CSM151B
Computer Systems Architecture

Week 3 Discussion

1/26/2018

Logistics

- HW2 due today
- HW3 due next Friday

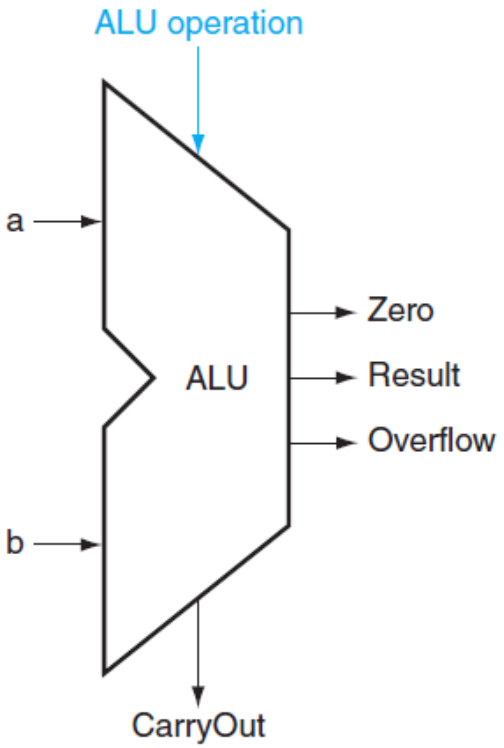
- HW1 solution has been uploaded on CCLE

Agenda

- ALU design
- Delay analysis

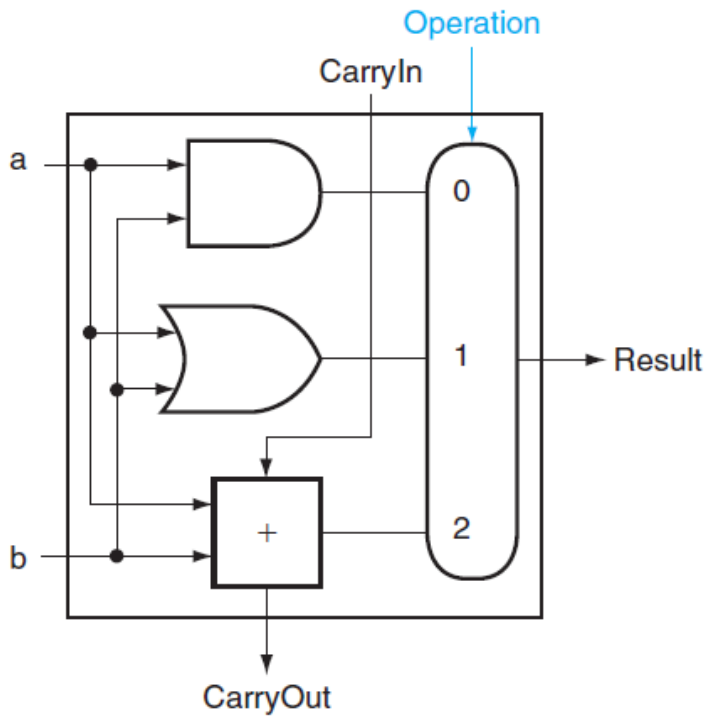
ALU Design

- Arithmetic logic unit (ALU) design
  - Input: two operand (32-bit), ALU control signals
  - Output: result, zero, overflow, carryout
  - Operation: add, subtract, and, or, nor, slt



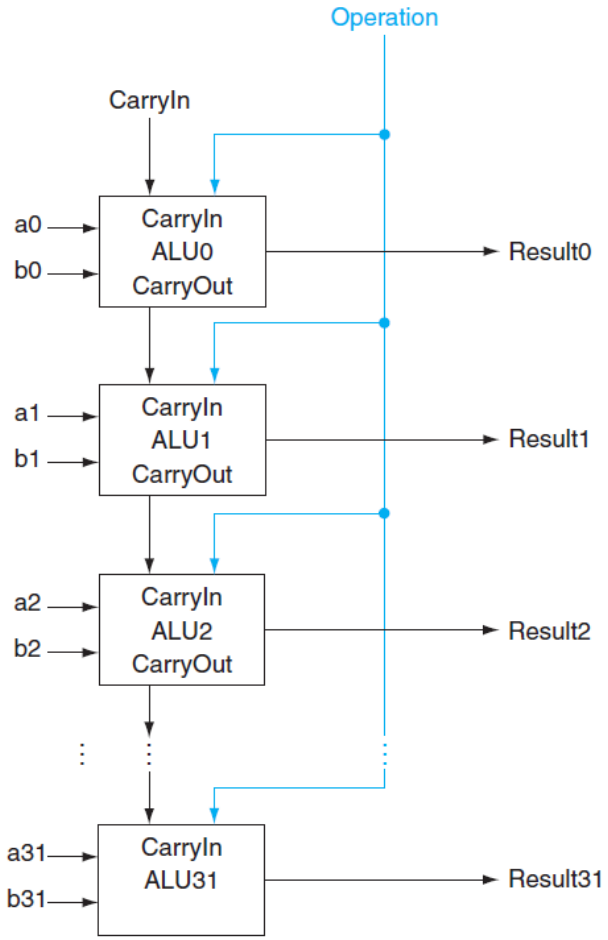
A 1-Bit ALU

- Input: two operand (1-bit), ALU control signal, carryin
- Output: result, carryout
- Operation: add, and, or



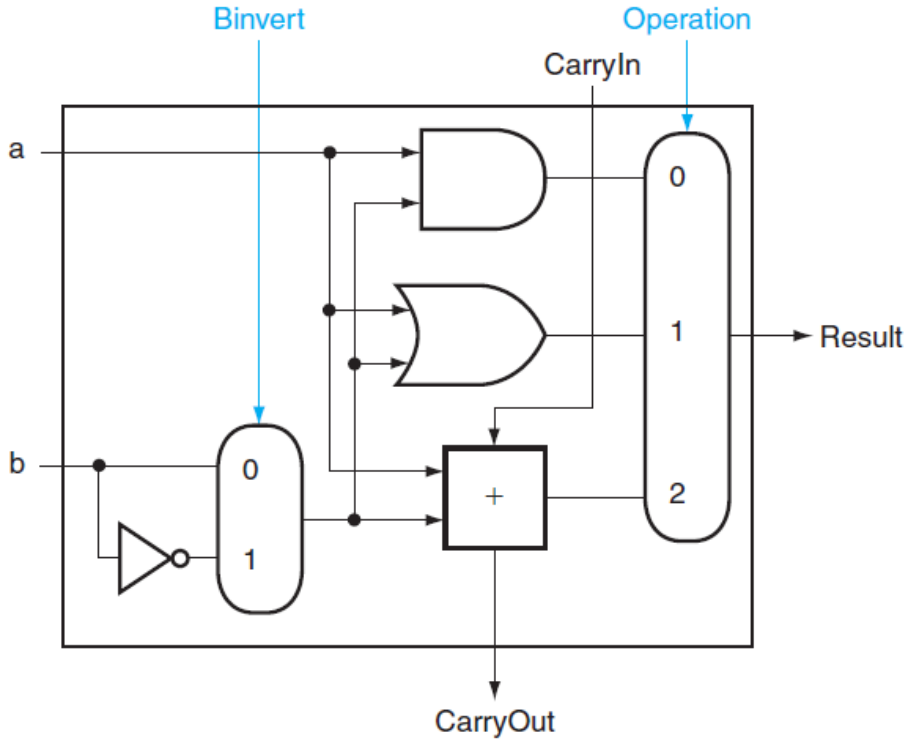
A 32-Bit ALU

- Ripple carry adder (1-bit ALU -> 32-bit ALU)



