Scilab Textbook Companion for Introduction To Mechatronics And Measurement Systems by David G. Alciatore And Michael B. Histand [1]

Created by
Sandeep Kumar C
B.tech
Electronics Engineering
Sreenidhi Institute Of Science And Technology
College Teacher
None
Cross-Checked by
None

June 10, 2019

[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: Introduction To Mechatronics And Measurement Systems

Author: David G. Alciatore And Michael B. Histand

Publisher: Tata Mcgraw Hill

Edition: 4

Year: 2012

ISBN: 978-0-07-338023-0

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 2

Electric Circuits and Components

Scilab code Exa 2.1 Resistance of a Wire

```
//Example 2.1, Page Number 16
//Resistance of a Wire
clc;

//Inputs
roh = 1.7*(10^-8)        //Resistivity of copper wire in ohm-metre.
D=(10^-3)       //Diameter of copper wire in metres.
rad=((D)/(2))       //Radius of copper wire in metres.
L=10      //Length of copper wire in metres.

//Outputs
A=(3.14)*(rad^2)       //Cross-sectional area of copper wire in metre-square
Rt=((roh*L)/(A))         //Total resistance of copper wire in ohm

//Results
```

```
mprintf("Cross-sectional area of copper wire: %.10f metre-square\n",A);
mprintf("Total wire resistance of copper wire is: %.5f ohm",Rt);

//Outputs
//Cross-sectional area of copper wire: 0.0000007850 metre-square (or) (7.8)*(10^-7) metre-square
//Total wire resistance of copper wire is: 0.21656 ohm
```

Scilab code Exa 2.2 `Resistance Color Codes`

```
//Example 2.2, Page Number 18
//Resistance Color Codes
clc;

//Inputs
//Colour bands:a = green, b =  brown, c = red, and tol = gold
//a=5                Green colour code
//b=1                  Brown colour code
c=2                //Red colour code
ab=51
tol=(0.05*((ab)*(10^c)))             //Tolerence band for given colour codes.

//Outputs
R_max=((ab)*(10^c))+tol        //Maximum resistance of given colour codes in ohms
R_min=((ab)*(10^c))-tol        //Minimum resistance of given colour codes in ohms

//Results
mprintf("Maximum resistance: %.5f ohms\n",R_max);
```

```
mprintf("Minimum resistance: %.5f ohms",R_min);

//Outputs
//Maximum resistance: 5355.00000 ohms
//Minimum resistance: 4845.00000 ohms

//Final answers in the solution of the book are
    rounded off.
```

Scilab code Exa 2.3 Kirchhoff Voltage Law

```
//Example 2.3, Page Number 23
// Kirchhoffs Voltage Law
clc;

//Inputs
Vin=10          //Input voltage in volts
Res=1000              //Resistance in ohms

//Outputs
//By Kirchhoff's Voltage Law
Vr=Vin                //Load Voltage across the load
    resistance in volts
//By ohms law
I=(Vr)/(Res)          //Load current flowing through
    load resistance in amperes

//Results
mprintf("Load current flowing through load
    resistance: %.5f ampere\n",I);

//Outputs
// Load current flowing through load resistance:
    0.01000 ampere (or) 10mA.
```

