Combinational Logic Design

Dr. Chandan Karfa
CSE IIT Guwahati

- Adder/Subtractor
- Mano, Chapter 4

Full Adder/Subtractor

- *A* - *B* is done by taking the 2's complement of *B* and adding it to *A* .
  - $A - B = A + (2^n - B) = 2^n + (A-B)$
- 2's complement can be obtained by taking the 1's complement and adding 1.
  - The 1's complement can be implemented with inverters, and a 1 can be added to the sum through the input carry.
- Inverters placed between each data input *B* and the input carry $C_0 = 1$
  - XOR with 1 for each bits of B

Full Adder/Subtractor

Unsigned number:

- A ≥ B: 2^n + (A-B) → 2^n will be discarded.
- A < B: 2^n - (B-A) → 2's complement of B-A

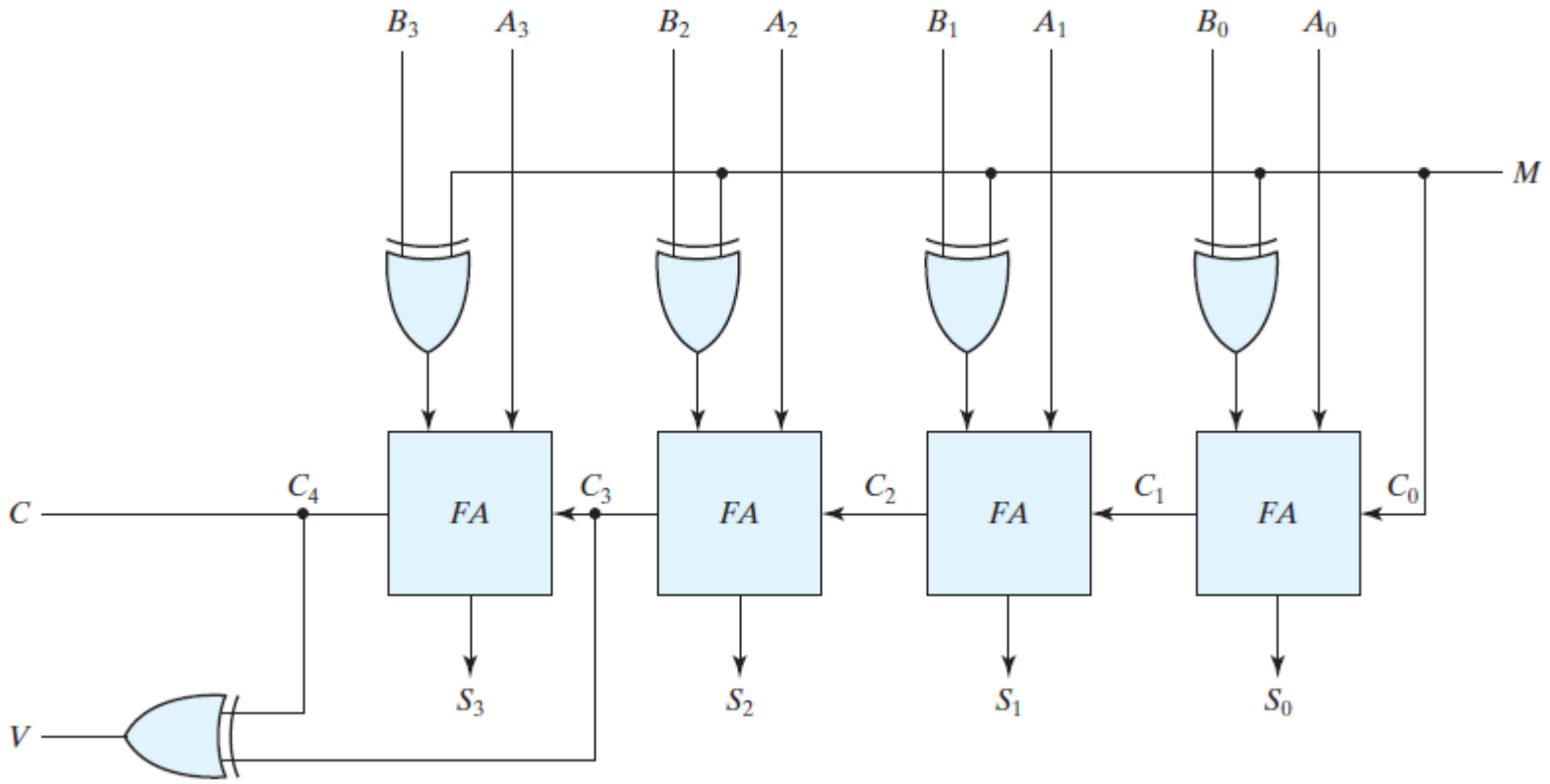
Signed number:

$$(\pm A) - (+B) = (\pm A) + (-B);$$
$$(\pm A) - (-B) = (\pm A) + (+B).$$

- Binary numbers in the signed-complement system are added and subtracted by the same basic addition and subtraction rules as are unsigned numbers.
- Therefore, computers need only one common hardware circuit to handle both types of arithmetic.

