MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

DEMODULADOR PARA SINAIS PTT-A3 DO ARGOS-3 EM FPGA.

Thiago Maia Souto

Relatório Final de Iniciação Científica do programa PIBIC, orientada pelo Dr. José Marcelo Lima Duarte.

URL do documento original:
<http://urlib.net/>

INPE
São José dos Campos
2022

PUBLICADO POR:

Instituto Nacional de Pesquisas Espaciais - INPE
Gabinete do Diretor (GB)
Serviço de Informação e Documentação (SID)
Caixa Postal 515 - CEP 12.245-970
São José dos Campos - SP - Brasil
Tel.:(012) 3945-6923/6921
Fax: (012) 3945-6919
E-mail: pubtc@sid.inpe.br

COMISSÃO DO CONSELHO DE EDITORAÇÃO E PRESERVAÇÃO DA PRODUÇÃO INTELECTUAL DO INPE (DE/DIR-544):
Presidente:
Marciana Leite Ribeiro - Serviço de Informação e Documentação (SID)
Membros:
Dr. Gerald Jean Francis Banon - Coordenação Observação da Terra (OBT)
Dr. Amauri Silva Montes - Coordenação Engenharia e Tecnologia Espaciais (ETE)
Dr. André de Castro Milone - Coordenação Ciências Espaciais e Atmosféricas (CEA)
Dr. Joaquim José Barroso de Castro - Centro de Tecnologias Espaciais (CTE)
Dr. Manoel Alonso Gan - Centro de Previsão de Tempo e Estudos Climáticos (CPT)
Drª Maria do Carmo de Andrade Nono - Conselho de Pós-Graduação
Dr. Plínio Carlos Alvalá - Centro de Ciência do Sistema Terrestre (CST)
BIBLIOTECA DIGITAL:
Dr. Gerald Jean Francis Banon - Coordenação de Observação da Terra (OBT)
Clayton Martins Pereira - Serviço de Informação e Documentação (SID)
REVISÃO E NORMALIZAÇÃO DOCUMENTÁRIA:
Simone Angélica Del Ducca Barbedo - Serviço de Informação e Documentação (SID)
Yolanda Ribeiro da Silva Souza - Serviço de Informação e Documentação (SID)
EDITORAÇÃO ELETRÔNICA:
Marcelo de Castro Pazos - Serviço de Informação e Documentação (SID)
André Luis Dias Fernandes - Serviço de Informação e Documentação (SID)

MINISTÉRIO DA CIÊNCIA, TECNOLOGIA, INOVAÇÕES E COMUNICAÇÕES
INSTITUTO NACIONAL DE PESQUISAS ESPACIAIS

DEMODULADOR PARA SINAIS PTT-A3 DO ARGOS-3 EM FPGA.

Thiago Maia Souto

Relatório Final de Iniciação Científica do programa PIBIC, orientada pelo Dr. José Marcelo Lima Duarte.

URL do documento original:
<http://urlib.net/>

INPE
São José dos Campos
2022

Dados Internacionais de Catalogação na Publicação (CIP)

Sobrenome, Nomes.

Cutter Demodulador para sinais PTT-A3 do ARGOS-3 em FPGA. / Thiago Maia Souto; Nome Completo do Autor2. – São José dos Campos : INPE, 2022.

ix + 12 p. ; ()

Relatório final (Iniciação Científica do programa PIBIC) – Instituto Nacional de Pesquisas Espaciais, São José dos Campos, AAAA.

Orientador : José Marcelo Lima Duarte.

1. Demodulador. 2. FPGA 3. SystemVerilog. 4. RTL. 5. Automação. 6. TCL. I. Título.

CDU 000.000

Esta obra foi licenciada sob uma Licença Creative Commons Atribuição-NãoComercial 3.0 Não Adaptada.

This work is licensed under a Creative Commons Attribution-NonCommercial 3.0 Unported License.

Informar aqui sobre marca registrada (a modificação desta linha deve ser feita no arquivo publicacao.tex).

ATENÇÃO! A FOLHA DE APROVAÇÃO SERÁ INCLUIDA POSTERIORMENTE.