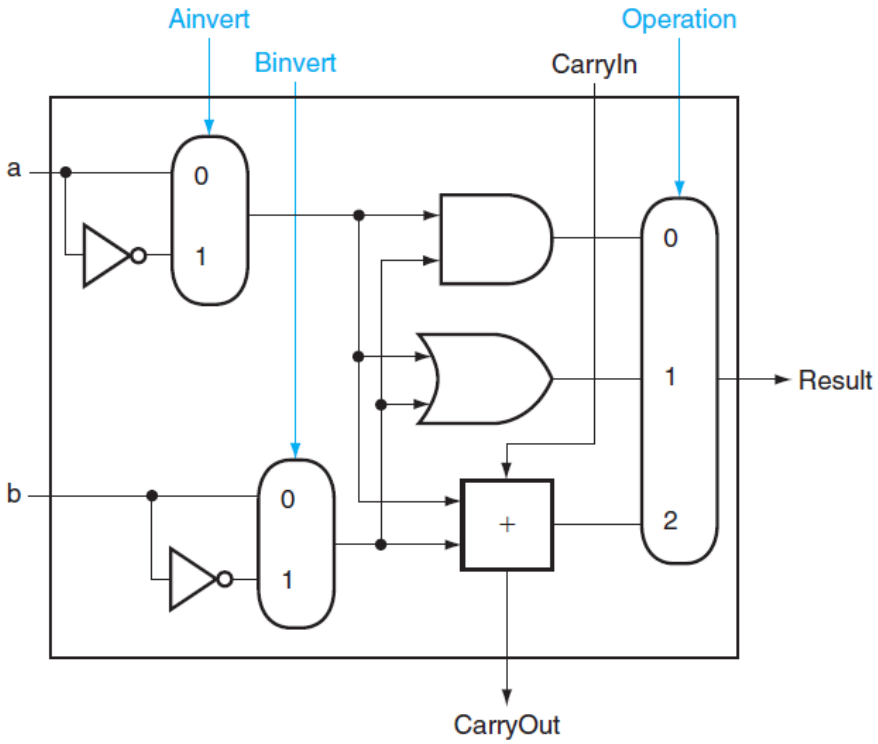
A 1-Bit ALU

- Subtraction?
  - a – b = a + (-b) = a + (~b + 1) = a + ~b + 1
- Operation: add, subtract, and, or



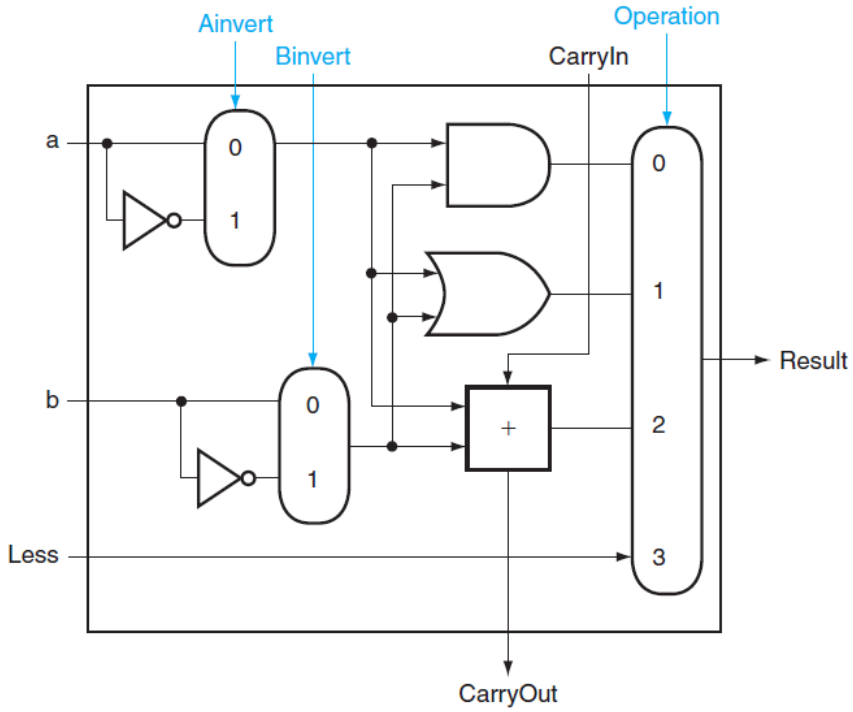
A 1-Bit ALU

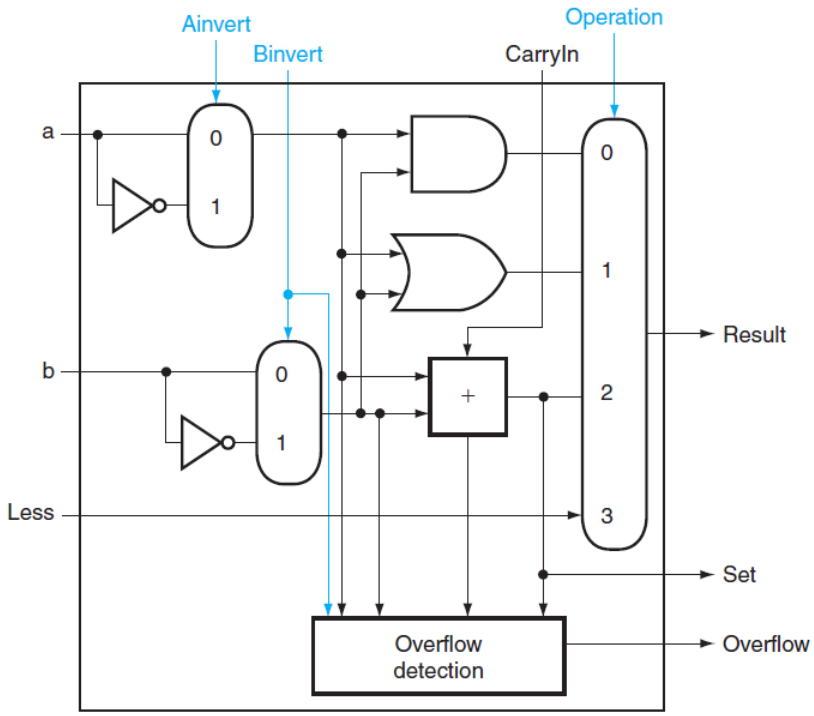
- NOR?
  - !(a | b) = !a & !b
- Operation: add, subtract, and, or, nor



A 1-Bit ALU

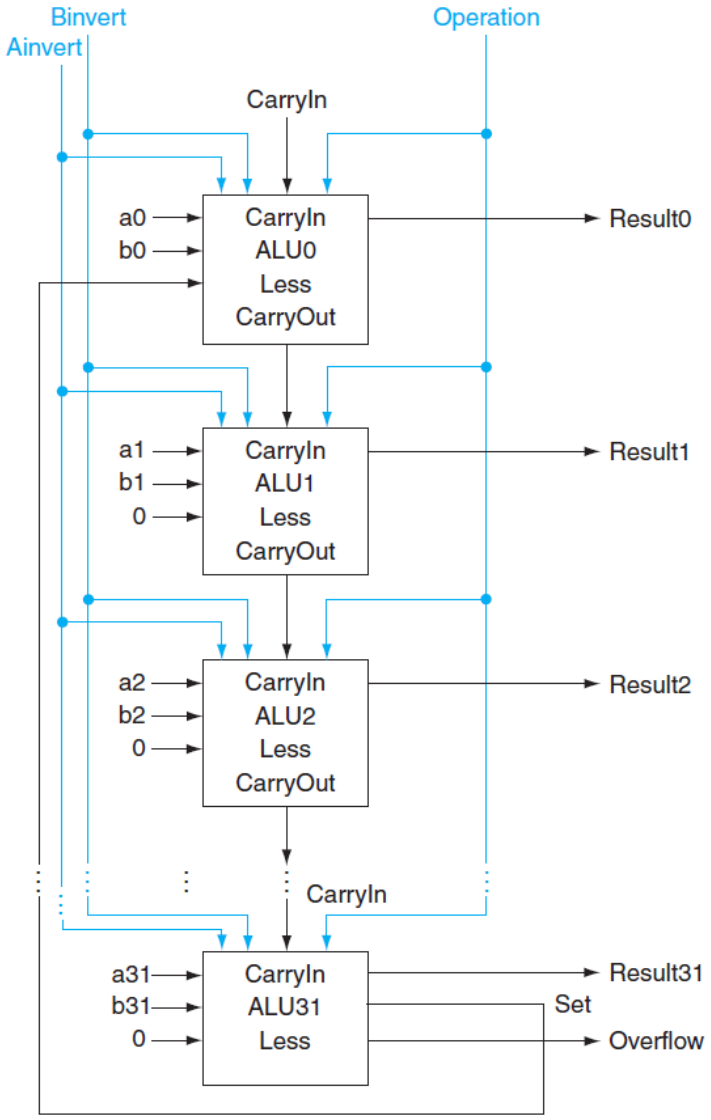
- Set on less than?
  - Less than: sign bit of (a – b)
  - Set lsb of result to be 1 if a < b



