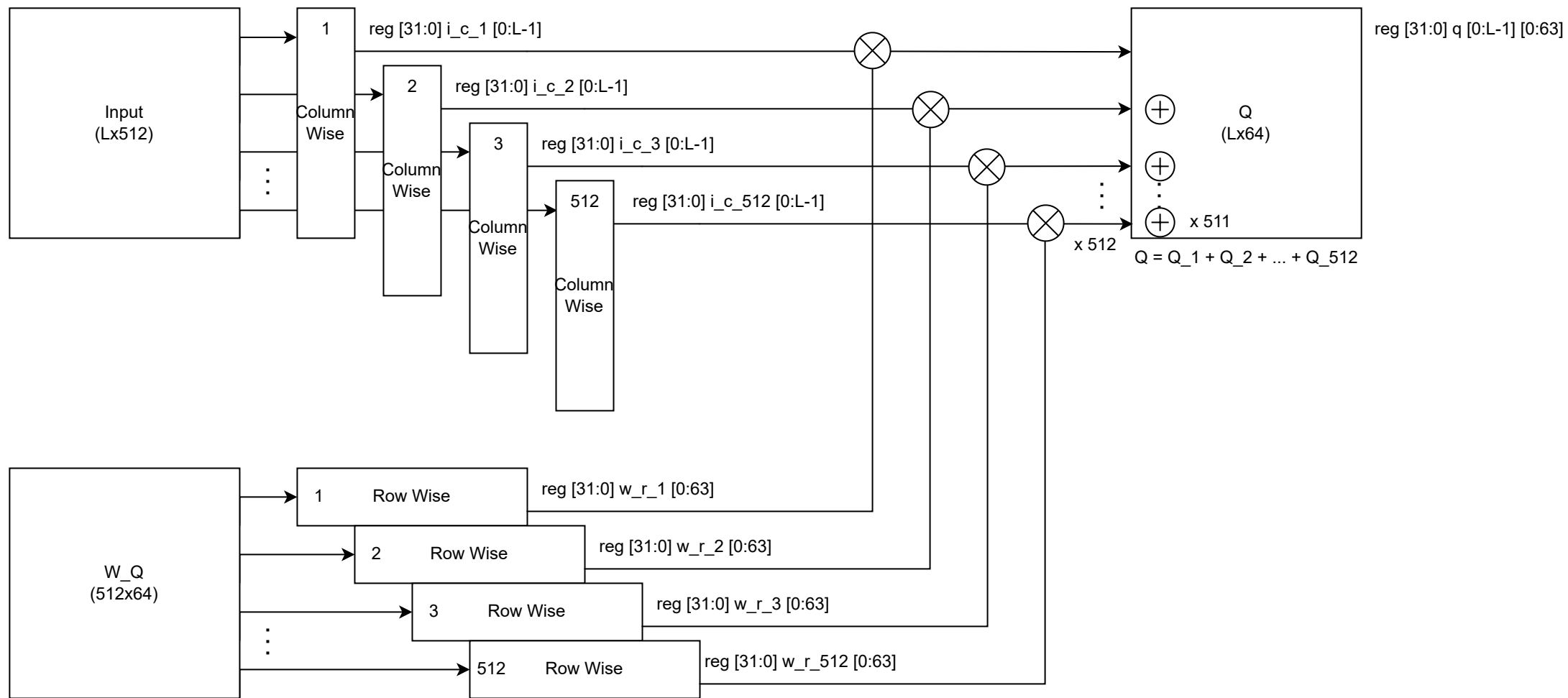


<개선한 점>

1. 곱셈과 덧셈연산에서 clk 비슷하게 맞추기
2. Input 행렬을 column 기준으로 잘라서 1cycle 연산이 끝났을 때 Q행렬의 모든 요소에 값이 하나씩 채워지게 하기
3. Q 행렬 요소에 adress를 부여하여 연산값이 주소를 보고 찾아갈 수 있게 하여 메모리 부담 줄이기
4. 덧셈은 한 번 더하는 구조에서 끝나게 설계
5. Input 행렬 불러오는 방식도 고안하기