Iniciação Científica do programa PIBIC

RESUMO

Inicialmente o objetivo do trabalho era o desenvolvimento de um demodulador para sinais PTT-A3 do ARGOS-3 em FPGA para ser incorporado ao sistema Environmental Data Collector (EDC). No entanto, devido a identificação da necessidade de otimização de performance do EDC, o objetivo inicial passou a ser implementar melhorias na versão do demodulador para sinais PTT-A2 já existente no EDC. O EDC é a carga útil que habilita um satélite a fazer parte do sistema GLOBAL OPEN COLLECTING DATA SYSTEM (GOLDS) de coleta de dados por satélite de baixa órbita terrestre. O trabalho consistiu no estudo da linguagem de descrição de hardware Verilog e estudo da documentação do projeto existente até então. A partir da análise do projeto já existente notou-se a utilização de uma quantidade considerável de blocos lógicos para síntese lógica do bloco de memória FIFO, isto porque a síntese lógica não estava utilizando os blocos de memória já disponíveis na FPGA, ocasionando um desperdício de recursos. Com isso, a primeira tarefa consistiu na implementação em Verilog de um novo bloco de memória FIFO capaz de ser sintetizado a partir dos blocos de memória da FPGA. Uma das dificuldades que surgiram durante a execução do projeto foi a atualização do software de desenvolvimento que exigiu a criação de um novo projeto na versão mais recente do software e em testes posteriores para validação das configurações adotadas. Já a segunda atividade desenvolvida consistiu na automação da criação e configuração do projeto para que não fosse necessário o desperdício de tempo no setup utilizando a ferramenta de desenvolvimento Libero.

Palavras-chave: Demodulador. RTL. SystemVerilog. TCL. FIFO. FPGA.

LISTA DE FIGURAS

Pág.

2.1 Visão geral do EDC representando apenas os principais sinais de interface do módulo M2S. Fonte: (INPE,) 3
2.2 Visão geral da arquitetura do EDC M2S. Fonte: (INPE,) 4
2.3 Visão geral da arquitetura do EDC Core. Fonte: (INPE,) 4
2.4 Árvore de dependências do módulo do EDC Core princiapl. Fonte: Autoria Própria. 5
2.5 Resultado do testbench da FIFO. Fonte: Autoria Própria. 6
2.6 Executando Script TCL para criação do projeto. Fonte: Autoria Própria. 7
2.7 Projeto do Libero criado a partir do script TCL. Fonte: Autoria Própria. 8

SUMÁRIO

Pág.

1 INTRODUÇÃO 1
1.1 Objetivos 1

2 DESENVOLVIMENTO 3
2.1 Revisão de literatura 3
2.1.1 Utilização de FPGAs em sistemas de comunicação 3
2.1.2 Arquitetura do EDC 3
2.2 Materiais e métodos 5
2.3 Resultados 6
2.4 Discussão dos resultados 8

3 CONCLUSÕES 10

REFERÊNCIAS BIBLIOGRÁFICAS 11

1 INTRODUÇÃO

O Sistema Brasileiro de Coleta de Dados (SBCDA), em operação desde 1993 consiste de um conjunto de Plataformas de Coletas de Dados (PCDs) espalhados pelo território brasileiro cujas informações coletadas são retransmitidas via satélite para uma plataforma receptora de dados (RAE, 2005). Além disso, o órgão responsável pelo desenvolvimento e operação do sistema é o Instituto Nacional de Pesquisas Espaciais (INPE).

Além disso, atualmente o SBCDA tem aproximadamente 450 PCDs ativas, quatro satélites em sua constelação e duas estações terrestres no Brasil. O Reduzido número de satélites e a idade avançada de alguns deles acaba desencorajando grandes usuários do sistema de permanecer usando-o. As limitações crescentes do SBCDA colocaram a continuidade do sistema em risco (DUARTE et al., 2021).

Para garantir a continuidade do SBCDA foi proposta a abertura da tecnologia do sistema para permitir a contribuição de outras institiuições para o seu desenvolvimento e esse novo sistema foi chamado Global Open CoLlecting Data System (GOLDS) (MATTIELO-FRANCISCO et al., 2018). Para isso foi proposta a expansão da constelação do SBCDA com a utilização de nanossatélites CubeSat cuja carga útil consiste no EDC tendo a função de receptor SBCDA.

