

Scilab Textbook Companion for
Introductory Circuit Analysis
by Robert L. Boylestad¹

Created by
Zaid Siddique
BE

Electronics Engineering
AIKTC, Mumbai University
College Teacher
None

Cross-Checked by
None

October 4, 2018

¹Funded by a grant from the National Mission on Education through ICT, <http://spoken-tutorial.org/NMEICT-Intro>. This Textbook Companion and Scilab codes written in it can be downloaded from the "Textbook Companion Project" section at the website <http://scilab.in>

Book Description

Title: Introductory Circuit Analysis

Author: Robert L. Boylestad

Publisher: Prentice Hall, Upper Saddle River, N.J

Edition: 10

Year: 2003

ISBN: 9780132359049

Scilab numbering policy used in this document and the relation to the above book.

Exa Example (Solved example)

Eqn Equation (Particular equation of the above book)

AP Appendix to Example(Scilab Code that is an Appednix to a particular Example of the above book)

For example, Exa 3.51 means solved example 3.51 of this book. Sec 2.3 means a scilab code whose theory is explained in Section 2.3 of the book.

Contents

List of Scilab Codes	4
1 Introduction	5
2 Current and Voltage	17
3 Resistance	20
4 Ohms Law Power and Energy	27
5 Series Circuits	37
6 Parallel Circuits	51
7 Series Parallel Networks	67
8 Methods of Analysis and Selected Topics dc	79
9 Network Theorems	87
10 Capacitors	101
11 Magnetic Circuits	120
12 Inductors	127

13 Sinusoidal Alternating Waveforms	137
14 The Basic Elements and Phasors	150
15 Series and Parallel ac Circuits	174
16 Series Parallel ac Networks	204
17 Methods of Analysis and Selected Topics ac	220
18 Network Theorems ac	239
19 Power ac	260
20 Resonance	270
21 Transformers	282
22 Polyphase Systems	290
23 Decibels Filters and Bode Plots	308
24 Pulse Waveforms and the RC Response	325
25 Nonsinusoidal Circuits	338
26 System Analysis An Introduction	346

List of Scilab Codes

Exa 1.1	Example 1	5
Exa 1.2	Example 2	7
Exa 1.3	Example 3	7
Exa 1.4	Example 4	8
Exa 1.5	Example 5	9
Exa 1.6	Example 6	9
Exa 1.7	Example 7	10
Exa 1.8	Example 8	11
Exa 1.9	Example 9	11
Exa 1.10	Example 10	12
Exa 1.11	Example 11	13
Exa 1.12	Example 12	13
Exa 1.13	Example 13	14
Exa 1.14	Example 14	14
Exa 1.15	Example 15	15
Exa 1.16	Example 16	16
Exa 2.1	Example 1	17
Exa 2.2	Example 2	17
Exa 2.3	Example 3	18
Exa 2.4	Example 4	18
Exa 2.5	Example 5	19
Exa 3.1	Example 1	20
Exa 3.2	Example 3	20
Exa 3.3	Example 3	21
Exa 3.4	Example 4	21
Exa 3.5	Example 5	22
Exa 3.6	Example 6	22
Exa 3.7	Example 7	23

Exa 3.8	Example 8	23
Exa 3.9	Example 9	24
Exa 3.10	Example 10	24
Exa 3.11	Example 11	25
Exa 3.12	Example 12	25
Exa 3.14	Example 14	26
Exa 4.1	Example 4	27
Exa 4.2	Example 2	27
Exa 4.3	Example 3	28
Exa 4.4	Example 4	28
Exa 4.5	Example 5	29
Exa 4.6	Example 6	29
Exa 4.7	Example 7	30
Exa 4.8	Example 8	30
Exa 4.9	Example 9	31
Exa 4.10	Example 10	31
Exa 4.11	Example 11	32
Exa 4.12	Example 12	32
Exa 4.13	Example 13	33
Exa 4.14	Example 14	33
Exa 4.15	Example 15	33
Exa 4.16	Example 16	34
Exa 4.17	Example 17	34
Exa 4.18	Example 18	35
Exa 4.19	Example 19	35
Exa 5.1	Example 1	37
Exa 5.2	Example 2	38
Exa 5.3	Example 3	38
Exa 5.4	Example 4	39
Exa 5.5	Example 5	40
Exa 5.6	Example 6	40
Exa 5.7	Example 7	41
Exa 5.8	Example 8	41
Exa 5.9	Exempl 9	42
Exa 5.10	Example 10	43
Exa 5.11	Example 11	43
Exa 5.12	Example 12	44
Exa 5.13	Example 13	44

Exa 5.14	Example 14	45
Exa 5.15	Example 15	45
Exa 5.16	Example 16	46
Exa 5.17	Example 17	46
Exa 5.18	Example 18	47
Exa 5.19	Example 19	47
Exa 5.20	Example 20	48
Exa 5.21	Example 21	48
Exa 5.22	Example 22	49
Exa 5.23	Example 23	50
Exa 5.24	Example 24	50
Exa 6.1	Example 1	51
Exa 6.2	Example 2	51
Exa 6.3	Example 3	52
Exa 6.4	Example 4	52
Exa 6.5	Example 5	53
Exa 6.6	Example 6	53
Exa 6.7	Example 7	54
Exa 6.8	Example 8	55
Exa 6.9	Example 9	55
Exa 6.10	Example 10	56
Exa 6.11	Example 11	57
Exa 6.12	Example 12	58
Exa 6.13	Example 13	59
Exa 6.14	Example 14	59
Exa 6.15	Example 15	60
Exa 6.16	Example 16	60
Exa 6.17	Example 17	61
Exa 6.18	Example 18	61
Exa 6.19	Example 19	62
Exa 6.20	Example 20	63
Exa 6.21	Example 21	63
Exa 6.22	Example 22	64
Exa 6.23	Example 23	64
Exa 6.24	Example 24	65
Exa 6.25	Example 25	65
Exa 6.26	Example 26	66
Exa 7.1	Example 1	67

Exa 7.2	Example 2	68
Exa 7.3	Example 3	69
Exa 7.4	Example 4	70
Exa 7.5	Example 5	71
Exa 7.6	Example 6	72
Exa 7.7	Example 7	73
Exa 7.8	Example 8	74
Exa 7.9	Example 9	75
Exa 7.10	Example 10	76
Exa 7.11	Example 11	76
Exa 7.12	Example 12	77
Exa 8.1	Example 1	79
Exa 8.2	Example 2	79
Exa 8.3	Example 3	80
Exa 8.4	Example 4	81
Exa 8.5	Example 5	82
Exa 8.6	Example 6	82
Exa 8.7	Example 7	83
Exa 8.8	Example 8	84
Exa 8.9	Example 9	85
Exa 9.1	Example 1	87
Exa 9.2	Example 2	87
Exa 9.3	Example 3	88
Exa 9.4	Example 4	89
Exa 9.5	Example 5	90
Exa 9.6	Example 6	90
Exa 9.7	Example 7	91
Exa 9.8	Example 8	92
Exa 9.9	Example 9	92
Exa 9.10	Example 10	93
Exa 9.11	Example 11	94
Exa 9.12	Example 12	94
Exa 9.13	Example 13	95
Exa 9.14	Example 14	96
Exa 9.15	Example 15	97
Exa 9.16	Example 16	97
Exa 9.17	Example 17	98
Exa 9.18	Example 18	99

Exa 9.19	Example 19	99
Exa 10.1	Example 1	101
Exa 10.2	Example 2	102
Exa 10.3	Example 3	102
Exa 10.4	Example 10	103
Exa 10.5	Example 5	104
Exa 10.6	Example 6	105
Exa 10.7	Example 7	106
Exa 10.8	Example 8	108
Exa 10.9	Example 9	109
Exa 10.10	Example 10	110
Exa 10.11	Example 11	112
Exa 10.12	Example 12	113
Exa 10.13	Example 13	114
Exa 10.14	Example 14	115
Exa 10.15	Example 15	116
Exa 10.16	Example 16	116
Exa 10.17	Example 17	117
Exa 10.18	Example 18	118
Exa 10.19	Example 19	119
Exa 11.1	Example 1	120
Exa 11.2	Example 2	120
Exa 11.3	Example 3	121
Exa 11.4	Example 4	122
Exa 11.5	Example 5	122
Exa 11.6	Example 6	123
Exa 11.7	Example 7	124
Exa 11.8	Example 8	125
Exa 11.9	Example 9	126
Exa 12.1	Example 1	127
Exa 12.2	Example 2	127
Exa 12.3	Example 3	128
Exa 12.4	Example 4	129
Exa 12.5	Example 5	129
Exa 12.6	Example 6	130
Exa 12.7	Example 7	131
Exa 12.8	Example 8	133
Exa 12.9	Example 9	134

Exa 12.10	Example 9	134
Exa 12.11	Example 11	135
Exa 12.12	Example 12	136
Exa 13.1	Example 1	137
Exa 13.2	Example 2	137
Exa 13.3	Example 3	138
Exa 13.4	Example 4	139
Exa 13.5	Example 5	139
Exa 13.6	Example 6	140
Exa 13.7	Example 7	140
Exa 13.8	Example 8	141
Exa 13.9	Example 9	141
Exa 13.10	Example 10	142
Exa 13.11	Example 11	142
Exa 13.13	Example 13	143
Exa 13.14	Example 14	143
Exa 13.15	Example 15	144
Exa 13.16	Example 16	145
Exa 13.17	Example 17	145
Exa 13.18	Example 18	145
Exa 13.19	Example 19	146
Exa 13.20	Example 20	146
Exa 13.21	Example 21	147
Exa 13.22	Example 22	147
Exa 13.23	Example 23	148
Exa 13.24	Example 24	148
Exa 14.1	Example 1	150
Exa 14.2	Example 2	150
Exa 14.3	Example 3	151
Exa 14.4	Example 4	152
Exa 14.5	Example 5	152
Exa 14.6	Example 6	153
Exa 14.7	Example 7	153
Exa 14.8	Example 8	154
Exa 14.9	Example 9	155
Exa 14.10	Example 10	155
Exa 14.11	Example 11	156
Exa 14.12	Example 12	157

Exa 14.13	Example 13	157
Exa 14.14	Example 14	158
Exa 14.15	Example 15	159
Exa 14.16	Example 16	160
Exa 14.17	Example 17	160
Exa 14.18	Example 18	161
Exa 14.19	Example 19	162
Exa 14.20	Example 20	162
Exa 14.21	Example 21	163
Exa 14.22	Example 22	164
Exa 14.23	Example 23	164
Exa 14.24	Example 24	165
Exa 14.25	Example 25	166
Exa 14.26	Example 26	167
Exa 14.29	Example 29	169
Exa 14.30	Example 30	170
Exa 14.31	Example 31	171
Exa 14.32	Example 32	172
Exa 15.1	Example 1	174
Exa 15.2	Example 2	175
Exa 15.3	Example 3	176
Exa 15.4	Example 4	177
Exa 15.5	Example 5	178
Exa 15.6	Example 6	180
Exa 15.7	Example 7	181
Exa 15.8	Example 8	182
Exa 15.9	Example 9	183
Exa 15.10	Example 10	185
Exa 15.11	Example 11	187
Exa 15.12	Example 12	190
Exa 15.13	Example 13	192
Exa 15.14	Example 14	195
Exa 15.15	Example 15	196
Exa 15.16	Example 16	198
Exa 15.17	Example 17	199
Exa 15.18	Example 18	200
Exa 16.1	Example 1	204
Exa 16.2	Example 2	206

Exa 16.3	Example 3	208
Exa 16.4	Example 4	210
Exa 16.5	Example 5	212
Exa 16.6	Example 6	214
Exa 16.7	Example 7	215
Exa 16.8	Example 8	217
Exa 17.1	Example 1	220
Exa 17.2	Example 2	221
Exa 17.3	Example 3	222
Exa 17.4	Example 4	223
Exa 17.5	Example 5	224
Exa 17.9	Example 9	226
Exa 17.12	Example 12	229
Exa 17.16	Example 16	231
Exa 17.17	Example 17	234
Exa 17.19	Example 19	234
Exa 17.20	Example 20	235
Exa 17.21	Example 21	237
Exa 18.1	Example 1	239
Exa 18.2	Example 2	241
Exa 18.3	Example 3	243
Exa 18.4	Example 4	244
Exa 18.5	Example 5	247
Exa 18.7	Example 7	248
Exa 18.8	Example 8	250
Exa 18.9	Example 9	252
Exa 18.14	Example 14	252
Exa 18.15	Example 15	254
Exa 18.16	Example 16	256
Exa 18.19	Example 19	257
Exa 18.20	Example 20	258
Exa 18.21	Example 21	259
Exa 19.1	Example 1	260
Exa 19.2	Example 2	261
Exa 19.3	Example 3	263
Exa 19.4	Example 4	265
Exa 19.5	Example 5	266
Exa 19.6	Example 6	267

Exa 19.7	Example 7	268
Exa 20.1	Example 1	270
Exa 20.2	Example 2	271
Exa 20.3	Example 3	272
Exa 20.4	Example 4	272
Exa 20.5	Example 5	273
Exa 20.6	Example 6	274
Exa 20.7	Example 7	275
Exa 20.8	Example 8	277
Exa 20.9	Example 9	278
Exa 20.10	Example 10	279
Exa 20.11	Example 11	280
Exa 21.1	Example 1	282
Exa 21.2	Example 2	283
Exa 21.3	Example 3	283
Exa 21.4	Example 4	284
Exa 21.5	Example 5	285
Exa 21.6	Example 6	286
Exa 21.7	Example 7	286
Exa 21.8	Example 8	287
Exa 21.10	Example 10	288
Exa 22.1	Example 1	290
Exa 22.2	Example 2	292
Exa 22.3	Example 3	294
Exa 22.4	Example 4	296
Exa 22.5	Example 5	297
Exa 22.6	Example 6	298
Exa 22.7	Example 7	299
Exa 22.8	Example 8	301
Exa 22.9	Example 9	304
Exa 23.1	Example 1	308
Exa 23.2	Example 2	308
Exa 23.3	Example 3	309
Exa 23.4	Example 4	310
Exa 23.5	Example 5	310
Exa 23.6	Example 6	312
Exa 23.7	Example 7	313
Exa 23.8	Example 8	315

Exa 23.9	Example 9	316
Exa 23.10	Example 10	317
Exa 23.11	Example 11	319
Exa 23.12	Example 12	321
Exa 24.1	Example 1	325
Exa 24.2	Example 2	326
Exa 24.3	Example 3	327
Exa 24.4	Example 4	327
Exa 24.5	Example 5	328
Exa 24.6	Example 6	329
Exa 24.7	Example 7	329
Exa 24.8	Example 8	330
Exa 24.9	Example 9	331
Exa 24.10	Example 10	332
Exa 24.11	Example 11	334
Exa 25.2	Example 2	338
Exa 25.3	Example 3	339
Exa 25.4	Example 4	340
Exa 25.5	Example 5	341
Exa 25.6	Example 6	341
Exa 25.7	Example 7	342
Exa 25.8	Example 8	343
Exa 26.1	Example 1	346
Exa 26.2	Example 2	346
Exa 26.3	Example 3	347
Exa 26.4	Example 4	348
Exa 26.5	Example 5	348
Exa 26.6	Example 6	349
Exa 26.7	Example 7	351
Exa 26.9	Example 9	352
Exa 26.11	Example 11	353
Exa 26.12	Example 12	353

Chapter 1

Introduction

Scilab code Exa 1.1 Example 1

```
1 //Chapter 1, Example 1.1
2 clc
3 //Variable Declaration
4 a1 = 532.6 // (a) unit less
   variable
5 a2 = 4.02 // (a) unit less
   variable
6 a3 = 0.036 // (a) unit less
   variable
7 b1 = 0.04 // (b) unit less
   variable
8 b2 = 0.003 // (b) unit less
   variable
9 b3 = 0.0064 // (b) unit less
   variable
10 c1 = 4.632 // (c) unit less
   variable
11 c2 = 2.4 // (c) unit less
   variable
12 d1 = 3.051 // (d) unit less
   variable
```



```

13 d2 = 802 // (d) unit less
    variable
14 e1 = 1402 // (e) unit less
    variable
15 e2 = 6.4 // (e) unit less
    variable
16 f1 = 0.0046 // (f) unit less
    variable
17 f2 = 0.05 // (f) unit less
    variable
18
19 // Calculation
20 // (a)
21 a = a1+a2+a3 // addition function
22 // (b)
23 b = b1+b2+b3 // addition function
24 // (c)
25 c = c1*c2 // Multiplication
    function
26 // (d)
27 d = d1*d2 // Multiplication
    function
28 // (e)
29 e = e1/e2 // Division function
30 // (f)
31 f = f1/f2 // Division function
32
33 // Results
34 printf("(a) %.1 f + %.2 f + %.3 f = %.3 f = %.1 f \n", a1,
    a2, a3, a, a)
35 printf("(b) %.2 f + %.3 f + %.4 f = %.4 f = %.2 f \n", b1,
    b2, b3, b, b)
36 printf("(c) %.3 f * %.1 f = %.4 f = %d \n", c1, c2, c, c)
37 printf("(d) %.3 f * %.0 f = %.3 f = %d \n", d1, d2, d, d)
38 printf("(e) %.0 f / %.1 f = %.4 f = %d \n", e1, e2, e, e)
39 printf("(f) %.4 f / %.2 f = %.4 f = %.2 f", f1, f2, f, f)

```

Scilab code Exa 1.2 Example 2

```
1 //Chapter 1, Example 1.2
2 clc
3 //Variable Declaration
4 a = 1000 // (a) unit less variable
5 b = 0.00001 // (b) unit less variable
6
7 //Calculation
8 //(a)
9 a1 = 1/a //inverse function
10 //(b)
11 b1 = 1/b //inverse function
12
13 //Results
14 printf("(a) %d^-3 \n", a1*10000)
15 printf("(b) %d^+5", round(b1/10000))
```

Scilab code Exa 1.3 Example 3

```
1 //Chapter 1, Example 1.3
2 clc
3 //Variable Declaration
4 a1 = 1000 // (a) unit less
   variable
5 a2 = 10000 // (a) unit less
   variable
6 b1 = 0.00001 // (b) unit less
   variable
7 b2 = 100 // (b) unit less
   variable
8
```

```

 9 // Calculation
10 //(a)
11 a = a1*a2 // multiplication
    function
12 //(b)
13 b = b1*b2 // multiplication
    function
14 // Results
15 printf("(a) %d^7 \n",a/1000000)
16 printf("(b) %d^-3",b*10000)

```

Scilab code Exa 1.4 Example 4

```

1 //Chapter 1, Example 1.4
2 clc
3 //Variable Declaration
4 a1 = 100000 // (a) unit less
    variable
5 a2 = 100 // (a) unit less
    variable
6 b1 = 1000 // (b) unit less
    variable
7 b2 = 0.0001 // (b) unit less
    variable
8
9 // Calculation
10 //(a)
11 a = a1/a2 // Division function
12 //(b)
13 b = b1/b2 // Division function
14 // Results
15 printf("(a) %d^3 \n",a/100)
16 printf("(b) %d^7",b/1000000)

```

Scilab code Exa 1.5 Example 5

```
1 //Chapter 1, Example 1.5
2 clc
3 //Variable Declaration
4 a1 = 100 // (a) unit less
   variable
5 b1 = 1000 // (b) unit less
   variable
6 c1 = 0.01 // (c) unit less
   variable
7
8 //Calculation
9 //(a)
10 a = a1^4 // multiplication
   function
11 //(b)
12 b = b1^-2 // multiplication
   function
13 //(c)
14 c = c1^-3 // multiplication
   function
15 //Results
16 printf(" (a) %d^8 \n", a/10000000)
17 printf(" (b) %d^-6 \n", b*10000000)
18 printf(" (c) %d^6", round(c/100000))
```

Scilab code Exa 1.6 Example 6

```
1 //Chapter 1, Example 1.6
2 clc
3 //Variable Declaration
```

```

4 a1 = 6300 // (a) unit less
   variable
5 a2 = 75000 // (a) unit less
   variable
6 b1 = 0.00096 // (b) unit less
   variable
7 b2 = 0.000086 // (b) unit less
   variable
8
9 // Calculation
10 // (a)
11 a = a1 + a2 // addition function
12 // (b)
13 b = b1 - b2 // subtraction function
14 // Results
15 printf("(a) %.1f x 10^3 \n", a/1000)
16 printf("(b) %.1f x 10^-5", b*100000)

```

Scilab code Exa 1.7 Example 7

```

1 // Chapter 1, Example 1.7
2 clc
3 // Variable Declaration
4 a1 = 0.0002 // (a) unit less
   variable
5 a2 = 0.000007 // (a) unit less
   variable
6 b1 = 340000 // (b) unit less
   variable
7 b2 = 0.00061 // (b) unit less
   variable
8
9 // Calculation
10 // (a)
11 a = a1*a2 // multiplication

```

```

        function
12 // (b)
13 b = b1*b2           // multiplication
        function
14 // Results
15 printf("(a) %d x 10-10 \n", a*10000000000)
16 printf("(b) %.1 f", b)

```

Scilab code Exa 1.8 Example 8

```

1 // Chapter 1, Example 1.8
2 clc
3 // Variable Declaration
4 a1 = 0.00047         // (a) unit less
        variable
5 a2 = 0.002           // (a) unit less
        variable
6 b1 = 690000         // (b) unit less
        variable
7 b2 = 0.00000013    // (b) unit less
        variable
8
9 // Calculation
10 // (a)
11 a = a1/a2           // division function
12 // (b)
13 b = b1/b2           // division function
14 // Results
15 printf("(a) %.1 f x 10-2 \n", a*100)
16 printf("(b) %.2 f x 1012", b/1000000000000)

```

Scilab code Exa 1.9 Example 9

```

1 //Chapter 1, Example 1.9
2 clc
3 //Variable Declaration
4 a1 = 0.00003 // (a) unit less
   variable
5 b1 = 90800000 // (b) unit less
   variable
6
7 //Calculation
8 //(a)
9 a = a1**3 //multiplication
   function
10 //(b)
11 b = b1**2 //multiplication
   function
12 //Results
13 printf("(a) %d x 10-15\n", a*10**15)
14 printf("(b) %.4f x 1014", b/10**14)

```

Scilab code Exa 1.10 Example 10

```

1 //Chapter 1, Example 1.10
2 clc
3 //Variable Declaration
4 a = 1000000 //in ohms
5 b = 1000000 //in meters
6 c = 0.0001 //in second
7 d = 0.000001 //in farad
8
9 //Results
10 printf("(a) %d x 106 ohms = %d megaohm \n", a/10**6,
   a/10**6)
11 printf("(b) %d x 103 meters = %d kilometers \n", b
   /10**4, b/10**4)
12 printf("(c) %.1f x 10-3 second = %.1f millisecond \n

```

```

    ",c*1000,c*10**3)
13 printf("(d) %d x 10-6 farad = %d microfarad",d
    *10**6,d*10**6)

```

Scilab code Exa 1.11 Example 11

```

1 //Chapter 1, Example 1.11
2 clc
3 //Variable Declaration
4 a = 41200 //in meter
5 b = 0.00956 //in Joule
6 c = 0.000768 //in sec
7 d1 = 8400 //in meter
8 d2 = 0.06 //in meter
9 e1 = 0.0003 //in sec
10
11 //Calculation
12 d = d1/d2 //in meter
13 e = e1^4 //in sec
14
15 //Results
16 printf("(a) %.1 f km \n",a/1000)
17 printf("(b) %.2 f mJ \n",b*1000)
18 printf("(c) %d us \n",c*10**6)
19 printf("(d) %d km \n",d/1000)
20 printf("(e) %.4 f ps",e*10**12)

```

Scilab code Exa 1.12 Example 12

```

1 //Chapter 1, Example 1.12
2 clc
3 //Variable Declaration
4 a = 20*10^3 //frequency in hertz

```



```

5 b = 0.01           //timeperiod in millisec
6 c = 0.002         //in Km
7
8 //Results
9 printf("(a) %.2f MHz \n", a/10**6)
10 printf("(b) %d us \n", b*10**3)
11 printf("(c) %d mm", c*10**6)

```

Scilab code Exa 1.13 Example 13

```

1 //Chapter 1, Example 1.13
2 clc
3 //Variable Declaration
4 a1 = 6.8           //time in minutes
5 a2 = 60           //60second in 1 minute
6 b1 = 0.24         //measurement in meter
7 b2 = 100          //100centimeter in 1 meter
8
9 //Calculation
10 a = a1*a2
11 b = b1*b2
12
13 //Results
14 printf("(a) %d s \n", a)
15 printf("(b) %d cm", b)

```

Scilab code Exa 1.14 Example 14

```

1 //Chapter 1, Example 1.14
2 clc
3 //Variable Declaration
4 a1 = 0.5           //half day
5 a2 = 24           //24hours in 1day

```

```

6 a3 = 60 //60minutes in 1hour
7 b1 =2.2 //yards
8 b2 = 3 //3ft in 1 yard
9 b3 = 12 //12inches in 1ft
10 b4 = 39.37 //in inches
11
12 //Calculation
13 a = a1*a2*a3
14 b = (b1*b2*b3)/b4
15
16 //Results
17 printf("(a) %d min \n",a)
18 printf("(b) %.3 f m",b)

```

Scilab code Exa 1.15 Example 15

```

1 //Chapter 1, Example 1.15
2 clc
3 //Variable Declaration
4 a1 =100 //in kilometers
5 a2 = 1000 //in meters
6 a3 = 39.37 //in inches
7 a4 = 12 //in inches
8 a5 = 5280 //in ft
9 b1 = 60 //in minutes
10 b2 = 4 //in hours
11
12 //Calculation
13 a = (a1*a2*a3)/(a4*a5)
14 b = b1/b2
15
16 //Results
17 printf("(a) %.2 f mi/h \n",a)
18 printf("(b) %d mi/h",b)

```

Scilab code Exa 1.16 Example 16

```
1 //Chapter 1, Example 1.16
2 clc
3 //Variable Declaration
4 a1 = 9
5 a2 = 3
6 c1 = 3
7 c2 = 9
8 c3 = 4
9 d1 = 1/4
10 d2 = 1/6
11 d3 = 2/3
12
13 //Calculation
14 a = sqrt(a1/a2)
15 b = (c1+c2)/c3
16 c = d1+d2+d3
17
18 //Results
19 printf("(a) %.2 f \n",a)
20 printf("(b) %.d \n",b)
21 printf("(c) %.2 f ",c)
```

Chapter 2

Current and Voltage

Scilab code Exa 2.1 Example 1

```
1 //Chapter 2, Example 2.1
2 clc
3 //Variable Declaration
4 q = 0.16 //charge in coulomb (C)
5 t = 64*10^-3 //time in second (sec)
6
7 //Calculation
8 I = q/t //current in ampere (A)
9
10 //Results
11 printf("I = %.2f A \n", I)
```

Scilab code Exa 2.2 Example 2

```
1 //Chapter 2, Example 2.2
2 clc
3 //Variable Declaration
4 Q = 641*10^-3 //charge in coulomb
```

```

5 I = 5*10^-3 //current in ampere
6 //Calculation
7 t = Q/I //time in seconds
8 //Results
9 printf("t = %.3f s",t/100)

```

Scilab code Exa 2.3 Example 3

```

1 //Chapter 2, Example 2.3
2 clc
3 //Variable Declaration
4 W = 60 //in joule
5 Q = 20 //in coulomb
6 //Calculation
7 V = W/Q //ans in volt
8 //Results
9 printf("V = %d V",V)

```

Scilab code Exa 2.4 Example 4

```

1 //Chapter 2, Example 2.4
2 clc
3 //Variable Declaration
4 Q = 50*10^-6 //charge in coulomb
5 V = 6 //in volt
6
7 //Calculation
8 W = Q*V //in microsec joule
9
10 //Results
11 printf("W = %d uJ",W*10^6)

```

Scilab code Exa 2.5 Example 5

```
1 //Chapter 2, Example 2.5
2 clc
3 //Variable Declaration
4 a = 450*10^-3 //current in millampere
5 b = 600*10^-3 //current in millampere
6 d = 45 //temperature in degree from
    graph 2.18.b
7 m = 60 //unit time in sec
8
9 //Calculation
10 life = (a/b)*m //life in minutes
11
12 //Results
13 printf("a. Life = %.d min \n",life)
14 printf("b. High temperature = %d degreeC",d)
```

Chapter 3

Resistance

Scilab code Exa 3.1 Example 1

```
1 //Chapter 3, Example 3.1
2 clc
3 //Variable Declaration
4 p = 10.37 //in CM.ohm/ft
5 l = 100 //length in ft
6 a = 400 //area in CM
7
8 //Calculation
9 R = p*l/a //resistance in ohm
10
11 //Results
12 printf("R = %.2 f ohm",R)
```

Scilab code Exa 3.2 Example 3

```
1 //Chapter 3, Example 3.3
2 clc
3 //Variable Declaration
```

```

4 p = 10.37           //in CM.ohm/ft
5 l = 3              //length in ft
6 a = 3.185*10^6     //area in cm
7
8 //Calculation
9 R = p*l/a          //resistance in ohm
10
11 //Results
12 printf("R = %.3f x 10^-6 ohm",R*10^6)

```

Scilab code Exa 3.3 Example 3

```

1 //Chapter 3, Example 3.3
2 clc
3 //Variable Declaration
4 p = 10.37           //in CM.ohm/ft
5 l = 3              //length in ft
6 a = 3.185*10^6     //area in cm
7
8 //Calculation
9 R = p*l/a          //resistance in ohm
10
11 //Results
12 printf("R = %.3f x 10^-6 ohm",R*10^6)

```

Scilab code Exa 3.4 Example 4

```

1 //Chapter 3, Example 3.4
2 clc
3 //Variable Declaration
4 l = 650             //in ft
5 o = 0.6282         // in ohm
6 L = 1000           //in ft

```



```

7 // Calculation
8 R = l*o/L //resistance in ohm
9
10 // Results
11 printf("R = %.3 f ohm" ,R)

```

Scilab code Exa 3.5 Example 5

```

1 //Chapter 3, Example 3.5
2 clc
3 //Variable Declaration
4 A = 6529.9 //area in cm
5
6 // Calculation
7 d = sqrt(6529.9)/1000 //answer in inch
8
9 // Results
10 printf("d = %.4 f in" ,d)

```

Scilab code Exa 3.6 Example 6

```

1 //Chapter 3, Example 3.6
2 clc
3 //Variable Declaration
4 p = 10.37 //in CM.ohm/ft
5 l = 100 //length in ft
6 r = 0.025 //in ohm
7
8 // Calculation
9 A = p*l/r //in CM
10
11 // Results
12 printf("A = %d cm" ,A)

```

Scilab code Exa 3.7 Example 7

```
1 //Chapter 3, Example 3.7
2 clc
3 //Variable Declaration
4 l = 3048 // in cm
5 d = 0.032 //in cm
6 r = 1.723*10^-6 //in ohm.cm
7 p = 3.14
8 r1 = 10.37 //in cm.ohm/ft
9 l1 = 100 //in ft
10 a = 159.79 //in cm
11
12 //Calculation
13 A = (p*d^2)/4
14 R = (r*l)/A //resistance in ohm
15 R1 = (r1*l1)/a //resistance in ohm
16 //Results
17 printf("R = %.1 f ohm \n",R)
18 printf("R = %.1 f ohm",R1)
```

Scilab code Exa 3.8 Example 8

```
1 //Chapter 3, Example 3.8
2 clc
3 //Variable Declaration
4 r = 100 //sheet resistance in ohm
5 l = 0.6 //length in cm
6 w = 0.3 //in cm
7
8 //Calculation
```

```
9 R = (r*l)/w           //resistance in ohm
10
11 //Results
12 printf("R = %d ohm",R)
```

Scilab code Exa 3.9 Example 9

```
1 //Chapter 3, Example 3.9
2 clc
3 //Variable Declaration
4 r = 50           //copper wire in ohm
5 t1 = 334.5      //in degree celsius
6 t2 = 254.5      //in degree celsius
7
8
9 //Calculation
10 R = (r*t1)/t2   //in ohm
11
12 //Results
13 printf("R = %.2 f ohm",R)
```

Scilab code Exa 3.10 Example 10

```
1 //Chapter 3, Example 3.10
2 clc
3 //Variable Declaration
4 r = 30           //copper wire in ohm
5 t1 = 194.5      //in degree celsius
6 t2 = 234.5      //in degree celsius
7
8 //Calculation
9 R = (r*t1)/t2   //in ohm
10
```

```
11 //Results
12 printf("R2 = %.2 f ohm" ,R)
```

Scilab code Exa 3.11 Example 11

```
1 //Chapter 3, Example 3.11
2 clc
3 //Variable Declaration
4 r1 = 120*10^-3 //in milliohm
5 t1 = 256 //in degree celsius
6 r2 = 100*10^-3 //in milliohm
7 t2 = 236 //in degree celsius
8 //Calculation
9 T = r1*(t1/r2)-t2 //in degree celsius
10
11 //Results
12 printf("T2 = %.1 f C" ,T)
```

Scilab code Exa 3.12 Example 12

```
1 //Chapter 3, Example 3.12
2 clc
3 //Variable Declaration
4 r =1000 //in ohm
5 p = 2500
6 t1 = 60 //in degree celsius
7 t2 = 20 //in degree celsius
8 a = 10^6
9
10 //Calculation
11 D = (r/a)*p*(t1-t2)
12 R = r+D //resistance in ohm
13 //Results
```

```
14 printf("R = %d ohm" ,R)
```

Scilab code Exa 3.14 Example 14

```
1 //Chapter 3, Example 3.14
2 clc
3 //Variable Declaration
4 a = 0.7
5 b = 1.4
6
7 //Calculation
8 Gn = a/b
9
10 //Results
11 printf("Gn = %.1 fGi" ,Gn)
```

Chapter 4

Ohms Law Power and Energy

Scilab code Exa 4.1 Example 4

```
1 //Chapter 4, Example 4.1
2 clc
3 //Variable Declaration
4 E = 9 //battery in volt
5 R = 2.2 //resistor in ohm
6
7
8 //Calculation
9 I = E/R //current in ampere
10
11 //Results
12 printf("I = %.2 f A",I)
```

Scilab code Exa 4.2 Example 2

```
1 //Chapter 4, Example 4.2
2 clc
3 //Variable Declaration
```

```

4 E = 120 //applied voltage
5 I = 500*10^-3 //current in ampere
6
7 //Calculation
8 R = E/I //in ohm
9
10 //Results
11 printf("R = %d ohm",R)

```

Scilab code Exa 4.3 Example 3

```

1 //Chapter 4, Example 4.3
2 clc
3 //Variable Declaration
4 V = 16 //voltage
5 R = 2*10^3 //in ohm
6
7 //Calculation
8 I = V/R //in miliampere
9
10 //Results
11 printf("I = %d mA",I*1000)

```

Scilab code Exa 4.4 Example 4

```

1 //Chapter 4, Example 4.4
2 clc
3 //Variable Declaration
4 I = 1.5 //current in ampere
5 R = 80 //in ohm
6
7 //Calculation
8 E = I*R //voltage

```

```
9
10 //Results
11 printf("E = %d V",E)
```

Scilab code Exa 4.5 Example 5

```
1 //Chapter 4, Example 4.5
2 clc
3 //Variable Declaration
4 V1 = 6 //in volts
5 I1 = 3*10^-3 //in miliampere
6 V2 = 2 //in volts
7 I2 = 1*10^-3 //in miliampere
8 //Calculation
9 Rdc = V1/I1 //in kiloohm
10 R = V2/I2 //in kiloohm
11
12 //Results
13 printf("Rdc = %d kohm \n",Rdc/10^3)
14 printf("R = %d kohm",R/10^3)
```

Scilab code Exa 4.6 Example 6

```
1 //Chapter 4, Example 4.6
2 clc
3 //Variable Declaration
4 V = 120 //in volts
5 I = 5 //in ampere
6
7 //Calculation
8 P = V*I //power in kwatt
9
10 //Results
```



```
11 printf("P = %.1f kW",P/10^3)
```

Scilab code Exa 4.7 Example 7

```
1 //Chapter 4, Example 4.7
2 clc
3 //Variable Declaration
4 I = 4 //in ampere
5 R = 5 //in ohm
6
7 //Calculation
8 P = (I*I)*R //power in watt
9
10 //Results
11 printf("P = %d W",P)
```

Scilab code Exa 4.8 Example 8

```
1 //Chapter 4, Example 4.8
2 clc
3 //Variable Declaration
4 I = 0.625 //in ampere
5 V = 120 //in volts
6
7
8 //Calculation
9 P = V*I //power in watt
10 R = V/I //in ohm
11
12 //Results
13 printf("P = %d W \n",P)
14 printf("R = %d ohm",R)
```

Scilab code Exa 4.9 Example 9

```
1 //Chapter 4, Example 4.9
2 clc
3 //Variable Declaration
4 P = 20*10^-3 //power in watt
5 R = 5*10^3 //in ohm
6
7 //Calculation
8 I = sqrt(P/R) //current in miliampere
9
10 //Results
11 printf("I = %d mA",I*1000)
```

Scilab code Exa 4.10 Example 10

```
1 //Chapter 4, Example 4.10
2 clc
3 //Variable Declaration
4 p1 = 1492 //output power
5 n = 0.75 //efficiency
6 E = 220 //applied voltage
7
8 //Calculation
9 Pi = p1/n //inputpower in watt
10 I = Pi/E //current in ampere
11
12 //Results
13 printf("Pi = %.2 f W \n",Pi)
14 printf("I = %.2 f A",I)
```

Scilab code Exa 4.11 Example 11

```
1 //Chapter 4, Example 4.11
2 clc
3 //Variable Declaration
4 n = 0.80 //efficiency
5 a2 = 120 //in volts
6 a3 = 8 //in ampere
7 b1 = 1 //motor in horsepower
8 b2 = 746 //power in watt
9 //Calculation
10 Po = n*a2*a3 //output power
11 I = Po*(b1/b2) //in horsepower
12 //Results
13 printf("H = %.3 f hp",I)
```

Scilab code Exa 4.12 Example 12

```
1 //Chapter 4, Example 4.12
2 clc
3 //Variable Declaration
4 n = 0.85 //efficiency
5 Wi = 50 //applied energy
6
7 //Calculation
8 Wo = n*Wi //output energy
9
10 //Results
11 printf("Wo = %.1 f J",Wo)
```

Scilab code Exa 4.13 Example 13

```
1 //Chapter 4, Example 4.13
2 clc
3 //Variable Declaration
4 n1 = 0.90 //efficiency1
5 n2 = 0.85 //efficiency2
6 n3 = 0.95 //efficiency3
7
8 //Calculation
9 nT = n1*n2*n3 //efficiency in percentage
10
11 //Results
12 printf("nT = %.1 f percentage",nT*100)
```

Scilab code Exa 4.14 Example 14

```
1 //Chapter 4, Example 4.14
2 clc
3 //Variable Declaration
4 n1 = 0.40 //efficiency1
5 n2 = 0.85 //efficiency2
6 n3 = 0.95 //efficiency3
7
8 //Calculation
9 nT = n1*n2*n3 //efficiency in percentage
10
11 //Results
12 printf("nT = %.1 f percentage",nT*100)
```

Scilab code Exa 4.15 Example 15

```
1 //Chapter 4, Example 4.15
```

```

2  clc
3  //Variable Declaration
4  cost = 9
5  r1 = 5360*103           //in kWh
6  r2 = 4650*103           //pervious reading
7
8  //Calculation
9  R = r1-r2               //used
10 M = R*(cost/103)        //in dollar
11
12 //Results
13 printf("M = %.2f dollar",M/100)

```

Scilab code Exa 4.16 Example 16

```

1  //Chapter 4, Example 4.16
2  clc
3  //Variable Declaration
4  a1 = 60                 //bulb in watt
5  a2 = 24                 //hours per day
6  a3 = 365                //days
7
8  //Calculation
9  Pt = a1*a2*a3           //power in Wh
10 W = Pt/1000             //energy in kWh
11
12 //Results
13 printf("W = %.2f kWh",W)

```

Scilab code Exa 4.17 Example 17

```

1  //Chapter 4, Example 4.17
2  clc

```

```

3 //Variable Declaration
4 p1 = 4*10^3 //energy in kWh
5 p2 = 205 //power in watt
6
7 //Calculation
8 Pt = p1*1000 //power in watt
9 W = Pt/p2 // in h
10 //Results
11 printf("W = %.2 f h" ,W/1000)

```

Scilab code Exa 4.18 Example 18

```

1 //Chapter 4, Example 4.18
2 clc
3 //Variable Declaration
4 p1 = 5 //motor in horsepower
5 p2 = 746 //in W/hp
6 p3 = 2 //in h
7 c = 9 //in euro/kWh
8
9 //Calculation
10 Pt = p1*p2*p3 //power
11 W = Pt/1000 //in kWh
12 cost = W*c //unit
13
14 //Results
15 printf("Cost = %.2 f unit",cost)

```

Scilab code Exa 4.19 Example 19

```

1 //Chapter 4, Example 4.19
2 clc
3 //Variable Declaration

```

```
4 w1 = 1200           //toaster in watt
5 w2 = 50             //bulbs in watt
6 w3 = 400           //washing machine in watt
7 w4 = 4800          //electric clothes in watt
8 c = 9              //cost in unit/kWh
9 //Calculation
10 W = ((w1*0.5)+(6*w2*4)+(w3*0.75)+(w4*0.33))/1000
      //energy in kWh
11 cost = W*c         //in euro
12
13 //Results
14 printf("Cost = %.1f unit",cost)
```

Chapter 5

Series Circuits

Scilab code Exa 5.1 Example 1

```
1 //Chapter 5, Example 5.1
2 clc
3 //Variable Declaration
4 r1 = 2 //resistance1
5 r2 = 1 //resistance2
6 r3 = 5 //resistance3
7 E = 20 //supply voltage
8
9
10
11 //Calculation
12 rt = r1+r2+r3 //total resistance
13 I = E/rt //current
14 V1 = I*r1 //voltage1
15 V2 = I*r2 //voltage2
16 V3 = I*r3 //voltage3
17 P1 = V1*I //power1
18 P2 = I*I*r2 //power2
19 P3 = V3*V3/r3 //power3
20 Pdel1 = E*I //power delivered by source
21 Pdel = P1+P2+P3 //Total power
```



```

22 //Results
23 printf("a. RT = %d ohm \n",rt)
24 printf("b. Is = %.1 f A \n",I)
25 printf("c. V1 = %d V\n",V1)
26 printf("   V2 = %.1 f V \n",V2)
27 printf("   V3 = %.1 f V \n",V3)
28 printf("d. P1 = %.1 f W \n",P1)
29 printf("   P2 = %.2 f W \n",P2)
30 printf("   P3 = %.2 f W \n",P3)
31 printf("e. Pdel = %d W \n",Pdel1)
32 printf("   %d W = %d W (checks) \n",Pdel1,Pdel)

```

Scilab code Exa 5.2 Example 2

```

1 //Chapter 5, Example 5.2
2 clc
3 //Variable Declaration
4 N = 3 //number of resistance
5 R1 = 7 //resistance1
6 R2 = 4 //resistance2
7 E = 50 //supply voltage
8 //Calculation
9 Rt = N*R1+R2 //total resistance
10 I = E/Rt //current in ampere
11 V2 = I*R2 //voltage2
12 //Results
13 printf("RT = %d ohm \n",Rt)
14 printf("I = %d A \n",I)
15 printf("V2 = %d V",V2)

```

Scilab code Exa 5.3 Example 3

```

1 //Chapter 5, Example 5.3

```

```

2  clc
3  //Variable Declaration
4  Rt = 12*103           //total resistance
5  R2 = 4*103           //resistance2
6  R3 = 6*103           //resistance3
7  I = 6*10-3          //current in ampere
8  //Calculation
9  R1 = Rt-(R2+R3)       //resistance1
10 E = I*Rt              //voltage
11 //Results
12 printf("R1 = %d kohm \n",R1/103)
13 printf("E = %d V",E)

```

Scilab code Exa 5.4 Example 4

```

1  //Chapter 5, Example 5.4
2  clc
3  //Variable Declaration
4  e1 = 16                //supply voltage1
5  v2 = 4.2              //voltage2
6  e2 = 9                //supply voltage2
7  e = 32                //supply voltage
8  v = 12                //voltage
9  v3 = 6                //voltage of R2
10 v4 =14                //voltage of R3
11 //Calculation
12 v1 = e1-v2-e2         //voltage1
13 vx = e-v              //unknown voltage
14 vx = v3+v4           //unknown voltage
15 //Results
16 printf("V1 = %.1 f V \n",v1)
17 printf("Vx = %d V \n",vx)
18 printf("Vx = %d V",vx)

```

Scilab code Exa 5.5 Example 5

```
1 //Chapter 5, Example 5.5
2 clc
3 //Variable Declaration
4 e1 = 25 //supply voltage1
5 e2 = 15 //supply voltage2
6 e3 = 20 //supply voltage3
7 //Calculation
8 v1 = e1+e2 //voltage1
9 v2 = -e3 //voltage2
10 //Results
11 printf("V1 = %d V \n",v1)
12 printf("V2 = %d V",v2)
```

Scilab code Exa 5.6 Example 6

```
1 //Chapter 5, Example 5.6
2 clc
3 //Variable Declaration
4 v1 = 60 //supply voltage
5 v2 = 40 //voltage1
6 v3 = 30 //voltage2
7 v4 = -6 //voltage1
8 v5 = 14 //supply voltage
9 v6 = 2 //voltage2
10 //Calculation
11 va = v1+v3-v2 //unknown voltage
12 vb = v4-v5+v6 //unknown voltage
13 //Results
14 printf("Vx = %d V \n",va)
15 printf("Vx = %d V \n",vb)
```

Scilab code Exa 5.7 Example 7

```
1 //Chapter 5, Example 5.7
2 clc
3 //Variable Declaration
4 r1 = 4 //resistor1
5 r2 = 6 //resistor2
6 e = 20 //supply voltage
7
8 //Calculation
9 rt = r1+r2 //total resistance
10 i = e/rt //current
11 v1 = i*r1 //voltage1
12 v2 = i*r2 //voltage2
13 p4 = v1*v1/r1 //power of R1
14 p6 = (i*i)*r2 //power of R2
15 pe = e*i //power of source
16 pe1 = p4+p6 //Total power
17 sv = v1+v2 //Total voltage
18 //Results
19 printf("a. Rt = %d ohm \n",rt)
20 printf("b. I = %d A \n",i)
21 printf("c. V1 = %d V \n",v1)
22 printf(" V2 = %d V \n",v2)
23 printf("d. P4 = %d W \n",p4)
24 printf(" P6 = %d W \n",p6)
25 printf("e. Pe = %d W \n",pe)
26 printf(" %d W = %d W (checks)\n",pe,pe1)
27 printf("f. %d V = %d V (checks)",e,sv)
```

Scilab code Exa 5.8 Example 8

```

1 //Chapter 5, Example 5.8
2 clc
3 //Variable Declaration
4 e = 54 //supply voltage
5 v1 = 18 //voltage1
6 v3 = 15 //voltage3
7 r2 = 7 //resistance2
8 //Calculation
9 v2 = e-v1-v3 //voltage2
10 i = v2/r2 //current in ampere
11 r1 = v1/i //resistance1
12 r3 = v3/i //resistance3
13
14 //Results
15 printf("a. V2 = %d V \n",v2)
16 printf("b. I = %d A \n",i)
17 printf("c. R1 = %d ohm \n",r1)
18 printf(" R3 = %d ohm \n",r3)

```

Scilab code Exa 5.9 Examp1 9

```

1 //Chapter 5, Example 5.9
2 clc
3 //Variable Declaration
4 r1 = 4 //resistance1
5 r2 = 7 //resistance2
6 e = 37.5 //supply voltage
7 i = 2.5 //current in ampere
8
9 //Calculation
10 rt = 2*r1+r2 //total resistance
11 i = e/rt //current in ampere
12 v = i*r2 //voltage of R2
13
14 //Results

```

```
15 printf("I = %.1 f A \n", i)
16 printf("V7ohm = %.1 f V \n", v)
```

Scilab code Exa 5.10 Example 10

```
1 //Chapter 5, Example 5.10
2 clc
3 //Variable Declaration
4 r1 = 20 //resistor1
5 e = 64 //supply voltage
6 r2 = 60 //resistor2
7
8 //Calculation
9 rt = r1+r2 //total resistance
10 v1 = r1*e/rt //voltage1
11
12 //Results
13 printf("V1 = %d V", v1)
```

