Scilab Textbook Companion for Advanced Measurements And Instrumentation by A. K. Sawhney[1]

Created by
Avishek Goyal
Measurement
Electrical Engineering
Thapar University patiala
College Teacher
Dr. Sunil Kumar Singla
Cross-Checked by
Ganesh R

August 10, 2013

[1]Funded 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: Advanced Measurements And Instrumentation

Author: A. K. Sawhney

Publisher: Dhanpat Rai And Co., New Delhi

Edition: 1

Year: 2004

ISBN: 9781111991432

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

Chapter 1

Measurement of phase and frequency

Scilab code Exa 1.1 To Calculate the inductance of the circuit and resonant frequency

```
// 1.1
clc;
c1=10^-6;
f1=60;
L1=1/(4*%pi*%pi*(f1^2)*c1);
printf("inductance of the circuit 1 = %.2f H", L1)
f2=50;
w=2*%pi*f2;
R1=100;
Z1=complex(R1,((w*L1)-(1/w*c1)));
//Z2=complex(100+j*((2*%pi*50*L2)-(1/(2*%pi
    *50*1.5*10^-6)))));
//for equal currents in two circuits Z1=Z2
disp('inductance of circuit 2 L2=9.82 H')
L2=9.82;
C2=1.5*10^-6;
Rf2=(1/(2*%pi))*(1/(L2*C2))^0.5;
printf("Resonant frequency of the circuit 1 = %.2f
```

Hz", `Rf2)`

Chapter 2

Primary sensing elements and transducers

Scilab code Exa 2.1 To Calculate the displacement of the free end

```
// 2.1
clc;
t=0.35;
P=1500*10^3;
E=180*10^9;
r=36.5;
x=16;
y=3;
a=%pi*36.5*10^-3;
da=(0.05*a*P/E)*((r/t)^0.2)*((x/y)^0.33)*((x/t)^3);
printf("Displacement of the free end = %.2f m", da)
```

Scilab code Exa 2.2 To calculate the natural length of the spring and dispacement

```
// 2.2
```

```
clc;
P=100*10^3;
A=1500*10^-6;
F=P*A;
Cs=F/3;
Ls=Cs+40;
printf("Natural length of spring = %.2f mm", Ls)
P1=10*10^3;
F1=P1*A;
Ss=3+2*.5;
D=F1/Ss;
printf("\nDisplacement of point C = %.2f mm", D)
```

Scilab code Exa 2.3 To calculate the deflection at center and natural frequency of the diaphragm

```
// 2.3
clc;
D=15*10^-3;
P=300*10^3;
sm=300*10^6;
t=[3*D^2*P/(16*sm)]^0.5*10^3;
printf("Thickness = %.2f mm",t)
P=150*10^3;
v=0.28;
E=200*10^9;
dm=3*(1-v^2)*D^4*P/(256*E*t^3);
printf("\nDeflection at center for Pressure of 150
   kN/m2= %.4f mm",dm)
d=8900;
wn=(20*t*10^-3/D^2)*(E/(3*d*(1-v^2)))^0.5;
printf("\nNatural frequency of the diaphragm =%.0f
   rad/sec",wn)
```

Scilab code Exa 2.4 To calculate the angle of twist

```
// 2.4
clc;
T=100;
G=80*10^9;
d=2*15*10^-3;
th=16*T/(%pi*G*d^3)
printf("Angle of twist= %.6f rad",th)
```

Scilab code Exa 2.5 To calculate Reynoids number and differential pressure and deflection at the center of diaphragm

```
// 2.5
clc;
d=60*10^-3;
Q=80*10^-3;
A=(%pi/4)*d^2;
v=Q/A;
vi=10^-3;
de=10^3;
Re=v*de*d/vi;
printf("Reynoids number = %.2f mm",Re)
d2=60*10^-3;
d1=100*10^-3;
A2=(%pi/4)*d2^2;
M=1/[(1-(d2/d1)^2)^0.5];
Cd=0.99;
w=1*10^3;
Qact=80*10^-3;
Pd=[[Qact/(Cd*M*A2)]^2]*w/(2)*10^-3;
printf("\nDifferential pressure = %.0f kN/m2 ",Pd)
```

```
Po=0.28;
D=10*10^-3;
E=206*10^9;
t=0.2*10^-3;
dm=[3*(1-Po^2)*D^4*Pd]/(256*E*t^3);
def=dm*10^6;
printf("\nDeflection at the center of diaphragm = %
   .2f micro m",def)
```

Scilab code Exa 2.6 To calculate mean velocity of water and velocity of air

```
// 2.6
clc;
Pd=10*10^3;
d=1000;
VmeanW= (2*Pd/d)^0.5;
printf("Mean velocity of water = %.2f m/s",VmeanW)
d=0.65;
Va= (2*Pd/d)^0.5;
printf("\nVelocity of air= %.1f m/s",Va)
```

Scilab code Exa 2.7 To calculate depth of flow

```
// 2.7
disp('let coefficient of discharge Cd=1')
H1=0.9;
L=1.2;
g=9.81;
Q=(2/3)*L*(2*g)^0.5*(H1)^(1.5);
th=45;
H2={Q*(15/8)/[(2*g)^0.5*tan(th)]}^(-1/2.5);
printf("Depth of flow = %.1f m",H2)
```

Scilab code Exa 2.8 To calculate uncertinity in discharge

```
// 2.8
Cd=0.6;
H=0.5;
dH=0.01;
g=9.81;
Q=(8/15)*Cd*(2*g)^0.5*(H)^(2.5);
dQ=(2.5*dH/H)*Q;
printf("Uncertinity in discharge = %.4f m3/s",dQ)
```

Scilab code Exa 2.9 To calculate the displacement and resolution of the potentiometer

```
// 2.9
clc;
Rnormal=10000/2;
Rpl=10000/50;
Rc1=Rnormal-3850;
Dnormal=Rc1/Rpl;
printf("Displacement = %.2f mm",Dnormal)
Rc2=Rnormal-7560;
Dnormal=Rc2/Rpl;
printf("\nDisplacement = %.2f mm",Dnormal)
disp('One displacement is positive and other is negative so two displacements are in the opposite direction')
Re=10*1/200;
printf("\nResolution = %.2f mm",Re)
```

