

ESTUDO DE ARQUITETURA EM MICROCONTROLADORES E DEFINIÇÃO DE CONCEITO ÓTIMO PARA APLICAÇÃO DE BEAMFORMERS EM FPGA

HERMAN L. DOS SANTOS*, WAGNER ENDO*, PAULO R. SCALASSARA*

* *Universidade Tecnológica Federal do Paraná
Campus Cornélio Procópio
Cornélio Procópio, Paraná, Brasil*

Emails: herman@alunos.utfpr.edu.br, wendo@utfpr.edu.br, prscalassara@utfpr.edu.br

Abstract— In this paper we present a preliminary study on Texas Instrument®'s DSP TMS320F2833x/2823x family architecture. Through the study about one known architecture is intended to define a optimum one for a spatial filtering and signal source direction of arrival estimation applications through Array Processing. We search for assembly the architecture defined on a FPGA device for real time purposes. We discuss the peripherals that will be needed to implement the array and also the Arithmetic Logic Unit specification for the designed processor.

Keywords— Array Processing, DSP, Beamformer.

Resumo— Neste artigo é apresentado um estudo preliminar de arquitetura do DSP Texas Instrument® família TMS320F2833x/2823x. Através do estudo de uma arquitetura conhecida pretende-se definir uma ótima para aplicação de filtragem espacial e detecção de direção de chegada através de técnicas de *Array Processing*. Busca-se embarcar esta arquitetura em um dispositivo FPGA para utilização em tempo real. Discuti-se aqui a presença de periféricos necessários para a criação do sistema assim como a necessidade da Unidade Lógica Aritmética do processador.

Palavras-chave— Array Processing, DSP, Formação de Feixe.

1 Introdução

Aplicações de formação de feixe, do inglês *beam-forming*, são amplamente utilizadas em diversas áreas como radar, sonar, estimação de direção de chegada de uma fonte emissora de sinais, sismologia e bioengenharia (Van Trees, 2004). O conceito de *Array Processing* utilizando a formação de feixe para filtragem espacial e detecção de sinais se assemelha à mover uma antena, apontando para a direção que se deseja obter o sinal ou encontrando a direção de chegada, aqui denominada DOA, do inglês *Direction of Arrival*, através de uma informação *a priori* do sinal que se deseja. Porém, ao invés de utilizar-se de uma montagem que pode se mover no espaço, o direcionamento do captador de sinal é todo feito de forma eletrônica (Manolakis et al., 2005), o que será discutido mais adiante neste artigo.

Nestas aplicações utilizam-se diversos sensores, dando a eles diferentes ganhos no sinal detectado (Naidu, 2009), a fim de fazer a filtragem do sinal, como em um filtro de resposta impulsiva finita FIR (Van Veen and Buckley, 1988). Este processo, dependendo da técnica empregada, pode ser muito custoso computacionalmente, o que leva a discussão do que seria uma arquitetura de microprocessador ótima para embarcar um sistema de captura e processamento de um arranjo de sensores.

Dependendo da necessidade de precisão na aplicação, é necessário utilizar técnicas diferentes para calcular os ganhos mencionados, o que torna cada vez o processo mais custoso computacionalmente. Em Yasin et al. (2010), Qiang et al. (2008),

Zhang et al. (2008), Sarevska and Salem (2008), Tian et al. (2001), Treder et al. (2016) e Oswal et al. (2016) são apresentadas diversas aplicações de *beamformers*. Para embarcar qualquer uma das técnicas mencionadas é necessário a utilização de um processador para as operações aritméticas.

Processadores convencionais possuem limitações devido à suas arquiteturas de hardware e periféricos fixos. Porém, através do advento de dispositivos de arquitetura configurável, estes problemas de hardware podem ser sanados e o processamento otimizado, sendo uma alternativa atrativa (Vemuri and Harr, 2000). Um exemplo destes dispositivos são os *Field Programmable Gate Array* - FPGA.

