

PROPOSTA DE PLACA DE CIRCUITO IMPRESSO PARA CONSOLE DIRECIONADA À JOGOS RETRÔ

Alirio Freire de Oliveira Filho ¹

Márcio Renê Brandão Sousa²

Resumo

A cada nova geração de vídeo game lançado, fica visível o avanço tecnológico, nova tecnologia substituindo a antiga, isso ajuda no lançamento de novos jogos, mantendo os consoles em pé de igualdade com os jogos lançados para computador. Porém, os consoles das gerações passadas e a geração atual, coexistem até que, em algum momento, as fabricantes descontinuam, deixando de lançar conteúdo e conseqüentemente parando de fabricar o console. Diante desse cenário, este trabalho tem como objetivo propor uma placa de circuito impresso que seja compatível com antigos e novos jogos lançados para o console Master System.

Palavras-Chave: Consoles; Videogame; Retrô

1 Introdução

Os videogames para TV surgiram na década de 1970 quando do lançamento do console de jogos Odyssey, pela empresa Magnavox (DCI, 2020), com mais de cem mil unidades vendidas apenas no primeiro ano (Nascimento, 2015). Na mesma década, foi lançado o

¹ Centro Universitário Jorge Amado

² Doutor em Modelagem Computacional e Tecnologia Industrial pelo Centro universitário SENAI CIMATEC (2015); Mestre em Modelagem Computacional pelo Centro de Pós-Graduação e Pesquisa Visconde de Cairu (2007); Especialista em sistemas de Informação com ênfase em Componentes Distribuídos e Web pela Faculdade Ruy Barbosa (2001), Graduado em Engenharia Civil pela Universidade Católica do Salvador (1996) e Graduado em Processamento de Dados pela Universidade Salvador (1992). Atualmente professor de graduação do Centro Universitário Jorge Amado e da Faculdade de Tecnologia SENAI-CIMATEC

console Atari, pela Atari Inc., causando grande impacto no mercado de entretenimento ao disponibilizar o jogo Pong, e tempos depois, em 1983 a Gradiente iniciou a comercialização do console Atari no Brasil.

Nas décadas seguintes, novos videogames foram lançados, ganhando cada vez mais popularidade, principalmente entre o público infantojuvenil, dando origem a um mercado muito competitivo e inovador. Desde o lançamento dos primeiros videogames até os dias atuais, diversas gerações foram lançadas trazendo avanços tecnológicos significativos, contribuindo para a consolidação da indústria de jogos.

Em 2020, o mercado de jogos gerou \$177,8 bilhões, com uma perspectiva para 2023, de \$200 bilhões (Wijman, 2021). Nesse cenário, vemos um aumento na procura por jogos retrô. Como consequência, percebe-se um aumento no número de desenvolvedores indie, que são caracterizados por pequenas equipes ou mesmo uma única pessoa, com ou sem apoio financeiro de qualquer fonte.

Contudo, os desafios enfrentados pelos desenvolvedores na fabricação de novos jogos retrô são inúmeros, isto porque, com o lançamento das novas gerações de videogames e os avanços tecnológicos, os fabricantes de consoles encerraram a fabricação de gerações passadas, causando escassez de alguns chips utilizados nessas gerações.

Isto significa que, na atualidade, para lançar um jogo retrô destinado a consoles das décadas de 1980 e 1990, é necessário construir uma placa de circuito impresso nova, assim como em alguns casos, os chips denominados “Mappers”, que são utilizados para aumentar o tamanho da memória acessível pelo console, usada para armazenar o programa, gráficos e sons do jogo.

Dentre os consoles da década de 1980 e 1990 com grande repercussão no mercado de jogos, o Master System é um console de 3ª Geração produzido pela SEGA com 8 bits e Z80 (Zilog, 2016) como processador central, sendo portanto, objeto de estudo deste trabalho.

Diante desse cenário, este trabalho tem como objetivo propor uma placa de circuito impresso que seja compatível com antigos e novos jogos lançados para o console Master System.