Scilab code Exa 2.10 To plot the graph of error versus K

```
// 2.10
clc;
K=[0    0.25    0.5    0.75    1];
V=[0    -0.174    -0.454    -0.524    0];
plot(K,V)
```

Scilab code Exa 2.11 Calculating the output voltage

```
//2.11
clc;
RAB=125;
Rtotal=5000;
R2=75/125*Rtotal;
R4=2500;
ei=5;
eo=[(R2/Rtotal)-(R4/Rtotal)]*ei;
printf("Output voltage = %.1f V",eo)
```

Scilab code Exa 2.12 Calculating the maximum excitation voltage and the sensitivity

```
// 2.12
clc;
Rm=10000;
Rp=Rm/15;
R=600;
P=5;
ei= (P*R)^0.5;
printf("Maximum excitation voltage = %.1f V",ei)
S=ei/360;
printf("\nSensitivity = %.3f V/degree",S)
```

Scilab code Exa 2.13 Calculating the resolution of the potentiometer

```
// 2.13
clc;
Rwga=1/400;
Re=Rwga/5;
printf("Resolution = %.4f mm",Re)
```

Scilab code Exa 2.14 Checking the suitability of the potentiometer

```
// 2.14
clc;
mo=0.8;
sr=250;
sm=sr/mo;
R=sm*1*10^-3;
printf("Resolution of 1mm movement = %.4f degree/mm"
   ,R)
Rq=300/1000;
printf("\nRequired Resolution of 1mm movement = %.3f
    degree/mm",Rq)
disp('Since the resolution of potentiometer is
   higher than the resolution required so it is
   suitable for the application')
```

Scilab code Exa 2.15 Checking the suitability of the potentiometer

```
// 2.15
clc;
```

```
Pd=(10^2)/150;
printf("Power dissipation = %.3f W",Pd)
th_pot=80+Pd*30;
PDa=(10*10^-3)*(th_pot-35);
printf("\nPower dissipation = %.3f W",PDa)
disp('Since power dissipation is higher than the
    dissipation allowed so potentiometer is not
    suitable')
```

Scilab code Exa 2.16 Calculating the possions ratio

```
// 2.16
clc;
Gf=4.2;
v=(Gf-1)/2;
disp(v,'Possion s ratio=')
```

Scilab code Exa 2.17 Calculating the value of the resistance of the gauges

```
// 2.17
clc;
strain=-5*10^-6;
Gf=-12.1;
R=120;
dR_nickel=Gf*R*strain;
printf("Change in resistance of nickel = %.3f ohm",
    dR_nickel)
Gf=2;
R=120;
dR_nicrome=Gf*R*strain;
printf("\nChange in resistance of nicrome = %.3f ohm
    ",dR_nicrome)
```

Scilab code Exa 2.18 Calculating the percentage change in value of the gauge resistance

```
// 2.18
clc;
s=100*10^6;
E=200*10^9;
strain=s/E;
Gf=2;
r_per_unit=Gf*strain*100;
printf("Percentage change in resistance = %.1f ",
    r_per_unit)
```

Scilab code Exa 2.19 Calculating the Gauge factor

```
//2.19
clc;
b=0.02;
d=0.003;
I=(b*d^3)/12;
E=200*10^9;
x=12.7*10^-3;
l=0.25;
F=3*E*I*x/l^3;
x=0.15;
M=F*x;
t=0.003;
s=(M*t)/(I*2);
strain=s/E;
dR=0.152;
R=120;
Gf=(dR/R)/strain;
```

```
printf("Gauge factor = %.2f ",Gf)
```

Scilab code Exa 2.20 Calculating the change in length and the force applied

```
// 2.20
clc;
dR=0.013;
R=240;
l=0.1;
Gf=2.2;
dl=(dR/R)*l/Gf*10^6;
printf(" Change in length= %.1f um ",dl)

strain=dl*10^-6/l;
E=207*10^9;
s=E*strain;
A=4*10^-4;
F=s*A;
printf("\n Force= %.2f N ",F)
```

Scilab code Exa 2.21 To calculate the linear approximation

```
// 2.21
clc;
th1=30;
th2=60;
th0=th1+th2/2;
Rth1=4.8;
Rth2=6.2;
Rth0=5.5;
ath0=(1/Rth0)*(Rth2-Rth1)/(th2-th1);
printf("\n alpha at o degree= %.4f /degree C ",ath0)
```

```
disp('5.5[1+0.0085(th-45)]')
```

Scilab code Exa 2.22 Calculate the linear approximation

```
// 2.22
clc;
th1=100;
th2=130;
th0=th1+th2/2;
Rth1=573.40;
Rth2=605.52;
Rth0=589.48;
ath0=(1/Rth0)*(Rth2-Rth1)/(th2-th1);
printf("alpha at o degree= %.5f /degree C ",ath0)
disp('Linear approximation is: Rth= 589.48[1+0.00182(th-115)]')
```

Scilab code Exa 2.23 Calculate the resistance and the temperature

```
// 2.23
clc;
Rth0=100;
ath0=0.00392;
dth=65-25;
R65=Rth0*[1+ath0*dth];
printf("resistance at 65 degree C= %.2f ohm ",R65)
th={[(150/100)-1]/ath0}+25;
printf("\n Temperature = %.2f degree C ",th)
```

Scilab code Exa 2.24 Calculate the resistance

```
// 2.24
clc;
Rth0=10;
ath0=0.00393;
dth=150-20;
R150=Rth0*[1+ath0*dth];
printf("Resistance at 150 degree C=%.2f ohm",R150)
```

Scilab code Exa 2.25 Calculate the time

```
// Calculate the time
clc;
th=30;
th0=50;
tc=120;
t=-120*[log(1-(th/th0))];
printf("Time= %.2f s ",t)
```