A utilização de um receptor ao invés do transpoder analógico usado anteriormente permite cobertura global, uma vez que as mensagens podem ser armazenadas para posterior transmissão para as estações receptoras terrestre, o que não era possível na versão anterior. Além disso, o downlink agora pode ser feito através da multiplexação das mensagens recebidas no canal já existente de telemetria do satélite. Essa alteração permite a facilidade de implementação em nanossatélites removendo a necessidade de um transmissor adicional permitindo uma comunicação de downlink energeticamente mais eficiente (DUARTE et al., 2021).

1.1 Objetivos

Inicialmente o objetivo do trabalho era o desenvolvimento de um demodulador para sinais PTT-A3 do ARGOS-3 em FPGA para ser incorporado ao EDC. No entanto, devido a identificação da necessidade de otimização de performance e recursos, o objetivo passou a ser implementar melhorias na versão do demodulador para sinais PTT-A2 já existente no EDC.

As alterações a serem implementadas estavam relacionadas ao módulo da memória

FIFO, a qual possuía um comportamento assíncrono dificultando sua síntese lógica utilizando os blocos de memórias já contidos na FPGA. Tendo em vista que o sistema já estava codificado na linguagem de descrição de hardware Verilog, foi necessária uma revisão desta junto à metodologia de desenvolvimento RTL.

Por fim, os novos objetivos consistiram na implementação e verificação de uma memória FIFO em Verilog com comportamento síncrono, de forma a diminuir a quantidade de blocos lógicos necessários para a síntese lógica do EDC, diminuindo dessa forma a área de ocupação da FPGA.

Em relação a segunda atividade desenvolvida durante a vigência da bolsa, o objetivo consistiu na automação do fluxo de projeto a partir da criação de scripts TCL de forma a criar e configurar o projeto do EDC utilizando a ferramenta Libero SoC.

2 DESENVOLVIMENTO

2.1 Revisão de literatura

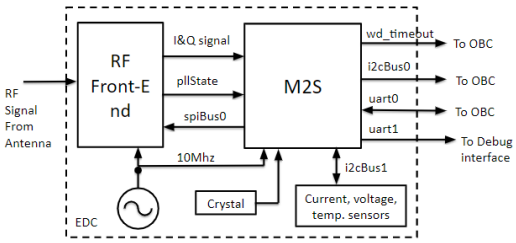
2.1.1 Utilização de FPGAs em sistemas de comunicação

Em virtude da capacidade das FPGAs de integrarem múltiplas funções ou até mesmo um sistema completo em um único chip sua adoção vêm se tornando algo comum entre projetistas de receptores digitais que utilizam essa tecnologia para implementação de seus sistemas (BREMEC et al., 2003). Dentro desse contexto, a escolha de uma FPGA para implementação da unidade de processamento do EDC também está relacionada com a capacidade de otimização do sistema para atender os requisitos específicos da aplicação.

2.1.2 Arquitetura do EDC

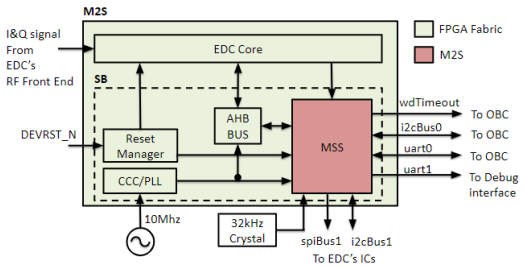
A unidade principal de processamento do EDC é a Microsemi SmartFusion 2 SoC FPGA (M2S), a qual contém uma FPGA baseada em memória flash (A qual reduz o tempo de configuração e oferece maior robustez diante da radiação espacial) e um Microcontroller Subsystem (MSS) contendo um microcontrolador ARM Cortex-M3. As funcionalidades dessa unidade consistem na coleta de informações sobre o estado do EDC, decodificação dos sinais das PCDs, adição de um cabeçalho às mensagens decodificadas indicando informações sobre tempo, frequência e potência do sinal e pela comunicação com o computador de bordo (INPE,).

