

Projeto de um Analisador Multicanal (MCA) baseado em Módulo Programável com FPGA

Design of a Multichannel-Analyzer based on FPGA Programmable Module*

Guilherme Augusto Santos da Silva[†]

Centro Brasileiro de Pesquisas Físicas – CBPF, Rua Dr. Xavier Sigaud, 150 – Urca – Rio de Janeiro – RJ - Brasil

Herman P. Lima Jr

Centro Brasileiro de Pesquisas Físicas – CBPF, Rua Dr. Xavier Sigaud,
150 – Urca – Rio de Janeiro – RJ - Brasil

Submetido: 31/08/2018 Aceito: 12/02/2019

Resumo: Este documento descreve o projeto de um Analisador Multicanal (MCA) utilizando como hardware um módulo de aquisição de dados baseado na tecnologia *Field Programmable Gate Array (FPGA)* e na linguagem de programação VHDL. Uma das aplicações imediatas do MCA é a caracterização de detectores a gás utilizando fontes radioativas de Ferro-55.

Palavras-chave: Termos de indexação: Detectores a Gás; Instrumentação; FPGA; VHDL.

Abstract: This paper describes the design of a Multichannel Analyzer (MCA) with an available scientific purpose data acquisition module. A high-speed analog-to-digital converter and a Field Programmable Gate Array (FPGA) available on the module makes it well suited for such development. A very useful foreseen application of that MCA is the characterization of gas detectors by using Iron-55 radiation sources.

Keywords: Gas-filled Chamber Detectors; Instrumentation; FPGA; VHDL.

Introdução

A detecção e a medida de radiação ionizante são importantes em várias áreas da ciência, engenharia e saúde. Equipamentos capazes de identificar, quantificar e processar a radiação ionizante são amplamente estudados e aprimorados para atender os mais diversos objetivos. O desenvolvimento constante de novos instrumentos científicos fornece recursos inovadores que possibilitam avanços nos métodos e nos processos de detecção.

Um destes equipamentos é o Analisador Multicanal (*Multichannel Analyzer - MCA*). Este equipamento se destina a realizar a detecção e a medida diferencial da quantidade de eventos ocorridos em janelas definidas de energia da radiação, através da quantização da amplitude do pulso detectado ($dN/dH \times H$). Este tipo de medida é crucial pois fornece informações importantes sobre a natureza de um evento radioativo, bem como a definição do comportamento de um detector e sua eletrônica de leitura.

O avanço na tecnologia que pode ser utilizada nestes equipamentos contribuiu muito para o aperfeiçoamento das medidas de radiação existentes possibilitando, assim, sua aplicação em diversas áreas da ciência, bem como em processos industriais de alto desempenho.

O que se propõe neste trabalho é a implementação de

um Analisador Multicanal utilizando conversores analógico-digitais de alta velocidade (125 MHz) e a tecnologia conhecida por lógica programável (dispositivos FPGA), inicialmente afim de realizar a caracterização dos detectores a gás com fonte radioativa de Fe-55 desenvolvidos no LSD (Laboratório de Sistemas de Detecção / CBPF).

Analisadores Multicanal (MCA)

Em última análise, o Analisador Multicanal é um instrumento usado para se obter o espectro de energia de determinados eventos. Ele pode ser descrito como uma sequência de instrumentos conhecidos como Analisador de um Único Canal (Single-Channel Analyzer – SCA), ou Discriminador Diferencial.

Um SCA é ajustado para que sua saída transmita um pulso lógico se, e somente se, em sua entrada seja reconhecida uma amplitude de pulso dentro de um intervalo pré-determinado.

É natural imaginar que agrupando-se uma sequência de SCAs, onde cada um corresponde a uma amplitude de pulso diferente, para uma mesma janela temporal de eventos consegue-se detectar simultaneamente várias amplitudes de pulso.

Considerando uma única entrada, e uma sequência de três SCAs, cada um ligado em um dispositivo de registro, que acumula cada evento detectado, sendo o primeiro deste correspondente à menor amplitude de pulso observado, e o último a maior amplitude do mesmo, ao final da observação tem-se um perfil de distribuição de amplitude dos eventos em cada SCA.

Cada intervalo do SCA é o que em um MCA recebe o

*Revista Brasileira de Ciência, Tecnologia e Inovação (RBCTI), n. 01, volume 01, jul./dez. ano 2013, p. XX-XX. Brazilian Journal of Science, Technology and Innovation.

[†]Electronic address: silvaga@cbpf.br

nome de *canal*. Cada canal corresponde a uma faixa de amplitude (altura) do pulso (H).

Medindo-se o número total de eventos em um determinado canal (dN), dividindo este número pelo diferencial de altura de pulso de cada canal (dH), e os distribuindo de acordo com a altura de pulso (H), temos uma distribuição diferencial destas alturas de pulso. Este processo define o funcionamento básico de um MCA.

Um MCA apresenta algumas características gerais, como por exemplo o número de canais. A escolha do número de canais é ditada principalmente por dois fatores:

- a resolução necessária;
- número total de contagens por canal

Quanto maior o número de canais disponíveis, naturalmente menor será largura de cada canal, o que idealmente leva o resultado da distribuição discreta o mais próximo possível do sinal original contínuo. Se não houvesse nenhuma restrição na quantidade de contagens que podem ser acumuladas como, por exemplo, limite de memória no dispositivo de armazenamento utilizado, a quantidade de canais poderia ser tão grande quanto desejado. Obviamente sempre existem limitações, sejam de ordem técnica, ou de relação custo-benefício.