Scilab code Exa 2.26 To Calculate the resistance

```
//2.26
clc;
R25=100;
ath=-0.05;
dth=35-25;
R35=R25*[1+ath*dth];
printf("Resistance at 35 degree C= %.2f ohm ",R35)
```

Scilab code Exa 2.27 Calculating the resistance

```
// 2.27
clc;
Ro=3980;
Ta=273;
//3980= a*3980*exp(b/273)
Rt50=794;
Ta50=273+50;
//794= a*3980*exp(b/323)
//on solving
//a=30*10^-6, b=2843
Ta40=273+40;
Rt40=(30*10^-6)*3980*exp(2843/313);
printf("Resistance at 40 degree C= %.2f ohm ",Rt40)
Rt100=(30*10^-6)*3980*exp(2843/373);
printf("\nResistance at 100 degree C= %.2f ohm ",
   Rt100)
```

Scilab code Exa 2.28 calculating the change in temperature

```
// 2.28
clc;
th=((1-1800/2000)/0.05)+70;
dth=th-70;
printf("Change in temperature= %.1f degree C ",dth)
```

Scilab code Exa 2.29 calculating the frequencies of oscillation

```
// 2.29
clc;
C=500*10^-12;
R20=10000*(1-0.05*(20-25));
f20=1/(2*%pi*R20*C);
```

```
printf("Frequency of oscillation at 20 degree C = %.2f Hz ",f20)
R25=10000*(1-0.05*(25-25));
f25=1/(2*%pi*R25*C);
printf("\nFrequency of oscillation at 25 degree C = %.2f Hz ",f25)
R30=10000*(1-0.05*(30-25));
f30=1/(2*%pi*R30*C);
printf("\nFrequency of oscillation at 30 degree C = %.2f Hz ",f30)
```

Scilab code Exa 2.30 Calculating the sensitivity and maximum output voltage

```
// 2.30
clc;
Se_thermocouple=500-(-72);
printf("Sensitivity of thermocouple= %.1f micro V/degree C",Se_thermocouple)
Vo=Se_thermocouple*100*10^-6;
printf("\nMaximum output voltage= %.2f V ",Vo)
```

Scilab code Exa 2.31 Calculating the temperature

```
// 2.31
clc;
ET=27.07+0.8;
printf("Required e.m.f.= %.2f mV ",ET)
disp('Temperature corresponding to 27.87 mV is 620 degree C')
```

Scilab code Exa 2.32 Calcating the series resistance and approximate error

```
// 2.32
clc;
Rm=50;
Re=12;
E=33.3*10^-3;
i=0.1*10^-3;
Rs=(E/i)-Rm-Re;
printf("Series resistance=%.2f ohm",Rs)
Re=13;
i1=E/(Rs+Re+Rm);
AE=[(i1-i)/i]*800;
printf("\nApproximate error due to rise in
   resistance of 1 ohm in Re=%.2f degree C",AE)
R_change=50*0.00426*10;
i1=E/(Rs+Re+Rm+R_change);
AE=[(i1-i)/i]*800;
printf("\nApproximate error due to rise in Temp. of
   10=%.2f degree C",AE)
```

Scilab code Exa 2.33 Calculate the values of resistance R1 and R2

```
// 2.33
clc;
E_20=0.112*10^-3;// emf at 20degree C
E_900=8.446*10^-3;
E_1200=11.946*10^-3;
E1=E_900-E_20;
E2=E_1200-E_20;
//E1=1.08*R1/(R1+2.5+R2         (i)
//E2=1.08*(R1+2.5)/(R1+2.5+R2         (ii)
//on solving (i) and (ii)
R1=5.95;
```

```
R2=762.6;
printf("Value of resistance R1=%.2f ohm",R1)
printf("\nValue of resistance R2=%.2f ohm",R2)
```

Scilab code Exa 2.34 Design the circuit

```
// 2.34
clc;
th=20;
Vz=2.73+th*10*10^-3;
Voffset=-2.73;
Vout=Vz+Voffset;
Rbias=(5-0.2)/10^-3;
Rzero=500;
th=50;
Vz=2.73+th*10*10^-3;
VmaxT=Vz+Voffset;
Vsupply=5;
Rl=(VmaxT*Rbias)/(Vsupply-VmaxT);
printf("Value of resistance R1=%.2f ohm",R1)
disp('value of resistance RL>>Rl')
```

Scilab code Exa 2.35 Calculate the change in inductance

```
// 2.35
clc;
L1=2;
La=1-0.02;
Lnew=2/La;
dl=Lnew-L1;
printf("Change in inductance=%.2f mH",dl)
```

Scilab code Exa 2.36 Calculate the percentage linearity

```
// 2.36
clc;
linearity_percentage=(0.003/1.5)*100;
printf("percentage linearity=%.2f ",
   linearity_percentage)
```

Scilab code Exa 2.37 Calculate senstivity of the LVDT Instrument and resolution of instrument in mm

```
// 2.37
clc;
displacement=0.5;
Vo=2*10^-3;
Se_LVDT=Vo/displacement;
printf("senstivity of the LVDT=%.3f V/mm",Se_LVDT)
Af=250;
Se_instrument=Se_LVDT*Af;
printf("\nSenstivity of the instrument=%.1f V/mm",
   Se_instrument)
sd=5/100;
Vo_min=50/5;
Re_instrument=1*1/1000;
printf("\nresolution of instrument=%.3f mm",
   Re_instrument)
```

Scilab code Exa 2.38 calculate the deflection maximum and minimum force

```
// 2.38
clc;
b=0.02;
t=0.004;
I=(1/12)*b*t^3;
F=25;
l=0.25;
E=200*10^9;
x=(F*l^3)/(3*E*I);
printf("deflection=%.2f m",x)
DpF=x/F;
Se=DpF*0.5*1000;
Re=(10/1000)*(2/10);
F_min=Re/Se;
F_max=10/Se;
printf("\nminimum force=%.2f N",F_min)
printf("\nmaximum force=%.2f N",F_max)
```

