CHAPTER 3

DENSITY

DESIGN AND DEVELOPMENT OF
COMPUTER BASED MEASUREMENT SYSTEM

Electronic Balance  Serial Interface  PC-AT 386
The density $d$ of any liquid is given by the following general formula $^{1-2}$

$$d = \frac{m}{v}$$

where $m$ is the mass of the liquid and $v$ is the volume of the liquid. When volume is kept constant, the density is proportional to mass $m$. Measurement of mass will enable one to determine the density. In the present study a single stem pycnometer whose volume is fixed is used.

The mass is measured with a digital electronic balance which is interfaced suitably with a personal computer. On implementation and execution of suitable software the system displays density of the given sample. The block diagram of the computer based density measurement system is shown in Fig.3.1. The system consists of the following functional blocks.

1. Single stem pycnometer
2. Electronic balance
3. Personal computer
4. Serial interface
5. Temperature control bath

The hardware and software features of the measurement system are described in the following pages.
FIG 3.1: BLOCK DIAGRAM OF COMPUTER BASED DENSITY MEASUREMENT SYSTEM
SECTION 3.2
HARDWARE FEATURES

1. SINGLE STEM Pycnometer:

In the present investigation a single stem pycnometer, similar to that of Wood and Brusie\textsuperscript{4} and Scatchard et al\textsuperscript{5}, with slight modifications was chosen for the purpose. This pycnometer is shown schematically in Fig.3.2. The bulb has 18.2 cm\textsuperscript{3} capacity and 1 mm internal diameter. The pycnometer was filled with the experimental liquid using a hypodermic syringe and cannula. The corrections for buoyancy were readily applied\textsuperscript{3,4}.

2. ELECTRONIC BALANCE:

The block diagram of the electronic balance is shown in Fig.3.3. It consists of a strain gauge as a transducer to convert the mass into electrical output. The electrical output is processed by means of signal conditioner. The processed signal is converted into digital. The value of the mass is displayed on the dot matrix LCD display. User can program the balance for different options using the membrane key pad. All these tasks are controlled by a dedicated microprocessor/microcontroller. The balance has the serial RS232 connector which is useful to interact with the host personal computers.

In the present study an electronic balance (Denver Instruments Company, Model-M120) which is designed on this principle is shown in Fig.3.4. The specifications and features of the balance\textsuperscript{6} are given below.

\begin{itemize}
  \item Maximum Capacity = 120 grams
  \item Minimum Capacity = 1 mg
  \item Readability = 0.1 mg
\end{itemize}
FIG 3.2 : SINGLE STEM PYCNOMETER
FIG 3.3 : BLOCK DIAGRAM OF ELECTRONIC BALANCE
Front View

3 Sliding Glass Doors
Weighing Pan
Pan Ring
Level Indicator Vial
Dual Display
Keypad

Turn clockwise to raise or counterclockwise to lower the balance.

Rear View

Power Cord Receptacle
RS-232 I/O Receptacle

FIG 3.4: ELECTRONIC BALANCE (Denver Make)
RS-232 bi-directional interface, 5 different formats.
Automatic calibration with built-in NIST (National Institute of Standards and Technology) traceable weights.

Output Specifications:
Output can be in one of the following forms:

<table>
<thead>
<tr>
<th>Type</th>
<th>Stable</th>
<th>Unstable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1</td>
<td>$1+0000.0002$</td>
<td>$U+0000.0002$</td>
</tr>
<tr>
<td>Type 2</td>
<td>$S+0000.0003g$</td>
<td>$SD+0000.0003g$</td>
</tr>
<tr>
<td>Type 3</td>
<td>$ST+0000.0003$</td>
<td>$US+000.0003$</td>
</tr>
<tr>
<td>Type 4</td>
<td>$+0000.0003$</td>
<td>$+000.0003$</td>
</tr>
<tr>
<td>Type 5</td>
<td>$+0000.0003$ grams</td>
<td>$+0000.0003$ unstable</td>
</tr>
</tbody>
</table>

The output string is terminated with <cr> <lf>

RS-232 Connector:
The mating connector is a 9 pin subminiature D socket (male). Pins used are as follows.

