Interfacing the Fully Automated Diagnostic Analyser To the PC

Consolidated Cybernetics Co-operative Private Ltd. (CCPL)

P-837

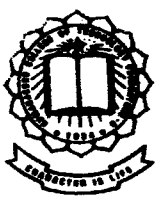
PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE OF MASTER OF SCIENCE
IN APPLIED SCIENCE - SOFTWARE ENGINEERING
OF BHARATHIAR UNIVERSITY, COIMBATORE.

Submitted by
Ms. S. Renuga Devi
Reg. no - 9937S0086

Under the Guidance of
Prof. Dr. S. Thangaswamy Ph.D
Head of Computer Science and Engineering Department
Kumaraguru College Of Technology
Coimbatore.

Mr. S. Shanmugavel
Senior Software Engineer
Consolidated Cybernetics Co-operative Private Ltd.



Department of Computer Science and Engineering
Kumaraguru College of Technology
Coimbatore - 641 006.

SEPTEMBER, 2002.

CONSOLIDATED CYBERNETICS
CO. PRIVATE LTD.

u Nagar, First Floor, Puliakulam Road, P.N. Palayam, Coimbatore-641 037, India
: 25/7, Devagiammal Street, Shenoy Nagar, Chennai - 600 030. Ph : 044-6449890

Phone : 0422-216001, 217489
Fax : 0422-217767
email : ccplindia@vsnl.com

HR/CERT/2002-2003/215
September 10, 2002

CERTIFICATE

This is to certify that the project report tilted, **"Interfacing the Fully Automated Diagnostic Analyser to the PC"**, which is being submitted by **Ms. S. Renuga Devi** in partial fulfillment of the requirements for the award of the degree of **Mater of Science in Applied Science – Software Engineering** of Bharathiar University, Coimbatore is a bonafide work carried out by her under the guidance of **Mr. S. Shanmugavel, Sr. Software Engineer**, at Consolidated Cybernetics Co. Pvt. Ltd., Coimbatore during the period May 2002 to September 2002 to our satisfaction.

For **CONSOLIDATED CYBERNETICS CO. PVT. LTD.,**

S. SHANMUGAVEL
SR. SOFTWARE ENGINEER

CERTIFICATE

Department of Computer Science and Engineering
Kumaraguru College of Technology
Coimbatore - 641 006.

This is to certify that that the project work entitled
"Interfacing the Fully Automated Diagnostic Analyser to the PC"
has been submitted by

Ms. S. Renuga Devi

in partial fulfillment of the award of the degree of
Master of Science in Applied Science - Software Engineering of
Bharathiar University, Coimbatore.
During the academic year 2002-2003

S.J. 24/9/02

Guide

S.J. 24/9/02

Head of the Department

Certified that the candidate was examined by us in the Project Work Viva Voice Examination held on 26.9.2002 and the University Register Number was **9937S0086** .

26/9

Internal Examiner

External Examiner

DECLARATION

I here by declare that this project work entitled "Interfacing the Fully Automated Diagnostic Analyser to the PC" is a record of original project work done by me under the guidance of Prof. Dr. S. Thangaswamy Head of the Computer Science and Engineering Department as internal guide and Mr. S. Shanmugavel, Senior Software Engineer as external guide, and this project work has not formed the basis for the award of any Degree / Diploma / Associate ship / Fellowship on similar titles to any other candidates of any university.

S. Renuga Devi

S. Renuga Devi

DATE: 26. 9. 2002

S.JL 7- 7 24/9/02

Internal Guide,
Prof. Dr. S. Thangaswamy,
Head of the Computer Science and Engineering Department,
Kumaraguru College Of Technology,
Coimbatore.

S. Shanmugavel

External Guide,
Mr. S. Shanmugavel ,
Senior Software Engineer,
Consolidated Cybernetics Co.Pvt.Ltd.
Coimbatore.

Dedicated to my beloved

Parents and Teachers

Acknowledgements

I would like to begin with a special note of gratitude to my sincere and heart felt thanks to our Principal **Dr. K. K. Padmanaban B.Sc. (Engg), M.Tech, Ph.D** and our HOD **Prof. Dr. S. Thangaswamy Ph.D** for giving me the needed encouragement in starting this project and carrying out successfully.

I also take an immense pleasure with gratitude especially in thanking my course coordinator **Asst. Prof. Mrs. S. Devaki B.E, M.S.** and my profound gratitude and thanks to my internal project guide **Prof. Dr. S. Thangaswamy Ph.D** for constantly encouraging me to pursue new goals and ideas and have given their tremendous guidance and suggestions throughout my project.

I would like to express my sincere thanks to **Mr. M. A. Kadersha, Head Of Software Development**, "Consolidated Cybernetics Co. Pvt. Ltd.", for providing me an opportunity to do the project in their organization.

My grateful thanks to **Mr. S. Shanmugavel, Senior Software Engineer,** for rendering his excellent guidance for the successful and timely completion of the project.

I am proud of my parents and relatives for their unending support, love, understanding and encouragement throughout the endeavor.

I also thank my friends for encouraging me to finish this project in a successful way.

Above all, I thank God Almighty, who had always showered abundant blessings on me.

Similarly a data from the processor's program of the Fully Automated Diagnostic Analyser is taken to the transmit buffer of the Fully Automated Diagnostic Analyser, then the data is sent through the port, through the RS232 null modem cable to the PC.

The **third module** deals with sending the request from the PC to the Fully Automated Diagnostic Analyser. The test serial number is got as an input on the PC. The input is converted into string. As only one character at a time can be sent through the serial cable, split the converted string into single characters and then the modified input that is the characters are transmitted one at a time through the RS232 Null modem cable, which is a standard interface, to the receive buffer of USART of the Fully Automated Diagnostic Analyser.

The **fourth module** deals with retrieving the data from the Fully-Automated Diagnostic Analyser. Here the characters that are sent by the PC are collected as a string. The real-time data which is in the text format is converted into binary format, and this binary format data is stored in a file. As the length of the results and details may vary for various test serial numbers, the binary format data is required to find the corresponding details and results of the requested test serial number.

The **fifth module** deals with receiving the data from the Fully Automated Diagnostic Analyser. After receiving the test serial number from the PC, in the Fully Automated Diagnostic Analyser it is required to check whether the given test serial number is available in the binary file, that had been created in fourth module. If it is available the corresponding result and details should be retrieved. As this result will be in binary format it is required to convert this binary result into string. The string is then splitted into characters and then it is sent to the receive buffer of the PC through the RS232 null modem serial cable.

The **sixth module** deals with error checking. Here in order to avoid transmission errors odd parity checking is done, this is applied to both, the Fully Automated Diagnostic Analyser and in PC. So that in the Fully Automated Diagnostic Analyser it is checked whether the test serial number from the PC is transmitted properly. Similarly in PC it is checked whether the details that are transmitted by the Fully Automated Diagnostic Analyser have been received by the PC properly. If there occurs a transmission error either in PC or in Fully Automated Diagnostic Analyser the transmission error message will be displayed on the screen of the PC. Then the retransmission of the results and details from the Fully Automated Diagnostic Analyser is requested by the PC. A provision is also made such that if

INDEX

1. INTRODUCTION

1.1 ABOUT AUTO - DIAGNOSTIC ANALYSER

The Diagnostic Analyser is used by the Pathologists for various bio-chemical tests such as blood, drugs etc. The Auto-Diagnostic Analyser is that which is automated and that the Pathologist have to just place the samples and the reagents in their respective partitions. The process of pipetting will be done with Hamilton Syringes with Valve blocks. The mixing will be done in the reaction rotor. The measurement and the calculations for obtaining the results will be done automatically by the appropriate scales and software. The result can be viewed in the screen of a computer attached to the automated diagnostic analyser. The disposal of the waste samples and reagents and the washing of the test tubes will be done automatically. Thus the entire process takes place automatically in the auto - diagnostic analyser.

1.1.1 Types of Auto - Diagnostic Analysers:

Generally there are two types of automated diagnostic analysers.

1. Semi-Auto Diagnostic Analyser
2. Fully Automated Diagnostic Analyser

➢ ***Semi-Auto Diagnostic Analyser :***

The semi automated diagnostic analyser **does not store the results of the various tests that is been done**, in the memory. It does the job of testing and printing out the result directly. This result could be understood only by the Pathologist as it will not be much detailed, it just contains chemicals and their respective quantities.

➢ ***Fully-Automated Diagnostic Analyser :***

The Fully Automated Diagnostic Analyser **stores the results of various tests that are done,** in the processor program. It does the job of testing according to the present procedures and printing out the result. This result could be understood only by the pathologist as it will not be much detailed, it just contains chemical names and their respective quantities.

1.1.2 Fully Automated Diagnostic Analyser Functions:

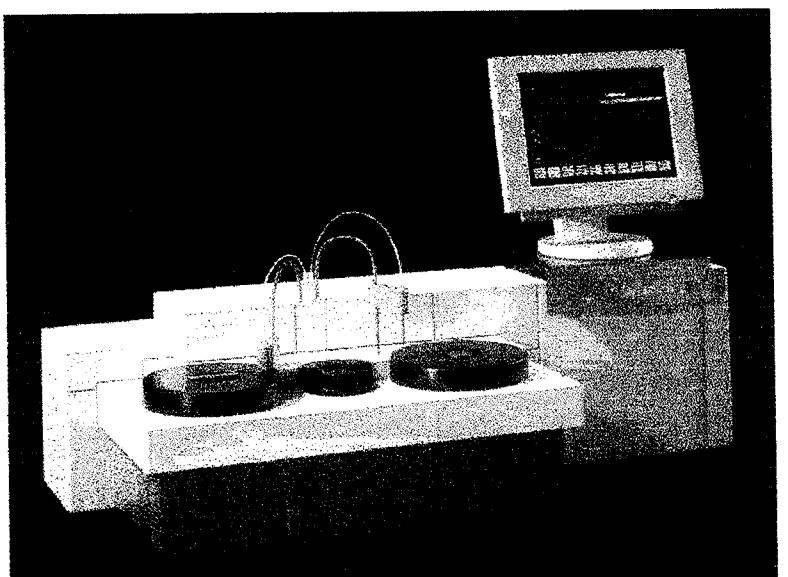
The project deals with Fully Automated Diagnostic Analyser and its functions are as follows.

The Fully Automated Diagnostic Analyser consists of three partitions.

- For placing the sample(in the right)
- Reaction Rotor(in the middle)
- For placing the reagents(in the left)

Fig 1.

Fully Automated Diagnostic Analyser



All the three partitions consist of small holes for placing the test tubes or bottles and each and every hole has a unique number.

The sample for testing is taken in a test tube and is placed inside the sample partition. Then the required reagents for carrying out the test are placed in the reagent partition. For pipetting Hamilton syringe and valve block is used.

Fig 2.

Sample Partition

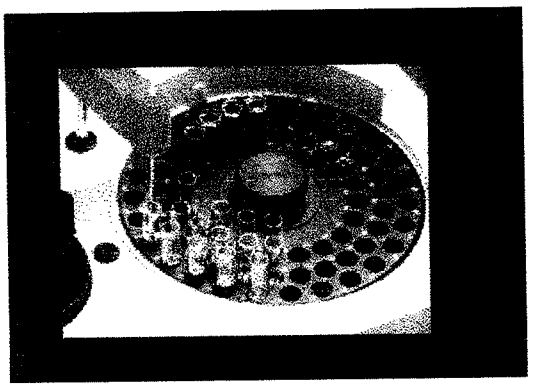


Fig 3.

Reaction Rotor

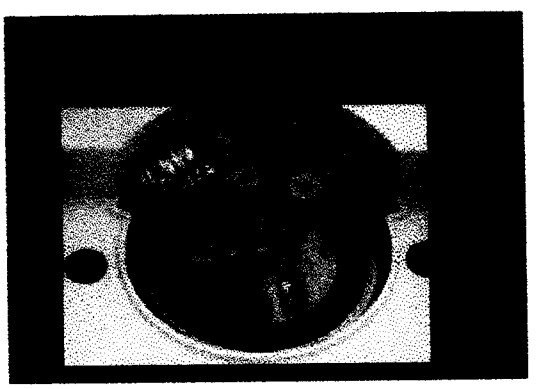
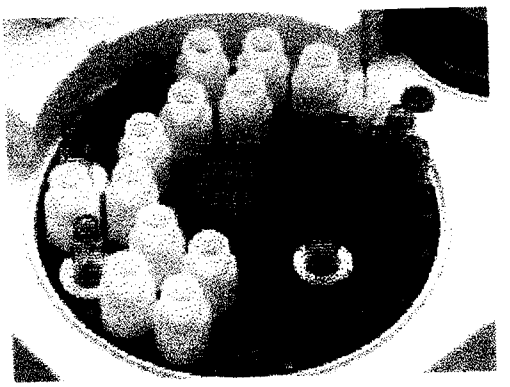


Fig 4.

Reagent Partition



The pathologist sends the command for the required test by checking the check boxes in the user screen of the analyser. The Hamilton syringe and valve block automatically moves to the reagent partition and it absorbs a small fixed amount of reagent and then it moves and it pipettes out in reaction rotor(in middle),and then it moves to the sample partition absorbs the required amount of sample and then moves to the reaction rotor and pipettes out into the reagent that was pipetted out earlier. Valve block prevents the flow of the liquid in the Hamilton Syringe, and when the valve block is released liquid flows out. The controlling of valve block is automatically taken care by the analyser itself. The mixing is done in the reaction rotor. As the test is being carried out there will be graphical representation of the sample partition that indicates all the holes that are used to place the test tubes and their respective numbers on the screen. The sample with the reagent for which the testing is going on is indicated by circle which serves as a marker. On the side of this graphical representation the test tube hole number will be printed and the corresponding chemicals and their quantities will be printed .

