

Scilab Textbook Companion for
Power Electronics
by P. S. Bimbira¹

Created by
Navdeep Kumar
B.E.
Electrical Engineering
Thapar University
College Teacher
Dr. Sunil Kumar Singla
Cross-Checked by
Lavitha Pereira

September 16, 2013

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: Power Electronics

Author: P. S. Bimbra

Publisher: Khanna Publishers, New Delhi

Edition: 5

Year: 2012

ISBN: 978-81-7409-279-3

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
2 Power Semiconductor Diodes and Transistors	14
3 Diode Circuits and Rectifiers	18
4 Thyristors	31
5 Thyristor Commutation Techniques	45
6 Phase Controlled Rectifiers	51
7 Choppers	74
8 Inverters	89
9 AC Voltage Controllers	104
10 Cycloconverters	110
11 Some Applications	113
12 Electric Drives	117
13 Power Factor Improvement	147
14 Miscellaneous Topics	152

List of Scilab Codes

Exa 2.1	to find switching freq of the transistor	14
Exa 2.2	to determine avg power loss due to collector current and also to find peak instantaneous power loss due to collector current	15
Exa 2.3	to obtain power loss during turn off time and turn off period and also instant power loss during fall time	16
Exa 2.4	to find switching freq of the transistor	17
Exa 3.2	to find the conduction time of diode peak current through diode and final steady state voltage	18
Exa 3.6	to determine conduction time of diode and rate of change of current	18
Exa 3.7	to find the time required to deliver a charge of 200Ah	19
Exa 3.8	to calculate the power delivered to the heater and find peak diode current and input pf	19
Exa 3.9	to find 1 the value of avg charging current 2 power supplied to battery and that dissipated in the resistor 3 supply pf 4 charging time 5 rectifier efficiency and PIV of diode	20
Exa 3.10	to determine the effect of reverse recovery time on the avg output voltage	21
Exa 3.11	to determine 1 avg value of output voltage and output current 2 avg and rms values of diode currents 3 rms values of output and input currents and supply pf	21
Exa 3.12	to calculate 1 peak load current 2 dc load current 3 dc diode current 4 percentage regulation from no load to full load	22
Exa 3.13	to design a zener voltage regulator to meet given specifications	23

Exa 3.14	to find R1 and R2	24
Exa 3.15	to find the VA rating of the transformer	24
Exa 3.16	to determine 1 avg value of output voltage 2 input current distortion factor 3 input displacement factor 4 input pf 5 input current harmonic factor and 6 crest ratio	25
Exa 3.17	to determine diode rating and transformer rating	25
Exa 3.18	to calculate avg value of output voltage avg and rms values of diode current and power delivered to diode	26
Exa 3.19	determine the ratings of diodes and of 3ph deltastar transformer	27
Exa 3.20	to determine 1 power delivered to battery and load 2 input displacement factor 3 current distortion factor 4 input pf 5 input HF 6 transformer rating	28
Exa 3.21	design capacitor filter and avg value of output voltage with and without filter	29
Exa 3.22	to find the value of L with different R and CRF without L	29
Exa 3.23	to design LC filter	30
Exa 4.2	to plot allowable gate voltage as the function of gate current and to find avg gate power dissipation	31
Exa 4.3	to compute gate source resistance	31
Exa 4.4	to compute gate source resistance trigger voltage and trigger current	32
Exa 4.5	to compute 1 resistance 2 triggering freq 3 duty cycle of the triggering pulse	33
Exa 4.6	to compute min width of gate pulse current	33
Exa 4.8	to calculate trigger voltage and trigger current	34
Exa 4.9	to compute avg on current rating for half sine wave current for various conduction angles	34
Exa 4.10	to compute avg on current rating for half sine wave current for various conduction angles	35
Exa 4.11	to calculate surge current rating and power rating	36
Exa 4.12	to find maximum value of the remedial parameter	36
Exa 4.16	to calculate fault clearance time	37
Exa 4.17	1 to calculate the max values of change in current and change in voltage for SCR 2 rms and avg current rating and 3 suitable voltage rating	37

Exa 4.19	1 to check heat sink selection is satisfactory 2 to choose heat sink for given requirements and ckt efficiency 3 compute case and junction temp	38
Exa 4.20	to find total avg power loss and percentage inc in device rating	39
Exa 4.21	to determine voltage across each SCR and discharge current across each capacitor	40
Exa 4.22	to calculate number of series and parrallel units of SCRs	41
Exa 4.23	to calculate the resistance	42
Exa 4.25	to compute value of various resistances	42
Exa 4.26	to compute max and min values of R and the corresponding freq	43
Exa 4.27	to find max and min firing angles for triac	43
Exa 5.1	to determine 1 conduction time and 2 voltage across thyristor	45
Exa 5.2	to calculate 1 conduction time for auxillery thyristor 2 voltage across and 3 circuit turn off time for main thyristor	45
Exa 5.3	to determine 1 peak value of current 2 value of capacitor C	46
Exa 5.4	to calculate 1 value of current 2 circuit turn off time for main and auxillery thyristor	47
Exa 5.5	to compute min value of C	47
Exa 5.6	to find circuit turn off time	48
Exa 5.7	to find conduction time of thyristor	48
Exa 5.8	1 to calculate value of Capacitor 2 determine value of Resistance	48
Exa 5.9	calculate 1 time at which commutation of main thyristor gets initiated 2 ckt turn off time	49
Exa 5.11	to find current in R and L	49
Exa 5.12	To find the current in R and L and voltage across C .	50
Exa 6.1	to calculate the power absorbed in the heater element	51
Exa 6.2	1 to find value of charging current 2 power supplied to battery and dissipated by resistor 3 calculate the supply pf	51
Exa 6.3	1 to find value of charging current 2 power supplied to battery and dissipated by resistor 3 calculate the supply pf	52

Exa 6.4	to find ckt turn off time avg output voltage and avg load current	53
Exa 6.5	to determine 1 rectification efficiency 2 form factor 3 voltage ripple factor 4 transformer utilisation factor 5 PIV of thyristor	54
Exa 6.6	to find power handled by mid pt convertor and single phase bridge convertor	55
Exa 6.7	compute firing angle delay and pf	55
Exa 6.9	to find avg output current and power delivered	56
Exa 6.10	to find avg value of load current and new value under given changed conditions	56
Exa 6.11	to calculate the input and output performance parameters for full conductor	57
Exa 6.12	to calculate the input and output performance parameters for single phase semi conductor	58
Exa 6.13	determine 1 firing angle 2 avg and rms values of load current 3 rectification efficiency	59
Exa 6.15	to calculate 1 avg value of load volatage 2 avg and rms current and PIV 3 avg power dissipation in each thyristor	59
Exa 6.17	to compute firing angle delay and supply pf	60
Exa 6.18	to find commutation time and reverse voltage across SCR	61
Exa 6.19	to find the magnitude of per phase input supply voltage	61
Exa 6.20	to find magnitude of input per phase supply voltage	62
Exa 6.21	to find magnitude of input per phase supply voltage	62
Exa 6.22	to compute firing angle delay and supply pf	63
Exa 6.23	to calculate rectification efficiency TUF and input power factor	63
Exa 6.24	to find DF CDF THD and pf and to calculate the active and reative input powers	64
Exa 6.25	calculate the power delivered to load and input pf	65
Exa 6.26	calculate overlap angle for different firing angles	66
Exa 6.28	to calculate firing angle delay and overlap angle	67
Exa 6.29	to calculate firing angle firing angle delay and overlap angle	67
Exa 6.31	calculate the peak value of circulating current	68
Exa 6.32	to determine 1 avg output voltage 2 avg output current 3 avg and rms values of thyristor currents 4 pf	68
Exa 6.33	to determine 1 avg output voltage 2 angle of overlap 3 pf	69

Exa 6.34	calculate the generator mean voltage	70
Exa 6.35	find the mean value of E	70
Exa 6.36	to find avg current through battery	71
Exa 6.37	to determine 1 avg output voltage 2 avg output current 3 avg and rms values of thyristor currents 4 avg and rms values of diodes currents 5 i pf 6 ckt turn off time	71
Exa 6.38	to calculate peak value of circulating currents and of both convertors	72
Exa 6.39	to estimate triggering angle for no current transients and for worst transients	73
Exa 7.2	to calculate 1 avg and rms values of output voltage 2 chopper efficiency	74
Exa 7.3	to compute pulse width of output voltage and avg value of new output voltage	74
Exa 7.4	to find time ratio for chopper	75
Exa 7.5	to compute pulse width of output voltage and avg value of new output voltage	75
Exa 7.11	1 to find whether load current is cont 2 calculate value of avg op current 3 to find its max and min values 4 find rms values of various harmonics 5 compute avg value of supply current 6 ip power power absorbed by load counter emf and power loss in res	76
Exa 7.12	1 to find whether load current is cont or not 2 to calculate avg output voltage and avg current 3 to compute max and min values of output current	77
Exa 7.13	to find the chopping freq	78
Exa 7.14	to find the value of external inductance to be added in series	79
Exa 7.15	1 to calculate min and max value of load current 2 max value of ripple current 3 avg and rms values of load current 4 rms value of chopper current	79
Exa 7.17	to find the time for which current flows	80
Exa 7.18	to calculate values of commutating capacitor and commutating inductor	81
Exa 7.19	to calculate the value of commutating component C and commutating component L	81

Exa 7.20	to compute 1 effective on period 2 peak current through main and auxillary thyristor 3 their turn off times 4 total commutation intervals 5 capacitor voltage 6 time needed to recharge the capacitor	82
Exa 7.21	to calculate values of 1 commutating components C and L and 2 min and max output voltages	83
Exa 7.22	calculate 1 value of commutating inductor and capacitor 2 max capacitor voltage 3 peak commutating current	84
Exa 7.23	to compute 1 turn off time of main thyristor 2 total commutation interval 3 turn off time of auxillary thyristor	84
Exa 7.24	to calculate min and max value of load current	85
Exa 7.27	calculate 1 range of speed control 2 range of duty cycle	86
Exa 7.28	determine chopping freq and duty cycle	86
Exa 7.29	to determine the higher limit of current pulsation chopping freq and duty cycle ratio	87
Exa 8.3	to find the value of C for having load commutation	89
Exa 8.4	to find the power delivered	89
Exa 8.5	to find the power delivered to the load	90
Exa 8.6	to find rms value of thyristor and diode currents	90
Exa 8.7	to find 1 rms value of fundamental load current 2 power absorbed by the load and fundamental power 3 rms and peak currents of each thyristor 4 conduction of thyristors and diodes	91
Exa 8.8	to determine 1 fundamental rms output voltage 2 total output power and fundamenetal freq power 3 avg and peak currents of each thyristor 4 input pf 5 distortion factor and THD and 6 harmonic factor foe lowest harmonic	92
Exa 8.9	to calculate 1 rms value of output voltage and fundamental component of output voltage 2 output power 3 fundamental freq output power 4 avg and peak currents of each transistor 5 peak reverse blocking voltage 6 harmonic factor for third harmonic 7 THD	93
Exa 8.10	1 to calculate THD of output voltage and its distortion factor 2 to calculate THD of output current and its distortion factor 3 load power and avg dc source current 4 conduction time of each transistor and diode 5 peak and rms current of each transistor	94

Exa 8.11	to determine 1 rms value of load current 2 rms value of thyristor current 3 load power	96
Exa 8.12	to determine power delivered to load for 1 square wave output 2 quasi square wave output 3 symmetrically placed pulses	96
Exa 8.14	to determine the value of source inductance and also find value of commutating capacitors	98
Exa 8.15	to check whether the ckt will commutate by itself or not and the voltage across the capacitor and inductor at the commutation	98
Exa 8.16	to calculate output freq	99
Exa 8.17	To determine 1 ckt turn off time 2 max possible operating freq	100
Exa 8.18	to calculate power delivered to the load and avg and rms values of thyristor current	101
Exa 8.19	to find the value of capacitor C for given conditions . .	101
Exa 8.20	to calculate 1 rms values of phase and line voltages 2 rms value of fundamental component of phase and line voltages 3 THD for voltages 4 load power and avg source current 5 avg value of thyristor current	102
Exa 9.1	to determine 1 rms value of output voltage 2 power delivered to the load and input pf 3 avg input current . .	104
Exa 9.2	to calculate 1 rms value of output voltage 2 load power and input pf 3 avg and rms currents of thyristors . . .	105
Exa 9.3	to determine 1 rms output voltage 2 input pf 3 avg and rms thyristor currents	105
Exa 9.4	to calculate 1 max values of avg and rms thyristor currents 2 min ckt turn off time	106
Exa 9.5	to calculate 1 control range of firing angle 2 max value of rms load current 3 max power and pf 4 max value of avg and rms thyristor currents 5 max possible value of current rating	106
Exa 9.6	to calculate extinction angle and rms value of output voltage	107
Exa 9.8	to calculate 1 rms value of output voltage 2 rms value of current for upper thyristors 3 rms value of current for lower thyristors 4 transformer VA rating 5 input pf . .	108

Exa 10.2	to calculate 1 rms value of output voltage 2 rms current of each converter 3 rms current of each thyristor and 4 input pf	110
Exa 10.4	to compute 1 value of fundamental rms output voltage 2 rms output current 3 output power	111
Exa 10.5	to compute 1 value of fundamental rms output voltage 2 rms output current 3 output power Here 6 pulse bridge convertor is employed	111
Exa 10.7	to calculata rms value of load voltage for various firing angle delays	112
Exa 11.1	to calculate 1 firing angle of the rectifier 2 output voltage 3 dc link voltage	113
Exa 11.2	To calculate rms current and peak reverse voltage ratings for each of thyristor valves	113
Exa 11.3	to determine voltage and current rating of 1 thyristor and 2 diodes of the bridge	114
Exa 11.4	to find the value of parameters of R2 C and load resistance	115
Exa 11.5	to calculate 1 depth of heat of penetration 2 heat generated per unit cylinder surface area and 3 heat generated per unit cylinder volume	115
Exa 11.6	to calculate reqd capacitor size	116
Exa 12.1	To determine 1 firing angle delay of armature converter 2 rms value of thyristor and freewheeling diode currents 3 input pf of armature converter	117
Exa 12.2	calculate the avg armature current and the motor torque	118
Exa 12.3	to determine 1 motor current 2 motor torque 3 input pf	118
Exa 12.4	to determine 1 rated armature current 2 firing angle delay of armature convertor 3 speed regulation at full load 4 input pf of armature convertor and drive	119
Exa 12.5	to calculate 1 delay angle of field converter 2 delay angle of armature converter 3 power fed back to the supply	120
Exa 12.6	To compute 1 motor speed 2 torque developed	121
Exa 12.7	to compute firing angle and motor speed	121
Exa 12.8	to calculate the speed of the motor	122
Exa 12.9	to compute rms value of source and thyristor currents avg value of thyristor currents and input supply pf	123

Exa 12.10	To find 1 no load speed 2 firing angle and supply pf 3 speed regulation	124
Exa 12.11	to calculate 1 transformer phase turn ratio 2 firing angle delay	125
Exa 12.12	to calculate firing angle for different given conditions .	126
Exa 12.13	to evaluate the time taken for the speed to reach 1000rpm	127
Exa 12.14	to determine the 3rd and 5th harmonic components of line current	127
Exa 12.15	to calculate 1 rms and avg value of thyristor current 2 pf of ac source 3 motor speed	128
Exa 12.16	to determine 1 input power from source 2 input resistance of chopper drive 3 motor speed and 4 motor torque	128
Exa 12.17	to calculate avg load current	129
Exa 12.18	to determine 1 range of speed control 2 range of duty cycle	130
Exa 12.19	To calculate the min and max values of armature current and armaure current extrusion	130
Exa 12.21	to determine 1 power returned to the dc supply 2 equivalent load resistance of motor acting as generator 3 min and max breaking speeds 4 speed during regenerative braking	131
Exa 12.22	to determine max current in terms of rated currents at given speeds	132
Exa 12.23	to calculate 1 motor speed at rated load 2 slip at which max torque occurs 3 max torque	133
Exa 12.24	To calculate 1 current and pf at the instant of starting and under max torque conditions 2 starting and max torques	134
Exa 12.25	calculate 1 slip for max torque 2 starting and max torques 3 supply voltage reqd	135
Exa 12.27	to calculate 1 the value of chopper resistance R 2 inductor current 3 duty cycle 4 rectified output voltage 5 efficiency	136
Exa 12.28	To calculate 1 value of chopper duty cycle 2 efficiency for power input 3 input power factor	137
Exa 12.29	to calculate 1 rotor rectified voltage 2 inductor angle 3 delay angle of inveter 4 efficiency 5 motor speed . . .	138
Exa 12.30	to find firing angle advance of inveter	139

Exa 12.31	to find firing angle advance of inveter	140
Exa 12.32	to find the voltage ratio of the transformer	141
Exa 12.33	to calculate 1 supply voltage 2 armature current 3 excitation voltage 4 load angle 5 pull out torque	141
Exa 12.34	to calculate 1 load angle 2 line current 3 input pf	142
Exa 12.35	To calculate motor speed	142
Exa 12.36	to calculate avg motor torque	143
Exa 12.37	to calculate avg motor torque	143
Exa 12.38	to calculate avg motor torque	144
Exa 12.39	to calculate avg motor torque	144
Exa 12.40	to calculate avg motor torque	145
Exa 12.41	to calculate the value of load current and source current	145
Exa 13.1	To calculate load voltage voltage regulation system utilisation and energy consumed	147
Exa 13.2	to calculate the capacitance reqd	148
Exa 13.3	to find reqd values of capacitor and inductor	149
Exa 13.4	to find the firing angle of the TCR	149
Exa 13.5	to calculate the effective inductance at different firing angles	150
Exa 13.6	to find value of inductance	151
Exa 14.1	to calculate dc output voltage rms value of output voltage DF PF and HF	152

Chapter 2

Power Semiconductor Diodes and Transistors

Scilab code Exa 2.1 to find switching freq of the transistor

```
1 clear all;
2 clc;
3 B=40;
4 R_c=10; //ohm
5 V_cc=130; //V
6 V_B=10; //V
7 V_CES=1.0; //V
8 V_BES=1.5; //V
9 I_CS=(V_cc-V_CES)/R_c; //A
10 I_BS=I_CS/B; //A
11 R_B=(V_B-V_BES)/I_BS; printf("value of R.B in
    saturated state= %.3f ohm",R_B);
12 P_T=V_BES*I_BS+V_CES*I_CS; printf("\n power loss
    in transistor= %.3f W",P_T);
13
14 ODF=5;
15 I_B=ODF*I_BS;
16 R_B=(V_B-V_BES)/I_B; printf("\nvalue of R.B for
    an overdrive factor 5 =%.2f ohm",R_B);
```

```

17 P_T=V_BES*I_B+V_CES*I_CS;    printf("\npower loss in
    transistor=%0.2 f W",P_T)
18
19 B_f=I_CS/I_B;    printf("\nforced current gain=%0.0 f"
    ,B_f);

```

Scilab code Exa 2.2 to determine avg power loss due to collector current and also to find peak instantaneous power loss due to collector current

```

1 clear all;
2 clc;
3
4 I_CEO=2*10^-3; //A
5 V_CC=220; //V
6 P_dt=I_CEO*V_CC; //instant. power loss during delay
    time
7
8 t_d=.4*10^-6; //s
9 f=5000;
10 P_d=f*I_CEO*V_CC*t_d; //avg power loss during delay
    time
11
12 V_CES=2; //V
13 t_r=1*10^-6; //s
14 I_CS=80; //A
15 P_r=f*I_CS*t_r*(V_CC/2-(V_CC-V_CES)/3); //avg power
    loss during rise time
16
17 t_m=V_CC*t_r/(2*(V_CC-V_CES));
18 P_rm=I_CS*V_CC^2/(4*(V_CC-V_CES)); //instant. power
    loss during rise time
19
20 P_on=P_d+P_r;    printf(" avg power loss during turn
    on=%0.4 f W",P_on);
21 P_nt=I_CS*V_CES;    printf("\ninstantaneous power

```



```

    loss during turn on=%0.0f W",P_nt);
22 t_n=50*10^-6;
23 P_n=f*I_CS*V_CES*t_n;    printf("\navg power loss
    during conduction period=%0.0f W",P_n);

```

Scilab code Exa 2.3 to obtain power loss during turn off time and turn off period and also instant power loss during fall time

```

1 clear all
2 clc;
3 I_CEO=2*10^-3; //A
4 V_CC=220; //V
5 t_d=.4*10^-6; //s
6 f=5000;
7 V_CES=2; //V
8 t_r=1*10^-6; //s
9 I_CS=80; //A
10 t_n=50*10^-6; //s
11 t_0=40*10^-6; //s
12 t_f=3*10^-6; //s
13 P_st=I_CS*V_CES; // instant. power loss during t_s
14 P_s=f*I_CS*V_CES*t_f; //avg power loss during t_s
15 P_f=f*t_f*(I_CS/6)*(V_CC-V_CES); //avg power loss
    during fall time
16 P_fm=(I_CS/4)*(V_CC-V_CES); //peak instant power
    dissipation
17
18 P_off=P_s+P_f;    printf("total avg power loss
    during turn off=%0.0f W",P_off);
19 P_0t=I_CEO*V_CC;    printf("\ninstantaneous power
    loss during t_0=%0.2f W",P_0t);
20 P_0=f*I_CEO*V_CC*t_0; //avg power loss during t_s
21 P_on=14.9339; //W    from previous eg
22 P_n=40; //W    from previous eg
23 P_T=P_on+P_n+P_off+P_0;    printf("\ntotal power

```

```
loss=0.3 f W',P_T);
```

Scilab code Exa 2.4 to find switching freq of the transistor

```
1 clear all
2 clc;
3 I_CS=100;
4 V_CC=200;
5 t_on=40*10^-6;
6 P_on=(I_CS/50)*10^6*t_on*(V_CC*t_on/2-(V_CC*10^6*
   t_on^2/(40*3))); //energy during turn on
7 t_off=60*10^-6;
8 P_off=(I_CS*t_off/2-(I_CS/60)*10^6*(t_off^2)/3)*((
   V_CC/75)*10^6*t_off); //energy during turn off
9 P_t=P_on+P_off; //total energy
10 P_avg=300;
11 f=P_avg/P_t; printf("allowable switching
   frequency=%0.1f Hz",f);
12 //in book ans is: f=1123.6 Hz. The difference in
   results due to difference in rounding of of
   digits
```

Chapter 3

Diode Circuits and Rectifiers

Scilab code Exa 3.2 to find the conduction time of diode peak current through diode and final steady state voltage

```
1 clear all;
2 clc;
3 V_s=400; //V
4 V_o=100; //V
5 L=100; //uH
6 C=30; //uF
7 t_o=%pi*sqrt(L*C);    printf("conduction time of
   diode=%0.2f us",t_o);
8 //in book solution is t_o=54.77 us. The ans is
   incorrect as %pi is not multiplied in ans.
   Formulae mentioned in correct.
9 I_p=(V_s-V_o)*sqrt(C/L);    printf("\npeak current
   through diode=%0.2f A",I_p);
10 v_D=-V_s+V_o;    printf("\nvoltage across diode=%0.0f
   V",v_D);
```

Scilab code Exa 3.6 to determine conduction time of diode and rate of change of current

```

1  clc
2  clear all
3  R=10; //ohm
4  L=.001; //H
5  C=5*10^-6; //F
6  V_s=230; //V
7  xi=R/(2*L);
8  w_o=1/sqrt(L*C);
9  w_r=sqrt((1/(L*C))-(R/(2*L))^2);
10 t=%pi/w_r;    printf('conduction time of diode=%0.3 f
    us ',t*10^6);
11 t=0;
12 //di=di/dt
13 di=V_s/L;    printf('\nrate of change of current at
    t=0 is %0.0 f A/s ',di);

