大學部專題簡介 (FFT)

Speaker: 俞子豪
Adviser: 吳安宇教授
2006/05/04
Outline

- DFT
- FFT algorithm
  - Decimation in time
  - Decimation in frequency
- FFT HW Architecture
  - Single processing element
  - Pipelined architecture
  - CORDIC based FFT
  - DHT based FFT
  - Radix $2^2$ FFT
- Topic
DFT (Discrete Fourier Transform)

\[ X(k) = \sum_{n=0}^{N-1} x(n) W_N^{nk} \]
\[ W_N = e^{-j\frac{2\pi}{N}}, k = 0, 1, \ldots, N-1 \]

- \( x(n) \): time domain sequence
- \( X(k) \): frequency domain spectrum

IDFT

\[ x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-nk} \]
\[ W_N = e^{-j\frac{2\pi}{N}}, n = 0, 1, \ldots, N-1 \]
Application

- Communication: modulation/demodulation
  - DAB (digital audio broadcasting)
  - DVB-T (digital video broadcasting-terrestrial)
  - ADSL (asymmetric digital subscriber line)
  - VDSL (very high bit rate digital subscriber line)
- Fast Convolution
- Spectrum Estimation
- Digital Signal Processing…
Outline

- DFT
- FFT algorithm
  - Decimation in time
  - Decimation in frequency
- FFT hardware architecture
  - Single processing element
  - Pipelined architecture
- Final project requirement
Fast Fourier Transform (FFT)

- An algorithm to compute DFT efficiently
- Tukey and Cooley, in 1965
- Computation Complexity
  - Direct DFT: $N^2$
  - FFT: $N \cdot \log_2(N)$
- Divide and conquer

Diagram:

```
N
/   \
/     \
N/2    N/2
/   \
/     \
N/4    N/4
/     \
/     \
N/4    N/4
```

N/4

2006/05/04
FFT – Decimation in Time

\[ FFT_N(k, f) = \sum_{n=0}^{N-1} f(n) e^{-j2\pi kn/N} \]

\[ FFT_N(k, f) = \sum_{n'=0}^{N/2-1} f(2n') e^{-j2\pi k(2n')/N} + \sum_{n'=0}^{N/2-1} f(2n' + 1) e^{-j2\pi k(2n' + 1)/N} \]

\[ = \sum_{n'=0}^{N/2-1} f(2n') e^{-j2\pi k n'/(N/2)} + e^{-j2\pi k/N} \sum_{n'=0}^{N/2-1} f(2n' + 1) e^{-j2\pi k n'/(N/2)} \]

\[ = FFT_{N/2}(k, fe) + e^{-j2\pi k/N} FFT_{N/2}(k, fo) \]

where \( fe(n') = f(2n') \), \( fo(n') = f(2n' + 1) \)
Radix-2 DIT Butterfly

\[ X_{i-1}(k) \]

\[ X_{i-1}(m) \]

Twiddle factor \( W^n \)

\[ X_i(k) \]

\[ X_i(m) \]
8-point DIT FFT Signal Flow

DFT-2

Bit reverse order

Normal order

DFT-4
FFT – Decimation in Frequency

\[ FFT_N(k, f) = \sum_{n=0}^{N/2-1} f(n) e^{-j2\pi kn/N} + \sum_{n=N/2}^{N-1} f(n) e^{-j2\pi kn/N} \]

\[ FFT_N(2k', f) = \sum_{n=0}^{N/2-1} \{f(n) + f(n + N/2)\} e^{-j2\pi k'n/(N/2)} \]

\[ FFT_{N/2}(k', f_E) = \sum_{n=0}^{N/2-1} f_E(n) e^{-j2\pi k'n/(N/2)} \]

\[ FFT_N(2k' + 1, f) = \sum_{n=0}^{N/2-1} \{f(n) - f(n + N/2)\} e^{-j2\pi n/N} e^{-j2\pi k'n/(N/2)} \]

\[ FFT_{N/2}(k', f_O) = \sum_{n=0}^{N/2-1} f_O(n) e^{-j2\pi k'n/(N/2)} \]
Radix-2 DIF Butterfly

\[ X_{i-1}(k) \rightarrow + \rightarrow X_i(k) \]

\[ X_{i-1}(m) \rightarrow -1 \rightarrow X_i(m) \]

Twiddle factor \( W^n \)
8-point DIF FFT Signal Flow

Normal order

000
001
010
011
100
101
110
111

DFT-4

DFT-2

Bit reverse order

000
100
010
110
001
101
011
111
Outline

- DFT
- FFT algorithm
  - Decimation in time
  - Decimation in frequency
- FFT Hardware architecture
  - Single processing element
  - Pipelined architecture
- Final project requirement
Single Processing Element Architecture

- Only one butterfly PE
- Address generator
Pipelined Architecture: Single Delay Feedback

R2SDF

input $x[n]$  \[\times\]  twiddle factor  \[\times\]  twiddle factor  \[\times\]  twiddle factor  \[\times\]  output $X[n]$  

R4SDF

Radix-4  BF  \[\times\]  Radix-4  BF  \[\times\]  Radix-4  BF  \[\times\]  Radix-4  BF
Radix-$2^2$ FFT

- Reduce hardware cost
  - Compared to R2SDF: fewer multiplier
  - Compared to R4SDF: fewer adder
Final Project Requirement

- Study – Radix-2^2 FFT algorithm.
- Verilog Coding – 64-point Radix-2^2 FFT.