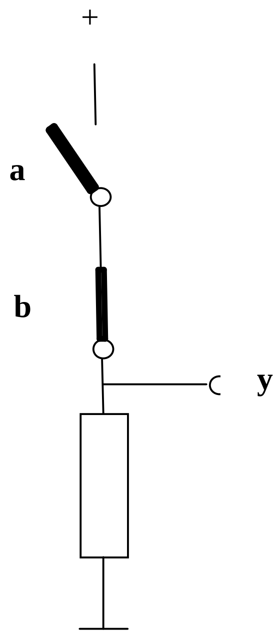
a	b	y
0	0	0

$0 \wedge 0 = 0$

Boolean Algebra → Serial Switches

Conjunction (AND): ∧

y = a ∧ b



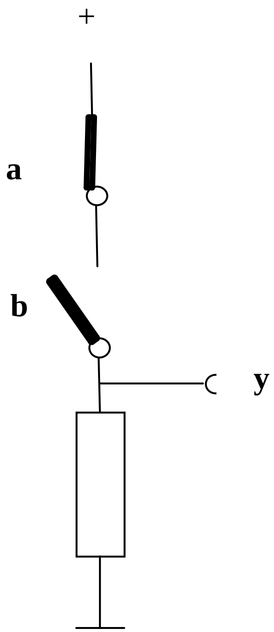
a	b	y
0	0	0
0	1	0

0 ∧ 0 = 0
0 ∧ 1 = 0

Boolean Algebra → Serial Switches

Conjunction (AND): ∧

$y = a \wedge b$



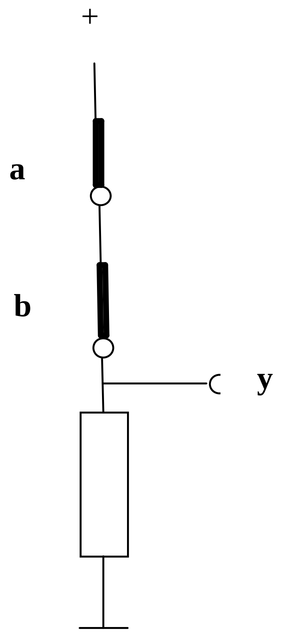
a	b	y
0	0	0
0	1	0
1	0	0

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

Boolean Algebra → Serial Switches

Conjunction (AND): ∧

$y = a \wedge b$



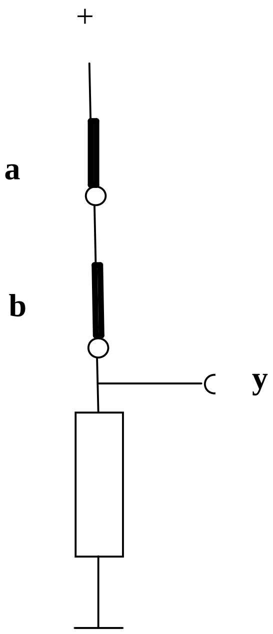
a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

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

Boolean Algebra → Serial Switches

SYNTAX → **Conjunction (AND): ∧**

$y = a \wedge b$



a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

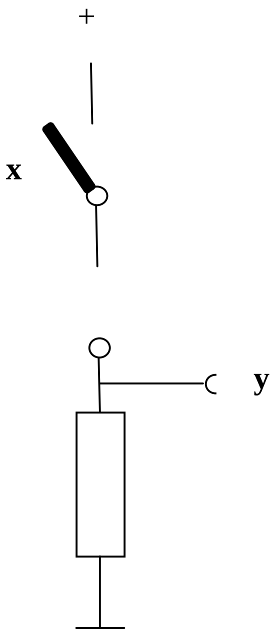
$0 \wedge 0 = 0$
$0 \wedge 1 = 0$
$1 \wedge 0 = 0$
$1 \wedge 1 = 1$