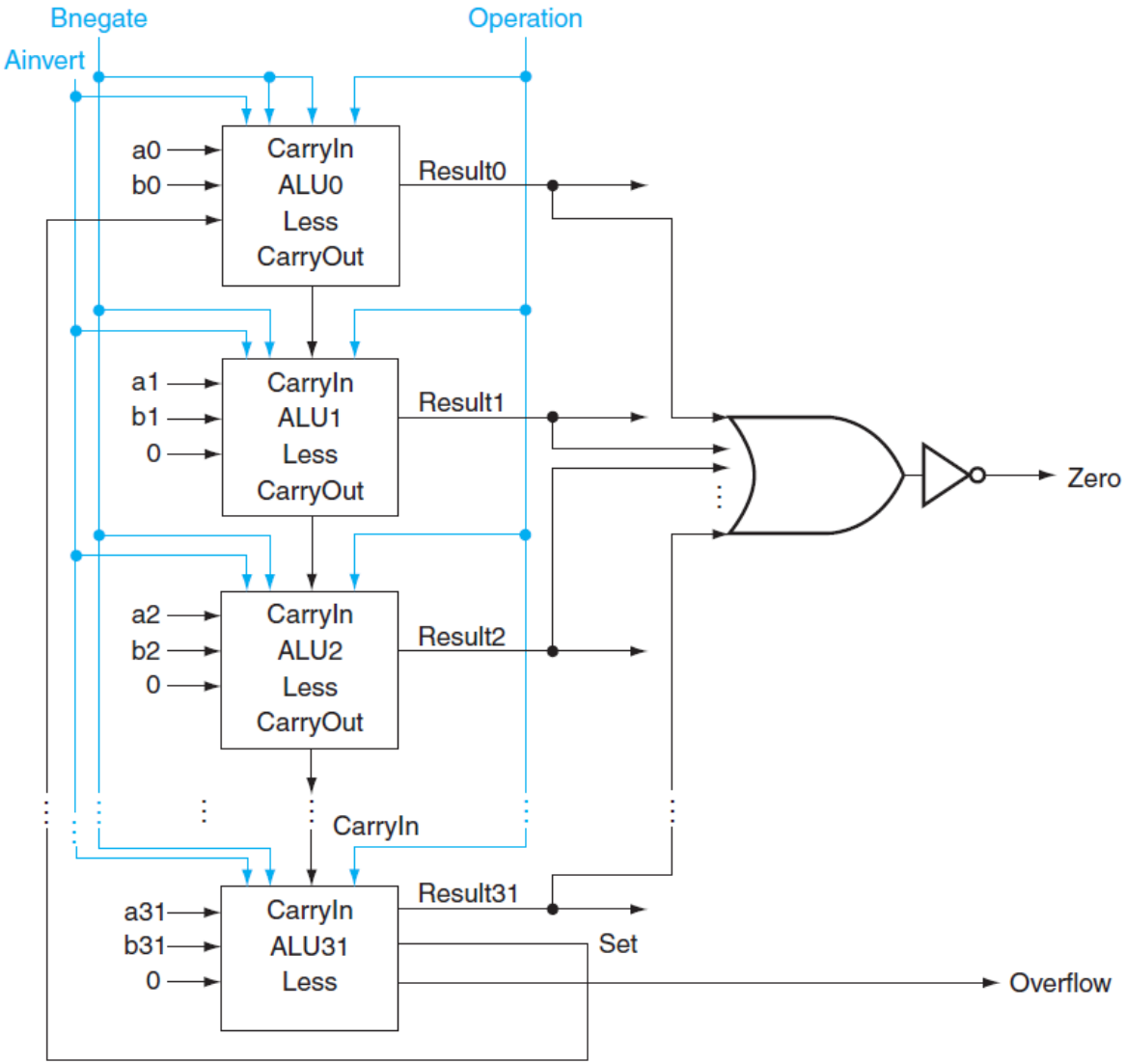


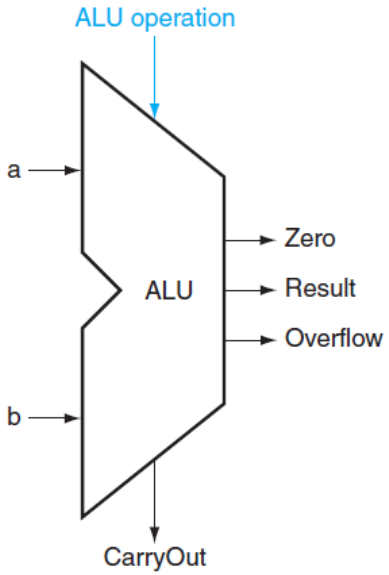
A 32-Bit ALU

- Operation
  - Add
  - Subtract
  - AND
  - OR
  - NOR
  - slt



A 32-Bit ALU

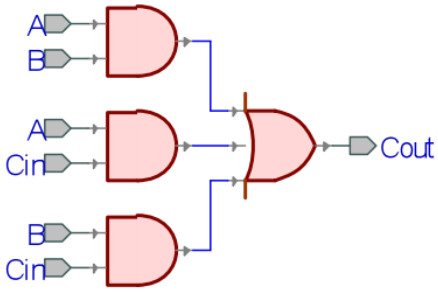

Bnegate
Ainvert
Operation
a0
b0
CarryIn
ALU0
Less
CarryOut
Result0
a1
b1
0
CarryIn
ALU1
Less
CarryOut
Result1
Zero
a2
b2
0
CarryIn
ALU2
Less
CarryOut
Result2
CarryIn
a31
b31
0
CarryIn
ALU31
Less
Result31
Set
Overflow

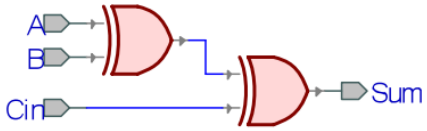

ALU operation
a
b
ALU
Zero
Result
Overflow
CarryOut

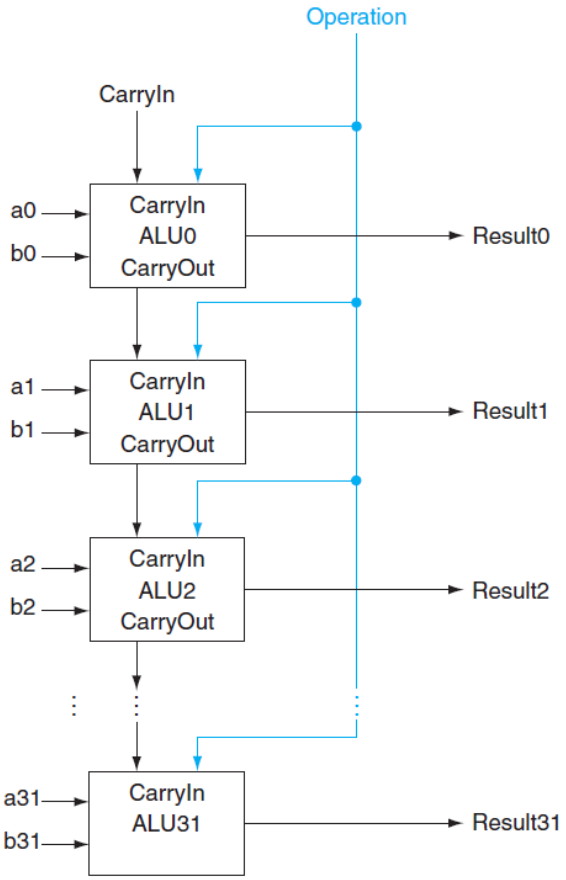
Delay Analysis

16-bit Ripple Carry Adder

- Worst case delay for 16-bit Ripple Carry Adder
  - Si = Ci ^ Ai ^ Bi
    Ci+1 = Ai & Bi + Ai & Ci + Bi & Ci
  - Delay(S) = 2 gate delays
  - Delay(C) = 2 gate delays
  - Worst case delay = 2 * 16 gate delays

