EECS 270 FA24
Midterm Exam Review

February 20th, 2025
By: Mick Gordinier

Midterm Exam Logistics (Date/Time)

- Midterm Exam Date/Time: **Monday February 24, 6:00 pm - 8:00 pm**
  - A - L: 220 CHRYS
  - M - V: G906 COOL
  - W - Z: 1109 FXB

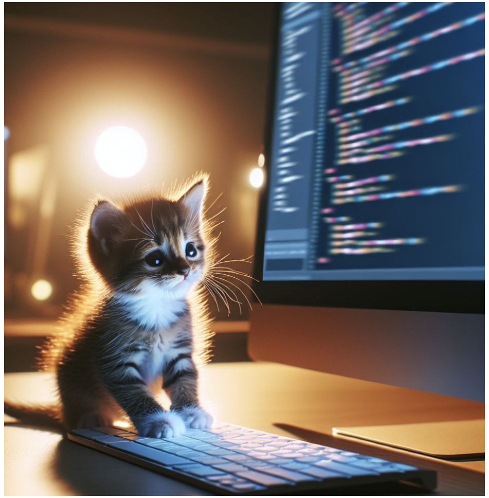
- SSD Accommodations
  - **Monday February 24, 5:00 pm - 8:00 pm**, EECS 270 Lab (2322 EECS)

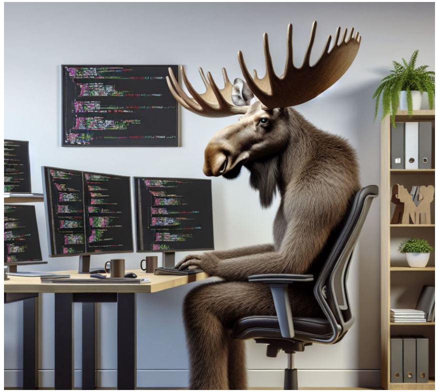
- Closed book except for **one 8.5"x11" sheet**
- No electronics

DOUBLE CHECK THE CANVAS ANNOUNCEMENT TO CONFIRM / UPDATES

Midterm Exam Logistics (Topics Covered)

2. Bits Everywhere!
3. Timing and Delay
4. Boolean Algebra
5. Switching Functions
6. Positive Binary Numbers
7. Binary Arithmetic I
8. Binary Arithmetic ||
9. Combinational Blocks

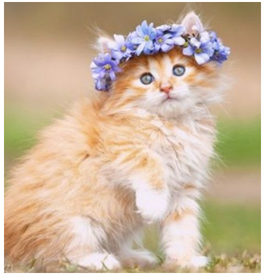




Let’s Get Started

Boolean Algebra

Formal Definition of Boolean Algebra

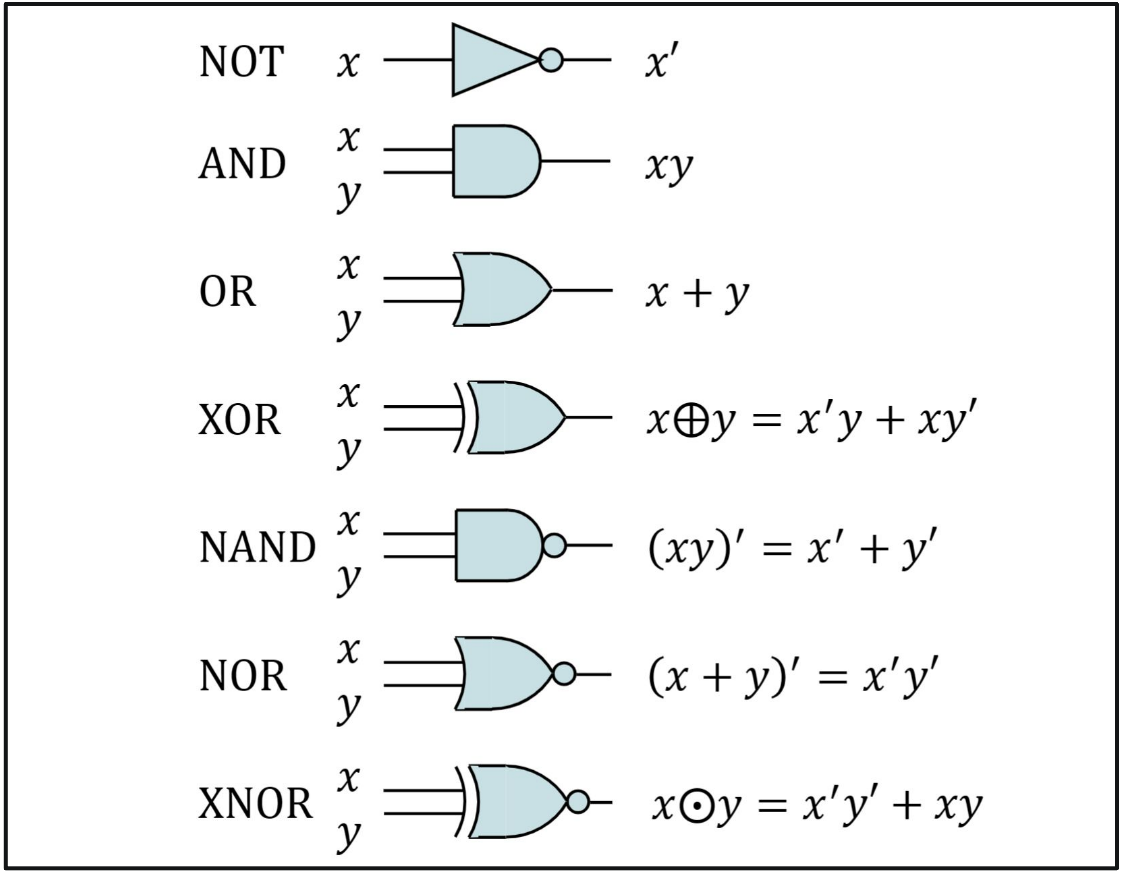


- Boolean/Switching Algebra
  - Basic “language” for combinational and sequential switching circuits
- Base Set: $B_2 = \{0, 1\}$
- Big 3 Operations
  - **NOT (x’)**
  - **AND (x & y)**
  - **OR (x | y)**

Postulate	Defines	A	B = A^D
P1	Switching Variables	$x = 0$ iff $x \neq 1$	$x = 1$ iff $x \neq 0$
P2	NOT	$0' = 1$	$1' = 0$
P3		$0 \cdot 0 = 0$	$1 + 1 = 1$
P4	AND / OR	$1 \cdot 1 = 1$	$0 + 0 = 0$
P5		$0 \cdot 1 = 1 \cdot 0 = 0$	$0 + 1 = 1 + 0 = 1$

- Postulates/Axioms - Mostly defining how the big three operations works
- Duality: $0 \leftrightarrow 1$, $\& \leftrightarrow |$ **(NOTE: THE LITERALS DO NOT NEGATE)**

Some of the Logic Gates You Should Know

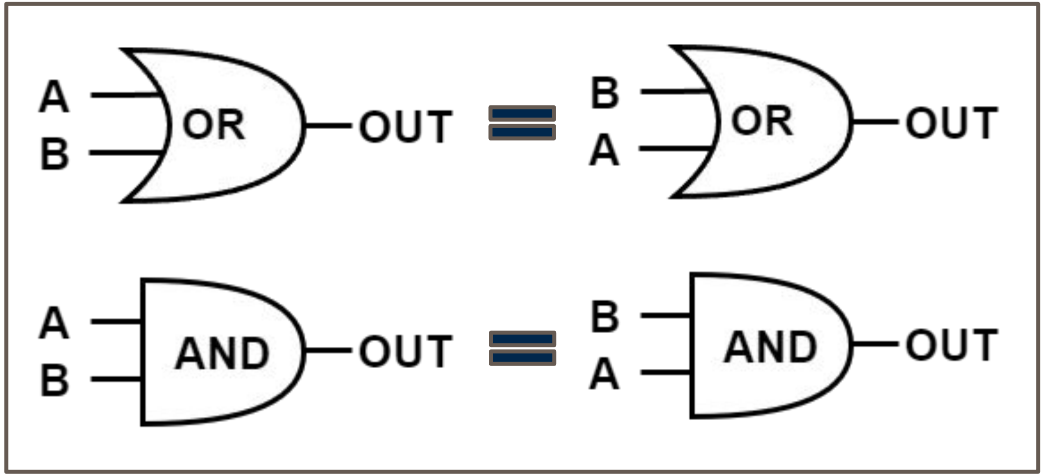


Building Theorems From Postulates

	A	Name	B
T1	$x \cdot 1 = x$	Identities	$x + 0 = x$
T2	$x \cdot 0 = 0$	Null Elements	$x + 1 = 1$
T3	$x \cdot x = x$	Idempotency	$x + x = x$
T4		Involution $(x')' = x$	
T5	$x \cdot x' = 0$	Complements	$x + x' = 1$
T6	$x \cdot y = y \cdot x$	Commutativity	$x + y = y + x$
T7	$x \cdot (x + y) = x$	Absorption	$x + (x \cdot y) = x$
T8	$x \cdot (x' + y) = x \cdot y$	No Name	$x + (x' \cdot y) = x + y$
T9	$(x \cdot y) \cdot z = x \cdot (y \cdot z)$	Associativity	$(x + y) + z = x + (y + z)$
T10	$x \cdot (y + z) = x \cdot y + x \cdot z$	Distributivity	$x + (y \cdot z) = (x + y) \cdot (x + z)$
T11	$x \cdot y + x' \cdot z + y \cdot z$ $= x \cdot y + x' \cdot z$	Consensus	$(x + y) \cdot (x' + z) \cdot (y + z)$ $= (x + y) \cdot (x' + z)$
T12		De Morgan's $f(x_1, \ldots, x_n, 0, 1, \cdot, +)' = f(x_1', \ldots, x_n', 1, 0, +, \cdot)$	

More Core Boolean Algebra Theorems

	A	Name	B
T6	$x \cdot y = y \cdot x$	Commutativity	$x + y = y + x$
T9	$(x \cdot y) \cdot z = x \cdot (y \cdot z)$	Associativity	$(x + y) + z = x + (y + z)$
T10	$x \cdot (y + z) = x \cdot y + x \cdot z$	Distributivity	$x + (y \cdot z) = (x + y) \cdot (x + z)$



De Morgan's Laws (Really Good One to Know)

- How we distribute a function inversion

T12	De Morgan's $f(x_1, \ldots, x_n, 0, 1, \cdot, +)' = f(x_1', \ldots, x_n', 1, 0, +, \cdot)$

$(X \bullet Y)' = X' + Y'$

$(X + Y)' = X' \bullet Y'$

$(X \bullet (X + Y))'$
$= X' + (X + Y)'$
$= X' + (X' \bullet Y')$
$= (X' + X') \bullet (X' + Y')$
$= (X') \bullet (X' + Y') = \mathbf{X'}$

$[(W \bullet Y \bullet 1) + ((Z \bullet Y) + 0)]'$
$= (W \bullet Y \bullet 1)' \bullet ((Z \bullet Y) + 0)'$
$= (W' + Y' + 0) \bullet ((Z' + Y') \bullet 1)$
$= (W' + Y') \bullet (Z' + Y')$
$= \mathbf{Y' + (W' \bullet Z')}$

Dual and De Morgan's Misconceptions

1. **Dual ($f^D(x)$) != Negation ($(f(x))'$)**
   a. Dual - Constant 0 ↔ 1 , & ↔ | **(NOTE: THE LITERALS DO NOT NEGATE)**
      i. Used primarily in Theory Proving
      ii. (X + Y) = 1 → Dual → (X • Y) = 0
      iii. (X' + Y)' = 0 → Dual → (X' • Y)' = 1

   b. De Morgan's - Used for handling negation and distributing it
      i. (X + Y)' = X' • Y'
      ii. (X • Y)' = X' + Y'

2. **Dual ($f^D(x)$) != Original ($f(x)$) (usually)**
   a. $(X + Y)^D = (X \cdot Y)$!= $(X + Y)$

