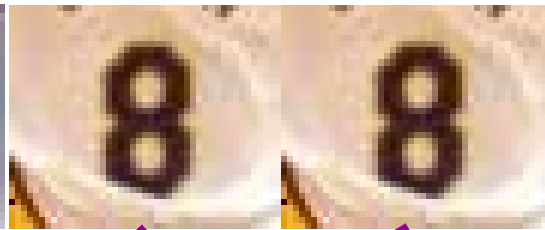
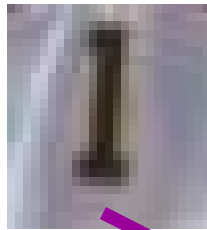


CNS



Scored



CNS



100 Wilt Chamberlain, Philadelphia, 3/2/1962
81 Kobe Bryant, LA Lakers, 1/22/2006
78 Wilt Chamberlain, Philadelphia, 12/8/1961
73 David Thompson, Denver, 4/9/1978
73 Wilt Chamberlain, San Francisco, 11/16/1962
73 Wilt Chamberlain, Philadelphia, 1/13/1962
72 Wilt Chamberlain, San Francisco, 11/3/1962
71 David Robinson, San Antonio, 4/24/1994
71 Elgin Baylor, LA Lakers, 11/15/1960
70 Wilt Chamberlain, San Francisco, 3/10/1963
69 Michael Jordan, Chicago, 3/28/1990

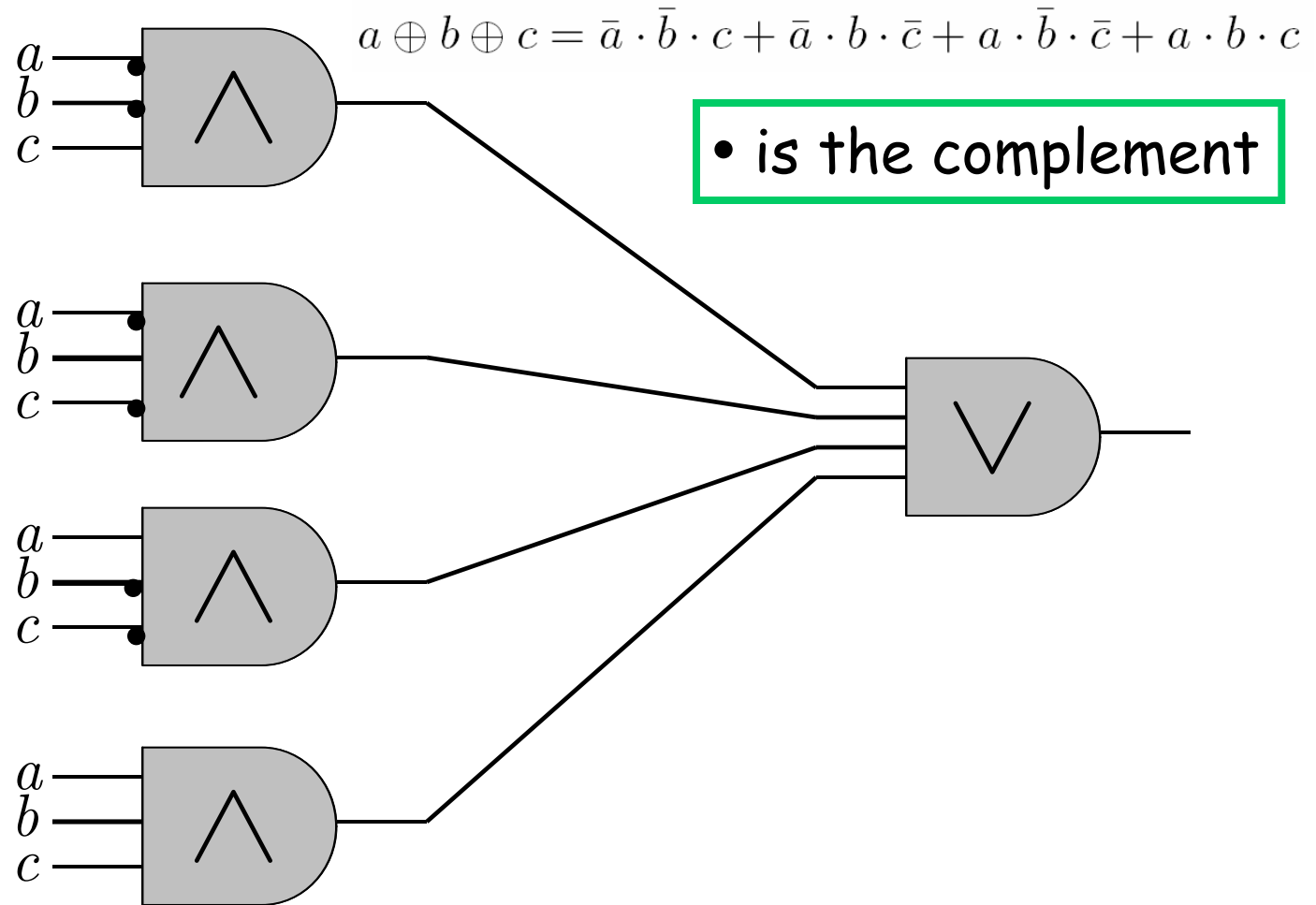
CNS 188a Overview

- Boolean algebra as an axiomatic system
- Boolean functions and their representations using Boolean formulas and spectral methods
- Implementing Boolean functions with **relay** circuits, circuits of **AON** (AND, OR, NOT) gates and **LT** (Linear Threshold) gates
- Analyzing the complexity (size and depth) of circuits
- Relations (as opposed to functions) and their implementation in circuits
- Feedback and convergence in LT circuits