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 2.4 Circuit Analysis

```
//Example 2.4 , Page Number 28
//Circuit Analysis
clc;

//Inputs
//Given resistances in circuit:
R1=1000                //Resistance in ohms
R2=2000                //Resistance in ohms
R3=3000                //Resistance in ohms
R4=4000                //Resistance in ohms
R5=5000                //Resistance in ohms
R6=6000                //Resistance in ohms
//Given voltage sources in circuit:
V1=10                  //Voltage source in volts
V2=20                  //Voltage source in volts

//Solving given resistances into an equivalent circuit:
//Outputs
//Resistors R2 and R4 are in series ,with an equivalent resistance of(R2+R4) ,and this is in parallel with resistor R3
R234=(((R2+R4)*(R3))/((R2+R4)+R3))      //Equivalent resistance of R2,R3,R4 in ohms
R56=((R5*R6)/(R5+R6))      //Equivalent resistance of R5,R6 in ohms
//Applying KVL in left loop:
Iout=(V1/R1)                    //Current through resistance R1 in amperes
```

```
//Voltage drop across resistance R234 determined from Voltage division rule in the assumed current direction of I234 throiugh R234:
V234=((V1-V2)*((R234)/(R234+R56)))          //Voltage across resistance R234 in volts
//The desired output voltage Vout calculated from V1 and voltage drop V234 across R234 resistance is :
Vout=((V1)-(V234))                  //Desired output voltage to be calculated in volts

//Results
mprintf("Equivalent resistance of R2,R3,R4: %.5f ohms\n",R234);
mprintf("Equivalent resistance of R5,R6 : %.5f ohms\n",R56);
mprintf("Current through resistance R1: %.5f ampere\n",Iout);
mprintf("Voltage across resistance R234: %.5f volts\n",V234);
mprintf("Desired output voltage to be calculated: %.5f volts\n",Vout);

//Outputs
//Equivalent resistance of R2,R3,R4: 2000.00000 ohms
//Equivalent resistance of R5,R6 : 2727.27273 ohms
//Current through resistance R1: 0.01000 ampere
//Voltage across resistance R234: -4.23077 volts
//Desired output voltage to be calculated: 14.23077 volts
```

This code can be downloaded from the website wwww.scilab.in This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 2.5 `Input and Output Impedance`

```
//Example 2.5, Page Number 34
//Input and Output Impedance
clc;

//Inputs
//Given parameters of the circuit:
Vs=10                //Given source voltage in volts
R1=1000                 //Resistance in ohms
R2=1000                 //Resistance in ohms
Zin=1*(10^6)              //Input impedence of voltmeter in ohms
Zout=50              //Output impedence of voltage source in ohms

//Outputs
Req=((R1*R2)/(R1+R2))      //Equivalent resistance due to resistance R1 and R2 in ohms
R_eq=(((Req*Zin)/(Req+Zin))+Zout)           //Equivalent resistance due to resistance R1,input impedence of voltmeter Zin and output impedence of voltage source Zout in ohms
Vm=(((R_eq-Zout)/(R_eq))*(Vs))           //Actual voltage Vm measured by voltmeter in volts
//The measured voltage Vm equals Vs for Zin=infinty and Zout=0.

//Results
mprintf("Equivalent resistance due to resistance R1 and R2: %.5f ohms\n",Req);
mprintf("Equivalent resistance due to resistance R1, input impedence of voltmeter Zin and output impedence of voltage source Zout: %.5f ohms\n",
```

```
R_eq);
mprintf("Actual voltage Vm measured by voltmeter: %.5f volts\n",Vm);

//Outputs
//Equivalent resistance due to resistance R1 and R2: 500.00000 ohms
//Equivalent resistance due to resistance R1,input impedence of voltmeter Zin and output impedence of voltage source Zout: 549.75012 ohms
//Actual voltage Vm measured by voltmeter: 9.09050 volts
```

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 2.6 AC Signal Parameters

```
//Example 2.6, Page Number 38
//AC Signal Parameters
clc;

//Inputs
//V(t)=5.sin( t 1 )                    Input Voltage in volts

//Outputs
Vm=5.0       //Signal amplitude in volts
w=1            //Signal radian frequency in radian per second
f=((w)/(2*3.14))              //Signal frequency in hertz
phi=-1              //Phase angle of the given input signal in radian
phi_deg=((180)/(3.14)*(phi))          //Phase angle in
```

```
degrees

//Results
mprintf("Signal amplitude: %.5f volts\n",Vm);
mprintf("Signal radian frequency: %.5f rad/sec\n",w);
mprintf("Signal frequency: %.5f hertz\n",f);
mprintf("Phase angle of the given input signal: %.5f radians\n",phi);
mprintf("Phase angle: %.5f degrees\n",phi_deg);

//Outputs
//Signal amplitude: 5.00000 volts
//Signal radian frequency: 1.00000 rad/sec
//Signal frequency: 0.15924 hertz
//Phase angle of the given input signal: -1.00000 radians
//Phase angle: -57.32484 degrees
```