<table>
<thead>
<tr>
<th>PIN#</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Case Ground</td>
</tr>
<tr>
<td>2</td>
<td>Data Input to Balance</td>
</tr>
<tr>
<td>3</td>
<td>Data Output from Balance</td>
</tr>
<tr>
<td>7</td>
<td>Signal Ground</td>
</tr>
</tbody>
</table>

The balance uses RS-232C interface, with the default settings at 8 data bits, Parity Off and two(2) stop bits.
2. PERSONAL COMPUTER:

The personal computer whose configuration is described in section 2.2 is used in the present study.

3. SERIAL INTERFACE:

The use of time-share computer terminals became more widespread. Modems were developed so that terminals could use phone lines to communicate with distant computers. Modems and other devices used to send serial data are often referred to as data communication equipment or DCE. The terminals or computers that are sending or receiving the data are referred to as data terminal equipment or DTE. In response to the need for signal and handshake standards between DTE and DCE, the Electronic Industries Association (EIA) developed EIA standard RS-232C. This standard describes the functions of 25 signal and handshake pins for serial data transfer. It also describes the voltage levels, impedance levels, rise and fall times, maximum bit rate and maximum capacitance for these signal lines. The voltage levels for all RS-232C signals are as follows.

For a logic High between -3 and -15 V under load and -25 V under no load.
For a logic Low between +3 and +15 V under load and +25 V under no load.
Voltages such as ±12 V are commonly used.

Interfacing of Electronic Balance with Personal Computer:

The interfacing of electronic balance with the personal computer through a serial interface cable designed for this purpose is shown in Fig 3.5.
FIG 3.5 INTERFACE OF ELECTRONIC BALANCE WITH PC
The pin numbers and its function of the serial I/O connector (9 pin D type Male connector) of the balance are given below.

<table>
<thead>
<tr>
<th>PIN #</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Case ground</td>
</tr>
<tr>
<td>2</td>
<td>Data input to balance (Rx)</td>
</tr>
<tr>
<td>3</td>
<td>Data output from balance (Tx)</td>
</tr>
<tr>
<td>4</td>
<td>No connection</td>
</tr>
<tr>
<td>5</td>
<td>No connection</td>
</tr>
<tr>
<td>6</td>
<td>No connection</td>
</tr>
<tr>
<td>7</td>
<td>Signal Ground (GND)</td>
</tr>
<tr>
<td>8</td>
<td>No connection</td>
</tr>
<tr>
<td>9</td>
<td>No connection</td>
</tr>
</tbody>
</table>

The pin numbers and its function of the serial connector (9 pin D type Male connector) of the personal computer are given below.

<table>
<thead>
<tr>
<th>PIN #</th>
<th>FUNCTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Carrier detect</td>
</tr>
<tr>
<td>2</td>
<td>Receive Data (Rx)</td>
</tr>
<tr>
<td>3</td>
<td>Transmit Data (Tx)</td>
</tr>
<tr>
<td>4</td>
<td>Data Terminal Ready (DTR)</td>
</tr>
<tr>
<td>5</td>
<td>Signal Ground (GND)</td>
</tr>
<tr>
<td>6</td>
<td>Data Set Ready (DSR)</td>
</tr>
<tr>
<td>7</td>
<td>Request To Send (RTS)</td>
</tr>
<tr>
<td>8</td>
<td>Clear To Send (CTS)</td>
</tr>
<tr>
<td>9</td>
<td>Ring Indicator (RI)</td>
</tr>
</tbody>
</table>

There is no handshake between the balance and the computer. The handshake signals of the serial port of the computer are interconnected by connecting RTS to CTS and DSR to DTS.
1. The serial port of the personal computer should be configured suitably (which is called initialization) by sending the control word to the control register.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port selected</td>
<td>COM2</td>
</tr>
<tr>
<td>Baud rate</td>
<td>300</td>
</tr>
<tr>
<td>Data bits</td>
<td>8</td>
</tr>
<tr>
<td>Parity</td>
<td>OFF</td>
</tr>
<tr>
<td>Stop bits</td>
<td>2</td>
</tr>
</tbody>
</table>

2. Reads the data from the balance.
3. The data is stored and displayed.

The detailed program developed in C language on these lines finds as a subroutine in the main program which is presented in section 3.3.

