

Baruch Zoltan Francisc

**STRUCTURA SISTEMELOR DE CALCUL
CU APLICAȚII**

STRUCTURA SISTEMELOR DE CALCUL CU APLICAȚII

Dr. ing. Baruch Zoltan Francisc

Universitatea Tehnică din Cluj-Napoca

Catedra de Calculatoare

Editura TODESCO

Cluj-Napoca, 2001

Recenzenți: Prof. dr. ing. KALMAN PUSZTAI
Conf. dr. ing. EUGEN LUPU

CUPRINS

PREFAȚĂ	xi
1. CALCULATORUL DIDACTIC CD	1
1.1. PREZENTAREA GENERALĂ A CALCULATORULUI DIDACTIC	1
1.2. INSTRUCȚIUNILE CALCULATORULUI DIDACTIC	2
1.2.1. Limbajul de descriere utilizat	2
1.2.2. Structura cuvântului	5
1.2.3. Setul de instrucțiuni	6
1.2.4. Exemple de programe	7
1.3. STRUCTURA CALCULATORULUI DIDACTIC	9
1.3.1. Structura generală	9
1.3.2. Semnalele de comandă	10
1.3.3. Structura detaliată	11
1.4. EXECUȚIA INSTRUCȚIUNILOR	13
1.5. APLICAȚII	16
2. DISPOZITIVE DE ÎNMULȚIRE ȘI ÎMPĂRȚIRE BINARĂ	17
2.1. ÎNMULȚIREA BINARĂ	17
2.1.1. Metode de înmulțire binară	17
2.1.2. Înmulțirea prin metoda directă	18
2.1.3. Înmulțirea prin metoda Booth	23
2.1.4. Înmulțirea pe grupe de cifre	27
2.2. ÎMPĂRȚIREA BINARĂ	30
2.2.1. Principiul împărțirii binare	30
2.2.2. Metode de împărțire binară	32
2.2.3. Metoda refacerii restului parțial	32
2.2.4. Metoda fără refacerea restului parțial	36
2.3. APLICAȚII	39
3. DISPOZITIVE DE ÎNMULȚIRE ȘI ÎMPĂRȚIRE ZECIMALĂ	40
3.1. ÎNMULȚIREA ZECIMALĂ	40
3.1.1. Metoda adunării repetate	40

3.1.2. Metoda celor nouă multipli ai deînmulțitului.....	41
3.1.3. Metoda componentelor din dreapta și din stânga	43
3.2. ÎMPĂRȚIREA ZECIMALĂ	46
3.2.1. Metoda refacerii restului parțial.....	47
3.2.2. Metoda fără refacerea restului parțial	48
3.2.3. Metoda celor nouă multipli ai împărțitorului.....	50
3.3. APLICAȚII.....	52
4. CALCULATOARE MICROPROGRAMATE.....	53
4.1 PRINCIPIUL CALCULATOARELOR MICROPROGRAMATE	53
4.1.1. Introducere.....	53
4.1.2. Microprogramarea pe orizontală și pe verticală.....	54
4.1.3. Structura unei unități de comandă microprogramate	55
4.2. EXEMPLU DE CALCULATOR MICROPROGRAMAT PE ORIZONTALĂ.....	57
4.2.1. Structura căii de date	57
4.2.2. Formatul microinstrucțiunilor.....	59
4.2.3. Secvențierea microinstrucțiunilor	61
4.2.4. Limbajul de microasamblare.....	63
4.3. EXEMPLU DE CALCULATOR MICROPROGRAMAT PE VERTICALĂ.....	64
4.3.1. Microinstrucțiunile verticale.....	64
4.3.2. Structura secțiunii de control	65
4.4. APLICAȚII.....	67
5. SISTEMUL DE PROIECTARE ACTIVE-HDL	69
5.1. PREZENTARE GENERALĂ A SISTEMULUI ACTIVE-HDL	69
5.2. COMPONENTELE SISTEMULUI ACTIVE-HDL.....	70
5.2.1. Editorul pentru codul sursă.....	70
5.2.2. Editorul pentru scheme bloc	71
5.2.3. Editorul pentru diagrame de stare	72
5.2.4. Compilatorul.....	73
5.2.5. Simulatorul	74
5.2.5.1. Etapele simulării	74
5.2.5.2. Definierea stimulilor	74
5.2.5.3. Execuția simulării	76
5.2.6. Editorul pentru forme de undă	76
5.2.7. Componenta <i>Design Explorer</i>	77
5.2.8. Componenta <i>Design Browser</i>	77
5.2.9. Componenta <i>Library Manager</i>	77
5.2.9.1. Structura bibliotecilor.....	78
5.2.9.2. Fereastra <i>Library Manager</i>	78
5.2.9.3. Operații cu biblioteci.....	79
5.2.10. Fereastra <i>Console</i>	79
5.2.11. Fereastra <i>Processes</i>	80
5.2.12. Fereastra <i>Watch</i>	80
5.2.13. Fereastra <i>Call Stack</i>	81
5.2.14. Fereastra <i>Dataflow</i>	81
5.2.15. Fereastra <i>List</i>	82

