

Vitor Amadeu Souza

Implementando uma

FFT

no

Arduino

©2016 by Cerne Tecnologia e Treinamento Ltda.

©2016 by Vitor Amadeu Souza

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

Janeiro de 2016

Direitos reservados por:

Cerne Tecnologia e Treinamento Ltda

Produção: Cerne Tecnologia e Treinamento

E-mail da Empresa: cerne@cerne-tec.com.br

Home Page: www.cerne-tec.com.br.com.br

Atendimento ao Consumidor: sac@cerne-tec.com.br

Contato com o Autor: viktor@cerne-tec.com.br



FEITO NO BRASIL

Agradecimentos

Agradeço ao Professor Pablo Diniz pela excelente aula ministrada sobre DFT/FFT, que me permitiu desmistificar este assunto e entender com clareza o seu funcionamento. Agradeço também a a minha querida esposa Renata Leal todo o apoio dispensado para conclusão desta obra.

“A única certeza que existe é a incerteza.”

Heráclito

Kits Didáticos e Gravadores da Cerne Tecnologia

A Cerne tecnologia têm uma linha completa de aprendizado para os microcontroladores da família ARDUINO, 8051, Holtek, dsPIC, ARM e etc. Veja os detalhes de cada um nas figuras abaixo:



Kit Cerne Arduino

- Microcontrolador ATMEGA8;
- Comunicação serial RS232;
- Alimentação de 12V;
- Pinos de I/O;
- Gravação ICSP.

Uma linha completa de componentes para o desenvolvimento de seus projetos eletrônicos como displays, PICs, botões, leds, cristais e etc.

Visite a nossa página na Internet, no endereço www.cerne-tec.com.br e conheça melhor nossos serviços e produtos.



www.cerne-tec.com.br

Sumário

Capítulo I: Metodologia de desenvolvimento	7
1. Introdução	7
Capítulo II: Conversor Analógico Digital.....	8
1. Introdução	8
2. Esquema elétrico	12
3. Programando o Arduino	13
Capítulo III: DFT no Arduino.....	14
1. Análise no tempo e frequência	14
2. Frequência de Nyquist	16
3. DFT - Transformada de Fourier Discreta.....	17
4. Análise de frequência	23
5. Esforço computacional	25
6. Hardware de testes.....	27
7. Esquema elétrico	29
8. Fluxograma simplificado.....	30
9. Código fonte	30
Capítulo IV: DFT no PC.....	33
1. Introdução	33
2. Esquema elétrico	34
3. Fluxograma do Arduino	34
4. Código fonte do Arduino	35
5. Fluxograma do PC	36
6. Código fonte em Delphi.....	37
Capítulo V: FFT.....	49
1. Transformada Rápida de Fourier (FFT).....	49
2. DFT de Números Reais.....	52
3. DFT de Números Complexos.....	53
4. Transformada Inversa Rápida de Fourier (IFFT).....	54

5. Decimação no tempo.....	56
6. Radix	62
7. Butterfly.....	63
8. Função FFT em Pascal	66
9. Simulação DFTxFFT	71
Capítulo VI: FFT no Arduino	78
1. Introdução	78
2. Função FFT em C.....	78
3. Fluxograma Simplificado.....	83
4. Código Fonte.....	84
Capítulo VII: FFT no PC	88
1. Introdução	88
2. Fluxograma do PIC	88
3. Código Fonte do PIC.....	89
4. Fluxograma do PC.....	91
5. Código Fonte em Delphi	92
Referências.....	95

Capítulo I

Metodologia de desenvolvimento

1. Introdução

Esta literatura é uma continuação da obra *Arduino – Prático e Objetivo* (2011) e *Programação para Arduino – Avançado* (2014) do mesmo autor e editora, onde o objetivo é implementar uma FFT (Transformada Rápida de Fourier) de modo a identificar qual a frequência fundamental que está sendo fornecida a entrada analógica da placa Arduino e mostrá-la através de um LCD.