Scilab code Exa 5.11 Example 11

```
1 //Chapter 5, Example 5.11
2 clc
3 //Variable Declaration
4 r1 = 2000 //resistance1
5 r2 = 5000 //resistance2
6 r3 = 8000 //resistance3
7 e = 45 //supply voltage
8
9 //Calculation
10 rt = r1+r2+r3 //total resistance
11 v1 = r1*e/rt //voltage1
12 v3 = r3*e/rt //voltage3
```

```

13 //Results
14 printf("V1 = %d V \n",v1)
15 printf("V2 = %d V",v3)

```

Scilab code Exa 5.12 Example 12

```

1 //Chapter 5, Example 5.12
2 clc
3 //Variable Declaration
4 r1 = 2000 //resistance1
5 r2 = 5000 //resistance2
6 r3 = 8000 //resistance3
7 e = 45 //supply voltage
8
9
10 //Calculation
11 rt = r1+r2+r3 //Total resistance
12 r = r1+r2 //resistance of R1 or R2
13 v = r*e/rt //volage
14 //Results
15 printf("V = %d V",v)

```

Scilab code Exa 5.13 Example 13

```

1 //Chapter 5, Example 5.13
2 clc
3 //Variable Declaration
4 e = 20 //supply voltage
5 i = 4*10^-3 //current
6 r2 = 1000 //resistance2
7
8 //Calculation
9 r1 = 4*r2 //resistance1

```

```
10
11 //Results
12 printf("R2 = %d Kohm \n",r2/1000)
13 printf("R1 = %d kohm",r1/10^3)
```

Scilab code Exa 5.14 Example 14

```
1 //Chapter 5, Example 5.14
2 clc
3 //Variable Declaration
4 va = 16 //voltage Va
5 vb = 20 //voltage Vb
6
7 //Calculation
8 vab = va-vb //voltage Vab
9
10 //Results
11 printf("Vab = %d V",vab)
```

Scilab code Exa 5.15 Example 15

```
1 //Chapter 5, Example 5.15
2 clc
3 //Variable Declaration
4 vab = 5 //voltage Vab
5 vb = 4 //voltage Vb
6
7 //Calculation
8 va = vab + vb //voltage Va
9
10 //Results
11 printf("Va = %d V",va)
```

Scilab code Exa 5.16 Example 16

```
1 //Chapter 5, Example 5.16
2 clc
3 //Variable Declaration
4 va = 20 //voltage Va
5 vb = -15 //voltage Vb
6
7 //Calculation
8 vab = va-vb //voltage Vab
9
10 //Results
11 printf("Vab = %d V",vab)
```

Scilab code Exa 5.17 Example 17

```
1 //Chapter 5, Example 5.17
2 clc
3 //Variable Declaration
4 e1 = 10 //supply voltage1
5 v1 = 4 //voltage1
6 e2 = 20 //supply voltage2
7 va = 10 //voltage Va
8
9 //Calculation
10 vb = e1-v1 //voltage Vb
11 vc = vb-e2 //voltage Vc
12 vac = va-vc //voltage Vac
13
14 //Results
15 printf("Vb = %d V \n",vb)
16 printf("Vc = %d V \n",vc)
```

```
17 printf("Vac = %d V \n", vac)
```

Scilab code Exa 5.18 Example 18

```
1 //Chapter 5, Example 5.18
2 clc
3 //Variable Declaration
4 r2 = 25 //resistance
5 r1 = 20 //resistance in ohm
6 e1 = -19 //supply voltage in V
7 e2 = 35 //supply voltage in V
8 //Calculation
9 r = r1+r2 //total resistance in
   ohm
10 e = e2-e1 //total supply voltage
   in V
11 i = e/r //current in ampere A
12 vab = i*r2 //in voltage Vab
13 vcb = -i*r1 //in voltage Vcd
14 vc = e1 //in voltage Vc
15
16 //Results
17 printf("Vab = %d V \n", vab)
18 printf("Vcb = %d V \n", vcb)
19 printf("Vc = %d V \n", vc)
```

Scilab code Exa 5.19 Example 19

```
1 //Chapter 5, Example 5.19
2 clc
3 //Variable Declaration
4 r1 = 4 //resistance1
5 e = 24 //supply voltage
```

```

6 r2 = 2 //resistance2
7
8 //Calculation
9 v1 = (r1*e)/(r1+r2) //voltage1
10 v2 = (r2*e)/(r1+r2) //voltage2
11
12 //Results
13 printf("V1 = %d V \n",v1)
14 printf("V2 = %d V \n",v2)

```

Scilab code Exa 5.20 Example 20

```

1 //Chapter 5, Example 5.20
2 clc
3 //Variable Declaration
4 r1 = 2 //resistor1
5 e = 10 //supply voltage
6 r2 = 3 //resistor2
7 r3 = 5 //resistor3
8
9 //Calculation
10 rt = r1+r2+r3 //total resistance
11 vab = r1*e/rt //voltage Vab
12 vb = (r2+r3)*e/rt //voltage Vb
13 vc = 0 //coltage Vc
14 vb1 = e-vab //voltage Vb
15 //Results
16 printf("a. Vab = +%d V \n",vab)
17 printf("b. Vb = %d V \n",vb)
18 printf(" or Vb = %d V \n",vb1)
19 printf("c. Vc = ground potential = %d V",vc)

```

Scilab code Exa 5.21 Example 21

```

1 //Chapter 5, Example 5.21
2 clc
3 //Variable Declaration
4 vn1 = 40 //voltage with no load
5 il = 77*10^-3 //load current in ampere
6 rl = 0.5*10^3 //load resistor in ohm
7
8 //Calculation
9 rint = (vn1/il)-rl //internal resistance in ohm
10
11 //Results
12 printf("Rint = %.2 f ohm",rint)

```

Scilab code Exa 5.22 Example 22

```

1 //Chapter 5, Example 5.22
2 clc
3 //Variable Declaration
4 e = 30 //supply voltage
5 rint = 2 //internal resistance
6 rl = 13 //load resistance
7 vn1 = 30 //voltage with no load
8
9
10 //Calculation
11 il = e/(rint+rl) //load current in ampere
12 vl = vn1-il*rint //load voltage in volts
13 plostm = il*il*rint //power lost in Watt
14
15 //Results
16 printf("Vl = %d V \n",vl)
17 printf("Plost = %d W",plostm)

```

Scilab code Exa 5.23 Example 23

```
1 //Chapter 5, Example 5.23
2 clc
3 //Variable Declaration
4 vn1 = 120 //voltage with no load
5 vf1 = 100 //voltage with load
6
7 //Calculation
8 vr = ((vn1-vf1)/vf1)*100 //voltage regulation in
    percent
9 //Results
10 printf("VR = %d percent",vr)
```

Scilab code Exa 5.24 Example 24

```
1 //Chapter 5, Example 5.24
2 clc
3 //Variable Declaration
4 rint = 19.48 //internal resistance
    in ohm
5 rl = 500 //load resistor in ohm
6
7 //Calculation
8 vr = (rint/rl)*100 //voltage regulation
    in percent
9
10 //Results
11 printf("VR = %.1f percent",vr)
```

Chapter 6

Parallel Circuits

Scilab code Exa 6.1 Example 1

```
1 //Chapter 6, Example 6.1
2 clc
3 //Variable Declaration
4 g1 = 1/3 //conductance in seimens
5 g2 = 1/6 //conductance in seimens
6
7 //Calculation
8 gt = g1+g2 //total conductance in
  seimens
9 rt = 1/gt //resistance in ohm
10
11 //Results
12 printf("Gt = %.1 f S \n",gt)
13 printf("Rt = %d ohm",rt)
```

Scilab code Exa 6.2 Example 2

```
1 //Chapter 6, Example 6.2
```

```

2  clc
3  //Variable Declaration
4  g1 = 0.5           //conductance in seimens
5  g2 = 1/10         //conductance in seimens
6
7  //Calculation
8  gt = g1 + g2      //total conductance in
    seimens
9  rt = 1/gt         //resistance in ohm
10
11 //Results
12 printf("Gt = %.1 f S \n",gt)
13 printf("Rt = %.3 f ohm",rt)

```

Scilab code Exa 6.3 Example 3

```

1  //Chapter 6, Example 6.3
2  clc
3  //Variable Declaration
4  r1 = 1/2           //resistance in ohm
5  r2 = 1/4           //resistance in ohm
6  r3 = 1/5           //resistance in ohm
7
8  //Calculation
9  rt = r1+r2+r3
10 r = 1/rt           //total resistance in
    ohm
11
12
13 //Results
14 printf("Rt = %.3 f ohm",r)

```

Scilab code Exa 6.4 Example 4

```

1 //Chapter 6, Example 6.4
2 clc
3 //Variable Declaration
4 r1 = 12 //resistor in ohm fig
   6.9
5 n1 = 3 //number of component
6 r2 = 2 //resistor in ohm fig
   6.10
7 n2 = 4 //number of component
8 //Calculation
9 rt1 = r1/n1 //total resistance in
   ohm fig 6.9
10 rt2 = r2/n2 //total resistance in
   ohm fig 6.10
11
12 //Results
13 printf("a. Rt = %d ohm \n",rt1)
14 printf("b. Rt = %.1 f ohm",rt2)

```

Scilab code Exa 6.5 Example 5

```

1 //Chapter 6, Example 6.5
2 clc
3 //Variable Declaration
4 r1 = 3 //resistor in ohm
5 r2 = 6 //resistor in ohm
6 //Calculation
7 rt = r1*r2/(r1+r2) //resistance in ohm
8
9 //Results
10 printf("Rt = %d ohm",rt)

```

Scilab code Exa 6.6 Example 6


```

1 //Chapter 6, Example 6.6
2 clc
3 //Variable Declaration
4 r1 = 2 //resistor in ohm
5 r2 = 4 //resistor in ohm
6 r3 = 5 //resistor in ohm
7
8 //Calculation
9 rt = 1/((1/r1)+(1/r2)+(1/r3)) //total resistance
    in ohm
10 rt1 = (r1*r2)/(r1+r2) //R1 and R2
    parallel
11 rt2 = (rt1*r3)/(rt1+r3) //Rt1 and R3
    parallel
12
13 //Results
14 printf("Rt = %.3f ohm \n",rt)
15 printf("Rt = %.3f ohm \n",rt2)

```

Scilab code Exa 6.7 Example 7

```

1 //Chapter 6, Example 6.7
2 clc
3 //Variable Declaration
4 r = 6 //resistor in ohm
5 n = 3 //number of component
6 r2 = 9 //resistor in ohm
7 r4 = 72 //resistor in ohm
8
9 //Calculation
10 rt1 = r/n //resistance in ohm
11 rt2 = (r2*r4)/(r2+r4) //R2 and R4 parallel
12 rt3 = (rt1*rt2)/(rt1+rt2) //total resistance in
    ohm
13

```

```
14
15 //Results
16 printf("Rt = %.1 f ohm",rt3)
```

Scilab code Exa 6.8 Example 8

```
1 //Chapter 6, Example 6.8
2 clc
3 //Variable Declaration
4 rt = 9000 //total resistance in
   ohm
5 r1 = 12000 //resistor in ohm
6
7 //Calculation
8 r2 = (rt*r1)/(r1-rt) //required resistor in
   ohm
9
10 //Results
11 printf("R2 = %d kohm",r2/10^3)
```

Scilab code Exa 6.9 Example 9

```
1 //Chapter 6, Example 6.9
2 clc
3 //Variable Declaration
4 rt = 16000 //total resistance in
   ohm
5 r = 1.75
6
7
8 //Calculation
9 r1 = r*rt //required resistance
   in ohm
```

```

10
11 //Results
12 printf("R1 = %d kohm",r1/10^3)

```

Scilab code Exa 6.10 Example 10

```

1 //Chapter 6, Example 6.10
2 clc
3 //Variable Declaration
4 r1 = 30 //resistance in ohm
   fig 6.18
5 r2 = 30 //resistance in ohm
   fig 6.18
6 r3 = 30 //resistance in ohm
   fig 6.19
7 r4 = 1000 //resistance in ohm
   fig 6.19
8 r5 = 0.1 //resistance in ohm
   fig 6.20
9
10 //Calculation
11 rt = (r1*r2)/(r1+r2) //total
   resistance in ohm
12 rt1 = (rt*r3)/(rt+r3) //total
   resistance in ohm
13 rt2 = (rt*r4)/(rt+r4) //total
   resistance in ohm
14 rt3 = (rt*r5)/(rt+r5) //total
   resistance in ohm
15 //Results
16 printf("a. Rt = %d ohm \n",rt)
17 printf("b. Rt = %d ohm \n",rt1)
18 printf("c. Rt = %.3 f ohm \n",rt2)
19 printf("d. Rt = %.3 f ohm",rt3)

```

Scilab code Exa 6.11 Example 11

```
1 //Chapter 6, Example 6.11
2 clc
3 //Variable Declaration
4 r1 = 9 //resistance in ohm
5 r2 = 18 //resistance in ohm
6 e = 27 //voltage supply
7
8 //Calculation
9 rt = (r1*r2)/(r1+r2) //total resistance
   in ohm
10 is = e/rt //source current in
   ampere
11 i1 = e/r1 //current in ampere
12 i2 = e/r2 //current in ampere
13 is1 = i1+i2 //applying KCL
14 p1 = e*i1 //power of R1 in
   watt
15 p2 = e*i2 //power of R2 in
   watt
16 ps = e*is //power supply in
   watt
17 ps1 = p1+p2 //power supply in
   watt
18
19 //Results
20 printf("a. Rt = %d ohm \n",rt)
21 printf("b. Is = %.1 f A \n",is)
22 printf("c. I1 = %d A \n",i1)
23 printf(" I2 = %.1 f A \n",i2)
24 printf(" %.1 f A = %.1 f A \n",is1,is)
25 printf("d. P1 = %d W \n",p1)
26 printf(" I2 = %.1 f W \n",p2)
```

```
27 printf("e. Ps = %.1 f W \n", ps)
28 printf(" Ps = %.1 f W \n", ps1)
```

Scilab code Exa 6.12 Example 12

```
1 //Chapter 6, Example 6.12
2 clc
3 //Variable Declaration
4 r1 = 10 //resistance in ohm
5 r2 = 20 //resistance in ohm
6 rt = 4 //total resistance in
  ohm
7 i1 = 4 //current of R1 in
  ampere
8
9
10 //Calculation
11 r = (1/rt)-(1/r1)-(1/r2)
12 r3 = (1/r) //resistance in
  ohm
13 e = i1*r1 //supply voltage
14 is = e/rt //source current
  in ampere
15 i2 = e/r2 //current of R2
  in ampere
16 p2 = i2*i2*r2 //power of R2 in
  watt
17
18 //Results
19 printf("a. R3 = %d ohm \n", r3)
20 printf("b. E = %d V \n", e)
21 printf("c. Is = %d A \n", is)
22 printf("d. I2 = %d A \n", i2)
23 printf("e. P2 = %d W", p2)
```

Scilab code Exa 6.13 Example 13

```
1 //Chapter 6, Example 6.13
2 clc
3 //Variable Declaration
4 i1 = 2 //current in ampere
5 i2 = 3 //current in ampere
6 i5 = 1 //current in ampere
7
8 //Calculation
9 i3 = i1+i2 //current in ampere
10 i4 = i3+i5 //current in ampere
11
12 //Results
13 printf("a: I3 = %d A \n",i3)
14 printf("b: I4 = %d A",i4)
```

Scilab code Exa 6.14 Example 14

```
1 //Chapter 6, Example 6.14
2 clc
3 //Variable Declaration
4 i = 5 //entering current in
    ampere
5 i2 = 4 //leaving current in
    ampere
6
7 //Calculation
8 i1 = i-i2 //applying KCL, in
    ampere
9 i3 = i1 // in ampere
10 i4 = i2 // in ampere
```

```

11 i5 = i3+i4                                //applying KCL, in
    ampere
12 //Results
13 printf(" I1 = %d A \n",i1)
14 printf(" I3 = %d A \n",i3)
15 printf(" I4 = %d A \n",i4)
16 printf(" I5 = %d A",i5)

```

Scilab code Exa 6.15 Example 15

```

1 //Chapter 6, Example 6.15
2 clc
3 //Variable Declaration
4 i1 = 4                                     //current in ampere
5 i2 = 3                                     //current in ampere
6 i4 = 1                                     //current in ampere
7
8 //Calculation
9 i3 = i1+i2                                 //current in ampere
10 i5 = i3-i4                                //current in ampere
11 //Results
12 printf(" I3 = %d A \n",i3)
13 printf(" I5 = %d A",i5)

```

Scilab code Exa 6.16 Example 16

```

1 //Chapter 6, Example 6.16
2 clc
3 //Variable Declaration
4 i1 = 10                                    //current in ampere
5 i2 = 12                                    //current in ampere
6 i5 = 8                                    //current in ampere
7

```

```

8 //Calculation
9 i7 = i1
10 i3 = i2-i1 //current in ampere
11 i4 = i2-i5 //current in ampere
12 i6 = i4-i3 //current in ampere
13 i71 = i5+i6 //current in ampere
14 //Results
15 printf(" I7 = %d A \n",i7)
16 printf(" I3 = %d A \n",i3)
17 printf(" I4 = %d A \n",i4)
18 printf(" I6 = %d A \n",i6)
19 printf(" %d A = %d A \n",i7,i71)

```

Scilab code Exa 6.17 Example 17

```

1 //Chapter 6, Example 6.17
2 clc
3 //Variable Declaration
4 r1 = 4000 //resistance
5 is = 6 //source current in
   ampere
6 r2 = 8000 //resistance in ohm
7
8 //Calculation
9 i2 = (r1*is)/(r1+r2) //current in ampere
10
11 //Results
12 printf(" I2 = %d A",i2)

```

Scilab code Exa 6.18 Example 18

```

1 //Chapter 6, Example 6.18
2 clc

```



```

3 //Variable Declaration
4 r1 = 6 //resistance in ohm
5 r2 = 24 //resistance in ohm
6 r3 = 48 //resistance in ohm
7 i = 42*10^-3 //current in ampere
8
9 //Calculation
10 r = (1/r1)+(1/r2)+(1/r3)
11 rt = 1/r //total resistance in
    ohm
12 i1 = (rt*i)/r1 //current in ampere
13 r4 = (r2*r3)/(r2+r3) //resistance in ohm
14 i11 = (r4*i)/(r4+r1) //current in ampere
15
16 //Results
17 printf("I1 = %.2f mA \n",i1*1000)
18 printf("I1 = %.2f mA",i11*1000)

```

Scilab code Exa 6.19 Example 19

```

1 //Chapter 6, Example 6.19
2 clc
3 //Variable Declaration
4 r1 = 2 //resistance in ohm
5 r2 = 4 //resistance in ohm
6 i = 12 //current in ampere
7
8 //Calculation
9 i1 = (r2*i)/(r1+r2) //current in ampere
10 i2 = i-i1 //current in ampere
11 i21 = (r1*i)/(r1+r2) //current in ampere
12 i3 = i //current in ampere
13 i31 = i1+i2 //current in ampere
14
15 //Results

```

```

16 printf(" I1 = %d A \n", i1)
17 printf(" I2 = %d A \n", i2)
18 printf(" I2 = %d A \n", i21)
19 printf(" I3 = %d A \n", i3)
20 printf(" I3 = %d A \n", i31)

```

Scilab code Exa 6.20 Example 20

```

1 //Chapter 6, Example 6.20
2 clc
3 //Variable Declaration
4 r2 = 7 //resistance in ohm
5 i = 27*10^-3 //current in ampere
6 i1 = 21*10^-3 //current in ampere
7
8
9 //Calculation
10 r1 = (r2*(i-i1))/i1 //resistance in ohm
11 i2 = i-i1 //current in ampere
12 v2 = i2*r2 //voltage in volt
13 v1 = v2 //voltage in volt
14 r11 = v1/i1 //resistance in ohm
15
16 //Results
17 printf("R1 = %.f ohm \n", r1)
18 printf("R1 = %.f ohm", r11)

```

Scilab code Exa 6.21 Example 21

```

1 //Chapter 6, Example 6.21
2 clc
3 //Variable Declaration
4 e = 20 //supply voltage

```

```

5
6 //Calculation
7 vab = e //output Vab
8
9 //Results
10 printf("Vab = %d V",vab)

```

Scilab code Exa 6.22 Example 22

```

1 //Chapter 6, Example 6.22
2 clc
3 //Variable Declaration
4 e1 = 10 //supply voltage
5 e2 = 30 //supply voltage
6
7 //Calculation
8 vab = e1 //voltage Vab
9 vcd = e1-e2 //voltage Vcd
10
11 //Results
12 printf("Vab = %d V \n",vab)
13 printf("Vcd = %d V",vcd)

```

Scilab code Exa 6.23 Example 23

```

1 //Chapter 6, Example 6.23
2 clc
3 //Variable Declaration
4 i = 0 //current in ampere
5 r = 15000 //resistance in ohm
6
7 //Calculation
8 v = i*r //voltage

```

```
9
10 //Results
11 printf("I = %d A \n",i)
12 printf("V = %d V",v)
```

Scilab code Exa 6.24 Example 24

```
1 //Chapter 6, Example 6.24
2 clc
3 //Variable Declaration
4 e = 18 //supply voltage
5 r1 = 5000 //resistance in ohm
6
7 //Calculation
8 i = e/r1 //current in ampere
9 v = e //voltage
10 //Results
11 printf("I = %.1f mA \n",i*1000)
12 printf("V = %d V",v)
```

Scilab code Exa 6.25 Example 25

```
1 //Chapter 6, Example 6.25
2 clc
3 //Variable Declaration
4 v = 0 //voltage
5 e = 6 //supply voltage
6 r1 = 2 //resistance in ohm
7
8 //Calculation
9 i = e/r1 //current in ampere
10
11 //Results
```

```
12 printf("V = %d V \n",v)
13 printf("I = %d A",i)
```

Scilab code Exa 6.26 Example 26

```
1 //Chapter 6, Example 6.26
2 clc
3 //Variable Declaration
4 e = 20 //supply voltage
5 r = 1*10^6 //resistance in ohm
6 rab1 = 11*10^6 //load resistance
   in ohm
7 rm = 2*10^6
8 //Calculation
9 vab = e //voltage Vab
10 vab2 = (rab1*e)/(rab1+r) //Vab by voltage
   divider rule
11 vab3 = (rm*e)/(rm+r) //Vab for the
   internal resistance
12
13 //Results
14 printf("a. Vab = %d V \n",vab)
15 printf("b. Vab = %.2f V \n",vab2)
16 printf("c. Vab = %.2f V \n",vab3)
```

Chapter 7

Series Parallel Networks

Scilab code Exa 7.1 Example 1

```
1 //Chapter 7, Example 7.1
2 clc
3 //Variable Declaration
4 rb = 12000 //resistance in ohm
5 rc = 6000 //resistance in ohm
6 e = 54 //voltage
7 ra = 2000 //resistance in ohm
8 //Calculation
9 rbc = (rb*rc)/(rb+rc) //RB and RC are
    parallel
10 rt = ra+rbc //the equivalent
    resistance in ohm
11 is = e/rt //source current Is in
    ampere
12 ia = is
13 ib = (rc*is)/(rb+rc) //applying current
    divider rule
14 ic = (rb*is)/(rb+rc) //applying current
    divider rule
15 ic1 = is-ib //applying KCL
16
```

```

17
18 //Results
19 printf("Rbc = %d kohm \n",rbc/10^3)
20 printf("Rt = %d kohm \n",rt/10^3)
21 printf("Ia = %d mA \n",ia*1000)
22 printf("Ib = %.f mA \n",ib*1000)
23 printf("Ic = %.f mA \n",ic*1000)
24 printf("Ic = %.f mA \n",ic1*1000)

```

Scilab code Exa 7.2 Example 2

```

1 //Chapter 7, Example 7.2
2 clc
3 //Variable Declaration
4 r1 = 4 //resistance in ohm
5 r2 = 4 //resistance in ohm
6 r3 = 4 //resistance in ohm
7 r4 = 0.5 //resistance in ohm
8 r5 = 1.5 //resistance in ohm
9 e = 10 //voltage
10 is = 2 //current through R1 in
    ampere
11
12 //Calculation
13 ra = r1 //RA is R1 in ohms
14 rb = (r2*r3)/(r2+r3) //RB is parallel
    combination of R2 and R3 in ohms
15 rc = r4+r5 //RC is equivalent
    resistor of R4 and R5 in ohms
16 rbc = (rb*rc)/(rb+rc) //parallel combination
    of RB and RC in ohms
17 rt = ra+rbc //resistance in ohm
18 ia = is //same current in ampere
19 ib = ia/2 //current in ampere
20 ic = ib //current in ampere

```

```

21 ir2 = ib/2           //current in ampere
22 ir3 = ir2           //current in ampere
23 va = ia*ra          //voltage
24 vb = ib*rb          //voltage
25 vc = vb             //voltage
26 v = e-va-vb        //applying KVL
27
28 //Results
29 printf("Rt = %d ohm \n",rt)
30 printf("Is = %d A \n",is)
31 printf("Ia = %d A \n",ia)
32 printf("Ib = %d A \n",ib)
33 printf("Ir2 = Ir3 = %.1f A \n",ir2)
34 printf("Va = %d V \n",va)
35 printf("Vb = %d V \n",vb)
36 printf("Vc = %d V \n",vc)

```

Scilab code Exa 7.3 Example 3

```

1 //Chapter 7, Example 7.3
2 clc
3 //Variable Declaration
4 r1 = 9           //resistance in ohms
5 r2 = 6           //resistance in ohms
6 r3 = 6           //resistance in ohms
7 r4 = r3          //resistance in ohms
8 r5 = 3           //resistance in ohms
9 r6 = 3           //resistance in ohms
10 e = 16.8        //source voltage
11
12 //Calculation
13 ra = (r1*r2)/(r1+r2) //parallel R1 and R2
    in ohms
14 rb = r3 + ((r4*r5)/(r4+r5)) //resistance in ohms
15 rc = r6         //resistance in ohms

```



```

16 rt = ra+((rb*rc)/(rb+rc)) //total resistance in
    ohms
17 is = e/rt //current in ampere
18 ia = is //current in ampere
19 ib = (rc*ia)/(rc+rb) //applying the current
    divider rule
20 ic = ia-ib //by KCL, current in
    ampere
21 va = ia*ra //by ohm's law ,
    voltage
22 vb = ib*rb //by ohm's law ,
    voltage
23 i1 = (r2*ia)/(r2+r1) //applying the current
    divider rule
24 i2 = ia-i1 //current in ampere
25
26 //Results
27 printf("Rt = %.1f ohm \n",rt-0.1)
28 printf("Is = %d A \n",is+0.1)
29 printf("Ia = Is = %d A \n",ia+0.1)
30 printf("Ib = %d A \n",ib+0.3)
31 printf("Ic = %d A \n",ic)
32 printf("Va = %.1f A \n",va)
33 printf("Vb = %d A \n",vb)
34 printf("I1 = %.1f A \n",i1)
35 printf("I2 = %.1f A",i2)

```

Scilab code Exa 7.4 Example 4

```

1 //Chapter 7, Example 7.4
2 clc
3 //Variable Declaration
4 e = 12 //voltage
5 r4 = 8 //resistance in ohms
6 r2 = 3 //resistance in ohms

```

```

7 r3 = 6 //resistance in ohms
8 r1 = 4 //resistance in ohms
9 rc = r1 //resistance in ohms
10
11 //Calculation
12 i4 = e/r4 //current through R4 in
    ampere
13 rd = (r2*r3)/(r2+r3) //resistance in ohms
14 v2 = (rd*e)/(rd+rc) //applying voltage
    divider rule
15
16 //Results
17 printf("I4 = %.1f A \n", i4)
18 printf("V2 = %d V", v2)

```

Scilab code Exa 7.5 Example 5

```

1 //Chapter 7, Example 7.5
2 clc
3 //Variable Declaration
4 r1 = 6 //resistance in ohms
5 r2 = 6 //resistance in ohms
6 r3 = 2 //resistance in ohms
7 r4 = 8 //resistance in ohms
8 r5 = 12 //resistance in ohms
9 e = 24 //voltage
10
11 //Calculation
12 r = (r1*r2)/(r1+r2) //parallel combination
    of R1 and R2
13 ra = (r*r3)/(r+r3) //resistance A
    combination in ohm
14 rb = (r4*r5)/(r4+r5) //resistance B
    combination in ohm
15 rt = ra+rb //total resistance in

```

```

    ohms
16 is = e/rt //current
17 v1 = is*ra //voltage across R'A'
18 v5 = is*rb //voltage across R'B'
19 i4 = v5/r4 //current in ampere
    through R'B'
20 i2 = v1/r2 //current in ampere
    through R2
21
22 //Results
23 printf("Rt = %d ohm \n",rt)
24 printf("Is = %d A \n",is)
25 printf("V1 = %.1 f V \n",v1)
26 printf("V5 = %.1 f V \n",v5)
27 printf("I4 = %.1 f A \n",i4)
28 printf("I2 = %.1 f A \n",i2)

```

Scilab code Exa 7.6 Example 6

```

1 //Chapter 7, Example 7.6
2 clc
3 //Variable Declaration
4 r1 = 5 //resistance in ohms
5 r2 = 3 //resistance in ohms
6 r3 = 6 //resistance in ohms
7 r4 = 2 //resistance in ohms
8 e = 12 //voltage
9
10 //Calculation
11 v1 = (r1*e)/(r1+r2) //voltage divider rule
12 v3 = (r3*e)/(r3+r4) //voltage divider rule
13 vab = v3-v1 //applying KVL, in
    voltage
14 i1 = v1/r1 //by ohm's law, in
    amperes

```

```

15 i3 = v3/r3           //by ohm's law, in
    amperes
16 is = i1+i3         //applying KCL, in
    amperes
17
18 //Results
19 printf("a. V1 = %.1f V \n",v1)
20 printf("   V3 = %d V \n",v3)
21 printf("   Vab = %.1f V \n",vab)
22 printf("b. Is = %d A \n",is)

```

Scilab code Exa 7.7 Example 7

```

1 //Chapter 7, Example 7.7
2 clc
3 //Variable Declaration
4 e1 = 6           //voltage
5 e2 = 18         //voltage
6 r1 = 6           //resistance in ohms
7 r2 = 5           //resistance in ohms
8 r3 = 7           //resistance in ohms
9 r4 = 6           //resistance in ohms
10
11 //Calculation
12 v2 = -e1         //in voltage
13 v1 = e2+e1       //total voltage
14 i1 = v1/r1       //by ohm's law, in amperes
15 i2 = e1/r4       //by ohm's law, in amperes
16 i3 = e1/(r2+r3) //by ohm's law, in amperes
17 i = i1+i2+i3    //applying KCL, in amperes
18
19 //Results
20 printf("V2 = %d V \n",v2)
21 printf("V1 = %d V \n",v1)
22 printf("I = %.1f A",i)

```

Scilab code Exa 7.8 Example 8

```
1 //Chapter 7, Example 7.8
2 clc
3 //Variable Declaration
4 r1 = 40*103 //resistance in ohms
5 r2 = 4*103 //resistance in ohms
6 rc = 10*103 //resistance in ohms 'C'
7 re = 1*103 //resistance in ohms 'E'
8 vcc = 22 //power supply Vcc
9 vb = 2 //base voltage
10 vbe = 0.7 //base emitter voltage
11
12 //Calculation
13 v2 = vb
14 ve = v2-vbe //applying KVL to lower
    loop
15 ie = ve/re //by ohm's law, in ampere
16 v1 = vcc-v2 //applying KVL to upper
    loop
17 ic = ie // in ampere
18 vc = vcc-(ie*rc) //applying KVL , in
    voltaage
19 vbc = vb-vc //in voltaage
20 vce = vc-ve //in voltaage
21
22 //Results
23 printf("Ve = %.1 f V \n",ve)
24 printf("Ie = %.1 f mA \n",ie*1000)
25 printf("V1 = %d V \n",v1)
26 printf("Vbc = %d V \n",vbc)
27 printf("Vce = %.1 f V",vce)
```

Scilab code Exa 7.9 Example 9

```
1 //Chapter 7, Example 7.9
2 clc
3 //Variable Declaration
4 r123 = 24*10^3 //combining resistance
   1,2,3 in ohms
5 r4 = 24*10^3 //resistance in ohms
6 r5 = 12*10^3 //resistance in ohms
7 r6 = 12*10^3 //resistance in ohms
8 r7 = 9*10^3 //resistance in ohms
9 r89 = 9*10^3 //combining resistance
   8,9 in ohms
10 e = 72 //voltage
11
12 //Calculation
13 r = (r123*r4)/(r123+r4) //parallel resistance
   R123 and R4 in ohm
14 i5 = e/(r+r5) //by ohm's law, in
   ampere
15 rr = (r7*r89)/(r7+r89) //parallel resistor R7
   and R89, in ohms
16 v7 = (rr*e)/(rr+r6) //voltage divider rule,
   in volts
17 i6 = v7/rr //current through rr,
   in ampere
18 is = i5+i6 //soucre current by KCL
   , in ampere
19
20 //Results
21 printf(" I5 = %d mA \n", i5*1000)
22 printf(" V7 = %.1 f V \n", v7)
23 printf(" I6 = %.2 f mA \n", i6*1000)
24 printf(" Is = %.2 f mA \n", is*1000)
```

Scilab code Exa 7.10 Example 10

```
1 //Chapter 7, Example 7.10
2 clc
3 //Variable Declaration
4 e1 = 20 //emf in voltage
5 e2 = 5 //emf in voltage
6 e3 = 8 //emf in voltage
7
8
9 //Calculation
10 v1 = e1-e3 //voltage
11 v2 = e2-v1 //voltage
12 v3 = e3-v2 //voltage
13
14 //Results
15 printf("V1 = %d V \n",v1)
16 printf("V2 = %d V \n",v2)
17 printf("V3 = %d V \n",v3)
```

Scilab code Exa 7.11 Example 11

```
1 //Chapter 7, Example 7.11
2 clc
3 //Variable Declaration
4 e = 72 //emf in voltage
5 is = 50*10^-3 //current source in ampere
6 vr3 = 12 //R3 voltage
7 vl1 = 60 //RL1 voltage
8 vl2 = 20 //RL2 voltage
9 ir1= 30*10^-3 //R1 current in ampere
10 ir2 = 20*10^-3 //R2 current in ampere
```

```

11 vr2 = 20 //R2 voltage
12
13 // Calculation
14 r3 = vr3/is //resistance in ohms
15 vr1 = v11-v12 //voltage
16 r1 = vr1/ir1 //resistance in ohms
17 r2 = vr2/ir2 //resistance in ohms
18
19 // Results
20 printf("R3 = %d ohm \n",r3)
21 printf("R1 = %.2f Kohm \n",r1/1000)
22 printf("R2 = %d kohm",r2/1000)

```

Scilab code Exa 7.12 Example 12

```

1 //Chapter 7, Example 7.12
2 clc
3 //Variable Declaration
4 e = 120 //voltage supply
5 r1 = 4000 //resistance in ohms
6 r2 = 6000 //resistance in ohms
7 r3 = 12000 //resistance in ohms
8 r4 = 30000 //resistance in ohms
9 r5 = 10000 //resistance in ohms
10
11 // Calculation
12 r = (r1*r3)/(r1+r3) //parallel combination R1
    and R3
13 rr = (r2*r4)/(r2+r4) //parallel combination R2
    and R4
14 v1 = (r*e)/8000 //voltage
15 v2 = (rr*e)/8000 //voltage
16
17 // Results
18 printf("V1 = %d V \n",v1)

```



```
19 printf("V2 = %d V \n", v2)
```

Chapter 8

Methods of Analysis and Selected Topics dc

Scilab code Exa 8.1 Example 1

```
1 //Chapter 8, Example 8.1
2 clc
3 //Variable Declaration
4 i = 10*10-3 //source current
5 r = 20*103 //resistance in ohms
6
7 //Calculation
8 i1 = i //current in ampere
9 v1 = i1*r //by ohm's law, in volts
10 vs = v1 //in volts
11
12 //Results
13 printf("I1 = %d mA \n", i1*1000)
14 printf("Vs = %d V", vs)
```

Scilab code Exa 8.2 Example 2

```

1 //Chapter 8, Example 8.2
2 clc
3 //Variable Declaration
4 e = 12 //voltage
5 r = 4 //resistance in ohm
6 i = 7 //current source in
    ampere
7
8 //Calculation
9 vs = e //voltage = Vs
10 i2 = e/r //by ohm's law, in
    amperes
11 i1 = i-i2 //applying kirchoff's
    current law
12
13 //Results
14 printf("Vs = %d V \n",vs)
15 printf("I2 = %d A \n",i2)
16 printf("I1 = %d A",i1)

```

Scilab code Exa 8.3 Example 3

```

1 //Chapter 8, Example 8.3
2 clc
3 //Variable Declaration
4 r2 = 1 //resistance in ohms
5 i = 6 //current in amperes
6 r1 = 2 //resistance in ohms
7
8 //Calculation
9 i1 = (r2*i)/(r2+r1) //applying the current
    divider rule
10 v1 = i1*r1 //by ohm's law, in volts
11 vs = v1+20 //applying kirchoff's
    current law

```

```

12
13 //Results
14 printf("I1 = %d A \n",i1)
15 printf("Vs = %d V",vs)

```

Scilab code Exa 8.4 Example 4

```

1 //Chapter 8, Example 8.4
2 clc
3 //Variable Declaration
4 e = 6 //voltage
5 rs = 2 //resistance in ohm
6 rl = 4 //load resistance in
    ohm
7 i = 3 //source current in
    ampere
8 rl2 = 1000 //replace load
    resistance in ohm
9
10 //Calculation
11 il = e/(rs+rl) //current in
    ampere
12 il1 = (rs*i)/(rs+rl) //applying the current
    divider rule
13 il2 = e/(rs+rl2) //current in ampere
14 il3 = e/rl2 //current in ampere
15
16 //Results
17 printf("a. IL = %d A \n",il)
18 printf(" IL = %d A \n",il1)
19 printf("b. IL = %.2 f mA \n",il2*1000)
20 printf("c. IL = %d mA",il3*1000)

```

Scilab code Exa 8.5 Example 5

```
1 //Chapter 8, Example 8.5
2 clc
3 //Variable Declaration
4 rs = 3000 //resistance in ohm
5 i = 9*10^-3 //source current in
    ampere
6 r1 = 6000 //load resistance in
    ohm
7 e = 27 //voltage
8 r12 = 10 //replace load
    resistance in ohm
9
10 //Calculation
11 il = (rs*i)/(rs+r1) //applying the current
    divider rule
12 il1 = e/(rs+r1) //current in ampere
13 il2 = (rs*i)/(rs+r12) //applying the current
    divider rule
14 il3 = i // current in ampere
15
16 //Results
17 printf("a. IL = %d mA \n",il*1000)
18 printf(" IL = %d mA \n",il1*1000)
19 printf("b. IL = %.2f mA \n",il2*1000)
20 printf("c. IL = %d mA",il3*1000)
```

Scilab code Exa 8.6 Example 6

```
1 //Chapter 8, Example 8.6
2 clc
3 //Variable Declaration
4 i1 = 6 //current in ampere figs.
    8.11
```

```

5 i2 = 10 //current in ampere figs.
   8.11
6 r1 = 3 //resistance in ohms figs.
   8.11
7 r2 = 6 //resistance in ohms figs.
   8.11
8 i3 = 7 //current in ampere figs.
   8.12
9 i4 = 3 //current in ampere figs.
   8.12
10 i5 = 4 //current in ampere figs.
    8.12
11 r3 = 4 //resistance in ohms figs.
    8.12
12
13
14 //Calculation
15 is = i2-i1 //current in ampere
16 rs = (r1*r2)/(r1+r2) //R1 and R2 are parallel
17 is1 = i3+i5-i4 //current in ampere
18 rs1 = r3 //in ohms
19
20 //Results
21 printf("Is = %d A \n",is)
22 printf("Rs = %d ohm \n",rs)
23 printf("Is = %d A \n",is1)
24 printf("Rs = %d ohm \n",rs1)

```

Scilab code Exa 8.7 Example 7

```

1 //Chapter 8, Example 8.7
2 clc
3 //Variable Declaration
4 i1 = 4 //current in ampere
5 i2 = 6 //current in ampere

```

```

6 r1 = 8 //resistance in ohm
7 r2 = 24 //resistance in ohm
8 rl = 14 //load resistance in ohm
9
10 //Calculation
11 is = i1+i2 //combining current in ampere
12 rs = (r1*r2)/(r1+r2) //R1 and R2 are parallel
13 il = (rs*is)/(rs+rl) //applying current divider
    rule
14
15 //Results
16 printf("Is = %d A \n",is)
17 printf("Rs = %d ohm \n",rs)
18 printf("IL = %d A",il)

```

Scilab code Exa 8.8 Example 8

```

1 //Chapter 8, Example 8.8
2 clc
3 //Variable Declaration
4 i1 = 4 //current in ampere
5 r1 = 3 //resistance in ohm
6 e2 = 5 //voltage
7 r2 = 2 //resistance in ohm
8
9 //Calculation
10 es = i1*r1 //by ohm's law
11 rs = r1 //equal
12 i2 = (es+e2)/(rs+r2)
13
14 //Results
15 printf("I2 = %.1f A ",i2)

```

Scilab code Exa 8.9 Example 9

```
1 //Chapter 8, Example 8.9
2 clc
3 //Variable Declaration
4
5 //three equations substituting the coefficients to
   variables
6 i11=2
7 i12=0
8 i13=4
9 i1=2
10
11 i21=0
12 i22=1
13 i23=4
14 i2=6
15
16 i31=1
17 i32=1
18 i33=-1
19 i3=0
20
21 //Calculation
22
23 //for I1
24 d=[i11 i12 i13;i21 i22 i23;i31 i32 i33]
25 di1=[i1 i12 i13;i2 i22 i23;i3 i32 i33]
26 ia1=det(di1)/det(d)
27
28 //for I2
29 di2=[i11 i1 i13 ; i21 i2 i23 ; i31 i3 i33]
30 ia2=det(di2)/det(d)
31
32 //for I3
33 di3=[i11 i12 i1 ; i21 i22 i2 ; i31 i32 i3]
34 ia3=det(di3)/det(d)
35
```



```
36 // Results
37 printf( 'I1 = %d A \n', ia1)
38 printf( 'I2 = %d A \n', ia2)
39 printf( 'I3 = %d A \n', ia3)
```

Chapter 9

Network Theorems

Scilab code Exa 9.1 Example 1

```
1 //Chapter 9, Example 9.1
2 clc
3 //Variable Declaration
4 rsc = 0
5 r1 = 6 //resistance in ohms
6 e = 30 //voltage
7 r1 = 6 //resistance in ohms
8
9
10 //Calculation
11 i1 = (rsc)/(rsc+r1) //applied the current
    divider rule, in amperes
12 i11 = e/r1 //applying ohm's law
13 i = i1+i11 //current in amperes
14
15 //Results
16 printf("I = %d A \n",i)
```

Scilab code Exa 9.2 Example 2

```

1 //Chapter 9, Example 9.2
2 clc
3 //Variable Declaration
4 r1 = 24 //resistance in ohms
5 r2 = 12 //resistance in ohms
6 r3 = 4 //resistance in ohms
7 e1 = 54 //voltage
8 e2 = 48 //voltage
9
10 //Calculation
11 rt = r1+(r2*r3)/(r2+r3) //total resistance in
    ohms
12 i = e1/rt //current in amperes in
    loop
13 i31 = (r2*i)/(r2+r3) //using current divider
    rule , in amperes
14 rt2 = r3+(r1*r2)/(r1+r2) //total resistance in
    ohms
15 i33 = e2/rt2 //current in amperes in
    loop
16 i3 = i33-i31 //total current in
    amperes through R3
17
18 //Results
19 printf("I3 = %.1 f A", i3)

```

Scilab code Exa 9.3 Example 3

```

1 //Chapter 9, Example 9.3
2 clc
3 //Variable Declaration
4 r1 = 12 //resistance in ohms
5 e = 36 //voltage
6 r2 = 6 //resistance in ohms
7 i = 9 //source current in amperes

```

```

8
9 //Calculation
10 i21 = e/(r1+r2) //current in amperes
11 i22 = (r1*i)/(r1+r2) //applying current in
    amperes divider rule
12 i2 = i21+i22 //total current in amperes
    through the R2
13 p = i2*i2*r2 //power in watt to R2
14
15 //Results
16 printf("I2 = %d A \n",i2)
17 printf("P = %d W \n",p)

```

Scilab code Exa 9.4 Example 4

```

1 //Chapter 9, Example 9.4
2 clc
3 //Variable Declaration
4 r1 = 6000 //resistance in ohms
5 r2 = 12000 //resistance in ohms
6 i = 6*10^-3 //current in amperes
7 e = 9 //voltage
8
9 //Calculation
10 i21 = (r1*i)/(r1+r2) //current in amperes
    divider rule
11 i22 = e/(r1+r2) //effect of 9-V
    voltage source
12 i2 = i21+i22 //sum of the current
    in amperes
13
14 //Results
15 printf("I2 = %.1 f mA", i2*1000)

```

Scilab code Exa 9.5 Example 5

```
1 //Chapter 9, Example 9.5
2 clc
3 //Variable Declaration
4 r1 = 2 //resistance in ohms
5 r2 = 4 //resistance in ohms
6 e1 = 12 //voltage
7 e2 = 6 //voltage
8 i = 3 //current in amperes
9
10
11 //Calculation
12 i11 = e1/(r1+r2) //effect of the 12-V
    source
13 i21 = e2/(r1+r2) //effect of the 6-V
    source
14 i23 = (r2*i)/(r1+r2) //applying the current
    divider rule, in amperes
15 i1 = i21+i23-i11 //current in
    amperes
16
17 //Results
18 printf("I1 = %d A \n",i1)
```

Scilab code Exa 9.6 Example 6

```
1 //Chapter 9, Example 9.6
2 clc
3 //Variable Declaration
4 e1= 9 //voltage
5 r1 = 3 //resistance in ohms
```

```

6 r2 = 6 //resistance in ohms
7 r11 =2 //resistance in ohms
8 r12 = 10 //resistance in ohms
9 r13 = 100 //resistance in ohms
10
11 //Calculation
12 rth = (r1*r2)/(r1+r2) //R1 and R2 are
    parallel
13 eth = (r2*e1)/(r2+r1) //applying voltage
    divider rule
14 il1 = eth/(rth+r11) //RL = 20 ohm
15 il2 = eth/(rth+r12) //RL = 10 ohm
16 il3 = eth/(rth+r13) //RL = 100 ohm
17
18 //Results
19 printf("Rth = %d ohm \n",rth)
20 printf("Eth = %d V \n",eth)
21 printf("IL = %.1 f A \n",il1)
22 printf("IL = %.1 f A \n",il2)
23 printf("IL = %.3 f A \n",il3)

```

Scilab code Exa 9.7 Example 7

```

1 //Chapter 9, Example 9.7
2 clc
3 //Variable Declaration
4 r1 = 4 //resistance in ohms
5 r2 = 2 //resistance in ohms
6 i1 = 12 //current in amperes source
7 i2 = 0
8
9 //Calculation
10 rth = r1+r2 //total resistance in ohms
11 eth = i1*r1 //voltage
12

```

```

13 //Results
14 printf("Rth = %d ohm \n",rth)
15 printf("Eth = %d V",eth)

```

Scilab code Exa 9.8 Example 8

```

1 //Chapter 9, Example 9.8
2 clc
3 //Variable Declaration
4 r1 = 6 //resistance in ohms
5 r2 = 4 //resistance in ohms
6 e1 = 8 //voltage
7
8 //Calculation
9 rth = (r1*r2)/(r1+r2) //R1 and R2 are parallel
10 eth = (r1*e1)/(r1+r2) //applying voltage divider
    rule
11
12 //Results
13 printf("Rth = %.1f ohm \n",rth)
14 printf("Eth = %.1f V",eth)

```

Scilab code Exa 9.9 Example 9

```

1 //Chapter 9, Example 9.9
2 clc
3 //Variable Declaration
4 r1 = 6 //resistance in ohms
5 r2 = 4 //resistance in ohms
6 r3 = 3 //resistance in ohms
7 r4 = 12 //resistance in ohms
8 e = 72 //voltage
9 r22 = 12 //resistance in ohms

