

Projeto de Laboratório de Sistema Digitais
SdSlideShow

Data: 20/06/2023 Grupo: 28



| Aluno | Número USP |
|--------------------------|------------|
| Bruno Barreto Rangel | 10738332 |
| Lucas Rodrigues Giacone | 11831901 |
| Henrique Hiroki Nakamura | 11200803 |

Sumário

| | |
|-------------------------------------|-----------|
| Descrição do projeto..... | 3 |
| Motivações..... | 3 |
| Concorrência..... | 3 |
| Requisitos do projeto..... | 4 |
| Diagrama de Blocos..... | 4 |
| Implementação..... | 5 |
| Diagrama de Arquitetura..... | 5 |
| Conclusão..... | 11 |
| Referências..... | 12 |

Descrição do projeto

A proposta do projeto SdSlideShow é fazer uma releitura do antigo projetor de slides, por meio da leitura de um cartão de memória SD que conta as imagens que serão apresentadas em um dispositivo de imagem utilizando comunicação VGA através FPGA para realizar o processamento da imagem lida contando também com um controle para passar e retornar a imagem anterior. O projeto será registrado no GitHub no seguinte endereço: <https://github.com/PCS-Poli-USP/projeto-final-projeto-28.git>

Motivações

Além de resgatar a ideia de apresentação instantânea fornecida pelo antigo aparelho de slides, a portabilidade e a possibilidade de exibir conteúdo digital para uma grande quantidade de pessoas, possibilita proporcionar entretenimento pessoal, apresentação de ideias e compartilhamento de informações e experiências, se mostrando um projeto de potencial relevante.

Concorrência

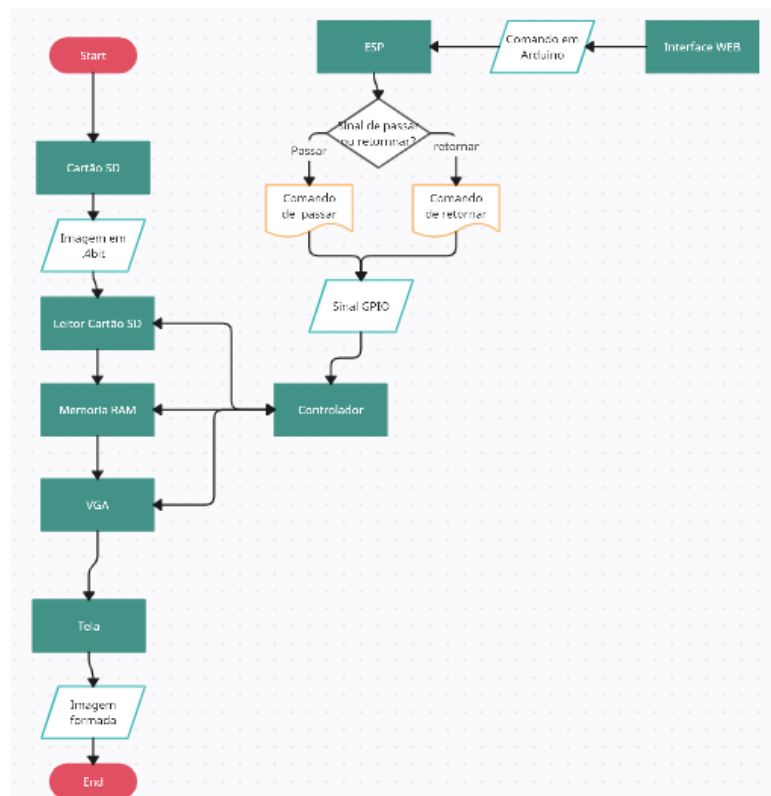
Após uma breve pesquisa na internet notou-se que existem produtos capazes de realizar a leitura de um cartão SD e realizar a projeção de imagem, sendo a maioria deles mini projetores portáteis, entretanto nenhum deles apresenta a possibilidade de conectar-se a qualquer entrada VGA para poder transmitir a imagem lida no cartão de memória, sendo incompatíveis com a transmissão para telas, sendo um diferencial do nosso projeto para os produtos já existentes.

Requisitos do projeto

Com o decorrer do desenvolvimento do projeto foram observados vários requisitos não antes estudados em sala para a implementação do que foi proposto. VGA, leitura de cartão SD, utilização da memória ram sendo partes essenciais para o funcionamento do projeto foram os principais objetos de estudo para entendermos os requisitos necessários para que conseguíssemos atingir o objetivo proposto. Além disso, foi proposto pelo professor Bruno de Carvalho Albertini para que utilizássemos o ESP8266, na parte de web server, para funcionar como um controle tanto da leitura do SD quanto das imagens a serem mostradas, tornando-o mais interativo e consequentemente aumentando o nível de complexibilidade do projeto.

