We acknowledge and pay our respects to the Kaurna people,
the traditional custodians whose ancestral lands we gather on.

We acknowledge the deep feelings of attachment and relationship of the Kaurna people to country and we respect and value their past, present and ongoing connection to the land and cultural beliefs.

Computer Systems

Lecture 04: Gates, Boolean Arithmetic, Sequential Logic

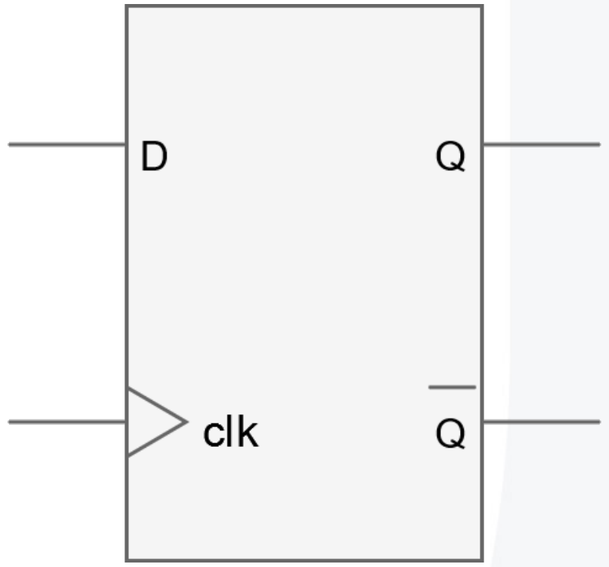
Review and Exercises

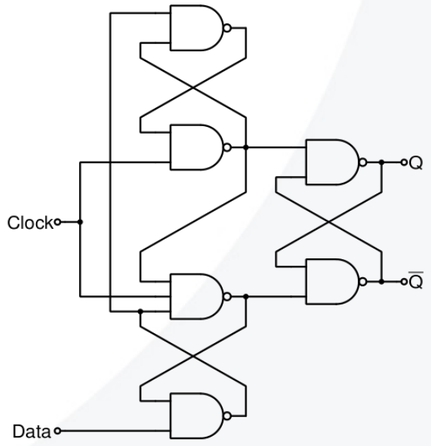
THE UNIVERSITY of ADELAIDE

DFFs

All sequential chips can be based on one low-level sequential gate, called a 'data flip flop', or DFF.

- DFF takes an input and holds/delays that until the next clock 'tick'.
- The output Q of the DFF is whatever the input D was at the previous clock 'tick'.
- DFFs can be made from NAND gates using some clever feedback loops. (we don't expect you to memorise this).

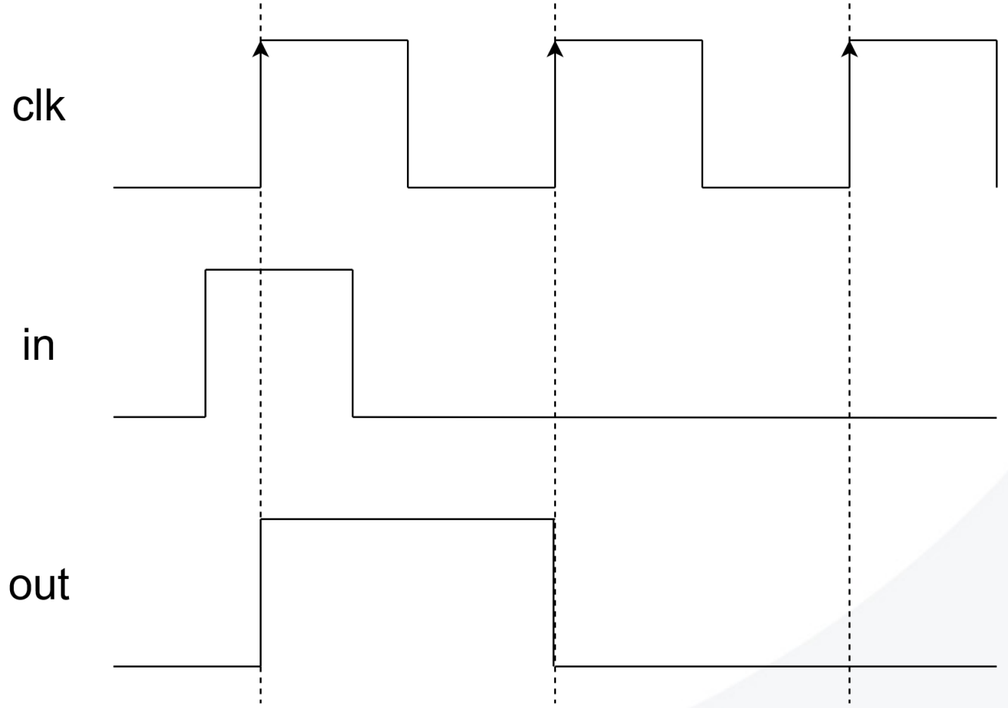




THE UNIVERSITY of ADELAIDE

Problem of using DFF to maintain state

If we using a DFF alone its state can only maintain for one clock cycle.

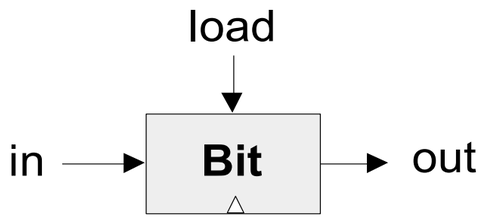


Let’s build a bit! (1-bit register)

We want to be able to:

Change the state of the bit

Retain that state until we want to change it.



if load(t-1) then out(t)=in(t-1)
else out(t)=out(t-1)

Now we can tell the bit when to change, by controlling the load signal.