```

Scilab code Exa 3.7 to find the time required to deliver a charge of 200Ah

```

1  clear all;
2  clc;
3  I_or=100; //A
4  R=1; //assumption
5  V_m=I_or*2*R;
6  I_o=V_m/(%pi*R);
7  q=200; //Ah
8  t=q/I_o;    printf("time required to deliver charge=
    %0.04 f hrs",t);

```

Scilab code Exa 3.8 to calculate the power delivered to the heater and find peak diode current and input pf

```

1  clear all;
2  clc;

```

```

3 V_s=230; //V
4 P=1000; //W
5 R=V_s^2/P;
6 V_or=sqrt(2)*V_s/2;
7 P_h=V_or^2/R;    printf("power delivered to the
    heater=%0.0f W",P_h);
8 V_m=sqrt(2)*230;
9 I_m=V_m/R;    printf("\npeak value of diode current=
    %0.4f A",I_m);
10 pf=V_or/V_s;    printf("\ninput power factor=%0.3f",
    pf);

```

Scilab code Exa 3.9 to find 1 the value of avg charging current 2 power supplied to battery and that dissipated in the resistor 3 supply pf 4 charging time 5 rectifier efficiency and PIV of diode

```

1 clear all;
2 clc;
3 V_s=230; //V
4 V_m=V_s*sqrt(2);
5 E=150; //V
6 theta1=asind(E/(sqrt(2)*V_s));
7 R=8; //ohm
8 f=50; //Hz
9 I_o=(1/(2*pi*R))*((2*sqrt(2)*V_s*cosd(theta1))-E*(
    pi-2*theta1*pi/180));
10 printf("avg value of charging current=%0.4f A",I_o);
11 P_d=E*I_o;    printf("\npower delivered to battery=%
    0.2f W",P_d);
12 I_or=sqrt((1/(2*pi*R^2))*((V_s^2+E^2)*(pi-2*theta1
    pi/180)+V_s^2*sind(2*theta1)-4*V_m*E*cosd(
    theta1)));    printf("\nrms value of the load
    current=%0.4f A",I_or);
13 pf=(E*I_o+I_or^2*R)/(V_s*I_or);    printf("\nsupply
    pf=%0.3f",pf);

```

```

14 P_dd=I_or^2*R;      printf("\npower dissipated in the
    resistor=%0.2f W",P_dd);
15 q=1000.00; //Wh
16 t=q/P_d;      printf("\ncharging time=%0.3f hr",t);
17 n=P_d*100/(P_d+P_dd);      disp(n,"rectifier
    efficiency (%)");
18 PIV=sqrt(2)*V_s+E;      printf("PIV of diode=%0.2f V",
    PIV);
19 //solutions have small variations due to difference
    in rounding off of digits

```

Scilab code Exa 3.10 to determine the effect of reverse recovery time on the avg output voltage

```

1 clear all;
2 clc;
3 V_s=230; //V;
4 t_rr=40*10^-6; //s reverde recovery time
5 V_o=2*sqrt(2)*V_s/%pi;
6 V_m=sqrt(2)*V_s;
7 disp("when f=50Hz");
8 f=50;
9 V_r=(V_m/%pi)*(1-cosd(2*%pi*f*t_rr*180/%pi));
10 v_avg=V_r*100/V_o*10^-3;      printf("percentage
    reduction in avg o/p voltage=%0.3fx10^-3",v_avg);
11
12 disp("when f=2500Hz");
13 f=2500;
14 V_r=(V_m/%pi)*(1-cosd(2*%pi*f*t_rr*180/%pi));
15 v_avg=V_r*100/V_o;      printf("percentage reduction
    in avg o/p voltage=%0.3f",v_avg);

```

Scilab code Exa 3.11 to determine 1 avg value of output voltage and output current 2 avg and rms values of diode currents 3 rms values of output and input currents and supply pf

```

1 clear all;
2 clc;
3 V_s=230; //V
4 R=10; //ohm
5 V_m=sqrt(2)*V_s;
6 V_o=2*V_m/%pi; printf("avg value of o/p voltage=%
    .2f V",V_o);
7 I_o=V_o/R; printf("\navg value of o/p current=%0.3
    f A",I_o);
8 I_DA=I_o/2; printf("\navg value of diode current=
    %0.3f A",I_DA);
9 I_Dr=I_o/sqrt(2); printf("\nrms value of diode
    current=%0.3f A",I_Dr);
10
11 printf("\nrms value of o/p current=%0.3f A",I_o);
12 printf("\nrms value of i/p current=%0.3f A",I_o);
13 pf=(V_o/V_s); printf("\nsupply pf=%0.2f",pf);

```

Scilab code Exa 3.12 to calculate 1 peak load current 2 dc load current 3 dc diode current 4 percentage regulation from no load to full load

```

1 clear all;
2 clc;
3 V_s=230; //V
4 R=1000; //ohm
5 R_D=20; //ohm
6 V_m=sqrt(2)*V_s;
7 I_om=V_m/(R+R_D); printf("peak load current=%0.4f
    A",I_om);
8 I_o=I_om/%pi; printf("\ndc load current=%0.5f A",
    I_o);

```

```

9  V_D=I_o*R_D-V_m/%pi;    printf("\ndc diode voltage=%
    .1 f V",V_D);
10 V_on=V_m/%pi;    printf("\nat no load, load voltage=
    %.3 f V",V_on);
11 V_o1=I_o*R;    printf("\nat given load, load voltage
    =%.3 f V",V_o1);
12 vr=(V_on-V_o1)*100/V_on;    printf("\nvoltage
    regulation(in percent)=%.3 f",vr);

```

Scilab code Exa 3.13 to design a zener voltage regulator to meet given specifications

```

1  clear all;
2  clc;
3  V_L=6.8; //V
4  V_smax=20*1.2; //V
5  V_smin=20*.8; //V
6  I_Lmax=30*1.5; //mA
7  I_Lmin=30*.5; //mA
8  I_z=1; //mA
9
10 R_smax=(V_smax-V_L)/((I_Lmin+I_z)*10^-3);    printf(
    "max source resistance=%0.0 f ohm",R_smax);
11 R_smin=(V_smin-V_L)/((I_Lmax+I_z)*10^-3);    printf(
    "\nmin source resistance=%0.0 f ohm",R_smin); //in
    book solution, error is committed in putting the
    values in formulae(printing error) but solution
    is correct
12 R_Lmax=V_L*1000/I_Lmin;    printf("\nmax load
    resistance=%0.1 f ohm",R_Lmax);
13 R_Lmin=V_L*1000/I_Lmax;    printf("\nmin load
    resistance=%0.1 f ohm",R_Lmin);
14
15 V_d=.6; //V
16 V_r=V_L-V_d;    printf("\nvoltage rating of zener

```



```
diode=%0.1f V" ,V_r);
```

Scilab code Exa 3.14 to find R1 and R2

```
1 clear all;
2 clc;
3 I2=200*10^-6; //A
4 V_z=20; //V
5 R_G=500; //ohm
6 R2=(V_z/I2)-R_G;    printf("R2=%0.1f kilo-ohm" ,R2
    /1000);
7
8 V_v=25; //V
9 I1=I2;
10 R1=(V_v-V_z)/I1;    printf("\nR1=%0.0f kilo-ohm" ,R1
    /1000);
```

Scilab code Exa 3.15 to find the VA rating of the transformer

```
1 clear all;
2 clc;
3 V_s=2*230; //V
4 V_o=(sqrt(2)*V_s)/%pi;
5 R=60; //ohm
6 P_dc=(V_o)^2/R;
7 TUF=0.2865;
8 VA=P_dc/TUF;    printf("kVA rating of the
    transformer=%0.1f kVA" ,VA/1000);
```

Scilab code Exa 3.16 to determine 1 avg value of output voltage 2 input current distortion factor 3 input displacement factor 4 input pf 5 input current harmonic factor and 6 crest ratio

```

1 clear all;
2 clc;
3 tr=0.5;    //turns ratio
4 I_o=10;
5 V=230;
6 V_s=V/tr;
7 V_m=sqrt(2)*V_s;
8 V_o=2*V_m/%pi;    printf("o/p voltage=%0.2 f V",V_o);
9 phi1=0;    //displacement angle=0 as fundamental
             component of i/p source current in phase with
             source voltage
10 DF=cosd(phi1);    printf("\ndistortion factor=%0.0 f",
    DF);
11 I_s1=4*I_o/(sqrt(2)*%pi);
12 I_s=sqrt(I_o^2*%pi/%pi);
13 CDF=I_s1/I_o;    printf("\ncurrent displacement factor
    =%0.1 f",CDF);
14 pf=CDF*DF;    printf("\ni/p pf=%0.2 f",pf);
15 HF=sqrt((I_s/I_s1)^2-1);    printf("\nharmonic
    factor=%0.2 f",HF);
16 CF=I_o/I_s;    printf("\ncrest factor=%0.2 f",CF);

```

Scilab code Exa 3.17 to determine diode rating and transformer rating

```

1 clear all;
2 clc;
3 V_o=230;
4 R=10;
5 V_s=V_o*%pi/(2*sqrt(2));
6 I_o=V_o/R;
7 I_m=sqrt(2)*V_s/R;    printf("peak diode current=%0.2

```

```

    f A",I_m);
8 I_DAV=I_m/%pi;    printf("\nI_DAV=%0.2 f A",I_DAV);
    //avg value of diode current
9 I_Dr=I_m/2;    printf("\nI_Dr=%0.2 f A",I_Dr);    //
    rms value of diode current
10 PIV=sqrt(2)*V_s;    printf("\nPIV=%0.1 f V",PIV);
11 I_s=I_m/sqrt(2);
12 TF=V_s*I_s;    printf("\ntransformer rating=%0.2 f kVA
    ",TF/1000);

```

Scilab code Exa 3.18 to calculate avg value of output voltage avg and rms values of diode current and power delivered to diode

```

1 clear all;
2 clc;
3 tr=5;
4 V=1100;
5 R=10;
6 disp("in case of 3ph-3pulse type");
7 V_ph=V/tr;
8 V_mp=sqrt(2)*V_ph;
9 V_o=3*sqrt(3)*V_mp/(2*%pi);    printf("avg o/p
    voltage=%0.1 f V",V_o);
10 I_mp=V_mp/R;
11 I_D=(I_mp/%pi)*sin(%pi/3);    printf("\navg value of
    diode current=%0.3 f A",I_D);
12 I_Dr=I_mp*sqrt((1/(2*%pi))*(%pi/3+.5*sin(2*%pi/3)));
    printf("\nrms value of diode current=%0.2 f A",
    I_Dr);
13 V_or=V_mp*sqrt((3/(2*%pi))*(%pi/3+.5*sin(2*%pi/3)));
14 P=(V_or^2)/R;    printf("\npower delivered=%0.1 f W",P
    );
15
16 disp("in case of 3ph-M6 type");
17 V_ph=V_ph/2;

```

```

18 V_mp=sqrt(2)*V_ph;
19 V_o=3*V_mp/(%pi);    printf(" avg o/p voltage=%0.2 f V"
    ,V_o);
20 I_mp=V_mp/R;
21 I_D=(I_mp/%pi)*sin(%pi/6);    printf("\navg value of
    diode current=%0.4 f A",I_D);
22 I_Dr=I_mp*sqrt((1/(2*%pi))*(%pi/6+.5*sin(2*%pi/6)));
    printf("\nrms value of diode current=%0.3 f A",
    I_Dr);
23 V_or=V_mp*sqrt((6/(2*%pi))*(%pi/6+.5*sin(2*%pi/6)));
24 P=(V_or^2)/R;    printf("\npower delivered=%0.0 f W",P
    );

```

Scilab code Exa 3.19 determine the ratings of diodes and of 3ph deltastar transformer

```

1 clear all;
2 clc;
3 V_o=400;
4 R=10;
5 V_ml=V_o*%pi/3;
6 V_s=V_ml/(sqrt(2)*sqrt(3));
7 I_m=V_ml/R;
8 I_s=.7804*I_m;
9 tr=3*V_s*I_s;    printf(" transformer rating=%0.1 f VA"
    ,tr);
10
11 I_Dr=.5518*I_m;    printf("\nrms value of diode
    current=%0.3 f A",I_Dr);
12 I_D=I_m/%pi;    printf("\navg value of diode current
    =%0.3 f A",I_D);
13 printf("\npeak diode current=%0.2 f A",I_m);
14 PIV=V_ml;    printf("\nPIV=%0.2 f V",PIV);

```

Scilab code Exa 3.20 to determine 1 power delivered to battery and load
 2 input displacement factor 3 current distortion factor 4 input pf 5 input HF
 6 transformer rating

```

1 clear all;
2 clc;
3 V_1=230;
4 E=240;
5 R=8;
6 V_m1=sqrt(2)*V_1;
7 V_o=3*V_m1/%pi;
8 I_o=(V_o-E)/R;
9 P_b=E*I_o;    printf("power delivered to battery=%0.1
    f W",P_b);
10 P_d=E*I_o+I_o^2*R;    printf("\npower delivered to
    load=%0.2 f W",P_d);
11
12 phi1=0;
13 DF=cosd(phi1);
14 printf("\ndisplacement factor=%0.0 f",DF);
15 I_s1=2*sqrt(3)*I_o/(sqrt(2)*%pi);
16 I_s=sqrt(I_o^2*2*%pi/(3*%pi));
17 CDF=I_s1/I_s;    printf("\ncurrent distortion factor
    =%0.3 f",CDF);
18 pf=DF*CDF;    printf("\ni/p pf=%0.3 f",pf);
19 HF=sqrt(CDF^-2-1);    printf("\nharmonic factor=%0.4 f
    ",HF);
20 tr=sqrt(3)*V_1*I_o*sqrt(2/3);    printf("\
    ntransformer rating=%0.2 f VA",tr);
21 //answers have small variations from the book due to
    difference in rounding off of digits

```

Scilab code Exa 3.21 design capacitor filter and avg value of output voltage with and without filter

```

1 clear all;
2 clc;
3 f=50; //Hz
4 V=230;
5 V_m=sqrt(2)*V;
6 R=400;
7 RF=0.05;
8 C=(1/(4*f*R))*(1+(1/(sqrt(2)*RF))); printf("
    capacitor value=%0.1f uF",C/10^-6);
9 V_o=V_m*(1-1/(4*f*R*C)); printf("\no/p voltage
    with filter=%0.3f V",V_o);
10 V_o=2*V_m/%pi; printf("\no/p voltage without
    filter=%0.2f V",V_o);

```

Scilab code Exa 3.22 to find the value of L with different R and CRF without L

```

1 clear all;
2 clc;
3 f=50;
4 CRF=0.05;
5 R=300;
6 L=sqrt((CRF/(.4715*R))^-2-R^2)/(2*2*%pi*f);
    printf("L=%0.4f H",L);
7 R=30;
8 L=sqrt((CRF/(.4715*R))^-2-R^2)/(2*2*%pi*f);
    printf("\nL=%0.4f H",L);
9 L=0;
10 CRF=.4715*R/sqrt(R^2+(2*2*%pi*f*L)^2); printf("\
    nCRF=%0.4f",CRF);

```

Scilab code Exa 3.23 to design LC filter

```
1 clear all;
2 clc;
3 R=50;
4 L_L=10*10^-3;
5 f=50;
6 w=2*%pi*f;
7 C=10/(2*w*sqrt(R^2+(2*w*L_L)^2)); printf("C=%0.2 f
    uF", C*10^6);
8 VRF=0.1;
9 L=(1/(4*w^2*C))*((sqrt(2)/(3*VRF))+1); printf("\
    nL=%0.3 f mH", L*10^3);
```

Chapter 4

Thyristors

Scilab code Exa 4.2 to plot allowable gate voltage as the function of gate current and to find avg gate power dissipation

```
1 clc
2 clear all
3 P=.5; //P=V_g*I_g
4 V=[0.01:0.01:.25];
5 plot2d(P./V);
6 xlabel('I_g in A');
7 ylabel('V_g in V');
8
9 t=poly(0, 't');
10 P1=P*t/(2*t);
11 printf('average gate power dissipation(in watts)');
12 disp(P1);
```

Scilab code Exa 4.3 to compute gate source resistance

```
1 clear all;
2 clc;
```



```

3 P=.5;      //P=V_g*I_g
4 s=130;     //s=V_g/I_g
5 I_g=sqrt(P/s);
6 V_g=s*I_g;
7 E=15;
8 R_s=(E-V_g)/I_g;    printf("gate source resistance=%g
    .2 f ohm",R_s);
9 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 4.4 to compute gate source resistance trigger voltage and trigger current

```

1 clear all;
2 clc;
3 R_s=120;    //slope of load line is -120V/A. This
    gives gate source resistance
4 printf("gate source resistance=%g.0 f ohm",R_s);
5
6 P=.4;      //P=V_g*I_g
7 E_s=15;
8 //E_s=I_g*R_s+V_g,    after solving this
9 //120*I_g^2-15*I_g+0.4=0    so
10 a=120;    b=-15;    c=0.4;
11 D=sqrt((b^2)-4*a*c);
12 I_g=(-b+D)/(2*a);    V_g=P/I_g;
13 printf("\ntrigger current=%g.2 f mA",I_g*10^3);
    printf("\nthen trigger voltage=%g.3 f V",V_g);
14
15 I_g=(-b-D)/(2*a);    V_g=P/I_g;
16 printf("\ntrigger current=%g.2 f mA",I_g*10^3);
    printf("\nthen trigger voltage=%g.2 f V",V_g);

```

Scilab code Exa 4.5 to compute 1 resistance 2 triggering freq 3 duty cycle of the triggering pulse

```

1 clear all;
2 clc;
3 //V_g=1+10*I_g
4 P_gm=5; //P_gm=V_g*I_g
5 //after solving, eqn becomes 10*I_g^2+I_g-5=0
6 a=10; b=1; c=-5;
7 I_g=(-b+sqrt(b^2-4*a*c))/(2*a);
8 E_s=15;
9 //using E_s=R_s*I_g+V_g
10 R_s=(E_s-1)/I_g-10; printf("reistance=%0.3f ohm",
    R_s);
11
12 P_gav=.3; //W
13 T=20*10^-6;
14 f=P_gav/(P_gm*T); printf("\ntriggering freq=%0.0f
    kHz",f/1000);
15
16 dl=f*T; printf("\nduty cycle=%0.2f",dl);

```

Scilab code Exa 4.6 to compute min width of gate pulse current

```

1 clear all;
2 clc;
3 I=.1;
4 E=200;
5 disp("in case load consists of (a)L=.2H");
6 L=.2;
7 t=I*L/E; printf("min gate pulse width=%0.0f us",t
    *10^6);

```

```

8 disp(" (b)R=20ohm in series with L=.2H");
9 R=20;
10 t=(-L/R)*log(1-(R*I/E));    printf("min gate pulse
    width=%0.3 f us",t*10^6);
11 disp(" (c)R=20ohm in series with L=2H");
12 L=2;
13 t=(-L/R)*log(1-(R*I/E));    printf("min gate pulse
    width=%0.2 f us",t*10^6);

```

Scilab code Exa 4.8 to calculate trigger voltage and trigger current

```

1 clc
2 clear all
3 E_s=16;
4 R_s=128;
5 P=.5;
6 y=poly([P -E_s R_s], 'i', 'coeff');
7 a=roots(y);
8 printf('trigger current=%0.1 f mA',a(1)*1000);
9 printf('\ntrigger voltage=%0.0 f V',P/a(1));

```

Scilab code Exa 4.9 to compute avg on current rating for half sine wave-current for various conduction angles

```

1 clear all;
2 clc;
3 function [I_TAV]=theta(th)
4     I_m=1;    //supposition
5     I_av=(I_m/(2*pi))*(1+cosd(th));
6     I_rms=sqrt((I_m/(2*pi))*((180-th)*pi/360+.25*
    sind(2*th)));
7     FF=I_rms/I_av;
8     I_rms=35;

```

```

9      I_TAV=I_rms/FF;
10  endfunction
11  disp("when conduction angle=180");
12  th=0;
13  I_TAV=theta(th);
14  printf("avg on current rating=%0.3 f A",I_TAV);
15  disp("when conduction angle=90");
16  th=90;
17  I_TAV=theta(th);
18  printf("avg on current rating=%0.3 f A",I_TAV);
19  disp("when conduction angle=30");
20  th=150;
21  I_TAV=theta(th);
22  printf("avg on current rating=%0.3 f A",I_TAV);

```

Scilab code Exa 4.10 to compute avg on current rating for half sine wave current for various conduction angles

```

1  clear all;
2  clc;
3  clear all
4  function [I_TAV]=theta(th)
5      n=360/th;
6      I=1;      //supposition
7      I_av=I/n;
8      I_rms=I/sqrt(n);
9      FF=I_rms/I_av;
10     I_rms=35;
11     I_TAV=I_rms/FF;
12  endfunction
13  disp("when conduction angle=180");
14  th=180;
15  I_TAV=theta(th);
16  printf("avg on current rating=%0.3 f A",I_TAV);
17  disp("when conduction angle=90");

```

```

18 th=90;
19 I_TAV=theta(th);
20 printf("avg on current rating=%0.1f A",I_TAV);
21 disp("when conduction angle=30");
22 th=30;
23 I_TAV=theta(th);
24 printf("avg on current rating=%0.4f A",I_TAV);

```

Scilab code Exa 4.11 to calculate surge current rating and power rating

```

1 clear all;
2 clc;
3 f=50; //Hz
4 I_sb=3000;
5 t=1/(4*f);
6 T=1/(2*f);
7 I=sqrt(I_sb^2*t/T); printf("surge current rating=
    %0.2f A",I);
8 r=(I_sb/sqrt(2))^2*T; printf("\nI^2*t rating=%0.0f
    A^2.s",r);

```

Scilab code Exa 4.12 to find maximum value of the remedial parameter

```

1 clc
2 clear all
3 V_s=300; //V
4 R=60; //ohm
5 L=2; //H
6 t=40*10^-6; //s
7 i_T=(V_s/R)*(1-exp(-R*t/L));
8 i=.036; //A
9 R1=V_s/(i-i_T);

```

```
10 printf("maximum value of remedial parameter=%0.3 f
    kilo-ohm", R1/1000);
```

Scilab code Exa 4.16 to calculate fault clearance time

```
1 clear all;
2 clc;
3 V_p=230*sqrt(2);
4 R=1+((1)^-1+(10)^-1)^-1;
5 A=V_p/R;
6 s=1; //s
7 t_c=20*A^-2*s; printf("fault clearance time=%0.4 f
    ms", t_c*10^3);
```

Scilab code Exa 4.17 1 to calculate the max values of change in current and change in voltage for SCR 2 rms and avg current rating and 3 suitable voltage rating

```
1 clear all;
2 clc;
3 V_s=sqrt(2)*230; //V
4 L=15*10^-6; //H
5 I=V_s/L; //I=(di/dt)_max
6 printf("(di/dt)_max=%0.3 f A/usec", I/10^6);
7 R_s=10; //ohm
8 v=I*R_s; //v=(dv/dt)_max
9 printf("\n(dv/dt)_max=%0.2 f V/usec", v/10^6);
10
11 f=50; //Hz
12 X_L=L*2*pi*f;
13 R=2;
14 I_max=V_s/(R+X_L); printf("\nI_rms=%0.3 f A", I_max)
    ;
```