Full Adder/Subtractor

- M=1 → subtractor
- M=0 →adder



Overflow

- Two numbers with *n* digits each are added and the sum is a number occupying *n* + 1 digits
  - a result that contains *n* + 1 bits cannot be accommodated by an *n* -bit word.
  - Overflow occurs
- For unsigned number:
  - two positive numbers are added.
  - reflected by the carry out bits

Overflow for signed numbers

- the leftmost bit always represents the sign
- negative numbers are in 2’s-complement form.
- When two signed numbers are added, the sign bit is treated as part of the number.
- the end carry does not indicate an overflow.
- overflow cannot occur after an addition if one number is positive and the other is negative,

Overflow for signed numbers

- Addition of two 8 bits signed numbers. (range: -128 to 127)

carries:	0 1	carries:	1 0
+70	0 1000110	−70	1 0111010
+80	0 1010000	−80	1 0110000
+150	1 0010110	−150	0 1101010

- If the carry out of the sign bit position is taken as the sign bit of the result, then the nine-bit answer so obtained will be correct.
- since the answer cannot be accommodated within eight bits, we say that an overflow has occurred.

Overflow for signed numbers

- Overflow condition can be detected by observing the carry into the sign bit position and the carry out of the sign bit position.
- If these two carries are not equal, an overflow has occurred.
- If the two carries are applied to an exclusive-OR gate, an overflow is detected when the output of the gate is equal to 1

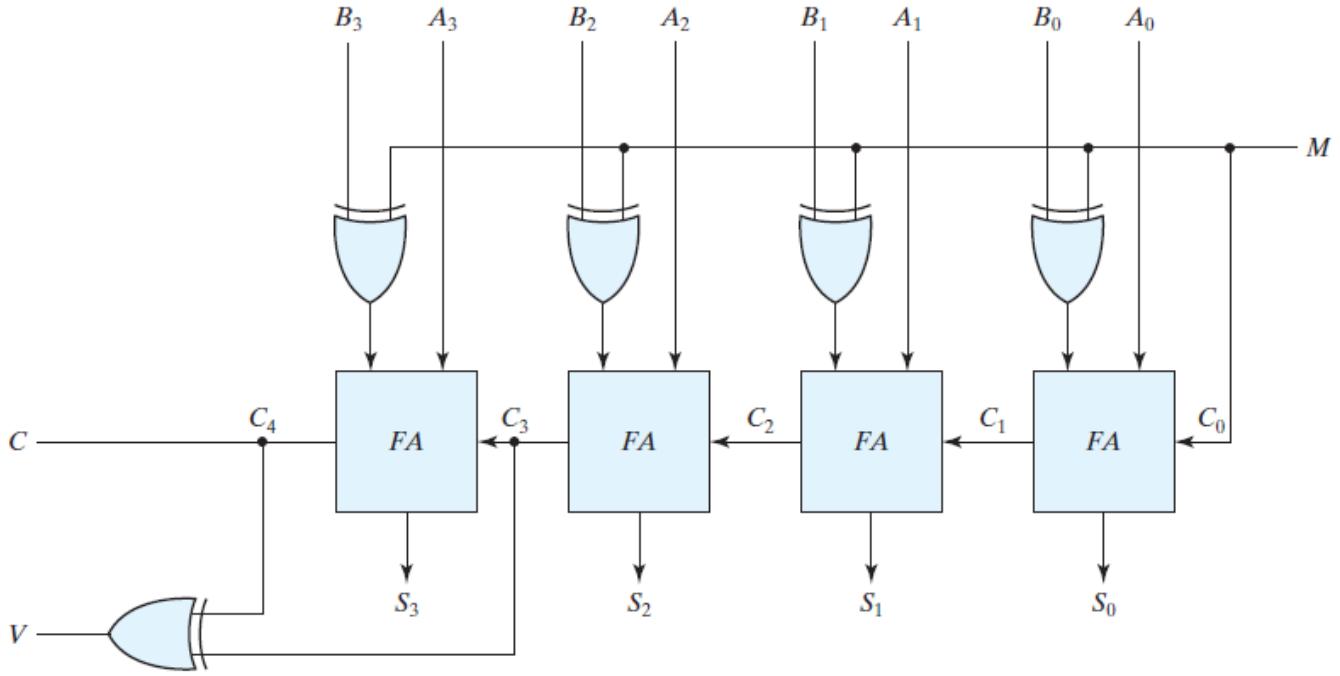
Overflow for signed numbers

Unsigned numbers:

- the *C* bit detects an overflow

Signed numbers:

- the *V* bit detects an overflow
- *V* = 0 → then no overflow
- *V* = 1 → overflow



Decimal adder

- Arithmetic operations directly in the decimal number system represent decimal numbers in binary coded form
- An adder for such a computer must employ arithmetic circuits that accept coded decimal numbers and present results in the same code.
- A decimal adder requires a minimum of nine inputs and five outputs, since four bits are required to code each decimal digit and the circuit must have an input and output carry
- Depends on the code used.

