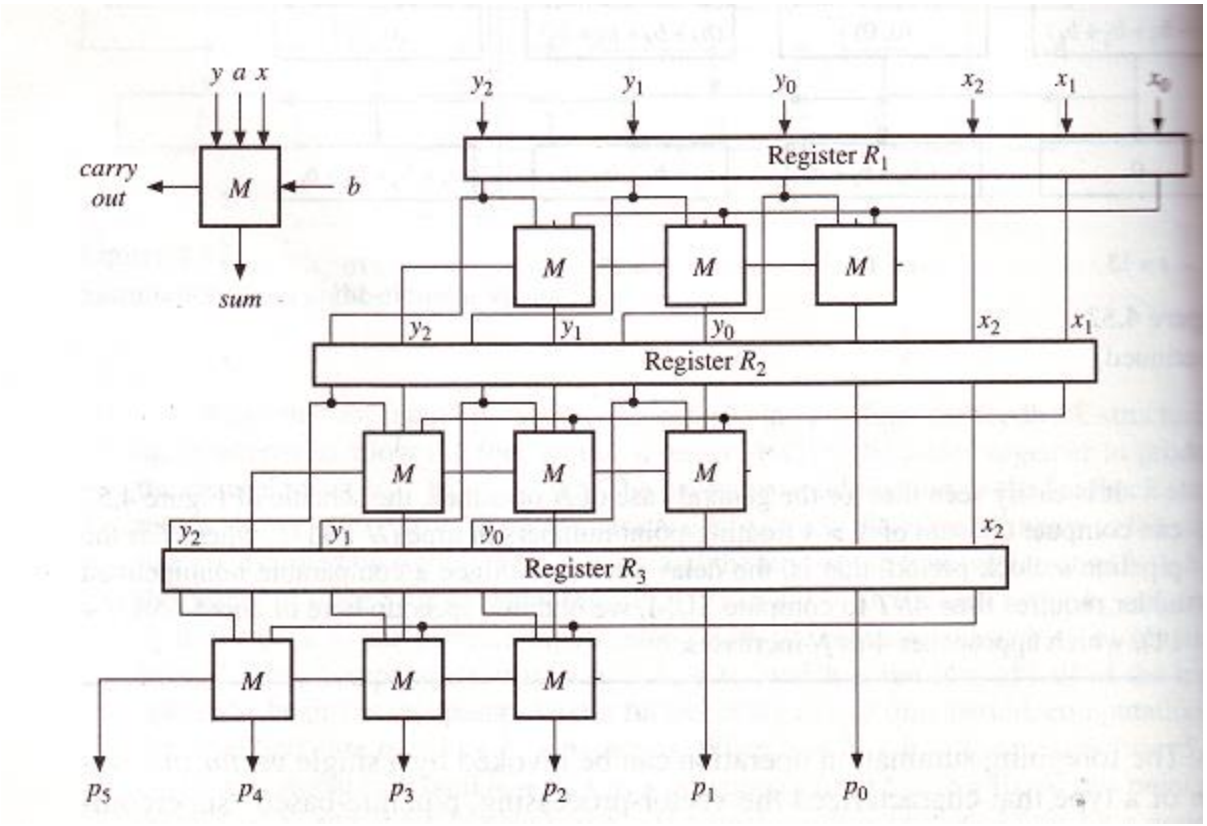
Datapath Design

Book of John P. Hayes

PP: 284-292

Pipelined Multiplier

Our task is to multiply two n-bit fixed-point binary numbers $X = x_{n-1}x_{n-2}\ldots x_1x_0$ and $Y = y_{n-1}y_{n-2}\ldots y_0$



Pipelined Multiplier

- ✔ The n cells in each stage S_i, $0<=i<=n-1$ computes the partial product of the form $P_i = P_{i-1} + x_i 2^i Y$
- ✔ Buffer register R_i stores the partial product, the multiplicand Y and the unused multiplier bits.

Advantages:

- ✔ An n-stage multiplier pipeline can overlap the computation of n separate products.
- ✔ It generates a new result every clock cycle, after the delay of the pipeline.

Pipelined Multiplier

Disadvantages:

✔ Slow speed of the carry propagation in each stage.

✔ The number of M cells needed in n^2.

