# COL215 Digital Logic and System Design

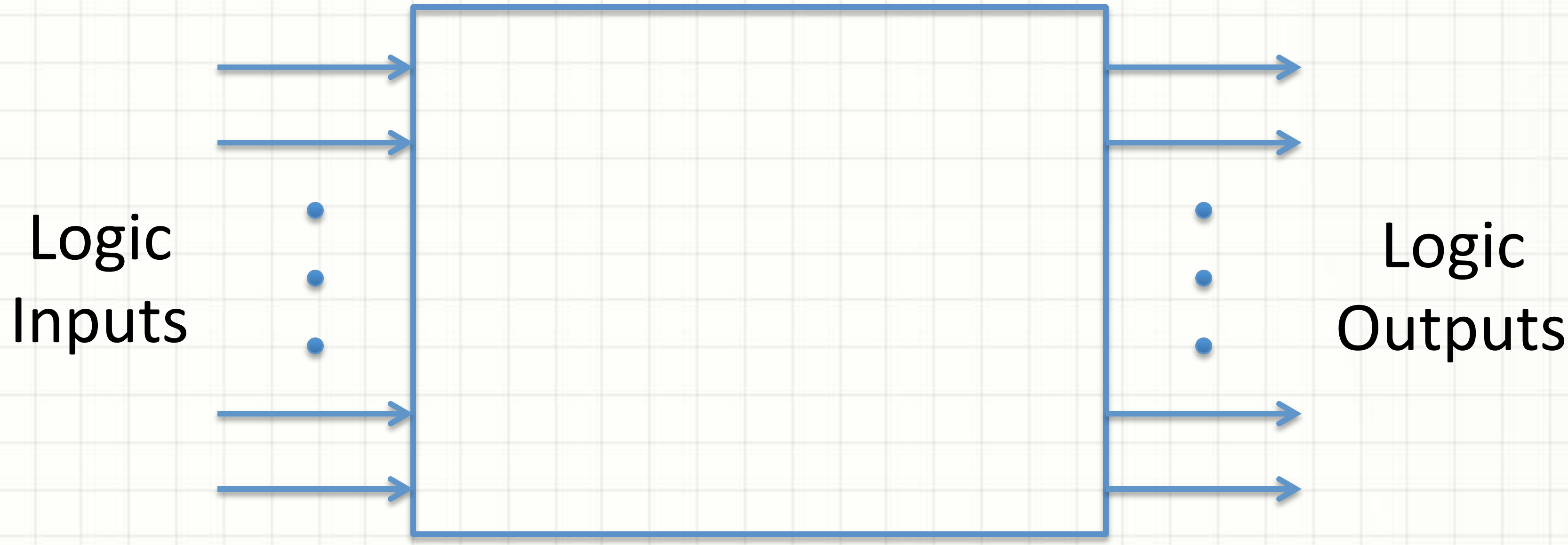
Combinational Logic Circuits

26 July 2017

# Lecture Outline

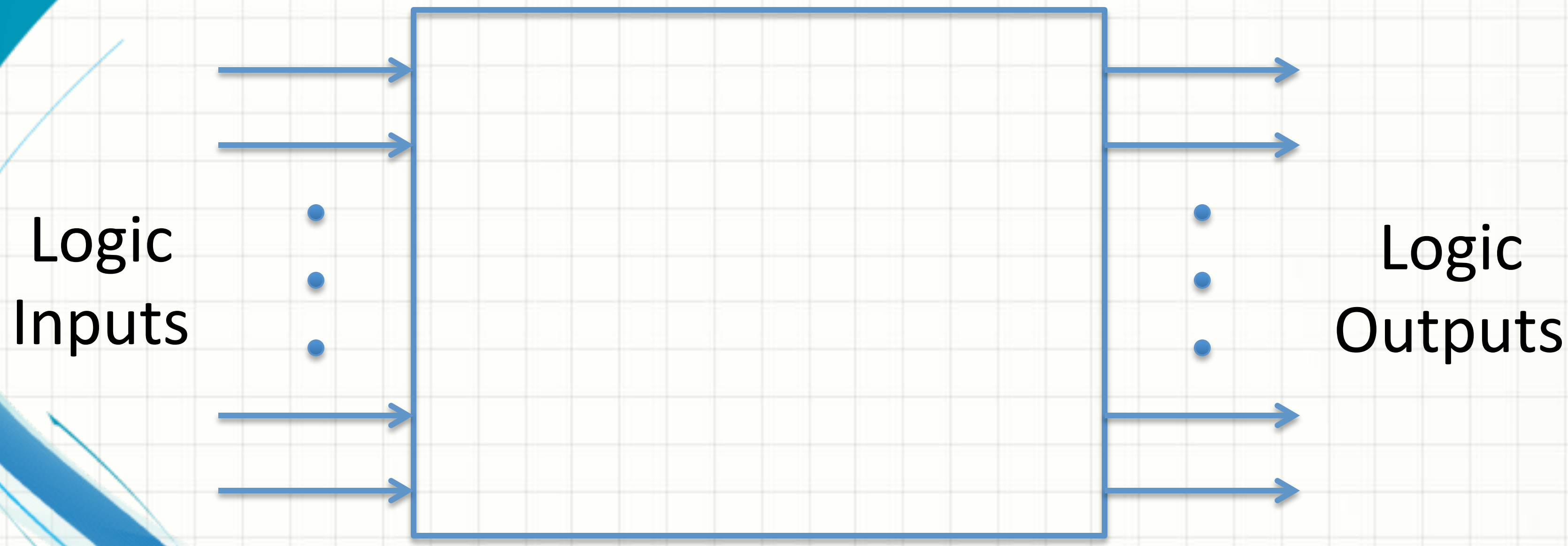
- Logic circuits
- Combinational vs sequential circuits
- Building blocks – Logic gates
- Putting gates together
- Logic simplification
- Boolean Algebra
- Special forms of circuits