↑ SEMANTICS

Boolean Algebra → Serial Switches

Conjunction (AND): ∧

$y = a \wedge b$



a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

$0 \wedge 0 = 0$

$0 \wedge 1 = 0$

$1 \wedge 0 = 0$

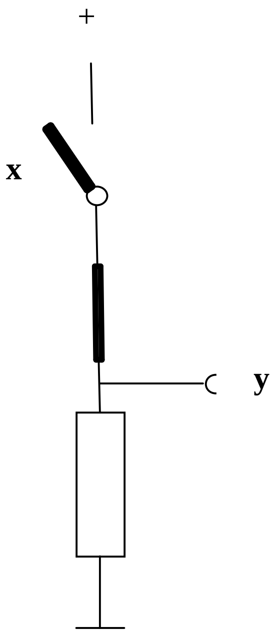
$1 \wedge 1 = 1$

$x \wedge 0 = 0$

Boolean Algebra → Serial Switches

Conjunction (AND): ∧

$y = a \wedge b$



a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

$0 \wedge 0 = 0$

$0 \wedge 1 = 0$

$1 \wedge 0 = 0$

$1 \wedge 1 = 1$

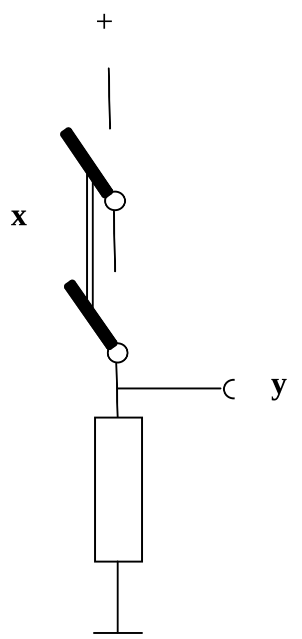
$x \wedge 0 = 0$

$x \wedge 1 = x$

Boolean Algebra → Serial Switches

Conjunction (AND): $\wedge$

$y = a \wedge b$



a	b	y
0	0	0
0	1	0
1	0	0
1	1	1

$0 \wedge 0 = 0$

$0 \wedge 1 = 0$

$1 \wedge 0 = 0$

$1 \wedge 1 = 1$

$x \wedge 0 = 0$

(Neutral Element 1) $x \wedge 1 = x$

(Idempotence) $x \wedge x = x$

Boolean Algebra

- Algebra on two-valued (binary) variables ← boolean variables
- Boolean operators:
  - Conjunction (AND): $\wedge$

    $a \wedge b = 1$ if and only if $a=1$ and $b=1$

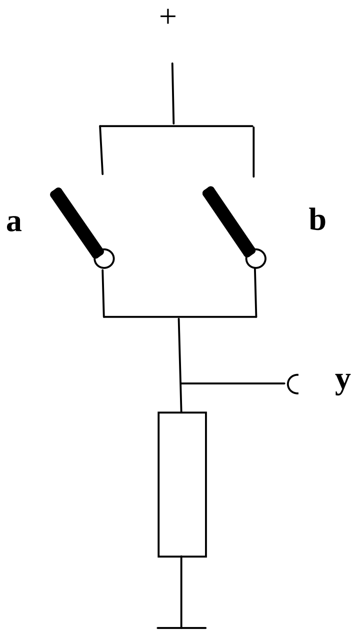
  - Disjunction (OR): $\vee$

    $a \wedge b = 1$ if and only if $a=1$ or $b=1$

Boolean Algebra → Parallel Switches

Disjunction (OR): $\vee$

$y = a \vee b$



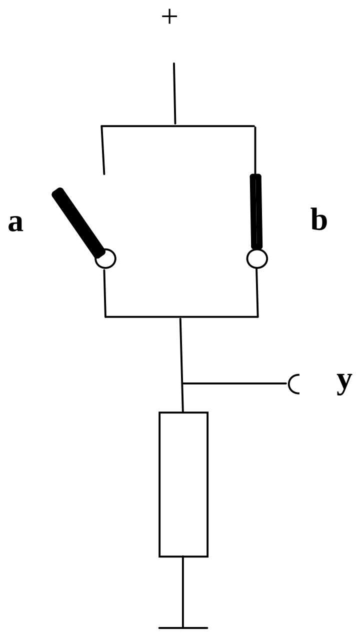
a	b	y
0	0	0

$0 \vee 0 = 0$

Boolean Algebra → Parallel Switches

Disjunction (OR): ∨

$y = a \lor b$



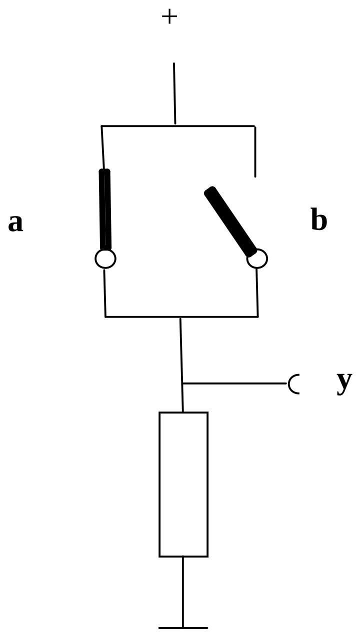
a	b	y
0	0	0
0	1	1

$0 \lor 0 = 0$
$0 \lor 1 = 1$

Boolean Algebra → Parallel Switches

Disjunction (OR): ∨

$y = a \vee b$



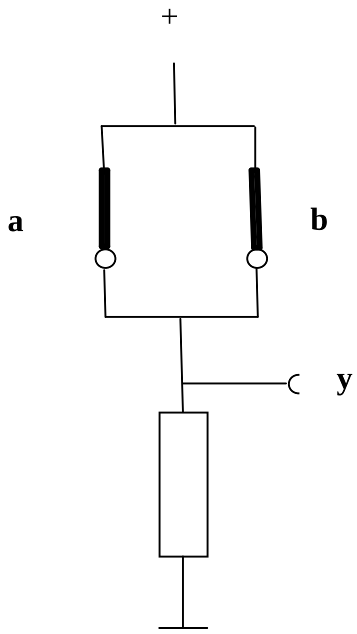
a	b	y
0	0	0
0	1	1
1	0	1

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

Boolean Algebra → Parallel Switches

Disjunction (OR): ∨

y = a ∨ b



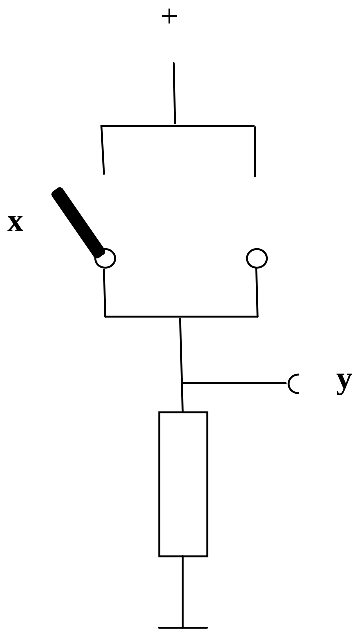
a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

0 ∨ 0 = 0
0 ∨ 1 = 1
1 ∨ 0 = 1
1 ∨ 1 = 1

Boolean Algebra → Parallel Switches

Disjunction (OR): ∨

$y = a \vee b$



a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

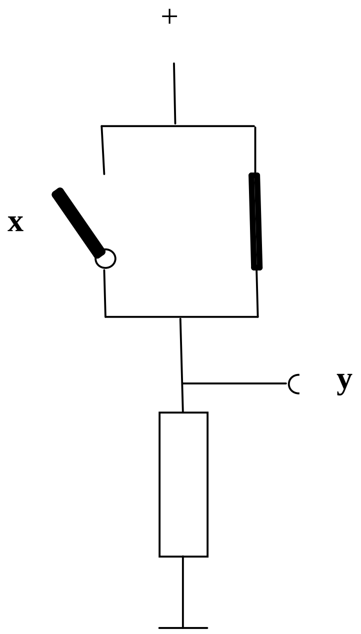
$0 \vee 0 = 0$
$0 \vee 1 = 1$
$1 \vee 0 = 1$
$1 \vee 1 = 1$

(Neutral Element 0) **$x \vee 0 = x$**

Boolean Algebra → Parallel Switches

Disjunction (OR): $\vee$

$y = a \vee b$



a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

$0 \vee 0 = 0$

$0 \vee 1 = 1$

$1 \vee 0 = 1$

$1 \vee 1 = 1$

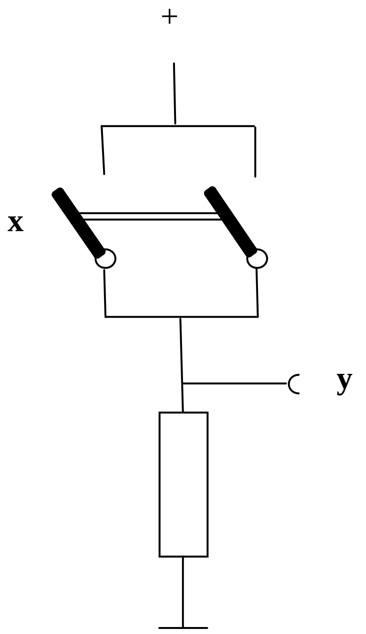
(Neutral Element 0) $x \vee 0 = x$

$x \vee 1 = 1$

Boolean Algebra → Parallel Switches

Disjunction (OR): $\vee$

$$y = a \vee b$$



a	b	y
0	0	0
0	1	1
1	0	1
1	1	1

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

(Neutral Element 0) $x \vee 0 = x$

$x \vee 1 = 1$

(Idempotence) $x \vee x = x$

Boolean Algebra

- Algebra on two-valued (binary) variables ← boolean variables
- Boolean operators:
  - Conjunction (AND): ∧

