CSARCH Lecture Series: Sequential Circuity Binary Multiplier

Sensei RL Uy
College of Computer Studies
De La Salle University
Manila, Philippines

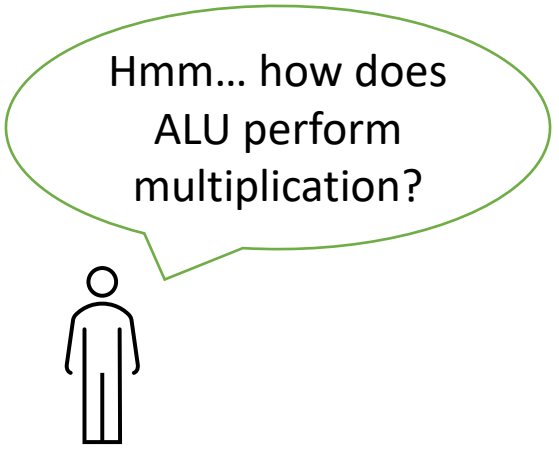
Copyright Notice

Overview

Reflect on the following question:

- How does Arithmetic and Logic Unit (ALU) perform multiplication?

```
int main()
{
    int var, var1, var2;
    var = 5;
    var1 = 2;
    var2 = var * var1;
}
```



Overview

- This sub-module describes how Arithmetic Logic Unit (ALU) performs multiplication
- The objective is as follows:
  - ✓Describe the process of performing sequential circuit binary multiplier

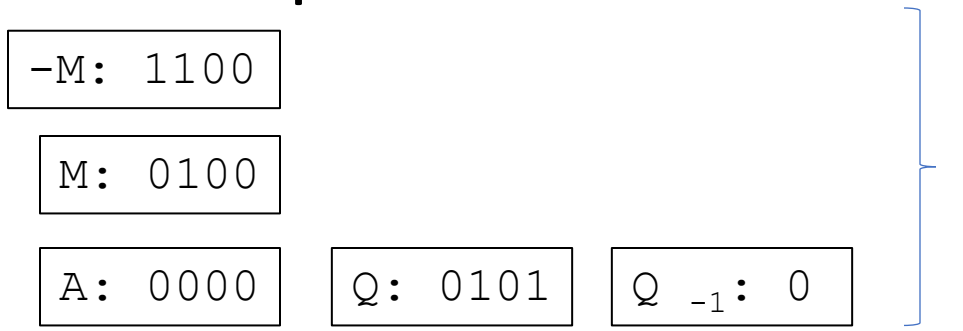
ALU Multiplication

- One of the method which Arithmetic and Logic Unit (ALU) uses to perform multiplication is called sequential circuit binary multiplier

Sequential Circuit Binary Multiplier

- Initialization
  - A ← 0, Q_{-1} ← 0
  - *M* gets multiplicand.
  - *Q* gets multiplier.
- Loop for each bit of multiplier
  - If Q_0 Q_{-1} = 01 then A ← A+M
  - else if Q_0 Q_{-1} = 10 then A ← A - M
  - Arithmetic Shift right A Q_0 Q_{-1}.
- Result contained in register combination *AQ*.