Diagrama de Blocos

Foi desenvolvido um diagrama de blocos de alto nível com a ideia de como funcionará o projeto de maneira simplificada.



Note que temos duas principais entradas de dados: o cartão SD e a interface WEB do ESP. Tudo começa com a leitura do cartão SD, com imagens carregadas a serem utilizadas.

Após essa leitura, há um componente de controle que carrega e aloca as imagens na memória RAM da placa, esse controle recebe comandos do usuário, utilizando a interface WEB, por meio do ESP e de um servidor local.

A partir disso, o controlador troca as imagens a ser lidas no cartão SD que são enviadas para a RAM e as exibe na tela por meio da VGA

Implementação

O início do projeto decorreu de maneira bem fluida e sem demasiados problemas, conseguimos estudar todos os requisitos previstos para as semanas iniciais e ficamos à frente do cronograma previsto. Entretanto nas semanas seguintes ocorreram atrasos devido a feriados e a impossibilidade de frequentar os open labs nesses dias para testar a integridade dos códigos desenvolvidos. Tentamos correr para alcançar o planejamento e conseguimos de certa forma ter todas as partes funcionando individualmente do projeto. Entretanto na parte de implementação conjunta das partes apareceram problemas devido a alta performance que o projeto exigia da FPGA. Vale ressaltar que tivemos dificuldade em entender o funcionamento da memória ram, o documento era muito extenso e continha muitas informações não úteis ao projeto o que acarretou na demora para filtrar e consequentemente implementar o desejado para o projeto. Também foi implementada uma UART que recebesse do ESP o ip local em que está hospedado o web server para controle do SlideShow, sendo possível a sua visualização nos displays da placa.

Para o SD foi utilizado um código open source que utiliza o cartão em modo SPI como se fosse uma ROM. Não foi possível elaborar os estados pois esta é muito complexa e o RTL é inviável de se ver devido ao seu tamanho e complexidade. Código obtido da xesscorp : https://github.com/xesscorp/VHDL_Lib/blob/master/SDCard.vhd

Diagrama de Arquitetura

Segue os diagramas de arquitetura das partes separadas:

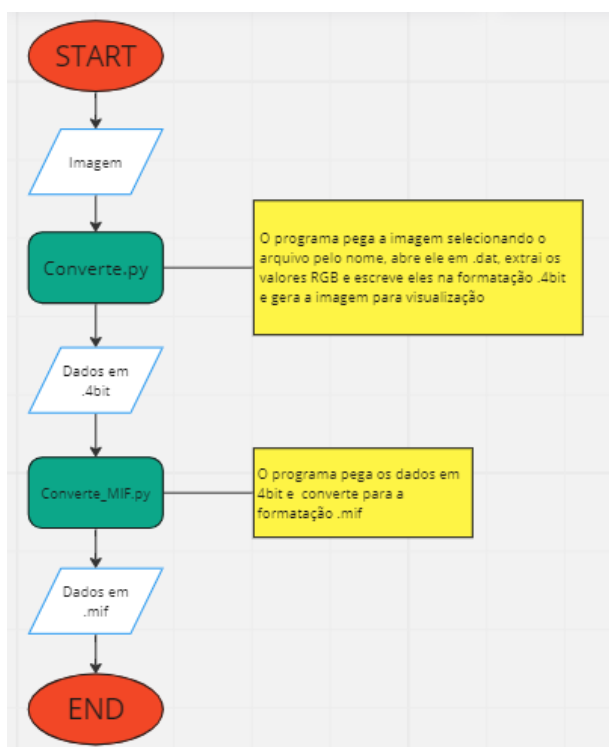


Figura (1): Diagrama de blocos do tratamento dos dados de imagem

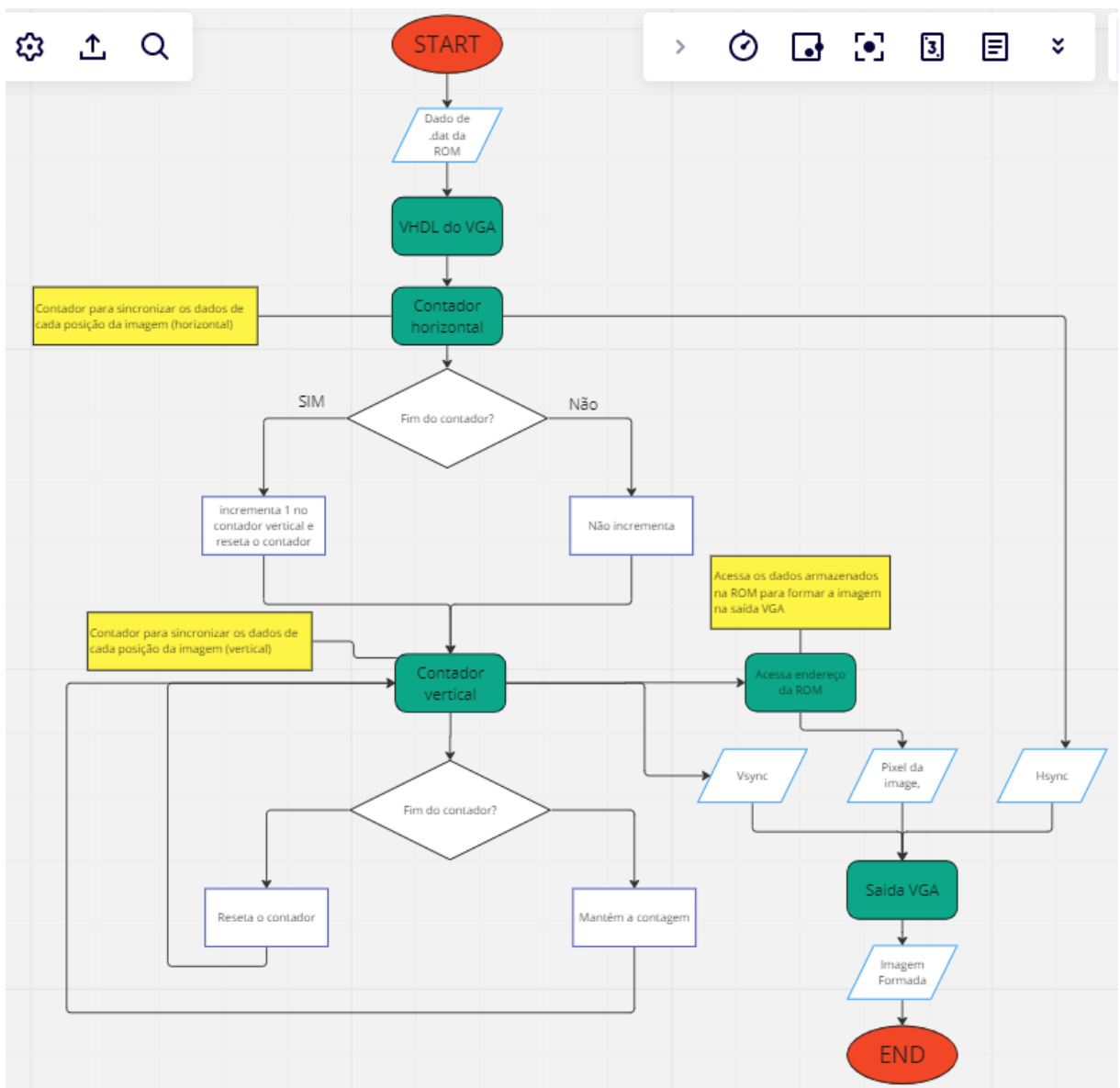


Figura (2): Diagrama de blocos do VGA

