

ÍNDICE GERAL

AGRADECIMENTOS V

PREFÁCIO VII

ÍNDICE DAS SIMULAÇÕES XXI

1 - INTRODUÇÃO AO MUNDO DOS COMPUTADORES 1

1.1	O computador como ferramenta.....	2
1.2	A importância dos computadores	4
1.3	Processamento da informação.....	5
1.4	Estrutura básica de um computador	6
1.5	O mundo com apenas dois símbolos	9
1.6	Interação pessoa-computador	11
1.7	A gestão de um computador	14
1.8	A evolução dos computadores	16
1.9	Perspetivas de evolução futura.....	27
1.10	Conclusões.....	29

2 - O MUNDO BINÁRIO 31

2.1	Circuitos eletrónicos analógicos	32
2.2	Circuitos eletrónicos digitais	33
2.2.1	Funcionamento básico	33
2.2.2	Diagramas temporais	35
2.2.3	Portas lógicas.....	36
2.3	Álgebra de Boole.....	40
2.4	Funções lógicas	41
2.5	Circuitos combinatórios.....	48
2.5.1	Síntese de circuitos combinatórios	48
2.5.2	<i>Multiplexers</i>	50
2.5.3	Descodificadores	53
2.5.4	ROMs	56

2.6	Circuitos sequenciais.....	59
2.6.1	Elementos biestáveis	59
2.6.1.1	Trinco SR.....	59
2.6.1.2	Trinco D.....	60
2.6.1.3	Báscula D.....	61
2.6.2	Registos	65
2.6.3	Portas lógicas de três estados (<i>tristate</i>).....	67
2.6.4	Banco de registos.....	68
2.6.5	Contadores.....	70
2.6.6	Registos de deslocamento.....	75
2.6.7	Máquinas de estados.....	76
2.6.7.1	Modelo das máquinas de estados	76
2.6.7.2	Diagramas de estados.....	78
2.6.7.3	Máquinas de estados sintetizadas.....	85
2.6.7.4	Máquinas de estados microprogramadas	85
2.7	Representação de números.....	89
2.7.1	Números em base 10 (decimais) e 2 (binários).....	89
2.7.2	Números em base 16 (hexadecimais)	91
2.7.3	Potências de 2.....	93
2.7.4	Quantos <i>bits</i> para representar um número?.....	95
2.7.5	Representação de números negativos	96
2.7.6	Representação de números em complemento para 2	97
2.7.7	Extensão do número de <i>bits</i> de um número.....	99
2.8	Operações aritméticas.....	101
2.8.1	Soma de dois números binários	101
2.8.2	Subtração de dois números binários	102
2.8.3	Excesso.....	103
2.8.4	Multiplicação de dois números binários	105
2.8.5	Divisão de dois números binários.....	107
2.9	Conclusões.....	111
2.10	Exercícios	112
3 - O MEU PRIMEIRO COMPUTADOR 115		
3.1	Componentes básicos de um computador	116
3.2	RAM – a memória para guardar informação.....	117
3.3	O processador (PEPE-8).....	121
3.3.1	Unidade de dados	122
3.3.1.1	Registo na unidade de dados.....	122
3.3.1.2	Unidade aritmética e lógica (ALU).....	125
3.3.1.3	Funcionamento da unidade de dados	130
3.3.2	Unidade de controlo	133
3.3.2.1	Sinais de controlo.....	133
3.3.2.2	Contador de Programa (PC).....	134
3.3.2.3	Um programa simples	135
3.3.2.4	Constantes no programa.....	137

3.3.2.5	Saltos no programa	143
3.3.2.6	Funcionamento detalhado do programa	145
3.3.3	O processador (PEPE-8) e as memórias	148
3.3.3.1	Processador (PEPE-8).....	148
3.3.3.2	Memória de dados.....	149
3.3.3.3	Memória de instruções.....	150
3.4	Programação em baixo nível de um computador	152
3.4.1	Instruções em vez de sinais de controlo	152
3.4.2	Linguagem <i>assembly</i>	155
3.4.3	Implementação das instruções	158
3.4.4	Programação em linguagem <i>assembly</i>	163
3.4.5	Programação do PEPE-8 em <i>assembly</i> : contagem de <i>bits</i>	166
3.5	Periféricos	168
3.5.1	Estrutura do <i>hardware</i>	168
3.5.2	Programação com periféricos	176
3.5.2.1	Uso de periféricos de saída	177
3.5.2.2	Uso de periféricos de entrada.....	179
3.6	Soluções específicas ou genéricas?	181
3.7	Conclusões.....	182
3.8	Exercícios	183