Devido a forma como é construído, um FPGA pode assumir diversas arquiteturas, sendo estas programadas pelo usuário, de acordo com suas limitações internas. Em Netti et al. (2015), é apresentada a implementação em tempo real de *Beamformer* em um dispositivo FPGA, que é a intenção de estudo deste artigo. Também são apresentados em diversas fontes, como Fei et al. (2011) e Fan (2011), sistemas de aquisição e processamento de dados utilizando estes dispositivos. A desvantagem na utilização destes dispositivos está em sua complexidade de operação. Por isso será comparado a utilização de lógica reconfigurável com a utilização de processadores de alto desempenho (DSPs) para a análise de viabilidade de projeto.

Este artigo é organizado da seguinte forma. Na Seção 2, são apresentados conceitos necessários para a organização do módulo de aquisição de dados. Consequentemente, na Seção 3, são apresentadas e analisadas arquiteturas de processadores

convencionais. Na Seção 4 é apresentada a definição de arquitetura ótima obtida para a aquisição destes dados. A Seção 5 contém as conclusões obtidas.

2 Aquisição e Processamento de Dados

Nesta seção serão feitos comentários e análises sobre aquisição em tempo real de dados e tempo de processamento. Primeiramente, é apresentado dados referentes à aquisição. São apresentados conceitos de armazenamento para futuro processamento de dados e de processamento de arranjo de sensores.

2.1 Aquisição de Dados

Em aplicações de aquisição e processamento de sinais é indispensável que se tenha dados sobre o sinal a ser coletado, como sua frequência. Na aquisição de dados temporais não se deve perder informações devido à sua frequência de amostragem. Para garantir que serão coletadas amostras suficientes, utiliza-se o critério de Nyquist. Este teorema diz que um sinal deve ser amostrado com, no mínimo, o dobro da frequência máxima onde se encontram informações relevantes (Oppenheim et al., 1989), Eq. (1).

$$f_s \geq 2B \quad (1)$$

sendo B a frequência máxima do sinal de interesse e f_s a frequência de amostragem mínima.

Considera-se uma aplicação com múltiplos sensores trabalhando em paralelo. As ondas de um sinal se propagam por um meio e são captadas por sensores. Estes sensores estão espaçados um do outro e, portanto, o tempo de chegada em cada sensor será diferente, ocorrendo um atraso. Este atraso temporal de chegada gera atrasos de fase quando comparados os elementos no mesmo instante de tempo. Por estes sinais serem periódicos, um atraso de fase maior que 2π rad gera uma ambiguidade. Esta ambiguidade gera erros no processamento do sinal e pode ser evitado analisando as frequências que serão captadas no arranjo e arranjando o espaçamento entre os elementos.

Seja λ o comprimento de onda de um sinal que se propaga em um espaço com velocidade de propagação c a uma frequência f , tem-se que o comprimento de onda deste sinal é dado segundo a Equação (2).

$$\lambda = \frac{c}{f} \quad (2)$$

Segundo Manolakis et al. (2005), para prevenir a ocorrência de ambiguidades espaciais é necessário que o espaçamento entre os sensores esteja ajustado de acordo com o comprimento de onda e do meio que este se propaga. Portanto, a Equação

(3), onde d é a distância em metros, dá o espaçamento máximo possível.

$$d \leq \frac{\lambda}{2} \quad (3)$$

Apesar de imprimir uma maior definição no resultado, a utilização de sensores muito próximos não traz vantagens em questões de hardware. Quanto mais próximos os sensores estiverem menor será o atraso entre os sinais.

Em sistemas regidos por um processador digital trabalha-se com tempos discretos. Portanto, quanto mais próximos estiverem os sensores maior a necessidade de uma amostragem cada vez maior a fim de prevenir ambiguidades na leitura temporal.

2.2 Aplicação de Técnicas de Filtragem e Estimção de Direção de Chegada

As técnicas mais simples, porém efetivas, e que garantem possibilidade de generalização da arquitetura nesta aplicação são a aplicação da filtragem através da formação de feixe, conhecida como *Matched Filter* e a obtenção da matriz de correlação cruzada dos sinais captados. Estas técnicas funcionam bem para um sinal isolado, sem a presença de interferência (Jan et al., 1995).