```

15 disp("when conduction angle=90");
16 FF=%pi/sqrt(2);
17 I_TAV=I_max/FF;    printf("I_TAV=%0.3 f A", I_TAV);
18 disp("when conduction angle=30");
19 FF=3.98184;
20 I_TAV=I_max/FF;    printf("I_TAV=%0.3 f A", I_TAV);
21
22 printf("\nvoltage rating=%0.3 f V", 2.75*V_s); //rating
    is taken 2.75 times of peak working voltage
    unlike 2.5 to 3 times as mentioned int book.

```

Scilab code Exa 4.19 1 to check heat sink selection is satisfactory 2 to choose heat sink for given requirements and ckt efficiency 3compute case and junction temp

```

1 clear all;
2 clc;
3 T_jm=125;
4 th_jc=.15;    //degC/W
5 th_cs=0.075;    //degC/W
6
7 dT=54;    //dT=T_s-T_a
8 P_av=120;
9 th_sa=dT/P_av;
10 T_a=40;    //ambient temp
11 P_av=(T_jm-T_a)/(th_sa+th_jc+th_cs);
12 if((P_av-120)<1)
13     disp("selection of heat sink is satisfactory");
14 end
15 dT=58;    //dT=T_s-T_a
16 P_av=120;
17 th_sa=dT/P_av;
18 T_a=40;    //ambient temp
19 P_av=(T_jm-T_a)/(th_sa+th_jc+th_cs);
20 if((P_av-120)<1)

```

```

21     disp("selection of heat sink is satisfactory");
22 end
23
24 V_m=sqrt(2)*230;
25 R=2;
26 I_TAV=V_m/(R*pi);
27 P_av=90;
28 th_sa=(T_jm-T_a)/P_av-(th_jc+th_cs);
29 dT=P_av*th_sa;
30 disp("for heat sink");    printf("T_s-T_a=%0.2 f degC"
    ,dT);    printf("\nP_av=%0.0 f W",P_av);
31 P=(V_m/2)^2/R;
32 eff=P/(P+P_av);    printf("\nckt efficiency=%0.3 f pu"
    ,eff);
33
34 a=60;    //delay angle
35 I_TAV=(V_m/(2*pi*R))*(1+cosd(a));
36 printf("\nI_TAV=%0.2 f A",I_TAV);
37 dT=46;
38 T_s=dT+T_a;
39 T_c=T_s+P_av*th_cs;    printf("\ncase temp=%0.2 f degC"
    ,T_c);
40 T_j=T_c+P_av*th_jc;    printf("\njunction temp=%0.2 f
    degC",T_j);

```

Scilab code Exa 4.20 to find total avg power loss and percentage inc in device rating

```

1 clear all;
2 clc;
3 T_j=125;    //degC
4 T_s=70;    //degC
5 th_jc=.16;    //degC/W
6 th_cs=.08;    //degC/W
7 P_av1=(T_j-T_s)/(th_jc+th_cs);    printf("total avg

```



```

    power loss in thristor sink combination=%0.2f W",
    P_av1);
8
9 T_s=60;      //degC
10 P_av2=(T_j-T_s)/(th_jc+th_cs);
11
12 inc=(sqrt(P_av2)-sqrt(P_av1))*100/sqrt(P_av1);
    printf("\npercentage inc in rating=%0.2f",inc);

```

Scilab code Exa 4.21 to determine voltage across each SCR and discharge current across each capacitor

```

1 clear all;
2 clc;
3 R=25000;
4 I_l1=.021;    //I_l=leakage current
5 I_l2=.025;
6 I_l3=.018;
7 I_l4=.016;
8 //V1=(I-I_l1)*R;
9 //V2=(I-I_l2)*R;
10 //V3=(I-I_l3)*R;
11 //V4=(I-I_l4)*R;
12 //V=V1+V2+V3+V4
13 V=10000;
14 I_l=I_l1+I_l2+I_l3+I_l4;
15 //after solving
16 I=((V/R)+I_l)/4;
17 R_c=40;
18 V1=(I-I_l1)*R;    printf("voltage across SCR1=%0.0f V
    ",V1);
19 V2=(I-I_l2)*R;    printf("\nvoltage across SCR2=%0.0f
    V",V2);
20 V3=(I-I_l3)*R;    printf("\nvoltage across SCR3=%0.0f
    V",V3);

```

```

21 V4=(I-I_14)*R;    printf("\nvoltage across SCR4=%0.0 f
    V",V4);
22
23 I1=V1/R_c;    printf("\ndischarge current through
    SCR1=%0.3 f A",I1);
24 I2=V2/R_c;    printf("\ndischarge current through
    SCR2=%0.3 f A",I2);
25 I3=V3/R_c;    printf("\ndischarge current through
    SCR3=%0.3 f A",I3);
26 I4=V4/R_c;    printf("\ndischarge current through
    SCR4=%0.3 f A",I4);

```

Scilab code Exa 4.22 to calculate number of series and parrallel units of SCRs

```

1 clear all;
2 clc;
3 V_r=1000;    //rating of SCR
4 I_r=200;    //rating of SCR
5 V_s=6000;    //rating of String
6 I_s=1000;    //rating of String
7 disp("when DRF=.1");
8 DRF=.1;
9 n_s=V_s/(V_r*(1-DRF));    printf("number of series
    units=%0.0 f",ceil(n_s));
10 n_p=I_s/(I_r*(1-DRF));    printf("\nnumber of
    parrallel units=%0.0 f",ceil(n_p));
11 disp("when DRF=.2");
12 DRF=.2;
13 n_s=V_s/(V_r*(1-DRF));    printf("number of series
    units=%0.0 f",ceil(n_s));
14 n_p=I_s/(I_r*(1-DRF));    printf("\nnumber of
    parrallel units=%0.0 f",ceil(n_p));

```

Scilab code Exa 4.23 to calculate the resistance

```
1 clear all;
2 clc;
3 V1=1.6; //on state voltage drop of SCR1
4 V2=1.2; //on state voltage drop of SCR2
5 I1=250; //current rating of SCR1
6 I2=350; //current rating of SCR2
7 R1=V1/I1;
8 R2=V2/I2;
9 I=600; //current to be shared
10 //for SCR1, I*(R1+R)/(total resistance)=k*I1
    (1)
11 //for SCR2, I*(R2+R)/(total resistance)=k*I2
    (2)
12 //(1)/(2)
13 R=(R2*I2-R1*I1)/(I1-I2);
14 printf("reqd value of resistance=%0.3f ohm",R);
```

Scilab code Exa 4.25 to compute value of various resistances

```
1 clear all;
2 clc;
3 f=2000; //Hz
4 C=0.04*10^-6;
5 n=.72;
6 R=1/(f*C*log(1/(1-n))); printf("R=%0.2f kilo-ohm",
    R/1000);
7 V_p=18;
8 V_BB=V_p/n;
9 R2=10^4/(n*V_BB); printf("\nR2=%0.2f ohm",R2);
10 I=4.2*10^-3; //leakage current
```

```

11 R_BB=5000;
12 R1=(V_BB/I)-R2-R_BB;      printf("\nR1=%0.0 f ohm",R1);

```

Scilab code Exa 4.26 to compute max and min values of R and the corresponding freq

```

1 clear all;
2 clc;
3 V_p=18;
4 n=.72;
5 V_BB=V_p/n;
6 I_p=.6*10^-3;
7 I_v=2.5*10^-3;
8 V_v=1;
9 R_max=V_BB*(1-n)/I_p;      printf("R_max=%0.2 f kilo-ohm
    ",R_max/1000);
10 R_min=(V_BB-V_v)/I_v;     printf("\nR_min=%0.2 f kilo-
    ohm",R_min/1000);
11
12 C=.04*10^-6;
13 f_min=1/(R_max*C*log(1/(1-n)));  printf("\nf_min=%0
    .3 f kHz",f_min/1000);
14 f_max=1/(R_min*C*log(1/(1-n)));  printf("\nf_max=%0
    .2 f kHz",f_max/1000);

```

Scilab code Exa 4.27 to find max and min firing angles for triac

```

1 clear all;
2 clc;
3 R1=1000;
4 C=.5*10^-6;
5 f=50;
6 w=2*pi*f;

```

```

7 V_s=230;
8 X_c=1/(w*C);
9 v_c=30;
10 R=0;
11 Z=sqrt((R+R1)^2+X_c^2);
12 phi=atand(X_c/(R+R1));
13 I1=V_s/(Z*complex(cosd(-phi),sind(-phi)));
14 V_c=I1*X_c*complex(cosd(-90),sind(-90));
15 a=abs(V_c); //magnitude of V_c
16 b=-atand(imag(V_c)/real(V_c)); //argument of V_c
17 //v_c=sqrt(2)*a*sind(a1-b)
18 a1=asind(v_c/(sqrt(2)*a))+b; printf("min angle=%
    .1f deg",a1);
19
20 R=25000;
21 Z=sqrt((R+R1)^2+X_c^2);
22 phi=atand(X_c/(R+R1));
23 I1=V_s/(Z*complex(cosd(-phi),sind(-phi)));
24 V_c=I1*X_c*complex(cosd(-90),sind(-90));
25 a=abs(V_c); //magnitude of V_c
26 b=-atand(imag(V_c)/real(V_c)); //argument of V_c
27 //v_c=sqrt(2)*a*sind(a2-b)
28 a2=asind(v_c/(sqrt(2)*a))+b; printf("\nmax angle=
    %.2f deg",a2);

```

Chapter 5

Thyristor Commutation Techniques

Scilab code Exa 5.1 to determine 1 conduction time and 2 voltage across thyristor

```
1 clear all;
2 clc;
3 L=5*10^-3;
4 C=20*10^-6;
5 V_s=200;
6 w_o=sqrt(1/(L*C));
7 t_o=%pi/w_o;    printf(" conduction time of thyristor
   =%.5 f ms",t_o*1000);
8 printf("\nvoltage across thyristor=%.0 f V",-V_s);
```

Scilab code Exa 5.2 to calculate 1 conduction time for auxillary thyristor 2 voltage across and 3 circuit turn off time for main thyristor

```
1 clear all;
2 clc;
```

```

3 C=20*10^-6;
4 L=5*10^-6;
5 V_s=230;
6 I_p=V_s*sqrt(C/L);
7 w_o=sqrt(1/(L*C));
8 t_o=%pi/w_o;    printf("conduction time of auxillary
    thyristor=%0.3f us",t_o*10^6);
9 I_o=300;
10 //a=w_o*(t3-t2)=asind(I_o/(2*V_s));
11 a=asind(I_o/(2*V_s));
12 V_ab=V_s*cosd(a);    printf("\nvoltage across main
    thyristor=%0.3f V",V_ab);
13 t_c=C*V_ab/I_o;    printf("\nckt turn off time=%0.3f
    us",t_c*10^6);

```

Scilab code Exa 5.3 to determine 1 peak value of current 2 value of capacitor C

```

1 clear all;
2 clc;
3 V_s=200;
4 R1=10;
5 R2=100;
6 I1=V_s*(1/R1+2/R2);    printf("peak value of current
    through SCR1=%0.0f A",I1);
7 I2=V_s*(2/R1+1/R2);    printf("\npeak value of
    current through SCR2=%0.0f A",I2);
8 t_c1=40*10^-6;
9 fos=2;    //factor of safety
10 C1=t_c1*fos/(R1*log(2));
11 C2=t_c1*fos/(R2*log(2));
12 if(C1>C2)
13     printf("\nvalue of capacitor=%0.4f uF",C1*10^6);
14 else
15     printf("\nvalue of capacitor=%0.4f uF",C2*10^6);

```

16 `end`

Scilab code Exa 5.4 to calculate 1 value of current 2 circuit turn off time for main and auxillary thyristor

```
1 clear all;
2 clc;
3 V_s=230;
4 L=20*10^-6;
5 C=40*10^-6;
6 I_o=120;
7 I_p=V_s*sqrt(C/L);
8 printf("current through main thyristor=%0.2f A",I_o+
   I_p);
9 printf("\ncurrent through auxillary thyristor=%0.0f A
   ",I_o);
10
11 t_c=C*V_s/I_o;    printf("\ncircuit turn off time
   for main thyristor=%0.2f us",t_c*10^6);
12 w_o=sqrt(1/(L*C));
13 t_c1=%pi/(2*w_o);    printf("\ncircuit turn off time
   for auxillary thyristor=%0.2f us",t_c1*10^6);
```

Scilab code Exa 5.5 to compute min value of C

```
1 clear all;
2 clc;
3 C_j=25*10^-12;
4 I_c=5*10^-3;    //charging current
5 V_s=200;
6 R=50;
7 C=(C_j*V_s)/(I_c*R);    printf("Value of C=%0.2f uF",
   C*10^6);
```

Scilab code Exa 5.6 to find circuit turn off time

```
1 clear all;
2 clc;
3 V_s=200;
4 R=5;
5 C=10*10^-6;
6 //for turn off  $V_s*(1-2*\exp(-t/(R*C)))=0$ , so
   after solving
7 t_c=R*C*log(2); printf(" circuit turn off time=%0.4
   f us",t_c*10^6);
```

Scilab code Exa 5.7 to find conduction time of thyristor

```
1 clear all;
2 clc;
3 R=1;
4 L=20*10^-6;
5 C=40*10^-6;
6 w_r=sqrt(((1/(L*C))-(R/(2*L))^2);
7 t_1=%pi/w_r; printf(" conduction time of thyristor
   =%0.3f us",t_1*10^6);
```

Scilab code Exa 5.8 1 to calculate value of Capacitor 2 determine value of Resistance

```
1 clear all;
2 clc;
3 dv=400*10^-6; //dv=dv_T/dt (V/s)
```

```

4 V_s=200;
5 R=20;
6 C=V_s/(R*dv);
7 C_j=.025*10^-12;
8 C_s=C-C_j;    printf(" C_s=%0.3 f uF" ,C_s/10^6);
9
10 I_T=40;
11 R_s=1/((I_T/V_s)-(1/R));    printf("\nR_s=%0.3 f ohm" ,
    R_s);
12 //value of R_s in book is wrongly calculated

```

Scilab code Exa 5.9 calculate 1 time at which commutation of main thyristor gets initiated 2 ckt turn off time

```

1 clear all;
2 clc;
3 V_s=200;
4 C=20*10^-6;
5 L=.2*10^-3;
6 i_c=10;
7 i=V_s*sqrt(C/L);
8 w_o=1/sqrt(L*C);
9 t_1=(1/w_o)*asin(i_c/i);    printf(" reqd time=%0.0 f
    us" ,t_1*10^6);
10
11 t_o=%pi/w_o;
12 t_c=t_o-2*t_1;    printf("\nckt turn off time=%0.1 f
    us" ,t_c*10^6);
13 //solution in book wrong, as wrong values are
    selected while filling the formuleas

```

Scilab code Exa 5.11 to find current in R and L

```

1 clear all;
2 clc;
3 L=1;
4 R=50;
5 V_s=200;
6 tau=L/R;
7 t=.01;
8 i=(V_s/R)*(1-exp(-t/tau));
9 Vd=.7;
10 t=8*10^-3;
11 i1=i-t*Vd;    printf("current through L=%0.4f A",i1);
12 i_R=0;        //current in R at t=.008s
13 printf("\ncurrent through R=%0.0f A",i_R);

```

Scilab code Exa 5.12 To find the current in R and L and voltage across C

```

1 clc
2 clear all
3 L=1; //H
4 R=50; //ohm
5 V_s=200; //V
6 tau=L/R;
7 t=.01; //s
8 i=(V_s/R)*(1-exp(-t/tau));    printf("current in R,L
    =%0.3f A",i);
9 C=1*10^-6; //F
10 V_c=sqrt(L/C)*i;
11 printf("\nvoltage across C=%0.3f kV",V_c/1000);

```

Chapter 6

Phase Controlled Rectifiers

Scilab code Exa 6.1 to calculate the power absorbed in the heater element

```
1 clear all;
2 clc;
3 V=230;
4 P=1000;
5 R=V^2/P;
6 disp("when firing angle delay is of 45deg");
7 a=%pi/4;
8 V_or=(sqrt(2)*V/(2*sqrt(%pi)))*sqrt((%pi-a)+.5*sin
    (2*a));
9 P=V_or^2/R;    printf("power absorbed=%.2 f W" ,P);
10
11 disp("when firing angle delay is of 90deg");
12 a=%pi/2;
13 V_or=(sqrt(2)*V/(2*sqrt(%pi)))*sqrt((%pi-a)+.5*sin
    (2*a));
14 P=V_or^2/R;    printf("power absorbed=%.2 f W" ,P);
```

Scilab code Exa 6.2 1 to find value of charging current 2 power supplied to battery and dissipated by resistor 3 calculate the supply pf

```

1 clear all;
2 clc;
3 V=230;
4 E=150;
5 R=8;
6 th1=asind(E/(sqrt(2)*V));
7 I_o=(1/(2*pi*R))*(2*sqrt(2)*230*cosd(th1)-E*(pi-2*
    th1*pi/180));
8 printf("avg charging curent=%0.4 f A",I_o);
9
10 P=E*I_o;    printf("\npower supplied to the battery=
    %0.2 f W",P);
11 I_or=sqrt((1/(2*pi*R^2))*((V^2+E^2)*(pi-2*th1*pi
    /180)+V^2*sind(2*th1)-4*sqrt(2)*V*E*cosd(th1)));
12 P_r=I_or^2*R;    printf("\npower dissipated by the
    resistor=%0.3 f W",P_r);
13
14 pf=(P+P_r)/(V*I_or);    printf("\nsupply pf=%0.3 f",pf
    );

```

Scilab code Exa 6.3 1 to find value of charging current 2 power supplied to battery and dissipated by resistor 3 calculate the supply pf

```

1 clear all;
2 clc;
3 V=230;
4 E=150;
5 R=8;
6 a=35;
7 th1=asind(E/(sqrt(2)*V));
8 th2=180-th1;
9 I_o=(1/(2*pi*R))*(sqrt(2)*230*(cosd(a)-cosd(th2))-E
    *((th2-a)*pi/180));
10 printf("avg charging curent=%0.4 f A",I_o);
11

```

```

12 P=E*I_o;    printf("\npower supplied to the battery=
    %.2f W",P);
13 I_or=sqrt((1/(2*pi*R^2))*((V^2+E^2)*((th2-a)*pi
    /180)-(V^2/2)*(sind(2*th2)-sind(2*a))-2*sqrt(2)*V
    *E*(cosd(a)-cosd(th2))));
14 P_r=I_or^2*R;    printf("\npower dissipated by the
    resistor=%.2f W",P_r);
15
16 pf=(P+P_r)/(V*I_or);    printf("\nsupply pf=%.4f",pf
    );
17 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 6.4 to find ckt turn off time avg output voltage and avg load current

```

1 clear all;
2 clc;
3 B=210;
4 f=50;    //Hz
5 w=2*pi*f;
6 a=40;    //firing angle
7 V=230;
8 disp(" for R=5ohm and L=2mH");
9 R=5;
10 L=2*10^-3;
11 t_c=(360-B)*pi/(180*w);    printf(" ckt turn off
    time=%.3f msec",t_c*1000);
12 V_o=(sqrt(2)*230/(2*pi))*(cosd(a)-cosd(B));
    printf("\navg output voltage=%.3f V",V_o);
13 I_o=V_o/R;    printf("\navg output current=%.4f A",
    I_o);
14
15 disp(" for R=5ohm, L=2mH and E=110V");

```

```

16 E=110;
17 R=5;
18 L=2*10^-3;
19 th1=asind(E/(sqrt(2)*V));
20 t_c=(360-B+th1)*%pi/(180*w);    printf("ckt turn off
    time=%0.3f msec",t_c*1000);
21 V_o=(sqrt(2)*230/(2*%pi))*(cosd(a)-cosd(B));
    printf("\navg output voltage=%0.3f V",V_o);
22 I_o=(1/(2*%pi*R))*(sqrt(2)*230*(cosd(a)-cosd(B))-E
    *((B-a)*%pi/180));    printf("\navg output
    current=%0.4f A",I_o);
23 V_o=R*I_o+E;    printf("\navg output voltage=%0.3f V"
    ,V_o);

```

Scilab code Exa 6.5 to determine 1 rectification efficiency 2 form factor 3 voltage ripple factor 4 transformer utilisation factor 5 PIV of thyristor

```

1 clear all;
2 clc;
3 V_s=230;
4 f=50;
5 R=10;
6 a=60;
7 V_m=(sqrt(2)*V_s);
8 V_o=V_m/(2*%pi)*(1+cosd(a));
9 I_o=V_o/R;
10 V_or=(V_m/(2*sqrt(%pi)))*sqrt((%pi-a*%pi/180)+.5*
    sind(2*a));
11 I_or=V_or/R;
12 P_dc=V_o*I_o;
13 P_ac=V_or*I_or;
14 RE=P_dc/P_ac;    printf("rectification efficiency=%0
    .4f",RE);
15 FF=V_or/V_o;    printf("\nform factor=%0.3f",FF);
16 VRF=sqrt(FF^2-1);    printf("\nvoltage ripple factor

```

```

    =%.4 f" ,VRF);
17 TUF=P_dc/(V_s*I_or);    printf("\nt/f utilisation
    factor=%.4 f" ,TUF);
18 PIV=V_m;    printf("\nPIV of thyristor=%.2 f V" ,PIV);

```

Scilab code Exa 6.6 to find power handled by mid pt convertor and single phase bridge convertor

```

1 clear all;
2 clc;
3 V=1000;
4 fos=2.5;    //factor of safety
5 I_TAV=40;
6 disp("for mid pt convertor");
7 V_m=V/(2*fos);
8 P=(2*V_m/%pi)*I_TAV;    printf("power handled=%.3 f
    kW" ,P/1000);
9 disp("for bridge convertor");
10 V_m=V/(fos);
11 P=(2*V_m/%pi)*I_TAV;    printf("power handled=%.3 f
    kW" ,P/1000);

```

Scilab code Exa 6.7 compute firing angle delay and pf

```

1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 R=.4;
6 I_o=10;
7 I_or=I_o;
8 E=120;

```



```

9 a=acosd((E+I_o*R)*%pi/(2*V_m));      printf(" firing
    angle delay=%0.2 f deg",a);
10 pf=(E*I_o+I_or^2*R)/(V_s*I_or);      printf("\n pf=%0.4 f
    ",pf);
11
12 E=-120;
13 a=acosd((E+I_o*R)*%pi/(2*V_m));      printf("\n firing
    angle delay=%0.2 f deg",a);
14 pf=(-E*I_o-I_or^2*R)/(V_s*I_or);      printf("\n pf=%0.4 f
    ",pf);

```

Scilab code Exa 6.9 to find avg output current and power delivered

```

1 clear all;
2 clc;
3 V_s=230;
4 f=50;
5 a=45;
6 R=5;
7 E=100;
8 V_o=((sqrt(2)*V_s)/(2*%pi))*(3+cosd(a));
9 I_o=(V_o-E)/R;      printf(" avg o/p current=%0.3 f A",
    I_o);
10 P=E*I_o;      printf("\n power delivered to battery=%0.4
    f kW",P/1000);

```

Scilab code Exa 6.10 to find avg value of load current and new value under given changed conditions

```

1 clear all;
2 clc;
3 V_s=230;
4 f=50;

```

```

5 a=50;
6 R=6;
7 E=60;
8 V_o=((sqrt(2)*2*V_s)/(pi))*cosd(a);
9 I_o=(V_o-E)/R;    printf("avg o/p current=%0.3 f A",
    I_o);
10
11 //ATQ after applying the conditions
12 V_o=((sqrt(2)*V_s)/(pi))*cosd(a);
13 I_o=(V_o-E)/R;    printf("\navg o/p current after
    change=%0.4 f A",I_o);