Scilab code Exa 2.7 AC Circuit Analysis

```
//Example 2.7, Page Number 42
//AC Circuit Analysis
clc;

//Inputs
R1=1000                              //Resistance in ohms
R2=3000                              //Resistance in ohms
C=0.2*(10^-6)                        //Capacitance in farads
L=0.5                      //Inductance in henry
//Vin=5*cos(3000*t+(pi/2))          Input Voltage in volts
```

```
//From the given input signal we can derive below parameters:
w=3000                //Input signal radian frequency in radian per second

//Outputs
Vin=(5*%i)                       //Input signal in rectangular form with signal amplitude of 5V and phase of 90 degrees in volts
Zc=(((-1)/(w*C))*(%i))           //Complex form of the capacitor impedance in ohms
Zl=(((w*L))*(%i))          //Complex form of the capacitor impedance in ohms
R2_Zl=((R2)+(Zl))              //Complex form of combined impedence of R2 and Zl in ohms
R2_Zl_Zc=(((R2_Zl)*(Zc))/(R2_Zl+Zc))            //Complex form of combined impedance of parallel combination of R2_Zl and Zc in ohms
Zeq=((R2_Zl_Zc)+(R1))                           //Complex form of equivalent impedance of the entire circuit in ohms
I1=(Vin/Zeq)                        //Current through R1 resistance in amperes
[mag0,phase0]=polar(I1)                          //Polar form of current I1 with magnitude in amperes and phase in radian

//Using current divison rule:
I=(((R2+Zl)/((R2+Zl)+Zc))*(I1))          //Current through capacitor in amperes
[mag1,phase1]=polar(I)                         //Polar form of current I with magnitude in amperes and phase in radian

//so the capacitor current leads the input reference by 159.8 or 2.789 rad, and the resulting current is
//I(t)=2.22cos(3000t+2.789)mA
```

```

//Results
disp(Vin,"Input signal in rectangular form in volts:");
disp(Zc,"Complex form of the capacitor impedance in ohms:");
disp(Zl,"Complex form of the inductance impedance in ohms:");
disp(R2_Zl,"Complex form of combined impedence of R2 and Zl in ohms:");
disp(R2_Zl_Zc,"Complex form of combined impedance of parallel combination of R2_Zl and Zc in ohms:");
disp(Zeq,"Complex form of equivalent impedance of the entire circuit in amperes:");
disp(I1,"Current through R1 resistance in amperes:");
disp(I,"Current through capacitor in amperes:");
disp(mag0,"Polar form of current I1 with magnitude in amperes:");
disp(phase0,"Polar form of current I1 with phase in radian:");
disp(mag1,"Polar form of current I with magnitude in amperes:");
disp(phase1,"Polar form of current I with phase in radian:");
printf("Resulting current in amperes:0.0022224*cos(3000*t+ 2.7886023) amperes")

//Outputs
// Complex form of the inductance impedance in ohms:1500.i
//Complex form of combined impedence of R2 and Zl in ohms:3000. + 1500.i
//Complex form of combined impedance of parallel combination of R2_Zl and Zc in ohms:923.07692 - 1615.3846i
//Complex form of equivalent impedance of the entire circuit in amperes:1923.0769 - 1615.3846i
```

```
//Current through R1 resistance in amperes:-0.0012805 + 0.0015244i
//Current through capacitor in amperes:-0.0020854 + 0.0007683i
//Polar form of current I1 with magnitude in amperes:0.0019908
//Polar form of current I1 with phase in radian:2.2694562
//Polar form of current I with magnitude in amperes:0.0022224
//Polar form of current I with phase in radian:2.7886023
//Resulting current in amperes:0.0022224*cos(3000*t+2.7886023) amperes



//Phase angles of polar forms are in radian.
```