A vantagem da definição da aplicação destas técnicas é que, principalmente a obtenção de matriz de correlação cruzada de um vetor todo, é custosa computacionalmente e existem técnicas mais refinadas que se pode trabalhar com menos dados. Portanto, para a aplicação desta forma de estimar a direção de chegada implementa-se a Equação (4), onde $x_M + 1$ é o conjunto de sinais captados pelo elemento seguinte menos espaçado de x_M .

$$x_M[n] \otimes x_{M+1}[n] = \sum_{k=0}^N x_M^*[n] x_{M+1}[n+k] \quad (4)$$

sendo $x_M^*[n]$ o complexo conjugado de $x_M[n]$ e N o comprimento do sinal em amostras.

Será então gerado um vetor com o comprimento dos vetores somados. O pico da função é correspondente ao máximo de similaridade. Como x_k e x_{k+1} são sinais similares de mesmo comprimento e com atraso temporal entre os mesmos, pode-se encontrar a direção de chegada do sinal pela quantidade de amostras de diferença entre o pico de correlação e o ponto central do vetor. Espera-se que os vetores de correlação cruzada do arranjo possuam o mesmo tempo de atraso.

A aplicação do *beamformer* é feito para um ângulo de direcionamento, que é encontrado pelo processo descrito anteriormente. Deve-se deslocar os sinais de forma a alinhar suas fases para o ângulo escolhido e então somá-los. Tem-se assim uma descaracterização dos sinais em ângulos que

não sejam o escolhido para o direcionamento e reconstrução dos sinais que são emitidos do ângulo de escolha. Este método é descrito pela Equação (5) (Manolakis et al., 2005).

$$y(n) = \frac{1}{\sqrt{M}} \left[1 \ e^{j2\pi u_s} \ \dots \ e^{j2\pi(M-1)u_s} \right] \times \left\{ \begin{bmatrix} s(n) \\ e^{-j2\pi u_s} s(n) \\ \vdots \\ e^{-j2\pi(M-1)u_s} s(n) \end{bmatrix} + \mathbf{w}(n) \right\} \quad (5)$$

sendo y a saída do arranjo, M o número de elementos, s o sinal de interesse captado e $\mathbf{w}$ o ruído térmico dos sensores.

Sinais temporais são armazenados em vetores contendo os dados amostrados a cada período. Para o processamento de vetores é necessário que estes estejam armazenados em memórias. Portanto, para realizar a aquisição e processamento dos sinais, precisa-se que os mesmos estejam arquivados em uma ou mais memórias que se conectam a uma Unidade Lógica Aritmética, como na figura 1.

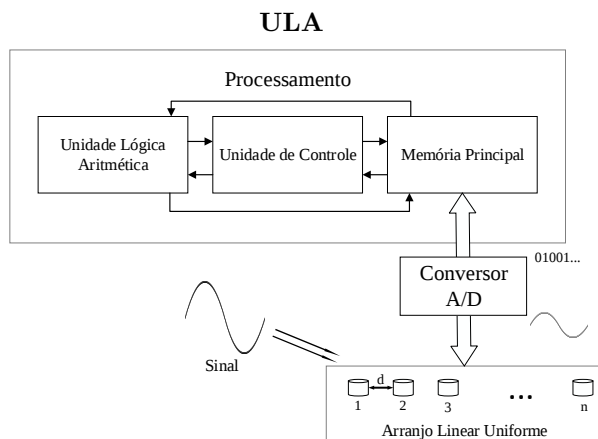


Figura 1: Captação de sinais em um arranjo de sensores e conexão com unidade de processamento

Os dados se agrupam em uma memória de forma binária em palavras de ordem 2^n e os dados coletados em circuitos integrados de conversores A/D podem variar dependendo da qualidade do circuito integrado e deve ser especificado de acordo com o excursionamento de magnitude presente no sinal captado.

A quantidade de amostras e o espaço binário que estas vão ocupar para armazenamento dependerá da frequência dos sinais de trabalho e devem ser especificadas de acordo com o caso. Na Seção 4 será definida a arquitetura para se trabalhar com um arranjo de microfones para sinais de áudio.

3 Texas Instruments TMS320F2833x/2823x

Nesta seção será feito um estudo sobre arquiteturas já conhecidas de microcontroladores e DSPs. Serão apresentados estudos sobre o DSP da família C2833x/C2823x de fabricação Texas Instruments, apresentando as limitações do mesmo para processamento de múltiplos sinais paralelos.

