



Atividade Laboratorial 2

Guia de Laboratório

ARQUITETURA DE COMPUTADORES

DEPARTAMENTO DE ENGENHARIA ELETRÓNICA E DE
TELECOMUNICAÇÕES E COMPUTADORES

Março de 2023

1 Objetivos

Este trabalho de Arquitetura de Computadores (AC) tem como objetivo o estudo dos processos de tradução de programas escritos em linguagem *assembly* P16 para código máquina e o seu carregamento em memória, bem como do funcionamento do barramento do processador P16. A componente experimental do trabalho é realizada sobre a placa SDP16 [3].

2 Requisitos

- Placa SDP16 com cabo USB;
- Computador pessoal com uma instalação nativa, ou em máquina virtual, do sistema operativo Microsoft Windows 10, ou de uma versão superior a esta;
- Assemblador p16as para o processador P16;
- Ferramenta de edição de código-fonte *assembly* P16, e.g. Notepad++ ou Visual Studio Code.

3 Trabalho de preparação à atividade laboratorial

As tarefas indicadas nesta secção constituem o trabalho de preparação para esta atividade laboratorial, pelo que deverão ser realizadas antecipadamente à sessão em laboratório. **Os grupos de alunos/as que não cumprirem este requisito ficarão impedidos de realizar a aula laboratorial.**

3.1 Preparação do ambiente de trabalho

1. No seu computador pessoal, crie uma diretoria com o nome `lab02` dentro da diretoria base que guarda os trabalhos realizados em AC no corrente semestre letivo.
2. Copie para essa diretoria o "Guia de codificação das instruções do P16" [1], que contém a descrição do mapa de codificação das instruções do P16. Este documento está disponível para descarregamento na página de meta disciplina de AC na plataforma Moodle.
3. Copie ainda para essa diretoria o "Manual de Utilização da Placa de Desenvolvimento SDP16" [3]. Este documento também está disponível para descarregamento na página de meta disciplina de AC na plataforma Moodle.

3.2 Análise de programas e sua codificação em linguagem máquina do P16

1. Analise o programa apresentado na Listagem 1, escrito em linguagem *assembly* P16, e indique a sua funcionalidade.
2. Indique, justificando, o número de ciclos de relógio gastos na execução de cada iteração do ciclo `loop` implementado no programa (linhas 3 a 7).
3. Sabendo que o programa será localizado em memória a partir do endereço `0x0000`, indique os valores que deverão ser associados aos símbolos `var2_addr`, `var1`, `var2` e `var3`. Justifique a sua resposta.

1 Utiliza a instrução `load` para buscar o endereço em que o `var2` se encontra e guarda-o no `r0`. Entra num ciclo infinito, guardando o 1 byte do que está guardado no endereço do `r0` em `r1(load)`, incrementa uma unidade guardando o valor final no `r1`. Posteriormente irá guardar em memória o valor de `r1` no 1º byte do endereço de `r0`. Regressando ao início do `loop`.

2 Como sabemos cada instrução apresenta 5 fases, ou seja 5 ciclos de clock; Instruction Fetch, Instruction Decode, Execute, Memory, Righ Back. Tanto a instrução load como store utilizam as 5 fases, no entanto, a instrução add nao necessita de acesso para leitura ou escrita utilizando apenas 4 fases. E por fim a instrução branch não apresenta a necessidade das ultimas duas fases ficando apenas com 3 fases. Deste modo o numero total de ciclos é 17, 5+4+5+3.

```

1      .text
2      ldr    r0, var2_addr
3  loop:
4      5 ldrb  r1, [r0, #0]
5      4 add  r1, r1, #1
6      5 strb  r1, [r0, #0]
7      3 b    loop
8
9  var2_addr:
10     .word  var2
11
12     .data
13  var1:
14     .byte  0xAC
15  var2:
16     .byte  126
17  var3:
18     .word  0x2022
    
```

Listagem 1: Programa de teste.

3 Tendo em conta que iniciamos o programa com o endereço 0x0000, e como cada instrução ocupa 16 bits determinamos que os valores dos endereços associados aos símbolos var2_addr e var1 são 0x000A e 0x000C. Não esquecendo que a instrução .byte ocupa apenas 8 bits, o valor do endereço associado ao símbolo var2 é 0x000D e o do símbolo var3 é 0x000E.

- Traduza, manualmente, o programa para código máquina do P16. Use uma tabela com o formato indicado na Tabela 1 para registar o resultado dessa codificação, considerando que cada linha da tabela deverá corresponder apenas a uma instrução do programa. Represente em notação hexadecimal, usando quatro dígitos, os valores dos endereços de memória e das instruções do programa.

