

Subsistema multiplataforma para controle de aquisição, visualização e organização de dados do Espectrômetro Digital de RM: ToRM Console

Multi-platform subsystem for controlling acquisition, visualization and data organization of an NMR Digital Spectrometer: ToRM Console

Danilo M. D. D. da Silva, Daniel C. Pizetta, Guilherme M. Freire, Felipe B. Coelho, Gustavo V. Lourenço, Rodrigo R. M. Corrêa, Mateus J. Martins, Edson L. G. Vidoto e Alberto Tannús

Centro de Imagens e Espectroscopia 'in vivo' por Ressonância Magnética (CIERMag), Instituto de Física de São Carlos (IFSC), Universidade de São Paulo (USP) – São Carlos (SP), Brasil.

Resumo

Neste trabalho serão apresentados os resultados recentes do desenvolvimento do Espectrômetro Digital de Ressonância Magnética do CIERMag – subsistema para controle de aquisição, visualização e organização dos dados. Alguns aspectos como a arquitetura e funcionalidades serão mostrados, incluindo uma demonstração com o método CPMG para o cálculo do tempo de relaxação transversal (T_2) utilizando o sistema. Com estas conquistas, o ToRM Console está sendo preparado para ser um sistema de MRI em um futuro próximo.

Palavras-chaves: desenvolvimento de software, instrumentação de ressonância magnética.

Abstract

In this work, we present the recent results from the development of the CIERMag NMR Digital Spectrometer – a subsystem for controlling acquisition, visualization and data organization. Some aspects of the architecture and features will be shown, including a demonstration with CPMG method for transversal relaxation time (T_2) measurement using the system. With these achievements, ToRM Console is now being prepared to be an MRI scanner in a near future.

Keywords: magnetic resonance, software development, instrumentation.

Introdução

As técnicas de ressonância magnética apresentam um papel importante em diversas áreas do conhecimento devido a algumas características fundamentais em certas aplicações, como, por exemplo, não ser invasiva ou destrutiva.

No entanto, estudos sobre a difusão e incorporação da tecnologia de ressonância magnética no Brasil¹ demonstraram que a participação do país nesses temas é consideravelmente baixa em relação aos países desenvolvidos. Tanto os equipamentos utilizados para fins clínicos, quanto aqueles localizados nos centros de pesquisas nacionais são de alto custo, justificando, em

parte, o *déficit* de uso dessa tecnologia. Além disso, a falta de flexibilidade no desenvolvimento de métodos de sequências de pulsos, bem como a dificuldade para se realizar tal tarefa em equipamentos comerciais são consequências do uso de *softwares* e *hardwares* proprietários fechados. Sendo assim, a utilização de novas técnicas possui forte dependência com os fabricantes desses equipamentos, inviabilizando em algumas delas o seu desenvolvimento ou adaptação a novos problemas.

Neste sentido, o Centro de Imagens e Espectroscopia 'in vivo' por Ressonância Magnética tem se empenhado na criação de um espectrômetro completamente digital de ressonância magnética, versátil o suficiente para que possa

Autor correspondente: Danilo Mendes Dias Delfino da Silva – Centro de Imagens e Espectroscopia in vivo por Ressonância Magnética (CIERMag) – Instituto de Física de São Carlos – Universidade de São Paulo (USP) – Avenida Trabalhador são-carlense, 400 – Centro – CEP: 13566-590 – São Carlos (SP), Brasil – E-mail: danilomendes@usp.br / danilomendesdias@gmail.com

ser utilizado em um Tomógrafo (TDRM – Projeto ToRM-15), com concepção completa e proprietária do CIERMag.

Neste projeto, duas características são fundamentais para contornar os problemas apontados anteriormente: o uso de *hardware* reconfigurável, por meio do uso de *Field Programmable Gate Arrays (FPGAs)*² e a existência de uma linguagem própria para o desenvolvimento de sequências de pulsos (a linguagem F). Esses recursos permitem o controle da tecnologia utilizada e uma maior imunidade à obsolescência. A Figura 1 mostra o espectrômetro do CIERMag ligado a um osciloscópio

Este trabalho tem por finalidade demonstrar os resultados de parte do sistema que compõe o Projeto ToRM-15, o subsistema ToRM Console, em uma contínua iteração no processo de concepção e desenvolvimento. O papel do ToRM Console é prover um ambiente de execução e visualização em tempo real de experimentos de RM da forma flexível, adequada ao ambiente de pesquisa no qual cada vez mais se encontram aplicações para as técnicas de RM.