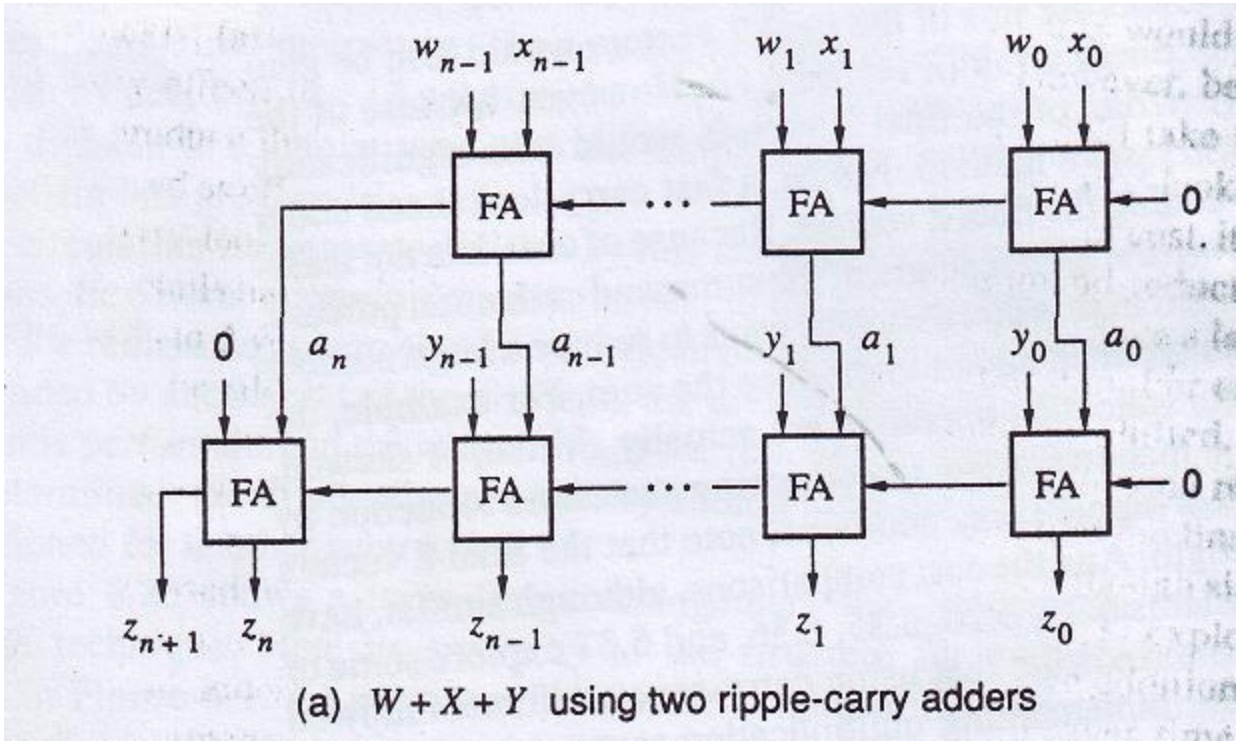
✔ The capacity of the buffer is huge.

✔ It is costly in hardware.

Carry Save Addition

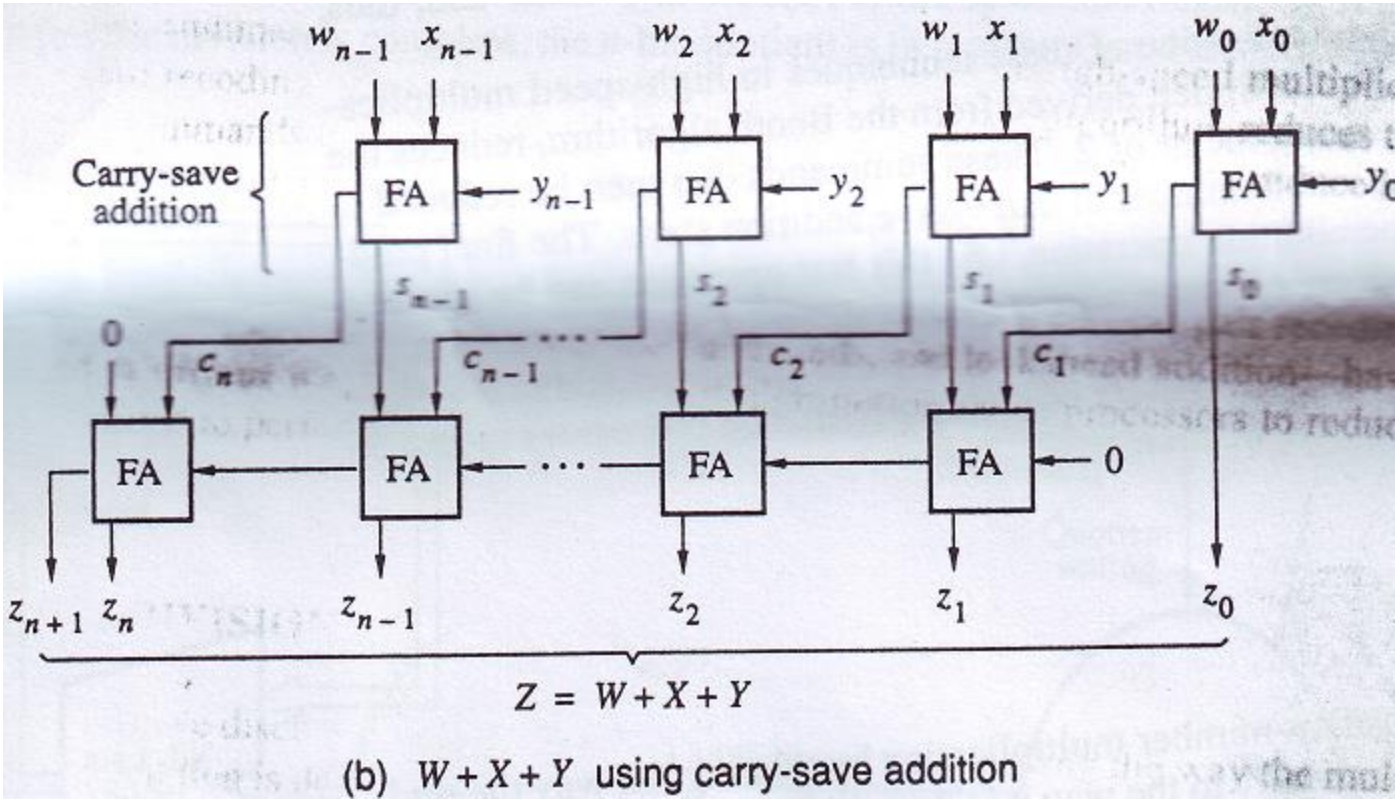
- ✔ An n-bit carry save adder consists of n disjoint full adders. It's input is three n-bit numbers to be added and the output consists of the n-sum bits forming a word *S* and the n carry bits forming a word *C*.
- ✔ Carry connections are shifted to the left correspond to normal carry propagation.
- ✔ To obtain the final result, *S* and *C* must be added by a conventional adder with carry propagation.

