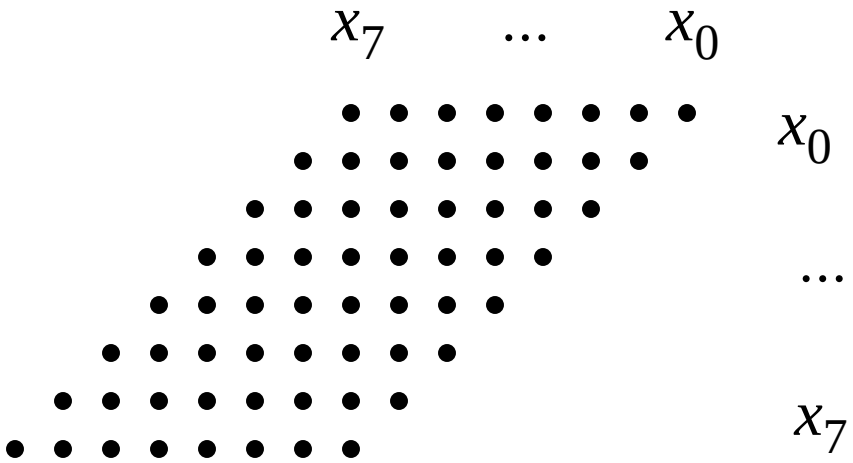
# SQUARING

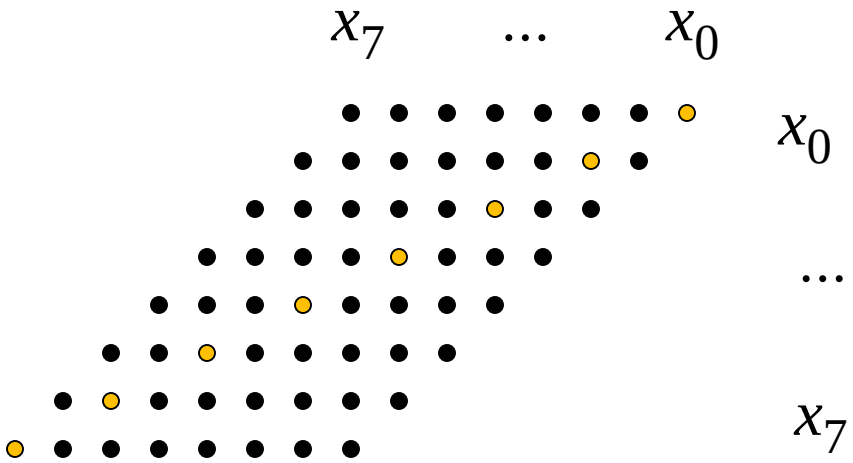
# Squaring

- $x^2$ can be calculated with about half the hardware of a full multiply (for a dedicated squaring block, of course)
- This simplification method begins with a non-Booth straightforward encoding of the *multiplier*'s bits



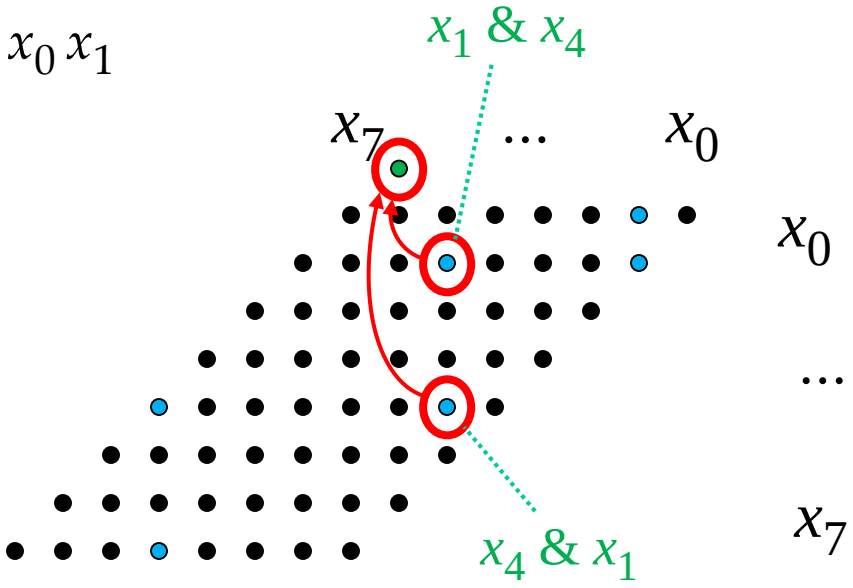
# Squaring

- Simplification #1: The partial product bits on the diagonal ($x_0$ & $x_0$, $x_1$ & $x_1$, …) can be replaced by the single input bit with no computation for that partial product bit
- 0 AND 0 = 0
  1 AND 1 = 1
  $x_i$ AND $x_i = x_i$



# Squaring

- Simplification #2: Pairs of equivalent bit products ($x_1\,x_0$ and $x_0\,x_1$, …) can be replaced by one bit product shifted over one column
- $x_1\,x_0, x_0\,x_1 \;=\; 2 \times x_0\,x_1$



# Squaring

- The end result is:
  - Approximately 1/2 of the partial product array is deleted
  - Generation of partial product bits along the diagonal of the array require no logic
  - The remaining approximately 1/2 of the partial product array is shifted one bit to the left (multiplied by 2)