2 Materiais e Métodos

Para atingir o objetivo proposto neste projeto, foi desenvolvida uma placa de circuito impresso, cujo diagrama esquemático elaborado através do software EasyEDA (EasyEDA, 2021), pode ser visto na Fig. 1. Nela é possível perceber os principais componentes de forma enumerada: (1) Interface de comunicação, (2) Chip Mapper, (3) Memória ROM, (4)

Memória Salve, (5) Regulador de tensão, (6) Capacitor, (7) Bloco para bateria e memória SRAM.

2.1 INTERFACE DE COMUNICAÇÃO

A interface de comunicação representa a forma de conexão do console com o cartucho do jogo. Ela fornece alimentação de 5 volts, com todo o barramento de endereço, data e alguns sinais de controle do processador Z80.

O processador Z80 utilizado no console Master System gerencia os acessos a interface de comunicação do cartucho disponibilizando todo o seu barramento de data, endereço e alguns sinais de controle. Na Tabela 1 temos os sinais utilizados na comunicação.

O processador em questão, usa 16-bits de endereçamento, permitindo a CPU acesso a 64KB de memória. Os 48KB do espaço de endereçamento de memória é utilizado para a interface de comunicação entre Z80 e cartucho, deixando 16KB de espaço de endereçamento para a memória interna do console.

2.2 Chip Mapper

Para o chip mapper, optou-se pelo uso de um Complex Programmable Logic Device (CPLD), foi escolhido o XC9572XL (Xilinx, 2007) que opera com tensão de 3.3V mas, é tolerante a entradas de 5.0V. Com 72 macrocells e suporte a frequências de até 178MHz.

Na implementação lógica do chip mapper, o endereçamento total de \$FFFF foi dividido em 4 slots de \$4000 bytes, sendo os três primeiros utilizados pelo mapper e o quarto e último sendo reservado para memória interna do console. A saber, slots referem-se a uma área de endereço do processador em que qualquer parte da memória ROM pode ser mapeada.

Registadores foram utilizados para parametrização do mapper, que ocorre durante uma escrita nos seguintes endereços: \$FFFC, \$FFFD, \$FFFE e \$FFFF. Esses endereços estão mapeados na região do quarto slot \$C000-\$FFFF, região destinada a memória interna do console, e por este motivo, mesmo que o mapper não implemente a leitura desses registradores, uma tentativa de leitura retornará o último valor escrito, pois a memória interna irá conter o valor.