4 - ARQUITETURA BÁSICA DE UM PROCESSADOR 187

4.1	Banco de registos	188
4.2	Endereços de dados e de instruções	194
4.2.1	Memórias de dados e de instruções separadas: <i>caches</i>	194
4.2.2	Espaço de endereçamento e mapa de endereços.....	196
4.3	Impacte da largura das instruções.....	198
4.4	Endereçamento de <i>byte</i> e de palavra.....	200
4.5	Codificação das instruções.....	204
4.6	Registos.....	207
4.7	<i>Bits</i> de estado	209
4.8	Conjunto de instruções	214
4.9	Instruções de salto	217
4.10	Instruções de transferência de dados.....	220
4.10.1	Combinações de operandos	220
4.10.2	Transferências entre registos	222
4.10.3	Transferência de uma constante para um registo.....	223
4.10.4	Transferências entre um registo e a memória	227
4.10.4.1	Endereços constantes e em registos	227
4.10.4.2	Modos de acesso à memória em 16 <i>bits</i>	228

4.10.4.3	Acesso à memória em 16 bits com índice variável	229
4.10.4.4	Acesso à memória em 16 bits sem índice	230
4.10.4.5	Acesso à memória em 16 bits com índice constante	231
4.10.4.6	Instruções de acesso à memória em 16 bits	233
4.10.4.7	Acesso à memória em 8 bits	234
4.10.4.8	Acesso à memória em 8 bits e 16 bits	239
4.10.5	Transferências para memória de uma constante ou memória	241
4.11	Instruções aritméticas	241
4.11.1	Instruções aritméticas mais simples	242
4.11.1.1	Soma e excesso: série de Fibonacci	243
4.11.1.2	Soma e transporte: números grandes	244
4.11.2	Multiplicação e divisão	245
4.12	Instruções lógicas	249
4.12.1	Funcionalidade das instruções lógicas	249
4.12.2	Expressões booleanas	252
4.12.3	Instruções de manipulação de um só bit	254
4.12.4	Operações lógicas com máscaras	258
4.12.4.1	Funcionamento das máscaras	258
4.12.4.2	Máscaras AND	259
4.12.4.3	Máscaras OR	261
4.12.4.4	Máscaras XOR	263
4.13	Instruções de deslocamento	265
4.13.1	Instruções de deslocamento linear	266
4.13.2	Instruções de deslocamento circular (rotações)	270
4.14	Modos de endereçamento	272
4.15	Conclusões	275
4.16	Exercícios	276

5 - PROGRAMAÇÃO DE UM COMPUTADOR 279

5.1	Um problema simples	280
5.1.1	Modo de atuação de um ser humano	280
5.1.2	Modo de atuação de um computador	280
5.2	Modelação do problema com fluxogramas	281
5.3	Programação em alto nível	284
5.4	Mapeamento da programação de alto nível em linguagem assembly	287
5.5	Dados, declarações e diretivas	289
5.5.1	Constantes simbólicas e a diretiva EQU	289
5.5.2	Variáveis	290
5.5.2.1	Tipos das variáveis	290
5.5.2.2	Acesso a variáveis de tipos de dados estruturados	291
5.5.2.3	Diretivas WORD, TABLE e STRING	292
5.5.3	A diretiva PLACE	295
5.5.4	Apontadores	299

5.6	Instruções	302
5.6.1	Atribuição e expressões.....	304
5.6.2	Decisão.....	305
5.6.2.1	Decisão simples.....	305
5.6.2.2	Decisão múltipla.....	306
5.6.3	Iteração.....	308
5.7	Rotinas	310
5.7.1	Estruturação do código.....	310
5.7.1.1	Funções nas linguagens de alto nível.....	310
5.7.1.2	Rotinas em linguagem <i>assembly</i>	314
5.7.1.3	Variante com apontadores.....	318
5.7.2	Mecanismo de chamada e retorno.....	320
5.7.2.1	Endereço de retorno.....	320
5.7.2.2	Chamada de rotinas com endereço de retorno em registo.....	321
5.7.2.3	Chamada de rotinas com endereço de retorno na memória (pilha).....	325
5.7.2.4	Qual dos mecanismos de chamada de rotinas se deve usar?.....	335
5.7.2.5	Variantes do funcionamento da pilha.....	337
5.7.3	Outras utilizações da pilha em rotinas.....	338
5.7.3.1	Guarda de registos nas rotinas.....	338
5.7.3.2	Variáveis locais.....	346
5.7.3.3	Passagem de parâmetros e do resultado.....	349
5.7.3.4	Contextos de chamada das rotinas.....	355
5.7.3.5	Recursividade.....	360
5.8	Gestão dos dados	364
5.8.1	Quando os registos não chegam.....	364
5.8.2	Cálculo de expressões.....	365
5.8.3	Execução de instruções imbricadas.....	366
5.8.4	Tabelas.....	367
5.8.4.1	Tabelas de uma só dimensão.....	367
5.8.4.2	Tabelas multidimensionais.....	372
5.8.4.3	Tabelas de apontadores.....	378
5.8.5	Estruturas de dados dinâmicas (montão).....	382
5.8.6	Listas ligadas.....	386
5.9	Desenvolvimento de programas	393
5.9.1	Ciclo de desenvolvimento.....	393
5.9.2	Programação em alto nível ou em linguagem <i>assembly</i> ?.....	398
5.9.3	Desenvolvimento em linguagem <i>assembly</i>	398
5.9.4	Ambientes de desenvolvimento.....	400
5.9.4.1	Computador alvo e hospedeiro.....	400
5.9.4.2	Sistemas embebidos.....	405
5.10	Conclusões	408
5.11	Exercícios	409