5.3. EXEMPLU DE PROIECTARE	82
5.3.1. Descrierea proiectului	82
5.3.2. Crearea unui nou proiect.....	83
5.3.2.1. Introducerea numelui proiectului	83
5.3.2.2. Crearea unui fișier sursă.....	83
5.3.2.3. Adăugarea porturilor	84
5.3.3. Editarea și compilarea codului sursă.....	84
5.3.3.1. Descrierea arhitecturii numărătorului	84
5.3.3.2. Specificarea bibliotecii pentru compilare	85
5.3.3.3. Compilarea	85
5.3.4. Vizualizarea structurii proiectului.....	85
5.3.5. Simularea	86
5.3.5.1. Inițializarea simulării	86
5.3.5.2. Asignarea stimulatoarelor.....	86
5.3.5.3. Execuția simulării	87
5.3.5.4. Editarea formelor de undă	87
5.3.5.5. Trasarea codului sursă.....	87
5.3.5.6. Setarea punctelor de întrerupere	88
5.3.5.7. Validarea punctelor de întrerupere	88
5.3.5.8. Vizualizarea rezultatelor simulării sub formă textuală	88
5.3.6. Adăugarea unui nou fișier la proiect.....	88
5.3.7. Utilizarea editorului pentru automate de stare	89
5.3.7.1. Editarea porturilor automatului de stare	89
5.3.7.2. Plasarea simbolurilor stărilor.....	89
5.3.7.3. Editarea simbolului unei stări.....	90
5.3.7.4. Adăugarea tranzițiilor.....	91
5.3.7.5. Editarea tranzițiilor	91
5.3.7.6. Adăugarea condițiilor pentru tranziții	92
5.3.7.7. Adăugarea acțiunilor pentru generarea ieșirilor	93
5.3.7.8. Specificarea proprietăților automatului	93
5.3.7.9. Generarea codului sursă	95
5.3.8. Crearea unui fișier de nivel ierarhic superior.....	95
5.3.9. Compilarea întregului proiect	95
5.3.10. Simularea proiectului.....	96
5.4. APLICAȚII.....	96
6. ELEMENTE DE BAZĂ ALE LIMBAJULUI VHDL	97
6.1. UNITĂȚI DE PROIECTARE	97
6.2. ENTITĂȚI	98
6.2.1. Nume de entități: identificatori	98
6.2.2. Declararea entităților	98
6.2.3. Semnale și porturi	100
6.2.3.1. Conceptul de semnal	100
6.2.3.2. Declararea porturilor	101
6.2.3.3. Modul porturilor.....	102
6.2.3.4. Tipul porturilor.....	103
6.2.4. Generice.....	103
6.3. ARHITECTURI	104
6.3.1. Definierea arhitecturilor	105
6.3.2. Declararea semnalelor	105
6.3.3. Asignarea valorilor la semnale.....	106

6.3.4. Stiluri de descrieri arhitecturale	108
6.3.4.1. Descrieri funcționale	108
6.3.4.2. Descrieri de tipul fluxului de date	109
6.3.4.3. Descrieri structurale	110
6.3.4.4. Compararea descrierilor arhitecturale	112
6.4. MODELAREA PENTRU SIMULARE	114
6.4.1. Simularea bazată pe evenimente	114
6.4.2. Drive de semnal	115
6.4.3. Ciclul de simulare	118
6.4.4. Drive multiple și funcții de rezoluție	121
6.5. MODELAREA PENTRU SINTEZĂ	123
6.6. APLICAȚII	124
7. TIPURI, ATRIBUTE ȘI OPERATORI ÎN LIMBAJUL VHDL	127
7.1. OBIECTE DE DATE	127
7.1.1. Constante	127
7.1.2. Variabile	128
7.1.3. Fișiere	128
7.2. TIPURI DE DATE	129
7.2.1. Prezentare generală a tipurilor de date	129
7.2.1.1. Clasificarea tipurilor de date	129
7.2.1.2. Tipuri care se pot utiliza pentru sinteza logică	130
7.2.1.3. Tipuri standard	130
7.2.1.4. Operatori standard	131
7.2.2. Tipuri scalare	132
7.2.2.1. Tipuri enumerate	132
7.2.2.2. Tipuri întregi	141
7.2.2.3. Tipuri flotante	144
7.2.2.4. Tipuri fizice	145
7.2.3. Tipuri compuse	146
7.2.3.1. Tablouri	146
7.2.3.2. Înregistrări	152
7.3. ATRIBUTE	154
7.3.1. Atribute pentru tipuri	154
7.3.2. Atribute pentru tablouri	156
7.3.3. Atribute pentru semnale	158
7.4. OPERATORI	160
7.4.1. Setul de operatori standard și precedența acestora	160
7.4.2. Operatori logici	161
7.4.3. Operatori relaționali	163
Interpretarea sintezei operatorilor relaționali pentru tipurile întregi și enumerate	163
Interpretarea sintezei operatorilor relaționali pentru tablouri	164
7.4.4. Operatori de deplasare	166
7.4.5. Operatori aritmetici	167
Interpretarea sintezei operatorilor aritmetici	168
7.4.6. Operatorul de concatenare	173
7.5. APLICAȚII	174