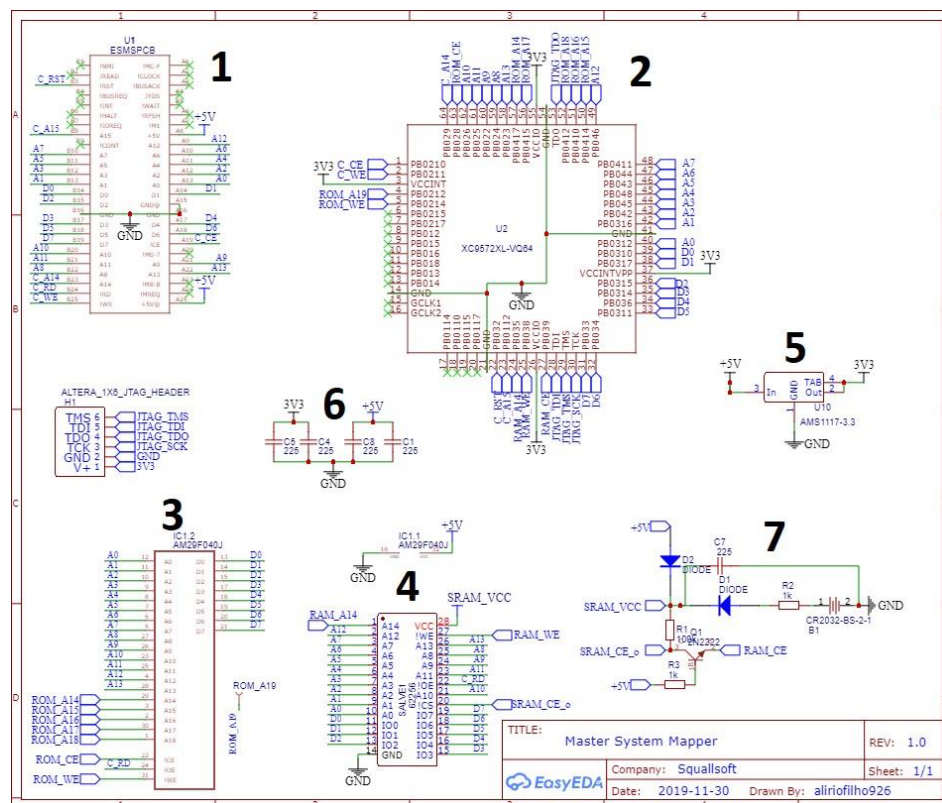


Figura 1: Diagrama esquemático da placa de circuito impresso

Tabela 1: Sinais utilizados na interface de comunicação

Sinal	Direção	Definição
Endereço 0 - 15	Output	Barramento de 16bits de endereço do processador Z80
Data 0 - 7	Input / Output	Barramento de 8bits data do processador Z80
Reset	Output	Sinal de reset
CE	Output	Indica quando a CPU esta acessando a região do cartucho
RD	Output	Indica quando a CPU esta lendo no barramento de data
WR	Output	Indica quando a CPU esta escrevendo no barramento de data
+5V	Output	Alimentação de 5.0 volts
GND	Output	Referência de tensão 0

Para controlar o acesso a memória de save, foi criado um registrador no endereço \$FFFC que possui apenas um bit sendo utilizado, porém no chip mapper oficial da SEGA existem outros bits sendo utilizados. Essas funções foram desabilitadas, sem causar impacto nesse projeto.

Conforme pode ser visto na Tabela 2. Cada bit controla uma função interna e quando o bit 4 desse registrador está em nível lógico 1, a memória de save fica disponível para acesso no endereço \$8000-\$BFFF, comumente chamado de slot 2, não permitindo acesso a memória ROM. Para preservar o vetor de interrupção que fica localizado no início do primeiro banco da memória rom, optou-se por travar o registrador \$FFFD no banco 0. Desse modo, todos os acessos no slot 0 \$0000-\$3FFF retorna os primeiros 16384 bytes da memória ROM.

Tabela 2: \$FFFC - RAM Acesso.

Bit	Função
7	Sem uso
6	Sem uso
5	Sem uso
4	Sem uso
3	Habilita RAM (\$8000-\$BFFF)
2	Sem uso
1	Sem uso
0	Sem uso

Os demais registradores \$FFFE e \$FFFF representam os slots 1 e 2 respectivamente, onde parte da memória ROM pode ser mapeada Tabela 3, permitindo ao console acessar, em blocos de \$4000 bytes por vez todo conteúdo das memórias contidas no cartucho, e, dessa forma, superar o limite de \$FFFF bytes imposto pelo espaço de endereçamento do processador Z80. Durante o processo de inicialização do console, o sinal de **reset** permanece em nível lógico baixo até que a alimentação interna do console estabilize, quando a alimentação estiver estabilizada, o sinal de **reset** vai a nível lógico alto liberando o processador Z80 para início da leitura do programa.

Tabela 3: \$FFFD-\$FFFF - ROM Bancos.

Registro	Slot	Endereço da Janela
\$FFFD	0	\$0000-\$3FFF
\$FFFE	1	\$4000-\$7FFF
\$FFFF	2	\$8000-\$BFFF

Conforme pode ser visto na Fig. 2, a lógica do chip mapeado utiliza o sinal de *reset* disponível na interface de comunicação para inicializar os registradores internos com os valores padrões Tabela 4.

Tabela 4: Registro.

Registro	Valor	Descrição
\$FFFC	0	RAM disable
\$FFFD	0	\$0000-\$3FFF
\$FFFE	1	\$4000-\$7FFF
\$FFFF	2	\$8000-\$BFFF

Finalizado o ciclo de **reset** do console, o mapper está pronto para aceitar acessos de escritas nos registros e leitura nos slots. Ocorrendo uma leitura no endereço do slot 0 (\$0000-\$3FFF) a região \$0000-\$3FFF da memória estará mapeada. Ocorrendo uma leitura no endereço do slot 1 (\$4000-\$7FFF), a região \$4000-\$7FFF da memória estará mapeada. Ocorrendo uma leitura no endereço do slot 2 (\$8000-\$BFFF) a região \$8000-\$BFFF da memória estará mapeada. Permitindo que o programa consiga acessar os parâmetros de inicialização localizados em \$1FF0 ou \$3FF0 ou \$7FF0.

