10주차 예비보고서

전공 : 국제한국학과 학년 : 4학년 학번 : 20181202 이름 : 김수미

**1.** **4-Bit Adder 및 Subtractor 이진 병렬 연산 기능에 대하여 조사하시오.**

1) 4-Bit Binary Parallel Adder/Subtractor

6주차 실습 시간에 Half/Full Adder와 Half/Full Subtractor에 배운 적이 있었다. 이러한 Adder와 Subtractor를 N개씩 병렬로 연결하면 N 비트의 Add, Subtract 연산을 수행하는 회로를 만들 수 있다.

|  |
| --- |
| 4-Bit Binary Parallel Adder |
|  |
| 4-Bit Binary Parallel Subtractor |
|  |

즉 4-bit Binary Parallel Adder는 두개의 4-bit 2진수 간의 덧셈 연산을 수행할 수 있으며, 4-bit Binary Parallel Subtractor는 두개의 4-bit 2진수 간의 뺄셈 연산을 수행할 수 있다. 4-bit Binary Parallel Adder는 위 그림처럼 1-bit Full Adder 4개가 병렬적으로 연결된 형태를 띠며 4-bit Binary Parallel Subtractor는 1-bit Full Subtractor 4개가 병렬적으로 연결된 형태를 띤다.

**2. Look ahead carry 대하여 조사하시오.**

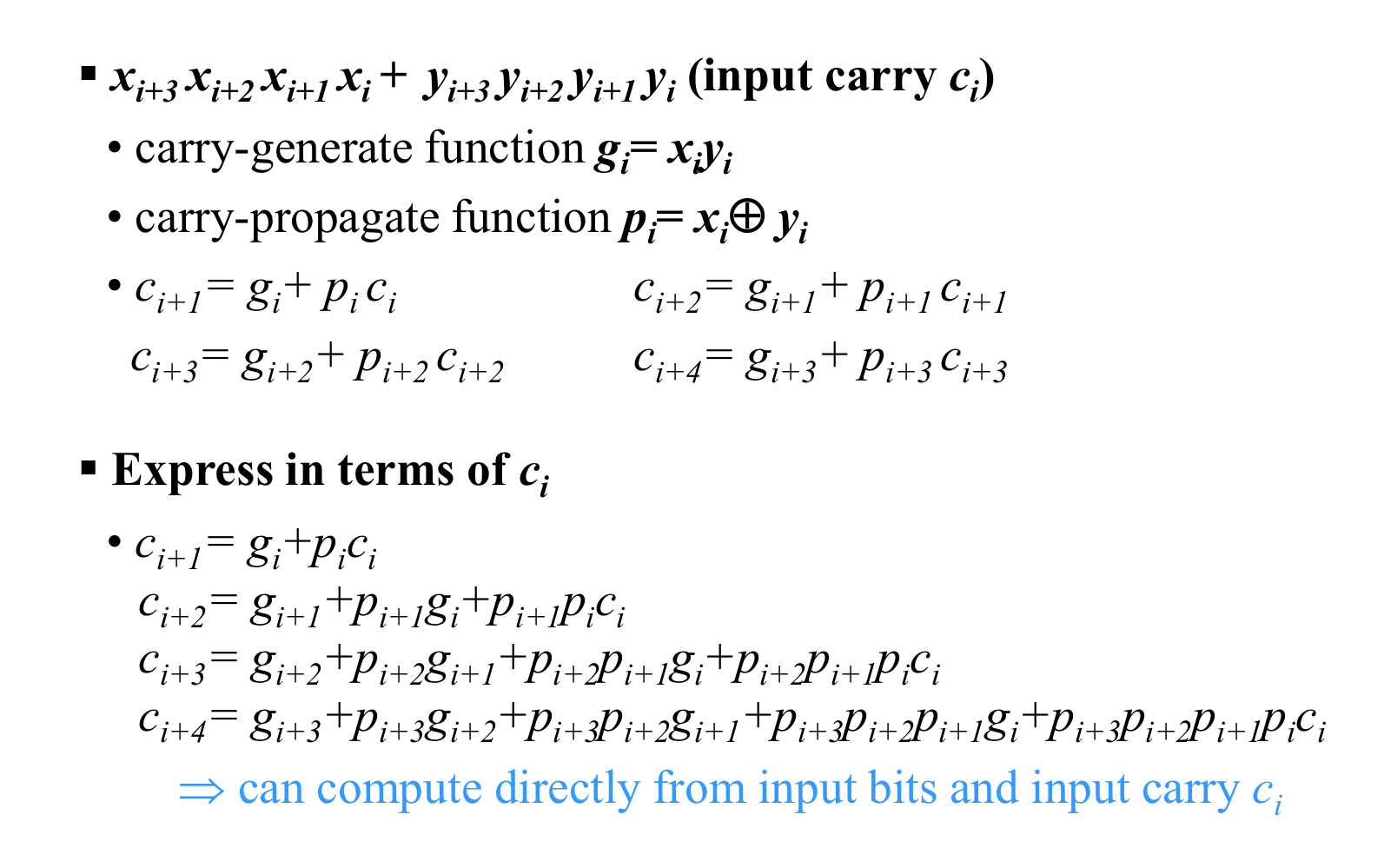
Look ahead carry란 ripple-carry chain 회로의 연산에서 발생하는 delay를 줄이기 위해 사용하는 연산 방식이다. 여러개의 bit에 대한 덧셈 연산에서 원래는 carry가 다음 자리 비트의 add 연산으로 전달되며 계산이 순차적으로 진행되는데, 모든 비트의 carry에 대한 식을 풀어서 계산함으로써 각 bit 자리수의 carry를 이전 bit로 부터의 carry 전달이 필요 없이 한번에 계산할 수 있게 되어 게이트 통과에 대한 delay를 감소시킬 수 있는 연산 방식이다. 아래는 4bit 덧셈 연산에서 발생하는 carry를 look ahead carry 방식으로 풀어 전개한 것이다.