É de suma importância numa aplicação, em que são utilizados diversos sensores, que haja meio de se armazenar os sinais capturados pelos mesmos. Para isso, é necessário que haja módulos de conversão destes sinais contínuos para registros de dados de forma simultânea.

A família 2833x/2823x fabricada pela Texas Instruments são DSPs de desempenho médio, possuindo frequência de operação 100 (F283332) ou 150 (F283334 e F283335) MHz. Possuem temporizadores de 32 bits e memórias extensas de palavras de 16 bits. O conversor A/D é de 12 bits e com taxa máxima de 12.5 MHz (Instruments, 2007).

Segundo o manual do fabricante o módulo A/D destes dispositivos atua como 2 módulos de 8 canais independentes, podendo ser configurado como um de 16 canais em cascata. Sendo o módulo mais importante para a obtenção de sinais analógicos, explicitaremos suas características e limitações.

O conversor AD deste sistema funciona da seguinte forma. O processador envia um sinal de clock, denominado SYSCLKOUT, a um *prescaler* de alta velocidade, que controla o temporizador interno de conversão através de um *High Speed Clock*, (HSPCLK), e está ligado a um bloco de controle do sistema.

O bloco de controle do sistema envia dois sinais ao módulo, HALT e ADCENCLK. O primeiro é utilizado para deixar o módulo A/D em baixo consumo e parar o HSPCLK, parando o módulo. O ADCENCLK serve para, ao iniciar ou na ocorrência de reset, serve para que todos os registradores e modos fiquem em estado padrão. Este sinal também habilita o funcionamento do módulo (Instruments, 2007). O diagrama de blocos parcial, fornecido pelo fabricante, está representado na Figura 2.

A limitação deste módulo dá-se por conta dos canais do conversor A/D e como estes funcionam. Como dito anteriormente este módulo atua com 16 canais (ADCINA0:7 e ADCINB0:7) divididos entre dois multiplexadores e cada qual com seu circuito segurador de ordem zero. O conversor recebe os valores destas conversões e armazena-os em 16 registradores de resultado. Parte deste sistema está ilustrado na Figura 3.

Portanto este microcontrolador não possui capacidade para aquisição paralela devido à presença de apenas dois circuito S/H e que não trabalham em paralelo. Em sua taxa de aquisição má-

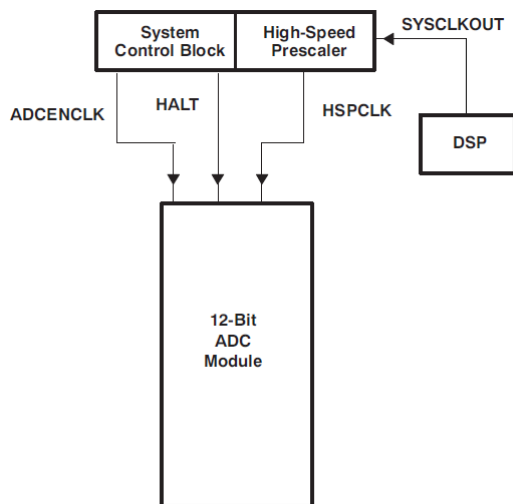


Figura 2: Comunicação e controle do DSP com o conversor AD. (Instruments, 2007)

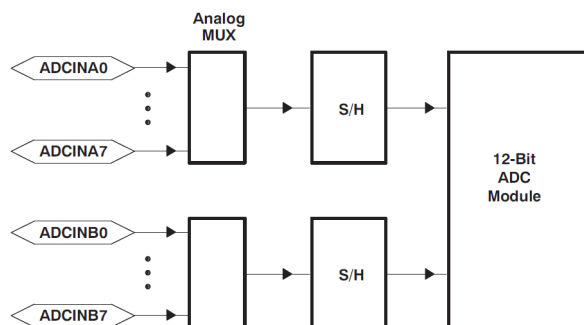


Figura 3: Comunicação dos pinos com o conversor A/D. (Instruments, 2007)

xima, têm-se a aquisição de uma amostra a cada 80 ns. Dependendo do tipo de sinal e meio com que se trabalha pode-se ter um atraso de fase a gerar ambiguidade (Seção 2.1) em caso de uma aquisição multiplexada.