The details regarding the test tube hole number which is called as the sample number, the details of the person who has undergone the test such as person's name which is called as the sample name, date of birth of the person ,sex, number of tests undergone by the person, and the details of the result such as the type of test for

respective quantities and their respective units and also a flag that is produced regarding the working of the equipment gets stored in the processor program.

1.1.3 Fully-Automated Diagnostic Analyser Specifications:

The Fully Automated Diagnostic Analyser supports bi-directional host connection. It contains a 25 pin female serial port that supports asynchronous communication. It consists of Hamilton syringes with valve block for pipetting and consists of specific software for calculating the quantity of each and every chemical present in a sample. Consists of a good memory and the results are stored.

1.2 ORGANIZATION PROFILE:

Consolidated Cybernetics is a leading Software Development Company situated in Coimbatore, Bangalore and Chennai. Established in 1993 as D I G I T A L (DEC, USA).Consolidated Cybernetics offers high quality, reliable software regarding medicine field for hospitals and laboratories under the name "Medisoft". **Medisoft** is specially designed to cater the needs of the health care industry. High Quality Standards have been maintained to meet the requirements of world leaders .

The domains of Consolidated Cybernetics are

- **Health Care Industries**
- **Discrete Manufacturing Industries**

Partners of Consolidated Cybernetics are

- **Tata Consultancy Services**
- **AL-ROMAN** Technologies **DUBAI**

The CEO, Mr. P R. Rangaswami, Managing Director and Chief Executive Officer is the visionary and force behind Cybernetics. He takes care of long term planning, strategies and partnership. Has expertise in Finance, Costing, Software, Project Consultancy & Management, Has worked for large corporations like Neyveli Lignite Corporation, Bharat Heavy Electricals Ltd, & Lakshmi Machine Works. Ltd.

Mr. M A Kadersha - Head – Software development has a strong background in software engineering and has over 12 years of professional experience in Design, Development and Implementation of software projects. Has developed and implemented projects for large international clients.

R Balaji - Head – Business development

Vision

To go beyond the limits of possible to become a

contributing and respectable member of the Society

Quality Policy

'WE SHALL DESIGN AND
DEVELOP QUALITY PRODUCTS,
PROVIDE SOFTWARE SOLUTIONS,
ALWAYS MEETING THE
CUSTOMER REQUIREMENTS AND
SHALL DELIVER ON TIME.'

1.3 ABOUT THE PROJECT:

The project is interfacing the Fully Automated Diagnostic Analyser to a PC. It deals with identifying the COM port address of the PC and the analyser, communicating through the COM port, transmitting the data through the RS232 standard interface cable, retrieving the requested data from the processor program by making use of USART of the Fully Automated Diagnostic Analyser. The error checking is also done using parity checking and if there are any transmission errors, such as data being lost during the transmission through the RS232 cable, then retransmission of data from the analyser is requested by the PC, else the data is stored in PC. This concept can be implemented for embedded systems with serial ports for interfacing, where it helps in sharing of data, saving the memory of the embedded systems.

1.3.1 Existing system and its limitations:

The Fully Automated Diagnostic Analyser contains valuable software that controls the entire equipment, the equipment is shielded with security software such that it does not allow any of the software to be loaded into the equipment. Anomaly it accepts the software that can deal with interfacing alone. No other software is accepted by the operating system of the Fully Automated Diagnostic Analyser in order to avoid the foulness in the testing, calculating results and also in maintaining the quality of the equipment.

The result details that are stored in the processor's program in the Fully Automated Diagnostic Analyser are not enough as for giving as a report, that is used by doctors and other persons in the medicine field. So that it requires a software in order to capture the details for a particular patient using the test serial number from the Fully Automated Diagnostic Analyser, which in turn is used by another software "Laboratory Management", in order to present it in the form of a clear report, to investigate and also for billing purposes. There is an inconvenience among the users such as doctors or receptionists of the laboratory such that they have to go to the room where the Fully Automated Diagnostic Analyser is placed to view the result for each and every person

in order to enter the results in "Laboratory Management " software. This is a discomfort to users.

The result that is been received from the Fully Automated Diagnostic Analyser will be in the following format.

N;385;Joseph;;M;6;GLU;81;mg/dl;a;UREA;28;mg/dl;;ALB;3.7;g/dl;N;TBIL;0.7;mg/dl ;a;DBIL;0.4;mg/dl;a;TPR;6.9;g/dl;;

The above format of the result is to be presented clearly as follows.

Sample number : 385
Sample name : Joseph
Date of Birth : NILL
Sex : M
Number of tests : 6

Type : N
Test name : GLU
Result : 81
Unit : mg/dl
Flag : a

Type : N
Test name : UREA
Result : 28
Unit : mg/dl
Flag : NILL

Type : N
Test name : ALB
Result : 3.7
Unit : g/dl

Type : N
Test name : TBIL
Result : 0.7
Unit : mg/dl
Flag : a

Type : N
Test name : DBIL
Result : 0.4
Unit : mg/dl
Flag : a

Type : N
Test name : TPR
Result : 6.9
Unit : g/dl
Flag : NILL

Here Type indicates the type of the test undergone for instance whether it is normal, calibrator, pediatric tests. Then Test name indicates the name of the chemical that is been tested in the sample. Result indicates the quantity of the chemical in the sample that is been tested. Unit indicates the standard of measurement for the chemical that is been tested in the sample. Flag deals with working of the Fully Automated Diagnostic Analyser.

In the above description we can see that Date of Birth is NILL, this is because the patients date of birth does not exist in the Fully Automated Diagnostic Analyser, as it might not be entered by the Pathologist in the Fully Automated Diagnostic Analyser.

It can also be seen that in some cases the flag is NILL, this is because the Fully Automated Diagnostic Analyser by default assigns some flags in the results, these flags deal with working of the Fully Automated Diagnostic Analyser.

1.3.2 Proposed system and its advantages:

To overcome the problem associated with the existing system, interfacing is considered to be very essential. Keeping these objectives in mind the "Interfacing the Fully Automated Diagnostic Analyser", software is developed.

The " Interfacing the Fully Automated Diagnostic Analyser" software is expected to revolutionize the present day Bio - Medical Instrumentation field with its state of the facilities moving hand - in - hand with today's fast paced technologies. Using this software either the doctor in the hospital or a receptionist in a laboratory can capture the results of a particular person, by sitting in front of the PC that is interfaced to the Fully Automated Diagnostic Analyser, without entering into the room where the equipment, Fully Automated Diagnostic Analyser is placed. The PC is any general purpose compatible and that in turn can be used for various other purposes also. This serves to be very flexible for doctors and other users such as receptionists, in saving the time otherwise for which they have to move to the equipment's room in order to see the result. The results if required can be stored in the PC and it is confidential and accurate as per it is stored in the Fully Automated Diagnostic Analyser.

The software with some modifications can be applied to all the equipment that contains a serial port. Thus the software can be widely used in Bio - Medical Instrumentation and Electrical and Electronics Instrumentation and also in Electrical and Electronics Communication fields.

The "Interfacing the Fully Automated Diagnostic Analyser", comprises of six modules. They are

- Identifying the Port addresses of PC and Fully Automated Diagnostic Analyser
- Establishing the communication between the Fully Automated Diagnostic Analyser and the Personal Computer(PC)
- Sending the request from the PC to the Fully Automated Diagnostic Analyser
- Retrieving the data from the Fully Automated Diagnostic Analyser
- Receiving the data from the Fully Automated Diagnostic Analyser
- Error Checking.

1. Identifying the Port addresses of PC and Fully Automated Diagnostic Analyser :

Usually there are three communication ports in a PC, such that two serial communications port COM1, COM2 and one parallel communication port. The serial communication port of the PC is 9 pin male port that supports asynchronous communication and bi-directional host connection which is considered to be the Data Terminal Equipment (DTE). The serial communication port of the Fully Automated Diagnostic Analyser is 25 pin female port that supports asynchronous communication and bi-directional host connection which is considered to be the Data Communication Equipment(DCE). The serial communication port address are identified for both the PC and Fully Automated Diagnostic Analyser. Then one of the serial port of the PC is selected. A null-modem RS232 standard interface cable is used to connect the serial port of the PC and the serial port of the Fully-Automated Diagnostic Analyser.

2. Establishing the communication between the Fully Automated Diagnostic Analyser and the Personal Computer(PC) :

The second module deals with establishing the communication between the Fully-Automated Diagnostic Analyser and the PC. In this module the baud rate, that is a measure of transmission speed, equivalent to about one bit per second, is set to 19200BPS. It is required that the baud rate should be the same in both the PC and the Fully Automated Diagnostic Analyser, in order to facilitate a perfect serial communication between the COM ports.

In order to transmit the data from the PC the data should be taken to the transmit buffer of USART of the PC. From the transmit buffer the data should be transmitted through the port, through the RS232 null modem cable to the receive buffer of USART of the Fully Automated Diagnostic Analyser. Then the transmitted data from the PC to the Fully Automated Diagnostic Analyser that is now in the receive buffer of the analyser can be viewed.

Similarly a data from the processor's program of the Fully Automated Diagnostic Analyser is taken to the transmit buffer of the Fully Automated Diagnostic Analyser, then the data is sent through the port, through the RS232 null modem cable to the PC.

Thus communication is established between the Fully Automated Diagnostic

3. Sending the request from the PC to the Fully Automated Diagnostic Analyser:

The third module deals with sending the request from the PC to the Fully-Automated Diagnostic Analyser. The test serial number is got as an input on the PC. The input is converted into string. As only one character at a time can be sent through the serial cable, split the converted string into single characters and then the modified input that is the characters are transmitted one at a time through the RS232 serial cable, which is a standard interface, to the receive buffer of USART of the Fully Automated Diagnostic Analyser.

4. Retrieving the data from the Fully Automated Diagnostic Analyser:

The fourth module deals with Retrieving the data from the processor program of the FullyAutomated Diagnostic Analyser. Here the characters that are sent by the PC are collected as a string. The realtime data which is in the text format is converted into binary format, and this binary format data is stored in a file. As the length of the results and details may vary for various test serial numbers, the binary format data is required to find the corresponding details and results of the requested test serial number.

5. Receiving the data from the Fully Automated Diagnostic Analyser:

The fifth module deals with receiving the data from the Fully Automated Diagnostic Analyser. After receiving the test serial number from the PC, in the Fully Automated Diagnostic Analyser it is required to check whether the given test serial number is available in the binary file, that had been created in fourth module. If it is available the corresponding result and details should be retrieved. As this result will be in binary format it is required to convert this binary result into string. The string is then splitted into characters and then it is sent to the receive buffer of the PC through the RS232 null modem serial cable.

6. Error Checking:

The sixth module deals with Error checking, here in order to avoid transmission errors, odd parity checking is done, this is applied to both, the Fully Automated Diagnostic [illegible] the test serial number from the PC is transmitted properly, and also in PC so

hat the details that are transmitted by the Fully Automated Diagnostic Analyser have been eceived by the PC properly. If there occurs a transmission error either in PC or in Fully Automated Diagnostic Analyser the transmission error message will be displayed on the screen of the PC

and the retransmission of the results and details from the Fully Automated Diagnostic Analyser is requested by the PC. A provision is also made such that if any of the test serial number is not found in the Fully Automated Diagnostic Analyser then a message will be displayed on the PC.

2. SYSTEM REQUIREMENTS

2.1 Purpose:

The project is concerned with a product Fully Automated Diagnostic Analyser. The name of the Fully Automated Diagnostic Analyser is VITALAB SELECTRA. Here VITALAB is the name of the company and SELECTRA represents the model of the Diagnostic Analyser. The project is to be done in "Consolidated Cybernetics Co. Pvt. Ltd", which is a software development company, that deals mainly in developing software for medicine field. The Fully Automated Diagnostic Analyser is available in HITECH LAB and SHEELA HOSPITAL. This project is to be done for HITECH LAB.

2.2 Problem Statement:

The Fully Automated Diagnostic Analyser does various tests such as blood, urine, drugs etc. The results of the tests are stored in the processor program in the Fully Automated Diagnostic Analyser. The Fully Automated Diagnostic Analyser should be interfaced with a PC in order to capture the results from the Fully Automated Diagnostic Analyser as per the request given in the PC by the user. The result brought to the PC serves as a reference, which can be used for investigations, billing purposes and also used to present a clear report which will be given as a hardcopy to the patient. This interfacing plays an important role in which it provides a great convenience for the users as they can capture the results from the Fully Automated Diagnostic Analyser through the PC by sitting in their seats without going to the room where the Fully Automated Diagnostic Analyser is placed.

2.3 Scope:

The " Interfacing the Fully Automated Diagnostic Analyser to the PC" software is expected to revolutionize the present day Bio - Medical Instrumentation field with its state of the facilities moving hand - in - hand with today's fast paced technologies. Using this

results of a particular person, by sitting in front of the PC that is interfaced to the Fully Automated Diagnostic Analyser, without entering into the room where the equipment, Fully Automated Diagnostic Analyser is placed. The PC is any general purpose compatible and that in turn can be used for various other purposes also. This serves to be very flexible for doctors and other users such as receptionists, in saving the time otherwise for which they have to move to the equipment's room in order to see the result. The results if required can be stored in the PC for using with other software and it is confidential and accurate as per it is stored in the Fully Automated Diagnostic Analyser. The result brought to the PC serves as a reference, which can be used for investigations, billing purposes and also used to present a clear report which will be given as a hardcopy to the patient.

The software with some modifications can be applied to all the equipment that contains a serial port. Thus the software can be widely used in Bio - Medical Instrumentation, Electrical and Electronic Instrumentation and also in Electrical and Electronic Communication fields.

2.4 Overview:

The project deals with Fully Automated Diagnostic Analyser and its functions are as follows.

The Fully Automated Diagnostic Analyser consists of three partitions.