$$\text{Min(PE)} = (2 \times 64) \times 512 = 65k7\|$$

$$\text{Max(PE)} = (256 \times 64) \times 512 = 8388k7\|$$

$$\text{Min(PE)} = (2 \times 64) \times 511 = 65k7\|$$

$$\text{Max(PE)} = (256 \times 64) \times 511 = 8372k7\|$$



$$\text{Min(Register 32bit)} = \text{PE} \times 4B = 260kB$$

$$\text{Max(Register 32bit)} = \text{PE} \times 4B = 33MB$$



$$\text{Min(Register 32bit)} = 2 \times 64 \times 4B = 512B$$

$$\text{Max(Register 32bit)} = 256 \times 64 \times 4B = 65536B = 66kB$$

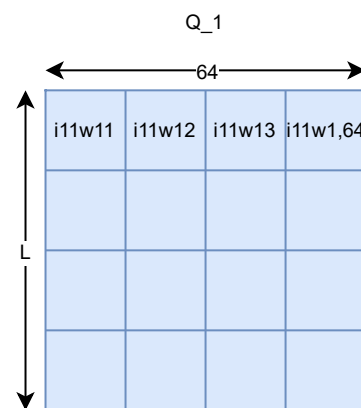
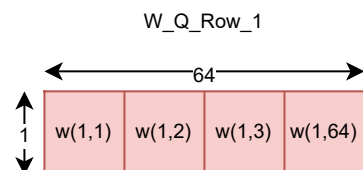
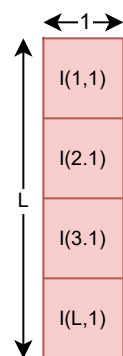
$$\text{Min(clk)} = (4\text{clk} \times 512) = 2048\text{clk}$$

$$\text{Max(clk)} = (4\text{clk} \times 512) = 2048\text{clk}$$

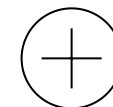
$$\text{Min(clk)} = (4\text{clk} \times 511) = 2044\text{clk}$$

$$\text{Max(clk)} = (4\text{clk} \times 511) = 2044\text{clk}$$

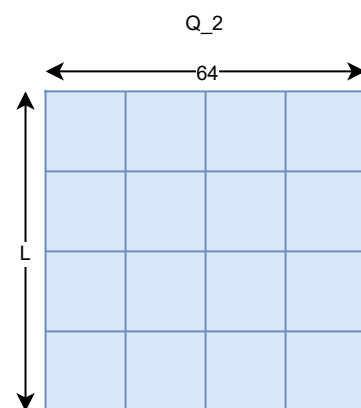
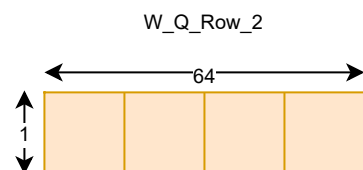
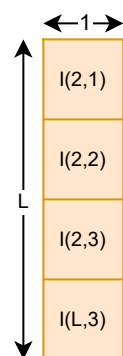
Input_Column_1



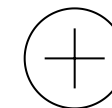
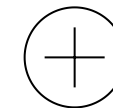
Text



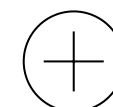
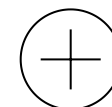
Input_Column_2



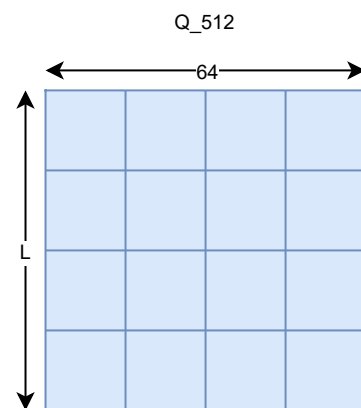
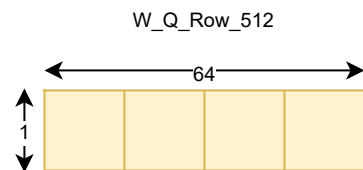
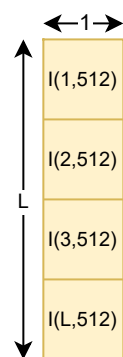
x 512



x 511



Input_Column_512



<추후 개선방안>

1. 한 요소 x 한 요소가 아닌 여러 요소를 동시에 곱하는 multiplier구조 설계
--> Q행렬을 쪼개는 방식으로 설계(몇개를 동시에 연산할지, 직/병렬구조 고안)
--> 하드웨어 사이즈 감소, 클럭수 증가(hw size / clk 는 반비례 관계)
2. Input 행렬을 쪼개는 방식과 Q 행렬에 address를 지정하는 것은 SRAM에서 해야하므로 추후에 설계
3. Input Column 및 W_Q Row는 쪼개졌다고 fix 후 그 이후 과정 도식화하기
4. Register 사이즈는 같거나 아주 조금 줄었을 것, 어쩔 수 없으니 무시하고 설계