Para um caso de multiplexação em um arranjo contendo 15 sensores, utilizando este dispositivo e extraindo sua capacidade máxima de 12,5 MHz, através do sombreamento dos canais A e B e desconsiderando qualquer outra utilização para processamento, haverá uma diferença de $1,2 \mu s$ entre a aquisição do primeiro e do último sensor. Isso, em um sinal de alta frequência com componentes importantes na faixa de 417 kHz não obedecerá o critério de Nyquist, gerando os erros mencionados. Este erro de tempo se propaga conforme a aquisição de um vetor de sinais para o processamento.

Conclui-se que este sistema somente é utilizável em caso de arranjos pequenos e de sinais de baixa ou média frequência. Como exemplo, Lin et al. (2016) utiliza um conceito de *Array Processing* para a construção de um radar de onda contínua modelada em frequência - FMCW - em uma aplicação automotiva. Estes sinais possuem largura de banda de até 1500 MHz, tornando in-

viável a utilização deste dispositivo em um sistema como este. Na próxima seção, será apresentada uma proposta de construção de arquitetura para múltiplas aquisições, assim como o circuito para implementação da mesma em um FPGA.

4 Arquitetura para Embarque em FPGA

Pretende-se trabalhar com sinais de voz na localização de fontes emissoras e filtragem em ambientes ruidosos. A frequência de um sinal de voz emitida por meio de fala pode variar até 4 kHz (Li et al., 2003). Portanto de acordo com a Equação (1) deve-se garantir que o sistema capte as amostras a pelo menos 8 kS/s.

O problema da amostragem paralela, tema central deste artigo, se dá pela incapacidade dos microcontroladores clássicos realizarem este tipo de aquisição, como mostrado na Seção 3. Será utilizado então um dispositivo de arquitetura configurável, o FPGA, junto de memórias e conversores A/D a serem integrados no sistema.

Todo FPGA trabalha apenas com lógica digital. Portanto estes não possuem conversores A/D, sendo necessário integrar periféricos para o processamento dos mesmos. São disponíveis em mercado conversores A/D que atuam com saída paralela ou serial. Por trabalharem apenas com lógica digital, os circuitos FPGA não possuem capacidade de aquisição de sinais para processamento. Deve-se então integrar tais circuitos externamente e utilizá-lo como processador entre entradas, armazenamento e saída.

4.1 Módulo de Conversão de Sinais

O conversor A/D é o elemento de maior importância em um sistema digital que necessita trabalhar com sinais analógicos. Estes são disponíveis em diversas configurações.

Estes circuitos atuam sob pulsos de *clock* e têm em sua entrada um sinal de tensão analógica. Para o caso adotado nesta seção, um sinal proveniente de um microfone. O conversor então digitaliza este sinal, transformando seus níveis de tensão em bits de acordo com as tensões elétricas de referência colocadas nos mesmos. A saída em níveis lógicos 0 ou 1 é dada de forma paralela ou serial, com uma frequência e resolução dependendo do dispositivo utilizado.

As topologias de conversão são diversas como: rampa, aproximações sucessivas, rampa dupla, *flash*, sigma-delta e tensão-frequência (TOCCI and WIDMER, n.d.). Cada uma destas formas de conversão utilizadas nestes componentes possuem suas particularidades e fica por conta do projetista a escolha de qual circuito utilizará.

Em aplicações de *array processing* deve-se garantir o paralelismo entre os sinais. Portanto deve-se utilizar um circuito conversor A/D para cada

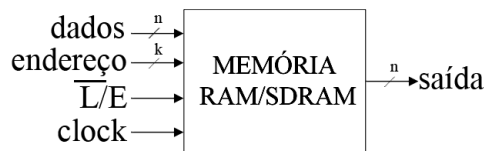


Figura 4: Memória RAM genérica de n bits e k bits de endereços

elemento no arranjo. Portanto, o processamento torna-se possível apenas com a presença destes componentes. O próximo elemento a ser discutido é a memória. A utilização de memórias é importante para armazenamento dos dados obtidos para processamento.