Scilab code Exa 2.39 calculating the sensitivity of the transducer

```
// 2.39
clc;
disp('permittivity of the air e0=8.85*10^-12')
e0=8.85*10^-12;
w=25*10^-3;
d=0.25*10^-3;
Se=-4*e0*w/d;
printf("sensitivity of the transducer=%.2f F/m",Se)
```

Scilab code Exa 2.40 Calculate the value of the capacitance afte the application of pressure

```
// 2.40
```

```
clc;
C1=370*10^-12;
d1=3.5*10^-3;
d2=2.9*10^-3;
C2=C1*d1*10^12/d2;
printf("the value of the capacitance afte the application of pressure=%.2f pF",C2)
```

Scilab code Exa 2.41 Calculate the change in frequency of the oscillator

```
// 2.41
clc;
fo1=100*10^3;
d1=4;
d2=3.7;
fo2=[(d2/d1)^0.5]*fo1;
dfo=fo1-fo2/10^-3;
printf("change in frequency of the oscillator=%.1f kHz",dfo)
```

Scilab code Exa 2.42 Calculate the dielectric stress change in value of capacitance

```
// 2.42
clc;
L_air=(3.1-3)/2;
D_stress=100/L_air;
e0=8.85*10^-12;
l=20*10^-3;
D2=3.1;
D1=3;
C=(2*%pi)*e0*l*10^12/(log(D2/D1));
printf("Capacitance=%.1f pF",C)
```

```
l=(20*10^-3)-(2*10^-3);
C_new=(2*%pi)*e0*l/(log(D2/D1));
C_change=C-C_new*10^12;
printf("\nchange in Capacitance=%.1f pF",C_change)
```

Scilab code Exa 2.43 Calculate the value of time constant phase shift series resistance amplitude ratio and voltage sensitivity

```
//2.43
clc;
M=0.95;
w=2*%pi*20;
tc=(1/w)*[(M^2)/(1-M^2)]^0.5;
printf("Time constant=%.2f s",tc)
ph={(%pi/2)-[atan(w*tc)]}*(180/%pi);
printf("\nPhase shift=%.1f deg",ph)
C=(8.85*10^-12*300*10^-6)/(0.125*10^-3);
R=tc*10^-6/C;
printf("\nSeries resistance=%.0f Mohm",R)
M=1/(1+(1/(2*%pi*5*tc)^2))^0.5;
printf("\nAmplitude ratio=%.1f ",M)
Eb=100;
x=0.125*10^-3;
Vs=Eb/x;
printf("\nVoltage sensitivity=%.1f V/m",Vs)
```

Scilab code Exa 2.44 Calculate the change in capacitance and ratio

```
//2.44
clc;
e0=8.85*10^-12;
A=500*10^-6;
d=0.2*10^-3;
```

```
C=e0*A/d;
d1=0.18*10^-3;
C_new=e0*A/d1;
C_change=C_new-C;
Ratio=(C_change/C)/(0.02/0.2);
printf("\nratio of per unit change of capacitance to per unit change of diaplacement=%.2f",Ratio)
d1=0.19*10^-3;
e1=1;
d2=0.01*10^-3;
e2=8;
C=(e0*A)/((d1/e1)+(d2/e2));
d1_new=0.17*10^-3;
C_new=(e0*A)/((d1_new/e1)+(d2/e2));
C_change=C_new-C;
Ratio=(C_change/C)/(0.02/0.2);
printf("\n New ratio of per unit change of capacitance to per unit change of diaplacement=%.2f",Ratio)
```

Scilab code Exa 2.47 Calculate the output voltage and charge sensitivity

```
// 2.47
clc;
g=0.055;
t=2*10^-3;
P=1.5*10^6;
Eo=g*t*P;
printf("Output voltage=%.0f V",Eo)
e=40.6*10^-12;
d=e*g*10^12;
printf("\n Charge sensitivity=%.2f pC/N",d)
```

Scilab code Exa 2.48 Calculate the force

```
// 2.48
clc;
g=0.055;
t=1.5*10^-3;
Eo=100;
P= Eo/(g*t);
A=25*10^-6;
F=P*A;
printf("\n Force=%.0f N",F)
```

Scilab code Exa 2.49 Calculate the strain charge and capacitance

```
// 2.49
clc;
A=25*10^-6;
F=5;
P=F/A;
d=150*10^-12;
e=12.5*10^-9;
g=d/(e);
t=1.25*10^-3;
Eo=(g*t*P);
strain=P/(12*10^6);
Q=d*F*10^12;
C=Q/Eo;
printf("\n strain=%.4f ",strain)
printf("\n Charge=%.0f pC",Q)
printf("\n capacitance=%.0f pF",C)
```

Scilab code Exa 2.50 Calculate peak to peak voltage swing under open and loaded conditions and calculate maximum change in crystal thickness

```
// 2.50
clc;
d=2*10^-12;
t=1*10^-3;
Fmax=0.01;
e0=8.85*10^-12;
er=5;
A=100*10^-6;
Eo_peak_to_peak=2*d*t*Fmax*10^3/(e0*er*A);
printf("\n peak voltage swing under open conditions=
   %.2f mV",Eo_peak_to_peak)
Rl=100*10^6;
Cl=20*10^-12;
d1=1*10^-3;
Cp=e0*er*A/d1;
C=Cp+Cl;
w=1000;
m=[w*Cp*Rl/[1+(w*C*Rl)^2]^0.5];
El_peak_to_peak=[2*d*t*Fmax*10^3/(e0*er*A)]*m;
printf("\n peak voltage swing under loaded
   conditions=%.2f mV",El_peak_to_peak)
E=90*10^9;
dt=2*Fmax*t*10^12/(A*E);
printf("\n Maximum change in crystal thickness=%.2f
   pm",dt)
```