This code can be downloaded from the website wwww.scilab.in

Chapter 3

Semiconductor Electronics

Scilab code Exa 3.1design Zener Diode Voltage Regulator Design

```
//Design Example 3.1, Page Number 84
//Zener Diode Voltage Regulator Design
clc;

//Inputs
Rl_max=240          //Maximum load resistance in ohms
Pz_max=1                    //Maximum power rating of the zener diode in watts
Vin=24          //Nominal input voltage in volts'
Vz=15                   //Zener breakdown voltage in volts

//Outputs
R_min=(((Vin-Vz)/(Pz_max+((Vz^2)/(Rl_max))))*(Vz))          //Minimum required current-limiting resistance in ohms

//Results
mprintf("Minimum required current-limiting resistance: %.5f ohms\n",R_min);
//The closest acceptable standard resistance value
```

```
   is 75 ohms

//Outputs
//Minimum required current-limiting resistance:
   69.67742 ohms
```

Scilab code Exa 3.2 Zener Regulation Performance

```
//Example 3.2, Page Number 83
//Zener Regulation Performance
clc;

//Inputs
Vin_min=20          //Minimum input voltage in volts
Vin_max=30             //Maximum input voltage in
   volts
Vz=15                  //Zener breakdown voltage in
   volts
P_dissp=1              //Maximum power rating of
   zener diode in watts
Rd=14               //Zener dynamic resistance in
   ohms
Delta_Vin=(Vin_max-Vin_min)          //Input
   voltage range in volts

//Outputs
//To limit the maximum power dissipation to less
   than 1W, the current through the diode must be
   limited to:
Iz_max=((P_dissp)/(Vz))          //Maximum zener
   current through zener diode in amperes
R_min=((Vin_max-Vz)/(Iz_max))         //Minimum load
    resistance for keeping the zener diode in
   breakdown region in ohms
//The closest acceptable standard resistance value
```

```
is 240
R_min_practical=240                              //Minimum load resistance for keeping the zener diode in breakdown region in ohms
Delta_Vout=(((Rd)/(Rd+R_min_practical))*(Delta_Vin))           //Output voltage range in volts
//When zener diode is in breakdown region Vout=Vz.
Volt_reg=((Delta_Vout)/(Vz))*100           //Voltage regulation percentage

//Results
mprintf("Maximum zener current through zener diode: %.5f ampere\n",Iz_max);
mprintf("Minimum load resistance for keeping the zener diode in breakdown region: %.5f ohms\n",R_min);
mprintf("Output voltage range: %.5f volts\n",Delta_Vout);
mprintf("Voltage regulation percentage: %.5f \n",Volt_reg);

//Outputs
//Maximum zener current through zener diode: 0.06667 ampere
//Minimum load resistance for keeping the zener diode in breakdown region: 225.00000 ohms
//Output voltage range: 0.55118 volts
////Voltage regulation percentage: 3.6745
```

Scilab code Exa 3.2design LED Switch

```
//Design Example 3.2, Page Number 98
//LED Switch
clc;

