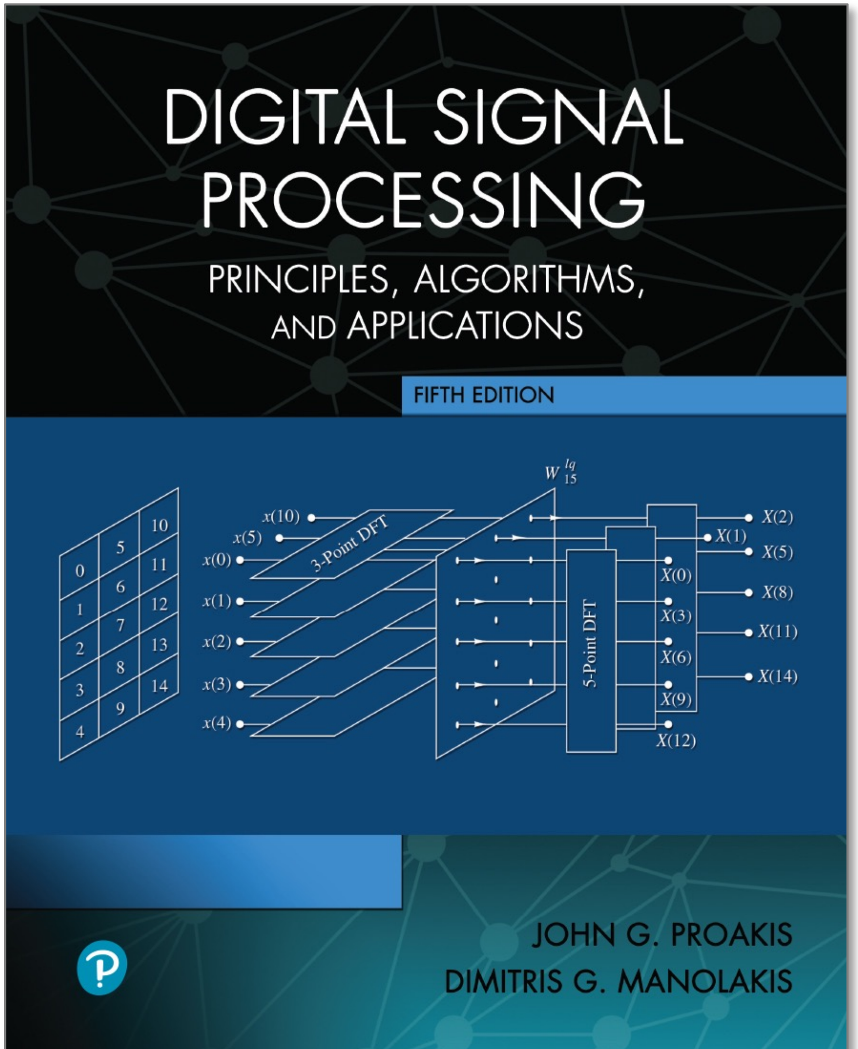
Digital Signal Processing

Fifth Edition



Chapter 8

Efficient Computation of the DFT: Fast Fourier Transform Algorithms

Figure 8.1.1 Two dimensional data array for storing the sequence $x(n)$, $0 \le n \le N-1$.

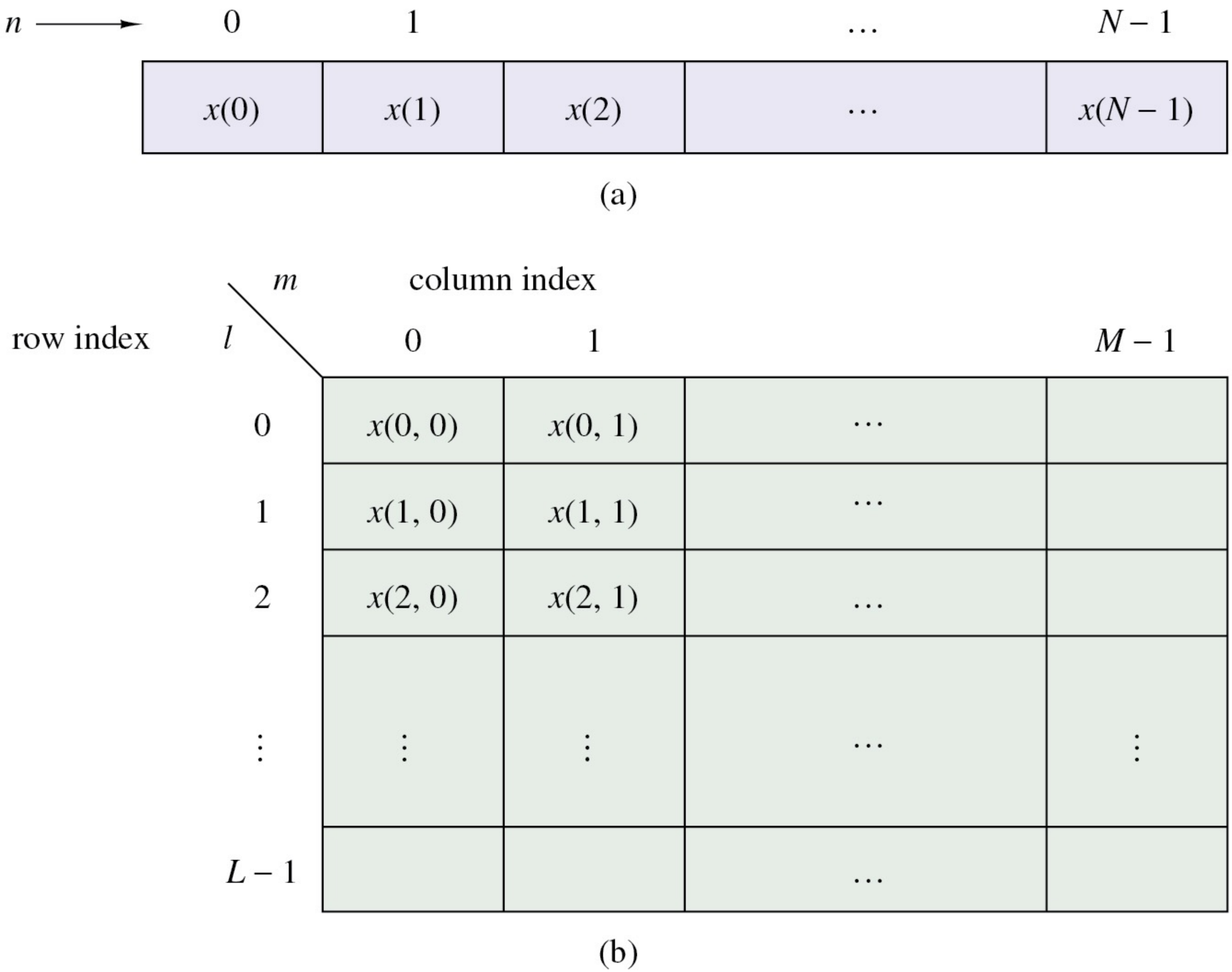


Figure 8.1.2 Two arrangements for the data arrays.

Row-wise $n = Ml + m$

$l \backslash m$	0	1	2		$M-1$
0	$x(0)$	$x(1)$	$x(2)$	$\cdots$	$x(M-1)$
1	$x(M)$	$x(M+1)$	$x(M+2)$	$\cdots$	$x(2M-1)$
2	$x(2M)$	$x(2M+1)$	$x(2M+2)$	$\cdots$	$x(3M-1)$
	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
$L-1$	$x((L-1)M)$	$x((L-1)M+1)$	$x((L-1)M+2)$	$\cdots$	$x(LM-1)$

(a)

Column-wise $n = l + mL$

$l \backslash m$	0	1	2		$M-1$
0	$x(0)$	$x(L)$	$x(2L)$	$\cdots$	$x((M-1)L)$
1	$x(1)$	$x(L+1)$	$x(2L+1)$	$\cdots$	$x((M-1)L+1)$
2	$x(2)$	$x(L+2)$	$x(2L+2)$	$\cdots$	$x((M-1)L+2)$
	$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$
$L-1$	$x(L-1)$	$x(2L-1)$	$x(3L-1)$	$\cdots$	$x(LM-1)$

(b)

Figure 8.1.3 Computation of N = 15-point DFT by means of 3-point and 5-point DFTs.

