# 第四章 快速傅里叶变换

*Fast Forurier Transform*

# 第四章　快速傅里叶变换

*Fast Forurier Transform*

## 4.4  快速傅里叶反变换的实现方法

华东理工大学信息科学与工程学院  万永菁

FFT算法流图也可以用于离散傅里叶逆变换(Inverse Discrete Fourier Transform，简称IDFT)。

比较DFT和IDFT的运算公式：

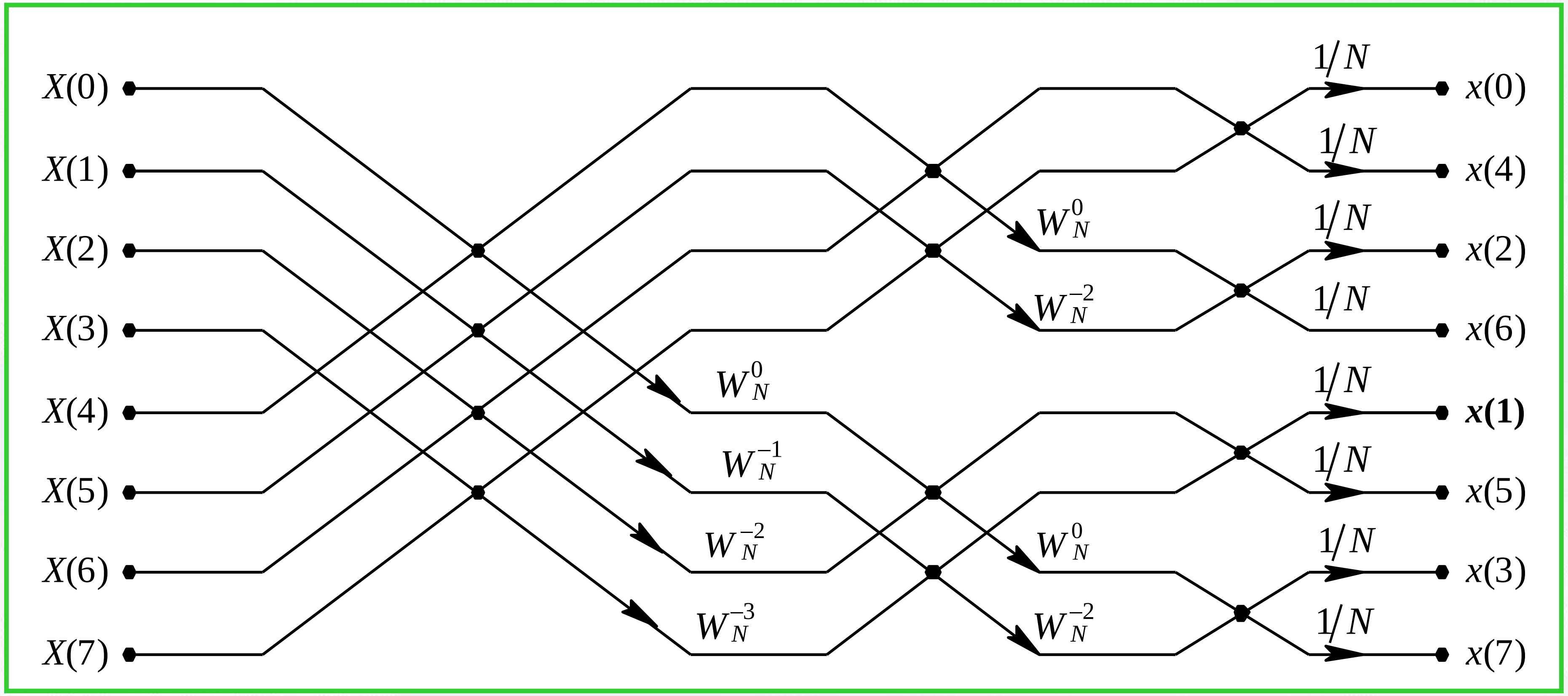
$$X(k)=\mathrm{DFT}[x(n)]=\sum_{n=0}^{N-1}x(n)W_N^{nk}\quad 0\le k\le N-1$$

$$x(n)=\mathrm{IDFT}[X(k)]=\frac{1}{N}\sum_{k=0}^{N-1}X(k)W_N^{-nk}\quad 0\le n\le N-1$$