```

Scilab code Exa 6.11 to calculate the input and output performance parameters for full conductor

```

1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 a=45;
6 R=10;
7 V_o=(2*V_m/pi)*cosd(a);
8 I_o=V_o/R;
9 V_or=V_m/sqrt(2);
10 I_or=I_o;
11 P_dc=V_o*I_o;
12 P_ac=V_or*I_or;
13 RE=P_dc/P_ac;    printf("rectification efficiency=%0
    .4 f",RE);
14 FF=V_or/V_o;    printf("\nform factor=%0.4 f",FF);
15 VRF=sqrt(FF^2-1);    printf("\nvoltage ripple factor
    =%0.4 f",VRF);
16 I_s1=2*sqrt(2)*I_o/pi;
17 DF=cosd(a);
18 CDF=.90032;

```

```

19 pf=CDF*DF;      printf("\n pf=%0.5 f",pf);
20 HF=sqrt((1/CDF^2)-1);      printf("\n HF=%0.5 f",HF);
21 printf("\n active power=%0.2 f W",P_dc);
22 Q=2*V_m*I_o*sind(a)/%pi;      printf("\n reactive power
    =%0.3 f Var",Q);

```

Scilab code Exa 6.12 to calculate the input and output performance parameters for single phase semi conductor

```

1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 a=45;
6 R=10;
7 V_o=(V_m/%pi)*(1+cosd(a));
8 I_o=V_o/R;
9 V_or=V_s*sqrt((1/%pi)*((%pi-a*%pi/180)+sind(2*a)/2))
;
10 I_or=I_o;
11 P_dc=V_o*I_o;
12 P_ac=V_or*I_or;
13 RE=P_dc/P_ac;      printf("rectification efficiency=%0
    .4 f",RE);
14 FF=V_or/V_o;      printf("\n form factor=%0.3 f",FF);
15 VRF=sqrt(FF^2-1);      printf("\n voltage ripple factor
    =%0.3 f",VRF);
16 I_s1=2*sqrt(2)*I_o*cosd(a/2)/%pi;
17 DF=cosd(a/2);      printf("\n DF=%0.4 f",DF);
18 CDF=2*sqrt(2)*cosd(a/2)/sqrt(%pi*(%pi-a*%pi/180));
    printf("\n CDF=%0.4 f",CDF);
19 pf=CDF*DF;      printf("\n pf=%0.4 f",pf);
20 HF=sqrt((1/CDF^2)-1);      printf("\n HF=%0.4 f",HF);
21 printf("\n active power=%0.3 f W",P_dc);
22 Q=V_m*I_o*sind(a)/%pi;      printf("\n reactive power=%0

```

```

    .2 f Var",Q);
23 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 6.13 determine 1 firing angle 2 avg and rms values of load current 3 rectification efficiency

```

1 clear all;
2 clc;
3 V_s=230;
4 R=10;
5 V_ml=sqrt(2)*V_s;
6 V_om=3*V_ml/(2*pi);
7 V_o=V_om/2;
8 th=30;
9 a=acosd((2*pi*sqrt(3)*V_o/(3*V_ml)-1))-th;
    printf("delay angle=%0.1f deg",a);
10 I_o=V_o/R;    printf("\navg load current=%0.3f A",I_o
    );
11 V_or=V_ml/(2*sqrt(pi))*sqrt((5*pi/6-a*pi/180)+.5*
    sind(2*a+2*th));
12 I_or=V_or/R;    printf("\nrms load current=%0.3f A",
    I_or);
13 RE=V_o*I_o/(V_or*I_or);    printf("\nrectification
    efficiency=%0.4f",RE);

```

Scilab code Exa 6.15 to calculate 1 avg value of load volatage 2 avg and rms current and PIV 3 avg power dissipation in each thyristor

```

1 clear all;
2 clc;
3 V=400;

```

```

4 V_m1=sqrt(2)*V;
5 v_T=1.4;
6 disp("for firing angle = 30deg");
7 a=30;
8 V_o=3*V_m1/(2*pi)*cosd(a)-v_T;    printf(" avg
    output voltage=%0.3 f V",V_o);
9 disp("for firing angle = 60deg");
10 a=60;
11 V_o=3*V_m1/(2*pi)*cosd(a)-v_T;    printf(" avg
    output voltage=%0.2 f V",V_o);
12
13 I_o=36;
14 I_TA=I_o/3;    printf("\navg current rating=%0.0 f A",
    I_TA);
15 I_Tr=I_o/sqrt(3);    printf("\nrms current rating=%0
    .3 f A",I_Tr);
16 printf("\nPIV of SCR=%0.1 f V",V_m1);
17
18 P=I_TA*v_T;    printf("\npower dissipated=%0.1 f W",P)
    ;

```

Scilab code Exa 6.17 to compute firing angle delay and supply pf

```

1 clear all;
2 clc;
3 E=200;
4 I_o=20;
5 R=.5;
6 V_o=E+I_o*R;
7 V_s=230;
8 V_m1=sqrt(2)*V_s;
9 a=acosd(V_o*pi/(3*V_m1));    printf(" firing angle
    delay=%0.3 f deg",a);
10 th=120;
11 I_s=sqrt((1/pi)*I_o^2*th*pi/180);

```

```

12 P=E*I_o+I_o^2*R;
13 pf=P/(sqrt(3)*V_s*I_s);    printf("\npf=%0.3 f",pf);
14
15 V_o=E-I_o*R;
16 a=acosd(-V_o*%pi/(3*V_ml));    printf("\nfiring
    angle delay=%0.2 f deg",a);

```

Scilab code Exa 6.18 to find commutation time and reverse voltage across SCR

```

1 clear all;
2 clc;
3 V=230;
4 f=50;
5 w=2*%pi*f;
6 disp("for firing angle delay=0deg");
7 a=0;
8 t_c=(4*%pi/3-a*%pi/180)/w;    printf("commutation
    time=%0.2 f ms",t_c*1000);
9 printf("\npeak reverse voltage=%0.2 f V",sqrt(2)*V);
10
11 disp("for firing angle delay=30deg");
12 a=30;
13 t_c=(4*%pi/3-a*%pi/180)/w;    printf("commutation
    time=%0.2 f ms",t_c*1000);
14 printf("\npeak reverse voltage=%0.2 f V",sqrt(2)*V);

```

Scilab code Exa 6.19 to find the magnitude of per phase input supply voltage

```

1 clear all;
2 clc;
3 a=30;

```

```

4 R=10;
5 P=5000;
6 V_s=sqrt(P*R*2*%pi/(2*3)/(%pi/3+sqrt(3)*cosd(2*a)/2)
    );
7 V_ph=V_s/sqrt(3);    printf("per phase voltage , V_ph
    =%.3f V",V_ph);
8 I_or=sqrt(P*R);
9 V_s=I_or*%pi/(sqrt(2)*3*cosd(a));
10 V_ph=V_s/sqrt(3);
11 printf("\nfor constant load current");
12 printf("\nV_ph=%.2f V",V_ph);

```

Scilab code Exa 6.20 to find magnitude of input per phase supply voltage

```

1 clear all;
2 clc;
3 a=30;
4 R=10;
5 P=5000;
6 V_s=sqrt(P*R*4*%pi/(2*3)/(2*%pi/3+sqrt(3)*(1+cosd(2*
    a))/2));
7 V_ph=V_s/sqrt(3);    printf("per phase voltage , V_ph
    =%.3f V",V_ph);
8 I_or=sqrt(P*R);
9 V_s=I_or*2*%pi/(sqrt(2)*3*(1+cosd(a)));
10 V_ph=V_s/sqrt(3);
11 printf("\nfor constant load current");
12 printf("\nV_ph=%.2f V",V_ph);

```

Scilab code Exa 6.21 to find magnitude of input per phase supply voltage

```

1 clear all;
2 clc;

```

```

3 a=90;
4 R=10;
5 P=5000;
6 V_s=sqrt(P*R*4*%pi/(2*3)/((%pi-%pi/2)+(sind(2*a))/2)
    );
7 V_ph=V_s/sqrt(3);    printf("per phase voltage , V_ph
    =%.2 f V",V_ph);
8 I_or=sqrt(P*R);
9 V_s=I_or*2*%pi/(sqrt(2)*3*(1+cosd(a)));
10 V_ph=V_s/sqrt(3);
11 printf("\nfor constant load current");
12 printf("\nV_ph=%.1 f V",V_ph);

```

Scilab code Exa 6.22 to compute firing angle delay and supply pf

```

1 clear all;
2 clc;
3 E=200;
4 I_o=20;
5 R=.5;
6 V_o=E+I_o*R;
7 V_s=230;
8 V_m1=sqrt(2)*V_s;
9 a=acosd(V_o*2*%pi/(3*V_m1)-1);    printf("firing
    angle delay=%.2 f deg",a);
10 a1=180-a;
11 I_sr=sqrt((1/%pi)*I_o^2*(a1*%pi/180));
12 P=V_o*I_o;
13 pf=P/(sqrt(3)*V_s*I_sr);    printf("\n pf=%.4 f",pf);

```

Scilab code Exa 6.23 to calculate rectification efficiency TUF and input power factor


```

1 clear all;
2 clc;
3 V_s=400;
4 f=50;
5 I_o=15;
6 a=45;
7 I_TA=I_o*120/360;
8 I_Tr=sqrt(I_o^2*120/360);
9 I_sr=sqrt(I_o^2*120/180);
10 V_ml=sqrt(2)*V_s;
11 V_o=3*V_ml*cosd(a)/%pi;
12 V_or=V_ml*sqrt((3/(2*%pi))*(%pi/3+sqrt(3/2)*cosd(2*a
    )));
13 I_or=I_o;
14 P_dc=V_o*I_o;
15 P_ac=V_or*I_or;
16 RE=P_dc/P_ac;    printf("rectification efficiency=%
    .5 f",RE);
17 VA=3*V_s/sqrt(3)*I_sr;
18 TUF=P_dc/VA;    printf("\nTUF=%0.4 f",TUF);
19 pf=P_ac/VA;    printf("\ninput pf=%0.3 f",pf);

```

Scilab code Exa 6.24 to find DF CDF THD and pf and to calculate the active and reactive input powers

```

1 clear all;
2 clc;
3 I=10;
4 a=45;
5 V=400;
6 f=50;
7 DF=cosd(a);
8 printf("DF=%0.3 f",DF);
9 I_o=10;
10 I_s1=4*I_o/(sqrt(2)*%pi)*sin(%pi/3);

```

```

11 I_sr=I_o*sqrt(2/3);
12 I_o=1; //suppose
13 CDF=I_s1/I_sr; printf("\nCDF=%0.3 f",CDF);
14 THD=sqrt(1/CDF^2-1); printf("\nTHD=%0.5 f",THD);
15 pf=CDF*DF; printf("\nPF=%0.4 f",pf);
16 P=(3*sqrt(2)*V*cosd(a)/%pi)*I; printf("\nactive
    power=%0.2 f W",P);
17 Q=(3*sqrt(2)*V*sind(a)/%pi)*I; printf("\nreactive
    power=%0.2 f Var",Q);

```

Scilab code Exa 6.25 calculate the power delivered to load and input pf

```

1 //calculate the power delivered to load and i/p pf
2
3 clc;
4 disp("for firing angle=30deg");
5 a=30;
6 V=400;
7 V_ml=sqrt(2)*V;
8 V_o=3*V_ml*cosd(a)/%pi;
9 E=350;
10 R=10;
11 I_o=(V_o-E)/R;
12 I_or=I_o;
13 P=V_o*I_o; printf("power delivered to load=%0.2 f W
    ",P);
14 I_sr=I_o*sqrt(2/3);
15 VA=3*V/sqrt(3)*I_sr;
16 pf=P/VA; printf("\n pf=%0.4 f",pf);
17
18 disp("for firing advance angle=60deg");
19 a=180-60;
20 V=400;
21 V_ml=sqrt(2)*V;
22 V_o=3*V_ml*cosd(a)/%pi;

```

```

23 E=-350;
24 R=10;
25 I_o=(V_o-E)/R;
26 I_or=I_o;
27 P=-V_o*I_o;    printf("power delivered to load=%0.2 f
    W",P);
28 I_sr=I_o*sqrt(2/3);
29 VA=3*V/sqrt(3)*I_sr;
30 pf=P/VA;    printf("\n pf=%0.4 f",pf);
31 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 6.26 calculate overlap angle for different firing angles

```

1 clear all;
2 clc;
3 a=0;
4 u=15;
5 i=cosd(a)-cosd(a+u);
6 disp("for firing angle=30deg");
7 a=30;
8 u=acosd(cosd(a)-i)-a;    printf("overlap angle=%0.1 f
    deg",u);
9 disp("for firing angle=45deg");
10 a=45;
11 u=acosd(cosd(a)-i)-a;    printf("overlap angle=%0.1 f
    deg",u);
12 disp("for firing angle=60deg");
13 a=60;
14 u=acosd(cosd(a)-i)-a;    printf("overlap angle=%0.2 f
    deg",u);

```

Scilab code Exa 6.28 to calculate firing angle delay and overlap angle

```
1 clear all;
2 clc;
3 E=400;
4 I_o=20;
5 R=1;
6 V_o=E+I_o*R;
7 f=50;
8 w=2*%pi*f
9 L=.004;
10 V=230; //per phase voltage
11 V_m1=sqrt(6)*V;
12 a=acosd(%pi/(3*V_m1)*(V_o+3*w*L*I_o/%pi)); printf
    (" firing angle delay=%.3 f deg",a);
13 u=acosd(%pi/(3*V_m1)*(V_o-3*w*L*I_o/%pi))-a;
    printf("\noverlap angle=%.2 f deg",u);
14 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.
```

Scilab code Exa 6.29 to calculate firing angle firing angle delay and overlap angle

```
1 clear all;
2 clc;
3 V=400;
4 f=50;
5 w=2*%pi*f;
6 R=1;
7 E=230;
8 I=15;
9 V_o=-E+I*R;
10 V_m1=sqrt(2)*V;
11 a=acosd(V_o*2*%pi/(3*V_m1)); printf(" firing angle
```

```

    =%.3 f deg",a);
12 L=0.004;
13 a=acosd((2*%pi)/(3*V_m1)*(V_o+3*w*L*I/(2*%pi)));
    printf("\nfiring angle delay=%.3 f deg",a);
14 u=acosd(cosd(a)-3*f*L*I/V_m1)-a;    printf("\
    noverlap angle=%.3 f deg",u);
15 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 6.31 calculate the peak value of circulating current

```

1 clear all;
2 clc;
3 V=230; //per phase
4 V_m1=sqrt(3)*sqrt(2)*V;
5 f=50;
6 w=2*%pi*f;
7 a1=60;
8 L=0.015;
9 i_cp=(sqrt(3)*V_m1/(w*L))*(1-sind(a1));    printf("
    circulating current=%.4 f A",i_cp);

```

Scilab code Exa 6.32 to determine 1 avg output voltage 2 avg output current 3 avg and rms values of thyristor currents 4 pf

```

1 clear all;
2 clc;
3 V=230;
4 V_m=sqrt(2)*V;
5 a=30;
6 V_o=2*V_m*cosd(a)/%pi;    printf(" avg o/p voltage=%.
    3 f V",V_o);

```

```

7 R=10;
8 I_o=V_o/R;    printf("\navg o/p current=%0.2 f A",I_o)
;
9 I_TA=I_o*pi/(2*pi);    printf("\navg value of
thyristor current=%0.3 f A",I_TA);
10 I_Tr=sqrt(I_o^2*pi/(2*pi));    printf("\nrms value
of thyristor current=%0.2 f A",I_Tr);
11 I_s=sqrt(I_o^2*pi/(%pi));
12 I_o=I_s;
13 pf=(V_o*I_o/(V*I_s));    printf("\n pf=%0.4 f",pf);
14 //Answers have small variations from that in the
book due to difference in the rounding off of
digits.

```

Scilab code Exa 6.33 to determine 1 avg output voltage 2 angle of overlap
3 pf

```

1 clear all;
2 clc;
3 V=230;
4 V_m=sqrt(2)*V;
5 a=30;
6 L=.0015;
7 V_o=2*V_m*cosd(a)/%pi;
8 R=10;
9 I_o=V_o/R;
10 f=50;
11 w=2*pi*f;
12 V_ox=2*V_m*cosd(a)/%pi-w*L*I_o/%pi;    printf(" avg o
/p voltage=%0.3 f V",V_ox);
13 u=acosd(cosd(a)-I_o*w*L/V_m)-a;    printf("\n angle
of overlap=%0.3 f deg",u);
14 I=I_o;
15 pf=V_o*I_o/(V*I);    printf("\n pf=%0.4 f",pf);
16 //Answers have small variations from that in the

```

book due to difference in the rounding off of digits.

Scilab code Exa 6.34 calculate the generator mean voltage

```
1 clear all;
2 clc;
3 V=415;
4 V_m1=sqrt(2)*V;
5 a1=35; //firing angle advance
6 a=180-a1;
7 I_o=80;
8 r_s=0.04;
9 v_T=1.5;
10 X_l=.25; //reactance=w*L
11 E=-3*V_m1*cosd(a)/%pi+2*I_o*r_s+2*v_T+3*X_l*I_o/%pi;
    printf("mean generator voltage=%0.3f V",E);
```

Scilab code Exa 6.35 find the mean value of E

```
1 clear all;
2 clc;
3 V=415;
4 V_m1=sqrt(2)*V;
5 R=.2;
6 I_o=80;
7 r_s=0.04;
8 v_T=1.5;
9 X_l=.25; //reactance=w*L
10
11 disp("when firing angle=35deg");
12 a=35;
```

```

13 E=-(-3*V_m1*cosd(a)/%pi+I_o*R+2*I_o*r_s+2*v_T+3*X_l*
    I_o/%pi);    printf("mean generator voltage=%%.3 f
    V",E);
14 disp("when firing angle advance=35deg");
15 a1=35;
16 a=180-a1;
17 E=(-3*V_m1*cosd(a)/%pi+I_o*R+2*I_o*r_s+2*v_T+3*X_l*
    I_o/%pi);    printf("mean generator voltage=%%.3 f
    V",E);

```

Scilab code Exa 6.36 to find avg current through battery

```

1 clear all;
2 clc;
3 R=5;
4 V=230;
5 V_mp=sqrt(2)*V;
6 a=30;
7 E=150;
8 B=180-asind(E/V_mp);
9 I_o=(3/(2*%pi*R))*(V_mp*(cosd(a+30)-cosd(B))-E*((B-a
    -30)*%pi/180));
10 printf("avg current flowing=%%.2 f A",I_o);

```

Scilab code Exa 6.37 to determine 1 avg output voltage 2 avg output current 3 avg and rms values of thyristor currents 4 avg and rms values of diodes currents 5 i pf 6 kct turn off time

```

1 clear all;
2 clc;
3 a=30;
4 V=230;
5 V_m=sqrt(2)*V;

```



```

6 V_o=V_m*(1+cosd(a))/%pi;    printf(" avg o/p voltage=
    %.3f V",V_o);
7 E=100;
8 R=10;
9 I_o=(V_o-E)/R;    printf("\navg o/p current=%0.2f A",
    I_o);
10 I_TA=I_o*%pi/(2*%pi);    printf("\navg value of
    thyristor current=%0.2f A",I_TA);
11 I_Tr=sqrt(I_o^2*%pi/(2*%pi));    printf("\nrms value
    of thyristor current=%0.3f A",I_Tr);
12 printf("\navg value of diode current=%0.2f A",I_TA);
13 printf("\nrms value of diode current=%0.3f A",I_Tr);
14 I_s=sqrt(I_o^2*(1-a/180)*%pi/(%pi));
15 I_or=I_o;
16 P=E*I_o+I_or^2*R;
17 pf=(P/(V*I_s));    printf("\n pf=%0.4f",pf);
18 f=50;
19 w=2*%pi*f;
20 t_c=(1-a/180)*%pi/w;    printf("\ncircuit turn off
    time=%0.2f ms",t_c*1000);

```

Scilab code Exa 6.38 to calculate peak value of circulating currents and of both convertors

```

1 clear all;
2 clc;
3 V=230;
4 V_m=sqrt(2)*V;
5 L=0.05;
6 f=50;
7 w=2*%pi*f;
8 a=30;
9 i_cp=2*V_m*(1-cosd(a))/(w*L);    printf(" peak value
    of circulating current=%0.3f A",i_cp);
10 R=30;

```

```
11 i_l=V_m/R;
12 i1=i_cp+i_l;    printf("\npeak value of current in
    convertor 1=%0.3 f A",i1);
13 i2=i_cp;    printf("\npeak value of current in
    convertor 2=%0.3 f A",i2);
```

Scilab code Exa 6.39 to estimate triggering angle for no current transients and for worst transients

```
1 clear all;
2 clc;
3 f=50;
4 w=2*%pi*f;
5 R=5;
6 L=0.05;
7 disp("for no current transients");
8 phi=atand(w*L/R);    printf("triggering angle=%0.2 f
    deg",phi);
9 disp("for worst transients");
10 phi=90+atand(w*L/R);    printf("triggering angle=%0.2
    f deg",phi);
```

Chapter 7

Choppers

Scilab code Exa 7.2 to calculate 1 avg and rms values of output voltage
2 chopper efficiency

```
1 clear all;
2 clc;
3 a=.4; //duty cycle ,a=T_on/T
4 V_s=230;
5 R=10;
6 V=a*(V_s-2); printf("avg o/p voltage=%0.1f V",V);
7 V_or=sqrt(a*(V_s-2)^2); printf("\nrms value of o/
  p voltage=%0.1f V",V_or);
8 P_o=V_or^2/R;
9 P_i=V_s*V/R;
10 n=P_o*100/P_i; printf("\nchopper efficiency in
  percentage=%0.2f",n);
```

Scilab code Exa 7.3 to compute pulse width of output voltage and avg
value of new output voltage

```
1 clear all;
```

```

2  clc;
3  V_i=220;
4  V_o=660;
5  a=1-V_i/V_o;
6  T_on=100;    //microsecond
7  T=T_on/a;
8  T_off=T-T_on;    printf("pulse width of o/p voltage=
    %.0f us",T_off);
9
10 T_off=T_off/2;
11 T_on=T-T_off;
12 a=T_on/T;
13 V_o=V_i/(1-a);    printf("\nnew o/p voltage=%.0f V",
    V_o);

```

Scilab code Exa 7.4 to find time ratio for chopper

```

1  clear all;
2  clc;
3  I_1=12;
4  I_2=16;
5  I_0=(I_1+I_2)/2;
6  R=10;
7  V_0=I_0*R;
8  V_s=200;
9  a=V_0/V_s;
10 r=a/(1-a);    printf("time ratio (T_on/T_off)=%.3f",r
    );

```

Scilab code Exa 7.5 to compute pulse width of output voltage and avg value of new output voltage

```

1  clear all;

```

```

2  clc;
3  V_o=660;
4  V_s=220;
5  a=(V_o/V_s)/(1+(V_o/V_s));
6  T_on=120;
7  T=T_on/a;
8  T_off=T-T_on;    printf("pulse width o/p voltage=%0.0
    f us",T_off);
9
10 T_off=3*T_off;
11 T_on=T-T_off;
12 a=T_on/(T_on+T_off);
13 V_o=V_s*(a/(1-a));    printf("\nnew o/p voltage=%0.2 f
    V",V_o);

```

Scilab code Exa 7.11 1 to find whether load current is cont 2 calculate value of avg op current 3 to find its max and min values 4 find rms values of various harmonics 5 compute avg value of supply current 6 ip power power absorbed by load counter emf and power loss in res

```

1  clear all;
2  clc;
3  R=1;
4  L=.005;
5  T_a=L/R;
6  T=2000*10^-6;
7  E=24;
8  V_s=220;
9  T_on=600*10^-6;
10 a=T_on/T;
11 a1=(T_a/T)*log(1+(E/V_s)*((exp(T/T_a))-1));
12 if(a1<a)
13     disp("load current in continuous");
14 else
15     disp("load current in discont.");

```