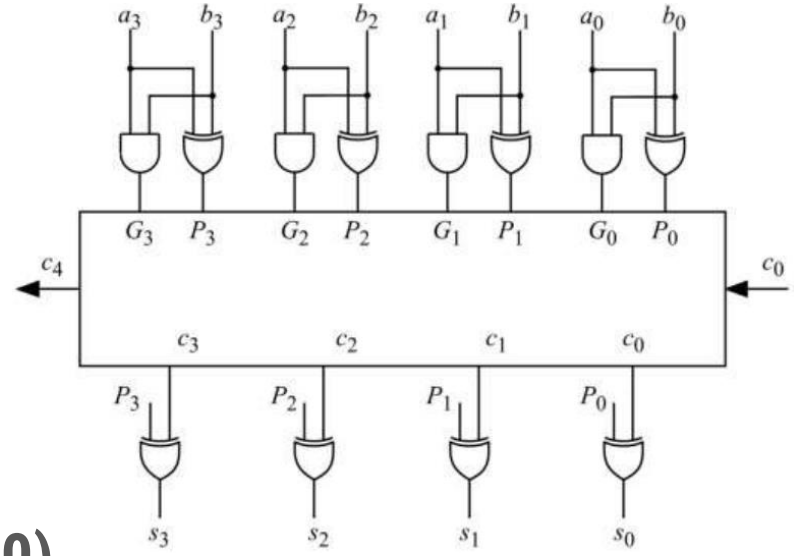






16-bit Carry Lookahead Adder

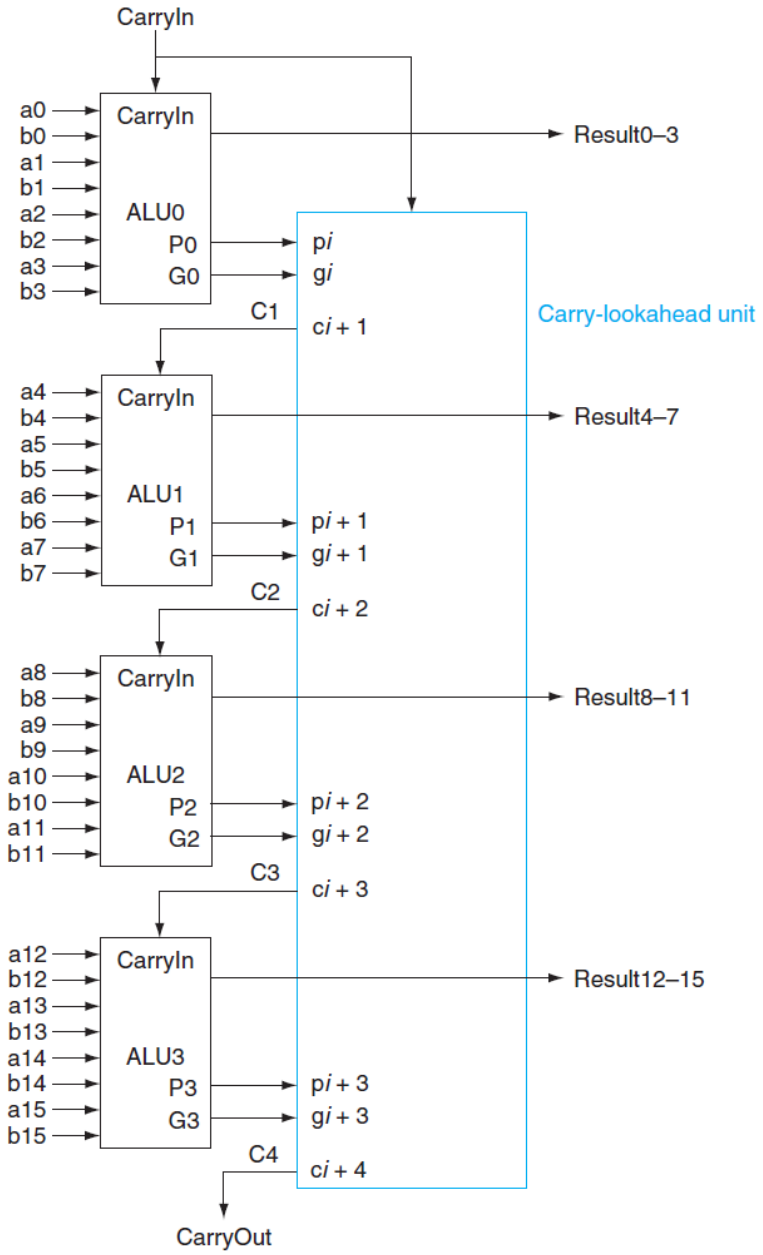
- For calculating Cout, we will use two additional signals at 1-bit adder
  - Generate (gi) = Ai & Bi
  - Propagate (pi) = Ai ^ Bi



- 4-bit CLA
  - c1 = g0 + (p0 & c0)
    c2 = g1 + (p1 & g0) + (p1 & p0 & c0)
    c3 = g2 + (p2 & g1) + (p2 & p1 & g0) + (p2 & p1 & p0 & c0)
    c4 = g3 + (p3 & g2) + (p3 & p2 & g1) + (p3 & p2 & p1 & g0)
    + (p3 & p2 & p1 & p0 & c0)
  - Delay(c4) = 2 + max{delay(gi), delay(pi)} = 2 + 1 = 3 gate delays