Distribution → “No Name” Theorem

- Distribution
  - $A \cdot (X + Y) = (A \cdot X) + (A \cdot Y)$ (AND distributes across the ORs)
  - $A + (X \cdot Y) = (A + X) \cdot (A + Y)$ (OR distributes across the ANDs) (DUALITY)

$X' + XY = ?$

Distribution → "No Name" Theorem

- Distribution
  - **$A \cdot (X + Y) = (A \cdot X) + (A \cdot Y)$** (AND distributes across the ORs)
  - **$A + (X \cdot Y) = (A + X) \cdot (A + Y)$** (OR distributes across the ANDs) (DUALITY)

$X' + XY = (X' + X) \cdot (X' + Y)$ (Distribution)

$= ?$

Distribution → "No Name" Theorem

- Distribution
  - $A \cdot (X + Y) = (A \cdot X) + (A \cdot Y)$ (AND distributes across the ORs)
  - $A + (X \cdot Y) = (A + X) \cdot (A + Y)$ (OR distributes across the ANDs) (DUALITY)

$$X' + XY = (X' + X) \cdot (X' + Y) \qquad \text{(Distribution)}$$

$$= (1) \cdot (X' + Y) \qquad \text{(Complements)}$$

$$= ?$$

Distribution → "No Name" Theorem

- Distribution
  - **$A \cdot (X + Y) = (A \cdot X) + (A \cdot Y)$** (AND distributes across the ORs)
  - **$A + (X \cdot Y) = (A + X) \cdot (A + Y)$** (OR distributes across the ANDs) (DUALITY)

$$
\begin{aligned}
X' + XY &= (X' + X) \cdot (X' + Y) && \text{(Distribution)} \\
&= (1) \cdot (X' + Y) && \text{(Complements)} \\
&= X' + Y && \text{(Identities)}
\end{aligned}
$$

(NO NAME THEOREM)

More Theorem Practice

Let D = (X □ Y)   and   P = ((X □ Y) • Z)’

→ Q = (D • P)’   **(What Gate is this)?**

= (De Morgan’s)

= (Substitution)

= …

= ??? (GOAL: USE “No Name” Theorem)

- “No Name” Theorem (Derived from Distribution)
  - **X • (X’ + Y) = X • Y**
  - **X + (X’ • Y) = X + Y**

More Theorem Practice

Let $D = (X \square Y)$ and $P = ((X \square Y) \bullet Z)'$

$\rightarrow Q = (D \bullet P)'$ **(NAND)**

= (De Morgan's)

= (Substitution)

= …

= ??? (GOAL: USE "No Name" Theorem)

- "No Name" Theorem (Derived from Distribution)
  - $\mathbf{X \bullet (X' + Y) = X \bullet Y}$
  - $\mathbf{X + (X' \bullet Y) = X + Y}$

More Theorem Practice

Let D = (X □ Y)   and   P = ((X □ Y) • Z)'

→ Q = (D • P)'   **(NAND)**

= D' + P'   **(De Morgan's)**

= (Substitution)

= …

= ??? (GOAL: USE "No Name" Theorem)

- "No Name" Theorem (Derived from Distribution)
  - **X • (X' + Y) = X • Y**
  - **X + (X' • Y) = X + Y**

More Theorem Practice

Let $D = (X \square Y)$ and $P = ((X \square Y) \cdot Z)'$

$\rightarrow Q = (D \cdot P)'$ **(NAND)**

$= D' + P'$ **(De Morgan's)**

$= (X \square Y)' + (((X \square Y) \cdot Z)')'$ **(Substitution)**

= …

= ??? (GOAL: USE "No Name" Theorem)

- "No Name" Theorem (Derived from Distribution)
  - $\mathbf{X \cdot (X' + Y) = X \cdot Y}$
  - $\mathbf{X + (X' \cdot Y) = X + Y}$

More Theorem Practice

Let $D = (X \square Y)$ and $P = ((X \square Y) \cdot Z)'$

$\rightarrow Q = (D \cdot P)'$ **(NAND)**

$= D' + P'$ **(De Morgan's)**

$= (X \square Y)' + (((X \square Y) \cdot Z)')'$ **(Substitution)**

$= (X \square Y)' + ((X \square Y) \cdot Z)$ **(NOT/NOT Cancellation)**

= …

= ??? (GOAL: USE "No Name" Theorem)

- "No Name" Theorem (Derived from Distribution)
  - $\mathbf{X \cdot (X' + Y) = X \cdot Y}$
  - $\mathbf{X + (X' \cdot Y) = X + Y}$

More Theorem Practice

Let $D = (X \square Y)$ and $P = ((X \square Y) \cdot Z)'$

$\rightarrow Q = (D \cdot P)'$ **(NAND)**

$= D' + P'$ **(De Morgan's)**

$= (X \square Y)' + (((X \square Y) \cdot Z)')'$ **(Substitution)**

$= (X \square Y)' + ((X \square Y) \cdot Z)$ **(NOT/NOT Cancellation)**

$= A' + (A \cdot Z)$ **(Let $A = (X \square Y)$)**

= …

= ??? (GOAL: USE "No Name" Theorem)

- "No Name" Theorem (Derived from Distribution)
  - **$X \cdot (X' + Y) = X \cdot Y$**
  - **$X + (X' \cdot Y) = X + Y$**

More Theorem Practice

Let $D = (X \square Y)$ and $P = ((X \square Y) \cdot Z)'$

$\rightarrow Q = (D \cdot P)'$ **(NAND)**

$= D' + P'$ **(De Morgan's)**

$= (X \square Y)' + (((X \square Y) \cdot Z)')'$ **(Substitution)**

$= (X \square Y)' + ((X \square Y) \cdot Z)$ **(NOT/NOT Cancellation)**

$= A' + (A \cdot Z)$ **(Let $A = (X \square Y)$)**

$= A' + Z$ **(No Name Theorem!!)**

= ...

- "No Name" Theorem (Derived from Distribution)
  - $\mathbf{X \cdot (X' + Y) = X \cdot Y}$
  - $\mathbf{X + (X' \cdot Y) = X + Y}$

More Theorem Practice

Let $D = (X \square Y)$ and $P = ((X \square Y) \bullet Z)'$

$\rightarrow Q = (D \bullet P)'$ **(NAND)**

$= D' + P'$ **(De Morgan's)**

$= (X \square Y)' + (((X \square Y) \bullet Z)')'$ **(Substitution)**

$= (X \square Y)' + ((X \square Y) \bullet Z)$ **(NOT/NOT Cancellation)**

$= A' + (A \bullet Z)$ **(Let $A = (X \square Y)$)**

$= A' + Z$ **(No Name Theorem!!)**

$= (X \square Y)' + Z$ **(Substitution)**

$= \ldots$

More Theorem Practice

Let $D = (X \square Y)$ and $P = ((X \square Y) \bullet Z)'$

$\rightarrow Q = (D \bullet P)'$ **(NAND)**

$= D' + P'$ **(De Morgan's)**

$= (X \square Y)' + (((X \square Y) \bullet Z)')'$ **(Substitution)**

$= (X \square Y)' + ((X \square Y) \bullet Z)$ **(NOT/NOT Cancellation)**

$= A' + (A \bullet Z)$ **(Let $A = (X \square Y)$)**

$= A' + Z$ **(No Name Theorem!!)**

$= (X \square Y)' + Z$ **(Substitution)**

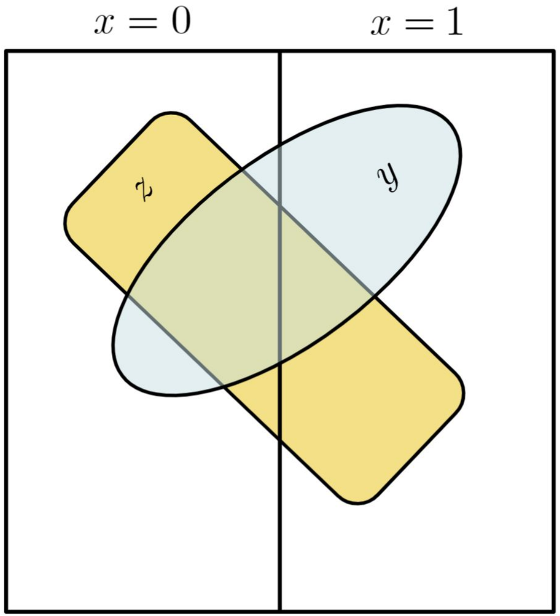
$\mathbf{= (X \odot Y) + Z}$ **(DONE)**

Consensus Understanding

10

Consensus Venn Diagram

$$Consensus(x' \cdot y + x \cdot z) = y \cdot z$$



Going Through Proving Consensus

$$
\begin{aligned}
XY + X'Z + YZ &= ? \\
&= ? \\
&= ? \\
&= ? \\
&= ? \\
&= ? \\
&= \mathbf{XY + X'Z}
\end{aligned}
$$

Going Through Proving Consensus

$XY + X'Z + YZ = XY + X'Z + (YZ \bullet (1))$ (Identities)

$= ?$

$= ?$

$= ?$

$= ?$

$= ?$

$= \mathbf{XY + X'Z}$

Going Through Proving Consensus

$$
\begin{aligned}
XY + X'Z + YZ &= XY + X'Z + (YZ \bullet (1)) && \text{(Identities)} \\
&= XY + X'Z + (YZ \bullet (X + X')) && \text{(Complement)} \\
&= ? \\
&= ? \\
&= ? \\
&= ? \\
&= \mathbf{XY + X'Z}
\end{aligned}
$$

Going Through Proving Consensus

$$
\begin{aligned}
XY + X'Z + YZ &= XY + X'Z + (YZ \bullet (1)) && \text{(Identities)}\\
&= XY + X'Z + (YZ \bullet (X + X')) && \text{(Complement)}\\
&= XY + X'Z + (XYZ + X'YZ) && \text{(Distribution)}\\
&= ?\\
&= ?\\
&= ?\\
&= \mathbf{XY + X'Z}
\end{aligned}
$$

Going Through Proving Consensus

$$
\begin{aligned}
XY + X'Z + YZ &= XY + X'Z + (YZ \bullet (1)) && \text{(Identities)}\\
&= XY + X'Z + (YZ \bullet (X + X')) && \text{(Complement)}\\
&= XY + X'Z + (XYZ + X'YZ) && \text{(Distribution)}\\
&= (XY + XYZ) + (X'Z + X'YZ) && \text{(Commutativity)}\\
&= ?\\
&= ?\\
&= \mathbf{XY + X'Z}
\end{aligned}
$$

Going Through Proving Consensus

$XY + X'Z + YZ$	$= XY + X'Z + (YZ \bullet (1))$	(Identities)
	$= XY + X'Z + (YZ \bullet (X + X'))$	(Complement)
	$= XY + X'Z + (XYZ + X'YZ)$	(Distribution)
	$= (XY + XYZ) + (X'Z + X'YZ)$	(Commutativity)
	$= (XY \bullet (1 + Z)) + (X'Z \bullet (1 + Y))$	(Distribution)
	$= ?$	
	$= \mathbf{XY + X'Z}$	

Going Through Proving Consensus

$$
\begin{aligned}
XY + X'Z + YZ &= XY + X'Z + (YZ \bullet (1)) && \text{(Identities)}\\
&= XY + X'Z + (YZ \bullet (X + X')) && \text{(Complement)}\\
&= XY + X'Z + (XYZ + X'YZ) && \text{(Distribution)}\\
&= (XY + XYZ) + (X'Z + X'YZ) && \text{(Commutativity)}\\
&= (XY \bullet (1 + Z)) + (X'Z \bullet (1 + Y)) && \text{(Distribution)}\\
&= (XY \bullet (1)) + (X'Z \bullet (1)) && \text{(Null Elements)}\\
&= \mathbf{XY + X'Z}
\end{aligned}
$$

Going Through Proving Consensus

$$
\begin{aligned}
XY + X'Z + YZ &= XY + X'Z + (YZ \bullet (1)) && \text{(Identities)} \\
&= XY + X'Z + (YZ \bullet (X + X')) && \text{(Complement)} \\
&= XY + X'Z + (XYZ + X'YZ) && \text{(Distribution)} \\
&= (XY + XYZ) + (X'Z + X'YZ) && \text{(Commutativity)} \\
&= (XY \bullet (1 + Z)) + (X'Z \bullet (1 + Y)) && \text{(Distribution)} \\
&= (XY \bullet (1)) + (X'Z \bullet (1)) && \text{(Null Elements)} \\
&= \mathbf{XY + X'Z} && \text{(Identities)}
\end{aligned}
$$