A resolução de um detector é, usualmente, dada pela seguinte pela equação 1, e representado na figura 1:

$$R = \frac{FWHM}{H_0} \quad (1)$$

onde:

FWHM é a Largura a Meia Altura do sinal (*Full Width at Half Maximum*)

H_0 é o valor central do pico de sinal

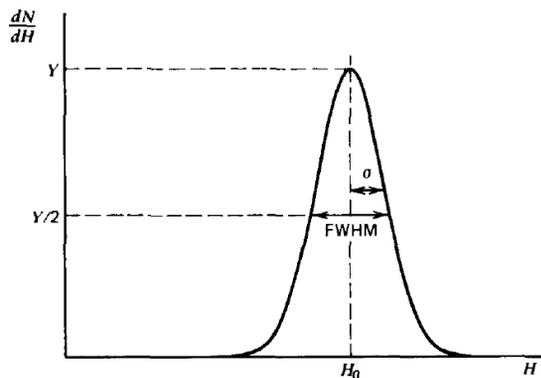


Figure 1: Definição da resolução de um detector.

Fonte: (KNOLL, G. F. Radiation Detection and Measurement. 4a edição, pag. 116)

A resolução a ser considerada no projeto do MCA deve ser levada em conta para o cálculo do número mínimo de canais necessários para que uma distribuição seja relativamente fiel ao sinal original observado.

Radiações

O projeto do MCA, como citado anteriormente, destina-se a montar o espectro de energia de uma determinada emissão radioativa. Mais especificamente para este projeto e sua aplicação imediata onde detectores a gás são utilizados, emissões radioativas na faixa dos raios-X.

Podemos separar as radiações em 4 categorias:

- Elétrons Acelerados
- Partículas Pesadas Carregadas
- Radiação Eletromagnética
- Radiação por Nêutrons

Na categoria Radiação Eletromagnética estão os raios-X, observados e estudados pioneiramente pelo físico alemão Wilhelm Röntgen em 1895. Caracteriza-se por fótons com origem na eletrosfera do átomo e com energias características de até uma centena de keV.

Posicionado no espectro eletromagnético entre os raios UV (ultravioleta) e os raios Gama (γ), os raios-X se diferem deste último exatamente por sua origem: raios Gama são originados de transições do núcleo atômico.

O experimento de caracterização dos detectores a gás utiliza uma fonte radioativa Fe-55, isótopo que é uma fonte de raios-X bem conhecida, e largamente utilizado para propósitos experimentais científicos. A fonte radioativa Fe-55 apresenta as seguintes características:

Meia-vida ($T_{1/2}$): 2,7 anos

Tipo de Radiação:

- Raios X/Gama – 6keV ($\sim 25\%$) e 7keV ($\sim 3\%$)
- Elétrons Auger – 5,2 keV

Para aplicações no Laboratório de Sistema de Medidas (LSD), parte integrante do Centro Brasileiro de Pesquisas Físicas (CBPF), a fonte radioativa é utilizada em experimentos como a aquisição de imagens de pequenos objetos, experimentos que utilizam o mesmo módulo de aquisição de dados, porém se valendo do dispositivo TDC (*Time-to-Digital Converter*).

Módulo SPRO e suas características

Desenvolvido no LSD, o módulo de aquisição de dados SPRO, sigla em inglês para *Standalone Prototype*, que é utilizado como *hardware* base para o projeto foi pensado de maneira a atender diferentes aplicações em física experimental. O SPRO contém conversores analógico-digital (ADC), conversor tempo-digital (TDC), amplificadores diferenciais, além de um circuito de interface USB para comunicação com computadores pessoais, como demonstrado na **Figura 2**.

Além dos componentes citados, o SPRO possui um dispositivo FPGA, modelo Cyclone II EP2C20Q240C8, do fabricante ALTERA (atual Intel), no qual é programado o circuito de processamento digital do MCA.

O módulo funciona com tensão de alimentação $+6 V_{DC}$ e consumo máximo de 1,5 A. Contém um conector (PROG)

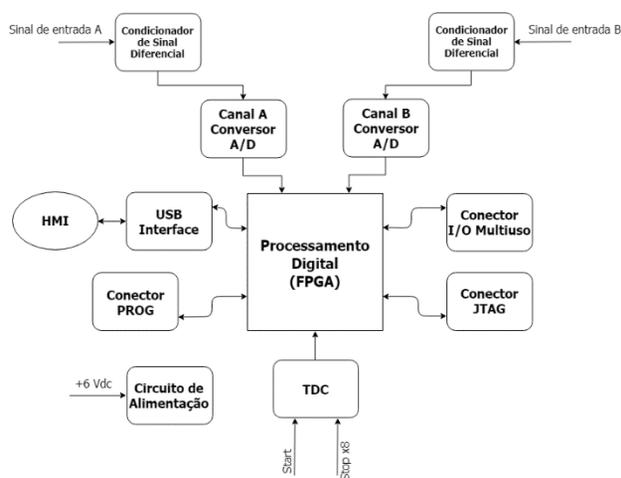


Figure 2: Diagrama em blocos do SPRO **Fonte:** (LSD, CBPF)

para configuração não volátil do FPGA, um segundo para configuração volátil e verificação interna dos sinais no FPGA (Joint Test Access Group – JTAG), além de um terceiro conector com 10 entradas/saídas (I/O) digitais (CON3).

FPGA

Com capacidade de 18.752 elementos lógicos, o Cyclone II EP2C20Q240C8 é um dos dispositivos de uma família de baixo custo de FPGAs, lançada em 2004, que sucede a família Cyclone de 2002. Suas principais características são listadas na Tabela 1.

Tabela 1. Características da FPGA Cyclone II EP2C20Q240C8

Característica	Qtd
Elementos lógicos	18.752
Blocos de memória RAM (M4K)	52
Bits RAM	239.616
PLLs (<i>Phase-Locked Loop</i>)	4
Nº máximo de pinos I/O	315

O chip conta com 240 pinos no total, em um encapsulamento tipo PQFP (*Plastic Quad Flat Pack*), com espessura de 4,1 mm.

Além de um clock global, o FPGA tem 4 unidades PLL (*Phase-locked Loop*), que sincronizam sinais internos de clock com sinais externos.