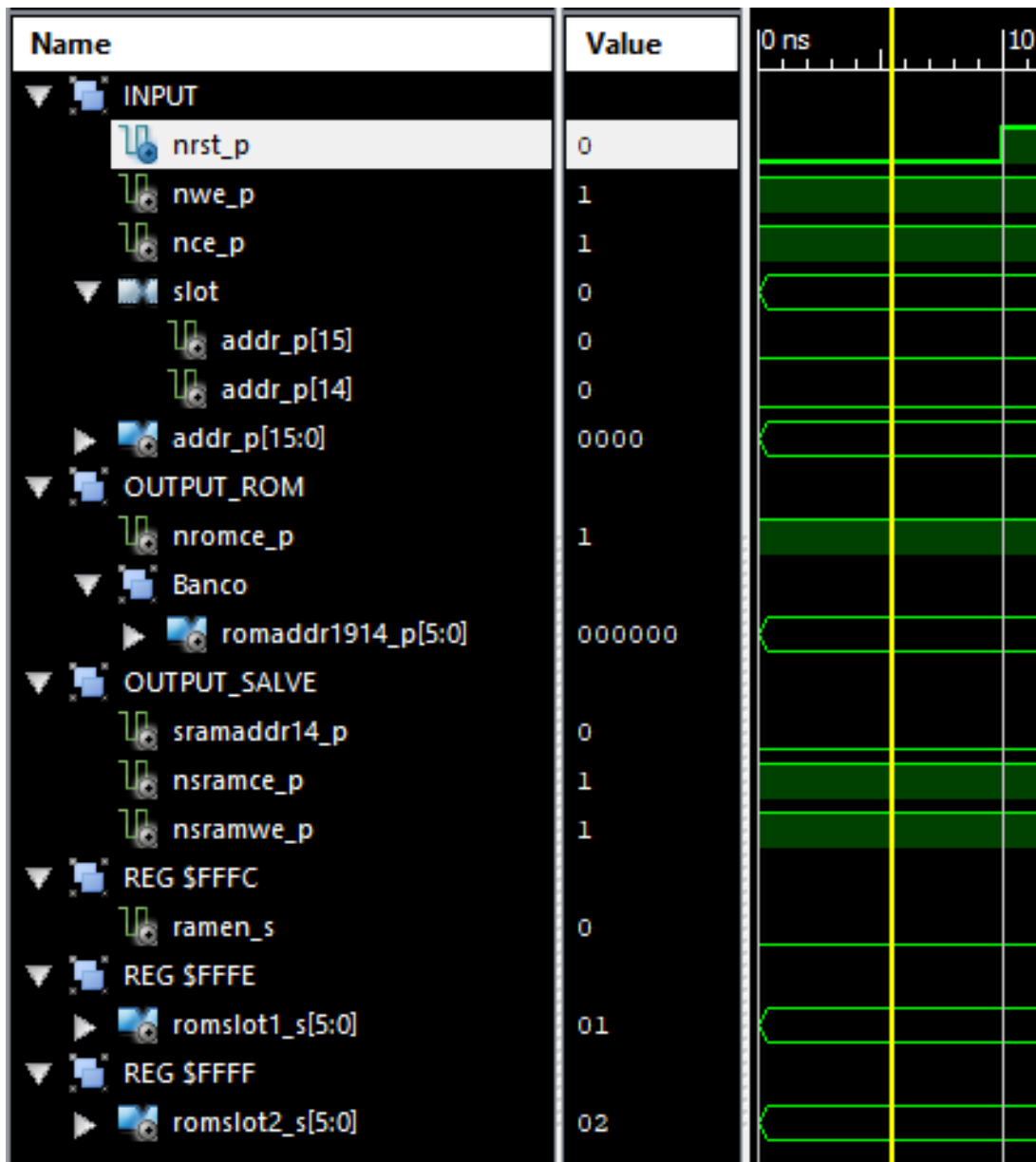


Figura 2: Mapper reset waveform

Tabela 5: Resultado da compilação da lógica do mapper

Uso de Macrocells	Uso de Pterms	Uso de Registros	Uso de Pinos	Block de funções
26/72 (37%)	80/360 (23%)	15/72 (21%)	38/52 (74%)	51/216 (24%)