```

```

10
11
12 //Calculation
13 rth = ((r1*r3)/(r1+r3))+((r2*r4)/(r2+r4)) //
    total resistance in ohms
14 v1 = (r1*e)/(r1+r3) //voltage
    divider rule R1
15 v2 = (r22*e)/(r2+r4) //voltage
    divider rule R2
16 eth = v2-v1 //applying
    kirchoff 's voltage law
17
18 //Results
19 printf("Rth = %d ohm \n",rth)
20 printf("Eth = %d V",eth)

```

Scilab code Exa 9.10 Example 10

```

1 //Chapter 9, Example 9.10
2 clc
3 //Variable Declaration
4 r1 = 0.8*10^3 //resistance in ohms
5 r2 = 4000 //resistance in ohms
6 r3 = 6000 //resistance in ohms
7 r4 = 1.4*10^3 //resistance in ohms
8 e1= 6 //voltage
9 e2 = 10 //voltage
10
11 //Calculation
12 r12 = (r1*r2)/(r1+r2) //R1 and R2 are
    parallel
13 rth = r4+((r12*r3)/(r12+r3)) //total
    resistance in ohm
14 r23 = (r2*r3)/(r2+r3) //R2 and R3 are
    parallel

```



```

15 v31 = (r23*e1)/(r23+r1)           //applying
    voltage divider rule R23
16 eth1 = v31                       //voltage
17 v32 = (r12*e2)/(r12+r2)         //applying
    voltage divider rule R12
18 eth2 = v32                       //voltage
19 eth = eth1-eth2                  //voltage
20
21 //Results
22 printf("Rth = %d kohm \n",rth/10^3)
23 printf("Eth = %d V",eth)

```

Scilab code Exa 9.11 Example 11

```

1 //Chapter 9, Example 9.11
2 clc
3 //Variable Declaration
4 r1 = 3                             //resistance in ohms
5 r2 = 6                             //resistance in ohms
6 e = 9                             //voltage
7
8 //Calculation
9 rn = (r1*r2)/(r1+r2)              //R1 and R2 are
    parallel
10 in = e/r1                        //current in amperes
11
12 //Results
13 printf("Rn = %d ohm \n",rn)
14 printf("In = %d A",in)

```

Scilab code Exa 9.12 Example 12

```

1 //Chapter 9, Example 9.12

```

```

2  clc
3  //Variable Declaration
4  r1 = 5           //resistance in ohms
5  r2 = 4           //resistance in ohms
6  i = 10          //current in amperes
7
8  //Calculation
9  rn = r1+r2      //total resistance in ohms
10 in = (r1*i)/(r1+r2) //applying the current in
    amperes divider rule
11
12 //Results
13 printf("Rn = %d ohm \n",rn)
14 printf("In = %.3 f A",in)

```

Scilab code Exa 9.13 Example 13

```

1  //Chapter 9, Example 9.13
2  clc
3  //Variable Declaration
4  r1 = 4           //resistance in ohms
5  r2 = 6           //resistance in ohms
6  r3 = 9           //resistance in ohms
7  r4 = 10          //resistance in ohms
8  e1 = 7           //voltage
9  e2 = 12          //voltage
10 i = 8            //current source in
    amperes
11
12 //Calculation
13 rn = (r1*r2)/(r1+r2) //R1 and R2 are
    parallel
14 in1= e1/r1        //current in amperes
15 in2 = i           //short circuited
16 in = in2-in1     //in amperes

```

```

17
18 //Results
19 printf("Rn = %.1 f ohm \n",rn)
20 printf("In = %.2 f A \n",in)

```

Scilab code Exa 9.14 Example 14

```

1 //Chapter 9, Example 9.14
2 clc
3 //Variable Declaration
4 rint1 = 2.5 //resistance in ohms
   of dc generator
5 rint2 = 0.5 //resistance in ohms
   of battery
6 rint3 = 40 //resistance in ohms
   of laboratory
7 n = 0.75 //efficiency in
   decimal form
8
9 //Calculation
10 r11 = rint1 //for the dc generator
   , in ohms
11 r12 = rint2 //for the battery, in
   ohms
12 r13 = rint3 //for the laboratory
   supply, in ohms
13 R11 = (n*r11)/(1-n) //RL dc gen. for
   efficiency, in ohms
14 R12 = (n*r12)/(1-n) //RL battery for
   efficiency, in ohms
15 R13 = (n*r13)/(1-n) //RL laboratory for
   efficiency, in ohms
16
17 //Results
18 printf("a. RL = %.1 f ohm \n",r11)

```

```
19 printf("    RL = %.1 f ohm \n",r12)
20 printf("    RL = %d ohm \n",r13)
21 printf("b. RL = %.1 f ohm \n",R11)
22 printf("    RL = %.1 f ohm \n",R12)
23 printf("    RL = %d ohm \n",R13)
```

Scilab code Exa 9.15 Example 15

```
1 //Chapter 9, Example 9.15
2 clc
3 //Variable Declaration
4 rs = 40*10^3 //resistance in ohms
5 in =10*10^-3 //current in amperes
   source
6
7 //Calculation
8 r1 = rs //load resistance in ohms
9 pl = (in*in*rs)/4 //maximum power in watt
10
11 //Results
12 printf("RL = %d kohm \n",r1/10^3)
13 printf("PLmax = %d W \n",pl)
```

Scilab code Exa 9.16 Example 16

```
1 //Chapter 9, Example 9.16
2 clc
3 //Variable Declaration
4 r1 = 6 //resistance in ohms
5 r2 = 3 //resistance in ohms
6 r3 = 8 //resistance in ohms
7 e = 12 //voltage
8
```

```

9 //Calculation
10 rth = r3+(r1*r2)/(r1+r2) //total resistance
    in ohms
11 eth = (r2*e)/(r2+r1) //voltage divider
12 PLmax = (eth*eth)/(4*rth) //maximum power in
    watt
13
14 //Results
15 printf("Rth = %d ohm \n",rth)
16 printf("Eth = %d V \n",eth)
17 printf("PLmax = %.1 f W",PLmax)

```

Scilab code Exa 9.17 Example 17

```

1 //Chapter 9, Example 9.17
2 clc
3 //Variable Declaration
4 r1 = 3 //resistance in ohms
5 r2 = 10 //resistance in ohms
6 r3 = 2 //resistance in ohms
7 e = 68 //voltage
8 i = 6 //current in amperes
9
10 //Calculation
11 rth = r1+r2+r3 //total resistance in
    ohms
12 rl = rth //load resistance in
    ohms
13 v2 = i*r2 //voltage
14 eth = v2 + e //applying kirchoff's
    voltage law
15 plmax = (eth*eth)/(4*rth) //maximum power in watt
16
17 //Results
18 printf("Rth = %d ohm \n",rth)

```

```
19 printf("PLmax = %.2 f W" ,plmax)
```

Scilab code Exa 9.18 Example 18

```
1 //Chapter 9, Example 9.18
2 clc
3 //Variable Declaration
4 r1 = 5 //resistance in ohms
5 r2 = 4 //resistance in ohms
6 r3 = 2 //resistance in ohms
7 e1 = 10 //voltage
8 e2= 16 //voltage
9 e3 = 8 //voltage
10 r1 = 3 //load resistance in ohms
11
12 //Calculation
13 Eeq = ((e1/r1)-(e2/r2)+(e3/r3))/((1/r1)+(1/r2)+(1/r3
    ))
14 Req = 1/((1/r1)+(1/r2)+(1/r3)) //
    resistance in ohms
15 il = Eeq/(Req+r1) //load current in
    amperes
16 vl = il*r1 //load voltage
17
18 //Results
19 printf("Eeq = %.3 f V \n",Eeq)
20 printf("Req = %.3 f ohm \n",Req)
21 printf("IL = %.3 f V \n",il)
22 printf("VL = %.3 f V ",vl)
```

Scilab code Exa 9.19 Example 19

```
1 //Chapter 9, Example 9.19
```

```

2  clc
3  //Variable Declaration
4  i1 = 5 //source current in
      amperes
5  i2 = 5/3 //source current in
      amperes
6  g1 = 1 //conductance
7  g2 = 1/6 //conductance
8  r3 = 2 //resistance in ohms
9  //Calculation
10 it = i1+i2 //total current in
      amperes
11 gt = g1+g2 //total conductance in
      mho
12 Eeq = it/gt //voltage
13 Req = 1/gt //equivalent resistance
14 i2ohm = Eeq/(Req+r3) //current in amperes of
      2-ohm
15 Eeq1 = Eeq
16 Req1 = Req
17
18 //Results
19 printf("a. Eeq = %f V \n",Eeq) //converted to
      decimals
20 printf(" Req = %f ohm \n",Req) //converted to
      decimals
21 printf(" I2ohm = %d A \n",i2ohm)
22 printf("b. Eeq = %f V \n",Eeq1) //converted to
      decimals
23 printf(" Req = %f ohm ",Req1) //converted to
      decimals

```

Chapter 10

Capacitors

Scilab code Exa 10.1 Example 1

```
1 //Chapter 10, Example 10.1
2 clc
3 //Variable Declaration
4 c11=5*10^-6 //capacitance in
   farad
5 c12=0.1*10^-6 //capacitance in
   farad
6 c13=20*10^-6 //capacitance in
   farad
7 c14=1000*10^-12 //capacitance in
   farad
8
9 //Calculation
10 c1 = 3*(c11) //capacitance in
   farad
11 c2 = 0.5*(c12) //capacitance in
   farad
12 c3 = 2.5*(c13) //capacitance in
   farad
13 c4 = 5*4*(c14)/(1/8) //capacitance in
   farad
```



```

14
15 //Results
16 printf("a. C = %.f uF \n",c1*10^6)
17 printf("b. C = %.2f uF \n",c2*10^6)
18 printf("c. C = %.f uF \n",c3*10^6)
19 printf("d. C = %.2f uF \n",c4*10^6)

```

Scilab code Exa 10.2 Example 2

```

1 //Chapter 10, Example 10.2
2 clc
3 //Variable Declaration
4 a = 0.01 //area of plate in meter
   square
5 d = 1.5*10^-3 //distance in meter
6 eo = 8.85*10^-12 //permittivity
7 v = 450 //voltage
8
9 //Calculation
10 c = eo*a/d //capacitance in farad
11 E = v/d //electric field in V/m
12 q = c*v //charge on each plate in
   Coulomb
13
14 //Results
15 printf("a. C = %.f pF \n",c*10^12)
16 printf("b. E = %d x 10^3 V/m \n",E/1000)
17 printf("c. Q = %.2f nC \n",q*10^9)

```

Scilab code Exa 10.3 Example 3

```

1 //Chapter 10, Example 10.3
2 clc

```

```

3 //Variable Declaration
4 v = 450 //voltage
5 d = 1.5*10^3 //distance in meter
6 a = 0.01 //area of plate in meter
    square
7 er = 5 //permittivity
8 eo = 8.855*10^-12 //permittivity
9 C = 59*10^-12 //capacitance in farad
10
11 //Calculation
12 E = v/d //electric field in V/m
13 q = er*eo*E*a //charge on each plate in
    coulomb
14 c = er*C //capacitance in farad
15
16 //Results
17 printf("E = %.1f V/m \n",E) //wrong answer in
    textbook
18 printf("Q = %f nC \n",q*10^9) //wrong answer in
    textbook
19 printf("C = %d pF \n",c*10^12)

```

Scilab code Exa 10.4 Example 10

```

1 //Chapter 10, Example 10.4
2 clc
3 //Variable Declaration
4 c = 0.2*10^-6 //capacitance in farad
5 a = 0.3 //area of plate in meter
    square
6 er = 6 //permittivity
7 di = 200 //dielectric strength
8
9 //Calculation
10 d = (8.85*er*a)/(10^12*c) //find distance

```

```

    in micrometer
11 d1 = d*(39.371)*1000           //converting
    micrometer to mils
12 d2 = di*d1
13
14 //Results
15 printf("d = %.2f um \n",d*10^6)
16 printf("V = %.2f V \n",d2)

```

Scilab code Exa 10.5 Example 5

```

1 //Chapter 10, Example 10.5
2 clc
3 //Variable Declaration
4 r = 8000           //resistance in ohms
5 e = 40            //voltage
6 c = 4*10^-6       //capacitor in farad
7 m = -1/(32*10^-3)
8
9 //Calculation
10 t = r*c           //time constant
11 vc = e            //in volts
12 ic = e/r         //in ampere
13 vr = e            //in volts
14 b = 5*t           //time constant
15
16 //for plotting graph
17 x1 = 0 : t : 1
18 y1=vc*(1- exp(-x1/t))
19 i1=ic*(1- exp(-x1/t))
20 vr1=vc*(exp(-x1/t))
21
22 subplot(221)
23 plot(x1, y1)
24 xlabel("t");

```

```

25 ylabel("vc (V)");
26 subplot(222)
27 plot(x1, i1)
28 xlabel("t");
29 ylabel("ic (mA)");
30 subplot(223)
31 plot(x1, vr1)
32 xlabel("t");
33 ylabel("vR (V)");
34
35 //Results
36 printf("a. t = %d ms \n",t*10^3)
37 printf("    Vc = %d*(1-e^-t/(32*10^-3)) \n",vc)
38 printf("    ic = (%d x 10^-3)*(e^-t/(32*10^-3)) \n",
        ic*1000)
39 printf("    vr = %d*(e^-t/(32*10^-3)) \n",vr)
40 printf("b = %d ms",b*10^3)

```

Scilab code Exa 10.6 Example 6

```

1 //Chapter 10, Example 10.6
2 clc
3 //Variable Declaration
4 t = 32*10^-3 //time in seconds
5 e = 40 //voltage
6 r = 8000 //resistance in ohms
7
8 //Calculation
9 vc = e //voltage
10 ic = -e/r //in ampere
11 vr = -e //voltage
12
13 //for plotting graph
14 x1 = 0 : t : 1
15 y1=vc*(exp(-x1/t))

```

```

16 i1=ic*(exp(-x1/t))
17 vr1=-vc*(exp(-x1/t))
18
19 subplot(221)
20 plot(x1, y1)
21 xlabel("t");
22 ylabel("vc (V)");
23 subplot(222)
24 plot(x1, i1)
25 xlabel("t");
26 ylabel("ic (mA)");
27 subplot(223)
28 plot(x1, vr1)
29 xlabel("t");
30 ylabel("vR (V)");
31
32 // Results
33 printf("a. t = %d ms \n",t*10^3)
34 printf("   Vc = %d*(e^-t/(32*10^-3)) \n",vc)
35 printf("   ic = (%d x 10^-3)*(e^-t/(32*10^-3)) \n",
        ic*1000)
36 printf("   vr = %d*(e^-t/(32*10^-3)) \n",vr)

```

Scilab code Exa 10.7 Example 7

```

1 //Chapter 10, Example 10.7
2 clc
3 //Variable Declaration
4 r1 = 100*10^3 //resistance in ohms
5 r2 = 200*10^3 //resistance in ohms
6 e = 10 //voltage
7 c = 0.05*10^-6 //capacitance in farad
8 t3=0.045
9
10 //Calculation

```

```

11 vc = e //voltage across the
    capacitor
12 t = r1*c //time constant
13 ic = e/r1 //current across the
    capacitor
14 t1 = r2*c //time constant
15 ic2 = -e/r2 //in ampere
16
17
18 //for plotting graph
19 x1 = 0 : 0.01 : t3
20 x2 = t3 : 0.01 : 1
21 y1=vc*(1-exp(-x1/t))
22 y2=vc*(exp(-x2/t1))
23 i1=ic*(exp(-x1/t))
24 i2=ic2*(exp(-x2/t1))
25
26 subplot(221)
27 plot(x1, y1)
28 xlabel("t")
29 ylabel("vc (V)")
30 xtitle('Vc Charging')
31 subplot(222)
32 plot(x2, y2)
33 xlabel("t")
34 ylabel("vc (V)")
35 xtitle('Vc Discharging')
36 subplot(223)
37 plot(x1, i1)
38 xlabel("t")
39 ylabel("ic (mA)")
40 xtitle('ic Charging')
41 subplot(224)
42 plot(x2, i2)
43 xlabel("t")
44 ylabel("ic (mA)")
45 xtitle('ic Discharging')
46

```

```

47 //Results
48 printf("a. vc = %d(1-e^-t/(5*10^-3)) \n",vc)
49 printf("b. ic = (%.1f x 10^-3)(e^-t/(5*10^-3)) \n",
        ic*10^3)
50 printf("c. vc = %d V \n",vc)
51 printf("    ic = %d A \n",ic)
52 printf("d. vc = %d(e^-t/(10*10^-3)) \n",vc)
53 printf("    ic = (%.2f x 10^-3)(e^-t/(10*10^-3)) \n",
        ic2*10^3)

```

Scilab code Exa 10.8 Example 8

```

1 //Chapter 10, Example 10.8
2 clc
3 //Variable Declaration
4 r1 = 5000 //resistance in ohms
5 e = 20 //voltage
6 r2 = 1000 //resistance in ohms
7 r3 = 3000 //resistance in ohms
8 c = 10*10^-6 //capacitance in farad
9
10
11 //Calculation
12 vc = e //voltage across capacitor
13 t1 = (r1+r3)*c //time constant
14 t2 = (r2+r3)*c //time constant
15 ic = e/(r1+r3) //current across capacitor
16 vc1 = 0.632*e //Vc when switch at
    position2
17 ic1 = -vc1/(r2+r3) //Ic when switch at
    position2
18
19
20 //for plotting graph
21 x1 = 0 : 0.001 : t1

```

```

22 x2 = t1 : 0.001 : 0.4
23 y1=vc*(1- exp(-x1/t1))
24 y2=vc1*(exp(-x2/t2))
25 i1=ic*(exp(-x1/t1))
26 i2=ic1*(exp(-x2/t2))
27
28 subplot(221)
29 plot(x1, y1)
30 plot(x2, y2)
31 xlabel("t")
32 ylabel("vc (V)")
33
34 subplot(222)
35 plot(x1, i1)
36 plot(x2, i2)
37 xlabel("t")
38 ylabel("ic (mA)")
39
40 //Results
41 printf("a. vc = %d(1-e^-t/(80*10^-3)) \n",vc)
42 printf("b. ic = (%.1f x 10^-3)(e^-t/(80*10^-3)) \n",
         ic*10^3)
43 printf("c. vc = %.2f(e^-t/(40*10^-3)) \n",vc1)
44 printf("ic = %.2f x 10^-3(e^-t/(40*10^-3)) \n",ic1
         *10^3)

```

Scilab code Exa 10.9 Example 9

```

1 //Chapter 10, Example 10.9
2 clc
3 //Variable Declaration
4 r1 = 2.2*10^3 //resistance in ohms
5 r2 = 1.2*10^3 //resistance in ohms
6 c = 3.3*10^-6 //capacitance in farad
7 e = 24 //voltage

```



```

8  vf = 24                                     //voltage
9  vi = 4                                     //voltage across
    capacitor
10
11 //Calculation
12 t = (r1+r2)*c                               //time constant
13 vc = (vi-vf)                               //Vc when switch close
14 im = (e-(vf+vc))/(r1+r2)                   //current in amperes
15 ic = im                                     //current during the
    transient peroid
16
17
18 //for plotting graph
19 x1 = 0 : 0.001 : (5*t)
20 y1=vf-(-vc*exp(-x1/t))
21 i1=(im*exp(-x1/t))
22 subplot(221)
23 plot(x1, y1)
24 xlabel("t");
25 ylabel("vc (V)");
26 subplot(222)
27 plot(x1, i1)
28 xlabel("t");
29 ylabel("ic (mA)");
30
31
32 //Results
33 printf("a. vc = %dV %dV(e^-t/(11.22*10^-3ms)) \n",vf
    ,vc)
34 printf("b. ic = %.2fmA(e^-t/(11.22*10^-3ms)) \n",ic
    *10^3)

```

Scilab code Exa 10.10 Example 10

```
1 //Chapter 10, Example 10.10
```

```

2  clc
3  //Variable Declaration
4  r1 = 60*10^3           //resistance in ohms
5  r2 = 30*10^3           //resistance in ohms
6  r3 = 10*10^3           //resistance in ohms
7  vi = 5.44              //voltage
8  vf = 0                  //voltage
9  c=0.2*10**-6           //capacitance in farad
10 e=21                    //emf in voltage
11
12 //Calculation
13 rth = ((r1*r2)/(r1+r2))+r3 //applying tevenin's
    theorem
14 eth = r2*e/(r2+r1)       //in volts
15 t=rth*c                  //time constant
16 vc1 = eth                //vc at switch
    position1
17 ic = eth/r2              //ic at switch
    position1
18 vc = vf +(vi-vf)         //vc at switch position2
19 ii = vc/r3               //current in amperes
20 t1=r3*c                  //time constant
21 ic1 = -ii                //ic at switch position2
22
23 //for plotting graph
24 x1 = 0 : 0.001 : 0.035
25 x2 = 0 : 0.0005 : 0.05
26 y1=eth*(1-exp(-x1/t))
27 y2=vc*(exp(-x2/t1))
28 i1=ic*(exp(-x1/t))
29 i2=ic1*(exp(-x2/t1))
30
31 subplot(221)
32 plot2d(x1, y1)
33 xlabel("t")
34 ylabel("vc (V)")
35 xtitle('Vc Charging')
36 subplot(222)

```

```

37 plot2d(x2, y2)
38 xlabel("t")
39 ylabel("vc (V)")
40 xtitle('Vc Discharging')
41 subplot(223)
42 plot2d(x1, i1)
43 xlabel("t")
44 ylabel("ic (mA)")
45 xtitle('ic Charging')
46 subplot(224)
47 plot2d(x2, i2)
48 xlabel("t")
49 ylabel("ic (mA)")
50 xtitle('ic Discharging')
51
52 //Results
53 printf("a. vc = %d(1-e^-t/6ms) \n",vc1)
54 printf("    ic = (%.3f x 10^-3)(e^-t/6ms) \n",ic
    *10^3)
55 printf("b. vc = %.2f(e^-t/2ms) \n",vc)
56 printf("    ic = (%.2f x 10^-3)(e^-t/2ms) \n",ic1
    *10^3)

```

Scilab code Exa 10.11 Example 11

```

1 //Chapter 10, Example 10.11
2 clc
3 //Variable Declaration
4 r2 = 5*10^3 //resistance in ohms
5 r1 = 7*10^3 //resistance in ohms
6 r3 = 18*10^3 //resistance in ohms
7 r4 = 2*10^3 //resistance in ohms
8 e = 120 //voltage
9 c = 40*10^-6 //capacitance in farad
10 vi = 40 //voltage across

```

```

    capacitor
11
12 // Calculation
13 eth = (r3*e)/(r3+r1+r4) //in
    volts
14 rth = ((r2+r3)*(r2+r3))/((r2+r3)+(r2+r3)) //
    resistance in ohms
15 t = rth*c //time
    constant
16 vc = vi //Vc after
    closing switch
17
18 // Results
19 printf("vc = %dV - %dV(e^-t/0.44s) \n", eth, vc)

```

Scilab code Exa 10.12 Example 12

```

1 //Chapter 10, Example 10.12
2 clc
3 //Variable Declaration
4 r1 = 6 //resistance in ohms
5 r2 = 10 //resistance in ohms
6 c = 500*10^-6 //capacitance in farad
7 i = 20*10^-3 //source current in ampere
8
9 // Calculation
10 rth = r1+r2 //resistance in ohms
11 eth = i*r1 //voltage
12 t = rth*c //time constant
13 vc = eth //voltage
14
15 // Results
16 printf("vc = %.2 f(1-e^-t/8ms) \n", vc)

```

Scilab code Exa 10.13 Example 13

```
1 //Chapter 10, Example 10.13
2 clc
3 //Variable Declaration
4 delv = 4 //change in voltage
5 delt = 2*10^-3 //change in time (sec)
6 c = 2*10^-6 //capacitor in farad
7 delv1 = 0 //change in voltage
8 delt1 = 6*10^-3 //change in time (sec)
9
10 //Calculation
11 ic = c*delv/(delt) //current in amperes
12 ic1 = c*delv1/delt //current in amperes
13 ic2 = -c*delv/delt1 //current in amperes
14
15 x1 = 0: 0.0009 : 0.002
16 x3 = 0.005 : 0.0009: 0.011
17
18 ic11=[ic ic ic]
19 ic12=[ic2 ic2 ic2 ic2 ic2 ic2 ic2]
20 plot(x1, ic11, 'LineWidth', 2)
21 plot(x3, ic12)
22 a=gca(); // Handle on axes entity
23 a.x_location = "origin";
24 xlabel("t")
25 ylabel("ic (mA)")
26 xtitle('Average Current')
27
28
29
30 //Results
31 printf("a. ic = %d mA \n", ic*10^3)
32 printf("b. ic = %d \n", ic1)
```

```
33 printf("c. ic = %.2 f mA \n", ic2*10^3)
```

Scilab code Exa 10.14 Example 14

```
1 //Chapter 10, Example 10.14
2 clc
3 //Variable Declaration
4 c1 = 200*10^-6 //capacitance in farad
5 c2 = 50*10^-6 //capacitance in farad
6 c3 = 10*10^-6 //capacitance in farad
7 e = 60 //voltage
8
9 //Calculation
10 ctt = (1/c1)+(1/c2)+(1/c3)
11 ct = 1/ctt //total capacitance
    in farad
12 qt = ct*e //charge on each
    plate
13 v1 = qt/c1 //voltage across each
    capacitor
14 v2 = qt/c2 //voltage across each
    capacitor
15 v3 = qt/c3 //voltage across each
    capacitor
16 e = v1+v2+v3 //voltage
17
18 //Results
19 printf("a. Ct = %d uF \n", ct*10^6)
20 printf("b. Qt = %d uF \n", qt*10^6)
21 printf("c. V1 = %.1 f V \n", v1)
22 printf(" V2 = %.1 f V \n", v2)
23 printf(" V3 = %.1 f V \n", v3)
24 printf(" E = %d V \n", e)
```

Scilab code Exa 10.15 Example 15

```
1 //Chapter 10, Example 10.15
2 clc
3 //Variable Declaration
4 c1 = 800*10^-6 //capacitance in farad
5 c2 = 60*10^-6 //capacitance in farad
6 c3 = 1200*10^-6 //capacitance in farad
7 e = 48 //voltage
8
9 //Calculation
10 ct = c1+c2+c3 //total capacitance in
    farad
11 q1 = c1*e //charge on each plate in
    coulombs
12 q2 = c2*e //charge on each plate in
    coulombs
13 q3 = c3*e //charge on each plate in
    coulombs
14 qt = q1+q2+q3 //total charge in coulombs
15
16 //Results
17 printf("a. Ct = %d uF \n",ct*10^6)
18 printf("b. Q1 = %.1 f mC \n",q1*10^3)
19 printf(" Q2 = %.2 f mC \n",q2*10^3)
20 printf(" Q3 = %.1 f mC \n",q3*10^3)
21 printf("c. Qt = %.2 f mC \n",qt*10^3)
```

Scilab code Exa 10.16 Example 16

```
1 //Chapter 10, Example 10.16
2 clc
```

```

3 //Variable Declaration
4 c1 = 3*10^-6 //capacitance in farad
5 c2 = 4*10^-6 //capacitance in farad
6 c3 = 2*10^-6 //capacitance in farad
7 e = 120 //voltage
8
9 //Calculation
10 ct1 = c2+c3 //capacitance in farad
11 ct = (c1*ct1)/(c1+ct1) //capacitance in
    farad
12 qt = ct*e //charge in coulombs
13 q1 = qt //charge of C1 in coulombs
14 v1 = q1/c1 //voltage1 of C1 in
    coulombs
15 qt1 = q1 //charge in coulombs
16 vt1 = qt1/ct1 //voltage total
17 q2 = c2*vt1 //charge of C2
18 q3 = c3*vt1 //charge of C3
19
20 //Results
21 printf("Qt = %d uC \n",qt*10^6)
22 printf("Q1 = %d uC \n",q1*10^6)
23 printf("V1 = %d V \n",v1)
24 printf("Vt = %d uC \n",vt1)
25 printf("Q2 = %d uC \n",q2*10^6)
26 printf("Q3 = %d uC \n",q3*10^6)

```

Scilab code Exa 10.17 Example 17

```

1 //Chapter 10, Example 10.17
2 clc
3 //Variable Declaration
4 r1 = 4 //resistance in ohms
5 r2 = 8 //resistance in ohms
6 e = 24 //voltage

```



```

7 c1 = 20*10^-6           //capacitance in farad
8
9
10 //Calculation
11 vc = (r2*e)/(r1+r2)    //voltage across capacitor
12 q1 = c1*vc             //charge in coulombs
13
14 //Results
15 printf("Vc = %d V \n",vc)
16 printf("Q1 = %d uC \n",q1*10^6)

```

Scilab code Exa 10.18 Example 18

```

1 //Chapter 10, Example 10.18
2 clc
3 //Variable Declaration
4 r1= 2           //resistance in ohms
5 r2 = 7         //resistance in ohms
6 r3 =8         //resistance in ohms
7 e = 72        //voltage
8 c1 = 2*10^-6  //capacitance in farad
9 c2 = 3*10^-6  //capacitance in farad
10
11 //Calculation
12 vc2 = (r2*e)/(r1+r2) //voltage across C2
13 vc1 = (r1*e)/(r1+r2) //voltage across C1
14 q1 = c1*vc1       //charge of C1
15 q2 = c2*vc2       //charge of C2
16
17 //Results
18 printf("Vc2 = %d V \n",vc2)
19 printf("Vc1 = %d V \n",vc1)
20 printf("Q1 = %d uC \n",q1*10^6)
21 printf("Q2 = %d uC \n",q2*10^6)

```

Scilab code Exa 10.19 Example 19

```
1 //Chapter 10, Example 10.19
2 clc
3 //Variable Declaration
4 c1 = 2*10^-6 //capacitance in farad
5 v1 = 16 //voltage
6 c2 = 3*10^-6 //capacitance in farad
7 v2 = 56 //voltage
8
9 //Calculation
10 wc = c1*v1*v1/2 //energy of charge in joule
11 wc1 = c2*v2*v2/2 //energy of charge in joule
12
13 //Results
14 printf("Wc = %d uJ \n",wc*10^6)
15 printf("Wc = %d uJ \n",wc1*10^6)
```

Chapter 11

Magnetic Circuits

Scilab code Exa 11.1 Example 1

```
1 //Chapter 11, Example 11.1
2 clc
3 //Variable Declaration
4 o = 6*10^-5 //flux in webers
5 a = 1.2*10^-3 //area in square meters
6
7 //Calculation
8 b = o/a //Tesla in T
9
10
11 //Results
12 printf("B = %d x 10^-2 T \n",b*10^2)
```

Scilab code Exa 11.2 Example 2

```
1 //Chapter 11, Example 11.2
2 clc
3 //Variable Declaration
```

```

4 a = 0.25 //area
5 b = 1.2 //tesla in T
6 //Calculation
7 A = a*(1/39.37)*(1/39.37) //area in square
meter
8 o = b*A //flux in webers
9
10 //Results
11 printf("O = %.3f x 10^-4 Wb ",o*10^4)

```

Scilab code Exa 11.3 Example 3

```

1 //Chapter 11, Example 11.3
2 clc
3 //Variable Declaration
4 o = 4*10^-4 //magnetic flux in
webers
5 a = 2*10^-3 //area in square meter
6 h = 170 //magnetizing force in
At/m
7 l = 0.16 //mean length in meter
8 n = 400 //number of turns
9 uo = 12.56*10^-7 //magnetic permeability
(uo)
10
11 //Calculation
12 b = o/a //the flux density
13 I = h*l/n //applying ampere's
circuital law in mA
14 u = b/h //the permeability of
the material in Wb/A.m
15 ur = u/uo //relative permeability
(ur)
16 //Results
17 printf("a. I = %d mA \n",I*10^3)

```

```

18 printf("b U = %f x 10^-3 Wb/A.m \n",u*10^3)
19 printf(" Ur = %.2f \n",ur)

```

Scilab code Exa 11.4 Example 4

```

1 //Chapter 11, Example 11.4
2 clc
3 //Variable Declaration
4 o = 3.5*10^-4 //magnetic flux in Wb
5 a = 6.452*10^-4 //area in square meter
6 hefab = 70 //magnetizing force
   sheet steel in At/m
7 hbcde = 1600 //magnetizing force
   cast iron in At/m
8 n = 50 //number of turns
9
10 //Calculation
11 lefab = 4+4+4 //length of the
   material in inch
12 lbcde = 0.5+4+0.5 //length of the
   material in inch
13 lbcde1 = lbcde*(1/39.37) //in meters
14 lefab1 = lefab*(1/39.37) //in meters
15 b = o/a //flux density in T
16 q = (hefab*lefab1)+(hbcde*lbcde1) //in At
17 i = q/n //applying ampere's
   circuital law
18
19 //Results
20 printf("I = %.2f A \n",i)

```

Scilab code Exa 11.5 Example 5

```

1 //Chapter 11, Example 11.5
2 clc
3 //Variable Declaration
4 o = 1.5*10^-5 //magnetic flux in Wb
5 a = 0.15*10^-3 //area in square meter
6 habcda = 20 //in At/m
7 i1 = 2 //primary current in
  ampere
8 n1 = 60 //number of turn in 1
  st winding
9 n2 = 30 //number of turn in 2
  nd winding
10 labcda = 0.16 //in meter
11
12 //Calculation
13 b = o/a //flux
  density
14 i2 = ((n1*i1)-(habcda*labcda))/n2 //secondary
  current
15
16 //Results
17 printf(" I2 = %.2 f A \n", i2)

```

Scilab code Exa 11.6 Example 6

```

1 //Chapter 11, Example 11.6
2 clc
3 //Variable Declaration
4 o = 0.75*10^-4 //magnetic flux in
  Wb
5 a = 1.5*10^-4 //area in square
  meter
6 hcore = 280 //H(cast steel) in
  At/m
7 hg = 3.98*10^5 //in At/m

```

```

8  icore = 100*10^-3           //in meter
9  ig = 2*10^-3              //in meter
10 n = 200                   //number of turns
11
12 //Calculation
13 b = o/a                    //flux density
14 i = ((hcore*icore)+(hg*ig))/n //in ampere
15
16 //Results
17 printf("I = %.2f A \n", i)

```

Scilab code Exa 11.7 Example 7

```

1 //Chapter 11, Example 11.7
2 clc
3 //Variable Declaration
4 o2 = 1.5*10^-4           //magnetic flux in Wb
5 a = 6*10^-4             //area in square meter
6 hbcde = 40              //in At/m
7 ibcde = 0.2             //in meter
8 ibe = 0.05              //in meter
9 b1 = 0.97               //tesla in T
10 hefab = 400            //in At/m
11 iefab = 0.2            //in meter
12 n = 50                  //number of turns
13 h1 = 40                 //in At/m
14 uo = 12.57*10^-7       //magnetic permeability
15 h2 = 160                // in At/m
16 b3 = 0.25              //tesla in T
17
18 //Calculation
19 b2 = o2/a               //flux density in T
20 hbe = (hbcde*ibcde)/ibe //in At/m
21 o1 = b1*a              //flux in Wb
22 ot = o1+o2             //flux in Wb

```

```

23 b = ot/a //flux density in T
24 i = ((hefab*iefab)+(hbe*ibe))/n //in ampere
25 u = b3/h1 //permeability
26 ur = u/uo //permeability
27 u1 = b1/h2 //permeability
28 ur1 = u1/uo //permeability
29 u4 = b/hefab //
    permeability
30 ur2 = u4/uo //permeability
31
32 //Results
33 printf("I = %.2f A \n", i) //required
    curent
34 printf("Ur = %.1f \n", ur)
35 printf("For section be, Ur = %d \n", ur1)
36 printf("For section efab, Ur = %.2f \n", ur2)

```

Scilab code Exa 11.8 Example 8

```

1 //Chapter 11, Example 11.8
2 clc
3 //Variable Declaration
4 n = 60 //numbers of turns
5 i = 5 //in ampere
6 labcda = 0.3 //in meter
7 b = 0.39 //in T
8 a = 2*10^-4
9
10 //Calculation
11 habcda = (n*i)/labcda //by Ampere's
    circuital law
12 o = b*a //magnetic flux in
    Wb
13
14 //Results

```



```
15 printf("O = %.2f*10-4 Wb", o*104)
```

Scilab code Exa 11.9 Example 9

```
1 //Chapter 11, Example 11.9
2 clc
3 //Variable Declaration
4 n = 100 //number of turns
5 i = 4 //in ampere
6 ig = 0.001 //in meter
7 uo = 12.56*10-7 //permeability
8 hcore = 850 //in At/m
9 lcore = 0.16 //in meter
10 a = 0.003 //area in meter
    square
11 hcore1 = 1500 //in At/m
12 lcore1 = 0.16 //in meter
13
14 //Calculation
15 hg = (n*i)/ig //in At/m
16 bg = uo*hg //in T
17 o = bg*a //flux in Wb
18 ni1 = (hcore*lcore)+(hg*ig) //applying
    Ampere's circuital law
19 o1 = (1-0.3)*o //magnetic flux
20 b = o1/a //in T
21 hgig = (7.96*105)*b*ig //in At/m
22 ni = (hcore*lcore)+(hgig) //in At
23 hga = n*i //in At
24
25 //Results
26 printf("NI = %.2f At > %d At \n", ni, hga)
27 printf("O = %.3f x 10-3 Wb", o1*103)
```

Chapter 12

Inductors

Scilab code Exa 12.1 Example 1

```
1 //Chapter 12, Example 12.1
2 clc
3 //Variable Declaration
4 d = 4*10^-3 //diameter in meter square
5 n = 100 //number of turns
6 uo = 12.56*10^-7 //air in Wb/A.m
7 l = 0.1 //length of the core in meter
8
9 //Calculation
10 a = (%pi*d*d)/4 //area of the core in square
    meter
11 lo = (n*n*uo*a)/l //inductance of the air-coil
    in H
12
13 //Results
14 printf("Lo = %.2 f uH",lo*10^6)
```

Scilab code Exa 12.2 Example 2

```

1 //Chapter 12, Example 12.2
2 clc
3 //Variable Declaration
4 ur = 2000 //permeability of the core
5 lo = 1.58*10^-6 //inductance of the coil
6
7 //Calculation
8 l = ur*lo //inductance in H
9
10 //Results
11 printf("L = %.2 f mH",l*10^3)

```

Scilab code Exa 12.3 Example 3

```

1 //Chapter 12, Example 12.3
2 clc
3 //Variable Declaration
4 i = 10*10^-3 //current in milli
   ampere
5 l = 4*10^-3 //inductor in henry
6 t = 2*10^-3 //time in millisecond
7 t1 = 5*10^-3 //time in millisecond
8 i1 = 0 //current
9
10 //Calculation
11 v12 = (l*i1)/t //voltage in
   millivolts
12 v1 = (l*i)/t //voltage in
   millivolts
13 v11 = (-l*i)/t1 //voltage
14
15 //Results
16 printf("a. VL = %d \n",v12)
17 printf("b. VL = %d mV \n",v1*10^3)
18 printf("c. VL = %d mV \n",v11*10^3)

```

```
19 printf("d. VL = %d",v12)
```

Scilab code Exa 12.4 Example 4

```
1 //Chapter 12, Example 12.4
2 clc
3 //Variable Declaration
4 l = 4 //inductance in H
5 r1 = 2000 //resistor1 in ohm
6 e = 50 //voltage
7
8 //Calculation
9 to = l/r1 //in millisecond
10 im = e/r1 //in milliAmpere
11 il = im //load current
12 v1 = e //load voltage
13
14 //Results
15 printf("iL = (%d x 10^-3)(1-e^-t/(%dx10^-3)) \n",il
        *10^3,to*10^3)
16 printf("VL = %de^-t/(%dx10^-3) \n",v1,to*10^3)
```

Scilab code Exa 12.5 Example 5

```
1 //Chapter 12, Example 12.5
2 clc
3 //Variable Declaration
4 e = 16 //battery volts
5 r1 = 2.2*10^3 //resistance1 in kohm
6 r2 = 6.8*10^3 //resistance2 in kohm
7 l = 100*10^-3 //inductance in mH
8 ii = 4*10^-3 //load current in mA
9
```

```

10 //Calculation
11 rt = r1+r2 //total resistance
12 if = e/(r1+r2) //in mA
13 to = 1/rt //time constant in
    microsecond
14 il = if //current through the coil-
15 il2 = ii-if //once the switch is closed
16 vr1 = ii*r1 //voltage of resistance1
17 vr2 = ii*r2 //voltage of resistance2
18 vm = e-vr1-vr2 //voltage across the coil
19 vl = vm
20
21 //Results
22 printf("a. iL = %.2 f mA",il*10^3)
23 printf(" + %.2 f mAe^-t/%.2 fus \n",il2*10^3,to*10^6)
24 printf("b. VL = %d*e^-t/%.2 fus \n",vl,to*10^6)
25 printf("c. iL = %.2 f mA ",il*10^3)

```

Scilab code Exa 12.6 Example 6

```

1 //Chapter 12, Example 12.6
2 clc
3 //Variable Declaration
4 r1 = 2000 //resistor1 in kohm
5 r2 = 3000 //resistor2 in kohm
6 l = 4 //inductor in henry
7 e = 50 //battery voltage
8
9
10 //Calculation
11 to = 1/r1 //time constant in ms
12 vl = e //load voltage
13 im = e/r1 //current in milliampere
14 il = im //load current
15 vr1 = e //voltage across r1

```

```

16 vr2 = vr1 //voltage across r2
17 to1 = 1/(r1+r2)
18 vi = (1+(r2/r1))*e //voltage in volts
19 vl1 = -vi //voltage at switch open
20 im1 = e/r1 //current in milliampere
21 il1 = im1 //current at switch open
22 vr11 = e //voltage across at r1 switch
    open
23 vr22 = (-r2/r1)*e //voltage across at r2 switch
    open
24
25 //Results
26 printf("a. VL = %d*e^-t/(%d*10^-3) \n",vl,to*10^3)
27 printf("    iL = %d*10^-3(1-e^-t/%d*10^-3) \n",il
    *10^3,to*10^3)
28 printf("    VR1 = %d(1-e^-t/%d*10^-3) \n",vr1,to
    *10^3)
29 printf("    VR2 = %d V\n",vr2)
30 printf("b. iL = %.3 fe^-t/(0.8*10^-3) \n",il)
31 printf("    VL = -%de^-t/(%.1 f*10^-3) \n",vi,to1
    *10^3)
32 printf("    VR1 = %de^-t/(%.1 f*10^-3) \n",vr11,to1
    *10^3)
33 printf("    VR2 = %de^-t/(%.1 f*10^-3)",vr22,to1*10^3)

```

Scilab code Exa 12.7 Example 7

```

1 //Chapter 12, Example 12.7
2 clc
3 //Variable Declaration
4 r1 = 20000 //resistor1 in kohm
5 r2 = 4000 //resistor2 in kohm
6 r3 = 16000 //resistor3 in kohm
7 l = 80 //inductor in mH
8 e = 12 //supply voltage

```

```

9  n = 2                                //number of turns
10
11 //Calculation
12 rth = r1/n                            //by thevenin
    theorem
13 to = 1/rth                            //time in
    microsecond
14 eth = (r2+r3)*e/(r1+r2+r3)           //applying voltage
    divider
15 im = eth/rth                          //current in
    milliampere
16 il = im                               //load current
17 vl = eth                               //load voltage
18
19 //Results
20 printf("a. iL = (%.1f*10^-3)(1-e^-t/(%d*10^-6)) \n",
    il*10^3,to*10^3)
21 printf("    VL = %de^-t/(%d*10^-6) \n",vl,to*10^3)
22
23
24
25 //Graph
26 x1=0:(1*10**-3):(50*10**-3)
27 vc=v1*(exp(-x1/to))                  //for VL
28 ic=il*(1-exp(-x1/to))                //for iL
29
30 subplot(221)
31 plot(x1,vc)
32 xlabel("t(s)")
33 ylabel("VL (V)")
34 xtitle("VL for the R-L network")
35 subplot(222)
36 plot(x1,ic)
37 xlabel("t(s)")
38 ylabel("iL (mA)")
39 xtitle("iL for the R-L network\ ")

```

Scilab code Exa 12.8 Example 8

```
1 //Chapter 12, Example 12.8
2 clc
3 //Variable Declaration
4 r1 = 2.2*10^3 //resistor1 in kohm
5 r2 = 8.2*10^3 //resistor2 in kohm
6 r3 = 1000 //resistor3 in kohm
7 i = 12*10^-3 //sources current
8 e = 6 //voltage
9 l = 680*10^-3 //inductor in mH
10 i = 12*10^-3 //current source in mA
11 E = 26.4 //voltage
12
13 //Calculation
14 ii = -e/r3 //using ohm's law
15 rth = r1+r2 //applying thevenin's
    theorem
16 eth = i*r1 //in volts
17 if = E/rth //in milliampere
18 to = l/rth //time constant
19 il = if //in milliampere
20 il1 = ii-if //in milliampere
21
22 //Results
23 printf("a. ii = %d mA \n",ii*10^3)
24 printf("b. iL = %.2f mA %.2fmAe^-t/(%.2fus) ",il
    *10^3,il1*10^3,to*10^6)
25
26 //Graph
27 x1=0:(0.1*10**-3):(0.5*10**-3)
28 ic=il+(il1*exp(-x1/to)) //for iL
29 plot(x1,ic)
30 xlabel("t(s)")
```



```
31 ylabel("iL (mA)")
32 xtitle("The current iL for the network")
```

Scilab code Exa 12.9 Example 9

```
1 //Chapter 12, Example 12.9
2 clc
3 //Variable Declaration
4 l = 1.2 //L2 and L3 are equal
5 n = 2 //number of turns
6 l4 = 1.8 //inductance L4 in henry
7 l1 = 0.56 //inductance L1 in henry
8
9 //Calculation
10 lt = l/n //total inductance in
    henry
11 lt11 = (lt*l4)/(lt+l4) //total inductance in
    henry
12 lt2 = l1+lt11 //total inductance in
    henry
13 //Results
14 printf("LT = %.2 f H",lt2)
```

Scilab code Exa 12.10 Example 9

```
1 //Chapter 12, Example 12.10
2 clc
3 //Variable Declaration
4 r1 = 2 //resistance1 in ohms
5 r2 = 3 //resistance2 in ohms
6 e = 10 //voltage in volts
7
8 //Calculation
```

```

9  i1 = e/(r1+r2)           //load current in ampere
10 vc = (r2*e)/(r1+r2)     //load voltage in volts
11
12 //Results
13 printf("IL = %d A \n",i1)
14 printf("Vc = %d V ",vc)

```

Scilab code Exa 12.11 Example 11

```

1 //Chapter 12, Example 12.11
2 clc
3 //Variable Declaration
4 r1 = 2           //resistance1 in ohms
5 r2 = 5           //resistance2 in ohms
6 r3 = 1           //resistance3 in ohms
7 r4 = 4           //resistance4 in ohms
8 r5 = 7           //resistance5 in ohms
9 e = 50          //voltage in volts
10
11 //Calculation
12
13 i1 = e/(r1+r3+r5) //current1 in ampere
14 i2= i1           //equal
15 v2 = i2*r5       //voltage2 by ohm's
    law
16 v1 = (r3+r5)*e/(r1+r3+r5) //voltage1 by voltage
    divider
17
18 //Results
19 printf("I1 = %d A \n",i1)
20 printf("V2 = %d V \n",v2)
21 printf("V1 = %d V \n",v1)

```

Scilab code Exa 12.12 Example 12

```
1 //Chapter 12, Example 12.12
2 clc
3 //Variable Declaration
4 r1 = 3 //resistance1
5 l = 6*10^-3 //inductance in mH
6 e = 15 //voltage in volts
7 r2 = 2 //resistance2
8 //Calculation
9 im = e/(r1+r2) //inductance current
10 ws = (l*im*im)/2 //energy stored
11
12 //Results
13 printf("Ws = %d mJ",ws*10^3)
```

Chapter 13

Sinusoidal Alternating Waveforms

Scilab code Exa 13.1 Example 1

```
1
2 //Chapter 13, Example 13.1
3 clc
4 //Variable Declaration
5 f1 = 60 //frequency in hertz
6 f2 = 1000 //frequency in hertz
7
8 //Calculation
9 t1 = 1/f1 //time in milli second
10 t2 = 1/f2 //time in milli second
11
12 //Results
13 printf("T = %.2 f ms \n",t1*10^3)
14 printf("T = %d ms ",t2*10^3)
```