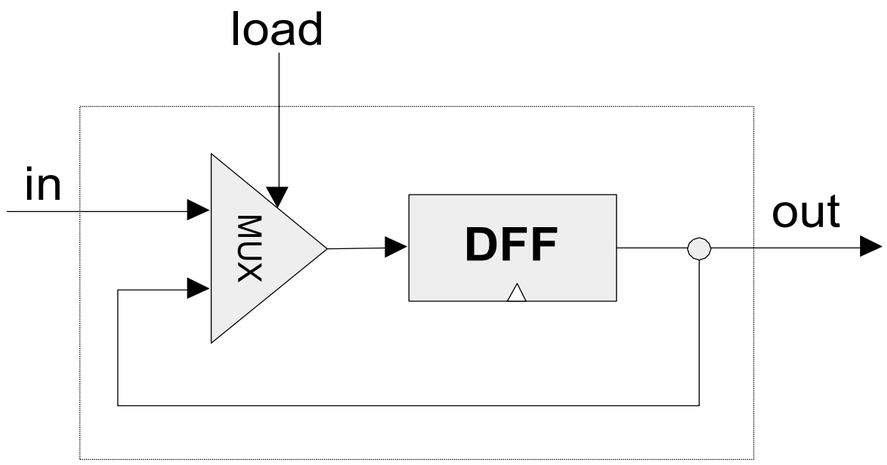
Can we build this from the DFF and gates we already know?

Bit (1-bit register) implementation

Notice how we use the load signal.

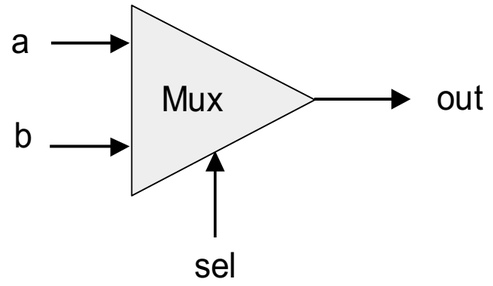
Now we have read logic and write logic.

Implementation



Canonical Form – Sum of Product (SoP)

a	b	sel	out
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



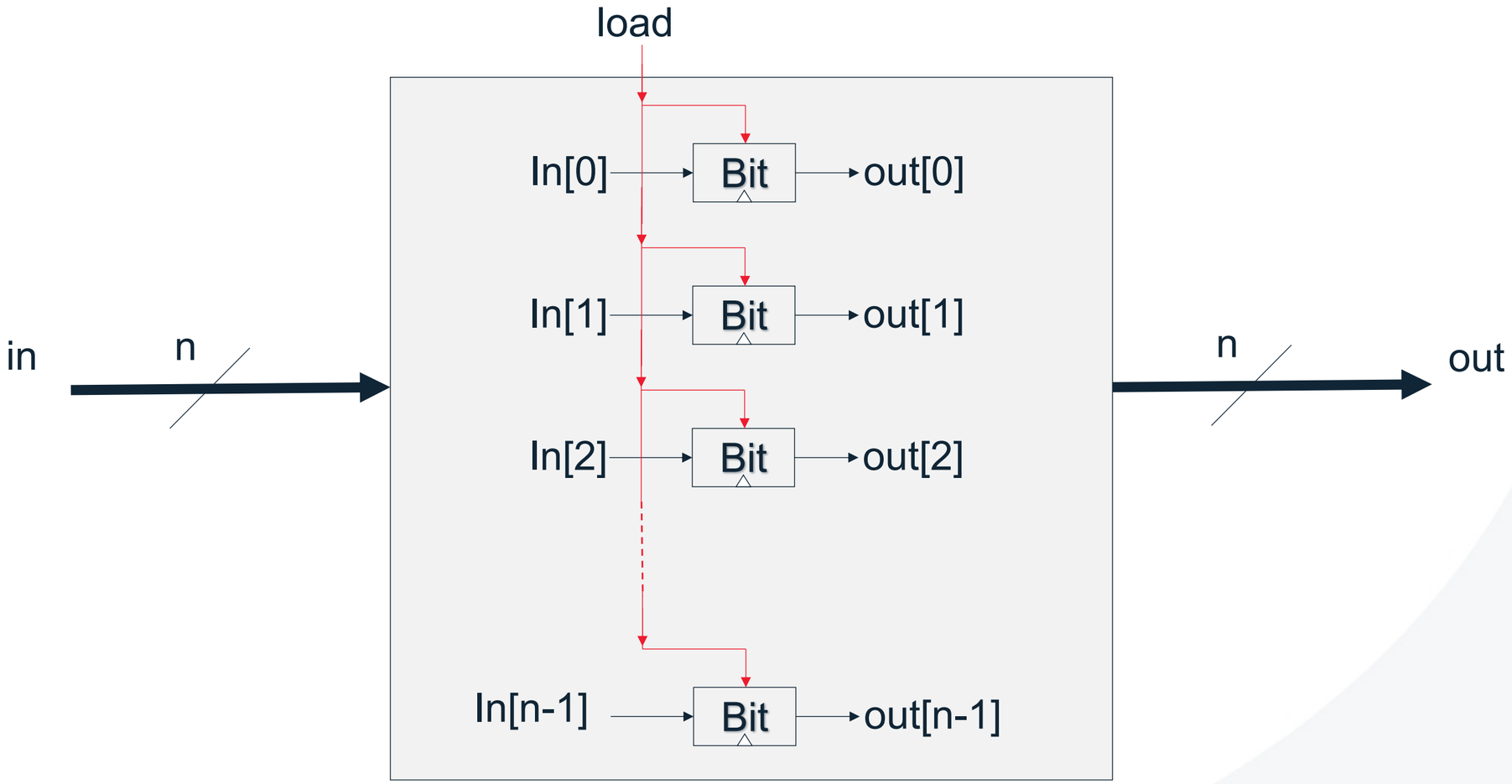
$$\mathbf{out} = (\bar{\mathbf{a}} \cdot \mathbf{b} \cdot \mathbf{sel}) + (\mathbf{a} \cdot \bar{\mathbf{b}} \cdot \overline{\mathbf{sel}}) + (\mathbf{a} \cdot \mathbf{b} \cdot \overline{\mathbf{sel}}) + (\mathbf{a} \cdot \mathbf{b} \cdot \mathbf{sel})$$

1. Find all lines with out = 1, ignore all lines with out = 0.
2. Write inputs in product (And), if the input is 1 write as it is, otherwise, write in bar (Not) form.
3. Repeat 2. for each line with out = 1.
4. Use sum (Or) to connect each term.