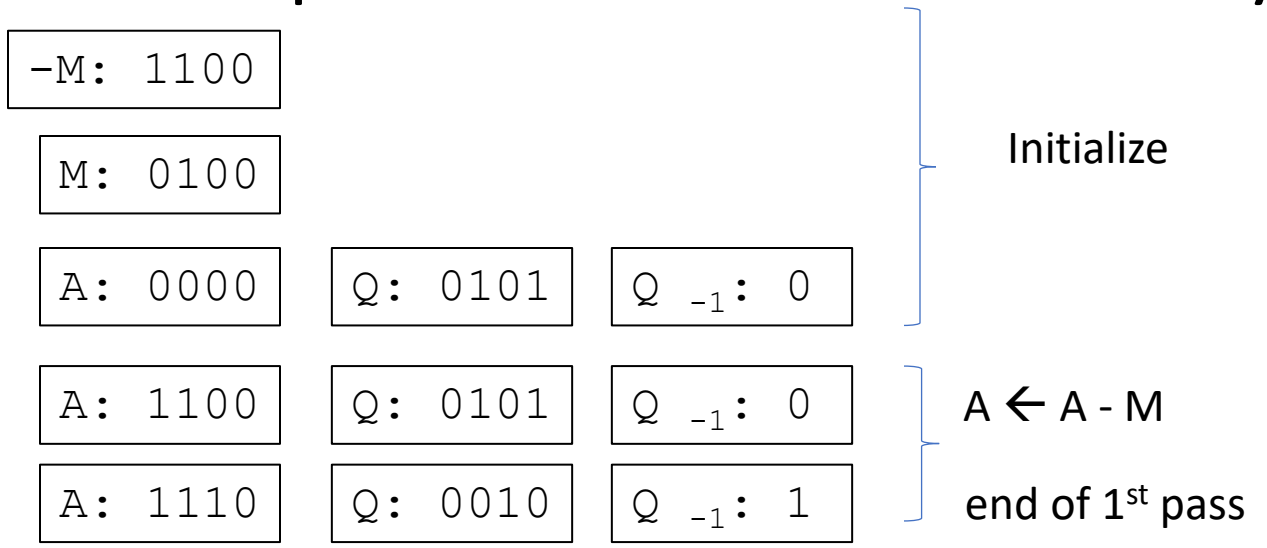
Sequential Circuit Binary Multiplier



0100 (M) * 0101 (Q)

- Initialization
  - A ← 0, Q_{-1} ← 0
  - *M* gets multiplicand.
  - *Q* gets multiplier.
- Loop for each bit of multiplier
  - If Q_0 Q_{-1} = 01 then A ← A+M
  - else if Q_0 Q_{-1} = 10 then A ← A - M
  - Arithmetic Shift right A Q_0 Q_{-1}.
- Result contained in register combination *AQ*.