Scilab code Exa 13.2 Example 2

```

1
2 //Chapter 13, Example 13.2
3 clc
4 //Variable Declaration
5 t = 20*10-3 //time in milli second
6
7
8 //Calculation
9 f = 1/t //frequency in hertz
10
11 //Results
12 printf("f = %d Hz",f)

```

Scilab code Exa 13.3 Example 3

```

1
2 //Chapter 13, Example 13.3
3 clc
4 //Variable Declaration
5 d = 4 //division
6 t= 50*10-6 //time in microsecond
7 d1 = 2 //division
8 v = 0.1 //in volts
9
10 //Calculation
11 t1 = d*t //time in microsecond
12 f = (1/t1) //frequency in hertz
13 vm = d1*v //in volts
14
15 //Results
16 printf("T = %d us \n",t1*106)
17 printf("f = %d kHz \n",f/103)
18 printf("Vm = %.1 f V \n",vm)

```

Scilab code Exa 13.4 Example 4

```
1
2 //Chapter 13, Example 13.4
3 clc
4 //Variable Declaration
5 f = 60 //frequency in hertz
6
7
8 //Calculation
9 w = 2*3.14*f //in rad/s
10
11 //Results
12 printf("w = %d rad/s",w)
```

Scilab code Exa 13.5 Example 5

```
1 //Chapter 13, Example 13.5
2 clc
3 //Variable Declaration
4 w = 500 //in rad/s
5
6
7 //Calculation
8 t = (2*3.14)/w //time in milliseconds
9 f = 1/t //frequency in hertz
10
11 //Results
12 printf("T = %.2 f ms \n",t*103)
13 printf("f = %.2 f Hz ",f)
```

Scilab code Exa 13.6 Example 6

```
1 //Chapter 13, Example 13.6
2 clc
3 //Variable Declaration
4 l = (3.14/2)           //in rad/s
5 w = 200               //in rad/s
6
7 //Calculation
8 t = 1/w               //time in millisecond
9
10 //Results
11 printf("t = %.2f ms \n",t*10^3)
```

Scilab code Exa 13.7 Example 7

```
1 //Chapter 13, Example 13.7
2 clc
3 //Variable Declaration
4 f = 60                //frequency in hertz
5 t= 5*10^-3           //time in milli second
6
7 //Calculation
8 l = 2*3.14*f*t        //in rad/s
9 lo = (180*1)/3.14     //in degree
10
11
12 //Results
13 printf("alpha = %.3f rad \n",l)
14 printf("alpha(degree) = %d degree",lo)
```

Scilab code Exa 13.8 Example 8

```
1 //Chapter 13, Example 13.8
2 clc
3 //Variable Declaration
4 e1 = 5 //in volts
5 l = 0.8*3.14 //in degree
6
7 //Calculation
8 e = e1*sind(40) //in volts
9 lo = 180*l/3.14 //in degree
10 e2 = 5*sind(lo) //in volts
11
12 //Results
13 printf("e = %.3f V \n", e)
14 printf("e = %.3f V \n", e2)
```

Scilab code Exa 13.9 Example 9

```
1 //Chapter 13, Example 13.9
2 clc
3 //Variable Declaration
4 v = 4 //in volts
5 em = 10 //in volts
6 w = 377 //in rad/s
7 //Calculation
8 l1 = asind(v/em) //alpha1 in degree
9 l2 = 180-(l1) //alpha2 in degree
10 l3 = (%pi/180)*(l1) //in rad
11 t1 = l3/w //in millisecond
12 l4 = (%pi/180)*l2 //in rad
13 t2 = l4/w //in millisecond
```



```

14 //Results
15 printf("a.  alpha1 = %.3 f \n",l1)
16 printf("    alpha2 = %.3 f \n",l2)
17 printf("b.  t1 = %.2 f ms \n",t1*10^3)
18 printf("    t2 = %.2 f ms \n",t2*10^3)

```

Scilab code Exa 13.10 Example 10

```

1 //Chapter 13, Example 13.10
2 clc
3 //Variable Declaration
4 w = 314 //in rad/s
5
6 //Calculation
7 t = (2*3.14)/w //in millisecond
8 t1 = t/2 //in millisecond
9 t2 = t/4 //in millisecond
10 t3 = t/12 //in millisecond
11
12 //Results
13 printf("360: T = %d ms \n",t*10^3)
14 printf("180: T/2 = %d ms \n",t1*10^3)
15 printf("90:  T/4 = %d ms \n",t2*10^3)
16 printf("30:  T/12 = %.2 f ms \n",t3*10^3)

```

Scilab code Exa 13.11 Example 11

```

1 //Chapter 13, Example 13.11
2 clc
3 //Variable Declaration
4 w = 1000 //in rad/s
5 t = 2*10^-3 //in millisecond
6 i1 = 6*10^-3 //in milliampere

```

```

7
8 //calculation
9
10 ap1 = (180*2)/3.14           //alphain degree
11 sin114 = 0.9093             //in degree
12 i = i1*sin114               //in milliampere
13
14 //Results
15 printf("i = %.2f mA ",i*10^3)

```

Scilab code Exa 13.13 Example 13

```

1 //Chapter 13, Example 13.13
2 clc
3 //Variable Declaration
4 v1 = 10                       //in volts
5 t1 = 1*10^-3                 //in millisecond
6 v2 = 14                       //in volts
7 v3 = 6                         //in volts
8
9 //Calculation
10 g = ((v1*t1)-(v1*t1))/(2*10^-3)
11 g1 = ((v2*t1)-(v3*t1))/(2*10^-3)
12
13
14 //Results
15 printf("a. G = %d V \n",g)
16 printf("b. G = %d V \n",g1)

```

Scilab code Exa 13.14 Example 14

```

1 //Chapter 13, Example 13.14
2 clc

```

```

3 //Variable Declaration
4 v1 = 3 //in volts
5 v2 = 1 //in volts
6 t1 = 4*10^-3 //in millisecond
7 v3 = 10 //in volts
8 t2 = 2*10^-3 //in millisecond
9 v4 = 4 //in volts
10 v5 = 2 //in volts
11
12 //Calculation
13 g = ((v1*t1)-(v2*t1))/(8*10^-3)
14 g1 = (-(v3*t2)+(v4*t2)-(v5*t2))/(10*10^-3)
15
16
17 //Results
18 printf("a. G = %d V \n",g)
19 printf("b. G = %.1 f V \n",g1)

```

Scilab code Exa 13.15 Example 15

```

1
2 //Chapter 13, Example 13.15
3 clc
4 //Variable Declaration
5 a = 2 //in volts
6 b = 2 //in volts
7 am = 1 //in volts
8 //Calculation
9 g = (a*am-b*am)/(2*3.14) //in volts
10
11 //Results
12 printf("G = %d V ",g)

```

Scilab code Exa 13.16 Example 16

```
1 //Chapter 13, Example 13.16
2 clc
3 //Variable Declaration
4 v1 = 16 //in volts
5 v2 = 2 //in volts
6
7
8 //Calculation
9 g1 = (v1+v2)/2 //in volts
10 g2 = g1-(16) //in volts
11
12
13 //Results
14 printf("G = %d mv ",g2)
```

Scilab code Exa 13.17 Example 17

```
1
2 //Chapter 13, Example 13.17
3 clc
4 //Variable Declaration
5 a = 2 //in volts
6 b = 0 //in volts
7 am = 10 //in volts
8 //Calculation
9 g = (a*am+b)/(2*3.14) //in volts
10
11 //Results
12 printf("G = %.2 f V ",g)
```

Scilab code Exa 13.18 Example 18

```

1 //Chapter 13, Example 13.18
2 clc
3 //Variable Declaration
4 g = 2 //in volts
5
6 //Results
7 printf("G = %d mV ",g)

```

Scilab code Exa 13.19 Example 19

```

1 //Chapter 13, Example 13.19
2 clc
3 //Variable Declaration
4 a = 0.707 //in volts
5 b = 12*10^-3 //in milliampere
6 c = 169.73 //in volts
7
8 //Calculation
9 i = a*b //in milliampere
10 v = a*c //in volts
11 //Results
12 printf("a. Irms = %.3 f mA \n",i*10^3)
13 printf("b. Irms = %.3 f mA \n",i*10^3)
14 printf("c. Vrms = %d mA ",v)

```

Scilab code Exa 13.20 Example 20

```

1 //Chapter 13, Example 13.20
2 clc
3 //Variable Declaration
4 pdc = 3.6 //power in watt
5 vdc= 120 //in volts
6

```

```

7
8 // Calculation
9 idc = pdc/vdc //in milliampere
10 im = sqrt(2)*idc //in milliampere
11 em = sqrt(2)*vdc //in millivolts
12
13 // Results
14 printf("Idc = %d mA \n",idc*10^3)
15 printf("Im = %.2 f mA \n",im*10^3)
16 printf("Em = %.2 f mA \n",em)

```

Scilab code Exa 13.21 Example 21

```

1 //Chapter 13, Example 13.21
2 clc
3 //Variable Declaration
4 x1 = 4 //in second
5 y1 = 9 //in volts
6 x2 = 4 //in second
7 y2 = 1 //in volts
8 x3 = 8 //in volts
9
10 // Calculation
11 v = sqrt(((y1*x1)+(y2*x2))/x3) //in volts
12
13 // Results
14 printf("Vrms = %.3 f V ",v)

```

Scilab code Exa 13.22 Example 22

```

1 //Chapter 13, Example 13.22
2 clc
3 //Variable Declaration

```

```

4 y1 = 100 //in volts
5 x1 = 2 //in second
6 y2 = 16 //in volts
7 y3 = 4 //in volts
8
9 //Calculation
10 v = sqrt(((y1*x1)+(y2*x1)+(y3*x1))/10) //in
    volts
11
12 //Results
13 printf("Vrms = %.3f V ",v)

```

Scilab code Exa 13.23 Example 23

```

1 //Chapter 13, Example 13.23
2 clc
3 //Variable Declaration
4 y1 = 1600 //in volts
5 x = 10*10^-3 //time in millisecond
6
7 //Calculation
8 v = sqrt(((y1*x)+(y1*x))/(20*10^-3)) //in volts
9
10 //Results
11 printf("Vrms = %d V ",v)

```

Scilab code Exa 13.24 Example 24

```

1 //Chapter 13, Example 13.24
2 clc
3 //Variable Declaration
4 v1 = 20 //in volts
5 v2 = 25 //in volts

```

```
6 vm1 = v1 //in volts
7 vm2 = 15 //in volts
8
9 //Calculation
10 m = 1.11*v1 //in volts
11 vr1 = 0.707*vm1 //in volts
12 vdc = vr1 //in volts
13 vr2 = 0.707*vm2 //in volts
14 //Results
15 printf("Meter indication = %.1f V \n",m)
16 printf("Vrms = %.2f V \n",vr1)
17 printf("Vrms = Vdc = %d V \n",vdc)
18 printf("Vrms = %.1f V \n",vr2)
```

Chapter 14

The Basic Elements and Phasors

Scilab code Exa 14.1 Example 1

```
1 //Chapter 14, Example 14.1
2 clc
3 //Variable Declaration
4 vm = 100 //voltage magnitude of (a)
5 r = 10 //resistance in ohms
6 vm1 = 25 //voltage magnitude of (b)
7
8
9 //Calculation
10 i = vm/r //current of (a)
11 i1 = vm1/r //current of (b)
12
13 //Results
14 printf("a. i = %dsin377t \n",i)
15 printf("b. i = %.1 fsin(377t+60dgree) \n",i1)
```

Scilab code Exa 14.2 Example 2

```
1 //Chapter 14, Example 14.2
2 clc
3 //Variable Declaration
4 im = 40 //current magnitude in
    ampere
5 r = 5 //resistance in ohms
6
7 //Calculation
8 v = im*r //voltage in volt
9
10 //Results
11 printf("v = %dsin(377t+30degree) ",v)
```

Scilab code Exa 14.3 Example 3

```
1 //Chapter 14, Example 14.3
2 clc
3 //Variable Declaration
4 w = 377 //in rad/s
5 l = 0.1 //inductance in ohms
6 im = 10 //current of (a)
7 im1= 7 //current of (b)
8
9 //Calculation
10 x1 = w*l //inductance reactance
11 vm = im*x1 //voltage of (a)
12 vm1 = im1*x1 //voltage of (b)
13
14 //Results
15 printf("a. v = %dsin(377t+90degree) \n",vm)
16 printf("b. v = %.1 fsin(377t+20degree)",vm1)
```

Scilab code Exa 14.4 Example 4

```
1 //Chapter 14, Example 14.4
2 clc
3 //Variable Declaration
4 w = 20 //in rad/s
5 l = 0.5 //inductance in henry
6 vm = 100 //voltage in volts
7
8
9 //Calculation
10 xl = w*l //inductance reactance
11 im = vm/xl //current in ampere
12
13 //Results
14 printf("i = %d sin(20t - 90degree)",im)
```

Scilab code Exa 14.5 Example 5

```
1 //Chapter 14, Example 14.5
2 clc
3 //Variable Declaration
4 w = 400 //in rad/s
5 c = 1*10^-6 //capacitance in
   microfarad
6 vm = 30 //voltage in volts
7
8
9 //Calculation
10 xc = 1/(w*c) //capacitance
   reactance
11 im = vm/xc //current in ampere
```

```

12
13 //Results
14 printf("i = %d x 10^-3 sin(400t + 90degree)",im
        *10^3)

```

Scilab code Exa 14.6 Example 6

```

1 //Chapter 14, Example 14.6
2 clc
3 //Variable Declaration
4 w = 500 //in rad/s
5 c = 100*10^-6 //capacitance in
    micro farad
6 im = 40 //current in ampere
7
8
9 //Calculation
10 xc = 1/(w*c) //capacitance
    reactance
11 vm = im*xc //voltage in volts
12
13 //Results
14 printf("v = %d sin(500t - 30degree)",vm)

```

Scilab code Exa 14.7 Example 7

```

1 //Chapter 14, Example 14.7
2 clc
3 //Variable Declaration
4 vm1 = 100 //Vm of (a)
5 im1 = 20 //Im of (a)
6 vm2 = 1000 //Vm of (b)
7 im2 = 5 //Im of (b)

```

```

8  vm3 = 500 //Vm of (c)
9  im3 = 1 //Im of (c)
10 vm4 = 50 //Vm of (d)
11 im4 = 5 //Im of (d)
12 w = 377 //in rad/s
13 w1 = 157 //in rad/s
14 //Calculation
15 r = vm1/im1 //resistance of (a)
16 xl = vm2/im2 //inductance
    reactance
17 l = xl/w //inductance in henry
18 xc = vm3/im3 //capacitance
    reactance
19 c = 1/(w1*xc) //capacitance in
    farad
20 r1 = vm4/im4 //resistance of (d)
21
22 //Results
23 printf("a. R = %d ohm \n",r)
24 printf("b. L = %.3 f H \n",l)
25 printf("c. C = %.2 f uF \n",c*10^6)
26 printf("d. R = %d ohm \n",r1)

```

Scilab code Exa 14.8 Example 8

```

1 //Chapter 14, Example 14.8
2 clc
3 //Variable Declaration
4 l = 200*10^-3 //inductance in henry
5 r = 5*10^3 //resistance in ohms
6 xl = 2*3.14*1 //reactance inductor
7
8 //Calculation
9 f= r/xl //frequency in hertz
10

```

```
11 //Results
12 printf("f = %.2f kHz ",f*10^-3)
```

Scilab code Exa 14.9 Example 9

```
1 //Chapter 14, Example 14.9
2 clc
3 //Variable Declaration
4 l = 5*10^-3 //inductance in henry
5 c = 0.1*10^-6 //capacitance in
   farad
6 //calculation
7 xl = 2*3.14*l //inductance
   reactance
8 xc = 1/(2*3.14*c) //capacitance
   reactance
9 f= 1/(2*3.14*sqrt(l*c)) //frequency in hertz
10
11 //Results
12 printf("f = %.2f kHz",f*10^-3)
```

Scilab code Exa 14.10 Example 10

```
1 //Chapter 14, Example 14.10
2 clc
3 //Variable Declaration
4 vm = 10 //voltage in volts
5 im = 5 //current in ampere
6
7
8 //Calculation
9 p = (vm*im)/2 //power
   dissipation in watt
```

```

10 r= (vm/im) //resistance in
    ohm
11 p1 = ((0.707*vm)*(0.707*vm))/r //power in watt
12 p2 = ((0.707*im)*(0.707*im))*r //power in watt
13
14 //Results
15 printf("P = %d W \n",p)
16 printf("P = %.2 f W \n",p1)
17 printf("P = %.2 f W \n",p2)

```

Scilab code Exa 14.11 Example 11

```

1 //Chapter 14, Example 14.11
2 clc
3 //Variable Declaration
4 vm = 100 //peak voltage (a)
5 im = 20 //peak current
6 ov = 40 //theta voltage in
    degree (a)
7 oi = 70 //theta current in
    degree (a)
8 vm1 = 150 //peak voltage (b)
9 im1 = 3 //peak current (b)
10 ov1 = -70 //theta voltage in
    degree (b)
11 oi1 = -50 //theta current in
    degree (b)
12
13 //Calculation
14 p = ((vm*im)/2)*0.866 //power (a) in watt
15 p1 = ((vm1*im1)/2)*0.9397 //power (b) in watt
16
17 //Results
18 printf("P = %d W \n",p)
19 printf("P = %.2 f W \n",p1)

```

Scilab code Exa 14.12 Example 12

```
1 //Chapter 14, Example 14.12
2 clc
3 //Variable Declaration
4 a = cosd(40-(-20)) //leading
5 b = cosd(80-30) //lagging
6 p = 100 //in watt
7 v = 20 //in voltage
8 i = 5 //in ampere
9
10 //Calculation
11 fp = p/(v*i) //power factor
12
13 //Results
14 printf("a. Fp = %.1f leading \n",a)
15 printf("b. Fp = %.4f leading \n",b)
16 printf("c. Fp = %d \n",fp)
```

Scilab code Exa 14.13 Example 13

```
1 //Chapter 14, Example 14.13
2 clc
3 //Variable Declaration
4 c1=complex(3,4)
5 c2=complex(0,-6)
6 c3=complex(-10,-20)
7
8 //Calculation
9 //Function to create plot on complex plane
10 function complexPlot(x,y,z)
```



```

11     xpts = [0 x];
12     ypts = [0 y];
13     plot(xpts, ypts);
14     xtitle(z);
15     xlabel("Real");
16     ylabel("Imaginary");
17 endfunction
18
19 //Plot
20 scf(1)
21 complexPlot(real(c1), imag(c1), "3 + j4")
22 scf(2)
23 complexPlot(real(c2), imag(c2), "0 - j6")
24 scf(3)
25 complexPlot(real(c3), imag(c3), "-10 - j20")

```

Scilab code Exa 14.14 Example 14

```

1 //Chapter 14, Example 14.14
2 clc
3
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9
10
11 [cr1,ci1]=polar2rect(5,30)
12 [cr2,ci2]=polar2rect(7,-120)
13 [cr3,ci3]=polar2rect(-4.2,60)
14
15 //Calculation
16 //Function to create plot on complex plane
17 function complexPlot(x,y,z)

```

```

18     xpts = [0 x];
19     ypts = [0 y];
20     plot(xpts, ypts);
21     xtitle(z);
22     xlabel("Real");
23     ylabel("Imaginary");
24 endfunction
25
26 //Plot
27 scf(1)
28 complexPlot(cr1,ci1,"5 < 30 degree")
29 scf(2)
30 complexPlot(cr2,ci2,"7 < 120 degree")
31 scf(3)
32 complexPlot(cr3,ci3,"4.2 < 60 degree")

```

Scilab code Exa 14.15 Example 15

```

1 //Chapter 14, Example 14.15
2 clc
3 funcprot(0)
4
5 //Rectangle to Polar Conversion
6 function [r,th] = rect2polar(x,y)
7     r=sqrt((x**2)+(y**2))
8     th=atand(y/x)
9 endfunction
10
11
12 //Variable Declaration
13 x = 3 //real part
14 y = 4 //imaginary part
15
16 //Calculation
17 [a,b]=rect2polar(x,y) //function of conversion

```

```
18
19 //Results
20 printf("C = %d < %.2f degree", a,b)
```

Scilab code Exa 14.16 Example 16

```
1 //Chapter 14, Example 14.16
2 clc
3
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9
10
11 //Variable Declaration
12 a=10 //radius
13 b=45 //angular in degree
14
15 //Calculation
16 [r,i]=polar2rect(a,b) //function of conversion
17
18 //Results
19 printf("C = %.2f + j %.2f",r,i)
```

Scilab code Exa 14.17 Example 17

```
1 //Chapter 14, Example 14.17
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
```

```

6      th= 180-(atand(y/x))
7  endfunction
8
9  //Variable Decleration
10 a = 6           //real part
11 b = 3          //imaginary
12
13 //Calculation
14 [r,th] = rect2polar(a,b) //function of
      conversion
15
16 //Results
17 printf("C = %.2f < %.2f degree",r,th)

```

Scilab code Exa 14.18 Example 18

```

1 //Chapter 14, Example 14.18
2 clc
3
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9
10 //Variable Declaration
11 a = 10           //real value
12 b = 230         //angle in degree
13 //Calculation
14 [r,i]=polar2rect(a,b) //function of conversion
15
16 //Results
17 printf("C = %.3f  j%.3f",r,i)

```

Scilab code Exa 14.19 Example 19

```
1 //Chapter 14, Example 14.19
2 clc
3 //addition
4 function [c1,c2]=addition(x1,x2,y1,y2)
5     c1 = x1+x2
6     c2 = y1+y2
7 endfunction
8
9 //Variable Declaration
10 x1=complex(2,4)           //complex form c1
11 y1 = complex(3,1)       //complex form c2
12 x2 = complex(3,6)       //complex form c1
13 y2 = complex(-6,3)      //complex form c2
14
15 //Calculation
16 [a,b]=addition(real(x1),real(y1),imag(x1),imag(y1))
17           //function of conversion
18 [c,d]=addition(real(x2),real(y2),imag(x2),imag(y2))
19           //function of conversion
20
21 //Results
22 printf("a. C = %d + j%d \n", a,b)
23 printf("b. C = %d + j%d", c,d)
```

Scilab code Exa 14.20 Example 20

```
1 //Chapter 14, Example 14.20
2 clc
3 //subtraction
4 function [c1,c2]=subtraction(x1,x2,y1,y2)
```

```

5     c1 = x1-x2
6     c2 = y1-y2
7     endfunction
8 //Variable Declaration
9 x1 = complex(1,4)           //complex form c2
10 y1 = complex(4,6)         //complex form c1
11 x2 = complex(-2,5)        //complex form c2
12 y2 = complex(3,3)         //complex form c1
13 //Calculation
14 [a,b]=subtraction(real(y1),real(x1),imag(y1),imag(x1
    ))           //function of conversion
15 [c,d]=subtraction(real(y2),real(x2),imag(y2),imag(x2
    ))           //function of conversion
16
17
18 //Results
19 printf("a. C = %d + j%d \n", a,b)
20 printf("b. C = %d %dj", c,d)

```

Scilab code Exa 14.21 Example 21

```

1 //Chapter 14, Example 14.12
2 clc
3 //Variable Declaration
4 a1 = 2           //real value
5 a2 = 3           //real value
6 a3= 2           //real value
7 a4 = 4           //real value
8
9 //Calculation
10 x1 = a1+a2       //real value
11 x2 = a3+a4       //real value
12
13 //Results
14 printf("a. %d<45 \n",x1)

```

```
15 printf("b. %d<0", x2)
```

Scilab code Exa 14.22 Example 22

```
1 //Chapter 14, Example 14.22
2 clc
3 function [c1, c2]=equ(x1, x2, y1, y2)
4     c1 = (x1*x2)-(y1*y2)
5     c2 = (y1*x2)+(x1*y2)
6     endfunction
7 //Variable Declaration
8 x1 = complex(2,3)           //complex form of c1
9 y1 = complex(5,10)        //complex form of c2
10 x2 = complex(-2,-3)       //complex form of c1
11 y2 = complex(4,-6)        //complex form of c2
12 //Calculation
13 [a,b]=equ(real(x1), real(y1), imag(x1), imag(y1))
14 [c,d]=equ(real(x2), real(y2), imag(x2), imag(y2))
15
16 //Results
17 printf("a. C = %d + j%d \n", a,b)
18 printf("b. C = %d + j%d", c,d)
```

Scilab code Exa 14.23 Example 23

```
1 //Chapter 14, Example 14.23
2 clc
3 //multiplication
4 function [c1, c2]=equ(x1, x2, y1, y2)
5     c1 = (x1*x2)
6     c2 = (y1+y2)
7     endfunction
8 //Variable Declaration
```

```

9 x1 = 5 //real value
10 x2 = 10 //real value
11 y1 = 20 //angle in degree
12 y2 = 30 //angle in degree
13 x3 = 2 //real value
14 x4 = 7 //real value
15 y3 = -40 //angle in degree
16 y4 = 120 //angle in degree
17
18 //Calculation
19 [a,b]=equ(x1,x2,y1,y2) //function of
    multiplication
20 [c,d]=equ(x3,x4,y3,y4) //function of
    multiplication
21 //Results
22 printf("a. C = %d < %d degree \n", a,b)
23 printf("b. C = %d < %d degree", c,d)

```

Scilab code Exa 14.24 Example 24

```

1 //Chapter 14, Example 14.24
2 clc
3 //multiplication
4 function [C1,C2]=equ2(x1,x2,y1,y2)
5     C1 = ((x1*x2)+(y1*y2))/((y1*y1)+(y2*y2))
6     C2 = ((x2*y1)-(x1*y2))/((y1*y1)+(y2*y2))
7     endfunction
8
9 //Variable Declaration
10 x1 = complex(1,4) //complex form of c1
11 y1 = complex(4,5) //complex form of c2
12 x2 = complex(-4,-8) //complex form of c1
13 y2 = complex(6,-1) //complex form of c2
14 //Calculation
15 [a,b]=equ2(real(x1),real(y1),imag(x1),imag(y1)) //

```



```

    function
16 [c,d]=equ2(real(x2),real(y2),imag(x2),imag(y2)) //
    function
17 //Results
18 printf("C1/C2 = %.3 f + j%.3 f \n",a,b)
19 printf("C1/C2 = %.3 f + j%.3 f \n",c,d)

```

Scilab code Exa 14.25 Example 25

```

1 //Chapter 14, Example 14.25
2 clc
3 //division
4 function [c1,c2]=equ(x1,x2,y1,y2)
5     c1 = (x1/x2)
6     c2 = (y1-y2)
7     endfunction
8 //Variable Declaration
9 x1 = 15 //real value
10 x2 = 2 //real value
11 y1 = 10 //angle in degree
12 y2 = 7 //angle in degree
13 x3 = 8 //real value
14 x4 = 16 //real value
15 y3 = 120 //angle in degree
16 y4 = -50 //angle in degree
17 //Calculation
18 [a,b]=equ(x1,x2,y1,y2) //function of
    division
19 [c,d]=equ(x3,x4,y3,y4) //function of
    division
20 //Results
21 printf("C1/C2 = %.1 f < %d degree\n",a,b)
22 printf("C1/C2 = %.1 f < %d degree\n",c,d)

```

Scilab code Exa 14.26 Example 26

```
1 //Chapter 14, Example 14.26
2 clc
3 //Polar to Rectangle conversion
4 function [r,i]= polar2rect(x,y)
5     r=x*cosd(y)
6     i=x*sind(y)
7 endfunction
8
9 //division
10 function [q,x]=division(x1,x2,y1,y2)
11     q=x1/x2
12     x=y1-y2
13 endfunction
14 //subtraction
15 function [c1,c2]=subtraction(x1,x2,y1,y2)
16     c1 = x1-x2
17     c2 = y1-y2
18 endfunction
19 //addition
20 function [c1,c2]=addition(x1,x2,y1,y2)
21     c1 = x1+x2
22     c2 = y1+y2
23 endfunction
24 //Rectangle to Polar Conversion
25 function [r,th] = rect2polar(x,y)
26     r=sqrt((x**2)+(y**2))
27     if x==0 then
28         th=90
29     else
30         th=atand(y/x)
31     end
32 endfunction
```

```

33 //multiplication
34 function [r,s] = voltdivider(x1,x2,y1,y2)
35     r=x1*x2
36     s=y1+y2
37 endfunction
38 //Variable Declaration
39 x1 = complex(2,3) //complex form of x1
40 x2 = complex(4,6) //complex form of x2
41 y1 = complex(7,7) //complex form of x3
42 y2 = complex(3,-3) //complex form of x4
43 x3 = 50 //real value
44 y3 = 30 //angle in degree
45 x4 = complex(5,5) //complex form of x5
46 x5 = 10 //real value
47 y5 = -20 //angle in degree
48 x6 = 2 //real value
49 y6 = 20 //angle in degree
50 x7 = complex(3,4) //complex form of x6
51 x8 = complex(8,-6) //complex form of x7
52 x9 = 3 //real value
53 y9 = 27 //angle in degree
54 x10 = 6 //real value
55 y10 = -40 //angle in degree
56 //Calculation
57 [a,b]=addition(real(x1),real(x2),imag(x1),imag(x2))
//function of addition
58 [c,d]=subtraction(real(y1),real(y2),imag(y1),imag(y2)
)) //function of subtraction
59 [aa,bb]=rect2polar(a,b) //function of conversion
60 [cc,dd]=rect2polar(c,d) //function of conversion
61 [e,f]=voltdivider(aa,cc,bb,-dd) //function of
multiplication
62 [g,h]=voltdivider(cc,cc,dd,-dd) //function of
multiplication
63 [m,n]=division(e,g,f,h) //function of division
64 [mm,nn]=polar2rect(m,n) //function of conversion
65 [o,p]=rect2polar(real(x4),imag(x4)) //function of
conversion

```

```

66 [oo,pp]=voltdivider(x3,o,y3,p) //function of
    multiplication
67 [ee,ff]=division(oo,x5,pp,y5) //function of
    division
68 [gg,hh]=voltdivider(x6,x6,y6,y6) //function of
    multiplication
69 [i,j]=rect2polar(real(x7),imag(x7)) //function of
    conversion
70 [ii,jj]=voltdivider(gg,i,hh,j) //function of
    multiplication
71 [s,t]=rect2polar(real(x8),imag(x8)) //function of
    conversion
72 [ss,tt]=division(ii,s,jj,t) //function of division
73 [v,w]=polar2rect(x9,y9) //function of conversion
74 [vv,ww]=polar2rect(x10,y10) //function of
    conversion
75 [ma,na]=subtraction(v,vv,w,ww) //function of
    subtraction
76
77 //Results
78 printf("a. = %.3f + j%.3f \n",mm,nn)
79 printf("b. = %.2f < %d degree \n",ee,ff)
80 printf("c. = %.1f < %d degree \n",ss,tt)
81 printf("d. = %.3f + j %.3f \n",ma,na)

```

Scilab code Exa 14.29 Example 29

```

1 //Chapter 14, Example 14.29
2 clc
3 function [t,p]=time2phasor(x,y)
4     t=x*0.707
5     p=y
6 endfunction
7
8 //Variable Declaration

```

```

 9 x1 = 50           //real value of time
10 y1 = 0           //angle in degree
11 x2 = 69.6       //peak voltage
12 y2 = 72         //angle in degree
13 x3 = 45         //peak voltage
14 y3 = 90         //angle in degree
15 //Calculation
16 [a,b]=time2phasor(x2,y2) //function of conversion
17 [c,d]=time2phasor(x3,y3) //function of conversion
18 //Results
19 printf("a. %d < %d degree \n",x1,y1)
20 printf("b. %.2f < %d degree \n",a,b)
21 printf("c. %.2f < %d degree \n",c,d)

```

Scilab code Exa 14.30 Example 30

```

1 //Chapter 14, Example 14.30
2 clc
3 function [m,n]=phasor2time(x,y)
4     m=sqrt(2)*x
5     n=y
6 endfunction
7
8 //Variable Declaration
9 x1= 10           //real value
10 y1 = 30         //angle in degree
11 x2 = 115        //real value
12 y2 = -70        //angle in degree
13
14 //Calculation
15 [a,b]=phasor2time(x1,y1) //function of conversion
16 [c,d]=phasor2time(x2,y2) //function of conversion
17
18 //Results
19 printf("a. I = %.2f sin(377t + %d degree)\n",a,b)

```

```
20 printf("b. V = %.1f sin(377t %d degree)\n",c,d)
```

Scilab code Exa 14.31 Example 31

```
1 //Chapter 14, Example 14.31
2 clc
3 funcprot(0)
4 //time to phasor
5 function [m,n]=time2phasor(x,y)
6     m=0.707*x
7     n=y
8 endfunction
9
10 //Polar to Rectangle conversion
11 function [r,i]= polar2rect(x,y)
12     r=x*cosd(y)
13     i=x*sind(y)
14 endfunction
15 //addition
16 function [c1,c2]=addition(x1,x2,y1,y2)
17     c1 = x1+y1
18     c2 = y2+x2
19 endfunction
20
21 //Rectangle to Polar Conversion
22 function [r,th] = rect2polar(x,y)
23     r=sqrt((x**2)+(y**2))
24     th=atand(y/x)
25 endfunction
26 //phasor to time domain
27 function [m,n]=phasor2time(x,y)
28     m=sqrt(2)*x
29     n=y
30 endfunction
31 //Variable Declaration
```

```

32 x1=50           //maximum voltage of Va
33 y1=30           //angle in degree of Va
34 x2=30           //maximum voltage of Vb
35 y2=60           //angle in degree of Vb
36 // Calculation
37 [a,b]=time2phasor(x1,y1) //function of conversion
38 [c,d]=time2phasor(x2,y2) //function of conversion
39 [e,f]=polar2rect(a,b) //function of conversion
40 [g,h]=polar2rect(c,d) //function of conversion
41 [i,j]=addition(e,f,g,h) //function of addition
42 [k,l]=rect2polar(i,j) //function of conversion
43 [o,p]=phasor2time(k,l) //function of conversion
44 // Results
45 printf("ein = %.2f sin(377t + %.2f degree)",o,p)

```

Scilab code Exa 14.32 Example 32

```

1 //Chapter 14, Example 14.32
2 clc
3 funcprot(0)
4 function [m,n]=time2phasor(x,y)
5     m=0.707*x
6     n=y
7 endfunction
8
9 //Polar to Rectangle conversion
10 function [r,i]= polar2rect(x,y)
11     r=x*cosd(y)
12     i=x*sind(y)
13 endfunction
14 //subtraction
15 function [c1,c2]=subtraction(x1,x2,y1,y2)
16     c1 = x1-y1
17     c2 = x2-y2
18 endfunction

```

```

19
20 //Rectangle to Polar Conversion
21 function [r,th] = rect2polar(x,y)
22     r=sqrt((x**2)+(y**2))
23     if x==0 then
24         th=-90
25     elseif x<0 then
26         th=atand(y/x)+180
27     else
28         th=atand(y/x)
29         if (x<0) & (y<0) then
30             th=th-180
31         end
32     end
33 endfunction
34 //phasor to time domain
35 function [m,n]=phasor2time(x,y)
36     m=sqrt(2)*x
37     n=y
38 endfunction
39 //Variable Declaration
40 x1= 120*10^-3 //maximum voltage of iT
41 y1=60 //angle in degree
42 x2 = 80*10^-3 //maximum voltage of i1
43 y2 = 0 //angle in degree
44 //Calculation
45 [a,b]=time2phasor(x1,y1) //function of conversion
46 [c,d]=time2phasor(x2,y2) //function of conversion
47 [e,f]=polar2rect(a,b) //function of conversion
48 [g,h]=polar2rect(c,d) //function of conversion
49 [i,j]=subtraction(e,f,g,h) //subtraction
50 [k,l]=rect2polar(i,j) //function of conversion
51 [o,p]=phasor2time(k,l) //function of conversion
52 //Results
53 printf("i2 = %.1f x 10^-3 sin(wt + %.2f degree) ",o
        *10^3,p)

```

Chapter 15

Series and Parallel ac Circuits

Scilab code Exa 15.1 Example 1

```
1 //Chapter 15, Example 15.1
2 clc
3 funcprot(0)
4 //time to phasor
5 function [m,n]=time2phasor(x,y)
6     m=0.707*x
7     n=y
8 endfunction
9
10 //phasor to time domain
11 function [m,n]=phasor2time(x,y)
12     m=sqrt(2)*x
13     n=y
14 endfunction
15 //Variable Declaration
16 x=100 //amplitude
17 y=0
18 r = 5 //resistor
19 y1=0
20 //Calculation
21 [a,b]=time2phasor(x,y)
```

```

22 i= a/r
23 [c]=phasor2time(i,y1)
24
25 //graph
26 x1 = 0 : 1 : 1000           //time in
    seconds
27 v = x*sin(377*x1)         //in voltage
28 ip = i*sin(377*x1)      //current in
    amperes
29 plot(x1, v)
30 plot(x1, ip)
31 xlabel("t")
32 ylabel("i V")
33
34 //Results
35 printf("i = %.1f sin wt", c)

```

Scilab code Exa 15.2 Example 2

```

1 //Chapter 15, Example 15.2
2 clc
3 //time to phasor
4 function [m,n]=time2phasor(x,y)
5     m=0.707*x
6     n=y
7 endfunction
8
9
10 //phasor to time domain
11 function [m,n]=phasor2time(x,y)
12     m=sqrt(2)*x
13     n=y
14 endfunction
15 //Variable Declaration
16 x=4           //amplitude

```

```

17 y=30 //in degrees
18 r=2 //in ohms
19
20 // Calculation
21 [a,b]=time2phasor(x,y)
22 c= a*r
23 [d,e]=phasor2time(c,y)
24
25 //graph
26 x1 = -5 : 1 : 20 //time in
    seocnds
27 v = d*sind((377*x1)+e) //voltage
28 ip = x*sind((377*x1)+e) //current in
    ampere
29 plot(x1, v)
30 plot(x1, ip)
31 xlabel("t")
32 ylabel("i V")
33 //Results
34 printf("v = %.1f sin(wt + %d degree)", d,e)

```

Scilab code Exa 15.3 Example 3

```

1 //Chapter 15, Example 15.3
2 clc
3 //time to phasor
4 function [m,n]=time2phasor(x,y)
5     m=0.707*x
6     n=y
7 endfunction
8
9
10 //phasor to time domain
11 function [m,n]=phasor2time(x,y)
12     m=sqrt(2)*x

```

```

13     n=y
14 endfunction
15 //Variable Declaration
16 x=24 //amplitude
17 y=0 //in degrees
18 r=3 //in ohms
19 d=90 //in degrees
20 //Calculation
21 [a,b]=time2phasor(x,y)
22 c= a/r
23 i=d
24 [f,e]=phasor2time(c,i)
25
26 //graph
27 x1 = 0 : 10 : 500 //time in
    seocnds
28 v = x*sind(x1) //voltage
29 ip = f*sind((x1)-d) //current in ampere
30 plot(x1, v)
31 plot(x1, ip)
32 xlabel("t")
33 ylabel("i V")
34 xtitle("Waveforms for Example 15.3")
35
36
37 //Results
38 printf("i = %.1f sin(wt - %d degree)", f,e)

```

Scilab code Exa 15.4 Example 4

```

1 //Chapter 15, Example 15.4
2 clc
3 //time to phasor
4 function [m,n]=time2phasor(x,y)
5     m=0.707*x

```

```

6     n=y
7     endfunction
8
9
10    //phasor to time domain
11    function [m,n]=phasor2time(x,y)
12        m=sqrt(2)*x
13        n=y
14    endfunction
15    //Variable Declaration
16    x=5           //amplitude
17    y=30         //in degrees
18    r=4          //in ohms
19    d=90         //in degrees
20    //Calculation
21    [a,b]=time2phasor(x,y)
22    c= a*r
23    i=b+d
24    [f,e]=phasor2time(c,i)
25
26    //graph
27    x1 = 0 : 1 : 500           //time in
        seocnds
28    v = f*sind((x1)+(d+y))    //voltage
29    ip = x*sind((x1)+y)       //current in ampere
30    plot(x1, v)
31    plot(x1, ip)
32    xlabel("t")
33    ylabel("i V")
34
35    //Results
36    printf("i = %d sin(wt + %d degree)", f,e)

```

Scilab code Exa 15.5 Example 5

```

1 //Chapter 15, Example 15.5
2 clc
3 //time to phasor
4 function [m,n]=time2phasor(x,y)
5     m=0.707*x
6     n=y
7 endfunction
8
9
10 //phasor to time domain
11 function [m,n]=phasor2time(x,y)
12     m=sqrt(2)*x
13     n=y
14 endfunction
15 //Variable Declaration
16 x=15 //amplitude
17 y=0 //in degrees
18 r=2 //in ohms
19 d=-90 //in degrees
20 //Calculation
21 [a,b]=time2phasor(x,y)
22 c= a/r
23 i=b-d
24 [f,e]=phasor2time(c,i)
25
26 //graph
27 x1 = 0 : 1 : 500 //time in
    seocnds
28 v = x*sind(x1) //voltage
29 ip = f*sind(x1-d) //current in ampere
30 plot(x1, v)
31 plot(x1, ip)
32 xlabel("t")
33 ylabel("i V")
34
35 //Results
36 printf("i = %.1f sin(wt + %d degree)", f,e)

```

Scilab code Exa 15.6 Example 6

```
1 //Chapter 15, Example 15.6
2 clc
3 //time to phasor
4 function [m,n]=time2phasor(x,y)
5     m=0.707*x
6     n=y
7 endfunction
8
9
10 //phasor to time domain
11 function [m,n]=phasor2time(x,y)
12     m=sqrt(2)*x
13     n=y
14 endfunction
15 //Variable Declaration
16 x=6 //amplitude
17 y=-60 //in degrees
18 r=0.5 //in ohms
19 d=-90 //in degrees
20
21 //Calculation
22 [a,b]=time2phasor(x,y)
23 c= a*r
24 i=b+d
25 [f,e]=phasor2time(c,i)
26
27
28 //graph
29 x1 = 0 : 1 : 500 //time in
    seocnds
30 v = f*sind((x1)+(d+y)) //voltage
31 ip = x*sind((x1)+ y) //current in
```

```

    ampere
32 plot(x1, v)
33 plot(x1, ip)
34 xlabel("t")
35 ylabel("i V")
36
37
38 //Results
39 printf("v = %.1f sin(wt%d degree)", f,e)

```

Scilab code Exa 15.7 Example 7

```

1 //Chapter 15, Example 15.7
2 clc
3 //Polar to Rectangle conversion
4 function [r,i]= polar2rect(x,y)
5     r=x*cosd(y)
6     i=x*sind(y)
7 endfunction
8
9 //addition
10 function [c1,c2]=addition(x1,x2,y1,y2)
11     c1 = x1+x2
12     c2 = y1+y2
13 endfunction
14 //Rectangle to Polar Conversion
15 function [r,th] = rect2polar(x,y)
16     r=sqrt((x**2)+(y**2))
17     th=atand(y/x)
18 endfunction
19 //Variable Declaration
20 r = 4 //resistance in
    ohms
21 ro = 0 //angle in degree
22 x1 = 8 //inductive

```



```

    reactance in ohms
23 xlo = 90 //angle in degree
24 //Calculation
25 [a,b]=polar2rect(r,ro) //Z1
26 [c,d]=polar2rect(xl,xlo) //Z2
27 [e,f]=addition(a,c,b,d) //Z1+Z2
28 [g,h]=rect2polar(e,f) //ZT
29
30 //Results
31 printf("ZT = %.3f ohm < %.2f degree",g,h)

```

Scilab code Exa 15.8 Example 8

```

1 //Chapter 15, Example 15.8
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     th=atand(y/x)
7 endfunction
8
9
10 //Polar to Rectangle conversion
11 function [r,i]= polar2rect(x,y)
12     r=x*cosd(y)
13     i=x*sind(y)
14 endfunction
15
16 //addition
17 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
18     c1 = x1+x2+x3
19     c2 = y1+y2+y3
20 endfunction
21
22 //Variable Declaration

```

```

23 r=6                                //resistance in
    ohms
24 ro = 0                             //angle in degree
25 xl = 10                             //in ohms
26 xlo = 90                            //angle in degree
27 xc = 12                             //in ohms
28 xco = -90                           //angle in degree
29 //Calculation
30 [a,b]=polar2rect(r,ro)              //Z1
31 [c,d]=polar2rect(xl,xlo)           //Z2
32 [e,f]=polar2rect(xc,xco)           //Z3
33 [g,h]=addition(a,c,e,b,d,f)        //Z1+Z2+Z3
34 [i,j]=rect2polar(g,h)              //ZT
35 //Results
36 printf("ZT = %.3f ohm < %.2f degree",i,j)

```

Scilab code Exa 15.9 Example 9

```

1 //Chapter 15, Example 15.9
2 clc
3 funcprot(0)
4
5 function [r,s] = voltdivider(x1,x2,y1,y2)
6     r=x1*x2
7     s=y1-y2
8 endfunction
9
10 //Polar to Rectangle conversion
11 function [r,i]= polar2rect(x,y)
12     r=x*cosd(y)
13     i=x*sind(y)
14 endfunction
15
16 function [c1,c2]=addition(x1,x2,y1,y2)
17     c1 = x1+x2

```

```

18     c2 = y1+y2
19 endfunction
20
21 //Rectangle to Polar Conversion
22 function [r,th] = rect2polar(x,y)
23     r=sqrt((x**2)+(y**2))
24     th=atand(y/x)
25 endfunction
26 //division
27 function [q,x]=division(x1,x2,y1,y2)
28     q=x1/x2
29     x=y1-y2
30 endfunction
31
32 //Variable Declaration
33 x1=4           //real value
34 x2=100        //real value
35 y1=-90        //angle in degree
36 y2=0          //angle in degree
37 r=3           //real value
38 x3=3          //real value
39 x4=100        //real value
40 y3=0          //angle in degree
41 y4=0          //angle in degree
42 //Calculation
43 [a,b]=voltdivider(x1,x2,y1,y2) //function of
    multiplication
44 [c,d]=polar2rect(x1,y1) //function of
    conversion
45 [e,f]=polar2rect(r,y2) //function of conversion
46 [i,j]=addition(e,f,c,d) //function of addition
47 [o,p]=rect2polar(i,j) //function of conversion
48 [m,n]=division(a,o,b,p) //function of division
49 [aa,ba]=voltdivider(x3,x4,y3,y4) //function of
    multiplication
50 [ma,na]=division(aa,o,ba,p) //function of
    division
51

```

```

52 //Results
53 printf("Vc = %d V < %.2f degree \n",m,n)
54 printf("Vr = %d V < %.2f degree",ma,na)

```

Scilab code Exa 15.10 Example 10

```

1 //Chapter 15, Example 15.10
2 clc
3 funcprot(0)
4 //addition
5 function [c1,c2]=addition1(x1,x2,y1,y2)
6     c1 = x1+x2
7     c2 = y1+y2
8     endfunction
9 //multiplication
10 function [r,s] = voltdivider(x1,x2,y1,y2)
11     r=x1*x2
12     s=y1+y2
13 endfunction
14
15 //Polar to Rectangle conversion
16 function [r,i]= polar2rect(x,y)
17     r=x*cosd(y)
18     i=x*sind(y)
19 endfunction
20 //addition
21 function [c1,c2,c3]=addition(x1,x2,x3,y1,y2,y3)
22     c1 = x1+x2+x3
23     c2 = y1+y2+y3
24 endfunction
25
26 //Rectangle to Polar Conversion
27 function [r,th] = rect2polar(x,y)
28     r=sqrt((x**2)+(y**2))
29     if x==0 then

