Fast Fourier Transform summary

G. Fountoukidou, Prof. Ioannis Pitas
Aristotle University of Thessaloniki
pitas@csd.auth.gr
www.aiia.csd.auth.gr
Version 2.2.1

Artificial Intelligence &
Information Analysis Lab

Fast Fourier Transform

- **DFT to FFT**
- Decimation in Time (DIT) FFT
- Decimation in Frequency (DIF) FFT
- FFT Computation issues
- Goertzel Algorithm
- Bluestein Algorithm

Fast Fourier Transform

Discrete Fourier Transform (***DFT***) of a signal $x(n)$:

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

- N complex roots of unity:

$$W_N = e^{-i\frac{2\pi}{N}}, \qquad W_N^N = 1.$$

Inverse Discrete Fourier Transform (***IDFT***):

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_n^{-nk}.$$

Fast Fourier Transform

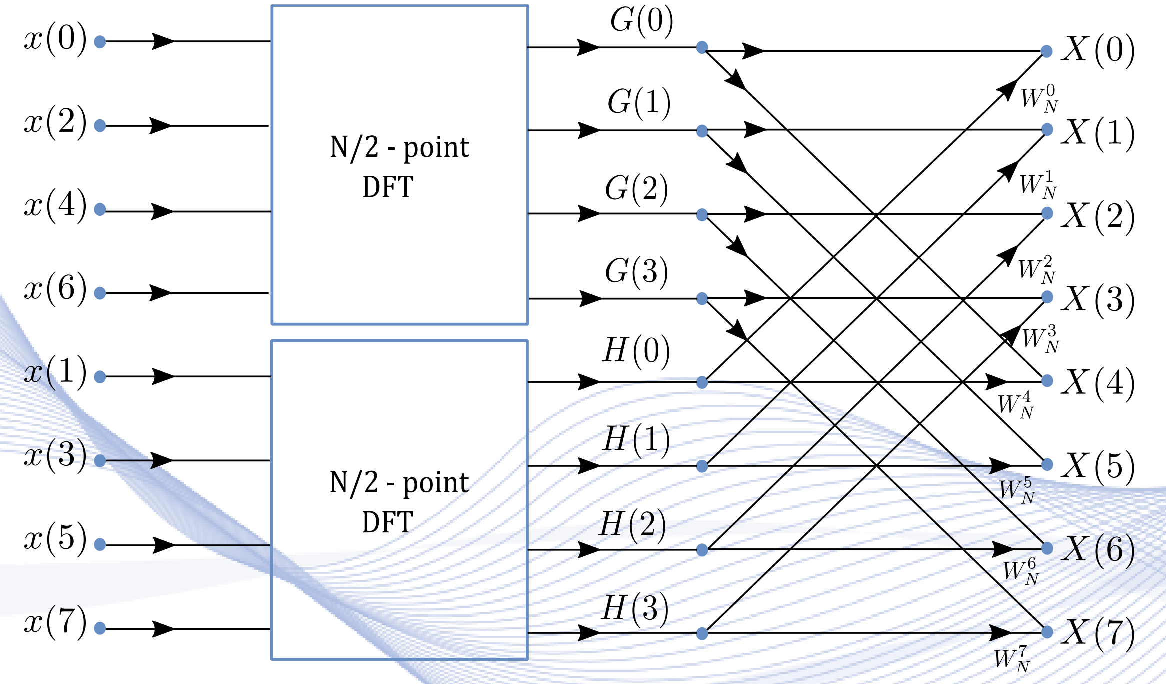
- DFT computation requires complex multiplications and additions.
- Each complex multiplication requires 4 real multiplications.
- DFT or IDFT computation by definition requires 2 for loops.
- Their computation complexity is $O(N^2)$.

- The most important advantage of DFT is that it can be calculated very fast using the ***Fast Fourier Transform*** (***FFT***) algorithm.