Material e Métodos

Além do ToRM Console, outras aplicações estão sendo desenvolvidas pelo grupo para que o espectrômetro possa funcionar de maneira completa. Entre elas, o desenvolvimento de um compilador para a linguagem F (além da concepção da própria linguagem), designada para desenvolvimento de sequência de pulsos de forma bastante simplificada e um respectivo ambiente integrado de desenvolvimento (ToRM IDE). Este último irá permitir o desenho de métodos de RM com a mesma abordagem que se utiliza para projetos de *software*, acrescida da possibilidade de programação gráfica, mais intuitiva.

Alguns recursos essenciais para o entendimento do funcionamento do ToRM Console serão explicados a seguir. A Figura 2 mostra a interação entre os subsistemas que compõem o espectrômetro.

ToRM ACQServer

Devido a escolha da arquitetura do sistema, o ToRM Console não se comunica diretamente com o espectrômetro. A vantagem dessa abordagem é a possibilidade de execução remota, via *internet* ou rede local, bem como o acompanhamento de um mesmo experimento por vários computadores.

Para que isso fosse possível, o ToRM ACQServer foi desenvolvido pelo grupo com o intuito de fornecer uma camada de comunicação direta com o *hardware*, bem como prover uma interface de comunicação com o ToRM Console. Na sua concepção, este subsistema pode ser interpretado como um *Hardware Abstraction Layer (HAL)*, que “conhece” a estrutura do *hardware* do espectrômetro e se comunica com as camadas superiores através de um protocolo uniforme.

Atualmente, a comunicação com o ToRM Console é feita pelo protocolo FTP com algumas alterações para permitir envio de comandos e sequências de pulsos compiladas para o espectrômetro. O ACQServer se comunica

com o *hardware* do espectrômetro através de USB ou *Gigabit Ethernet*, com precedência para este último.

Aba de parâmetros

Provida de reaproveitamento de código do ToRM IDE por possuir funcionalidades muito semelhantes, a aba de parâmetros permite com que o usuário possa alterar os valores correntes dos parâmetros. Dois grupos de parâmetros são definidos: parâmetros do sistema e parâmetros do método, ambos sujeitos a validação anteriormente à execução de cada *scan*.

A validação é feita através da verificação de limites superiores e inferiores nos valores dos parâmetros pré-determinados na etapa de desenvolvimento e scripts de validação. Esses scripts, escritos em Python pelo próprio desenvolvedor do método, são necessários para garantir que complexas inter-relações entre os parâmetros sejam satisfeitas.

Escolha de linguagens e bibliotecas

Com o aumento da velocidade da evolução tecnológica, fatores como obsolescência e portabilidade devem ser considerados no desenvolvimento de *software* de sistemas que devem possuir tempo de vida longo. Dessa maneira, a linguagem de programação *Python* mostrou-se bem atrativa por permitir o funcionamento do mesmo código em múltiplos sistemas operacionais *desktop*, incluindo *Microsoft Windows* e *Linux*, e possuir confiáveis e ricas bibliotecas para os mais diversos fins³. Essas bibliotecas são mantidas por uma forte comunidade de *software* livre e garantem rapidez no desenvolvimento de rotinas de processamento de sinais, interface com usuários, gráficos, acesso a bases de dados⁴⁻⁷, etc.