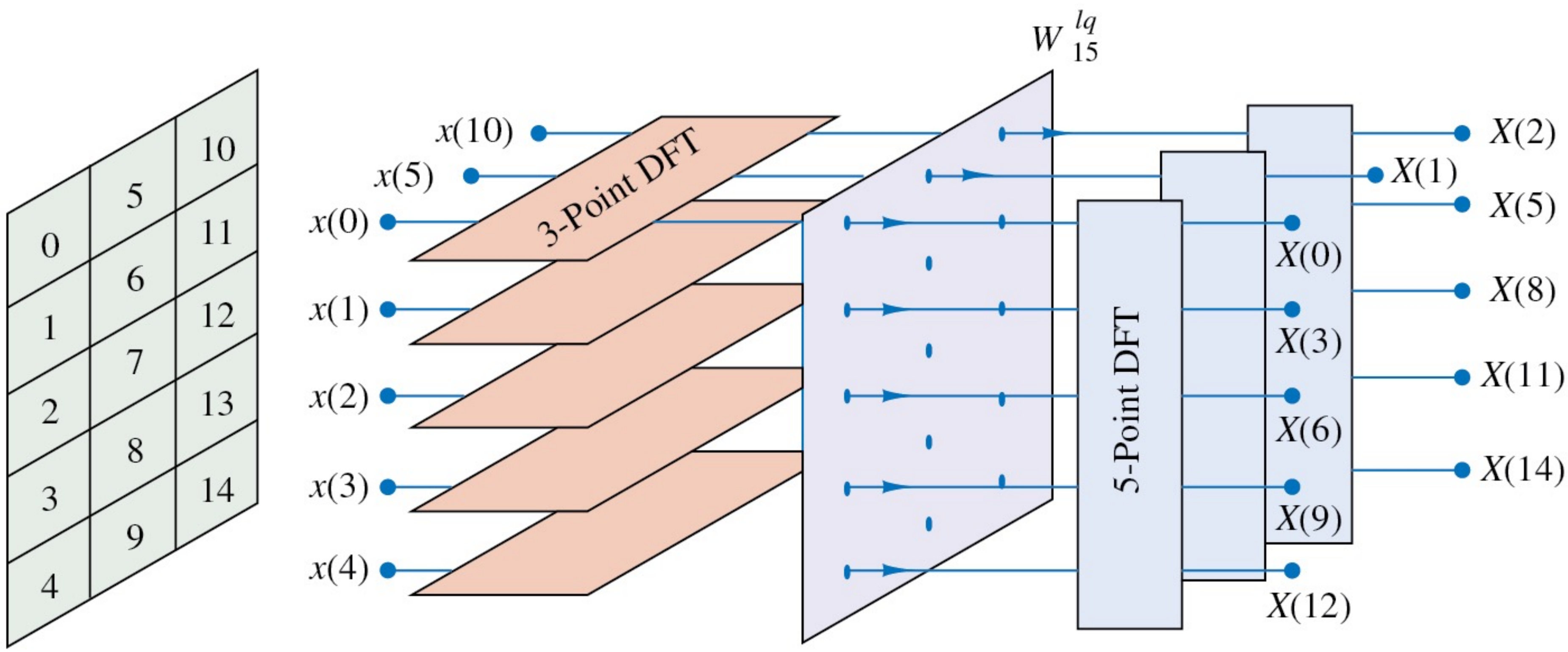


Figure 8.1.4 First step in the decimation-in-time algorithm.

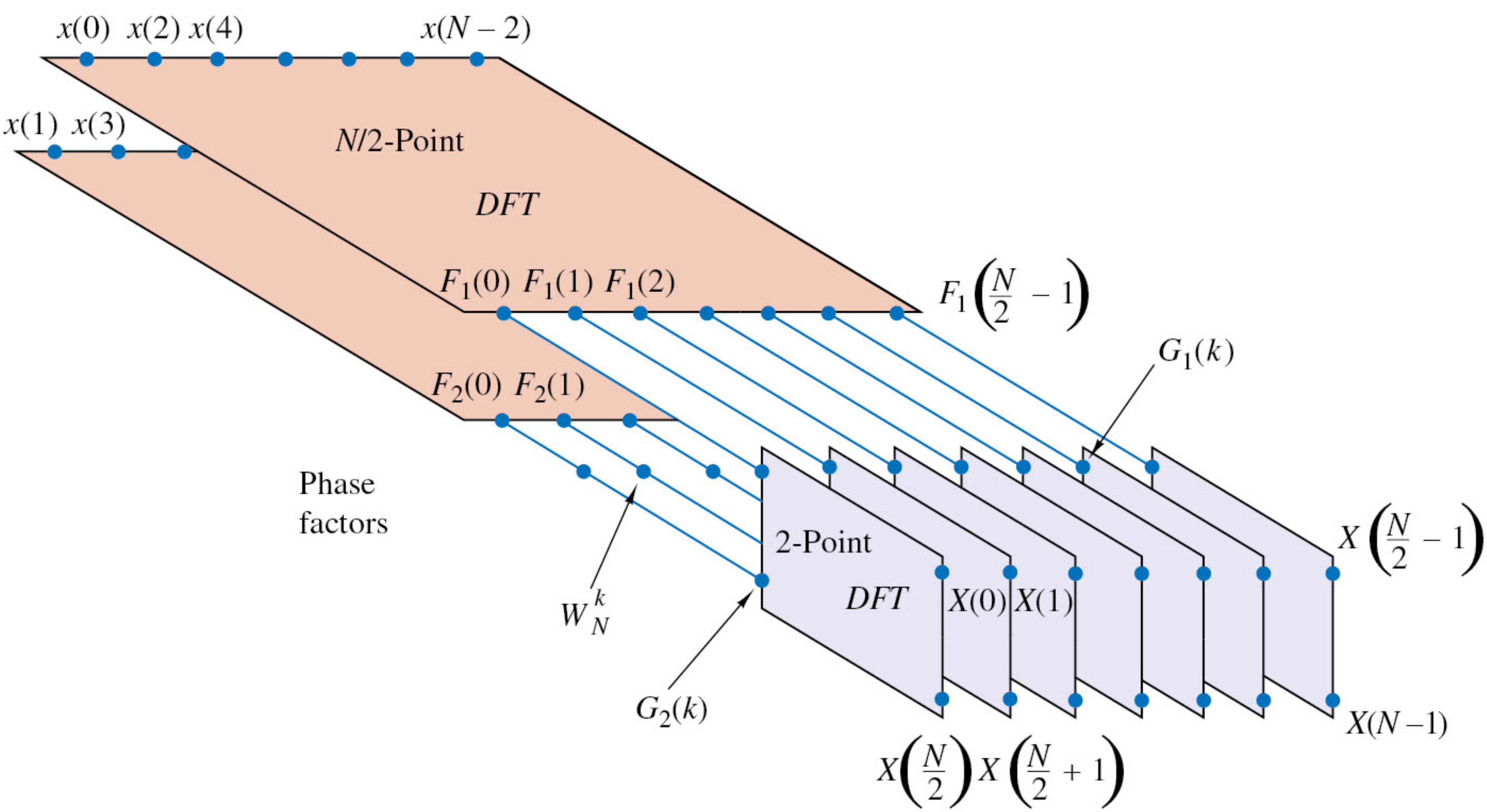


Table 8.1 Comparison of Computational Complexity for the Direct Computation of the DFT Versus the FFT Algorithm

Number of Points, N	Complex Multiplications in Direct Computation, N^2	Complex Multiplications in FFT Algorithm, $(N/2)\log_2 N$	Speed Improvement Factor
4	16	4	4.0
8	64	12	5.3
16	256	32	8.0
32	1,024	80	12.8
64	4,096	192	21.3
128	16,384	448	36.6
256	65,536	1,024	64.0
512	262,144	2,304	113.8
1,024	1,048,576	5,120	204.8

Figure 8.1.5 Three stages in the computation of an N = 8-point DFT.

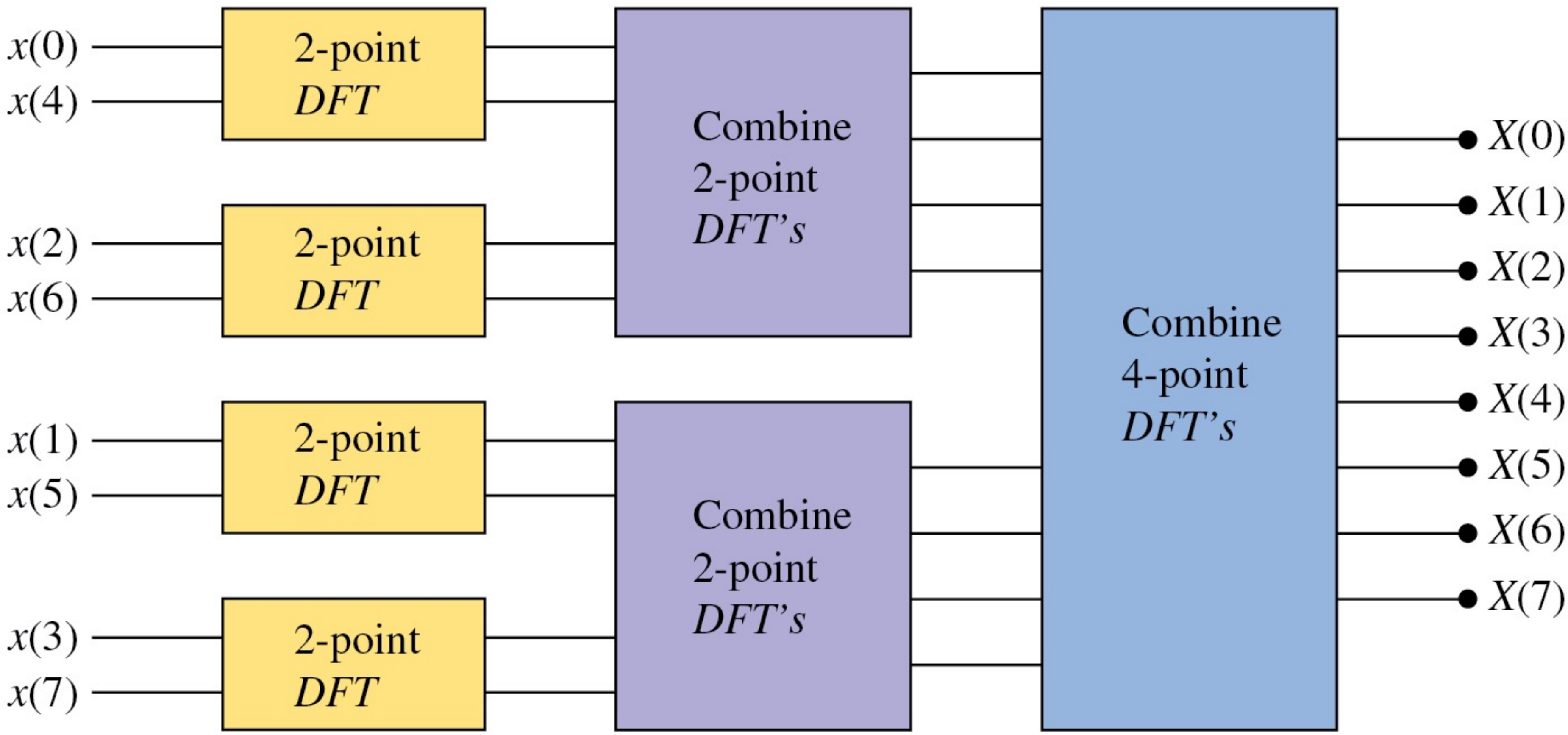
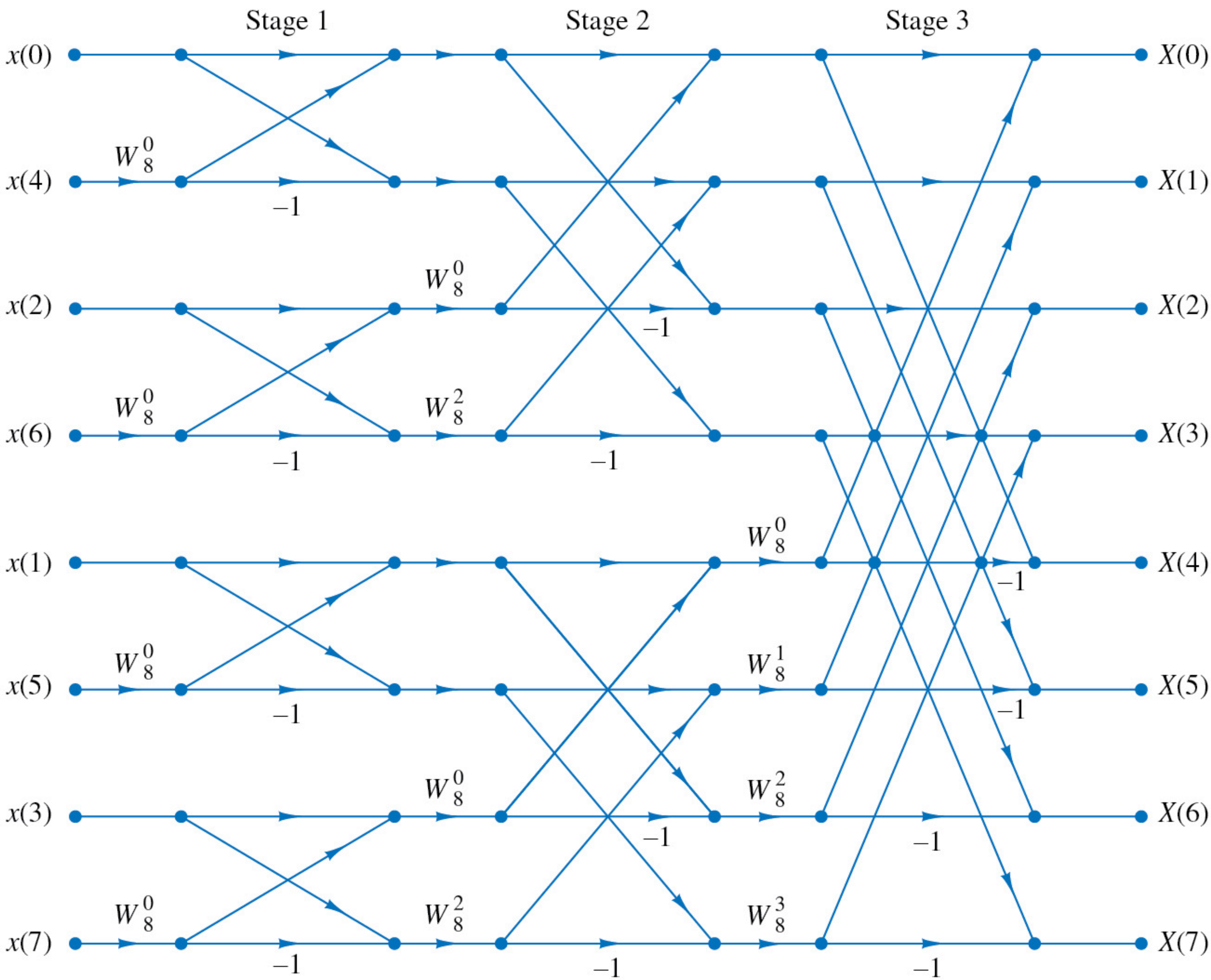