6.1	Interligação dos componentes de um computador	414
6.1.1	Barramentos.....	414
6.1.2	Operações de leitura e escrita	417
6.1.3	Descodificação de endereços (de palavra).....	420
6.1.3.1	Seleção de dispositivo a aceder.....	420
6.1.3.2	Implementação do mapa de endereços.....	421
6.1.3.3	Descodificação parcial dos endereços.....	426
6.1.3.4	Descodificação de mapas de endereços irregulares	429
6.1.3.5	Descodificação de endereços programável	431
6.1.4	Descodificação de endereços (de <i>byte</i>).....	434
6.1.5	Impacte do endereçamento de <i>byte</i>	439
6.1.5.1	Organização da memória em <i>bytes</i>	439
6.1.5.2	Endereçamento <i>little-endian</i> e <i>big-endian</i>	442
6.1.5.3	Alinhamento dos acessos	447
6.1.6	Ciclos de acesso à memória/periféricos.....	450
6.1.6.1	Ligação ao barramento de dados.....	450
6.1.6.2	Ciclos de leitura e escrita	452
6.1.6.3	Temporizações no acesso aos dispositivos.....	455
6.1.6.4	Acesso a dispositivos lentos.....	460
6.2	Exceções	463
6.2.1	Princípios básicos	463
6.2.2	Interrupções.....	466
6.2.2.1	Pinos de interrupção.....	466
6.2.2.2	Controlo do atendimento de interrupções	468
6.2.2.3	Comportamento das interrupções.....	470
6.2.2.4	Mecanismo básico de atendimento de interrupções.....	472
6.2.2.5	Programação com interrupções.....	473
6.2.2.6	Controlador de interrupções.....	480
6.2.3	Outras exceções	482
6.2.3.1	Invocação explícita e retorno de uma exceção	482
6.2.3.2	Exceções predefinidas.....	484
6.3	Tipos de periféricos	487
6.3.1	O que é um periférico?	487
6.3.2	Periféricos de memória de massa	487
6.3.3	Periféricos gráficos.....	490
6.3.4	Periféricos de comunicação.....	492
6.3.4.1	Princípios básicos.....	492
6.3.4.2	Comunicação paralela.....	494
6.3.4.3	Comunicação série.....	496
6.4	Arquitetura do sistema de periféricos	510
6.4.1	Barramentos hierárquicos.....	510
6.4.2	Modos de transferência de dados.....	512
6.4.2.1	Transferência por teste (<i>polling</i>).....	512
6.4.2.2	Transferência por interrupções.....	514
6.4.2.3	Transferência por acesso directo à memória (DMA).....	515
6.4.2.4	Transferência por processador de entradas/saídas.....	520

6.5	Exemplos de computadores completos	521
6.5.1	Classes de computadores	521
6.5.2	O PC	523
6.5.2.1	Arquitetura original	523
6.5.2.2	Evolução nos processadores	525
6.5.2.3	Evolução nas memórias	533
6.5.2.4	Evolução nos periféricos	536
6.5.2.5	Evolução nos barramentos	538
6.5.3	O microcontrolador	541
6.5.3.1	Características básicas	541
6.5.3.2	CREPE: um microcontrolador baseado no PEPE	544
6.6	Avaliação de desempenho dos computadores	548
6.6.1	O que é o desempenho	548
6.6.2	Programas de avaliação (<i>benchmarks</i>)	549
6.6.3	A lei de Amdahl	552
6.6.4	Avaliação do desempenho do processador	553
6.6.5	Avaliação do desempenho da memória	555
6.6.6	O impacte do compilador	557
6.6.7	A filosofia RISC	559
6.6.8	Avaliação do desempenho dos periféricos	561
6.7	Conclusões	566
6.8	Exercícios	566