Resultados

Controle de aquisição e visualização dos experimentos em tempo real

Como podemos visualizar na Figura 3, o controle de aquisição e visualização dos dados possui as seguintes

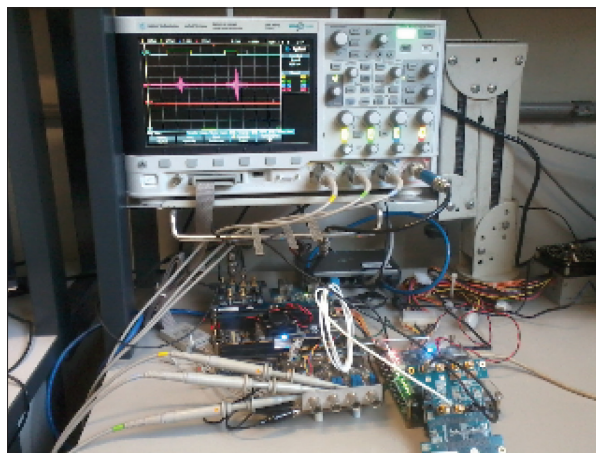


Figura 1. Espectrômetro CIERMag em execução para fins de testes.

etapas: obtenção dos dados através do ACQServer por meio de protocolo FTP, filtragem dos dados, visualização e armazenamento dos dados organizados. Essas tarefas são executadas concorrentemente, com o auxílio da biblioteca padrão *multiprocessing* do Python, com o uso de filas e eventos que atuam como variáveis compartilhadas com proteção para leitura/escrita por vários processos simultaneamente.

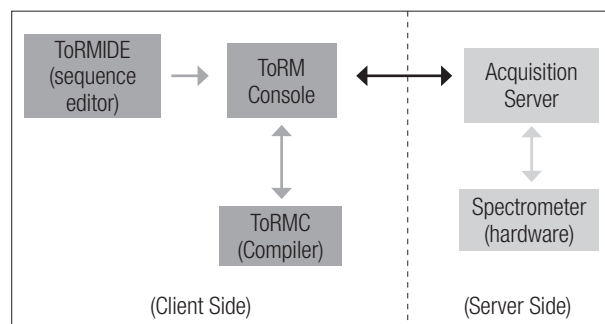


Figura 2. Arquitetura simplificada do sistema.

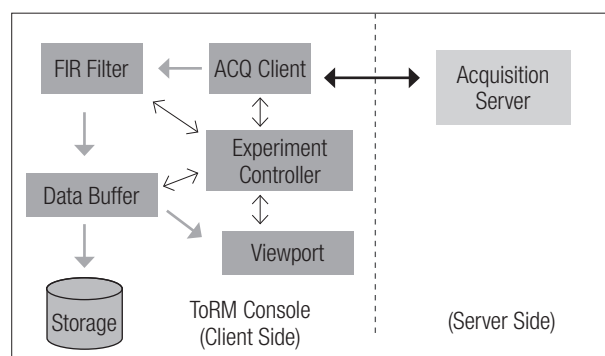


Figura 3. Arquitetura do sistema de aquisição e visualização do ToRM Console.

Os índices dos dados recém-chegados são calculados e fornecidos pelo controlador do experimento, que contém a estrutura dos *loops* da sequência de pulsos. Desta forma, é possível, por exemplo, saber em tempo real quantas médias já foram realizadas. Além disso, os dados filtrados finais são armazenados em formato HDF5⁸, um formato que permite armazenamento rápido e organizado de grandes arquivos.

Filtros FIR

Os filtros FIR são filtros digitais de resposta ao impulso finita. Por não utilizarem realimentação, os erros de arredondamento não são acumulados. Utilizamos filtros FIR para criação de filtros passa-baixa, de modo a remover componentes de frequência fora da região de interesse e definir a banda final com que se pretende analisar os sinais de RM. Após a filtragem, o ruído presente em frequências maiores do que a frequência de corte será eliminado e, obedecendo ao critério de Nyquist, poderão ser aplicadas decimações, de modo a reduzir a quantidade de dados sem perda de informação. O ruído presente na região de frequência de interesse poderá ser removido com uso de médias.

Embora o *hardware* do espectrômetro possua alguns filtros digitais em sua implementação, eles não podem ser facilmente alterados. Sendo assim, optou-se pela aplicação adicional de filtros na etapa de aquisição pelo ToRM Console, uma vez que estes filtros podem atuar em cascata.