Como recurso fundamental para a implementação do MCA, o FPGA contém blocos de memória RAM (*Random Access Memory*) do tipo M4K. Cada bloco é constituído de 4096 bits, acrescidos de bits de paridade, totalizando 4608 bits. Neste tipo de memória são armazenados a contagem de eventos identificados pelo MCA, que posteriormente serão acessados pela interface USB.

Conversor Analógico Digital (ADC)

O módulo SPRO disponibiliza um conversor A/D com dois canais independentes (dual).

Com taxa de amostragem de 125 MSPS (*Mega Samples per Second*), o conversor do fabricante Analog Devices, modelo AD9627-125, fornece um sinal diferencial de clock (DCO), de 125 MHz, que sincroniza os dados convertidos na saída. Como premissa de projeto, a busca de pico do sinal convertido será, naturalmente, feita de forma síncrona com este clock.

O dispositivo que irá ler o conversor AD9627-125 deve utilizar a subida do clock DCO para capturar o sinal de entrada no canal A convertido, enquanto a borda de descida deve ser utilizada para o canal B, como pode ser observado na Figura 3.

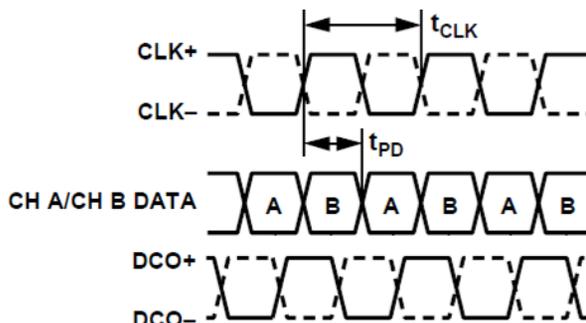


Figure 3: Diagrama de onda ADC9627-125 **Fonte:** (<http://www.analog.com>)

O ADC é configurado para operação em modo LVDS (*Low Voltage Differential Signaling*), que se mostra mais eficaz em termos de transmissão de sinais em alta velocidade, pois é menos suscetível a interferências eletromagnéticas.

Para o desenvolvimento deste projeto, apenas o canal A do ADC está sendo considerado como entrada para o MCA.

Interface de Comunicação USB

A Interface Homem-Máquina com o SPRO é feito por uma Universal Serial Bus (USB). Comum nos dias de hoje em todos os computadores de uso pessoal / laboratorial / corporativo, é o método mais eficaz por exigir pouco do usuário para que seja configurado propriamente. O SPRO conta com um circuito integrado (IC - sigla em inglês) da empresa FTDI Chip (Future Technology Devices International), do modelo FT245BM, que implementa o protocolo de comunicação entre os dispositivos conectados, como por exemplo, SPRO e computador com o programa do usuário.

A arquitetura do IC é baseada em memórias FIFO¹, com uma velocidade de transferência de até 1 Mbyte/s.

A programação da interface no SPRO foi realizada previamente onde foi desenvolvida de forma a facilitar a adição de novas funcionalidades / experimentos, evitando que essa

1

FIFO – FIFO (First-In First-Out) é um tipo de memória sequencial onde o primeiro dado escrito será sempre o primeiro dado lido.

parte em si necessite ser totalmente programada a cada aplicação desenvolvida no módulo.

Desenvolvimento do MCA

O desenvolvimento do projeto do MCA para o módulo de aquisição de dados SPRO utiliza as seguintes premissas:

- Detecção de Pico do Sinal utilizando um canal do ADC, sincronizado com o clock do mesmo (DCO).
- Utilização de uma memória single-port RAM, configurada através de uma *Megafunction* e criada a partir

do *plug-in MegaWizard*, presente no Altera Quartus II Web Edition².

- Integração do bloco à Interface USB do módulo SPRO.

No diagrama em blocos do MCA, ver Figura 4, pode-se observar os blocos responsáveis pela detecção de pico (“*peak_detector*”) assim como, a parte responsável pelo gerenciamento da memória e contagem cada evento para que esse seja acumulado ao ser detectado (“*histogram*”).

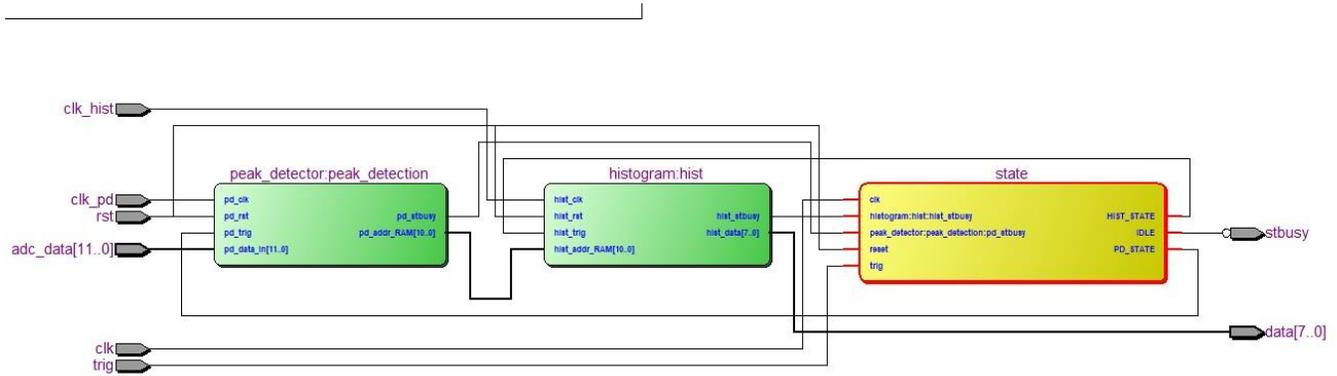


Figure 4: Diagrama em blocos do componente “MCA_comp”

Fonte: (LSD, CBPF)

Esses são partes de um bloco maior chamado “*MCA_comp*”, ao qual faz parte uma máquina de estados (“*state*”) projetada para gerenciar os sinais internos de trigger para cada um dos blocos supracitados. Esse é o componente mapeado dentro da entidade principal: “*angra*”.