Carry Save Addition



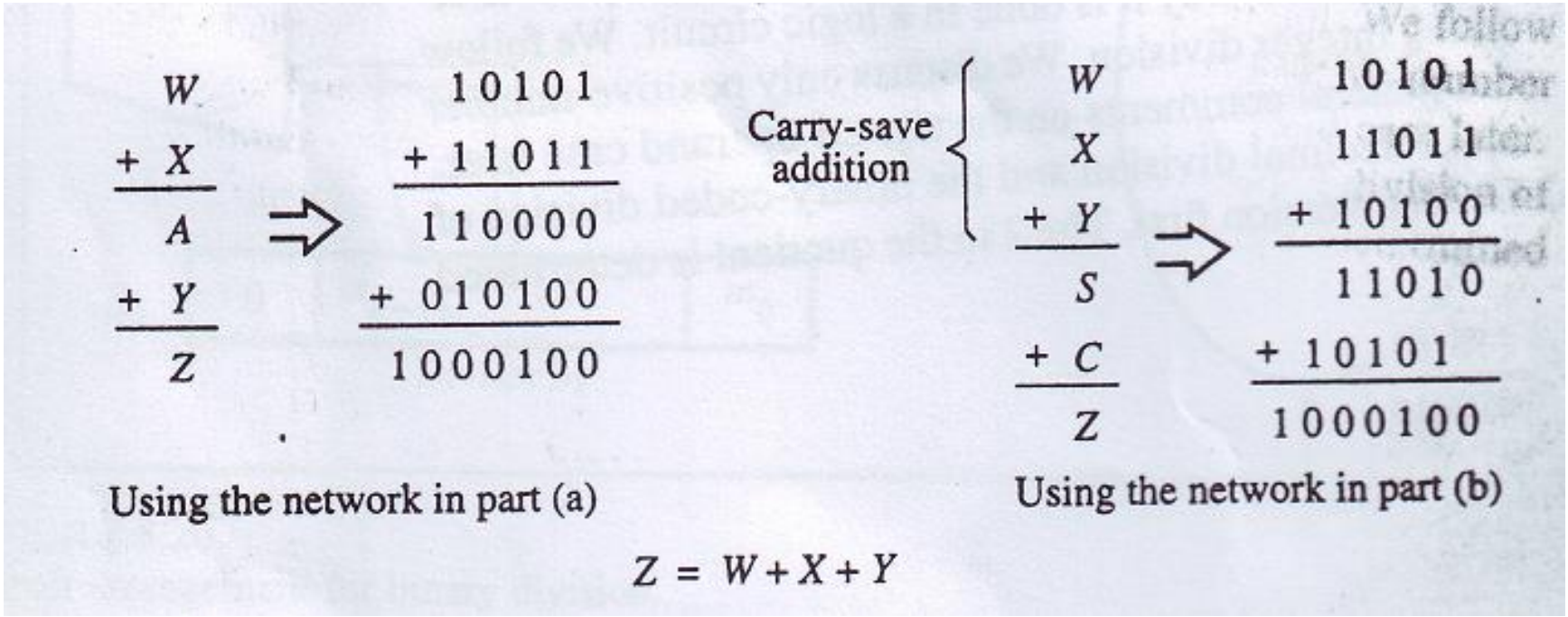
(a) $W + X + Y$ using two ripple-carry adders

Carry Save Addition



(b) $W + X + Y$ using carry-save addition

Carry Save Addition



Carry Save Addition

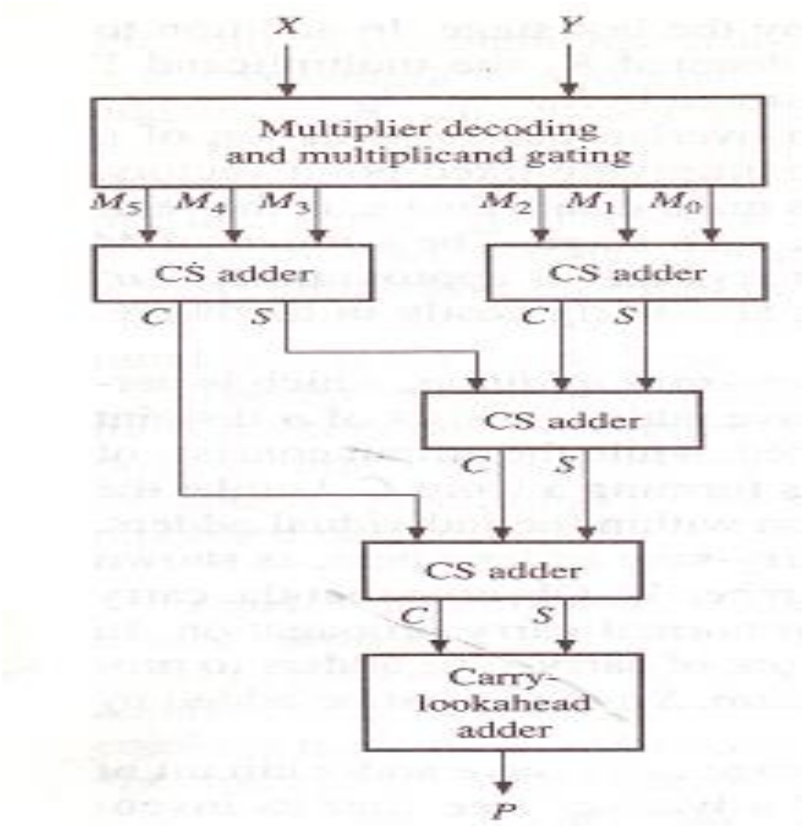
Advantages:

- ✔ All bits of *S* and *C* vectors are produced in a short, fixed amount of time.
- ✔ Carry propagation takes place only in the second row.
- ✔ Since all bits of *S* and *C* are available in parallel, a carry-lookahead adder can be used effectively to add the *S* and *C*.