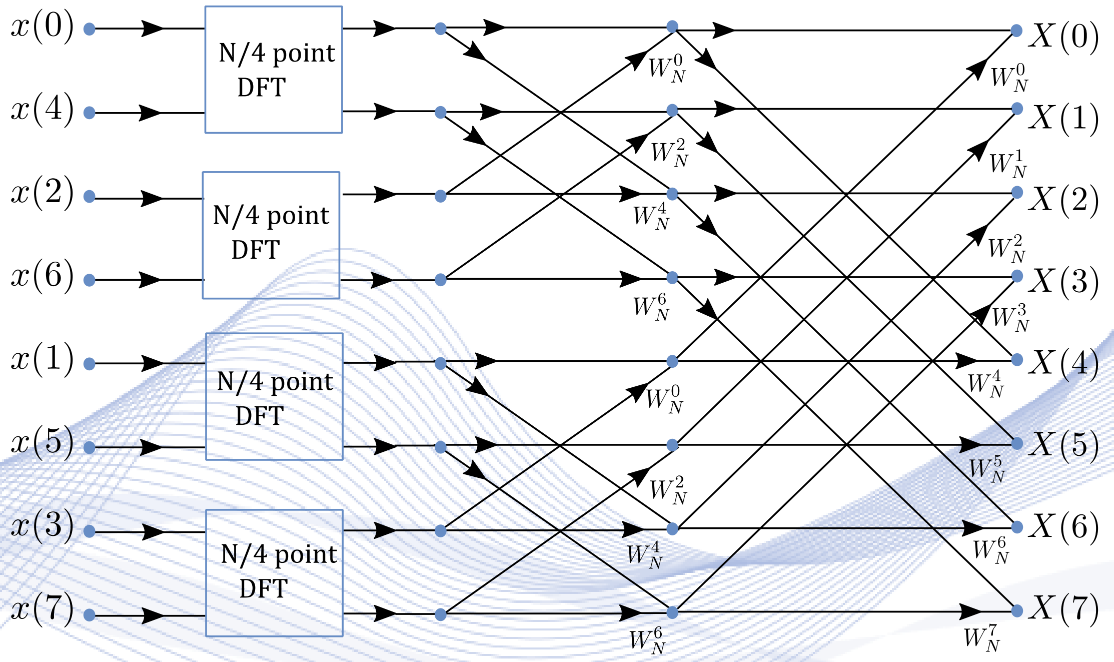
Fast Fourier Transform

- DFT to FFT
- **Decimation in Time (DIT) FFT**
- Decimation in Frequency (DIF) FFT
- FFT Computation issues
- Goertzel Algorithm
- Bluestein Algorithm

Decimation In Time FFT



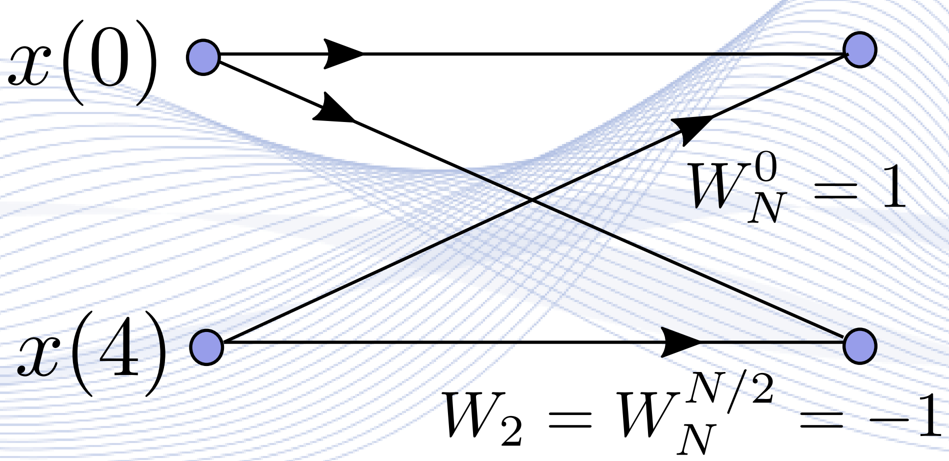
First FFT Stage for $N = 8$.



Second FFT Stage for $N = 8$.

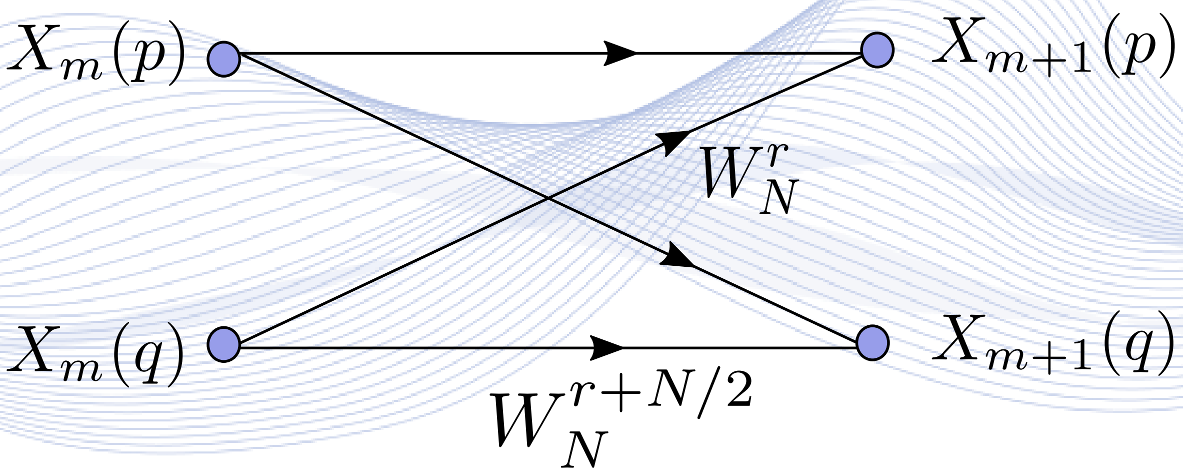
Decimation In Time FFT

- This method results in the so−called ***Decimation in Time FFT*** (***DIT FFT***):
- This decimation process is continued, by breaking the calculation of the DFTs of length $N/2$ into 4 DFTs of length $N/4$, etc., until we come to DFTs of length 2:



Decimation In Time FFT

- Each stage consists of $N/2$ basic computation structures called "***butterflies***".
- Each butterfly has 2 complex multiplication and two complex additions.



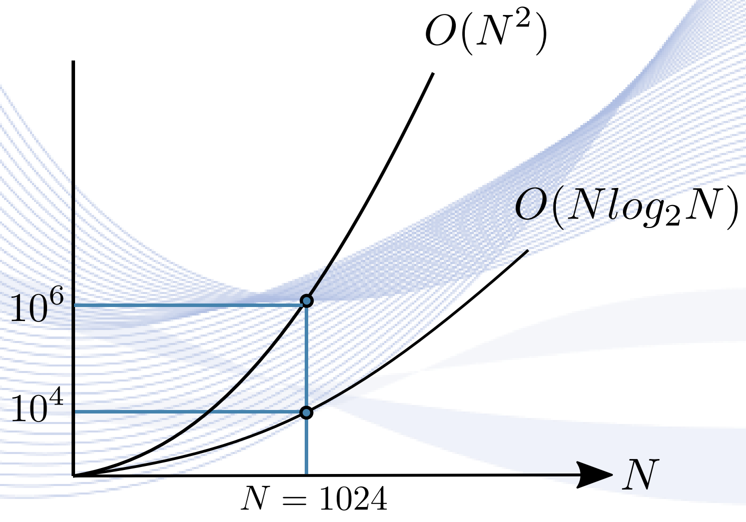
Decimation In Time FFT

- DIT FFT has $n = \log_2 N$ steps. Each step has $\frac{N}{2}$ butterflies.
- Therefore, it requires a total of:

$$M = 2(\frac{N}{2}) \log_2 N = Nlo\,g_2\,N\,.$$

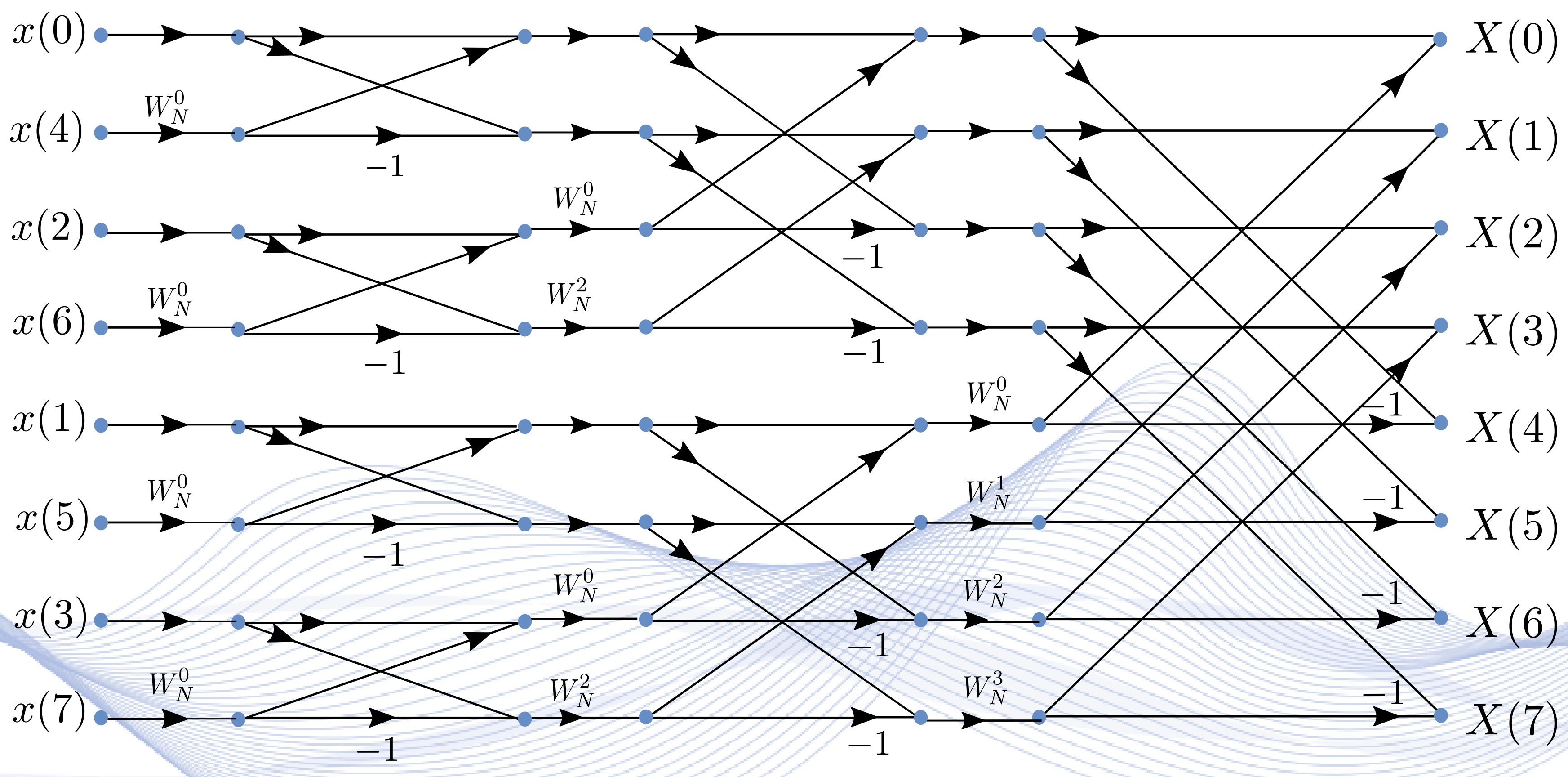
complex multiplications and $Nlog_2\,N$ complex additions.

