

ÍNDICE GERAL

AGRADECIMENTOS	VII
PREFÁCIO	IX
ÍNDICE GERAL	XIII
ÍNDICE DAS SIMULAÇÕES.....	XXIII
ÍNDICE DAS FIGURAS	XXVII
ÍNDICE DAS TABELAS.....	XLI
ÍNDICE DOS PROGRAMAS.....	LI
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 Interacção pessoa-computador	11
1.7 A gestão de um computador.....	14
1.8 A evolução dos computadores	16
1.9 Perspectivas de evolução futura.....	24
1.10 Conclusões.....	26
2 - O MUNDO BINÁRIO	27
2.1 Circuitos electrónicos analógicos	28
2.2 Circuitos electrónicos digitais	29

2.2.1	Funcionamento básico	29
2.2.2	Diagramas temporais	31
2.2.3	Portas lógicas	32
2.3	Álgebra de Boole	36
2.4	Funções lógicas	37
2.5	Circuitos combinatórios.....	44
2.5.1	Síntese de circuitos combinatórios	44
2.5.2	<i>Multiplexers</i>	46
2.5.3	Descodificadores	49
2.5.4	ROMs	52
2.6	Circuitos Sequenciais.....	55
2.6.1	Elementos bi-estáveis	55
2.6.1.1	Trinco SR	55
2.6.1.2	Trinco D	56
2.6.1.3	Báscula D	57
2.6.2	Registos	61
2.6.3	Portas lógicas de três estados (<i>tristate</i>).....	63
2.6.4	Banco de registos.....	64
2.6.5	Contadores.....	66
2.6.6	Registos de deslocamento.....	71
2.6.7	Máquinas de estados.....	72
2.6.7.1	Modelo das máquinas de estados	72
2.6.7.2	Diagramas de estados.....	74
	Microondas simples.....	74
	Semáforo simples	76
	Semáforo com botão para peões.....	77
2.6.7.3	Máquinas de estados sintetizadas.....	81
2.6.7.4	Máquinas de estados microprogramadas	81
2.7	Representação de números.....	85
2.7.1	Números em base 10 (decimais) e 2 (binários).....	85
2.7.2	Números em base 16 (hexadecimais)	87
2.7.3	Potências de 2	89
2.7.4	Quantos <i>bits</i> para representar um número?	91
2.7.5	Representação de números negativos	92
2.7.6	Representação de números em complemento para 2	93
2.7.7	Extensão do número de <i>bits</i> de um número	95
2.8	Operações aritméticas.....	97
2.8.1	Soma de dois números binários	97
2.8.2	Subtração de dois números binários	98
2.8.3	Excesso	99
2.8.4	Multiplicação de dois números binários	101
2.8.5	Divisão de dois números binários.....	103
2.9	Conclusões.....	107
2.10	Exercícios	108

3 - O MEU PRIMEIRO COMPUTADOR	111
3.1 Componentes básicos de um computador	112
3.2 RAM – a memória para guardar informação	113
3.3 O processador (PEPE-8).....	117
3.3.1 Unidade de dados	118
3.3.1.1 Registo na unidade de dados	118
3.3.1.2 Unidade aritmética e lógica (ALU).....	121
3.3.1.3 Funcionamento da unidade de dados	126
3.3.2 Unidade de controlo	128
3.3.2.1 Sinais de controlo.....	128
3.3.2.2 Contador de Programa (PC).....	129
3.3.2.3 Um programa simples	131
3.3.2.4 Constantes no programa.....	133
3.3.2.5 Saltos no programa	138
3.3.2.6 Funcionamento detalhado do programa	140
3.3.3 O processador (PEPE-8) e as memórias	143
3.3.3.1 Processador (PEPE-8).....	143
3.3.3.2 Memória de dados.....	144
3.3.3.3 Memória de instruções	145
3.4 Programação em baixo nível de um computador.....	147
3.4.1 Instruções em vez de sinais de controlo	147
3.4.2 Linguagem <i>assembly</i>	150
3.4.3 Implementação das instruções	153
3.4.4 Programação em linguagem <i>assembly</i>	158
3.4.5 Programação do PEPE-8 em <i>assembly</i> : contagem de <i>bits</i>	161
3.5 Periféricos	163
3.5.1 Estrutura do <i>hardware</i>	163
3.5.2 Programação com periféricos	171
3.5.2.1 Uso de periféricos de saída	172
3.5.2.2 Uso de periféricos de entrada.....	174
3.6 Soluções específicas ou genéricas?	176
3.7 Conclusões.....	177
3.8 Exercícios	178
4 - ARQUITECTURA BÁSICA DE UM PROCESSADOR	181
4.1 Banco de registos	182
4.2 Endereços de dados e de instruções.....	188
4.2.1 Memórias de dados e de instruções separadas: <i>caches</i>	188
4.2.2 Espaço de endereçamento e mapa de endereços.....	190
4.3 Impacte da largura das instruções.....	192