- For placing the sample(in the right)
- Reaction Rotor(in the middle)
- For placing the reagents(in the left)

All the three partitions consist of small holes for placing the test tubes or bottles and each and every hole has a unique number.

The sample for testing is taken in a test tube and is placed inside the sample partition. Then the required reagents for carrying out the test are placed in the reagent partition. For pipetting Hamilton syringe and valve block is used.

The Pathologist sends the command for the required test by checking the check boxes in the user screen of the analyser. The Hamilton syringe and valve block automatically moves to the reagent partition and it absorbs a small amount of reagent and then it moves and it pipettes out in reaction rotor(in middle),and then it moves to the sample partition absorbs the required amount of sample and then moves to the reaction rotor and pipettes out into the reagent that was pipetted out earlier. Valve block prevents the flow of the liquid in the Hamilton Syringe, and when the valve block is released liquid flows out. The controlling of valve block is automatically taken care by the analyser itself. The mixing is done in the reaction rotor. As the test is being carried out there will be graphical representation of the sample partition that indicates all the holes that are used to place the test tubes and their respective numbers on the screen. The sample with the reagent for which the testing is going on is indicated by circle which serves as a marker. On the side of this graphical representation the test tube hole number will be printed and the corresponding chemicals and their quantities will be printed .

The details regarding the test tube hole number which is called as the sample number, the details of the person who has undergone the test such as person's name which is called as the sample name, date of birth of the person, sex, number of tests undergone by the person, and the details of the result such as the type of test for instance whether it is normal, calibrator, pediatric and the name of the chemicals, their respective quantities and their respective units and also a flag that is produced regarding the working of the equipment gets stored in the processor program. These details and results are required for billing purposes, investigations and for also preparing reports.

As the Fully Automated Diagnostic Analyser contains valuable software that controls the entire equipment, the equipment is shielded with security software such that it does not allow any of the software to be loaded into the equipment. Anomaly it accepts the software that can deal with interfacing alone. No other software is accepted by the operating system of the Fully Automated Diagnostic Analyser in order to avoid the foulness in the testing, calculating results and also in maintaining the quality of the equipment.

In this case it is required to transfer the data to a separate Personal Computer (PC). So that some other software in the PC can make use of this data and it helps out in giving out a detailed report and which serves also as a reference. For this interfacing the Fully Automated Diagnostic Analyser to the PC is required and this is done through RS232 interface.

When the operator types the test serial number in the PC, this test serial number is taken to the transmit buffer of USART of the PC, then this test serial number from the transmit buffer of USART of the PC is transmitted through the RS232 interface cable, through the COM port towards the receive buffer of USART of the Fully Automated Diagnostic Analyser. The results and details regarding the test is stored in text format in the Fully Automated Diagnostic Analyser, as the number of tests undergone varies from one person to another, it is required to convert the text format results into binary format, so that all the results can be captured clearly.

The test serial number now in the receive buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the processor program of the Fully Automated Diagnostic Analyser.

The test serial number which is the operator's request, from the PC, which is now in the receive buffer of USART of the Fully Automated Diagnostic Analyser should be seen whether it matches with the test serial number stored in the binary format file of the analyser. Once it gets matched the details corresponding to that test serial number is reterived and it is then converted into text format and then it is sent to the transmit buffer of USART of the Fully Automated Diagnostic Analyser. Then the data from the transmit buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the receive buffer of USART of the PC through the RS232 null modem cable, through the COM port.

The PC should accept the details store those details in a buffer and error checking is done, if there is any error then request is passed to the Fully Automated Diagnostic Analyser to resend else the required details are placed in PC in the text format for further uses.

2.5 GENERAL DESCRIPTION

2.5.1 Product Perspective:

The Fully Automated Diagnostic Analyser VITALAB SELECTRA is used by the Pathologists for various bio-chemical tests such as blood, drugs etc. Here

VITALAB is the name of the company and SELECTRA represents the model of the Diagnostic analyser.

Generally there are two types of automated diagnostic analysers.

1. Semi-Auto Diagnostic Analyser
2. Fully Automated Diagnostic Analyser

- ***Semi-Auto Diagnostic Analyser:***

The semi automated diagnostic analyser **does not store the results of the various tests that is been done**, in the memory. It does the job of testing and printing out the result directly. This result could be understood only by the Pathologist as it will not be much detailed, it just contains chemicals and their respective quantities.

- ***Fully-Automated Diagnostic Analyser:***

The Fully Automated Diagnostic Analyser **stores the results of various tests that is been done,** in the processor program. It does the job of testing and printing out the result. This result could be understood only by the Pathologist as it will not be much detailed, it just contains chemicals and their respective quantities.

In order to interface the Fully Automated Diagnostic Analyser to the PC the following technical components and terms are to be known.

- RS232 :

RS232 is a very common physical interface. It is a US engineering industry standard which has been in use since the 1960s .It defines a 25 pin D plug connector with unbalanced transmission circuits on which logic one is represented by -3 to -25 volts, logic zero by +3 to +25 volts.

RS232 is defined for a full duplex connection with separate circuits for transmitting and receiving.

USART:

Universal Synchronous/Asynchronous Receiver-Transmitter, the USART is a serial communication interface, which is capable of being programmed for either synchronous or asynchronous operation.

Handshaking:

Handshaking refers to the internal communications protocol by which data is transferred from the hardware port to the receive buffer. When a character of data arrives at the serial port, the communications device has to move it into the receive buffer so that the program can read it. A handshaking protocol ensures that data is not lost due to buffer overrun, where data arrives at the port too quickly for the communications device to move the data into the receive buffer.

Software Flow Control:

Instead of using voltage levels on pins, the device may transmit a special character X OFF down its transmit line (pin 2) to tell stop sending characters. When it wants resume, it sends an X ON character.

Bi-Directional Communication:

The serial port on the PC is a full-duplex device meaning that it can send and receive data at the same time. In order to be able to do this, it uses separate lines for transmitting and receiving data. The Fully Automated Diagnostic Analyser also supports bi-directional communication.

Serial Interface:

An interface that can be used for serial communication, in which only one bit is transmitted at a time.

➢ Serial Communication:

All IBM PC and compatible computers are typically equipped with two serial ports and one parallel port. Although these two types of ports are used for communicating with external devices, they work in different ways.

A parallel port sends and receives data eight bits at a time over 8 separate wires. This allows data to be transferred very quickly; however, the cable required is more bulky because of the number of individual wires it must contain. Parallel ports are typically used to connect a PC to a printer and are rarely used for much else. A serial port sends and receives data one bit at a time over one wire. While it takes eight times as long to transfer each byte of data this way, only a few wires are required.

➢ Asynchronous Serial communication:

The serial ports on IBM-style PCs are asynchronous devices and therefore only support asynchronous serial communications.

Asynchronous means "no synchronization", and thus does not require sending and receiving idle characters. However, the beginning and end of each byte of data must be identified by start and stop bits. The start bit indicates when the data byte is about to begin and the stop bit signals when it ends. The requirement to send these additional two bits cause asynchronous communications to be slightly slower than synchronous however it has the advantage that the processor does not have to deal with the additional idle characters.

An asynchronous line that is idle is identified with a value of 1, (also called a mark state). By using this value to indicate that no data is currently being sent, the devices are able to distinguish between an idle state and a disconnected line. When a character is about to be transmitted, a start bit is sent. A start bit has a value of 0, (also called a space state). Thus, when the line switches from a value of 1 to a value of 0, the receiver is alerted that a data character is about to come down the line.

- Duplex Communication:

Another important characteristic of digital communication is the extent to which simultaneous two-way transfers of data can be achieved.

In a simple connection, the hardware configuration is such that only one-way communication is possible (for example, from a computer to a printer that cannot send status signals back to the computer). In a half-duplex connection, two-way transfer of data is possible, but only in one direction at a time. That is, the two parties to the connection take turns transmitting and receiving data. In a full-duplex connection, both parties can send and receive data simultaneously. The Serial Driver supports full-duplex operation.

2.5.2 Product Functions:

The Fully Automated Diagnostic Analyser consists of three partitions.

- For placing the sample
- Reaction Rotor(in the middle)
- For placing the reagents

All the three partitions consist of small holes for placing the test tubes or bottles and each and every hole has a unique number.

The sample for testing is taken in a test tube and is placed inside the sample partition. Then the required reagents for carrying out the test are placed in the reagent partition. For pipetting Hamilton syringe and valve block is used.

The Pathologist sends the command for the required test by checking the check boxes in the user screen of the analyser. The Hamilton syringe and valve block automatically moves to the reagent partition and it absorbs a small amount of reagent and then it moves and it pipettes out in reaction rotor(in middle),and then it moves to the sample partition absorbs the required amount of sample and then moves to the reaction rotor and pipettes out into the reagent that was pipetted out earlier. Valve block prevents the flow of the liquid in the Hamilton Syringe, and when the valve block is released liquid flows out. The controlling of valve block is automatically taken care by the analyser itself. The mixing is done in the

sample partition that indicates all the holes that are used to place the test tubes and their respective numbers on the screen. The sample with the reagent for which the testing is going on is indicated by circle which serves as a marker. On the side of this graphical representation the test tube hole number will be printed and the corresponding chemicals and their quantities will be printed .

The details regarding the test tube hole number which is called as the sample number, the details of the person who has undergone the test such as person's name which is called as the sample name, date of birth of the person ,sex, number of tests undergone by the person, and the details of the result such as the type of test for instance whether it is normal, calibrator, pediatric and the name of the chemicals, their respective quantities and their respective units and also a flag that is produced regarding the working of the equipment gets stored in the processor program. These details and results are required for billing purposes, investigations and for also preparing reports.

As the Fully Automated Diagnostic Analyser contains valuable software that controls the entire equipment, the equipment is shielded with security software such that it does not allow any of the software to be loaded into the equipment. Anomaly it accepts the software that can deal with interfacing alone. No other software is accepted by the operating system of the Fully Automated Diagnostic Analyser in order to avoid the foulness in the testing, calculating results and also in maintaining the quality of the equipment.

In this case it is required to transfer the data to a separate Personal Computer (PC). So that some other software in the PC can make use of this data and it helps out in giving out a detailed report and which serves also as a reference. For this interfacing the Fully Automated Diagnostic Analyser to the PC is required and this is done through RS232 interface.

When the operator types the test serial number in the PC, this test serial number is taken to the transmit buffer of USART of the PC, then this test serial number from the transmit buffer of USART of the PC is transmitted through the RS232 interface cable, through the COM port towards the receive buffer of USART of the Fully Automated Diagnostic Analyser. The results and details regarding the test is stored in text format in the Fully Automated Diagnostic Analyser, as the number of tests undergone varies from one person to another, it is required to convert the text format results into binary format, so that all the results can be captured clearly.

The test serial number now in the receive buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the processor program of the Fully Automated Diagnostic Analyser.

The test serial number which is the operator's request, from the PC, which is now in the receive buffer of USART of the Fully Automated Diagnostic Analyser should be seen whether it matches with the test serial number stored in the binary format file of the analyser. Once it gets matched the details corresponding to that test serial number is reterived and it is then converted into text format and then it is sent to the transmit buffer of USART of the Fully Automated Diagnostic Analyser. Then the data from the transmit buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the receive buffer of USART of the PC through the RS232 null modem cable, through the COM port.

The PC should accept the details store those details in a buffer and error checking is done, if there is any error then request is passed to the Fully Automated Diagnostic Analyser to resend else the required details are placed in PC in the text format for further uses.

2.5.3 User Characteristics:

This product can be used only by the person, who is been given the authorization to access it, by the manager of the HITECH LAB. The person should have the knowledge of operating the computer. The software is designed as a user friendly interface product. The user can interact with the system by using the keyboard for giving the input. The result will be displayed for each test serial number.

2.5.4 General Constraints:

The distance between the Fully Automated Analyser and the PC should be within 50ft.

2.6 SPECIFIC REQUIREMENTS

2.6.1 Functional Requirements:

The various functions that are to be done in order to interface the DCE, the Fully Automated Diagnostic Analyser and the DTE, the PC, are

- Identifying the COM port addresses of the PC and the Fully Automated Diagnostic Analyser
- Establishing the communication between the PC and the Fully Automated Diagnostic Analyser through the serial communication port
- Transmitting the data by making use of the USART through the RS232 standard interface cable
- Retrieving the requested data from the processor program of the Fully Automated Diagnostic Analyser
- Receiving the data that is sent from the Fully Automated Diagnostic Analyser in the PC.
- The error checking is also done using odd parity checking and if there are any transmission errors, such as data being lost during the transmission through the RS232 cable, then retransmission of data from the Fully Automated Diagnostic Analyser is requested by the PC, else the data is stored in PC.

2.6.2 Performance Requirements:

Security:

Username and password is given to login the software, such that only the person, who is given the authorization to access it by the manager of the HITECH LAB, can use the product.

Availability:

The data retrieved using this software will be available in the PC and it can be used at any time without turning on the Fully Automated Diagnostic Analyser.

Capacity:

One record at a time is retrieved through the RS232 interface. The baud rate is set to 19200BPS.

Response time:

As the request is sent from the PC the response should be within 2 minutes, if there is no response then an error message will be generated and the PC sends a request to, resend the data, to the Fully Automated Diagnostic Analyser.

2.6.3 Design Constraints

Hardware Limitations:

Processor Pentium III 550 MHz

Floppy Disk Drive 1.44 MB

Hard Disk 20GB

System RAM 64MB

Serial Ports : 2

- one for analyser
- one for host

Printer Port: 1

RS232 Cable

External Interface Requirements:

RS232 null modem cable is used as the external interface between the Fully Automated Diagnostic Analyser (DCE) and PC (DTE). Here DTE is of 9 pin and DCE is of 25 pin.