- We already have a reduction in computational complexity from $O(N^2)$ to $O(N \log_2 N)$.



Artificial Intelligence & Information Analysis Lab

Decimation In Time FFT



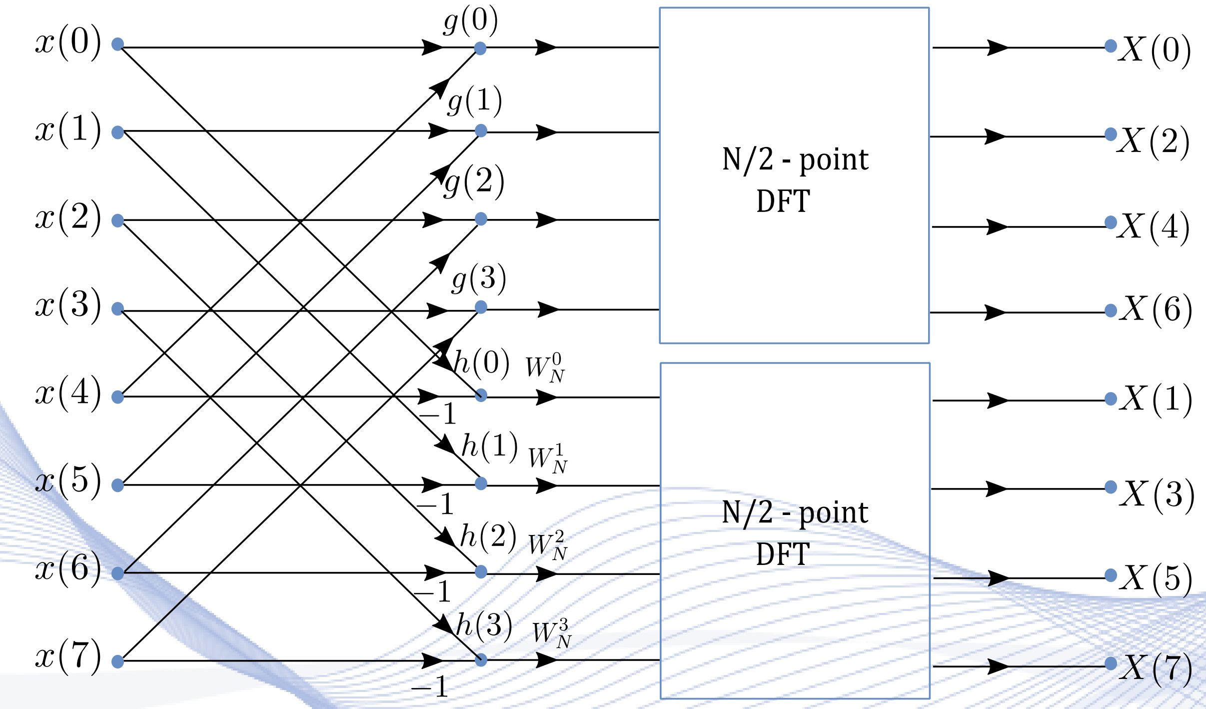
DIT FFT $N = 8$.