XOR of Three Variables with AON

Depth = 2

Size = 5



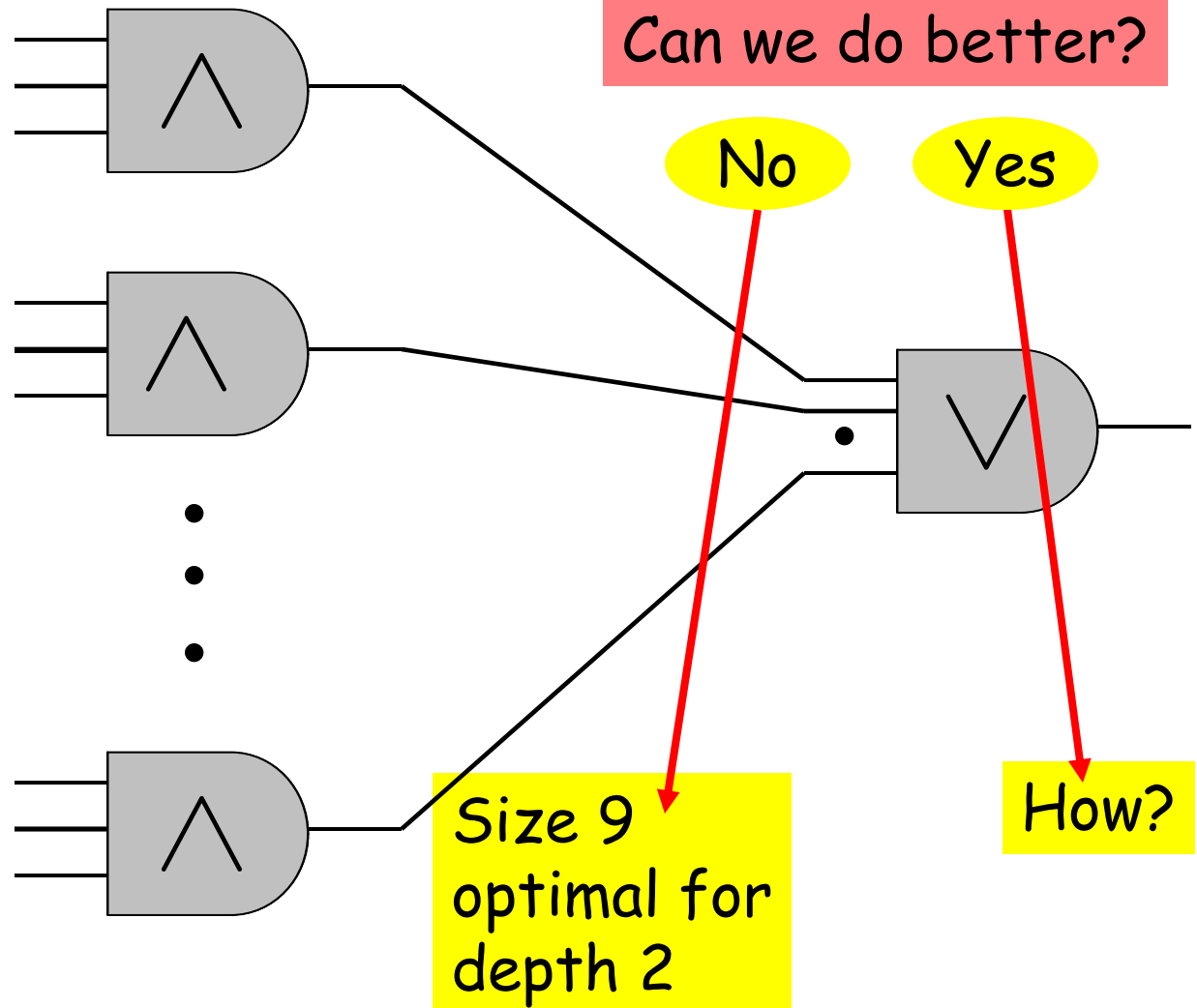
XOR of Four Variables with AON

Depth = 2

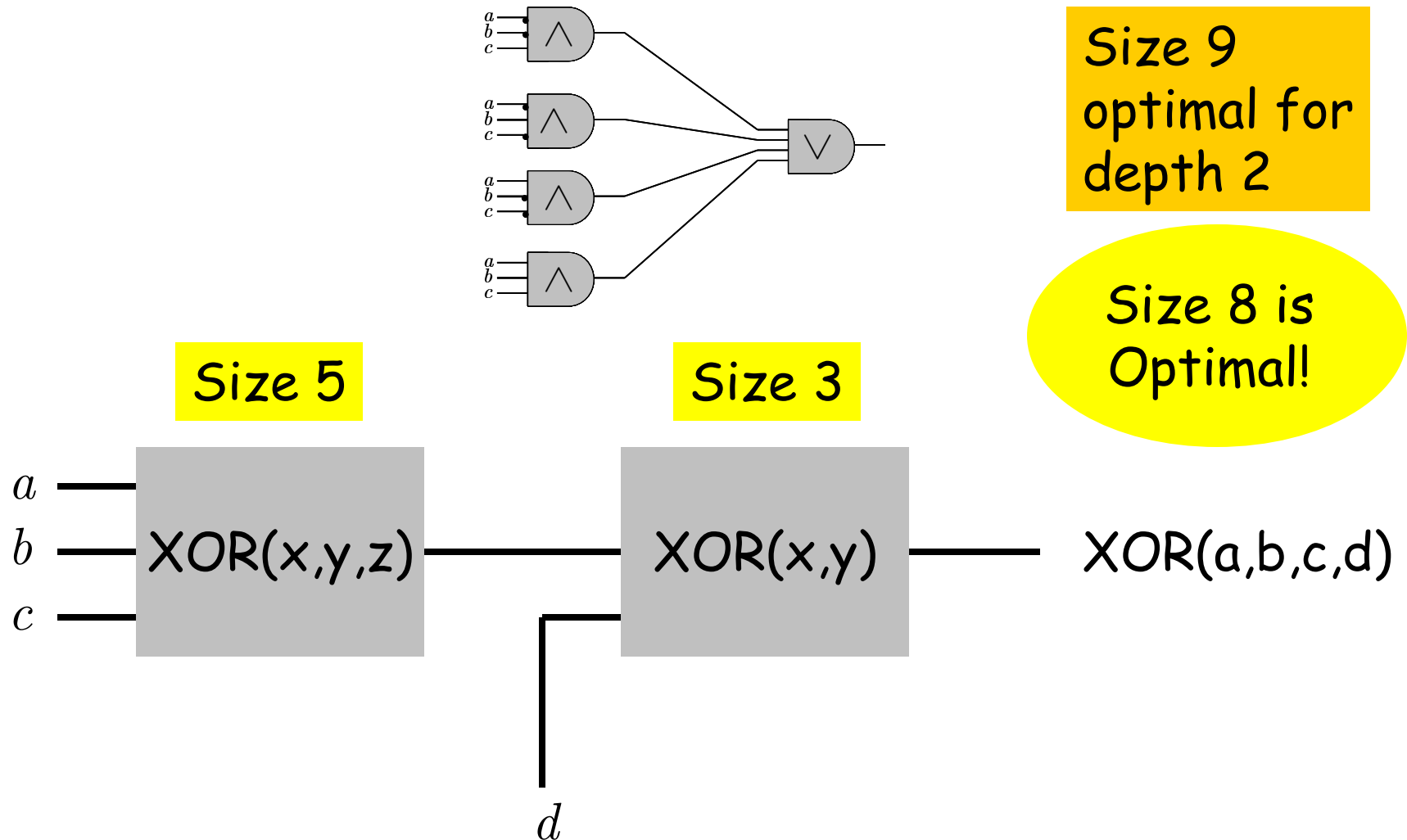
Size = 9

4 *AND*s for $|X| = 1$

4 *AND*s for $|X| = 3$

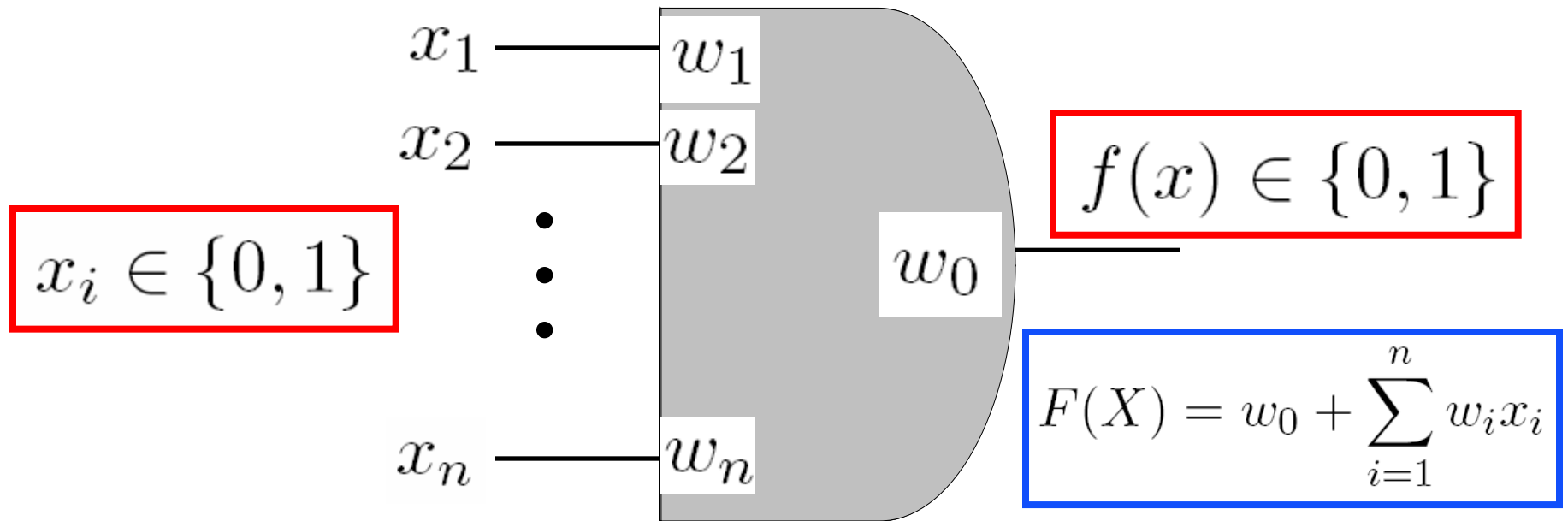


Size 8 XOR AON Circuit of Four Variables



Neuron - Neural Gate

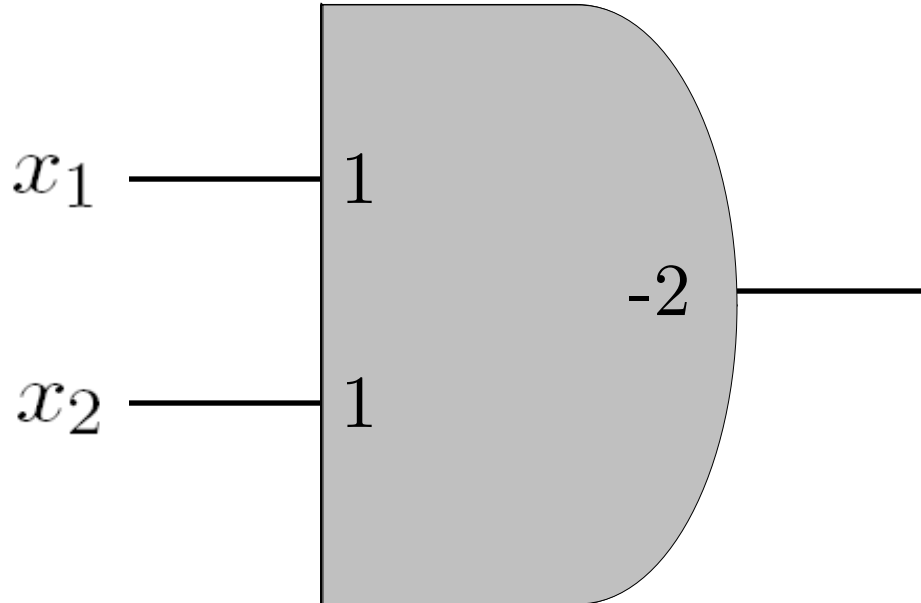
Linear Threshold (LT) gate



$$f(X) = \text{sgn}(F(X)) = \begin{cases} 0 & \text{if } F(X) < 0 \\ 1 & \text{if } F(X) \geq 0 \end{cases}$$

Can We Compute an AND Function with an LT Gate?

$$AND(x_1, x_2)$$



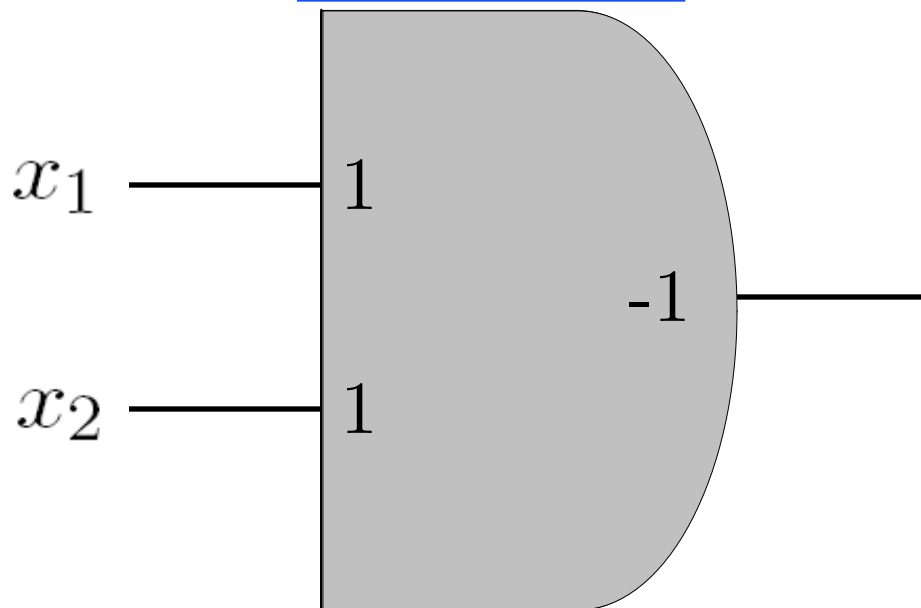
$$F(x_1, x_2) = -2 + x_1 + x_2$$

$$F(X) = w_0 + \sum_{i=1}^n w_i x_i$$

x_1	x_2	$F(X)$	$f(X)$
0	0	-2	0
0	1	-1	0
1	0	-1	0
1	1	0	1

Can We Compute an OR Function with an LT Gate?