Figura 2.1 - Visão geral do EDC representando apenas os principais sinais de interface do módulo M2S. Fonte: (INPE,)



A arquitetura do módulo M2S do EDC é composta pelo MSS e o EDC Core, um periférico customizado para o MSS implementado na FPGA. Toda a configuração do hardware do MSS foi realizada com o auxílio da aplicação proprietária da Microsemi (fabricante do SoC). A figura 2.2 representa a composição do bloco M2S.

Figura 2.2 - Visão geral da arquitetura do EDC M2S. Fonte: (INPE,)

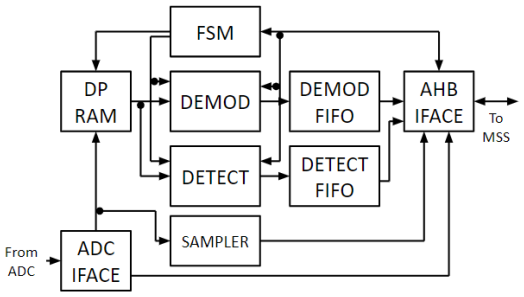


O foco deste trabalho é o bloco EDC Core localizado no topo da figura 2.2 cujas funções consistem em:

- Realizar a interface com o circuito receptor de Radiofrequência (RF) para recepção dos dados de entrada do EDC.
- Fornecer acesso de leitura para o sinal de entrada.
- Auxiliar no processo de detecção dos sinais das PCDs através do cálculo do espectro do sinal de entrada e aplicar um limiar de detecção nas amostras do espectro.
- Demodular os sinais das PCDs presentes no sinal de entrada.
- Computar o valor quadrático médio do sinal de entrada.

Uma visão geral da arquitetura do EDC Core pode ser observada na figura 2.3.

Figura 2.3 - Visão geral da arquitetura do EDC Core. Fonte: (INPE,)

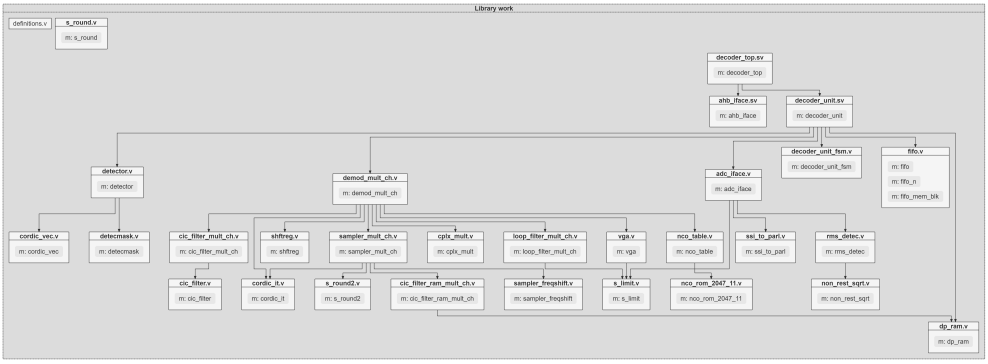


Como pode ser visto na figura 2.3, os blocos de memória FIFO constituem um componente de extrema importância para o sistema.

2.2 Materiais e métodos

O início do trabalho consistiu em analisar a documentação produzida até então sobre o EDC Core e avaliação dos códigos HDL em Verilog que descrevem os blocos utilizados. Tendo em vista a dificuldade de trabalhar com sistemas já em desenvolvimento, foi necessária uma análise profunda dos códigos que possibilitou a construção de uma árvore de dependências do módulo principal do EDC Core com a utilização da ferramenta Sigasi Studio.

Figura 2.4 - Árvore de dependências do módulo do EDC Core princiapl. Fonte: Autoria Própria.



Com o auxílio da figura 2.4 foi possível notar que o módulo FIFO era instanciado em apenas um único módulo, indicando que as alterações realizadas seriam feitas apenas neste módulo ao invés de em vários blocos do sistema. Com isso, foi feita uma análise no código da antiga FIFO e notou-se que ele utilizava uma memória RAM do tipo assíncrona, o que inviabilizava a síntese lógica do bloco nas memórias RAMs síncronas da FPGA, ocasionando uma maior ocupação de blocos lógicos.

A implementação desenvolvida para solucionar esse problema consistiu em reescrever o código da FIFO utilizando o bloco de memória RAM síncrono já instanciado pelo sistema que era sintetizado de maneira correta nos blocos de memória RAM da FPGA. Em seguida, foi desenvolvido um código de testbench para realizar a verificação do comportamento da FIFO projetada.