1) $a + 0 =$ ____________________

2) $\bar{a} \cdot 0 =$ ____________________

3) $a + \bar{a} =$ ____________________

4) $a + a =$ ____________________

5) $a + ab =$ ____________________

6) $a + \bar{a}b =$ ____________________

7) $a(\bar{a} + b) =$ ____________________

8) $ab + \bar{a}b =$ ____________________

9) $(\bar{a} + \bar{b})(\bar{a} + b) =$ ____________________

10) $a(a + b + c + \ldots) =$ ____________________

For (11),(12), (13), $f(a,b,c) = a + b + c$

11) $f(a,b,ab) =$ ____________________

12) $f(a,b,\bar{a} \cdot \bar{b}) =$ ____________________

13) $f[a,b,\overline{(ab)}] =$ ____________________

14) $y + y\bar{y} =$ ____________________

15) $xy + x\bar{y} =$ ____________________

16) $\bar{x} + y\bar{x} =$ ____________________

17) $(w + \bar{x} + y + \bar{z})y =$ ____________________

18) $(x + \bar{y})(x + y) =$ ____________________

19) $w + [w + (wx)] =$ ____________________

20) $x[x + (xy)] =$ ____________________

21) $\overline{(\bar{x} + \bar{x})} =$ ____________________

22) $\overline{(x + \bar{x})} =$ ____________________

23) $w + (w\bar{x}yz) =$ ____________________

24) $\bar{w} \cdot \overline{(wxyz)} =$ ____________________

25) $xz + \bar{x}y + zy =$ ____________________

26) $(x + z)(\bar{x} + y)(z + y) =$ ____________________

27) $\bar{x} + \bar{y} + xy\bar{z} =$ ____________________

SRC

For Additional (HARD) Online Practice

https://www.boolean-algebra.com/quiz

Randomly generates boolean equations and requires you to simplify

Simplify the equation

$$(BBB+BB)A+\overline{BD}$$

$D+\overline{B}$	BA	$A+\overline{B}+\overline{D}$

Simplify the equation

$$\overline{ADC}+A(BB+CC)$$

$CD+CB+\overline{A}$	1	$CA+DBC$

Simplify the equation

$$BD+(\overline{AB1}+AA)D$$

1	D	$D+\overline{A}$

Simplify the equation

$$\overline{DD}+D(DA+C0)A$$

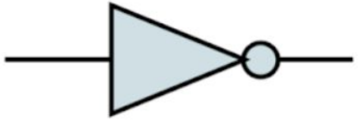
$\overline{D}+\overline{A}$	$\overline{D}+A$	$\overline{A}+D$

NAND x, y

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

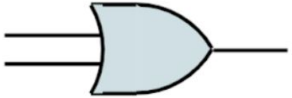
Proving NAND is Functionally Complete

NOT x



x'

OR x, y



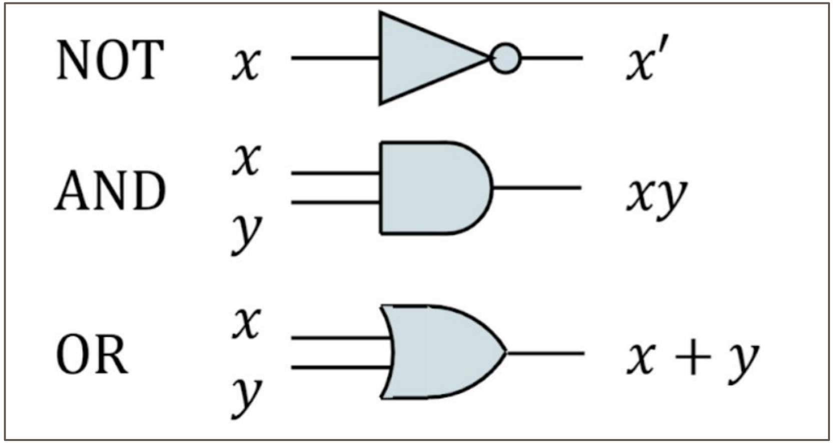
$x + y$

AND x, y

xy

Functional Completeness

- A set of operations that is able to express every switching operation
  - In this case, the set needs to be able to produce NOT, AND, OR operations
- To prove a set is functionally complete
  - Must build an **AND**, **NOT**, and **OR** operation from the set

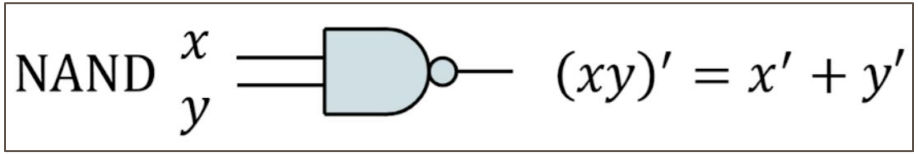


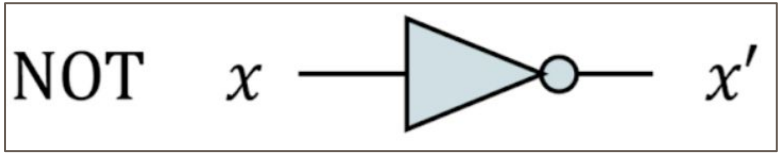
NAND is Functional Complete (Creating the NOT Gate)

HINT: It only requires one NAND Gate

X	Y	Out
0	0	1
0	1	1
1	0	1
1	1	0

X	Out
0	1
1	0



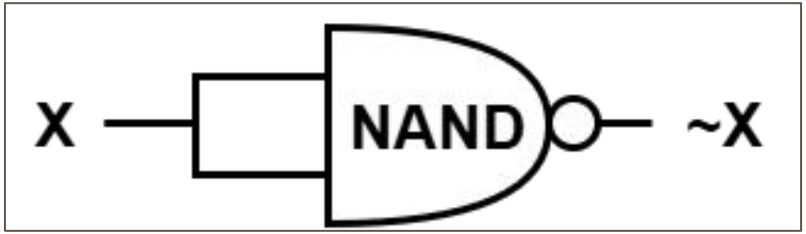


NAND is Functional Complete (Creating the NOT Gate)

HINT: It only requires one NAND Gate

X	Out = ~(X & X)
0	1
1	0

X	Out
0	1
1	0



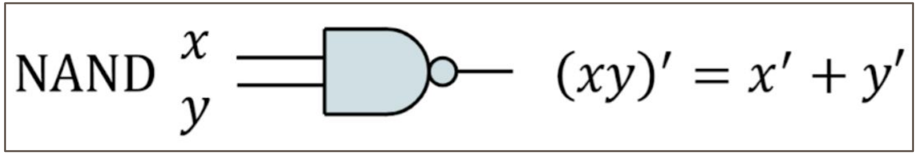
NAND is Functional Complete (Creating the AND Gate)

HINT: How Does a NAND relate to an AND

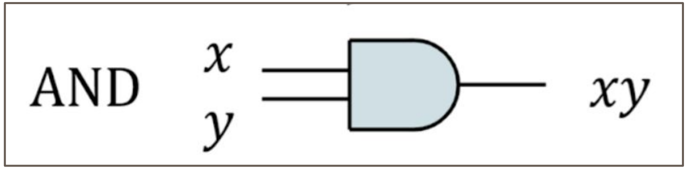
X	Y	Out
0	0	1
0	1	1
1	0	1
1	1	0

→

X	Y	Out
0	0	0
0	1	0
1	0	0
1	1	1

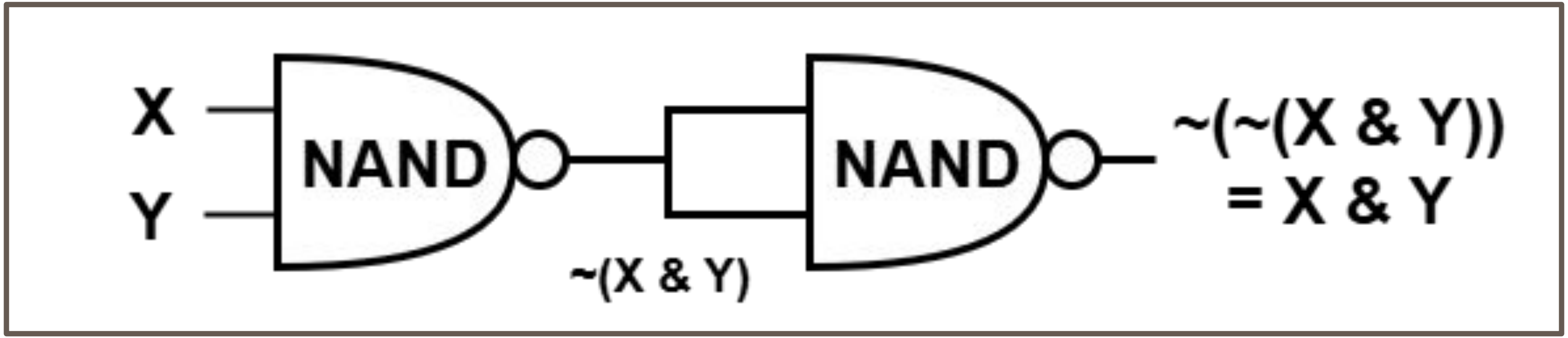


→



NAND is Functional Complete (Creating the AND Gate)

$$(X \cdot Y) = [(X \cdot Y)']'$$

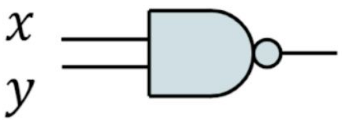


NAND is Functional Complete (Creating the OR Gate)

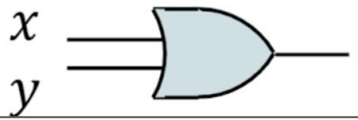
HINT: Think about De Morgan's

X	Y	Out
0	0	1
0	1	1
1	0	1
1	1	0

X	Y	Out
0	0	0
0	1	1
1	0	1
1	1	1

NAND 

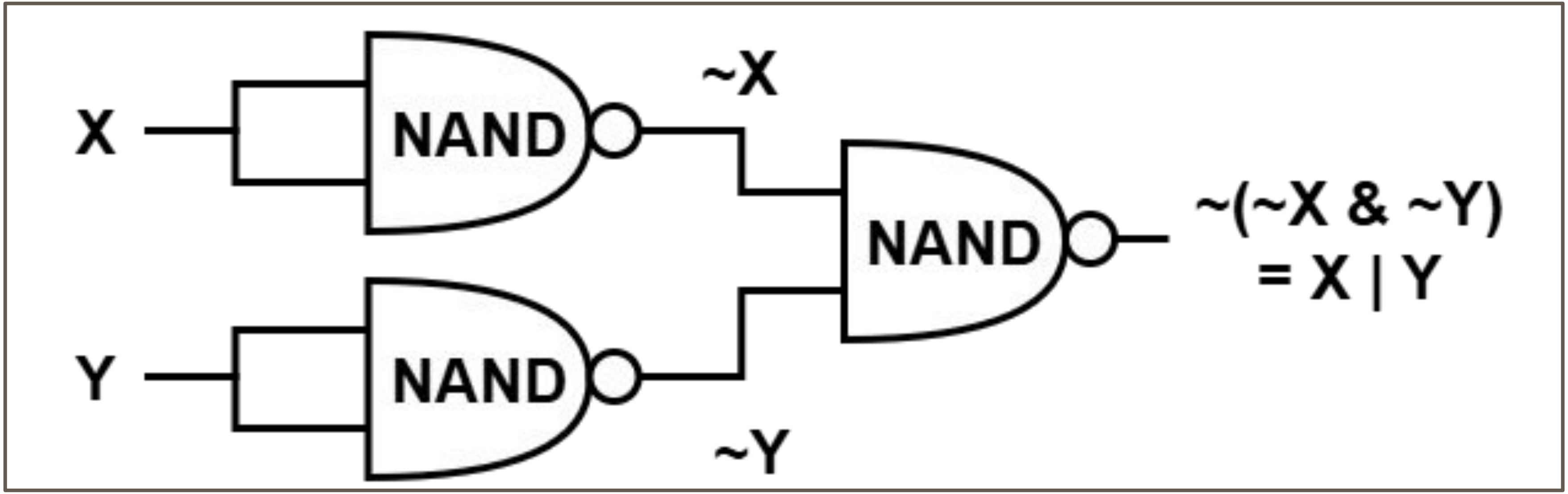
$(xy)' = x' + y'$

OR 

$x + y$

NAND is Functional Complete (Creating the OR Gate)

$$(X + Y) = (X' \cdot Y')'$$



Canonical Sum of Products (SOP)