THE UNIVERSITY of ADELAIDE

Bigger registers

We can wire up n 1-bit registers in parallel to build a n-bit register.

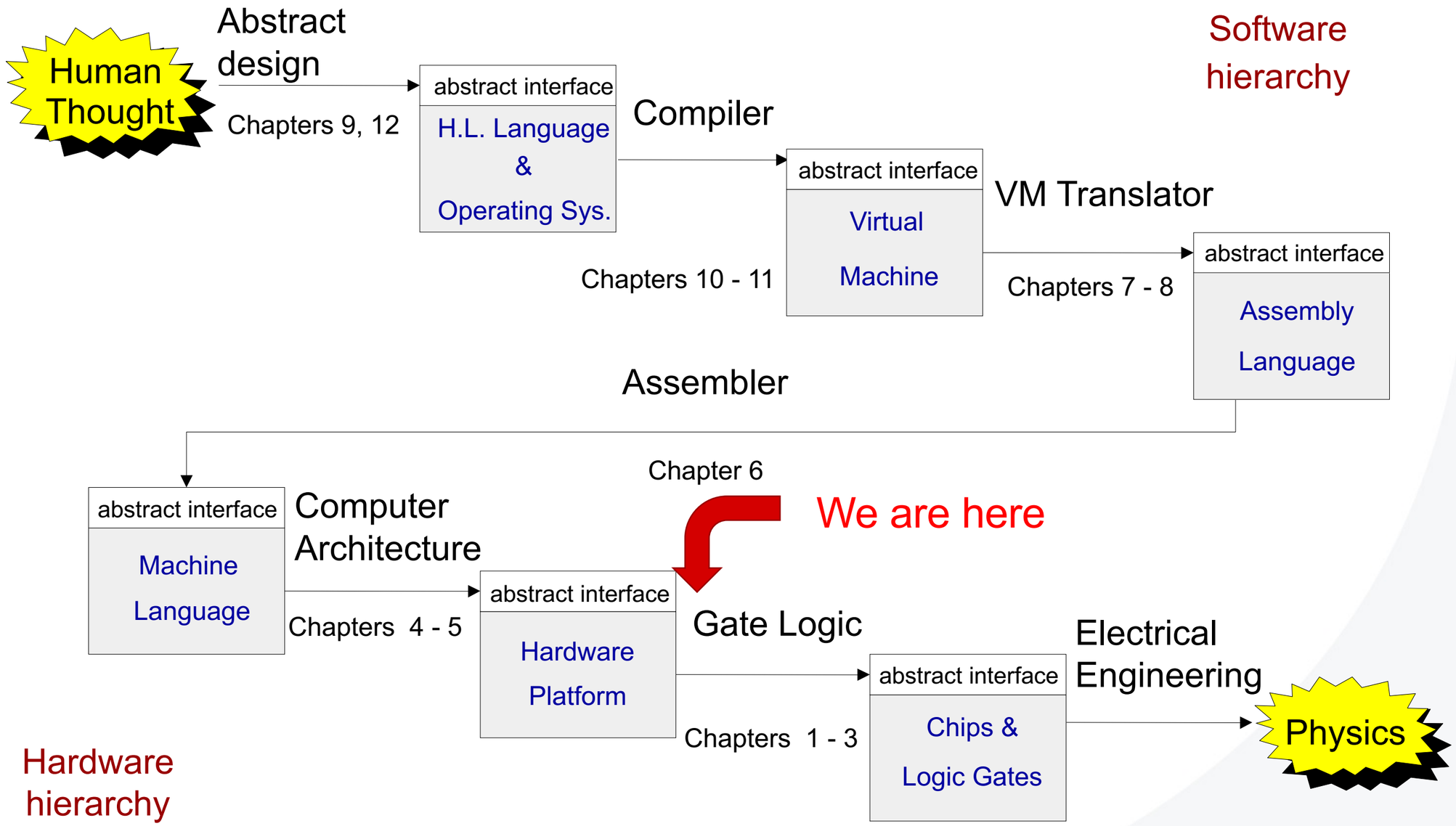


Summary

- Boolean Logic can be used to perform basic binary arithmetic such as addition.
- We can combine basic 1-bit adder chips to add larger numbers.
- Sequential Logic allows us to perform operations that rely on previous results and state.
- DFFs allow us to store state by delaying output. This allows us to build registers and other data storage structures.

THE UNIVERSITY of ADELAIDE

Our Journey

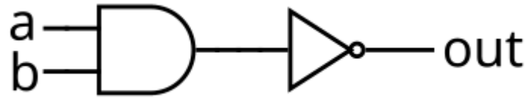


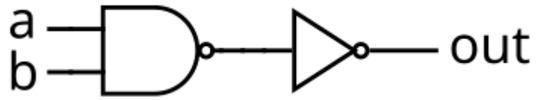
(Abstraction–implementation paradigm)

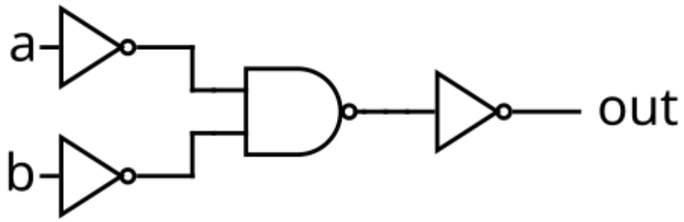
THE UNIVERSITY of ADELAIDE

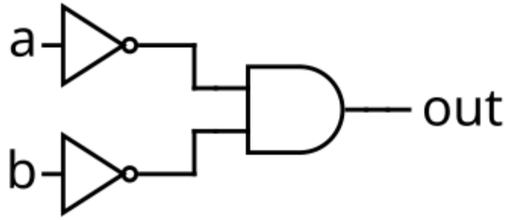
Quiz questions 1

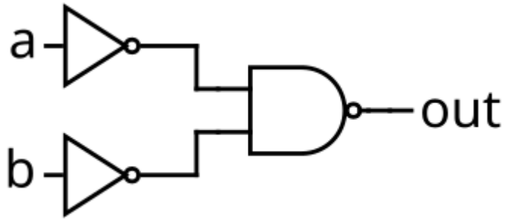
Which of the following logic circuits is equivalent to an OR gate?