4.4	Endereçamento de <i>byte</i> e de palavra	194
4.5	Codificação das instruções	198
4.6	Registos	201
4.7	<i>Bits</i> de estado	203
4.8	Conjunto de instruções	208
4.9	Instruções de salto	211
4.10	Instruções de transferência de dados	214
4.10.1	Combinações de operandos	214
4.10.2	Transferências entre registos	216
4.10.3	Transferência de uma constante para um registo	217
4.10.4	Transferências entre um registo e a memória	221
4.10.4.1	Endereços constantes e em registos	221
4.10.4.2	Modos de acesso à memória em 16 <i>bits</i>	222
4.10.4.3	Acesso à memória em 16 <i>bits</i> com índice variável	223
4.10.4.4	Acesso à memória em 16 <i>bits</i> sem índice	224
4.10.4.5	Acesso à memória em 16 <i>bits</i> com índice constante	225
4.10.4.6	Instruções de acesso à memória em 16 <i>bits</i>	227
4.10.4.7	Acesso à memória em 8 <i>bits</i>	228
4.10.4.8	Acesso à memória em 8 <i>bits</i> e 16 <i>bits</i>	233
4.10.5	Transferências para memória de uma constante ou memória	235
4.11	Instruções aritméticas	235
4.11.1	Instruções aritméticas mais simples	236
4.11.1.1	Soma e excesso: série de Fibonacci	237
4.11.1.2	Soma e transporte: números grandes	238
4.11.2	Multiplicação e divisão	240
4.12	Instruções Lógicas	244
4.12.1	Funcionalidade das instruções lógicas	244
4.12.2	Expressões booleanas	247
4.12.3	Instruções de manipulação de um só <i>bit</i>	248
4.12.4	Operações lógicas com máscaras	252
4.12.4.1	Funcionamento das máscaras	252
4.12.4.2	Máscaras AND	253
4.12.4.3	Máscaras OR	255
4.12.4.4	Máscaras XOR	257
4.13	Instruções de deslocamento	259
4.13.1	Instruções de deslocamento linear	260
4.13.2	Instruções de deslocamento circular (rotações)	264
4.14	Modos de endereçamento	266
4.15	Conclusões	269
4.16	Exercícios	270

5 - PROGRAMAÇÃO DE UM COMPUTADOR	273
5.1 Um problema simples	274
5.1.1 Modo de actuação de um ser humano.....	274
5.1.2 Modo de actuação de um computador.....	274
5.2 Modelação do problema com fluxogramas	275
5.3 Programação em alto nível.....	278
5.4 Mapeamento da programação de alto nível em linguagem <i>assembly</i>	281
5.5 Dados, declarações e directivas	283
5.5.1 Constantes simbólicas e a directiva EQU	283
5.5.2 Variáveis.....	284
5.5.2.1 Tipos das variáveis.....	284
5.5.2.2 Acesso a variáveis de tipos de dados estruturados	285
5.5.2.3 Directivas WORD, TABLE e STRING	286
5.5.3 A directiva PLACE.....	289
5.5.4 Apontadores.....	292
5.6 Instruções	296
5.6.1 Atribuição e expressões	298
5.6.2 Decisão	299
5.6.2.1 Decisão simples	299
5.6.2.2 Decisão múltipla	300
5.6.3 Iteração	302
5.7 Rotinas.....	304
5.7.1 Estruturação do código	304
5.7.1.1 Funções nas linguagens de alto nível.....	304
5.7.1.2 Rotinas em linguagem <i>assembly</i>	308
5.7.1.3 Variante com apontadores.....	312
5.7.2 Mecanismo de chamada e retorno	314
5.7.2.1 Endereço de retorno	314
5.7.2.2 Chamada de rotinas com endereço de retorno em registo.....	315
5.7.2.3 Chamada de rotinas com endereço de retorno na memória (pilha)	319
5.7.2.4 Qual dos mecanismos de chamada de rotinas se deve usar?	329
5.7.2.5 Variantes do funcionamento da pilha.....	331
5.7.3 Outras utilizações da pilha em rotinas	332
5.7.3.1 Guarda de registos nas rotinas	332
5.7.3.2 Variáveis locais.....	340
5.7.3.3 Passagem de parâmetros e do resultado	342
5.7.3.4 Contextos de chamada das rotinas	349
5.7.3.5 Recursividade.....	354
5.8 Gestão dos dados	358
5.8.1 Quando os registos não chegam	358
5.8.2 Cálculo de expressões.....	359
5.8.3 Execução de instruções imbricadas	360
5.8.4 Tabelas	361
5.8.4.1 Tabelas de uma só dimensão.....	361

