Durham University

COMP1071 - Digital Electronics

Circuits

Ioannis Ivrissimtzis

email: ioannis.ivrissimtzis@durham.ac.uk

Room: MCS 2023

Slide acknowledgements: Eleni Akrida and Farshad Arvin

Overview of today's lecture

- Adder
- Subtractor
- Decoder
- Tristates
- Mux
- Simple ALU

Key circuits

Combinatorial / combinational circuits:

(Output depends only on current input)

Adders – add the contents of two registers

Decoders – decodes an n-digit binary number into 2^n data lines

Multiplexors – use a binary number to select an input

Sequential circuits

(Output depends on state and input, i.e. history of input)

Latches / flip-flops – basic memory element

Half-Adder

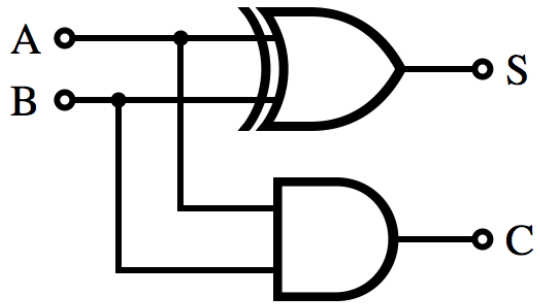
Based on 8 simple rules:

0 + 0 = 0	Carry + 0 + 0 = 1
0 + 1 = 1	Carry + 0 + 1 = 0 with Carry
1 + 0 = 1	Carry + 1 + 0 = 0 with Carry
1 + 1 = 0 with Carry	Carry + 1 + 1 = 1 with Carry

Inputs: A, B

Outputs: Sum, Carry

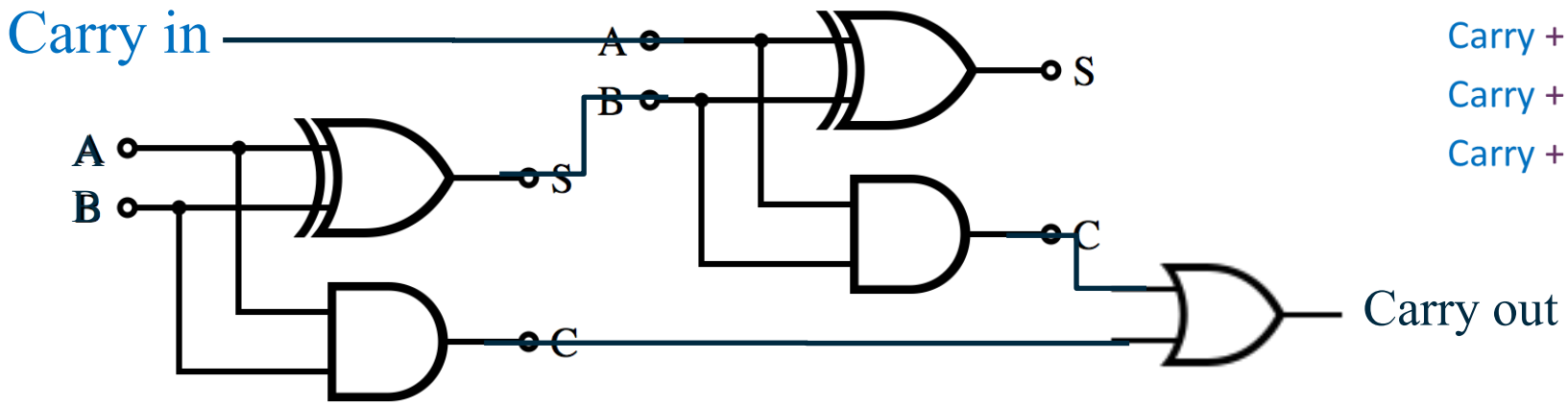
A	B	Sum	Carry
0	0	**0**	**0**
0	1	**1**	**0**
1	0	**1**	**0**
1	1	**0**	**1**



Adder

Input is not just A and B, but A, B and **the carry from the previous bit**

Use two half-adders: Add A and B first, then add in the carried bit:



Carry + 0 + 0 = 1

Carry + 0 + 1 = 0 with Carry

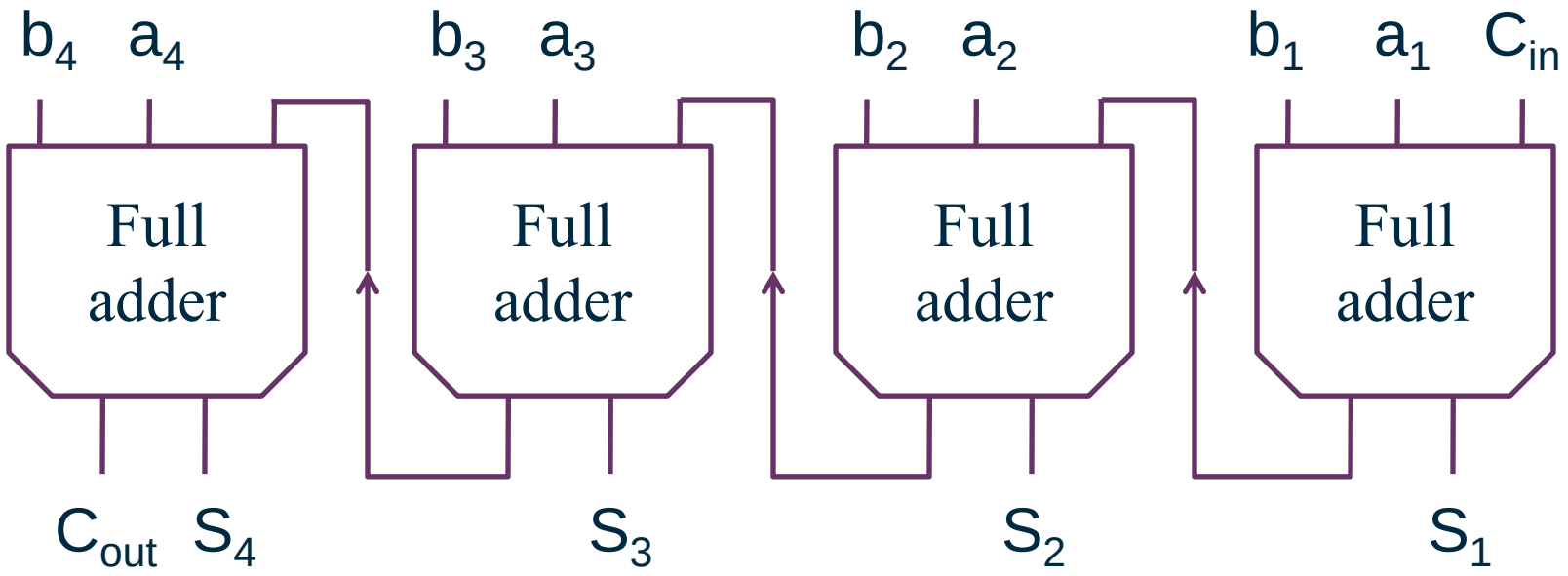
Carry + 1 + 0 = 0 with Carry

Carry + 1 + 1 = 1 with Carry

“Full adder”

Chaining adders – ripple carry adder

How to add two 4 bit registers:

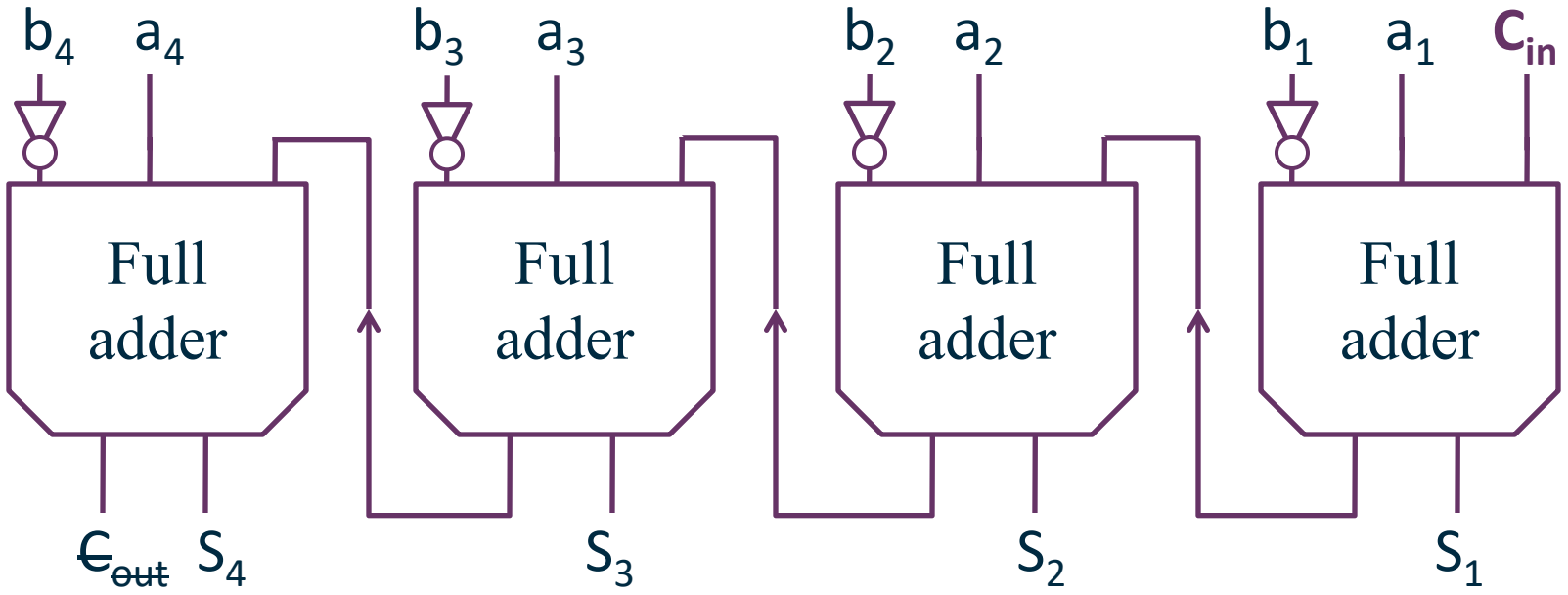


More full-adders can be chained together to give 8-, 16- or 32-bit adders.

Subtractor

Recall that we can create a **twos-complement negative** by inverting each bit and adding 1.

We subtract b from a by **adding -b**.



The NOT gates and setting $C_{in}=1$ give the effect of converting b to -b.

Beyond Simple Logic Gates - Decoder

A decoder has *N* inputs and 2^N outputs. Asserts exactly one of its outputs depending on the binary number represented by the *N* input bits.

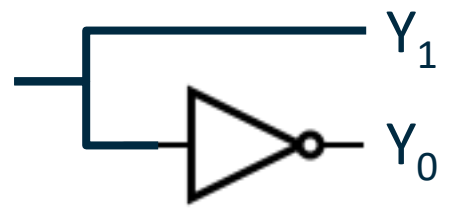
The outputs are called *one-hot*, because exactly one is “hot” (high) at a given time.

N inputs, $\mathbf{2^N}$ outputs.

Only one output is high at a time.

Which one depends on the input.