Scilab code Exa 2.51 Calculate the minimum frequency and phase shift

```
// 2.51
clc;
M=0.95;
tc=1.5*10^-3;
w=(1/tc)*[(M^2)/(1-M^2)]^0.5;
printf("\n Minimum frequency=%.2f rad/sec",w)
ph={(%pi/2)-[atan(w*tc)]}*(180/%pi);
```

```
printf("\n Phase shift=%.2f deg",ph)
```

Scilab code Exa 2.52 calculate sensitivity of the transducer high frequency sensitivity Lowest frequency and Calculate external shunt capacitance and high frequency sensitivity after connecting the external shunt capacitance

```
//2.52
clc;
Kq=40*10^-3;
Cp=1000*10^-12;
K=Kq/Cp;
printf(" Sensitivity of the transducer=%.2f V/m",K)
Cc=300*10^-12;
Ca=50*10^-12;
C=Cp+Cc+Ca;
Hf=Kq/C;
printf("\n High frequency sensitivity =%.2f V/m",Hf)
R=1*10^6;
tc=R*C;
M=0.95;
w=(1/tc)*[(M^2)/(1-M^2)]^0.5;
f=w/(2*%pi);
printf("\n Minimum frequency=%.2f sec",f)
disp('now f=10Hz')
f=10;
w=2*%pi*f;
tc=(1/w)*[(M^2)/(1-M^2)]^0.5;
C_new=tc/R;
Ce=(C_new-C)*10^6;
printf("\n External shunt capacitance=%.2f pF",Ce)
Hf_new=Kq/C_new;
printf("\n new value of high frequency sensitivity=%
    .2f V/m",Hf_new)
```

Scilab code Exa 2.53 Calculate output voltage 10 ms after the application of impulse

```
// 2.53
clc;
R=10^6;
C=2500*10^-12;
tc=R*C;
t=2*10^-3;
d=100*10^-12;
F=0.1;
el=10^3*{d*F*[exp(-t/tc)]/C};
printf("Voltage just before t=2ms =%.2f mV",e1)
el_after=10^3*{d*F*[exp(-t/tc)-1]/C};
disp(el_after,'voltage just after t=2ms (mV)')
printf("Voltage just after t=2ms =%.2f mV",el_after)
disp('when t=10ms')
t=10*10^-3;
T=2*10
e_10=10^3*{d*F*[exp((-T/tc)-1)]*{exp(-(t-T))/tc}/C}
printf("output voltage 10 ms after the application
    of impulse =%.0f mV",e_10)
```

Scilab code Exa 2.54 prove time constant should be approximately 20T to keep undershoot within 5 percent

```
// 2.54
clc;
disp('Let T=1');
T=1;
el=0.95;
tc=-T/log(el);
```

```
printf("Time constant =%.2f s",tc)
disp('as T=1 so time constant should be
    approximately equal to 20T')
```

Scilab code Exa 2.55 Calculate the output voltage

```
//2.55
clc;
Kh=-1*10^-6;
I=3;
B=0.5;
t=2*10^-3;
Eh=Kh*I*B*10^3/t;
printf("output voltage =%.2f mV",Eh)
```

Scilab code Exa 2.56 Calculate the external resistance and dark current

```
//2.56
clc;
R1=(30/10*10^-3)-1000;
printf("External resistance required =%.3f ohm",R1)
Id=30*10^3/((2*10^3)+(100*10^3))
printf("\nDark current =%.2f mA",Id)
```

Scilab code Exa 2.57 Calculate the output voltage of bridge

```
//2.57
clc;
Vb=10-(10/((2*10^3))*10^3);
disp(Vb,'Potential of point b, Vb=')
```

```
Vd=10-(10/((3*10^3))*2*10^3);
disp(Vd,'Potential of point d, Vd=')
Ebd=Vb-Vd;
printf("\nOutout voltage of bridge =%.2f V",Ebd)
```

Chapter 3

Measurement of non electrical quantities

Scilab code Exa 3.1 To calculate the optimum setting

```
// 3.1
clc;
Aou=700*25*1/100;
Aol=100*25*1/100;
AouPtP= 2*Aou;
AolPtP= 2*Aol;
Se1=1;
D1=AouPtP/Se1;
printf("\ndeflection of screen corresponding to
    maximum pressure for sensitivity of 1mV/mm =%.1f
    mm",D1)
disp('sinch the length of the screen is 100mm so
    waveform is out of range and hence sensitivity
    setting of 1mV/mm should not be used')
Se2=5;
D2=AouPtP/Se2;
printf("\ndeflection of screen corresponding to
    maximum pressure for sensitivity of 5mV/mm =%.1f
    mm",D2)
```

```
disp('delection is within the range')
Se3=20;
D3=AouPtP/Se3;
printf("\ndeflection of screen corresponding to
    maximum pressure for sensitivity of 20mV/mm =%.1f
     mm",D3)
disp('delection is within the range')
Se4=100;
D4=AouPtP/Se4;
printf("\ndeflection of screen corresponding to
    maximum pressure for sensitivity of 10mV/mm =%.1f
     mm",D4)
disp('delection is within the range')
Se5=500;
D5=AouPtP/Se5;
printf("\ndeflection of screen corresponding to
    maximum pressure for sensitivity of 500mV/mm =%.1
    f mm",D5)
disp('delection is within the range')
disp('since the sensitivity of 5mV/mm gives higher
    deflection so it is the optimum sensitivity')
```

Scilab code Exa 3.2 Calculating the radius of curvature

```
//3.2
clc;
tA=1;
tB=1;
m=tA/tB;
EB=147;
EA=216;
T2=200;
T1=25;
n=EB/EA;
T=T2-T1;
```

```
A=12.5*10^-6;
B=1.7*10^-6;
a=3*(1+m)^2;
b=(1+m*n)*((m^2)+1/(m*n));
c= (6*(A-B)*T*(1+m)^2);
r=(a+b)/c;
printf("\nRadius of curvature =%.2f mm",r)
```

Scilab code Exa 3.3 Calculating the vertical displacement

```
//3.3
clc;
t=2;
T2=180;
T1=20;
T=T2-T1;
A=12.5*10^-6;
r=t/(2*T*A);
printf("\nRadius of curvature =%.0f mm",r)
Th=40/500;
y=r*(1-cos(Th));
printf("\nvertical displacement =%.0f mm",y)
```