$$OR(x_1, x_2)$$

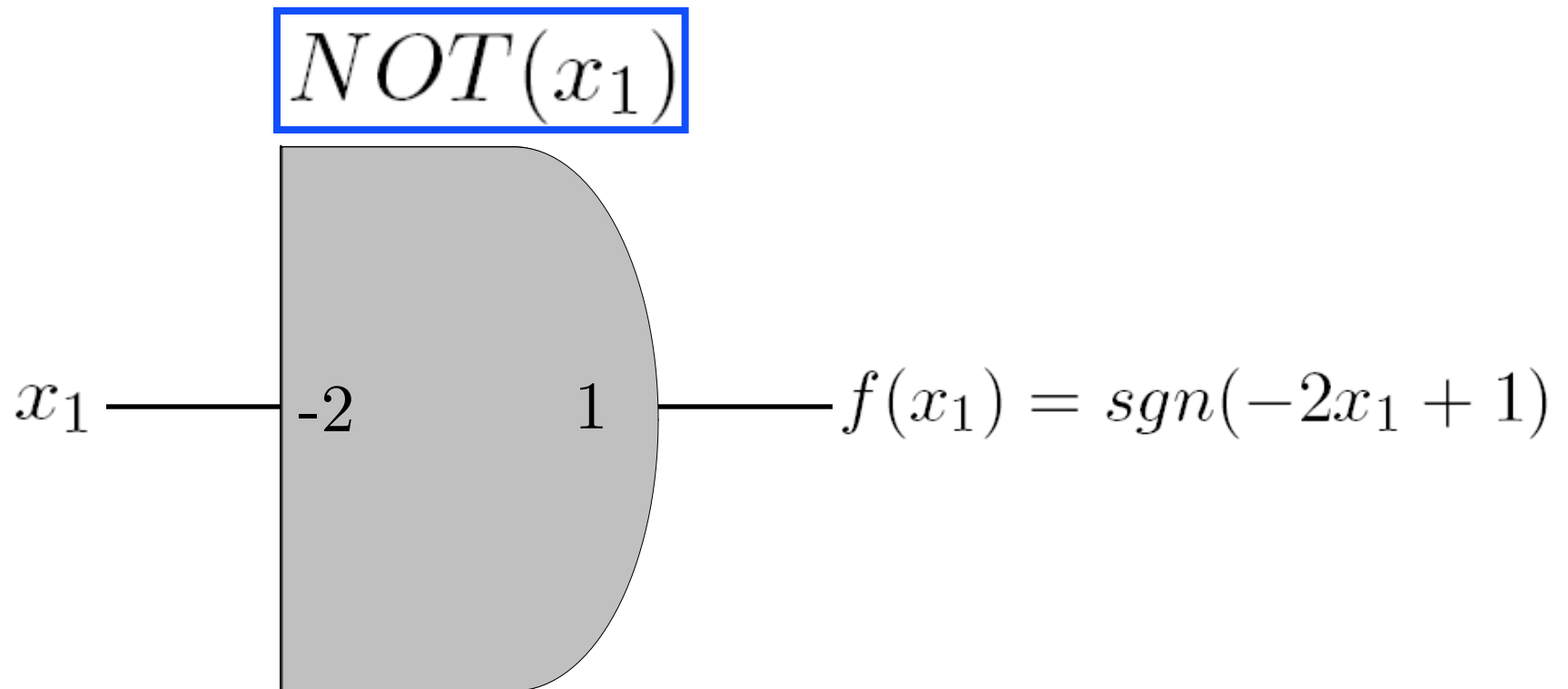


$$F(x_1, x_2) = -1 + x_1 + x_2$$

$$F(X) = w_0 + \sum_{i=1}^n w_i x_i$$

x_1	x_2	$F(X)$	$f(X)$
0	0	-1	0
0	1	0	1
1	0	0	1
1	1	1	1

Can We Compute a NOT with an LT Gate?



Can we compute **NOT**
without sgn ?

$$\bar{x}_1 = NOT(x_1) = 1 - x_1$$

More Variables for AND?

$$F(x_1, x_2) = -2 + x_1 + x_2$$

$$F(x_1, x_2, \dots, x_n) = -n + x_1 + x_2 + \dots + x_n$$

$$|X| = \sum_{i=1}^n x_i$$

$$-n + |X|$$

$$F(X) = \begin{cases} 0 & \text{if } |X| = n \\ < 0 & \text{if } |X| < n \end{cases}$$

Hence $\text{sgn}(F(X))$ is an AND

More Variables for OR?

$$F(x_1, x_2) = -1 + x_1 + x_2$$

$$F(x_1, x_2, \dots, x_n) = -1 + x_1 + x_2 + \dots + x_n$$

$$|X| = \sum_{i=1}^n x_i$$

$$-1 + |X|$$

$$F(X) = \begin{cases} -1 & \text{if } |X| = 0 \\ \geq 0 & \text{if } |X| \geq 1 \end{cases}$$

Hence $\text{sgn}(F(X))$ is an OR

AON vs. LT Circuits

Theorem (LT/AON):

LT circuits are at least as powerful as AON circuits:

(i) **Size** of an **AON** circuit for a Boolean function f is greater/equal
size of an optimal **LT** circuit for the function f

(ii) **Depth** of an **AON** circuit for a Boolean function f is greater/equal
depth of an optimal **LT** circuit for the function f

Proof:

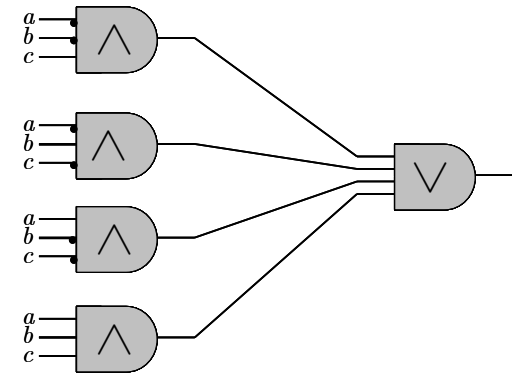
By the previous reductions:
AND, *OR* and *NOT* can be implemented by a single LT gate