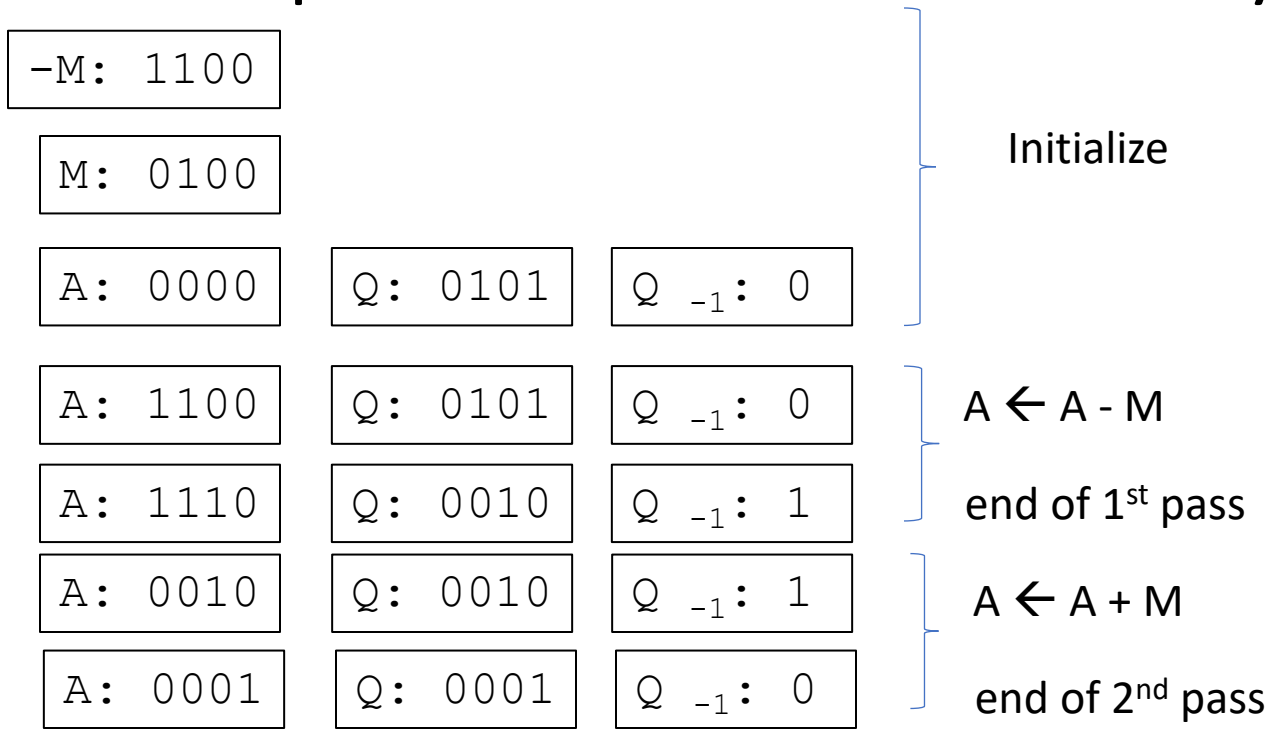
Sequential Circuit Binary Multiplier



0100 (M) * 0101 (Q)

- Initialization
  - A ← 0, Q_{-1} ← 0
  - *M* gets multiplicand.
  - *Q* gets multiplier.
- Loop for each bit of multiplier
  - If Q_0 Q_{-1} = 01 then A ← A+M
  - else if Q_0 Q_{-1} = 10 then A ← A - M
  - Arithmetic Shift right A Q_0 Q_{-1}.
- Result contained in register combination *AQ*.