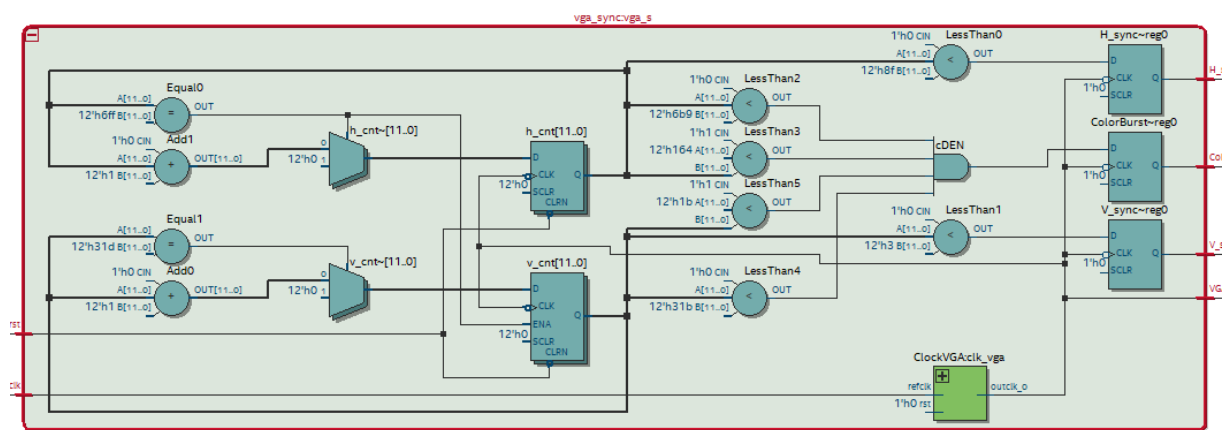


Figura (3): RTL do VGA

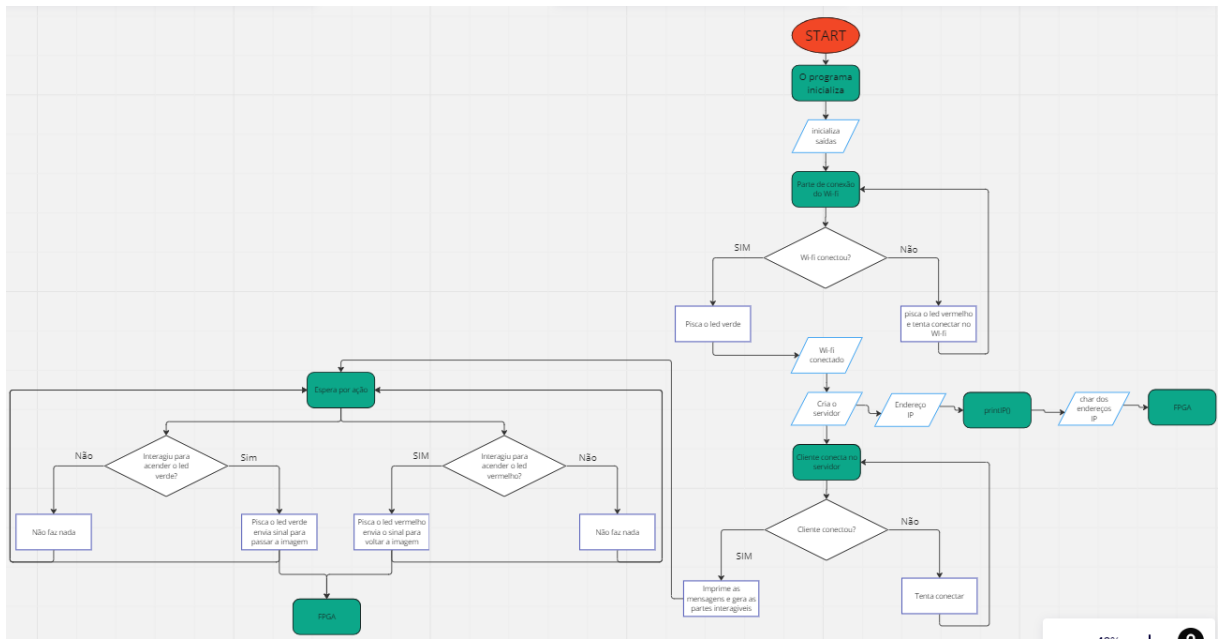


Figura (4): Diagrama de blocos do Web Server do arduino.