4.2 Memórias RAM/SDRAM

O próximo circuito que deve ser integrado ao sistema são memórias para gravação dos dados. Para os dados que serão processados são necessários vetores de amostras dos mesmos para a aplicação das técnicas. Necessita-se então que se tenha uma quantidade n_s de amostras.

Cada aquisição temporal pode ser salva em uma posição de memória de um dispositivo SDRAM, que apresentam funcionamento rápido. Opta-se por utilização de memórias SDRAM pois almeja-se um sistema que funcione em tempo real, portanto, não irá salvar dados para futura utilização.

O processo de leitura e escrita na memória RAM é controlada por ciclos de *clock* e por nível lógico em um pino de escrita/leitura. Em aplicações de processamento de sinal, utiliza-se estas memórias para armazenar temporariamente os dados adquiridos nos sensores. Portanto, para cada aquisição realizada pelos M sensores, habilita-se a escrita na memória RAM e retém-se os valores das amostras daquele instante de tempo. Estes dados estarão salvos na memória RAM enquanto houver alimentação da mesma. O diagrama de uma memória RAM genérica pode ser visto na Figura 4.

4.3 Montagem do sistema de aquisição de dados sonoros

Nas subseções anteriores, foram apresentados conceitos de periféricos que devem ser integrados ao sistema processador de um *array*. Será gerado então um sistema virtual que pode ser aplicado. Este sistema pode ser embarcado em qualquer dispositivo que se trabalhe com paralelismo e tenha sua arquitetura reconfigurável. A Tabela 1 apresenta características de projeto para trabalho com estes sinais.

onde B é a frequência máxima do sinal de interesse, f_s é a frequência de amostragem adotada,

Tabela 1: Parâmetros de trabalho com dados provenientes de voz humana

Parâmetro	Unidades
B	4 kHz
f_s	44.1 kS/s
c	343,3 m/s
d	4,29 cm
n_s	1024

c é a velocidade do som no ar, d é o espaçamento entre os sensores e n_s é a quantidade de amostras que necessita-se para aplicação da correlação cruzada.

Com estes dados, pode-se fazer a escolha dos componentes que serão utilizados. Como objeto de estudo não haverá preocupação com números de pinos em componentes. Definimos como 8 o número de elementos no arranjo. Define-se um conversor A/D de 44.1 kS/s e 8 bits. Escolhe-se esta frequência de amostragem para uma melhora na resolução do sistema. Tem-se, por iteração de filtragem e/ou estimação, 8192 bytes de dados. Como será necessário utilizar espaço de memória para as matrizes de correlação cruzada, que ocupam $n_s \times 2 \times (M - 1) \times 8$ bits, são necessários então de 14336 bytes adicionais. Este valor pode ser encontrado facilmente em memórias comerciais, tendo ela espaço de palavra de 2 bytes e 11 bits de endereços ou 1 byte de palavra e 13 bits de endereços. Esta escolha fica a escolha do projetista.

A ULA deste sistema deve contar com circuitos somadores e multiplicadores e ser capaz de acessar a memória de modo randômico. Seus barramentos devem conter 8 bits para um registrador de instruções, este com pelo menos 3 bits de instrução para as operações de soma, escrita, apagar, deslocamento e transferir, que são as operações básicas para realização dos procedimentos descritos anteriormente.

Um exemplo que pode ser aplicado neste sistema é a utilização do FPGA Cyclone IV EP4CE115F29, presente no kit DE2-115 da ALTERA. Este FPGA conta com 114480 elementos lógicos configuráveis, memória de 3888 Kb, além das duas memórias de 64MB presentes no kit, e mais de 200 pinos que podem ser utilizados pelo usuário.

5 Conclusões

Neste estudo, apresentou-se o que é necessário para a montagem de um dispositivo de aquisição e processamento de múltiplos dados destinado à *array processing*. A generalização proposta torna acessível a implementação para qualquer sistema desde que sejam obedecidos os critérios de construção também apresentados aqui.

Objetiva-se então a implementação deste sis-

tema em um FPGA para trabalhos futuros e aplicação de técnicas mais refinadas. Pode-se perceber que a complexidade do sistema aumenta conforme deseja-se um melhor desempenho, aumentando o número de sensores, a frequência de amostragem ou as técnicas a serem empregadas.