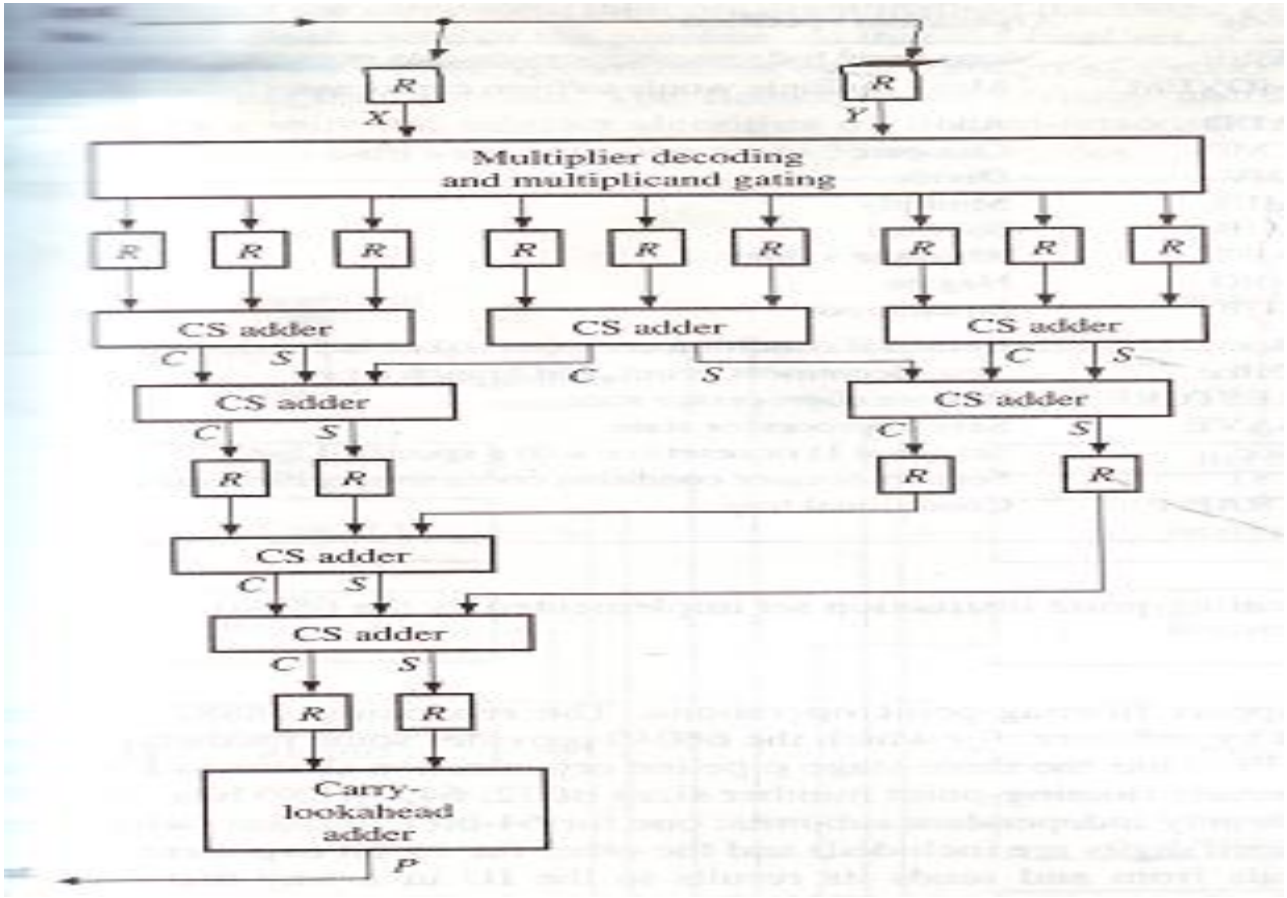
Multiplication Using Carry Save Adder

- ✔ Multiplication can be performed using a multi-stage carry-save adder circuit.
- ✔ The inputs are n-terms of the form $M_i = x_i Y 2^k$.
- ✔ The desired product is $\sum M_i$.
- ✔ In pipelined carry save multiplier the complete operation requires four full-adder unit delays to do the carry-save additions, followed by a full-addition operation on the final 2 vectors.
- ✔ When the number of M_i is large, the time saved is proportionally greater but the number of carry-save adders required can be excessive.