- Minterm - AND term of N literals
  - Ex. $m_5(A, B, C, D) = A'BC'D$
  - **m_i = 1 for exactly one combination of variables, and 0 for all others**

- Canonical SOP - Sum of Minterms (all minterms that result in f=1)

- Ex. $f(x, y, z) = \boldsymbol{\Sigma_{x,\,y,\,z}\,(0, 2, 3, 6, 7)}$

$$= \mathbf{(x'y'z') + (x'yz') + (x'yz) + (xyz') + (xyz)}$$

Decimal	*x y z*	*f*
0	000	1
1	001	0
2	010	1
3	011	1
4	100	0
5	101	0
6	110	1
7	111	1

Canonical Product of Sums (POS)

- Maxterm - OR term of N literals
  - Ex. $M_5(A, B, C, D) = (A + B' + C + D')$
  - **M_i = 0 for exactly one combination of variables, and 1 for all others**
  - $M_i = m_i'$ [Ex. $M_5 = m_5' = (A'BC'D)' = (A + B' + C + D')$]

- Canonical POS - Product of Maxterms (all maxterms that result in f=0)

- Ex. $f(x, y, z) = \prod_{x, y, z} (1, 4, 5)$

$$= (x + y + z') \bullet (x' + y + z) \bullet (x' + y + z')$$

Decimal	$x\,y\,z$	f
0	000	1
1	001	0
2	010	1
3	011	1
4	100	0
5	101	0
6	110	1
7	111	1

Conversion of Bases

Base Conversion (Common Bases/Radices)

Common Name	**Base/Radix**	**Single Digit Range**
Binary	2	{0, 1}
Decimal	10	{0, 1, 2, …, 8, 9}
Hexadecimal	16	{0, 1, …, 9, A, B, …, F}
Octal	8	{0, 1, …, 7}

Converting from Binary → Decimal

Unsigned N-bit Binary value = $x_{(N-1)}x_{(N-2)}..x_1x_0$

→ Decimal value = $(2^{(N-1)} * x_{(N-1)}) + (2^{(N-2)} * x_{(N-2)}) + \ldots + (2^{(1)} * x_{(1)}) + (2^{(0)} * x_{(0)})$

Ex. $10101_2 = [2^4 * (1)] + [2^3 * (0)] + [2^2 * (1)] + [2^1 * (0)] + [2^0 * (1)]$

$= (2^4) + (2^2) + (2^0)$

$= (16) + (4) + (1)$

$= \mathbf{21_{10}}$

Converting from ANY Base (B) → Decimal

Unsigned N-digit value in base B = $x_{(N-1)}x_{(N-2)}..x_1x_0$

→ Decimal value = $(B^{(N-1)} * x_{(N-1)}) + (B^{(N-2)} * x_{(N-2)}) + \ldots + (B^{(1)} * x_{(1)}) + (B^{(0)} * x_{(0)})$

Ex. $2E3D_{16} = [16^3 * (2)] + [16^2 * (E == 14)] + [16^1 * (3)] + [16^0 * (D == 13)]$

$= (4096 * 2) + (256 * 14) + (16 * 3) + (1 * 13)$

$= (8192) + (3584) + (48) + (13)$

$= \mathbf{11{,}837_{10}}$

Converting from Decimal → Binary

- Will require special long division

Ex. $85_{10} = ?_2$

$2\overline{)85}$
$2\overline{)42}$ r1
$2\overline{)21}$ r0
$2\overline{)10}$ r1
$2\overline{)5}$ r0
$2\overline{)2}$ r1
$2\overline{)1}$ r0
0 r1

$\boxed{\therefore 85_{10} = 1010101_2}$

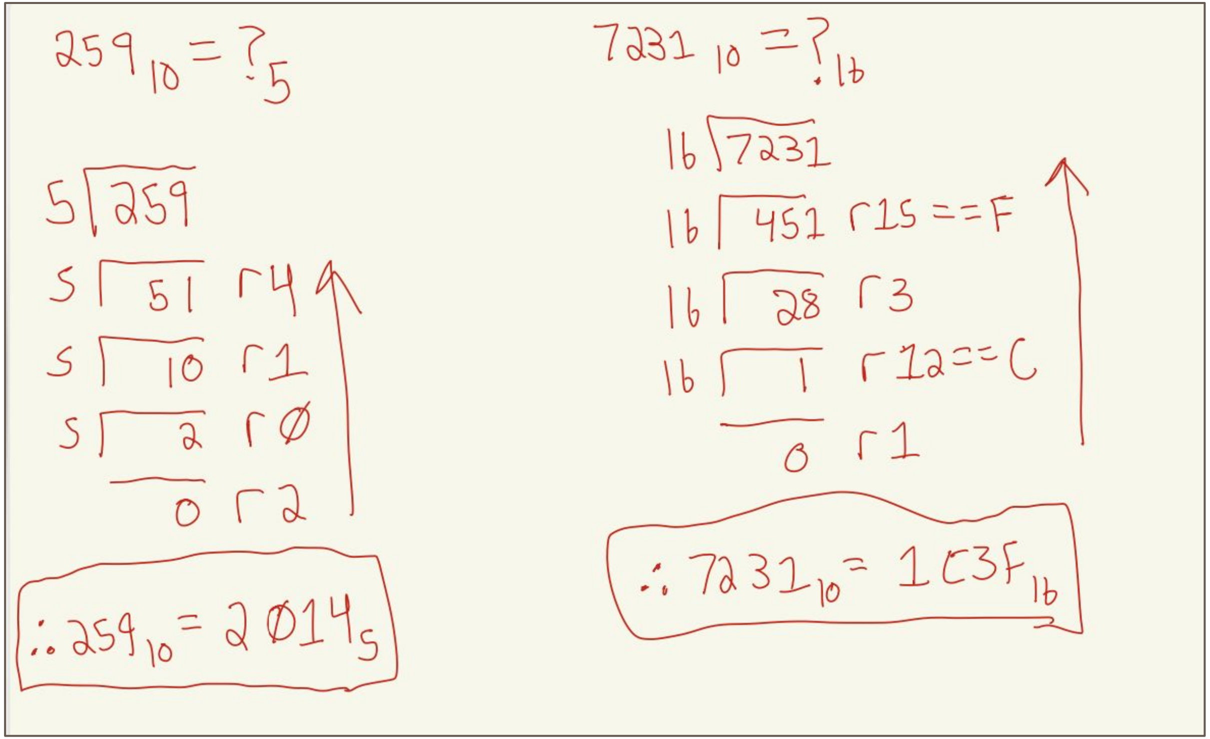
Ex. $44_{10} = ?_2$

$2\overline{)44}$
$2\overline{)22}$ r0
$2\overline{)11}$ r0
$2\overline{)5}$ r1
$2\overline{)2}$ r1
$2\overline{)1}$ r0
$2\overline{)0}$ r1

$\therefore 44_{10} = 101100_2$

Converting from Decimal → ANY Base (B)

- Same steps as binary, but with a **different divisor**



Steps to Convert from ANY Base (B_1) → ANY Base (B_2)

1. Convert B_1 → Decimal
2. Convert from Decimal → B_2

$4546_7 = ?_{13}$

1. Base 7 → 10

$$
\begin{aligned}
&(6 \cdot 7^0) \\
+&(4 \cdot 7^1) \\
+&(5 \cdot 7^2) \\
+&(4 \cdot 7^3) \\
\hline
&6 \\
+&28 \\
+&245 \\
+&1372 \\
&\boxed{1651_{10}}
\end{aligned}
$$

2. Base 10 → 13

$$
\begin{array}{l l}
13 \,\overline{)\,1651} & \\
13 \,\overline{)\,127} & r\,0 \\
13 \,\overline{)\,9} & r\,10 == A \\
\quad 0 & r\,9
\end{array}
$$

$$\boxed{\therefore 4546_7 = 9A0_{13}}$$

Trick with Octal ↔ Binary ↔ Hexadecimal Conversions

- Very easy to convert between these bases

$$1011010001_2 = ?_{16}$$

$$\rightarrow \underbrace{0010}_{2}\ \underbrace{1101}_{D}\ \underbrace{0001}_{1}$$

$$\boxed{\therefore 1011010001 = 2D1_{16}}$$

$$= ?_8$$

$$\rightarrow \underbrace{001}_{1}\ \underbrace{011}_{3}\ \underbrace{010}_{2}\ \underbrace{001}_{1}{}_2$$

$$\boxed{\therefore 1011010001_2 = 1321_8}$$

Trick with Octal ↔ Binary ↔ Hexadecimal Conversions

$5D3F_{16} = ?_8$

1. Convert to Binary

$5D3F_{16} = \boxed{0101\ 1101\ 0011\ 1111_2}$

2. Binary → Octal

$$\underbrace{101}_{5}\ \underbrace{110}_{6}\ \underbrace{100}_{4}\ \underbrace{111}_{7}\ \underbrace{111}_{7}{}_2$$

$$\boxed{\therefore 5D3F_{16} = 56477_8}$$

Practice Problems!

1. Convert $123D_{16}$ to base 10
2. Convert $1000\_1001\_1010_2$ to base 10
3. Convert $B23D_{16}$ to binary
4. Convert BDF_{16} to octal
5. Convert 381_{10} to base 5
6. Convert 3210_5 to base 3

Practice Problems!

1. Convert $123D_{16}$ to base 10
    a. 4669_{10}
2. Convert $1000\_1001\_1010_2$ to base 10
    a. 2202_{10}
3. Convert $B23D_{16}$ to binary
    a. $1011\_0010\_0011\_1101_2$
4. Convert BDF_{16} to octal
    a. 5737_8
5. Convert 381_{10} to base 5
    a. 3011_5
6. Convert 3210_5 to base 3
    a. 1202213_3

Number Representation

Signed Magnitude

Unique Characteristics:
???

4-bit Range: **(?, ?)**

Most positive 4-bit number binary representation:
????

Most negative 4-bit number binary representation:
????

Multiple Representations? : **Y / N**

If so, what?:

1's Complement

Unique Characteristics:
???

4-bit Range: **(?, ?)**

Most positive 4-bit number binary representation:
????

Most negative 4-bit number binary representation:
????

Multiple Representations? : **Y / N**

If so, what?:

2's Complement

Unique Characteristics:
???

4-bit Range: **(?, ?)**

Most positive 4-bit number binary representation:
????

Most negative 4-bit number binary representation:
????

Multiple Representations? : **Y / N**

If so, what?:

Signed Magnitude

Unique Characteristics:
MSB is an unweighted sign bit
0 → Positive, 1 → Negative

4-bit Range: **(-7, +7)**

Most positive 4-bit number binary representation:
$\mathbf{0111_{2C} \rightarrow +7_{10}}$

Most negative 4-bit number binary representation:
$\mathbf{1111_{2C} \rightarrow -7_{10}}$

Multiple Representations? : **Y**

If so, what?:
$\mathbf{0000_{SM} \rightarrow 0_{10}}$
$\mathbf{1000_{SM} \rightarrow -0_{10} = 0_{10}}$

1's Complement

Unique Characteristics:
???

4-bit Range: **(?, ?)**

Most positive 4-bit number binary representation:
????

Most negative 4-bit number binary representation:
????

Multiple Representations? : **Y / N**

If so, what?:

2's Complement

Unique Characteristics:
???

4-bit Range: **(?, ?)**

Most positive 4-bit number binary representation:
????

Most negative 4-bit number binary representation:
????