LT gates are MORE Powerful

XOR of three variables with AON

Depth = 2

Size = 5



XOR of four variables with AON

Depth = 2

Size = 9

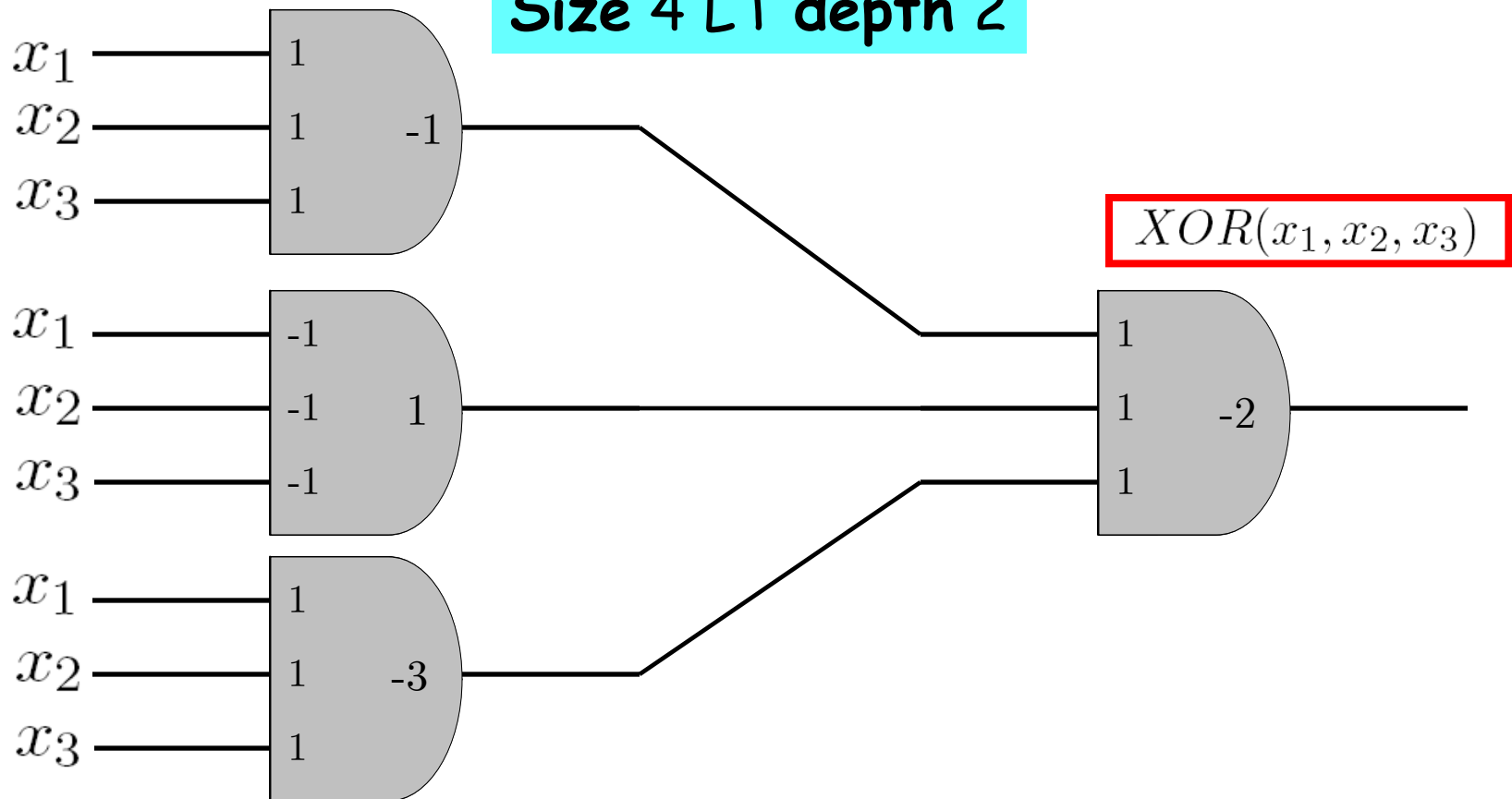
Size is optimal for depth 2

XOR of four variables with AON,
size 8 is optimal for any depth

LT gates are MORE Powerful

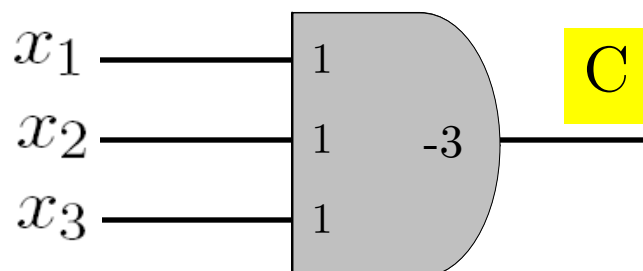
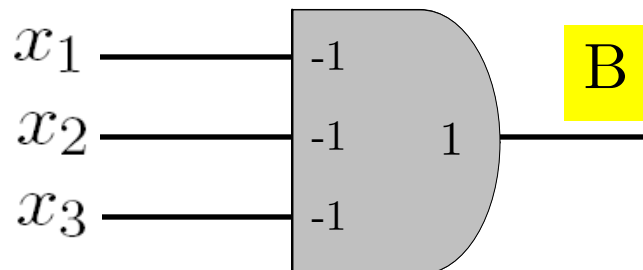
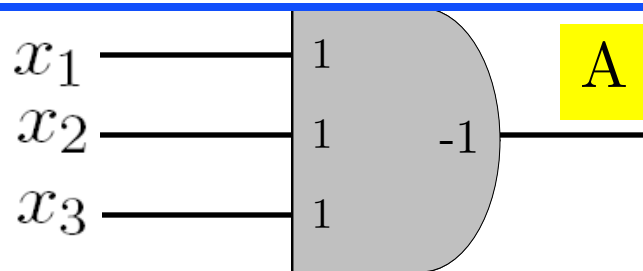
Size 5 is optimal for AON depth 2

Size 4 LT depth 2

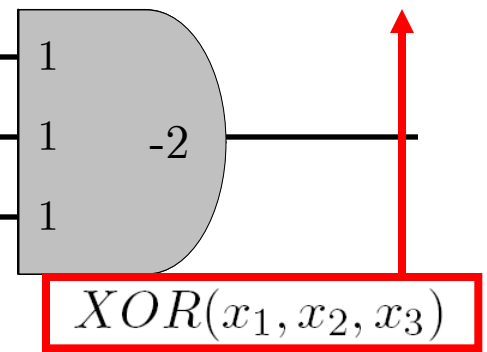


LT gates are MORE Powerful

LT-I = LT layered, inputs go to first layer only

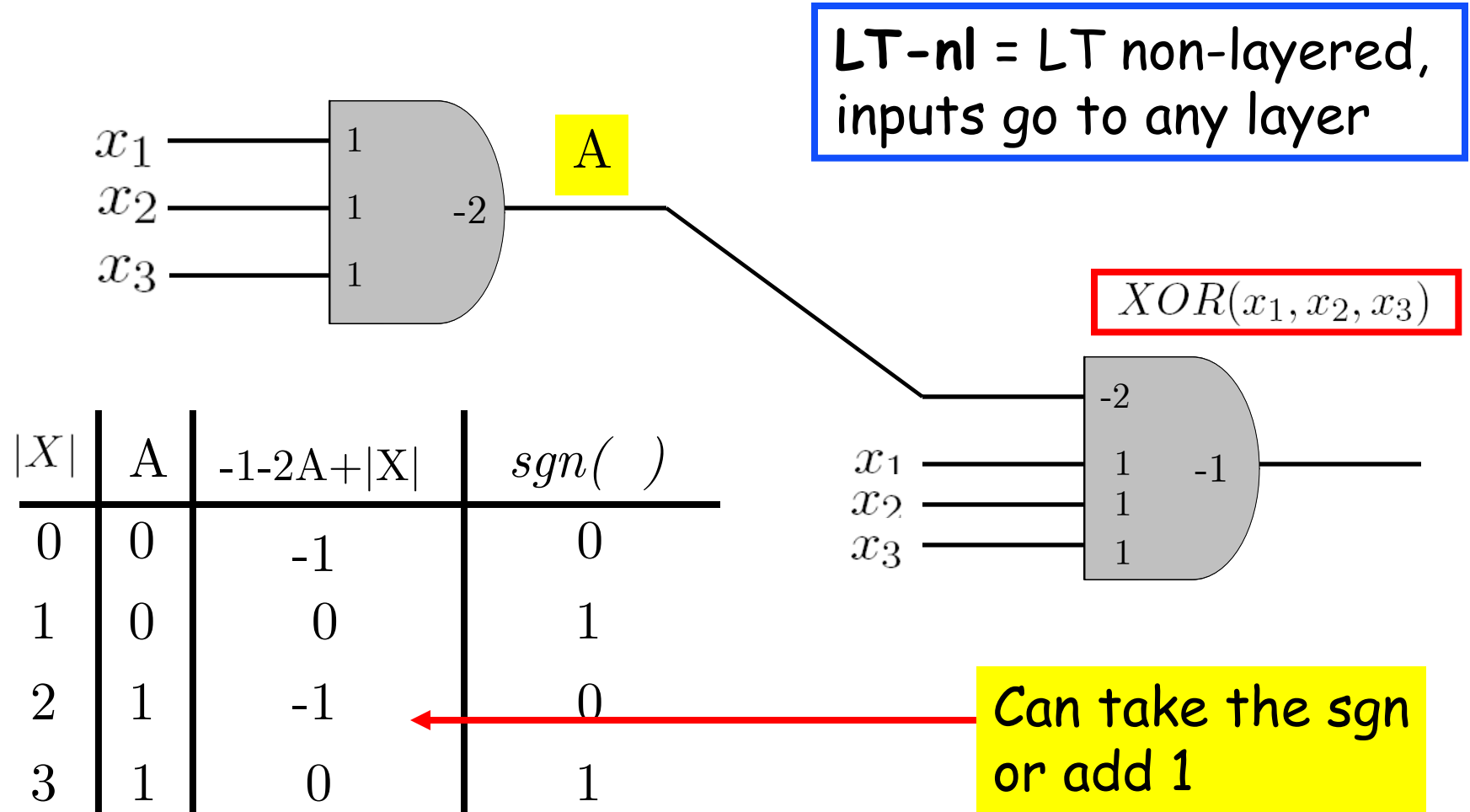


$ X $	A	B	C	A+B+C	-2+A+B+C
0	0	1	0	1	-1
1	1	1	0	2	0
2	1	0	0	1	-1
3	1	0	1	2	0



Can take the sgn or add 1

LT gates are EVEN MORE Powerful



XOR Function: Size of LT vs. AON in Depth 2

	$XOR(x_1, x_2, x_3)$	$XOR(x_1, x_2, \dots, x_n)$
AON	5	$2^{n-1} + 1^*$
LT-I	4	$n + 1^*$
LT-nl	2	$\left\lfloor \frac{n}{2} \right\rfloor + 1^*$

* = it is optimal

Exponential gap in size

XOR with a Single LT Gate

Is it possible to compute $XOR(x_1, x_2)$ with a single LT element?

Answer : NO

Proof: By contradiction
assume it is possible and reach a contradiction

$$XOR(x_1, x_2) = \text{sgn}(w_0 + w_1x_1 + w_2x_2)$$

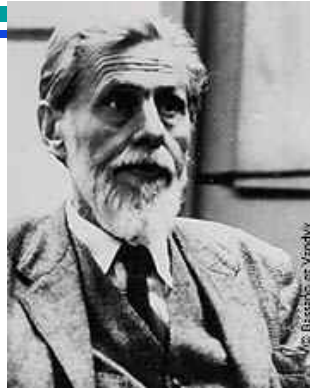
$x_1 = 0$	$x_2 = 0$	$\longrightarrow$	$w_0 < 0$	$\longrightarrow$	$2w_0 + w_1 + w_2 < 0$
$x_1 = 1$	$x_2 = 0$	$\longrightarrow$	$w_0 + w_1 \geq 0$	$\longrightarrow$	$2w_0 + w_1 + w_2 \geq 0$
$x_1 = 0$	$x_2 = 1$	$\longrightarrow$	$w_0 + w_2 \geq 0$	$\longrightarrow$	$2w_0 + w_1 + w_2 \geq 0$
$x_1 = 1$	$x_2 = 1$	$\longrightarrow$	$w_0 + w_1 + w_2 < 0$	$\longrightarrow$	$2w_0 + w_1 + w_2 \geq 0$

Contradiction

History on LT

Warren McCulloch
1899 - 1969

Neurophysiologist, MD



Walter Pitts
1923 - 1969

Logician, Autodidact



Computing with LT (neural) gates: connection between circuits and neural networks, **1943**

Warren McCulloch arrived in early 1942 to the University of Chicago, invited Pitts, who was still homeless, to live with his family.