User Interfaces:

The user can interact through keyboard and mouse in order to get

Hardware Interfaces and Software Interfaces with other systems:

Hardware Interface : RS232 standard interface null modem cable

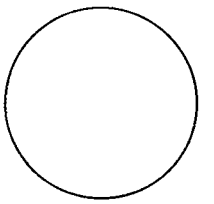
Software Interface : 'C'

2.6.4 High-Level Data Flow Diagram:

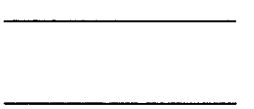
Data Flow Diagram (DFD) is a graphical technique that depicts information flow and the transforms that are applied as data move from input to output. DFD may be used to represent a system or software at any level of abstraction flow and functional details.

Graphical Notations used in DFD:

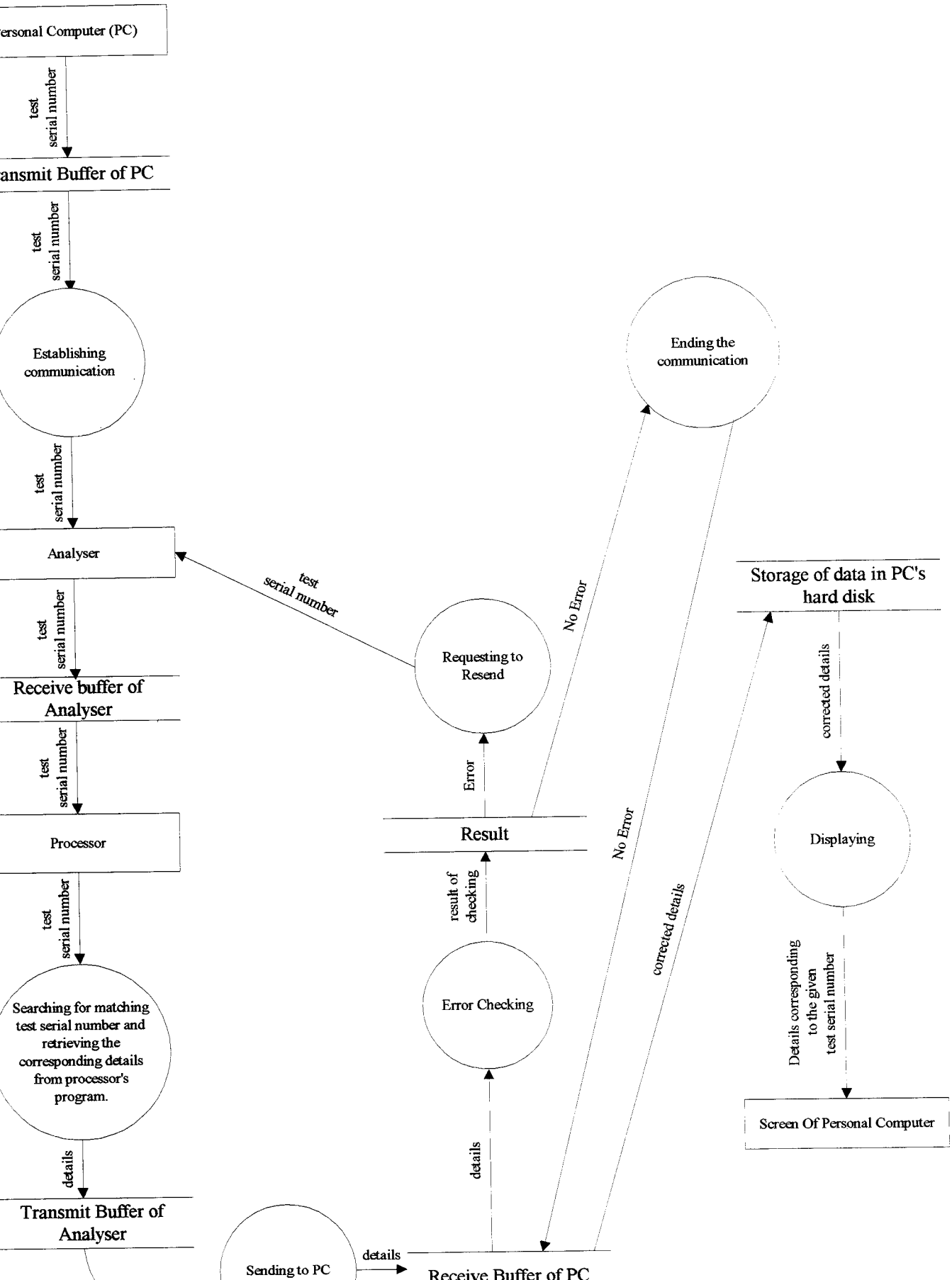
Inputs/Outputs to the Software 

Processing the inputs 

Data Flows 

Data Stores 

DFD For Interfacing The Fully Automated Diagnostic Analyser To PC



2.6.5 Other Requirements:

Operations Required By User:

The corresponding details from the Fully Automated Diagnostic Analyser under the given test serial number (i.e. input) will be displayed on the screen of the PC.

The output format of the result is as follows:

Sample Input:

Enter the Sample Number (or) Test Serial Number : 98

Output Format:

N;98;Ganesan;;M;4;GLU;97;mg/dl;a;UREA;29;mg/dl;;UA;3.2;mg/dl;N;ALT;43;U/l;N;

Sample number : 98
Sample name : Ganesan
Date of Birth : NILL
Sex : M
Number of tests : 4

Type : N
Test name : GLU
Result : 97
Unit : mg/dl
Falg : a

Type : N
Test name : UREA
Result :29

Type : N
Test name : CHOL
Result : 222
Unit : mg/dl
Flag : a

Type : N
Test name : TRIG
Result : 83
Unit : mg/dl
Flag : NILL

Type : N
Test name : Ca
Result : 11.1
Unit : mg/dl
Flag : NILL

Type : N
Test name : DHDL
Result : 51.9
Unit : mg/dl
Flag : NILL

3. DESIGN DOCUMENT

3.1 EXTERNAL DESIGN SPECIFICATIONS

3.1.1 Logical format of data files

In this software there are seven files they are Results.TXT file, Anstruct.H file, Test.C file, Realdata.DAT file, Port.C file, PC.C file, Analyser.C file,Single.TXT file.

Results.TXT file:

Results.TXT file contains the results and details of each and every person who has undergone the tests and there exists semicolons which are treated as delimiters. This file exists in the Fully Automated Diagnostic Analyser.

Eg:

Here is the sample result that exists in the Results.TXT file in the Fully Automated Diagnostic Analyser.

N;384;Thomas;;M;5;GLU;127;mg/dl;Na;CHOL;222;mg/dl;a;TRIG;83;mg/dl;;Ca;11.1;mg/dl;;DHDL;51.9;mg/dl;;

Description of the above result:

Sample number (or) Test serial number : 384
Sample name : Thomas
Date of Birth : NILL
Sex : M
Number of tests : 5

Test No : 1
Type : N

Test name : GLU
Result : 127
Unit : mg/dl
Flag :Na

Test No : 2

Type : N
Test name : CHOL
Result : 222
Unit : mg/dl
Flag : a

Test No : 3

Type : N
Test name : TRIG
Result : 83
Unit : mg/dl
Flag : NILL

Test No : 4

Type : N
Test name : Ca
Result : 11.1
Unit : mg/dl
Flag : NILL

Test No : 5

Type : N
Test name : DHDL
Result : 51.9
Unit : mg/dl
Flag : NILL

Here Type indicates the type of the test undergone for instance whether it is normal, calibrator, pediatric tests. Then Test name indicates the name of the chemical that is been tested in the sample. Result indicates the quantity of the chemical in the sample that is been tested. Unit indicates the standard of measurement for the chemical that is been tested in the sample. Flag deals with working of the Fully Automated Diagnostic Analyser.

In the above description we can see that Date of Birth is NILL, this is because the patients date of birth does not exist in the Fully Automated Diagnostic Analyser, as it might not be entered by the Pathologist in the Fully Automated Diagnostic Analyser.

It can also be seen that in Test No: 3,4 and 5 the flag is NILL, this is because the Fully Automated Diagnostic Analyser by default assigns some flags in the results, these flags deal with working of the Fully Automated Diagnostic Analyser.

Anstruct.H File:

This file is used to initialize the structure format in order to handle the data that is the results and details in the Fully Automated Diagnostic Analyser. Here the result structure consists of type of test, sample number, sample name, date of birth, sex, number of test and result. Which in turn in the test structure sample number, test name, result, unit and flag are present.This Anstruct.H file is used in the Test.C file for binary conversion and also in Analyser.C file inorder to handle the fields of data.

Test.C file:

This file uses the Anstruct.H file, Results.Txt file. In this file the results from Results.Txt file are read, the results that are read are put in the form of different fields using structures and then these results are converted to binary format and it is written into Realdata.DAT file. Thus this file is mainly used for binary conversion.

Realdata.DAT file:

This file contains the binary format of the results of the Fully Automated

Port.C File:

This file is used to identify the port addresses of the PC and the Fully Automated Diagnostic Analyser. The port address of the PC was found to be COM1 that is serial communication port one was 0x3F8 and COM2 that is serial communication port two was 0x2F8. The port of the analyser was found to be 0x2F8 which is the same as that of COM2 of PC.

PC.C File:

In this file the serial communication settings are done such as defining the Port addresses here as the serial communication port address of the Fully Automated Diagnostic Analyser address is 0x2F8 and one of the serial communication port address that is COM2 of the PC is 0x2F8 the communication is to established using the port with address 0x2F8 since the port address is same in both PC and Fully Automated Diagnostic Analyser , then setting the baud rate to 19,200 BPS, turning on the pins DTR(Data Terminal Ready), RTS (Request to send). Then enabling the FIFO control register by using its address. Then setting the number of bits to be transmitted such as 8 bits which contains one parity bit and one stop bit. The parity bit is assigned for odd parity checking. Then this file also deals with establishing the communication with the Fully Automated Diagnostic Analyser that is if data is passed from PC it should reach the Fully Automated Diagnostic Analyser and similarly if the data is sent from Fully Automated Diagnostic Analyser the PC should accept the data.

Analyser.C File:

This file mainly reads the Realdata.DAT file, retrieves the data corresponding to the request sent by the PC. This retrieved result which is in binary format is converted to text format and the retrieved result in the text format is written to Single.TXT file. The data in this Single.TXT file is read and it is sent to the PC character by character.

In this file the serial communication settings are done such as defining the Port addresses here as the serial communication port address of the Fully Automated Diagnostic Analyser address is 0x2F8 and one of the serial communication port address that is COM2 of

ince the port address is same in both PC and Fully Automated Diagnostic Analyser , then etting the baud rate to 19,200 BPS, turning on the pins DTR(Data Terminal Ready), RTS Request to send). Then enabling the FIFO control register by using its address. Then setting he number of bits to be transmitted such as 8 bits which contains one parity bit and one stop oit. The parity bit is assigned for odd parity checking. Then this file also deals with establishing the communication with the PC that is accepting the input from the PC character by character and transmitting the results and details to the PC character by character. Then if there are any transmission errors or the data for the request is not found then it should the message to the PC.

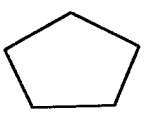
Single.TXT File:

This file contains a single record that is of text format which is been converted from binary format. During each cycle of retrieving the data from the Fully Automated Diagnostic Analyser the text format of data which is been converted from binary format will be written to this file and after sending the data to the PC successfully this file will be cleared, for the next record to be stored and sent.

3.2 ARCHITECTURAL DESIGN SPECIFICATIONS

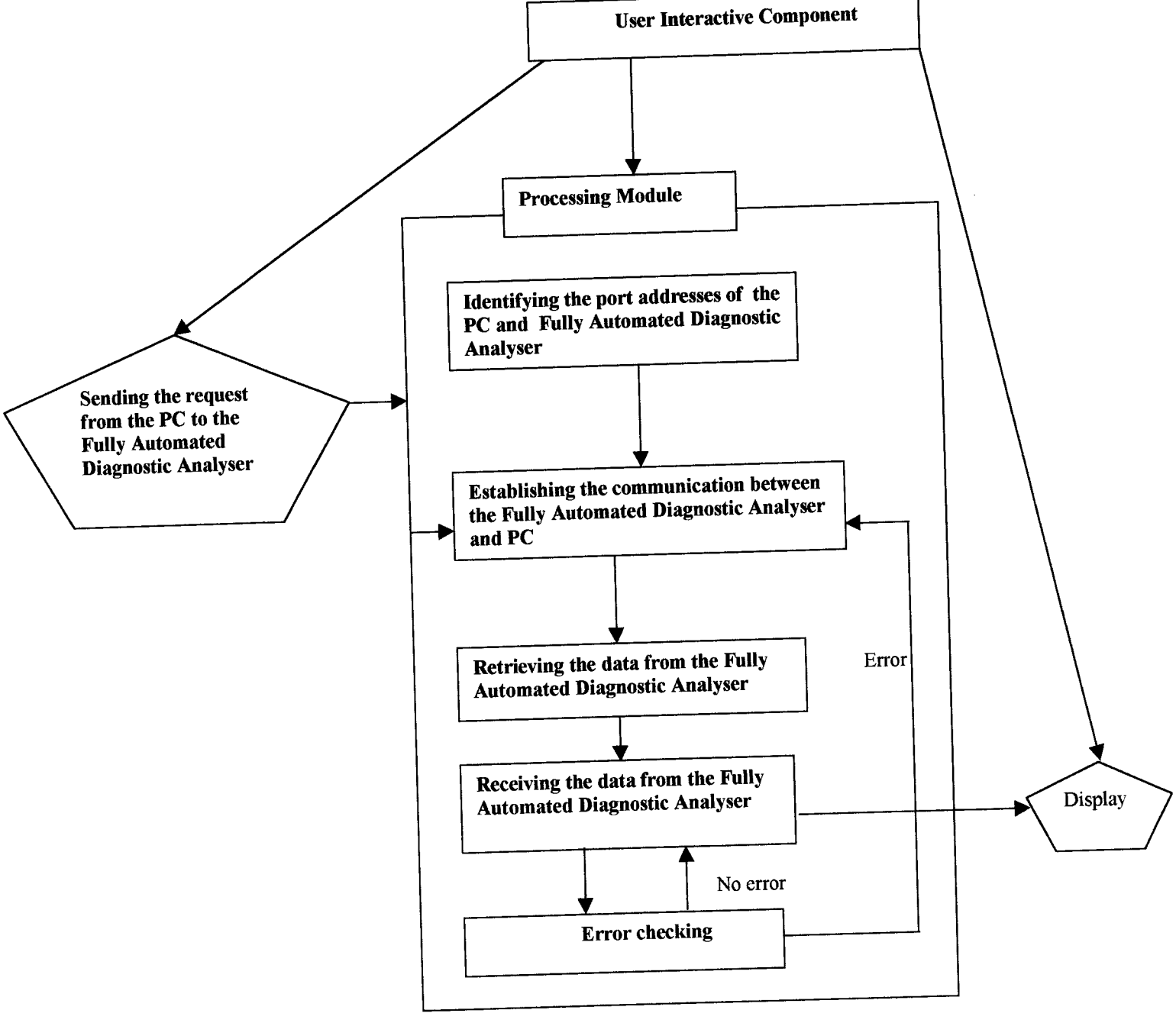
3.2.1 Flowchart :

The flowchart explains the details about the product structure. What are the inputs and outputs to the product structure, how the product is compressed into six modules and its name is also mentioned.



