1. Complete the truth table for a half adder

Input A	Input B	Sum (S)	Carry (C)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Explanation:
A Half Adder performs binary addition of two bits.

- Sum = A ⊕ B
- Carry = A · B

---

2. What is the main difference between a half adder and a full adder?

- **Half Adder:** Adds two input bits (A and B). It cannot handle carry input from a previous addition.
- **Full Adder:** Adds three bits — A, B, and a carry input (Cin) — and produces a Sum and Carry output.
    - Formula:
        - Sum = A ⊕ B ⊕ Cin
        - Carry = (A · B) + (Cin · (A ⊕ B))

A full adder includes the carry input; a half adder does not.

---

3. Using the full 4-bit adder in the slide, draw the full 2-bit adder. Then, add the following binary numbers:

Let's perform both additions step by step.

(a) 01 + 11

```
   0 1
+  1 1
------
  1 00
```

Result: 100 (3 + 1 = 4 in decimal)

(b) 10 + 11

```
   1 0
+  1 1
------
  1 01
```

Result: 101 (2 + 3 = 5 in decimal)

---

4. Using the full 4-bit subtractor, draw the logic circuit of a 2-bit subtractor. Label the inputs and outputs. Then compute:

Subtraction uses the formula:
A – B = A + (2's complement of B)
That means:

1. Invert each bit of B
2. Add 1 to the least significant bit (LSB)

(a) 10 – 01

```
  10
- 01
-----
  01
```

Result: 01 (2 – 1 = 1 in decimal)

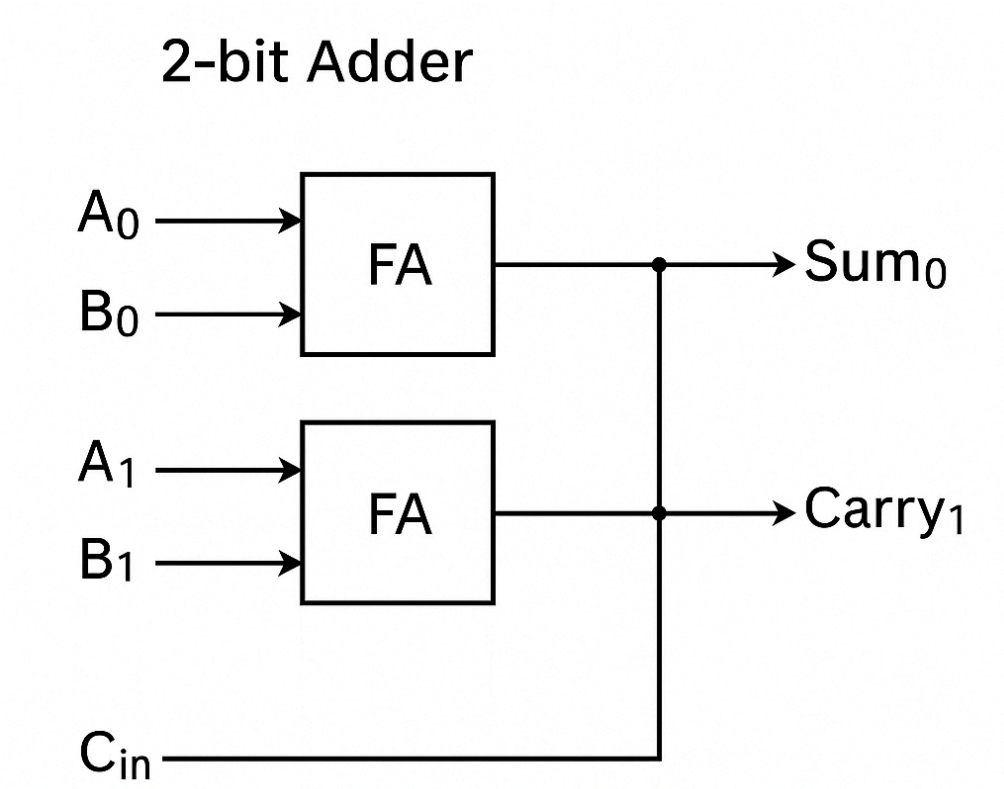
(b) 11 – 10

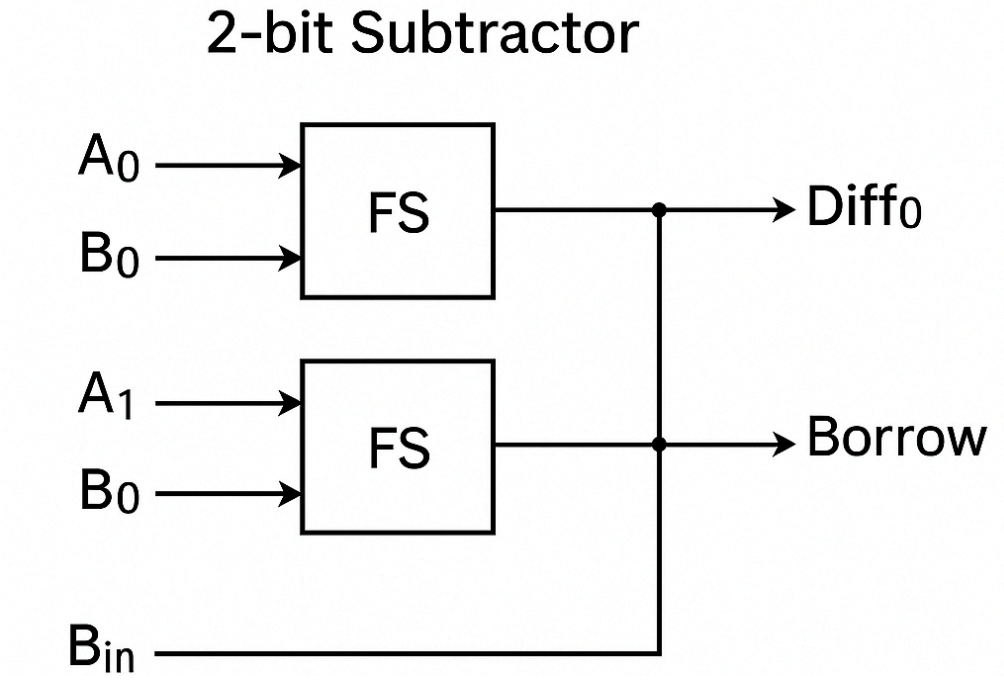
```
  11
- 10
-----
  01
```

Result: 01 (3 – 2 = 1 in decimal)