```

16 end
17 I_o=(a*V_s-E)/R;    printf(" avg o/p current=%0.0 f A",
    I_o);
18 I_mx=(V_s/R)*((1-exp(-T_on/T_a))/(1-exp(-T/T_a)))-E/
    R;    printf("\nmax value of steady current=%0.2 f
    A", I_mx);
19 I_mn=(V_s/R)*((exp(T_on/T_a)-1)/(exp(T/T_a)-1))-E/R;
    printf("\nmin value of steady current=%0.3 f A"
    , I_mn);
20
21 f=1/T;
22 w=2*%pi*f;
23 I1=(2*V_s/(sqrt(2)*%pi)*sind(180*a))/(sqrt(R^2+(w*L
    ^2)));    printf("\nfirst harmonic current=%0.4 f A"
    , I1);
24 I2=(2*V_s/(2*sqrt(2)*%pi)*sind(2*180*a))/(sqrt(R^2+(
    w*L*2)^2));    printf("\nsecond harmonic current=
    %0.4 f A", I2);
25 I3=(2*V_s/(3*sqrt(2)*%pi)*sind(3*180*a))/(sqrt(R^2+(
    w*L*3)^2));    printf("\nthird harmonic current=%0
    .5 f A", I3);
26
27 I_TAV=a*(V_s-E)/R-L*(I_mx-I_mn)/(R*T);    printf("\
    navg supply current=%0.4 f A", I_TAV);
28
29 P1=I_TAV*V_s;
30 printf("\ni/p power=%0.2 f W", P1);
31 P2=E*I_o;
32 printf("\npower absorbed by load emf=%0.0 f W", P2);
33 printf("\npower loss in resistor=%0.2 f W", P1-P2);
34 I_or=sqrt(I_o^2+I1^2+I2^2+I3^2);
35 printf("\nrms value of load current=%0.3 f A", I_or);

```

Scilab code Exa 7.12 1 to find whether load current is cont or not 2 to calculate avg output voltage and avg current 3 to compute max and min

values of output current

```
1 clear all;
2 clc;
3 R=1;
4 L=.001;
5 V_s=220;
6 E=72;
7 f=500;
8 T_on=800*10^-6;
9 T_a=L/R;
10 T=1/f;
11 m=E/V_s;
12 a=T_on/T;
13 a1=(T_a/T)*log(1+m*(exp(-T/T_a)-1));
14 if(a1>a)
15     disp("load current is continuous");
16 else
17     disp("load current is discontinuous");
18 end
19 t_x=T_on+L*log(1+((V_s-E)/272)*(1-exp(-T_on/T_a)));
20 //Value of t_x wrongly calculated in the book so ans
    of V_o and I_o varies
21 V_o=a*V_s+(1-t_x/T)*E;    printf("avg o/p voltage=%g
    .2f V",V_o);
22 I_o=(V_o-E)/R;    printf("\navg o/p current=%g.2f A",
    I_o);
23 printf("\nmin value of load current=%g.0f A",0);
24 I_mx=(V_s-E)/R*(1-exp(-T_on/T_a));    printf("\nmax
    value of load current=%g.1f A",I_mx);
```

Scilab code Exa 7.13 to find the chopping freq

```
1 clear all;
2 clc;
```

```

3 a=.2;
4 V_s=500;
5 E=a*V_s;
6 L=0.06;
7 I=10;
8 T_on=(L*I)/(V_s-E);
9 f=a/T_on;    printf("chopping freq=%0.2 f Hz",f);

```

Scilab code Exa 7.14 to find the value of external inductance to be added in series

```

1 clear all;
2 clc;
3 a=.5;
4 pu=.1; //pu ripple
5 //x=T/T_a
6 //y=exp(-a*x)
7 y=(1-pu)/(1+pu);
8 //after solving
9 x=log(1/y)/a;
10 f=1000;
11 T=1/f;
12 T_a=T/x;
13 R=2;
14 L=R*T_a;
15 Li=.002;
16 Le=L-Li;    printf("external inductance=%0.3 f mH",Le
    *1000);

```

Scilab code Exa 7.15 1 to calculate min and and max value of load current
2 max value of ripple current 3 avg and rms values of load current 4 rms value
of chopper current


```

1 clear all;
2 clc;
3 R=10;
4 L=.015;
5 T_a=L/R;
6 f=1250;
7 T=1/f;
8 a=.5;
9 T_on=a*T;
10 V_s=220;
11 I_mx=(V_s/R)*((1-exp(-T_on/T_a))/(1-exp(-T/T_a)));
    printf("max value of load current=%0.3f A",I_mx
    );
12 I_mn=(V_s/R)*((exp(T_on/T_a)-1)/(exp(T/T_a)-1));
    printf("\nmin value of load current=%0.2f A",I_mn)
    ;
13 dI=I_mx-I_mn;    printf("\nmax value of ripple
    current=%0.3f A",dI);
14 V_o=a*V_s;
15 I_o=V_o/R;    printf("\navg value of load current=%0
    .2f A",I_o);
16 I_or=sqrt(I_mx^2+dI^2/3+I_mx*dI);    printf("\nrms
    value of load current=%0.3f A",I_or);
17 I_chr=sqrt(a)*I_or;    printf("\nrms value of
    chopper current=%0.3f A",I_chr);
18 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 7.17 to find the time for which current flows

```

1 clear all;
2 clc;
3 L=0.0016;
4 C=4*10^-6;

```

```

5 w=1/sqrt(L*C);
6 t=%pi/w;    printf("time for which current flows=%%.2
    f us",t*10^6);

```

Scilab code Exa 7.18 to calculate values of commutating capacitor and commutating inductor

```

1 clear all;
2 clc;
3 t_q=20*10^-6;
4 dt=20*10^-6;
5 t_c=t_q+dt;
6 I_0=60;
7 V_s=60;
8 C=t_c*I_0/V_s;    printf("value of commutating
    capacitor=%%.0f uF",C*10^6);
9 L1=(V_s/I_0)^2*C;
10 L2=(2*t_c/%pi)^2/C;
11 if(L1>L2)
12     printf("\nvalue of commutating inductor=%%.0f uH"
        ,L1*10^6);
13 else
14     printf("\nvalue of commutating inductor=%%.0f uH"
        ,L2*10^6);
15 end

```

Scilab code Exa 7.19 to calculate the value of commutating component C and commutating component L

```

1 clear all;
2 clc;
3 t=100*10^-6;
4 R=10;

```

```

5 //V_s*(1-2*exp(-t/(R*C)))=0
6 C=-t/(R*log(1/2));    printf("Value of comutating
    component C=%0.3f uF",C*10^6);
7 disp("max permissible current through SCR is 2.5
    times load current");
8 L=(4/9)*C*R^2;    printf("value of comutating
    component L=%0.1f uH",L*10^6);
9 disp("max permissible current through SCR is 1.5
    times peak diode current");
10 L=(1/4)*C*R^2;    printf("value of comutating
    component L=%0.2f uH",L*10^6);

```

Scilab code Exa 7.20 to compute 1 effective on period 2 peak current through main and auxillary thyristor 3 their turn off times 4 total commutation intervals 5 capacitor voltage 6 time needed to recharge the capacitor

```

1 clear all;
2 clc;
3 T_on=800*10^-6;
4 V_s=220;
5 I_o=80;
6 C=50*10^-6;
7 T=T_on+2*V_s*C/I_o;    printf("effective on period=%0
    .0f us",T*10^6);
8
9 L=20*10^-6;
10 C=50*10^-6;
11 i_T1=I_o+V_s*sqrt(C/L);    printf("\npeak current
    through main thyristor=%0.2f A",i_T1);
12 i_TA=I_o;    printf("\npeak current through
    auxillary thyristor=%0.0f A",i_TA);
13
14 t_c=C*V_s/I_o;    printf("\nturn off time for main
    thyristor=%0.1f us",t_c*10^6);
15 t_c1=(%pi/2)*sqrt(L*C);    printf("\nturn off time

```

```

    for auxillary thyristor=%0.3f us",t_c1*10^6);
16
17 printf("\ntotal commutation interval=%0.0f us",2*t_c
    *10^6);
18
19 t=150*10^-6;
20 v_c=I_o*t/C-V_s;    printf("\ncapacitor voltage=%0.0f
    V",v_c);
21
22 printf("\ntime needed to recharge the capacitor=%0.0f
    us",2*V_s*C/I_o*10^6);

```

Scilab code Exa 7.21 to calculate values of 1 commutating components C and L and 2 min and max output voltages

```

1 clear all;
2 clc;
3 I_o=260;
4 V_s=220;
5 fos=2;//factor of safety
6 t_off=18*10^-6;
7 t_c=2*t_off;
8 C=t_c*I_o/V_s;    printf("Value of C=%0.3f uF",C
    *10^6);
9 L=(V_s/(.8*I_o))^2*C;    printf("\nvalue of L=%0.3f
    uH",L*10^6);
10
11 f=400;
12 a_mn=%pi*f*sqrt(L*C);
13 V_omn=V_s*(a_mn+2*f*t_c);    printf("\nmin value of
    o/p voltage=%0.3f V",V_omn);
14 V_omx=V_s;    printf("\nmax value of o/p voltage=%0.0
    f V",V_omx);

```

Scilab code Exa 7.22 calculate 1 value of commutating inductor and capacitor 2 max capacitor voltage 3 peak commutating current

```

1 clear all;
2 clc;
3 x=2;
4 t_q=30*10^-6;
5 dt=30*10^-6;
6 t_c=t_q+dt;
7 V_s=230;
8 I_o=200;
9 L=V_s*t_c/(x*I_o*(%pi-2*asin(1/x))); printf("
    value of commutating inductor=%0.3f uH",L*10^6);
10 C=x*I_o*t_c/(V_s*(%pi-2*asin(1/x))); printf("\
    nvalue of commutating capacitor=%0.3f uF",C*10^6);
11 V_cp=V_s+I_o*sqrt(L/C); printf("\npeak capacitor
    voltage=%0.0f V",V_cp);
12 I_cp=x*I_o; printf("\npeak commutataing current=%0
    .0f A",I_cp);
13
14 x=3;
15 L=V_s*t_c/(x*I_o*(%pi-2*asin(1/x))); printf("\
    nvalue of commutating inductor=%0.3f uH",L*10^6);
16 C=x*I_o*t_c/(V_s*(%pi-2*asin(1/x))); printf("\
    nvalue of commutating capacitor=%0.3f uF",C*10^6);
17 V_cp=V_s+I_o*sqrt(L/C); printf("\npeak capacitor
    voltage=%0.2f V",V_cp);
18 I_cp=x*I_o; printf("\npeak commutataing current=%0
    .0f A",I_cp);

```

Scilab code Exa 7.23 to compute 1 turn off time of main thyristor 2 total commutation interval 3 turn off time of auxillery thyristor

```

1 clear all;
2 clc;
3 V_s=230;
4 C=50*10^-6;
5 L=20*10^-6;
6 I_cp=V_s*sqrt(C/L);
7 I_o=200;
8 x=I_cp/I_o;
9 t_c=(%pi-2*asin(1/x))*sqrt(C*L);    printf("turn off
    time of main thyristor=%0.2f us",t_c*10^6);
10 th1=asind(1/x);
11 t=(5*%pi/2-th1*%pi/180)*sqrt(L*C)+C*V_s*(1-cosd(th1)
    )/I_o;    printf("\ntotal commutation interval=%0
    .3f us",t*10^6);
12 t=(%pi-th1*%pi/180)*sqrt(L*C);    printf("\nturn off
    time of auxillary thyristor=%0.3f us",t*10^6);

```

Scilab code Exa 7.24 to calculate min and max value of load current

```

1 clear all;
2 clc;
3 tc=.006;
4 R=10;
5 L=R*tc;
6 f=2000;
7 T=1/f;
8 V_o=50;
9 V_s=100;
10 a=V_o/V_s;
11 T_on=a*T;
12 T_off=T-T_on;
13 dI=V_o*T_off/L;
14 I_o=V_o/R;
15 I2=I_o+dI/2;    printf("max value of load current=%0
    .3f A",I2);

```

```
16 I1=I_o-dI/2;    printf("\nmin value of load current=
    %.3f A",I1);
```

Scilab code Exa 7.27 calculate 1 range of speed control 2 range of duty cycle

```
1 clear all;
2 clc;
3 I_a=30;
4 r_a=.5;
5 V_s=220;
6 a=I_a*r_a/V_s;    printf("min value of duty cycle=%
    .3f",a);
7 printf("\nmin Value of speed control=%0.0f rpm",0);
8 a=1;
9 printf("\nmax value of duty cycle=%0.0f",a);
10 k=.1; //V/rpm
11 N=(a*V_s-I_a*r_a)/k;    printf("\nmax value of speed
    control=%0.0f rpm",N);
```

Scilab code Exa 7.28 determine chopping freq and duty cycle

```
1 clear all;
2 clc;
3 V_t=72;
4 I_a=200;
5 r_a=0.045;
6 N=2500;
7 k=(V_t-I_a*r_a)/N;
8 E_a=k*1000;
9 L=.007;
10 Rm=.045;
11 Rb=0.065;
```

```

12 R=Rm+Rb;
13 T_a=L/R;
14 I_mx=230;
15 I_mn=180;
16 T_on=-T_a*log(-((V_t-E_a)/R-I_mx)/((I_mn)-(V_t-E_a)/
    R));
17 R=Rm;
18 T_a=L/R;
19 T_off=-T_a*log(-((-E_a)/R-I_mn)/((I_mx)-(-E_a)/R));
20 T=T_on+T_off;
21 f=1/T;    printf("chopping freq=%0.1f Hz",f);
22 a=T_on/T;    printf("\nduty cycle ratio=%0.3f",a);

```

Scilab code Exa 7.29 to determine the higher limit of current pulsation chopping freq and duty cycle ratio

```

1 clear all;
2 clc;
3 I_mx=425;
4 I_lt=180;    //lower limit of current pulsation
5 I_mn=I_mx-I_lt;
6 T_on=.014;
7 T_off=.011;
8 T=T_on+T_off;
9 T_a=.0635;
10 a=T_on/T;
11 V=(I_mx-I_mn*exp(-T_on/T_a))/(1-exp(-T_on/T_a));
12 a=.5;
13 I_mn=(I_mx-V*(1-exp(-T_on/T_a)))/(exp(-T_on/T_a));
14 T=I_mx-I_mn;    printf("higher limit of current
    pulsation=%0.0f A",T);
15 T=T_on/a;
16 f=1/T;    printf("\nchopping freq=%0.3f Hz",f);
17 printf("\nduty cycle ratio=%0.2f",a);
18 //Answers have small variations from that in the

```


book due to difference in the rounding off of
digits.

Chapter 8

Inverters

Scilab code Exa 8.3 to find the value of C for having load commutation

```
1 clear all;
2 clc;
3 T=.1*10^-3;
4 f=1/T;
5 k=15*10^-6; //k=th/w;
6 th=2*%pi*f*k;
7 X_l=10;
8 R=2;
9 X_c=R*tan(th)+X_l;
10 C=1/(2*%pi*f*X_c); printf("value of C=%.3f uF",C
    *10^6);
```

Scilab code Exa 8.4 to find the power delivered

```
1 clear all;
2 clc;
3 V_s=230;
4 V_01=2*V_s/(sqrt(2)*%pi);
```

```

5 R=2;
6 I_01=V_01/R;
7 P_d=I_01^2*R;    printf("power delivered to load=%0.1
      f W" ,P_d);
8 V=V_s/2;
9 I_s=sqrt(2)*I_01/%pi;
10 P_s=V*I_s;
11 printf("\npower delivered by both sources=%0.1f W" ,2*
      P_s);

```

Scilab code Exa 8.5 to find the power delivered to the load

```

1 clear all;
2 clc;
3 V_s=230;
4 V_01=4*V_s/(%pi*sqrt(2));
5 R=1;
6 X_L=6;
7 X_c=7;
8 I_01=V_01/sqrt(R^2+(X_L-X_c)^2);
9 P=I_01^2*R;    printf("power delivered to the source
      =%0.3f kW" ,P/1000);
10 I_s=sqrt(2)*I_01*(2*cosd(45))/%pi;
11 P_s=V_s*I_s;    printf("\npower from the source=%0.3f
      kW" ,P_s/1000);

```

Scilab code Exa 8.6 to find rms value of thyristor and diode currents

```

1 clear all;
2 clc;
3 disp("when load R=2 ohm");
4 V_01=230;
5 R=2;

```

```

6 I_01=V_01/R;
7 I_m=I_01*sqrt(2);
8 I_T1=I_m/2;    printf("rms value of thyristor
    current=%0.2f A",I_T1);
9 I_D1=0;    printf("\nrms value of diode current=%0.0f
    A",I_D1);
10
11 disp("when load R=2ohm, X_L=8ohm and X_C=6ohm");
12 X_L=8;
13 X_C=6;
14 I_01=V_01/sqrt(R^2+(X_L-X_C)^2);
15 phi1=atand((X_L-X_C)/R);
16 I_T1=I_T1*sqrt(2)*.47675;    printf("rms value of
    thyristor current=%0.3f A",I_T1);
17 I_D1=.1507025*I_m/sqrt(2);    printf("\nrms value of
    diode current=%0.3f A",I_D1);

```

Scilab code Exa 8.7 to find 1 rms value of fundamental load current 2 power absorbed by the load and fundamental power 3 rms and peak currents of each thyristor 4 conduction of thyristors and diodes

```

1 clear all;
2 clc;
3 //v_o=4*V_s/%pi*(sind(wt)+sind(3*wt)/3+sind(5*wt)/5)
4 V_s=230;
5 R=4;
6 f=50;
7 w=2*%pi*f;
8 L=0.035;
9 C=155*10^-6;
10 X_L=w*L;
11 X_C=1/(w*C);
12 Z1=sqrt(R^2+(X_L-X_C)^2);
13 phi1=-atand((X_L-X_C)/R);
14 Z3=sqrt(R^2+(X_L*3-X_C/3)^2);

```

```

15 phi3=atand((X_L*3-X_C/3)/R);
16 Z5=sqrt(R^2+(X_L*5-X_C/5)^2);
17 phi5=atand((X_L*5-X_C/5)/R);
18
19 I_m1=4*V_s/(Z1*%pi);
20 I_01=I_m1/sqrt(2);    printf("rms value of
    fundamental load current=%0.2 f A",I_01);
21 I_m3=4*V_s/(3*Z3*%pi);
22 I_m5=4*V_s/(5*Z5*%pi);
23 I_m=sqrt(I_m1^2+I_m3^2+I_m5^2);
24 I_0=I_m/sqrt(2);
25 P_0=(I_0)^2*R;    printf("\nload power=%0.1 f W",P_0);
26 P_01=(I_01)^2*R;    printf("\nfundamental load power
    =%0.1 f W",P_01);
27 printf("\nrms value of thyristor current=%0.3 f A",I_m
    /2);
28
29 t1=(180-phi1)*%pi/(180*w);    printf("\nconduction
    time for thyristor=%0.3 f ms",t1*1000);
30 t1=(phi1)*%pi/(180*w);    printf("\nconduction time
    for diodes=%0.3 f ms",t1*1000);

```

Scilab code Exa 8.8 to determine 1 fundamental rms output voltage 2 total output power and fundamenetal freq power 3 avg and peak currents of each thyristor 4 input pf 5 distortion factor and THD and 6 harmonic factor foe lowest hamonic

```

1 clear all;
2 clc;
3 V_s=230;
4 V_01=2*V_s/(sqrt(2)*%pi);    printf("fundamental rms
    o/p voltage=%0.3 f V",V_01);
5 R=10;
6 I_01=V_01/R;
7 P=I_01^2*R;    printf("\nfundamental power to load=%0

```

```

    .1 f W",P);
8 V_or=sqrt((V_s/2)^2);
9 P=V_or^2/R;    printf("\ntotal o/p power to load=%0.1
    f W",P);
10
11 I_TP=V_s/(2*R);
12 printf("\navg SCR current=%0.2 f A",I_TP*180/360);
13
14 I_or=I_TP;
15 pf=I_01^2*R/(V_or*I_or);    printf("\ni/p pf=%0.3 f",
    pf);
16
17 DF=V_01/V_or;    printf("\ndistortion factor=%0.1 f",
    DF);
18
19 V_oh=sqrt(V_or^2-V_01^2);
20 THD=V_oh/V_01;    printf("\nTHD=%0.3 f",THD);
21
22 V_03=V_01/3;
23 HF=V_03/V_01;    printf("\nharmonic factor=%0.4 f",HF)
    ;

```

Scilab code Exa 8.9 to calculate 1 rms value of output voltage and fundamental component of output voltage 2 output power 3 fundamental freq output power 4 avg and peak currents of each transistor 5 peak reverse blocking voltage 6 harmonic factor for third harmonic 7 THD

```

1 clear all;
2 clc;
3 V_s=60;
4 R=3;
5 V_or=sqrt(V_s^2*%pi/%pi);    printf("rms value of o/
    p voltage=%0.0 f V",V_or);
6 V_01=4*V_s/(sqrt(2)*%pi);    printf("\nfundamental
    component of rms voltage=%0.2 f V",V_01);

```

```

7 P_o=V_or^2/R;    printf("\no/p power=%.0 f W",P_o);
8 P_01=V_01^2/R;    printf("\nfundamental freq o/p
    power=%.2 f W",P_01);
9
10 I_s=V_s/R;    printf("\npeak current=%.0 f A",I_s);
11 I_avg=I_s*%pi/(2*%pi);    printf("\navg current of
    each transistor=%.0 f A",I_avg);
12
13 printf("\npeak reverse blocking voltage=%.0 f V",V_s)
    ;
14
15 V_03=V_01/3;
16 HF=V_03/V_01;    printf("\nharmonic factor=%.4 f",HF)
    ;
17
18 V_oh=sqrt(V_or^2-V_01^2);
19 THD=V_oh/V_01;    printf("\nTHD=%.4 f",THD);

```

Scilab code Exa 8.10 1 to calculate THD of output voltage and its distortion factor
2 to calculate THD of output current and its distortion factor
3 load power and avg dc source current
4 conduction time of each transistor and diode
5 peak and rms current of each transistor

```

1 clear all;
2 clc;
3 V_s=220;
4 R=6;
5 f=50;
6 w=2*%pi*f;
7 L=0.03;
8 C=180*10^-6;
9 X_L=w*L;
10 X_C=1/(w*C);
11
12 V_or=sqrt(V_s^2*%pi/%pi);

```

```

13 V_01=4*V_s/(sqrt(2)*%pi);
14 V_oh=sqrt(V_or^2-V_01^2);
15 THD=V_oh/V_01;    printf("THD of voltage=%0.4 f",THD);
16 DF=V_01/V_or;    printf("\nDF=%0.1 f",DF);
17
18 Z1=sqrt(R^2+(X_L-X_C)^2);
19 phi1=-atand((X_L-X_C)/R);
20 Z3=sqrt(R^2+(X_L*3-X_C/3)^2);
21 phi3=atand((X_L*3-X_C/3)/R);
22 Z5=sqrt(R^2+(X_L*5-X_C/5)^2);
23 phi5=atand((X_L*5-X_C/5)/R);
24 Z7=sqrt(R^2+(X_L*7-X_C/7)^2);
25 phi7=atand((X_L*7-X_C/7)/R);
26
27 I_01=19.403;
28 I_m1=4*V_s/(Z1*%pi);
29 I_m3=4*V_s/(3*Z3*%pi);
30 I_m5=4*V_s/(5*Z5*%pi);
31 I_m7=4*V_s/(7*Z7*%pi);
32 I_m=sqrt(I_m1^2+I_m3^2+I_m5^2+I_m7^2);
33 I_or=I_m/sqrt(2);
34 I_oh=sqrt((I_m^2-I_m1^2)/2);
35 THD=I_oh/I_01;    printf("\nTHD of current=%0.4 f",THD
    );
36 DF=I_01/I_or;    printf("\nDF=%0.3 f",DF);
37
38 P_o=I_or^2*R;    printf("\nload power=%0.1 f W",P_o);
39 I_avg=P_o/V_s;    printf("\navg value of load
    current=%0.2 f A",I_avg);
40
41 t1=(180-phi1)*%pi/(180*w);    printf("\nconduction
    time for thyristor=%0.0 f ms",t1*1000);
42 t1=1/(2*f)-t1;    printf("\nconduction time for
    diodes=%0.0 f ms",t1*1000);
43
44 I_p=I_m1;    printf("\npeak transistor current=%0.2 f
    A",I_p);
45 I_t1=.46135*I_p;    printf("\nrms transistor current