Figure 8.1.6 Eight-point decimation-in-time FFT algorithm.



Pearson

Figure 8.1.7 Basic butterfly computation in the decimation-in-time FFT algorithm.

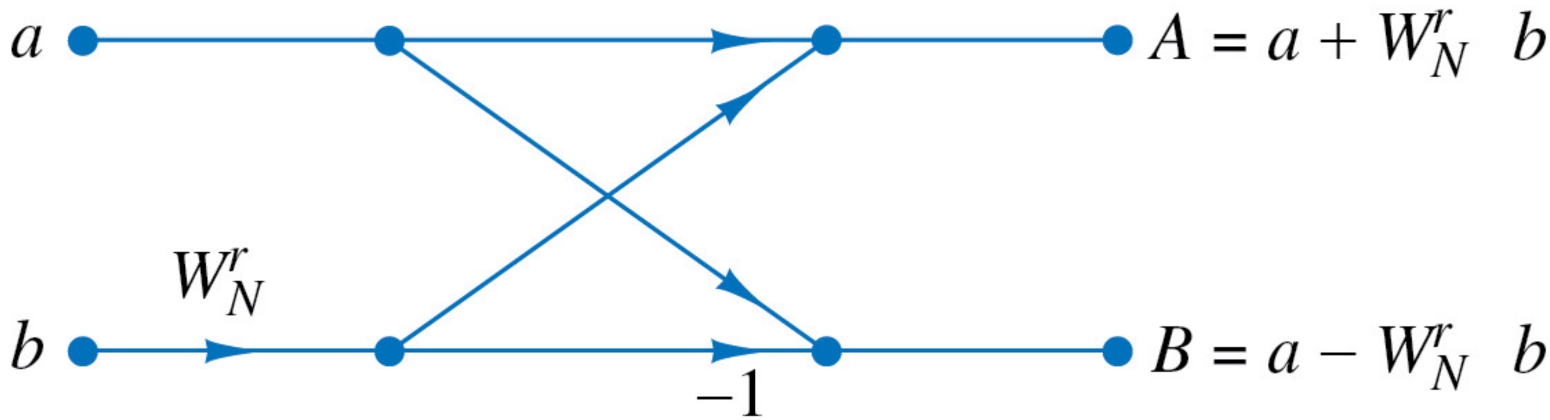
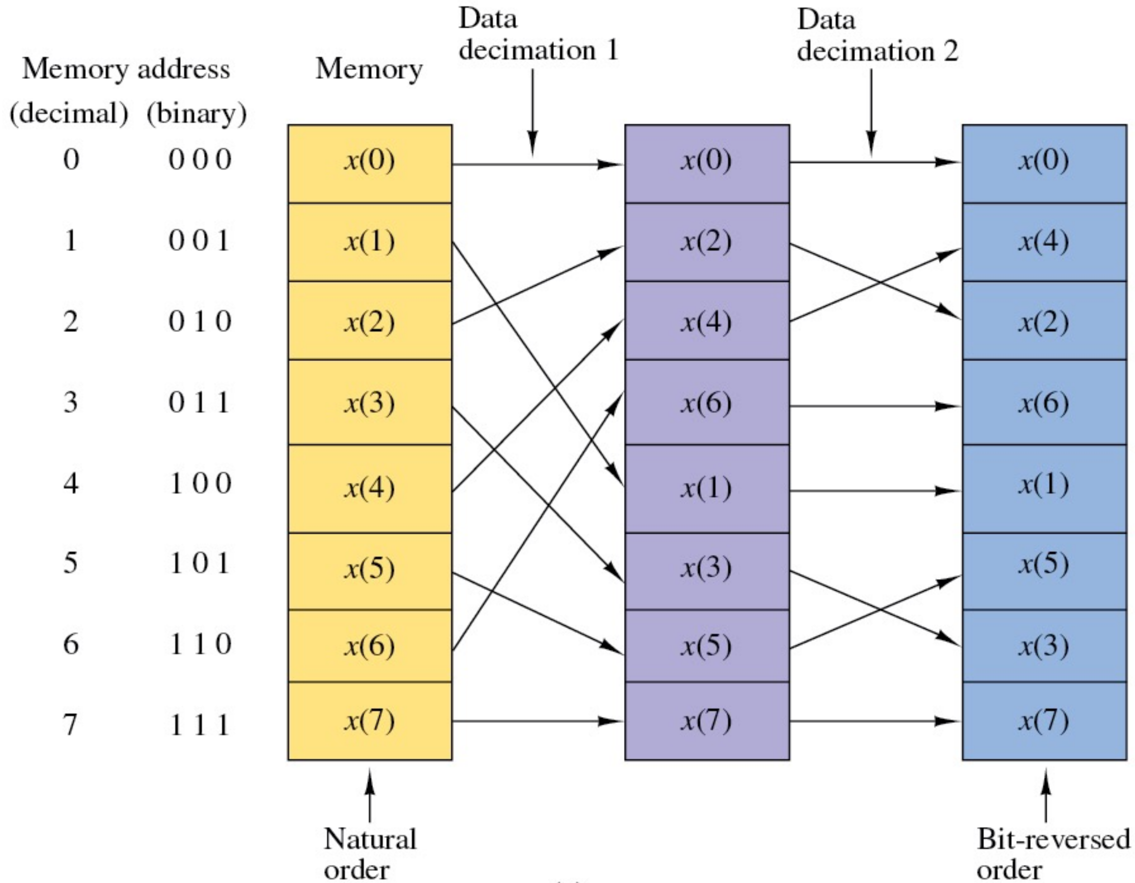


Figure 8.1.8 Shuffling of the data and bit reversal.



(a)

$(n_2 n_1 n_0)$	$\rightarrow$	$(n_1 n_0 n_2)$	$\rightarrow$	$(n_0 n_1 n_2)$
(0 0 0)	$\rightarrow$	(0 0 0)	$\rightarrow$	(0 0 0)
(0 0 1)	$\rightarrow$	(0 1 0)	$\rightarrow$	(1 0 0)
(0 1 0)	$\rightarrow$	(1 0 0)	$\rightarrow$	(0 1 0)
(0 1 1)	$\rightarrow$	(1 1 0)	$\rightarrow$	(1 1 0)
(1 0 0)	$\rightarrow$	(0 0 1)	$\rightarrow$	(0 0 1)
(1 0 1)	$\rightarrow$	(0 1 1)	$\rightarrow$	(1 0 1)
(1 1 0)	$\rightarrow$	(1 0 1)	$\rightarrow$	(0 1 1)
(1 1 1)	$\rightarrow$	(1 1 1)	$\rightarrow$	(1 1 1)

(b)

Figure 8.1.9 First stage of the decimation-in-frequency FFT algorithm.