Cada um dos registradores, \$FFFD, \$FFFE e \$FFFF pode ser representado como um array de bits, que determina a quantidade de bancos de 16384 bytes de cada slot, como também define o tamanho total de memória gerenciada pelo mapper. Para esse projeto, foram utilizados 6 bits, podendo gerenciar até 64 bancos de 16384 bytes, conforme as equações Eqs. (1) e (2).

$$nBancos = 2^6 \quad (1)$$

, logo

$$RomSize = nBancos * 16384 \quad (2)$$

, sendo nBancos o número de bancos, e RomSize o tamanho total gerenciado pelo mapper, também será a definição do tamanho da memória ROM. Na Tabela 5 vemos o resultado da compilação.

2.3 Memória ROM

Para o armazenamento do programa principal foi utilizado o Chip AM29F040 (MXIC, 2012) memória flash, com 4 Megabits (524,288 x 8 Bit) CMOS 5.0 volts. Este chip foi escolhido por ser uma memória Flash, com possibilidade de ser apagado e regravado.

2.4 Memória de Salve

Para a memória de salve, foi utilizada uma Ferroelectric RAM (FeRAM ou FRAM), Ferroelectric Random-Access Memory, que é um tipo de memória não volátil, cujo

modelo é FM1808 (Cypress, 2019) com 32768 bytes, sem a necessidade de uso de bateria.

2.5 Regulador de Tensão e Capacitores

Foi utilizado regulador de tensão linear o chip AMS1117 (AMS, 2007) para reduzir o potencial de tensão de 5.0V (voltagem de trabalho do console), para 3.3V (voltagem alvo do mapper).

Para filtrar ruídos da linha de alimentação, foram adicionados capacitores de filtragem e desacoplamento.

2.6 Bloco bateria

Esse bloco, formado por componentes discretos, tem a função de manter uma voltagem de *standby*, utilizando bateria, para o caso da utilização de memória SRAM em substituição da FRAM no que tange a memória de salve.

2.7 Software utilizados

Para a confecção do diagrama esquemático da placa de circuito impresso Fig. 3, foi utilizado o EasyEDA, software disponibilizado de forma gratuita na web. Para descrever o comportamento do mapper foi utilizado o software ISE WebDesign na versão 14.7 e na simulação utilizou-se o ISim, ambos disponibilizado pela Xilinx.

No que tange a linguagem de descrição, foi escolhida VHDL que significa VHSIC ou *Linguagem de Descrição de Hardware de Circuitos Integrados com Altíssima Velocidade*.

3 Resultados e Discussão

Para validar a placa proposta, foram realizados dois tipos de testes, sendo o primeiro o teste de bancada, que após alcançados os resultados satisfatórios, foram disponibilizados 3 protótipos montados da versão final da placa de circuito impresso.

3.1 Testes de bancada

Foram realizados testes em cada registrador usando a ferramenta ISim, a fim de garantir seu pleno funcionamento validando o diagrama esquemático e lógica empregada na descrição de hardware em VHDL,

Inicialmente, foram efetuados testes com o registrador \$FFFE que corresponde ao slot 1, respondendo a ciclos de leitura do console no intervalo de endereço \$4000 até \$7FFF,

Durante os testes desse registrador, conforme pode ser visto na Fig. 4, foi constatado que a lógica está de acordo com o proposto neste artigo.