Para a implementação dos filtros, utilizamos a biblioteca *Scipy*⁹. Os coeficientes dos filtros — ou janelas — podem ser escolhidos dinamicamente. Assim, o operador pode escolher a janela entre as opções disponíveis: “*blackman*”, “*hamming*”, “*hanning*”, “*hann*”, entre outras, além do número de coeficientes do filtro. A frequência de corte deve ser informada na aba de parâmetros. As Figuras 4 e 5 ilustram a aplicação do filtro em um conjunto de dados sintetizados contendo ruído.

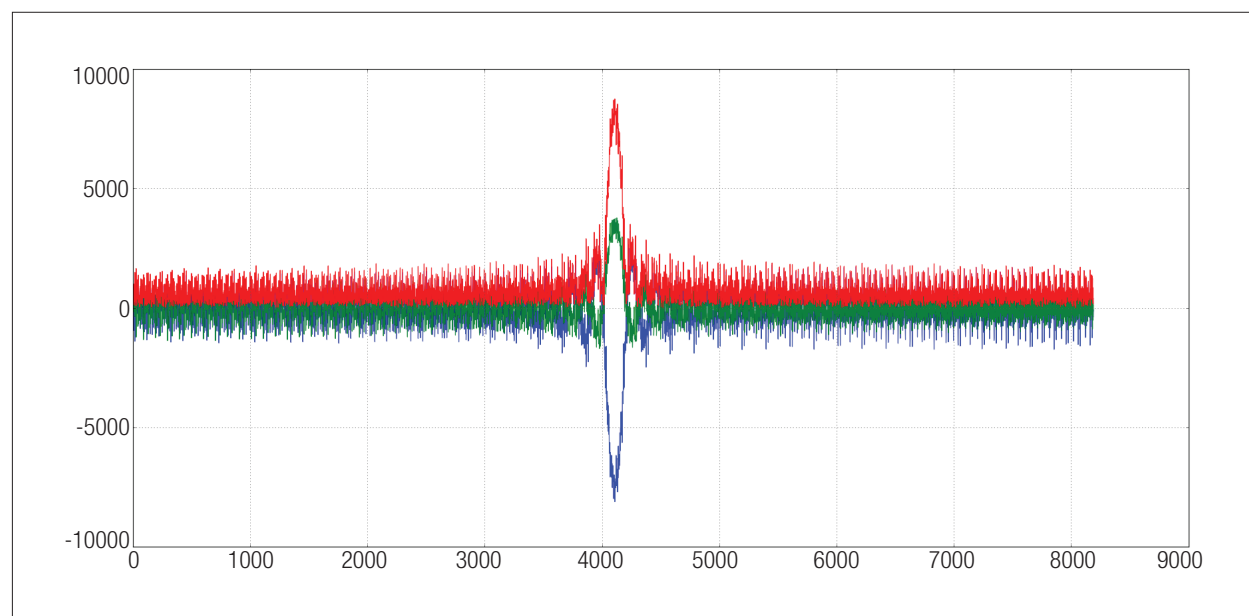


Figura 4. Aquisição de sinal de eco sintetizado com ruído em banda de 500 KHz.

Organização dos dados

A organização de dados do sistema consiste na criação de uma matriz multidimensional de coeficientes complexos, que depende do número de aquisições em cada dimensão e do número de pontos complexos em cada aquisição. Entende-se por dimensão do experimento o número de iterações ou *loops* utilizados na implementação da sequência de pulsos.

Como não existe obrigatoriedade de uso de nomenclatura reservada na declaração de *loops* na linguagem F (um dos principais atributos desta abordagem), a chave "Type" é utilizada para marcar alguns tipos de *loops* especiais. São eles : *Average*, *Slices*, *Split*, *PhE1*, *PhE2* e *PhE3* *EchoTrain* e *EchoTrainPhE1*, *EchoTrainPhE2* e *EchoTrainPhE3*.

Cada tipo especial corresponde a um tratamento distinto durante a etapa de organização. Deste modo, uma estrutura em formato JSON é criada após o processo de compilação contendo a composição dos *loops* que será necessária para a reconstrução e visualização dos dados em tempo de aquisição pelo ToRM Console.