O bloco “*MCA_comp*” conta com dois domínios distintos de clock entre seus blocos internos. Essa separação torna-se necessária à medida que tanto a máquina de estados de triggers internos, quanto a interface USB e o bloco “*histogram*”, utilizam um clock de frequência muito menor do que o utilizado pelo ADC, e consequentemente pelo bloco “*peak_detector*”.

Além dos sinais de clock, o bloco tem como entrada os sinais *rst* e *trig*, bem como o barramento de dados vindo do conversor A/D, com 12 bits, chamado de *ADC_DATA*.

Abaixo uma breve descrição de cada um dos sinais indicados na Figura 4:

- **a rst:** utilizado pelo módulo de aquisição de dados, SPRO, com a finalidade de restaurar condições iniciais do circuito;
- **b trig:** entrada do pulso que dispara a análise do sinal analógico sendo medido;
- **c adc_data[11..0]:** barramento de dados de 12 bits proveniente do conversor A/D. É a representação digital da amplitude do sinal de entrada;
- **d clk_pd:** sinal de clock que sincroniza a busca e

detecção do pico de energia do sinal. É o mesmo clock utilizado pelo conversor A/D;

- **e clk_hist:** sinal de clock específico para o bloco de histograma. É o mesmo clock utilizado pela interface USB do módulo SPRO;
- **f clk:** sinal de clock que sincroniza a máquina de estado responsável pelo gerenciamento dos triggers internos;
- **g stbusy:** sinal que indica que um evento (pulso) está sendo analisado pelo bloco do MCA. Enquanto esse sinal estiver em nível lógico “1”, o MCA está ocupado e não identificará nenhum outro sinal de trigger, ou poderá ser acessado pela interface USB;
- **h data[7..0]:** barramento de dados de 8 bits que é enviado à interface USB. É o resultado final, onde o conteúdo de cada canal indica a contagem dos eventos detectados.

Máquina de Estado

Para a programação do componente “*MCA_comp*” foram desenvolvidas máquinas de estado finitas (*FSM: Finite State Machine*), que auxiliaram principalmente na sincronização entre os diversos componentes projetados.

A máquina de estado do componente principal, ver Figura 5, que gerencia os triggers internos, contém 6 estados:

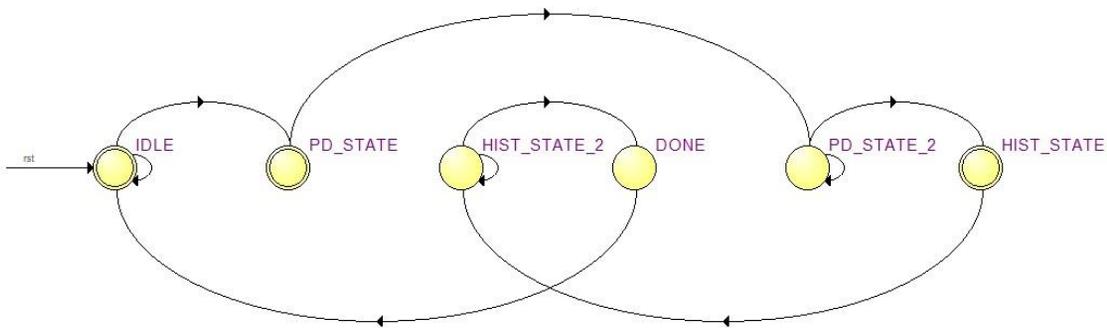


Figure 5: Máquina de Estados do bloco MCA_comp

Fonte:(LSD, CBPF)

- 1) IDLE
- 2) PD_STATE
- 3) PD_STATE_2
- 4) HIST_STATE
- 5) HIST_STATE_2
- 6) DONE

A máquina sempre inicia no estado IDLE e aguarda o sinal de trigger externo para início de funcionamento. Recebendo o sinal de trigger, e não havendo sinal rst (RESET) ativo, procede ao estado PD_STATE, onde ativa um sinal que dará

início ao bloco “*peak_detector*”, e durará exatamente um período de clock. Ao passar para o estado PD_STATE_2, o sinal que foi ativado retorna para nível lógico baixo. Desta forma, cria-se um pulso de trigger com duração de apenas um período de clock.

A máquina é síncrona com o clock de 20 MHz fornecido disponível no módulo SPRO.

Após o fim do bloco “*peak_detector*”, a máquina passa ao estado HIST_STATE que, utilizando o mesmo modo operacional citado acima, inicia o bloco “*histogram*”.

Cada bloco, “*peak_detector*” e “*histogram*”, possui sua própria máquina de estado, como descrita a seguir.

O bloco referente a detecção de pico do sinal analisado contém 4 estados – ver Figura 6:

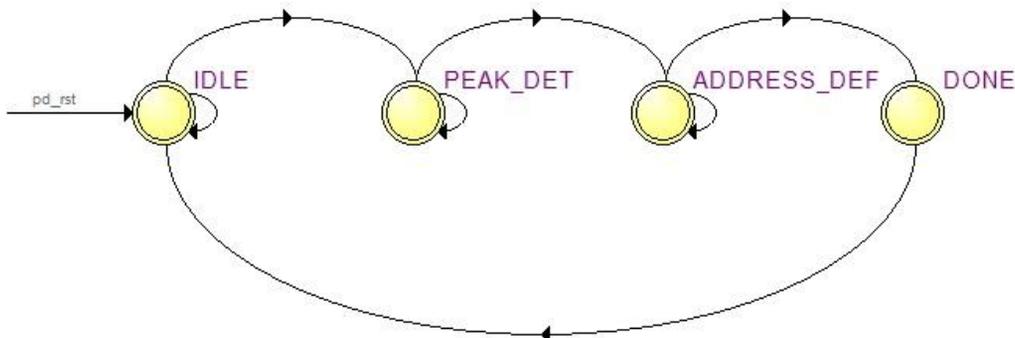


Figure 6: Máquina de Estados do bloco “peak_detector”

Fonte:(LSD, CBPF)

- 1) IDLE
- 2) PEAK_DET
- 3) ADDRESS_DEF
- 4) DONE

A máquina sempre inicia no estado IDLE, que também é o estado de destino do bloco quando esse identifica um nível alto na entrada rst (RESET). Para avançar ao estado seguinte, PEAK_DET, além de não haver um sinal de reset, o sinal interno de trigger vindo da máquina de estados que o gerencia (citado acima), deve também estar em nível alto.