# Logic Circuits



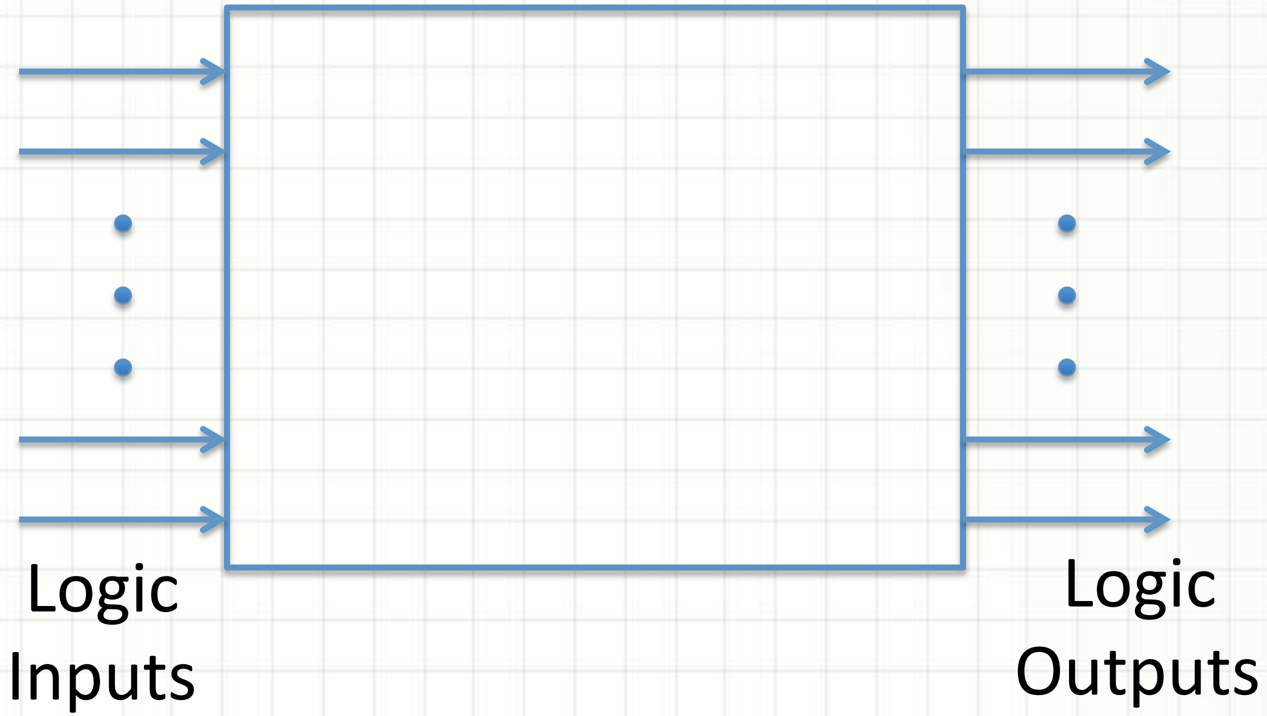
# Combinational Logic Circuits

Output values depend on the current combination of input values



# Sequential Logic Circuits

Output values depend on the current and previous input values



# Dependence on previous input values

- The circuit “remembers” the previous input values !
- Practical circuits have “finite” memory
- Previous input sequences can be represented by a finite number of states

# Example: Car Interior Light Control

- Switch OFF/DOOR/ON
- Doors open/close (4)

# More sophisticated control

- Switch OFF/DOOR/ON
- Doors open/close (4)
- Door key
- Ignition key
- Gradual fading
- Time-out

# Typical sequence: Entry

- Unlock the door
- Open the door
- Enter the car
- Close the door
- Put key in ignition switch
- Switch on ignition