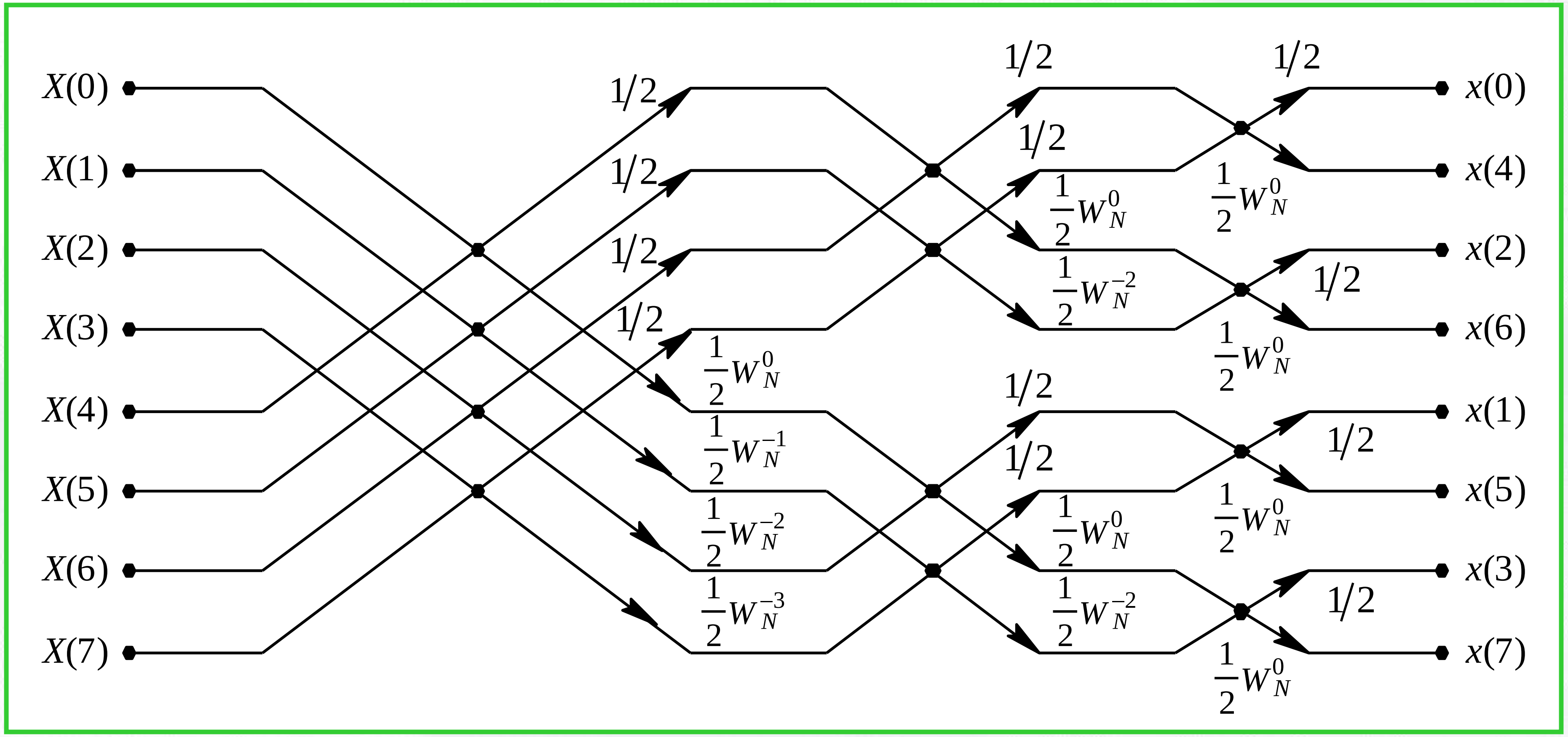
(1) 旋转因子： $W_N^k \to W_N^{-k}$

(2) 系数： $1/N$， $N=2^M$

## DIT—IFFT运算流图



## DIT—IFFT运算流图(防止溢出)



如果希望直接调用FFT子程序计算IFFT，则可用下面的方法：

$$x(n)=\frac{1}{N}\sum_{k=0}^{N-1}X(k)W_N^{-kn}$$

$$x^*(n)=\frac{1}{N}\sum_{k=0}^{N-1}X^*(k)W_N^{kn}$$

对上式两边再同时取共轭，得：

$$x(n)=\frac{1}{N}\left[\sum_{k=0}^{N-1}X^*(k)W_N^{kn}\right]^*=\frac{1}{N}\left\{\mathrm{FFT}\left[X^*(k)\right]\right\}^*$$

# 第四章　快速傅里叶变换

*Fast Forurier Transform*

## 4.4 快速傅里叶反变换的实现方法

华东理工大学信息科学与工程学院 万永菁