```

```

30         th=-90
31     else
32         th=atand(y/x)
33     end
34 endfunction
35
36
37     function [q,x]=division(x1,x2,y1,y2)
38         q=x1/x2
39         x=y1-y2
40     endfunction
41
42 //Variable Declaration
43 x1=6 //real value of Zr
44 x2=50 //real value of E
45 y1=0 //angle in degree of Zr
46 y2=30 //angle in degree of E
47 r2=9 //real value of Zl
48 s2=90 //angle in degree of Zl
49 r3=17 //real value of Zc
50 s3=-90 //angle in degree of
    Zc
51 //Calculation
52 [a,b]=voltdivider(x1,x2,y1,y2) //function of
    multiplication
53 [c,d]=polar2rect(x1,y1) //function of
    conversion
54 [e,f]=polar2rect(r2,s2) //function of
    conversion
55 [ee,ff]=polar2rect(r3,s3) //function of
    conversion
56 [i,j]=addition(c,e,ee,d,f,ff) //function of
    addition
57 [o,p]=rect2polar(i,j) //function of
    conversion
58 [m,n]=division(a,o,b,p) //function of division
59 [aa,ba]=voltdivider(r2,x2,s2,y2) //function
    of conversion

```

```

60 [ma,na]=division(aa,o,ba,p)           //function of
    division
61 [ab,bb]=voltdivider(r3,x2,s3,y2)      //function
    of conversion
62 [mb,nb]=division(ab,o,bb,p)          //function of
    division
63 [za,ya]=addition1(e,ee,f,ff)         //function of
    addition
64 [zb,yb]=rect2polar(za,ya)            //function of
    conversion
65 [zc,yc]=voltdivider(zb,x2,yb,y2)     //function of
    multiplication
66 [zd,yd]=division(zc,o,yc,p)          //function of
    division
67
68 //Results
69 printf("Vr = %d V < %.2f degree \n",m,n)
70 printf("Vl = %d V < %.2f degree \n",ma,na)
71 printf("Vc = %d V < %.2f degree \n",mb,nb)
72 printf("V1 = %d V < %.2f degree",zd,yd)

```

Scilab code Exa 15.11 Example 11

```

1 //Chapter 15, Example 15.11
2 clc
3 funcprot(0)
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9
10 //addition
11 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
12     c1 = x1+x2+x3

```

```

13     c2 = y1+y2+y3
14 endfunction
15
16 //Rectangle to Polar Conversion
17 function [r,th] = rect2polar(x,y)
18     r=sqrt((x**2)+(y**2))
19     th=atand(y/x)
20 endfunction
21 //division
22 function [q,x]=division(x1,x2,y1,y2)
23     q=x1/x2
24     x=y1-y2
25 endfunction
26
27 //multiplication
28 function [r,s] = voltdivider(x1,x2,y1,y2)
29     r=x1*x2
30     s=y1+y2
31 endfunction
32 //Variable Declaration
33 r1=6 //resistance in ohms
34 r2=4 //resistance in ohms
35 l1 = 0.05 //inductance in henry
36 l2 = 0.05 //inductance in henry
37 c1 = 200*10^-6 //capacitance in farad
38 c2 = 200*10^-6 //capacitance in farad
39 w = 377 //in rad/s
40 z = 20 //source in volts
41 //Calculation
42 rt = r1+r2 //total resistance in ohms
43 yt = 0 //angle in degree
44 lt = l1+l2 //in henry
45 ct = c1/2 //in farad
46 xl = w*lt //in ohms
47 xy = 90 //angle in degree
48 xc = 1/(w*ct) //in ohms
49 cy = -90 //angle in degree
50 [a,b]=polar2rect(rt,yt) //function of

```

```

conversion
51 [c,d]=polar2rect(xl,xy) //function of
conversion
52 [e,f]=polar2rect(xc,cy) //function of
conversion
53 [i,j]=addition(a,c,e,b,d,f) //function of addition
54 [m,n]=rect2polar(i,j) //function of conversion
55 [o,p]=division(z,m,yt,n) //function of division
56 [w,ww]=voltdivider(o,rt,p,yt) //function of
multiplication
57 [x,y]=voltdivider(o,xl,p,xy) //function of
multiplication
58 [v,k]=voltdivider(o,xc,p,cy) //function of
multiplication
59 fp1 = cosd(48.16) //in degree
60 fp2 = (rt/m) //lagging
61 pt= z*o*fp1 //power in watt
62 q=x-v //in volts
63 e= sqrt((w*w)+(q*q)) //in volts
64 eo = 0 //in volts
65 [aa,bb]=voltdivider(rt,e,yt,eo) //function of
multiplication
66 [cc,dd]=division(aa,m,bb,n) //function of
division
67 [ee,ff]=voltdivider(xc,e,cy,eo) //function of
multiplication
68 [ii,jj]=division(ee,m,ff,n) //function of
division
69 //Results
70 printf("a. Zt = %d ohm < %.2f degree \n",m,n)
71 printf(" I = %.3f A < %.2f degree \n",o,p)
72 printf(" Vr = %.2f V < %.2f degree \n",w,ww)
73 printf(" Vl = %.2f V < %.2f degree \n",x,y)
74 printf(" Vc = %.2f V < %.2f degree \n",v,k)
75 printf("b. Fp = %.3f lagging \n",fp1)
76 printf(" Fp = %.3f lagging \n",fp2)
77 printf("c. Pt = %.2f W \n",pt)
78 printf("e. E = %d V \n",e)

```



```
79 printf(" f. Vr = %.1f V < %.2f degree \n",cc,dd)
80 printf(" Vc = %.2f V < %.2f degree",ii,jj)
```

Scilab code Exa 15.12 Example 12

```
1 //Chapter 15, Example 15.12
2 clc
3 funcprot(0)
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9 //adition
10 function [c1,c2]=addition(x1,x2,y1,y2)
11     c1 = x1+x2
12     c2 = y1+y2
13 endfunction
14
15 //Rectangle to Polar Conversion
16 function [r,th] = rect2polar(x,y)
17     r=sqrt((x**2)+(y**2))
18     th=atand(y/x)
19 endfunction
20 //division
21 function [q,x]=division(x1,x2,y1,y2)
22     q=x1/x2
23     x=y1-y2
24 endfunction
25
26 //multiplication
27 function [r,s] = voltdivider(x1,x2,y1,y2)
28     r=x1*x2
29     s=y1+y2
30 endfunction
```

```

31
32 //Function to create plot on complex plane
33 function complexPlot(x1,y1,x2,y2,x3,y3,z)
34     xpts1 = [0 x1];
35     ypts1 = [0 y1];
36
37     xpts2 = [0 x2];
38     ypts2 = [0 y2];
39
40     xpts3 = [0 x3];
41     ypts3 = [0 y3];
42     scf(1);
43     clf(1);
44     set(gca()," auto_clear"," off")
45     plot(xpts1,ypts1,'ro-')
46     plot(xpts2,ypts2,'ro-')
47     plot(xpts3,ypts3,'ro-')
48     set(gca()," auto_clear"," on")
49     a = gca();
50     a.x_location=" origin";
51     a.y_location=" origin";
52     set(gca()," data_bounds",matrix
53         ([-0.1,0.1,-0.2,0.2],2,-1))
54     xtitle(z);
55     xlabel(" Real");
56     ylabel(" Imaginary");
57 endfunction
58 //Variable Declaration
59 g = 0.05 //in admittance
60 d = 0 //in degrees
61 b1 = 0.1 //in admittance
62 d1 = -90 //in degrees
63 zr= 20 //in impedance
64 zro = 0 //in angle
65 z1 = 10 //in impedance
66 zlo = 90 //in degrees
67

```

```

68 // Calculation
69 [a,b]=polar2rect(g,d)
70 [c,d]=polar2rect(bl,d1)
71 [e,f]=addition(a,c,b,d)
72 [i,j]=rect2polar(e,f)
73 m= (1/i)
74 n= (-j)
75 [o,p]=voltagedivider(zr,zl,zro,zlo)
76 [q,r]=polar2rect(zr,zro)
77 [s,t]=polar2rect(zl,zlo)
78 [oo,pp]=addition(q,s,r,t)
79 [qq,rr]=rect2polar(oo,pp)
80 [ss,tt]=division(o,qq,p,rr)
81
82 // Plot
83 complexPlot(a,b,c,d,e,f," Admittance Diagram")
84
85 // Results
86 printf("Yr = %.2 f S + j%d \n",a,b)
87 printf("Yl = %d %.1 fj s \n",c,d)
88 printf("Yt = %.2 f S %.1 fj S \n",e,f)
89 printf("Zt = %.2 f ohm < %.2 f degree \n",m,n)
90 printf("Zt = %.2 f ohm < %.2 f degree \n",ss,tt)

```

Scilab code Exa 15.13 Example 13

```

1 //Chapter 15, Example 15.13
2 clc
3 funcprot(0)
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9 //adition

```

```

10 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
11     c1 = x1+x2+x3
12     c2 = y1+y2+y3
13 endfunction
14
15 //Rectangle to Polar Conversion
16 function [r,th] = rect2polar(x,y)
17     r=sqrt((x**2)+(y**2))
18     th=atand(y/x)
19 endfunction
20 //division
21 function [q,x]=division(x1,x2,y1,y2)
22     q=x1/x2
23     x=y1-y2
24 endfunction
25
26     //multiplication
27 function [r,s] = voltdivider(x1,x2,x3,y1,y2,y3)
28     r=x1*x2*x3
29     s=y1+y2+y3
30 endfunction
31
32 //multiplication
33 function [r,s] = voltdivider1(x1,x2,y1,y2)
34     r=x1*x2
35     s=y1+y2
36 endfunction
37
38
39 //Function to create plot on complex plane
40 function complexPlot(x1,y1,x2,y2,x3,y3,x4,y4,z)
41     xpts1 = [0 x1];
42     ypts1 = [0 y1];
43
44     xpts2 = [0 x2];
45     ypts2 = [0 y2];
46
47     xpts3 = [0 x3];

```

```

48     ypts3 = [0 y3];
49
50     xpts4 = [0 x4];
51     ypts4 = [0 y4];
52     scf(1);
53     clf(1);
54     set(gca()," auto_clear"," off")
55     plot(xpts1,ypts1,'ro-')
56     plot(xpts2,ypts2,'ro-')
57     plot(xpts3,ypts3,'ro-')
58     plot(xpts4,ypts4,'ro-')
59     set(gca()," auto_clear"," on")
60     a = gca();
61     a.x_location=" origin";
62     a.y_location=" origin";
63     set(gca()," data_bounds",matrix
        ([-0.1,0.25,-0.2,0.2],2,-1))
64     xtitle(z);
65     xlabel(" Real");
66     ylabel(" Imaginary");
67 endfunction
68
69 //Variable Declaration
70 r = 5
71 g = 1/r
72 x1 = 8
73 d = 0
74 b1 = 1/x1
75 d1 = -90
76 xc = 20
77 bc = 1/xc
78 dc = 90
79 zr= 5
80 zro = 0
81 z1 = 8
82 zlo = 90
83 zc = 20
84 zco = -90

```

```

85 // Calculation
86 [a,b]=polar2rect(g,d)
87 [c,d]=polar2rect(bl,d1)
88 [cc,dd]=polar2rect(bc,dc)
89 [e,f]=addition(a,c,cc,b,d,dd)
90 [i,j]=rect2polar(e,f)
91 m= (1/i)
92 n= (-j)
93 [o,p]=voltdivider(zr,zl,zc,zro,zlo,zco)
94 [mm,nn]=voltdivider1(zr,zl,zro,zlo)
95 [ma,na]=voltdivider1(zl,zc,zlo,zco)
96 [mb,nb]=voltdivider1(zr,zc,zro,zco)
97 [q,r]=polar2rect(mm,nn)
98 [s,t]=polar2rect(ma,na)
99 [sa,ta]=polar2rect(mb,nb)
100 [oo,pp]=addition(q,s,sa,r,t,ta)
101 [qq,rr]=rect2polar(oo,pp)
102 [ss,tt]=division(o,qq,p,rr)
103
104 // Plot
105 complexPlot(a,b,c,d,cc,dd,e,f," Admittance Diagram")
106
107
108 // Results
109 printf(" a. Yr = %.1f S + j%d \n",a,b)
110 printf("    Yl = %d %.3fj s \n",c,d)
111 printf("    Yc = %d + j%.3f S \n",cc,dd)
112 printf(" b. Yt = %.2f S < %.3f degree \n",i,j)
113 printf(" c. Zt = %.2f ohm < %.2f degree \n",m,n)
114 printf("    Zt = %.2f ohm < %.2f degree \n",ss,tt)

```

Scilab code Exa 15.14 Example 14

```

1 //Chapter 15, Example 15.14
2 clc

```

```

3 //real and imaginary part
4 function [r,i]=realnimaginary(x,y)
5     r= x/((x*x)+(y*y))
6     i= y/((x*x)+(y*y))
7 endfunction
8
9
10 //Variable Declaration
11 r = 6 //real part of Z
12 xc = 8 //imaginary part of Z
13 r1 = 10 //real part of Z
14 xl = 4 //imaginary part of Z
15 xc1 = -0.1 //imaginary part of Z
16
17 //Calculation
18 [a,b]=realnimaginary(r,xc) //function
19 xo = xl+xc1 //imaginary part of Z
20 [c,d]=realnimaginary(r1,xo) //function
21 //Results
22 printf("a. Y = %.2f S + j%.2f S \n",a,b)
23 printf("b. Y = %.3f S - j%.3f S ",c,d)

```

Scilab code Exa 15.15 Example 15

```

1 //Chapter 15, Example 15.15
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8
9 //Rectangle to Polar Conversion
10 function [r,th] = rect2polar(x,y)
11     r=sqrt((x**2)+(y**2))

```

```

12     th=atand(y/x)
13 endfunction
14
15 //Polar to Rectangle conversion
16 function [r,i]= polar2rect(x,y)
17     r=x*cosd(y)
18     i=x*sind(y)
19 endfunction
20
21 function [c1,c2]=addition(x1,x2,y1,y2)
22     c1 = x1+x2
23     c2 = y1+y2
24 endfunction
25
26 //division
27 function [q,x]=division(x1,x2,y1,y2)
28     q=x1/x2
29     x=y1-y2
30 endfunction
31 //Variable Declaration
32 z1 = 4           //real value of Z1
33 zlo = 90        //angle in degree
34 it = 20        //real value of It
35 ito = 0        //angle in degree
36 zr = 3         //real value of Zr
37 zro = 0        //angle in degree
38
39 //Calculation
40 [a,b]=voldivider(z1,it,zlo,ito) //function of
    multiplication
41 [c,d]=polar2rect(zr,zro) //function of
    conversion
42 [e,f]=polar2rect(z1,zlo) //function of
    conversion
43 [i,j]=addition(c,d,e,f) //function of addition
44 [o,p]=rect2polar(i,j) //function of
    conversion
45 [m,n]=division(a,o,b,p) //function of division

```



```

46 [aa,bb]=voldivider(zr,it,zro,ito) //function of
    multiplication
47 [mm,nn]=division(aa,o,bb,p) //function of
    division
48 //Results
49 printf("Ir = %d A < %.2f degree \n",m,n)
50 printf("It = %d A < %.2f degree \n",mm,nn)

```

Scilab code Exa 15.16 Example 16

```

1 //Chapter 15, Example 15.16
2 clc
3 //division
4 function [q,x]=division(x1,x2,y1,y2)
5     q=x1/x2
6     x=y1-y2
7     endfunction
8 //Rectangle to Polar Conversion
9 function [r,th] = rect2polar(x,y)
10    r=sqrt((x**2)+(y**2))
11    if x==0 then
12        th=90
13    else
14        th=atand(y/x)
15    end
16 endfunction
17
18 //multiplication
19 function [r,s] = voldivider(x1,x2,y1,y2)
20    r=x1*x2
21    s=y1+y2
22 endfunction
23 //Variable Declaration
24 zc= 2 //real value of Zc
25 zco = -90 //angle in degree

```

```

26 it = 5 //real value of It
27 itt = 30 //angle in degree
28 zr = 1 //real part of Zrl
29 zl = 8 //imaginary part of Zl
30 xx = zl-zc //imaginary part of Zcrl
31 //Calculation
32 [a,b]=voldivider(zc,it,zco,itt) //function of
multiplication
33 [c,d]=rect2polar(zr,xx) //function of
conversion
34 [e,f]=division(a,c,b,d) //function of division
35 [m,n]=rect2polar(zr,zl) //function of
conversion
36 [i,j]=voldivider(m,it,n,itt) //function of
multiplication
37 [s,t]=division(i,c,j,d) //function of division
38 //Results
39 printf("Irl = %.3f A < %.2f degree \n",e,f)
40 printf("Ic = %.3f A < %.2f degree",s,t)

```

Scilab code Exa 15.17 Example 17

```

1 //Chapter 15, Example 15.17
2 clc
3 //Variable Declaration
4 rp = 8*10^3 //R in kilohms
5 xl = 9*10^3 //Xl in kilohms
6 xc = 4*10^3 //Xc in kilohms
7
8
9 //Calculation
10 xp = xl-xc //in kilohms
11 rs = (rp*(xp*xp))/((xp*xp)+(rp*rp)) //in kilohms
12 xs = ((rp*rp)*xp)/((xp*xp)+(rp*rp)) //in kilohms
13

```

```

14 //Results
15 printf("Rs = %.3 f kohm \n",rs/10^3)
16 printf("Xs = %.3 f kohm ",xs/10^3)

```

Scilab code Exa 15.18 Example 18

```

1 //Chapter 15, Example 15.18
2 clc
3 funcprot(0)
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
10     c1 = x1+x2+x3
11     c2 = y1+y2+y3
12     endfunction
13 //Rectangle to Polar Conversion
14 function [r,th] = rect2polar(x,y)
15     r=sqrt((x**2)+(y**2))
16     th=atand(y/x)
17 endfunction
18
19 //division
20 function [q,x]=division(x1,x2,y1,y2)
21     q=x1/x2
22     x=y1-y2
23     endfunction
24
25     //multiplication
26 function [r,s] = voltdivider(x1,x2,y1,y2)
27     r=x1*x2
28     s=y1+y2
29 endfunction

```

```

30
31 function [c1,c2]=addition1(x1,x2,y1,y2)
32     c1 = x1+x2
33     c2 = y1+y2
34     endfunction
35 //Variable Declaration
36 r1 = 10 //resistance1 in ohms
37 r2 = 40 //resistance2 in ohms
38 l1 = 6*10^-3 //inductance1 in
    millihenry
39 l2 = 12*10^-3 //inductance2 in
    millihenry
40 c1 = 80*10^-6 //capacitance1 in
    microfarad
41 c2 = 20*10^-6 //capacitance1 in
    microfarad
42 w = 1000 //in rad/s
43 ia= 12 //current source in
    ampere
44 iao = 0 //angle in degree
45 //Calculation
46 rt = (r1*r2)/(r1+r2) //total resistance in
    ohms
47 lt = (l1*l2)/(l1+l2) //total inductance in
    millihenry
48 ct = c1+c2 //total capacitance in
    microfarad
49 xl = w*lt //reactance in ohms
50 xlo = 90 //angle in degree
51 xc = 1/(w*ct) //reactance in ohms
52 g = 1/rt //real value of Yr
53 go = 0 //angle in degree
54 bl = 1/xl //real value of Bl
55 blo = -90 //angle in degree
56 bc = 1/xc //real value of Bc
57 bco = 90 //angle in degree
58 [a,b]=polar2rect(g,go) //function of conversion
59 [c,d]=polar2rect(bl,blo) //function of conversion

```

```

60 [e,f]=polar2rect(bc,bco) //function of conversion
61 [i,j]=addition(a,c,e,b,d,f) //function of addition
62 [m,n]=rect2polar(i,j) //function of conversion
63 [mm,nn]=division(ia,m,iao,n) //function of division
64 [o,p]=division(mm,xl,nn,xlo) //function of division
65 fp = (g/m) //lagging
66 po = mm*ia*cosd(nn) //power in watt
67 zt1 = 1/m //real vlue in ohms of
    Zt
68 zt2 = -n //angle in degree
69 [vv,ww]=polar2rect(zt1,zt2) //function of
    conversion
70 l=vv/w //inductance in
    millihenry
71 [ss,tt]=voldivider(ia,zt1,iao,zt2) //function of
    multiplication
72 ppo= (ia*ia)*vv //power in watt
73 rp = ((vv*vv)+(ww*ww))/vv //resistance in ohms
74 xp = ((vv*vv)+(ww*ww))/ww //resistance in ohms
75 g1=1/rp //real value in ohms of
    Yt
76 g1o=0 //angle in degree
77 b11=1/xp //real value
78 b11o=-90 //angle in degree
79 [za,zb]=polar2rect(g1,go) //function of
    conversion
80 [zc,zd]=polar2rect(b11,b11o) //function of
    conversion
81 [ze,zf]=addition1(za,zc,zb,zd) //function of
    addition
82 [zx,zy]=rect2polar(ze,zf) //function of
    conversion
83 //Results
84 printf("Yt = %.3f S < %.3f degree \n",m,n)
85 printf("E = %.3f V < %.3f degree \n",mm,nn)
86 printf("IL = %.3f A < %.3f degree \n",o,p)
87 printf("Fp = %.3f lagging \n",fp)
88 printf("P = %.2f W \n",po)

```

```
89 printf("Zt = %.3f ohm + j%.3f ohm \n",vv,ww)
90 printf("L = %.3f mH \n",l*10^3)
91 printf("E = %.3f V < %.3f degree \n",ss,tt)
92 printf("P = %.2f W \n",ppo)
93 printf("Rp = %d ohm \n",rp)
94 printf("Xp = %.3f ohm \n",xp)
95 printf("Yt = %.3f S < %.3f degree \n",zx,zy)
```

Chapter 16

Series Parallel ac Networks

Scilab code Exa 16.1 Example 1

```
1 //Chapter 16, Example 16.1
2 clc
3
4 //multiplication
5 function [r,s] = voltdivider(x1,x2,y1,y2)
6     r=x1*x2
7     s=y1+y2
8 endfunction
9 //Rectangle to Polar Conversion
10 function [r,th] = rect2polar(x,y)
11     r=sqrt((x**2)+(y**2))
12     th=atand(y/x)
13 endfunction
14 //Polar to Rectangle conversion
15 function [r,i]= polar2rect(x,y)
16     r=x*cosd(y)
17     i=x*sind(y)
18 endfunction
19 //division
20 function [q,x]=division(x1,x2,y1,y2)
21     q=x1/x2
```

```

22         x=y1-y2
23     endfunction
24     //addition
25     function [c1,c2]=addition(x1,x2,y1,y2)
26         c1 = x1+x2
27         c2 = y1+y2
28     endfunction
29 //Variable Declaration
30 z1 = 1           //real value of impedancel
31 z1o = 0         //angle in degree
32 xc = 2          //real value of Xc
33 xco = -90       //angle in degree
34 x1 = 3          //real value of Xl
35 x1o = 90        //angle in degree
36 ev = 120        //real value of E
37 eo = 0          //angle in degree
38
39
40 //Calculation
41 [a,b]=voltdivider(xc,x1,xco,x1o) //function of
    multiplication
42 x= -xc+x1       //real value
43 xo= 90          //angle in degree
44 [c,d]=division(a,x,b,xo) //function of division
45 [aa,bb]=polar2rect(z1,z1o) //function of conversion
46 [cc,dd]=polar2rect(c,d) //function of conversion
47 [e,f]=addition(aa,cc,bb,dd) //function of addition
48 [ee,ff]=rect2polar(e,f) //function of conversion
49 [m,n]=division(ev,ee,eo,ff) //function of
    division
50 [v,w]=voltdivider(m,z1,n,z1o) //function of
    multiplication
51 [vv,ww]=voltdivider(m,c,n,d) //function of
    multiplication
52 [q,r]=division(vv,xc,ww,xco) //function of
    division
53 pdel = (m*m)*z1 //power in watt
54 fp = cosd(n)    //lagging

```



```

55 //Results
56 printf("a. Zt = %.2f ohm < %.2f degree \n",ee,ff)
57 printf("b. Is = %.2f A < %.2f degree \n",m,n)
58 printf("c. Vr = %.2f V < %.2f degree \n",v,w)
59 printf("   Vc = %.2f V < %.2f degree \n",vv,ww)
60 printf("d. Ic = %.2f A < %.2f degree \n",q,r)
61 printf("e. Pdel = %.2f W \n",pdel)
62 printf("f. Fp = %.3f leading \n",fp)

```

Scilab code Exa 16.2 Example 2

```

1 //Chapter 16, Example 16.2
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8 //Rectangle to Polar Conversion
9 function [r,th] = rect2polar(x,y)
10    r=sqrt((x**2)+(y**2))
11    if x==0 then
12        th=-90
13    else
14        th=atand(y/x)
15    end
16 endfunction
17 //Polar to Rectangle conversion
18 function [r,i]= polar2rect(x,y)
19    r=x*cosd(y)
20    i=x*sind(y)
21 endfunction
22 //addition
23 function [c1,c2]=addition(x1,x2,y1,y2)
24    c1 = x1+x2

```

```

25     c2 = y1+y2
26 endfunction
27 //division
28 function [q,x]=division(x1,x2,y1,y2)
29     q=x1/x2
30     x=y1-y2
31 endfunction
32
33 //Variable Declaration
34 x=complex(3,4) //complex form of Z1
35 y=complex(0,-8) //complex form of Z2
36 i = 50 //real value of current in
    ampere
37 io = 30 //angle in degree
38 //Calculation
39
40 [a,b]=rect2polar(real(x),imag(x)) //function of
    conversion
41 [c,d]=rect2polar(real(y),imag(y)) //function of
    conversion
42 [aa,bb]=voldivider(c,i,d,io) //function og
    multiplication
43 [cc,dd]=addition(real(y),real(x),imag(y),imag(x))
44 [e,f]=rect2polar(cc,dd) //function of conversion
45 [ee,ff]=division(aa,e,bb,f) //function of division
46 [ac,bc]=voldivider(a,i,b,io) //function og
    multiplication
47 [ea,fa]=division(ac,e,bc,f) //function of division
48 [m,n]=polar2rect(ee,ff) //function of conversion
49 [mm,nn]=polar2rect(ea,fa) //function of
    conversion
50 [ma,na]=addition(m,mm,n,nn) //function of addition
51 [mb,nb]=rect2polar(ma,na) //function of
    conversion
52 //Results
53 printf("a. I1 = %d A < %.2f degree \n",ee,ff)
54 printf("b. I2 = %d A < %.2f degree \n",ea,fa)
55 printf("c. %d A < %d degree = %d A < %d degree (

```

```
checks) ",mb,nb,mb,nb)
```

Scilab code Exa 16.3 Example 3

```
1 //Chapter 16, Example 16.3
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=-90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12 //Rectangle to Polar Conversion
13 function [r,th] = rect2polar1(x,y)
14     r=sqrt((x**2)+(y**2))
15     if x==0 then
16         th=90
17     else
18         th=atand(y/x)
19     end
20 endfunction
21
22 //Polar to Rectangle conversion
23 function [r,i]= polar2rect(x,y)
24     r=x*cosd(y)
25     i=x*sind(y)
26 endfunction
27
28 //addition
29 function [c1,c2]=addition(x1,x2,y1,y2)
30     c1 = x1+x2
31     c2 = y1+y2
```

```

32     endfunction
33
34     //multiplication
35 function [r,s] = voltdivider(x1,x2,y1,y2)
36     r=x1*x2
37     s=y1+y2
38 endfunction
39
40 //division
41 function [q,x]=division(x1,x2,y1,y2)
42     q=x1/x2
43     x=y1-y2
44 endfunction
45 //Variable Declaration
46 z1=complex(5,0)           //complex form of Z1
47 z2= complex(0,-12)       //complex form of Z2
48 z3 = complex(0,8)        //complex form of Z3
49 e1 = 20                   //real part of voltage
50 e1o = 20                   //angle in degree
51
52 //Calculation
53 [a,b]=rect2polar(real(z2),imag(z2)) //function of
    conversion
54 [aa,bb]=voltdivider(a,e1,b,e1o) //function of
    multiplication
55 [c,d]=addition(real(z1),real(z2),imag(z1),imag(z2))
    //function of addition
56 [cc,dd]=rect2polar(c,d) //function of conversion
57 [e,f]=division(aa,cc,bb,dd) //function of
    division
58 [ee,ff]=rect2polar1(real(z3),imag(z3)) //function
    of conversion
59 [g,h]=division(e1,ee,e1o,ff) //function of
    division
60 [gg,hh]=division(e1,cc,e1o,dd) //function of
    division
61 [o,p]=polar2rect(g,h) //function of conversion
62 [oo,pp]=polar2rect(gg,hh) //function of

```

```

        conversion
63 [m,n]=addition(o,oo,p,pp) //function of addition
64 [mm,nn]=rect2polar(m,n) //function of conversion
65 //Results
66 printf("a. Vc = %.2f V < %.2f degree \n",e,f)
67 printf("b. I1 = %.1f A < %.2f degree \n",g,h)
68 printf(" I2 = %.2f A < %.2f degree \n",gg,hh)
69 printf(" Is = %.2f A < %.2f degree \n",mm,nn)

```

Scilab code Exa 16.4 Example 4

```

1 //Chapter 16, Example 16.4
2 clc
3
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9 //Rectangle to Polar Conversion
10 function [r,th] = rect2polar(x,y)
11     r=sqrt((x**2)+(y**2))
12     th=atand(y/x)
13 endfunction
14 //subtraction
15 function [c1,c2]=subtraction(x1,x2,y1,y2)
16     c1 = x1-x2
17     c2 = y1-y2
18     endfunction
19 //multiplication
20 function [r,s] = voltdivider(x1,x2,y1,y2)
21     r=x1*x2
22     s=y1+y2
23 endfunction
24 function [c1,c2]=addition(x1,x2,y1,y2)

```

```

25     c1 = x1+x2
26     c2 = y1+y2
27     endfunction
28
29 //division
30 function [q,x]=division(x1,x2,y1,y2)
31     q=x1/x2
32     x=y1-y2
33     endfunction
34 //Variable Declaration
35 x=complex(3,4)      //complex form of Z1
36 y=complex(8,-6)   //complex form of Z2
37 ea = 100           //source in volts
38 eo = 0             //angle in degree
39 xo = 90
40 //Calculation
41 [a,b]=rect2polar(real(x),imag(x)) //function of
    conversion
42 [c,d]=rect2polar(real(y),imag(y)) //function of
    conversion
43 [e,f]=voltdivider(a,c,b,d) //function of
    conversion
44 [g,h]=addition(real(x),real(y),imag(x),imag(y))
45 [gg,hh]=rect2polar(g,h) //function of conversion
46 [i,j]=division(e,gg,f,hh) //function of division
47 [k,l]=division(ea,i,eo,j) //function of division
48 [m,n]=division(ea,a,eo,b) //function of division
49 [mm,nn]=division(ea,c,eo,d) //function of division
50 [o,p]=voltdivider(m,real(x),n,eo) //function of
    multiplication
51 [oo,pp]=voltdivider(mm,real(y),nn,eo) //function of
    multiplication
52 [q,r]=voltdivider(real(x),ea,eo,eo) //function of
    multiplication
53 [s,t]=polar2rect(real(x),eo) //function of
    conversion
54 [ss,tt]=polar2rect(imag(x),xo) //function of
    conversion

```

```

55 [sa,ta]=addition(s,ss,t,tt)      //function of
    addition
56 [sb,tb]=rect2polar(sa,ta)
57 [qa,ra]=division(q,sb,r,tb)    //function of
    division
58 [v,w]=polar2rect(oo,pp)        //function of conversion
59 [vv,ww]=polar2rect(o,p)        //function of conversion
60 [va,wa]=subtraction(v,vv,w,ww) //function of
    subtraction
61 [vb,wb]=rect2polar(va,wa)      //function of conversion
62 //Results
63 printf("Zt = %.3f ohm < %.2f degree \n",i,j)
64 printf("Is = %.2f A < %.2f degree \n",k,l)
65 printf("I1 = %d A < %.2f degree \n",m,n)
66 printf("I2 = %d A < %.2f degree \n",mm,nn)
67 printf("Vr1 = %d V < %.2f degree \n",o,p)
68 printf("Vr2 = %d V < %.2f degree \n",oo,pp)
69 printf("Vr1 = %d V < %.2f degree \n",qa,ra)
70 printf("Vab = %d V < %.2f degree \n",vb,wb)

```

Scilab code Exa 16.5 Example 5

```

1 //Chapter 16, Example 16.5
2 clc
3
4 //multiplication
5 function [r,s] = voltdivider(x1,x2,y1,y2)
6     r=x1*x2
7     s=y1+y2
8 endfunction
9 function [c1,c2]=addition(x1,x2,y1,y2)
10     c1 = x1+x2
11     c2 = y1+y2
12 endfunction
13 //Rectangle to Polar Conversion

```

```

14 function [r,th] = rect2polar(x,y)
15     r=sqrt((x**2)+(y**2))
16     th=atand(y/x)
17 endfunction
18
19 //division
20 function [q,x]=division(x1,x2,y1,y2)
21     q=x1/x2
22     x=y1-y2
23 endfunction
24 //Variable Declaration
25 ra = 50*10^3           //real value of Ra
26 rao = 0                //angle in degree
27 rb = 3.3*10^3         //real value of Rb
28 rbo = 0                //angle in degree
29 x=complex(1*10^3,-159.16) //complex form of Z2
30 ia = 4*10^-3          //real value of current
    in milliampere
31 iao = 0                //angle in degree
32 y=complex(1*10^3,-0.796) //complex form of Z2
33
34 //Calculation
35 z1 = (ra*rb)/(ra+rb)   //Z1 in kilo ohms
36 [a,b]=voltagedivider(z1,ia,rao,iao) //function of
    multiplication
37 [c,d]=addition(z1,real(x),rao,imag(x))
38 [g,h]=rect2polar(c,d) //function of conversion
39 [j,k]=division(a,g,b,h) //function of division
40 [m,n]=voltagedivider(j,real(x),k,rao) //function of
    multiplication
41 [cc,dd]=addition(z1,real(y),rao,imag(y))
42 [gg,hh]=rect2polar(cc,dd)
43 [jj,kk]=division(a,gg,b,hh) //function of division
44 [mm,nn]=voltagedivider(jj,real(y),kk,rao) //function
    of multiplication
45
46
47 //Results

```



```

48 printf("a. VL = %.3f V < %.3f degree \n",m,n)
49 printf("b. VL = %.3f V < %.3f degree \n",mm,nn)

```

Scilab code Exa 16.6 Example 6

```

1 //Chapter 16, Example 16.6
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8 function [c1,c2]=addition(x1,x2,y1,y2)
9     c1 = x1+x2
10    c2 = y1+y2
11    endfunction
12
13 //division
14 function [q,x]=division(x1,x2,y1,y2)
15     q=x1/x2
16     x=y1-y2
17    endfunction
18 //Rectangle to Polar Conversion
19 function [r,th] = rect2polar(x,y)
20     r=sqrt((x**2)+(y**2))
21     th=atand(y/x)
22 endfunction
23
24 //Variable Declaration
25 it = 2.626*10^-3 //real value of current in
    milliampere
26 ito = 51.02 //angle in degree
27 z1 = 1.545*10^3 //real value of Z1
28 z1o = 0 //angle in degree
29 x=complex(10*10^3,-20*10^3) //complex form of Z2

```

```

30
31 // Calculation
32 [a,b]=voldivider(z1,it,z1o,it0) //function of
    multiplication
33 [c,d]=addition(z1,real(x),z1o,imag(x)) //function
    of addition
34 [e,f]=rect2polar(c,d) //function of conversion
35 [m,n]=division(a,e,b,f) //function of division
36 [o,p]=rect2polar(real(x),imag(x)) //function of
    conversion
37 [mm,nn]=voldivider(m,o,n,p) //function of
    multiplication
38
39
40
41 // Results
42 printf("a. I = %.3f mA < %.3f degree \n",m*10^3,n)
43 printf("b. V = %.3f V < %.3f degree \n",mm,nn)

```

Scilab code Exa 16.7 Example 7

```

1 //Chapter 16, Example 16.7
2 clc
3 //Polar to Rectangle conversion
4 function [r,i]= polar2rect(x,y)
5     r=x*cosd(y)
6     i=x*sind(y)
7 endfunction
8 //Rectangle to Polar Conversion
9 function [r,th] = rect2polar(x,y)
10    r=sqrt((x**2)+(y**2))
11    th=atand(y/x)
12 endfunction
13 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
14    c1 = x1+x2+x3

```

```

15     c2 = y1+y2+y3
16 endfunction
17     //multiplication
18 function [r,s] = voltdivider(x1,x2,y1,y2)
19     r=x1*x2
20     s=y1+y2
21 endfunction
22
23 //division
24 function [q,x]=division(x1,x2,y1,y2)
25     q=x1/x2
26     x=y1-y2
27 endfunction
28
29
30 //Variable Declaration
31 z1 = 10           //real value of Z1
32 z1o = 0          //angle in degree
33 z2 = complex(3,4) //complex form of Z2
34 z3 = complex(8,-6) //complex form of Z3
35 ea = 200         //source voltage in volts
36 eo = 0           //angle in degree
37 //Calculation
38 [a,b]=rect2polar(real(z2),imag(z2)) //function of
    conversion
39 [c,d]=rect2polar(real(z3),imag(z3)) //function of
    conversion
40 y1=1/z1          //admittance1
41 y1o = 0          //angle in degree
42 y2 = 1/a         //admittance2
43 ya2 = -b         //angle in degree
44 y3 = 1/c         //admittance3
45 ya3 = -d         //angle in degree
46 [e,f]=polar2rect(y2,ya2) //function of conversion
47 [g,h]=polar2rect(y3,ya3) //function of conversion
48 [m,n]=addition(y1,e,g,y1o,f,h) //function of
    addition
49 [o,p]=rect2polar(m,n) //function of conversion

```

```

50 [s,t]=voltdivider(ea,o,eo,p)      //function of
    multiplication
51 [sa,ta]=division(ea,z1,eo,z1o)   //function of
    division
52 [sb,tb]=division(ea,a,eo,b)     //function of division
53 [sc,tc]=division(ea,c,eo,d)     //function of division
54 zt1 = 1/o                        //real value of Zt
55 zt2 = -p                          //angle in degree
56 [v,w]=polar2rect(s,t)           //function of conversion
57 [va,wa]=polar2rect(sa,ta)       //function of conversion
58 [vb,wb]=polar2rect(sb,tb)       //function of conversion
59 [vc,wc]=polar2rect(sc,tc)       //function of conversion
60 [vv,ww]=addition(va,vb,vc,wa,wb,wc) //function of
    addition
61 //Results
62 printf("a. I = %.1f A < %.3f degree \n",s,t)
63 printf("b. I1 = %d A < %d degree \n",sa,ta)
64 printf("   I2 = %d A < %.2f degree \n",sb,tb)
65 printf("   I3 = %d A < %.2f degree \n",sc,tc)
66 printf("c. %d %dj = %d %dj (checks)\n",v,w,vv,ww)
67 printf("d. Zt = %.3f ohm < %.3f degree \n",zt1,zt2)

```

Scilab code Exa 16.8 Example 8

```

1 //Chapter 16, Example 16.8
2 clc
3 funcprot(0)
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)
6     r=sqrt((x**2)+(y**2))
7     th=atand(y/x)
8 endfunction
9 //multiplication
10 function [r,s] = voltdivider(x1,x2,y1,y2)
11     r=x1*x2

```

```

12     s=y1+y2
13 endfunction
14
15 //division
16 function [q,x]=division(x1,x2,y1,y2)
17     q=x1/x2
18     x=y1-y2
19 endfunction
20
21 //Polar to Rectangle conversion
22 function [r,i]= polar2rect(x,y)
23     r=x*cosd(y)
24     i=x*sind(y)
25 endfunction
26
27 function [c1,c2]=addition(x1,x2,y1,y2)
28     c1 = x1+x2
29     c2 = y1+y2
30 endfunction
31
32 function [c1,c2]=subtraction(x1,x2,y1,y2)
33     c1 = x1-x2
34     c2 = y1-y2
35 endfunction
36 //Variable Declaration
37 z1 = 4                //real value of Z1
38 z1o = 0              //angle in degree
39 z2=complex(9,-7)     //complex form of Z2
40 z3=complex(8,6)     //complex form of Z3
41 ev = 100            //source voltage in volts
42 eo = 0              //angle in degree
43
44 //Calculation
45 [a,b]=rect2polar(real(z2),imag(z2)) //function
    of conversion
46 [c,d]=rect2polar(real(z3),imag(z3)) //function
    of conversion
47 [e,f]=voltdivider(a,c,b,d) //function of

```

```

    multiplication
48 [ee,ff]=addition(real(z2),real(z3),imag(z2),imag(z3)
    )
49 [m,n]=rect2polar(ee,ff) //function of
    conversion
50 [ea,fa]=division(e,m,f,n) //function of division
51 [eb,fb]=polar2rect(ea,fa) //function of
    conversion
52 [g,h]=addition(z1,eb,z1o,fb) //function of
    addition
53 [gg,hh]=rect2polar(g,h) //function of
    conversion
54 [o,p]=division(ev,gg,eo,hh) //function of
    division
55 fp=gg/gg
56 [i,j]=voltdivider(a,o,b,p) //function of
    multiplication
57 [q,r]=division(i,m,j,n) //function of division
58 [qa,ra]=polar2rect(q,r) //function of
    conversion
59 [qb,rb]=polar2rect(o,p) //function of
    conversion
60 [s,t]=subtraction(qb,qa,rb,ra) //function of
    subtraction
61 [ss,tt]=rect2polar(s,t) //function of
    conversion
62 pt = ev*o*cosd(1.5) //power in watt
63 //Results
64 printf("a. Zt = %.3f ohm < %.1f degree \n",gg,hh)
65 printf("b. I = %.2f A < %.1f degree \n",o,p)
66 printf("c. Fp = %d \n",fp)
67 printf("I2 = %.2f A < %d degree \n",q,r)
68 printf("I1 = %.1f A < %.2f degree \n",ss,tt)
69 printf("Pt = %.2f W \n",pt)

```

Chapter 17

Methods of Analysis and Selected Topics ac

Scilab code Exa 17.1 Example 1

```
1 //Chapter 17, Example 17.1
2 clc
3 //division
4 function [q,x]=division(x1,x2,y1,y2)
5     q=x1/x2
6     x=y1-y2
7     endfunction
8 //Variable Declaration
9 ea = 100           //source voltage in volts
10 eo = 0           //angle in degree
11 za = 5           //real value of Z
12 zo = 53.13       //angle in degree
13 //Calculation
14 [a,b]=division(ea,za,eo,zo) //function of division
15
16
17
18 //Results
19 printf("I = %d A < %.2f degree \n",a,b)
```

Scilab code Exa 17.2 Example 2

```
1 //Chapter 17, Example 17.2
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12 //multiplication
13 function [r,s] = voltdivider(x1,x2,y1,y2)
14     r=x1*x2
15     s=y1+y2
16 endfunction
17
18 //Polar to Rectangle conversion
19 function [r,i]= polar2rect(x,y)
20     r=x*cosd(y)
21     i=x*sind(y)
22 endfunction
23
24 function [c1,c2]=addition(x1,x2,y1,y2)
25     c1 = x1+x2
26     c2 = y1+y2
27 endfunction
28
29 //division
30 function [q,x]=division(x1,x2,y1,y2)
31     q=x1/x2
32     x=y1-y2
```



```

33     endfunction
34 //Variable Declaration
35 zc = 4           //real value of Zc
36 zco = -90       //angle in degree
37 z1 = 6         //real value of Z1
38 zlo = 90       //angle in degree
39 ia = 10        //real value of I
40 iao = 60       //angle in degree
41
42 //Calculation
43 [aa,bb]=voldivider(zc,z1,zco,zlo) //function of
    multiplication
44 [a,b]=polar2rect(zc,zco)         //function of
    conversion
45 [c,d]=polar2rect(z1,zlo)         //function of
    conversion
46 [e,f]=addition(a,c,b,d)         //function of addition
47 [ee,ff]=rect2polar(e,f)         //function of
    conversion
48 [g,h]=division(aa,ee,bb,ff)     //function of
    division
49 [i,j]=voldivider(ia,g,iao,h)    //function of
    multiplication
50
51 //Results
52 printf("Z = %d V < %d degree \n",g,h)
53 printf("E = %d V < %d degree \n",i,j)

```

Scilab code Exa 17.3 Example 3

```

1 //Chapter 17, Example 17.3
2 clc
3
4 //division
5 function[q,x]=division(x1,x2,y1,y2)

```

```

6         q=x1/x2
7         x=y1-y2
8         endfunction
9 //Variable Declaration
10 ea = 20 //source voltage in volts
11 eo = 0 //angle in degree
12 z1 = 5*10^3 //real value of Z
13 z1o = 0 //angle in degree
14
15
16 //Calculation
17 [a,b]=division(ea,z1,eo,z1o) //function of
    division
18
19
20
21 //Results
22 printf("I =( %d X 10^-3 V)A < %d degree",a*10^3,b)

```

Scilab code Exa 17.4 Example 4

```

1 //Chapter 17, Example 17.4
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8 //Variable Declaration
9 ia= 100 //real value of I
10 iao = 0 //angle in degree
11 z1 = 40*10^3 //real value of Z
12 z1o = 0 //angle in degree
13
14

```

```

15 // Calculation
16 [a,b]=voltdivider(ia,z1,iao,z1o)
17
18
19 // Results
20 printf("E = (%d x 10^6I)V < %d degree",a/10^6,b)

```

Scilab code Exa 17.5 Example 5

```

1 //Chapter 17, Example 17.5
2 clc
3 funcprot(0)
4 //addition
5 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
6     c1 = x1+x2+x3
7     c2 = y1+y2+y3
8     endfunction
9 //subtaction
10 function [c1,c2]=subtraction(x1,x2,y1,y2)
11     c1 = x1-x2
12     c2 = y1-y2
13     endfunction
14 //multiplication
15 function [r,s] = voltdivider(x1,x2,y1,y2)
16     r=x1*x2
17     s=y1+y2
18 endfunction
19 //Rectangle to Polar Conversion
20 function [r,th] = rect2polar(x,y)
21     r=sqrt((x**2)+(y**2))
22     if x==0 then
23         th=-90
24     else
25         th=atand(y/x)
26         if (x<0) & (y<0) then

```

```

27         th=th-180
28     end
29 end
30 endfunction
31 //Rectangle to Polar Conversion
32 function [r,th] = rect2polar1(x,y)
33     r=sqrt((x**2)+(y**2))
34     if x==0 then
35         th=90
36     else
37         th=atand(y/x)
38         if (x<0) & (y<0) then
39             th=th-180
40         end
41     end
42 endfunction
43 //division
44 function [q,x]=division(x1,x2,y1,y2)
45     q=x1/x2
46     x=y1-y2
47 endfunction
48
49 //Polar to Rectangle conversion
50 function [r,i]= polar2rect(x,y)
51     r=x*cosd(y)
52     i=x*sind(y)
53 endfunction
54 //Variable Declaration
55 z1 = complex(0,2) //complex form of Z1
56 z2 = complex(4,0) //complex form of Z2
57 z3 = complex(0,-1) //complex form of Z3
58 e1 = 2 //source voltage in volts
59 e1o = 0 //angle in degree
60 e2 = 6 //source voltage in volts
61 e2o = 0 //angle in degree
62
63 //Calculation
64 a= ((e1-e2)*real(z2)) //real part of numerator

```

```

65 b=e1*imag(z3) //imaginary part of
    numerator
66 [c,d]=rect2polar(a,b) //function of
    conversion
67
68 [e,f]=rect2polar1(real(z1),imag(z1)) //z1
    function of conversion
69 [g,h]=rect2polar(real(z2),imag(z2)) //z2 function
    of conversion
70 [i,j]=rect2polar(real(z3),imag(z3)) //z3 function
    of conversion
71 [ca,da]=voltdivider(e,g,f,h) //z1*z2
    function of multiplication
72 [cb,db]=voltdivider(e,i,f,j) //z1*z3
    function of multiplication
73 [cc,dd]=voltdivider(g,i,h,j) //z2*z3
    function of multiplication
74 [s,t]=polar2rect(ca,da) //function of
    conversion
75 [sa,ta]=polar2rect(cb,db) //function of
    conversion
76 [sb,tb]=polar2rect(cc,dd) //function of
    conversion
77
78 [o,p]=addition(s,sa,sb,t,ta,tb) //function of
    addition
79 [m,n]=rect2polar(o,p) //function of
    conversion
80 [mm,nn]=division(c,m,d,n) //function of
    division
81 //Results
82 printf("I1 = %.2f A < %.2f degree \n",mm,nn)