O livro também contempla o envio para o PC de um buffer referente às amostras analógicas, para que este possa calcular a FFT e plotar em um gráfico a frequência fundamental e harmônicos nela presente. Todas as etapas de programação são apresentadas, tanto do Arduino quanto do Delphi para programar o PC.

A placa didática utilizada foi a Cerne Arduino, onde tal kit estão à venda no site www.cerne-tec.com.br. No entanto, outros modelos compatíveis como a Arduino UNO podem ser utilizadas, em função da compatibilidade entre as mesmas.

Capítulo II

Conversor Analógico Digital

1. Introdução

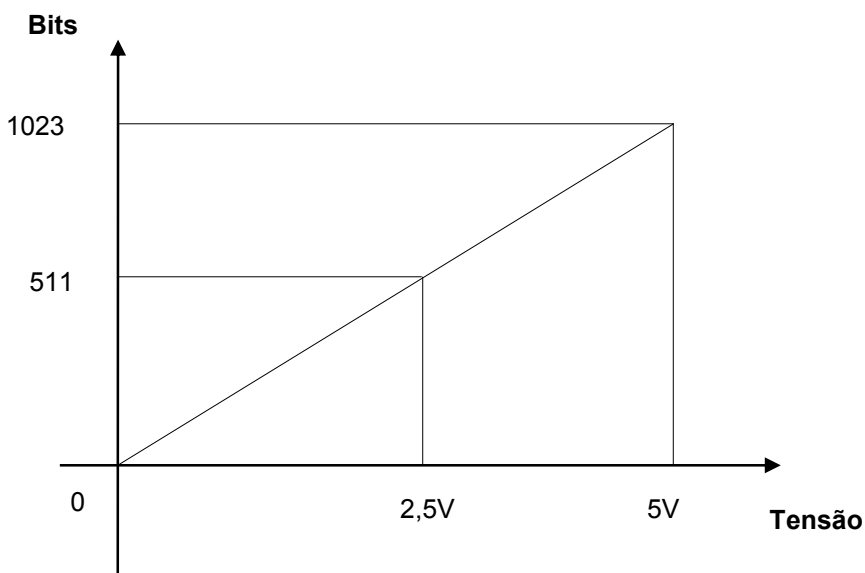
O microcontrolador (MCU) utilizado no Arduino trabalha internamente com grandezas digitais enquanto que o mundo externo em si é analógico. Sendo assim, há internamente no MCU um dispositivo chamado conversor analógico digital (CAD), onde este tem a incumbência de converter uma grandeza analógica e permitir assim ao MCU ser capaz de trabalhar com esta informação externa.

Na placa Arduino tais entradas são reconhecidas como A0, A1, A2, A3, A4 e A5, ou seja, a mesma disponibiliza 6 entradas analógicas.

Tais entradas normalmente funcionam dentro da faixa de alimentação do MCU, indicando que se o Arduino está sendo alimentado em 5V conclui-se que tais entradas devem variar dentro desta faixa de alimentação.

O conversor ADC do MCU AVR é de 10 bits, logo o mesmo fornece uma resolução de 1024 pontos, já que 2^{10} é igual a este valor. O sensor conectado a uma das entradas analógicas do Arduino fornece um valor binário na faixa de 0 a 1023.

Sendo assim, podemos modelar o seguinte gráfico no qual converte a escala de tensão para os bits referentes à conversão AD:



Observe que há um gráfico linear referente à tensão com o resultado do conversor AD. Ou seja, se a entrada de tensão do

Arduino for de 5V teremos como resultado AD o valor 1023. Se for 2,5V o valor da conversão será de 511 e assim linearmente.

Apesar do conversor AD ser de 10 bits, não somos obrigados a usar este valor para a conversão. Em nossos exemplos, usaremos a conversão de 8 bits por ser o suficiente para aplicar a DFT e assim obter o valor da frequência fundamental. Sendo assim, o gráfico apresentado anteriormente ficaria da seguinte forma caso o conversor AD seja de 8 bits:

