

INTEGRAÇÃO DA SIMULAÇÃO COMPUTACIONAL A DISPOSITIVOS RECONFIGURÁVEIS PARA APOIAR O ENSINO DE SISTEMAS COMPLEXOS

Explorando o uso do CompSim no nível técnico.

Camila Valdez Ribeiro
camila.ribeiro00@academico.ifs.edu.br

Luiz Fernando Batista Morato
fernandomorato93@gmail.com

Micael Domingos Cruz de Jesus
micaelloiro@hotmail.com

Milena Santos Do Nascimento
milena.nascimento068@academico.ifs.edu.br

Pedro Silva dos Santos
pedro.santos061@academico.ifs.edu.br

Edson Barbosa Lisboa
ebl2@cin.ufpe.br

Guilherme Álvaro Rodrigues Maia Esmeraldo
guilhermealvaro@ifce.edu.br

Resumo – Em disciplinas de Organização e Arquitetura de Computadores (OAC) há uma carência de simuladores que abordem de forma simples e abrangente o complexo conteúdo. Assim, o objetivo é que o CompSim contribua na aprendizagem dos alunos através das funcionalidades do ambiente de simulação que possibilitam a interação com um hardware real através da integração de microcontroladores e/ou dispositivos eletrônicos reconfiguráveis, proporcionando uma melhor perspectiva acerca do funcionamento dos dispositivos, tanto em hardware quanto em software. Desta forma é possível validar conceitos complexos em computação. O trabalho foi dividido em sete atividades fazendo uso de Arduinos, um Módulo didático Zilocchi e diversos componentes eletrônicos, além da plataforma CompSim. Foram desenvolvidos cinco códigos principais em *Assembly* para o processador virtual Cariri e um modelo de esteira robótica com Arduino para experimentos com a plataforma. Foi desenvolvido material para futuros trabalhos, desse modo, contribui-se para o aprimoramento do ensino de sistemas complexos.

Palavras-Chave: Eletrônica; Organização e Arquitetura de Computadores; CompSim.

INTRODUÇÃO

A disciplina de Organização e Arquitetura de Computadores (OAC) está presente em cursos técnicos e superiores, no entanto, segundo Zorzo *et al.* (2017), é uma das disciplinas mais complexas presentes em cursos de Computação.

Por isso, o conteúdo de OAC é complexo e difícil de assimilar por ter detalhes eletrônicos presentes no funcionamento de computadores de difícil visualização e que necessitam de um alto nível de abstração para a sua compreensão, o que pode tornar o processo de ensino-aprendizagem pouco atrativo. Uma solução para a visualização desses detalhes é a utilização de uma ferramenta de software para a simulação de plataformas computacionais e que, adicionalmente, permita a interação com dispositivos eletrônicos reais. O CompSim é uma ferramenta que inclui a plataforma de hardware simulável e configurável. Esta plataforma, denominada Mandacaru, contém um processador conceitual (o Cariri), memória RAM, cache, periféricos de entrada e saída além de barramentos compartilhados. A Figura 1 ilustra a interface gráfica do CompSim (LEDS/LEA, 2017).

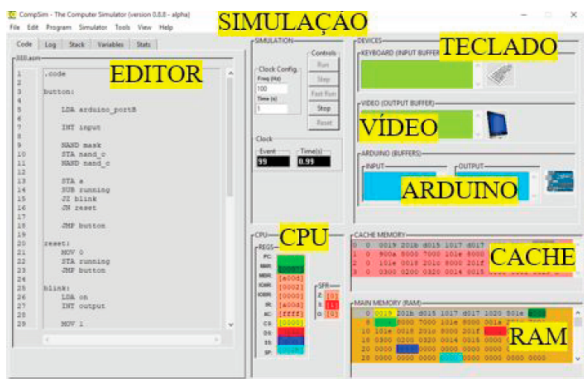


Figura 1 - Plataforma simulável Mandacaru.
Fonte: LEDS/LEA, 2017.

MATERIAL E MÉTODOS

O estudo iniciou com perspectiva de ser do tipo didático, considerando que as atividades realizadas possuem caráter referencial prático e/ou teórico para discentes e docentes. A pesquisa foi segmentada em sete atividades, sendo: estudo do simulador CompSim, experimentos de controle de *hardware*, controle de registrador em FPGA, implementação da biblioteca de funções booleanas no Cariri, Esteira Arduino, implementação do PWM e do sensor ultrassônico na Esteira Arduino, comunicação da Esteira Arduino com o CompSim. Posteriormente, foram realizadas pesquisas bibliográficas, a fim de aprimorar os conhecimentos teórico e prático em Arquitetura e Organização de Computadores (AOC).

Para a execução das atividades foram utilizados os seguintes componentes e equipamentos: Esteira Arduino em MDF para Projetos, Arduino UNO R3 + Cabo USB, Motor DC 3-6V 80RPM com Caixa de Redução, Sensor Ultrassônico de Distância HC-SR04, Driver Controlador para motor DC - IRF520, Intel Galileo Gen 2, Módulo didático Zilocchi MPL-D10EX, LEDs, botões, dentre outros.

Os experimentos foram divididos desta maneira:

Estudo do simulador CompSim, no intuito de compreender seu funcionamento e suas funcionalidades;

Experimentos de controle de *hardware*: o controle do *hardware* (LEDs, botões) foi realizado por meio do comando da placa

Arduino através da plataforma de simulação Mandacaru do CompSim, conforme o fluxograma presente na Figura 2;

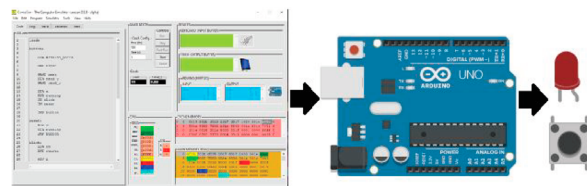


Figura 2 - Fluxograma CompSim-Hardware.
Fonte: Próprios autores, 2020.

Controle de registrador em FPGA: utilizou-se a placa Arduino como intermediário na comunicação do CompSim com o registrador de deslocamento (figura 3), feito em FPGA para a realização de operações binárias;

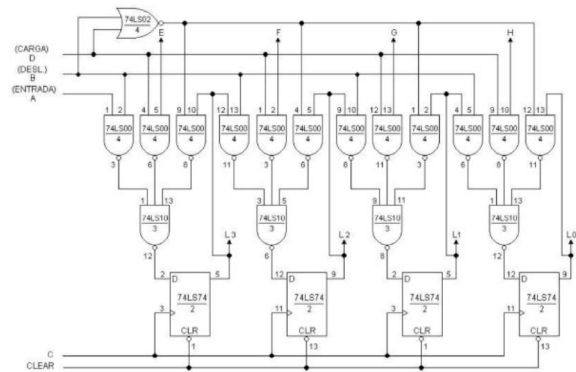


Figura 3 - Registrador de deslocamento com carregamento paralelo. Fonte: DATAPOOL, 2020.

Implementação da biblioteca de funções booleanas no Cariri: dentre as funções lógicas elementares, o processador Cariri apenas suporta nativamente a função NAND, escolhida pois o postulado da álgebra de Boole estabelece a “suficiência de NAND” a partir da qual pode-se implementar todas as demais funções.

Assim, percebeu-se a necessidade de enriquecer o conjunto de instruções do simulador por meio do desenvolvimento de uma biblioteca de funções booleanas que permitiram uma análise comportamental das operações binárias;

Esteira Arduino: foram realizados experimentos a partir da integração entre a esteira e o Arduino UNO R3;

Implementação do PWM e do sensor ultrassônico na Esteira Arduino: foi utilizado

o PWM (Modulação de Largura de Pulso), uma técnica que obtém sinais analógicos por meios digitais, assim foi feita a contenção da potencial/velocidade do motor, já o sensor ultrassônico de distância HC-SR04 transmite um pulso de som de alta frequência através do pino *Trigger* (entrada), o pino *Echo*(saída) recebe esse pulso, com isso, pode-se calcular a distância entre o sensor de sonar e o objeto, devem estar entre 2-400 cm. Desta maneira, foi realizada a detecção/contagem dos objetos;

Comunicação da Esteira Arduino com o CompSim: controlou-se a esteira através do simulador, conforme o diagrama presente na Figura 4.

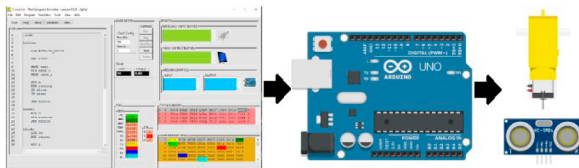


Figura 4 - Diagrama da comunicação CompSim-Arduino-Sensor Ultrassônico-Motor. **Fonte:** Próprios autores, 2020.

RESULTADOS E DISCUSSÃO

Um dos objetivos deste projeto foi desenvolver, validar e testar as funções lógicas para proporcionar uma melhor perspectiva do funcionamento dessas operações, tanto em *hardware* quanto em *software* e validar conceitos simples e complexos empiricamente.