Ao passar para o estado de PEAK_DET, o bloco passa a monitorar e comparar o dado proveniente do conversor A/D, a cada ciclo do sinal de clock DCO. Esta etapa dura um tempo pré-determinado, que leva em consideração a duração média característica do sinal sendo detectado. Pode ser facilmente alterada no código fonte. A comparação feita a cada sinal recebido é simplesmente observar se o valor atual recebido do conversor A/D no instante “N” é maior que o valor recebido anteriormente, guardado em memória temporária interna. Se este for o caso, o valor do instante “N” substitui o maior valor armazenado anteriormente. Caso contrário, sendo o valor atual em “N” menor que o já previamente armazenado, este atual é ignorado.

Expirando-se a janela de tempo de busca do valor máximo, a máquina passa ao próximo estado, ADDRESS_DEF. Neste estado, simplesmente confirma-se o valor máximo colocando-o na porta de saída do bloco. A partir deste momento este valor corresponde ao endereço que será utilizado pela memória RAM para armazenar a contagem de cada nível de energia.

Após um ciclo de clock, a máquina passa ao estado DONE e, após mais um período, retorna ao estado IDLE.

O bloco responsável pelo gerenciamento da memória RAM, e adição de um novo evento registrado, contém os seguintes estados – ver Figura 7:

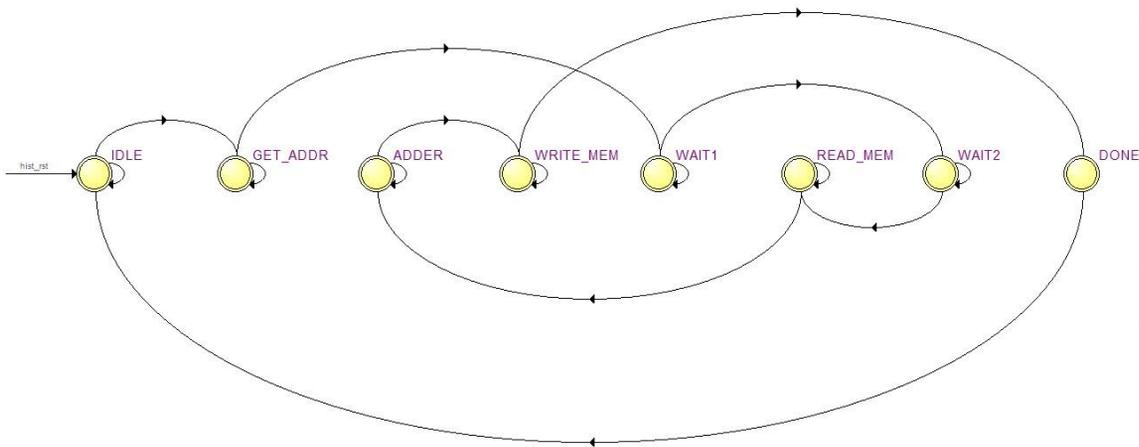


Figure 7: Diagrama em blocos do componente "histogram"

Fonte: (LSD, CBPF)

- 1) IDLE
- 2) GET_ADDR
- 3) WAIT1
- 4) WAIT2
- 5) READ_MEM
- 6) ADDER
- 7) WRITE_MEM
- 8) DONE

Esta máquina sempre inicia, assim como a anterior, no estado IDLE. Assim como anteriormente, é para este estado que a máquina retorna quando identifica um nível alto na entrada rst (RESET). Para que a máquina passe para o próximo estado, quando a detecção de pico termina, um sinal interno do bloco de MCA de trigger é disparado.

Após o sinal de trigger interno ser detectado, a máquina passa para o estado GET_ADDR, onde é definido internamente o endereço que será acessado na memória RAM.

A seguir, ocorrem dois estados idênticos, que tem como função realizar o sincronismo necessário para acesso ao dado de memória do endereço escolhido: WAIT1 e WAIT2.

Os estados WAIT1 e WAIT2 se fazem necessários pois no funcionamento do bloco de memória RAM há uma latência entre a definição do endereço a ser lido, que é feito sempre na borda de subida do clock. Após o bloco de memória identificar o endereço, ele precisa de dois ciclos para que o conteúdo deste endereço fique disponível na saída do mesmo.

Ao final dos dois estados de sincronização, o conteúdo do endereço lido é escrito em uma variável temporária no estado READ_MEM.

No ciclo seguinte, passando ao estado ADDER, o valor armazenado será incrementado (somado de 1), significando que mais um evento naquele canal específico foi detectado. Assim, conseguimos um efeito de contador para cada janela de energia, como deve funcionar um MCA.

Para encerrar a operação, após a soma, a máquina passa ao estado responsável por regravar esse valor no mesmo endereço de memória, substituindo o conteúdo primeiramente lido alguns estados atrás. Esse armazenamento acontece no estado WRITE_MEM.

Por fim a máquina finalmente passa ao estado DONE, que indica o final do processo, retornando um ciclo depois para o estado IDLE.

Não somente esse processo, mas também o componente principal passará para os estados de DONE e logo após IDLE, que neste momento indicarão que o evento foi detectado e contabilizado, podendo agora aguardar um próximo trigger externo ou ter seus dados colhidos pela Interface Homem-Máquina.