Já a segunda parte do trabalho consistiu em implementar um script TCL que realizasse a criação do projeto do EDC na ferramenta Libero de forma automatizada. Com esse intuito, foi necessária uma pesquisa mais aprofundada na documentação da ferramenta Libero, tendo em vista a dificuldade de utilização de sua interface

gráfica.

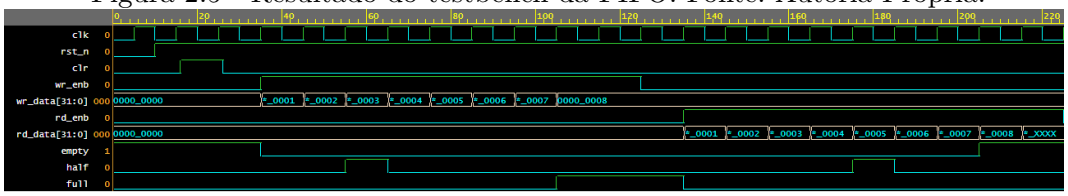
Após a pesquisa, descobriu-se que a ferramenta era capaz de exportar scripts TCL a partir dos comandos realizados na interface gráfica. No entanto, algumas configurações específicas não eram suportadas nativamente por essa função, o que necessitou mais uma pesquisa na documentação até descobrir a funcionalidade de exportar scripts TCL a partir de blocos já configurados, o que permitiu a criação de um script capaz de fazer toda a etapa de criação e configuração dos blocos necessários.

Uma vez criado o script foram necessários alguns ajustes manuais para garantir que arquivos de constraint fossem inseridos de forma adequada, além da inclusão do firmware durante a fase de inicialização da memória, permitindo que todo o processo fosse realizado a partir da execução do script TCL criado.

2.3 Resultados

O testebench realizado teve como objetivo verificar se os processos de escrita e leitura, assim como os sinais indicativos do estado da FIFO estavam funcionando de acordo com o esperado. Para isso, instanciou-se uma FIFO com uma capacidade de 8 amostras de tamanho de 32 bits cada por motivos de simplificação. Além disso, foram testados dois casos, um deles responsável por realizar a escrita de dados na FIFO até enche-la e outro responsável por realizar a leitura dos dados da FIFO até esvaziá-la, enquanto cada caso era testado, verificava-se se os sinais de estado (Vazia, Metade Cheia, Cheia) estavam respondendo de forma adequada.

Figura 2.5 - Resultado do testbench da FIFO. Fonte: Autoria Própria.



Uma vez tendo verificado o comportamento da FIFO, foi realizada a substituição do módulo anterior pelo novo módulo no código do EDC Core. Por fim, foi realizada a síntese lógica do sistema com a nova FIFO e comparou-se os relatórios de ocupação dos blocos da FPGA antes e depois da mudança. Na tabela 2.1 encontra-se o resultado obtido.

Em relação a segunda atividade implementada, foram criados dois scripts:

Tabela 2.1 - Comparação entre a porcentagem usada de recursos lógicos a partir da síntese lógica da antiga implementação e a nova.

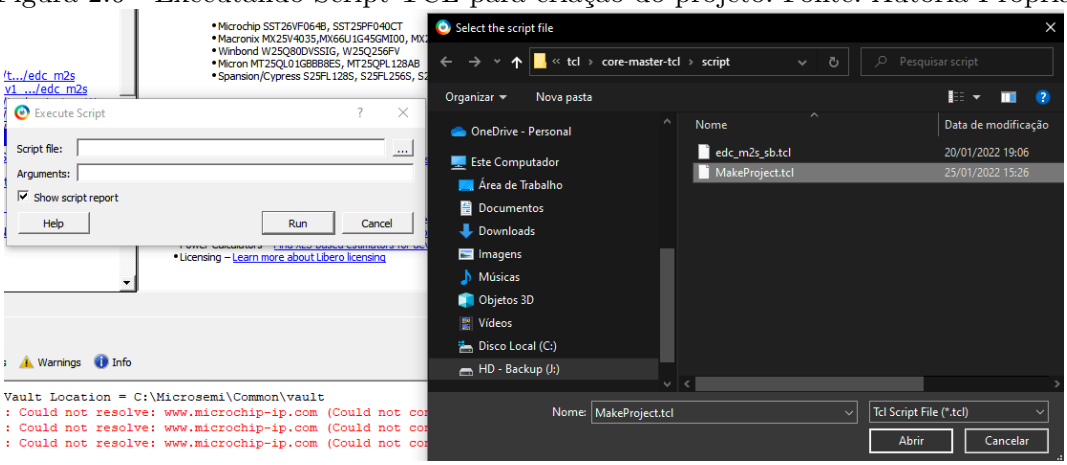
Percentual Usado	Antigo(%)	Novo(%)	Redução(%)
4LUT	50,02	49,29	0,73
DFF	48,74	48,57	0,16
MACC	88,24	82,35	5,89
RAM64x18	67,65	55,88	11,77
RAM1K18	48,39	64,52	-16,13