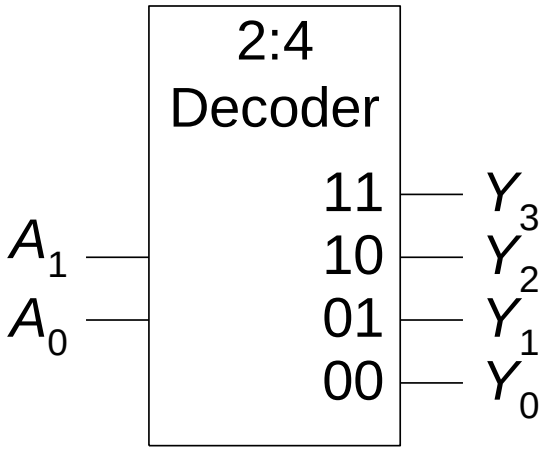
1-bit decoder: A



Durham University

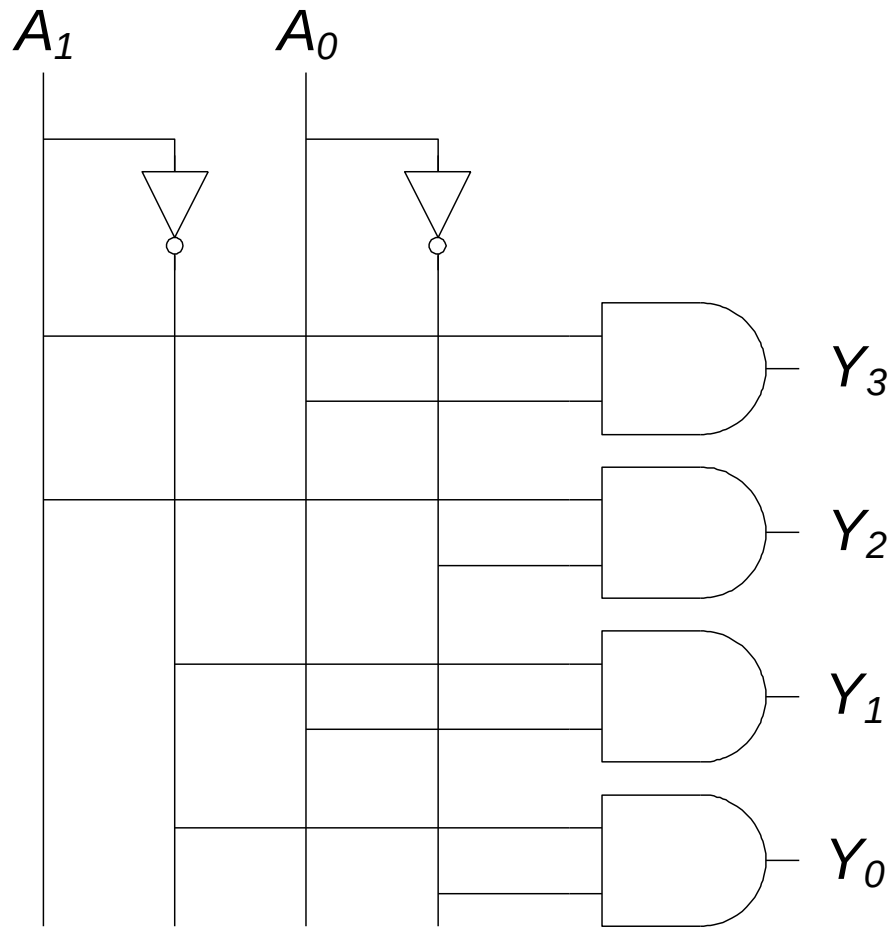
Decoder

2-bit decoder:



Unique 1 in each row

A_1	A_0	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0



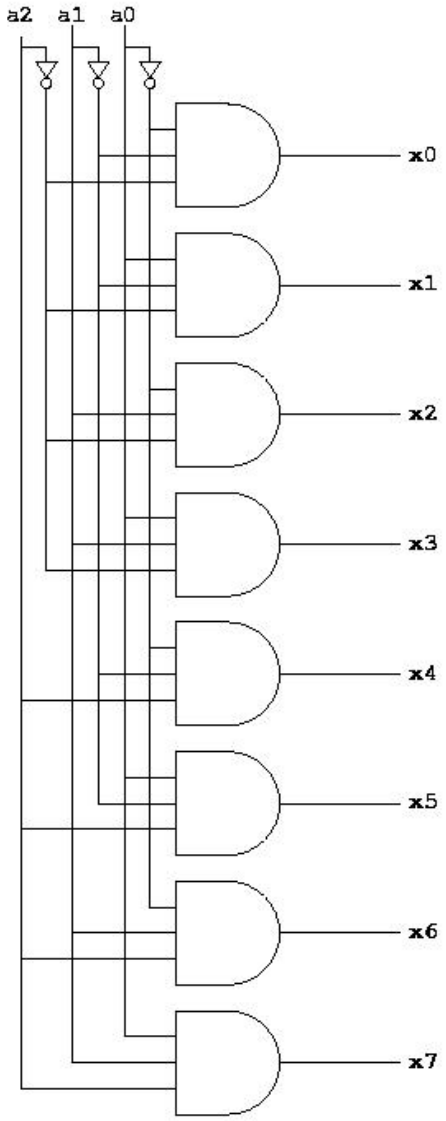
Decoder

Larger decoders require multi-input AND gates to be constructed.

This requires a lot of circuitry.

Can create deeper circuits with fewer transistors (i.e. less silicone), at the cost of slower response.

3-bit or ‘3-8’ decoder

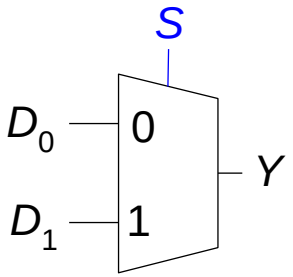


Beyond Simple Logic Gates - Mux (multiplexor)

Chooses 1 of many inputs to steer to its single output under the direction of control inputs (selector)

e.g., if the input to a circuit can come from several places a Mux is one way to funnel the multiple sources selectively to the single input.