Artificial Intelligence & Information Analysis Lab

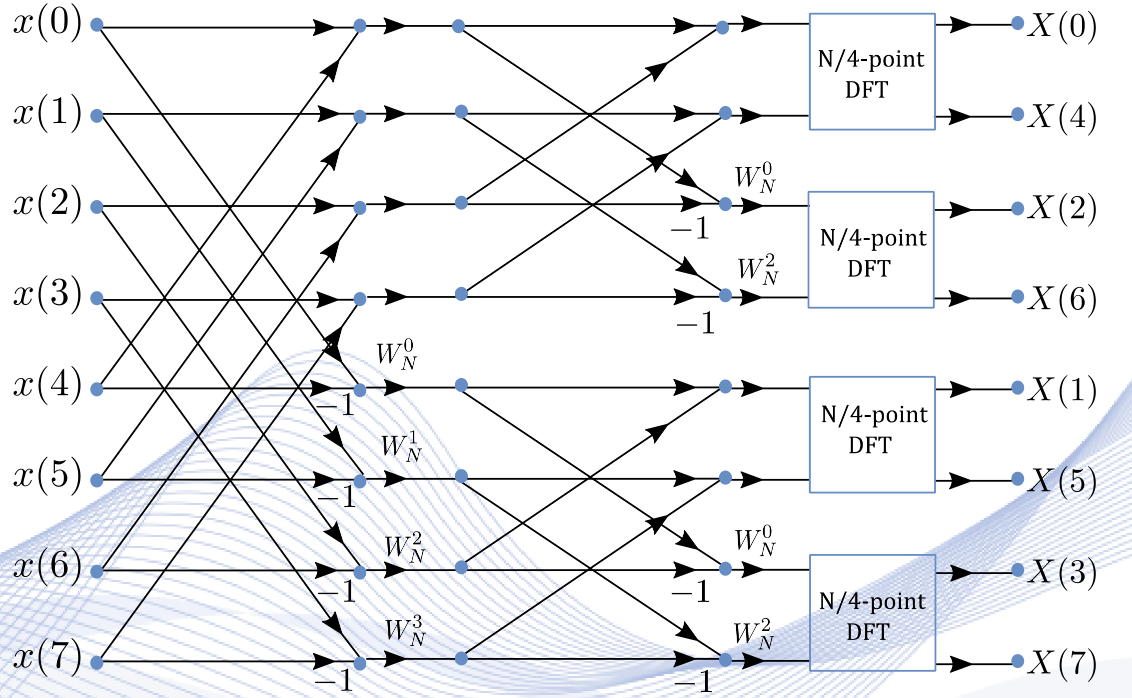
Fast Fourier Transform

- DFT to FFT
- Decimation in Time (DIT) FFT
- **Decimation in Frequency (DIF) FFT**
- FFT Computation issues
- Goertzel Algorithm
- Bluestein Algorithm

