PATIENT ID CARD

PROJECT WORK DONE AT

KUMARAGURU COLLEGE OF TECHNOLOGY

PROJECT REPORT

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENT

FOR THE AWARD OF

BACHELOR OF ENGINEERING DEGREE IN INFORMATION TECHNOLOGY

OF BHARATHIAR UNIVERSITY, COIMBATORE

SUBMITTED BY

RAMYA VENKATACHALAM

0027S0096

B. SARANYA

0027S00102

GUIDED BY

Ms. S. DEVAKI B.E., M.S., ASSISTANT PROFESSOR

Department of Computer Science and Engineering

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006

MARCH 2004

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006

CERTIFICATE

This is to certify that the project entitled

"PATIENT ID CARD"

is the bonafide work of

RAMYA VENKATACHALAM
0027S0096

B. SARANYA
0027S00102

and submitted in partial fulfillment of the requirement for the award of Bachelor of Engineering Degree in Information Technology of Bharathiar university, Coimbatore

Head of the department
(Dr. S. THANGASAMY)

Guide
(Ms. S.DEVAKI)

Submitted for University Examination held on 26.3.2008

Internal Examiner

External Examiner

DECLARATION

DECLARATION

We,

Ramya Venkatachalam 0027S0096
B. Saranya 0027S0102

declare that, this project titled "Patient ID Card" was done by us and to the best of our knowledge, so far, a similar work has not been submitted to the Bharathiar University or any other institution, for fulfillment of the requirement for the award of Bachelor of Engineering Degree in Information Technology.

This report is submitted on the partial fulfillment of the requirements for the award of degree of Bachelor of Information Technology of Bharathiar University.

Place: COIMBATORE
Date: 22-03-2004

Ramya
Ramya Venkatachalam

B. Saranya
B. Saranya

Dedicated to

our Beloved Parents

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

My heartfelt thanks to **Dr K.K.Padmanaban B.Sc.,(Engg), M.Tech,Ph.D,** Principal, **Kumaraguru College of Technology, Coimbatore** for having granted permission to undertake this project work.

I wish to extend my sincere gratitude to **Dr S.Thangaswamy Ph.D.,** Head of the **Department of Computer Science & Engineering, Kumaraguru College of Technology, Coimbatore** for the interest and support he gave us during this project.

My deepest gratitude goes to our project guide, **Ms S.Devaki, BE, MS, Assistant Professor, Department of Computer Science & Engineering,** for the valuable guidance, encouragement and support she gave us throughout the course of this project.

Our sincere thanks to our class advisor **Ms P. Sudha, BE, Lecturer, Department of Information Technology** for the encouragement and support throughout the phases of our project.

We also express our thanks to all other faculty members and technical supporting staff of the Department of Computer Science and Engineering and the Department of Information Technology for their support.

Last but not the least, we would like to thank our parents and all our friends for their enthusiasm and encouragement without whom this project wouldn't have been a success.

SYNOPSIS

SYNOPSIS

The medical world is full of information. If this information does not get to the doctor in time it may lead to serious consequences. This project is mainly to bring the information about a patient to the doctors hands very easily so that the doctor can think of further treatment.

The purpose of our project entitled 'PATIENT ID CARD' is to develop a patient identification card using smart card which will contain the details of a patient which can be used instead of the regular case sheet.

The card contains details of the patient such as

- Personal details
- Medical history
- Family history
- Immunization and investigations

All the above information will be stored in the card. This is useful because it provides a quick reference to the patient's health condition in case of emergency situations. The data can be updated and stored in the card.

This card relieves the manual work involved in updating and verification of the patient details and also the discrepancies, which may occur during updation.

CONTENTS

CONTENTS

INTRODUCTION

1. INTRODUCTION

The purpose of the project is to develop a "Patient ID card" which helps the doctor to view and update patient details, it also helps the patient as it is easily portable.

1.1 EXISTING SYSTEM AND ITS LIMITATIONS

The approach that is being currently used by doctors is manual entry of patient details, test results and other investigations. The doctor prepares a case sheet for the patient, which the patient has to carry with him when he goes to meet the doctor. This method is suitable when the patient is under normal condition. Under normal circumstances the patient who goes for the periodical checkups to the doctor makes it a point to carry the case sheets.

This kind of approach fails under abnormal conditions. The patient is required to carry his case sheets always. In case of any emergency the patient is just rushed to the hospital and time should not be wasted in looking for the case sheets.

The patient is then made to undergo a routine cycle of tests in order to identify the cause of abnormality as well as to think of further course of treatment. Due to all these tests the process of treatment will be further delayed.