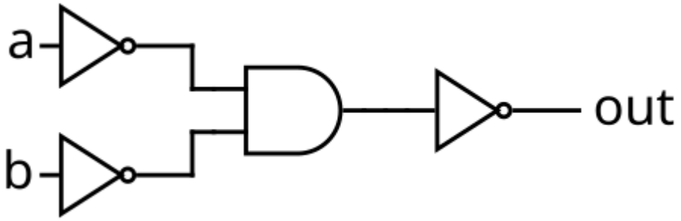












THE UNIVERSITY of ADELAIDE

In electrical and computer engineering, De Morgan's laws are commonly written as:

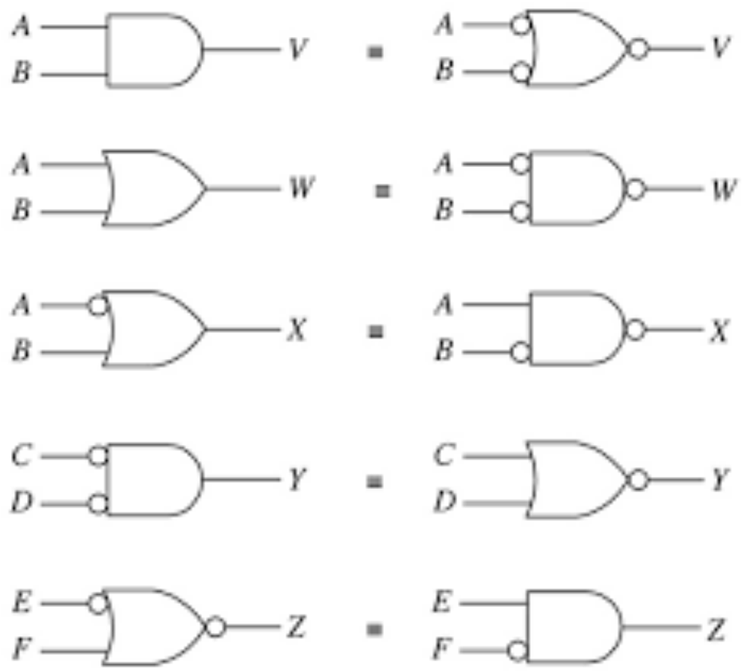
$$\overline{(A \cdot B)} \equiv (\overline{A} + \overline{B})$$

and

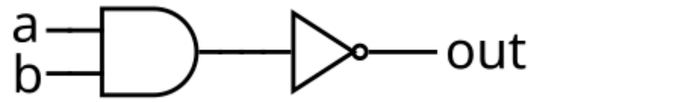
$$\overline{A + B} \equiv \overline{A} \cdot \overline{B},$$

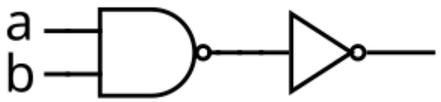
where:

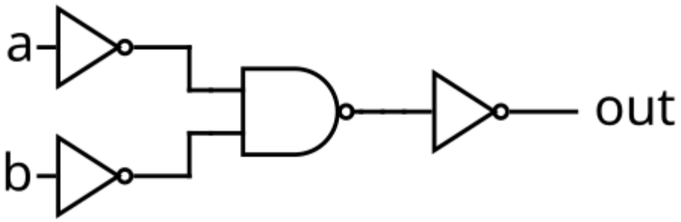
- $\cdot$ is the logical AND,
- $+$ is the logical OR,
- the $\overline{\text{overbar}}$ is the logical NOT of what is underneath the overbar.

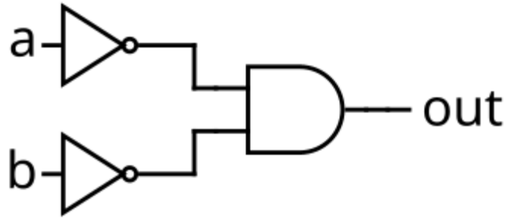


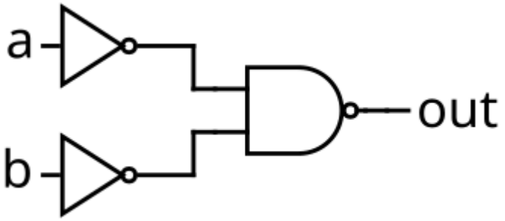
quivalent to an OR gate?

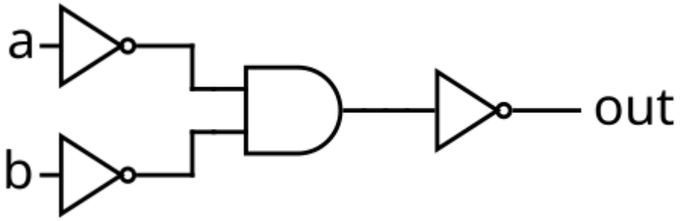












THE UNIVERSITY of ADELAIDE

Quiz questions 2

Why do we use two's complement to represent negative numbers in binary?