Agradecimentos

Agradecemos à instituição de ensino UTFPR por ter possibilitado a realização deste trabalho e à Fundação Araucária processo nº 338/12 pelo kit utilizado.

Referências

- Fan, Y. (2011). FPGA-based data acquisition system, *Signal Processing, Communications and Computing (ICSPCC), 2011 IEEE International Conference on*, pp. 1–3.
- Fei, W., Zhijie, W., Hong, C. and Yi, X. (2011). High-speed data acquisition system based on FPGA/SoPC, *Electronic Measurement Instruments (ICEMI), 2011 10th International Conference on*, Vol. 1, pp. 24–27.
- Instruments, T. (2007). *TMS320F2883x/2823x*. Revised August 2012.
- Jan, E., Svaizer, P. and Flanagan, J. L. (1995). Matched-filter processing of microphone array for spatial volume selectivity, *Circuits and Systems, 1995. ISCAS '95., 1995 IEEE International Symposium on*, Vol. 2, pp. 1460–1463 vol.2.
- Li, Y., Ho, K. C. and Kwan, C. (2003). Design of broad-band circular ring microphone array for speech acquisition in 3-d, *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on*, Vol. 5.
- Lin, J.-J., Li, Y.-P., Hsu, W.-C. and Lee, T.-S. (2016). Design of an FMCW radar baseband signal processing system for automotive application, *Springer* **5**(1): 1–16.
- Manolakis, D. G., Ingle, V. K. and Kogon, S. M. (2005). *Statistical and adaptive signal processing: spectral estimation, signal modeling, adaptive filtering, and array processing*, Vol. 46, Artech House Norwood.
- Naidu, P. S. (2009). *Sensor array signal processing*, CRC press.
- Netti, A., Diodati, G., Camastra, F. and Quaranta, V. (2015). FPGA implementation of a real-time filter and sum beamformer for acoustic antenna, *Internoise2015* **129**: 279–291.
- Oppenheim, A. V., Schaffer, R. W., Buck, J. R. et al. (1989). *Discrete-time signal processing*, Vol. 2, Prentice hall Englewood Cliffs, NJ.
- Oswal, A., Jha, A., Neal, S., Reid, A., Bradbury, D., Aston, P., Limousin, P., Foltynie, T., Zrinzo, L., Brown, P. and Litvak, V. (2016). Analysis of simultaneous MEG and intracranial LFP recordings during deep brain stimulation: A protocol and experimental validation, *Journal of Neuroscience Methods* **261**: 29–46.
- Qiang, Q., Minglu, J. and Qian, L. (2008). Cyclostationary LMS beamforming algorithm, *Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08. 4th International Conference on*, pp. 1–4.
- Sarevska, M. and Salem, A.-B. (2008). Antenna array beamforming algorithms for smart antenna system, *World Academy of Science, Engineering and Technology* **24**: 941–945.
- Tian, Z., Bell, K. and Van Trees, H. (2001). A recursive least squares implementation for LCMP beamforming under quadratic constraint, *IEEE Transactions on Signal Processing* **49**(6): 1138–1145.
- TOCCI, R. and WIDMER, N. S. (n.d.).
- Treder, M., Porbadnigk, A., Shahbazi Avarvand, F., Müller, K.-R. and Blankertz, B. (2016). The LDA beamformer: Optimal estimation of ERP source time series using linear discriminant analysis, *NeuroImage* **129**: 279–291.
- Van Trees, H. L. (2004). *Detection, estimation, and modulation theory, optimum array processing*, John Wiley & Sons.
- Van Veen, B. and Buckley, K. (1988). Beamforming: A versatile approach to spatial filtering, *IEEE ASSP Magazine* **5**(2): 4–24.
- Vemuri, R. and Harr, R. (2000). Configurable computing: Technology and applications, *Computer* **33**(4): 39–40.
- Yasin, M., Akthar, P., Khan, M. and Naqvi, S. (2010). Enhanced sample matrix inversion is a better beamformer for a smart antenna system, *World Applied Sciences Journal* **10**(10): 1167–1175.
- Zhang, X., Wang, Z. and Xu, D. (2008). Wavelet packet transform-based least mean square beamformer with low complexity, *Progress In Electromagnetics Research* **86**: 291–304.