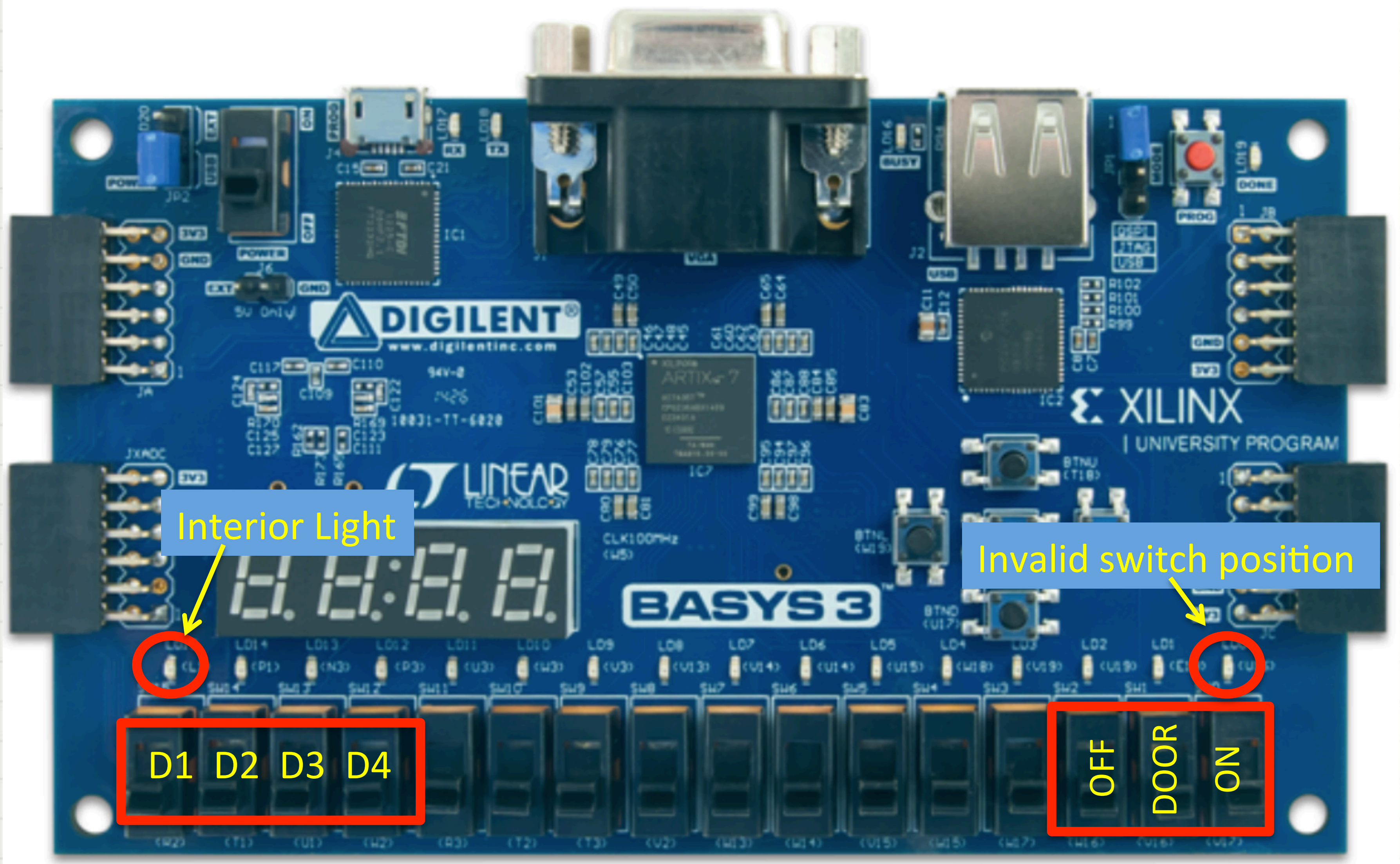
# Typical sequence: Exit

- Switch off ignition
- Take ignition key out
- Open the door
- Exit the car
- Close the door
- Lock the door

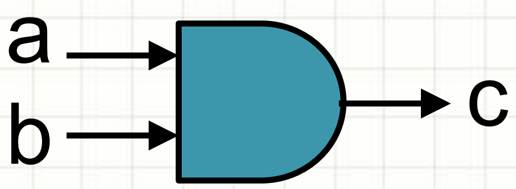
# Additional Logic

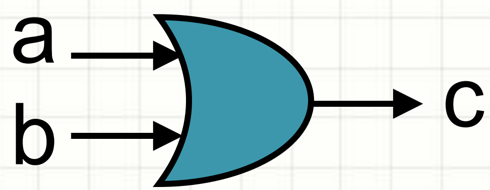
- When the light is off, opening any door turns it on and closing the door makes it turn off gradually
- When the light is on, it times out in 8-10 sec if there is no event