Input/Output Module

Processing Module


User Interactive Component
Processing Module
Identifying the port addresses of the PC and Fully Automated Diagnostic Analyser
Sending the request from the PC to the Fully Automated Diagnostic Analyser
Establishing the communication between the Fully Automated Diagnostic Analyser and PC
Retrieving the data from the Fully Automated Diagnostic Analyser
Error
Receiving the data from the Fully Automated Diagnostic Analyser
Display
No error
Error checking

.2.2 Parameter Specifications:

The input data consists of the following parameters.

nput Data	Type	Member Variables
Sample number	char	samp_number[12]

The output consists of the following parameters.

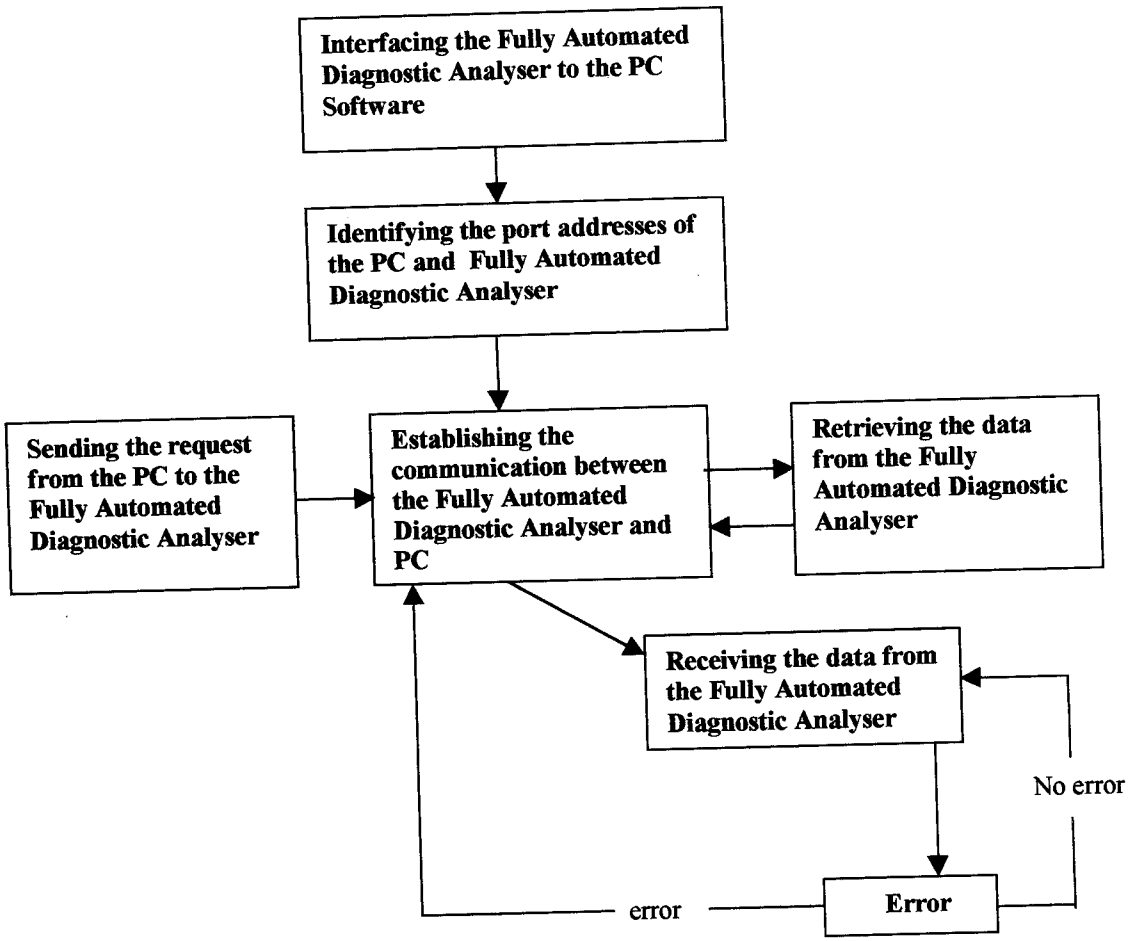
Output Data	Type	Member Variables
Type of test	char	type[1]
Sample number	char	samp_number[12]
Sample name	char	sample_name[30]
Date of Birth	char	date_of_birth[11]
Sex	char	sex[1]
Number of tests	char	num_of_test[2]
Test name	char	test_name[4]
Results	char	result[250]
Unit	char	unit[6]
Flag	char	flag[22]

3.2.3 Functional description:

The "Interfacing the Fully Automated Diagnostic Analyser" software is decomposed into six modules based on the function they perform. The six main modules are as follows.

- Identifying the Port addresses of PC and Fully Automated Diagnostic Analyser
- Establishing the connection between the Fully Automated Diagnostic Analyser and the Personal Computer(PC)

- Retrieving the data from the Fully Automated Diagnostic Analyser
- Receiving the data from the Fully Automated Diagnostic Analyser
- Error Checking.



1. Identifying the Port addresses of PC and Fully Automated Diagnostic Analyser :

Module Used:

Here the Identifying the Port addresses of PC and Fully Automated Diagnostic Analyser is used.

Processing Narrative:

Usually there are three communication ports in a PC, such that two serial communications port COM1, COM2 and one parallel communication port. The serial communication port of the PC is 9 pin male port that supports asynchronous communication and bi-directional host connection which is considered to be the Data Terminal Equipment (DTE). The serial communication port of the Fully Automated Diagnostic Analyser is 25 pin female port that supports asynchronous communication and bi-directional host connection which is considered to be the Data Communication Equipment(DCE). The serial communication port address are identified for both the PC and Fully Automated Diagnostic Analyser. Then one of the serial port of the PC is selected. A null-modem RS232 standard interface cable is used to connect the serial port of the PC and the serial port of the Fully-Automated Diagnostic Analyser.

Comments:

Here the serial communication port addresses of the PC such that COM1 was found to be 0x2F8 and COM2 was found to be 0x3F8.The serial communication port address of the Fully Automated Diagnostic Analyser was found to be 0x2F8.

2. Establishing the communication between the Fully Automated Diagnostic Analyser and the Personal Computer(PC):

Module Used:

Here the Establishing the connection between the Fully Automated Diagnostic Analyser and the Personal Computer(PC) module is used.

Processing Narrative:

In this module the baud rate, that is a measure of transmission speed, equivalent to about one bit per second, is set to 19,200 BPS. It is required that the baud rate should be the

same in both the PC and the Fully Automated Diagnostic Analyser, in order to facilitate a perfect serial communication between the COM ports.

In order to transmit the data from the PC the data should be taken to the transmit buffer of USART of the PC. From the transmit buffer the data should be transmitted through the port, through the RS232 null modem cable to the receive buffer of USART of the Fully Automated Diagnostic Analyser. Then the transmitted data from the PC to the Fully Automated Diagnostic Analyser that is now in the receive buffer of the analyser can be viewed.

Similarly a data from the processor's program of the Fully Automated Diagnostic Analyser is taken to the transmit buffer of the Fully Automated Diagnostic Analyser, then the data is sent through the port, through the RS232 null modem cable to the PC.

Thus communication is established between the Fully Automated Diagnostic Analyser and the PC.

Comments:

The serial communication port address of the Fully Automated Diagnostic Analyser address is 0x2F8 and one of the serial communication port address that is COM2 of the PC is 0x2F8 and that the communication is to be established using these two ports which has the same address. COM1 which has the address 0x3F8 should not be connected to Fully Automated Diagnostic Analyser as the port address differs and that the communication cannot be established using COM1.

3. Sending the request from the PC to the Fully Automated Diagnostic Analyser:

Module Used:

Here the Sending the request from the PC to the Fully Automated Diagnostic Analyser module is used.

Processing Narrative:

The test serial number is got as an input on the PC. The input is converted into string. As only one character at a time can be sent through the serial cable, split the converted string into single characters and then the modified input that is the characters are transmitted one at a time through the RS232 serial cable, which is a standard interface, to the receive buffer of USART of the Fully Automated Diagnostic Analyser.

Comment:

Using serial communication ports and serial interface data can be transmitted such that one bit at a time.

4. Retrieving the data from the Fully Automated Diagnostic Analyser:

Module Used:

Here the Retrieving the data from the Fully Automated Diagnostic Analyser module is used.

Processing Narrative:

The module deals with Retrieving the data from the processor program of the Fully Automated Diagnostic Analyser. Here the characters that are sent by the PC are collected as a string. The real-time data which is in the text format is converted into binary format, and this binary format data is stored in a file. As the length of the results and details may vary for various test serial numbers, the binary format data is required to find the corresponding details and results of the requested test serial number.

Comment:

In binary format the details of patient and complete record of results are retrieved

5. Receiving the data from the Fully Automated Diagnostic Analyser:

Module Used:

Here the Receiving the data from the Fully Automated Diagnostic Analyser module is used.

Processing Narrative:

After receiving the test serial number from the PC, in the Fully Automated Diagnostic Analyser it is required to check whether the given test serial number is available in the binary file, that had been created in fourth module. If it is available the corresponding result and details should be retrieved. As this result will be in binary format it is required to convert this binary result into string. The string is then splitted into characters and then it is sent to the receive buffer of the PC through the RS232 null modem serial cable.

Comment:

The receiving process will be done as one bit at a time because of serial communication.

6. Error Checking:

Module Used:

Here the Error Checking module is used.

Processing Narrative:

Here in order to avoid transmission errors, odd parity checking is done, this is applied to both, the Fully Automated Diagnostic Analyser so that the test serial number from the PC is transmitted properly, and also in PC so that the details that are transmitted by the Fully Automated Diagnostic Analyser have been received by the PC properly. If there occurs a

error message will be displayed on the screen of the PC and the retransmission of the results and details from the Fully Automated Diagnostic Analyser is requested by the PC. A provision is also made such that if any of the test serial number is not found in the Fully Automated Diagnostic Analyser then a message will be displayed on the PC.

Comment:

The foulness in results due to transmission error is avoided. Any requested data which is not found in Fully Automated Diagnostic Analyser is also informed.

3.3 Packing Specification:

The software "Interfacing the Fully Automated Diagnostic Analyser to the PC" is left on the systems such as Fully Automated Diagnostic Analyser and in the PC. The organization needs to run the executable program form the respective systems. There is no special packing done for this software.

4. TEST PLANS

Type of test : Functional test

Machine Configuration :

Processor: Pentium III 550 MHz

Floppy Disk Drive: 1.44 MB

Hard Disk: 20GB

System RAM: 64MB

Serial Ports: 2

- one for analyser
- one for host

Printer Port: 1

External Interface: RS232 Cable

Test assumption: Invalid input

Exact Test Stimuli:

If the user enters an input that is the test serial number in the PC which is not found in the Fully Automated Diagnostic Analyser then a message should be sent from the Fully Automated Diagnostic Analyser to the PC such that the test serial number is not found in the Fully Automated Diagnostic Analyser.

Expected outcome:

The PC displays the message "The test serial number not found in the Fully Automated Diagnostic Analyser".

2. **Type of test** : Performance test

Machine Configuration :

Processor: Pentium III 550 MHz

Floppy Disk Drive: 1.44 MB

Hard Disk: 20GB

System RAM: 64MB

Serial Ports: 2

- one for analyser
- one for host

Printer Port: 1

External Interface: RS232 Cable

Test assumption: Overcoming transmission errors from Fully Automated Diagnostic Analyser

Exact Test Stimuli:

When the Fully Automated Diagnostic Analyser sends the details and results to the PC character by character in some occasions the data might be lost or there might be additional junk characters. In such a case the PC on identifying the transmission error should display a transmission error message and ask the user to wait and then it should request the Fully Automated Diagnostic Analyser to resend the data.

Expected outcome:

The PC displays the message "Transmission error Please Wait".

3. **Type of test** : Performance test

Machine Configuration :

Processor: Pentium III 550 MHz

Floppy Disk Drive: 1.44 MB

Hard Disk: 20GB

System RAM: 64MB

Serial Ports: 2

- one for analyser
- one for host

Printer Port: 1

Test assumption: Overcoming transmission errors from the PC.