Instrução	Endereço	Código máquina
ldr r0, var2_addr		

Tabela 1: Tabela exemplo para o registo da codificação para código máquina.

3.3 Entrega do trabalho de preparação

- Usando a aplicação navegadora de Internet (*browser*) da sua preferência, aceda à página de AC para a sua turma na plataforma Moodle do ISEL e selecione a atividade com o título "Preparação relativa à Atividade Laboratorial 2".
- Carregue no botão "Responder ao teste" para iniciar a atividade.

Atenção: Espera-se que apenas um dos elementos de cada grupo de alunos/as submeta as respostas através desta atividade.

4 Trabalho a realizar no laboratório

4.1 Preparação da placa SDP16

- Dirija-se à sala de apoio aos laboratórios e requisiite uma placa SDP16 e um cabo USB.
- Repita o procedimento descrito no ponto 4.1 do guia de laboratório da "Atividade Laboratorial 1" de AC [2] para validar o funcionamento da placa SDP16. Contacte o docente caso detete alguma anomalia no funcionamento da placa que lhe foi atribuída.

4.2 Abertura da ficha de aferição de conhecimentos da atividade laboratorial

1. Usando a aplicação navegadora de Internet (*browser*) da sua preferência, aceda à página de AC para a sua turma na plataforma Moodle e selecione a atividade com o título "Ficha de aferição de conhecimentos relativa à Atividade Laboratorial 2".
2. Carregue no botão "Responder ao teste" para iniciar a atividade, que tem uma duração máxima prevista de 3 horas e 30 minutos. Quando iniciar a atividade, o temporizador começará a contagem decrescente e não é possível parar ou fazer pausas.
3. Utilize os botões "Página seguinte" e "Página anterior" para navegar entre as várias perguntas que compõem a atividade.

Atenção: Espera-se que apenas um dos elementos de cada grupo de alunos/as submeta as respostas através desta atividade.

4.3 Carregamento manual de programas na placa SDP16

1. Posicione o interruptor SW6 da placa SDP16 na posição 'ON' para fornecer energia elétrica à placa.
2. Posicione o interruptor SW3 da placa SDP16 na posição 'DMA' para habilitar o funcionamento do módulo DMA. Este módulo possibilita o acesso a qualquer palavra do espaço de endereçamento do processador P16 para escrita de um novo valor, em notação hexadecimal, usando o teclado matricial alfanumérico de 16 teclas instalado na placa, ou a consulta do seu valor atual, em código binário, usando o mostrador de sinais.
3. Pressione o botão de pressão SW2 ('RESET') da placa SDP16 para fazer *hard reset* ao circuito de controlo do módulo DMA.
4. Posicione o interruptor SW5 da placa SDP16 na posição 'ADDR' para poder definir o endereço da posição de memória a aceder utilizando o teclado alfanumérico instalado na placa SDP16.
5. Utilize o teclado alfanumérico da placa SDP16 para estabelecer o valor do endereço de memória onde deverá ficar localizada a primeira instrução do programa, conforme definido no ponto 4 da secção 3.2. Verifique o estabelecimento deste valor por inspeção do estado dos LED 'A0' a 'A15' do mostrador de sinais da placa SDP16.
6. Posicione o interruptor SW5 da placa SDP16 na posição 'DATA' para poder definir o conteúdo da posição de memória estabelecida no ponto 5.
7. Repita os seguintes passos para cada uma das instruções constantes do programa apresentado na Listagem 1:
 - i) Introduza os quatro dígitos do código máquina da instrução, em notação hexadecimal, utilizando o teclado alfanumérico da placa SDP16;
 - ii) Valide o valor introduzido por inspeção do estado dos LED 'D0' a 'D15' do mostrador de sinais da placa SDP16;
 - iii) Pressione o botão de pressão SW7 ('NEXT') da placa SDP16 para selecionar o endereço em que será localizada a instrução seguinte do programa.
8. Reposicione o interruptor SW5 da placa SDP16 na posição 'ADDR' e, utilizando o teclado alfanumérico, estabeleça o endereço da posição de memória associada ao símbolo `var2_addr` do programa apresentado na Listagem 1. Verifique o estabelecimento deste valor por inspeção do estado dos LED 'A0' a 'A15' do mostrador de sinais da placa SDP16.

