



PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Instituto de Ciências Exatas e de Informática

Exercício prático Arquitetura de Computadores II

Vitor Costa Oliveira Rolla

Pontifícia Universidade Católica de Minas Gerais

09 de junho de 2024

Resumo

O projeto consiste no desenvolvimento de um sistema integrado para simulação e execução de operações de uma Unidade Lógica Aritmética (ULA) de 4 bits. Este sistema é dividido em duas partes principais: um **hardware externo** implementado em Arduino e um **software para PC** que atua como interface de usuário e compilador.

1 Hardware Externo (Arduino)

O componente de hardware externo é uma **ULA de 4 bits** projetada para operar com dois operandos de 4 bits, denominados A e B, e uma instrução de 4 bits. A ULA é implementada em um microcontrolador Arduino e possui as seguintes características:

- **Entradas:** Recebe dados (operandos A e B) e instruções através da porta serial do Arduino. Os valores são interpretados em formato hexadecimal.
- **Saídas:** O resultado das operações da ULA é exibido em quatro LEDs, conectados aos pinos digitais 13, 12, 11 e 10 do Arduino, representando o bit mais significativo (MSB) ao menos significativo (LSB), respectivamente.
- **Conjunto de Instruções:** A ULA suporta um conjunto específico de 16 instruções lógicas e aritméticas, conforme detalhado na Tabela 1.

Tabela 1: Conjunto de Instruções da ULA

Função	Mnemônico	Código Hexa
Lógico um	umL	0
Lógico zero	zeroL	1
$A+B'$	AonB	2
$A'+B'$	nAonB	3
$(A.B)'$	AeBn	4
B'	nB	5
A'	nA	6
$nA \times nB$	nAxnB	7
$A \oplus B$	AxB	8
A copiaA	copiaA	9
B copiaB	copiaB	A
$A.B$	AeB	B
$A.B'$	AenB	C
$A'.B$	nAeB	D
$A+B$	AoB	E
$(A'.B)'$	nAeBn	F

- **Memória Interna:** O programa do Arduino gerencia um vetor de memória que atua como registradores e área de programa. As quatro primeiras posições são dedicadas a variáveis de controle e dados:
 - **PC (Program Counter):** Índice da instrução atual na memória.
 - **W:** Armazena o resultado da última operação da ULA.

- **X:** Armazena o operando A.
- **Y:** Armazena o operando B.
- **Execução de Programa:** O Arduino é capaz de carregar um programa (sequência de instruções) na memória, executar as instruções sequencialmente, atualizar os valores das variáveis na memória e exibir o estado da memória (DUMP) após cada execução de instrução nos LEDs e na saída serial.

2 Software para PC

O software para PC, a ser desenvolvido em linguagens como C, C++, C#, Java ou Python, serve como uma interface para o usuário interagir com a ULA do Arduino. Suas funcionalidades incluem:

- **Leitura de Programa Fonte:** Lê um arquivo de texto (`.ula`) contendo o programa em mnemônicos (ex: `testeula.ula`).
- **Geração de Código Hexadecimal:** Transforma o programa fonte em um arquivo de código hexadecimal (`.hex`), onde cada linha representa os valores (operandos e instrução) a serem enviados para o Arduino (ex: `testeula.hex`).
- **Comunicação Serial:** Envia os dados e instruções do arquivo hexadecimal para o Arduino via porta serial, permitindo a execução remota do programa na ULA.

3 Fluxo de Operação

O ciclo de operação do sistema envolve a criação de um programa fonte pelo usuário, sua conversão para código hexadecimal pelo software no PC, o envio desse código para o Arduino, onde é carregado na memória e executado pela ULA. Os resultados são visualizados nos LEDs do Arduino e monitorados através de *dumps* de memória na saída serial.

Este projeto visa demonstrar a compreensão da arquitetura de computadores, programação de microcontroladores e desenvolvimento de software de interface, com ênfase na clareza do código e na correta implementação das especificações da ULA.

4 Arduino

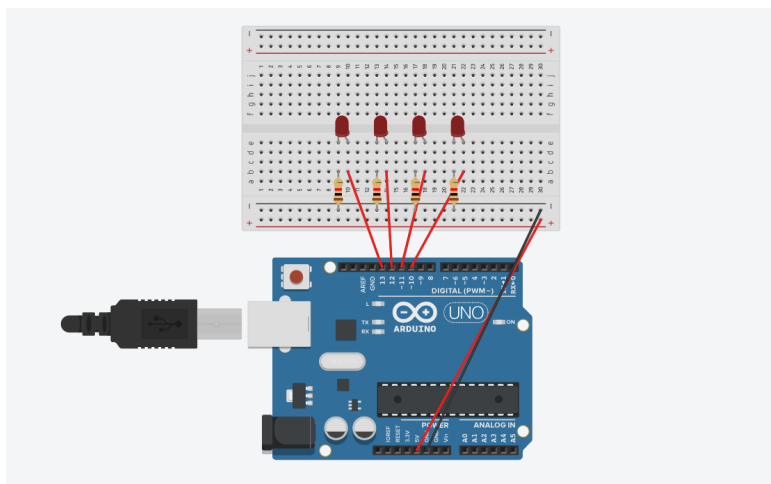


Figura 1: Circuito Tinkercad