16-bit Carry Lookahead Adder

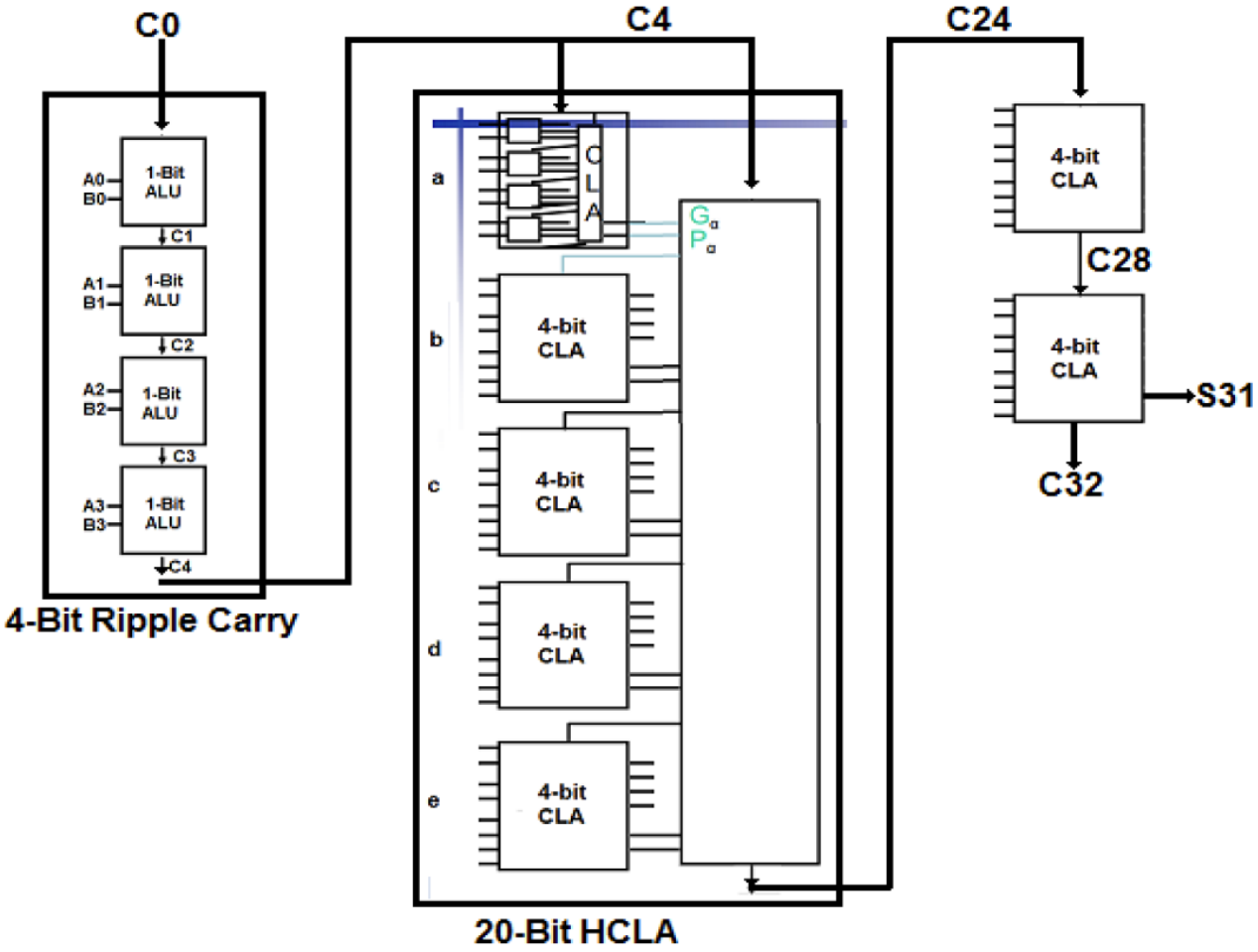
- **Hierarchical CLA**
  - P0 = p3 & p2 & p1 & p0
    G0 = g3 + (p3 & g2) + (p3 & p2 & g1)
    + (p3 & p2 & p1 & g0)
  - C1 = G0 + (P0 & c0)
    C2 =
    C3 =
    C4 = G3 + (P3 & G2) + (P3 & P2 & G1)
    + (P3 & P2 & P1 & G0)
    + (P3 & P2 & P1 & P0 & c0)
  - Delay(C4) = 2 + max{delay(P), delay(G)}
    = 2 + delay(G)
    = 2 + 2 + max{delay(p), delay(g)}
    = 2 + 2 + 1
    = 5 gate delays



Sample Question

- Consider the 32-bit adder as shown, find the delay of C32
- Assume the delay time sheet

Fan-In	Delay
1	1T
2	2T
3	3T
4	5T
5	8T
6	13T



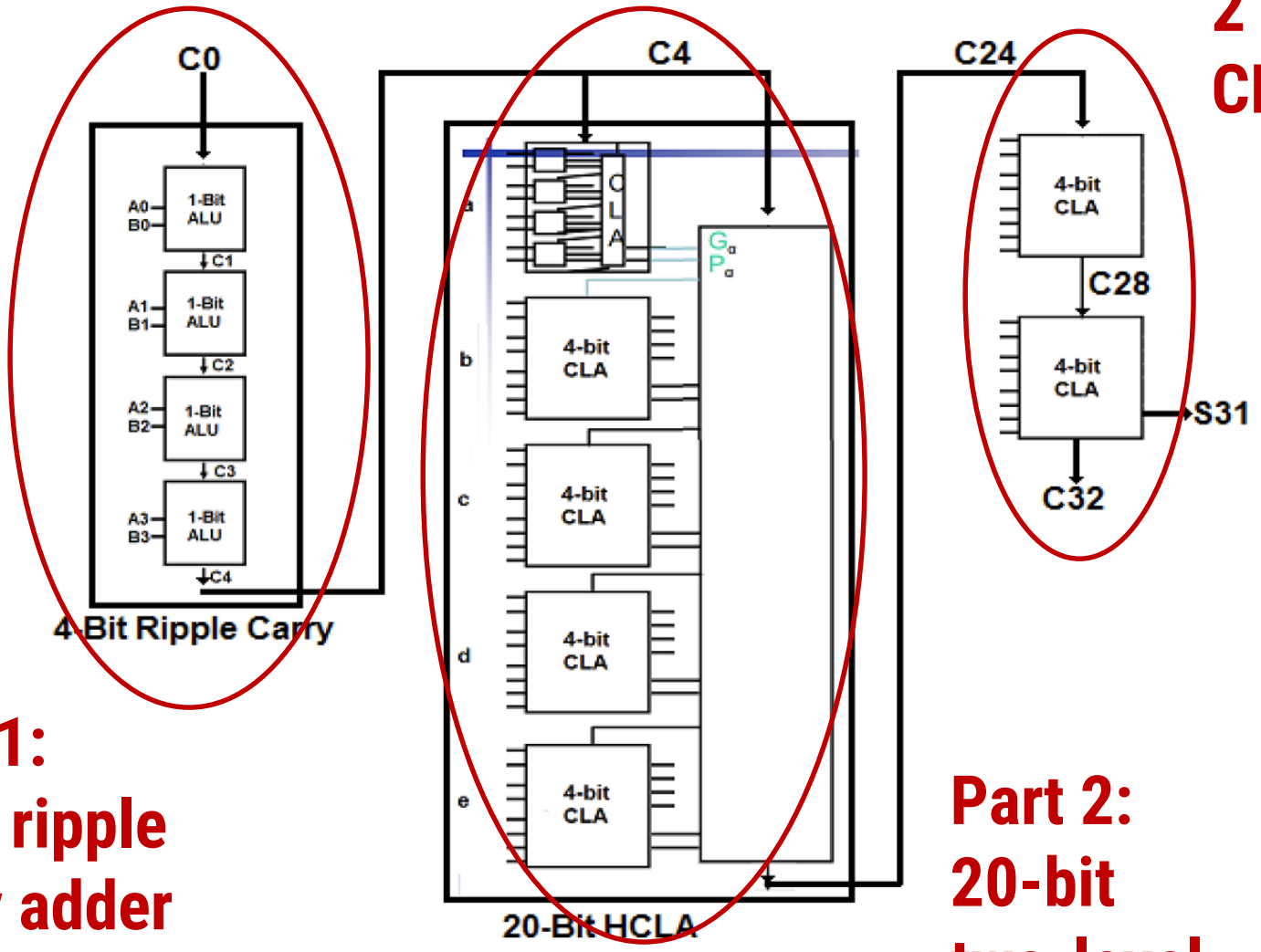
Steps

1. Find the critical path

2. Understand the components of the ALU

3. Calculate time delay for different parts in order
   - When calculating time delay for different part, decompose it as a simple ALU as we have discussed before