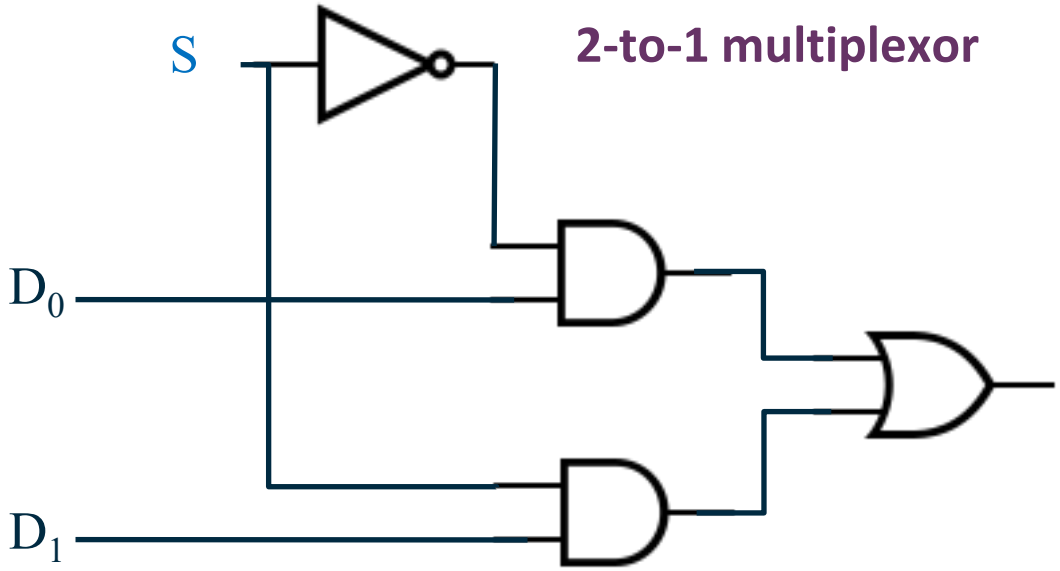
Beyond Simple Logic Gates - Mux (multiplexor)



S	D_1	D_0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

Compressed version of the truth table

S	Y
0	D_0
1	D_1



A multiplexor has **k+2^k inputs** and **1 output**.
The first k inputs (the selector *S*) represent a binary number.
The output takes the value of one of the remaining **2^k** inputs, the one indexed by the selector.

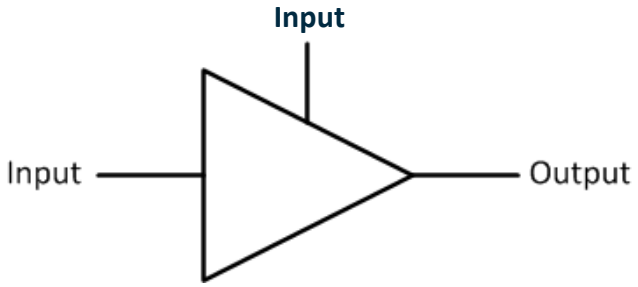
Durham University

Overview of today's lecture

- Adder
- Subtractor
- Decoder
- Tristates
- Mux
- Simple ALU

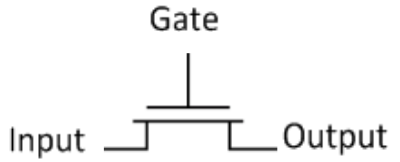
Tristate

In contrast to a normal buffer which is either 1 or 0 at its output, a tristate buffer can be electrically disconnected from the bus wire, i.e., it will have no effect on any other signals currently on the bus.



Input	Enable	Output
0	0	**Floating**
0	1	**0**
1	0	**Floating**
1	1	**1**

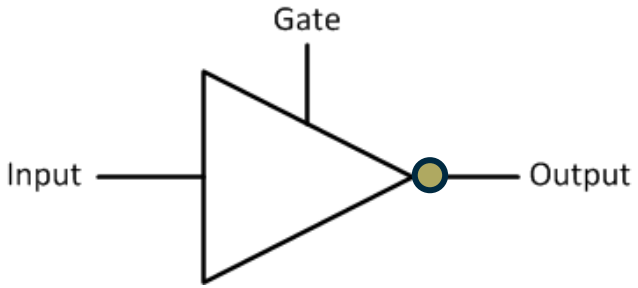
Linear truth table



A single transistor could work, but the output is not driven.