A dimensão do tipo *Average* é colapsada durante a aquisição, pois os dados com o mesmo índice de *Average* ou de média, serão somados e, no fim da aquisição, normalizados.

Loops do tipo *Split*, normalmente mais externos, são utilizados para admitir um número de fatias, ou *slices*, maior do que o tempo de repetição possa permitir. Se um determinado tempo de repetição permite, por exemplo, a realização de 4 *slices*, então é possível, na mesma sequência, obter 8 *slices* caso exista um *loop Split* com duas iterações. No entanto, durante organização final dos dados, a dimensão *Split* será

colapsada, uma vez que ela não possui significado, e a dimensão de *Slices* terá mais elementos.

Loops de *EchoTrain* são utilizados apenas para visualização de múltiplos ecos em tempo real, uma vez que a visualização padrão é a de uma aquisição por vez e cada eco é gerado em uma aquisição diferente.

Os *loops PhE1*, *PhE2* e *PhE3* são utilizados para marcar as direções de codificação de fase e são de uso obrigatório no desenvolvimento de sequências multidimensionais para MRI.

Por fim, *loops* de *EchoTrainPhE1*, *EchoTrainPhE2*, *EchoTrainPhE3* são usados para permitir a visualização de múltiplos ecos, cada um deles com uma codificação de fase distinta. O respectivo identificador de direção de codificação de fase deverá, obrigatoriamente, ser utilizado em um *loop* mais externo. Para que isso seja possível, os índices deverão ser recalculados na reorganização de dados. A Figura 6 demonstra a execução de um método que contém *loops* do tipo *Average* e *Echo*

Método CPMG para cálculo de T2

Para demonstrar o funcionamento do sistema, foi realizado um experimento de relaxometria utilizando o método CPMG^{10,11} em uma amostra contendo solução de CuSO₄ em alta concentração.

Os seguintes parâmetros foram utilizados: 256 ecos, contendo 20 pontos cada, tempo de aquisição de 200 ms, sem a necessidade de utilização de médias. O valor obtido para o valor de relaxação transversal (T_2) foi de 14,5 ms, conforme mostrado na Figura 7.

A Figura 8 ilustra a execução de outro scan utilizando o método CPMG, desta vez contendo apenas 64 ecos.