4. TEMPERATURE CONTROL BATH:

The temperature control bath, whose design features are described in section 2.2, is used in the present study.
SECTION 3.3
SOFTWARE FEATURES

The role of the software in the density measurement is to control the following activities.

1. To initialize the balance.
2. To make the data acquisition from the balance
3. To read the mass of the samples and to store.
4. To make deferent functional units of the system to work in a systematic and sequential manner.
5. To compute and display the density.
6. To indicate the hardware defect if any.

The necessary software package is developed on these lines for the effective functioning of the system. The high level language C is chosen for this purpose. The flowchart of the program is shown in Fig.3.6. The detailed program developed for this purpose follows.
FIG. 3.6: FLOW CHART OF THE PROGRAM
/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*   PROGRAM FOR THE MEASUREMENT OF  *
*       LIQUID DENSITY         *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

#include <stdio.h>
#include <conio.h>
#include <bios.h>

// DECLARATION OF VARIABLES
#define COM2 1 // Serial Port2
#define DATA_READY 0x100
#define TRUE 1
#define FALSE 0
#define SETTINGS 0x42 // Baud rate = 300, 8bits, no parity

FILE *cal,*mes;
char file[12],liq[20];
float den,temp,mf;
float mass,vol;
int m[12];
int DONE = FALSE;

float measure();
float calibrate();
float gmass();
void init();
void test();
int menu();
void beep1();

/* * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*       MAIN PROGRAM          *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * */

void main()
{
    int opt;
    clrscr();
    init();
    /* Initialisation */
    again:
    opt=menu();

if (opt==1)
{
    measure();
}
else if(opt==2)
{
    calibrate();
}
else if(opt==3)
{
    goto endp;
}
else
{
    goto again;
}
goto again;
endp:

// SUBROUTINE FOR MENU

int menu()
{
    int sel;
    clrscr();
    gotoxy(22,2);
    printf("COMPUTER BASED MEASUREMENT OF DENSITY");
    gotoxy(25,7);
    printf("1. MEASUREMENT OF DENSITY");
    gotoxy(25,9);
    printf("2. CALIBRATION OF DENSITOMETER");
    gotoxy(25,11);
    printf("3. QUIT");
    gotoxy(30,16);
    printf("SELECT ? = ");
    scanf("%d", &sel);
    return sel;
}

// SUBROUTINE FOR THE MEASUREMENT OF DENSITY

float measure()
{
    float mass1, vol1, tcal;
    int status;
clrscr();
printf("COMPUTER BASED MEASUREMENT OF DENSITY\n\n");
printf("Enter the Temperature(°C) = ");
scanf("%f", &temp);

if ((cal=fopen("DEN.CAL", "rt")) == NULL)
{
    fprintf(stderr, "Cannot open DEN.CAL\n");
    printf("Press any key to continue\n");
    getch();
    goto clprg;  // close programme
}

cal = fopen("DEN.CAL", "rt");
scanf(cal, "%f,%f,%f", &tcal, &massl, &voll);
if (tcal != temp)
{
    printf("\nInstrument is to be calibrate for this

temperature...\n");
    beep1();
    printf("Press any key to continue....\n");
    getch();
    goto clprg;  // close programme
}

printf("Enter filename = ");
scanf("%12s", file);
printf("Name of the Liquid = ");
scanf("%20s", liq);
printf("Place the Pycnometer in the Balance...
");
printf("Press any key to start....\n");
getch();