```

Scilab code Exa 17.9 Example 9

```

1 //Chapter 17, Example 17.9
2 clc
3
4
5 //Polar to Rectangle conversion
6 function [r,i]= polar2rect(x,y)
7     r=x*cosd(y)
8     i=x*sind(y)
9 endfunction
10
11 //addition
12 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
13     c1 = x1+x2+x3
14     c2 = y1+y2+y3
15 endfunction
16 //Rectangle to Polar Conversion
17 function [r,th] = rect2polar(x,y)
18     r=sqrt((x**2)+(y**2))
19     if x==0 then
20         th=90
21     else
22         th=atand(y/x)
23     end
24 endfunction
25
26 function [c1,c2]=subtraction(x1,x2,y1,y2)
27     c1 = x1-x2
28     c2 = y1-y2
29 endfunction
30 //multiplication
31 function [r,s] = voltdivider(x1,x2,y1,y2)
32     r=x1*x2
33     s=y1+y2
34 endfunction
35
36 //division
37 function [q,x]=division(x1,x2,y1,y2)
38     q=x1/x2

```

```

39         x=y1-y2
40     endfunction
41 //Variable Declaration
42 z1 = complex(1,2)    //complex form of Z1
43 z2 = complex(4,-8)  //complex form of Z2
44 z3 = complex(0,6)   //complex form of Z3
45 e1 = 8              //source of voltage in volts
46 e1o = 20           //angle in degree
47 e2 = 10            //source of voltage in volts
48 e2o = 0            //angle in degree
49
50 //Calculation
51 [a,b]=rect2polar(real(z2),imag(z2))    //z2
    function of conversion
52 [aa,bb]=voltdivider(a,e1,b,e1o)        //function of
    multiplication
53 [ac,bc]=polar2rect(aa,bb)             //function of
    conversion
54 [c,d]=rect2polar(real(z1),imag(z1))    //z1
    function of conversion
55 [ca,da]=voltdivider(c,e2,d,e2o)        //function of
    multiplication
56 [cb,db]=polar2rect(ca,da)             //function of
    conversion
57 [cc,dd]=subtraction(ac,cb,bc,db)       //function of
    subtraction
58 [e,f]=rect2polar(cc,dd)               //function of conversion
59 [ea,fa]=voltdivider(c,a,d,b)          //Z1*z2
    function of multiplication
60 [ee,ff]=polar2rect(ea,fa)             //function of
    conversion
61 [g,h]=rect2polar(real(z3),imag(z3))    //z3
    function of conversion
62 [ga,ha]=voltdivider(c,g,d,h)          //z1*z3
    function of multiplication
63 [gg,hh]=polar2rect(ga,ha)             //function of
    conversion
64 [gb,hb]=voltdivider(a,g,b,h)          //z2*z3 function

```

```

        of multiplication
65 [m,n]=polar2rect(gb,hb)    //function of conversion
66 [mm,nn]=addition(ee,gg,m,ff,hh,n)    //function of
    addition
67 [ma,na]=rect2polar(mm,nn)    //function of
    conversion
68 [mb,nb]=division(e,ma,f,na)    //function of
    division
69
70
71 //Results
72 printf("I2 = %.2 f A < %.2 f degree",mb,nb)

```

Scilab code Exa 17.12 Example 12

```

1 //Chapter 17, Example 17.12
2 clc
3 funcprot(0)
4 //Polar to Rectangle conversion
5 function [r,i]= polar2rect(x,y)
6     r=x*cosd(y)
7     i=x*sind(y)
8 endfunction
9
10 //addition
11 function [c1,c2]=addition(x1,x2,y1,y2)
12     c1 = x1+x2
13     c2 = y1+y2
14 endfunction
15 //Rectangle to Polar Conversion
16 function [r,th] = rect2polar(x,y)
17     r=sqrt((x**2)+(y**2))
18     if x==0 then
19         th=90
20     else

```



```

21         th=atand(y/x)
22         if (x<0) & (y<0) then
23             th=th-180
24         end
25     end
26 endfunction
27
28 function [c1,c2]=subtraction(x1,x2,y1,y2)
29     c1 = x1-x2
30     c2 = y1-y2
31     endfunction
32 //multiplication
33 function [r,s] = voltdivider(x1,x2,y1,y2)
34     r=x1*x2
35     s=y1+y2
36 endfunction
37
38 //division
39 function [q,x]=division(x1,x2,y1,y2)
40     q=x1/x2
41     x=y1-y2
42     endfunction
43 //Variable Declaration
44 z1 = 24 //real part in
    milliampere
45 z1o = 0 //angle in degree
46 z2 = -0.539 //real part in
    milliampere
47 z2o = 21.80 //angle in degree
48 z3 = 0.5 //real part in
    milliampere
49 z3o = 0 //angle in degree
50 z4 = 4 //real part in milliampere
51 z4o = 0 //angle in degree
52 z5 = 2.5 //real part in
    milliampere
53 z5o= -2.29 //angle in degree
54

```

```

55 // Calculation
56 [a,b]=voltdivider(z1,z2,z1o,z2o) //z1*z2 function
    of multiplication
57 [aa,bb]=voltdivider(z3,z4,z3o,z4o) //z3*z4 function
    of multiplication
58 [ac,bc]=polar2rect(a,b) //function of
    conversion
59 [ad,bd]=polar2rect(aa,bb) //function
    of conversion
60 [c,d]=addition(ac,ad,bc,bd) //function of
    addition
61 [ca,da]=rect2polar(c,d) //function of
    conversion
62 [m,n]=voltdivider(z5,z2,z5o,z2o) //function of
    multiplication
63 [ma,na]=voltdivider(z3,z3,z3o,z3o) //function of
    multiplication
64 [mb,nb]=polar2rect(m,n) //function of
    conversion
65 [mc,nc]=polar2rect(ma,na) //function
    of conversion
66 [md,nd]=addition(mb,mc,nb,nc) //function of
    addition
67 [me,ne]=rect2polar(md,nd) //function
    of conversion
68 [o,p]=division(ca,me,da,ne) //function of
    division
69 // Results
70 printf("V1 = %.2 f V < %.2 f degree ",o,p)

```

Scilab code Exa 17.16 Example 16

```

1 //Chapter 17, Example 17.16
2 clc
3 //Rectangle to Polar Conversion

```

```

4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=-90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12 //Rectangle to Polar Conversion
13 function [r,th] = rect2polar1(x,y)
14     r=sqrt((x**2)+(y**2))
15     if x==0 then
16         th=90
17     else
18         th=atand(y/x)
19     end
20 endfunction
21 //multiplication
22 function [r,s] = voltdivider(x1,x2,y1,y2)
23     r=x1*x2
24     s=y1+y2
25 endfunction
26
27 //division
28 function [q,x]=division(x1,x2,y1,y2)
29     q=x1/x2
30     x=y1-y2
31 endfunction
32 //Polar to Rectangle conversion
33 function [r,i]= polar2rect(x,y)
34     r=x*cosd(y)
35     i=x*sind(y)
36 endfunction
37 //addition
38 function [c1,c2]=addition1(x1,x2,y1,y2)
39     c1 = x1+x2
40     c2 = y1+y2
41 endfunction

```

```

42 // addition
43 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
44     c1 = x1+x2+x3
45     c2 = y1+y2+y3
46     endfunction
47 //Variable Declaration
48 z1 = complex(1/4,0)
49 z2 = complex(0,1/5)
50 z3 = complex(0,1/-2)
51 i1 = 6
52 i1o = 0
53 i2 = 4
54 i2o = 0
55
56 // Calculation
57 [a,b]=addition1(real(z3),real(z2),imag(z3),imag(z2))
                                     //y3+y2
58 [aa,bb]=rect2polar(a,b)
59 [ac,bc]=voltdivider(-aa,i1,-bb,i1o)           //(y3+y2)/
        i1
60 [c,d]=rect2polar(real(z2),imag(z2))
61 [cc,dd]=voltdivider(i2,c,i2o,d)           //i2y2
62 [w,v]=polar2rect(ac,bc)
63 [ww,vv]=polar2rect(cc,dd)
64 [wa,va]=addition1(w,ww,v,vv)
65 [wb,vb]=rect2polar(wa,va)
66 [ca,da]=rect2polar(real(z1),imag(z1))
67 [cb,db]=rect2polar1(real(z3),imag(z3))
68 [e,f]=voltdivider(ca,cb,da,db)
69 [ee,ff]=polar2rect(e,f)
70 [g,h]=voltdivider(c,cb,d,db)
71 [gg,hh]=polar2rect(g,h)
72 [m,n]=voltdivider(ca,c,da,d)
73 [mm,nn]=polar2rect(m,n)
74 [o,p]=addition(ee,gg,mm,ff,hh,nn)
75 [oo,pp]=rect2polar(o,p)
76 [s,t]=division(wb,oo,vb,pp)
77 // Results

```

```
78 printf("V1 = %.2f V < %.2f degree ",s,t)
```

Scilab code Exa 17.17 Example 17

```
1 //Chapter 17, Example 17.17
2 clc
3 //Variable Declaration
4 z1= complex(7,8)
5 z2 = complex(4,5)
6 z3 = complex(0,-10)
7 z4 = complex(8,0)
8 e1 = 20
9 e1o = 0
10 i1 = 10
11 i1o = 20
12
13 //Calculation
14
15
16
17 //Results
```

Scilab code Exa 17.19 Example 19

```
1 //Chapter 17, Example 17.19
2 clc
3 //Variable Declaration
4
5
6 //Calculation
7
8
9
```

10 //Results

Scilab code Exa 17.20 Example 20

```
1 //Chapter 17, Example 17.20
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=-90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12
13 //multiplication
14 function [r,s] = voltdivider(x1,x2,y1,y2)
15     r=x1*x2
16     s=y1+y2
17 endfunction
18
19 //division
20 function [q,x]=division(x1,x2,y1,y2)
21     q=x1/x2
22     x=y1-y2
23 endfunction
24 //Polar to Rectangle conversion
25 function [r,i]= polar2rect(x,y)
26     r=x*cosd(y)
27     i=x*sind(y)
28 endfunction
29 //addition
30 function [c1,c2]=addition1(x1,x2,y1,y2)
31     c1 = x1+x2
```

```

32     c2 = y1+y2
33     endfunction
34 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
35     c1 = x1+x2+x3
36     c2 = y1+y2+y3
37     endfunction
38 //Variable Declaration
39 za=complex(0,-4)           //complex form of Za
40 zb=complex(0,-4)           //complex form of Zb
41 zc=complex(3,4)           //complex form of Zc
42 z4 = complex(2,0)         //complex form of Z4
43 z5 = complex(3,0)         //complex form of Z5
44
45 //Calculation
46 [a,b]=rect2polar(real(zb),imag(zb)) //function of
    conversion
47 [c,d]=rect2polar(real(zc),imag(zc)) //function of
    conversion
48 [e,f]=voltdivider(a,c,b,d) //function of
    multiplication
49 [g,h]=addition(real(za),real(zb),real(zc),imag(za),
    imag(zb),imag(zc)) //function of addition
50 [i,j]=rect2polar(g,h) //function of conversion
51 [m,n]=division(e,i,f,j) //function of
    division
52 [o,p]=polar2rect(m,n) //Z1 function of
    conversion
53 [aa,bb]=rect2polar(real(za),imag(za)) //function
    of conversion
54 [ee,ff]=voltdivider(aa,c,bb,d) //function of
    multiplication
55 [mm,nn]=division(ee,i,ff,j) //function of
    division
56 [oo,pp]=polar2rect(mm,nn) //Z2 function of
    conversion
57 [ac,bc]=voltdivider(aa,a,bb,b) //function of
    multiplication
58 [ad,bd]=division(ac,i,bc,j) //function of

```

```

    division
59 [ae,be]=polar2rect(ad,bd)      //Z3 function of
    conversion
60 [ma,na]=addition1(o,real(z4),p,imag(z4))    //
    function of addition
61 [mb,nb]=rect2polar(ma,na)      //ZT1
    function of conversion
62 [s,t]=addition1(oo,real(z5),pp,imag(z5))    //
    function of addition
63 [sa,ta]=rect2polar(s,t)      //ZT2
    function of conversion
64 [sb,tb]=volddivider(mb,sa,nb,ta)    //function
    of multiplication
65 [sc,tc]=addition1(ma,s,na,t)    //function of
    addition
66 [se,te]=rect2polar(sc,tc)      //function
    of conversion
67 [sd,td]=division(sb,se,tb,te)    //function of
    division
68 [sf,tf]=polar2rect(sd,td)      //ZT3 function
    of conversion
69 [si,ti]=addition1(ae,sf,be,tf)    //function of
    addition
70 [sv,tv]=rect2polar(si,ti)      //function
    of conversion
71
72
73 // Results
74 printf("ZT = %.2 f ohm < %.2 f degree \n",sv,tv)

```

Scilab code Exa 17.21 Example 21

```

1 //Chapter 17, Example 17.21
2 clc
3 //Variable Declaration

```



```
4 zt = complex(1,2)
5 zd = complex(3,6)
6
7 // Calculation
8 a= real(zd)/3
9 b= imag(zd)/3
10 c = a*3
11 d = b*3
12 e = c/2
13 f = d/2
14 g = (e*2)/3
15 h = (f*2)/3
16 // Results
17 printf("Zy = %d ohm + j %d ohm \n",a,b)
18 printf("Zt = %d ohm + j %d ohm \n",real(zt),imag(zt)
    )
19 printf("Zdel = %d ohm + j %d ohm \n",c,d)
20 printf("Z1 = %.1f ohm + j %d ohm \n",e,f)
21 printf("Zt = %d ohm + j %d ohm \n",g,h)
```

Chapter 18

Network Theorems ac

Scilab code Exa 18.1 Example 1

```
1 //Chapter 18, Example 18.1
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=-90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12 //multiplication
13 function [r,s] = voltdivider(x1,x2,y1,y2)
14     r=x1*x2
15     s=y1+y2
16 endfunction
17
18 //division
19 function [q,x]=division(x1,x2,y1,y2)
20     q=x1/x2
21     x=y1-y2
```

```

22     endfunction
23 //addition
24 function [c1,c2]=addition(x1,x2,y1,y2)
25     c1 = x1+x2
26     c2 = y1+y2
27     endfunction
28
29 //Polar to Rectangle conversion
30 function [r,i]= polar2rect(x,y)
31     r=x*cosd(y)
32     i=x*sind(y)
33 endfunction
34 //Variable Declaration
35 z1 = complex(0,4)           //complex form of Z1
36 z2 = complex(0,4)           //complex form of Z2
37 z3 = complex(0,-3)         //complex form of Z3
38 e1 = 10                     //real value of voltage
39 e1o = 0                     //angle in degree
40 e2 = 5                       //real value of voltage
41 e2o = 0                     //angle in degree
42 //Calculation
43 [a,b]=rect2polar(real(z2),imag(z2)) //function of
    conversion
44 [c,d]=rect2polar(real(z3),imag(z3)) //function of
    conversion
45 [e,f]=voltdivider(a,c,b,d) //function of
    multiplication
46 [g,h]=addition(real(z2),real(z3),imag(z2),imag(z3))
    //function of addition
47 [gg,hh]=rect2polar(g,h) //function of conversion
48 [m,n]=division(e,gg,f,hh) //z23 in ohms
49 [mo,no]=polar2rect(m,n)
50 [o,p]=addition(mo,real(z1),no,imag(z1)) //function
    of addition
51 [oo,pp]=rect2polar(o,p) //function of conversion
52 [s,t]=division(e1,oo,e1o,pp) //Is1 in ampere
53 [ss,tt]=polar2rect(s,t) //function of conversion
54 [aa,ba]=voltdivider(c,s,d,t) //function of

```

```

        multiplication
55 [ab,bb]=rect2polar(aa,ba) //function of conversion
56 [ac,bc]=division(ab,gg,bb,hh) //I in ampere
57 z12 = imag(z1)/2 //imaginary part of Z12
58 [cc,dd]= addition(real(z1),real(z3),z12,imag(z3))
59 [ca,da]=rect2polar(cc,dd) //function of conversion
60 [cb,db]=division(e2,ca,e2o,da) //Is2
61 i2 = cb/2 //in ampere
62 [ma,na]=polar2rect(cb,db) //function of conversion
63 [mm,nn]=addition(ss,ma,tt,na) //function of
        addition
64 [mb,nb]=rect2polar(mm,nn) //final I function
        of conversion
65
66
67 //Results
68 printf("I = %.2f A < %.2f degree \n",mb,nb)

```

Scilab code Exa 18.2 Example 2

```

1 //Chapter 18, Example 18.2
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12 //multiplication
13 function [r,s] = voltdivider(x1,x2,y1,y2)
14     r=x1*x2
15     s=y1+y2

```

```

16 endfunction
17 //addition
18 function [c1,c2]=addition(x1,x2,y1,y2)
19     c1 = x1+x2
20     c2 = y1+y2
21 endfunction
22 //division
23 function [q,x]=division(x1,x2,y1,y2)
24     q=x1/x2
25     x=y1-y2
26     endfunction
27
28 //Polar to Rectangle conversion
29 function [r,i]= polar2rect(x,y)
30     r=x*cosd(y)
31     i=x*sind(y)
32 endfunction
33 //Variable Declaration
34 z1 = complex(0,6)           //complex form of Z1
35 z2 = complex(6,-8)        //complex form of Z2
36 i1 = 2                    //current in ampere
37 e1= 20                    //real value of voltage
38 e1o = 30                  //angle in degree
39 //Calculation
40 [a,b]=addition(real(z1),real(z2),imag(z1),imag(z2))
    //function of addition
41 [aa,bb]=rect2polar(a,b)   //function of conversion
42 c=imag(z1)*i1            //imaginary part
43 [i,j]=rect2polar(real(z1),c) //function of
    conversion
44 [e,f]=division(i,aa,j,bb) //I1 function of
    division
45 [m,n]=division(e1,aa,e1o,bb) //I2 function of
    division
46 [o,p]=polar2rect(e,f)   //function of conversion
47 [oo,pp]=polar2rect(m,n) //function of conversion
48 [s,t]=addition(o,oo,p,pp) //function of addition
49 [ss,tt]=rect2polar(s,t) //function of conversion

```

```

50 //Results
51 printf("I = %.2f A < %.2f degree \n",ss,tt)

```

Scilab code Exa 18.3 Example 3

```

1 //Chapter 18, Example 18.3
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     th=atand(y/x)
7 endfunction
8
9
10 //Polar to Rectangle conversion
11 function [r,i]= polar2rect(x,y)
12     r=x*cosd(y)
13     i=x*sind(y)
14 endfunction
15
16 function [c1,c2]=addition(x1,x2,y1,y2)
17     c1 = x1+x2
18     c2 = y1+y2
19 endfunction
20 //Variable Declaration
21 i1= 1.9 //real value of current in
    ampere
22 i1o = 108.43 //angle in degree
23 r1 = 6 //resistance in ohms
24 i2 = 3.16 //real value of current in
    ampere
25 i2o = 48.43 //angle in degree
26 it = i1+i2 //real value of current in
    ampere
27 ito = i1o-i2o //angle in degree

```

```

28 ia = 4.42                //total current in ampere
29 iao = 70.2              //angle in degree
30 //Calculation
31 v1 = i1*r1              //in volts
32 v2 = i2*r1              //in volts
33 [a,b]=polar2rect(v1,i1o) //function of conversion
34 [c,d]=polar2rect(v2,i2o) //function of conversion
35 [e,f]=addition(a,c,b,d) //function of addition
36 [i,j]=rect2polar(e,f)   //function of conversion
37 vt = ia*r1              //in volts
38
39 //Results
40 printf("V6ohm = %.1f V < %.1f degree \n",i,j)
41 printf("V6ohm = %.1f V < %.1f degree (checks)\n",vt,
    iao)

```

Scilab code Exa 18.4 Example 4

```

1 //Chapter 18, Example 18.4
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8
9 //division
10 function [q,x]=division(x1,x2,y1,y2)
11     q=x1/x2
12     x=y1-y2
13 endfunction
14 //Rectangle to Polar Conversion
15 function [r,th] = rect2polar(x,y)
16     r=sqrt((x**2)+(y**2))
17     th=atand(y/x)

```

```

18 endfunction
19
20
21 //Polar to Rectangle conversion
22 function [r,i]= polar2rect(x,y)
23     r=x*cosd(y)
24     i=x*sind(y)
25 endfunction
26
27 function [c1,c2]=addition(x1,x2,y1,y2)
28     c1 = x1+x2
29     c2 = y1+y2
30 endfunction
31 //Variable Declaration
32 r1 = 0.5*10^3           //resistance1 in kilohms
33 r1o = 0                 //angle in degree
34 r2 = 1*10^3           //resistance2 in kilohms
35 r2o = 0                 //angle in degree
36 r3 = 3*10^3           //resistance3 in kilohms
37 r3o = 0                 //angle in degree
38 e1 = 12                 //source voltage in volts
39 xc = 10*10^3           //reactance in kilohms
40 xco = -90               //angle in degree
41 x1 = 2*10^3           //reactance in kilohms
42 x1o = 90                //angle in degree
43 e2 = 4                  //source voltage in volts
44 e2o = 0                 //angle in degree
45 //Calculation
46 rp1 = (r1*r3)/(r1+r3)   //in kilohms
47 v3 = (rp1*e1)/(rp1+r2) //V3(dc) in
    volts
48 [a,b]=voltagedivider(r2,xc,r2o,xco) //function of
    multiplication
49 [c,d]=polar2rect(r2,r2o) //function of conversion
50 [e,f]=polar2rect(xc,xco) //function of conversion
51 [g,h]=addition(c,e,d,f) //function of addition
52 [i,j]=rect2polar(g,h)   //function of conversion
53 [m,n]=division(a,i,b,j) //z2 function

```



```

    of division
54 [o,p]=polar2rect(r3,r3o) //function of conversion
55 [q,r]=polar2rect(x1,x1o) //function of conversion
56 [s,t]=addition(o,q,p,r) //function of addition
57 [v,w]=rect2polar(s,t) //z3 function
    of conversion
58 [ma,na]=voltdivider(m,v,n,w) //function of
    multiplication
59 [mb,nb]=polar2rect(m,n) //function of conversion
60 [mc,nc]=addition(mb,s,nb,t) //function of
    addition
61 [md,nd]=rect2polar(mc,nc) //Z2 + Z31
    function of conversion
62 [oo,pp]=division(ma,md,na,nd) //function of
    division
63 [qa,ra]=polar2rect(r1,r1o) //function of
    conversion
64 [qb,rb]=polar2rect(oo,pp) //function of
    conversion
65 [qc,rc]=addition(qa,qb,ra,rb) //function of
    addition
66 [qd,rd]=rect2polar(qc,rc) //Zt function
    of conversion
67 [va,wa]=division(e2,qd,e2o,rd) //Is function
    of division
68 [sa,ta]=voltdivider(m,va,n,wa) //Z2*Is
    function of multiplication
69 [sb,tb]=division(sa,md,ta,nd) //I3 function
    of division
70 [oa,pa]=voltdivider(sb,r3,tb,r3o) //V3(ac)
    function of multiplication
71 pb=oa*sqrt(2) //power in watt
72 //Results
73 printf("V3 = %.1f V \n",v3)
74 printf("V3 = %.1f + %.2f sin(wt %.2f degree)\n",v3,
    pb,pa)

```

Scilab code Exa 18.5 Example 5

```
1 //Chapter 18, Example 18.5
2 clc
3 funcprot(0)
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)
6     r=sqrt((x**2)+(y**2))
7     th=atand(y/x)
8 endfunction
9 function [c1,c2]=addition(x1,x2,y1,y2)
10     c1 = x1+x2
11     c2 = y1+y2
12 endfunction
13 //Polar to Rectangle conversion
14 function [r,i]= polar2rect(x,y)
15     r=x*cosd(y)
16     i=x*sind(y)
17 endfunction
18
19 //multiplication
20 function [r,s] = voltdivider(x1,x2,y1,y2)
21     r=x1*x2
22     s=y1+y2
23 endfunction
24 //Variable Declaration
25 z1 = 4           //real value of Z1
26 z1o = 0         //angle in degree
27 z2 = complex(6,8) //complex form of Z2
28 u=20           //unit
29 v=10           //real value of voltage
30 vo = 0         //angle in degree
31 h=100          //unit
32 iu = 20*10^-3 //current in milliampere
```

```

33 // Calculation
34 [a,b]=addition(z1,real(z2),z1o,imag(z2)) //function
    of addition
35 [c,d]=rect2polar(a,b) //function of conversion
36 e=1/c //I1
37 f=-d
38 [g,ha]=voltdivider(z1,e,z1o,f) //I2 function of
    multiplication
39 i=e*u*v
40 [m,n]=polar2rect(i,f) //I1 function of
    conversion
41 j=g*h*iu
42 [ma,na]=polar2rect(j,ha) //function of conversion)
43 [q,r]=addition(m,ma,n,na) //function of addition
44 [qa,ra]=rect2polar(q,r) //function of conversion
45 //Results
46 printf(" I2 = %.2 f A < %.2 f degree \n",qa,ra)

```

Scilab code Exa 18.7 Example 7

```

1 //Chapter 18, Example 18.7
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8
9 //division
10 function [q,x]=division(x1,x2,y1,y2)
11     q=x1/x2
12     x=y1-y2
13 endfunction
14
15 //division

```

```

16 function [q,x]=division1(x1,x2,y1,y2)
17     q=x1/x2
18     x=y1+y2
19     endfunction
20 //Rectangle to Polar Conversion
21 function [r,th] = rect2polar(x,y)
22     r=sqrt((x**2)+(y**2))
23     if x==0 then
24         th=-90
25     else
26         th=atand(y/x)
27     end
28 endfunction
29
30 //Polar to Rectangle conversion
31 function [r,i]= polar2rect(x,y)
32     r=x*cosd(y)
33     i=x*sind(y)
34 endfunction
35
36 function [c1,c2]=addition(x1,x2,y1,y2)
37     c1 = x1+x2
38     c2 = y1+y2
39     endfunction
40 //Variable Declaration
41 z1=complex(0,8)           //complex form of Z1
42 z2=complex(0,-2)        //complex form of Z2
43 ea= 10                   //source voltage in volts
44 eo = 0                   //angle in degree
45 //Calculation
46 [a,b]=rect2polar(real(z1),imag(z1)) //function of
    conversion
47 [c,d]=rect2polar(real(z2),imag(z2)) //function of
    conversion
48 [e,f]=voldivider(a,c,b,d) //function of
    multiplication
49 [g,h]=addition(real(z1),real(z2),imag(z1),imag(z2))
50 [gg,hh]=rect2polar(g,h) //function of conversion

```

```

51 [o,p]=division(e,gg,f,hh)    //function of division
52 [s,t]=voldivider(c,ea,d,eo) //function of
    multiplication
53 [ss,tt]=rect2polar(g,h)    //function of conversion
54 [oo,pp]=division1(s,ss,t,tt) //function of
    conversion
55
56
57 //Results
58 printf("Zth = %.2f ohm < %d degree \n",o,p)
59 printf("Eth = %.2f V < %d degree \n",oo,pp)

```

Scilab code Exa 18.8 Example 8

```

1 //Chapter 18, Example 18.8
2 clc
3 //multiplication
4 function [r,s] = voldivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8
9 //division
10 function [q,x]=division(x1,x2,y1,y2)
11     q=x1/x2
12     x=y1-y2
13 endfunction
14 //Rectangle to Polar Conversion
15 function [r,th] = rect2polar(x,y)
16     r=sqrt((x**2)+(y**2))
17     th=atand(y/x)
18 endfunction
19
20
21 //Polar to Rectangle conversion

```

```

22 function [r,i]= polar2rect(x,y)
23     r=x*cosd(y)
24     i=x*sind(y)
25 endfunction
26
27 function [c1,c2]=addition(x1,x2,y1,y2)
28     c1 = x1+x2
29     c2 = y1+y2
30     endfunction
31 //Variable Declaration
32 z1=complex(6,8)           //complex form of Z1
33 z2=complex(3,-4)        //complex form of Z2
34 z3=complex(0,5)         //complex form of Z3
35 ea= 10                   //real value of voltage
36 eo = 0                   //angle in degree
37 //Calculation
38 [aa,bb]=rect2polar(real(z1),imag(z1)) //function of
    conversion
39 [ac,bc]=rect2polar(real(z2),imag(z2)) //function of
    conversion
40 [a,b]=voltagedivider(aa,ac,bb,bc) //function of
    multiplication
41 [g,h]=addition(real(z1),real(z2),imag(z1),imag(z2))
    //function of addition
42 [i,j]=rect2polar(g,h) //function of conversion
43 [ii,jj]=division(a,i,b,j) //function of division
44 [ia,ja]=polar2rect(ii,jj) //function of conversion
45 [gg,hh]=addition(real(z3),ia,imag(z3),ja) //
    function of addition
46 [ga,ha]=rect2polar(gg,hh) //function of conversion
47 [c,d]=voltagedivider(ac,ea,bc,eo) //function of
    multiplication
48 [ca,da]=division(c,i,d,j) //function of division
49
50 //Results
51 printf("Zth = %.2f ohm < %.2f degree \n",ga,ha)
52 printf("Eth = %.2f V < %.2f degree \n",ca,da)

```

Scilab code Exa 18.9 Example 9

```
1 //Chapter 18, Example 18.8
2 clc
3
4 //Variable Declaration
5 zth = 2*10^3           //resistance in kilohms
6 rs = 0.5*10^3         //resistance1 in kilohms
7 rs1 = 2.3*10^3        //resistance2 in
   kilohms
8 r1 = 1*10^3           //resistance3 in kilohms
9
10 //Calculation
11 rsa = rs+rs1          //total resistance in
   kilohms
12 i1 = 1/rsa           //wrong answer in textbook
13 eth = -i1*zth        //in ohmEi
14 vl = (r1*eth)/((1*10^3)+(2*10^3)) //in Ei
15
16
17 //Results
18 printf("Zth = %d kohm \n",zth/10^3)
19 printf("Eth = %.2 fEi \n",eth)
20 printf("VL = %.2 fEi \n",vl)
```

Scilab code Exa 18.14 Example 14

```
1 //Chapter 18, Example 18.14
2 clc;
3
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)
```

```

6     r=sqrt((x**2)+(y**2))
7     if x==0 then
8         th=90
9     else
10        th=atand(y/x)
11    end
12 endfunction
13
14
15 //Polar to Rectangle conversion
16 function [r,i]= polar2rect(x,y)
17     r=x*cosd(y)
18     i=x*sind(y)
19 endfunction
20
21 function [c1,c2]=addition(x1,x2,y1,y2)
22     c1 = x1+x2
23     c2 = y1+y2
24     endfunction
25 //multiplication
26 function [r,s] = voltdivider(x1,x2,y1,y2)
27     r=x1*x2
28     s=y1-y2
29 endfunction
30
31 //division
32 function [q,x]=division(x1,x2,y1,y2)
33     q=x1/x2
34     x=y1-y2
35     endfunction
36 //Variable Declaration
37 z1=complex(3,4)    //complex form of Z1
38 z2=complex(0,-5) //complex form of Z2
39 e1 = 20           //real value of voltage
40 e1o = 0           //angle in degree
41
42 //Calculation
43 [a,b]=rect2polar(real(z1),imag(z1)) //function of

```



```

conversion
44 [c,d]=rect2polar(real(z2),imag(z2)) //function of
conversion
45 [ca,da]=voldivider(a,c,b,d) //function of
multiplication
46 [cb,db]=addition(real(z1),real(z2),imag(z1),imag(z2)
)
47 [cc,dd]=rect2polar(cb,db) //function of conversion
48 [e,f]=division(ca,cc,da,dd) //function of division
49 [o,p]=polar2rect(e,f) //function of conversion
50 [m,n]=division(e1,a,e1o,b) //function of division
51 //Resultss
52 printf("Zn = %.2f ohm %.2fj ohm \n",o,p)
53 printf("In = %d A < %.2f degree \n",m,n)

```

Scilab code Exa 18.15 Example 15

```

1 //Chapter 18, Example 18.15
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12
13
14 //Polar to Rectangle conversion
15 function [r,i]= polar2rect(x,y)
16     r=x*cosd(y)
17     i=x*sind(y)
18 endfunction

```

```

19
20 function [c1,c2]=addition(x1,x2,y1,y2)
21     c1 = x1+x2
22     c2 = y1+y2
23     endfunction
24 //multiplication
25 function [r,s] = voltdivider(x1,x2,y1,y2)
26     r=x1*x2
27     s=y1+y2
28 endfunction
29
30 //division
31 function [q,x]=division(x1,x2,y1,y2)
32     q=x1/x2
33     x=y1-y2
34     endfunction
35 //Variable Declaration
36 z1=complex(2,-4)           //complex form of z1
37 z2=complex(1,0)           //complex form of z2
38 z3=complex(0,5)           //complex form of z3
39 i = 3                       //current in ampere
40 //Calculation
41 [a,b]=addition(real(z1),real(z2),imag(z1),imag(z2))
42     //z1+z2 in comp
43 [aa,bb]=rect2polar(a,b)           //z1+z2 in
44     polar form
45 [ac,bc]=rect2polar(real(z3),imag(z3)) //z3 in
46     polar form
47 [ad,bd]=voltdivider(ac,aa,bc,bb) //z3(z1+z2
48     )
49 [ae,be]=addition(real(z3),a,imag(z3),b) //z3+(z1+
50     z2) in complex
51 [af,bf]=rect2polar(ae,be)           //function of
52     conversion
53 [ag,bg]=division(ad,af,bd,bf)       //function of
54     division
55 [ah,bh]=polar2rect(ag,bg)           //function of
56     conversion

```

```

49 za = real(z1)*i           //real part
50 zb = imag(z1)*i         //imaginary part
51 [c,d]=rect2polar(za,zb)  //function of
    conversion
52 [e,f]=division(c,aa,d,bb) //function of
    division
53 //Results
54 printf("Zn = %.2f ohm + j%.2f ohm \n",ah,bh)
55 printf("In = %.2f A < %.1f degree \n",e,f)

```

Scilab code Exa 18.16 Example 16

```

1 //Chapter 18, Example 18.16
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     th=atand(y/x)
7 endfunction
8 //multiplication
9 function [r,s] = voltdivider(x1,x2,y1,y2)
10     r=x1*x2
11     s=y1+y2
12 endfunction
13 //Variable Declaration
14 zn = complex(7.50,2.50) //complex form of Zn
15 in = 2.68 //real value of
    current
16 ino = -10.3 //angle in degree
17
18 //Calculation
19 [a,b]=rect2polar(real(zn),imag(zn)) //function of
    conversion
20 [c,d]=voltdivider(in,a,ino,b) //function of
    multiplication

```

```

21
22
23 //Results
24 printf("Eth = %.1f V < %.2f degree \n",c,d)

```

Scilab code Exa 18.19 Example 19

```

1 //Chapter 18, Example 18.19
2 clc
3 funcprot(0)
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)
6     r=sqrt((x**2)+(y**2))
7     if x==0 then
8         th=90
9     else
10         th=atand(y/x)
11     end
12 endfunction
13
14
15 //Polar to Rectangle conversion
16 function [r,i]= polar2rect(x,y)
17     r=x*cosd(y)
18     i=x*sind(y)
19 endfunction
20
21 //addition
22 function [c1,c2]=addition(x1,x2,y1,y2)
23     c1 = x1+x2
24     c2 = y1+y2
25     endfunction
26 //multiplication
27 function [r,s] = voltdivider(x1,x2,y1,y2)
28     r=x1*x2

```

```

29     s=y1+y2
30 endfunction
31
32 //division
33 function [q,x]=division(x1,x2,y1,y2)
34     q=x1/x2
35     x=y1-y2
36 endfunction
37 //Variable Declaration
38 z1 = complex(6,-8)
39 z2 = complex(0,8)
40 e1 = 9
41 e1o = 0
42 //Calculation
43 [a,b]=rect2polar(real(z1),imag(z1))
44 [c,d]=rect2polar(real(z2),imag(z2))
45 [e,f]=voltdivider(a,c,b,d)
46 [g,h]=addition(real(z1),real(z2),imag(z1),imag(z2))
47 [gg,hh]=rect2polar(g,h)
48 [o,p]=division(e,gg,f,hh)
49 [oo,pp]=polar2rect(o,p)
50 [aa,bb]=voltdivider(c,e1,d,e1o)
51 [cc,dd]=division(aa,gg,bb,hh)
52 pmax = (cc*cc)/(4*a)
53 //Results
54 printf("ZL = %.2f ohm + j%d ohm \n",oo,pp)
55 printf("Pmax = %.2f W ",pmax)

```

Scilab code Exa 18.20 Example 20

```

1 //Chapter 18, Example 18.20
2 clc
3 //Variable Declaration
4
5

```

```
6 // Calculation
7
8
9
10 // Results
```

Scilab code Exa 18.21 Example 21

```
1 //Chapter 18, Example 18.21
2 clc
3 //Variable Declaration
4 rth = 4 //reistance in ohms
5 xth = 7 //in ohms
6 Xload = 4 //load resistance in
   ohms
7 eth = 20 //source voltage in
   volts
8 //Calculation
9 r1 = sqrt((rth*rth)+ ((xth-Xload)*(xth-Xload))) //
   load resistance in ohms
10 rav = (rth+r1)/2 //resistance in ohms
11 p = (eth*eth)/(4*rav) //power in watt
12 pmax = (eth*eth)/(4*rth) //power in watt
13 //Results
14 printf("RL = %d ohm \n",r1)
15 printf("Rav = %.1 f ohm \n",rav)
16 printf("P = %.2 f W \n",p)
17 printf("Pmax = %d W \n",pmax)
```

Chapter 19

Power ac

Scilab code Exa 19.1 Example 1

```
1 //Chapter 19, Example 19.1, page 860
2 clc
3 //Initialisation
4 p1=100 //power in watt
5 p2=200 //power in watt
6 p3=300 //power in watt
7 var1=0 //volt-amperes
   reactive
8 var2=700 //volt-amperes
   reactive
9 var3=1500 //volt-amperes
   reactive
10
11 //Calculation
12 va1=sqrt((p1**2)+(var1**2))
   //volt-amperes
13 va2=sqrt((p2**2)+(var2**2))
   //volt-amperes
14 va3=sqrt((p3**2)+(var3**2))
   //volt-amperes
15 pt=p1+p2+p3 //total power in
```

```

    watt
16 qt=var3-var2 //total volt-
    amperes
17 st=sqrt((pt**2)+(qt**2)) //
    total volt-amperes reactive
18 fp=pt/st //power factor
19 i=st/p1 //current in ampere
20 teta=acosd(fp) //angle in
    degree
21
22 //Result
23 printf("VA of load 1 = %d \n",va1)
24 printf("VA of load 2 = %.1f \n",va2)
25 printf("VA of load 3 = %.2f \n",va3)
26 printf(" Total Watt, Pt = %d \n",pt)
27 printf(" Total volt-amperes reactive , Qt = %d \n",qt)
28 printf(" Total volt-amperes , St = %d \n",st)
29 printf(" Power factor , Fp = %.1f leading (C) \n",fp)
30 printf(" I = %d < + %.2f degree \n",i,teta)

```

Scilab code Exa 19.2 Example 2

```

1 //Chapter 19, Example 19.2, page 861
2 clc
3 funcprot()
4 function [r,th]=rect2pol(x,y)
5 //rectangle to polar coordinate conversion
6 r=sqrt(x^2+y^2);
7 th = atan(y,x)*180/%pi;
8 endfunction
9
10 //Initialisation
11 E=100 //emf in volt
12 r=6 //resistance in ohm
13 x1=7 //inductive reactance in ohm

```



```

14 xlang=90 //angle in degree
15 xc=15 //capacitive reactance in
    ohm
16 xcang=-90 //angle in degree
17 f1=60 //frequency in hertz
18
19
20 //Calculation
21 I=E/(r+complex(0,xl)+complex(0,-xc)) //
    current in ampere
22 [ir,iang]=rect2pol(real(I),imag(I))
    //conversion to polar form
23 vr=ir*r //in volt
24 vl=ir*xl //in volt
25 vlang=iang*xlang //
    in degree
26 vc=ir*xc //in volt
27 vcang=xcang*iang //
    in degree
28 pt=E*ir*cosd(iang)
    //power in watt
29 st=E*ir //VAR
30 qt=E*ir*sind(iang)
    //VA
31 fp=pt/st //power
    factor
32 wr=(vr*ir)/f1 //
    energy in joule
33 wl=(vl*ir)*(2*3.14*f1)**-1
    //energy in joule
34 wc=(vc*ir)/(2*3.14*f1)
    //energy in joule
35
36 //Result
37 printf(" (a) I = %d A < %.2f degree \n",ir,iang)
38 printf(" Pt = %d W \n",pt)
39 printf(" Qt = %d VAR \n",qt)
40 printf(" St = %d VA \n",st)

```

```

41 printf("      Fp = %.1f leading (C) \n",fp)
42 printf("(b) Wr = %d J \n",wr)
43 printf("(c) Wl = %.2f J \n",wl)
44 printf("(d) Wc = %.2f J \n",wc)

```

Scilab code Exa 19.3 Example 3

```

1 //Chapter 19, Example 19.3, page 861
2 clc
3 funcprot()
4 function [r,th]=rect2pol(x,y)
5 //rectangle to polar coordinate conversion
6   r=sqrt(x^2+y^2);
7   th = atan(y,x)*180/%pi;
8 endfunction
9
10 //Initialisation
11 n=12 //no of device
12 p1=60 //power in watt
13 q1=0 //in VAR
14 fp1=1 //power factor
15 p2=6400 //power in watt
16 q2=0 //in VAR
17 fp2=1 //power factor
18 n2=5 //no of device
19 p0=746 //power in watt
20 eta=0.82 //eta contat
21 fp3=0.72 //power factor
22 e=208 //emf in volt
23 r=9 //resistance in ohm
24 xc=12 //capacitive reactance
    in ohm
25 z=complex(9,-12)
26
27 //Calculation

```

```

28 tp1=n*p1 //power in watt
29 s1=tp1 //in VA
30 s2=p2 //in VA
31 p3=(p0*n2)/eta //power in watt
32 s3=p3/fp3 //in VA
33 q3=s3*sin(asin(fp3)) //in VAR
34 [zr,zang]=rect2pol(real(z),imag(z)) //function of
conversion
35 i=e/zr //in ampere
36 iang=0-zang
37 p4=(i**2)*r //power in watt
38 q4=(i**2)*xc //in VAR
39 s4=sqrt((p4**2)+(q4**2)) //in
VA
40 fp4=p4/s4 //power factor
41 pt=tp1+p2+p3+p4 //total power in
watt
42 qt=q1+q2+q3-q4 //in VAR(L)
43 st=sqrt((pt**2)+(qt**2)) //in VA
44 fpt=pt/st //power factor
45 tetat=acosd(fpt) //angle in degree
46 it=st/e //in ampere
47
48 //Result
49 printf("(a) Bulb \n")
50 printf(" P1 = %d W \n",tp1)
51 printf(" Q1 = %d VAR \n",q1)
52 printf(" S1 = %d VA\n",s1)
53 printf(" Fp1 = %d \n",fp1)
54 printf(" Heating elements \n")
55 printf(" P2 = %.1 f kW\n",p2/1000)
56 printf(" Q2 = %d VAR\n",q2)
57 printf(" S2 = %.1 f KVA\n",s2/1000)
58 printf(" Fp2 = %d \n",fp2)
59 printf(" Motor \n")
60 printf(" P3 = %.2 f W\n",p3)
61 printf(" Q3 = %.2 f VAR\n",q3)
62 printf(" S3 = %.2 f VA\n",s3)

```

```

63 printf("      Fp = %.2 f lagging \n",fp3)
64 printf("      Capacitive load \n")
65 printf("      P4 = %.2 f W\n",p4)
66 printf("      Q4 = %.2 f VAR\n",q4)
67 printf("      S4 = %.2 f VA\n",s4)
68 printf("      Fp = %.1 f lagging \n",fp4)
69 printf("(b) \n")
70 printf("      Pt = %.2 f W \n",pt)
71 printf("      Qt = %.2 f VAR \n",qt)
72 printf("      St = %.2 f VA\n",st)
73 printf("      Fp = %.3 f lagging \n",fpt)
74 printf("(c) \n")
75 printf("      I = %.2 f A < %.2 f degree \n",it,-tetat)

```

Scilab code Exa 19.4 Example 4

```

1 //Chapter 19, Example 19.4, page 864
2 clc
3 //Initialisation
4
5 //Polar to Rectangle conversion
6 function [r,i]= polar2rect(x,y)
7     r=x*cosd(y)
8     i=x*sind(y)
9 endfunction
10
11 ei=5000 //in VAR
12 v=100 //emf in volt
13 fp=0.6 //power factor
14
15 //Calculation
16 i=ei/v //current in ampere
17 tetat=-acosd(fp) //phase angle in degree
18 zt=v/i //impedance of the
    circuit in ohm

```

```

19 [a,b]=polar2rect(zt,teta)    //function of
    conversion
20 //Result
21 printf("I = %d A < %.2f degree \n",i,teta)
22 printf("Zt =%.1f ohm + j %.1f ohm",a,-b)

```

Scilab code Exa 19.5 Example 5

```

1 //Chapter 19, Example 19.5, page 865
2 clc
3 //Initialisation
4 hp1=746                //1 hp
5 eta=0.92              //eta
6 fp=0.6                //power factor
7 v=208                 //voltage
8 qc=5405.8             //in VAR
9 f=60                  //frequency in hertz
10
11
12 //Calculation
13 p0=5*hp1              //output power in watt
14 pi=p0/eta             //input power in watt
15 teta=acosd(fp)       //angle in degree
16 ql=pi*tand(teta)     //in VAR(L)
17 s=sqrt((pi**2)+(ql**2)) //in VA
18 xc=(v**2)/(qc)       //in ohms
19 c=1/(2*3.14*f*xc)    //in microfarad
20 i1=s/v                //in ampere
21 i2=pi/v               //in ampere
22 im=pi/(v*fp)         //in ampere
23
24
25 //Result
26 printf("(a) Power traingle , \n")
27 printf("    Pi = %.2f W \n",pi)

```

```

28 printf("      QL = %.2f VAR \n",q1)
29 printf("      S = %.2f VA \n",s)
30 printf("(b) C = %.1f uF \n",c*10**6)
31 printf("(c) At 0.6 Fp, I = %.2f A \n",i1)
32 printf("      At unity Fp, I = %.2f A \n",i2)
33 printf("(d) Im = %.2f A < %.2f degree \n",im,teta)

```

Scilab code Exa 19.6 Example 6

```

1 //Chapter 19, Example 19.6, page 867
2 clc
3 //Initialisation
4 s=20000 //in VA
5 fp=0.7 //power factor
6 p1=10000 //power in watt
7 p2=14000 //power in watt from fig.
      19.28
8 e=1000 //voltage
9 fp2=0.95 //power factor
10 f=60 //frequency in hertz
11
12 //Calculation
13 p=s*fp //power factor
14 teta = acosd(fp) //phase angle in
      degree
15 ql=s*sind(teta) //in VAR
16 st=sqrt(((p1+p2)**2)+(ql**2)) //in
      VA
17 it=st/e //current in ampere
18 teta2=acosd(fp2) //phase angle in
      degree
19 ql1=(p1+p2)*tand(teta2) //in VAR
20 qc=ql-ql1 //in VAR
21 xc=(e**2)/qc //capacitive
      impedance in ohm

```

```

22 c=1/(2*3.14*f*xc)                                     //
    capacitance ih farad
23 st1=sqrt(((p1+p2)**2)+(q11**2))                       //in
    VA
24 it1=st1/e                                             //current in ampere
25
26 //Result
27 printf("(a) It = %.2f A \n",it)
28 printf("      C = %.2f uF \n",c*10**6)
29 printf("(b) I = %.2f A \n",it1)

```

Scilab code Exa 19.7 Example 7

```

1 //Chapter 19, Example 19.7, page 872
2 clc
3 //Initialisation
4 p=75 //power in watt
5 i=5 //current in watt
6 e=120 //voltage
7 rad1=377 //angular frequency in rad/
    s
8 p2=80 //power in watt
9 i2=4 //current in watt
10 p3=52 //power in watt
11 i3=2 //current in watt
12
13 //Calculation
14 r=p/i**2 //resistance in
    ohm
15 zt=e/i //resistance in
    ohm
16 x1=sqrt((zt**2)-(r**2)) //resistance in
    ohm
17 l=x1/(rad1) //inductance
    in henry