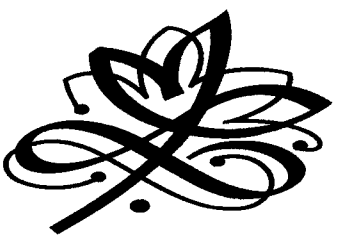
Exact Test Stimuli:

When the PC sends the input to the Fully Automated Diagnostic Analyser as character by character in some occasions the data might be lost or there might be additional junk characters. In such a case the Fully Automated Diagnostic Analyser on identifying the transmission error should send a transmission error message in input and ask the user to reenter the input.

Expected outcome:

The PC displays the message "Transmission error in Test serial number. Enter the Test serial number".

Project Legacy



5. PROJECT LEGACY

5.1 PROJECT DESCRIPTION

The project is concerned with a product Fully Automated Diagnostic Analyser. The Fully Automated Diagnostic Analyser does various tests such as blood, urine, drugs etc. The results of the tests are stored in the processor program in the Fully Automated Diagnostic Analyser. The Fully Automated Diagnostic Analyser should be interfaced with a PC in order to capture the results from the Fully Automated Diagnostic Analyser as per the request given in the PC by the user. The result brought to the PC serves as a reference, which can be used for investigations, billing purposes and also used to present a clear report which will be given as a hardcopy to the patient. This interfacing plays an important role in which it provides a great convenience for the users as they can capture the results from the Fully Automated Diagnostic Analyser through the PC by sitting in their seats without going to the room where the Fully Automated Diagnostic Analyser is placed.

5.2 INITIAL EXPECTATIONS

The objective of "Interfacing the Fully Automated Diagnostic Analyser to the PC" software is to capture the details and results from the Fully Automated Diagnostic Analyser as per the request that is given as input by the user in the PC, and the retrieved details and results from the Fully Automated Diagnostic Analyser should be stored in the PC and also the retransmission of data is requested whenever there occurs transmission errors. First the port addresses of the Fully Automated Diagnostic Analyser and the PC are identified, then the communication between the PC and the Fully Automated Diagnostic Analyser is established. The user's request according to which the details and results are to be captured from the Fully Automated Diagnostic Analyser is got as input. Then this input is to be sent to the Fully Automated Diagnostic Analyser which is dealed by the module Sending the request from the PC to the Fully-Automated Diagnostic Analyser. Then the retrieving process takes place in the Fully Automated Diagnostic Analyser which is dealed by the module Retrieving the data

from the Fully-Automated Diagnostic Analyser. Then the process of sending the retrieved details and results from the Fully-Automated Diagnostic Analyser to the PC, and making the PC to accept the data sent by the Fully-Automated Diagnostic Analyser is dealed in the module Receiving the data from the Fully-Automated Diagnostic Analyser. Then in order to avoid transmission errors and also to indicate if details and results of any user's request is not available in Fully-Automated Diagnostic Analyser is dealed in Error checking module.

5.3 SOLUTION STRATEGY:

The problem was approached in a step by step fashion. First and foremost the functions of the Fully Automated Diagnostic Analyser and its purpose is learnt. Then the functions of the serial communications port, Universal Synchronous Asynchronous Receiver Transmitter (USART), RS232 null modem cable are learnt. Then analysis is done on problem definition given by the Project Manager. Then the rough draft is made on the analysis and finally ends up in a solution by breaking down the defined problem into six modules.

It is required to transfer the data to a separate Personal Computer (PC), that is the DTE, from the DCE, that is the Fully Automated Diagnostic Analyser. So that some other software in the PC can make use of this data and it helps out in giving out a detailed report and for also as reference.

For this problem interfacing the Fully Automated Diagnostic Analyser to the PC is required and this is done through RS232 standard interface cable. When the operator types the test serial number in the PC, this test serial number is taken to the transmit buffer of USART of the PC, then this test serial number from the transmit buffer of USART of the PC is transmitted through the RS232 interface cable, through the COM port towards the receive buffer of USART of the Fully Automated Diagnostic Analyser. The results and details regarding the test is stored in text format in the Fully Automated Diagnostic Analyser, as the number of tests undergone varies from one person to another, it is required to convert the text format results into binary format, so that all the results can be captured clearly.

The test serial number now in the receive buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the processor program of the Fully Automated Diagnostic Analyser.

The test serial number which is the operator's request, from the PC, which is now in the receive buffer of USART of the Fully Automated Diagnostic Analyser should be seen whether it matches with the test serial number stored in the binary format file of the analyser. Once it gets matched the details corresponding to that test serial number is retrieved and it is then converted into text format and then it is sent to the transmit buffer of USART of the Fully Automated Diagnostic Analyser. Then the data from the transmit buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the receive buffer of USART of the PC through the RS232 null modem cable, through the COM port.

The PC should accept the details store those details in a buffer and error checking is done, if there is any error then request is passed to the Fully Automated Diagnostic Analyser to resend else the required details are placed in PC in the text format for further uses.

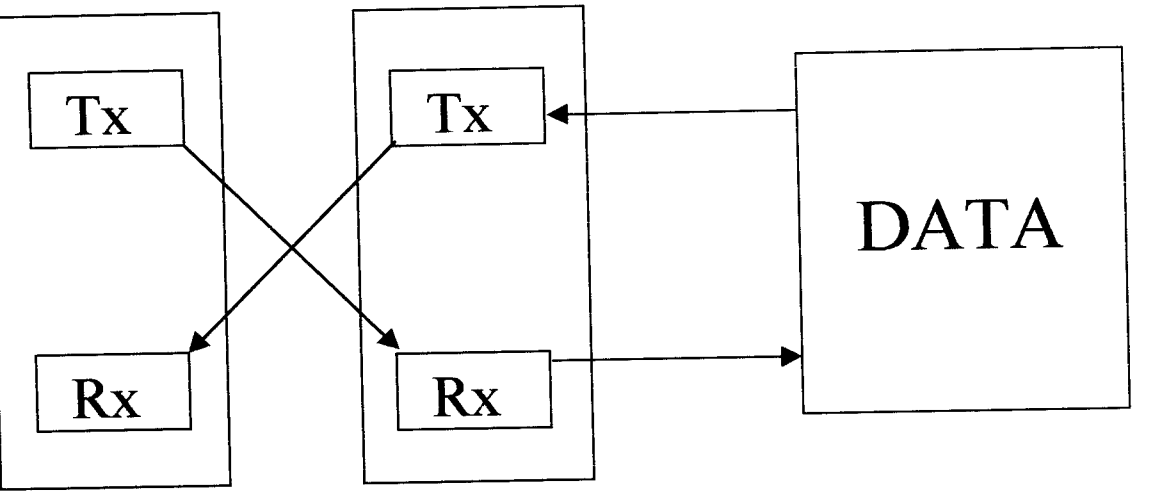
5.4 SOFTWARE PRODUCT FEATURES:

Interfacing the Fully Automated Diagnostic Analyser to the PC is done through RS232 standard interface cable. When the operator types the test serial number in the PC, this test serial number is taken to the transmit buffer of USART of the PC, then this test serial number from the transmit buffer of USART of the PC is transmitted through the RS232 interface cable, through the COM port towards the receive buffer of USART of the Fully Automated Diagnostic Analyser. The results and details regarding the test is stored in text format in the Fully Automated Diagnostic Analyser, as the number of tests undergone varies from one person to another, it is required to convert the text format results into binary format, so that all the results can be captured clearly.

Technical Representation of Software

PC (DTE) **Fully Automated Diagnostic Analyser (DCE)**

USART USART Processor program



The test serial number now in the receive buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the processor program of the Fully Automated Diagnostic Analyser.

The test serial number which is the operator's request, from the PC, which is now in the receive buffer of USART of the Fully Automated Diagnostic Analyser should be seen whether it matches with the test serial number stored in the binary format file of the analyser. Once it gets matched the details corresponding to that test serial number is reterived and it is then converted into text format and then it is sent to the transmit buffer of USART of the Fully Automated Diagnostic Analyser. Then the data from the transmit buffer of USART of the Fully Automated Diagnostic Analyser is transmitted to the receive buffer of USART of the PC through the RS232 null modem cable, through the COM port.

The PC should accept the details store those details in a buffer and error

Diagnostic Analyser to resend else the required details are placed in PC in the text format for further uses.

5.5 ACCEPTANCE CRITERIA:

The software has to establish a perfect communication between the PC and the Fully Automated Diagnostic Analyser.It should accept the requirement that is some input values from the user. It should also accept the data that is sent by the Fully Automated Diagnostic Analyser.

5.6 PROJECT PLAN

5.6.1 Life Cycle Mode:

The Spiral Model is proposed to be the life cycle mode followed while developing the product software. It provides the potential for the rapid development of incremental version of the software. The software is developed in the series of incremental releases. The Spiral Model has six tasks region.

Task Region 1:

- Terminology: Customer Communication
- Milestones: May 15th- May 17th
- Work Product: The project leader defines about interfacing the Fully Automated Diagnostic Analyser (DCE) to the PC (DTE). Reading some notes from the websites, the manual of the Fully Automated Diagnostic Analyser and some books related to Fully Automated Diagnostic Analyser and its use. The "Interfacing the Fully Automated Diagnostic Analyser to the PC" software should establish the communication between the PC and the Fully Automated Diagnostic Analyser, and it should also satisfy the user's request by retrieving the requested

details and results from the Fully Automated Diagnostic Analyser with the help of the PC.

Task Region 2:

- Terminology: Planning
- Milestones: May 18th - May 23rd
- Work Product: Analysis of the software product definition. What function the software product has to perform, Processing environment, Software features, Programming language to be used are all decided in this stage.

Task Region 3:

- Terminology : Risk Analysis
- Milestones: May 24th - May 28th
- Work Product:

Technical Risk:

In order to establish the communication between the Fully Automated Diagnostic Analyser and the PC, the details of the serial communication ports, USART in the PC and in the Fully Automated Diagnostic Analyser should be known. The details of the hardware interface that is the RS232 null modem cable should also be known. Then the details about the Fully Automated Diagnostic Analyser such as what type of software it can accept in its operating system are known.

Managerial Risk:

For every module time limits are set to be fixed for its completion. The project duration is between May 15th -Sept 11th. Within the given time slot for each module it has to be completed. The modules cannot be postponned because output of one module becomes the input

is a delay in completion of the whole project. Therefore the modules have to be completed in the appropriate time as per schedule.

Task Region 4:

- Terminology: Engineering
- Milestones: SRS Document May 28th - May 31st
- Work Product: Based on the needs of the customer, the software requirement specification is prepared. SRS includes Product

- Milestones: Design Document June 1st - June 17th
- Work Product: Based on the needs of the customer, the Design document is prepared. Designing plays an important role in coding. Once the design is framed well, the programmer can start the coding very easily. Design document includes external design specification, Architectural design overview and Detailed design Specification.

Task Region 5:

Terminology: Construction and Release.

Milestones	**Work Product**
June 18th - June 21st	Coding for Identifying the Port addresses of PC and the Fully Automated Diagnostic Analyser Module
June 22nd - June 28th	Coding for Establishing the communication between the Fully Automated Diagnostic Analyser and the PC module

June 28th - July 22nd	Sending the request from the PC to the Fully Automated Diagnostic Analyser module
July 22nd - Aug 9th	Retrieving the data from the Fully Automated Diagnostic Analyser module
Aug 10th - Aug 16th	Receiving the data from the Fully Automated Diagnostic Analyser module
Aug 17th- Aug 23rd	Error Checking module
Aug 24th- Aug 31st	Preparing user manual.
Aug 1st- Sept 3rd	Preparing Test Plan.
Sept 4th- Sept 6th	Installing Exe file.

Task Region 6:

- Terminology: Customer Evaluation
- Milestones: Sept 8th - Sept 9th
- Work Product: On seeing the execution, the customer gave the feedback. His feedback was the product works perfectly and it can be extended in future for most of the bio-medical equipments.

5.6.2 Team Structure :

The "Interfacing the Fully Automated Diagnostic Analyser to the PC" software is used for retrieving the details and results from the Fully Automated Diagnostic Analyser as per the user's request.

The project consists of one member under the guidance of Project leader.

5.6.3 Development Schedule :

In order to complete the project in a given time, the development schedule is framed and based on the time slots, the software product is developed. The development schedule consists of Milestones and Reviews.

Milestones	Reviews
May 17th - Product Definition	A rough draft is made to Product Definition and definitions are reviewed.
May 20th - Product Analysis	A rough draft is made to Product Analysis and the review is made on the analysis to do step by step fashion.
May 23rd - Programming language	The Programming Language is decided.
May 28th - Risk Analysis	A rough draft is made to Risk Analysis. The draft was reviewed and there was two introduction types of risks like Technical Risk and Managerial Risk.

May 31st - SRS	SRS general formats are reviewed.
June 17th - Design Documentation	Design Document was reviewed.
June 21st - Coding for Identifying the Port addresses of PC and the Fully Automated Diagnostic Analyser module.	The reviews were made on the logic.
June 28th - Coding for Establishing the communication between the Fully Automated Diagnostic Analyser and the PC module	The reviews were made on the technical initialization part
July 22nd - Sending the request from the PC to the Fully Automated Diagnostic Analyser module	The reviews were made on the logic.
Aug 9th - Retrieving the data from the	The reviews were made on the logic. Fully Automated Diagnostic Analyser module
Aug 16th - Receiving the data from the	The reviews were made on the logic. Fully Automated Diagnostic Analyser Module
Aug 23rd - Error Checking module	The reviews were made on the logic.
Aug 31st - Prepare user manual	Reviews were made on the rough draft of the user manual.
Sept 2nd - Test Plans	After doing the various test plans, reviews were made on rough draft of the

Sept 7th - Installing Exe program No Reviews

Sept 9th - Full project demonstration No Reviews

5.6.4 Programming Language :

'C' is the programming language that is been used in the software "Interfacing the Fully Automated Diagnostic Analyser to the PC".

'C' is a general purpose structured programming language that is powerful, efficient and compact . 'C' combines the features of a high-level language with the elements of the assembler and is thus ,close to both man and machine.