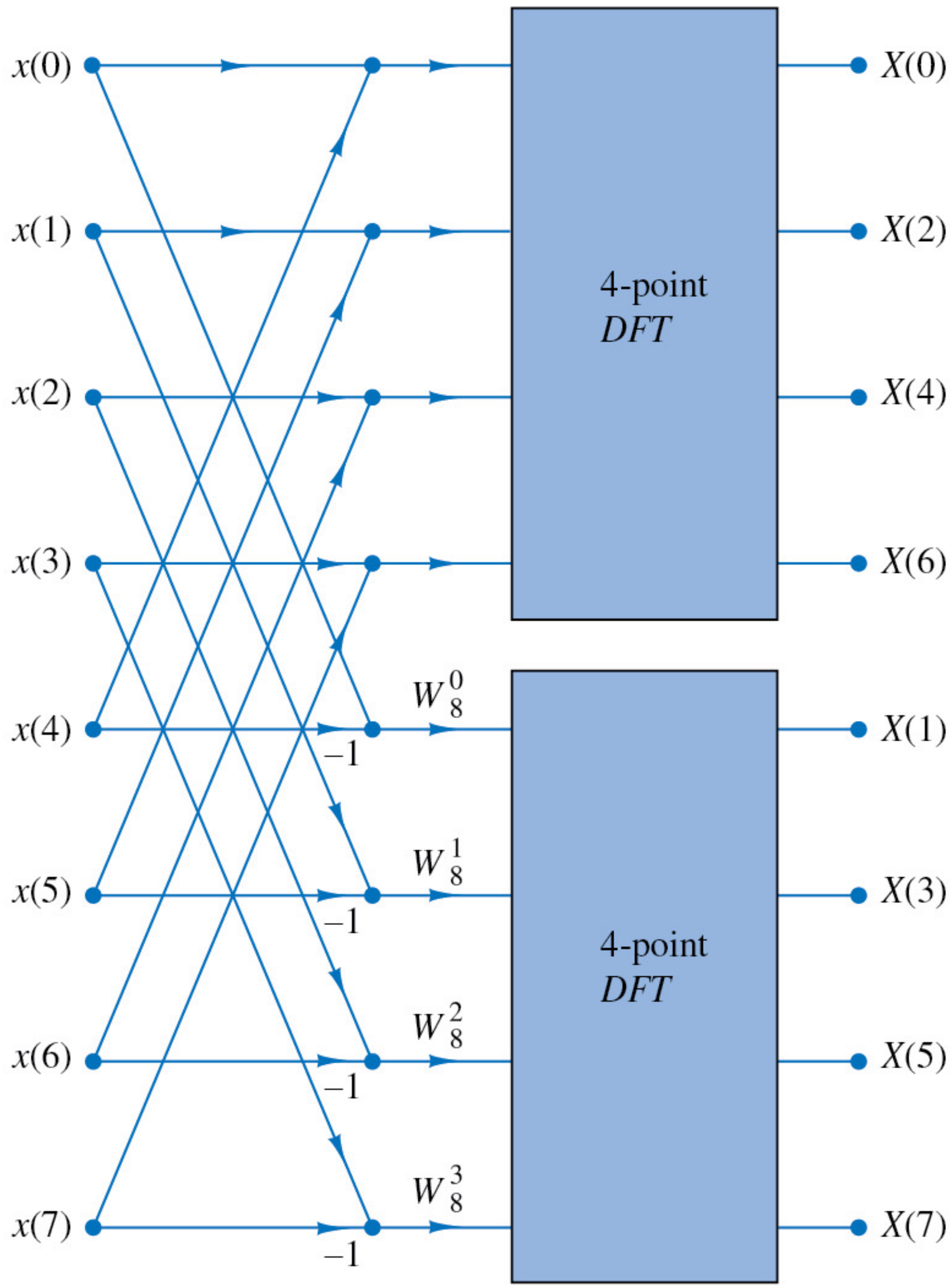


Figure 8.1.10 Basic butterfly computation in the decimation-in-frequency FFT algorithm.

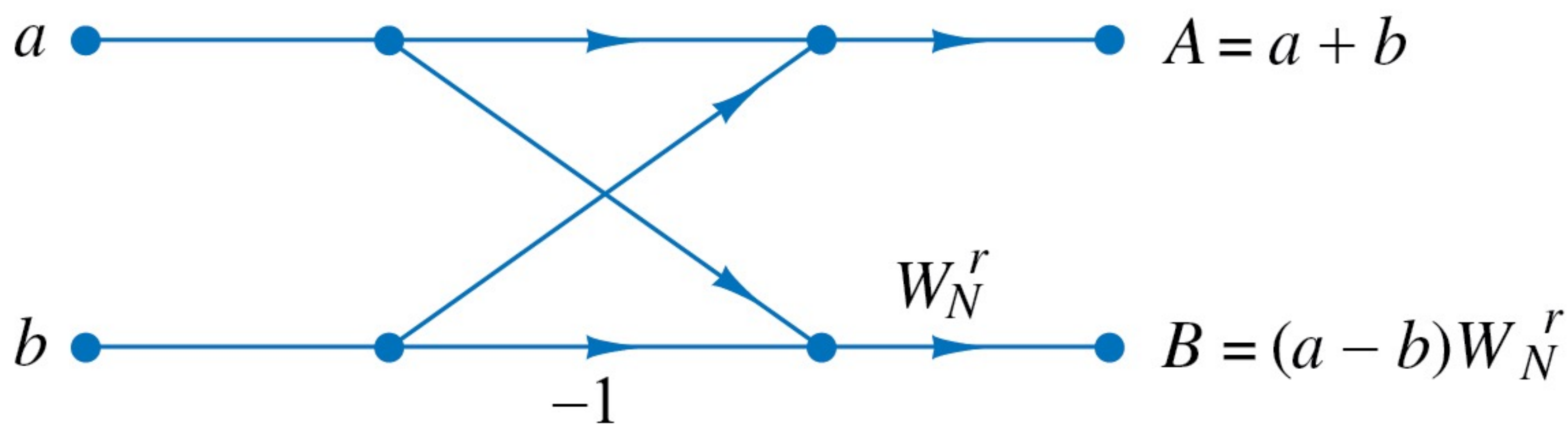


Figure 8.1.11 N = 8-point decimation-in-frequency FFT algorithm.

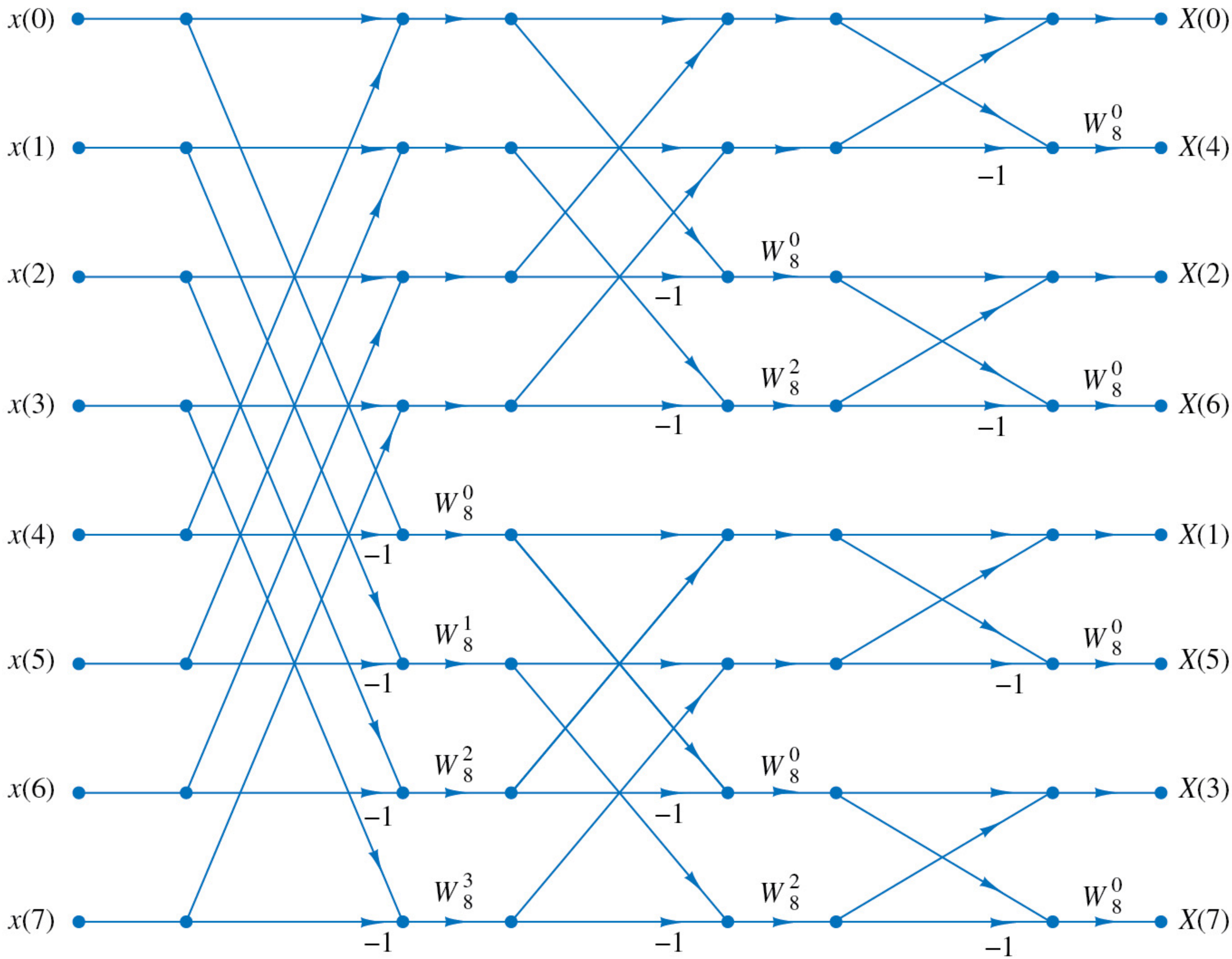


Figure 8.1.12 Basic butterfly computation in a radix-4 FFT algorithm.

