



F9.2 Logic circuits and expressions

Logic and computer systems · GCSE · OCR J277 2.4.1, AQA 8525 3.4.2,
Edexcel 1CP2 1.3.1 · about 15 min

Name _____

Class _____

Date _____

What this lesson is about

Combining gates, Boolean expressions, XOR, and the half adder.

- A circuit and its expression
- XOR
- Adding two bits
- From a problem to a circuit

Questions

6 marks in all

1. In $Q = (A \text{ AND } B) \text{ OR } (\text{NOT } C)$, $A = 0$, $B = 1$ and $C = 1$. What is Q ? [1 mark]

2. In $Q = (A \text{ AND } B) \text{ OR } (\text{NOT } C)$, $A = 0$, $B = 0$ and $C = 0$. What is Q ? [1 mark]

3. When does XOR output 1? [1 mark]

- ☐ A When the inputs are different
- ☐ B When both inputs are 1
- ☐ C When at least one input is 1
- ☐ D When both inputs are 0

4. In a half adder, which gate gives the carry bit? [1 mark]

- ☐ A AND
- ☐ B XOR
- ☐ C OR
- ☐ D NOT

5. What does this program print? [1 mark]

```
def XOR(a, b): return 1 if a != b else 0
for a, b in [(1, 1), (1, 0)]:
    print(a & b, XOR(a, b))
```

6. "Beep if the robot is close and moving, or it has bumped." Which expression matches? [1 mark]

- ☐ A (close AND moving) OR bumped
- ☐ B close AND (moving OR bumped)
- ☐ C (close OR moving) AND bumped
- ☐ D NOT close AND moving AND bumped

The task: a half adder

Write `AND(a, b)` and `XOR(a, b)` as functions of 0s and 1s. XOR must be built from AND, OR and NOT functions you also write, as `(A OR B) AND NOT (A AND B)`, not with `!=`. Print the half adder's truth table as four lines in the form `A=1 B=1 carry=1 sum=0`.

```
# the two lines every program starts with: the commands, then the robot
from bugbot import *
connect()

def AND(a, b): return 1 if a and b else 0
def OR(a, b): return 1 if a or b else 0
def NOT(a): return 1 - a
```

Plan your program here, then type it in and press Run.



Do it on the robot

www.bugbotlab.com/learn/f9-2-logic-circuits-and-expressions/

The simulator checks it and tells you when it passes. Nothing to install, no account.

Challenges

1. A full adder adds three bits: A, B and a carry in. Build one from two half adders and an OR gate, and print its eight-row truth table.
2. Write the expression for "the LED is green if the battery is OK and the way is clear, or the robot is parked", and print its truth table.
3. Show that `NOT (A AND B)` is always the same as `(NOT A) OR (NOT B)` by printing both columns.