Multiple Representations? : **Y / N**

If so, what?:

Signed Magnitude

Unique Characteristics:
MSB is an unweighted sign bit
0 → Positive, 1 → Negative

4-bit Range: **(-7, +7)**

Most positive 4-bit number binary representation:
$0111_{2C} \rightarrow +7_{10}$

Most negative 4-bit number binary representation:
$1111_{2C} \rightarrow -7_{10}$

Multiple Representations? : **Y**

If so, what?:
$0000_{SM} \rightarrow 0_{10}$
$1000_{SM} \rightarrow -0_{10} = 0_{10}$

1's Complement

Unique Characteristics:
MSB is $-(2^{(n-1)} - 1)$
Neg. rep. → Just flip bits!

4-bit Range: **(-7, +7)**

Most positive 4-bit number binary representation:
$0111_{1C} \rightarrow +7_{10}$

Most negative 4-bit number binary representation:
$1000_{1C} \rightarrow -7_{10}$

Multiple Representations? : **Y**

If so, what?:
$0000_{SM} \rightarrow 0_{10}$
$1111_{SM} \rightarrow 0_{10}$

2's Complement

Unique Characteristics:
???

4-bit Range: **(?, ?)**

Most positive 4-bit number binary representation:
????

Most negative 4-bit number binary representation:
????

Multiple Representations? : **Y / N**

If so, what?:

Signed Magnitude

Unique Characteristics:
MSB is an unweighted sign bit
0 → Positive, 1 → Negative

4-bit Range: **(-7, +7)**

Most positive 4-bit number binary representation:
$\mathbf{0111_{2C} \rightarrow +7_{10}}$

Most negative 4-bit number binary representation:
$\mathbf{1111_{2C} \rightarrow -7_{10}}$

Multiple Representations? : **Y**

If so, what?:
$\mathbf{0000_{SM} \rightarrow 0_{10}}$
$\mathbf{1000_{SM} \rightarrow -0_{10} = 0_{10}}$

1's Complement

Unique Characteristics:
MSB is $-(2^{(n-1)} - 1)$
Neg. rep. → Just flip bits!

4-bit Range: **(-7, +7)**

Most positive 4-bit number binary representation:
$\mathbf{0111_{1C} \rightarrow +7_{10}}$

Most negative 4-bit number binary representation:
$\mathbf{1000_{1C} \rightarrow -7_{10}}$

Multiple Representations? : **Y**

If so, what?:
$\mathbf{0000_{SM} \rightarrow 0_{10}}$
$\mathbf{1111_{SM} \rightarrow 0_{10}}$

2's Complement

Unique Characteristics:
MSB is $-(2^{(n-1)})$
Neg. rep. → Flip bits, add 1
No repeat representation!
Easy hardware implementation!

4-bit Range: **(-8, +7)**

Most positive 4-bit number binary representation:
$\mathbf{0111_{2C} \rightarrow +7_{10}}$

Most negative 4-bit number binary representation:
$\mathbf{1000_{2C} \rightarrow -8_{10}}$

Multiple Representations? : **N**

If so, what?: **N/A**

Binary Arithmetic

Half Adders

- Let's **design** a 1-bit half-adder

$$\begin{array}{r} X \\ +\,Y \\ \hline C\ S \end{array} \qquad \begin{array}{r} 0 \\ +\,0 \\ \hline ?\ ? \end{array} \qquad \begin{array}{r} 0 \\ +\,1 \\ \hline ?\ ? \end{array} \qquad \begin{array}{r} 1 \\ +\,0 \\ \hline ?\ ? \end{array} \qquad \begin{array}{r} 1 \\ +\,1 \\ \hline ?\ ? \end{array}$$

Half Adders

- Let's **design** a 1-bit half-adder

$$\begin{array}{r} X \\ +\ Y \\ \hline C\ S \end{array} \qquad \begin{array}{r} 0 \\ +\ 0 \\ \hline 0\ 0 \end{array} \qquad \begin{array}{r} 0 \\ +\ 1 \\ \hline 0\ 1 \end{array} \qquad \begin{array}{r} 1 \\ +\ 0 \\ \hline 0\ 1 \end{array} \qquad \begin{array}{r} 1 \\ +\ 1 \\ \hline 1\ 0 \end{array}$$

Half Adders - FINISH THE REST!

X	Y	C	S
0	0	?	?
0	1	?	?
1	0	?	?
1	1	?	?

C = ?

S = ?

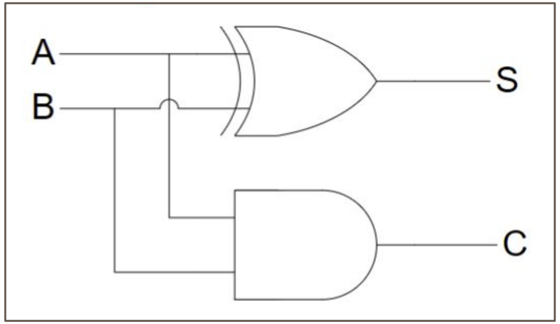
```
module half_adder(
  input x, y,
  output c, s);

  // finish module

endmodule
```

Half Adders

X	Y	C	S
0	0	**0**	**0**
0	1	**0**	**1**
1	0	**0**	**1**
1	1	**1**	**0**

$$C = X \bullet Y$$

$$S = (X' \bullet Y) + (X \bullet Y') = X \oplus Y$$




```
module half_adder(
  input x, y,
  output c, s);

  assign s = x ^ y;
  assign c = x & y;

endmodule
```

Full Adders - Introduction of Carry-In Bit

X	Y	Cin	C	S
0	0	0	?	?
0	0	1	?	?
0	1	0	?	?
0	1	1	?	?
1	0	0	?	?
1	0	1	?	?
1	1	0	?	?
1	1	1	?	?

C = ?

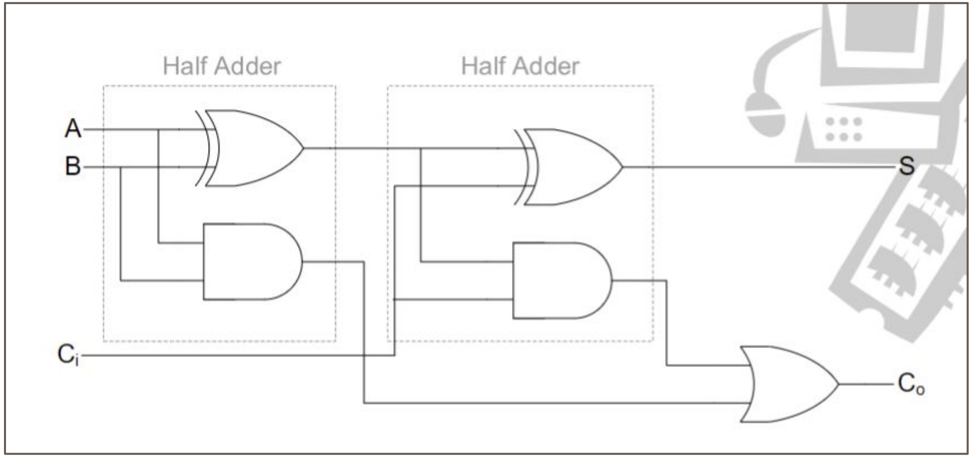
S = ?

Full Adders - Introduction of Carry-In Bit

X	Y	Cin	C	S
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**

$$\mathbf{C = (X \bullet Y) + X \bullet (Y \oplus Cin)}$$

$$\mathbf{S = X \oplus Y \oplus Cin}$$



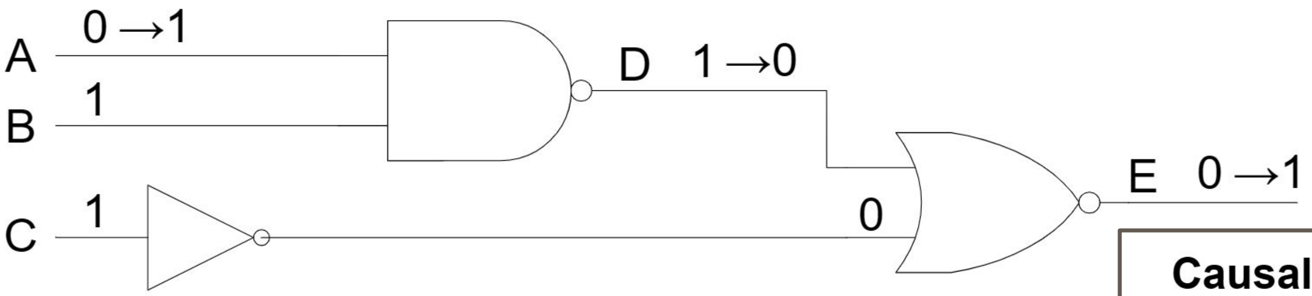
Propagation Delay

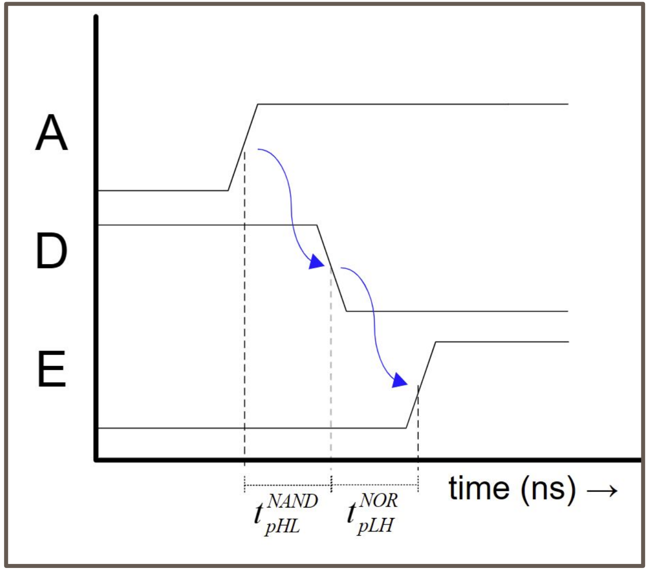
Output Changes are NOT Immediate

- Logical circuitry is built using physical wires and transistors
  - Each have some internal delay
  - In real implementations, need to account for delay

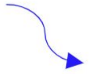
- **Propagation Delay** - Time it takes for the input change to be reflected onto the output

Causality Graphs and Terminology





Causality Arrow



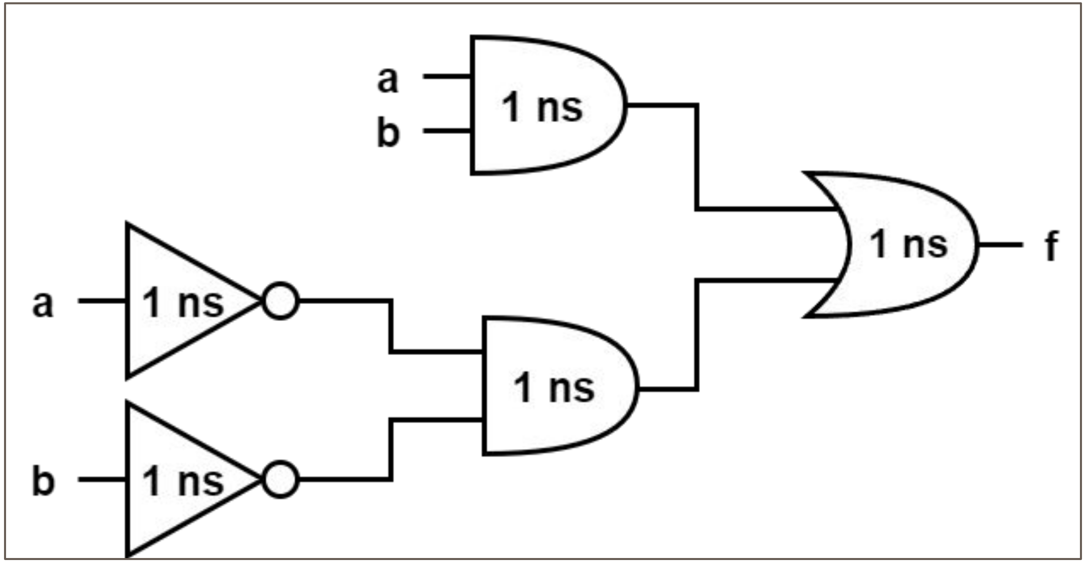
Gate Propagation Delay

$$t_{pHL}^{NAND}$$

gives NAND gate propagation delay from input to output when output is changing from H to L