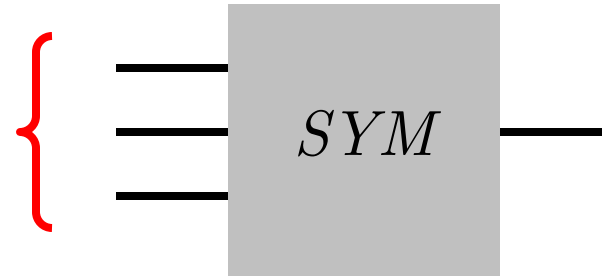
In the evenings McCulloch and Pitts collaborated. Pitts was familiar with the work of Gottfried Leibniz on computing and they considered the question of whether the nervous system could be considered a kind of universal computing device as described by Leibniz.

This led to their 1943 seminal neural networks paper:
A Logical Calculus of Ideas Immanent in Nervous Activity.

Symmetric Functions

AND, *OR* and *XOR* are symmetric Boolean functions

Permuting the inputs does
not change the output



$$XOR(x_1, x_2, x_3) = XOR(x_2, x_1, x_3) = XOR(x_2, x_3, x_1)$$

Definition: A Boolean function f is symmetric if

$$f(X) = f(\pi(X))$$

for an arbitrary permutation π

Questions on Symmetric Functions

AND, *OR* and *XOR* are symmetric Boolean functions

Q1: How many symmetric Boolean functions of n variables?

Q2: What are the symmetric functions that can be realized by a single LT gate?

Symmetric Functions

Definition: A Boolean function f is symmetric if

$$f(X) = f(\pi(X))$$

for an arbitrary permutation π

Theorem: A Boolean function $f(X)$ is symmetric if and only if it is a function of the number of 1's in X , namely $|X|$

Symmetric Functions

Theorem: A Boolean function $f(X)$ is symmetric if and only if it is a function of the number of 1's in X , namely $|X|$

Proof:

f a function of $|X| \longrightarrow f$ symmetric

Given: $|X_1| = |X_2| \longrightarrow f(X_1) = f(X_2)$

Need to prove: $f(X) = f(\pi(X)) \quad \forall \pi, \forall X$

However, $|X| = |\pi(X)| \quad \forall \pi, \forall X$

$\longrightarrow f(X) = f(\pi(X)) \quad \forall \pi, \forall X$

Symmetric Functions

Theorem: A Boolean function $f(X)$ is symmetric if and only if it is a function of the number of 1's in X , namely $|X|$

Proof:

f symmetric



f a function of $|X|$

Given: $f(X) = f(\pi(X)) \quad \forall \pi, \forall X$

Need to prove: $|X_1| = |X_2| \longrightarrow f(X_1) = f(X_2)$

However, $|X_1| = |X_2| \longrightarrow \exists \pi, X_2 = \pi(X_1)$

Hence, $f(X_1) = f(\pi(X_1)) = f(X_2)$

Questions on Symmetric Functions

Q1: How many symmetric Boolean functions of n variables?

Q2: What are the symmetric functions that can be realized by a single LT gate?

A1:

Theorem: The number of symmetric functions of n variables is: 2^{n+1}

Number of Symmetric Functions

Theorem: The number of symmetric functions of n variables is: 2^{n+1}

Proof:

f symmetric



f a function of $|X|$

Symmetric
function
table



$ X $	$f(X)$
0	*
1	*
2	*
•	•
•	•
•	•
n	*

* can be 0 or 1



2^{n+1} functions

Questions on Symmetric Functions

Q1: How many symmetric Boolean functions of n variables?

Q2: What are the symmetric functions that can be realized by a single LT gate?

Definitions:

(1) **SYM** = the class of Boolean symmetric functions

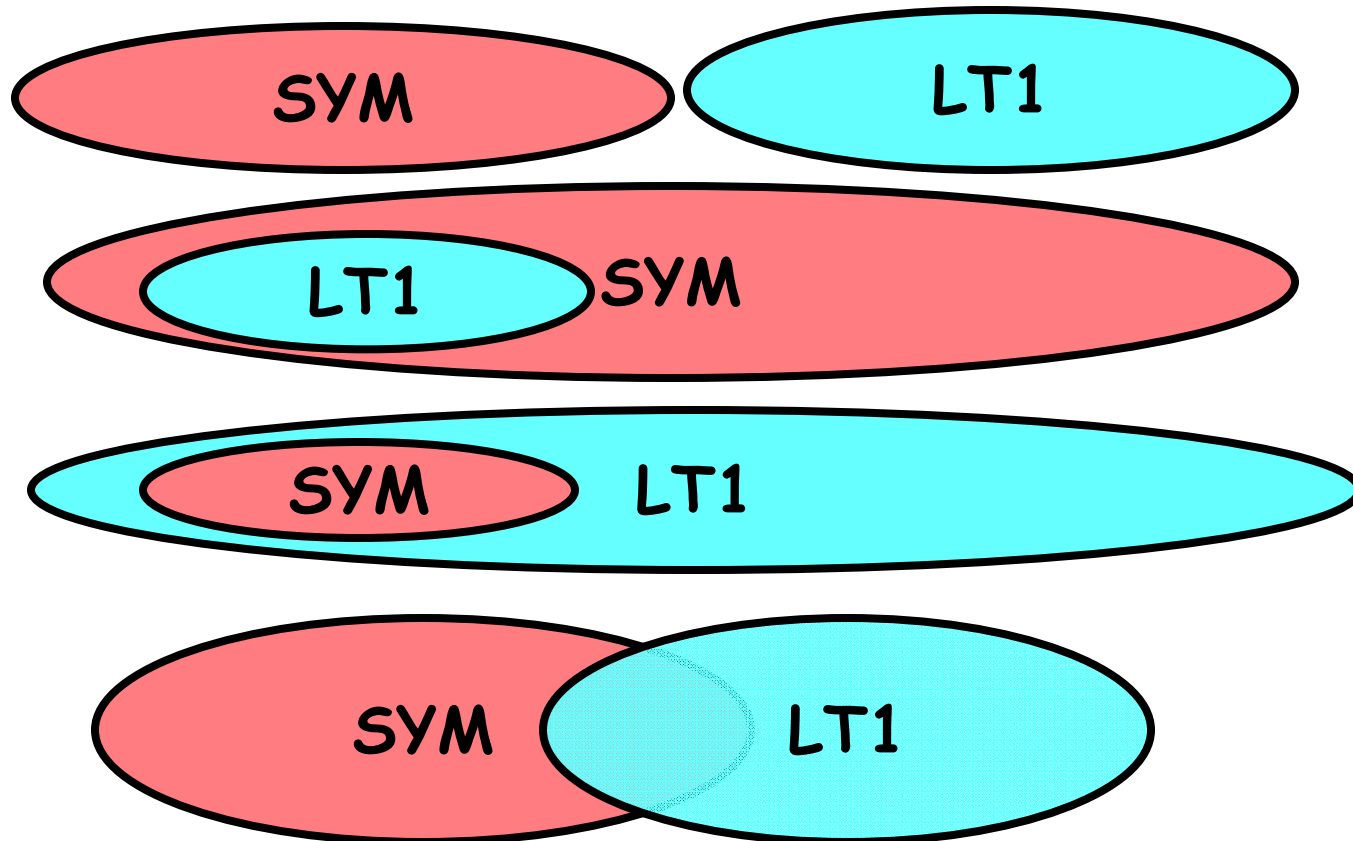
(2) **LT_1** = the class of Boolean functions that can be realized by a single LT gate.

Q2: How is **SYM** related to **LT_1** ??

Questions on Symmetric Functions

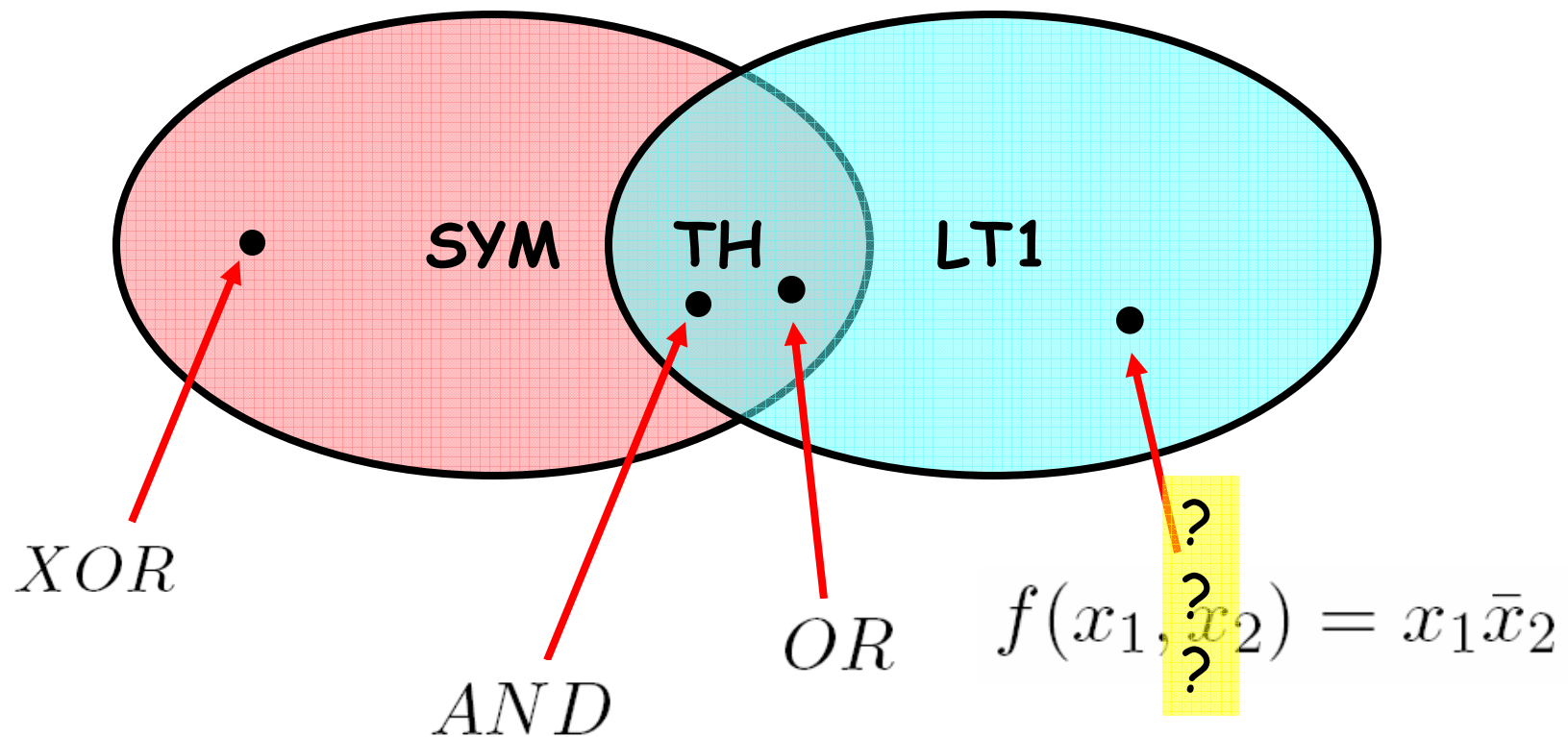
Q2: How is *SYM* related to *LT*₁ ??