Scilab code Exa 3.4 To calculate the true temperature

```
//3.4
clc;
Ta=1480+273;
Tf=0.8;
T=Tf^-0.25*Ta;
printf("\nTrue temperature =%.2f degree K",T)
Tc=T-273;
printf("\nTrue temperature =%.2f degree C",Tc)
```

Scilab code Exa 3.5 To calculate the error in the measurement of temperature

```
// 3.5
clc;
ATC1=1065;
AT=ATC1+273;
Em1=0.82;
Ta=(Em1^(-0.25))*AT;
Em2=0.75;
Taa=(Em2^-0.25)*Ta;
ATC2=Taa-273;
E=ATC1-ATC2;
printf("Error in temperature measurement=%.2f degree
    C",E)
```

Scilab code Exa 3.6 To calculate average flow rate and percentage decrease in voltage

```
// 3.6
clc;
EL=0.1;
Zo=250*10^3;
ZL=2.5*10^6;
Eo=EL*(1+(Zo/ZL));
B=0.1;
l=50*10^-3;
G=1000;
v=Eo/(B*l*G);
printf("Average flow rate=%.2f degree m/s",v)
Zon=1.2*250*10^3;
```

```
ELn=2*Eo/(1+(Zon/ZL));
PDV=[(0.2-ELn)/0.2]*100;
printf("Percentage decrease in voltage=%.2f degree m
   /s",PDV)
```

Chapter 4

Telemetry and data acquisition system

Scilab code Exa 4.1 To calculate the frequencies present in output

```
// 4.1
clc;
fc=1000;
disp('In addition to carrier frequency of 1000kHz the other upeer and lower frequencies are')
fs1=0.3;
fu1=fc+fs1;
printf("\nUpper side band frequency for modulating frequency of 300 Hz =%.1f kHz",fu1)
fl1=fc-fs1;
printf("\nLower side band frequency for modulating frequency of 300 Hz =%.1f kHz",fl1)
fs2=0.8;
fu2=fc+fs2;
printf("\nUpper side band frequency for modulating frequency of 800 Hz =%.1f kHz",fu2)
fl2=fc-fs2;
printf("\nLower side band frequency for modulating frequency of 800 Hz =%.1f kHz",fl2)
```

```
fs3=2;
fu3=fc+fs3;
printf("\nUpper side band frequency for modulating
   frequency of 2kHz =%.1f kHz",fu3)
fl3=fc-fs3;
printf("\nLower side band frequency for modulating
   frequency of 2kHz =%.1f kHz",fl3)
```

Scilab code Exa 4.2 To calculate the frequencies range occupied by the side bands

```
// 4.2
clc;
L=50*10^-6;
C=1*10^-9;
fc=1/(2*%pi*(L*C)^0.5);
fs1=10000;
fu1=(fc+fs1)*10^-3;
printf("\nUpper side band frequency =%.2f kHz",fu1)
fl1=(fc-fs1)*10^-3;
printf("\nLower side band frequency =%.2f kHz",fl1)
```

Scilab code Exa 4.3 To calculate the radiation power

```
// 4.3
clc;
Pc=50;
m=0.85;
Pt=Pc*(1+(m^2/2))
printf("Radiation Power =%.2f kW",Pt)
```

Scilab code Exa 4.4 To calculate the modulation indicies

```
// 4.4
clc;
delta=4.8;
Es=2.4;
K=delta/Es;
Es1=7.2;
delta1=K*Es1;
Es2=10;
delta2=K*Es2;
fs1=500*10^-3;
mf1=delta/fs1;
printf("\nmodulation index for Es (2.4) =%.1f",mf1)
mf2=delta1/fs1;
printf("\nmodulation index for Es(7.2)=%.1f",mf2)
mf3=delta2/fs1;
printf("\nmodulation indexfor Es(10) =%.1f",mf3)
```

Scilab code Exa 4.5 Calculating the carrier modulating frequencies modulation index maximum deviation and power dissipation

```
// 4.5
clc;
wc=6*10^8;
fc=(wc)/(2*%pi)*10^-3;
printf("\ncarrier frequency =%.1f kHz",fc)
ws=1250;
fs=(ws)/(2*%pi);
printf("\nmodulating frequency =%.1f Hz",fs)
mf=5;
delta=mf*fs;
printf("\nmaximum deviation =%.1f Hz",delta)
Rms=12/(2^0.5);
P=Rms^2/10;
```

```
printf("\nPower dissipated =%.1f W",P)
```

Scilab code Exa 4.6 Calculating bandwidth of intelligence and rise time

```
// 4.6
clc;
delta=10;
fs=2;
mf=delta/fs;
BW=16*mf;
printf("\nBand width =%.0f kHz",BW)
```

Scilab code Exa 4.7 Write down the voltage expression

```
// 4.7
clc;
fc=100*10^6;
wc=2*%pi*fc;
fs=6*10^3;
ws=2*%pi*fs;
delta=60*10^3;
mf=delta/fs;
mp=mf;
disp('epm=8sin(0.6283*10^9t+10 sin 37.7*10^3t)V')
disp('for a signal voltage of 4 V')
mp=4*10/3;
disp('epm=8sin(0.6283*10^9t+13.33 sin 37.7*10^3t)V')
disp('for a fs of 8 kHz')
disp('epm=8sin(0.6283*10^9t+13.33 sin 50.27*10^3t)V'
  )
```

Scilab code Exa 4.8 Calculating the quantization error

```
// 4.8
clc;
n=5;
Ql=2^n;
Range=(Ql-1)*1;
disp('range is 0-31 V with each step representing 1V
   ')
Qe=27.39-27;
printf("\nquattization error =%.1f V",Qe)
```