|  |
| --- |
| Look Ahead Carry (4bit Adder) |
|  |
| 4bit Look-ahead Adder |
|  |

**3. XOR을 활용한 2‘s complement 가감산에 대하여 조사하시오.**

2’s complement 가감산의 특징은 특정 수에서 음수를 빼는 경우, 이를 양수의 덧셈으로 바꾸어 계산하고 양수를 빼는 경우 음수의 덧셈으로 바꾸어 계산하는 등 모든 감산 연산을 가산 연산으로 변환하여 계산한다는 특징이 있다.

이 때 두 수의 가산 연산에서 carry가 generate 되는 상황은 더하는 자릿수의 bit가 모두 1 일 때 이다. 즉 and 게이트를 이용하여 carry가 generate 되는 상황을 감지할 수 있다. 두 수의 가산 연산에서 carry가 propagate 되는 상황은 더하는 자릿수의 bit 에서 둘 중 하나의 bit만 1로 set 되어 있는 경우이다. 아래 자릿수에서 올라온 carry bit가 현재 자릿수에서 다시 다음 자릿수로 전달되기 때문이다. 즉 xor 게이트를 이용하면 carry가 propagate 되는 상황을 감지할 수 있다.



**4. BCD 연산에 대하여 조사하시오.**

1) BCD 연산

BCD code의 경우 4bit 값을 피연산자로 사용하지만 0에서 9까지의 숫자만 표현하기 때문에 0000부터 1001까지의 이진수만 사용하고 1010부터 1111까지의 이진수는 사용하지 않는다(don’t care). 따라서 일반적인 2’complement 연산으로 계산하면 안되고, 추가적인 처리가 필요하다 : 4bit 이진수의 연산 결과값이 1010부터 1111까지 범위에 해당되는 경우, 6만큼, 즉 0110을 연산 결과에 더해주면 된다.

|  |
| --- |
| BCD 연산의 예시 |
|  |

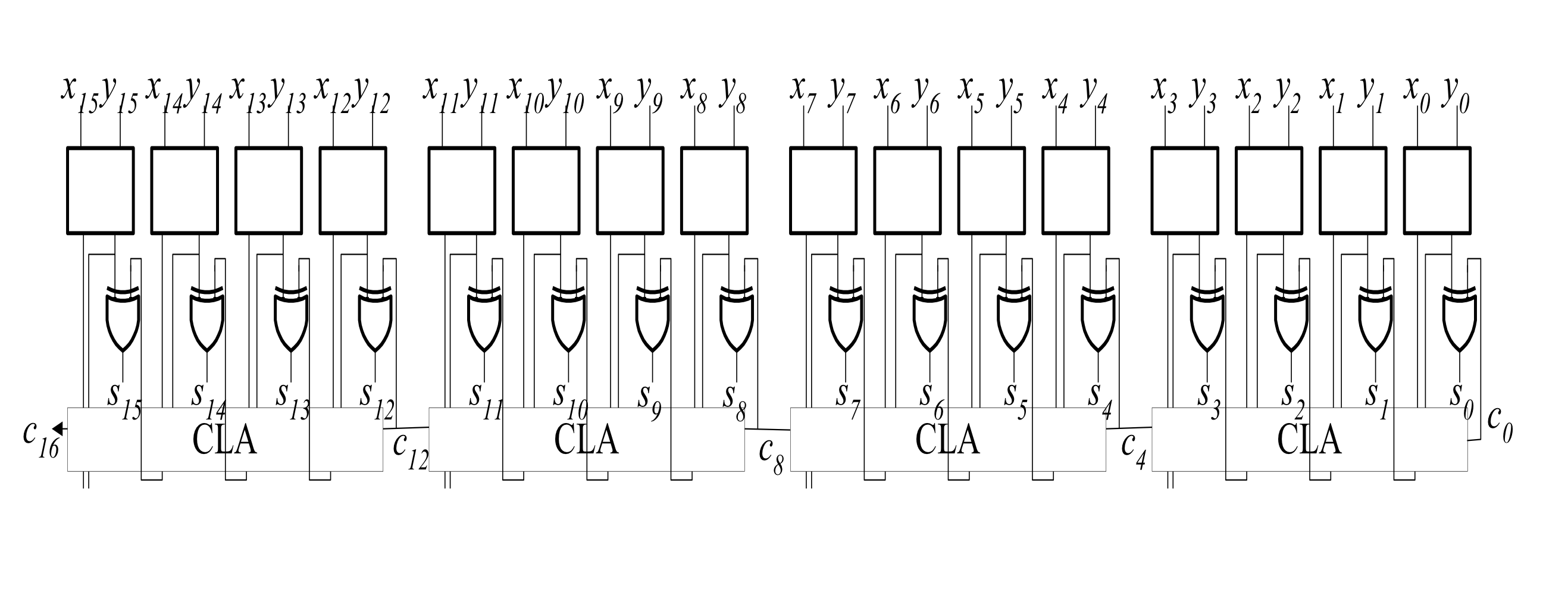
**5.** **ALU의 기능에 대하여 조사하시오.**

ALU(Arithmetic Logic Unit)란 덧셈, 뺄셈 같은 산술 연산이나 논리합(OR), 논리곱(AND), 등의 논리연산을 계산을 수행하는 디지털 회로이다. ALU는 덧셈 연산을 수행하는 Adder를 중심으로 연산 값을 저장하는 데이터와 연산 결과를 저장하는 레지스터, 보수를 만드는 보수기(complementary), 오버플로우를 검출하는 오버플로우 검출기 등으로 구성되어 있다.