Q3: Which class has more functions?



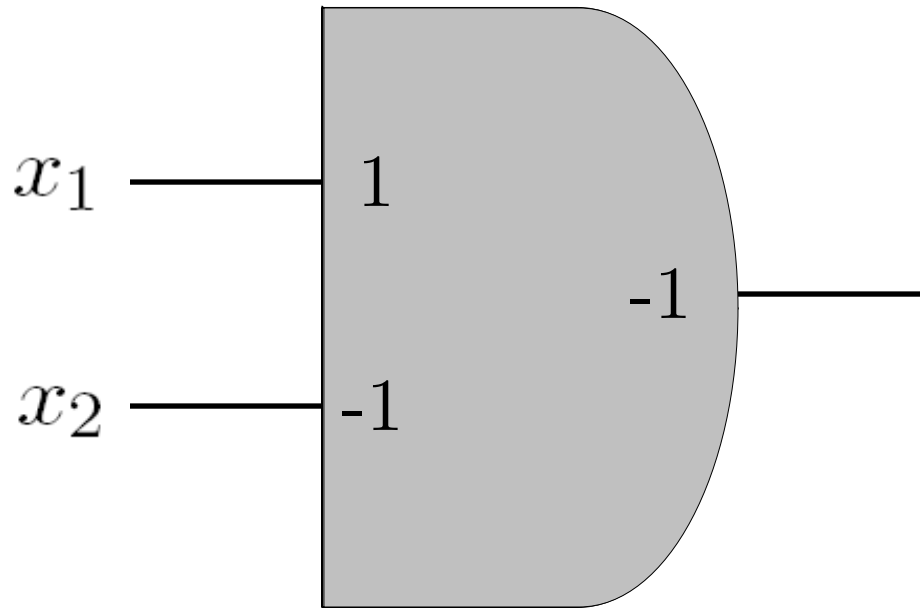
SYM and *LT*₁

Theorem: $SYM \cap LT_1 = TH$



LT₁ Function that is not Symmetric

$$f(x_1, x_2) = x_1 \bar{x}_2$$

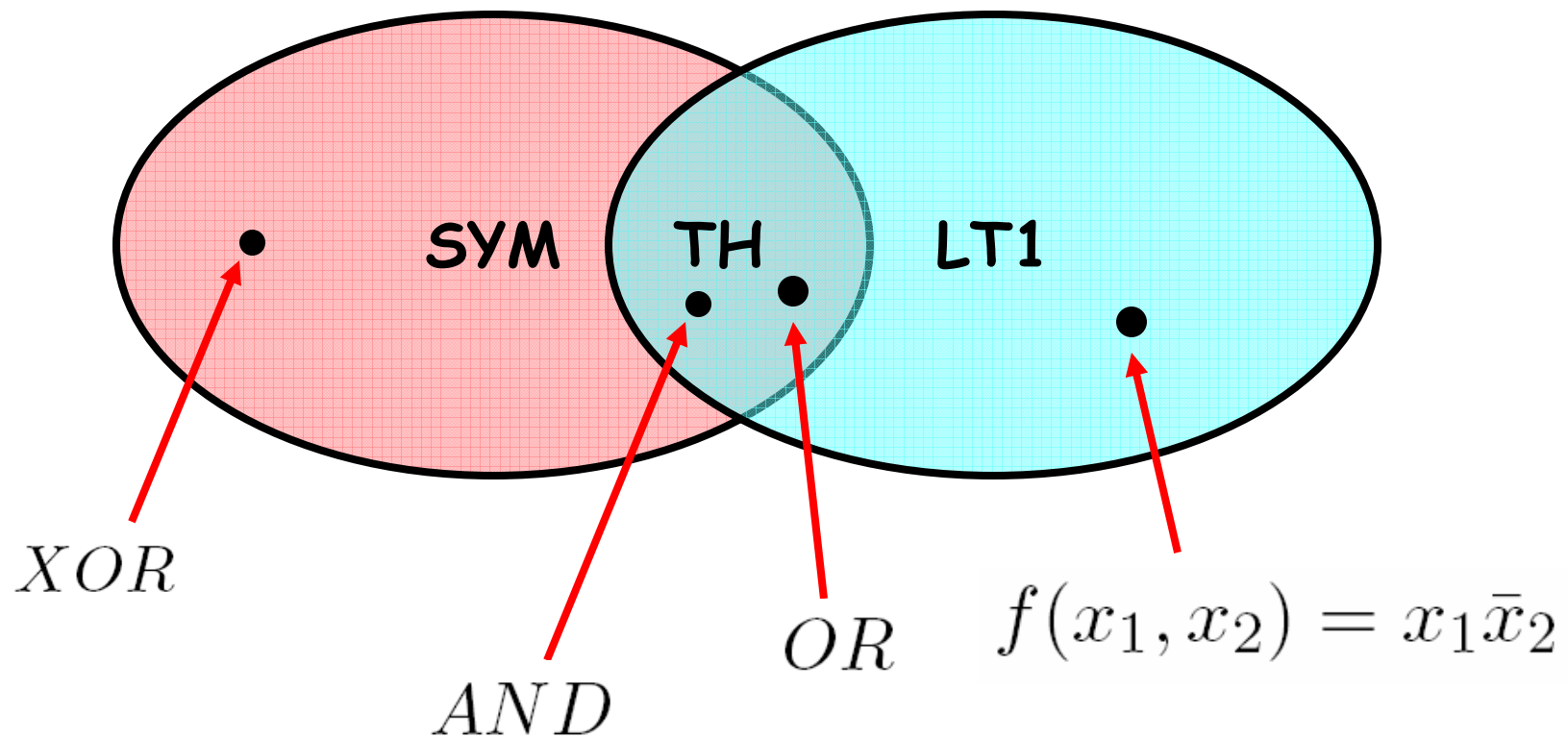


x_1	x_2	$F(X)$	$f(X)$
0	0	-1	0
0	1	-2	0
1	0	0	1
1	1	-1	0

$$F(x_1, x_2) = -1 + x_1 - x_2$$

SYM and *LT*₁

Q: What is *TH* ?



SYM and LT_1

Q: What is TH ?

Definition: A symmetric Boolean function is in TH if it has at most a single transition in the symmetric function table

$ X $	<i>AND</i>	<i>OR</i>	<i>XOR</i>
0	0	0	0
1	0	1	1
2	1	1	0



= a transition

Not in TH

The Class TH

Definition: A symmetric Boolean function is in TH if it has at most a single transition in the **symmetric function table**

$$TH_i(x_1, x_2, \dots, x_n) = \begin{cases} 1 & \text{if } |X| \geq i \\ 0 & \text{if } |X| < i \end{cases}$$

$$AND(X) = TH_n$$

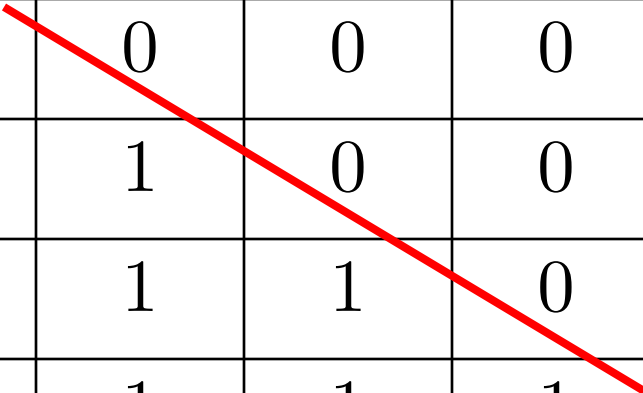
$$OR(X) = TH_1$$

$$MAJ(X) = TH_{\lceil \frac{n}{2} \rceil}$$

The Class TH

$$TH_i(x_1, x_2, \dots, x_n) = \begin{cases} 1 & \text{if } |X| \geq i \\ 0 & \text{if } |X| < i \end{cases}$$