Calculating Propagation Delay - Unit Delay

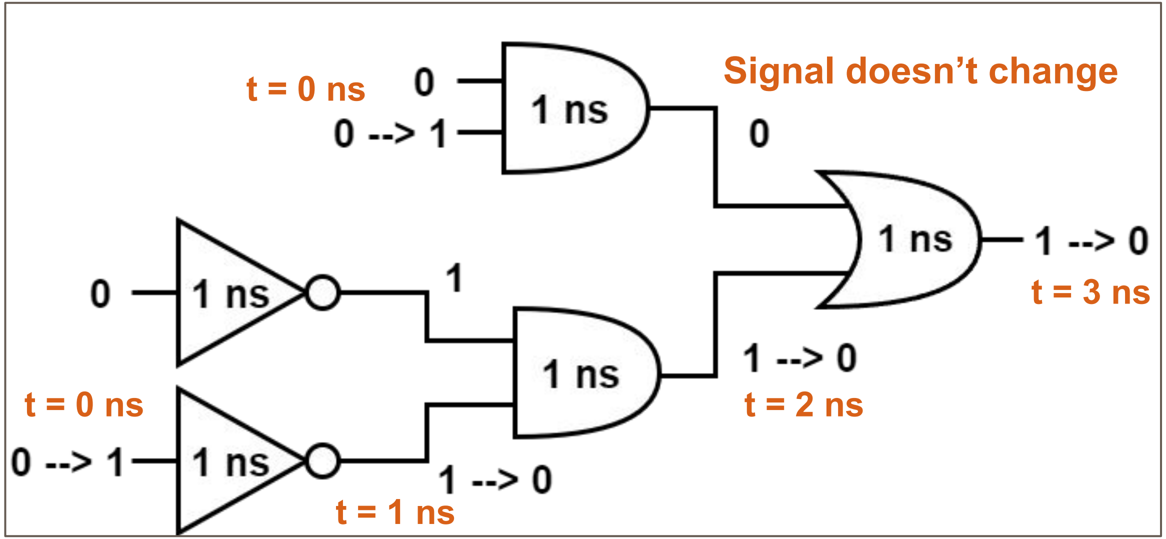
- **Unit Delay** - Takes **1 unit of delay** to propagate through the gate
  - We will be assuming 1 time unit = 1ns for these example problems



CHECK1 Circuit with Unit Delay Gates

Calculating Propagation Delay - Unit Delay

1. Assume gates are **SETTLED** at some initial combination of values
2. Update **one** of the controllable inputs (Ex. Updating ‘b’ from 0 → 1)
3. Let the signal propagate through the circuit until the output has **SETTLED**



CHECK1 Circuit with Unit Delay Gates

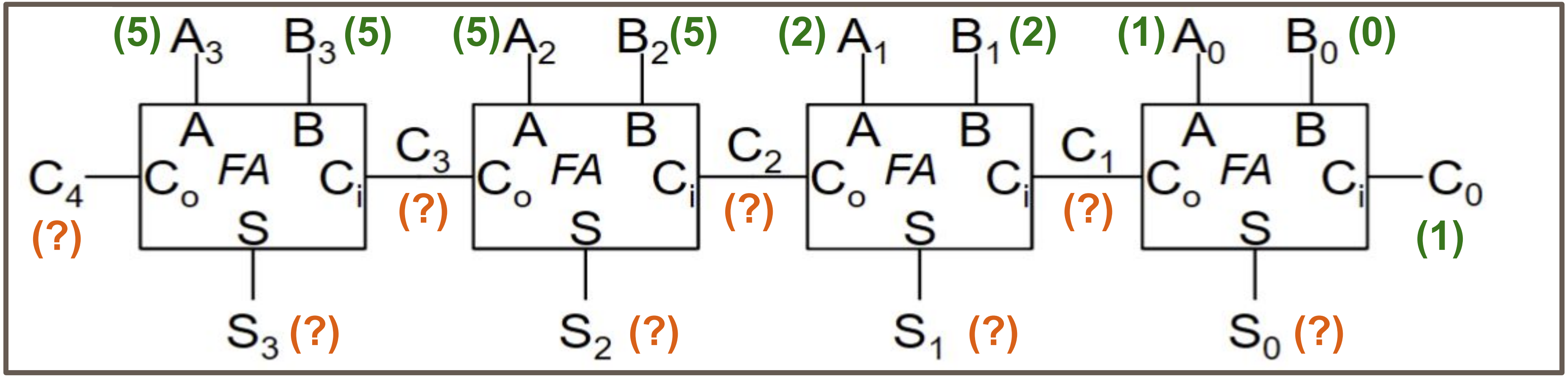
Starting with combination (a, b) = (0, 0)

→ $t^{b\to f}_{pHL}$ = 3 ns

Remember: pHL is the OUTPUT going from H→L

This is not true for all initial combinations of (a, b)