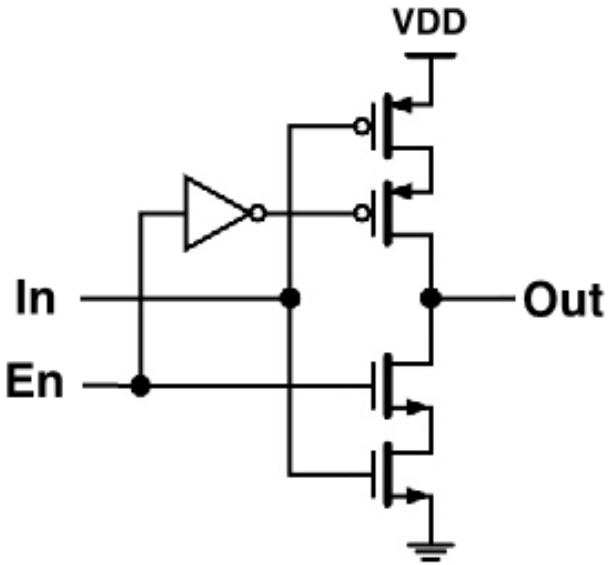
Issue when gate is ON and input is weak – the output will be weak as well

Inverting Tristate



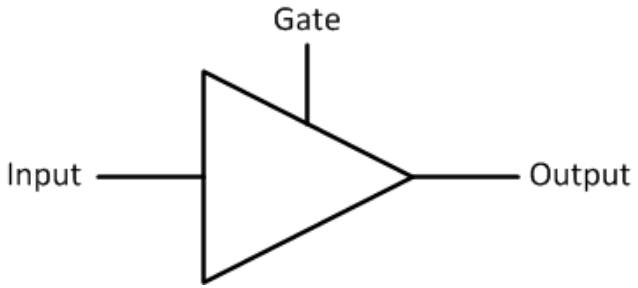
Input	Enable	Output
0	0	**Floating**
0	1	**1**
1	0	**Floating**
1	1	**0**

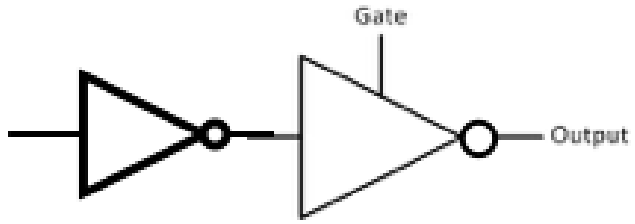
Linear truth table



Here the output is driven by direct connection to VDD or ground.

Tristate

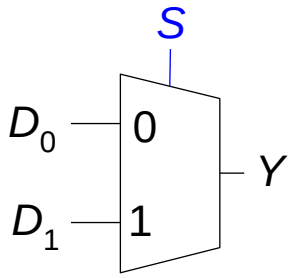




Input	Enable	Output
0	0	Floating
0	1	**0**
1	0	Floating
1	1	**1**

Linear truth table

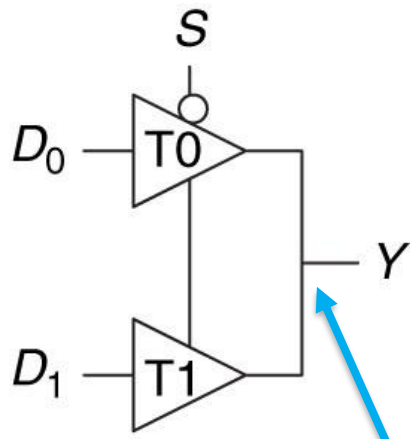
Mux (multiplexor)