$ X $	TH_0	TH_1	TH_2	TH_3
0	1	0	0	0
1	1	1	0	0
2	1	1	1	0
3	1	1	1	1



The Class TH

$$TH_i(x_1, x_2, \dots, x_n) = \begin{cases} 1 & \text{if } |X| \geq i \\ 0 & \text{if } |X| < i \end{cases}$$

$$\overline{TH}_i = \{f \mid \bar{f} = TH_i\}$$

$$TH = \{TH_i, \overline{TH}_i \mid 0 \leq i \leq n\}$$

$ X $	TH_0	TH_1	TH_2	TH_3	$\overline{TH}_0$	$\overline{TH}_1$	$\overline{TH}_2$	$\overline{TH}_3$
0	1	0	0	0	0	1	1	1
1	1	1	0	0	0	0	1	1
2	1	1	1	0	0	0	0	1
3	1	1	1	1	0	0	0	0

The Class TH

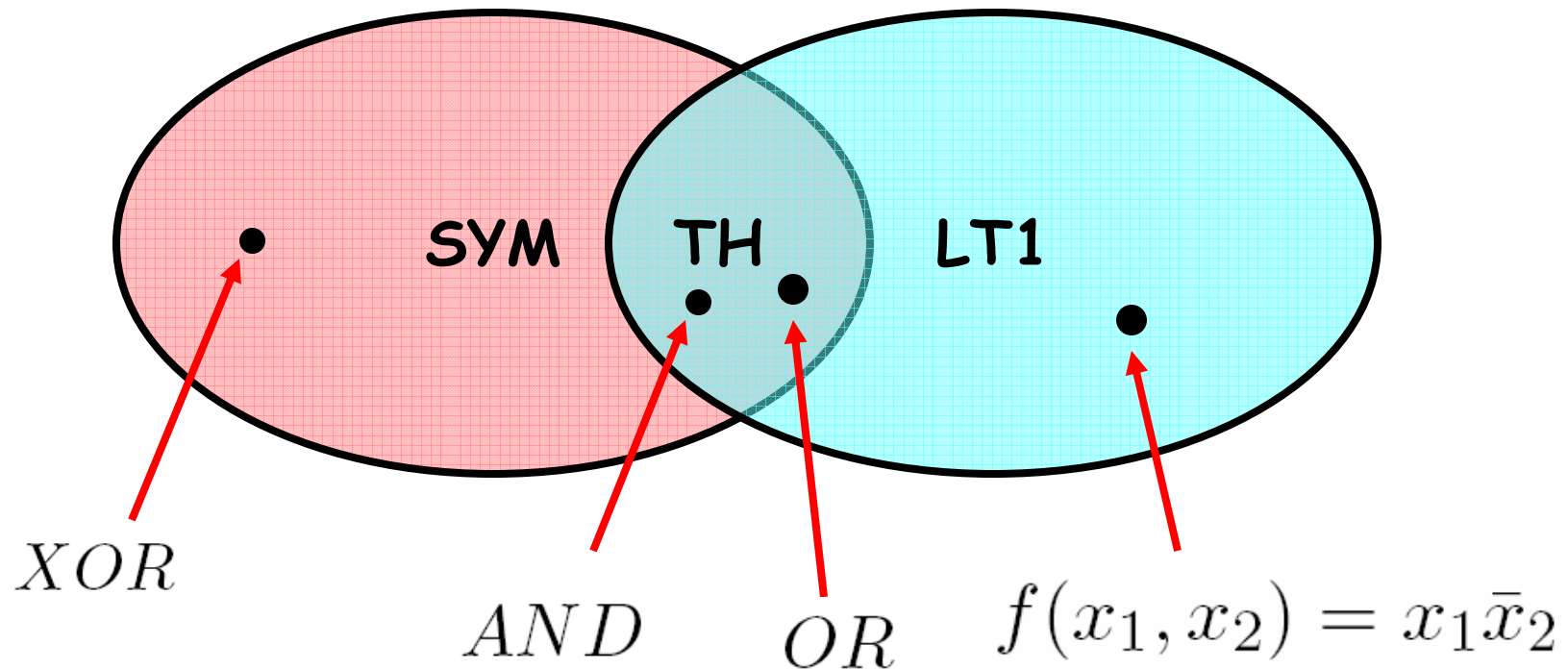
$$TH = \{TH_i, \overline{TH}_i \mid 0 \leq i \leq n\}$$

Q: what is $|TH|$?

A: $2n+2$

$ X $	TH_0	TH_1	TH_2	TH_3	$\overline{TH}_0$	$\overline{TH}_1$	$\overline{TH}_2$	$\overline{TH}_3$
0	1	0	0	0	0	1	1	1
1	1	1	0	0	0	0	1	1
2	1	1	1	0	0	0	0	1
3	1	1	1	1	0	0	0	0

Back to the Theorem



We know that: $TH \subset SYM$

First, prove that: $TH \subseteq LT_1$

Second, prove that: $SYM \cap LT_1 = TH$

$$SYM \cap LT_1 \supseteq TH$$

Easy Part of the Proof

We know that: $TH \subset SYM$

First, prove that:

$$TH \subseteq LT_1$$

$$SYM \cap LT_1 \supseteq TH$$

$$TH_i(x_1, x_2, \dots, x_n) = \begin{cases} 1 & \text{if } |X| \geq i \\ 0 & \text{if } |X| < i \end{cases}$$

$$TH_i(x_1, x_2, \dots, x_n) = \text{sgn}(-i + x_1 + x_2 + \dots + x_n)$$

$$= \begin{cases} 1 & \text{if } |X| \geq i \\ 0 & \text{if } |X| < i \end{cases}$$

$$\overline{TH}_i = \text{sgn}((i - 1) - (x_1 + x_2 + \dots + x_n))$$

Harder Part of the Proof

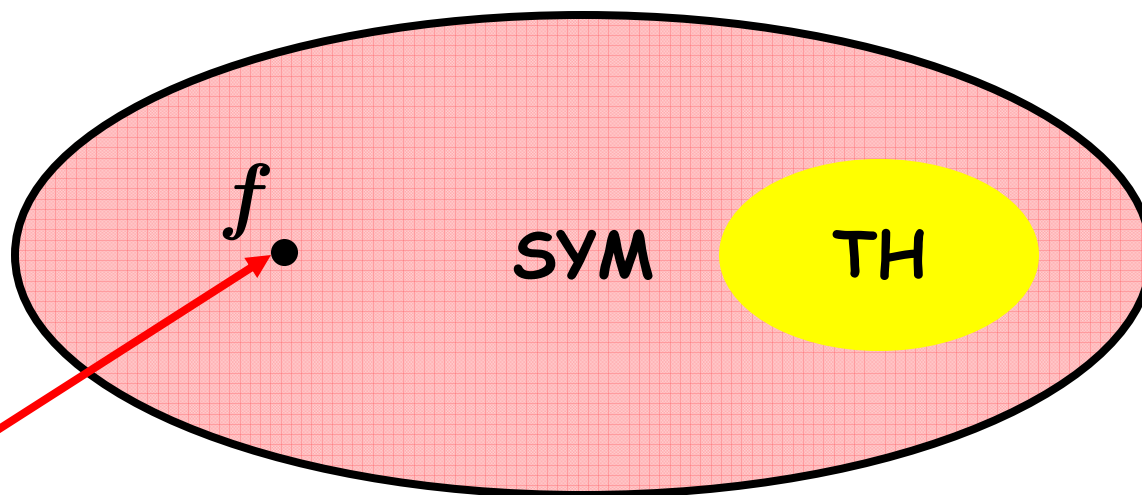
We proved that:

$$SYM \cap LT_1 \supseteq TH$$

Need to prove that:

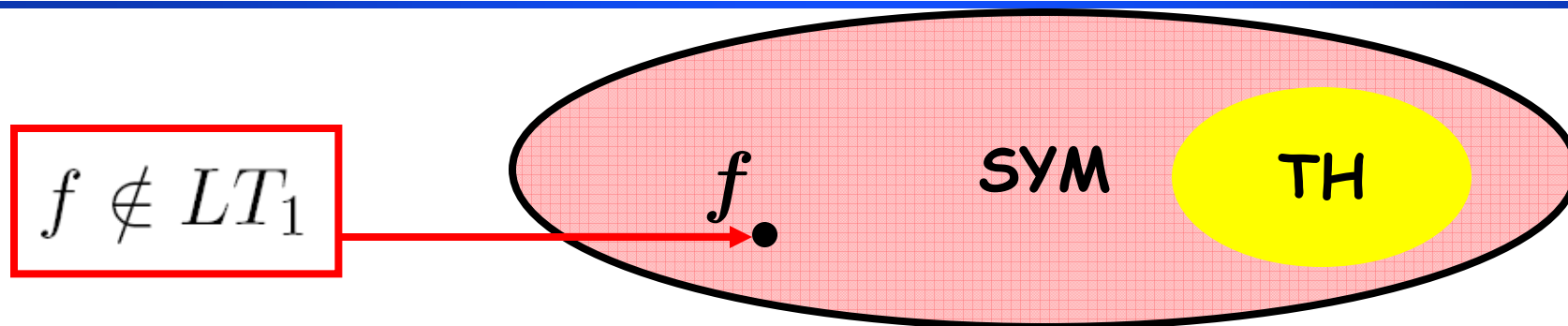
$$SYM \cap LT_1 = TH$$

Idea??