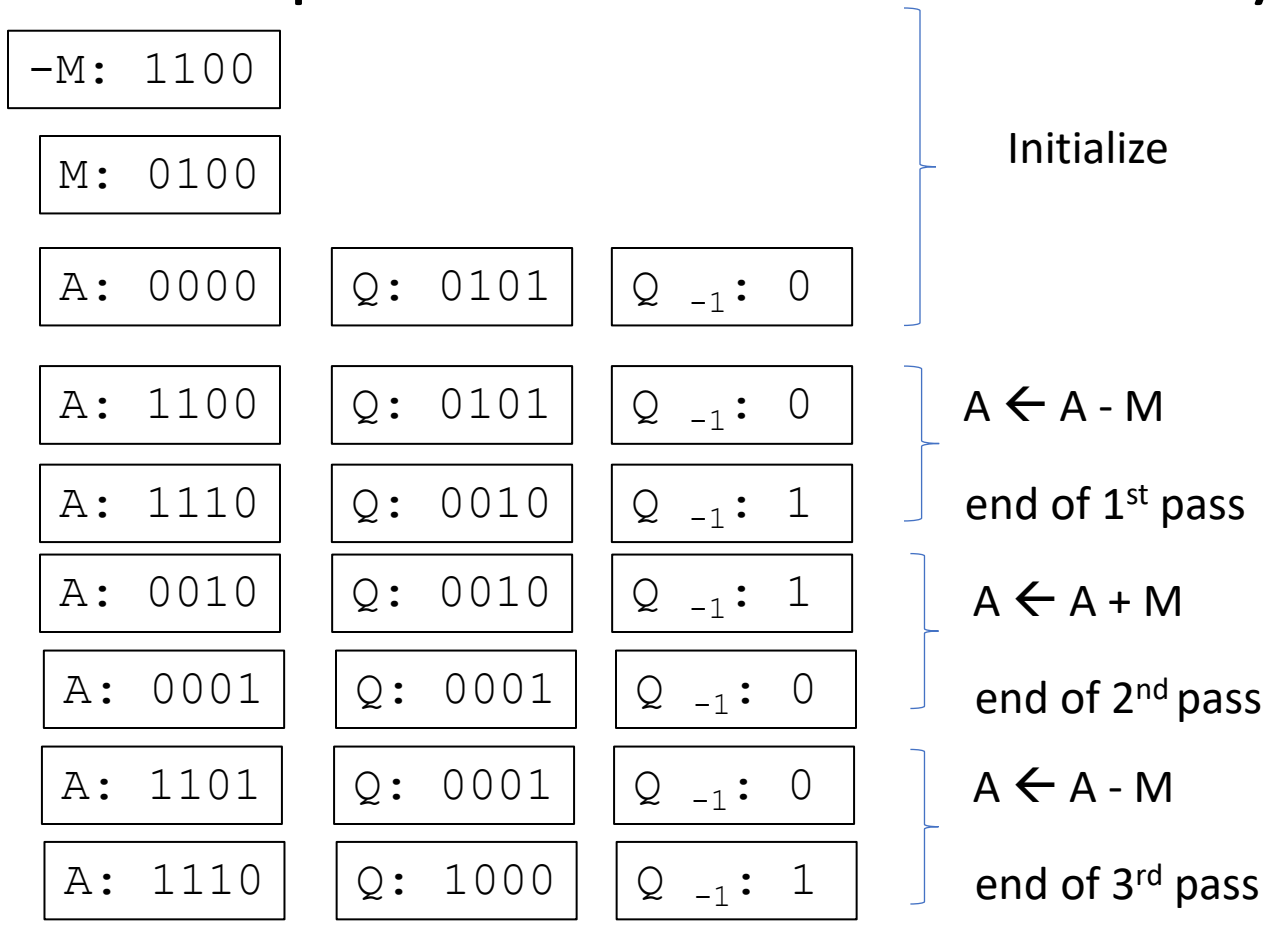
Sequential Circuit Binary Multiplier



0100 (M) * 0101 (Q)

- Initialization
  - A ← 0, Q_{-1} ← 0
  - *M* gets multiplicand.
  - *Q* gets multiplier.
- Loop for each bit of multiplier
  - If Q_0 Q_{-1} = 01 then A ← A+M
  - else if Q_0 Q_{-1} = 10 then A ← A - M
  - Arithmetic Shift right A Q_0 Q_{-1}.
- Result contained in register combination *AQ*.