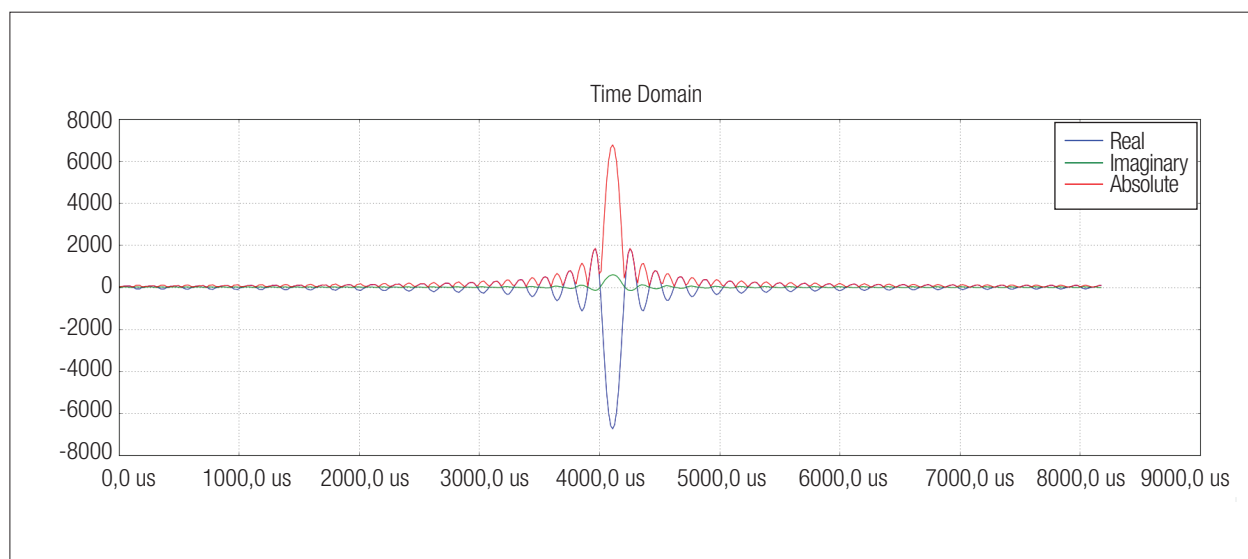


Figura 5. Aquisição de sinal de eco sintetizado com ruído (mesmo *scan* da Figura 4) após aplicação de filtro FIR com frequência de corte de 31,250 KHz e aplicação de 128 médias.

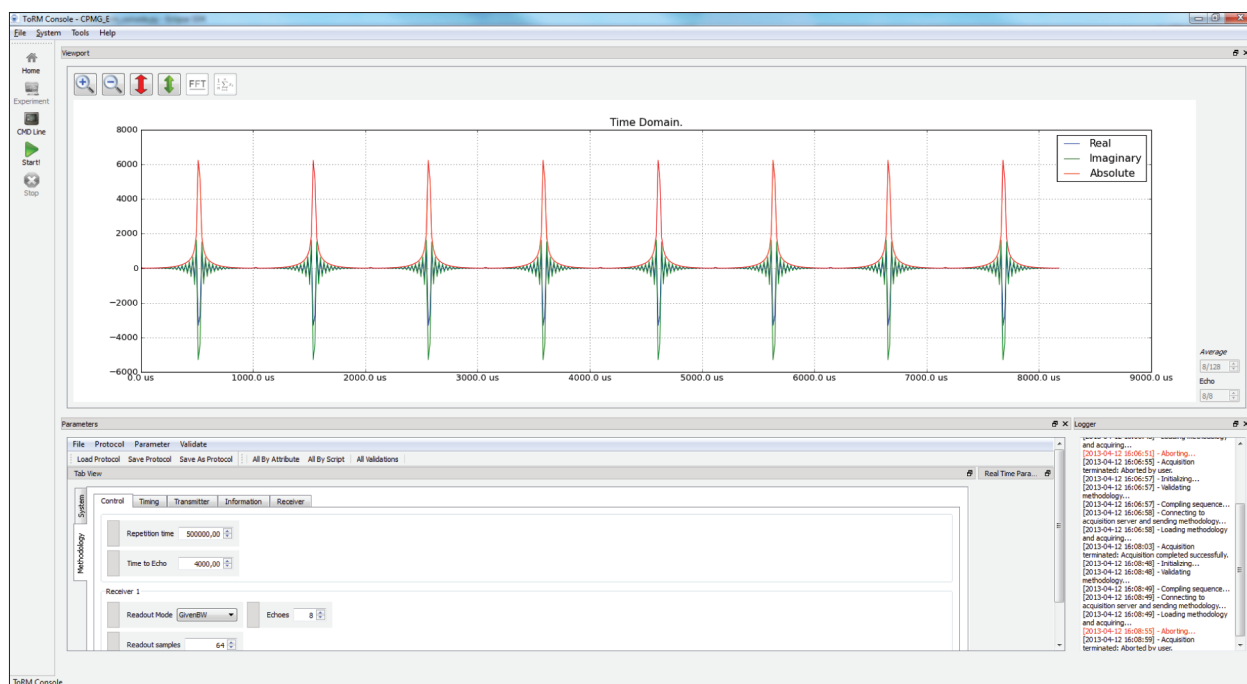


Figura 6. ToRM Console em execução com dados gerados por sintetizador de RF implementado em FPGA simulando 8 ecos.