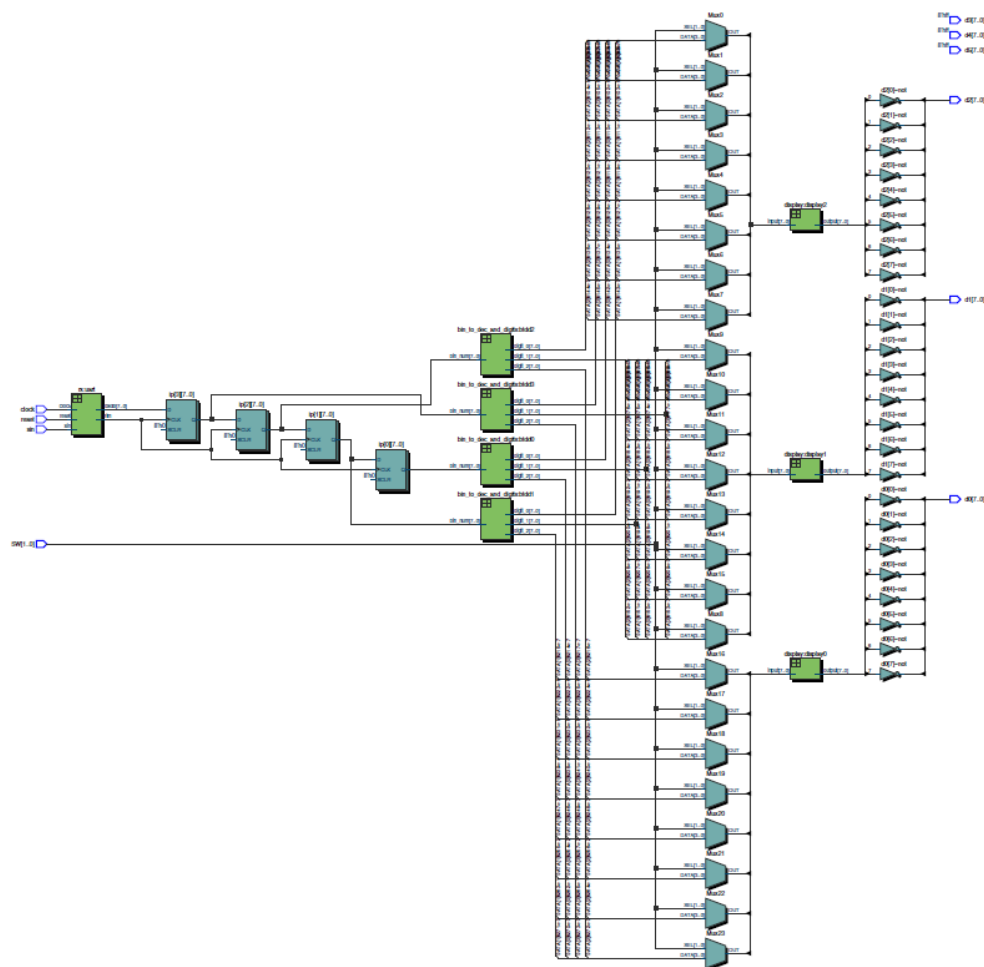


Figura (5): RTL do componente da UART do IP do webservice e de como se utilizam os displays.

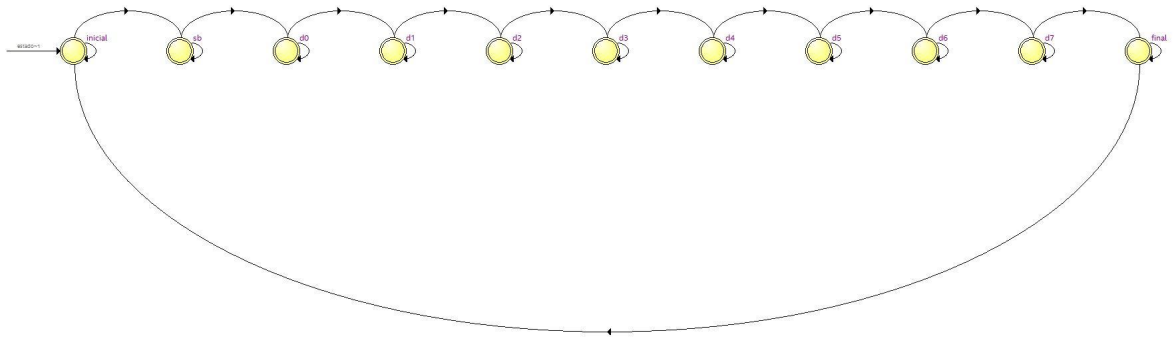


Figura (6): Diagrama da máquina de estados da UART utilizada para recepção do ip do web server..