BCD adder

- Numbers range 0-9.
- Adder results: 0-19 (including input carry)

Derivation of BCD Adder

Binary Sum					BCD Sum					Decimal
K	Z_8	Z_4	Z_2	Z_1	C	S_8	S_4	S_2	S_1	
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	2
0	0	0	1	1	0	0	0	1	1	3
0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	1	0	0	1	0	1	5
0	0	1	1	0	0	0	1	1	0	6
0	0	1	1	1	0	0	1	1	1	7
0	1	0	0	0	0	1	0	0	0	8
0	1	0	0	1	0	1	0	0	1	9
0	1	0	1	0	1	0	0	0	0	10
0	1	0	1	1	1	0	0	0	1	11
0	1	1	0	0	1	0	0	1	0	12
0	1	1	0	1	1	0	0	1	1	13
0	1	1	1	0	1	0	1	0	0	14
0	1	1	1	1	1	0	1	0	1	15
1	0	0	0	0	1	0	1	1	0	16
1	0	0	0	1	1	0	1	1	1	17
1	0	0	1	0	1	1	0	0	0	18
1	0	0	1	1	1	1	0	0	1	19

When the binary sum is equal to or less than 1001, the corresponding BCD number is identical, and therefore no conversion is needed.
When the binary sum is greater than 1001, the addition of binary 6 (0110) to the binary sum converts it to the correct BCD representation and also produces an output carry as required