```

```

18 r2=p2/i2**2 //resistance
    in ohm
19 r3=p3/i3**2 //resistance
    in ohm
20 zt3=e/i3 //resistance in
    ohm
21 x13=sqrt((zt3**2)-(r3**2)) //resistance in
    ohm
22 l3=x13/(rad1) //
    inductance in henry
23
24 //Result
25 printf("(a) R = %d ohm \n      L = %.2 f mH \n",r,l
    *10**3)
26 printf("(b) R = %d ohm \n",r2)
27 printf("(c) R = %d ohm \n      L = %.2 f mH",r3,l3
    *10**3)

```

Chapter 20

Resonance

Scilab code Exa 20.1 Example 1

```
1 //Chapter 20, Example 20.1
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8
9 //Variable Declaration
10 r = 2 //resistance in ohms
11 e1 = 10 //real value in volts
12 e1o = 0 //angle in degree
13 zt = 2 //real value in ohms
14 zto = 0 //angle in degree
15 x1 = 10 //real value in ohms
16 x1o = 90 //angle in degree
17 fs = 5000 //frequency in hertz
18
19 //Calculation
20 i = e1/r //in ampere
21 [a,b]=voltdivider(i,x1,e1o,x1o) //function of
```

```

    multiplication
22 [c,d]=voldivider(i,x1,e1o,-x1o) //function of
    multiplication
23 qs = x1/r //unit
24 bw = fs/qs //bandwidth in
    hertz
25 phpf=0.5*i*i*r //power in watt
26 //Results
27 printf("a. I = %d A < %d degree \n",i,e1o)
28 printf(" VL = %d V < %d degree \n",a,b)
29 printf(" Vc = %d V < %d degree \n",c,d)
30 printf("b. Qs = %d \n",qs)
31 printf("c. BW = %d Hz\n",bw)
32 printf("d. Pphf = %d W \n",phpf)

```

Scilab code Exa 20.2 Example 2

```

1 //Chapter 20, Example 20.2
2 clc
3
4 //Variable Declaration
5 fs = 4000 //frequency in hertz
6 bw = 400 //bandwidth in hertz
7 r = 10 //resistance in ohm
8 xc = 100 //in ohms
9
10 //Calculation
11 qs = fs/bw //unit
12 xl = qs*r //in ohms
13 l = xl/(2*pi*fs) //inductance in
    millihenry
14 c = 1/(2*pi*fs*xc) //capacitance in
    microfarad
15 //Results
16 printf("a. Qs = %d \n",qs)

```

```
17 printf("b. XL = %d ohms \n",x1)
18 printf("c. L = %.2 f mH \n",l*10^3)
19 printf("    C = %.3 f uF \n",c*10^6)
```

Scilab code Exa 20.3 Example 3

```
1 //Chapter 20, Example 20.3
2 clc
3
4 //Variable Declaration
5 x1 = 300           //in ohms
6 r = 5             //in ohms
7 fs = 12000        //in hertz
8
9
10 //Calculation
11 qs = x1 / r       //unit
12 bw = fs/qs        //bandwidth in hertz
13 f2 = fs + (bw/2) //frequency2 in hertz
14 f1 = fs-100       //frequency1 in hertz
15 //Results
16 printf("a. BW = %d Hz \n",bw)
17 printf("b. f2 = %d Hz \n",f2)
18 printf("    f1 = %d Hz \n",f1)
```

Scilab code Exa 20.4 Example 4

```
1 //Chapter 20, Example 20.4
2 clc
3
4 //Variable Declaration
5 bw = 200           //bandwidth in hertz
6 fs = 2800          //frequency in hertz
```

```

7 c = 101.5*10^-9           //capacitance in
    nanofarad
8 imax = 200*10^-3         //max current in
    milliampere
9
10
11 //Calculation
12 qs = fs/bw              //unit
13 l = 1/(4*pi*pi*fs*fs*c) //inductance in mH
14 xl = 2*pi*fs*l         //in ohm
15 r = xl/qs              //resistance in ohm
16 e = imax*r             //in volts
17 //Results
18 printf("a. BW = %d Hz \n",bw)
19 printf("   Qs = %d \n",qs)
20 printf("b. L = %.3f mH \n",l*10^3)
21 printf("   R = %d ohm \n",r)
22 printf("c. E = %d V \n",e)

```

Scilab code Exa 20.5 Example 5

```

1 //Chapter 20, Example 20.5
2 clc
3
4 //Variable Declaration
5 e = 120           //in volts
6 p = 16           //power in watt
7 ws = 10^5        //in rad/s
8
9 //Calculation
10 r = (e*e)/p      //resistance in ohm
11 fs = ws/(2*pi)  //frequency in hertz
12 bw = 0.15*fs    //bandwidth in hertz
13 l = r/(2*pi*bw) //inductance in mH
14 c = 1/(4*pi*pi*fs*fs*l) //capacitance in nF

```

```

15 x1 = 2*%pi*fs*l           //in ohm
16 qs = x1/r               // unit
17 a = 1/qs                 //unit
18
19 //Results
20 printf("a. P = %d ohm \n",r)
21 printf("b. BW = %.2 f Hz \n",bw)
22 printf("c. L = %d mH \n",l*10^3)
23 printf("   C = %.2 f nF \n",c*10^9)
24 printf("d. Qs = %.2 f \n",qs)
25 printf("e. BW/fs = %.2 f \n",a)

```

Scilab code Exa 20.6 Example 6

```

1 //Chapter 20, Example 20.6
2 clc
3
4 //Variable Declaration
5 l = 1*10^-3           //inductance in mH
6 c = 1*10^-6           //capacitance in uF
7 rs = 10*10^3          //in ohm
8 i = 10*10^-3         //current in milliampere
9
10
11 //Calculation
12 fp = 1/(2*%pi*sqrt(l*c)) //frequency in kHz
13 x1 = 2*%pi*fp*l       //in ohm
14 qp = rs/x1           //unit
15 bw = fp/qp           //bandwidth in Hz
16 f1 = (1/(4*%pi*c))*((1/rs)-sqrt((1/(rs*rs))+((4*c)/l
   ))) //frequency in kHz
17 f2 = (1/(4*%pi*c))*((1/rs)+sqrt((1/(rs*rs))+((4*c)/l
   ))) //frequency in kHz
18 vc = i*rs           //in volts
19 il = vc/x1         //load current in ampere

```

```

20 ic = vc/xl          //in ampere
21
22 //Results
23 printf("a. fp = %.2 f kHz \n",fp/10^3)
24 printf("b. Rs = %d kohm \n",rs/10^3)
25 printf("c. Qp = %.2 f \n",qp)
26 printf("   BW = %.2 f kHz \n",bw)
27 printf("   f1 = %.3 f kHz \n",-f1/10^3)
28 printf("   f2 = %.3 f kHz \n",f2/10^3)
29 printf("d. Vc = %d V \n",vc)
30 printf("e. IL = %.2 f A \n",il)
31 printf("   Ic = %.2 f A \n",ic)

```

Scilab code Exa 20.7 Example 7

```

1 //Chapter 20, Example 20.7
2 clc
3 funcprot(0)
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)
6     r=sqrt((x**2)+(y**2))
7     if x==0 then
8         th=90
9     else
10        th=atand(y/x)
11    end
12 endfunction
13 //multiplication
14 function [r,s] = voltdivider(x1,x2,y1,y2)
15     r=x1*x2
16     s=y1+y2
17 endfunction
18 function [c1,c2]=subtraction(x1,x2,y1,y2)
19     c1 = x1-x2
20     c2 = y1-y2

```

```

21     endfunction
22 //division
23 function [q,x]=division(x1,x2,y1,y2)
24     q=x1/x2
25     x=y1-y2
26     endfunction
27 //Variable Declaration
28 l = 0.3*10^-3           //inductance in mH
29 c = 100*10^-9          //capacitance in nF
30 r1 = 20                 //in ohm
31 xco = -90              //angle in degree
32 i = 2*10^-3           //current in mA
33
34 //Calculation
35 fs = 1/(2*pi*sqrt(l*c)) //frequency in
    kHz
36 fm = fs*(sqrt(1-(1/4)*((r1*r1*c/l)))) //
    frequency in kHz
37 fp = fs*(sqrt(1-((r1*r1*c/l)))) //
    frequency in kHz
38 x1 = 2*pi*fm*l         //in ohm
39 xc = 1/(2*pi*fm*c)     //in ohm
40 [a,b]=rect2polar(r1,x1) //function of
    conversion
41 [c,d]=voldivider(a,xc,b,xco) //function of
    multiplication
42 [e,f]=subtraction(r1,0,x1,xc) //function of
    subtraction
43 [ee,ff]=rect2polar(e,f) //function of
    conversion
44 [m,n]=division(c,ee,d,ff) //function of division
45 vc = i*m              //in mV
46 qp = x1/r1           //unit
47 bw = fp/qp           //bandwidth in kHz
48 ql = (2*pi*fs*l)/r1 //unit
49 ztp = ql*ql*r1       //in ohm
50 vc1 = i*ztp          //in mV
51 bw1 = fp/qp          //bandwidth in kHz

```

```

52
53 // Results
54 printf(" a. fs = %.2 f kHz \n", fs/10^3)
55 printf("    fm = %.2 f kHz \n", fm/10^3)
56 printf("    fp = %.2 f kHz \n", fp/10^3)
57 printf(" b. ZTm = %.2 f ohm < %.2 f degree \n", m, n)
58 printf("    VCmax = %.2 f mV \n", vc*10^3)
59 printf(" c. Qp = %.2 f \n", qp)
60 printf(" d. BW = %.2 f kHz \n", bw/10^3)
61 printf(" e. fs = %.2 f kHz \n", fs/10^3)
62 printf("    Ql = %.2 f \n", ql)
63 printf("    ZTp = %.2 f ohm < 0 degree \n", ztp)
64 printf("    VCmax = %.1 f mV \n", vc1*10^3)
65 printf("    BW = %.2 f kHz \n", bw1/10^3)

```

Scilab code Exa 20.8 Example 8

```

1 //Chapter 20, Example 20.8
2 clc
3
4 //Variable Declaration
5 l = 1*10^-3 //inductance in mH
6 fp = 0.04*10^6 //frequency in
    microfarad
7 rl = 10 //in ohm
8 rs = 40*10^3 //in kohm
9
10 //Calculation
11 xl = 2*pi*fp*l //in ohm
12 ql = xl/rl //unit
13 rp = ql*ql*rl //in kiloohm
14 ztp = (rs*rp)/(rs+rp) //in kiloohm
15 c = 1/(4*pi*pi*fp*fp*l) //capacitance in nF
16 qp = ztp/xl //unit
17 bw = fp/qp //bandwidth in kHz

```



```

18 f1 = (1/(4*%pi*c))*((1/ztp)-sqrt((1/(ztp*ztp))+((4*c
    )/1))) //frequency in kHz
19 f2 = (1/(4*%pi*c))*((1/ztp)+sqrt((1/(ztp*ztp))+((4*c
    )/1))) //frequency in kHz
20 //Results
21 printf("a. Ql = %.2 f \n",q1)
22 printf("b. Rp = %.2 f kohm \n",rp/10^3)
23 printf("c. ZTp = %.2 f kohm \n",ztp/10^3)
24 printf("d. C = %.2 f nF \n",c*10^9)
25 printf("e. Qp = %.2 f \n",qp)
26 printf("f. BW = %.2 f kHz \n",bw/10^3)
27 printf("    f1 = %.3 f kHz \n",-f1/10^3)
28 printf("    f2 = %.3 f kHz \n",f2/10^3)

```

Scilab code Exa 20.9 Example 9

```

1 //Chapter 20, Example 20.9
2 clc
3 stacksize(10000000);
4 //Variable Declaration
5 l = 5*10^-3 //inductance in mH
6 c = 50*10^-12 //capacitance in pF
7 rl = 100 //in ohm
8 rs = 50*10^3 //in kohm
9 i = 2*10^-3 //current in mA
10
11 //Calculation
12 fs = 1/(2*%pi*sqrt(l*c)) //frequency in kHz
13 xl = 2*%pi*fs*l //in kohm
14 ql = xl/rl //unit
15 fp = fs //frequency in kHz
16 rp = ql*ql*rl //in Megaohm
17 ztp=((rs*rp)/(rs+rp)) //in kohm
18 qp = ztp/xl //unit
19 bw = fp/qp //frequency in kHz

```

```

20 bw1 = (1/(2*%pi))*((r1/l)+(1/(rs*c)))           //
    frequency in kHz
21 vp = i*ztp                                     //in volts
22
23 //graph
24 x= 100 : 10000 : 500000
25 //x=318.31*10**3
26 rp1=(4*(%pi**2)*(l**2)*(x.^2))/r1
27 //rp1=r1**2/(4*(%pi**2)*(l**2)*(x.^2))
28 vp1=i*((rs*rp1)./(rs+rp1))
29
30 plot(x, vp1);
31 xtitle('Passband response')
32 xlabel('f')
33 ylabel('Vo')
34 //insufficient data from textbook for plotting a
    graph
35
36
37 //Results
38 printf("a. fp = fs = %.2 f kHz \n", fs/10^3)
39 printf("b. Qp = %.2 f \n", qp)
40 printf("c. BW = %.2 f kHz \n", bw/10^3)
41 printf("    BW = %.2 f kHz \n", bw1/10^3)
42 printf("d. Vp = %.2 f V \n", vp)

```

Scilab code Exa 20.10 Example 10

```

1 //Chapter 20, Example 20.10
2 clc
3 //Variable Declaration
4 fp = 318.31*10^3           //frequency in kHz
5 ql = 100                  //unit
6 rp = 1*10^6               //in Mega ohm
7 i = 2*10^-3              //current in mA

```

```

8
9 // Calculation
10 qp = ql //unit
11 bw = fp/qp //frequency in kHz
12 ztp = rp //in mega ohm
13 vp = i*ztp //in volts
14
15
16
17 // Results
18 printf("a. fp = %.2 f kHz \n",fp/10^3)
19 printf("b. Qp = %d \n",qp)
20 printf("c. BW = %.3 f kHz \n",bw/10^3)
21 printf("d. Vp = %d V ",vp)

```

Scilab code Exa 20.11 Example 11

```

1 //Chapter 20, Example 20.11
2 clc
3 //Variable Declaration
4 fp = 50000 //frequency in Hz
5 bw = 2500 //bandwidth in Hz
6 l = 1*10^-3 //inductance in mH
7 rl = 10 //resistance in ohm
8 vp = 10 //in volts
9 rs = 17.298*10^3 //in kohms
10 // Calculation
11 qp = fp/bw //unit
12 xl = 2*%pi*fp*l //in ohm
13 ql = xl/rl //unit
14 rp = ql*ql*rl //in ohm
15 xc = xl //in ohm
16 c = 1/(2*%pi*fp*xc) //in uF
17 ztp = (rs*rp)/(rs+rp) //in kohm
18 i = vp/ztp //current in mA

```

```
19 // Results
20 printf("Qp = %d \n", qp)
21 printf("Ql = %.1 f \n", ql)
22 printf("Rp = %.1 f ohm \n", rp)
23 printf("C = %.2 f uF \n", c*10^6)
24 printf("I = %.1 f mA \n", i*10^3)
```

Chapter 21

Transformers

Scilab code Exa 21.1 Example 1

```
1 //Chapter 21, Example 21.1
2 clc
3 //Initialisation
4 k=0.6 //coefficient of
      coupling
5 Lp=200*10**-3 //in Henry
6 Ls=800*10**-3 //in Henry
7 Np=50 //no of turns in
      primary
8 phip=450*10**-3 //in Wp/s
9 Ns=100 //no of turns in
      primary
10 ip=0.2/10**-3 //in Wp/s
11
12 //Calculation
13 M=k*sqrt(Lp*Ls) //Mutual Inductance
14 ep=Np*phip //induced voltage
15 es=k*Ns*phip //induced voltage
16 epi=Lp*ip //induced voltage
17 esi=M*ip //induced voltage
18
```

```

19 //Result
20 printf("(a) Mutual Inductance , M = %.3 f mH \n",M
    *1000)
21 printf("(b) Induced Voltage , ep = %.1 f V \n",ep)
22 printf("(c) Induced Voltage , es = %d V \n",es)
23 printf("(d) Induced Voltage , ep = %d V \n",epi)
24 printf("(e) Induced Voltage , es = %d V",esi)

```

Scilab code Exa 21.2 Example 2

```

1 //Chapter 21, Example 21.2
2 clc
3 //Initialisation
4 ep=200 //induced voltage
5 np=50 //no of turns
6 f=60 //frequency in hertz
7 es=2400 //induced voltage
8
9 //Calculation
10 phim=ep/(4.44*np*f) //in Wb/m
11 ns=(np*es)/ep //no of turns
12
13
14 //Result
15 printf("(a) Maximum Flux = %.2 f mWb \n",phim*10**3)
16 printf("(b) Secondary turns Ns = %d turns",ns)

```

Scilab code Exa 21.3 Example 3

```

1 //Chapter 21, Example 21.3
2 clc
3 //Initialisation
4 np=40 //no of turns

```

```

5 ns=5 //no of turns
6 vl=200 //voltage
7 zl=2*10**3 //resistance in ohm
8 is=0.1 //current in ampere
9
10 //Calculation
11 Ip=(ns/np)*is //current in amp
12 vg=(np/ns)*vl //voltage in volt
13 a=np/ns //turn ratio
14 zp=(a**2)*(zl) //in ohm
15
16
17 //Result
18 printf("(a) Ip = %.1 f mA \n",Ip*10**3)
19 printf(" Vg = %d V \n",vg)
20 printf("(b) Zp = %d Kohm",zp/10**3)

```

Scilab code Exa 21.4 Example 4

```

1 //Chapter 21, Example 21.4
2 clc
3 //Initialisation
4 p1=600 //power in watt
5 p2=400 //power in watt
6 p3=2000 //power in watt
7 vp=2400 //voltage in volt
8 v1=120 //voltage in volt
9 v2=240 //voltage in volt
10
11 //Calculation
12 pt=p1+p2+p3 //power in watt
13 Ip=pt/vp //current in ampere
14 R=vp/Ip //resistance in ohm
15 i1=p1/v1 //current in ampere
16 i2=p3/v2 //current in ampere

```

```

17 v1=1.73*vp           //voltage in volt
18 pt1=3*pt            //power in watt
19 a=vp/v2             //ratio
20
21 //Result
22 printf("(a) R = %d ohm \n",R)
23 printf("(b) I1 = %d A \n",i1)
24 printf("      I2 = %.2f A \n",i2)
25 printf("(c) V1 = %d V \n",v1)
26 printf("(d) Pt = %d kW \n",pt1/1000)
27 printf("(e) a = %d \n",a)

```

Scilab code Exa 21.5 Example 5

```

1 //Chapter 21, Example 21.5
2 clc
3 //Initialisation
4 E=120           //voltage in volt
5 rt1=512         //resistance in ohm
6 rt2=8           //resistance in ohm
7 R=8             //resistance in ohm
8 np=8            //no of turns
9 ns=1            //no of turns
10
11 //Calculation
12 is=E/(rt1+rt2) //current in ampere
13 p=(is**2)*R    //power in watt
14 zp=((np/ns)**2)*R //resistance in ohm
15 is2=E/(rt1+zp) //current in ampere
16 p2=(is2**2)*zp //power in watt
17
18
19 //Result
20 printf("(a) P = %.2f W \n",p)
21 printf("(b) Zp = %d ohm \n",zp)

```



```
22 printf("      P = %.3f W \n", p2)
```

Scilab code Exa 21.6 Example 6

```
1 //Chapter 21, Example 21.6
2 clc
3 //Initialisation
4 vp=70.7 //voltage in volt
5 pp=10 //power in watt
6 z1=8 //resistance in ohm
7 rt=500
8
9 //Calculation
10 pt=pp*4 //resulting power in watt
11 ip=pp/vp //current in ampere
12 zp=vp/ip //resistance in ohm
13 zp1=sqrt(zp/z1) //resistance in ohm
14 vl=vp/zp1 //voltage in volt
15 rt2=rt/2 //resistance in ohm
16 rt3=rt/3 //resistance in ohm
17 rt4=rt/4 //resistance in ohm
18
19 //Result
20 printf("(a) Resulting power = %d W \n", pt)
21 printf("(b) Zp = %d ohm \n", zp)
22 printf("(c) Zp = %d : 1 \n", round(zp1))
23 printf("(d) VL = %d V \n", round(vl))
24 printf("(e) One Speaker = %d ohm \n", rt)
25 printf(" Two Speaker = %d ohm \n", rt2)
26 printf(" Three Speaker = %d ohm \n", rt3)
27 printf(" Four Speaker = %d ohm \n", rt4)
```

Scilab code Exa 21.7 Example 7

```

1 //Chapter 21, Example 21.7
2 clc
3 funcprot()
4 function [r,th]=rect2pol(x,y)
5 //rectangle to polar coordinate conversion
6   r=sqrt(x^2+y^2);
7   th = atan(y,x)*180/%pi;
8 endfunction
9
10 //Initialisation
11 rp=1           //in ohm
12 a=2           //turn ratio
13 rs=1         //in ohm
14 xp=2         //in ohm
15 xs=2         //in ohm
16 ip=10        //in ampere
17 rl=60        //in ohm
18 vl=1200      //in volt
19
20 //Calculation
21 Re=rp+((a**2)*rs)           //in ohm
22 Xe=xp+((a**2)*xs)         //in ohm
23 c=Re+((a**2)*rl)
24 vg=ip*complex(c,Xe)        //in volt
25 [vgr,vgi]=rect2pol(real(vg),imag(vg)) //
    rectangle to polar conversion
26 vg2=a*vl           //in volt
27
28 //Result
29 printf("(a) Re = %d ohm \n",Re)
30 printf("   Xe = %d ohm \n",Xe)
31 printf("(b) Vg = %.2f V < %.2f degree \n",vgr,vgi)
32 printf("(c) Vg = %.2f V \n",vg2)

```

Scilab code Exa 21.8 Example 8

```

1 //Chapter 21, Example 21.8
2 clc
3 //Initialisation
4 L1=5           //in henry
5 L2=10          //in henry
6 L3=15          //in henry
7 M12=2          //in henry
8 M23=3          //in henry
9 M13=1          //in henry
10
11
12 //Calculation
13 Lt=L1+L2+L3+(2*M12)-(2*M23)-(2*M13)           //in
    henry
14
15
16 //Result
17 printf("Lt = %d H \n",Lt)

```

Scilab code Exa 21.10 Example 10

```

1 //Chapter 21, Example 21.10
2 clc
3 funcprot(0)
4 function [r,th]=rect2pol(x,y)
5 //rectangle to polar coordinate conversion
6   r=sqrt(x^2+y^2);
7   th = atan(y,x)*180/%pi;
8 endfunction
9
10 //Initialisation
11 Rp=3           //in ohm
12 Xlp=2400       //in ohm
13 W=400          //angular frequency in rad/s
14 M=0.9          //in henry

```

```

15 Rs=0.5           //in ohm
16 Rl=40           //in ohm
17 Xls=400         //in ohm
18
19 //Calculation
20 Zp = complex(Rp,Xlp) //in ohm
21 Zi=Zp+(((W*M)**2)/(complex(Rs+Rl,Xls))) //Input
    impedance in ohm
22 [Zir,Zid]=rect2pol(real(Zi),imag(Zi))
23
24 //Result
25 printf("Zi = %.1f ohm + j %d ohm\n",real(Zi),imag(Zi
    ))
26 printf("    = %.2f ohm < %.2f degree",Zir,Zid)

```

Chapter 22

Polyphase Systems

Scilab code Exa 22.1 Example 1

```
1 //Chapter 22, Example 22.1
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12
13 //division
14 function [q,x]=division(x1,x2,y1,y2)
15     q=x1/x2
16     x=y1-y2
17 endfunction
18 //Polar to Rectangle conversion
19 function [r,i]= polar2rect(x,y)
20     r=x*cosd(y)
21     i=x*sind(y)
```

```

22 endfunction
23 //addition
24 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
25     c1 = x1+x2+x3
26     c2 = y1+y2+y3
27 endfunction
28 //Variable Declaration
29 teta2 = -120 //angle in degree
30 teta3 = 120 //angle in degree
31 eo = 120 //in volt
32 van = 120 //in volt
33 vano = 0 //angle in degree
34 vbn = 120 //in volt
35 vbno = -120 //angle in degree
36 vcn = 120 //in volt
37 vcno = 120 //angle in degree
38 zan = complex(3,4) //complex form of Zan
39 zbn = 5 //in ohm
40 zbno = 53.13 //angle in degree
41 zcn = 5 //in ohm
42 zcno = 53.13 //angle in degree
43
44 //Calculation
45 e1 = sqrt(3)*eo //in volt
46 [a,b]=rect2polar(real(zan),imag(zan)) //function
    of conversion
47 [b,c]=division(van,a,vano,b) //function of
    division
48 [e,f]=division(vbn,zbn,vbno,zbno) //function of
    division
49 [g,h]=division(vcn,zcn,vcno,zcno) //function of
    division
50 [bb,cc]=polar2rect(b,c) //function of conversion
51 [ee,ff]=polar2rect(e,f) //function of conversion
52 [gg,hh]=polar2rect(g,h) //function of conversion
53 [m,n]=addition(bb,ee,gg,cc,ff,hh) //function of
    addition
54 //Results

```

```

55 printf("a. theta2 = %d degree and theta3 = +%d
    degree \n",teta2,teta3)
56 printf("b. EL = %d V \n",e1)
57 printf("c. Ian = %d A < %.2f degree \n",b,c)
58 printf("    Ibn = %d A < %.2f degree \n",e,f)
59 printf("    Icn = %d A < %.2f degree \n",g,h)
60 printf("d. IN = %d + j%d \n",m,n)

```

Scilab code Exa 22.2 Example 2

```

1 //Chapter 22, Example 22.2
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12
13 //division
14 function [q,x]=division(x1,x2,y1,y2)
15     q=x1/x2
16     x=y1-y2
17 endfunction
18 //Polar to Rectangle conversion
19 function [r,i]= polar2rect(x,y)
20     r=x*cosd(y)
21     i=x*sind(y)
22 endfunction
23 //addition
24 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
25     c1 = x1+x2+x3

```

```

26     c2 = y1+y2+y3
27     endfunction
28 //Variable Declaration
29 teta2 = -120           //angle in degree
30 teta3 = 120           //angle in degree
31 vab = 150             //in volts
32 vabo = 0              //angle in degree
33 vbc = 150             //in volts
34 vbco = -120           //angle in degree
35 vca = 150             //in volts
36 vcao = 120            //angle in degree
37 zab = complex(6,8)    //complex form of Zab
38 zbc = 10              //in ohm
39 zbco = 53.13          //angle in degree
40 zca = 10              //in ohm
41 zcao = 53.13          //angle in degree
42 io = 15               //in ampere
43
44 //Calculation
45 [a,b]=rect2polar(real(zab), imag(zab))           //
    function of conversion
46 [c,d]=division(vab,a,vabo,b)                    //function of
    division
47 [e,f]=division(vbc,zbc,vbco,zbco)               //function of
    division
48 [g,h]=division(vca,zca,vcao,zcao)               //function of
    division
49 il = sqrt(3)*io                                  //in ampere
50 //Results
51 printf("a. theta2 = %d degree and theta3 = +%d
    degree \n",teta2,teta3)
52 printf("b. Iab = %d A < %.2f degree \n",b,c)
53 printf("   Ibc = %d A < %.2f degree \n",e,f)
54 printf("   Ica = %d A < %.2f degree \n",g,h)
55 printf("c. IL = %.2f A \n",il)

```

Scilab code Exa 22.3 Example 3

```
1 //Chapter 22, Example 22.3
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12
13 //division
14 function [q,x]=division(x1,x2,y1,y2)
15     q=x1/x2
16     x=y1-y2
17 endfunction
18 //Polar to Rectangle conversion
19 function [r,i]= polar2rect(x,y)
20     r=x*cosd(y)
21     i=x*sind(y)
22 endfunction
23 //addition
24 function [c1,c2]=addition(x1,x2,x3,y1,y2,y3)
25     c1 = x1+x2+x3
26     c2 = y1+y2+y3
27 endfunction
28 //multiplication
29 function [r,s] = voltdivider(x1,x2,y1,y2)
30     r=x1*x2
31     s=y1+y2
32 endfunction
```

```

33 //Variable Declaration
34 teta2 = -120           //angle in degree
35 teta3 = 120           //angle in degree
36 vab = 120             //in volts
37 vabo = 0              //angle in degree
38 vbc = 120             //in volts
39 vbco = 120           //angle in degree
40 vca = 120             //in volts
41 vcao = -120          //angle in degree
42 zab1 = 5              //in volt
43 zab1o = 0             //angle in degree
44 zab2 = 5              //in volt
45 zab2o = -90           //angle in degree
46 zab = complex(5,-5)  //complex form Zab
47 zbc = 3.54            //in ohm
48 zbco = -45           //angle in degree
49 zca = 3.54            //in ohm
50 zcao = -45           //angle in degree
51 io = 34               //in ampere
52
53 //Calculation
54 [aa,bb]=voltdivider(zab1,zab2,zab1o,zab2o) //
    function of multiplication
55 [a,b]=rect2polar(real(zab),imag(zab)) //
    function of conversion
56 [s,t]=division(aa,a,bb,b) //function of
    division
57 [c,d]=division(vab,s,vabo,t) //function of
    division
58 [e,f]=division(vbc,zbc,vbco,zbco) //function of
    division
59 [g,h]=division(vca,zca,vcao,zcao) //function of
    division
60 il = sqrt(3)*io //in ampere
61 //Results
62 printf("a. theta2 = %d degree and theta3 = +%d
    degree \n",teta2,teta3)
63 printf("b. Iab = %.1f A < %d degree \n",c,d)

```

```

64 printf(" Ibc = %.1f A < %d degree \n",e,f)
65 printf(" Ica = %.1f A < %d degree \n",g,h)
66 printf(" c. IL = %.2f A \n",i1)

```

Scilab code Exa 22.4 Example 4

```

1 //Chapter 22, Example 22.4
2 clc
3 //multiplication
4 function [r,s] = voltdivider(x1,x2,y1,y2)
5     r=x1*x2
6     s=y1+y2
7 endfunction
8 //Variable Declaration
9 ian = 2 //in ampere
10 iano = 0 //angle in degree
11 ibn = 2 //in ampere
12 ibno = -120 //angle in degree
13 icn = 2 //in ampere
14 icno = 120 //angle in degree
15 zan = 10 //in ohm
16 zano = -53.13 //angle in degree
17 //Calculation
18 [a,b]=voltdivider(ian,zan,iano,zano) //function
    of multiplication
19 [c,d]=voltdivider(ibn,zan,ibno,zano) //function
    of multiplication
20 [e,f]=voltdivider(icn,zan,icno,zano) //function
    of multiplication
21 e1 = sqrt(3)*a //in volt
22 //Results
23 printf(" a. Van = %d V < %.2f degree \n",a,b)
24 printf(" Vbn = %d V < %.2f degree \n",c,d)
25 printf(" Vcn = %d V < %.2f degree \n",e,f)
26 printf(" b. El = %.1f V ",e1)

```

Scilab code Exa 22.5 Example 5

```
1 //Chapter 22, Example 22.5
2 clc
3 //Variable Declaration
4 vo = 100 //in volt
5 io = 20 //in ampere
6 teta = 53.13 //angle in degree
7 ro = 3 //in ohm
8 vr = 60 //in volt
9 el = 173.2 //in volt
10 il = 20 //in ampere
11 xo = 4 //in ohm
12 //Calculation
13 po = vo*io*cosd(teta) //in watt
14 po1 = io*io*ro //in watt
15 po2 = (vr*vr)/ro //in watt
16 pt = 3*po //in watt
17 pt1 = sqrt(3)*el*il*cosd(teta) //in watt
18 qo = vo*io*sind(teta) //in VAR
19 qo1 = io*io*xo //in
   VAR
20 qt = 3*qo //in VAR
21 qt1 = sqrt(3)*el*il*sind(teta) //in
   VAR
22 so = vo*io //in VA
23 st = 3*so //in VA
24 st1 = sqrt(3)*el*il //in VA
25 fp = pt1/st1 //lagging
26 //Results
27 printf(" a. Po = %d W \n",po)
28 printf(" Po = %d W \n",po1)
29 printf(" Po = %d W \n",po2)
30 printf(" Pt = %d W \n",pt)
```

```

31 printf(" Pt = %d W \n",pt1)
32 printf("b. Qo = %d VAR \n",qo)
33 printf(" Qo = %d VAR \n",qo1)
34 printf(" Qt = %d VAR \n",qt)
35 printf(" Qt = %d VAR \n",qt1)
36 printf("c. So = %d VA \n",so)
37 printf(" St = %d VA \n",st)
38 printf(" St = %d VA \n",st1)
39 printf("d. Fp = %.1f lagging \n",fp)

```

Scilab code Exa 22.6 Example 6

```

1 //Chapter 22, Example 22.6
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12 //Variable Declaration
13 zd = complex(6,-8) //complex form
    of Zd
14 e1 = 200 //in volt
15 io = 20 //in ampere
16 ro = 6 //in ohm
17 xo = 8 //in ohm
18 vo = 200 //in volt
19 zy = complex(4,3) //complex form of Zy
20 ro1 = 4 //in ohm
21 xo1 = 3 //in ohm
22 vo1 = 116 //in volt

```

```

23
24 // Calculation
25 [a,b]=rect2polar(real(zd),imag(zd)) //function
    of conversion
26 io = e1/a //in ampere
27 pt = 3*io*io*ro //power in watt
28 qt = 3*io*io*xo //in VAR
29 st = 3*vo*io //in VA
30 [c,d]=rect2polar(real(zy),imag(zy)) //function
    of conversion
31 io1 = (e1/sqrt(3))/(c) //in ampere
32 pty = 3*(io1*io1)*ro1 //power in
    watt
33 qty = 3*(io1*io1)*xo1 //in VAR
34 sty = 3*vo1*io1 //in VA
35 ptt = pt+pty //power in watt
36 qtt = qt-qty //in VAR
37 stt = sqrt((ptt*ptt)+(qtt*qtt)) //in VA
38 fp = ptt/stt //leading
39 // Results
40 printf("For the delta: \n")
41 printf("Pt = %d W \n",pt)
42 printf("Qt = %d VAR (C) \n",qt)
43 printf("St = %d VA \n",st)
44 printf("For he Y: \n")
45 printf("Pty = %d W \n",pty)
46 printf("Qty = %d VAR (L) \n",qty)
47 printf("Sty = %.2 f VA \n",sty)
48 printf("For the total load:\n")
49 printf("Pt = %d W \n",ptt)
50 printf("Qt = %.d VAR (C) \n",qtt)
51 printf("St = %.1 f VA \n",stt)
52 printf("Fp = %.3 f leading \n",fp)

```

Scilab code Exa 22.7 Example 7

```

1 //Chapter 22, Example 22.7
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=90
8     else
9         th=atand(y/x)
10    end
11 endfunction
12 //addition
13 function [c1,c2]=addition(x1,x2,y1,y2)
14     c1 = x1+x2
15     c2 = y1+y2
16     endfunction
17 //multiplication
18 function [r,s] = voltdivider(x1,x2,y1,y2)
19     r=x1*x2
20     s=y1+y2
21 endfunction
22
23 //Polar to Rectangle conversion
24 function [r,i]= polar2rect(x,y)
25     r=x*cosd(y)
26     i=x*sind(y)
27 endfunction
28
29 //Variable Declaration
30 v1 = 12000 //in volt
31 pt = 160000 //power in watt
32 vo = 6936.42 //in volt
33 voo = 0 //angle in degree
34 co = 0.86 //cos
35 ioo= -30.68 //angle in
    degree
36 z1 = 25 //in ohm
37 zlo = 53.13 //angle in

```

```

    degree
38 r1 = 15 //in ohm
39 //Calculation
40 vo = vl/sqrt(3) //in volt
41 io = pt/(3*vo*co) //in ampere
42 [a,b]=voltdivider(io,z1,i00,zlo) //function of
    multiplication
43 [c,d]=polar2rect(a,b) //function of
    conversion
44 [e,f]=polar2rect(vo,voo) //function of
    conversion
45 [g,h]=addition(c,e,d,f) //function of
    addition
46 [gg,hh]=rect2polar(g,h) //function of
    conversion
47 eab = sqrt(3)*gg //in volt
48 pline = 3*io*io*r1 //power line
    in watt
49 ptt= pt+pline //total power
50 co1= pt/(sqrt(3)*eab*io) //cos
51 n = (pt/ptt)*100 //efficiency in
    percentage
52 //Results
53 printf("a. Io = %.2f A \n",io)
54 printf(" Eab = %.2f V \n",eab)
55 printf("b. Fp = %.2f < 0.86 of load \n",co1)
56 printf("c. n = %.1f percent \n",n)

```

Scilab code Exa 22.8 Example 8

```

1 //Chapter 22, Example 22.8
2 clc
3 funcprot(0)
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)

```



```

6     r=sqrt((x**2)+(y**2))
7     if x==0 then
8         th=-90
9     else
10        th=atand(y/x)
11        if (x<0) & (y<0) then
12            th=th-180
13        end
14    end
15 endfunction
16 //Rectangle to Polar Conversion
17 function [r,th] = rect2polar1(x,y)
18     r=sqrt((x**2)+(y**2))
19     if x==0 then
20         th=90
21     else
22         th=-atand(y/x)
23     end
24 endfunction
25
26 //subtraction
27 function [c1,c2]=subtraction(x1,x2,y1,y2)
28     c1 = x1-x2
29     c2 = y1-y2
30 endfunction
31 //division
32 function [q,x]=division(x1,x2,y1,y2)
33     q=x1/x2
34     x=y1-y2
35 endfunction
36 //Polar to Rectangle conversion
37 function [r,i]= polar2rect(x,y)
38     r=x*cosd(y)
39     i=x*sind(y)
40 endfunction
41
42 //Variable Declaration
43 eab = 208 //real value

```

```

44 eabo = 0 //angle in degree
45 zab = 10 //real value
46 zabo = 0 //angle in degree
47 ebco = -120 //angle in degree
48 ecao = 120 //angle in degree
49 zbc = complex(15,20) //complex form of
    Zbc
50 zca = complex(12,12) //complex form of
    Zca
51
52 //Calculation
53 [a,b]=division(eab,zab,eabo,zabo) //Iab
    function of division
54 [c,d]=rect2polar(real(zbc),imag(zbc)) //
    function of conversion
55 [e,f]=division(eab,c,ebco,d) //Ibc
    function of division
56 [g,h]=rect2polar1(real(zca),imag(zca))
    //function of conversion
57 [m,n]=division(eab,g,ecao,h) //Ica
    function of division
58 [aa,bb]=polar2rect(a,b) //function of
    conversion
59 [mm,nn]=polar2rect(m,n) //function of
    conversion
60 [o,p]=subtraction(aa,mm,bb,nn) //function of
    subtraction
61 [s,t]=rect2polar(o,p) //Iaa function
    of conversion
62 [ee,ff]=polar2rect(e,f) //function of
    conversion
63 [oo,pp]=subtraction(ee,aa,ff,bb) //function of
    subtraction
64 [oa,pa]=rect2polar(oo,pp) //function of
    conversion
65 [ob,pb]=subtraction(mm,ee,nn,ff) //function of
    subtraction
66 [oc,pc]=rect2polar(ob,pb) //function of

```

```

        conversion
67 od=(eab*s)*cosd(5.55) //in
    watt
68 pd = (eab*oc)*cosd(70.65) //in
    watt
69 pt = od+pd //in watt
70 ptt = (a*a*zab)+(e*e*real(zbc))+(m*m*real(zca)) //
    in watt
71 //Results
72 printf("a. Iab = %.1 f A < %d degree \n",a,b)
73 printf("   Ibc = %.2 f A < %.2 f degree \n",e,f)
74 printf("   Ica = %.2 f A < %d degree \n",m,n)
75 printf("b. IAa = %.2 f A < %.2 f degree \n",s,t)
76 printf("   IBb = %.2 f A < %.2 f degree \n",oa,pa)
77 printf("   ICc = %.1 f A < %.2 f degree \n",oc,pc)
78 printf("c. P1 = %.2 f W \n",od)
79 printf("   P2 = %.1 f W \n",pd)
80 printf("d. PT = %.2 f W \n",pt)
81 printf("e. PT = %.2 f W \n",ptt)

```

Scilab code Exa 22.9 Example 9

```

1 //Chapter 22, Example 22.9
2 clc
3 //Rectangle to Polar Conversion
4 function [r,th] = rect2polar(x,y)
5     r=sqrt((x**2)+(y**2))
6     if x==0 then
7         th=-90
8     elseif x<0 then
9         th=atand(y/x)+180
10    else
11        th=atand(y/x)
12        if (x<0) & (y<0) then
13            th=th-180

```

```

14         end
15     end
16 endfunction
17
18 //Polar to Rectangle conversion
19 function [r,i]= polar2rect(x,y)
20     r=(x*cosd(y))/1000
21     i=(x*sind(y))/1000
22 endfunction
23
24 //addition
25 function [c1,c2]=addition(x1,x2,y1,y2)
26     c1 = x1+x2
27     c2 = y1+y2
28     endfunction
29 //addition
30 function [c1,c2]=addition1(x1,x2,x3,y1,y2,y3)
31     c1 = x1+x2+x3
32     c2 = y1+y2+y3
33     endfunction
34
35 function [c1,c2]=subtraction(x1,x2,x3,y1,y2,y3)
36     c1 = x1-x2-x3
37     c2 = y1-y2-y3
38     endfunction
39
40 function [c1,c2]=subtraction1(x1,x2,y1,y2)
41     c1 = x1-x2
42     c2 = y1-y2
43     endfunction
44 //multiplication
45 function [r,s] = voltdivider(x1,x2,y1,y2)
46     r=x1*x2
47     s=y1+y2
48 endfunction
49 //multiplication
50 function [r,s] = voltdivider1(x1,x2,y1,y2)
51     r=x1*x2

```

```

52     s=y1-y2
53 endfunction
54 //division
55 function [q,x]=division(x1,x2,y1,y2)
56     q=x1/x2
57     x=y1-y2
58 endfunction
59
60 function [c1,c2]=subtraction1(x1,x2,y1,y2)
61     c1 = x1-x2
62     c2 = y1-y2
63 endfunction
64 //Variable Declaration
65 eca = 200 //real value
66 ecao = 120 //angle in degree
67 z2 = 200 //real value
68 z2o = 0 //angle in degree
69 ebc = 200 //real value
70 ebco = -120 //angle in degree
71 z1 = 166 //real value
72 z1o = -90 //angle in degree
73 z3 = 200 //real value
74 z3o = 0 //angle in degree
75 eab = 200 //real value
76 eabo = 0 //angle in degree
77
78 //Calculation
79 [a,b]=voltdivider(eca,z2,ecao,z2o) //EcaZ2
80 [aa,bb]=voltdivider1(ebc,z1,ebco,z1o) //
    Ebcxz1
81 [e,f]=polar2rect(a,b) //
    function of conversion
82 [ee,ff]=polar2rect(aa,bb) //
    function of conversion
83 [ac,bc]=addition(e,ee,f,ff) //function
    of addiion
84 [ad,bd]=rect2polar(ac,bc) //num
85 [c,d]=voltdivider(z1,z2,z1o,z2o) //z1xz2

```

```

86 [ca,da]=voltdivider(z1,z3,z1o,z3o)           //z1xz3
87 [cb,db]=voltdivider(z2,z3,z2o,z3o)         //z2xz3
88 [cc,dd]=polar2rect(c,d)                     //
      function of conversion
89 [ce,de]=polar2rect(ca,da)                   //
      function of conversion
90 [cf,df]=polar2rect(cb,db)                   //
      function of conversion
91 [cg,dg]=addition1(cc,ce,cf,dd,de,df)        //function
      of addition
92 [ci,di]=rect2polar(cg,dg)                   //
      function of conversion
93 [ch,dh]=division(ad,ci,bd,di)              //function of
      division
94 [i,j]=voltdivider(eab,z3,eabo,z3o)          //function
      of multiplication
95 [ia,ja]=polar2rect(i,j)                     //
      function of conversion
96 [ic,jc]=voltdivider(ebc,z1,ebco,z1o)        //
      function of multiplication
97 [id,jd]=polar2rect(ic,jc)                   //
      function of conversion
98 [ib,jb]=subtraction1(id,ia,jd,ja)           //function
      of subtraction
99 [ie,je]=rect2polar(ib,jb)                   //num
100 [if,jf]=division(ie,ci,je,di)             //function of
      division
101
102 //Results
103 printf("Icn = %.2 f A < %.2 f degree \n",ch,dh)
104 printf("Ibn = %.2 f A < %.2 f degree \n",if,jf)

```

Chapter 23

Decibels Filters and Bode Plots

Scilab code Exa 23.1 Example 1

```
1 //Chapter 23, Example 23.1
2 clc
3 //Variable Declaration
4 d1=7/16 //from plot
5 d2=3/4 //from plot
6
7
8 //Calculation
9 d=d1/d2
10 c=10**d
11 value=(10**2)*c
12
13 //Results
14 printf("Value = %.1f \n",value)
```

Scilab code Exa 23.2 Example 2

```
1 //Chapter 23, Example 23.2
```

```

2  clc
3  //Variable Declaration
4  a1=0.004
5  a2=250000
6  a31=0.08
7  a32=240
8  a41=10**4
9  a42=10**-4
10 a5=10**4
11
12
13 //Calculation
14 a=log10(a1)
15 b=log10(a2)
16 c=log10(a31)+log10(a32)
17 d=log10(a41)-log10(a42)
18 e=log10(a5)
19
20 //Results
21 printf("(a) %.3f \n",a)
22 printf("(b) %.3f \n",b)
23 printf("(c) %.3f \n",c)
24 printf("(d) %.1f \n",d)
25 printf("(e) %d \n",e)

```

Scilab code Exa 23.3 Example 3

```

1 //Chapter 23, Example 23.3
2 clc
3 //Variable Declaration
4 vo=1.2 //output voltage in V
5 vi=2*10**-3 //input voltage
   in mV
6
7

```



```

8 // Calculation
9 dbv=20*log10(vo/vi) //dB gain
10
11 // Results
12 printf("dBv = %.2 f dB\n", dbv)

```

Scilab code Exa 23.4 Example 4

```

1 //Chapter 23, Example 23.4
2 clc
3 //Variable Declaration
4 vo=6.8 //output voltage in V
5 db=36 //dB gain
6
7
8 // Calculation
9 vi=vo/(10**(db/20)) //input voltage
10
11 // Results
12 printf("Vi = %.2 f mV\n", vi*10**3)

```

Scilab code Exa 23.5 Example 5

```

1 //Chapter 23, Example 23.5
2 clc
3 //Variable Declaration
4 r1=1000 //resistance in ohm
5 c1=500*10**-12 //capacitance in farad
6 f1=100000 //frequency in hertz
7 f2=1*10**6 //frequency in hertz
8 vi=20 //input voltage
9
10 // Calculation

```

```

11 fc=1/(2*3.14*r1*c1)           //cutoff
    frequency in hertz
12 xc=1/(2*3.14*c1*fc)          //impedance in ohm
13 xc1=1/(2*3.14*c1*f1)        //impedance in
    ohm
14 xc2=1/(2*3.14*c1*f2)        //impedance in
    ohm
15 vo1=vi/sqrt(((r1/xc1)**2)+1) //output voltage
16 vo2=vi/sqrt(((r1/xc2)**2)+1) //output voltage
17 voc=vi/sqrt(((r1/xc)**2)+1) //output voltage
18
19 //graph
20 //for frequency response
21 x= 0.1*fc : 10000 : fc*20
22 xc11=(2*3.14*c1*x).^ -1
23 vo=vi./sqrt(((r1./xc11)**2)+1)
24 plot2d('ln',x,vo)
25 plot(f1,vo1,'*')
26 plot(f2,vo2,'*')
27 plot(fc,voc,'*')
28 xtitle('Frequency response for the low-pass R-C
    network')
29 xlabel('f');
30 ylabel('Vo');
31
32 //for normalized plot
33 scf(2)
34 plot2d('ln',x,vo/vi)
35 plot(f1,vo1/vi,'*')
36 plot(f2,vo2/vi,'*')
37 plot(fc,voc/vi,'*')
38 xtitle('Normalized plot for the low-pass R-C network
    ')
39 xlabel('f');
40 ylabel('Av');
41
42 //Results
43 printf('(a) fc = %.2f kHz \n',fc)