S	D_1	D_0	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

S	Y
0	D_0
1	D_1

2-to-1 line multiplexor

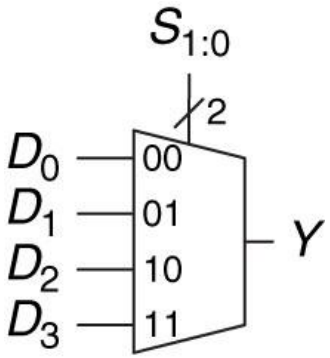


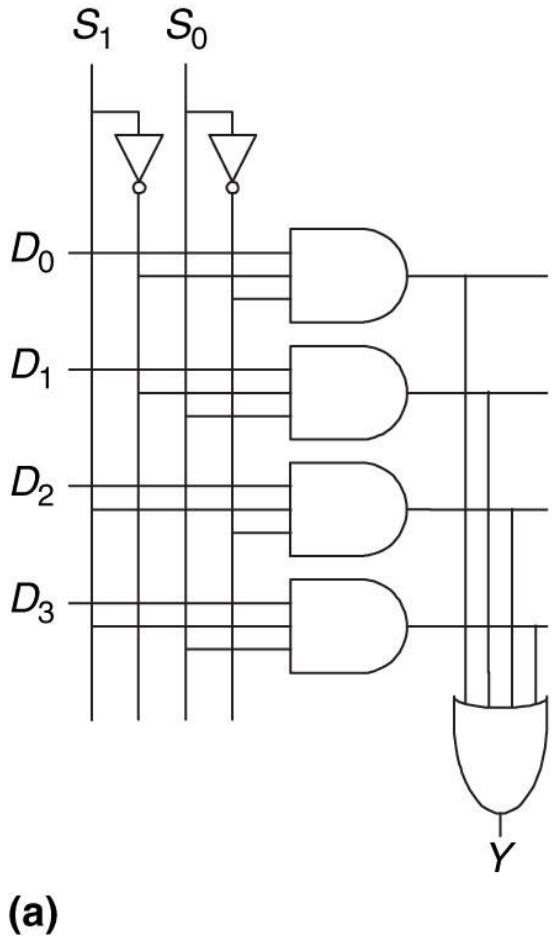
$$Y = D_0\overline{S} + D_1 S$$

It looks like Y is driven by two outputs which violates the combinational circuit rules, but this is not in fact the case; we have been very careful in how the selectors are arranged so that only one of $T0, T1$ is ever driving the value, and the other is floating.

Mux

4-to-1 line multiplexor





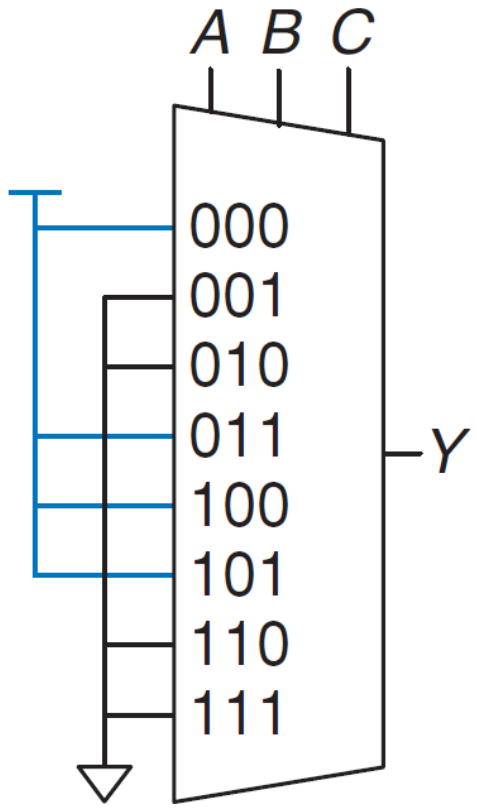
(a)

a) **two-level logic**, requiring multiple input gates.
b) Using **tri-state** buffers – take value of input if activated, otherwise floating value.
c) Using **hierarchical logic**

Mux (example 2.12 in Harris & Harris)

- A Mux can be used to implement combinational logic functions.
- An 8-input Mux can be used to implement functions in three variables. For example:

A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0



The control inputs A, B, C are used to select the minterms required at the output.

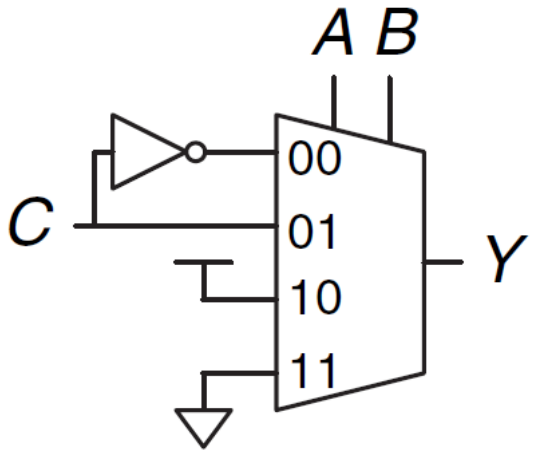
Mux (example 2.13 in Harris & Harris)

More economic designs than the one in the previous slide are possible.