    $a \wedge b = 1$ if and only if $a=1$ and $b=1$

  - Disjunction (OR): ∨

    $a \wedge b = 1$ if and only if $a=1$ or $b=1$

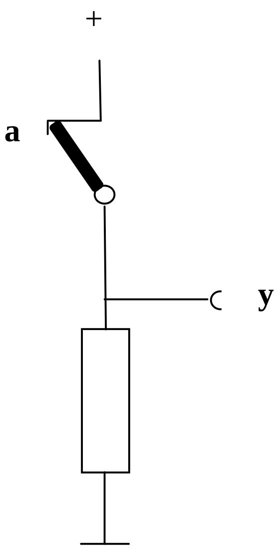
  - Negation (NOT): ¬

    $\neg a = 1$ if and only if $a=0$

Boolean Algebra

Negation (NOT): ¬

y = ¬ a



a	y
0	1
1	0

¬ 0 = 1
¬ 1 = 0

Boolean Algebra

- Algebra on two-valued (binary) variables ← boolean variables
- Boolean operators (lowest to highest precedence): ∧, ∨, ¬

a	b	a ∧ b
0	0	0
0	1	0
1	0	0
1	1	1

a	b	a ∨ b
0	0	0
0	1	1
1	0	1
1	1	1

a	¬a
0	1
1	0

Truth Tables **(Wahrheitstafel)**

Example: Construct the truth table of the *boolean function* **f=a ∨ (¬ b ∧ c).**

Boolean Algebra – Axioms/Laws (Gesetz)

Neutral elements (identity)	$a \wedge 1 = a$
	$a \vee 0 = a$
Zero elements	$a \wedge 0 = 0$
	$a \vee 1 = 1$
Idempotence	$a \wedge a = a$
	$a \vee a = a$
Negation	$a \wedge \neg a = 0$
	$a \vee \neg a = 1$
Commutative	$a \wedge b = b \wedge a$
	$a \vee b = b \vee a$
Associative	$a \wedge (b \wedge c) = (a \wedge b) \wedge c$
	$a \vee (b \vee c) = (a \vee b) \vee c$
Distributive	$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$
	$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$

Note: Every axiom has a *dual* one: replace $\vee$, 0 with $\wedge$, 1.

Duality: $\neg f(x,y) = g(\neg x, \neg y)$

Boolean Algebra – DeMorgan Axioms

- DeMorgan's Theorems for negating a boolean function:
  - Negate boolean variables;
  - Change $\wedge$ to $\vee$ and $\vee$ to $\wedge$;
- DeMorgan's axioms:

$$\neg (a \wedge b) = (\neg a) \vee (\neg b)$$ ← called a NAND

$$\neg (a \vee b) = (\neg a) \wedge (\neg b)$$ ← called a NOR

Boolean Algebra – Axioms with DeMorgan's Axioms

Neutral elements (identity)	$a \wedge 1 = a$
	$a \vee 0 = a$
Zero elements	$a \wedge 0 = 0$
	$a \vee 1 = 1$
Idempotence	$a \wedge a = a$
	$a \vee a = a$
Negation	$a \wedge \neg a = 0$
	$a \vee \neg a = 1$
Commutative	$a \wedge b = b \wedge a$
	$a \vee b = b \vee a$
Associative	$a \wedge (b \wedge c) = (a \wedge b) \wedge c$
	$a \vee (b \vee c) = (a \vee b) \vee c$
Distributive	$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$
	$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
DeMorgan's	$\neg (a \wedge b) = (\neg a) \vee (\neg b)$
	$\neg (a \vee b) = (\neg a) \wedge (\neg b)$

Proof using Axioms

- Axioms can be used to prove new *theorems:*
  - Theorems are of the form f=g (f, g are boolean functions);

  - Proof by using *simplifications* of f to yield g.
    - Simplification of f:
    - transforming f by repeated application of axioms;
    - yields an equivalent new function with fewer boolean operators.