```

```

44 printf('(b) At f = 100 kHz, Vo = %.2f V \n',vo1)
45 printf('    At f = 1 MHz, Vo = %.1f V \n',vo2)
46 printf("Vi = %.2f mV\n",vi*10**3)

```

Scilab code Exa 23.6 Example 6

```

1 //Chapter 23, Example 23.6
2 clc
3 //Variable Declaration
4 r1=20000 //resistance in ohm
5 c1=1200*10**-12 //capacitance in farad
6
7
8 //Calculation
9 fc=1/(2*3.14*r1*c1) //cutoff frequency in
    hertz
10 f=fc/2 //frequency in hertz
11 xc=1/(2*3.14*c1*f) //capacitive reactance
    in ohms
12 av=1/sqrt(1+((xc/r1)**2)) //voltage gain
13 teta=atand(xc/r1) //phase in degree
14
15 //graph
16 x= 0 : 500 : fc*3
17 xc1=(2*3.14*c1*x).^ -1
18 av1=r1./sqrt((r1**2)+(xc1**2)) //
    normalised gain
19 av2=xc1./sqrt((r1**2)+(xc1**2)) //
    normalised gain
20
21 teta1=atand(xc1/r1) //phase plot
22 teta2=atand(r1*xc1**-1) //phase plot
23 scf(1)
24 plot2d(x,av1)
25 xtitle('Normalised Plot – High Pass')

```

```

26 xlabel('f (log scale)')
27 ylabel('Av')
28
29 scf(2)
30 plot2d(x,av2)
31 xtitle('Normalised Plot - Low Pass')
32 xlabel('f (log scale)')
33 ylabel('Av')
34
35
36 scf(3)
37 plot2d(x,teta1)
38 xtitle('Phase Plot - High Pass')
39 xlabel('f (log scale)')
40 ylabel('teta')
41
42 scf(4)
43 plot2d(x,teta2)
44 xtitle('Phase Plot - Low Pass')
45 xlabel('f (log scale)')
46 ylabel('teta')
47
48 clc
49 //Results
50 printf('(a) fc = %.2f Hz \n',fc)
51 printf("Av = Vo / Vi = %.4f < %.2f degree",av,teta)

```

Scilab code Exa 23.7 Example 7

```

1 //Chapter 23, Example 23.7
2 clc
3 //Variable Declaration
4 r1=1000 //resistance in ohm
5 c1=1.5*10**-9 //capacitance in farad
6 r2=40000 //resistance in ohm

```

```

7  c2=4*10**-12          //capacitance in farad
8
9
10 //Calculation
11 fc1=1/(2*3.14*r1*c1)   //cutoff frequency in
    hertz
12 fc2=1/(2*3.14*r2*c2)   //cutoff frequency in
    hertz
13
14
15
16 //graph
17 x1= 0 : 1000 : fc1*200
18 x2= 0 : 1000 : fc2*1000
19 xc1=(2*3.14*c1*x1).^-1 //
    impedance in ohm
20 xc2=(2*3.14*c2*x2).^-1 //
    impedance in ohm
21 av1=r1./sqrt((r1**2)+(xc1**2)) //
    normalised gain
22 av2=xc2./sqrt((r2**2)+(xc2**2)) //
    normalised gain
23
24
25 scf(1)
26 //plot2d('ln',[x1 x2],[av1 av2])
27 plot2d('ln',x1,av1)
28 xtitle('Normalised Plot - High Pass')
29 xlabel('f')
30 ylabel('Vo')
31
32 scf(2)
33 plot2d('ln',x2,av2)
34 xtitle('Normalised Plot - Low Pass')
35 xlabel('f')
36 ylabel('Vo')
37
38

```

```

39 clc
40 //Results
41 printf('(a) High Pass filter , fc = %.1f kHz \n',fc1
        /1000)
42 printf('      Low Pass filter , fc = %.1f kHz \n',fc2
        /1000)
43 printf("(c) Av = Vo / Vi = %.4f < %.2f degree",av,
        teta)

```

Scilab code Exa 23.8 Example 8

```

1 //Chapter 23, Example 23.8
2 clc
3 //Variable Declaration
4 l1=1*10**-3 //inductance in henry
5 c1=0.01*10**-6 //capacitance in farad
6 r1=33 //resistance in ohm
7 r2=2 //resistance in ohm
8 vi=20*10**-3 //input voltage
9
10 //Calculation
11 fs=1/(2*3.14*sqrt(l1*c1)) //frequency in
    hertz
12 xl=2*3.14*fs*l1 //inductive
    reactance
13 qs=xl/(r1+r2) //Q factor
14 bw=fs/qs //bandwidth in
    hertz
15 vomax=(r1*vi)/(r1+r2) //max voltage
16 vo=0.707*vi*(r1/(r1+r2)) //voltage
17
18
19 //Insufficient data on textbook problem to plot
    graph
20 //graph

```

```

21 x=0:1000:fs*2
22 //ztp1=((2*3.14*x*l1)/(r1+r2)**2)*r2
23 x1=2*3.14*x*l1
24 xc=1/(2*3.14*x*c1)
25 zs=sqrt((r2**2)+(x1-xc')**2)
26 vo1=(zs*vi)./(zs+r1)
27 plot2d(x,vo1)
28 xtitle('Passband response')
29 xlabel('f')
30 ylabel('Vo')
31
32
33 //Results
34 clc
35 printf('(a) fs = %.1f Hz \n',fs)
36 printf('Qs = %.2f \n',qs)
37 printf('BW = %.2f kHz \n',bw/1000)
38 printf('Vomax = %.2f mV \n', vomax*1000)

```

Scilab code Exa 23.9 Example 9

```

1 //Chapter 23, Example 23.9
2 clc
3 //Variable Declaration
4 c=500*10**-12 //
   capacitance in farad
5 f1=200*10**3 //frequency
   in hertz
6 f2=600*10**3 //frequency
   in hertz
7
8
9 //Calculation
10 l1=1/(4*(%pi**2)*(f2**2)*c) //
   inductance in henry

```

```

11 xls=2*%pi*f1*l1 //inductive
    reactance in ohms
12 xc=1/(2*%pi*f1*c) //
    capacitive reactance in ohms
13 j=xc-xls //for
    series elements
14 lp=j/(2*%pi*f1) //inductance in henry
15
16 //Results
17 printf("Ls = %.1 f uH\n",l1*10**6)
18 printf("Lp = %.2 f mH",lp*10**3)

```

Scilab code Exa 23.10 Example 10

```

1 //Chapter 23, Example 23.10
2 clc
3 //Variable Declaration
4 r1=1000 //resistance in ohm
5 c1=0.1*10**-6 //capacitance in farad
6
7
8
9 //Calculation
10 f1=1/(2*3.14*r1*c1) //cutoff frequency in
    hertz
11
12 //for magnitude plot
13 f = 100:10:10*f1;
14 av = (1+(f1./f).^2).^(-1/2); // -10*log10
15 av1 = -20*log10(f1/f1);
16 f2 = f1/10;
17 av2 = -20*log10(f1/f2);
18 f3 = f1/2;
19 av3 = -20*log10(f1/f3);
20 f4 = f1;

```



```

21 av4 = -20*log10(f1/f4);
22 x = [f2 f3 f4 f1];
23 y = [av2 av3 av4 av1];
24 scf(1)
25 gainplot(f,av);
26 a = gca();
27 a.y_location = 'left';
28 a.x_location = 'top';
29 a.x_label.text = 'frequency';
30 a.y_label.text = 'Av';
31 a.title.text = 'Magnitude Plot';
32 plot2d(x,y);
33
34 //for phase plot
35 f01 = 10:10:10*f1;
36 teta = atand(f1./f)
37 f11=f1
38 teta1 = atand(f1./f11)
39 f12=f1/10
40 teta2 = atand(f1./f12)
41 f13=f1*10
42 teta3 = atand(f1./f13)
43 x1 = [f11 f12 f13 ];
44 y1 = [teta1 teta2 teta3];
45 scf(2)
46 phaseplot(f,av,teta)
47 b = gca();
48 b.x_label.text = 'frequency';
49 b.y_label.text = 'Av';
50 b.title.text = 'Phase Plot';
51 plot2d(x1,y1)
52
53 f5=1000 //
    frequency in hertz
54 av3 = 20*log10(1/sqrt((((f1/f5)**2)+1))); //gain
55
56
57 //Results

```

```

58 printf('(a) fc = %.2f Hz \n',f1)
59 printf('      Magnitude plot shown in window 1 \n')
60 printf('(b) |AvdB| = %.2f dB \n',av3)
61 printf('(c) Phase plot shown in window 2')

```

Scilab code Exa 23.11 Example 11

```

1 //Chapter 23, Example 23.11
2 clc
3 //Variable Declaration
4 r1=9100 //resistance in ohm
5 c1=0.47*10**-6 //capacitance in farad
6 r2=1000 //resistance in ohm
7
8
9 //Calculation
10 r12=(r1*r2)/(r1+r2) //parallel resistance in
    ohm
11 f1=1/(2*3.14*r1*c1) // frequency in hertz
12 fc=1/(2*3.14*r12*c1) // frequency in hertz
13 al=-20*log10((r1+r2)/r2) //maximum low-level
    attenuation
14
15 teta1=-(atand(f1/f1))+ (atand(fc/f1)) //in degrees
16 teta2=-(atand(f1/fc))+ (atand(fc/fc)) //in degrees
17 teta3=-(atand(f1/120))+ (atand(fc/120)) //in
    degrees
18
19 //for magnitude plot
20 f = 1:10:8*fc;
21 av=((1+((f1./f).^2))./(1+((fc./f).^2))).^(1/2)
    //-10*log10
22 f2 = f1
23 av2=10*log10((1+((f1./f2).^2))./(1+((fc./f2).^2)))
24 f3 = fc

```

```

25 av3=10*log10((1+((f1./f3).^2))./(1+((fc./f3).^2)))
26 x = [f2 f3];
27 y = [av2 av3];
28 scf(1)
29 gainplot(f,av);
30 a = gca();
31 a.y_location = 'left';
32 a.x_location = 'top';
33 a.x_label.text = 'frequency';
34 a.y_label.text = 'Av';
35 a.title.text = 'Ac dB versus frequency for the filter
    ';
36 plot2d(x,y);
37 plot(f2,av2,'*');
38 plot(f3,av3,'*');
39
40 //for phase plot
41 teta=-(atand(f1./f))+ (atand(fc./f))
42 scf(2)
43 plot2d('ln',f,teta)
44 plot(f1,teta1,'*')
45 plot(fc,teta2,'*')
46 plot(120,teta3,'*')
47 xtitle('phase vs frequency for the filter')
48 xlabel('frequency')
49 ylabel('phase in degree')
50
51
52 //Results
53 printf('(a) f1 = %.1f Hz \n',f1)
54 printf('    fc = %.2f Hz \n',fc)
55 printf('(b) At f1 = 32.2 Hz, teta = %.2f degree \n',
    teta1)
56 printf('    At fc = 376.26 Hz, teta = %.2f degree \n
    ',teta2)
57 printf('    At frequency midway between fc and f1 ,
    like 120 Hz, teta = %.2f degree \n',teta3)

```

Scilab code Exa 23.12 Example 12

```
1 //Chapter 23, Example 23.12
2 clc
3 f1=50 //frequency in hertz
4 f2=200 //frequency in hertz
5 f3=10000 //frequency in hertz
6 f4=20000 //frequency in hertz
7 f8=8500 //frequency in hertz
8 f9=1000 //frequency in hertz
9
10 //Calculations
11 AvdB1=-20*log10(sqrt((1+(f1/f3)^2)*(1+((f2/f3)^2))
    *(1+(f3/f3)^2)*(1+((f3/f4)^2))))
    //in dB
12 AvdB2=-20*log10(sqrt((1+(f8/f3)^2)*(1+((f8/f4)^2))))
    //in dB
13 BW=f8-f2 //in hertz
14 AvdB3=-20*log10(sqrt((1+(f1/f9)^2)*(1+((f2/f9)^2))))
    //in dB
15
16 //phase plot
17 f = 10:100:200000;
18 teta=(atand(f1./f))+ (atand(f2./f))-(atand(f./f3))-(
    atand(f./f4))
19 scf(1)
20 plot2d('ln',f,teta)
21 xgrid(1)
22 a = gca();
23 a.y_location = 'left';
24 a.x_location = 'top';
25 a.x_label.text = 'frequency';
26 a.y_label.text = 'Phase';
27 a.title.text = 'Phase response';
```

```

28
29 //for magnitude plot
30 av = -((1+(f1./f)^2).*((f2./f).^2))^(1/2);
31 av1 = -20*log10(f1/f1);
32 f12 = f1/10;
33 av2 = -20*log10(f1/f12);
34 f13 = f1/4;
35 av3 = -20*log10(f1/f13);
36 f14 = f1/2;
37 av4 = -20*log10(f1/f14);
38
39 x = [f12 f13 f14 f1];
40 y = [av2 av3 av4 av1];
41 scf(2)
42 gainplot(f,av);
43 a = gca();
44 a.y_location = 'left';
45 a.x_location = 'top';
46 a.x_label.text = 'frequency';
47 a.y_label.text = 'Av';
48 a.title.text = 'Bode';
49 plot2d(x,y);
50
51 f1=f2
52 av1 = -20*log10(f1/f1);
53 f12 = f1/10;
54 av2 = -20*log10(f1/f12);
55 f13 = f1/4;
56 av3 = -20*log10(f1/f13);
57 f14 = f1/2;
58 av4 = -20*log10(f1/f14);
59
60 x = [f12 f13 f14 f1];
61 y = [av2 av3 av4 av1];
62
63 gainplot(f,av);
64 a = gca();
65 a.y_location = 'left';

```

```

66 a.x_location = 'top';
67 a.x_label.text = 'frequency';
68 a.y_label.text = 'Av';
69 a.title.text = 'Bode';
70 plot2d(x,y);
71
72
73
74
75 av = -((1+(f./f3)^2).*(1+((f./f4).^2)))^(-1/2);
76
77 gainplot(f,av);
78 a = gca();
79 a.y_location = 'left';
80 a.x_location = 'top';
81 a.x_label.text = 'frequency';
82 a.y_label.text = 'Av';
83 a.title.text = 'Bode';
84 plot2d(x,y);
85
86
87 f1 = f3;
88 f = 0.1*f1:100:10*f1;
89 av = (1+(f/f1)^2)^(-1/2);
90 av1 = -20*log10(f1/f1);
91 f2 = f1*10;
92 av2 = -20*log10(f2/f1);
93 f3 = f1*4;
94 av3 = -20*log10(f3/f1);
95 f4 = f1*2;
96 av4 = -20*log10(f4/f1);
97
98 x = [f1 f4 f3 f2];
99 y = [av1 av4 av3 av2];
100 plot2d(x,y);
101
102 f1 = f4;
103 f = 0.1*f1:100:10*f1;

```

```
104 av = (1+(f/f1)^2)^(-1/2);
105 av1 = -20*log10(f1/f1);
106 f2 = f1*10;
107 av2 = -20*log10(f2/f1);
108 f3 = f1*4;
109 av3 = -20*log10(f3/f1);
110 f4 = f1*2;
111 av4 = -20*log10(f4/f1);
112
113 x = [f1 f4 f3 f2];
114 y = [av1 av4 av3 av2];
115 plot2d(x,y);
116
117
118 printf('for 10kHz,\n')
119 printf('AvdB = %.1f dB \n',AvdB1)
120 printf('for 8.5kHz, \n')
121 printf('AvdB = %.1f dB \n',AvdB2)
122 printf('BW = %.1f kHz \n',BW/1000)
123 printf('for midrange of bandwidth 1kHz, \n')
124 printf('AvdB = %.1f dB \n',AvdB3)
```

Chapter 24

Pulse Waveforms and the RC Response

Scilab code Exa 24.1 Example 1

```
1 //Chapter 24, Example 24.1
2 clc
3 //Initialisation
4 t1=12           //time period in ms
5 t2=7           //time period in ms
6 v1=8           //voltage
7 v2=4           //voltage
8 v3=12          //voltage
9 v4=11          //voltage
10 vb=-4         //voltage from graph
11
12 //Calculation
13 tp=t1-t2      //time period in ms
14 vmax=v1+v2    //maximum amplitude voltage
15 v=(v3+v4)/2   //voltage
16 vp=((v3-v4)/v)*100 //tilt in percent
17
18 //Result
19 printf("(a) Positive-going \n") //from graph
```



```

20 printf("(b) Vb = %d V\n",vb)
21 printf("(c) tp = %d ms \n",tp)
22 printf("(d) Vmax = %d V \n",vmax)
23 printf("(e) V (tilt in percentage) = %.3f percent",
    vp)

```

Scilab code Exa 24.2 Example 2

```

1 //Chapter 24, Example 24.2
2 clc
3 //Initialisation
4 vb=0 //voltage from graph
5 tilt=0 //from graph
6 div=4 //amplitude in div
7 perdiv=10 //mV per div
8 div1=3.2 //division
9 perdiv1=5 //microsec per div
10 div2=0.4 //division
11 div3=0.8 //division
12
13
14
15 //Calculation
16 amp=div*perdiv //amplitude
17 tp=div1*perdiv1 //from graph 24.9
18 tr=div2*perdiv1 //from graph 24.9
19 tf=div3*perdiv1 //from graph 24.9
20
21 //Result
22 printf("(a) Positive-going \n") //from graph
23 printf("(b) Vb = %d V\n",vb)
24 printf("(c) tilt = %d percent \n",tilt)
25 printf("(d) amplitude = %d mV \n",amp)
26 printf("(e) tp = %d us \n",tp)
27 printf("(f) tr = %d us \n",tr)

```

```
28 printf("      tf = %d us \n",tf)
```

Scilab code Exa 24.3 Example 3

```
1 //Chapter 24, Example 24.3
2 clc
3 //Initialisation
4 t1=15           //timeperiod in microsec
5 t2=6            //timeperiod in microsec
6 t3=8            //timeperiod in microsec
7
8
9 //Calculation
10 T=t1-t2        //timeperiod in microsec
11 prf=1/T        //frequency
12 tp=t3-t2       //timeperiod in microsec
13 duty=(tp/T)*100 //duty cycle
14
15 //Result
16 printf("(a) Pulse repition Frequency, prf = %.2f kHz
        \n",prf*1000)
17 printf("(b) Duty Cycle = %.2f percent \n",duty)
```

Scilab code Exa 24.4 Example 4

```
1 //Chapter 24, Example 24.4
2 clc
3 //Initialisation
4 div1=3.2        //no of division
5 ms=1            //ms per divisions
6 div2=0.8        //no of division
7
8
```

```

 9 // Calculation
10 T=div1*ms           //timeperiod in microsec
11 tp=div2*ms         //frequency
12 prf=1/T            //timeperiod in microsec
13 duty=(tp/T)*100    //duty cycle
14
15 // Result
16 printf("(a) Pulse repition Frequency, prf = %.2f Hz
    \n",prf*1000)
17 printf("(b) Duty Cycle = %.2f percent \n",duty)

```

Scilab code Exa 24.5 Example 5

```

1 //Chapter 24, Example 24.5
2 clc
3 // Initialisation
4 div1=2.6           //no of division
5 ms=10              //microsec per divisions
6 div2=0.2           //no of division
7
8
9
10 // Calculation
11 T=div1*ms          //timeperiod in microsec
12 tp=div2*ms        //frequency
13 prf=1/T            //timeperiod in microsec
14 duty=(tp/T)*100   //duty cycle
15
16 // Result
17 printf("(a) Pulse repition Frequency, prf = %d Hz \n
    ",prf*10**6)
18 printf("(b) Duty Cycle = %.2f percent \n",duty)

```

Scilab code Exa 24.6 Example 6

```
1 //Chapter 24, Example 24.6
2 clc
3 //Initialisation
4 t1=12*10**-6 //time period in ms
5 t2=2*10**-6 //time period in ms
6 v1=8*10**-3 //voltage
7 v2=4*10**-6 //voltage
8 v3=2*10**-3 //voltage
9 v4=6*10**-6 //voltage
10
11
12
13
14 //Calculation
15 T=t1-t2 //timeperiod in microsec
16 g=((v1*v2)+(v3*v4))/T //Average Value in Volts
17
18 //Result
19 printf("Average Value , G = %.1 f mV \n",g*10**3)
```

Scilab code Exa 24.7 Example 7

```
1 //Chapter 24, Example 24.7
2 clc
3 //Initialisation
4 duty = 0.28 //duty cycle
5 Vp = 7 //peak value
6 Vb = -3 //base-line voltage
7 t1=0.28 //time period
8 t2=0.72 //time period
9
10
11
```

```

12 // Calculation
13 Vav = (duty*Vp)+((1-duty)*Vb) //Average Value in
    Volts
14 g=(Vp*t1)+(Vb*t2) //Average Value
    in Volts
15
16 //Result
17 printf("(a) Average Value, Vav = %.1 f V \n",Vav)
18 printf("(c) G = %.1 f V \n",g)

```

Scilab code Exa 24.8 Example 8

```

1 //Chapter 24, Example 24.8
2 clc
3 //Initialisation
4 vi=2 //voltage in mV
5 e=8 //voltage in mV
6 r=100*10**3 //resistance in ohm
7 c=10**-6 //capacitance in farad
8
9
10 //Calculation
11 tau=r*c //time constant
12 vr=e-vi //voltage
13 irm=vr/r //current in amperes
14
15 //Result
16 printf("(a) Vc = %d V %d V e^(-t/%.f ms) \n",e,(vi-e
    ),tau*1000)
17 printf("(b) Irmax = %.2 f mA e^(-t/%.f ms)",irm*1000,
    tau*1000)
18
19
20 //Graph
21 x1=0:0.1:0.8

```

```

22 vc=e+(vi-e)*exp(-x1/tau)
23 ic=irm*exp(-x1/tau)
24 scf(221)
25 plot(x1,vc)
26 xlabel("t(s)")
27 ylabel("vc(V)")
28 xtitle("Vc for the network of Fig. 24.20.")
29 scf(222)
30 plot(x1,ic)
31 xlabel("t(s)")
32 ylabel("ic (mA)")
33 xtitle("ic for the network of Fig. 24.20.")

```

Scilab code Exa 24.9 Example 9

```

1 //Chapter 24, Example 24.9
2 clc
3 //Initialisation
4 vi=-4 //voltage in mV
5 vf=10 //voltage in mV
6 r=1*10**3 //resistance in ohm
7 c=0.01*10**-6 //capacitance in farad
8
9
10 //Calculation
11 tau=r*c //time constant
12 t=-tau*log(vf/(vf-vi)) //time in seconds
13
14 //Result
15 printf("Vc = %d mV %d mV e^(-t/%.f ms) \n",vf,(vi-vf
),tau*10**6)
16 printf("t = %.2f us", (t*10**6))
17
18 //Graph
19 x1=0:(10**-6):(80*10**-6)

```

```

20 vc=vf+(vi-vf)*exp(-x1/tau)
21 plot(x1,vc)
22 xlabel("t(s)")
23 ylabel("vc(V)")
24 xtitle("Vc for the network of Fig. 24.22.")

```

Scilab code Exa 24.10 Example 10

```

1 //Chapter 24, Example 24.10
2 clc
3 //Initialisation
4 f=1000 //frequency in hertz
5 r=5*10**3 //resistance in ohm
6 c=0.01*10**-6 //capacitance in farad
7 vi=0 //for charging phasse,
    voltage
8 vf=10 //voltage in mV
9 vi2=10 //for discharging phasse,
    voltage in mV
10 vf2=0 //voltage
11 t1=0 //for charging phase for
    ic
12 vr=vf
13
14
15 //Calculation
16 t=1/f //time period in sec
17 tp=t/2 //phase time in sec
18 tau=r*c //time constant in sec
19 irmax=vr/r //current in milliampere
20
21 //Result
22 printf("(a) tp = %.1 f ms \n",tp*10**3)
23 printf("tau = %.2 f ms \n",tau*10**3)
24 printf("Therefore , tp = 10*tau = T/2 \n")

```

```

25 printf("(b) for charging phase, Vc = %d mV(1 - e^(-t
    /%.2f ms)) \n",vf,tau*1000)
26 printf(" for discharging phase, Vc = %d mV(e^(-t/%
    .2f ms)) \n",vf,tau*1000)
27 printf("(c) Imax = %.2f uA e^(-t/%.f ms)",irmax
    *10**3,tau*1000)
28
29
30 //Graph
31 x1=0:(0.1*10**-3):(t)
32 x2=tp:(0.1*10**-3):(t)
33 x3=0:(0.1*10**-3):(tp)
34 x4=tp:(0.1*10**-3):(t)
35 vc=vf+(vi-vf)*(exp(-x1/tau)) //for Vc -
    Charging phase
36 vc2=vf2+(vi2-vf2)*exp(-x2/tau) //for Vc
    - Discharging phase
37 ic=irmax*exp(-x3/tau) //for Ic -
    Charging phase
38 ic2=-irmax*(exp(-x3/tau)) //for Ic -
    Discharging phase
39 subplot(221)
40 plot(x1,vc)
41 xlabel("t(s)")
42 ylabel("Vc(V)")
43 xtitle("Vc for the R-C network - Charging Phase")
44 subplot(222)
45 plot(x2,vc2)
46 xlabel("t(s)")
47 ylabel("Vc (V)")
48 xtitle("Vc for the R-C network - Discharging Phase")
49 subplot(223)
50 plot(x3,ic)
51 xlabel("t(s)")
52 ylabel("ic (mA)")
53 xtitle("Ic for the R-C network - Charging Phase")
54 subplot(224)
55 plot(x4,ic2)

```



```

56 xlabel("t(s)")
57 ylabel("ic (mA)")
58 xtitle("Ic for the R-C network - Discharging Phase")

```

Scilab code Exa 24.11 Example 11

```

1 //Chapter 24, Example 24.11
2 clc
3 //Initialisation
4 f=10*10**3 //frequency in hertz
5 r=5*10**3 //resistance in ohm
6 c=0.01*10**-6 //capacitance in farad
7 vi=0 //for charging phase,
// voltage
8 vf=10 //voltage in mV
9 vf2=0 //voltage
10 t1=0 //for charging phase for
// ic
11 vr=vf
12
13
14 //Calculation
15 t=1/f //time period in sec
16 tp=t/2 //phase time in sec
17 tau=r*c //time constant in sec
18 irmax=vr/r //current in milliampere
19
20 //first interval
21 vc1=vf+(vi-vf)*(exp(-tau/tau)) //in
// voltage
22 vi1=vc1 //in voltage
23 vc2=vc1*exp(-tau/tau) //in voltage
24 irmax2=(vi-vc1)/r //in
// milliamp
25

```

```

26     //for graph
27     x1=0:(0.01*10**-3):(tp)
28     x2=0:(0.01*10**-3):(tp)
29     vcc=vf+((vi-vf)*(exp(-x1/tau)))           //
        for vc - Charging phase
30     vcd=vc1*(exp(-x2/tau))                   //
        for vc - Discharging phase
31
32     icc=(irmax*10**-3)*exp(-x1/tau)
        //for Ic - Charging phase
33     icd=(irmax2*10**-3)*(exp(-x2/tau))
        //for Ic - Discharging phase
34     scf(1)
35     subplot(221)
36     plot(x1,vcc)
37     xlabel("t(s)")
38     ylabel("Vc(V)")
39     xtitle("Vc for the R-C network - Charging Phase"
        )
40     subplot(222)
41     plot(x2,vcd)
42     xlabel("t(s)")
43     ylabel("Vc (V)")
44     xtitle("Vc for the R-C network - Discharging
        Phase")
45     subplot(223)
46     plot(x1,icc)
47     xlabel("t(s)")
48     ylabel("ic (mA)")
49     xtitle("Ic for the R-C network - Charging Phase"
        )
50     subplot(224)
51     plot(x2,icd)
52     xlabel("t(s)")
53     ylabel("ic (mA)")
54     xtitle("Ic for the R-C network - Discharging
        Phase")
55

```

```

56 //second interval
57     vc1=vf+(vc2-vf)*(exp(-tau/tau))
        //in voltage
58     vi1=vc1
        //in voltage
59     vc2=vc1*exp(-tau/tau)
        //in voltage
60     irmax2=(vi-vc1)/r
        //in milliamp
61     x1=0:(0.01*10**-3):(tp)
62     x2=0:(0.01*10**-3):(tp)
63     vcc2=vf+((vc2-vf)*(exp(-x1/tau)))
        //for vc - Charging phase
64     vcd2=vc1*(exp(-x2/tau))
        //for vc - Discharging phase
65     icc=(irmax*10**-3)*exp(-x1/tau)
        //for Ic - Charging phase
66     icd=(irmax2*10**-3)*(exp(-x2/tau))
        //for Ic - Discharging phase
67     scf(2)
68     subplot(221)
69     plot(x1,vcc2)
70     xlabel("t(s)")
71     ylabel("Vc(V)")
72     xtitle("Vc for the R-C network - Charging Phase"
73           )
74     subplot(222)
75     plot(x2,vcd2)
76     xlabel("t(s)")
77     ylabel("Vc (V)")
78     xtitle("Vc for the R-C network - Discharging
79           Phase")
80     subplot(223)
81     plot(x1,icc)
82     xlabel("t(s)")
83     ylabel("ic (mA)")
84     xtitle("Ic for the R-C network - Charging Phase"
85           )

```

```
83     subplot(224)
84     plot(x2,icd)
85     xlabel("t(s)")
86     ylabel("ic (mA)")
87     xtitle("Ic for the R-C network - Discharging  
Phase")
```

Chapter 25

Nonsinusoidal Circuits

Scilab code Exa 25.2 Example 2

```
1 //Chapter 25, Example 25.2
2 clc
3 //Initialisation
4 A0=20
5 A10=0
6 A11=5*10**-3
7 A12=0
8 A13=0
9 A20=8
10 A21=0
11 B1=12
12 B2=0
13
14 //Calculation
15 v=A0 //from graph 25.12
16 i=A10+A11+A12+A13 //fourier series
17 v2=A20+A21 //fourier series
18 B=B1+B2 //fourier series
19
20 //Result
21 printf("(a) v = %d \n",v)
```

```

22 printf("(b) i = %d X 10^-3 sin wt \n",i*10**3)
23 printf("(c) v = %d + %d cos wt",v2,B)

```

Scilab code Exa 25.3 Example 3

```

1 //Chapter 25, Example 25.3
2 clc
3 //Initialisation
4 funcprot();
5 //Polar to Rectangle Conversion
6 function [r,i]= polar2rect(x,y)
7     r=x*cosd(y)
8     i=x*sind(y)
9 endfunction
10
11 //Rectangle to Polar Conversion
12 function [r,th] = rect2polar(x,y)
13     r=sqrt((x**2)+(y**2))
14     th=atand(y/x)
15 endfunction
16
17 //According to phasor algebra
18 a1=2
19 [r1,i1]=polar2rect(1,90)
20 [r2,i2]=polar2rect(2,0)
21
22 //Calculation
23 ar1=r1+r2
24 ai1=i1+i2
25 [ar2,ath2] = rect2polar(ar1,ai1)
26 x = 0 : 0.1 : 10
27 y = a1 + (ar2*sin(x+ath2*3.14/180))
28
29
30 //Result

```

```

31 plot(x, y)
32 xlabel("a = wt");
33 ylabel("V");
34 title('v = 2 + 1 cos a + 2 sin a');

```

Scilab code Exa 25.4 Example 4

```

1 //Chapter 25, Example 25.4
2 clc
3 //Initialisation
4 funcprot();
5 //Polar to Rectangle Conversion
6 function [r,i]= polar2rect(x,y)
7     r=x*cosd(y)
8     i=x*sind(y)
9 endfunction
10
11 //Rectangle to Polar Conversion
12 function [r,th] = rect2polar(x,y)
13     r=sqrt((x**2)+(y**2))
14     th=atand(y/x)
15 endfunction
16
17 //According to phasor algebra
18 [r1,i1]=polar2rect(1,0)
19 [r2,i2]=polar2rect(1,0)
20
21 //Calculation
22 ar1=r1+r2
23 ai1=i1+i2
24 [ar2,ath2] = rect2polar(ar1,ai1)
25 x = 0 : 0.1 : 10
26 y = ar2*sin(x+ath2*3.14/180)
27
28

```

```

29 //Result
30 plot(x, y)
31 xlabel("a = wt");
32 ylabel("V");
33 title('v = 2 + 1 cos a + 2 sin a');

```

Scilab code Exa 25.5 Example 5

```

1 //Chapter 25, Example 25.5
2 clc
3 //Initialisation
4 v=4 //in voltage
5 v2=6 //in voltage
6
7 //Calculation
8 x = 0 : 0.1 : 10
9 y = v + v2*sin(x)
10 vrmax=sqrt((v**2)+(v2**2)/2)
11
12 //Result
13 plot(x, y)
14 xlabel("wt");
15 ylabel("V");
16 title('v = 4 + 6 sin wt');
17 printf("(b) Vrmax = %.3f V",vrmax)

```

Scilab code Exa 25.6 Example 6

```

1 //Chapter 25, Example 25.6
2 clc
3 //Initialisation
4 v=20 //in voltage
5

```



```

6 // Calculation
7 v0=(4/%pi)*v
8 v1=v0
9 v2=(v0/3)
10 v3=(v0/5)
11 v4=(v0/7)
12 v5=(v0/9)
13 v6=(v0/11)
14 vrms=sqrt(((v1**2)+(v2**2)+(v3**2)+(v4**2)+(v5**2)+(
      v6**2))/2)
15
16 // Result
17 printf("(b) Vmax = %.3f V",vrms)

```

Scilab code Exa 25.7 Example 7

```

1 //Chapter 25, Example 25.7
2 clc
3 //Initialisation
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)
6     r=sqrt((x**2)+(y**2))
7     th=atand(y/x)
8 endfunction
9
10 i1=0
11 R=3
12 vc=12
13 C=(1/8)
14 E=10/sqrt(2)
15
16 // Calculation
17 vr=i1*R
18 xc=1/(2*C)
19

```

```

20 z=complex(R,xc)
21 [r1,th1]=rect2polar(real(z),-imag(z))
22 i2=E/r1
23 vr2=i2*R
24 vc1=i2*xc
25 vca=-th1-90
26
27 irms = sqrt((i2**2))
28 vrms = sqrt((vr2**2))
29 vcrms = sqrt((vc**2)+(vc1**2))
30
31 P=(irms**2)*R
32 //Result
33 printf("(a) 1. VR = %d V \t\n \tI = %d \t\n\tVc = %d
          V\t\n",vr,i1,vc)
34 printf("      2. for AC supply \n")
35 printf("          i = %.2f A < %.2f \n",i2,-th1)
36 printf("          Vr = %.2f A < %.2f \n",vr2,-th1)
37 printf("          Vc = %.2f A < %.2f \n",vc1,vca)
38 printf("(b) Irms = %.3f A \n\tVrms = %.3f V \n\
          tVcrms = %.3f V\t\n",irms,vrms,vcrms)
39 printf("(c) P = %d W \n",P)

```

Scilab code Exa 25.8 Example 8

```

1 //Chapter 25, Example 25.8
2 clc
3 //Initialisation
4 //Rectangle to Polar Conversion
5 function [r,th] = rect2polar(x,y)
6     r=sqrt((x**2)+(y**2))
7     th=atand(y/x)
8 endfunction
9
10 x1=0

```

```

11 R=6
12 e0=63.6
13 w=377
14 L=0.1
15 e1=70.71
16 e2=29.98
17 e2a=-90
18 w2=754
19
20 // Calculation
21 zt=R
22 i0=e0/R
23 vr0=i0*R
24 vl0=0
25 p0=(i0**2)*R
26
27 x11=w*L
28 zt1=complex(R,x11)
29 [zt1r,zt1a]=rect2polar(real(zt1),imag(zt1))
30 i1=e1/zt1r
31 vr1=i1*R
32 vl1=i1*x11
33 vl1a=90-zt1a
34 p1=(i1**2)*R
35
36
37 x12=w2*L
38 zt2=complex(R,x12)
39 [zt2r,zt2a]=rect2polar(real(zt2),imag(zt2))
40 i2=e2/zt2r
41 i2a=e2a-zt2a
42 vr2=i2*R
43 vl2=i2*x12
44 vl2a=90+i2a
45 p2=(i2**2)*R
46 Irms=sqrt((i0**2)+(i1**2)+(i2**2))
47 VRms=sqrt((e0**2)+(vr1**2)+(vr2**2))
48 Vlrms=sqrt((vl1**2)+(vl2**2))

```

```

49 //Result
50 printf("The Fourier series expansion for i is \n")
51 printf("i = %.1f + sqrt(2)(%.2f)sin(%d*t - %.2f) +
      sqrt(2)(%.3f)sin(%d*t - %.2f) \n",i0,i1,w,zt1a,i2
      ,w2,-i2a)
52 printf("Irms = %.2f A \n\n",Irms)
53 printf("The Fourier series expansion for VR is \n")
54 printf("Vr = %.1f + sqrt(2)(%.2f)sin(%d*t - %.2f) +
      sqrt(2)(%.3f)sin(%d*t - %.2f) \n",e0,vr1,w,zt1a,
      vr2,w2,-i2a)
55 printf("Vrms = %.2f V \n\n",VRms)
56 printf("The Fourier series expansion for VL is \n")
57 printf("VL = sqrt(2)(%.2f)sin(%d*t - %.2f) + sqrt(2)
      (%.3f)sin(%d*t - %.2f) \n",v11,w,v11a,v12,w2,-
      v12a)
58 printf("Vlrms = %.2f V \n\n",Vlrms)

```

Chapter 26

System Analysis An Introduction

Scilab code Exa 26.1 Example 1

```
1 //Chapter 26, Example 26.1, page 1153
2 clc
3 //Initialisation
4 eg=100*10**-3 //in Volt
5 ei=96*10**-3 //in Volt
6 rz=100 //in Ohms
7
8 //Calculation
9 vr=eg-ei //in Volt
10 i=vr/rz //in Ampere
11 zi=ei/i //in Ohms
12
13 //Result
14 printf("Input Impedance , Zi = %.1 f kOhm" ,zi/1000)
```

Scilab code Exa 26.2 Example 2

```

1 //Chapter 26, Example 26.2, page 1154
2 clc
3 //Initialisation
4 eg=2 //in Volt
5 eo=1.92 //in Volt
6 rz=2*10**3 //in Ohms
7
8 //Calculation
9 vr=eg-eo //in Volt
10 i=vr/rz //in Ampere
11 zo=eo/i //in Ohms
12
13 //Result
14 printf("Output Impedance , Zi = %.1 f kOhm" ,zo/1000)

```

Scilab code Exa 26.3 Example 3

```

1 //Chapter 26, Example 26.3, page 1154
2 clc
3 //Initialisation
4 Vr=2*10**-3 //in Volt
5 Rz=10 //in Ohms
6 Ei=50*10**-3 //in Volt
7 phi1=150 //in degree
8
9 //Calculation
10
11 i=Vr/Rz //in Ampere
12 zi=Ei/i //Zi in Ohms
13 phi2=180-phi1 //phase angle between Eg
    and Vr
14 re=zi*cos(phi2*3.14/180) //real number,
    R
15 im=zi*sin(phi2*3.14/180) //Imaginary
    number, Xl

```

```

16
17 //Result
18 printf("Zi = %.2f ohm + j %.1f ohm ",re,im)

```

Scilab code Exa 26.4 Example 4

```

1 //Chapter 26, Example 26.4, page 1157
2 clc
3 //Initialisation
4 Eo=-20 //output voltage
5 Ei=4*10**-3 //input voltage
6 Rl=2.2*10**3 //load Resistance in Ohm
7 Ro=50*10**3 //output Resistance in Ohm
8 Zi=1*1*10**3 //input impedance in Ohm
9 Rg=1*10**3 //Resistance in Ohm
10
11
12 //Calculation
13 Avn=Eo/Ei //no-load voltage gain
14 Av=Avn*Rl/(Rl+Ro) //loaded voltage gain
15 Avr=Av*Zi/(Zi+Rg) //loaded voltage gain
16 Ro=Rl*((Avn/Av)-1) //Resistance in Ohm
17
18 //Result
19 printf("(a) Avnl = %d \n",Avn)
20 printf("(b) Av = %.2f \n",Av)
21 printf("(c) Avr = %.2f \n",Avr)
22 printf("(d) Ro = %d kOhm",Ro/1000)

```

Scilab code Exa 26.5 Example 5

```

1 //Chapter 26, Example 26.5, page 1161
2 clc

```

```

3 //Initialisation
4 Rl=4.7*10**3 //load Resistance in Ohm
5 Ro=40*10**3 //output Resistance in Ohm
6 Avn=-960 //no-load voltage gain
7 Ri=2.7*10**3 //input Resistance in Ohm
8
9
10
11 //Calculation
12 Av=Avn*Rl/(Rl+Ro) //loaded voltage gain
13 Ai=(-Avn)*Ri/(Rl+Ro) //voltage gain, Ai
14 Rl2=2*Rl //Resistance in Ohm
15 Av2=Avn*Rl2/(Rl2+Ro) //loaded voltage gain
16 Ai2=(-Avn)*Ri/(Rl2+Ro) //voltage gain, Ai
17 Air=(-Av)*Ri/Rl //voltage gain, Air
18 Ag=(Av**2)*Ri/Rl //voltage gain, Ag
19 Ai2=-Ag/Av //voltage gain, Ai
20
21 //Result
22 printf("(a) Av = %.2 f \n",Av)
23 printf("(b) Avi = %.2 f \n",Ai)
24 printf("(c) Av = %.2 f, Ai = %.2 f\n",Av2,Ai2)
25 printf("(d) Air = %.2 f \n",Air)
26 printf("(e) Ag = %.2 f \n",Ag)
27 printf("(f) Ai = %.2 f",Ai2)

```

Scilab code Exa 26.6 Example 6

```

1 //Chapter 26, Example 26.6, page 1163
2 clc
3 //Initialisation
4 Avn1=1 //no-load voltage gain
5 Avn2=-600 //no-load voltage gain
6 Avn3=-1200 //no-load voltage gain
7 Zi1=50*10**3 //input impedance in Ohm

```



```

8 Zi2=1.8*10**3           //input impedance in Ohm
9 Zi3=1.2*10**3           //input impedance in Ohm
10 Rl=3.3*10**3           //input impedance in Ohm
11 Ro=25                   //output Resistance in Ohm
12 Ro2=40*10**3           //output Resistance in Ohm
13 Ro3=50*10**3           //output Resistance in Ohm
14
15
16 // Calculation
17 Av1=Avn1*Zi2/(Zi2+Ro)   //loaded voltage gain
18 Av2=Avn2*Zi3/(Zi3+Ro2) //loaded voltage gain
19 Av3=Avn3*Rl/(Rl+Ro3)   //loaded voltage gain
20 Ai=(-Avn1)*Zi1/(Zi2+Ro) //voltage gain, Ai
21 Ai2=(-Avn2)*Zi2/(Zi3+Ro2) //voltage gain,
    Ai
22 Ai3=(-Avn3)*Zi3/(Rl+Ro3) //voltage gain, Ai
23 Rl2=2*Rl                //Resistance in Ohm
24 Av21=Avn1*Rl2/(Rl2+Ro) //loaded voltage
    gain
25 Ai20=(-Avn1)*Zi1/(Zi2+Ro) //voltage gain,
    Ai
26 Air=(-Avn1)*Zi1/Rl      //voltage gain,
    Air
27 Ag=(Avn1**2)*Zi1/Rl    //voltage gain,
    Ag
28 Ai21=-Ag/Avn1          //voltage gain, Ai
29 Avt=Av1*Av2*Av3        //voltage gain
30 Ait=Ai*Ai2*Ai3         //voltage gain
31 Agt=-Avt*Ait           //voltage gain
32 Av2=Av1*Av2            //voltage gain
33 Ai22=Ai*Ai2            //voltage gain
34 Ai23=Av2*Zi1/Zi3       //voltage gain
35 Ag2=Av2*Ai23           //voltage gain
36 Ag22=(Av2**2)*(Zi1/Zi3) //voltage
    gain
37 Avt2=Avn1*Avn2*Avn3   //voltage gain
38
39

```

```

40 //Result
41 printf("(a) Av1 = %.3f, Av2 = %.3f, Av3 = %.3f \n
      Ai1 = %.3f, Ai2 = %.3f, Ai3 = %.3f \n",Av1,Av2,
      Av3,Ai,Ai2,Ai3)
42 printf("(b) Avt = %.2f, Ait = %.2f \n",Avt,Ait)
43 printf("(c) Agt = %.2f x 10^6 \n",Agt/10**6)
44 printf("(d) Av2 = %.2f, Ai2 = %.3f \n",Av2,Ai22)
45 printf("(e) Ai2 = %.2f \n",Ai23)
46 printf("(f) Ag2 = %.2f \n",Ag2)
47 printf("(g) Ag2 = %.2f \n",Ag22)
48 printf("(h) Avt = %.1f x 10^5",Avt2/(10**5))

```

Scilab code Exa 26.7 Example 7

```

1 //Chapter 26, Example 26.7, page 1166
2 clc
3 funcprot(0);
4 // A = p2z(R,Theta) - Convert from polar to
      rectangular form.
5 // R is a matrix containing the magnitudes
6 // Theta is a matrix containing the phase angles
      (in degrees).
7 function [A] = p2z(R,Theta)
8 A = R*exp(%i*pi*Theta/180);
9 endfunction
10
11 //Initialisation
12 z1 = p2z(3,0)
13 z2 = p2z(5,90)
14 z3 = p2z(4,-90)
15
16
17 //Calculation
18 z11 = z1+z3
19 z12 = z3

```

```

20 z22 = z2+z3
21
22 // Result
23 printf("Z11 = %d ohm - j %d ohm \n",real(z11),-imag(
    z11))
24 printf("Z12 = - j %d ohm \n",-imag(z12))
25 printf("Z22 = j %d ohm \n",imag(z22))

```

Scilab code Exa 26.9 Example 9

```

1 //Chapter 26, Example 26.9, page 1171
2 clc
3 funcprot(0);
4 // A = p2z(R,Theta) - Convert from polar to
    rectangular form.
5 // R is a matrix containing the magnitudes
6 // Theta is a matrix containing the phase angles
    (in degrees).
7 function [A] = p2z(R,Theta)
8 A = R*exp(%i*pi*Theta/180);
9 endfunction
10
11 //Initialisation
12 Y1 = p2z(0.2,0) //admittance
13 Y2 = p2z(0.02,-90) //admittance
14 Y3 = p2z(0.25,90) //admittance
15
16
17 //Calculation
18 y11 = Y1+Y2 //admittance
19 y12 = -Y2 //admittance
20 y22 = Y2+Y3 //admittance
21
22 //Result
23 printf("Y11 = %.1 f mS - j %.2 f mS \n",real(y11),-

```

```

    imag(y11))
24 printf("Y12 = j %.2 f mS \n", imag(y12))
25 printf("Y22 = j %.2 f mS \n", imag(y22))

```

Scilab code Exa 26.11 Example 11

```

1 //Chapter 26, Example 26.11, page 1177
2 clc
3 //Initialisation
4 hf=50 //short circuit current gain
5 ho=25*10**-6 //Output Admittance with input
   open circuited
6 zl=2*10**3 //load impedance
7 hi=1*10**3 //input impedance with output
   shorted
8 hr=4*10**-4 //reverse voltage amplification
   with input open circuited
9
10 //Calculation
11 Ai=hf/(1+(ho*zl)) //current
   gain
12 Av=-hf*zl/((hi*(1+(ho*zl)))-(hr*hf*zl)) //voltage
   gain
13
14 //Result
15 printf("Current gain , Ai = %.2 f \n",Ai)
16 printf("Voltage gain , Av = %ds \n",Av)

```

Scilab code Exa 26.12 Example 12

```

1 //Chapter 12, Example 12.12
2 clc
3 //Variable Declaration

```

```
4 r1 = 3 //resistance1
5 l = 6*10^-3 //inductance in mH
6 e = 15 //voltage in volts
7 r2 = 2 //resistance2
8 //Calculation
9 im = e/(r1+r2) //inductance current
10 ws = (l*im*im)/2 //energy stored
11
12 //Results
13 printf("Ws = %d mJ",ws*10^3)
```