$$f \notin LT_1$$

Harder Part of the Proof



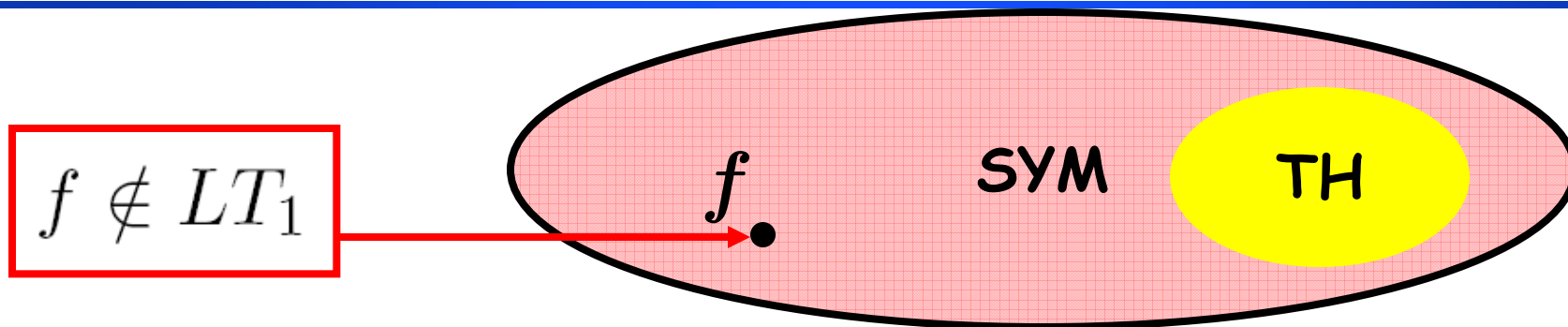
Idea: proof by contradiction, assume $f \in LT_1$ exists

The function f is in **SYM** and not **TH** , **wlog:**

$ X $	0	1	2	3	*	*	*	*	n
$f(X)$	0	1	1	1	1	1	1	1	0

without loss of generality

Harder Part of the Proof



Idea: proof by contradiction, assume $f \in LT_1$ exists

The function f is in **SYM** and not **TH** , **wlog:**

$ X $	0	1	2	3	*	*	*	*	n
$f(X)$	0	1	1	1	1	1	1	1	0

without loss of generality

Why WLOG?

Idea: proof by contradiction, assume $f \in LT_1$ exists

The function f is in **SYM** and not **TH**, **wlog**:

$ X $	0	1	2	3	4	5
$f(X)$	0	0	1	0	1	1

$$F(X) = w_0 + w_1x_1 + w_2x_2 + w_3x_3 + w_4x_4 + w_5x_5$$

$$x_3 = 1 \quad x_4 = 0 \quad x_5 = 0$$

$$F(X) = w_0 + w_1x_1 + w_2x_2 + w_3 + 0 + 0$$

$$1 \leq |X| = x_1 + x_2 + 1 \leq 3$$

Why WLOG?

Idea: proof by contradiction, assume $f \in LT_1$ exists

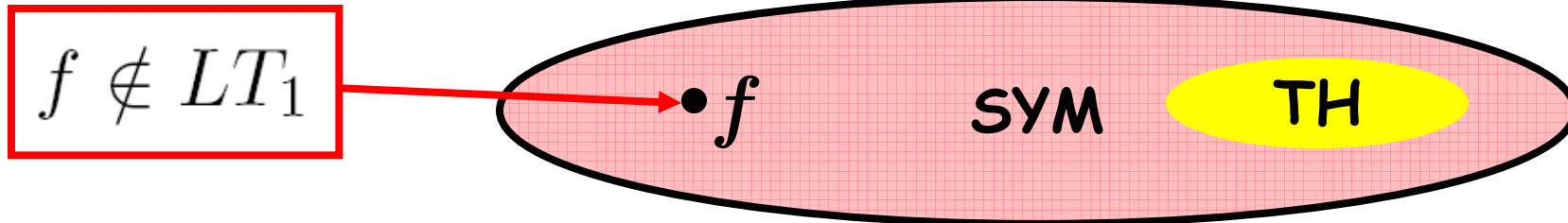
The function f is in **SYM** and not **TH**, **wlog**:

$ X $	0	1	2	3	4	5
$f(X)$	0	0	1	0	1	1

$$1 \leq |X| = x_1 + x_2 + 1 \leq 3$$

$ X $	$f(X)$
0	0
1	1
2	0

Harder Part of the Proof

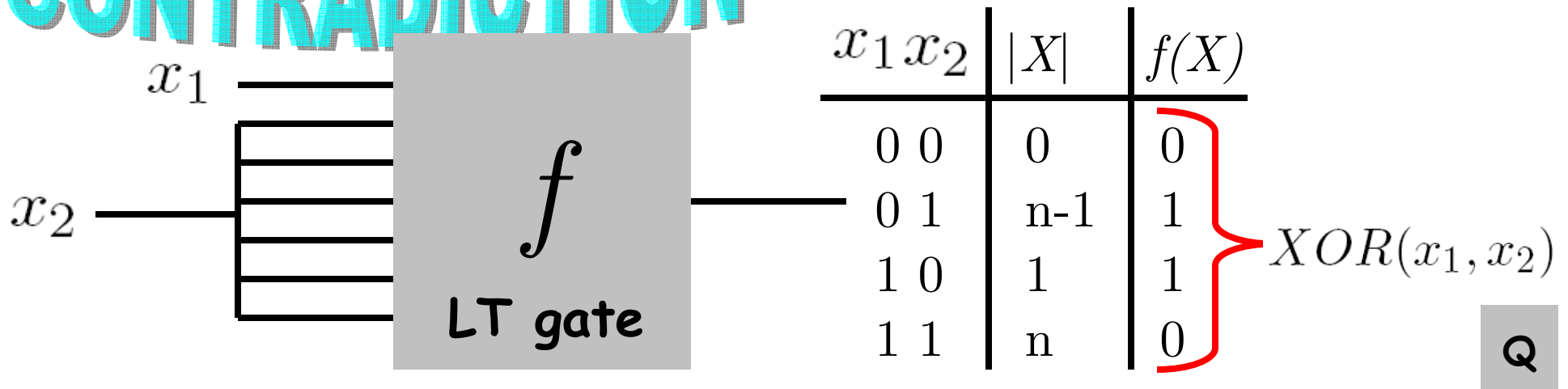


Idea: proof by contradiction, assume $f \in LT_1$ exists

WLOG f is:

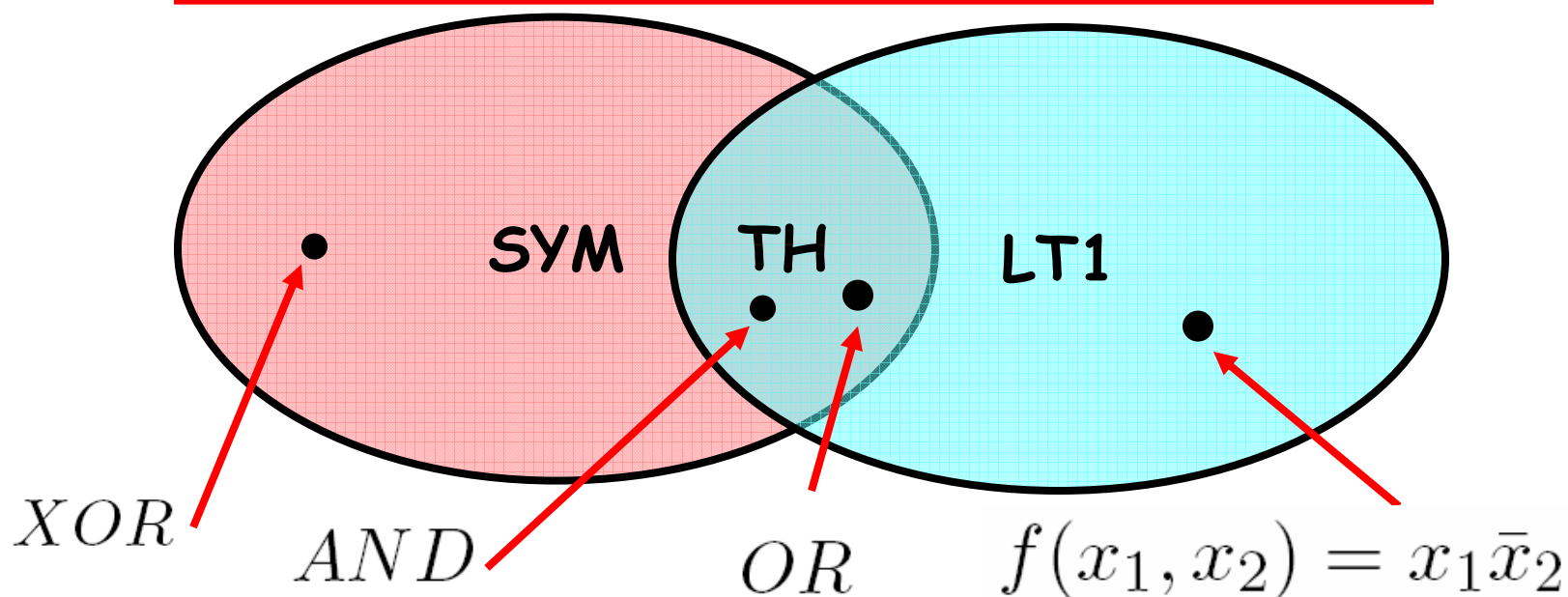
$ X $	0	1	2	3	*	*	*	*	n
$f(X)$	0	1	1	1	1	1	1	1	0

CONTRADICTION



We Proved the Theorem

Theorem: $SYM \cap LT_1 = TH$



$2n+2$ functions of **SYM** are in **LT1**,
what about the rest of **SYM**?

More Layers

$2n+2$ functions of **SYM** are in **LT₁**,
what about the rest of **SYM**?

Definition:

LT₂ = the class of Boolean functions that can be realized by a two layer circuit of **LT** gates, size is bounded by $\text{poly}(n)$, for example $n, n^2, n^3, \dots, n^k$ where k is a constant independent of n

Q: How is **SYM** related to **LT₂** ??

How is *SYM* related to LT_2 ?