- Examples:
  - $\neg(a \vee b) \wedge \neg((\neg a) \vee (\neg b)) = 0$
  - Absorption Axioms:

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

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

Boolean Algebra – Special Boolean Functions

XOR: Exclusive OR (Antivalenz)

$$y = a \neq b$$

y is 1 if exactly one of the variables a and b is 1.

a	b	y
0	0	0
0	1	1
1	0	1
1	1	0

$0 \neq 0 = 0$

$0 \neq 1 = 1$

$1 \neq 0 = 1$

$1 \neq 1 = 0$

$x \neq 0 = x$

$x \neq 1 = \neg x$

$x \neq x = 0$

Boolean Algebra – Special Boolean Functions

a implies b

if a is 1 then b is also 1

Implication $\Rightarrow$

$$y = a \Rightarrow b$$

y is 1 if and only if a is 0 or b is 1

a	b	y
0	0	1
0	1	1
1	0	0
1	1	1

Boolean Algebra – Special Boolean Functions

a implies b

if a is 1 then b is also 1

Implication ⇒

$$y = a \Rightarrow b = \neg a \vee b$$

y is 1 if and only if a is 0 or b is 1

a	b	y
0	0	1
0	1	1
1	0	0
1	1	1

Boolean Algebra – Special Boolean Functions

Equivalence $\Leftrightarrow$

$$y = a \Leftrightarrow b = (a \Rightarrow b) \wedge (b \Rightarrow a)$$

a	b	y
0	0	1
0	1	0
1	0	0
1	1	1

Boolean Algebra – Precedence of Boolean Operators

Operator	Name	Priority (Precedence)
$\neg$	Negation	4
$\wedge$	Conjunction	3
$\vee$	Disjunction	3
$\Rightarrow$	Implication	2
$\Leftrightarrow$	Equivalence	1

Example: $\neg$ a $\wedge$ b $\Rightarrow$ c $\vee$ d $\Leftrightarrow$ e

Boolean Algebra – truth value (Wahrheit) of a boolean function f

- f is a *tautology* if:

  f =1 in *all* situations (f is always 1)

  Example: $(a \wedge (a \Rightarrow b)) \Rightarrow b$ (Modus Ponens Axiom)

- f is *contradiction* if:

  f = 0 in *all* situations (f is always 0)

  Example: $a \wedge \neg a$

- f is *satisfiable (valid, erfüllbar)* if:

  f = 1 in *some* situation

  Example: $(a \Rightarrow b) \wedge (a \wedge b \Rightarrow c) \Rightarrow (a \Rightarrow c)$ is a tautology and thus satisfiable.
  $(a \Rightarrow b) \wedge c$ is satisfiable, but is not a tautology

Boolean Algebra – checking satisfiability is hard

I can't get no satisfaction

And I try,

And I try,

And I try,

And I try, …

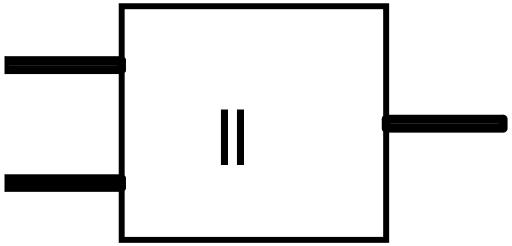
The Rolling Stones

Boolean Algebra – All Boolean Functions with 2 Parameters

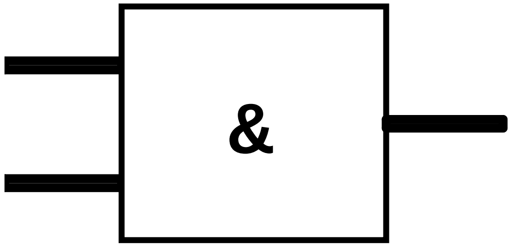
a 0011 b 0101	
0000	$y_0 = 0$
0001	$y_1 = a \wedge b$
0010	$y_2 = a \wedge \neg b$
0011	$y_3 = a$
0100	$y_4 = \neg a \wedge b$
0101	$y_5 = b$
0110	$y_6 = a \neq b$
0111	$y_7 = a \vee b$

a 0011 b 0101	
1111	$y_{15} = 1$
1110	$y_{14} = \neg a \vee \neg b$
1101	$y_{13} = \neg a \vee b$
1100	$y_{12} = \neg a$
1011	$y_{11} = a \vee \neg b$
1010	$y_{10} = \neg b$
1001	$y_9 = a \Leftrightarrow b$
1000	$y_8 = \neg a \wedge \neg b$