Operation	Result (Binary)	Result (Decimal)
01 + 11	100	4
10 + 11	101	5
10 – 01	01	1

Operation	Result (Binary)	Result (Decimal)
11 – 10	01	1





//
//
//

1️⃣ أكمل جدول الصواب (Truth Table) لـ Half Adder

A	B	Sum (S)	Carry (C)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

🔹 **الشرح:**
الـ **Half Adder** يجمع بتين (A وB).

- ناتج **Sum = A XOR B**
- ناتج **Carry = A AND B**

فمثلاً:

- إذا كانت A=1 وB=0 ⇒ Sum=1, Carry=0
- إذا كانت A=1 وB=1 ⇒ Sum=0, Carry=1

2️⃣ ما الفرق الرئيسي بين Half Adder وFull Adder؟

🔸 **Half Adder:**
يجمع (A وB)، ولا يأخذ بالحسبان أي حمل (Carry-in) من عملية سابقة. **بتين فقط**

المدخلات: A, B
المخرجات: Sum, Carry

🔸 **Full Adder:**
يجمع (A، B، وCarry-in من عملية سابقة). **3 بتات**

المدخلات: A, B, Cin
المخرجات: Sum, Carry-out

✅ **الفرق الرئيسي:**
الـ Full Adder (مثل 4-bit adder) يمكن استخدامه في عمليات الجمع المتسلسلة، بينما الـ Half Adder لا يمكنه التعامل مع الحمل من العمليات السابقة.

3️⃣ واحسب: 2 ارسم 2-bit adder، باستخدام 4-bit adder

أ) (01 + 11)

ب) (10 + 11)

يمكنك تمثيل **2-bit adder** من خلال توصيل **مجمّعين كاملين (Full Adders)**:

- أول FA لجمع البت الأقل (LSB)
- ثاني FA لجمع البت الأعلى (MSB) مع حمل من الأول

أ) (01 + 11)

```
   0 1
+  1 1
------
  1 00
```

✅ مع **00** = الناتج **Carry = 1** → **بالعشري 4) 100** (أي الناتج الفعلي هو).

ب) (10 + 11)

```
   1 0
+  1 1
------
  1 01
```

✅ مع **01** = الناتج **Carry = 1** → **بالعشري 5) 101** (الناتج الفعلي).

4️⃣ واحسب: 2 ارسم 2-bit subtractor، باستخدام 4-bit subtractor

أ) (10 – 01)

ب) (11 – 10)

نستخدم دائرة **Full Subtractor** في **الطرح الثنائي**
المعادلات الأساسية:

- **Difference = A XOR B XOR Bin**
- **Borrow = (¬A AND B) OR (Bin AND ¬(A XOR B))**

أ) (10 – 01)

```
  10
- 01
-----
  01
```

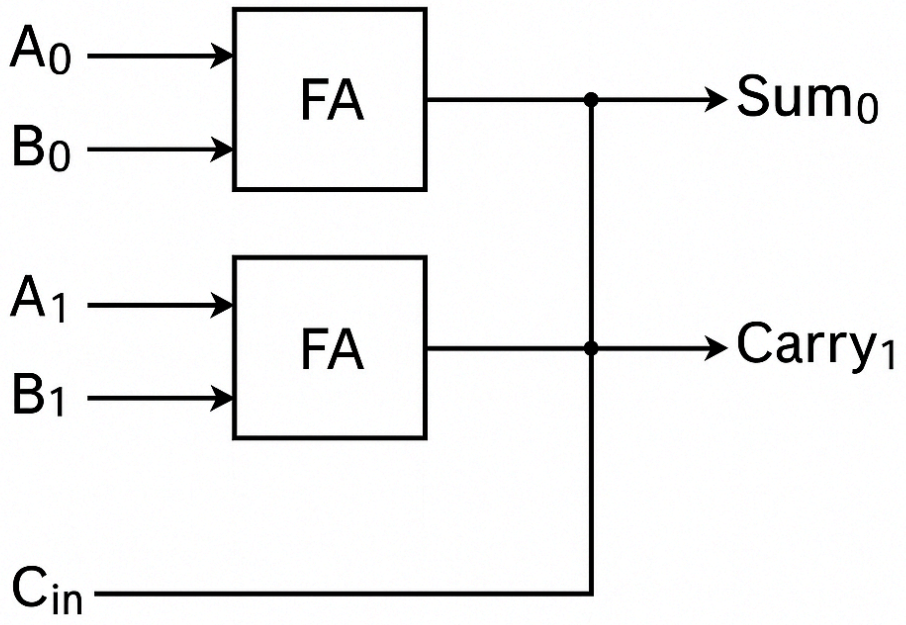
✅ بدون Borrow. الناتج = **01 (1 بالعشري)**

ب) 11 – 10 (10)

```
  11
- 10
-----
  01
```

✅ بدون Borrow. الناتج = **01 (1 بالعشري)**

2-bit Adder



2-bit Subtractor