Part 3:
2 x 4-bit
CLA

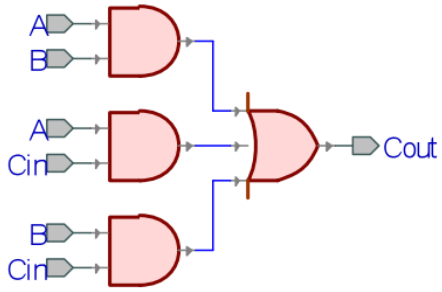


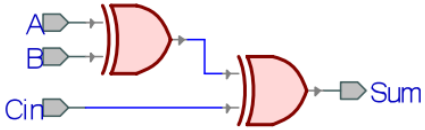
Part 1:
4-bit ripple
carry adder

Part 2:
20-bit
two-level
CLA

Part 1: 4-bit Ripple Carry Adder

Fan-In	Delay
1	1T
2	2T
3	3T
4	5T
5	8T
6	13T





Si = Ci ^ Ai ^ Bi

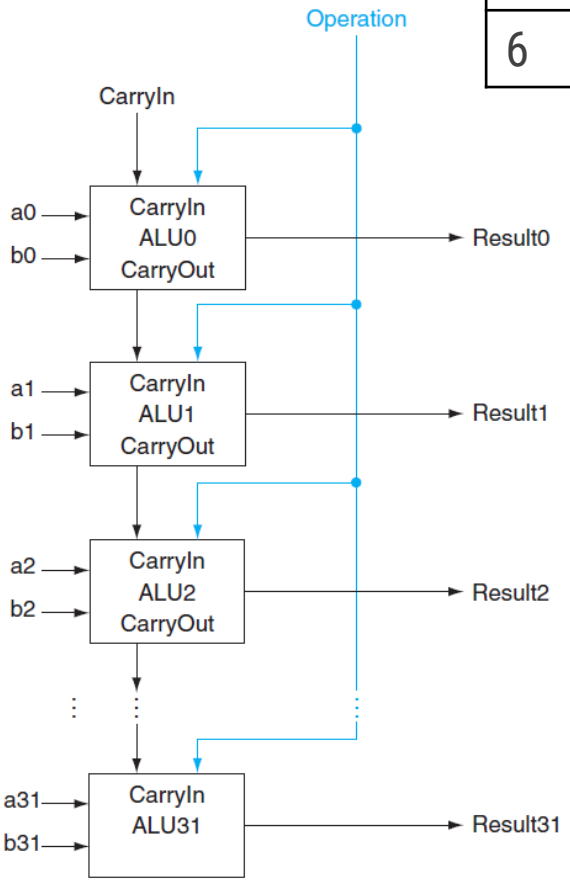
Ci+1 = Ai & Bi + Ai & Ci + Bi & Ci

Delay(S) = 2T + 2T = 4T

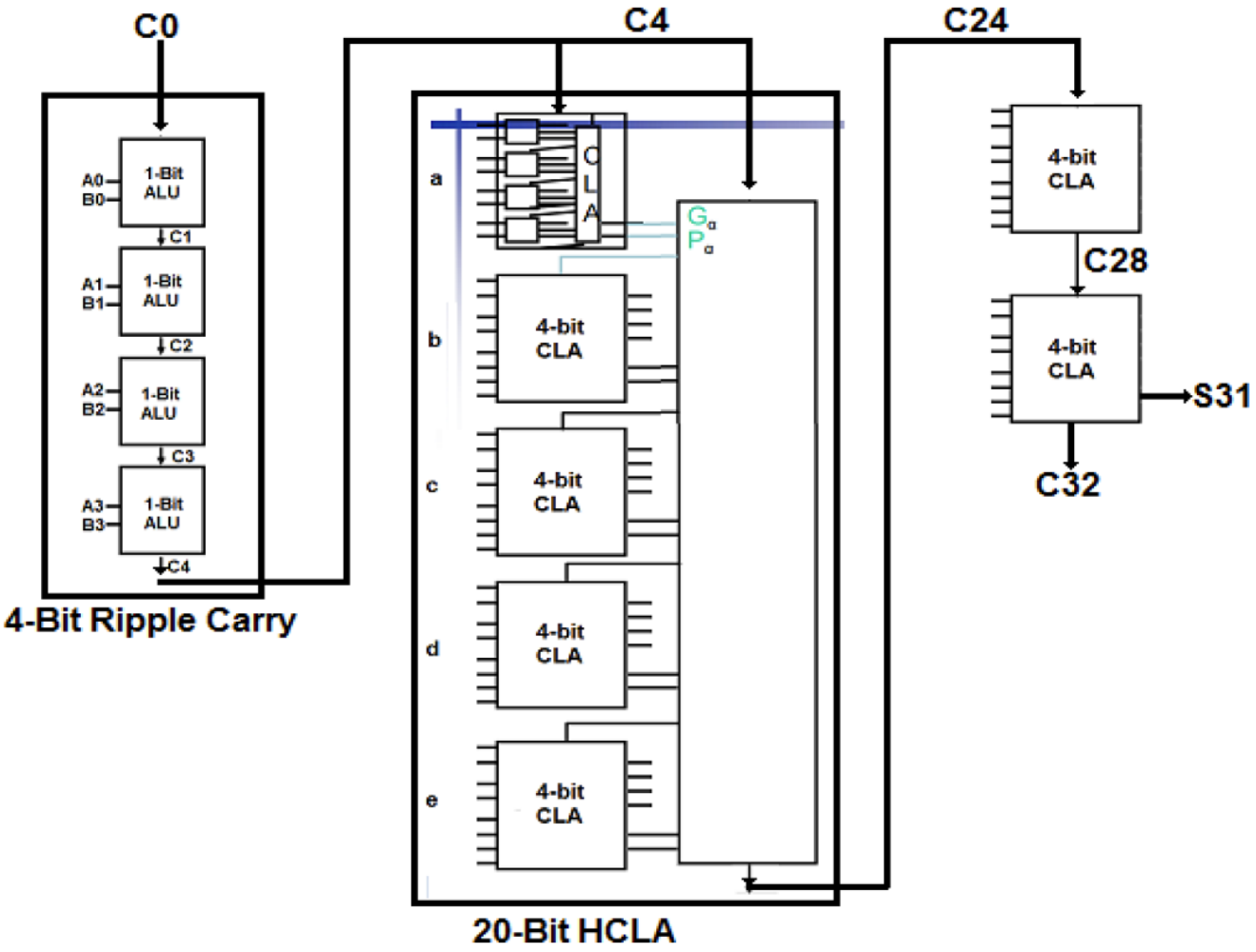
Delay(C) = 2T + 3T = 5T

Delay(Cin) = 0

Delay(C4) = 20T

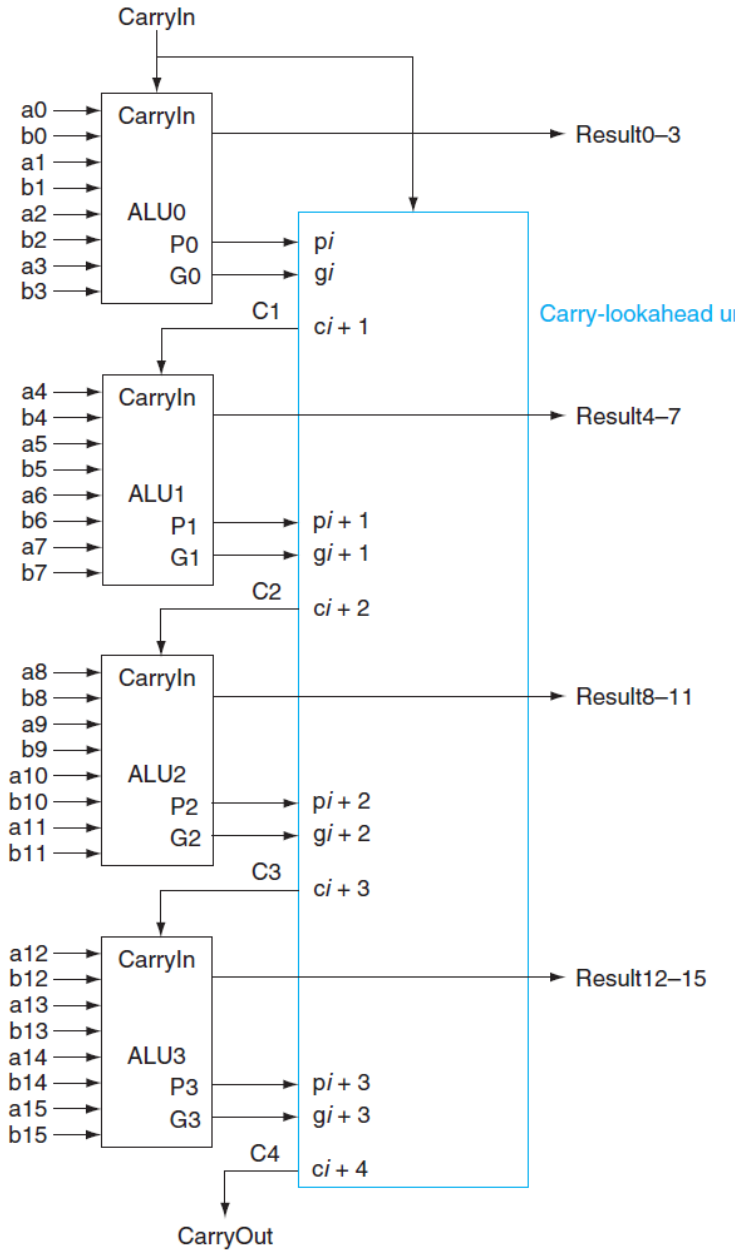


20T


C0
C4
C24
A0
B0
1-Bit ALU
C1
A1
B1
1-Bit ALU
C2
A2
B2
1-Bit ALU
C3
A3
B3
1-Bit ALU
C4
4-Bit Ripple Carry
a
b
c
d
e
CLA
Ga
Pa
4-bit CLA
4-bit CLA
4-bit CLA
4-bit CLA
4-bit CLA
20-Bit HCLA
4-bit CLA
C28
4-bit CLA
S31
C32