Decimation in Frequency FFT

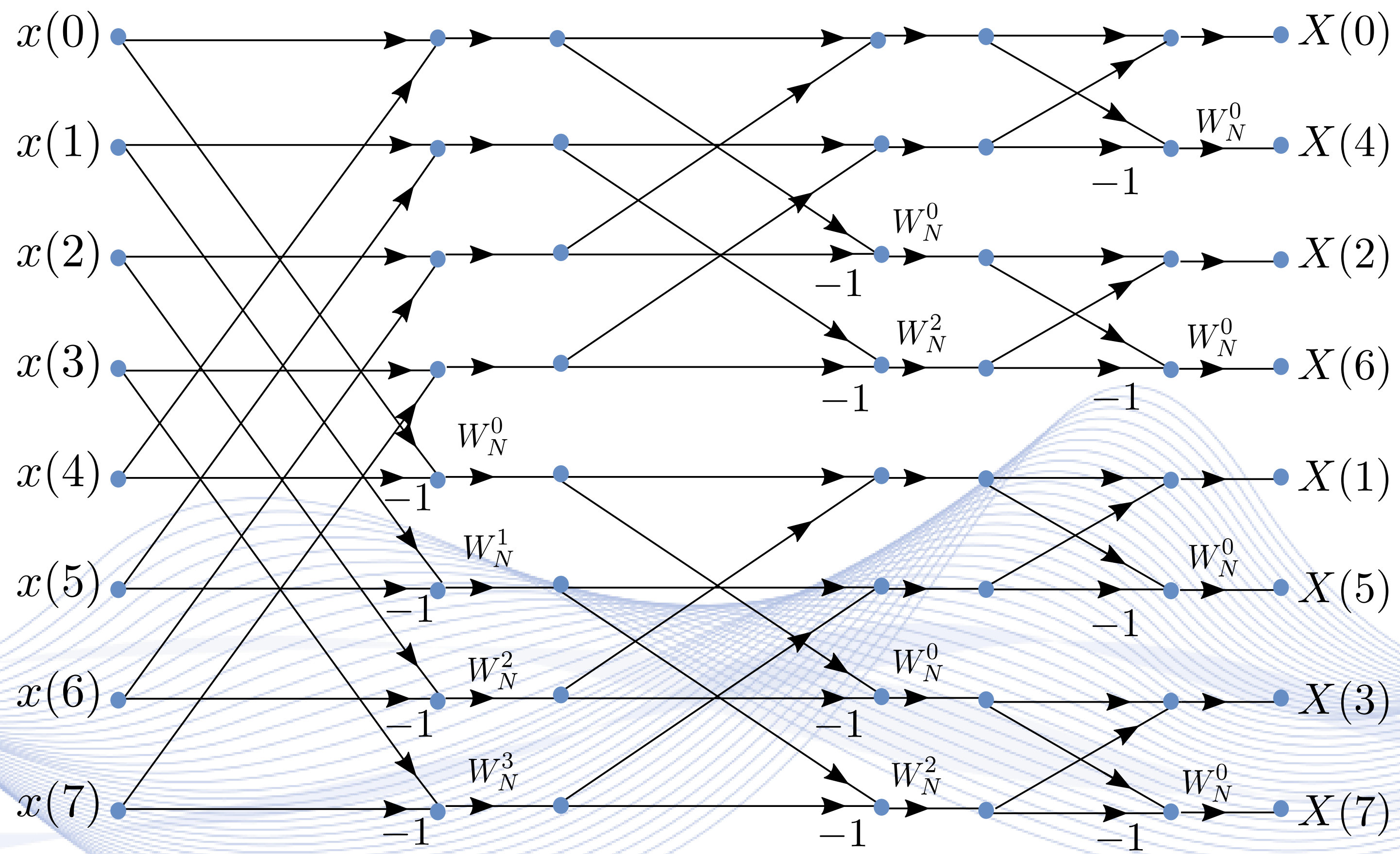


First DIF FFT stage $N = 8$.



Second DIF FFT stage $N = 8$.

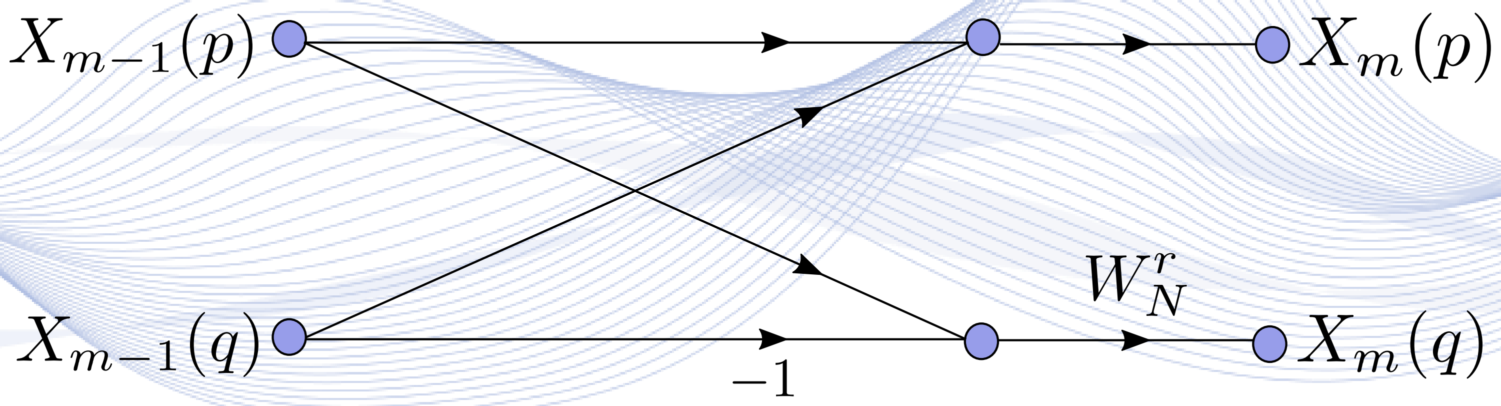
Decimation in Frequency



Final DIF FFT stage $N = 8$.

Decimation in Frequency FFT

- DIF FFT butterfly requires one complex multiplication and two complex additions.
- Therefore DIF FFT also requires $N \log_2 N$ complex additions and $(N/2) \log_2 N$ multiplications.



DIF FFT butterfly.