Ripple Carry Adder - CALCULATE SETTLE TIMES

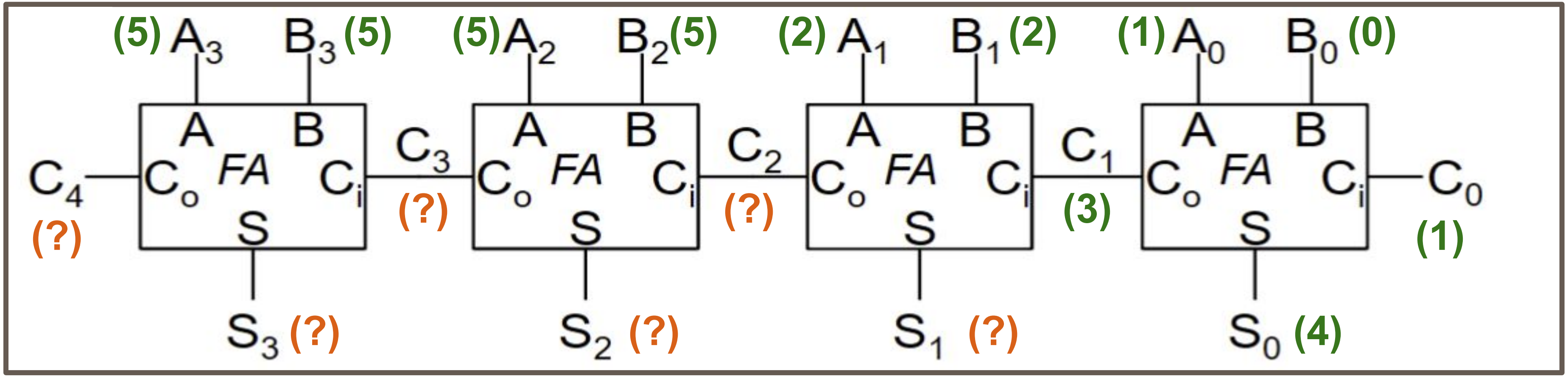


Input	Output	Delay
a or b	s	3
a or b	co	1
ci	s	2
ci	co	2

(x) - Known Settle Times

(?) - Times to Determine

Ripple Carry Adder - CALCULATE SETTLE TIMES

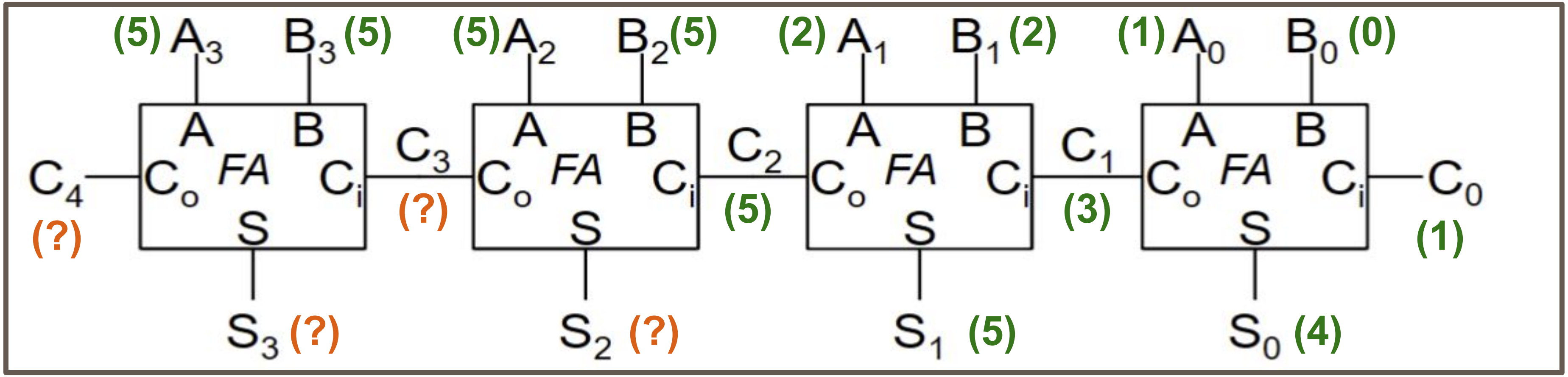


Input	Output	Delay
a or b	s	3
a or b	co	1
ci	s	2
ci	co	2

(x) - Known Settle Times

(?) - Times to Determine

Ripple Carry Adder - CALCULATE SETTLE TIMES

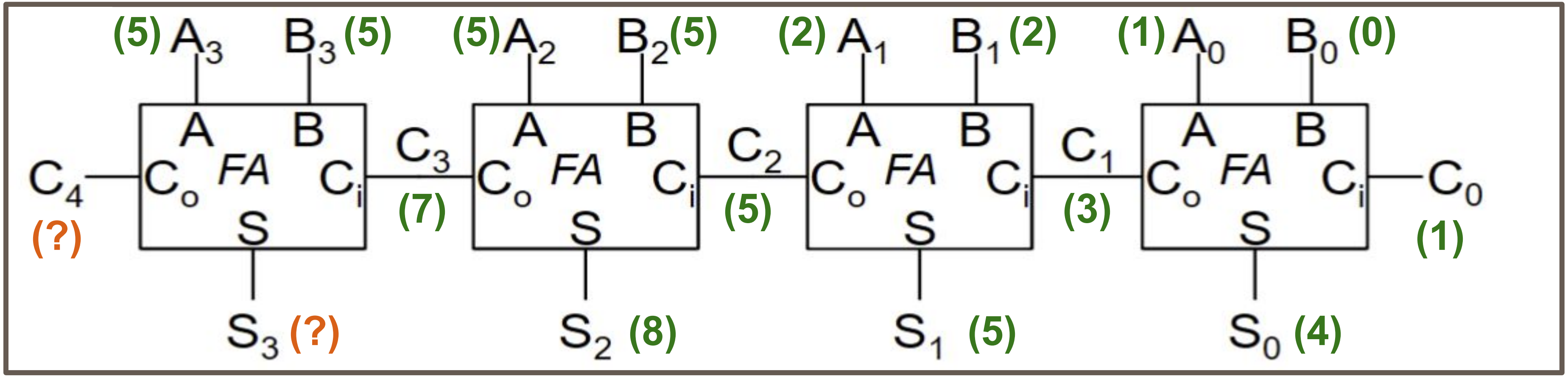


Input	Output	Delay
a or b	s	3
a or b	co	1
ci	s	2
ci	co	2

(x) - Known Settle Times

(?) - Times to Determine

Ripple Carry Adder - CALCULATE SETTLE TIMES

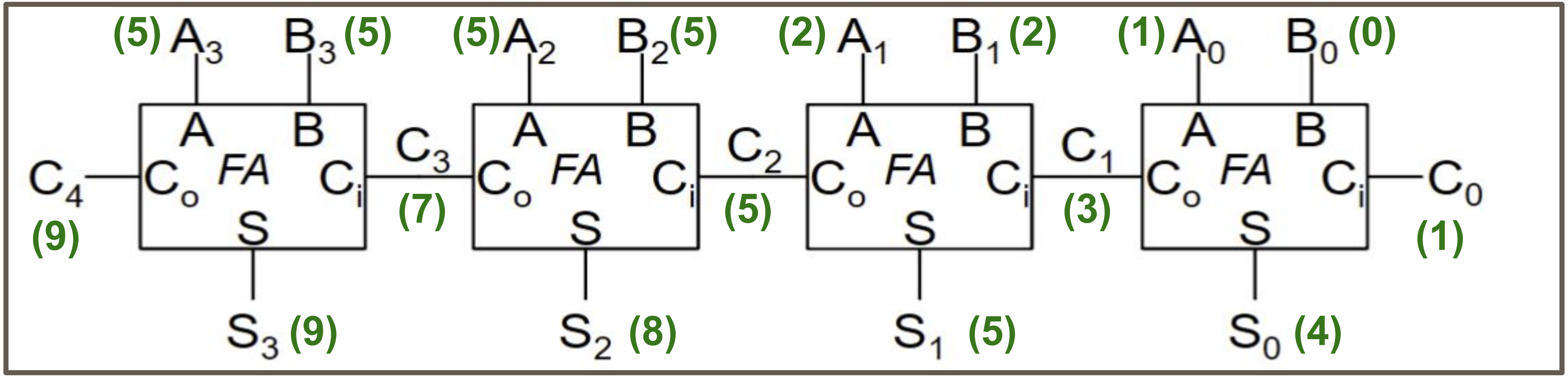


Input	Output	Delay
a or b	s	3
a or b	co	1
ci	s	2
ci	co	2

(x) - Known Settle Times

(?) - Times to Determine

Ripple Carry Adder - CALCULATE SETTLE TIMES

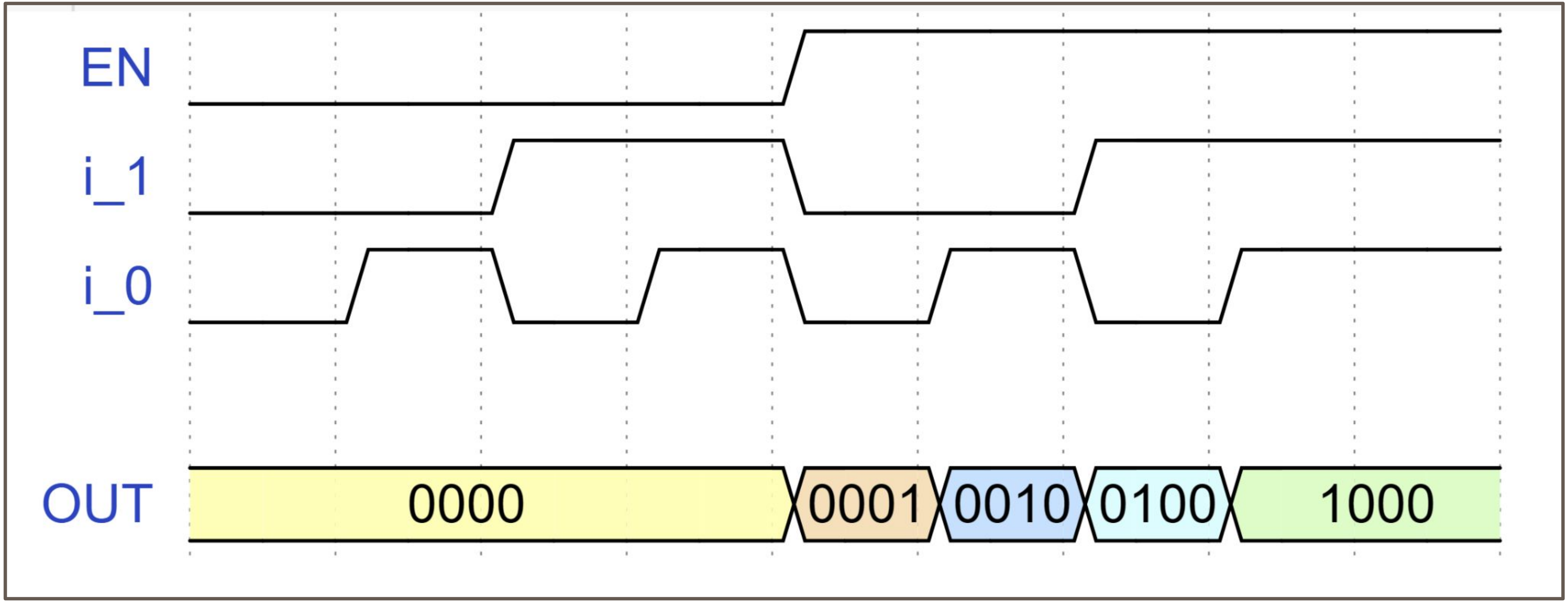


Input	Output	Delay
a or b	s	3
a or b	co	1
ci	s	2
ci	co	2

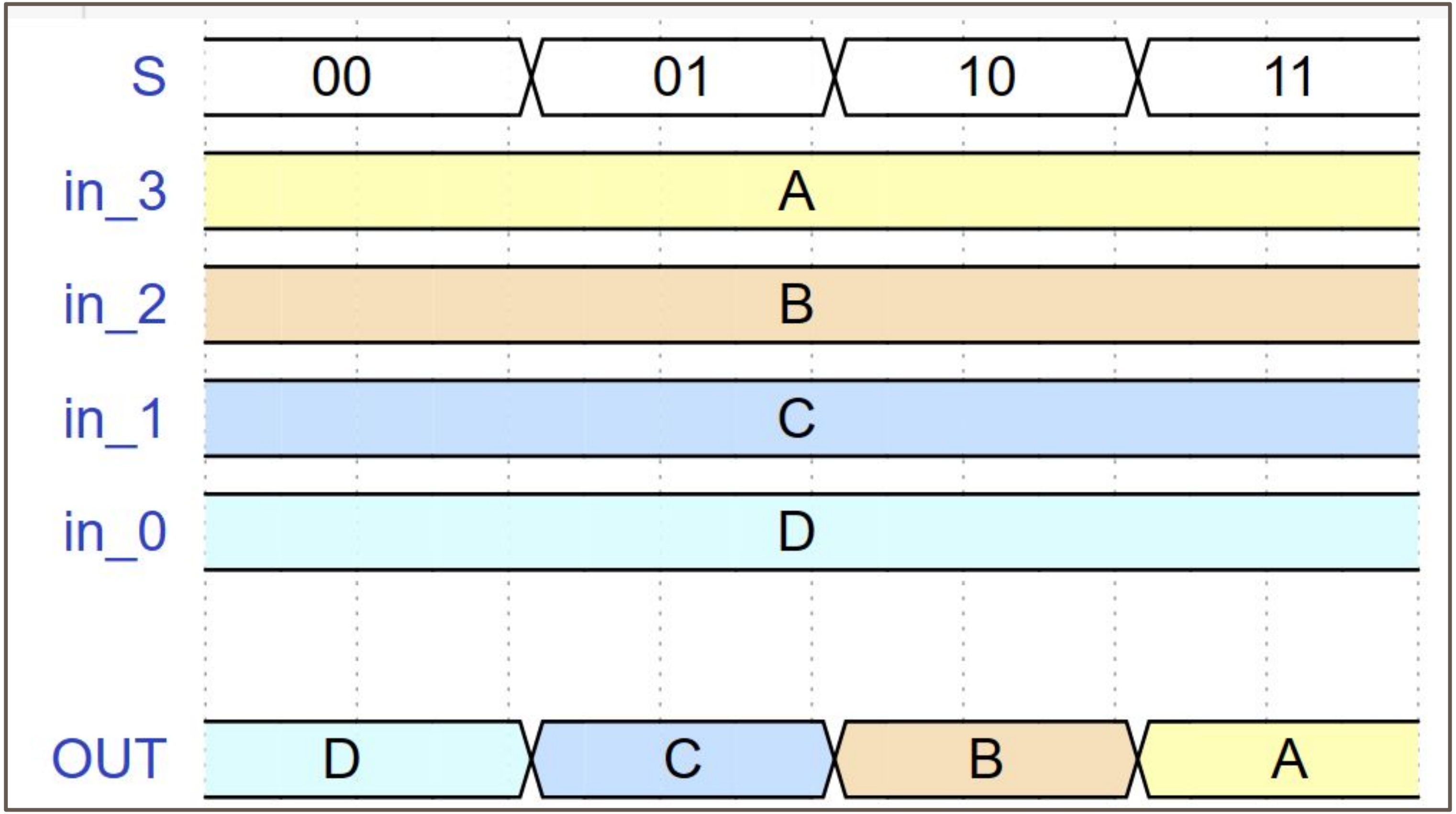
(x) - Known Settle Times
(?) - Times to Determine