```

```
//Inputs
Vout_min=0                //Output minimum voltage in volts
Vout_max=5                //Output maximum voltage in volts
Vcc=5                   //DC supply voltage in volts
Iout_max=5*(10^-3)             //Maximum output current in amperes
Vbe=0.7                        //Base−Emitter voltage in volts
Vce=0.2                        //Collector−Emitter voltage in volts
Rc=100                  //Collector resistance in ohms
Rb=10*(10^3)                //Base resistance in ohms
//The LED requires 20  40  mA to provide a bright display.
Vf_drop=2                 //Forward bias voltage drop of LED in volts

//Outputs
//When the digital output is 0V,the transistor is in cutoff:
Ib_cutoff=(Vcc-Vbe)/Rb                              //Base current when transistor is in cutoff in amperes
//When the digital output is 5V,the transistor is in saturation:
Ic_sat=(Vcc-Vf_drop-Vce)/Rc                         //Collector current when transistor is in saturation in amperes
//The 100 ohm collector resistance limits the LED current to a value within the desired range for the LED to be bright (20  40  ) mA

//Results
mprintf("Base current when transistor is in cutoff: %.5f amperes\n",Ib_cutoff);
mprintf("Collector current when transistor is in saturation: %.5f amperes\n",Ic_sat);
```

```

//Outputs
//Base current when transistor is in cutoff: 0.00043 amperes
//Collector current when transistor is in saturation : 0.02800 amperes
```

Scilab code Exa 3.4 Guaranteeing That a Transistor Is in Saturation

```
//Example 3.4, Page Number 94
//Guaranteeing That a Transistor Is in Saturation
clc;

//Inputs
Ic_max=200*(10^-3)                    //Maximum collector current in amperes
Vce_sat=0.2                           //Collector-Emitter saturation current in volts
Beta=100              //Common emitter DC current gain
Vcc=10                        //DC supply voltage in volts
Vbe=0.7                    //Base-Emitter saturation current in volts
Rb=10*(10^3)                  //Base resistor in ohms
Rc=(10^3)                  //Collector resistor in ohms

//Outputs
Ic=(Vcc-Ic_max)/Rc                    //Collector current in amperes
//Because the DC current gain hFE(Beta) is about 100,Ib must be at least Ic/100 or 0.098mA.Because Vbe=0.7 V
Ib=(Ic/100)                           //Base current in amperes
//Base current can be related to the input voltage as:
```

```
Vin_min=(Ib*Rb)+Vbe              //Minimum input
    voltage required for saturation in volts

//Results
mprintf("Minimum input voltage required for
    saturation: %.5f volts\n",Vin_min);


//Outputs
// Minimum input voltage required for saturation:
    1.68000 volts



//Normally you would use a voltage larger than this
    (e.g., 2 to 5 times larger) to ensure that the
    transistor is fully saturated, even with
    variances in parameters.
```

This code can be downloaded from the website wwww.scilab.in

Chapter 6

Digital Circuits

Scilab code Exa 6.1 `Binary Arithmetic`

```
//Example 6.1, Page Number 200
//Binary Arithmetic
clc;

//Inputs
//Binary to decimal conversion
a=bin2dec('1001')        //Binary input for addition
b=bin2dec('0011')        //Binary input for additiom
u=bin2dec('1001')        //Binary input for multiplication
v=bin2dec('0011')           //Binary input for multiplication

//Outputs
c=a+b                   //Addition of given numbers
d=dec2bin(c)             //Decimal to binary conversion
w=u*v                  //Multiplication of given numbers
x=dec2bin(w)             //Decimal to binary conversion
```

```

//Results
disp('Addition of the given binary numbers: ')
disp(d)
disp('Multiplication of the given binary numbers: ')
disp(x)

//Outputs
//Addition of the given binary numbers: 1100

//Multiplication of the given binary numbers: 11011
```