Part 2: 20-bit Two-level CLA

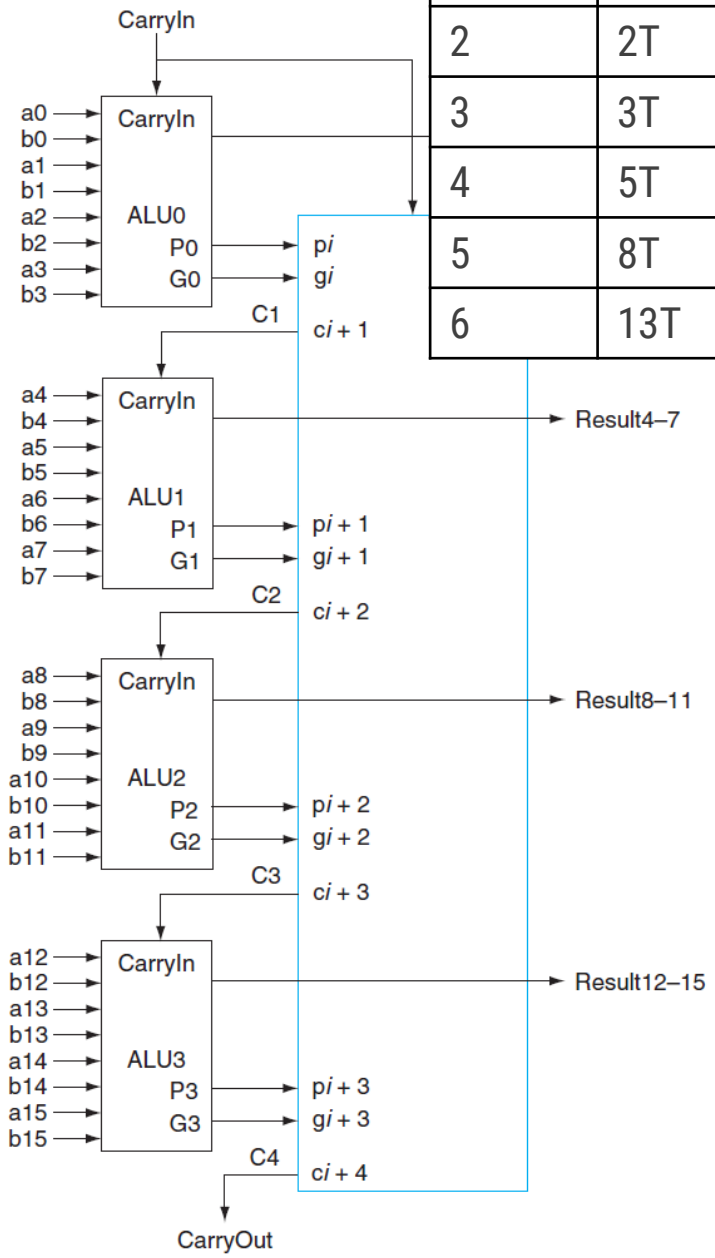
- **Level 1: 4-bit CLA**
  - P0 = p3 & p2 & p1 & p0
    G0 = g3 + (p3 & g2) + (p3 & p2 & g1)
    + (p3 & p2 & p1 & g0)
  - Delay(P0) = 5T + 2T = 7T **< Delay(Cin)**
    Delay(G0) = 10T + 2T = 12T **< Delay(Cin)**
  - The latency of computing Pi and Gi is hidden by the latency of computing C4
  - **Computing Pi and Gi is not on the critical path**
  - **We only need to analyze the second level**

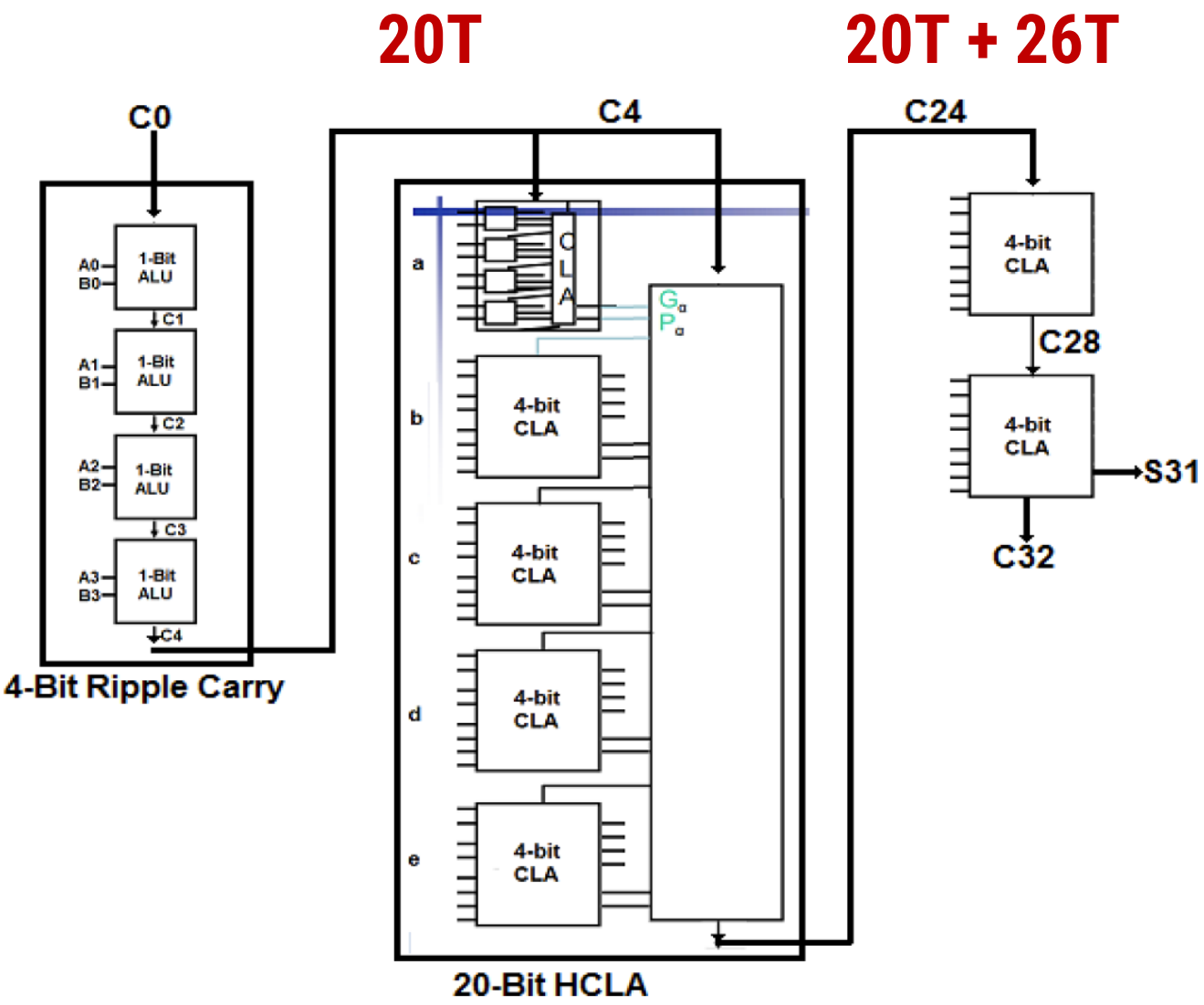


Part 2: 20-bit Two-level CLA

- **Level 2: 5-bit CLA**
  - C1 = G0 + (P0 & c0)
    C2 =
    C3 =
    C4 =
    C5 = G4 + (P4 & G3) + (P4 & P3 & G2)
    + (P4 & P3 & P2 & G1)
    + (P4 & P3 & P2 & P1 & G0)
    + (P4 & P3 & P2 & P1 & P0 & c0)
  - Delay(C5) = 13T + 13T = 26T
  - pi and gi from the first level is not considered here because they can be pre-computed before c0 arrives