Half Adder (Halbaddierwerk)

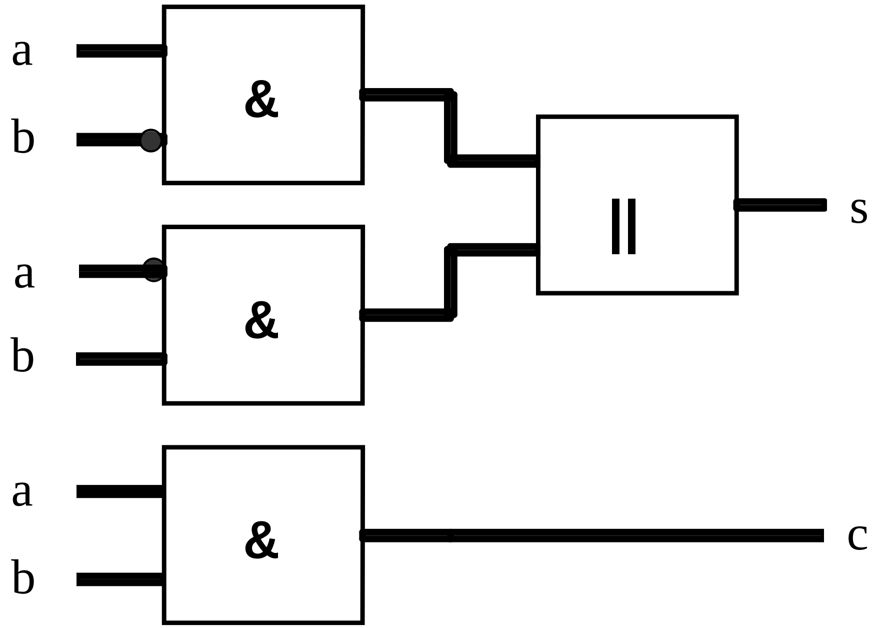


OR-Gate (Gatter)



AND-Gate

Half Adder



XOR

a	b	c	s
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

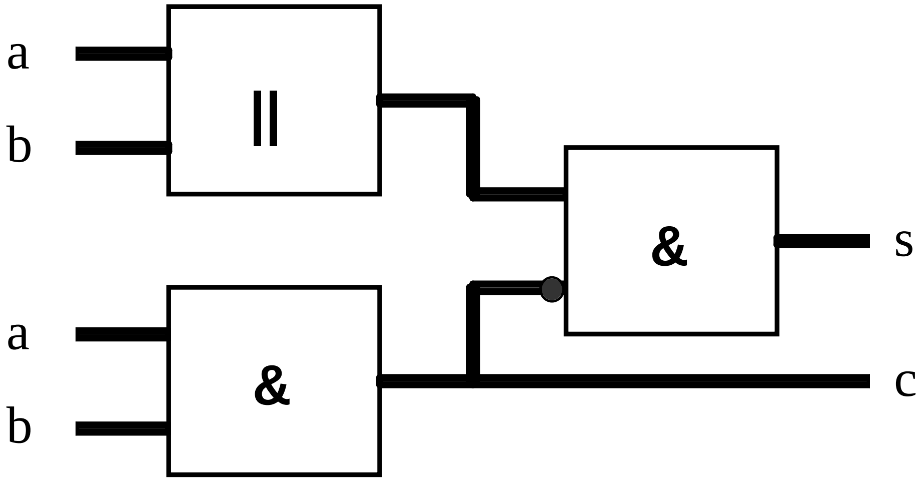
$s = (\neg a \wedge b) \vee (a \wedge \neg b)$
$c = (a \wedge b)$

Half Adder – Simplifications

$$s = (\neg a \wedge b) \vee (a \wedge \neg b) =$$
$$= ((\neg a \wedge b) \vee a) \wedge ((\neg a \wedge b) \vee \neg b) =$$
$$= (\neg a \vee a) \wedge (b \vee a) \wedge (\neg a \vee \neg b) \wedge (b \vee \neg b) =$$
$$= (b \vee a) \wedge (\neg a \vee \neg b) =$$
$$= (a \vee b) \wedge \neg (a \wedge b)$$

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

Half Adder - Optimized



a	b	c	s
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

$s = (a \lor b) \land \neg(a \land b)$
$c = (a \land b)$