A Carry-Save (Wallace Tree) Multiplier



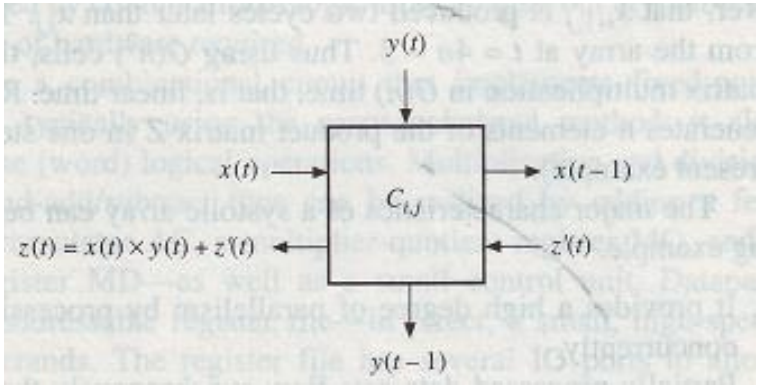
A Pipelined Carry-Save Multiplier



Systolic Array

- ✔ It is formed by interconnecting a set of identical data-processing cells in uniform manner.
- ✔ Data words flow synchronously from cell to cell and each cell performs a small step in the overall operation of the array.
- ✔ It permits data to flow through the cells in several directions at once.
- ✔ One dimensional systolic array is therefore a kind of pipeline with identical stages.
- ✔ It can used to implement various complex arithmetic operations such as matrix multiplication

Two-dimensional Systolic Array Performing Matrix Multiplication



Let X be an $n \times n$ matrix of fixed-point or floating-point numbers defined by

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots x_{1,n} \\ x_{2,1} & x_{2,2} & \dots x_{2,n} \\ & \dots & \\ x_{n,1} & x_{n,2} & \dots x_{n,n} \end{bmatrix}$$