```



```
=%.2 f A" ,I_t1);
```

Scilab code Exa 8.11 to determine 1 rms value of load current 2 rms value of thyristor current 3 load power

```
1 clear all;
2 clc;
3 V_s=450;
4 R=10;
5 disp(" for 180deg mode");
6 I_or=sqrt((V_s/(3*R))^2*2/3+(2*V_s/(3*R))^2*1/3);
   printf("rms value of load current=%.3 f A" ,I_or
   );
7 I_T1=sqrt((1/(2*pi))*((V_s/(3*R))^2*2*pi/3+(2*V_s
   /(3*R))^2*pi/3)); printf("\nrms value of load
   current=%.0 f A" ,I_T1);
8 P=3*I_or^2*R; printf("\npower delivered to load=%
   .1 f kW" ,P/1000);
9
10 disp(" for 120deg mode");
11 I_or=sqrt((1/(pi))*((V_s/(2*R))^2*2*pi/3));
   printf("rms value of load current=%.3 f A" ,I_or);
12 I_T1=sqrt((1/(2*pi))*((V_s/(2*R))^2*2*pi/3));
   printf("\nrms value of load current=%.2 f A" ,I_T1)
   ;
13 P=3*I_or^2*R; printf("\npower delivered to load=%
   .3 f kW" ,P/1000);
```

Scilab code Exa 8.12 to determine power delivered to load for 1 square wave output 2 quasi square wave ooutput 3 symmitrically placed pulses

```
1 clear all;
2 clc;
```

```

3 V_s=230;
4 R=10;
5 f=50;
6 w=2*pi*f;
7 L=0.03;
8 X_L=w*L;
9
10 V_or=sqrt(V_s^2*pi/pi);
11 V_01=4*V_s/(sqrt(2)*pi);
12
13 Z1=sqrt(R^2+(X_L)^2);
14 phi1=-atand((X_L)/R);
15 Z3=sqrt(R^2+(X_L*3)^2);
16 phi3=atand((X_L*3)/R);
17 Z5=sqrt(R^2+(X_L*5)^2);
18 phi5=atand((X_L*5)/R);
19 Z7=sqrt(R^2+(X_L*7)^2);
20 phi7=atand((X_L*7)/R);
21
22 disp("using square wave o/p");
23 I_m1=4*V_s/(sqrt(2)*Z1*pi);
24 I_m3=4*V_s/(sqrt(2)*3*Z3*pi);
25 I_m5=4*V_s/(sqrt(2)*5*Z5*pi);
26 I_m7=4*V_s/(sqrt(2)*7*Z7*pi);
27 I_m=sqrt(I_m1^2+I_m3^2+I_m5^2+I_m7^2);
28 P=I_m^2*R;    printf("power delivered=%0.2f W",P);
29
30 disp("using quasi-square wave o/p");
31 I_01=I_m1*sind(45);
32 I_03=I_m3*sind(3*45);
33 I_05=I_m5*sind(5*45);
34 I_07=I_m7*sind(7*45);
35 I_0=(I_01^2+I_03^2+I_05^2+I_07^2);
36 P=I_0*R;    printf("power delivered=%0.2f W",P);
37
38 disp("using two symmetrical spaced pulses");
39 g=(180-90)/3+45/2;
40 I_01=2*I_m1*sind(g)*sind(45/2);

```

```

41 I_03=2*I_m3*sind(g*3)*sind(3*45/2);
42 I_05=2*I_m5*sind(g*5)*sind(5*45/2);
43 I_07=2*I_m7*sind(g*7)*sind(7*45/2);
44 I_0=(I_01^2+I_03^2+I_05^2+I_07^2);
45 P=I_0*R;    printf("power delivered=%0.2 f W",P);

```

Scilab code Exa 8.14 to determine the value of source inductance and also find value of commutating capacitors

```

1 clear all;
2 clc;
3 f=50;
4 T=1/f;
5 I=.5;
6 di=I/T;    //di=di/dt
7 V_s=220;
8 L=V_s/di;    printf("source inductance=%0.1 f H",L);
9
10 t=20*10^-6;
11 fos=2;    //factor of safety
12 t_c=t*fos;
13 R=10;
14 C=t_c/(R*log(2));    printf("\ncommutating capacitor
    =%0.2 f uF",C*10^6);

```

Scilab code Exa 8.15 to check whether the ckt will commute by itself or not and the voltage across the capacitor and inductor at the commutation

```

1 clear all;
2 clc;
3 R=10;
4 L=.01;
5 C=10*10^-6;

```

```

6  if(R^2<4*L/C)
7      disp("ckt will commute on its own");
8  else
9      disp("ckt will not commute on its own");
10 end
11
12 xie=R/(2*L);
13 w_o=1/sqrt(L*C);
14 w_r=sqrt(w_o^2-xie^2);
15 phi=atand(xie/w_r);
16 t=%pi/w_r;
17 V_s=1;
18 v_L=V_s*(w_o/w_r)*exp(-xie*t)*cosd(180+phi);
    printf("voltage across inductor(*V_s)=%.5 f V",v_L
    );
19 v_c=V_s*(1-(w_o/w_r)*exp(-xie*t)*cosd(180-phi));
    printf("\nvoltage across capacitor(*V_s)=%.5 f V",
    v_c);
20 di=V_s/L;    printf("\ndi/dt*V_s (for t=0)=%.0 f A/s"
    ,di);

```

Scilab code Exa 8.16 to calculate output freq

```

1  clear all;
2  clc;
3  L=.006;
4  C=1.2*10^-6;
5  R=100;
6  T=%pi/sqrt(1/(L*C)-(R/(2*L))^2);
7  T_off=.2*10^-3;
8  f=1/(2*(T+T_off));    printf("o/p freq=%.2 f Hz",f);
9
10 disp(" for R=40ohm");
11 R=40;
12 T=%pi/sqrt(1/(L*C)-(R/(2*L))^2);

```

```

13 T_off=.2*10^-3;
14 f=1/(2*(T+T_off));    printf("upper limit o/p freq=%
    .1 f Hz",f);
15
16 disp("for R=140ohm");
17 R=140;
18 T=%pi/sqrt(1/(L*C)-(R/(2*L))^2);
19 T_off=.2*10^-3;
20 f=1/(2*(T+T_off));    printf("lower limit o/p freq=%
    .1 f Hz",f);

```

Scilab code Exa 8.17 To determine 1 ckt turn off time 2 max possible operating freq

```

1 clear all;
2 clc;
3 f=5000;
4 w=2*%pi*f;
5 R=3;
6 L=60*10^-6;
7 xie=R/(2*L);
8 C=7.5*10^-6;
9 w_o=1/sqrt(L*C);
10 w_r=sqrt(w_o^2-xie^2);
11 t_c=%pi*(1/w-1/w_r);    printf("ckt turn off time=%
    .2 f us",t_c*10^6);
12
13 fos=1.5;
14 t_q=10*10^-6;
15 f_max=1/(2*%pi*(t_q*fos/%pi+1/w_r));    printf("\
    nmax possible operating freq=%0.1 f Hz",f_max);
16 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 8.18 to calculate power delivered to the load and avg and rms values of thyristor current

```
1 clear all;
2 clc;
3 f=5000;
4 w=2*%pi*f;
5 R=3;
6 L=60*10^-6;
7 xie=R/(2*L);
8 C=7.5*10^-6;
9 w_o=1/sqrt(L*C);
10 w_r=sqrt(w_o^2-xie^2);
11 t1=%pi/(2*w_r);
12 V_s=220;
13 V_co=80;
14 I_omx=(V_s+V_co)*exp(-xie*t1)/(w_r*L);
15 I_rms=I_omx/sqrt(2);
16 P=I_rms^2*R;    printf("load power=%0.2 f W",P);
17 printf("\nrms value of thyristor current=%0.3 f A",
    I_omx/2);
18 I_SA=P/V_s;
19 printf("\navg thyristor current=%0.3 f A",I_SA/2);
20 //error in the book. wrong values are placed in the
    Lomx formulae. so all answer varies
```

Scilab code Exa 8.19 to find the value of capacitor C for given conditions

```
1 clear all;
2 clc;
3 t=20;
4 fos=2;    //factor of safety
```

```

5 t_c=t*fos;
6 n=1/3;
7 R=20;
8 C=n^2*t_c/(4*R*log(2));    printf("value of
    capacitor=%0.5 f uF",C);
9 //printing mistake in the answer in book.

```

Scilab code Exa 8.20 to calculate 1 rms values of phase and line voltages
2 rms value of fundamental component of phase and line voltages 3 THD for
voltages 4 load power and avg source current 5 avg value of thyristor current

```

1 clear all;
2 clc;
3 V_s=220;
4 V_p=sqrt(2)*V_s/3;    printf("rms value of phasor
    voltages=%0.2 f V",V_p);
5 V_L=sqrt(3)*V_p;    printf("\nrms value of line
    voltages=%0.2 f V",V_L);
6
7 V_p1=sqrt(2)*V_s/%pi;    printf("\nfundamental
    component of phase voltage=%0.3 f V",V_p1);
8 V_L1=sqrt(3)*V_p1;    printf("\nfundamental
    component of line voltages=%0.3 f V",V_L1);
9
10 V_oh=sqrt(V_L^2-V_L1^2);
11 THD=V_oh/V_L1;    printf("\nTHD=%0.7 f",THD);
12
13 V_a1=2*V_s/%pi;
14 V_a5=2*V_s/(5*%pi);
15 V_a7=2*V_s/(7*%pi);
16 V_a11=2*V_s/(11*%pi);
17 R=4;
18 L=0.02;
19 f=50;
20 w=2*%pi*f;

```

```

21 Z1=sqrt(R^2+(w*L)^2);
22 Z5=sqrt(R^2+(5*w*L)^2);
23 Z7=sqrt(R^2+(7*w*L)^2);
24 Z11=sqrt(R^2+(11*w*L)^2);
25 I_a1=V_a1/Z1;
26 I_a5=V_a5/Z5;
27 I_a7=V_a7/Z7;
28 I_a11=V_a11/Z11;
29 I_or=sqrt((I_a1^2+I_a5^2+I_a7^2+I_a11^2)/2);
30 P=3*I_or^2*R;    printf("\nload power=%.1f W",P);
31 I_s=P/V_s;    printf("\navg value of source current=
%.3f A",I_s);
32 I_TA=I_s/3;    printf("\navg value of thyristor
current=%.3f A",I_TA);

```

Chapter 9

AC Voltage Controllers

Scilab code Exa 9.1 to determine 1 rms value of output voltage 2 power delivered to the load and input pf 3 avg input current

```
1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 a=45;
6 V_or=(V_m/2)*sqrt(1/%pi*((2*%pi-a*%pi/180)+sind(2*a)
    /2)); printf("rms value of o/p voltage=%0.3f V"
    ,V_or);
7 R=20;
8 I_or=V_or/R;
9 P_o=I_or^2*R; printf("\nload power=%0.1f W",P_o);
10 I_s=I_or;
11 VA=V_s*I_s;
12 pf=P_o/VA; printf("\ni/p pf=%0.4f",pf);
13
14 V_o=sqrt(2)*V_s/(2*%pi)*(cosd(a)-1);
15 I_ON=V_o/R; printf("\navg i/p current=%0.4f A",
    I_ON);
```

Scilab code Exa 9.2 to calculate 1 rms value of output voltage 2 load power and input pf 3 avg and rms currents of thyristors

```

1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 a=45;
6 V_or=(V_s)*sqrt(1/%pi*((%pi-a*%pi/180)+sind(2*a)/2))
    ; printf("rms value of o/p voltage=%0.3 f V",
    V_or);
7 R=20;
8 I_or=V_or/R;
9 P_o=I_or^2*R; printf("\nload power=%0.2 f W",P_o);
10 I_s=I_or;
11 VA=V_s*I_s;
12 pf=P_o/VA; printf("\ni/p pf=%0.4 f",pf);
13
14 I_TA=sqrt(2)*V_s/(2*%pi*R)*(cosd(a)+1); printf("\
    navg thyristor current=%0.3 f A",I_TA);
15 I_Tr=sqrt(2)*V_s/(2*R)*sqrt(1/%pi*((%pi-a*%pi/180)+
    sind(2*a)/2)); printf("\nrms value of
    thyristor current=%0.4 f A",I_Tr);

```

Scilab code Exa 9.3 to determine 1 rms output voltage 2 input pf 3 avg and rms thyristor currents

```

1 clear all;
2 clc;
3 V_s=230;
4 n=6; //on cycles
5 m=4; //off cycles

```

```

6 k=n/(n+m);
7 V_or=V_s*sqrt(k);    printf("rms value of o/ voltage
    =%.3f V",V_or);
8 pf=sqrt(k);    printf("\ni/p pf=%.4f",pf);
9 R=15;
10 I_m=V_s*sqrt(2)/R;
11 I_TA=k*I_m/%pi;    printf("\navg thyristor current=%
    .4f A",I_TA);
12 I_TR=I_m*sqrt(k)/2;    printf("\nrms value of
    thyristor current=%.3f A",I_TR);

```

Scilab code Exa 9.4 to calculate 1 max values of avg and rms thyristor currents 2 min ckt turn off time

```

1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 R=3;
6 I_TAM=2*V_m/(2*%pi*R);    printf("max value of avg
    thyristor current=%.3f A",I_TAM);
7 I_TRM=V_m/(2*R);    printf("\nmax value of avg
    thyristor current=%.3f A",I_TRM);
8
9 f=50;
10 w=2*%pi*f;
11 t_c=%pi/w;    printf("\nckt turn off time=%.0f ms",
    t_c*1000);

```

Scilab code Exa 9.5 to calculate 1 control range of firing angle 2 max value of rms load current 3 max power and pf 4 max value of avg and rms thyristor currents 5 max possible value of current rating

```

1 clear all;
2 clc;
3 R=3;
4 X_L=4;
5 phi=atand(X_L/R);    printf("min firing angle=%0.2 f
    deg",phi);
6 printf("\nmax firing angle=%0.0 f deg",180);
7 V_s=230;
8 Z=sqrt(R^2+X_L^2);
9 I_or=V_s/Z;    printf("\nmax value of rms load
    current=%0.0 f A",I_or);
10 P=I_or^2*R;    printf("\nmax power=%0.0 f W",P);
11 I_s=I_or;
12 pf=P/(V_s*I_s);    printf("\ni/p pf=%0.1 f",pf);
13
14 I_TAM=sqrt(2)*V_s/(%pi*Z);    printf("\nmax value of
    avg thyristor current=%0.3 f A",I_TAM);
15 I_Tm=sqrt(2)*V_s/(2*Z);    printf("\nmax value of
    rms thyristor current=%0.3 f A",I_Tm);
16
17 f=50;
18 w=2*%pi*f;
19 di=sqrt(2)*V_s*w/Z;    printf("\ndi/dt=%0.0 f A/s",di)
    ;

```

Scilab code Exa 9.6 to calculate extinction angle and rms value of output voltage

```

1 clc
2 clear all
3 V=230;
4 R=3; //ohm
5 X_L=5; //ohm
6 a=120; //firing angle delay
7 phi=atand(X_L/R);

```

```

8 b=0;
9 i=1;
10 while i>0;
11     LHS=sind(b-a);
12     RHS=sind(a-phi)*exp(-(R/X_L)*(b-a)*%pi/180);
13     if abs(LHS-RHS)<=.01;
14         B=b;
15         i=2;
16         break;
17     end
18     b=b+.1
19 end
20 printf("extinction angle=%0.1f deg",B);//answer in
    the book is wrong as formulae for RHS is wrongly
    employed
21 V_or=sqrt(2)*V*sqrt((1/(2*%pi))*((B-a)*%pi/180+(sind
    (2*a)-sind(2*B))/2));
22 printf("\nrms value of output voltage=%0.2f V",V_or);
    //answer do not match due to wrong B in book

```

Scilab code Exa 9.8 to calculate 1 rms value of output voltage 2 rms value of current for upper thyristors 3 rms value of current for lower thyristors 4 transformer VA rating 5 input pf

```

1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 a=60;
6 R=20;
7 V_or=sqrt((V_m^2/(2*%pi))*(a*%pi/180-sind(2*a)/2)
    +(2*V_m^2/(%pi))*(%pi-a*%pi/180+sind(2*a)/2));
    printf("rms value of o/p voltage=%0.2f V",V_or)
    ;
8 I_T1r=(V_m/R)*sqrt(1/%pi*((%pi-a*%pi/180)+sind(2*a)

```

```

/2));    printf("\nrms value of current for upper
thyristors=%0.3f A",I_T1r);
9 I_T3r=(V_m/(2*R))*sqrt(1/%pi*((a*%pi/180)-sind(2*a)
/2));    printf("\nrms value of current for lower
thyristors=%0.3f A",I_T3r);
10 I1=sqrt(2)*I_T1r;
11 I3=sqrt((sqrt(2)*I_T1r)^2+(sqrt(2)*I_T3r)^2);
12 r=V_s*(I1+I3);    printf("\nt/f VA rating=%0.2f VA",r
);
13 P_o=V_or^2/R;
14 pf=P_o/r;    printf("\ni/p pf=%0.4f",pf);

```

Chapter 10

Cycloconverters

Scilab code Exa 10.2 to calculate 1 rms value of output voltage 2 rms current of each converter 3 rms current of each thyristor and 4 input pf

```
1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 R=10;
6 a=30;
7 V_or=(V_m/sqrt(2))*sqrt((1/%pi)*(%pi-a*%pi/180+sin(
    (2*a)/2));
8 I_or=V_or/R;    printf("rms value of o/p current=%0.2
    f A",I_or);
9 printf("\nrms value of o/p current for each
    convertor=%0.2f A",I_or/sqrt(2));
10 printf("\nrms value of o/p current for each
    thyristor=%0.3f A",I_or/2);
11 I_s=I_or;
12 pf=(I_or^2*R)/(V_s*I_s);    printf("\ni/p pf=%0.4f",
    pf);
```

Scilab code Exa 10.4 to compute 1 value of fundamental rms output voltage 2 rms output current 3 output power

```

1 clear all;
2 clc;
3 V_s=400;
4 V_ph=V_s/2;
5 a=160;
6 r=cosd(180-a);
7 m=3;
8 V_or=r*(V_ph*(m/%pi)*sin(%pi/m));    printf("rms o/p
      voltage=%0.3 f V",V_or);
9 R=2;
10 X_L=1.5;
11 th=atand(X_L/R);
12 Z=sqrt(R^2+X_L^2);
13 I_or=V_or/Z;    printf("\nrms o/p current=%0.2 f A",
      I_or);
14 printf("\nphase angle of o/p current=%0.2 f deg",-th)
15 P=I_or^2*R;    printf("\no/p power=%0.2 f W",P);

```

Scilab code Exa 10.5 to compute 1 value of fundamental rms output voltage 2 rms output current 3 output power Here 6 pulse bridge convertor is employed

```

1 clear all;
2 clc;
3 V_s=400;
4 V_ph=V_s/2;
5 V_l=V_ph*sqrt(3);
6 a=160;
7 r=cosd(180-a);
8 m=6;
9 V_or=r*(V_l*(m/%pi)*sin(%pi/m));    printf("rms o/p
      voltage=%0.2 f V",V_or);

```



```

10 R=2;
11 X_L=1.5;
12 th=atand(X_L/R);
13 Z=sqrt(R^2+X_L^2);
14 I_or=V_or/Z;    printf("\nrms o/p current=%.2 f A",
    I_or);
15 printf("\nphase angle of o/p current=%.2 f deg",-th)
16 P=I_or^2*R;    printf("\no/p power=%.2 f W",P);

```

Scilab code Exa 10.7 to calculata rms value of load voltage for various firing angle delays

```

1 clear all;
2 clc;
3 V_l=400;
4 V_ml=sqrt(2)*V_l;
5 m=6;
6 f=50;
7 w=2*pi*f;
8 L=.0012;
9 I=40;
10 disp("for firing angle=0deg");
11 a=0;
12 V_or=(V_ml*(m/pi)*sin(pi/m))*cosd(a);
13 V_omx=V_or-3*w*L*I/pi;
14 V_rms=V_omx/sqrt(2);    printf("rms value of load
    voltage=%.2 f V",V_rms);
15
16 disp("for firing angle=30deg");
17 a=30;
18 V_or=(V_ml*(m/pi)*sin(pi/m))*cosd(a);
19 V_omx=V_or-3*w*L*I/pi;
20 V_rms=V_omx/sqrt(2);    printf("rms value of load
    voltage=%.2 f V",V_rms);

```

Chapter 11

Some Applications

Scilab code Exa 11.1 to calculate 1 firing angle of the rectifier 2 output voltage 3 dc link voltage

```
1 clear all;
2 clc;
3 V_s=11000;
4 V_m1=sqrt(2)*V_s;
5 f=50;
6 w=2*pi*f;
7 I_d=300;
8 R_d=1;
9 g=20; //g=gamma
10 a=acosd(cosd(g)+pi/(3*V_m1)*I_d*R_d); printf("
    firing angle=%.3f deg",a);
11 L_s=.01;
12 V_d=(3/pi)*((V_m1*cosd(a))-w*L_s*I_d); printf("\
    nrectifier o/p voltage=%.1f V",V_d);
13 printf("\ndc link voltage=%.3f V",2*V_d/1000);
```

Scilab code Exa 11.2 To calculate rms current and peak reverse voltage ratings for each of thyristor valves

```

1 clear all;
2 clc;
3 V_d=(200+200)*10^3;
4 P=1000*10^6;
5 I_d=P/V_d;
6 //each thristor conducts for 120deg for a
  periodicity of 360deg
7 printf("rms current rating of thyristor=%0.2f A",I_d*
  sqrt(120/360));
8 a=0;
9 V_d=200*10^3;
10 V_m1=V_d*%pi/(3*cosd(a));
11 printf("\npeak reverse voltage across each thyristor
  =%0.2f kV",V_m1/2/1000);

```

Scilab code Exa 11.3 to determine voltage and current rating of 1 thyristor and 2 diodes of the bridge

```

1 clear all;
2 clc;
3 V_m=230;
4 V_s=230/sqrt(2);
5 pf=.8;
6 P=2000;
7 I_m=P/(V_s*pf);
8 I_Tr=I_m/sqrt(2);
9 I_TA=2*I_m/%pi;
10 fos=2; //factor of safety
11 printf("rms value of thyristor current=%0.2f A",fos*
  I_Tr);
12 printf("\navg value of thyristor current=%0.3f A",fos
  *I_TA);
13 PIV=V_m*sqrt(2);
14 printf("\nvoltage rating of thyristor=%0.2f V",PIV);
15

```