A Fig. 5 apresenta os testes do registrador \$FFFE que representa o slot 2, respondendo a ciclos de leitura do console no intervalo de endereço \$8000 até \$BFFF,

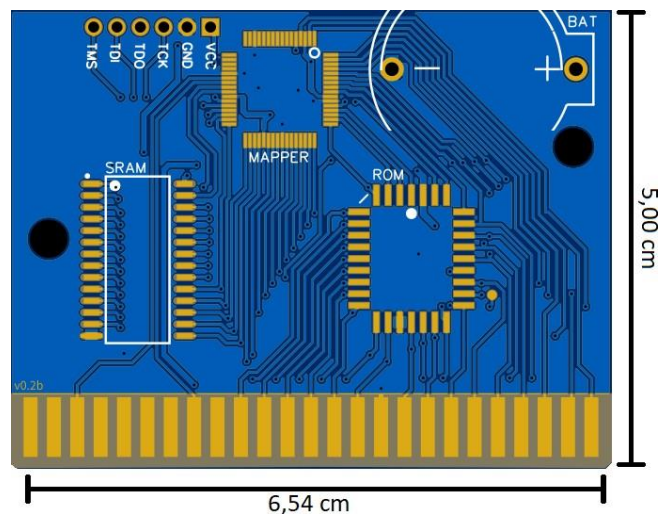


Figura 3: PCB Final com dimensões de 6,54 x 5,00cm

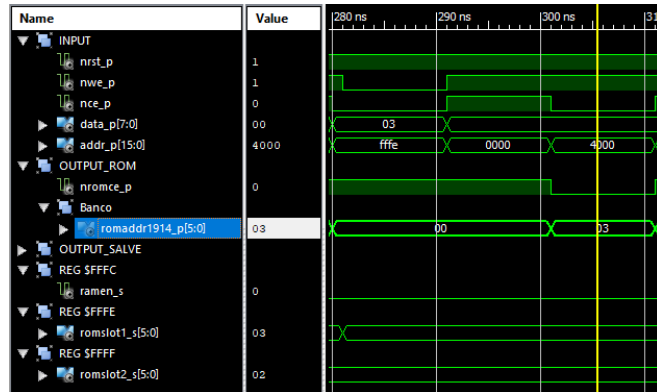


Figura 4: Acessando banco slot 1

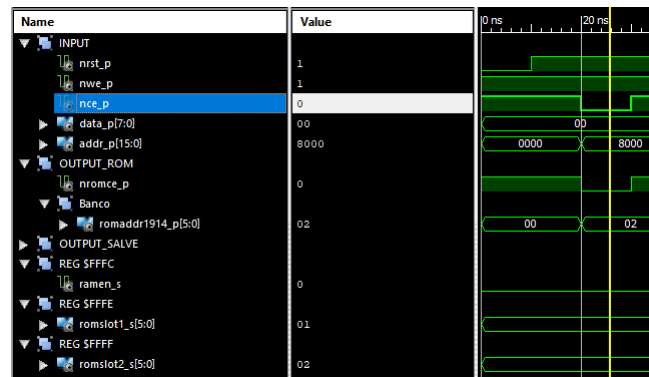


Figura 5: Acessando banco slot 2

Durante os testes desse registrador, conforme Fig. 5, foi constatado que a lógica está de acordo com o proposto neste artigo.

Na Fig. 6 é possível verificar o acesso ao registro \$FFFC e conseqüentemente o acesso a memória de salve, onde:

No tempo de 20 ns, temos a leitura no slot 2, com acesso da memória ROM,

No tempo de 40 ns, temos a escrita no registrador

\$FFFC do valor binário **00001000**, habilitando o acesso a memória de save,

No tempo de 60 ns, é possível verificar que a memória de save foi acionada como esperada,

No tempo de 80 ns, temos a escrita no registrador

\$FFFC do valor binário **00000000** no registrador \$FFFC, desabilitando o acesso à memória de save,

No tempo de 100 ns, é possível verificar que a memória ROM foi acionada.

3.2 Teste com usuários

Tendo sido validada a placa de circuito impresso e a lógica do mapper, foram montados 3 protótipos com a versão final da placa de circuito impresso e disponibilizado a 3 usuários para testes com os seguintes jogos gravados:

Protótipo **(1)Mortal Kombat** é um jogo de luta desenvolvido e publicado pela Midway Games, e recebe versões doméstica para diversos consoles pela Acclaim Entertainment.

Protótipo **(2)Phantasy Star** é um jogo de RPG desenvolvido e publicado pela SEGA.

Protótipo **(3)Sonic The Hedgehog 2** é um videogame de plataforma desenvolvido e publicado pela Sega.