mass = gmass();
if (mass <= massl)
{
    printf("Error in the Measurement...
");
    printf("\nPress any key to continue...
");
    getch();
    goto clprg;
}

den = (mass - massl) / voll;
//printf("\nName of the Liquid = %s\n", liq);
printf("Liquid Density = %.5f\n", den);
//printf("Temperature = %.2f\n", temp);
//printf("Empty Pycnometer Mass = %.6f\n", massl);
//printf("Pycnometer + Liquid Mass = %.6f\n", mass);
mes = fopen(file, "w+");
fprintf(mes, "Name of the Liquid = %s\n", liq);
fprintf(mes, "Density = %.5f\n", den);
fprintf(mes,"Temperature = %.2f\n",temp);
fprintf(mes,"Molefraction = %.4f\n",mf);
fprintf(mes,"Empty Pycnometer Mass = %.6f\n",massl);
fprintf(mes,"Pycnometer + Liquid Mass =%.6f\n",mass);
fclose(mes);
printf("\nPress any key to continue...\n");
getch();
clprg:
  return 0;
}

// SUBROUTINE FOR CALIBRATION

float calibrate()
{
  float mass1,mass2,voll;
  clrscr();
  printf("CALIBRATION OF DENSITOMETER\n\n");
  printf("Enter the Temperature(°C) = ");
  scanf("%f",&temp);
  printf("Keep Empty Pycnometer in the Balance...\n");
  printf("Press any key when ready...\n");
  getch();
  mass1 = gmass();
  printf("Keep Pycnometer with standard Liquid...\n");
  printf("Press any key when ready...\n");
  getch();
  mass2 = gmass();
  if(mass2<=mass1)
    {
      printf("Error in the Measurement...
");
      printf("\nPress any key to continue...\n");
      getch();
      goto clprgl;
    }
  printf("Enter the Density of the Standard = ");
  scanf("%f",&den);
  voll = (mass2-mass1)/den;
  printf("Empty Pycnometer Mass = %.6f\n",massl);
  printf("Volume of the Pycnometer = %.6f\n",voll);
  if ((cal=fopen("DEN.CAL","rt"))==NULL)
    {
      fprintf(stderr,"Cannot open DEN.CAL\n");
      printf("Press any key to continue.....\n");
      getch();
      goto clprgl;    // close programme
    }
cal = fopen("DEN.CAL", "w+"); 
fprintf(cal,"%.2f,%.6f,%.6fM,temp,mass1,voll); 
printf("Press any key to continue.....\n"); 
getch(); 
fclose(cal); 

clprgl: 
    return 0; 
} 

// SUBROUTINE TO GET MASS FROM BALANCE 

float gmass1() 
{
    float tmass;
    printf("Enter Mass = ");
    scanf("%f", &tmass);
    return (tmass);
}

float gmass() 
{
    int i,data;
    float tmass;
    do 
    {
        data = gcom();
    } 
    while (data !=43); 
    getch(data); 
    for(i=1;i<=10;i++) 
    {
        data=gcom();
        m[i] =data & OxOf;
    }
    tmass=m[2]*100.0+m[3]*10.0+m[4]*1.0+
    m[6]*0.1+m[7]*0.01+m[8]*0.001+m[9]*0.0001;
    getch(); 
    return(tmass);
}

// SUBROUTINE FOR READING DATA FROM SERIAL PORT 

gcom() 
{
    int status,data;
    DONE = 0;
    while (!DONE) 
    {
        status = bioscom(3, 0, COM2);
if (status & DATA READY)
  if ((data = bioscom(2, 0, COM2) & 0x7F)! = 0)
    }
    return(data);
}

// SUBROUTINE FOR INITIALIZATION OF SERIAL PORT
void init()
{
  bioscom(0, SETTINGS, COM2);
}

// SUBROUTINE FOR THE SERIAL PORT TESTING
void test()
{
  return;
}

// SUBROUTINE FOR THE BEEP SOUND
void beep1()
{
  sound(500);
  delay(500);
  nosound();
}

// ******* END OF THE PROGRAM *******

The program is tested and it is found to be successful in its implementation. A typical output of the program is presented in Fig.3.7.
MEASUREMENT OF DENSITY

Enter the temperature (°C) = 25
Enter filename = BENZ@25.DAT
Name of the liquid = BENZENE
Place the Pycnometer in the balance.....
Press any key when ready.....
Keep the Pycnometer with liquid.....
Press any key to start.....
Liquid density = 0.87232
Press any key to continue.....