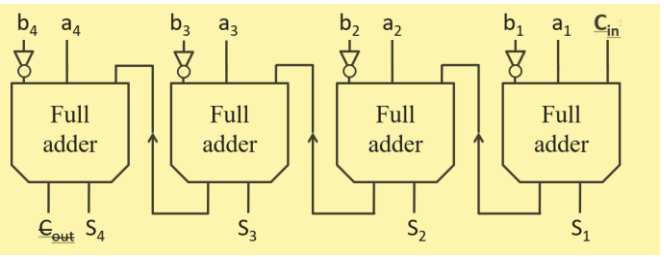
For example, reduce the truth table to 4 rows by letting the output depend on C.

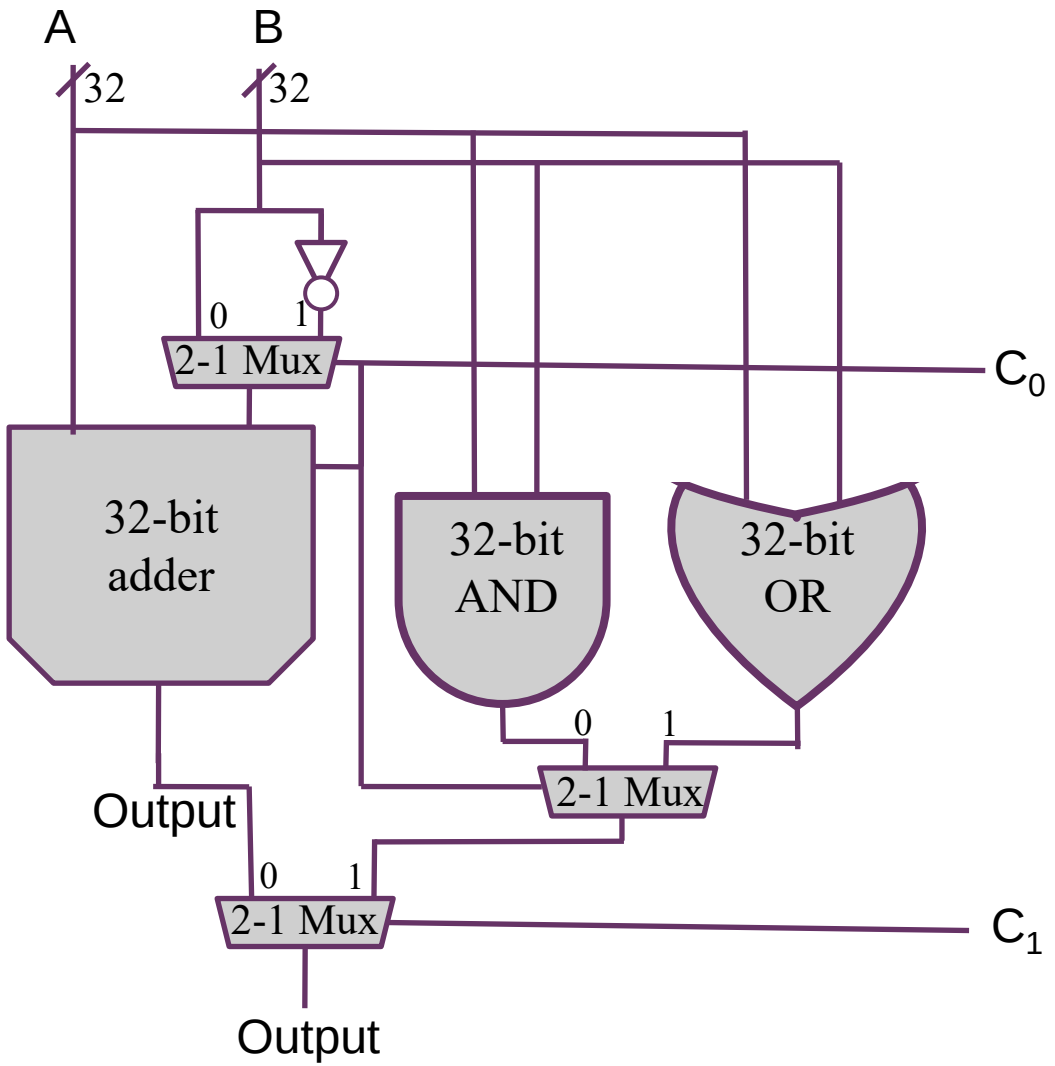
A	B	C	Y
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

A	B	Y
0	0	$\overline{C}$
0	1	C
1	0	1
1	1	0



Building a simple ALU (Arithmetic Logic Unit)





C_0	Fn
0	Add
1	Subtract

Summary

- Adder & Subtractor
- Decoder & Mux
- Tristates
- ALU