Fast Fourier Transform

- DFT to FFT
- Decimation in Time (DIT) FFT
- Decimation in Frequency (DIF) FFT
- **FFT Computation issues**
- Goertzel Algorithm
- Bluestein Algorithm

FFT Computation Issues

There are FFT algorithms for lengths different than a power of 2:

- ***Radix 4 FFT*** (N is power of 4),

- ***Prime Factor Algorithm*** (***PFA FFT***):
  - It calculates the DFT fast, if its length is a product of prime numbers:

$$N = p_1 p_2 \dots p_n.$$

- $p_1, p_2, \dots, p_n$: co-prime integers.

Fast Fourier Transform

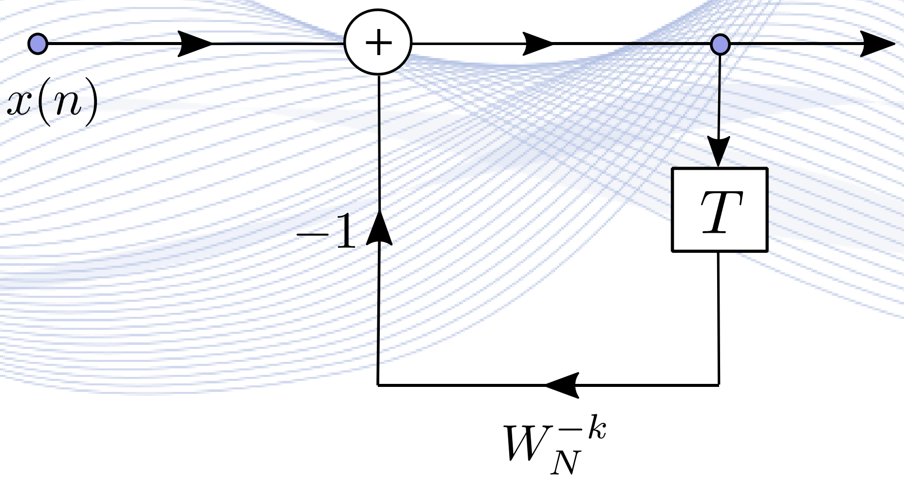
- DFT to FFT
- Decimation in Time (DIT) FFT
- Decimation in Frequency (DIF) FFT
- FFT Computation issues
- **Goertzel Algorithm**
- Bluestein Algorithm

Goertzel Algorithm

The IIR filter that has such an impulse response is given by:

$$H_k(z) = \frac{1}{1 - W_N^{-k} z^{-1}}.$$

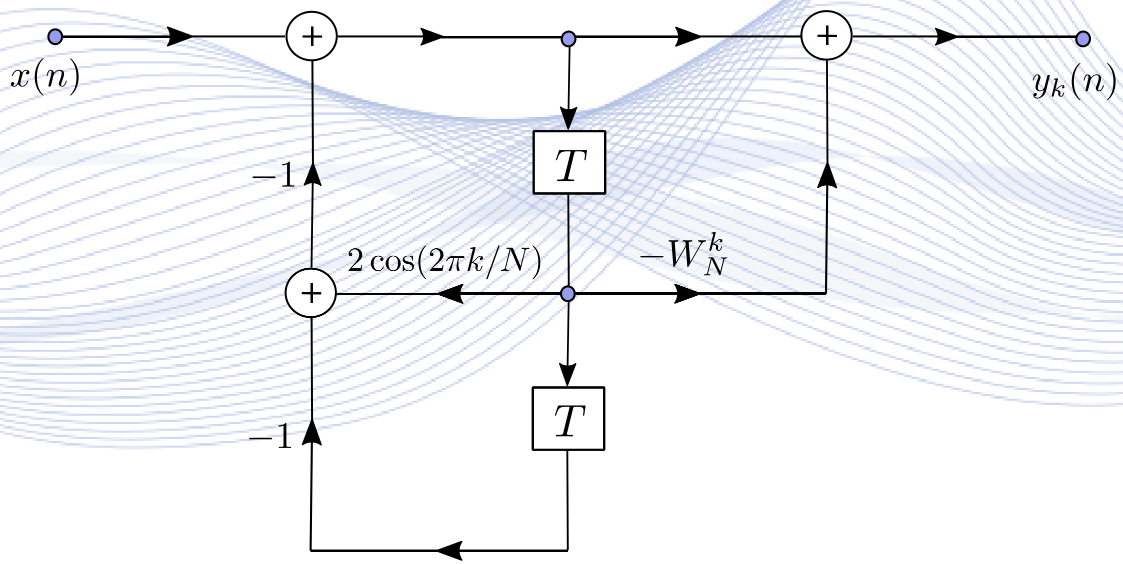
Therefore $X(k)$ calculation can be done using the IIR filter structure:



Goertzel Algorithm

An improved form of the ***Goetzel filter*** is given by the relation:

$$H_k(z) = \frac{1 - W_N^k z^{-1}}{(1 - W_N^{-k} z^{-1})(1 - W_N^k z^{-1})} = \frac{1 - W_N^k z^{-1}}{1 - 2\cos\left(\frac{2\pi k}{N}\right) z^{-1} + z^{-2}}.$$