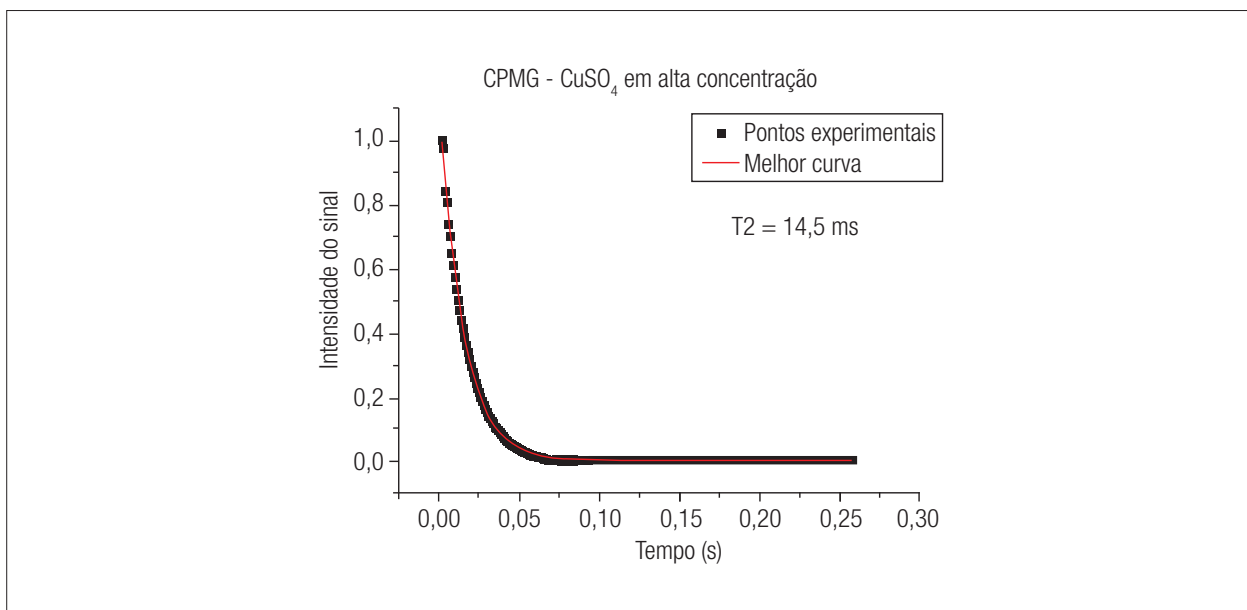


Figura 7. Demonstração de experimento de relaxometria com dados experimentais obtidos pelo ToRM Console utilizando método CPMG.

Discussão e Conclusões

Para atingir o objetivo final deste trabalho, que é a criação de um ambiente para aquisição e visualização de experimentos de RM, seja em espectroscopia ou imagens, algumas etapas foram necessariamente percorridas. Em 2013, o sistema de controle de

aquisição, a implementação dos filtros e a organização dos dados garantem a execução de testes de *hardware* e pequenos experimentos de relaxometria. A partir desse ponto, outras etapas poderão ser implementadas no sistema, como: criação de imagens e estabelecimento de um ambiente de processamento sobre os dados adquiridos.