Scilab code Exa 6.2 `Combinational Logic`

```
//Example 6.2, Page Number 204
//Combinational Logic
clc;

//Inputs
//Binary to decimal conversion
A=[0;0;0;0;1;1;1;1]        //Binary input A
B=[0;0;1;1;0;0;1;1]        //Binary input B
C=[0;1;0;1;0;1;0;1]        //Binary input C

//Outputs
Z=bitcmp(C,1)                           //Bit
   compliment of input C
D=bitand(A,B)                            //AND
   operation of inputs A,B
E=bitor(D,Z)                             //OR
   operation of D and Z(Compliment of C)
F=bitcmp((bitand(E,Z)),1)               //NAND
   operation of E and Z(Compliment of C)

//Results
```

```
disp('Input A')
disp("   A")
disp(A)
disp('Input B')
disp("   B")
disp(B)
disp('Input C')
disp("   C")
disp(C)
disp('AND operation of inputs A,B:')
disp("   D")
disp(D)
disp('OR operation of D and Z(Compliment of C):')
disp("   E")
disp(E)
disp('NAND operation of E and Z(Compliment of C):')
disp("   F")
disp(F)

//Outputs

//Input A        Input B     Input C     AND operation of inputs A,B:        OR operation of D and Z(Compliment of C):          NAND operation of E and Z(Compliment of C):

//  A                    B                C                       D                       E                       F
//  0.                   0                0                       0                       1                       0
//  0.                   0                1                       0                       0                       1
```

```
//  0.                    1                0                              0
                                                          1

    0
//  0.                    1                1                              0
                                                          0

    1
//  1.                    0                0                              0
                                                          1

    0
//  1.                    0                1                              0
                                                          0

    1
//  1.                    1                0                              1
                                                          1

    0
//  1.                    1                1                              1
                                                          1

    1
```

This code can be downloaded from the website wwww.scilab.in

Scilab code Exa 6.3 `Simplifying a Boolean Expression`

```
//Example 6.3, Page Number 207
//Simplifying a Boolean Expression
clc;

//Inputs
```

```
//Given boolean expression:
//X=(A.B.C)+(B.C)+(A'.B)

//Outputs
//Results
disp("Given boolean expression:")
disp('X=(A.B.C)+(B.C)+(A''B)')
disp("The above equation can be rewritten using the associative law and the fundamental law as Z.1=Z")
disp('X=A.(B.C)+1.(B.C)+(A''B)')
disp("Using distributive law")
disp('X=(A+1).(B.C)+(A''B)')
disp('X=(B.C)+(A''B)')
disp("Because A+1=1 and 1.(B.C )=B C")
disp("Furthermore, using the associative and distributive laws")
disp('X=B.(C+A'')')

//Outputs
// Given boolean expression:
//X=(A.B.C)+(B.C)+(A'B)
//The above equation can be rewritten using the associative law and the fundamental law as Z.1=Z
//X=A.(B.C)+1.(B.C)+(A'B)
//Using distributive law
//X=(A+1).(B.C)+(A'B)
//X=(B.C)+(A'B)
//Because A+1=1 and 1.(B.C )=B C
//Furthermore, using the associative and distributive laws
//X=B.(C+A')
```

Scilab code Exa 6.4 `Sum of Products and Product of Sums`

```
//Example 6.4, Page Number 212
//Sum of Products and Product of Sums
clc;

//Inputs
//Given inputs
A=[0;0;1;1]                 //Input A
B=[0;1;0;1]                  //Input B

//Outputs
S=bitxor(A,B)                         //Sum of A and B
C=bitand(A,B)                                    //Carry of A and B

//Results
disp("Input A")
disp(A)
disp("Input B")
disp(B)
disp("Sum of A and B:")
disp(S)
disp("Carry of A and B")
disp(C)
disp('The sum-of-products method applied to output S yields:')
disp('S=(A''+B)+(A+B'')')
disp('The product-of-sums method applied to output S yields:')
disp('S=(A+B).(A''+B'')')
disp('The sum-of-products method applied to output C yields:')
disp('C=A.B')
disp('The product-of-sums method applied to output C yields:')
disp('C=(A+B).(A+''B).(A''+B'')')

//Outputs

```

```
//Input A                    Input B              Sum of A and B:              Carry of A and B
// 0                          0                         0                        0
// 0                          1                         1                        0        .
// 1.                         0                         1                        0
// 1.                         1                         0                        1

//The sum-of-products method applied to output S yields:
//S=(A'+B)+(A+B')
//The product-of-sums method applied to output S yields:
//S=(A+B).(A'+B')
//The sum-of-products method applied to output C yields:
//C=A.B
//The product-of-sums method applied to output C yields:
//C=(A+B).(A+'B).(A'+B')


//If we use the product-of-sums result for S and the sum-of-products result for C,we obtain a circuit using the fewest number of gates
```