Turbo C Compiler is the most widely used professional software development tool on the micro computers.

Features of the 'C' Language:

C language as such offers only a handful of functions which form the core of the language,rest of the functions available in libraries are developed using the core functions offered by the language as building blocks. This feature expands the scope and the power of the language. This leads to functionally cohesive modules and , therefore, re-usable code.

Thus highly independent functions can be written and stored in a library containing other functions to be used as when necessary.

Thus C language makes it possible to use the language for systems programming, like development of compilers, interpreters, operating systems, graphics and general utilities, and also for a host of applications in the commercial environment.

5.6.5 Documents to be prepared:

It was suggested the following documents are to be prepared during the time of the project.

- A System Definition consisting of a Product Definition and a Project Plan.
- A Software Requirement Specification.
- A Design document consisting of external design and architectural design
- A Test Plan
- A User's manual
- A Project Legacy document.

5.6.6 Manner of Demonstration:

Reviews:

Every week on Wednesday the finished modules are explained to the project leader, reviewed and inputs are taken.

Documents:

Draft of every document is reviewed by the project leader before it is finalized. If there are any changes to the draft they are incorporated in the document.

Product:

Demo of the each module is given to the project leader as and when the module is completed. If any changes are required, they are incorporated in the module after the review.

5.7 CURRENT STATUS OF THE PROJECT

The "Interfacing the Fully Automated Diagnostic Analyser to the PC" software is able

Fully Automated Diagnostic Analyser and the PC, establishing the communication between the the Fully Automated Diagnostic Analyser and the PC, accepting the inputs from the user in the PC, then sending those input values to the the Fully Automated Diagnostic Analyser, retrieving the required results and details form the Fully Automated Diagnostic Analyser as per the user's request which was given as input, sending the retrieved details and results from the Fully Automated Diagnostic Analyser to the PC, making the PC to accept the data that is been sent by the Fully Automated Diagnostic Analyser and storing it in a file, then requesting the retransmission of the data whenever there occurs transmission errors.

5.8 ACTIVITIES / TIME LOG(S)

Time Logs	**Activities**
May 17th	Product Definition
May 20th	Product Analysis
May 23rd	Programming language
May 28th	Risk Analysis
May 31st	SRS
June 17th	Design Documentation
June 21st	Coding for Identifying the Port addresses of PC and the Fully Automated Diagnostic Analyser module

June 28th	Coding for Establishing the communication between the Fully Automated Diagnostic Analyser and the PC module
July 22nd	Sending the request from the PC to the Fully Automated Diagnostic Analyser module
Aug 9th	Retrieving the data from the Fully Automated Diagnostic Analyser module
Aug 16th	Receiving the data from the Fully Automated Diagnostic Analyser module
Aug 23rd	Error Checking module
Aug 31st	Prepare user manual
Sept 2nd	Test Plans
Sept 7th	Installing Exe program
Sept 9th	Full project demonstration

5.9 TECHNICAL LESSONS LEARNED

Many technical lessons were learned while working in this project. Out of many techniques the best one is selected, learnt and implemented in the case of Problem analysis

Based on the problem approach and its solution the software is selected. The software used is 'C'. 'C', plays a main role in order to attain perfection in interfacing the two devices.

Many electrical and electronics communication techniques were learnt to approach the problem. Technical terms like DTE, DCE, USART, Baud rate and details and pin configurations of serial port and RS232 null modem cable were learnt. Some websites and manuals were also referred to know about the Fully Automated Diagnostic Analyser. Some technical books were also referred in order to interface the Fully Automated Diagnostic Analyser to the PC.

5.10 MANAGERIAL LESSONS LEARNT

In this organization apart from the technical lessons, managerial lessons were also learnt, that includes - How to approach the problem patiently and how to discuss about the evaluation of the problem and express my solution regarding the problem in the conference. The way of communicating with my Project Leader and programmers and asking them suggestions and advises in some tough times while programming. The time slots were framed by me and the Project leader to finish each and every part of the module. Suppose if there is a delay in coding for a particular module, the time for the very next modules timing should be adjusted in order to finish the project in a given time slot. As and when the modules are finished the time taken to finish is noted in a sheet of paper and it is compared with the original time slot.

5.11 RECOMMENDATIONS TO FUTURE PROJECTS

In this organization the projects are mainly concerned with developing software for medical equipments. Many of these projects are concerned with electronics such as automating the medical equipments. Many projects can be done in future in other equipments which are used in surgery.

Conclusion

6. CONCLUSION

The " Interfacing the Fully Automated Diagnostic Analyser" software is expected to revolutionize the present day Bio - Medical Instrumentation field with its state of the facilities moving hand - in - hand with today's fast paced technologies. Using this software either the doctor in the hospital or a receptionist in a laboratory can capture the results of a particular person, by sitting in front of the PC that is interfaced to the Fully Automated Diagnostic Analyser, without entering into the room where the equipment, Fully Automated Diagnostic Analyser is placed. The PC is any general purpose compatible and that in turn can be used for various other purposes also. This serves to be very flexible for doctors and other users such as receptionists, in saving the time otherwise for which they have to move to the equipment's room in order to see the result. The results if required can be stored in the PC for using with other software and it is confidential and accurate as per it is stored in the Fully Automated Diagnostic Analyser.

The software with some modifications can be applied to all the equipment that contains a serial port. Thus the software can be widely used in Bio - Medical Instrumentation, Electrical and Electronic Instrumentation and also in Electrical and Electronic Communication fields.

The programming style used adds to the flexibility of the system and allows easy modification to the existing system. Appropriate comments and suggestions are specified throughout the code, which make the code easily understandable. Clear documentation and supporting documents along with appropriate diagrams have been produced which aids in understanding the overall system architecture.

Apart from these I gained a lot of technical knowledge, industrial mannerism and behavior. I was able to understand and learn more about the real world project life - cycle.

REFERENCES

Books:

- Mark Goodwin, **"Serial Communications in C and C++"**, MIS Press(Pitman),1992.
- **VITALAB SELECTRA** Manual book.
- Charles H.Roth, Jr. University Of Texas At Austin, **"Fundamentals Of Logic Design"**, Jaico Publishing House 2002.
- Ramesh S.Gaonkar, State University Of New York **"Microprocessor Architecture, Programming and Applications with 8085"**,Prentice Hall 1996
- Roger S. Pressman, **"Software Engineering"**, McGraw-Hill International Edition 2001.

Web Sites:

www.google.com

www.VitaLab.com

www.howstuffworks.com

www.ebooks.com

http://www.beyondlogic.org/about.html

A. USER'S MANUAL

A.1 INTRODUCTION:

A.1.1 Product rational and overview:

This Product is mainly concerned for interfacing the Fully Automated Diagnostic Analyser to the PC.

The objective of "Interfacing the Fully Automated Diagnostic Analyser to the PC" software is to capture the details and results from the Fully Automated Diagnostic Analyser as per the request that is given as input by the user in the PC, and the retrieved details and results from the Fully Automated Diagnostic Analyser should be stored in the PC and also the retransmission of data is requested whenever there occurs transmission errors. First the port addresses of the Fully Automated Diagnostic Analyser and the PC are identified, then the communication between the PC and the Fully Automated Diagnostic Analyser is established. The user's request according to which the details and results are to be captured from the Fully Automated Diagnostic Analyser is got as input. Then this input is to be sent to the Fully Automated Diagnostic Analyser which is dealed by the module Sending the request from the PC to the Fully Automated Diagnostic Analyser. Then the retrieving process takes place in the Fully Automated Diagnostic Analyser which is dealed by the module Retrieving the data from the Fully Automated Diagnostic Analyser. Then the process of sending the retrieved details and results from the Fully-Automated Diagnostic Analyser to the PC, and making the PC to accept the data sent by the Fully Automated Diagnostic Analyser is dealed in the module Receiving the data from the Fully Automated Diagnostic Analyser. Then in order to avoid transmission errors and also to indicate if details and results of any user's request is not available in Fully Automated Diagnostic Analyser is dealed in Error checking module.

A.1.2 Terminology:

The Fully Automated Diagnostic Analyser is an equipment that provides the identification of some or all constituents in a substance or compound. The Fully Automated Diagnostic Analyser is used by the Pathologists for various bio-chemical tests such as blood, drugs etc. The Fully Automated Diagnostic Analyser is that which is fully automated and that the Pathologist have to just place the samples and the reagents in their respective partitions. The process of pipetting will be done with Hamilton Syringes with Valve blocks. The mixing will be done in the reaction rotor. The measurement and the calculations for obtaining the results will be done automatically by the appropriate scales and software. The result can be viewed in the screen attached to the Fully Automated Diagnostic Analyser.The disposal of the waste samples and reagents and the washing of the test tubes will be done automatically. Thus the entire process takes place automatically in the Fully Automated Diagnostic Analyser.

Before going to the "Interfacing the Fully Automated Diagnostic Analyser to the PC" software there are some terms or technical words that has to be known.

- Diagnostic Analyser - An equipment that provides the identification of some or all constituents in a substance or compound.
- Interfacing - A process where a Hardware or Software forms a link between the devices and allows them to communicate with each other.
- Pathologist - A physician or other expert specializing in pathology.
- Pathology - The branch of medical science dealing with the causes, nature, and effects of diseases.
- Port - Any of the connections to a computer that allow the transfer of data.
- Communications Port - Any of the connections on a computer that permit the transfer of
- data as to a peripheral device or to another computer.
- Buffer - An area for temporary storage of data, often used to free resources or to compensate for a difference in transmission speeds between devices.
- Baud Rate - A measure of transmission speed, equivalent to about one bit per second.
- Serial communication - The various bits of data are transmitted one after another, lower order bit first. Eg:RS232 interface.

- Serial Interface - A connector that transmits data sequentially, one bit a time over a single line. E.g.: RS232 interface.
- Asynchronous Communication - The recognition and transmission of data as it becomes available rather than that at timed intervals. Asynchronous communication is more or less continuos because data is transmitted, as it becomes available without the constraints of a timing device.
- Packet - A block of computer data that is transmitted as a unit.
- Standard Interface - Any hardware or software connection between a computer and a peripheral that complies with an industry standard.
- Transmission - The transfer of signals between parts of a computer system.
- Transmission speed - The rate at which data can be sent and received usually expressed as baud rate.
- RS232 - Defines the connection between a terminal (DTE) and communication equipment (DCE), used in interfacing and it speeds up to 20Kbps over 50 ft.

Abbreviations:

- COM Port - Communications Port.
- DTE - Data Terminal Equipment (in this project DTE ->Personal Computer (PC))
- DCE - Data Communications Equipment (in this project DCE->FullyAutomated Diagnostic Analyser)
- USART - Universal Synchronous/Asynchronous Receiver Transmitter.

A.1.3 Basic Features:

The software is mainly developed for interfacing the Fully Automated Diagnostic Analyser to the PC. The "Interfacing the Fully Automated Diagnostic Analyser to the PC" software is able to do all the operations like identifying the serial communication port addresses of the Fully Automated Diagnostic Analyser and the PC, establishing the communication between the Fully Automated Diagnostic Analyser and the PC, accepting the inputs from the user in the PC, then sending those input values to the Fully Automated Diagnostic Analyser, retrieving the required results and details form the Fully Automated

details and results from the Fully Automated Diagnostic Analyser to the PC, making the PC to accept the data that is been sent by the Fully Automated Diagnostic Analyser and storing it in a file, then requesting the retransmission of the data whenever there occurs transmission errors.

A.2 GETTING STARTED

A.2.1 Loading of files:

Totally this software has got seven files they are Results.TXT file, Anstruct.H file, Test.C file, Realdata.DAT file, Port.C file, PC.C file, Analyser.C file, Single.TXT file.

Here follows the brief description of each and every file and how they are to be loaded and used.

Results.TXT file:

Results.TXT file contains the results and details of each and every person who has undergone the tests and there exists semicolons which are treated as delimiters.
This file exists in the Fully Automated Diagnostic Analyser.

Anstruct.H File:

This file is used to initialize the structure format in order to handle the data that is the results and details in the Fully Automated Diagnostic Analyser. Here the result structure consists of type of test, sample number, sample name, date of birth, sex, number of test and result. Which inturn in the test structure sample number, test name, result, unit and flag are present. This Anstruct.H file is used in the Test.C file for binary conversion and also in Analyser.C file in order to handle the fields of data.

This file is to be loaded in the Fully Automated Diagnostic Analyser.

Test.C file:

This file uses the Anstruct.H file, Results.Txt file. In this file the results from Results.Txt file are read, the results that are read are put in the form of different fields using structures and then these results are converted to binary format and it is written into Realdata.DAT file. Thus this file is mainly used for binary conversion.

This file is to be loaded in the Fully Automated Diagnostic Analyser.

Realdata.DAT file:

This file contains the binary format of the results of the Fully Automated Diagnostic Analyser which is been obtained from the Test.C file.

This file will be created in the Fully Automated Diagnostic Analyser by the Test.C file.

Port.C File:

This file is used to identify the port addresses of the PC and the Fully Automated Diagnostic Analyser. The port address of the PC was found to be COM1 that is serial communication port one was 0x3F8 and COM2 that is serial communication port two was 0x2F8. The port of the analyser was found to be 0x2F8 which is the same as that of COM2 of PC.

This file is to be loaded in both the Fully Automated Diagnostic Analyser and in the PC.

PC.C File:

In this file the serial communication settings are done such as defining the Port addresses here as the serial communication port address of the Fully Automated Diagnostic Analyser address is 0x2F8 and one of the serial communication port address that is COM2 of the PC is 0x2F8 the communication is to established using the port with address 0x2F8 since the port address is same in both PC and Fully Automated Diagnostic Analyser , then setting

to send). Then enabling the FIFO control register by using its address. Then setting the number of bits to be transmitted such as 8 bits which contains one parity bit and one stop bit. The parity bit is assigned for odd parity checking. Then this file also deals with establishing the communication with the Fully Automated Diagnostic Analyser that is if data is passed from PC it should reach the Fully Automated Diagnostic Analyser and similarly if the data is sent from Fully Automated Diagnostic Analyser the PC should accept the data.