7 - O PROCESSADOR EM DETALHE 571

7.1	Diagrama de blocos geral	572
7.2	Núcleo do processador	574
7.2.1	Caminho de dados	574
7.2.1.1	Funcionamento geral	574
7.2.1.2	Banco de registos	578
7.2.1.3	Gerador de constantes	580
7.2.1.4	Unidade aritmética e lógica (ALU)	581
7.2.2	Unidade de exceções	585
7.2.3	Unidade de controlo	586
7.2.4	Microprogramação	591
7.2.4.1	Circuito simples microprogramado	591
7.2.4.2	Microprogramação no PEPE	594
7.3	Processamento em estágios	601
7.3.1	Princípios de funcionamento	601
7.3.2	Cadeias de estágios	605
7.3.3	Implementação das cadeias de estágios	609
7.3.3.1	Cadeia de estágios de instruções	609
7.3.3.2	Cadeia de estágios de microinstruções	614
7.3.4	Exceções com processamento em estágios	616
7.3.5	Dependências de dados	619
7.3.6	Dependências de controlo	624

7.4	Interface de memória	627
7.5	Caches	628
7.5.1	Princípios de funcionamento das <i>caches</i>	628
7.5.2	Organização das <i>caches</i>	631
7.5.2.1	Princípios da organização	631
7.5.2.2	Mapeamento direto	633
7.5.2.3	Mapeamento associativo	636
7.5.2.4	Mapeamento associativo por conjuntos	639
7.5.3	Políticas de substituição de blocos	641
7.5.4	Políticas de escrita nas <i>caches</i>	642
7.5.5	Evolução do subsistema de <i>caches</i>	644
7.5.6	Casos em que não se quer <i>cache</i>	646
7.5.7	<i>Caches</i> no PEPE	648
7.6	Memória virtual	649
7.6.1	Hierarquia de memórias	649
7.6.2	Princípios de funcionamento da memória virtual	650
7.6.3	Tradução de endereços virtuais para físicos	654
7.6.4	Gestão das páginas	657
7.6.5	A TLB e o seu papel na tradução de endereços	661
7.6.6	Integração da memória virtual e das <i>caches</i>	663
7.6.7	Memória virtual no PEPE	667
7.7	Suporte para processos	671
7.7.1	Modelos de programação e de execução	671
7.7.2	Multiprogramação	672
7.7.3	Interação entre processos	678
7.7.3.1	Sincronização de baixo nível	678
7.7.3.2	Sincronização com semáforos	682
7.7.3.3	Comunicação	685
7.7.4	Programação cooperativa	685
7.7.5	Proteção	689
7.7.6	Gestores de periféricos	692
7.8	Conclusões	694
7.9	Exercícios	695

APÊNDICE A - MANUAL DE PROGRAMADOR DO PEPE 703

A.1	Pinos do módulo PEPE	703
A.2	Registos	704
A.2.1	Registos principais	704
A.2.2	Registos auxiliares	705
A.2.2.1	Configuração do núcleo	706
A.2.2.2	Configuração das <i>caches</i>	707
A.2.2.3	Configuração da memória virtual	708
A.3	Exceções	709
A.4	Conjunto de instruções	709

A.5 Programação do PEPE 716

APÊNDICE B - MANUAL DE PROGRAMADOR DO CREPE 719

B.1 Pinos do módulo CREPE 719

B.2 Registos auxiliares 719

B.3 Funcionamento dos periféricos 722

 B.3.1 Portos de entrada/saída 722

 B.3.2 Temporizadores 723

 B.3.3 UARTs 724

 B.3.4 Informação sobre o estado dos periféricos 726

B.4 Exceções 726

B.5 Exemplo de utilização 729

APÊNDICE C - INTRODUÇÃO AO SIMULADOR (SIMAC) 733

C.1 Desenho de circuitos 733

C.2 Simulação de circuitos 736

APÊNDICE D - COMPUTAÇÃO EM VÍRGULA FLUTUANTE 743

D.1 Representação em vírgula flutuante 743

D.2 A norma IEEE 754 745

D.3 Operações aritméticas em vírgula flutuante 748

APÊNDICE E - CODIFICAÇÃO DE CARACTERES EM ASCII 751

BIBLIOGRAFIA 753

ÍNDICE REMISSIVO 755