9. Coloque novamente o interruptor SW5 da placa SDP16 na posição 'DATA' e, utilizando o teclado alfanumérico da placa SDP16, introduza os quatro dígitos hexadecimais correspondentes ao valor do símbolo `var2_addr`. Valide o valor introduzido por inspeção do estado dos LED 'D0' a 'D15' do mostrador de sinais da placa SDP16
10. Reponha o interruptor SW5 da placa SDP16 na posição 'ADDR' e, utilizando o teclado alfanumérico, estabeleça o endereço associado ao símbolo `var1`. Verifique o estabelecimento deste valor por inspeção do estado dos LED 'A0' a 'A15' do mostrador de sinais da placa SDP16.
11. Recoloque o interruptor SW5 da placa SDP16 na posição 'DATA' e, com as adaptações necessárias, repita o procedimento indicado no ponto 7 para escrever na memória os valores iniciais das posições associadas aos símbolos `var1`, `var2` e `var3`.

4.4 Teste de programas na placa SDP16

1. Posicione o interruptor SW4 da placa SDP16 na posição 'STEP' para habilitar o funcionamento em modo passo-a-passo aquando da execução de programas na placa SDP16.
2. Posicione o interruptor SW3 da placa SDP16 na posição 'CPU' para habilitar o funcionamento do processador P16 instalado na placa SDP16.
3. Pressione o botão de pressão SW2 ('RESET') da placa SDP16 para fazer *hard reset* ao processador P16.
4. Verifique se o processador está a aceder ao endereço de memória pretendido analisando o estado dos LED 'A0' a 'A15' do mostrador de sinais da placa SDP16. Analise também o estado dos LED 'D0' a 'D15' deste mostrador para verificar se o conteúdo dessa posição de memória corresponde ao valor pretendido.
5. Usando uma tabela com o formato indicado na Tabela 2, em que cada linha deverá representar o resultado da execução de um ciclo máquina do processador, registe a atividade nos barramentos do processador para a execução passo-a-passo de duas iterações do ciclo implementado no troço de código programado na memória do sistema. Para tal, realize o seguinte procedimento:
 - i) Observe o estado dos LED "nRD", "nWRH" e "nWRL" do mostrador de sinais da placa SDP16 e registe nas colunas correspondentes da sua tabela os valores simbólicos associados (L – valor lógico zero; H – valor lógico um);
 - ii) Observe o estado dos LED 'A0' a 'A15' do mostrador de sinais da placa SDP16 e registe na coluna "Endereço" da sua tabela o valor correspondente, em notação hexadecimal, com quatro dígitos;
 - iii) Observe o estado dos LED 'D0' a 'D15' do mostrador de sinais da placa SDP16 e registe na coluna "Dados" da sua tabela o valor correspondente, em notação hexadecimal, com quatro dígitos;
 - iv) Pressione o botão de pressão SW7 ('NEXT') da placa SDP16 para avançar na execução do programa.

Instrução	Controlo			Endereço		Dados	
	nRD	nWRH	nWRL	A15 ... A0		D15 ... D0	
ldr r0, var2_addr							

Tabela 2: Tabela exemplo para o registo da atividade nos barramentos do processador.

4.5 Conclusão da ficha de aferição de conhecimentos da atividade laboratorial

No final da sessão de laboratório deverá submeter todas as respostas dadas às questões enunciadas na ficha de aferição de conhecimentos da atividade laboratorial para aferição do cumprimento dos objetivos propostos. Para tal, deverá usar o botão "Terminar Tentativa" que surge na página da última pergunta desta atividade.

Bibliografia

- [1] Dias, Tiago: *Conjunto de Instruções do P16 – Guia de Codificação das Instruções*. ISEL, Lisboa, Portugal, março 2022. https://iselppt.sharepoint.com/:b:/s/acp/Ecy-i_SPVx5LvplHIrKBwWkBuxZHYbvq0sVSB1eDZvi-Wg?e=dtgRK8 (Acedido em 13-03-2023).
- [2] Dias, Tiago: *Atividade Laboratorial 1 de Arquitetura de Computadores – Guia de Laboratório*. ISEL, Lisboa, Portugal, 3a edição, março 2023.
- [3] Paraíso, José e Tiago Dias: *Manual de Utilização da Placa de Desenvolvimento SDP16*. ISEL, Lisboa, Portugal, junho 2020. <https://iselppt.sharepoint.com/:b:/s/acp/EYLbn2wMpx5BsiVHpjEHmT4BQXC71rwSbEScMs-m0Ims2A?e=mZWNs9> (Acedido em 13-03-2023).