Scilab code Exa 4.9 Calculating the minimum carrier chnnel bandwidth

```
// 4.9
clc;
disp('For amplitude modulation')
MCCW=2*1;
printf("\nMinimum width of carrier channel =%.1f kHz
   ",MCCW)
disp('For frequency modulation')
MCCW=2*(1.5+1);
printf("\nMinimum width of carrier channel =%.1f kHz
   ",MCCW)
disp('For pulse code modulation')
MCCW=8*1;
printf("\nMinimum width of carrier channel =%.1f kHz
   ",MCCW)
```

Scilab code Exa 4.10 Calculating the fuel level

```
// 4.10
clc;
```

```
Fc=430-370;
disp('At 403 change in frequency')
Fc1=403-370;
Fuel_level=Fc1*3000/Fc;
printf("\nFuel level =%.1f L",Fuel_level)
```

Scilab code Exa 4.11 Calculating the lowest practical sampling rate

```
// 4.11
clc;
disp('for good quality data the sampling rate should
    be at least 5 times the data frequency for one
   channel')
channel=5;
f=50;
sampling_rate=5*channel*f;
printf("\nsampling rate =%.1f samples per second",
   sampling_rate)
```

Scilab code Exa 4.12 Calculating the maximum possible data transmission rate minimum sampling rate per channel and maximum number of channels

```
//4.12
clc;
Vs=7;
Vn=1;
fh=10^3;
H=2*fh*log2(1+(Vs/Vn));
printf("\nMaximum possible data transmission rate =%
   .1f bits per second",H)
Sampling_rate=2*fh;
```

```
printf("\nminimum sampling rate per channel =%.1f samples per second",Sampling_rate)
C_max=85714/2000;
printf("\nmaximum number of channels =%.0f ",C_max)
```

Scilab code Exa 4.13 Calculating the cut off frequency

```
//4.13
clc;
d_rate=100;
fc= 0.5*d_rate;
printf("cutt off frquency =%.1f kHz ",fc)
```

Scilab code Exa 4.14 Calculating the number of transmission channels

```
//4.14
clc;
disp('The modulated carrier will have a bandwidth of 100MHz+/- 1kHz.')
disp('therefore we can have 5 channels each transmitting a 1KHz data for 5kHz bandwidth')
```

Scilab code Exa 4.15 Calculating bandwidth of intelligence and rise time

```
// 4.15
clc;
Fd=7.5*165*10^3/100;
mf=5;
Bandwidth=Fd/mf;
```

```
printf("Bandwidth of intelligence =%.1f Hz ",Bandwidth)
Tr=0.35/Bandwidth*10^6;
printf("\nRise time=%.1f us ",Tr)
```

Chapter 5

Advanced measuring instruments

Scilab code Exa 5.1 calculating number of turns and current

```
// 5.1
clc;
D=8*10^-3;
A=D^2;
disp(A,'A=')
J=8*10^-3;
K=16*10^-3;
B=4*J*K;
disp (B,'B=')
disp('since A<B so the instrument is underdamped')
th=(100*%pi)/180;
i=10*10^-3;
F=0.2*10^-6;
G=(K*th+F)/i;
l=65*10^-3;
d=25*10^-3;
N=G/(B*l*d);
printf("\nNumber of turns=%.0f ",N)
i=F/G*10^6;
```

```
printf("\ncurrent required to overcome friction=%.1f uA ",i)
```

Scilab code Exa 5.2 Calculating the frequency range

```
// 5.2
clc;
eta=0.6;
fn=2400;
M=0.98;
//M=1/[[(1-u^2)^2]+(2*u*eta)^2]^0.5; ..........(i)
// On solving the above equation we get u=0.79
u=0.79;
fu=u*fn;
printf("\nupper value of range=%.0f Hz",fu)

//Now let M=1.02, on solving equation (i) we have u=0.29
u=0.29;
fl=u*fn;
printf("\nlower value of range=%.0f Hz",fl)
disp('So, the range of the frequency is from 696 to 1896 Hz')
```

Scilab code Exa 5.3 Calculating the phase displacement for the fundamental and 5th harmonic

```
// 5.3
clc;
eta=0.64;
u=0.1;
alpha_1=atand(2*eta*u/(1-u^2))
```

```
printf("\nphase displacement for the fundamental=%.2
    f degree",alpha_1)
u=0.5;
alpha_5=atand(2*eta*u/(1-u^2))
printf("\nphase displacement for the 5th harmonic=%
    .2f degree",alpha_5)
```

Scilab code Exa 5.4 Calculate the percentage errors in the production of harmonics

```
//5.4
clc;
To=1/2000;
T=1/50;
//Rn=1/(1+n^2*(To/T)^2)
R1=1/(1+1^2*(To/T)^2);
R3=1/(1+3^2*(To/T)^2);
R5=1/(1+5^2*(To/T)^2);
R7=1/(1+7^2*(To/T)^2);
R11=1/(1+11^2*(To/T)^2);
R13=1/(1+13^2*(To/T)^2);
PE3=(R3-1/1)*100;
printf("Percentage error for the production of 3rd
    harmonics=%.2f",PE3)
PE5=(R5-1/1)*100;
printf("\nPercentage error for the production of 5th
     harmonics=%.2f",PE5)
PE7=(R7-1/1)*100;
printf("\nPercentage error for the production of 7th
     harmonics=%.2f",PE7)
PE11=(R11-1/1)*100;
printf("\nPercentage error for the production of 11
    th harmonics=%.2f",PE11)
PE13=(R13-1/1)*100;
printf("\nPercentage error for the production of 13
```

```
    th harmonics=%.2f",PE13)
//displacement of nth harmonic alpha=atan2*n/((T/To)
    -n^2*(To/T))
alpha_1=atand(2*1/((T/To)-(1^2*(To/T))));
alpha_13=(atand(2*13/((T/To)-(13^2*(To/T)))));
alpha_1_equivalent_13=13*alpha_1;
phase_displacement_13=alpha_13-alpha_1_equivalent_13
    ;
printf("\n Displacement of 13th harmonic=%.2f degree
    ",phase_displacement_13)
```