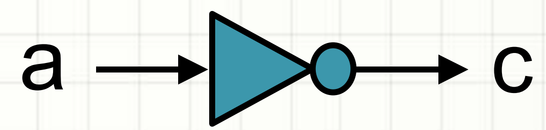
# Lab Exercise 1



# Building Blocks: Gates




c = a AND b

$c = a . b$

| a | b | c |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

c = a OR b

$c = a + b$

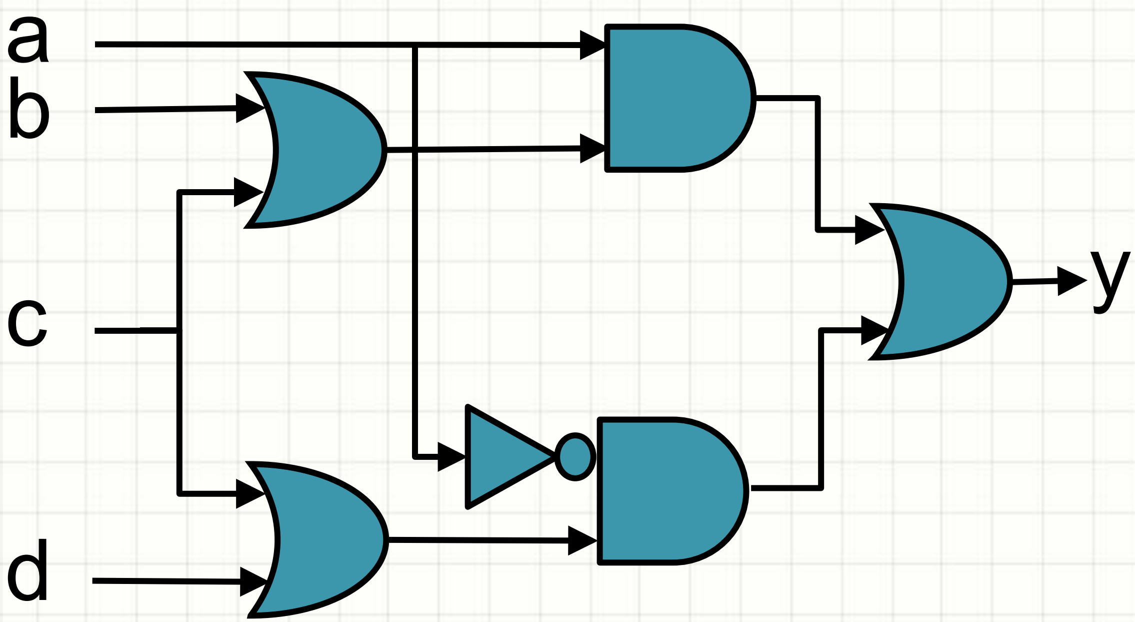
| a | b | c |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

c = NOT a

$c = \bar{a}$ or $c = a'$

| a | c |
|---|---|
| 0 | 1 |
| 1 | 0 |

# Putting Gates together

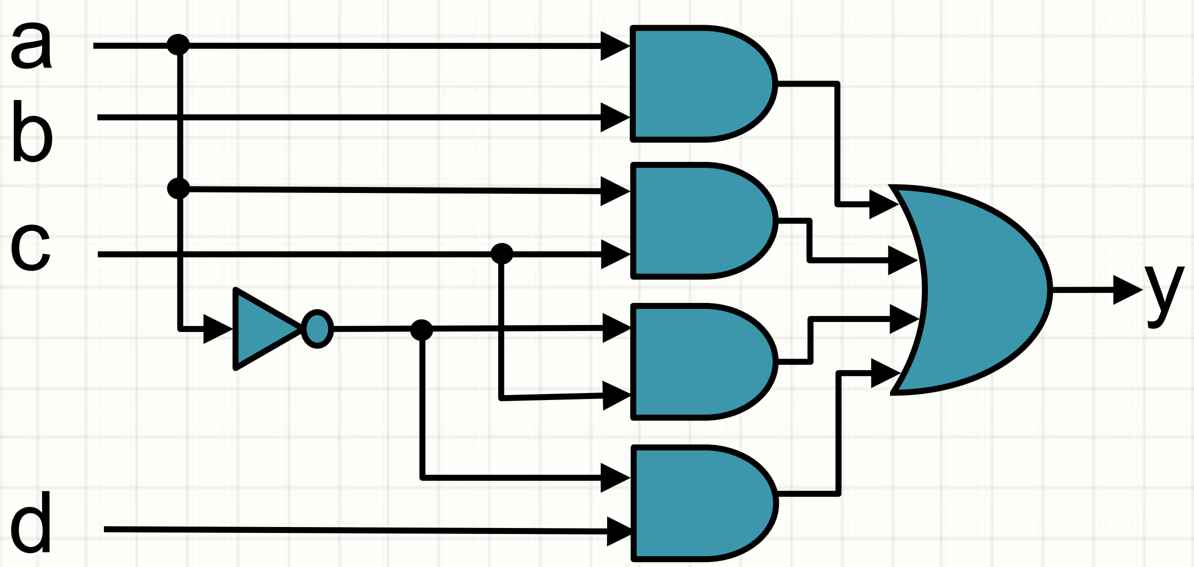



y = (a AND (b OR c)) OR (NOT a AND (c OR d))