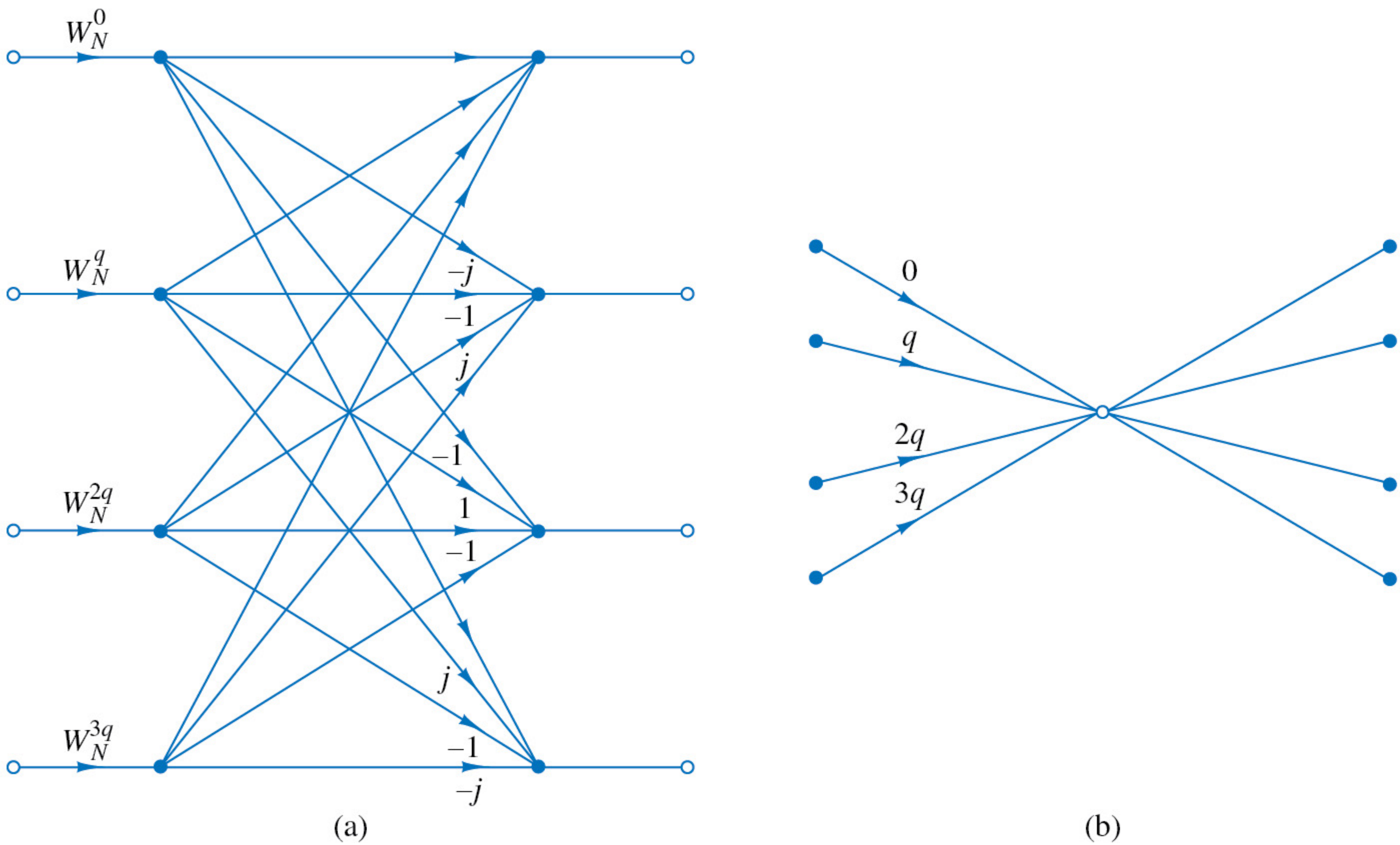


Figure 8.1.13 Sixteen-point radix-4 decimation-in-time algorithm with input in normal order and output in digit-reversed order. The integer multipliers shown on the graph represent the exponents on W_{16}.

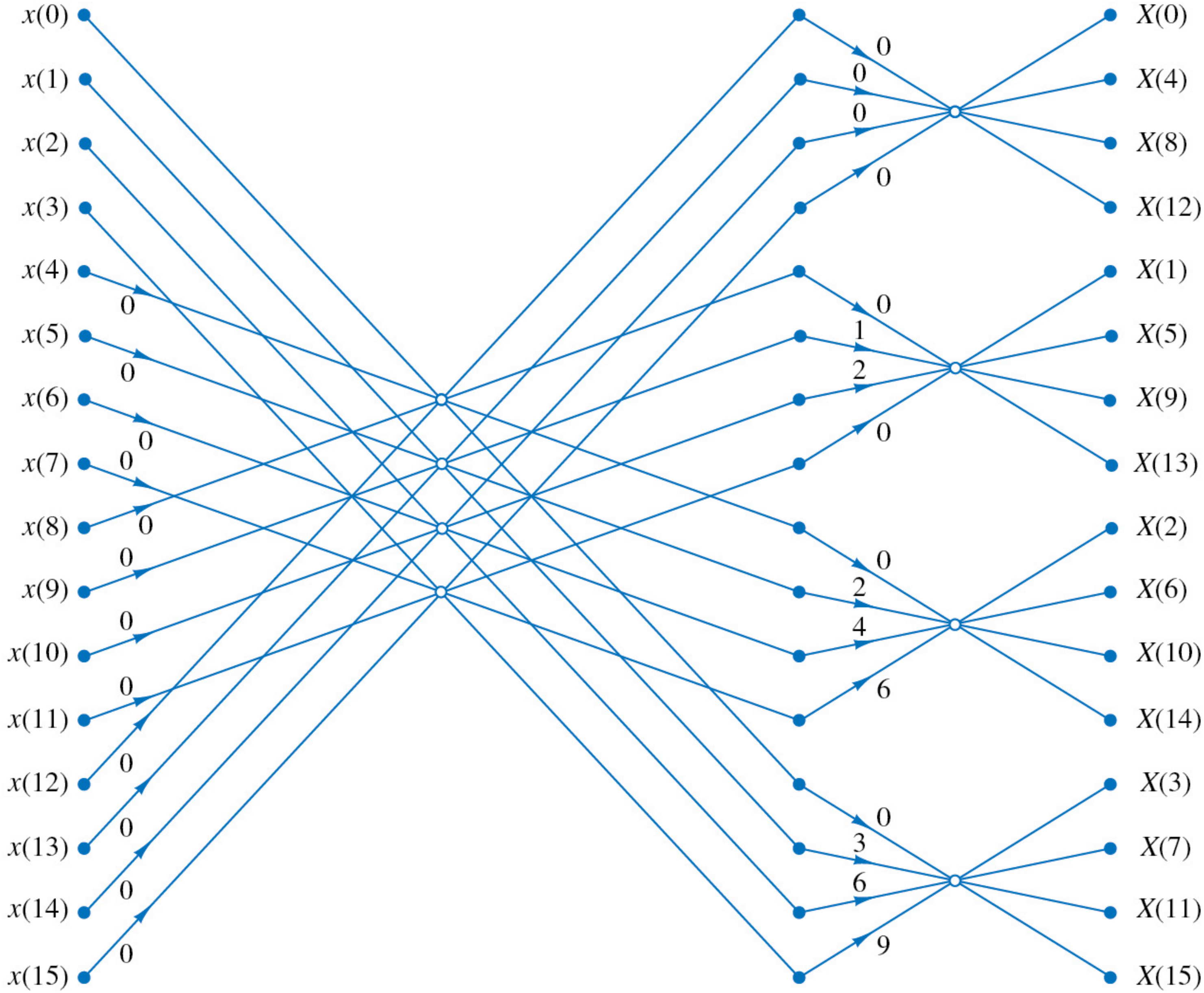


Figure 8.1.14 Sixteen-point, radix-4 decimation-in-frequency algorithm with input in normal order and output in digit-reversed order.