- **MakeProject.tcl**: Esse script é responsável pela criação do projeto Libero em um local adequado e pela configuração do projeto: definição do nome do projeto, escolha do modelo de FPGA utilizada e especificações elétricas da FPGA. Além disso esse script é responsável por importar todos os arquivos de design e de constraints de forma adequada e inicializar a memória RAM da FPGA com o firmaware.

- **edc_m2s_sb.tcl**: Esse script é responsável pela instanciação e configuração dos módulos fornecidos pela FPGA, tais como barramento de dados, GPIO, conexão SPI e I2C.

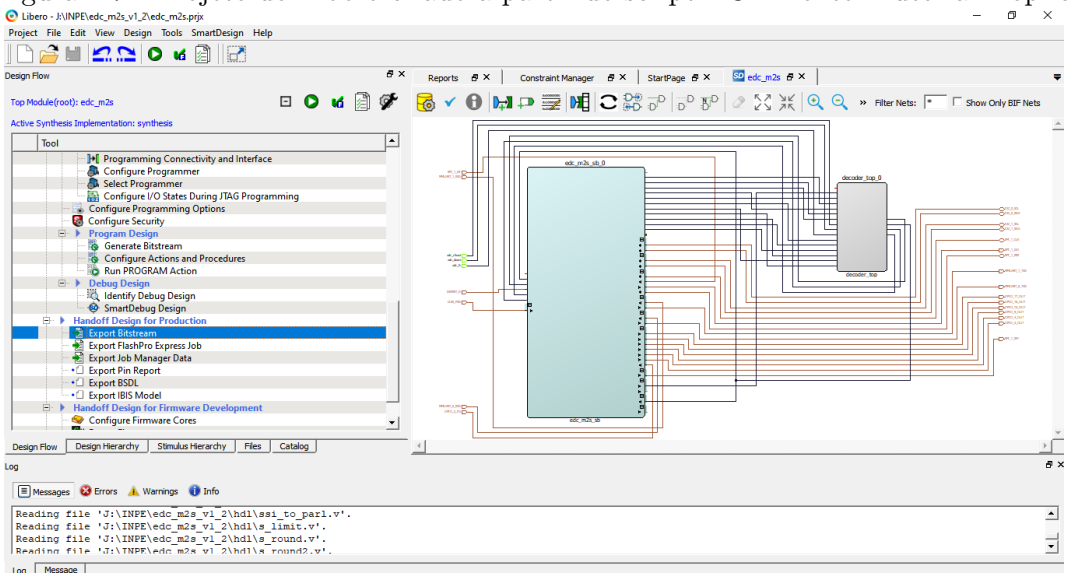
A partir dos scripts criados a rotina de criação e configuração de um projeto foi totalmente automatizada, de forma que basta executar a ferramenta Libero, clicar no menu "Project", escolher a opção "Execute Script File"e selecionar o arquivo tcl "MakeProject.tcl"como na figura 2.6 abaixo:

Figura 2.6 - Executando Script TCL para criação do projeto. Fonte: Autoria Própria.



Uma vez selecionado o arquivo basta apertar o botão "Run"na interface gráfica que toda a rotina será executada e ao final será gerado o projeto completo, o qual inclusive pode ser usado para gravação na FPGA. A figura 2.7 abaixo representa o projeto completamente criado e configurado.