$y = (a \cdot (b + c)) + (\bar{a} \cdot (c + d))$

| a | b | c | d | y |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

# Another Circuit




| a | b | c | d | y |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

y = (a AND b) OR (a AND c) OR
(NOT a AND c) OR (NOT a AND d)

$y = (a \cdot b) + (a \cdot c) + (\overline{a} \cdot c) + (\overline{a} \cdot d)$

**This is Sum of Products (SOP) form**

# How to compare two circuits?

- Compare truth tables, row by row, exhaustively!
- Check if two expressions are "equivalent"
- Apply rules, for example,

$$a \,.\, (b + c) \equiv (a \,.\, b) + (a \,.\, c)$$

Among equivalent expressions/circuits, can we identify the best one? Criteria: cost/delay

# Transforming logic expressions: Boolean Algebra

Set B, operations . & +, identity 0 & 1, inverse ’

| | |
|---|---|
| $a + a = a \, . \, a = a$ | idempotent |
| $a + b = b + a$ | commutative |
| $a \, . \, b = b \, . \, a$ | commutative |
| $a + (b + c) = (a + b) + c$ | associative |
| $a \, . \, (b \, . \, c) = (a \, . \, b) \, . \, c$ | associative |
| $a \, . \, (b + c) = (a \, . \, b) + (a \, . \, c)$ | distributive |
| $a + (b \, . \, c) = (a + b) \, . \, (a + c)$ | distributive |

# Boolean Algebra – contd.

| | | |
|---|---|---|
| $a \cdot 1 = a$ | $a + 0 = a$ | def: identity elements |
| $a \cdot a' = 0$ | $a + a' = 1$ | def: inverse/complement |
| $(a')' = a$ | | (by symmetry) |
| $a \cdot 0 = 0$ | $a + 1 = 1$ | (derived) |

# Derived results

$(a + b)' = a' . b'$ DeMorgan's theorem

$(a . b)' = a' + b'$

$x + (x . y) = x$ Absorption

$x . (x + y) = x$

$x + (x' . y) = x + y$

$x . (x' + y) = x . y$

$(x . y) + (y . z) + (x' . z) = (x . y) + (x' . z)$

$(x + y) . (y + z) . (x' + z) = (x + y). (x' + z)$

# Duality Principle

Take any Boolean Algebra equality

Replace . with + and vice versa

Replace 1 with 0 and vice versa

The equality will still hold

# Prove $[x + (x \cdot y) = x]$

$[x + (x \cdot y) = x]$

LHS $= x + (x \cdot y)$

$= (x \cdot 1) + (x \cdot y)$

$= x \cdot (1 + y)$

$= x \cdot 1$

$= x =$ RHS

# Prove $[x + (x' . y) = x + y]$

$[x + (x' . y) = x + y]$

LHS $= x + (x' . y)$

$= x + (x . y) + (x' . y)$

$= x + (x + x') . y$

$= x + 1 . y$

$= x + y =$ RHS

# Prove $(x.y)+(y.z)+(x'.z) = (x.y)+(x'.z)$

$[(x \, . \, y) + (y \, . \, z) + (x' \, . \, z) = (x \, . \, y) + (x' \, . \, z)]$