Assim, no trabalho foram desenvolvidos os códigos a seguir: Código 1: consiste na implementação entre o simulador-microcontrolador, na qual CompSim envia dados para o Arduino, que processa os dados e os envia para o LED e o botão. Assim, é possível avaliar o funcionamento do *hardware* por meio da plataforma CompSim;

```
blink:
    LDA on
    INT output
    MOV 1
    STA running
    LDA arduino_portD
    INT output
```

Figura 5 - Código 1 parcial: controle de LED. **Fonte:** Próprios autores, 2020

```
LDA a
NAND a
STA not_a
LDA c
NAND c
STA not_c
LDA not_a
NAND not_c
STA res

LDA res
NAND res
STA res
```

Figura 6 - Código 3 parcial: função NOR. **Fonte:** Próprios autores, 2020

Para a realização dos códigos 4 e 5, necessitou-se primeiro criá-los no IDE do Arduino para auxiliar na criação dos códigos em linguagem Assembly, ou seja, para o simulador CompSim. Desta maneira, O Código 4 foi gerado para controlar o motor a partir dos valores preestabelecidos da Tabela 1, tanto na IDE do Arduino quanto no CompSim.

Ciclo de Trabalho	$pwm = velocidade * 85$	Tensão Média
0%	$0 = 0 * 85$	0,0V
33%	$85 = 1 * 85$	1,7V
67%	$170 = 2 * 85$	3,3V
100%	$255 = 3 * 85$	5,0V

Tabela 1 - Relação entre ciclo de trabalho e tensão média de saída. **Fonte:** Próprios autores, 2020.

Assim, quando chega um novo dado, checa se ele está entre o intervalo de 0 a 3. Se for um dado válido, então faz-se uma soma que corresponde ao produto de 85 com a variável “velocidade” (correspondente a escolha do ciclo de trabalho) e aplica o total desta soma na variável “pwm”, que configura o ciclo de trabalho do motor.

Já o Código 5, feito no IDE do Arduino, primeiro é mandado o pulso para ativar *Trigger*, é realizado um *loop* para a contagem do tempo para saber o tamanho do pulso do *Echo*, com o tamanho do pulso do pulso é identificada a distância em que o objeto se encontra através da Equação 1:

$$\text{float distancia} = \text{pulso} * 0.034 / 2 * 5 \quad (1)$$

Logo, se o tamanho da distância for menor que 50 cm, o objeto é detectado e contabilizado.

Porém, não foi possível o agrupamento do sensor ultrassônico com o simulador, visto que há uma limitação de leitura do código inviabilizando a implementação no CompSim, que não disponibiliza uma frequência suficiente para a realização do cálculo do tamanho do pulso gerado pelo *Echo*. Com a Equação 2 foi possível calcular a frequência máxima que o CompSim disponibiliza.

$$f = \frac{n}{\Delta t} = \frac{4210}{30} \approx 140 \text{ Hz} \quad (2)$$

No qual n e Δt são, respectivamente, o número de eventos ocorridos e o tempo necessário para que ocorram em segundos.

Assim, cada evento no CompSim é executado em aproximadamente 7 milissegundos. O pulso do *Echo* funciona entre 25 milissegundos e 150 microssegundos, desta maneira, encontra-se dentro do alcance do sensor. Porém, fora da faixa que necessitaríamos para a determinação da distância do objeto.

CONCLUSÕES

Quando se iniciou o trabalho de pesquisa percebeu-se a grande dificuldades que os alunos tinham para aprender os conceitos inerentes a OAC e a sistemas digitais em geral. A aplicação metodológica do ambiente de simulação CompSim mostrou-se promissor como uma importante ferramenta para o aprendizado e ensino de Arquitetura e Organização de Computadores e sistemas digitais, incluindo um amplo conjunto de conteúdos. Alinhado aos

objetivos de controle de protótipos robóticos e teste de *software*, foi desenvolvido muito material para futuros trabalhos, incluindo, mas não limitado à implementação de novos sensores (como o sensor ultrassônico) e a implementação de uma biblioteca de funções lógicas no CompSim. Desse modo, espera-se o aprimoramento do ensino de sistemas complexos, com suporte de dispositivos e componentes para aplicações na indústria 4.0, IoT (*Internet of Things*), inteligência artificial, dentre outras.

REFERÊNCIAS

DATAPOOL. **MÓDULO 8810: MANUAL TEÓRICO E PRÁTICO**. Disponível em: <http://paginapessoal.utfpr.edu.br/pfrimer/sistemasdigitais/atividades-praticas/Modulo%208810_%20M%20anual%20Teorio%20e%20Pratico.pdf>. Acessado em 08 de out. 2020.

LEDS/LEA. **CompSim - The Computer Simulator**. Disponível em: <<http://compsim.crato.ifce.edu.br/classroom.html>>. Acessado em: 07 de out. 2020.

ZORZO, A. F.; NUNES, D.; MATOS, E. S.; STEINMACHER, I.; LEITE, J. C.; ARAÚJO, R.; CORREIA, R. C. M.; MARTINS, S. **Referenciais de Formação para os Cursos de Graduação em Computação**. SBC, 2017. p. 153.