Após a compilação do blobo “MCA_comp”, como descrito nos parágrafos anteriores, obteve-se o seguinte uso dos recursos no FPGA, como visto na Tabela 2:

Tabela 2. Recursos utilizados do FPGA

EP2C20Q240C8			
Característica	Valor Usado	Total Disponível	% Usada
Memória	22.016 bits	239.616 bits	9%
Elementos lógicos	1.558	18.752	8%
PLLs	1	4	25%
Pinos	111	142	78%

Os números na tabela acima mostram que o dispositivo EP2C20Q240C8 ainda comporta grande quantidade de recursos para inclusão de novos componentes e funcionalidades no mesmo módulo.

Resultados

Como parte dos testes que foram estabelecidos, simulações funcionais de cada bloco, assim como do bloco “MCA_comp”, o bloco principal, foram realizados para assegurar que o código escrito desempenhe exatamente o que foi previsto.

Simulação funcional

Utilizou-se o software ModelSim ALTERA STARTER

EDITION 10.1d, desenvolvido em parceria pelas empresas Altera e Mentor, especificamente para simular projetos com FPGAs fabricadas pela Altera.

Os resultados, primeiramente, evidenciaram que alterações no código deveriam ser feitas para que alguns cenários fossem cobertos, como por exemplo, a entrada de um sinal de trigger em meio ao funcionamento do componente, sem que esse causasse o reinício do processo, ou a alteração indevida de algum dado colhido.

Simulando os sinais de clock com os mesmos valores que são encontrados no módulo - 125 MHz (8 ns) e 20 Mhz (50 ns) - e uma janela de detecção arbitrária somente para fins de simulação de 5 ciclos do clock de 125 MHz, foram observados os seguintes resultados:

- Funcionamento sequencial de blocos e estados: observa-se que o componente respeita a premissa de funcionamento, onde os estados de cada máquina ocorrem na ordem desejada (ver em detalhes nas Figura 8 e Figura 9).
- Detecção de pico: durante o intervalo destinado a detecção de pico, os dados vindos do ADC são avaliados e somente o maior valor encontrado é registrado; no caso simulado o valor 00000000100.
- Janela de detecção: a janela de busca do pico escolhida foi gerada com a duração de 5 ciclos do sinal de clock do ADC (ver Figura 8).
- Gerenciamento de memória e armazenamento de evento: com o valor do pico energético encontrado, observa-se a utilização deste valor como endereço de memória, e ainda que o seu conteúdo é incrementado em um ao final dessa etapa (ver Figura 9).
- Tempo de processamento simulado: em simulação, com a janela de detecção de 5 ciclos do sinal de clock do ADC, o tempo de processamento completo foi de 642 ns por evento, contando a partir da transição positiva do trigger (Figura 10).

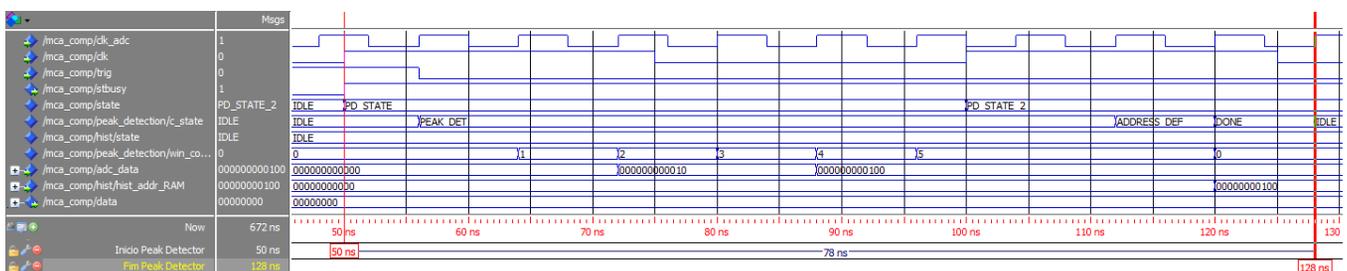


Figure 8: Simulação do código programado - detecção de pico
Fonte: (LSD, CBPF)

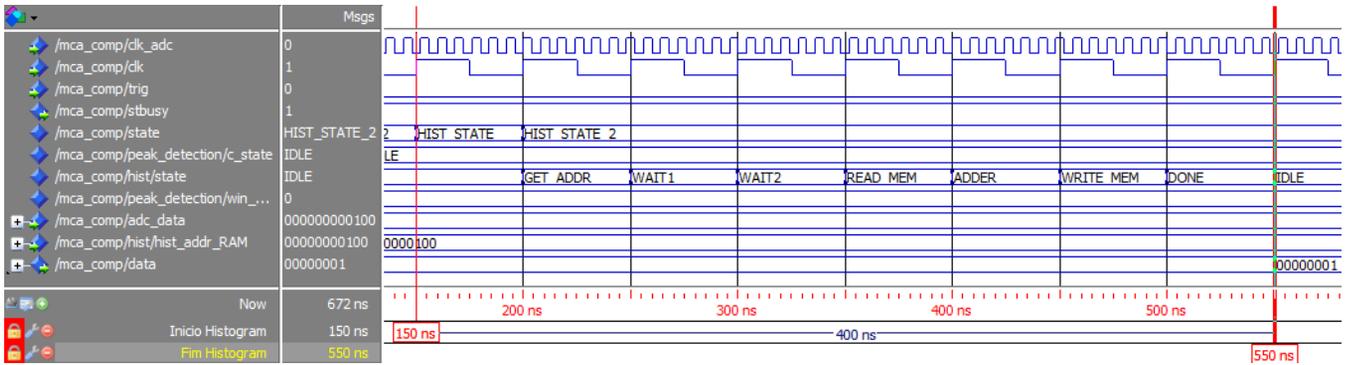


Figure 9: Simulação do código programado - gerenciamento de memória **Fonte:** (LSD, CBPF)