BCD Adder

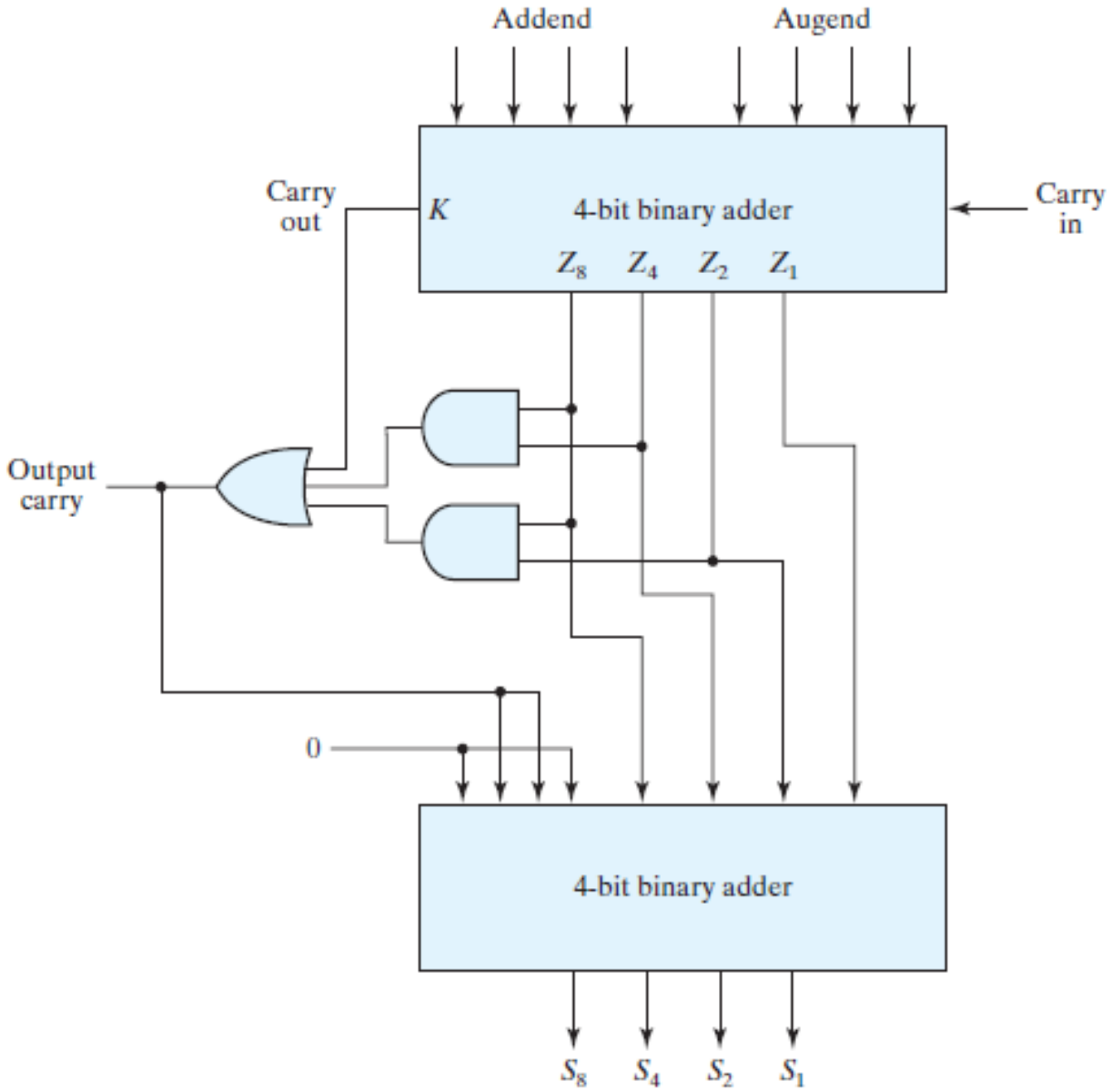
Corrections needed when

- $K = 1$.
- The other six combinations from 1010 through 1111 that need
- when 1 in position $Z8$.
  - To distinguish them from binary 1000 and 1001, which also have a 1 in position $Z8$, either $Z4$ or $Z2$ must have a 1.

$$C = K + Z_8Z_4 + Z_8Z_2$$

BCD Adder

$$C = K + Z_8Z_4 + Z_8Z_2$$



Binary Multiplier

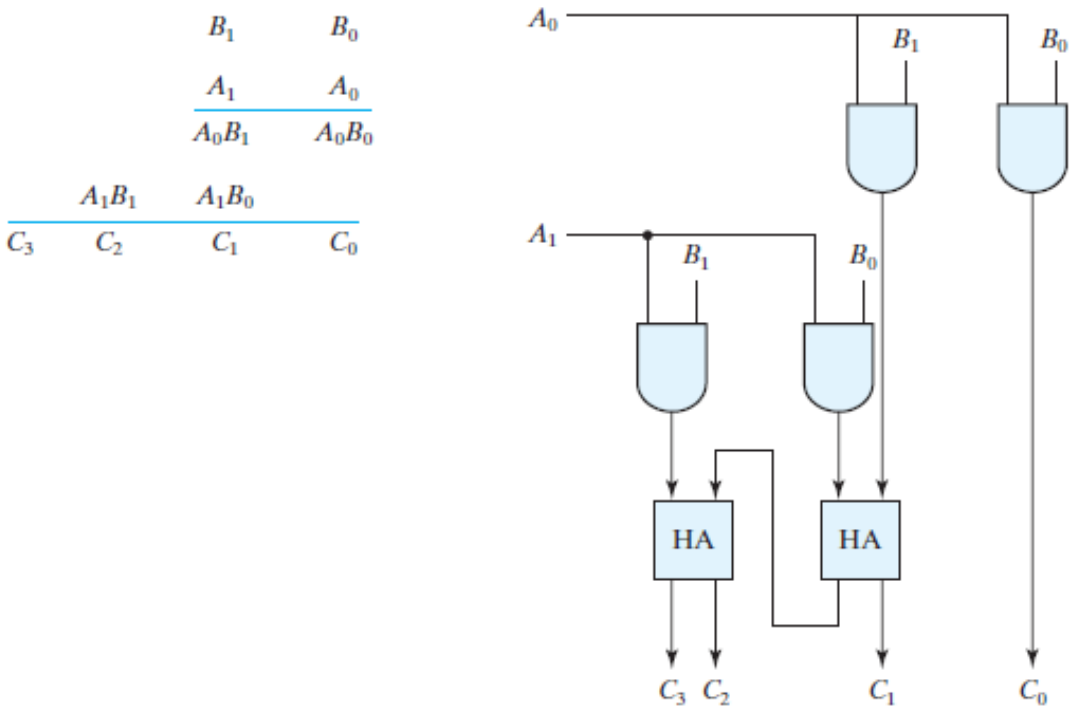


FIGURE 4.15
Two-bit by two-bit binary multiplier

- A bit of the multiplier is ANDed with each bit of the multiplicand in as many levels as there are bits in the multiplier.
- The binary output in each level of AND gates is added with the partial product of the previous level to form a new partial product.
- The last level produces the product.
- For J multiplier bits and K multiplicand bits, we need $J * K$2AND gates and $(J - 1) * K$ bit adders to produce a product of $(J + K)$ bits.

Binary Multiplier