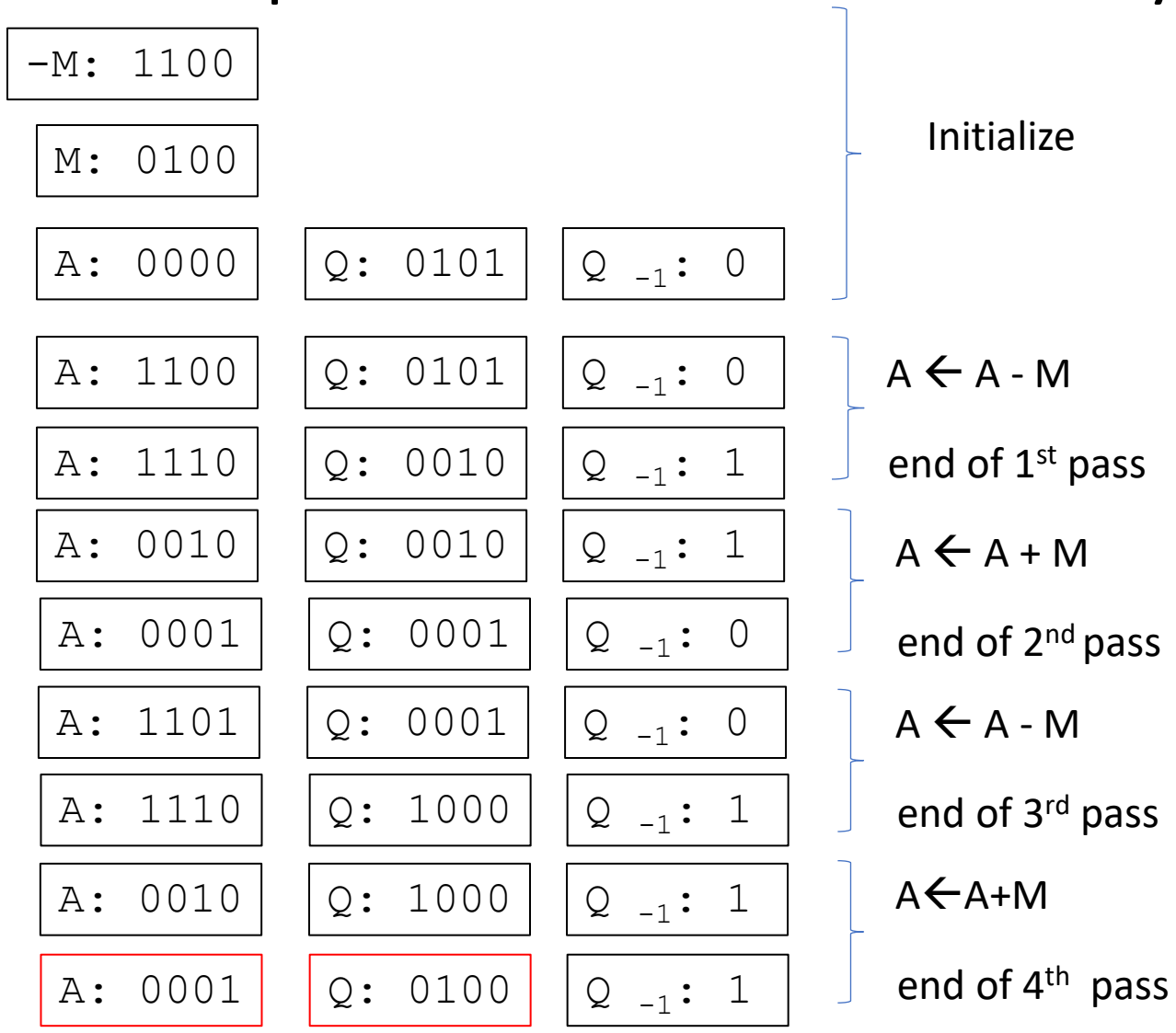
Sequential Circuit Binary Multiplier



0100 (M) * 0101 (Q)

- Initialization
  - A ← 0, Q_{-1} ← 0
  - *M* gets multiplicand.
  - *Q* gets multiplier.
- Loop for each bit of multiplier
  - If Q_0 Q_{-1} = 01 then A ← A+M
  - else if Q_0 Q_{-1} = 10 then A ← A - M
  - Arithmetic Shift right A Q_0 Q_{-1}.
- Result contained in register combination *AQ*.

Sequential Circuit Binary Multiplier



0100 (M) * 0101 (Q)

- Initialization
  - A ← 0, Q_{-1} ← 0
  - *M* gets multiplicand.
  - *Q* gets multiplier.
- Loop for each bit of multiplier
  - If Q_0 Q_{-1} = 01 then A ← A+M
  - else if Q_0 Q_{-1} = 10 then A ← A - M
  - Arithmetic Shift right A Q_0 Q_{-1}.
- Result contained in register combination *AQ*.

Try

Try: 11101 * 11010 (using sequential circuit binary multiplier)
Show the value of A and Q after the end of each pass

After this pass	A	Q
1st		
2nd		
3rd		
4th		
5th		

Try

Try: 11101 * 11010 (using sequential circuit binary multiplier)
Show the value of A and Q after the end of each pass

After this pass	A	Q
1st	00000	01101
2nd	11001	10110
3rd	00011	01011
4th	11011	00101
5th	11101	10011

To recall …

- What have we learned:
  - ✓Describe the process of performing sequential circuit binary multiplier