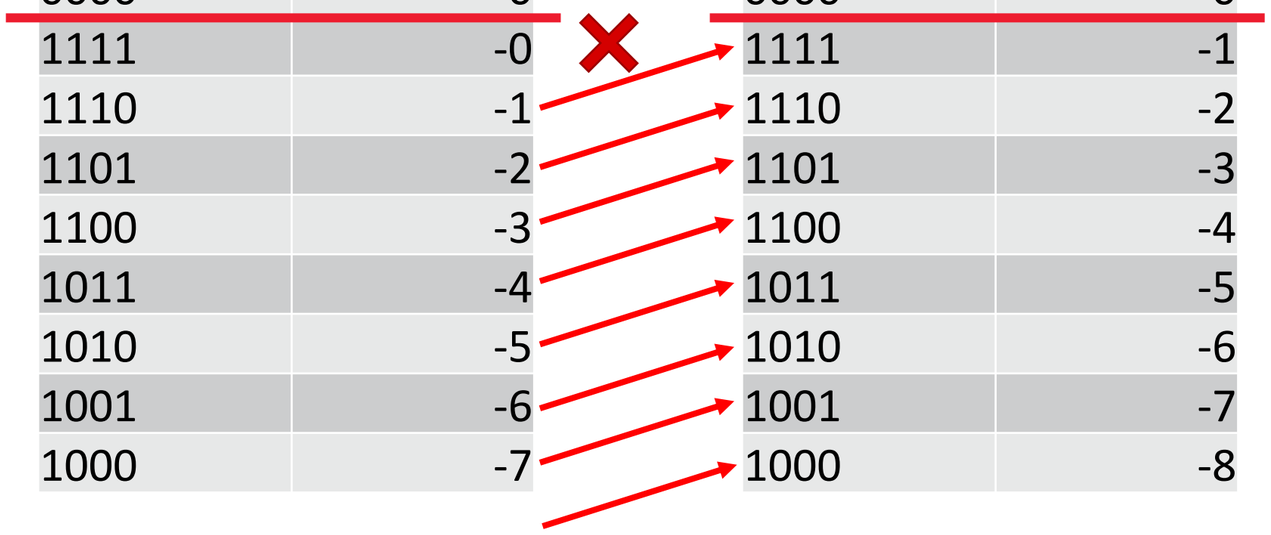
- ❑ **So that we can perform addition without worrying about the sign of the numbers**
- ❑ **So that there is only one representation of 0**
- ❑ **So that we can use the most significant bit as a sign bit**

THE UNIVERSITY of ADELAIDE

Number Representation: Two's Complement

One's Complete	Decimal
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-0
1110	-1
1101	-2
1100	-3
1011	-4
1010	-5
1001	-6
1000	-7

Two's Complete	Decimal
0111	7
0110	6
0101	5
0100	4
0011	3
0010	2
0001	1
0000	0
1111	-1
1110	-2
1101	-3
1100	-4
1011	-5
1010	-6
1001	-7
1000	-8



- An n-bit two's complement can represent 2^n numbers.
- From $-(2^{n-1})$ to $(2^{n-1}-1)$.
- Able to represent negative numbers.
- One representation of 0.

THE UNIVERSITY of ADELAIDE

Quiz questions 3

Consider the following diagram for a DMUX

sel	a	b
0	in	0
1	0	in

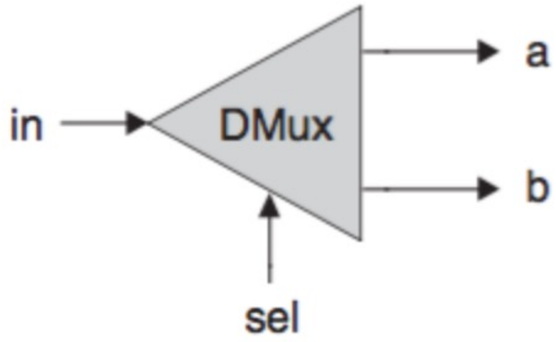


Figure 1.9 Demultiplexor.

where the unselected output wires are set to zero. Would a dmux still be a useful chip if these values were always set to 1? Explain your answer.

Quiz questions 3

Consider the following diagram for a DMUX， where the unselected output wires are set to zero. Would a DMux still be a useful chip if these values were always set to 1?

- No. It would render the DMux useless because all of the rest of the machine would be expecting a zero in this place rather than a one and there is nothing we could do to fix it.
- Yes. You might have to invert this signal or change the expected interpretation of this signal.
- No. The ones make the output of the DMux unpredictable.
- Yes. It doesn't matter what is on these inputs.

THE UNIVERSITY of ADELAIDE

Quiz questions 4

Which of the following statements about the HDL language used in this course are true.

- ❑ **HDL is a programming language**
- ❑ **HDL keywords are written in lowercase letters**
- ❑ **A chip definition consists of a header and a body. The header specifies the chip interface and the body its implementation.**
- ❑ **Names of chips and pins may be any sequence of letters and digits not starting with a digit.**

THE UNIVERSITY of ADELAIDE

Quiz questions 4

Which of the following statements about the H

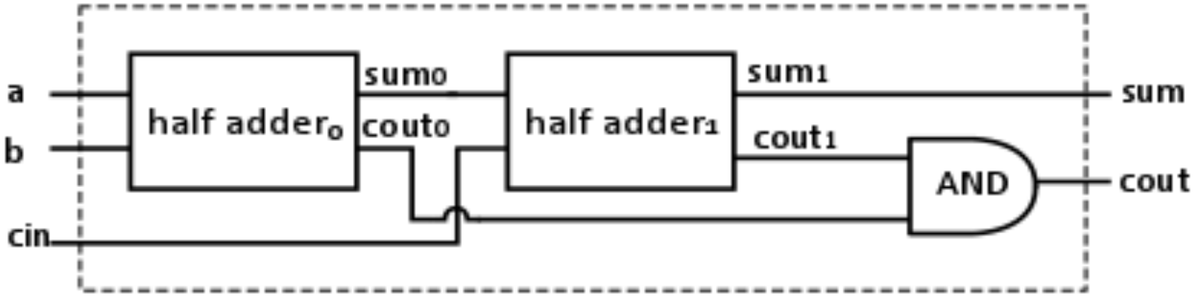
- ❑ **HDL is a programming language**
- ❑ **HDL keywords are written in lowerca**
- ❑ **A chip definition consists of a heade chip interface and the body its imple**
- ❑ **Names of chips and pins may be any starting with a digit.**

