1. Consider a 4-bit carry lookahead adder which has two levels with two 2-bit groups on first level.
   (a) Please find out the formula of G0, G1, P0, P1, C1, and C2 when adding two unsigned 4-bit numbers and a carry-in , say c0. (Please describe your answer in terms of p0~p3, g0~g3 and c0).
   (b) Compare the number of gate delays for the above carry lookahead adder with a 4-bit ripple carry adder.

2. Calculate the time necessary to perform a multiply of two 32-bit integers. Please
   (a) Use the optimized multiplication block diagram.
   Assume the operations of multiplier-bit0 checking and step 1a are merged as : add the multiplicand ANDed with bit0 of multiplier. So this step is always performed.
   Assume each step of operation takes 7 time units.
   (b) Use the fast multiplication hardware with 31 adders . Assume an adder takes 7 time units.

3. Please make a comparison and list the difference between the two versions of the multiplication block diagrams.

4. (a) Complete the table below using 1-bit Booth's algorithm to compute 3×-4 (i.e., $0011_2 \times 1100_2$)
   (b) Some particular bit patterns may make Booth's algorithm slow. Give an example and explain.

Iteration	Multiplicand	Product
0 (initial)	0011	
1		
2		
3		
4		

1 (a)

P0 = p1 p0

P1 = p3 p2

G0 = g1 + p1 g0

G1 = g3 + p3 g2

C1 = G0 + P0 c0

C2 = G1 + P1C1 = G1 + P1 G0 + P1 P0c0

(b) 4-bit ripple carry: 8 gate delay

4-bit carry lookahead: 2+2+1=5 gate delay

2. (a) add: 7tu shift: 7tu check for end: 7u

3x7(tu) x32(loops) = 672 tu

(b) 5 level x 7 tu = 35 tu

3

Multiplicand is 64 bits	Multiplicand is 32 bits
Multiplicand don't shift	Multiplicand don't shift
ALU is 64-bit	ALU is 32-bit
Product don't shift	Product must shift right
Multiplier is independent 32-bits	Multiplier is in low 32-bit of product

4.

Iteration	Multiplicand	Product	
0 (initial)	0011	0000 1100 0	
1	0011	0000 0110 0	NOP/shift
2	0011	0000 0011 0	NOP/shift
3	0011	1110 1001 1	Sub/shift
4	0011	1111 0100 1	NOP/shift

(b) With many short 1's in the sequence

e.g., 10101010