Fan-In	Delay
1	1T
2	2T
3	3T
4	5T
5	8T
6	13T

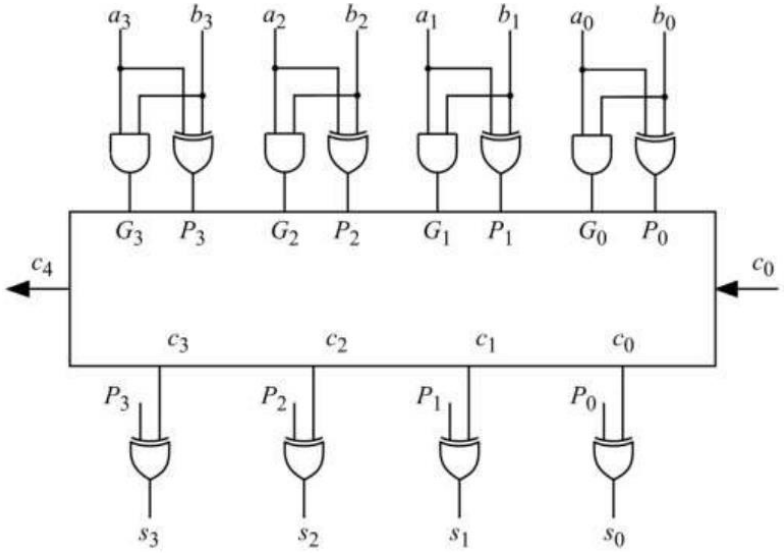


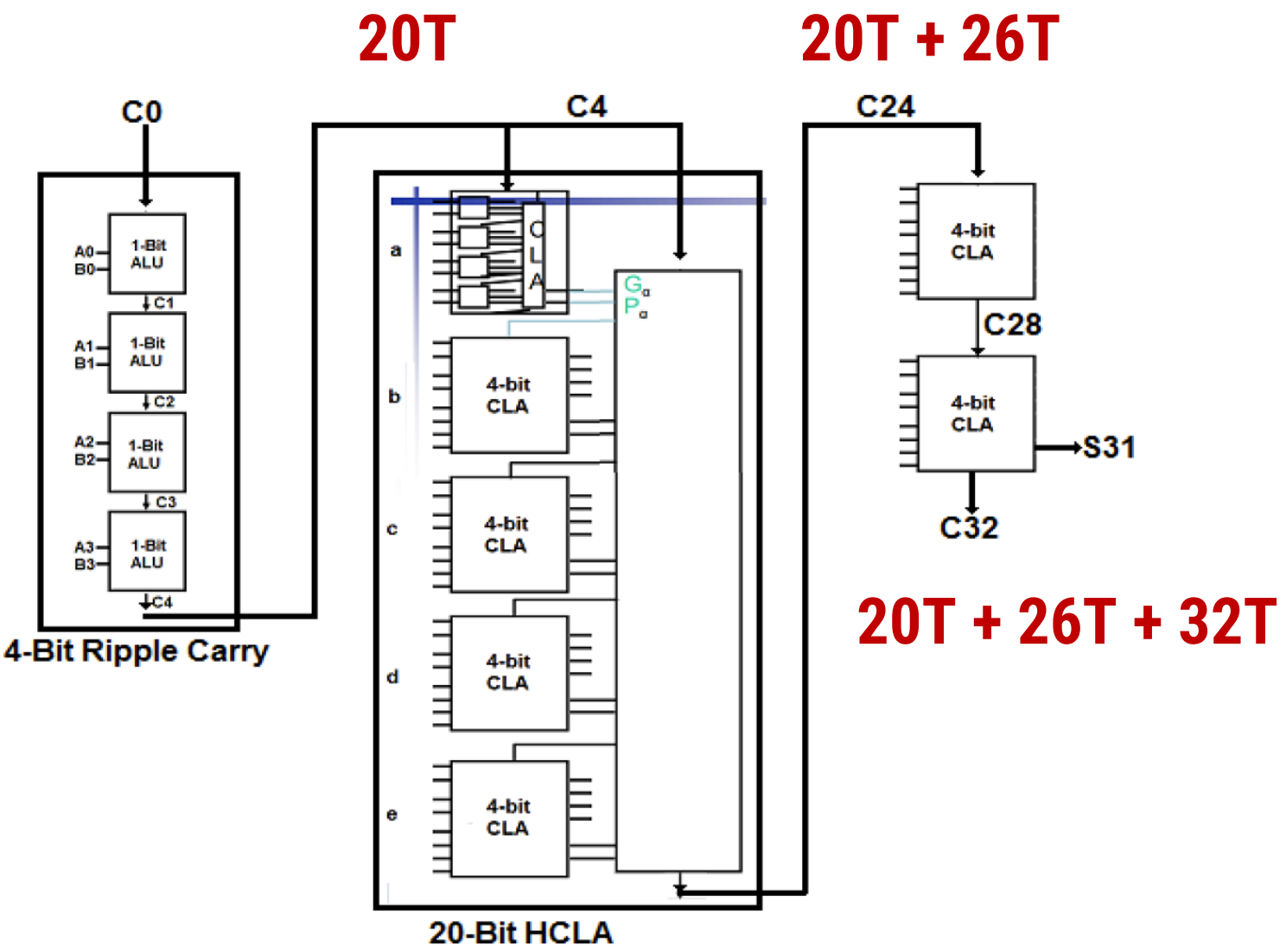

20T
20T + 26T
C0
C4
C24
A0
B0
1-Bit ALU
C1
A1
B1
1-Bit ALU
C2
A2
B2
1-Bit ALU
C3
A3
B3
1-Bit ALU
C4
4-Bit Ripple Carry
a
b
c
d
e
CLA
Ga
Pa
4-bit CLA
4-bit CLA
4-bit CLA
4-bit CLA
20-Bit HCLA
4-bit CLA
C28
4-bit CLA
S31
C32

Part 3: 2 x 4-bit CLA

Fan-In	Delay
1	1T
2	2T
3	3T
4	5T
5	8T
6	13T

- **4-bit CLA**
  - **c1 = g0 + (p0 & c0)**
    c2 = g1 + (p1 & g0) + (p1 & p0 & c0)
    c3 = g2 + (p2 & g1) + (p2 & p1 & g0) + (p2 & p1 & p0 & c0)
    c4 = g3 + (p3 & g2) + (p3 & p2 & g1) + (p3 & p2 & p1 & g0)
    + (p3 & p2 & p1 & p0 & c0)
  - **Delay(c4) = 16T**




20T
20T + 26T
C0
C4
C24
A0
B0
1-Bit ALU
C1
A1
B1
1-Bit ALU
C2
A2
B2
1-Bit ALU
C3
A3
B3
1-Bit ALU
C4
4-Bit Ripple Carry
a
b
c
d
e
CLA
Ga
Pa
4-bit CLA
4-bit CLA
4-bit CLA
4-bit CLA
4-bit CLA
20-Bit HCLA
4-bit CLA
C28
4-bit CLA
S31
C32
20T + 26T + 32T