Figura 2.7 - Projeto do Libero criado a partir do script TCL. Fonte: Autoria Própria.



2.4 Discussão dos resultados

Como pode ser analisado a partir da figura 2.5, a FIFO projetada respondeu conforme o esperado. Na primeira fase, tendo em vista que no início do teste a FIFO encontrava-se vazia, o sinal indicativo de estado Vazio estava em nível lógico alto, enquanto os demais mantinham-se em nível lógico baixo, situação que foi alterada após a primeira escrita de dados fazendo com o que o sinal de Vazia mudasse para nível lógico baixo. Uma vez que a capacidade da FIFO era de 8 amostras, ao escrever a quarta amostrada o sinal indicativo de Metade Cheia mudou para nível lógico alto, enquanto os outros mantiveram-se baixo até a próxima escrita. Por fim, ao escrever o oitavo dado na FIFO o sinal indicativo de Cheia mudou para nível lógico alto e permaneceu assim até que fosse realizada a primeira operação de leitura.

Na segunda fase, a qual realizou-se oito processos de leitura, atentou-se para os sinais de estado e também para os dados retornados em cada leitura para verificar se estavam de acordo com a ordem de escrita. Como pode-se verificar, os dados sendo lidos respeitaram a sequência de escrita. Além disso, ao realizar a quarta leitura o sinal indicativo de Metade Cheia mudou para nível lógico alto até que fosse

realizada a próxima leitura. Por fim, ao realizar o oitavo processo de leitura o sinal indicativo de Vazia assume o nível lógico alto e assim permanece. Portanto, a partir da análise das formas de onda da figura 2.5 foi possível constatar que a FIFO estava se comportando de acordo com o esperado.

Já em relação a tabela 2.1 foi possível notar que de fato houve uma redução geral no percentual de ocupação dos blocos lógicos da FPGA com a mudança da FIFO. Além disso, o percentual negativo na coluna de Redução do bloco RAM1K18 deve-se ao fato de que a mudança fez com que os blocos FIFO, antes sintetizados em elementos lógicos, agora fossem sintetizados usando os blocos de memória apropriados da FPGA.

No caso da segunda atividade, conforme testes realizados a partir dos scripts TCL criados, verificou-se que tudo estava de acordo com o esperado, garantindo a automação de fluxo de projeto.

3 CONCLUSÕES

A partir dos resultados obtidos é possível constatar que o objetivo do projeto de realizar a implementação e verificação de uma memória FIFO em Verilog com comportamento síncrono, de forma a diminuir a quantidade de blocos lógicos necessários para a síntese lógica do EDC, diminuindo dessa forma a área de ocupação da FPGA foi alcançado com sucesso.

Por fim, no caso da segunda atividade desenvolvida durante a vigência da bolsa, conforme verificado a partir dos resultados obtidos, conclui-se que o objetivo de automação do fluxo de trabalho foi alcançado com sucesso, permitindo a economia de tempo para realização do setup do projeto.

REFERÊNCIAS BIBLIOGRÁFICAS

BREMEC, S.; URŠIČ, R.; MAVRIČ, U. Advantages of implementing digital receivers in field programmable gate arrays (fpga). In: **Beam diagnostics and instrumentation for particle accelerators. Proceedings, 6th European Workshop, DIPAC 2003**. [S.l.: s.n.], 2003. p. 175–177. 3

DUARTE, J. M. L.; LIMA, R. S. de; RAMOS, V. S.; CARVALHO, M. J. de. A multiuser decoder based on spectrum analysis for the brazilian environmental data collecting system. **International Journal of Satellite Communications and Networking**, Wiley Online Library, v. 39, n. 2, p. 205–220, 2021. 1

INPE. **EDC M2S Core**. [S.l.]. vii, 3, 4

MATTIELO-FRANCISCO, F.; CARVALHO, M.; SANTOS, M.; GARBI, G. Fostering environmental data collection with golds constellation. **Oral presentation at the UN/Brazil Symposium on Basic Space Technology**, 2018. 1

RAE, J. Detector de sinais para os satélites do sistema brasileiro de coleta de dados usando análise espectral digital. **Instituto Tecnológico de Aeronáutica, São José dos Campos**, 2005. 1