8. INSTRUCȚIUNI SECVENȚIALE ȘI CONCURENTE ÎN LIMBAJUL VHDL.....	175
8.1. INSTRUCȚIUNI SECVENȚIALE	175
8.1.1. Procese.....	176
8.1.1.1. Structura și execuția unui proces.....	176
8.1.1.2. Procese cu liste de sensibilitate incomplete.....	177
8.1.1.3. Instrucțiunea wait	178
8.1.1.4. Procese combinaționale și procese secvențiale.....	180
8.1.2. Instrucțiunea secvențială de asignare a semnalelor.....	181
8.1.2.1. Execuția instrucțiunii secvențiale de asignare	181
8.1.2.2. Reacții inverse.....	182
8.1.2.3. Întârzierea inerțială.....	183
8.1.2.4. Întârzierea de transport.....	184
8.1.3. Variabile.....	184
8.1.3.1. Declararea și inițializarea variabilelor.....	185
8.1.3.2. Instrucțiunea de asignare a variabilelor	185
8.1.4. Instrucțiunea if.....	186
8.1.4.1. Sintaxa și execuția instrucțiunii if	186
8.1.4.2. Interpretarea sintezei instrucțiunii if	187
8.1.4.3. Instrucțiuni if incomplete.....	189
8.1.4.4. Instrucțiuni if în care apar variabile	192
8.1.5. Instrucțiunea case.....	193
8.1.6. Instrucțiuni de buclare	195
8.1.6.1. Instrucțiunea loop	196
8.1.6.2. Instrucțiunea while loop.....	196
8.1.6.3. Instrucțiunea for loop.....	197
8.1.6.4. Instrucțiunea next	200
8.1.6.5. Instrucțiunea exit	202
8.1.7. Instrucțiunea secvențială assert.....	203
8.2. INSTRUCȚIUNI CONCURENTE	205
8.2.1. Structura și execuția unei arhitecturi.....	205
8.2.2. Procese.....	207
8.2.3. Instrucțiuni concurente de asignare a semnalelor	207
8.2.3.1. Instrucțiunea de asignare simplă	207
8.2.3.2. Instrucțiunea de asignare condițională	208
8.2.3.3. Instrucțiunea de asignare selectivă	211
8.2.3.4. Instrucțiunea block.....	212
8.2.3.5. Instrucțiunea concurentă assert	214
8.3. DESCRIEREA UNOR CIRCUITE COMBINAȚIONALE	215
8.3.1. Multiplexoare	215
8.3.2. Codificatoare prioritare.....	216
8.4. DESCRIEREA UNOR CIRCUITE SECVENȚIALE	218
8.4.1. Circuite secvențiale sincrone și asincrone	218
8.4.2. Bistabile.....	218
8.4.3. Registre.....	220
8.4.4. Numărătoare	220
8.4.5. Resetarea componentelor sincrone	222
8.4.6. Buffere cu trei stări și semnale bidirecționale.....	225
8.5. APLICAȚII.....	227