Common Combinational Blocks

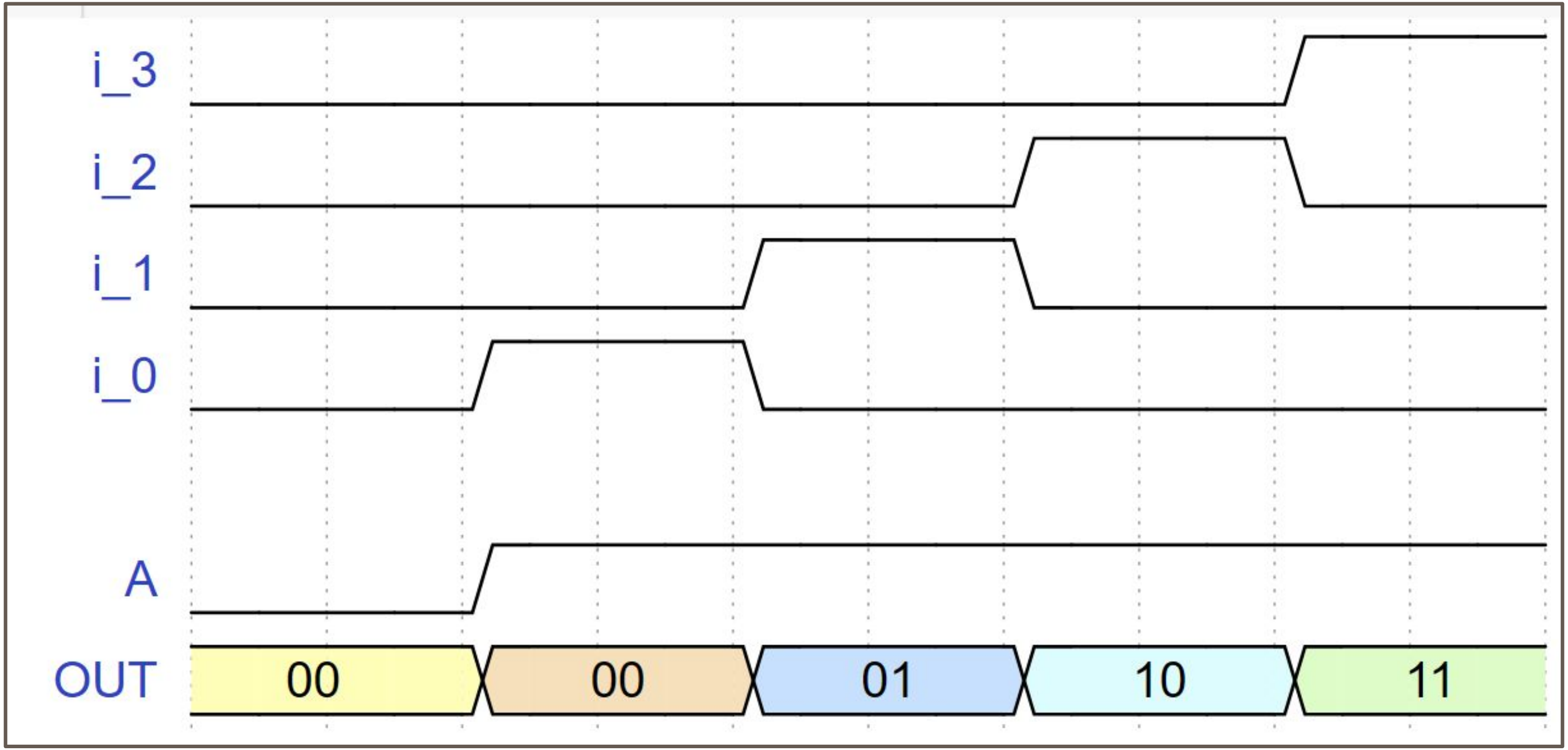
A



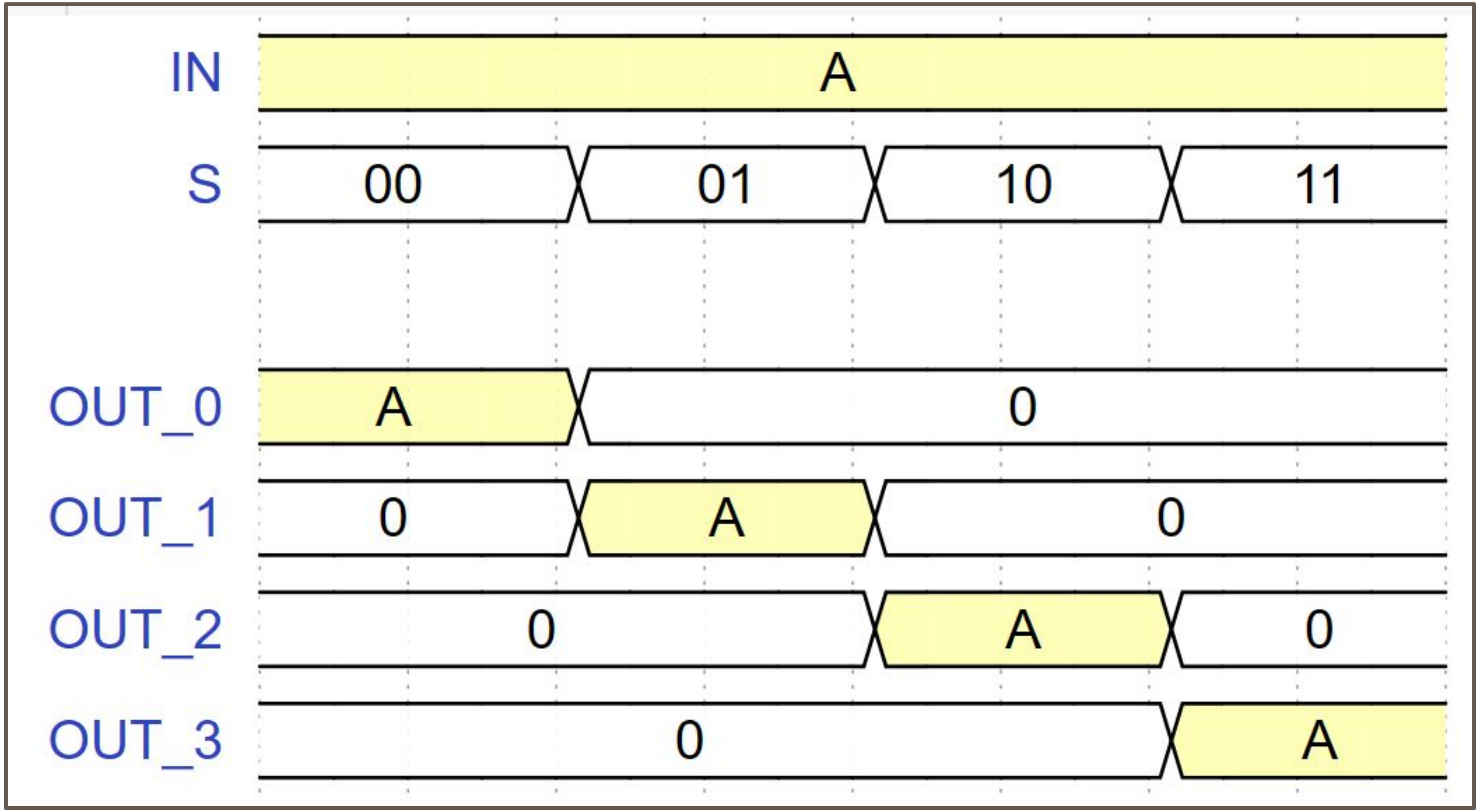
B



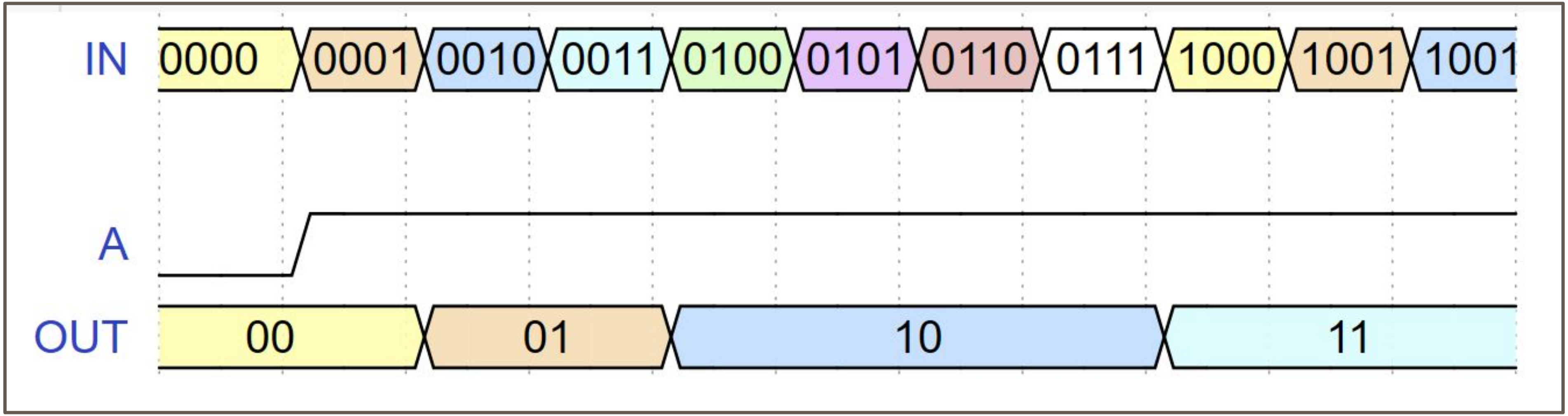
C



D



E (BONUS)

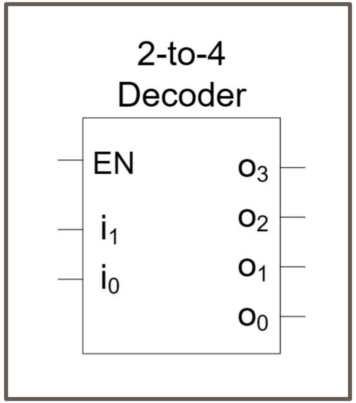

IN
0000
0001
0010
0011
0100
0101
0110
0111
1000
1001
1001
A
OUT
00
01
10
11

ANSWERS TO WAVEFORM QUESTIONS

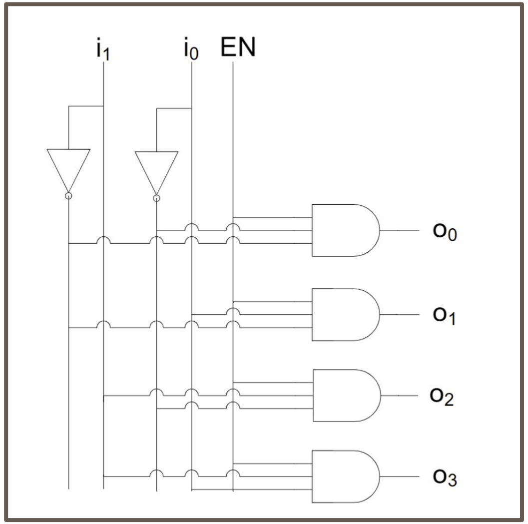
- A → DECODER
- B → MULTIPLEXER (MUX)
- C → ENCODER
- D → DEMULTIPLEXER (DEMUX)
- E → PRIORITY ENCODER

Common Combinational Blocks (Decoder)

- INPUT: Encoded bits (n bits)
- OUTPUT: One-Hot Representation of Bits (2^n bits)

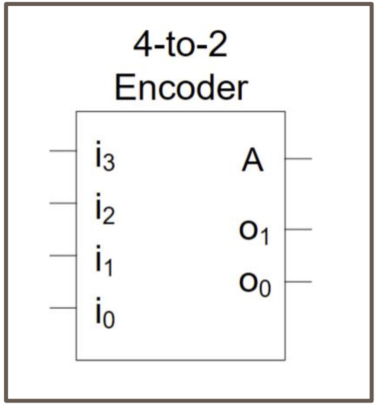


EN	i_1	i_0	$o_3o_2o_1o_0$
0	x	x	0000
1	0	0	0001
1	0	1	0010
1	1	0	0100
1	1	1	1000



Common Combinational Blocks (Encoder)

- INPUT: One-Hot Representation of Bits (2^n bits)
- OUTPUT: Encoded bits (n bits)



i_3	i_2	i_1	i_0	o_1o_0	A
0	0	0	0	00	0
1	0	0	0	11	1
0	1	0	0	10	1
0	0	1	0	01	1
0	0	0	1	00	1
Any Other Combo				dd	d

$$o_1 = i_3 + i_2$$

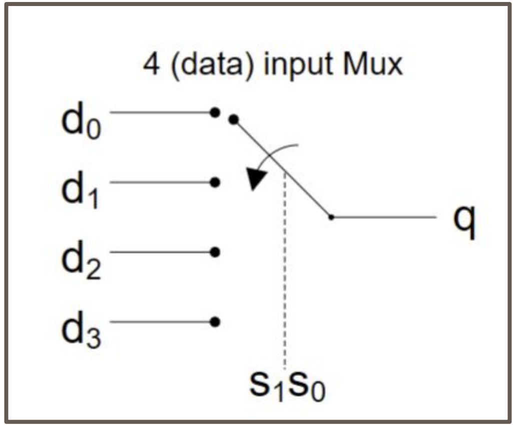
$$o_0 = i_3 + i_1$$

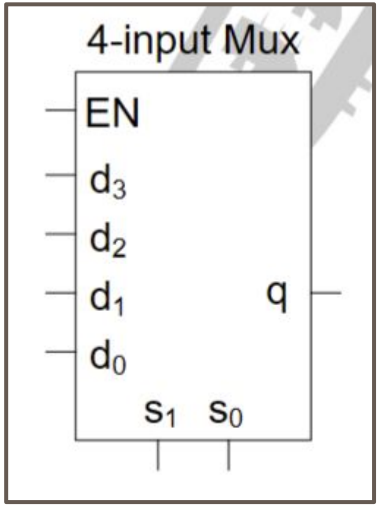
$$A = i_3 + i_2 + i_1 + i_0$$

Common Combinational Blocks (Multiplexer)

- Switch of which the output is driven from one of the n inputs

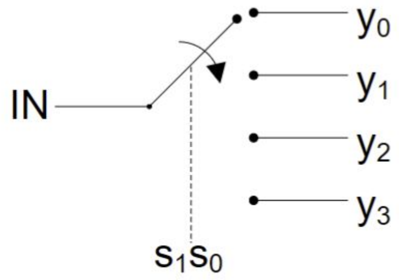
$$Q = s_1's_0'd_0 + s_1's_0d_1 + s_1s_0'd_2 + s_1s_0d_3$$



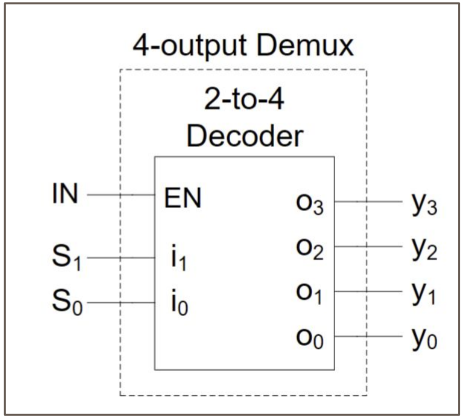


Common Combinational Blocks (Demultiplexer)

- Connects Input to one of the Output signals
- Dependent on the selector bits



s_1	s_0	y_0	y_1	y_2	y_3
0	0	IN	0	0	0
0	1	0	IN	0	0
1	0	0	0	IN	0
1	1	0	0	0	IN



Have a cute pet? See it featured on the next exam review!

https://forms.gle/ZS4YfU8RH4iJaa626