This file is to be loaded in the PC.

Analyser.C File:

This file mainly reads the Realdata.DAT file, retrieves the data corresponding to the request sent by the PC. This retrieved result which is in binary format is converted to text format and the retrieved result in the text format is written to Single.TXT file. The data in this Single.TXT file is read and it is sent to the PC character by character.

In this file the serial communication settings are done such as defining the Port addresses here as the serial communication port address of the Fully Automated Diagnostic Analyser address is 0x2F8 and one of the serial communication port address that is COM2 of the PC is 0x2F8 the communication is to be established using the port with address 0x2F8 since the port address is same in both PC and Fully Automated Diagnostic Analyser, then setting the baud rate to 19,200 BPS, turning on the pins DTR(Data Terminal Ready), RTS (Request to send). Then enabling the FIFO control register by using its address. Then setting the number of bits to be transmitted such as 8 bits which contains one parity bit and one stop bit. The parity bit is assigned for odd parity checking. Then this file also deals with establishing the communication with the PC that is accepting the input from the PC character by character and transmitting the results and details to the PC character by character. Then if there are any transmission errors or the data for the request is not found then it should the message to the PC.

This file is to be loaded in the Fully Automated Diagnostic Analyser.

Single.TXT File:

This file contains a single record that is of text format which is been converted from

Analyser the text format of data which is been converted from binary format will be written to this file and after sending the data to the PC successfully this file will be cleared, for the next record to be stored and sent.

This file will be created in the Fully Automated Diagnostic Analyser by the Analyser.C file.

Executing operation:

After loading the software in the respective systems such as Fully Automated Diagnostic Analyser and PC, the software in the Fully Automated Diagnostic Analyser is executed first then the software in the PC is executed. Once after the execution of the software in both the systems any number of inputs can be given one by one and that it will work continously until exiting and that the user can make use of this to full extent for retreiving the required details and results from the Fully Automated Diagnostic Analyser.The results and details that are retreived form the Fully Automated Diagnostic Analyser will be stored in a file in the PC for further uses.

A.2.2 Sample Run:

The following are the sample outputs got for the inputs from the software while being tested.

Enter the Sample number (or) Test Serial number :246

N;246;Krishnaraj;;M;7;ALP;242;U/l;;ALT;152;U/l;N;AST;153;U/l;N;ALB;3.7;g/dl;N; TBIL;6.4;mg/dl;Na;DBIL;5.0;mg/dl;Na;TPR;6.9;g/dl;r;

Sample number : 246
Sample name : Krishnaraj
Date of Birth : NILL
Sex : M
Number of tests : 7

Type : N
Test name : ALP
Result : 242
Unit : U/l
Falg : NILL

Type : N
Test name : ALT
Result :152
Unit : U/l
Flag : N

Type : N
Test name : AST
Result : 153
Unit : U/I
Flag : N

Type : N
Test name : ALB
Result : 3.7
Unit : g/dl
Flag : N

Type : N
Test name : TBIL
Result : 6.4
Unit : mg/dl
Flag : Na

Type : N
Test name : DBIL

Unit : mg/dl
Flag : Na

Type : N
Test name : TPR
Result : 6.9
Unit : g/dl
Flag : r

Do you want to continue? :y

Enter the Sample number (or) Test Serial number :385

N;385;Joseph;;M;6;GLU;81;mg/dl;a;UREA;28;mg/dl;;ALB;3.7;g/dl;N;TBIL;0.7;mg/dl
;a;DBIL;0.4;mg/dl;a;TPR;6.9;g/dl;;

Sample number : 385
Sample name : Joseph
Date of Birth : NILL
Sex : M
Number of tests : 6

Type : N
Test name : GLU
Result : 81
Unit : mg/dl
Flag : a

Type : N
Test name : UREA
Result : 28
Unit : mg/dl

Type : N
Test name : ALB
Result : 3.7
Unit : g/dl
Flag : N

Type : N
Test name : TBIL
Result : 0.7
Unit : mg/dl
Flag : a

Type : N
Test name : DBIL
Result : 0.4
Unit : mg/dl
Flag : a

Type : N
Test name : TPR
Result : 6.9
Unit : g/dl
Flag : NILL

Do you want to continue? :y

Enter the Sample number (or) Test Serial number :605

N;605;Sivnesan;;M;3;CHOL;149;mg/dl;a;TRIG;152;mg/dl;Na;DHDL;31.3;mg/dl;N;

Sample number : 605

Date of Birth : NILL

Sex : M

Number of tests : 3

Type : N
Test name : CHOL
Result : 149
Unit : mg/dl
Flag : a

Type : N
Test name : TRIG
Result : 152
Unit : mg/dl
Flag : Na

Type : N
Test name : DHDL
Result : 31.3
Unit : mg/dl
Flag : N

Do you want to continue? :n

Here are some of the outputs with missing and transmission errors.

Enter the Sample number (or) Test Serial number :220

Error : No such Sample Number (or) Test Serial Number found in Fully Automated Diagnostic Analyser.

Do you want to continue? :y

N;550;Poulson;;M;4;GLU;95;mg/dl;a;CHOL;186;mg/dl;a;UREA;23;mg/dl;;ALT;38;U/l;;

Sample number : 550

Sample name : Poulson

Date of Birth : NILL

Sex : M

Number of tests : 4

Type : N
Test name : GLU
Result : 95
Unit : mg/dl
Flag : a

Type : N
Test name : CHOL
Result : 186
Unit : mg/dl
Flag : a

Type : N
Test name : UREA
Result : 23
Unit : mg/dl
Flag : NILL

Type : N
Test name : ALT
Result : 38
Unit : U/I
Flag : NILL

Do you want to continue? :y

Enter the Sample number (or) Test Serial number :557

N;557;Jagadambal;;M;5;GLU;125;mg/dl;Na;CHOL;216;mg/dl;a;UREA;15;mg/dl;;ALP;116;U/l;;ALT;18;U/l;;

Sample number : 557

Sample name : Jagadambal

Date of Birth : NILL

Sex : M

Number of tests : 5

Type : N
Test name : GLU
Result : 125
Unit : mg/dl
Flag : Na

Type : N
Test name : CHOL
Result : 216
Unit : mg/dl
Flag : a

Type : N
Test name : UREA
Result : 15
Unit : mg/dl
Flag : NILL

Type : N
Test name : ALP

Unit : U/I
Flag : NILL

Type : N
Test name : ALT
Result : 18
Unit : U/I
Flag : NILL

Do you want to continue? :y

Enter the Sample number (or) Test Serial number :385

Error : Transmission Error Please Wait

N;385;Joseph;;M;6;GLU;81;mg/dl;a;UREA;28;mg/dl;;ALB;3.7;g/dl;N;TBIL;0.7;mg/dl
;a;DBIL;0.4;mg/dl;a;TPR;6.9;g/dl;;

Sample number : 385
Sample name : Joseph
Date of Birth : NILL
Sex : M
Number of tests : 6

Type : N
Test name : GLU
Result : 81
Unit : mg/dl
Flag : a

Type : N
Test name : UREA

Unit : mg/dl
Flag : NILL

Type : N
Test name : ALB
Result : 3.7
Unit : g/dl
Flag : N

Type : N
Test name : TBIL
Result : 0.7
Unit : mg/dl
Flag : a

Type : N
Test name : DBIL
Result : 0.4
Unit : mg/dl
Flag : a

Type : N
Test name : TPR
Result : 6.9
Unit : g/dl
Flag : NILL

Do you want to continue? :y

Enter the Sample number (or) Test Serial number :662

N;662;Savitha;;F;6;CHOL;154;mg/dl;a;UREA;16;mg/dl;;ALP;166;U/l;;ALT;9;U/l;;Ca

Sample number : 662

Sample name : Savitha

Date of Birth : NILL

Sex : F

Number of tests : 6

Type : N
Test name : CHOL
Result : 154
Unit : mg/dl
Flag : a

Type : N
Test name : UREA
Result : 16
Unit : mg/dl
Flag : NILL

Type : N
Test name : ALP
Result : 166
Unit : U/I
Flag : NILL

Type : N
Test name : ALT
Result : 9
Unit : U/I
Flag : NILL

Type : N

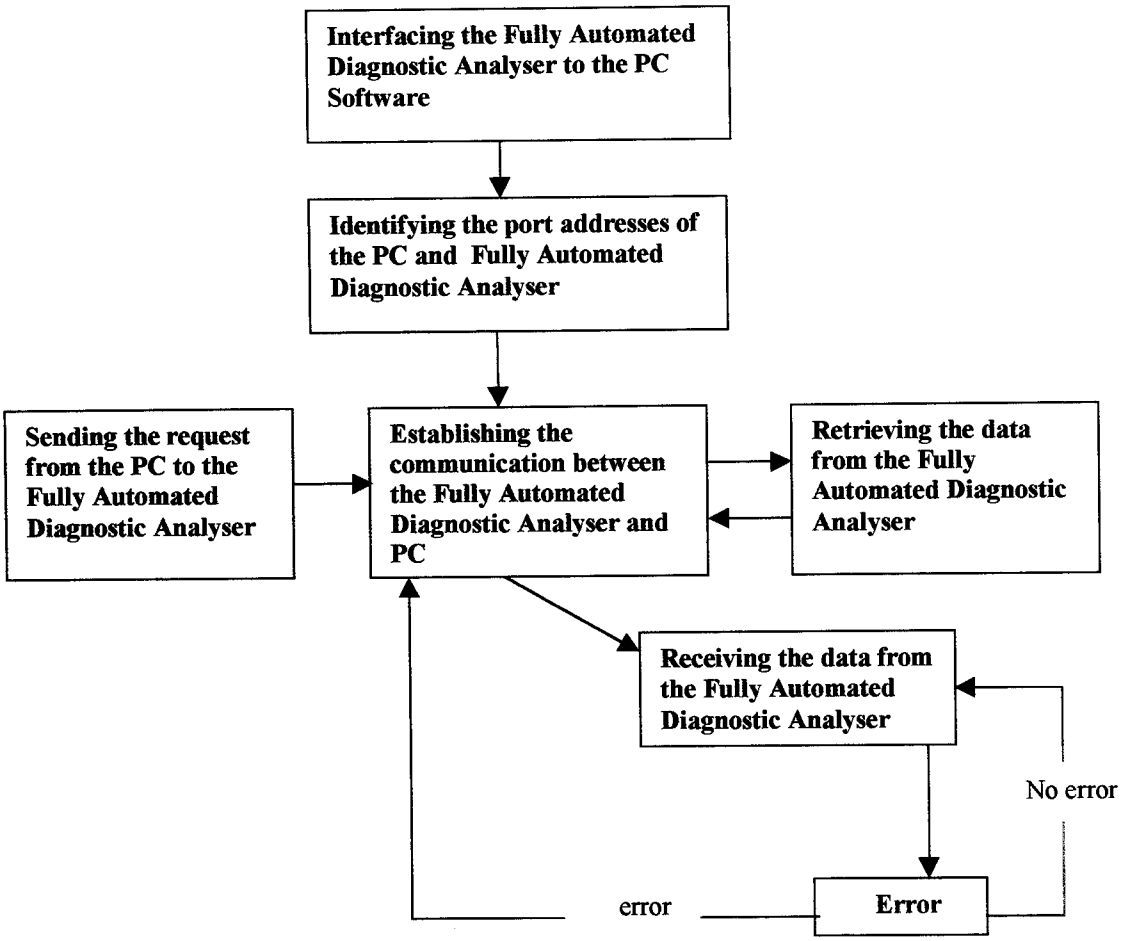
Result : 10.9
Unit : mg/dl
Flag : NILL
Type : N
Test name : Phos
Result : 3.8
Unit : mg/dl
Flag : NILL

Do you want to continue? :n

A.3 MODES OF OPERATION

The " Interfacing the Fully Automated Diagnostic Analyser", comprises of six modules. They are

- Identifying the Port addresses of PC and Fully Automated Diagnostic Analyser
- Establishing the connection between the Fully Automated Diagnostic Analyser and the Personal Computer(PC)
- Sending the request from the PC to the Fully Automated Diagnostic Analyser
- Retrieving the data from the Fully Automated Diagnostic Analyser
- Receiving the data from the Fully Automated Diagnostic Analyser
- Error Checking.



There are two modes of operation they are

- Keyboard
- Mouse

Keyboard:

With the help of the keyboard the user can select the menu option from the menu interface in the Laboratory management software for executing the software "Interfacing the Fully Automated Diagnostic Analyser" and can select the icon in the screen of the Fully Automated Diagnostic Analyser to run the software in the equipment.And also can

Mouse:

The user can select the menu option from the menu interface in the Laboratory management software for executing the software "Interfacing the Fully Automated Diagnostic Analyser" with the help of the left mouse button and can also select the icon in the screen of the Fully Automated Diagnostic Analyser to run the software in the equipment.

A.4 COMMAND SYNTAX AND SYSTEM OPTIONS:

The organization at present has a software that deals with laboratory management such as it deals with billing and management details. The organization will convert this software "Interfacing the Fully Automated Diagnostic Analyser to the PC" into executable format and links this software to the menu option provided in the Laboratory management software.

In the Fully Automated Diagnostic Analyser the software "Interfacing the Fully Automated Diagnostic Analyser to the PC" will be loaded to the location as per the rules provided for the equipment. Once loaded and the settings are done for the equipment a small icon will be displayed at the bottom end of the screen. So that when the user clicks this icon the software "Interfacing the Fully Automated Diagnostic Analyser to the PC" will run in the Fully Automated Diagnostic Analyser.