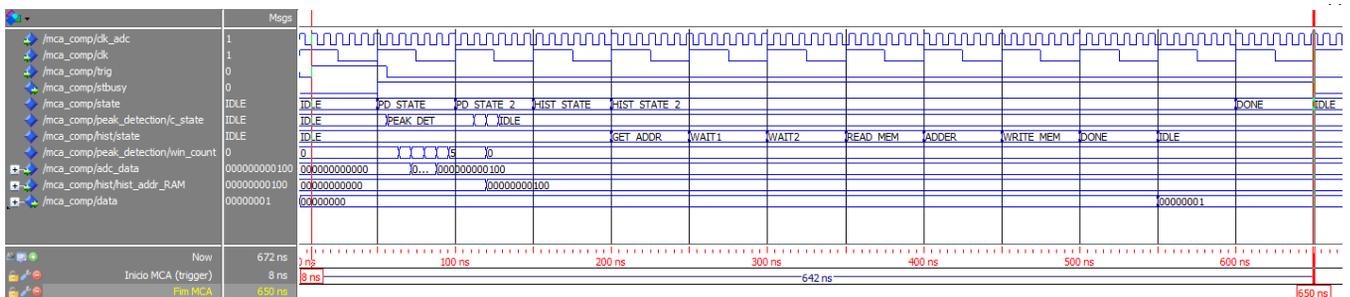


Figure 10: Simulação do código programado do MCA **Fonte:** (LSD, CBPF)

Verificação com SignalTap II

Após a simulação funcional, utilizou-se o recurso *SignalTap II Logic Analyzer*³ para se confirmar o comportamento do código proposto em funcionamento no FPGA.

Para essa verificação foi utilizado o pino 50 do FPGA, disponível para conexão externa através do pino 19 do conector 3 (CON3) do módulo SPRO (ver Figura 11), como entrada para o sinal de trigger (trig); o pino 20 é a referência para este sinal. O CON3 permite ao usuário do módulo conectar sinais de entrada ou saída diretamente aos pinos de I/O do FPGA, de acordo com a necessidade do projeto. O mesmo sinal de trigger é responsável por dar início ao analisador multicanal, assim como na aquisição dos dados pelo software.

O sinal de clock do conversor A/D (clk_pd) foi usado

como referência por ser este o sinal de maior frequência no circuito, e no próprio MCA. Foram monitorados os sinais das máquinas de estado de todos os blocos, assim como os sinais de entrada e saída, trigger e stbusy (que indica que o bloco está em uso para agentes externos). Uma janela de detecção de 5 ciclos do clock (clk_pd) foi arbitrada apenas para fins de análise de funcionamento. As condições de trigger, para efeitos de sincronização para aquisição dos dados, são as seguintes:

- 1) Detecção da borda de subida do sinal trig
- 2) stbusy estar em nível lógico "0"

Garantindo, assim, que a aquisição de dados fosse feita no início do processo de fato, como sendo um evento real. Sinais de trigger que por ventura venham a acontecer no meio de um processamento são ignorados pela lógica.

Como entrada analógica foi configurada, em um gerador de sinais, uma onda quadrada de 1 Hz, 0,5 Vpp, sendo seu

³

Recurso similar a um Analisador Lógico que permite a inspeção dos sinais internos da FPGA em tempo real.

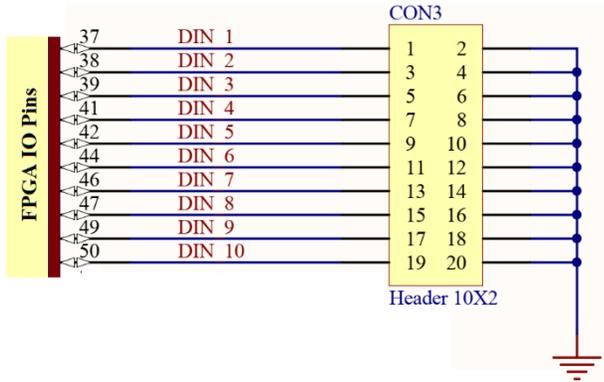


Figure 11: Conector de entradas e saídas digitais do módulo SPRO (CON3)

Fonte: (LSD, CBPF)

ponto mínimo em 0 V, com 98% de *duty cycle*⁴.

Na Figura 12 observa-se o comportamento geral do MCA e seus sinais mais importantes – o detalhamento dos estados será visto em detalhes a frente. O sinal *stbusy* encontra-se em nível lógico “1” por toda a detecção do evento; como projetado, esse impede que o MCA reinicie se outro evento ocorra antes do fim do processamento, mesmo o trigger estando em nível lógico “1” também.

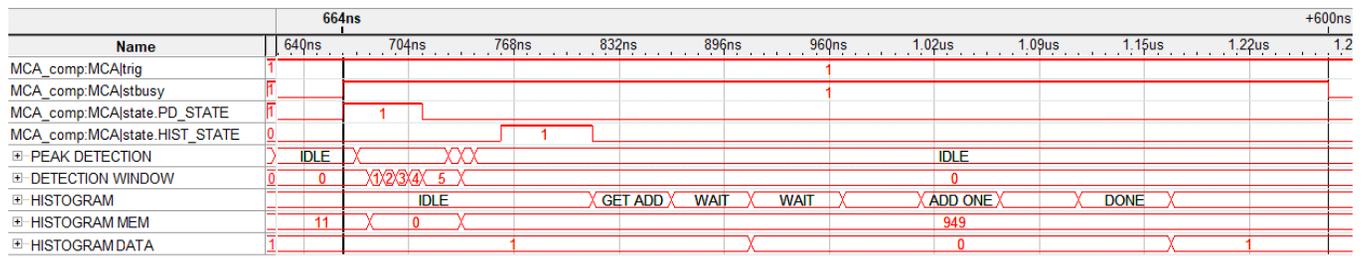


Figure 12: Comportamento geral verificado do MCA

Fonte: (LSD, CBPF)

Na Figura 13 tem-se o detalhamento do comportamento do bloco de detecção de pico. É possível observar que o valor 949 é capturado como sendo o pico energético do evento sendo processado, e assim é designado como o endereço

de memória no bloco gerenciador da memória. O endereço 949 é o valor discretizado correspondente a +0,5 V (\pm erro intrínseco do sistema), como configurado no gerador de sinais.