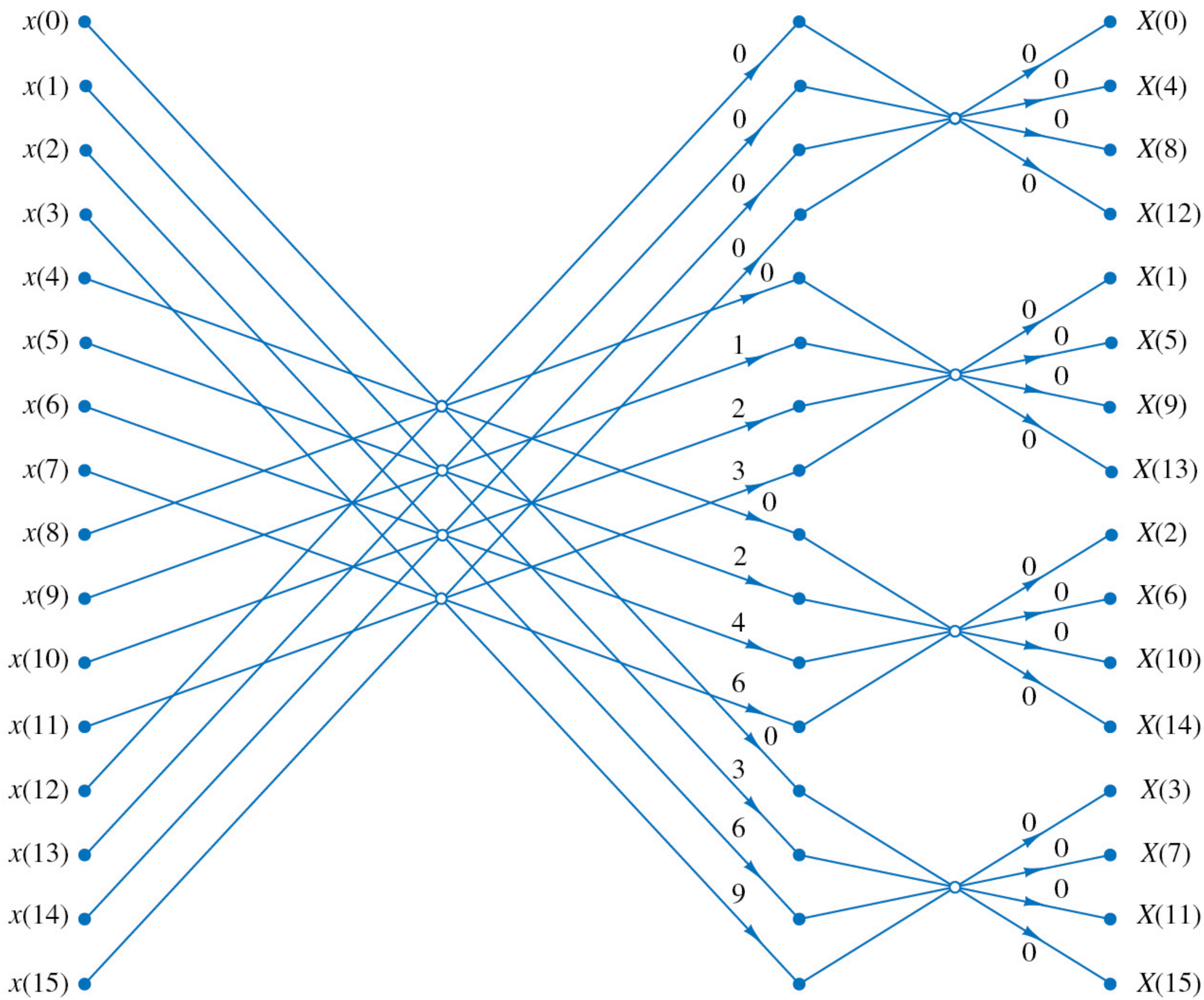


Figure 8.1.15 Length 32 split-radix FFT algorithms from paper by Duhamel (1986); reprinted with permission from the IEEE.

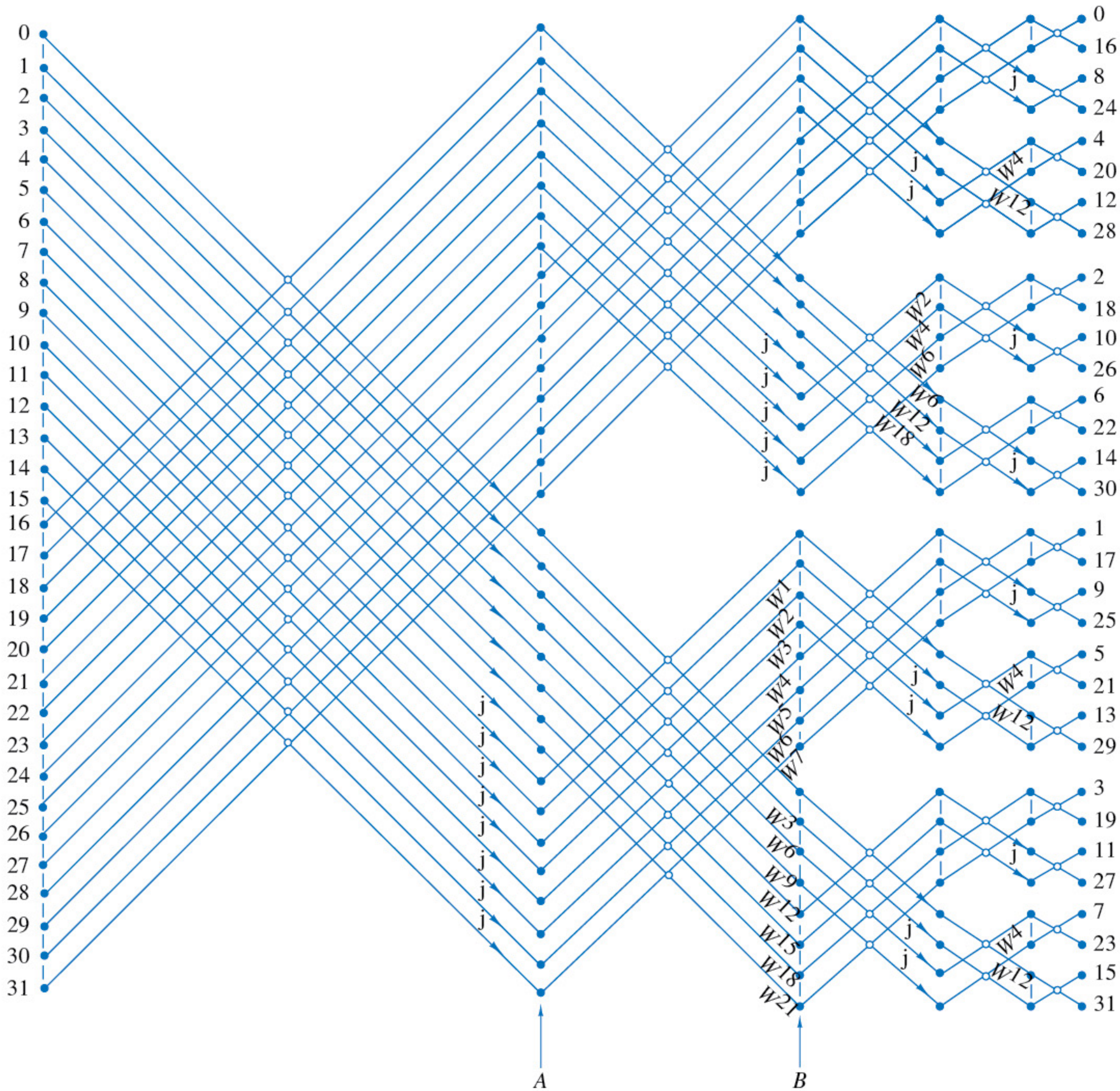


Figure 8.1.16 Butterfly for SRFFT algorithm.

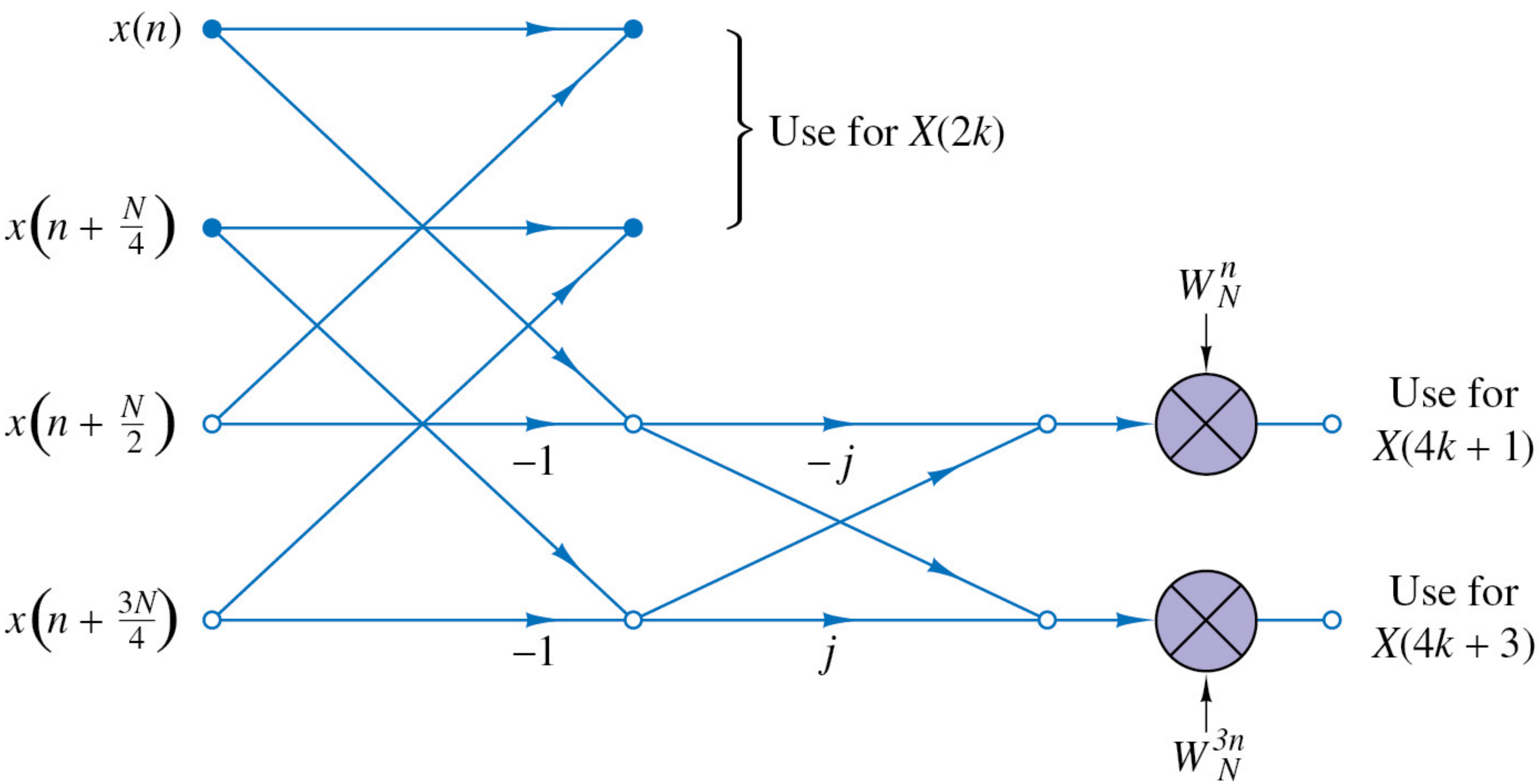


Table 8.2 Number of Nontrivial Real Multiplications and Additions to Compute an *N*-point Complex DFT

	Real Multiplications				Real Additions			
N	Radix 2	Radix 4	Radix 8	Split Radix	Radix 2	Radix 4	Radix 8	Split Radix
16	24	20		20	152	148		148
32	88			68	408			388
64	264	208	204	196	1,032	976	972	964
128	712			516	2,504			2,308
256	1,800	1,392		1,284	5,896	5,488		5,380
512	4,360		3,204	3,076	13,566		12,420	12,292
1,024	10,248	7,856		7,172	30,728	28,336		27,652

Source: Extracted from Duhamel (1986).