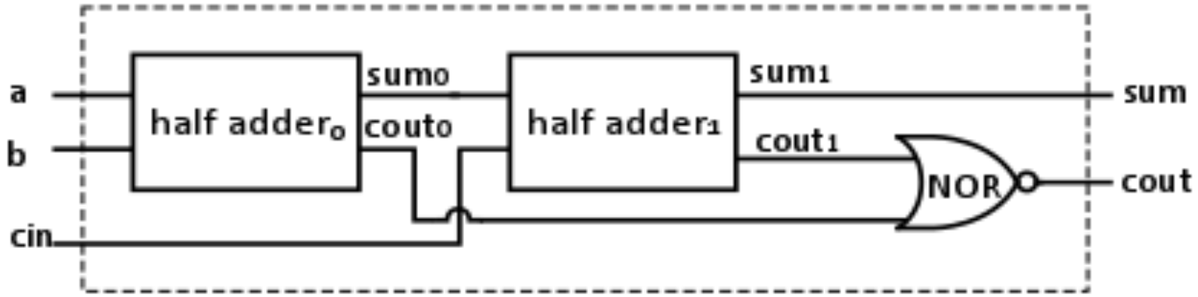
```
CHIP Xor {
    IN a, b;
    OUT out;

    PARTS:
    Not(in=a,out=na);
    Not(in=b,out=nb);
    And(a=na,b=b,out=c);
    And(a=a,b=nb,out=d);
    Or(a=c,b=d,out=out);

}
```

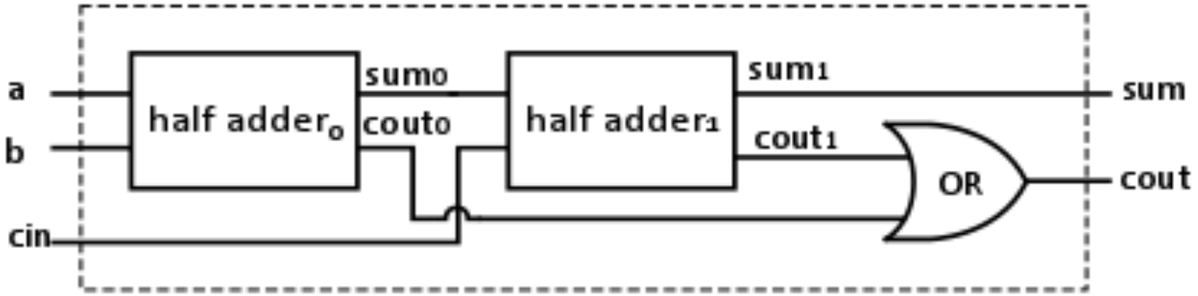
THE UNIVERSITY of ADELAIDE

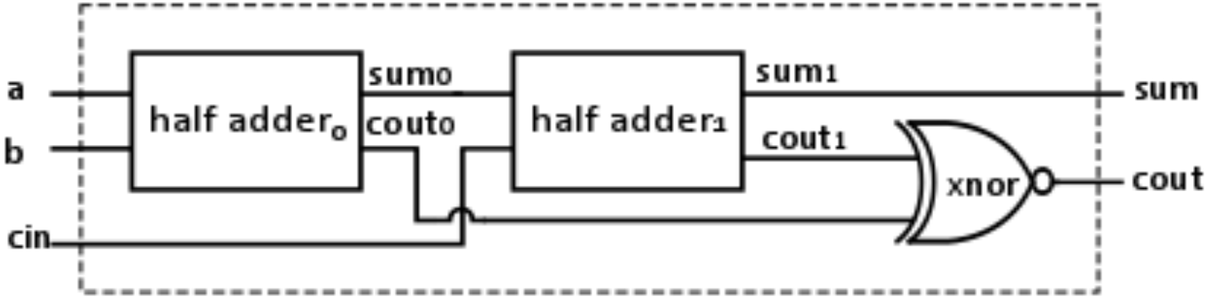
Quiz questions 5

Implementing a full adder using half adders.