5.8.4.2	Tabelas multidimensionais.....	366
5.8.4.3	Tabelas de apontadores.....	372
5.8.5	Estruturas de dados dinâmicas (montão).....	376
5.8.6	Listas ligadas.....	380
5.9	Desenvolvimento de programas.....	387
5.9.1	Ciclo de desenvolvimento.....	387
5.9.2	Programação em alto nível ou em linguagem <i>assembly</i> ?.....	392
5.9.3	Desenvolvimento em linguagem <i>assembly</i>	392
5.9.4	Ambientes de desenvolvimento.....	394
5.9.4.1	Computador alvo e hospedeiro.....	394
5.9.4.2	Sistemas embebidos.....	399
5.10	Conclusões.....	402
5.11	Exercícios.....	403
6	O COMPUTADOR COMPLETO.....	407
6.1	Interligação dos componentes de um computador.....	408
6.1.1	Barramentos.....	408
6.1.2	Operações de leitura e escrita.....	411
6.1.3	Descodificação de endereços (de palavra).....	414
6.1.3.1	Seleção de dispositivo a aceder.....	414
6.1.3.2	Implementação do mapa de endereços.....	415
6.1.3.3	Descodificação parcial dos endereços.....	420
6.1.3.4	Descodificação de mapas de endereços irregulares.....	423
6.1.3.5	Descodificação de endereços programável.....	425
6.1.4	Descodificação de endereços (de <i>byte</i>).....	428
6.1.5	Impacte do endereçamento de <i>byte</i>	433
6.1.5.1	Organização da memória em <i>bytes</i>	433
6.1.5.2	Endereçamento <i>little-endian</i> e <i>big-endian</i>	436
6.1.5.3	Alinhamento dos acessos.....	441
6.1.6	Ciclos de acesso à memória/periféricos.....	444
6.1.6.1	Ligação ao barramento de dados.....	444
6.1.6.2	Ciclos de leitura e escrita.....	446
6.1.6.3	Temporizações no acesso aos dispositivos.....	449
6.1.6.4	Acesso a dispositivos lentos.....	454
6.2	Excepções.....	457
6.2.1	Princípios básicos.....	457
6.2.2	Interrupções.....	460
6.2.2.1	Pinos de interrupção.....	460
6.2.2.2	Controlo do atendimento de interrupções.....	462
6.2.2.3	Comportamento das interrupções.....	464
6.2.2.4	Mecanismo básico de atendimento de interrupções.....	466
6.2.2.5	Programação com interrupções.....	467
6.2.2.6	Controlador de interrupções.....	474
6.2.3	Outras excepções.....	476
6.2.3.1	Invocação explícita e retorno de uma excepção.....	476

6.2.3.2	Excepções predefinidas.....	478
6.3	Tipos de periféricos	480
6.3.1	O que é um periférico?	480
6.3.2	Periféricos de memória de massa	481
6.3.3	Periféricos gráficos	484
6.3.4	Periféricos de comunicação	485
6.3.4.1	Princípios básicos.....	485
6.3.4.2	Comunicação paralela.....	488
6.3.4.3	Comunicação série	490
	Comunicação série assíncrona com a norma RS-232C	491
	Comunicação série síncrona com o barramento USB	494
6.4	Arquitectura do sistema de periféricos	504
6.4.1	Barramentos hierárquicos	504
6.4.2	Modos de transferência de dados.....	505
6.4.2.1	Transferência por teste (<i>polling</i>).....	505
6.4.2.2	Transferência por interrupções.....	507
6.4.2.3	Transferência por acesso directo à memória (DMA)	509
6.4.2.4	Transferência por processador de entradas/saídas	513
6.5	Exemplos de computadores completos.....	514
6.5.1	Classes de computadores	514
6.5.2	O PC	516
6.5.2.1	Arquitectura original	516
6.5.2.2	Evolução nos processadores.....	518
6.5.2.3	Evolução nas memórias	524
6.5.2.4	Evolução nos periféricos.....	526
6.5.2.5	Evolução nos barramentos	528
6.5.3	O microcontrolador	531
6.5.3.1	Características básicas	531
6.5.3.2	CREPE: um microcontrolador baseado no PEPE	533
6.6	Avaliação de desempenho dos computadores.....	538
6.6.1	O que é o desempenho	538
6.6.2	Programas de avaliação (<i>benchmarks</i>)	539
6.6.3	A lei de Amdahl	541
6.6.4	Avaliação do desempenho do processador	543
6.6.5	Avaliação do desempenho da memória	545
6.6.6	O impacte do compilador	547
6.6.7	A filosofia RISC	548
6.6.8	Avaliação do desempenho dos periféricos	551
6.7	Conclusões.....	556
6.8	Exercícios	556
7	O PROCESSADOR EM DETALHE	561
7.1	Diagrama de blocos geral	562