Table 8.3 Computational Complexity

Size of FFT $\nu = \log_2 N$	$c(\nu)$ Number of Complex Multiplications per Output Point
9	13.3
10	12.6
11	12.8
12	13.4
14	15.1

Figure 8.3.1 Direct form II realization of two-pole resonator for computing the DFT.

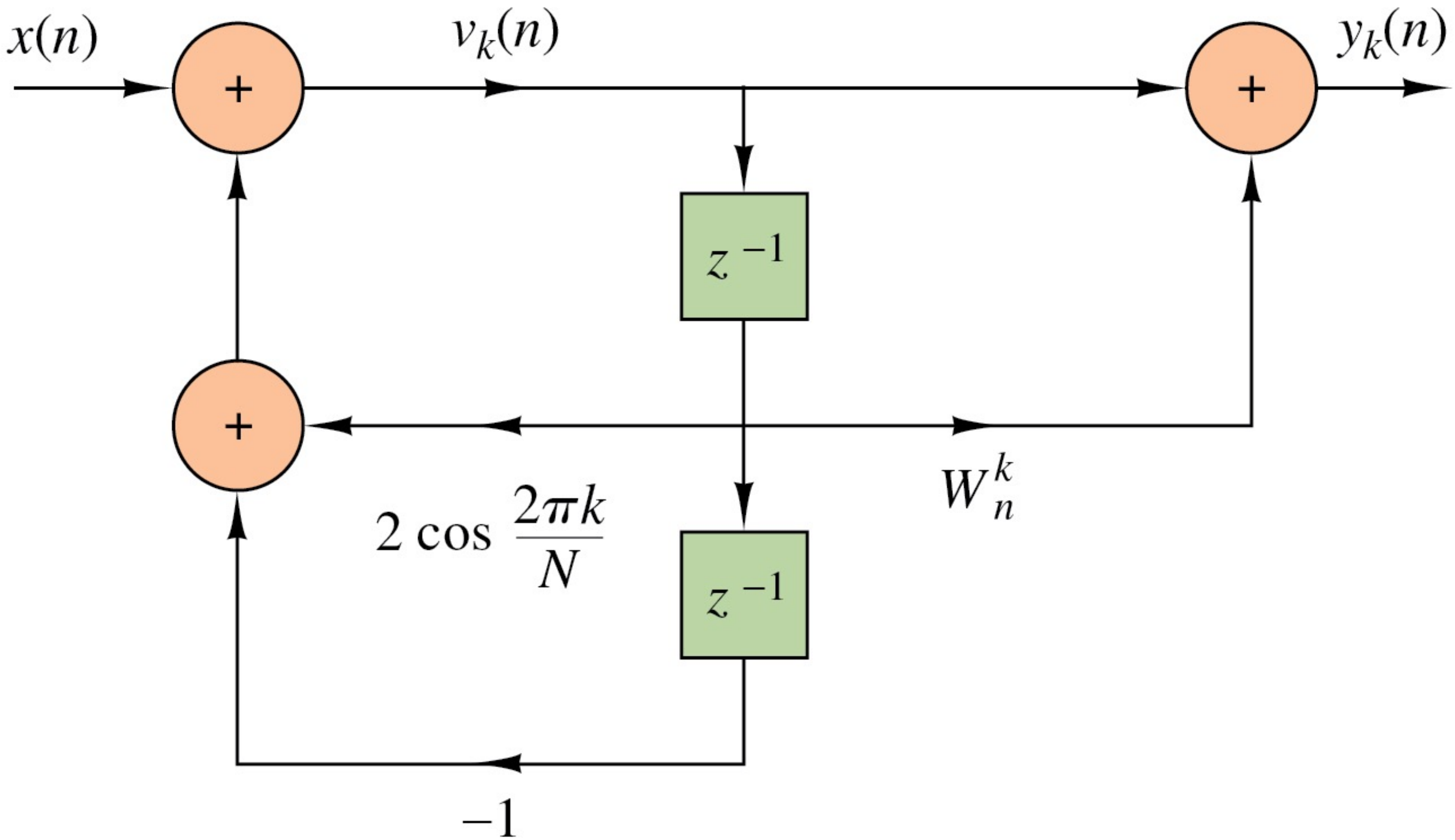
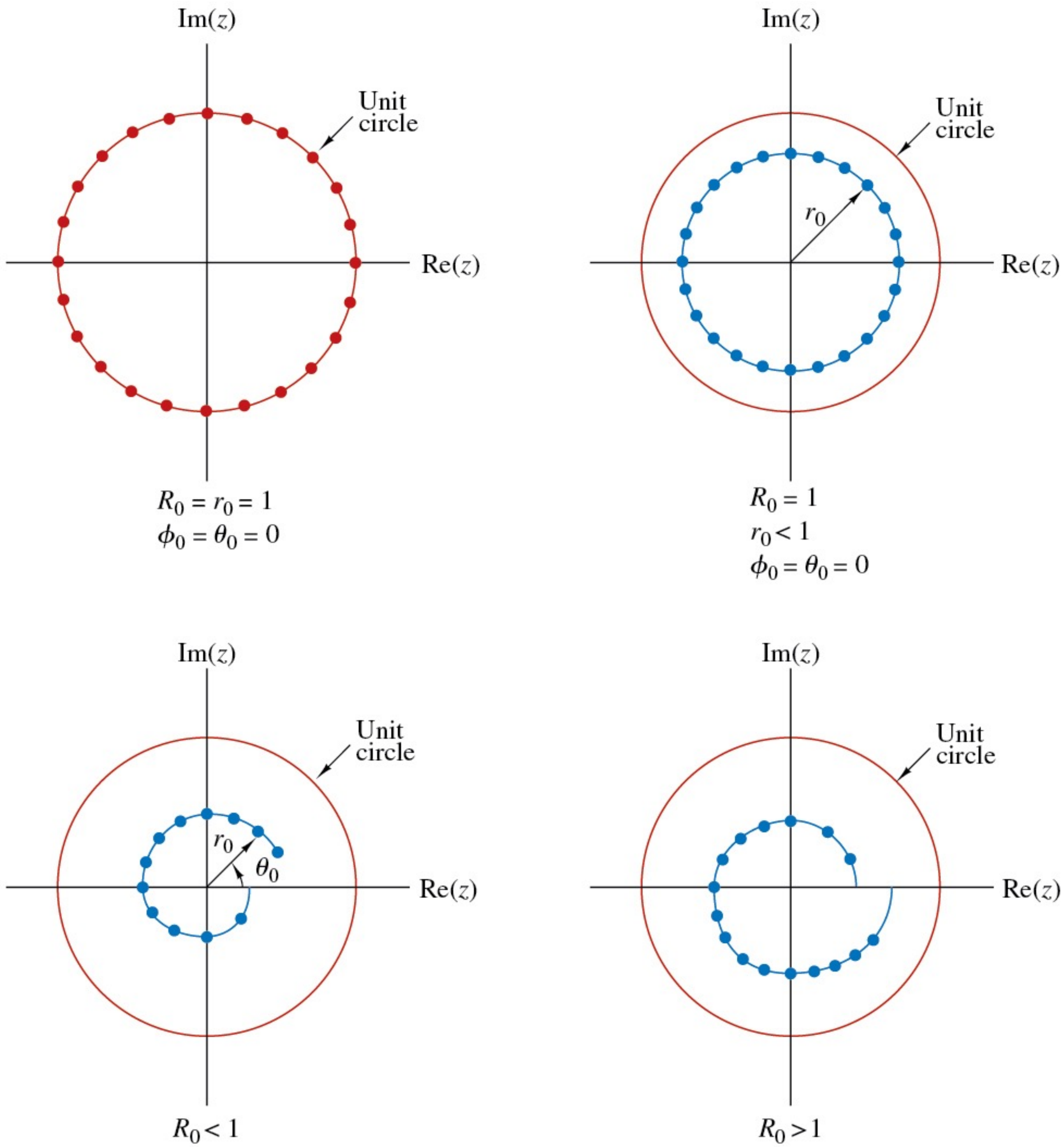


Figure 8.3.2 Some examples of contours on which we may evaluate the z-transform.



Pearson

Figure 8.3.3 Block diagram illustrating the implementation of the chirp-z transform for computing the DFT (magnitude only).