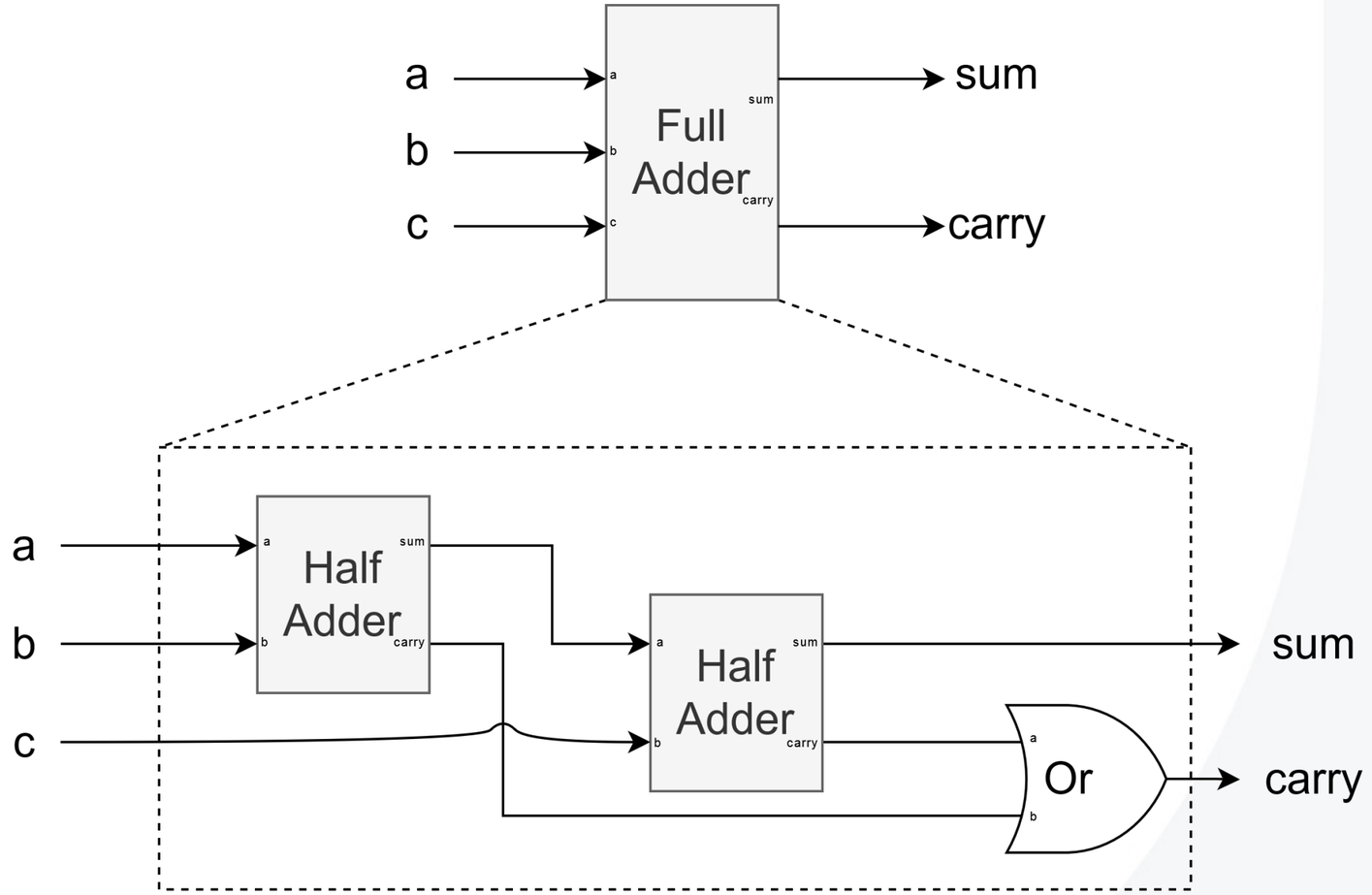




THE UNIVERSITY of ADELAIDE

Full adder (designed to add 3 bits)

a	b	c	sum	carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Implementation: can be based on <u>two half-adder and an Or gates</u>.

Exercise: 6-bit Two’s Complement 101010 to decimal

Exercise: 6-bit Two's Complement 101010 to decimal

digits	1	0	1	0	1	0
weights	-2^5	2^4	2^3	2^2	2^1	2^0

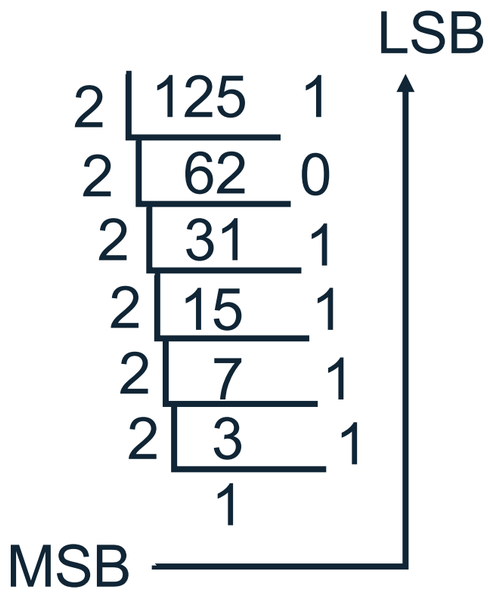
$$= 1 * (-2^5) + 0 * 2^4 + 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0$$
$$= -32 + 0 + 8 + 0 + 2 + 0$$
$$= -22$$

THE UNIVERSITY of ADELAIDE

Exercise: -125 to 8-bit Two’s complement

Exercise: -125 to 8-bit Two's complement

Consider 125 first



$125_{10} = 0111\ 1101_2$

$-125_{10} = 1000\ 0010_2 + 1_2$

$= 1000\ 0011_2$

Exercise: 4-bit Two’s Complement -5 + -5

Exercise: 4-bit Two's Complement -5 + -5

$5_{10} = 0101_2$

$-5_{10} = 1010_2 + 1_2$

$-5_{10} = 1011_2$

$$\begin{array}{r} 1\ 0\ 1\ 1 \\ +\ 1\ 0\ 1\ 1 \\ \hline 1\ 0\ 1\ 1\ 0 \end{array}$$

$0110_2 = 6_{10}$

Exercise: 16-bit Unsigned 32760 + 8

Exercise: 16-bit Unsigned 32760 + 8

Will this overflow?