Fig 3.7: A typical output of density measurement
SECTION 3.4
CALIBRATION AND MEASUREMENT PROCEDURE

The pycnometer is to be cleaned with a suitable solvent if oils or other organic materials are in contact with the interior surface. The residual contamination is usually removed by filling it with acetone and allowing it to stand overnight. After filling it with the liquid under investigation, the pycnometer is to be introduced into the temperature bath and it is allowed for at least half an hour to attain the set temperature. The experimental procedure adopted for finding the density of the given sample, is presented in the flow chart which is shown in Fig.3.8.

The working and performance of the system is tested with the standard samples. The triple distilled water and analytical reagent grade chemicals benzene, carbon tetrachloride, and n-heptane are used for this purpose. These samples are used after necessary purification and distillation following the procedure cited by Weisberger\textsuperscript{10}. The purity of the sample is checked by determining the densities at 25°C.

The results of the density measurements in benzene, carbon tetrachloride, n-heptane are presented in Table 3.1. The experimental results are in good agreement with the literature values\textsuperscript{11-14}.
FIG. 3.8: FLOW CHART FOR MEASUREMENT PROCEDURE

1. START
2. SET TEMPERATURE BATH AT DESIRED POINT
3. CLEAN & FILL THE PYCNOMETER AND KEEP IN BATH
4. SWITCH ON THE MEASUREMENT SYSTEM
5. PLACE THE PYCNOMETER IN THE BALANCE AND WAIT FOR SOME TIME
6. RUN THE MEASUREMENT PROGRAM
7. NOTE THE MASS OF THE PYCNOMETER
8. COMPUTE & DISPLAY DENSITY
9. STOP
### TABLE 3.1

**DENSITY COMPARISON WITH LITERATURE DATA AT 25°C**

<table>
<thead>
<tr>
<th>Liquid</th>
<th>Density (10³ Kg m⁻³)</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>benzene</td>
<td>0.87351</td>
<td>Present Study</td>
</tr>
<tr>
<td></td>
<td>0.87351</td>
<td>Scatchard and Ticknor⁵</td>
</tr>
<tr>
<td></td>
<td>0.873524</td>
<td>Kiyohara and Benson¹²</td>
</tr>
<tr>
<td></td>
<td>0.87368</td>
<td>Diaz Pena and Delgado¹⁶</td>
</tr>
<tr>
<td>carbon tetrachloride</td>
<td>1.58432</td>
<td>Present Study</td>
</tr>
<tr>
<td></td>
<td>1.58434</td>
<td>Timmermans¹⁵</td>
</tr>
<tr>
<td></td>
<td>1.58429</td>
<td>Barker et al¹⁷</td>
</tr>
<tr>
<td></td>
<td>1.58437</td>
<td>Scatchard and Ticknor⁵</td>
</tr>
<tr>
<td></td>
<td>1.58429</td>
<td>Wood and Gray¹⁸</td>
</tr>
<tr>
<td>n-heptane</td>
<td>0.67950</td>
<td>Present Study</td>
</tr>
<tr>
<td></td>
<td>0.67951</td>
<td>Rossini¹⁹</td>
</tr>
<tr>
<td></td>
<td>0.67951</td>
<td>Weissberger¹⁰</td>
</tr>
<tr>
<td></td>
<td>0.67950</td>
<td>Letcher²⁰</td>
</tr>
<tr>
<td></td>
<td>0.67958</td>
<td>Fortier et al²¹</td>
</tr>
</tbody>
</table>
REFERENCES


***