$$z_{i,j} = \sum_{k=1}^{n} x_{i,k} \times y_{k,j}$$

Two-dimensional Systolic Array Performing Matrix Multiplication

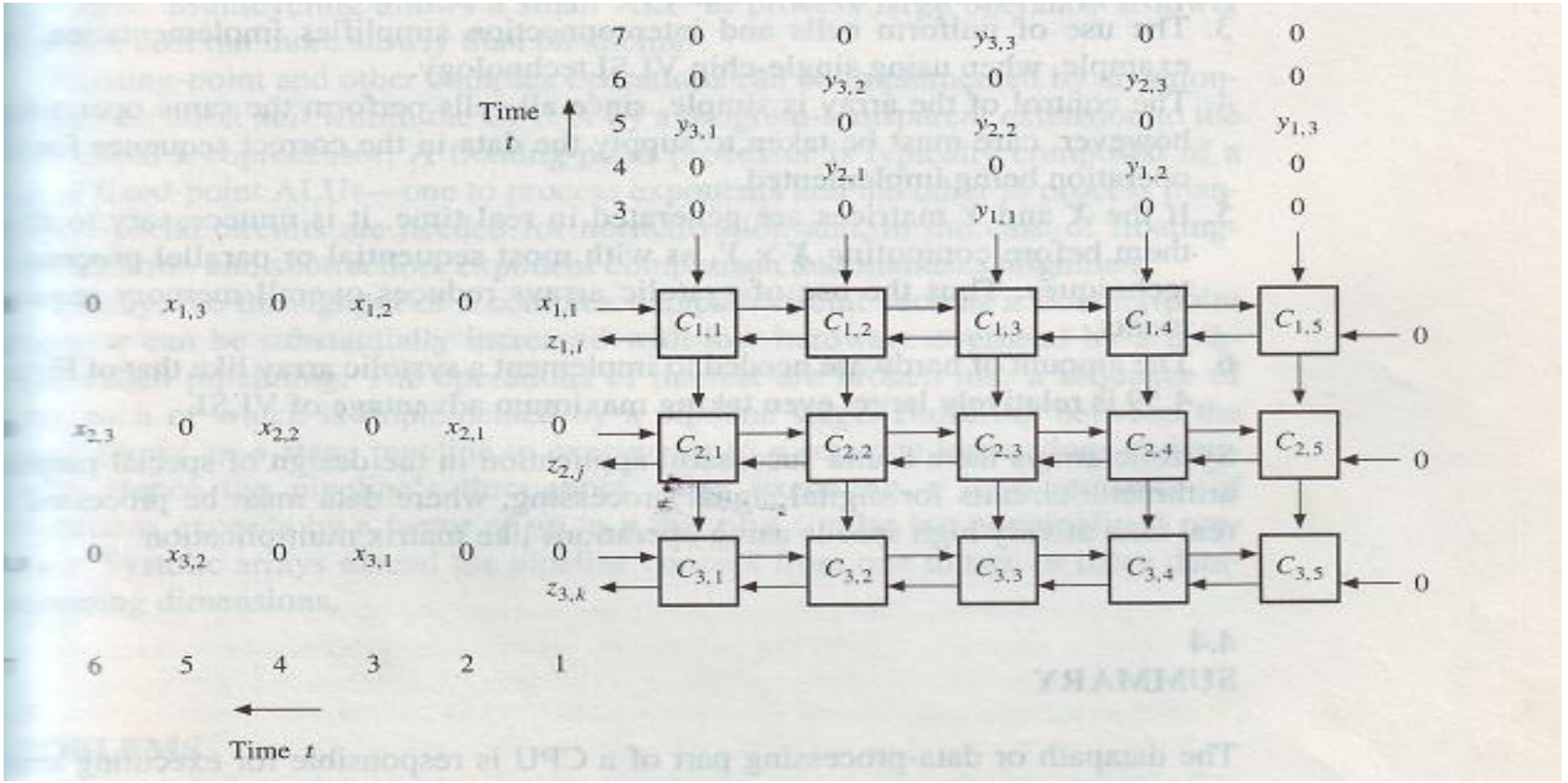
$$\begin{bmatrix} z11 & z12 & z13 \\ z21 & z22 & z23 \\ z31 & z32 & z33 \end{bmatrix} = \begin{bmatrix} x11 & x12 & x13 \\ x21 & x22 & x23 \\ x31 & x32 & x33 \end{bmatrix} * \begin{bmatrix} y11 & y12 & y13 \\ y21 & y22 & y23 \\ y31 & y32 & y33 \end{bmatrix}$$

$$z_{11} = x_{11}y_{11} + x_{12}y_{21} + x_{13}y_{31}$$

$$Z_{12} = x_{11}y_{12} + x_{12}y_{22} + x_{13}y_{32}$$

$$Z_{13} = x_{11}y_{13} + x_{12}y_{23} + x_{13}y_{33}$$

Two-dimensional Systolic Array Performing Matrix Multiplication



Two-dimensional Systolic Array Performing Matrix Multiplication

- ✔ The systolic matrix multiplier is constructed from n(2n-1) copies of $C_{i,j}$.
- ✔ The x and y operands are carefully ordered and separated by 0 so that the specific operand pairs $x_{i,k}$ and $y_{k,j}$ meet at an appropriate cell of the array.
- ✔ The z's emerge from the left side of $C_{i,j}$.
- ✔ Each row of cells eventually issues the corresponding row of the matrix product Z from its left side.

Major Characteristics of a Systolic Array

1. Provides a higher degree of parallelism.
2. Partially processed data flow synchronously through the array in pipeline fashion but in several directions at once, with complete results eventually appearing at the array boundary.
3. The use of uniform cells and interconnection simplifies the implementation, when using single chip.
4. The control is simple. But care must be taken to supply the data in the correct sequence.
5. X and Y need not to store before computing X × Y.
6. The amount of hardware needed to implement is relatively large.