```

16 I_Tr=I_m/(2);
17 I_TA=I_m/%pi;
18 printf("\nrms value of diode current=%%.3f A",fos*
    I_Tr);
19 printf("\navg value of diode current=%%.3f A",fos*
    I_TA);
20 printf("\nvoltage rating of diode=%%.2f V",PIV);

```

Scilab code Exa 11.4 to find the value of parameters of R2 C and load resistance

```

1 clear all;
2 clc;
3 V=200;
4 I=10;
5 R_L=V/I;    printf("value of load resistance=%%.0f
    ohm",R_L);
6 I_h=.005;    //holding current
7 R2=V/I_h;    printf("\nvalue of R2=%%.0f kilo-ohm",R2
    /1000);
8 t_c=20*10^-6;
9 fos=2; //factor of safety
10 C=t_c*fos/(R_L*log(2));    printf("\nvalue of C=%%.3f
    uF",C*10^6);

```

Scilab code Exa 11.5 to calculate 1 depth of heat of penetration 2 heat generated per unit cylinder surface area and 3 heat generated per unit cylinder volume

```

1 clear all;
2 clc;
3 u_r=10;
4 f=10000; //Hz

```

```

5 p=4*10^-8; //ohm-m
6 dl=(1/(2*pi))*sqrt(p*10^7/(u_r*f));    printf("
    depth of heat of penetration=%0.5 f mm",dl*1000);
7 l=.12; //length of cylinder
8 t=20; //no of turns
9 I=100;
10 H=t*I/l;
11 P_s=2*pi*H^2*sqrt(u_r*f*p*10^-7);    printf("\nheat
    generated per unit cylinder surface area=%0.3 f W/
    m^2",P_s);
12 d=.02; //diameter
13 P_v=4*H^2*p/(d*dl);    printf("\nheat generated per
    unit cylinder volume=%0.0 f W/m^3",P_v);
14 //answer of P_v varies as given in book as value of
    d is not taken as in formulae.

```

Scilab code Exa 11.6 to calculate reqd capacitor size

```

1 clear all;
2 clc;
3 f=3000;
4 t_qmin=30*10^-6;
5 f_r=f/(1-2*t_qmin*f);
6 R=.06;
7 L=20*10^-6;
8 C=1/(L*((2*pi*f_r)^2+(R/(2*L))^2));    printf("
    required capacitor size=%0.4 f F",C*10^6);
9 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Chapter 12

Electric Drives

Scilab code Exa 12.1 To determine 1 firing angle delay of armature converter 2 rms value of thyristor and freewheeling diode currents 3 input pf of armature converter

```
1 clear all;
2 clc;
3 T_e=15; //Nm
4 K_m=.5; //V-s/rad
5 I_a=T_e/K_m;
6 n_m=1000;
7 w_m=2*pi*n_m/60;
8 E_a=K_m*w_m;
9 r_a=.7;
10 V_t=E_a+I_a*r_a;
11 V_s=230;
12 V_m=sqrt(2)*V_s;
13 a=acosd(2*pi*V_t/V_m-1); printf(" firing angle
    delay=%.3f deg",a);
14 I_Tr=I_a*sqrt((180-a)/360); printf("\nrms value
    of thyristor current=%.3f A",I_Tr);
15 I_fdr=I_a*sqrt((180+a)/360); printf("\nrms value
    of freewheeling diode current=%.3f A",I_fdr);
16 pf=V_t*I_a/(V_s*I_Tr); printf("\ninput power
```

```
factor=%0.4f", pf);
```

Scilab code Exa 12.2 calculate the avg armature current and the motor torque

```
1 clear all;
2 clc;
3 V_m=330;
4 E_a=80;
5 r_a=4;
6 a=30;
7 I_a=((V_m/%pi*(1+cosd(a)))-E_a)/r_a;    printf(" avg
      armature current=%0.3f A", I_a);
8 n_s=1400;
9 w_m=2*%pi*n_s/60;
10 K_m=E_a/w_m;
11 T_e=K_m*I_a;    printf("\nmotor torque=%0.3f Nm", T_e)
    ;
```

Scilab code Exa 12.3 to determine 1 motor current 2 motor torque 3 input pf

```
1 clear all;
2 clc;
3 V_s=250;
4 V_m=sqrt(2)*V_s;
5 a=30;
6 k=0.03; //Nm/A^2
7 n_m=1000;
8 w_m=2*%pi*n_m/60;
9 r=.2; // r_a+r_s
10 V_t=V_m/%pi*(1+cosd(a));
```

```

11 I_a=V_t/(k*w_m+r);    printf("motor armature current
    =%.2f A",I_a);
12 T_e=k*I_a^2;    printf("\nmotor torque=%.3f Nm",T_e)
    ;
13 I_sr=I_a*sqrt((180-a)/180);
14 pf=(V_t*I_a)/(V_s*I_sr);    printf("\ninput power
    factor=%.4f",pf);

```

Scilab code Exa 12.4 to determine 1 rated armature current 2 firing angle delay of armature converter 3 speed regulation at full load 4 input pf of armature converter and drive

```

1 clear all;
2 clc;
3 V_s=400;
4 V_m=sqrt(2)*V_s;
5 V_f=2*V_m/pi;
6 r_f=200;
7 I_f=V_f/r_f;
8 T_e=85;
9 K_a=.8;
10 I_a=T_e/(I_f*K_a);    printf("rated armature current
    =%.2f A",I_a);
11 n_m=1200;
12 w_m=2*pi*n_m/60;
13 r_a=.2;
14 V_t=K_a*I_f*w_m+I_a*r_a;
15 a=acosd(V_t*pi/(2*V_m));    printf("\nfiring angle
    delay=%.2f deg",a);
16 E_a=V_t;
17 w_mo=E_a/(K_a*I_f);
18 N=60*w_mo/(2*pi);
19 reg=((N-n_m)/n_m)*100;    printf("\nspeed regulation
    at full load=%.2f",reg);
20 I_ar=I_a;

```



```

21 pf=(V_t*I_a)/(V_s*I_ar);    printf("\ninput power
    factor of armature convertor=%0.4f",pf);
22 I_fr=I_f;
23 I_sr=sqrt(I_fr^2+I_ar^2);
24 VA=I_sr*V_s;
25 P=V_t*I_a+V_f*I_f;
26 printf("\ninput power factor of drive=%0.4f",P/VA);
27 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 12.5 to calculate 1 delay angle of field converter 2 delay angle of armature converter 3 power fed back to the supply

```

1 clear all;
2 clc;
3 V_s=400;
4 V_m=sqrt(2)*V_s;
5 V_f=2*V_m/%pi;
6 a1=acosd(-V_f*%pi/(2*V_m));    printf("delay angle
    of field converter=%0.0f deg",a1);
7 r_f=200;
8 I_f=V_f/r_f;
9 T_e=85;
10 K_a=.8;
11 I_a=T_e/(I_f*K_a);
12 n_m=1200;
13 w_m=2*%pi*n_m/60;
14 r_a=.1;
15 I_a=50;
16 V_t=-K_a*I_f*w_m+I_a*r_a;
17 a=acosd(V_t*%pi/(2*V_m));    printf("\nfiring angle
    delay of armature converter=%0.3f deg",a);
18 printf("\npower fed back to ac supply=%0.0f W",-V_t*
    I_a);

```

Scilab code Exa 12.6 To compute 1 motor speed 2 torque developed

```
1 clear all;
2 clc;
3 V_t=220;
4 n_m=1500;
5 w_m=2*pi*n_m/60;
6 I_a=10;
7 r_a=1;
8 K_m=(V_t-I_a*r_a)/(w_m);
9 T=5;
10 I_a=T/K_m;
11 V_s=230;
12 V_m=sqrt(2)*V_s;
13 a=30;
14 V_t=2*V_m*cosd(a)/pi;
15 w_m=(V_t-I_a*r_a)/K_m;
16 N=w_m*60/(2*pi);    printf("motor speed=%.2 f rpm",N
    );
17 a=45;
18 n_m=1000;
19 w_m=2*pi*n_m/60;
20 V_t=2*V_m*cosd(a)/pi;
21 I_a=(V_t-K_m*w_m)/r_a;
22 T_e=K_m*I_a;    printf("\ntorque developed=%.3 f Nm",
    T_e);
23 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.
```

Scilab code Exa 12.7 to compute firing angle and motor speed

```

1 clear all;
2 clc;
3 V_t=220;
4 n_m=1000;
5 w_m=2*%pi*n_m/60;
6 I_a=60;
7 r_a=.1;
8 K_m=(V_t-I_a*r_a)/(w_m);
9 V_s=230;
10 V_m=sqrt(2)*V_s;
11 disp(" for 600rpm speed");
12 n_m=600;
13 w_m=2*%pi*n_m/60;
14 a=acosd((K_m*w_m+I_a*r_a)*%pi/(2*V_m)); printf("
    firing angle=%0.3f deg",a);
15
16 disp(" for -500rpm speed");
17 n_m=-500;
18 w_m=2*%pi*n_m/60;
19 a=acosd((K_m*w_m+I_a*r_a)*%pi/(2*V_m)); printf("
    firing angle=%0.3f deg",a);
20
21 I_a=I_a/2;
22 a=150;
23 V_t=2*V_m*cosd(a)/%pi;
24 w_m=(V_t-I_a*r_a)/K_m;
25 N=w_m*60/(2*%pi); printf("\nmotor speed=%0.3f rpm"
    ,N);

```

Scilab code Exa 12.8 to calculate the speed of the motor

```

1 clear all;
2 clc;
3 K_m=1.5;
4 T_e=50;

```

```

5 I_a=T_e/K_m;
6 r_a=0.9;
7 a=45;
8 V_s=415;
9 V_m1=sqrt(2)*V_s;
10 w_m=((3*V_m1*(1+cosd(a))/(2*pi))-I_a*r_a)/K_m;
11 N=w_m*60/(2*pi);    printf("motor speed=%.2 f rpm",N
    );

```

Scilab code Exa 12.9 to compute rms value of source and thyristor currents avg value of thyristor currents and input supply pf

```

1 clear all;
2 clc;
3 V_t=600;
4 n_m=1500;
5 w_m=2*pi*n_m/60;
6 I_a=80;
7 r_a=1;
8 K_m=(V_t-I_a*r_a)/(w_m);
9 V_s=400;
10 V_m=sqrt(2)*V_s;
11 disp("for firing angle=45deg and speed=1200rpm");
12 a=45;
13 n_m=1200;
14 w_m=2*pi*n_m/60;
15 I_a=(3*V_m*(1+cosd(a))/(2*pi)-K_m*w_m)/r_a;
16 I_sr=I_a*sqrt(2/3);    printf("rms value of source
    current=%.3 f A",I_sr);
17 printf("\nrms value of thyristor current=%.3 f A",I_a
    *sqrt(1/3));
18 printf("\navg value of thyristor current=%.2 f A",I_a
    *(1/3));
19 pf=(3/(2*pi)*(1+cosd(a)));    printf("\ninput power
    factor=%.3 f",pf);

```

```

20
21 disp(" for firing angle=90deg and speed=700rpm");
22 a=90;
23 n_m=700;
24 w_m=2*%pi*n_m/60;
25 I_a=(3*V_m*(1+cosd(a))/(2*%pi)-K_m*w_m)/r_a;
26 I_sr=I_a*sqrt(90/180);    printf("rms value of
    source current=%0.3f A",I_sr);
27 printf("\nrms value of thyristor current=%0.3f A",I_a
    *sqrt(90/360));
28 printf("\navg value of thyristor current=%0.3f A",I_a
    *(1/3));
29 pf=(sqrt(6)/(2*%pi)*(1+cosd(a)))*sqrt(180/(180-a));
    printf("\ninput power factor=%0.4f",pf);

```

Scilab code Exa 12.10 To find 1 no load speed 2 firing angle and supply pf 3 speed regulation

```

1 clear all;
2 clc;
3 V_s=400;
4 V_m=sqrt(2)*V_s;
5 a=30;
6 V_t=3*V_m*cosd(a)/%pi;
7 I_a=21;
8 r_a=.1;
9 V_d=2;
10 K_m=1.6;
11 w_m=(V_t-I_a*r_a-V_d)/K_m;
12 N=w_m*60/(2*%pi);    printf("speed of motor=%0.1f rpm
    ",N);
13
14 N=2000;
15 w_m=2*%pi*N/60;
16 I_a=210;

```

```

17 V_t=K_m*w_m+I_a*r_a+V_d;
18 a=acosd(V_t*%pi/(3*V_m));      printf("\nfiring angle=
    %.2f deg",a);
19 I_sr=I_a*sqrt(2/3);
20 pf=V_t*I_a/(sqrt(3)*V_s*I_sr);  printf("\nsupply
    power factor=%.3f",pf);
21
22 I_a=21;
23 w_m=(V_t-I_a*r_a-V_d)/K_m;
24 n=w_m*60/(2*%pi);
25 reg=(n-N)/N*100;      printf("\nspeed regulation(
    percent)=%.2f",reg);

```

Scilab code Exa 12.11 to calculate 1 transformer phase turn ratio 2 firing angle delay

```

1 clear all;
2 clc;
3 V_t=230;
4 V_l=V_t*%pi/(3*sqrt(2));
5 V_ph=V_l/sqrt(3);
6 V_in=400;    //per phase voltage input
7 printf("transformer phase turns ratio=%.3f",V_in/
    V_ph);
8
9 N=1500;
10 I_a=20;
11 r_a=.6;
12 disp("for motor running at 1000rpm at rated torque")
    ;
13 E_a1=V_t-I_a*r_a;
14 n=1000;
15 E_a=E_a1/1500*1000;
16 V_t=E_a+I_a*r_a;
17 a=acosd(V_t*%pi/(3*sqrt(2)*V_l));      printf("firing

```

```

    angle delay=%0.2 f deg", a);
18
19 disp("for motor running at -900rpm at half of rated
    torque");
20 I_a=.5*I_a;
21 n=-900;
22 V_t=n*E_a1/N+I_a*r_a;
23 a=acosd(V_t*%pi/(3*sqrt(2)*V_l));    printf(" firing
    angle delay=%0.3 f deg", a);

```

Scilab code Exa 12.12 to calculate firing angle for different given conditions

```

1 clear all;
2 clc;
3 V_s=400;
4 V_ml=sqrt(2)*V_s;
5 V_f=3*V_ml/%pi;
6 R_f=300;
7 I_f=V_f/R_f;
8 T_e=60;
9 k=1.1;
10 I_a=T_e/(k*I_f);
11 N=1000;
12 w_m=2*%pi*N/60;
13 r_a=.3;
14 V_t=k*I_f*w_m+I_a*r_a;
15 a=acosd(V_t*%pi/(3*V_ml));    printf(" firing angle=%
    .3 f deg", a);
16
17 N=3000;
18 w_m=2*%pi*N/60;
19 a=0;
20 V_t=3*V_ml*cosd(a)/%pi;
21 I_f=(V_t-I_a*r_a)/(w_m*k);

```

```

22 V_f=I_f*R_f;
23 a=acosd(V_f*%pi/(3*V_m1));    printf("\nfiring angle
    =%.3f deg",a);

```

Scilab code Exa 12.13 to evaluate the time taken for the speed to reach 1000rpm

```

1 clear all;
2 clc;
3 //after calculating
4 //t=w_m/6000-%pi/360
5
6 N=1000;
7 w_m=2*%pi*N/60;
8 t=w_m/6000-%pi/360;    printf("time reqd=%.5f s",t);
9 //printing mistake in the answer in book

```

Scilab code Exa 12.14 to determine the 3rd and 5th harmonic components of line current

```

1 clear all;
2 clc;
3 I_a=1;//supposition
4 a=60;
5 I_s1=2*sqrt(2)/%pi*I_a*sind(a);
6 I_s3=2*sqrt(2)/(3*%pi)*I_a*sind(3*a);
7 I_s5=2*sqrt(2)/(5*%pi)*I_a*sind(5*a);
8 per3=I_s3/I_s1*100;    printf("percent of 3rd
    harmonic current in fundamental=%.0f",per3);
9 per5=I_s5/I_s1*100;    printf("\npercent of 5th
    harmonic current in fundamental=%.0f",per5);

```

Scilab code Exa 12.15 to calculate 1 rms and avg value of thyristor current 2 pf of ac source 3 motor speed

```

1 clear all;
2 clear all;
3 clc;
4 I_a=60;
5 I_TA=I_a/3;    printf("avg thyristor current=%0.0f A"
    ,I_TA);
6 I_Tr=I_a/sqrt(3);    printf("\nrms thyristor current
    =%0.3f A",I_Tr);
7
8 V_s=400;
9 V_m=sqrt(2)*V_s;
10 I_sr=I_a*sqrt(2/3);
11 a=150;
12 V_t=3*V_m*cosd(a)/%pi;
13 pf=V_t*I_a/(sqrt(3)*V_s*I_sr);    printf("\npower
    factor of ac source=%0.3f",pf);
14
15 r_a=0.5;
16 K_m=2.4;
17 w_m=(V_t-I_a*r_a)/K_m;
18 N=w_m*60/(2*%pi);    printf("\nspeed of motor=%0.2f
    rpm",N);
19 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 12.16 to determine 1 input power from source 2 input resistance of chopper drive 3 motor speed and 4 motor torque

```

1 clear all;
2 clc;
3 I_a=300;
4 V_s=600;
5 a=.6;
6 V_t=a*V_s;
7 P=V_t*I_a;    printf("input power from source=%0.0 f
    kW", P/1000);
8 R_eq=V_s/(a*I_a);    printf("\nequivalent input
    resistance=%0.3 f ohm", R_eq);
9 k=.004;
10 R=.04+.06;
11 w_m=(a*V_s-I_a*R)/(k*I_a);
12 N=w_m*60/(2*pi);    printf("\nmotor speed=%0.1 f rpm"
    ,N);
13 T_e=k*I_a^2;    printf("\nmotor torque=%0.0 f Nm", T_e)
    ;

```

Scilab code Exa 12.17 to calculate avg load current

```

1 clear all;
2 clc;
3 T_on=10;
4 T_off=15;
5 a=T_on/(T_on+T_off);
6 V_s=230;
7 V_t=a*V_s;
8 r_a=3;
9 K_m=.5;
10 N=1500;
11 w_m=2*pi*N/60;
12 I_a=(V_t-K_m*w_m)/r_a;    printf("motor load current
    =%0.3 f A", I_a);

```

Scilab code Exa 12.18 to determine 1 range of speed control 2 range of duty cycle

```
1 clear all;
2 clc;
3 w_m=0;    printf("lower limit of speed control=%0.0f
               rpm",w_m);
4 I_a=25;
5 r_a=.2;
6 V_s=220;
7 K_m=0.08;
8 a=(K_m*w_m+I_a*r_a)/V_s;    printf("\nlower limit of
               duty cycle=%0.3f",a);
9
10 a=1;    printf("\nupper limit of duty cycle=%0.0f",a)
           ;
11 w_m=(a*V_s-I_a*r_a)/K_m;    printf("\nupper limit of
               speed control=%0.1f rpm",w_m);
```

Scilab code Exa 12.19 To calculate the min and max values of armature current and armature current extrusion

```
1 clear all;
2 clc;
3 clear all
4 T_e=30;
5 K_m=1.5;
6 I_a=T_e/K_m;
7 N=1000;
8 w_m=2*pi*N/60;
9 E_a=K_m*w_m;
10 r_a=0;
```

```

11 V_t=E_a+I_a*r_a;
12 V_s=220;
13 a=V_t/V_s;
14 f=400;
15 T=1/f;
16 T_on=a*T;
17 T_off=T-T_on;
18 L=0.02;
19 di=(V_s-E_a)/L; //di=di_a/dt, during on period
20 dii=(-E_a)/L; //di=di_a/dt, during off period
21 //I_mx=I_mn+di*T_on;
22 //I_a=(I_mx+I_mn)/2;
23 //after solving
24 I_mx=22.808; printf("maximum armature current=%.3
    f A",I_mx);
25 I_mn=2*I_a-I_mx; printf("\nminimum armature
    current=%.3 f A",I_mn);
26 printf("\narmature current extrusion=%.3 f A",I_mx-
    I_mn);
27 t=poly(0,'t');
28 i_a=addf('I_mn',mulf('t','di'));
29 printf("\narmature current expression during turn-on
    ");
30 disp(eval(i_a));
31 i_a=addf('I_mx',mulf('t','dii'));
32 printf("\narmature current expression during turn-
    off");
33 disp(eval(i_a));

```

Scilab code Exa 12.21 to determine 1 power returned to the dc supply
2 equivalent load resistance of motor acting as generator 3 min and max
breaking speeds 4 speed during regenerative braking

```

1 clear all;
2 clc;

```

```

3 a=.6;
4 V_s=400;
5 V_t=(1-a)*V_s;
6 I_a=300;
7 P=V_t*I_a;      printf("power returned=%0.0 f kW",P
    /1000);
8
9 r_a=.2;
10 K_m=1.2;
11 R_eq=(1-a)*V_s/I_a+r_a;      printf("\nequivalent load
    resistance=%0.4 f ohm",R_eq);
12
13 w_mn=I_a*r_a/K_m;
14 N=w_mn*60/(2*%pi);      printf("\nmin braking speed=%0
    .2 f rpm",N);
15 w_mx=(V_s+I_a*r_a)/K_m;
16 N=w_mx*60/(2*%pi);      printf("\nmax braking speed=%0
    .1 f rpm",N);
17
18 w_m=(V_t+I_a*r_a)/K_m;
19 N=w_m*60/(2*%pi);      printf("\nmax braking speed=%0.1
    f rpm",N);

```

Scilab code Exa 12.22 to determine max current in terms of rated currents at given speeds

```

1 clear all;
2 clc;
3 N=1500;
4 disp("when speed=1455rpm");
5 n=1455;
6 s1=(N-n)/N;
7 r=sqrt(1/3)*(2/3)/(sqrt(s1)*(1-s1));      printf("
    I_2mx/I_2r=%0.3 f",r);
8

```

```

9  disp("when speed=1350rpm");
10 n=1350;
11 s1=(N-n)/N;
12 r=sqrt(1/3)*(2/3)/(sqrt(s1)*(1-s1));    printf("
      I_2mx/I_2r=%0.3 f",r);

```

Scilab code Exa 12.23 to calculate 1 motor speed at rated load 2 slip at which max torque occurs 3 max torque

```

1  clear all;
2  clc;
3  Po=20000;
4  N=1440;
5  w_m=2*pi*N/60;
6  T_e=Po/w_m;
7  f1=120;
8  P=4;
9  w_s=4*pi*f1/P;
10 r2=.4;
11 x2=1.6;
12 f2=50;
13 Z1=r2+i*x2*f1/f2;
14 Z=abs(Z1);
15 ph=3;
16 V_s=400;
17 s=(ph/w_s)*(V_s/(Z*sqrt(3)))^2*(r2/T_e);
18 N=w_s*f1/(4*pi)*(1-s);    printf("motor speed at
      rated laod=%0.2 f rpm",N);
19 s_m=r2/imag(Z1);    printf("\nslip at which max
      torque occurs=%0.4 f",s_m);
20 T_em=(3/w_s)*(V_s/sqrt(3))^2/(2*imag(Z1));    printf
      ("\nmax torque=%0.3 f Nm",T_em);