|  |
| --- |
| ALU(Arithmetic Logic Unit)의 구조 |
|  |

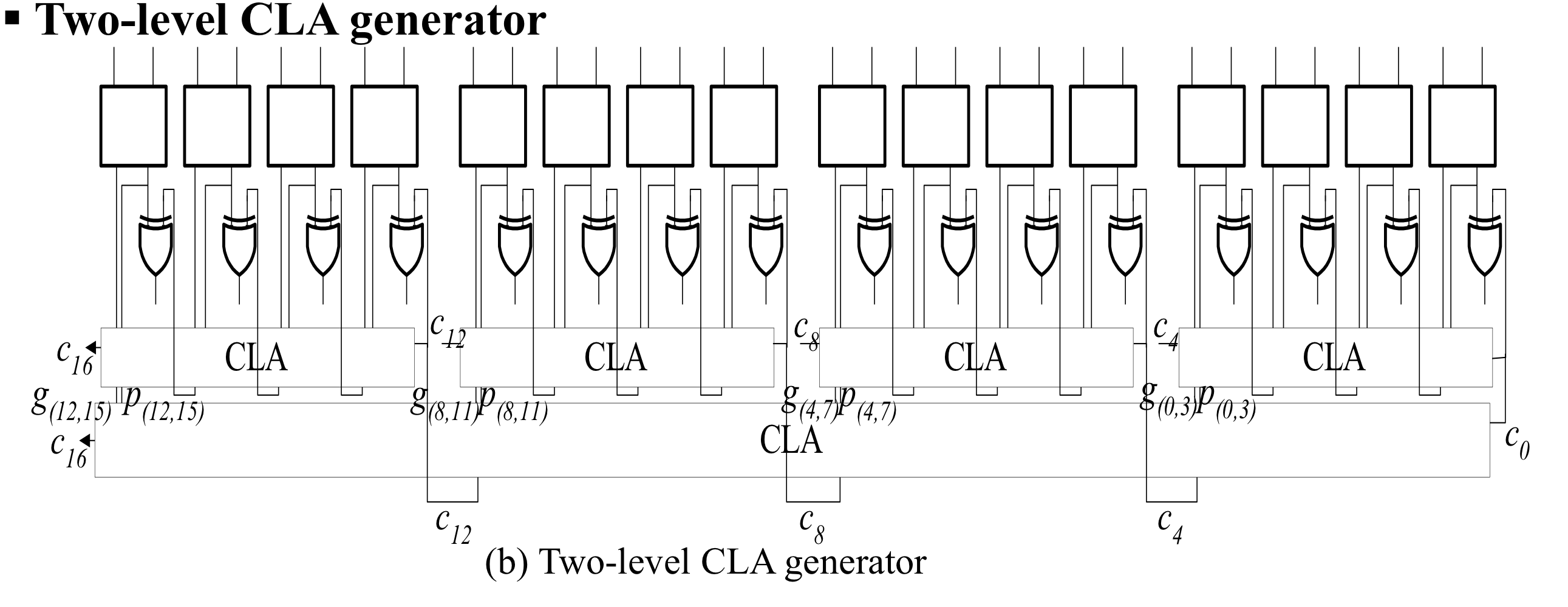
**6. 기타 이론**

1) Single-level 16-bit Carry Look-ahead Adder



4-bit carry look-ahead adder 4개를 이어서 16-bit carray look-ahead adder를 만들 수 있다. 4개의 parallel 구조를 serial 하게 연결한 형태이며 parallel 구조와 serial 구조가 섞인 형태를 띤다.

2) 2-level 16-bit Carry Look-ahead Adder



2-level 16-bit Carry Look-ahead Adder는 CLA generator를 이중으로 사용하여 Single-level 16-bit Carry Look-ahead Adder보다 gate를 통과하는데 소요되는 delay를 더 감소시킬 수 있는 구조이다.

|  |
| --- |
| 2-level 16-bit Carry Look-ahead Adder |
|  |