9. AUTOMATE DE STARE ÎN LIMBAJUL VHDL.....	231
9.1. EXEMPLU DE PROIECTARE	231
9.1.1. Proiectarea tradițională	231
9.1.2. Proiectarea utilizând limbajul VHDL	233
9.2. PROIECTAREA UNUI CONTROLER DE MEMORIE	235
9.3. TEHNICI PENTRU GENERAREA SEMNALELOR DE IEȘIRE.....	241
9.3.1. Ieșiri decodificate din biții de stare	241
9.3.2. Ieșiri decodificate în registre paralele de ieșire.....	241
9.3.3. Ieșiri codificate în cadrul biților de stare	244
9.3.4. Codificarea cu un bistabil pe stare	248
9.4. AUTOMATE DE STARE DE TIP MEALY	251
9.5. ALTE CONSIDERAȚII DE PROIECTARE	252
9.5.1. Codificarea stărilor utilizând tipuri enumerate.....	252
9.5.2. Codificarea explicită a stărilor	253
9.5.3. Toleranța la defecte a automatelor codificate cu un bistabil pe stare	254
9.6. APLICAȚII	254
10. PROIECTAREA STRUCTURALĂ ÎN LIMBAJUL VHDL.....	257
10.1. AVANTAJELE PROIECTĂRII STRUCTURALE	257
10.2. ELEMENTELE UNEI DESCRIERI STRUCTURALE.....	258
10.2.1. Exemplu de descriere structurală	258
10.2.2. Declarația componentelor	260
10.2.3. Instanțierea componentelor	260
10.2.4. Instanțierea directă a entităților	261
10.2.5. Specificarea configurației	262
10.3. BIBLIOTECI.....	265
10.4. PACHETE	266
10.4.1. Declarația pachetelor	266
10.4.2. Corpul pachetelor	267
10.4.3. Pachete cu declarații de componente	267
10.5. GENERICHE ȘI COMPONENTE PARAMETRIZATE	268
10.5.1. Principiul genericelor.....	268
10.5.2. Definierea entităților generice	269
10.5.3. Utilizarea componentelor generice	270
10.5.4. Tipul parametrilor generici	271
10.5.5. Construirea unei biblioteci de componente.....	272
10.6. INSTRUCȚIUNEA GENERATE	275
10.6.1. Instrucțiunea <code>for generate</code>	275
10.6.2. Instrucțiunea <code>if generate</code>	277
10.6.3. Instanțierea componentelor în instrucțiunile generate.....	279
10.7. APLICAȚII	280

11. SUBPROGRAME ÎN LIMBAJUL VHDL	282
11.1. ROLUL SUBPROGRAMELOR	282
11.2. FUNCȚII	282
11.2.1. Definirea funcțiilor	283
11.2.2. Utilizarea funcțiilor	285
11.2.3. Valori inițiale	286
11.2.4. Parametri cu domeniul nelimitat	287
11.2.5. Valori returnate cu domeniul nelimitat	289
11.2.6. Instrucțiuni <code>return</code> multiple	290
11.2.7. Redefinirea funcțiilor	292
11.2.8. Funcții pentru conversii de tip	292
11.2.8.1. Conversii de tip predefinite	292
11.2.8.2. Conversii de tip definite de utilizator	293
11.2.9. Funcții ca alternativă la instanțierea componentelor	297
11.2.10. Funcții problematice pentru sinteză	297
11.2.11. Definirea operatorilor ca și funcții	298
11.2.11.1. Operatori predefiniți	299
11.2.11.2. Redefinirea operatorilor	299
11.2.12. Funcții standard	301
11.2.13. Plasarea definiției funcțiilor	303
11.3. PROCEDURI	304
11.3.1. Definirea procedurilor	304
11.3.2. Apelul procedurilor	306
11.3.3. Parametri cu domeniul nelimitat	307
11.3.4. Parametri cu modul <code>inout</code>	308
11.3.5. Semnale utilizate ca parametri	308
11.3.6. Redefinirea procedurilor	310
11.4. APLICAȚII	311
12. CALCULATORUL DLX	313
12.1. ARHITECTURA CALCULATORULUI DLX	313
12.1.1. Setul de registre	313
12.1.2. Structura căii de date <i>pipeline</i> și execuția instrucțiunilor	314
12.1.3. Hazarduri	316
12.2. LIMBAJUL DE ASAMBLARE DLX	317
12.2.1. Expresii	317
12.2.2. Directive	317
12.2.3. Setul de instrucțiuni	318
12.2.3.1. Instrucțiuni pentru transferul datelor	318
12.2.3.2. Instrucțiuni aritmetice și logice	320
12.2.3.3. Instrucțiuni de control	321
12.2.3.4. Instrucțiuni pentru operații în virgulă mobilă	322
12.3. SIMULATORUL WINDLX	323
12.3.1. Prezentare generală	323
12.3.2. Apelul funcțiilor de sistem	324
12.3.2.1. Deschiderea unui fișier	325
12.3.2.2. Închiderea unui fișier	326
12.3.2.3. Citirea unui bloc dintr-un fișier	326
12.3.2.4. Scrierea unui bloc într-un fișier	327

12.3.2.5. Scrierea la dispozitivul standard de ieșire	327
12.4. UTILIZAREA SIMULATORULUI WINDLX.....	328
12.4.1. Lansarea și configurarea simulatorului	328
12.4.2. Încărcarea programelor de test.....	329
12.4.3. Simularea	329
12.4.3.1. Fereastra căii de date <i>Pipeline</i>	329
12.4.3.2. Fereastra de cod	329
12.4.3.3. Fereastra diagramei ciclurilor de ceas	330
12.4.3.4. Fereastra punctelor de întrerupere	330
12.4.3.5. Fereastra registrelor.....	331
12.4.3.6. Fereastra de statistici	332
12.5. APLICAȚII.....	332
BIBLIOGRAFIE	335