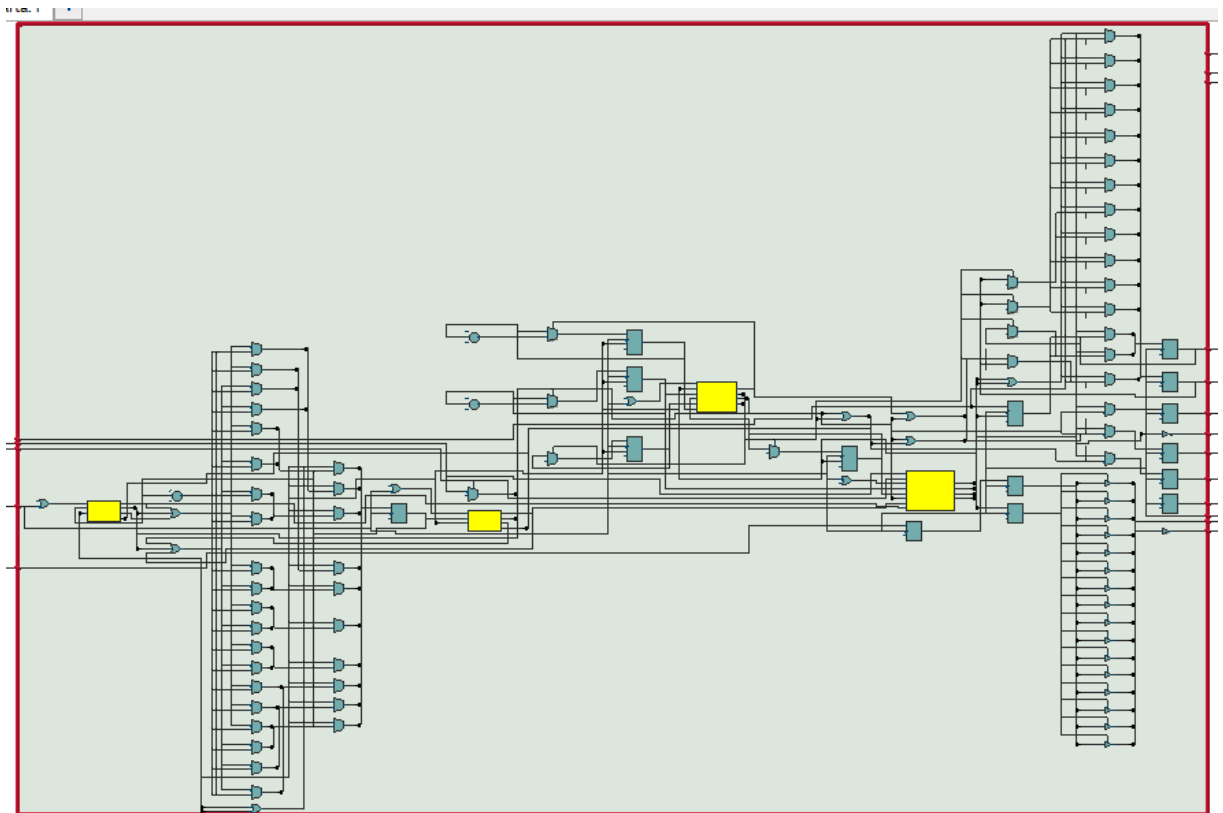


Figura (7): RTL do Geral

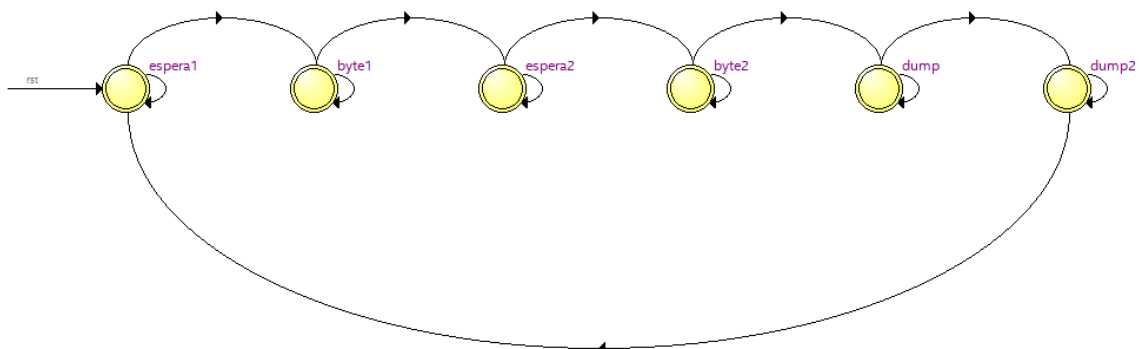


Figura (8): Máquina de estados da FIFO

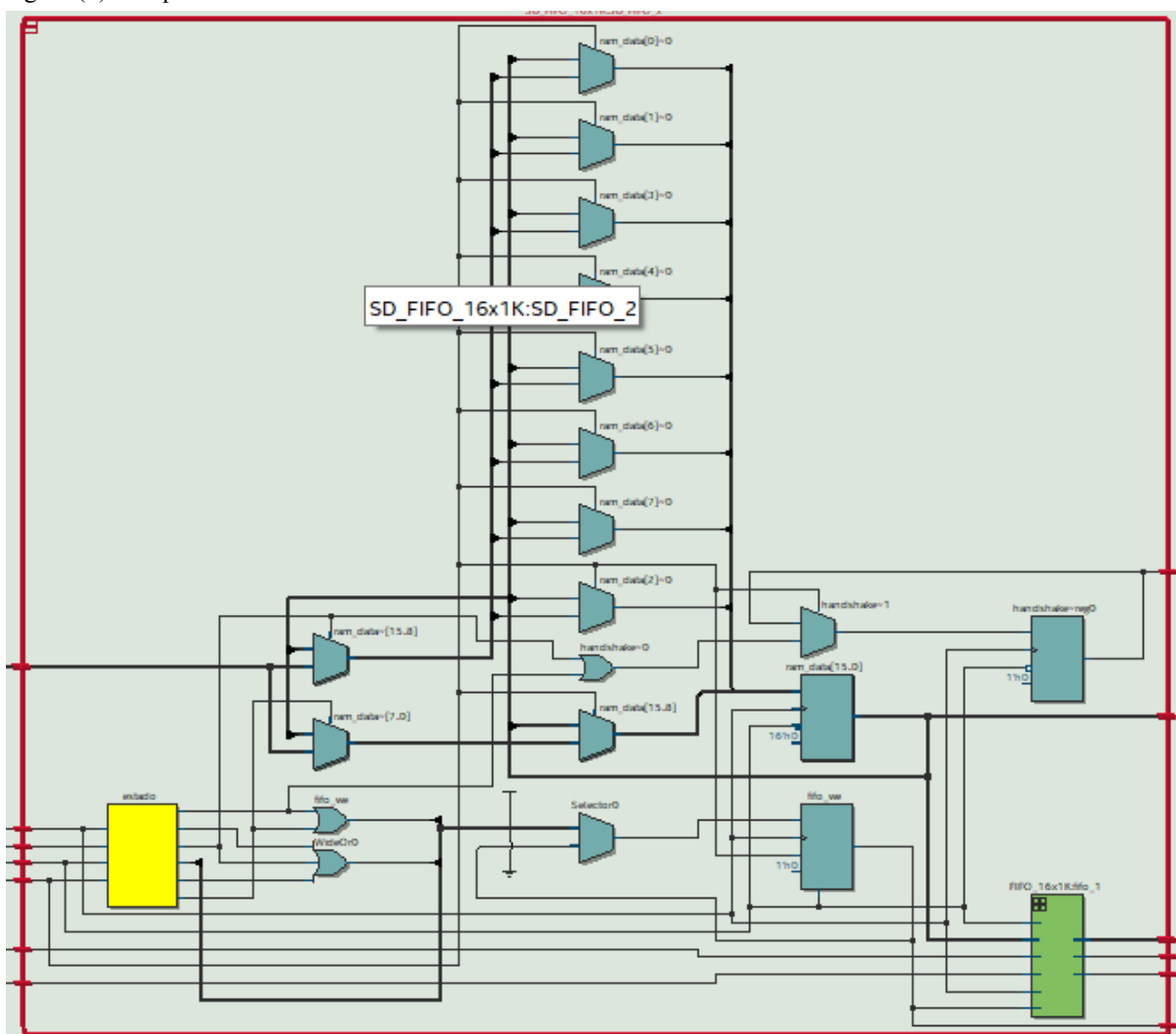


Figura (9): RTL da fifo

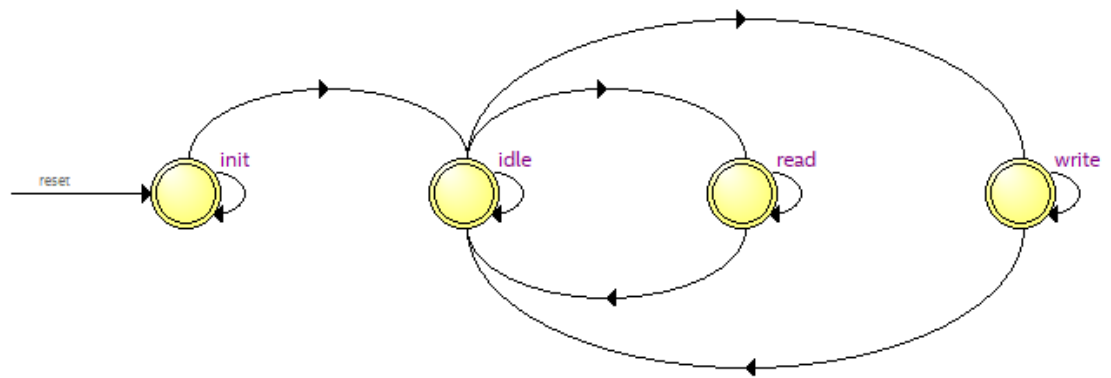


Figura (10): Máquina de estados do controle ram

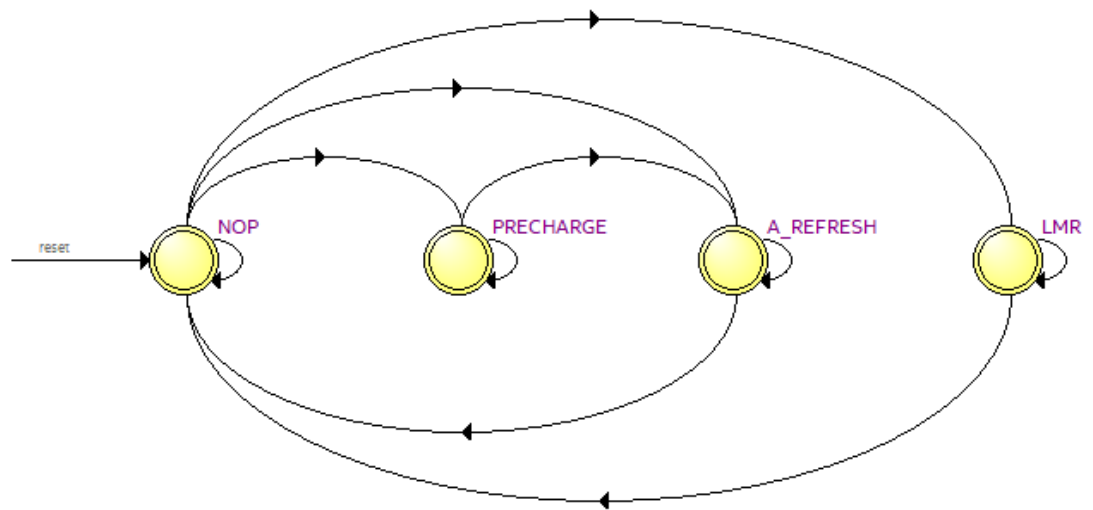


Figura (11): Máquina de estados do controle ram

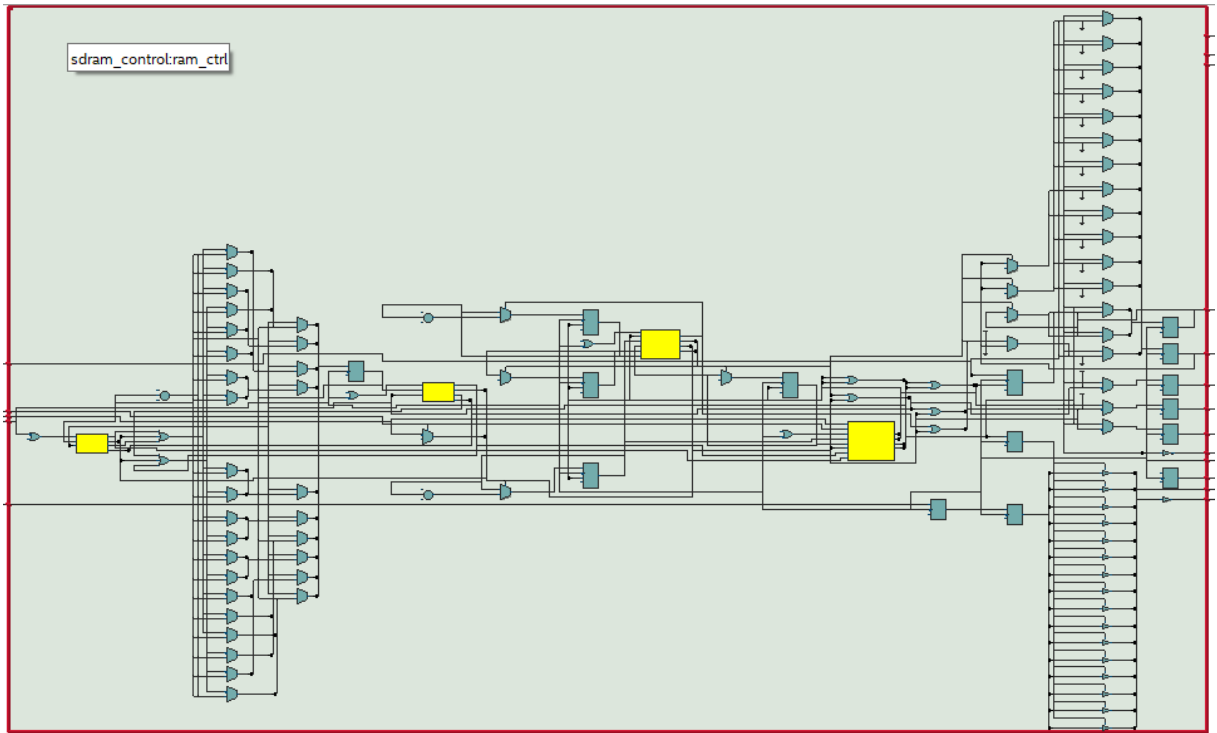


Figura (10): RTL do controle ram

Conclusão

O projeto foi concluído, apesar de ter sido “às pressas”. O grupo acredita que devido ao nível de complexibilidade para o funcionamento conjunto houve falta de