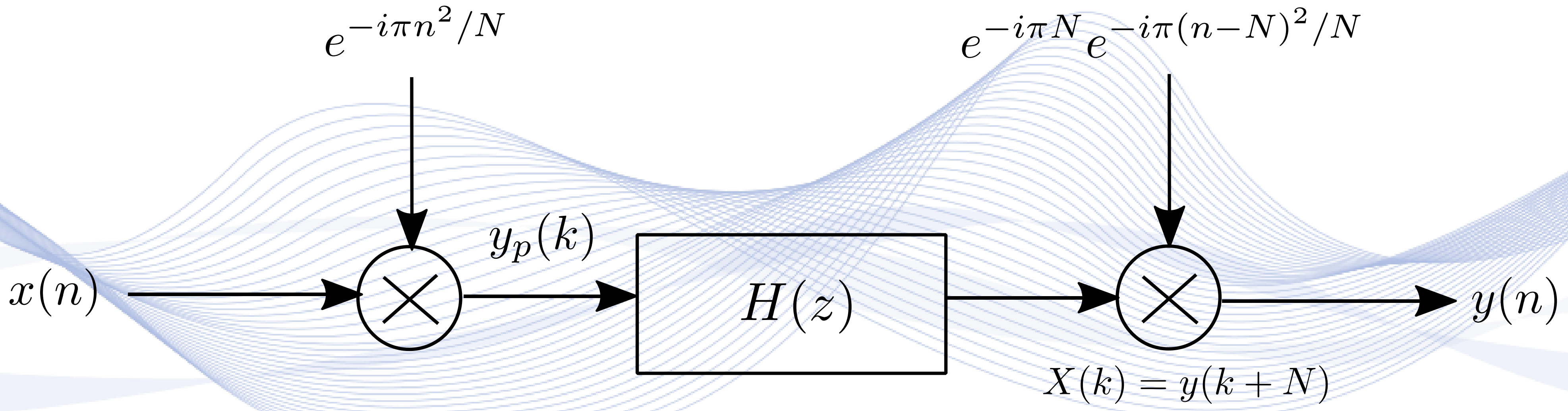


Fast Fourier Transform

- DFT to FFT
- Decimation in Time (DIT) FFT
- Decimation in Frequency (DIF) FFT
- FFT Computation issues
- Goertzel Algorithm
- **Bluestein Algorithm**

Bluestein Algorithm

If we input signal $x(n)e^{-\frac{i\pi n^2}{N}}$ in the filter and multiply the output by $e^{-i\pi N}e^{-i\pi(n-N)^2/N}$, then the filter output is the DFT $X(k)$.



Bluestein filter structure.

Bibliography

[OPP2013] A. Oppenheim, A. Willsky, Signals and Systems, Pearson New International, 2013.

[MIT1997] S. K. Mitra, Digital Signal Processing, McGraw-Hill, 1997.

[OPP1999] A.V. Oppenheim, Discrete-time signal processing, Pearson Education India, 1999.

[HAY2007] S. Haykin, B. Van Veen, Signals and systems, John Wiley, 2007.

[LAT2005] B. P. Lathi, Linear Systems and Signals, Oxford University Press, 2005.

[HWE2013] H. Hwei. Schaum's Outline of Signals and Systems, McGraw-Hill, 2013.

[MCC2003] J. McClellan, R. W. Schafer, and M. A. Yoder, Signal Processing, Pearson Education Prentice Hall, 2003.

Bibliography

[PHI2008] C. L. Phillips, J. M. Parr, and E. A. Riskin, Signals, Systems, and Transforms, Pearson Education, 2008.

[PRO2007] J.G. Proakis, D.G. Manolakis, Digital signal processing. PHI Publication, 2007.

[DUT2009] T. Dutoit and F. Marques, Applied Signal Processing. A MATLAB-Based Proof of Concept. New York, N.Y.: Springer, 2009

Bibliography

[PIT2000] I. Pitas, “Digital Image Processing Algorithms and Applications”, J. Wiley, 2000.

[PIT2021] I. Pitas, “Computer vision”, Createspace/Amazon, in press.

[PIT2017] I. Pitas, “Digital video processing and analysis” , China Machine Press, 2017 (in Chinese).

[PIT2013] I. Pitas, “Digital Video and Television” , Createspace/Amazon, 2013.

[NIK2000] N. Nikolaidis and I. Pitas, “3D Image Processing Algorithms”, J. Wiley, 2000.

Q & A

Thank you very much for your attention!

More material in
http://icarus.csd.auth.gr/cvml-web-lecture-series/

Contact: Prof. I. Pitas
pitas@csd.auth.gr