Foi elaborado um questionário com perguntas para que os usuários pudessem responder após o uso dos jogos, com intuito de validar o funcionamento da PCB. Durante o período de 01/11/2021 até 06/11/2021 foi possível aos usuários registarem, através do site <http://squallsoft.ddns.net/sms> respostas para as seguintes perguntas:

- (1) Houve alguma tentativa de iniciar o jogo mal sucedida?
- (2) Aconteceram travamentos do jogo durante o período em que foi utilizado o protótipo?

- (3) Foi possível efetuar o save no protótipo disponibilizado com o jogo Phantasy Star?
- (4) Houve perda de save no protótipo disponibilizado com o jogo Phantasy Star?
- (5) Houve alguma falha não listada nas perguntas que gostaria de informar?

Com base nos testes efetuados, apesar de uma amostra pequena, verificou-se com a análise do questionário aplicado Tabela 6 que não houveram incidentes registrados, o sistema de save sem a utilização de bateria se mostrou muito eficaz, mantendo os dados de save após o desligamento do console.

Tabela 6: Testes com usuários

Jogo	Questão 1	Questão 2	Questão 3	Questão 4
<i>Mortal Kombat</i>	Não	Não	N/A	Não
<i>Phantasy Star</i>	Não	Não	Sim	Não
<i>Sonic The Hedgehog 2</i>	Não	Não	N/A	Não

4 Conclusão

O presente trabalho trouxe uma compreensão da realidade no que tange o desenvolvimento de um circuito elétrico e elaboração da placa de circuito impresso, em consonância, possibilitou aos desenvolvedores de conteúdo indie destinados ao console Master System, uma solução completa à falta de componentes na confecção da eletrônica do cartucho.

A utilização da proposta desenvolvida no presente trabalho, mostrou-se viável nos testes de bancada, e nos testes com usuários constatado através de feedback.

O presente estudo deixa margem para melhorias, pois possibilita a utilização de memória ROM, com capacidades maiores que a utilizada no estudo atual, ademais, abre espaço para completude dos registradores analogamente aos demais mappers da SEGA.

Em estudos futuros, os autores tem uma expectativa numa versão futura, a adição das melhorias citadas e possíveis outras. Disponibilizando espaço ainda maior para desenvolvimento de *games* no master System.



Figura 6: Acesso a memória de salve

Referências

AMS (2007). *1A LOW Dropout Voltage Regulator*. Disponível em <http://www.advanced-monolithic.com/pdf/ds1117.pdf>. Acesso 29 de nov 2021.

Cypress (2019). *256-Kbit (32K x 8)Bytewise F-RAM Memory*. Disponível em <https://www.cypress.com/file/136446/download>. Acesso 28 de nov 2021.

DCI, R. (2020). *História do videogame: relembre os consoles que marcaram época*. Disponível em <https://www.dci.com.br/tecnologia-e-games/historia-do-videogame/4270/>. Acesso 2 de nov 2021.

EasyEDA (2021). *An Easier and Powerful Online PCB Design Tool*. Disponível em <https://easyeda.com/>. Acesso 9 de nov 2021.

MXIC (2012). *MX29F040C*. Disponível em <https://www.macronix.com/Lists/Datasheet/Attachments/7266/MX29F040C,%205V,%204Mb,%20v2.2.pdf>. Acesso 29 de nov 2021.

Nascimento, A. (2015). *INDÚSTRIA DE VIDEOGAMES*. Disponível em <https://cepein.femanet.com.br/BDigital/arqTccs/1211390345.pdf>. Acesso 29 de nov 2021.

Wijman, T. (2021). *Global Games Market to Generate 175.8Billionin2021; DespiteaSlightDecline, theMarketIsOnTracktoSurpass200 Billion in 2023*. Disponível em <https://newzoo.com/insights/articles/global-games-market-to-generate-175-8-billion-in-2021-despite-a-slight-decline-the-market-is-on-track-to-surpass-2>. Acesso 30 de nov 2021.

Xilinx (2007). *XC9572XL HIGH Performance CPLD*. Disponível em: https://www.xilinx.com/support/documentation/data_sheets/ds057.pdf. Acesso 6 de nov 2021.

Zilog (2016). *Z80 Microprocessors*. Disponível em <http://www.zilog.com/docs/z80/um0080.pdf>. Acesso 14 de nov 2021.