```

Scilab code Exa 12.24 To calculate 1 current and pf at the instant of starting and under max torque conditions 2 starting and max torques

```

1  clear all;
2  clc;
3  V1=400;
4  r1=.6;
5  r2=.4;
6  s=1;
7  x1=1.6;
8  x2=1.6;
9  disp("at starting in normal conditions");
10 I_n=V1/sqrt((r1+r2/s)^2+(x1+x2)^2);    printf("
    current=%0.2 f A",I_n);
11 pf=(r1+r2)/sqrt((r1+r2/s)^2+(x1+x2)^2);    printf("\
    npf=%0.4 f" ,pf);
12 f1=50;
13 w_s=4*pi*f1/4;
14 T_en=(3/w_s)*I_n^2*(r2/s);    printf("\nTorque
    developed=%0.2 f Nm",T_en);
15 disp("motor is operated with DOL starting");
16 I_d=V1/2/sqrt((r1+r2/s)^2+((x1+x2)/2)^2);    printf(
    " current=%0.0 f A",I_d);
17 pf=(r1+r2)/sqrt((r1+r2/s)^2+((x1+x2)/2)^2);
    printf("\npf=%0.2 f",pf);
18 f1=25;
19 w_s=4*pi*f1/4;
20 T_ed=(3/w_s)*I_d^2*(r2/s);    printf("\nTorque
    developed=%0.3 f Nm",T_ed);
21
22 disp("at max torque conditions");
23 s_mn=r2/sqrt((r1)^2+((x1+x2))^2);
24 I_n=V1/sqrt((r1+r2/s_mn)^2+(x1+x2)^2);    printf("
    current=%0.3 f A",I_n);
25 pf=(r1+r2/s_mn)/sqrt((r1+r2/s_mn)^2+(x1+x2)^2);
    printf("\npf=%0.4 f",pf);
26 f1=50;
27 w_s=4*pi*f1/4;

```

```

28 T_en=(3/w_s)*I_n^2*(r2/s_mn);    printf("\nTorque
    developed=%0.2 f Nm",T_en);
29 disp("motor is operated with DOL starting");
30 s_mn=r2/sqrt((r1)^2+((x1+x2)/2)^2);
31 I_d=V1/2/sqrt((r1+r2/s_mn)^2+((x1+x2)/2)^2);
    printf(" current=%0.3 f A",I_d);
32 pf=(r1+r2/s_mn)/sqrt((r1+r2/s_mn)^2+((x1+x2)/2)^2);
    printf("\n pf=%0.3 f",pf);
33 f1=25;
34 w_s=4*pi*f1/4;
35 T_en=(3/w_s)*I_d^2*(r2/s_mn);    printf("\nTorque
    developed=%0.3 f Nm",T_en);

```

Scilab code Exa 12.25 calculate 1 slip for max torque 2 starting and max torques 3 supply voltage reqd

```

1 clear all;
2 clc;
3 x1=1;
4 X_m=50;
5 X_e=x1*X_m/(x1+X_m);
6 V=231;
7 V_e=V*X_m/(x1+X_m);
8 x2=1;
9 r2=.4;
10 r1=0;
11 s_m=r2/(x2+X_e);    printf(" slip at max torque=%0.2 f"
    ,s_m);
12 s_mT=r2/(x2+X_m);    printf("\n slip at max torque=%0
    .5 f",s_mT);
13 f1=50;
14 w_s=4*pi*f1/4;
15 disp("for constant voltage input");
16 T_est=(3/w_s)*(V_e/sqrt(r2^2+(x2+X_e)^2))^2*(r2);
    printf(" starting torque=%0.3 f Nm",T_est);

```



```

17 T_em=(3/w_s)*V_e^2/(2*(x2+X_e));    printf("\
    nmaximum torque developed=%0.2f Nm",T_em);
18 disp("for constant current input");
19 I1=28;
20 T_est=(3/w_s)*(I1*X_m)^2/(r2^2+(x2+X_m)^2)*r2;
    printf("starting torque=%0.3f Nm",T_est);
21 T_em=(3/w_s)*(I1*X_m)^2/(2*(x2+X_m));    printf("\
    nmaximum torque developed=%0.3f Nm",T_em);
22 s=s_mT;
23 I_m=I1*(r2/s+%i*x2)/(r2/s+%i*(x2+X_m));
24 I_m=abs(I_m);
25 V1=sqrt(3)*I_m*X_m;    printf("\nsupply voltage reqd
    =%0.1f V",V1);

```

Scilab code Exa 12.27 to calculate 1 the value of chopper resistance R 2 inductor current 3 duty cycle 4 rectified output voltage 5 efficiency

```

1 clear all;
2 clc;
3 V=420;
4 V1=V/sqrt(3);
5 T_e=450;
6 N=1440;
7 n=1000;
8 T_L=T_e*(n/N)^2;
9 n1=1500;
10 w_s=2*pi*n1/60;
11 w_m=2*pi*n/60;
12 a=.8;
13 I_d=T_L*w_s/(2.339*a*V1);
14 k=0;
15 R=(1-w_m/w_s)*(2.339*a*V1)/(I_d*(1-k));    printf("
    value of chopper resistance=%0.4f ohm",R);
16
17 n=1320;

```

```

18 T_L=T_e*(n/N)^2;
19 I_d=T_L*w_s/(2.339*a*V1);    printf("\ninductor
    current=%0.3f A",I_d);
20
21 w_m=2*pi*n/60;
22 k=1-((1-w_m/w_s)*(2.339*a*V1)/(I_d*R));    printf("\
    nvalue of duty cycle=%0.4f",k);
23
24 s=(n1-n)/n1;
25 V_d=2.339*s*a*V1;    printf("\nrectified o/p voltage=
    %0.3f V",V_d);
26
27 P=V_d*I_d;
28 I2=sqrt(2/3)*I_d;
29 r2=0.02;
30 Pr=3*I2^2*r2;
31 I1=a*I2;
32 r1=0.015;
33 Ps=3*I1^2*r1;
34 Po=T_L*w_m;
35 Pi=Po+Ps+Pr+P;
36 eff=Po/Pi*100;    printf("\nefficiency (in percent)=%
    .2f",eff);

```

Scilab code Exa 12.28 To calculate 1 value of chopper duty cycle 2 efficiency for power input 3 input power factor

```

1 clear all;
2 clc;
3 V=400;
4 V_ph=V/sqrt(3);
5 N_s=1000;
6 N=800;
7 a=.7;
8 I_d=110;

```

```

9 R=2;
10 k=1-((1-N/N_s)*(2.339*a*V_ph)/(I_d*R));    printf("
        value of duty cycle=%0.3f",k);
11 P=I_d^2*R*(1-k);
12 I1=a*I_d*sqrt(2/3);
13 r1=.1;
14 r2=.08;
15 Pr=3*I1^2*(r1+r2);
16 P_o=20000;
17 P_i=P_o+Pr+P;
18 eff=P_o/P_i*100;    printf("\nefficiency=%0.2f",eff);
19
20 I11=sqrt(6)/%pi*a*I_d
21 th=43;
22 P_ip=sqrt(3)*V*I11*cosd(th);
23 pf=P_ip/(sqrt(3)*V*I11);    printf("\ninput power
        factor=%0.4f",pf);

```

Scilab code Exa 12.29 to calculate 1 rotor rectified voltage 2 inductor angle 3 delay angle of inverter 4 efficiency 5 motor speed

```

1 clear all;
2 clc;
3 V=420;
4 V1=V/sqrt(3);
5 N=1000;
6 w_m=2*%pi*N/60;
7 N_s=1500;
8 s=(N_s-N)/N_s;
9 a=.8;
10 V_d=2.339*a*s*V1;    printf("rectified voltage=%0.2f
        V",V_d);
11 T=450;
12 N1=1200;
13 T_L=T*(N/N1)^2;

```

```

14 f1=50;
15 w_s=4*%pi*f1/4;
16 I_d=w_s*T_L/(2.339*a*V1);    printf("\ninductor
    current=%0.2f A",I_d);
17 a_T=-.4;
18 a1=acosd(s*a/a_T);    printf("\ndelay angle of
    inverter=%0.2f deg",a1);
19
20 P_s=V_d*I_d;
21 P_o=T_L*w_m;
22 R_d=0.01;
23 P_i=I_d^2*R_d;
24 I2=sqrt(2/3)*I_d;
25 r2=0.02;
26 r1=0.015;
27 P_rol=3*I2^2*r2;
28 I1=a*I2;
29 P_sol=3*I1^2*r1;
30 P_i=P_o+P_rol+P_sol+P_i;
31 eff=P_o/P_i*100;    printf("\nefficiency=%0.2f",eff);
32 w_m=w_s*(1+(-a_T/a)*cosd(a1)-w_s*R_d*T_L/(2.339*a*V1
    )^2);
33 N=w_m*60/(2*%pi);    printf("\nmotor speed=%0.1f rpm"
    ,N);
34 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 12.30 to find firing angle advance of inveter

```

1 clear all;
2 clc;
3 V=700;
4 E2=V/sqrt(3);
5 N_s=1500;

```

```

6 N=1200;
7 s=(N_s-N)/N_s;
8 V_dd=.7;
9 V_dt=1.5;
10 V_d=3*sqrt(6)*s*E2/%pi-2*V_dd;
11 V1=415;
12 a=acosd((3*sqrt(2)*E2/%pi)^-1*(-V_d+2*V_dt));
13 printf(" firing angle advance=%0.2 f deg",180-a);

```

Scilab code Exa 12.31 to find firing angle advance of inveter

```

1 clear all;
2 clc;
3 V=700;
4 E2=V/sqrt(3);
5 N_s=1500;
6 N=1200;
7 s=(N_s-N)/N_s;
8 V_dd=.7;
9 V_dt=1.5;
10 a=0;
11 u=18; //overlap angle in case of rectifier
12 V_d=3*sqrt(6)*s*E2*(cosd(a)+cosd(a+u))/(2*%pi)-2*
    V_dd;
13
14 V1=415;
15 V_ml=sqrt(2)*V1;
16 u=4; //overlap angle in the inverter
17 //V_dc=-(3*V_ml*(cosd(a)+cosd(a+u))/(2*%pi)-2*V_dt);
18 //V_dc=V_d;
19 //after solving , (1+cosd(u))*cosd(a)-sind(u)*sind(a)
    )=-.6425
20 a=acosd(-.6425/(sqrt((1+cosd(u))^2+sind(u)^2)))-
    atand(sind(u)/(1+cosd(u)));
21 printf(" firing angle advance=%0.2 f deg",180-a);

```

Scilab code Exa 12.32 to find the voltage ratio of the transformer

```
1 clear all;
2 clc;
3 V=700;
4 E2=V;
5 N_s=1500;
6 N=1200;
7 s=(N_s-N)/N_s;
8 V1=415;
9 a_T=s*E2/V1;    printf("voltage ratio of the
                   transformer=%0.4f", a_T);
```

Scilab code Exa 12.33 to calculate 1 supply voltage 2 armature current 3 excitation voltage 4 load angle 5 pull out torque

```
1 clear all;
2 clc;
3 P=6;
4 N_s=600;
5 f1=P*N_s/120;
6 V=400;
7 f=50;
8 V_t=f1*V/f;    printf("supply freq=%0.0f Hz", V_t);
9 T=340;
10 N=1000;
11 T_L=T*(N_s/N)^2;
12 w_s=2*pi*N_s/60;
13 P=T_L*w_s;
14 I_a=P/(sqrt(3)*V_t);    printf("\narmature current=%0
    .2f A", I_a);
```

```

15 Z_s=%i*2;
16 X_s=f1/f*abs(Z_s);
17 V_t=V_t/sqrt(3);
18 Ef=sqrt(V_t^2+(I_a*X_s)^2);
19 printf("\nexcitation voltage=%.2f V",sqrt(3)*Ef);
20 dl=atand(I_a*X_s/V_t);    printf("\nload angle=%.2f
    deg",dl);
21 T_em=(3/w_s)*(Ef*V_t/X_s);    printf("\npull out
    torque=%.2f Nm",T_em);

```

Scilab code Exa 12.34 to calculate 1 load angle 2 line current 3 input pf

```

1 clear all;
2 clc;
3 P=4;
4 f=50;
5 w_s=4*%pi*f/P;
6 X_d=8;
7 X_q=2;
8 T_e=80;
9 V=400;
10 V_t=V/sqrt(3);
11 dl=(1/2)*asind(T_e*w_s/((3/2)*(V_t)^2*(1/X_q-1/X_d))
    );    printf("load angle=%.3f deg",dl);
12 I_d=V_t*cosd(dl)/X_d;
13 I_q=V_t*sind(dl)/X_q;
14 I_a=sqrt(I_d^2+I_q^2);    printf("\narmature current
    =%.2f A",I_a);
15 pf=T_e*w_s/(sqrt(3)*V*I_a);    printf("\ninput power
    factor=%.4f",pf);

```

Scilab code Exa 12.35 To calculate motor speed

```

1 clear all;
2 clc;
3 T_e=3;
4 K_m=1.2;
5 I_a=T_e/K_m;
6 r_a=2;
7 V=230;
8 E_a=(.263*sqrt(2)*V-I_a*r_a)/(1-55/180);
9 w_m=E_a/K_m;
10 N=w_m*60/(2*pi);    printf("motor speed=%.2 f rpm",N
    );

```

Scilab code Exa 12.36 to calculate avg motor torque

```

1 clear all;
2 clc;
3 K_m=1;
4 N=1360;
5 w_m=2*pi*N/60;
6 E_a=K_m*w_m;
7 //after calculations V_t, calculated
8 V_t=163.45;
9 r_a=4;
10 I_a=(V_t-E_a)/r_a;
11 T_e=K_m*I_a;    printf("motor torque=%.4 f Nm",T_e);

```

Scilab code Exa 12.37 to calculate avg motor torque

```

1 clear all;
2 clc;
3 K_m=1;
4 N=2100;
5 w_m=2*pi*N/60;

```



```

6 E_a=K_m*w_m;
7 //after calculations V_t, calculated
8 V_t=227.66;
9 r_a=4;
10 I_a=(V_t-E_a)/r_a;
11 T_e=K_m*I_a;    printf("motor torque=%0.4f Nm",T_e);

```

Scilab code Exa 12.38 to calculate avg motor torque

```

1 clear all;
2 clc;
3 K_m=1;
4 N=840;
5 w_m=2*%pi*N/60;
6 E_a=K_m*w_m;
7 V=230;
8 a=75;
9 V_t=sqrt(2)*V/%pi*(1+cosd(a));
10 r_a=4;
11 I_a=(V_t-E_a)/r_a;
12 T_e=K_m*I_a;    printf("motor torque=%0.4f Nm",T_e);
13 //Answers have small variations from that in the
    book due to difference in the rounding off of
    digits.

```

Scilab code Exa 12.39 to calculate avg motor torque

```

1 clear all;
2 clc;
3 K_m=1;
4 N=1400;
5 w_m=2*%pi*N/60;
6 E_a=K_m*w_m;

```

```

7 V=230;
8 a=60;
9 a1=212;
10 V_t=sqrt(2)*V/%pi*(cosd(a)-cosd(a1))+E_a*(180+a-a1)
    /180;
11 r_a=3;
12 I_a=(V_t-E_a)/r_a;
13 T_e=K_m*I_a;    printf("motor torque=%.3f Nm",T_e);

```

Scilab code Exa 12.40 to calculate avg motor torque

```

1 clear all;
2 clc;
3 K_m=1;
4 N=600;
5 w_m=2*%pi*N/60;
6 E_a=K_m*w_m;
7 V=230;
8 a=60;
9 V_t=2*sqrt(2)*V/%pi*(cosd(a));
10 r_a=3;
11 I_a=(V_t-E_a)/r_a;
12 T_e=K_m*I_a;    printf("motor torque=%.3f Nm",T_e);

```

Scilab code Exa 12.41 to calculate the value of load current and source current

```

1 clear all;
2 clc;
3 r1=.6;
4 r2=.4;
5 s=0.04;
6 x1=1.6;

```

```

7 x2=1.6;
8 Z=(r1+r2/s)+%i*(x1+x2);
9 V=400;
10 I1=V/Z;    printf("source current=%0.3f A and with %
    .1f deg phase",atand(imag(I1)/real(I1)),abs(I1));
11 I2=V/Z;
12 N=1500;
13 w_s=2*%pi*N/60;
14 T_e=(3/w_s)*abs(I2)^2*r2/s;    printf("\nmotor
    torque=%0.2f Nm",T_e);
15 N_r=N*(1-s);
16
17 f=45;
18 N_s1=120*f/4;
19 w_s=2*%pi*N_s1/60;
20 s1=(N_s1-N_r)/N_s1;
21 Z=(r1+r2/s1)+%i*(x1+x2)*f/50;
22 V=360;
23 I1=V/Z;    printf("\nsource current=%0.3f A and with
    %0.1f deg phase",atand(imag(I1)/real(I1)),abs(I1))
    ;
24 I2=V/Z;
25 T_e=(3/w_s)*abs(I2)^2*r2/s1;    printf("\nmotor
    torque=%0.2f Nm",T_e);

```

Chapter 13

Power Factor Improvement

Scilab code Exa 13.1 To calculate load voltage voltage regulation system utilisation and energy consumed

```
1 clear all;
2 clc;
3 V_s=250;
4 R_l=5;
5 I_l=20;
6 disp(" for pf=1");
7 V_l=sqrt(V_s^2-(R_l*I_l)^2);    printf(" load voltage
    =%.2f V",V_l);
8 reg=(V_s-V_l)/V_s*100;    printf("\nvoltage
    regulation=%.2f",reg);
9 pf=1;
10 P_l=V_l*I_l*pf;    //load power
11 P_r=V_s*I_l*pf;    //max powwible system rating
12 utf=P_l*100/P_r;    printf("\nsystem utilisation
    factor=%.3f",utf);
13 printf("\nenergy consumed(in units)=%.1f",P_l/1000);
14 disp(" for pf=.5");
15 pf=.5;
16 //(0.5*V_l)^2+(.866*V_l+R_l*I_l)^2=V_s^2
17 //after solving
```

```

18 V_1=158.35;      printf("load voltage=%0.2f V",V_1);
19 reg=(V_s-V_1)/V_s*100;      printf("\nvoltage
    regulation=%0.2f",reg);
20 P_l=V_1*I_1*pf;      //load power
21 P_r=V_s*I_1;      //max powwible system rating
22 utf=P_l*100/P_r;      printf("\nsystem utilisation
    factor=%0.3f",utf);
23 printf("\nenergy consumed(in units)=%0.2f",P_l/1000);

```

Scilab code Exa 13.2 to calculate the capacitance reqd

```

1 clear all;
2 clc;
3 f=50;
4 V_s=230;
5 disp("at no load");
6 I_m=2;
7 pf=.3;
8 I_c=I_m*sind(acosd(pf));
9 C=I_c/(2*%pi*f*V_s);      printf("value of capacitance
    =%0.3f uF",C*10^6);
10 disp("at half full load");
11 I_m=5;
12 pf=.5;
13 I_c=I_m*sind(acosd(pf));
14 C=I_c/(2*%pi*f*V_s);      printf("value of capacitance
    =%0.3f uF",C*10^6);
15 disp("at full load");
16 I_m=10;
17 pf=.7;
18 I_c=I_m*sind(acosd(pf));
19 C=I_c/(2*%pi*f*V_s);      printf("value of capacitance
    =%0.3f uF",C*10^6);

```

Scilab code Exa 13.3 to find reqd values of capacitor and inductor

```
1 clear all;
2 clc;
3 I_c=10;
4 f=50;
5 V_s=230;
6 C=I_c/(2*%pi*f*V_s); printf("value of capacitance
   =%.3f uF",C*10^6);
7 I_l=10;
8 L=V_s/(2*%pi*f*I_l); printf("\nvalue of inductor=
   %.3f mH",L*1000);
```

Scilab code Exa 13.4 to find the firing angle of the TCR

```
1 clc
2 clear all
3 V_s=230;
4 I_L=10;
5 X_L=V_s/I_L;
6 I_f1=6;
7 //B=2*a-sin(2*a)
8 B=2*%pi-I_f1*%pi*X_L/V_s;
9 a=0;
10 i=1;
11 for a= 0:.01:360
12     b=2*a*%pi/180-sind(2*a);
13     if abs(B-b)<=.001; //by hit and trial
14         i=2;
15         break;
16     end
17 end
```

```

18 printf("firing angle of TCR = %.1f deg",a);
19 //(a-.01)*180/%pi);

```

Scilab code Exa 13.5 to calculate the effective inductance at different firing angles

```

1 clear all;
2 clc;
3 L=.01;
4 disp("for firing angle=90deg");
5 a=90*%pi/180;
6 L_eff=%pi*L/(2*%pi-2*a+sin(2*a)); printf("
    effective inductance=%.0f mH",L_eff*1000);
7 disp("for firing angle=120deg");
8 a=120*%pi/180;
9 L_eff=%pi*L/(2*%pi-2*a+sin(2*a)); printf("
    effective inductance=%.3f mH",L_eff*1000);
10 disp("for firing angle=150deg");
11 a=150*%pi/180;
12 L_eff=%pi*L/(2*%pi-2*a+sin(2*a)); printf("
    effective inductance=%.2f mH",L_eff*1000);
13 disp("for firing angle=170deg");
14 a=170*%pi/180;
15 L_eff=%pi*L/(2*%pi-2*a+sin(2*a)); printf("
    effective inductance=%.3f H",L_eff);
16 disp("for firing angle=175deg");
17 a=175*%pi/180;
18 L_eff=%pi*L/(2*%pi-2*a+sin(2*a)); printf("
    effective inductance=%.2f H",L_eff);
19 disp("for firing angle=180deg");
20 a=180*%pi/180;
21 L_eff=%pi*L/(2*%pi-2*a+sin(2*a)); printf("
    effective inductance=%.3f H",L_eff);
22 //random value at firing angle =180 is equivalent to
    infinity as in answer in book

```

Scilab code Exa 13.6 to find value of inductance

```
1 clear all;
2 clc;
3 Q=100*10^3;
4 V_s=11*10^3;
5 f=50;
6 L=V_s^2/(2*%pi*f*Q);    printf(" effective inductance
    =%.4f H",L);
```

Chapter 14

Miscellaneous Topics

Scilab code Exa 14.1 to calculate dc output voltage rms value of output voltage DF PF and HF

```
1 clear all;
2 clc;
3 V_s=230;
4 V_m=sqrt(2)*V_s;
5 a1=0;
6 a2=45;
7 printf("for two single phase series semiconvertors")
  ;
8 V_0=V_m/%pi*(2+cosd(a1)+cosd(a2));    printf("\navg
  o/p voltage=%0.2 f V",V_0);
9 V_or=V_s*sqrt((1/%pi)*(4*%pi-3*a2*%pi/180+(3/2)*sind
  (2*a2)));    printf("\nrms value of o/p voltage=%0
  .2 f V",V_or);
10 DF=(3+cosd(a2))/(sqrt(2)*sqrt(5+3*cosd(a2)));
    printf("\nDF=%0.4 f",DF);
11 PF=sqrt(2/%pi)*(3+cosd(a2))/sqrt(4*%pi-3*a2*%pi/180)
    ;    printf("\nPF=%0.5 f",PF);
12 HF=sqrt((%pi*(%pi-(3/4)*a2*%pi/180)/(5+3*cosd(a2)))
    -1);    printf("\nHF=%0.5 f",HF);
13
```

```

14 printf("\n\nfor two single phase series full
    convertors");
15 a=45;
16 V_0=2*V_m/%pi*(1+cosd(a));    printf("\navg o/p
    voltage=%0.2 f V",V_0);
17 V_or=2*V_s*sqrt((1/%pi)*(%pi-a2*%pi/180+(1/2)*sind
    (2*a2)));    printf("\nrms value of o/p voltage=%0
    .2 f V",V_or);
18 DF=cosd(a2/2);    printf("\nDF=%0.4 f",DF);
19 PF=sqrt(2/(%pi*(%pi-a2*%pi/180))*(1+cosd(a2)));
    printf("\nPF=%0.4 f",PF);
20 HF=sqrt((%pi*(%pi-a2*%pi/180)/(4+4*cosd(a2)))-1);
    printf("\nHF=%0.2 f",HF);

```