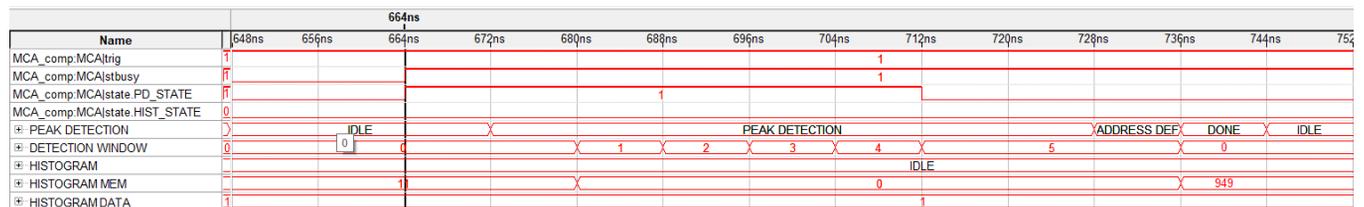


Figure 13: Comportamento verificado do bloco de detecção de pico

Fonte: (LSD, CBPF)

O comportamento do bloco de gerenciamento da memória é observado em detalhes na Figura 14. É possível observar que o conteúdo da memória após a definição do novo endereço é inicialmente 0. No estado de adição (ADD ONE

na figura) o valor encontrado é somado em um e reescrito no mesmo endereço. Ao final o conteúdo da memória (HISTOGRAM DATA) passa a ser um.

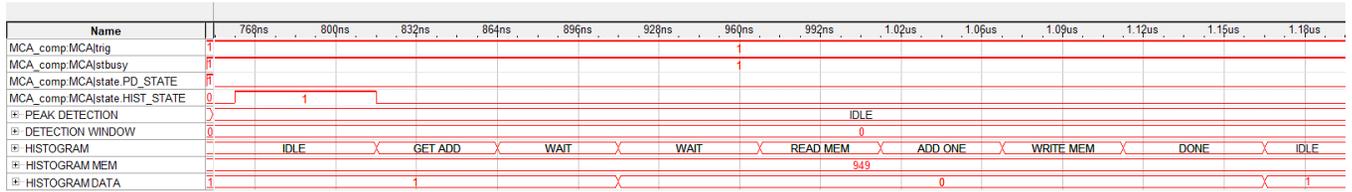


Figure 14: Comportamento verificado do bloco de gerenciamento da memória

Fonte: (LSD, CBPF)

Um segundo evento foi simulado com o mesmo nível de sinal de entrada, que terminou por encontrar o mesmo endereço de memória como pico. Neste confirma-se que

o evento subsequente ao primeiro é somado, caracterizando que este ocorreu uma segunda vez – ver Figura 15.

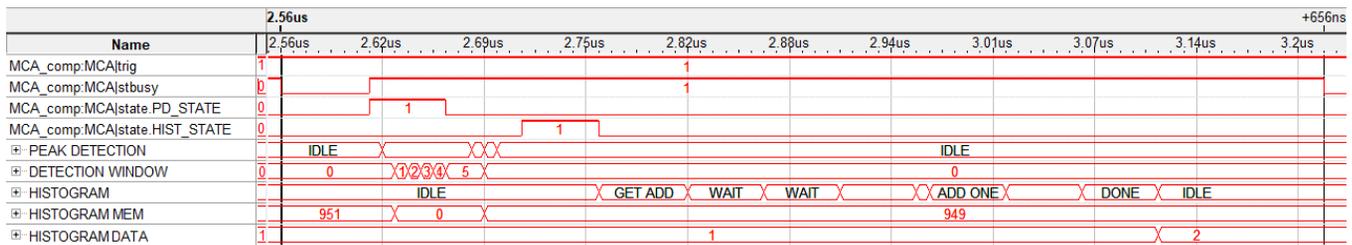


Figure 15: Comportamento do MCA em um Segundo evento de mesma energia

Fonte: (LSD, CBPF)

1. CONCLUSÃO

Foi projetado um MCA baseado em um conversor A/D de 125 MHz e um dispositivo FPGA para processamento digital da informação, ambos disponíveis no módulo SPRO. Os testes realizados, tanto por simulação quanto por inspeção interna dos sinais no FPGA, demonstram que o circuito desenvolvido funciona de acordo com o que foi predeterminado, estando pronto para testes com os experimentos reais envolvendo os detectores de raios-X.

Bibliography

[1] KNOLL, G. F. **Radiation Detection and Measurement**. 4th edição. Michigan, EUA. John Wiley & Sons Inc, 2010.

[2] Brown, S.; Vranesic, Z. **Fundamentals of Digital Logic with VHDL Design**. 3a edição. Toronto, Canadá. McGraw-Hill Companies, Inc, 2008.

[3] Lima Junior, H. P.; Barbosa, A. F.; Guedes, G. P.; Seixas, J. M. A **Fast Multichannel-Analyzer for Radiation Detection Applications** IEEE Transactions on Instrumentation and Measurements, v. 53-2, p. 378-383, 2004.

[4] Altera Corporation. **Cyclone II Device Handbook, Volume 1**; February 2008. Disponível em: <http://www.altera.com>. Acesso em: 23 mar. 2018.

[5] Altera Corporation. **RAM Megafunction User Guide**; March 2007. Disponível em: <http://www.altera.com>. Acesso em: 15 out 2017.

[6] Analog Devices, Inc. **12-Bit, 80 MSPS/105 MSPS/125 MSPS/150 MSPS, 1.8 V Dual Analog-to-Digital Converter**; 2010. Disponível em: <http://www.analog.com/>. Acesso em: 12 abr. 2018.

[7] The University of Manchester. **Nuclide Safety Data Sheet Iron-55; 2012. Disponível em**

<http://www.hep.man.ac.uk/safety/RadiationDataSheets/>.
Acesso em 18 abr. 2018.