

Table of Contents

SECTION 1	1.1	Introduction to the Fourier Integral	1-1
Definition and History	1.2	Use of the Fourier Transform	1-3
SECTION 2	2.1	The Discrete-Time Fourier Transform (DTFT)	2-1
The Discrete Fourier Transform	2.2	Windowing and Windowing Effects	2-4
	2.3	Sampling the Frequency Function	2-7
SECTION 3	3.1	Motivation	3-1
The Fast Fourier Transform	3.2	Divide and Conquer	3-2
	3.3	The Decimation-in-Time and Decimation-in-Frequency Radix-2 Fast Fourier Transforms	3-3
	3.4	The Decimation-in-Frequency Radix-2 Fast Fourier Transforms	3-9
SECTION 4	4.1	Required Hardware Support for FFT Calculation	4-1
Complex FFT on the Motorola DSP Family	4.2	Radix-2 DIT and DIF Butterflies	4-4
	4.3	Complexity of a Radix-2 DIT FFT	4-6
	4.4	Implementation on Motorola's DSP56001	4-6
	4.4.1	DSP56001 Architecture	4-6
	4.4.2	DIT Butterfly Kernel on DSP56001	4-9

Table of Contents

4.5	Implementation on Motorola's DSP96002	4-13
4.5.1	DSP96002 Architecture	4-13
4.5.2	DIT Butterfly Kernel on DSP96002	4-15
4.6	Implementation on Motorola's DSP56156	4-17
4.6.1	DSP56156 Architecture	4-17
4.6.2	DIT Butterfly Kernel on DSP56156	4-19
4.7	Scaling for Fixed-Point Processors (DSP56001/2 and DSP56156)	4-19
4.7.1	Scaling at the Input – Guard Bits	4-20
4.7.2	Scaling During the Passes – Auto-Scaling and Block Floating-Point	4-21
4.8	Twiddle Factors and On-Chip ROM	4-23
4.8.1	Twiddle Factors for DIT	4-23
4.8.2	Sine Table on the DSP56001/2	4-23
4.8.3	Sine and Cosine Tables on the DSP96002	4-24
4.9	Bit-Reversed Addressing	4-25
4.10	Implementation of a Radix-4 DIT FFT on DSP96002	4-26
4.10.1	Radix-4 DIT Butterfly Core	4-27
4.10.2	Radix-4 DIF Butterfly Core	4-31
4.11	Inverse FFT	4-32

Table of Contents

SECTION 5	5.1	Optimization	5-1
Optimizing	5.1.1	Minimum Memory Requirement —	
Performance		In-Place Calculation	5-3
of the FFT	5.1.2	Optimization for Faster Execution	5-5
	5.2	Example of Optimization	5-9
	5.2.1	Fully Optimized Complex FFT	
		for the DSP56001/2	5-9
	5.2.2	Fully Optimized Complex FFT	
		for the DSP96002	5-12
 SECTION 6	 6.1	 Real-Valued Input FFT Algorithm 1	 6-2
Real-Valued	6.1.1	Bergland Algorithm	6-2
Input FFT	6.1.2	Reordering	6-6
Algorithm	6.1.3	Performance Estimation	6-7
	6.2	Real-Valued Input FFT Algorithm 2	6-9
	6.2.1	Separating Two Real FFT	
		from One Complex FFT	6-9
	6.2.2	Rebuilding the DFT of a Real	
		Sequence from Two DFTs	6-11
	6.2.3	Performance Estimation	6-13
	6.3	Real-Valued Input FFT Algorithm 3	6-15
	6.4	The Goertzel Algorithm	6-18
	6.5	Real-Time Data Acquisition on	
		Motorola DSPs	6-20
	6.5.1	Fast Interrupt on DSP56001 for	
		Real-Time FFT Data Acquisition	6-21
	6.5.2	Real-Time Data Acquisition	
		on DSP96002	6-23

Table of Contents

SECTION 7	7.1	Two Dimensional FFTs on the DSP96002	7-1
Two	7.2	Discrete Cosine Transform on the DSP96002	7-2
Dimensional	7.2.1	One Dimensional Discrete Cosine Transform (DCT)	7-2
Fourier and	7.2.2	Two Dimensional DCT	7-5
Cosine			
Transforms			
SECTION 8	8.1	Most Popular Digital Signal Processors	8-1
Competitive	8.2	Performance of FFTs on Digital Signal Processors	8-2
Analysis of FFT	8.2.1	FFTs on Floating-Point DSPs	8-3
Performances	8.2.1.1	Complex FFT on Floating-Point DSPs	8-4
	8.2.1.2	Real FFT on Floating-Point DSPs	8-5
	8.2.2	FFT on Fixed-Point DSPs	8-6
	8.2.2.1	Complex Input FFT	8-6
	8.2.2.2	Real Input FFT	8-7
SECTION 9			
Conclusion			9-1
APPENDIX A	A.1	Optimized Complex FFT for the DSP96002	A-1
Fully Optimized			
Complex FFT			

Table of Contents

APPENDIX B	B.1	Faster real FFT for the DSP96002	B-1
Real-Valued	B.2	Real FFT for DSP56001/2	B-5
Input FFT			