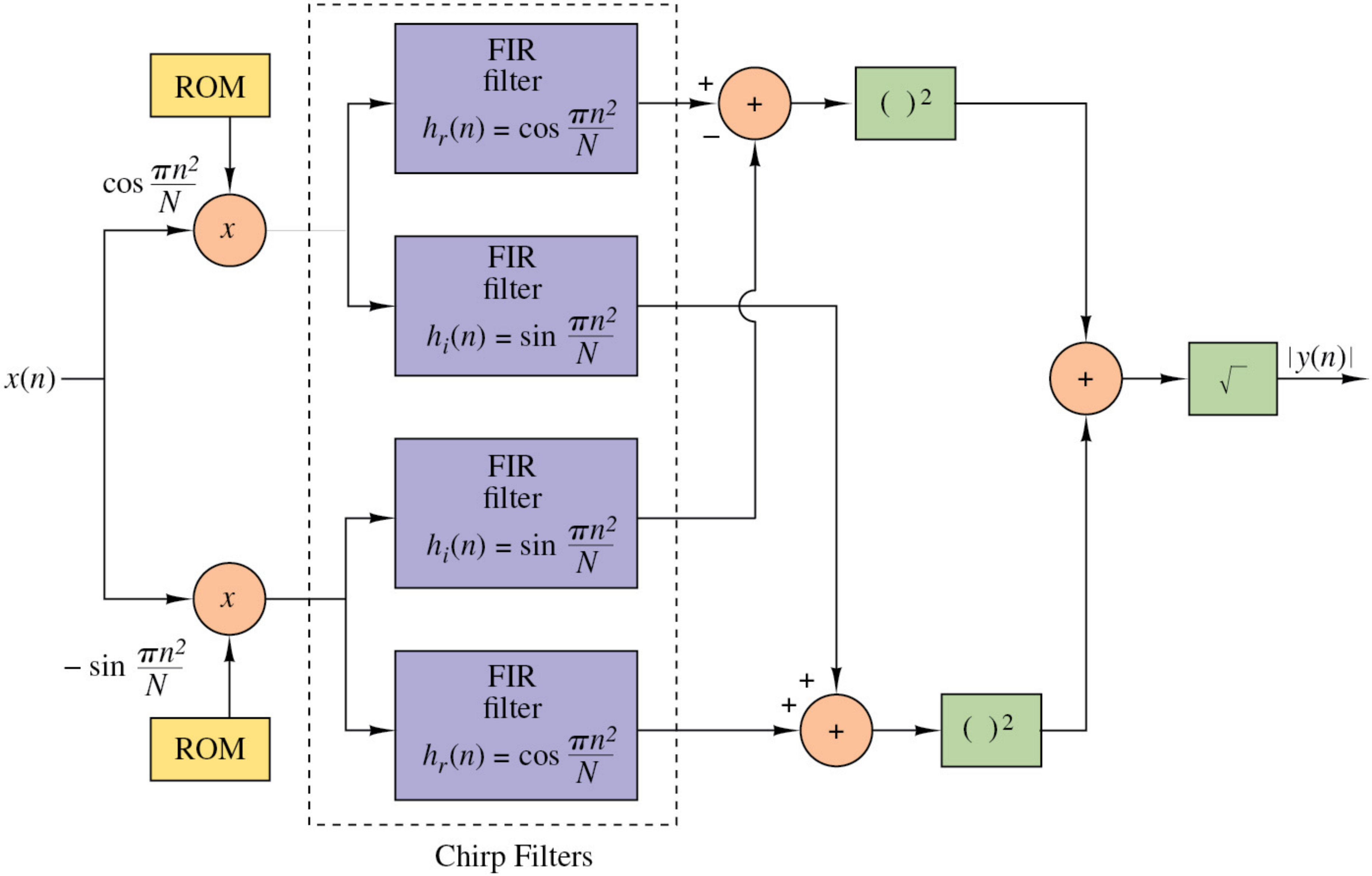


Figure 8.4.1 Decimation-in-time FFT algorithm.

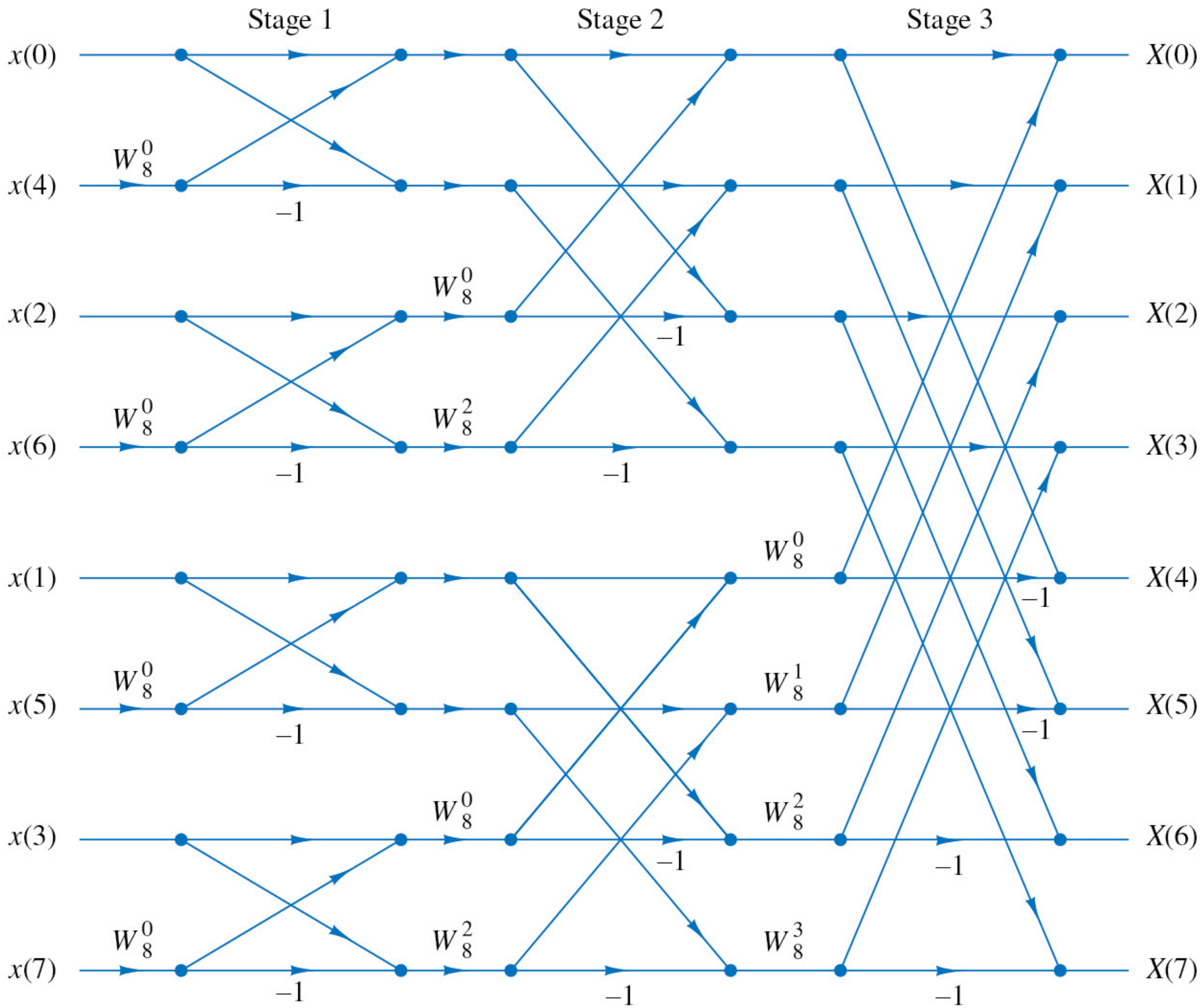
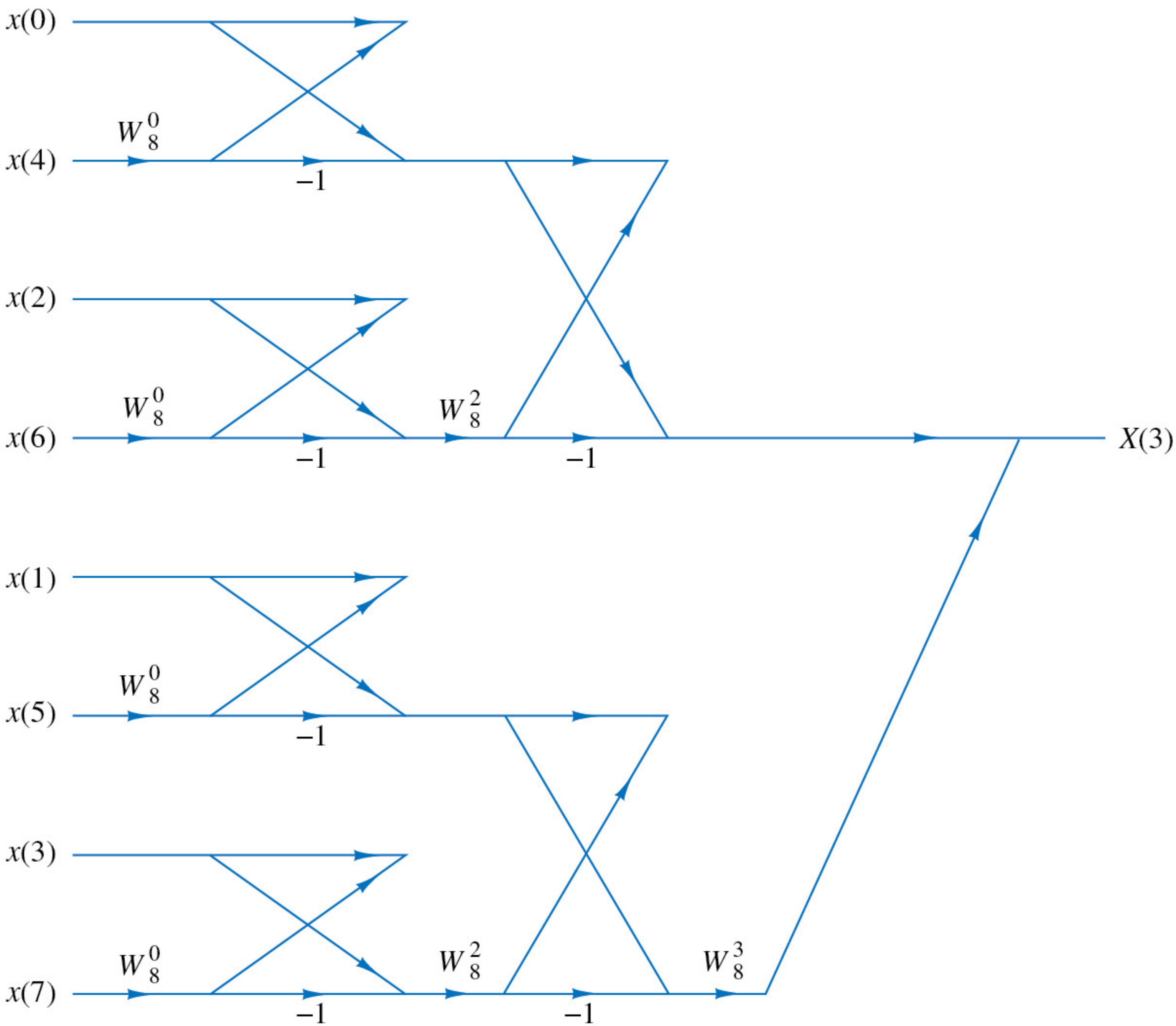


Figure 8.4.2 Butterflies that affect the computation of $X(3)$.



Copyright