Chapter 8

Data Acquisition

Scilab code Exa 8.2 Aperture Time

```
//Example 8.2, Page Number 355
//Aperture Time
clc;

//Inputs
n=10                    //No of bit resolution
BW=10*(10^3)                    //Bandwidth of signal in hertz

//Outputs
N=(2^n)                 //Number of output states
delta_Ta=(2/(N*2*3.14*BW))                //Aperture time in seconds
Ts=((1/(2*BW)))                       //Minimum sampling rate in seconds

//Results
mprintf("Number of output states: %.5f\n",N);
mprintf("Aperture time: %.13f seconds\n",delta_Ta);
mprintf("Minimum sampling rate: %.6f seconds\n",Ts);

```

```
//Outputs
//Number of output states: 1024.00000
//Aperture time: 0.0000000311007 seconds (or) 31.100 nanoseconds
//Minimum sampling rate: 0.000050 seconds (or) 50 microseconds


//Even for this low-resolution converter,the required aperture time (32 nsec) is much smaller than the required sample period (50,000 nsec).
```

Chapter 9

Sensors

Scilab code Exa 9.1 Strain Gage Resistance Changes

```
//Example 9.1, Page Number 395
//Strain Gage Resistance Changes
clc;

//Inputs
R=120                          //Strain gage in ohms
F=2                   //Gage factor
strain=100*(10^-6)             //Strain experienced by
    gage

//Outputs
R_change=R*F*strain                //Resistance of the
   gage change from the unloaded state to the loaded
    state in ohms

//Results
mprintf("Resistance of the gage change from the
   unloaded state to the loaded state: %.5f ohms\n",
   R_change);


```

```
//Outputs
//Resistance of the gage change from the unloaded state to the loaded state: 0.02400 ohms
```

Scilab code Exa 9.2 Thermocouple Configuration with Nonstandard Reference

```
//Example 9.2,Page Number 413
//Thermocouple Configuration with Nonstandard Reference
clc;

//Inputs
//A standard two-junction thermocouple configuration is being used to measure the temperature in a wind tunnel.
//The reference junction is held at a constant temperature of 10 Celsius.
//We have only a thermocouple table referenced to 0 Celsius .
Junc_Temp=[0;10;20;30;40;50;60;70;80;90;100]                    //Junction temperature in celsius
V_out=[0;0.507;1.019;1.536;2.058;2.585;3.115;3.649;4.186;4.725;5.268]    //Output voltages in millivolts
V_100_to_0=5.268*(10^-3)                    //Voltage measured for a temperature of 100 Celsius relative to a reference junction at 0 Celsius in volts
V_10_to_10=0.507*(10^-3)                    //Voltage measured for a temperature of 10 Celsius relative to a reference junction at 10 Celsius in volts

//Outputs
//By law of intermediate temperature:
```

```
V_100_to_10= (V_100_to_0)-(V_10_to_10)        // Voltage measured for a temperature of 100 Celsius relative to a reference junction at 10 Celsius in volts

//Results
disp('Junction temperature in celsius list')
disp(Junc_Temp)
disp('Output voltages in millivolts list')
disp(V_out)

mprintf("Voltage measured for a temperature of 100 Celsius relative to a reference junction at 10 Celsius: %.5f volts\n",V_100_to_10);


//Outputs
//Junction temperature in celsius list    Output voltages in millivolts list

//     0                0
//    10                0.507
//    20                1.019
//    30                1.536
//    40                2.058
//    50                2.585
```

```
//    60    3.115
//    70    3.649
//    80    4.186
//    90    4.725
//   100    5.268
//Voltage measured for a temperature of 100 Celsius relative to a reference junction at 10 Celsius: 0.00476 volts (or) 4.76 millivolts
```