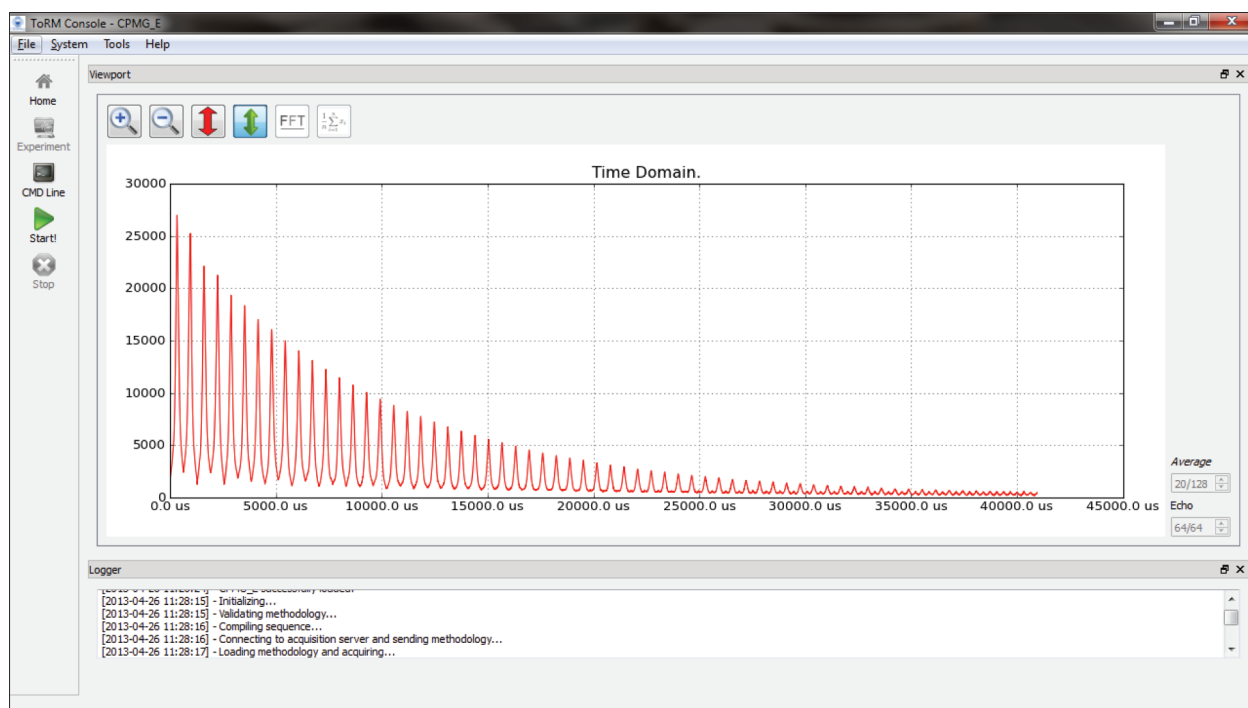


Figura 8. Outro experimento com método CPMG, agora utilizando 64 ecos, apenas para ilustrar a visualização no ToRM Console. Neste modo de visualização, o eixo temporal contabiliza apenas o tempo de aquisição.

Agradecimentos

Agradecemos a toda equipe do CIERMag, além das agências de fomento tecnológico pelo apoio ao projeto (FAPESP: 2005/5.6663-1 e CNPQ: 565.047/2010-8, CAPES, FINEP, FNS).

Referências

1. Andrade EIG, Daniel Resende Faleiros; Camila Lins Rodrigues; Catari Vilela Chaves; Eduardo da Mota e Albuquerque. Análise da evolução do conhecimento e da difusão e incorporação da tecnologia de ressonância magnética no Brasil. In: IV Congresso Brasileiro Ciências Sociais e Humanas em Saúde. Salvador; 2007.
2. Altera. FPGAs 2013. [cited 2013 May 10]. Available from: <http://www.altera.com/products/fpga.html>.
3. Python. Python Documentation. 2013. [cited 2013 May 10]. Available from: <http://www.python.org/doc>
4. The Scipy Community. NumPy Reference. 2013. [cited 2013 May 10]. Available from: <http://docs.scipy.org/doc/numpy/reference>
5. PySide - Pyside - Python for Qt. [cited 2013 May 10]. Available from: <http://qt-project.org/wiki/PySide/FAQ>
6. Matplotlib. User's Guide. 2013. [cited 2013 May 10]. Available from: <http://matplotlib.org/users/index.html>
7. MySQL Python. MySQLdb User's Guide. 2013. [cited 2013 May 10]. Available from: <http://mysql-python.sourceforge.net/MySQLdb.html>
8. The HDF Group. HDF5 Software Documentation. 2012. [cited 2013 May 10]. Available from: <http://www.hdfgroup.org/HDF5/doc/index.html>
9. The Scipy Community. Scipy. 2013. [cited 2013 May 10]. Available from: <http://docs.scipy.org/doc/numpy/reference>
10. Carr HY, Purcell EM. Effects of Diffusion on Free Precession in Nuclear Magnetic Resonance Experiments. *Physical Review*. 1954;94(3):630-8.
11. Meiboom S, Gill D. Modified Spin-Echo Method for Measuring Nuclear Relaxation Times. *Rev Sci Instrum*. 1958;29:688-91.