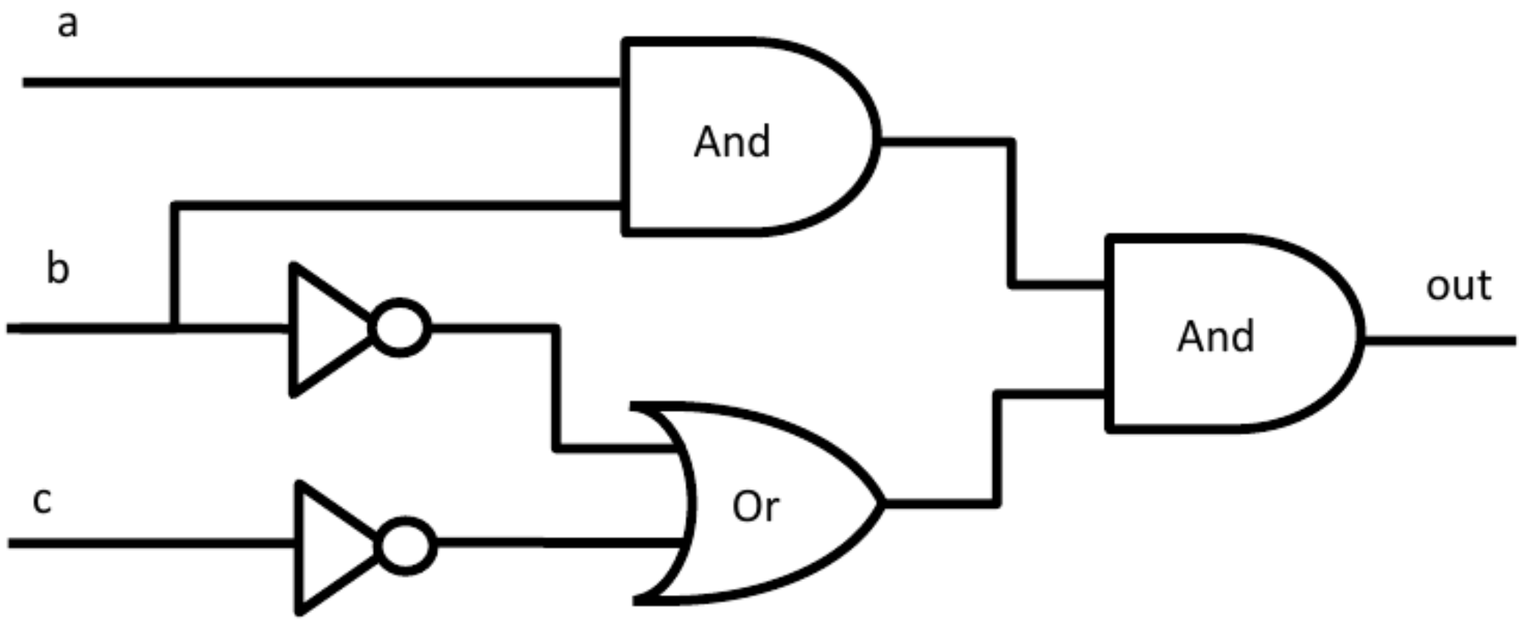
What is the largest 16-bit unsigned?

$$2^n - 1$$
$$= 2^{16} - 1$$
$$= 65536 > 32760$$

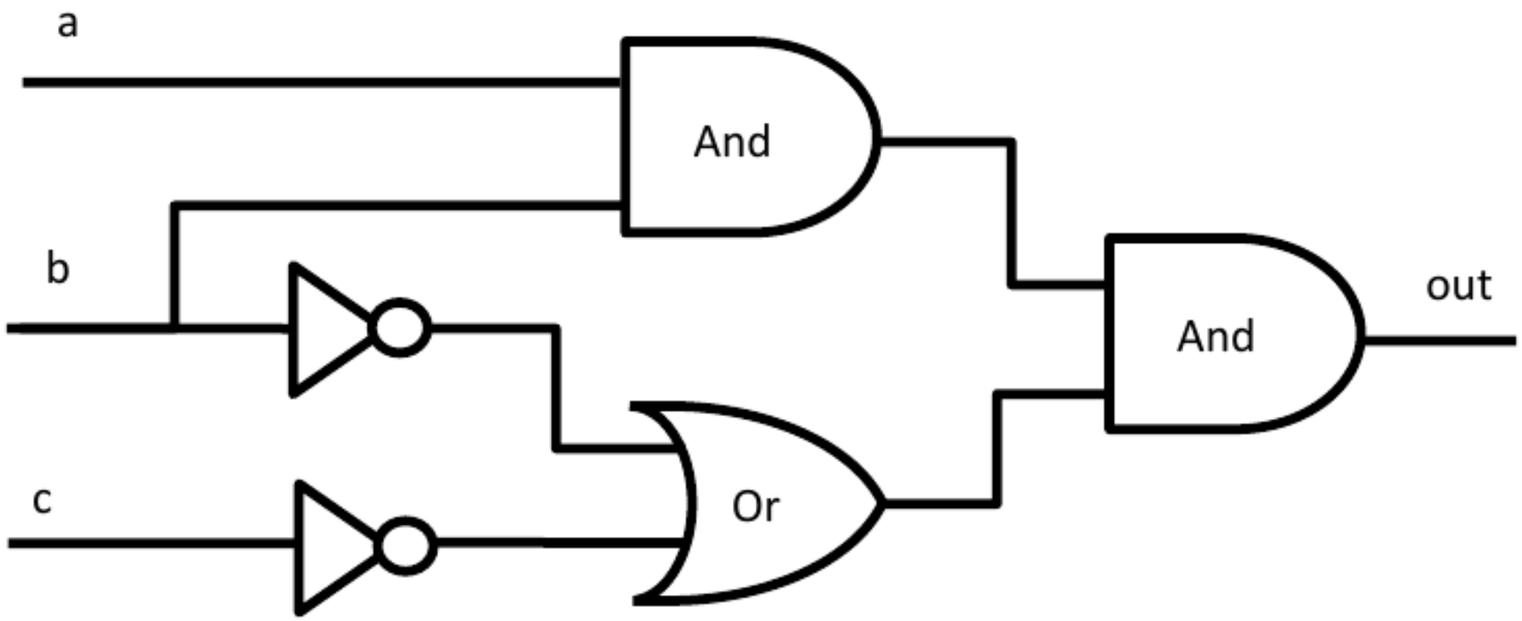
No overflow, so 32768 +8 = 32768.

THE UNIVERSITY of ADELAIDE

Exercise: Write logic equation and truth table based on circuit diagram



Exercise: Write logic equation and truth table based on circuit diagram



a	b	c	out
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

$$out = (a \cdot b) \cdot (\bar{b} + \bar{c})$$
$$= a \cdot b \cdot \bar{b} + a \cdot b \cdot \bar{c}$$
$$= 0 + a \cdot b \cdot \bar{c}$$
$$= a \cdot b \cdot \bar{c}$$

THE UNIVERSITY of ADELAIDE

Exercise: For a given truth table, write the logic equation and draw circuit diagram.

a	b	c	out
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

THE UNIVERSITY of ADELAIDE

This Week

- Review Chapters 2 & 3 of the Text Book (if you haven't already)
- Finish Assignment 1 (due this Sunday)
- Review Chapter 4 of the Text Book before next week.

THE UNIVERSITY of ADELAIDE