LHS $= (x \, . \, y) + (y \, . \, z) + (x' \, . \, z)$

$= (x \, . \, y) + (1 \, . \, y \, . \, z) + (x' \, . \, z)$

$= (x \, . \, y) + ((x + x') \, . \, y \, . \, z) + (x' \, . \, z)$

$= (x \, . \, y) + (x \, . \, y \, . \, z) + (x' \, . \, y \, . \, z) + (x' \, . \, z)$

$= (x \, . \, (y + (y \, . \, z))) + (x' \, . \, ((y \, . \, z) + z))$

$= (x \, . \, (y)) + (x' \, . \, (z))$

$= (x \, . \, y) + (x' \, . \, z) =$ RHS

# Prove $[(a + b)' = a' . b']$

$[(a + b)' = a' . b']$

$y = x'$ iff $[\, x . y = 0$ and $x + y = 1 \,]$

- $(a + b) . (a' . b')$

  $= (a . a' . b') + (b . a' . b') = (0 . b') + (a' . 0)$

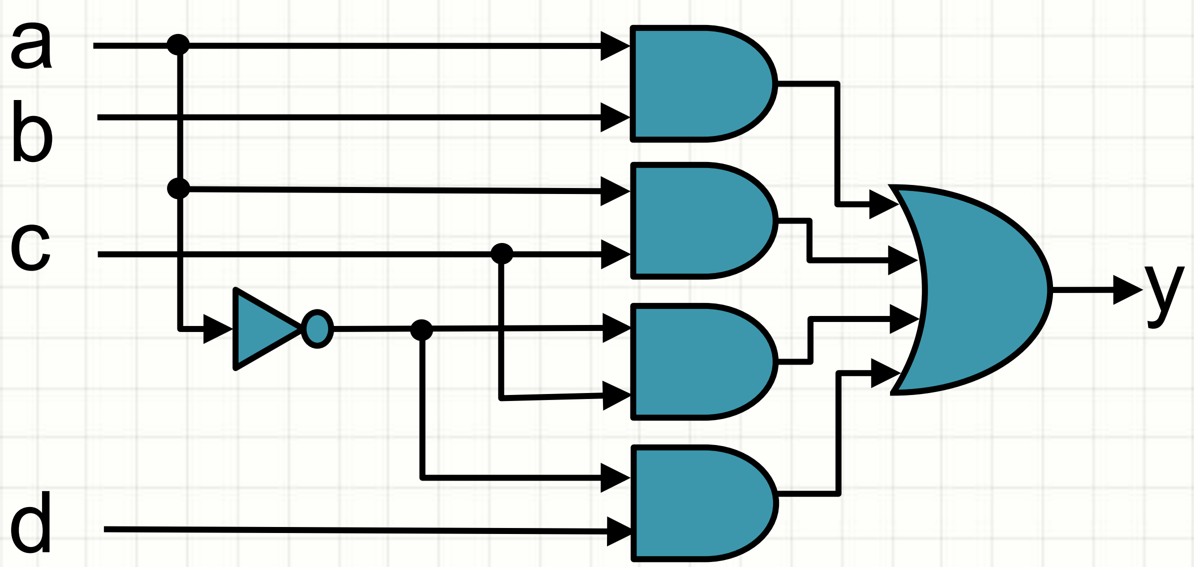
  $= 0 + 0 = 0$

- $(a + b) + (a' . b')$

  $= a + (b + (a' . b')) = a + (b + a')$

  $= (a + a') + b = 1 + b = 1$

# Finding POS form




| a | b | c | d | y |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 |

**Sum of Product (SOP) form**

$y = (a \,.\, b) + (a \,.\, c) + (a' \,.\, c) + (a' \,.\, d)$

$y' = a'b'c'd' + a'b\,c'd' + a\,b'c'd' + a\,b'c'd$

$\quad = a'c'd' + a\,b'c'$

$y = (a'c'd' + a\,b'c')' = (a'c'd')' \,.\, (a\,b'c')'$

$\quad = (a + c + d) \,.\, (a' + b + c)$

**This is Product of Sums (POS) form**

# Questions?