7.2	Núcleo do processador	564
7.2.1	Caminho de dados	564
7.2.1.1	Funcionamento geral	564
7.2.1.2	Banco de registos	568
7.2.1.3	Gerador de constantes	570
7.2.1.4	Unidade aritmética e lógica (ALU)	571
7.2.2	Unidade de excepções	575
7.2.3	Unidade de controlo	576
7.2.4	Microprogramação	581
7.2.4.1	Circuito simples microprogramado	581
7.2.4.2	Microprogramação no PEPE	584
7.3	Processamento em estágios	591
7.3.1	Princípios de funcionamento	591
7.3.2	Cadeias de estágios	595
7.3.3	Implementação das cadeias de estágios	599
7.3.3.1	Cadeia de estágios de instruções	599
7.3.3.2	Cadeia de estágios de microinstruções	604
7.3.4	Excepções com processamento em estágios	606
7.3.5	Dependências de dados	608
7.3.6	Dependências de controlo	613
7.4	Interface de memória	615
7.5	Caches	616
7.5.1	Princípios de funcionamento das <i>caches</i>	616
7.5.2	Organização das <i>caches</i>	619
7.5.2.1	Princípios da organização	619
7.5.2.2	Mapeamento directo	621
7.5.2.3	Mapeamento associativo	624
7.5.2.4	Mapeamento associativo por conjuntos	627
7.5.3	Políticas de substituição de blocos	629
7.5.4	Políticas de escrita nas <i>caches</i>	630
7.5.5	Evolução do subsistema de <i>caches</i>	632
7.5.6	Casos em que não se quer <i>cache</i>	634
7.5.7	<i>Caches</i> no PEPE	636
7.6	Memória virtual	637
7.6.1	Hierarquia de memórias	637
7.6.2	Princípios de funcionamento da memória virtual	638
7.6.3	Tradução de endereços virtuais para físicos	641
7.6.4	Gestão das páginas	645
7.6.5	A TLB e o seu papel na tradução de endereços	648
7.6.6	Integração da memória virtual e das <i>caches</i>	651
7.6.7	Memória virtual no PEPE	654
7.7	Suporte para Processos	658
7.7.1	Modelos de programação e de execução	658
7.7.2	Multiprogramação	659
7.7.3	Interação entre processos	665
7.7.3.1	Sincronização de baixo nível	665

7.7.3.2	Sincronização com semáforos.....	669
7.7.3.3	Comunicação	672
7.7.4	Programação cooperativa	673
7.7.5	Protecção	676
7.7.6	Gestores de periféricos	679
7.8	Conclusões.....	681
7.9	Exercícios	683
 APÊNDICE A MANUAL DE PROGRAMADOR DO PEPE 689		
A.1	Pinos do módulo PEPE	689
A.2	Registos	690
A.2.1	Registos principais	690
A.2.2	Registos auxiliares.....	691
A.2.2.1	Configuração do núcleo	692
A.2.2.2	Configuração das <i>caches</i>	693
A.2.2.3	Configuração da memória virtual	694
A.3	Excepções	695
A.4	Conjunto de instruções	695
A.5	Programação do PEPE	702
 APÊNDICE B MANUAL DE PROGRAMADOR DO CREPE 705		
B.1	Pinos do módulo CREPE.....	705
B.2	Registos auxiliares.....	705
B.3	Funcionamento dos Periféricos.....	708
B.3.1	Portos de entrada/saída	708
B.3.2	Temporizadores	709
B.3.3	UARTs	710
B.3.4	Informação sobre o estado dos periféricos	712
B.4	Excepções	712
B.5	Exemplo de utilização	714
 APÊNDICE C INTRODUÇÃO AO SIMULADOR (SIMAC)..... 719		
C.1	Desenho de circuitos.....	719
C.2	Simulação de circuitos	722
 APÊNDICE D COMPUTAÇÃO EM VÍRGULA FLUTUANTE 727		
D.1	Representação em vírgula flutuante.....	727

D.2	A norma IEEE 754.....	729
D.3	Operações aritméticas em vírgula flutuante.....	732
APÊNDICE E CODIFICAÇÃO DE CARACTERES EM ASCII.....		735
BIBLIOGRAFIA		737
ÍNDICE REMISSIVO		739