tempo hábil para concluir o projeto sem correr com o cronograma. Havia partes separadas do projeto funcionando para a utilização, porém o maior desafio era sintetizar tudo num componente só que realizasse as implementações de forma sincronizada e correta, com todas as funcionalidades desejadas.

Como o esperado em outros relatórios prévios, foi possível correr atrás dos atrasos no projeto e não só finalizá-lo, como também otimizá-lo em alguns aspectos, como no dataflow da ram e da FIFO, a junção de todos os componentes foi mais fácil após entendimento do funcionamento da RAM e de como se dá a transmissão das informações.

Tendo o projeto funcionando sua parte principal de leitura do SD, escrita e leitura da memória ram e projeção do VGA, o projeto foi finalizado sem a integração da parte do ESP, sendo substituída por controle em dois push buttons na própria FPGA, mas com todos os componentes funcionando a sua implementação para o controle seria trivial, ficando como possibilidade para o futuro.

Referências

- <https://github.com/xesscorp/VHDL_Lib/blob/master/SDCard.vhd> Acesso em: 06, junho de 2023.
- <<https://github.com/esp8266/Arduino/blob/master/cores/esp8266/HardwareSerial.h>> Acesso em: 12, junho de 2023.
- <datasheet: "IS42/45R86400D/16320D/32160D"> Acesso em: 15, junho de 2023.