Scilab code Exa 5.5 Calculating the speed of the tape

```
// 5.5
clc;
W_min=2.5*6.25*10^-6;
f=500000;
S_min=W_min*f;
printf("\nminimum tape speed=%.2f m/s",S_min)
```

Scilab code Exa 5.6 Calculating the number density of the tape

```
// 5.6
clc;
Num_per_sec=12000;
S=1.5*10^3;
Number_density=Num_per_sec/S;
printf("\nNumber density of the tape=%.0f numbers/mm
    ",Number_density)
```

Chapter 6

Cathode ray oscilloscope

Scilab code Exa 6.1 calculating the amplitude of voltage after 10 ms

```
// 6.1
clc;
Vcc=50;
t=10*10^-3;
R=500*10^3;
C=0.2*10^-6;
tc=R*C;
Vo=Vcc*[1-exp(-t/tc)];
printf("\namplitude of voltage after 10 ms=%.2f V",
   Vo)
```

Scilab code Exa 6.2 calculating the voltage across the capacitor after 50 microsecond

```
// 6.2
clc;
Vcc=4.76;
t=50*10^-6;
```

```
R=0.2*10^3;
C=0.2*10^-6;
tc=R*C;
Vo=Vcc*[exp(-t/tc)];
printf("\nvoltage across the capacitor after 50
    microsecond=%.2f V",Vo)
```

Scilab code Exa 6.3 Calculating the rise time

```
// 6.3
clc;
BW=10*10^6;
tr=0.35/BW*10^6;
printf("\nRise time=%.2f us",tr)
```

Scilab code Exa 6.4 Calculating the attenuation factor

```
// 6.4
clc;
R=(9*10^3)+(900+90+10);
Rt=100*10^3;
Attenuation=R/Rt;
Attenuation_factor=1/Attenuation;
printf("\nAttenuation factor=%.1f ",
    Attenuation_factor)
```

Scilab code Exa 6.5 Calculating the attenuation factor

```
// 6.5
clc;
```

```
R=10*10^3;
Ri=100*10^3;
Rt=100*10^3;
Rp=(Ri*R)/(Ri+R);
Attenuation=Rp/Rt;
Attenuation_factor=1/Attenuation;
printf("\nAttenuation factor=%.1f ",
    Attenuation_factor)
```

Scilab code Exa 6.6 Calculating the voltage per division voltage value

```
// 6.6
clc;
Vo=50*10^-3;
disp('For point A Attenuation_factor=400')
Attenuation_factor=400;
Vi=Attenuation_factor*Vo;
printf("\nvoltage per division value at point A=%.2f
    ",Vi)
disp('For point B Attenuation_factor=100')
Attenuation_factor=100;
Vi=Attenuation_factor*Vo;
printf("\nvoltage per division value at point B=%.2f
    ",Vi)
disp('For point C Attenuation_factor=40')
Attenuation_factor=40;
Vi=Attenuation_factor*Vo;
printf("\nvoltage per division value at point C=%.2f
    ",Vi)
disp('For point D Attenuation_factor=10')
Attenuation_factor=10;
Vi=Attenuation_factor*Vo;
printf("\nvoltage per division value at point D=%.2f
    ",Vi)
disp('For point E Attenuation_factor=4')
```

```
Attenuation_factor=4;
Vi=Attenuation_factor*Vo;
printf("\nvoltage per division value at point E=%.2f
   ",Vi)
disp('For point F Attenuation_factor=1')
Attenuation_factor=1;
Vi=Attenuation_factor*Vo;
printf("\nvoltage per division value at point F=%.2f
   ",Vi)
```

Scilab code Exa 6.7 Compare the output voltage of the voltage divider attenuator for dc and ac

```
//6.7
clc;
R2=100*10^3;
Vi=1;
R1=900*10^3;
Vo_dc=Vi*R2/(R1+R2);
k_dc=1/Vo_dc;
printf("\nAttenuationn for dc=%.1f", k_dc)
XC2=1592;
Vi=1;
XC1=3183;
Vo_ac=Vi*XC2/(XC1+XC2);
k_ac=1/Vo_ac;
printf("\nAttenuationn for ac=%.1f", k_ac)
disp('Therefore the attenuation with ac is different
    from that of dc')
```

Scilab code Exa 6.8 Calculating the maximum velocity of the beam of electrons

```
// 6.8
clc;
e=1.6*10^-19;
Ea=800;
m=9.1*10^-31;
Vox=(2*e*Ea/m)^0.5;
printf("\nmaximum velocity of the beam of electrons=
    %.2f m/s",Vox)
```

Scilab code Exa 6.9 Calculating the maximum velocity of the beam of electrons deflection sensitivity and deflection factor

```
// 6.9
clc;
e=1.6*10^-19;
Ea=2000;
m=9.1*10^-31;
Vox=(2*e*Ea/m)^0.5;
printf("\nmaximum velocity of the beam of electrons=
    %.2f m/s",Vox)
L=5;
ld=1.5*10^-2;
d=5*10^-3;
S=(L*ld/2*d*Ea);
printf("\ndeflection sensitivity=%.2f mm/V",S)
G=1/S;
printf("\nDeflection Factor=%.2f V/mm",G)
```

Scilab code Exa 6.10 Calculating the input voltage required for deflection of 3mm

```
// 6.10
clc;
```

```
Ea=2000;
L=0.3;
ld=2*10^-2;
d=5*10^-3;
D=3*10^-2;
Ed=(2*d*Ea*D)/(L*ld);
gain=100;
V_require=Ed/gain;
printf("\nInput voltage required for deflection of 3
   mm =%.1f V",V_require)
```

Scilab code Exa 6.11 Calculating the velocity of the beam and cutt off frequency

```
// 6.11
clc;
e=1.6*10^-19;
Ea=2000;
m=9.1*10^-31;
Vox=(2*e*Ea/m)^0.5;
printf("\nmaximum velocity of the beam of electrons=
   %.2f m/s",Vox)
l=50*10^-3;
fc=Vox/(4*l)*10^-6;
printf("\nCutt off frequency=%.2f MHz",fc)
```