

Illustrations

Figure 2-1	Fourier transform of a rectangular function	2-3
Figure 2-2	Windowing effects when windowing a single sine wave	2-6
Figure 3-1	The FFT principle in layman's terms	3-2
Figure 3-2	Decimation-in-Time of an N-Point FFT	3-4
Figure 3-3	Decimation-in-Time FFT: step two	3-4
Figure 3-4	Decimation-in-Time FFT: final step (2-Point DFT)	3-5
Figure 3-5	An 8-point, radix-2, Decimation-in-Time FFT	3-5
Figure 3-6	Rearrangement of the "Butterfly" building block of the DIT FFT	3-6
Figure 3-7	Rearrangement of the "Butterfly" building block of the DIF FFT	3-6
Figure 3-8	Rearrangement of the DIT computation of Figure 3-6	3-7
Figure 3-9	Decimation-in-Frequency concept	3-8
Figure 3-10	Complete 8-Point Radix-2 DIF FFT	3-8
Figure 4-1	Grouping of butterflies in the FFT calculation	4-5
Figure 4-2	DSP56001 architecture block diagram	4-7

Illustrations

Figure 4-3	A radix-2 DIT butterfly that needs less instruction cycles than a radix-2 DIF butterfly	4-8
Figure 4-4	The radix-2, DIT butterfly kernel on the DSP56001/DSP56002	4-9
Figure 4-5	A simple, triple-nested DO-loop radix-2 DIT FFT on DSP56000/DSP56001	4-11
Figure 4-6	DSP96002 architectural block diagram	4-14
Figure 4-7	The radix-2, DIT FFT Butterfly Kernel on the DSP96002	4-16
Figure 4-8	DSP56156 architectural block diagram	4-18
Figure 4-9	The butterfly core of the DSP56156	4-19
Figure 4-10	In-place bit-reversed to normal order conversion	4-26
Figure 4-11	A flow diagram of two stages in a radix-2 DIT butterfly	4-27
Figure 4-12	A flow diagram of a radix-4 DIT butterfly	4-29
Figure 4-13	Radix-4 DIT butterfly takes 17 instructions on the DSP96002	4-30
Figure 5-1	Trivial twiddle factors in a 512-point complex radix-2 DIT FFT	5-11
Figure 6-1	Non-redundancy calculation of the Cooley-Tukey radix-2 DIT FFT with real inputs	6-3

Illustrations

Figure 6-2	Bergland algorithm has only $\log_2(N)-1$ passes and one more addition and subtraction	6-5
Figure 6-3	(a) Butterfly of Bergland Algorithm with $W = 1$ (b) Butterfly of Bergland Algorithm with $W \neq 1$	6-6
Figure 6-4	C language code that generates Bergland order tables	6-7
Figure 6-5	Computation of the real-input, DIT FFT	6-17
Figure 6-6	DSP56001 assembly code that calculates energy of DFT coefficients by single parameter	6-19
Figure 6-7	Double buffering input data so that data input can work with the FFT program concurrently	6-21
Figure 6-8	Block diagram of the double buffering technique	6-22
Figure 7-1	The flow diagram of an 8-point discrete cosine transform	7-4
Figure A-1	Optimized Complex FFT for the DSP96002	A-1
Figure B-1	Faster real FFT for the DSP96002	B-1
Figure B-2	Real FFT for DSP56001/2	B-5