In the existing system as the patient's medical details are not known it is likely that the patient may be put up in wrong medication and also allowed to undergo tests which will not be required.

1.2 PROPOSED SYSTEM AND ITS ADVANTAGES

The equipment developed contains a smart card in which the details of the patient are to be stored. These cards are handy and portable and can be carried by the patient with ease.

Each and every patient is given a card containing his latest medical information. The card also provides the doctors a brief summary of the present medical condition of the patient.

The card cannot be tampered, as it is password protected. The card contains a unique identification number and it contains the information filed in the case sheets. The card can be used to store the personal medical details of the patient and the results of the various tests undergone.

The card thus provides a summary of the useful information maintained for a patient. This information would be handy in emergency conditions. The patient is also relieved from the burden of carrying the case sheets at all times.

The doctor is also relieved from writing long case sheets. The doctor is provided a user-friendly interface for entering the patient details. The details can also be updated and the doctor also can view them as well.

The GUI based environment supports ease of use and understand ability to every user, even to those unexposed to any computer application.

The proposed system is user-interactive, and at the same time, guarantees a reliable and consistent access to information on the card. This is a cost effective solution and at the same time does not required trained personal for entering data on the card.

SYSTEM REQUIREMENTS ANALYSIS

2. SYSTEM REQUIREMENT ANALYSIS

2.1 PRODUCT DEFINITION

The micro controller is a programmable single-chip integrated circuit (IC) that controls the various operations of a system. It plays a significant role in the everyday functioning of industrialized societies. It can be viewed as a data processing unit or a computing unit of a computer. This, embedded into a system meets special requirements such as cost effectiveness, low power and a high level of system integration.

Our "**Patient ID card**" provides the doctor with up-to-date information about the patient on reading the card and relieves the patient from carrying the case sheets at all times. The Patient ID card provides portability and also provides authentication for the data.

As applications become more demanding there is a need to have a powerful hardware to churn-out the information fast. It becomes important for the hardware and software to process the information as fast as possible. In our project the RS-232 is used for the purpose of communicating with the PC.

The entire working of this system can be divided into two parts the software part and the hardware part. Assembly level language is used for developing the software part of the project and the hardware part comprises the necessary electronic components and circuit boards. It also has a very interactive, user-friendly interface.

2.2 PROJECT PLAN

The current system of processing is done manually and therefore to reduce the burden of manual labour, to monitor the patient's health, to record the details and to update the details of the patient, a system has to be developed.

To overcome these difficulties and to enhance the capabilities, we propose to develop a Micro Controller Unit for writing data into the card and for reading the data from the card. The system also has a user-friendly front end and a powerful back end for computerizing the entire procedure. We use VISUAL BASIC 6.0 as front end and MS Access as back end apart from storing the data on the card.

2.3 SOFTWARE TOOLS REQUIRED

MPLAB :

MPLAB is a windows-based Integrated Development Environment(IDE) for the Microchip Technology Incorporated PIC Microcontroller families.

MPLAB allows you to write, debug, and Optimise PIC micro applications for firmware product designs.

MPLAB tools allows us to:

- Assemble, Compile and Link source code
- Debug the executable logic by watching the program flow with the simulator, or in real time with the MPLAB-ICE emulator
- Making timing measurements
- View variables in watch windows
- Program Firmware with PICSTART Plus or PRO MATE II

MPLAB is an easy-to-learn and use Integrated Development Environment (IDE).The IDE provides firmware development engineers the flexibility to develop and debug firmware for Microchip's PICmicro Microcontroller families. The MPLAB IDE runs under Microsoft Windows 3.1x,Windows 95,98,NT, and 2000.

MPLAB provides functions that allows you to

- Create and edit source files
- Create files into projects
- Debug source code
- Debug Executable Logic using the Simulator or Emulators

ASSEMBLY CODE :

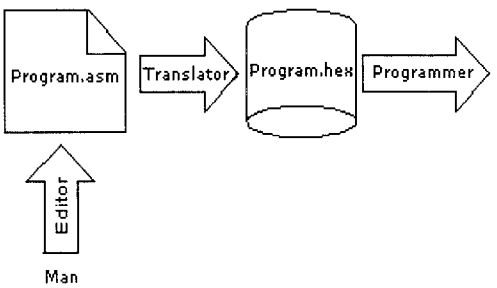
The ability to communicate is of great importance in any field. However, it is only possible if both communication partners know the same language, i.e follow the same rules during communication. Using these principles as a starting point, we can also define communication that occurs between microcontrollers and man . Language that microcontroller and man use to communicate is called "assembly language". An Assembly language is capable of producing an ASCII file.

Basic elements of assembly language are:

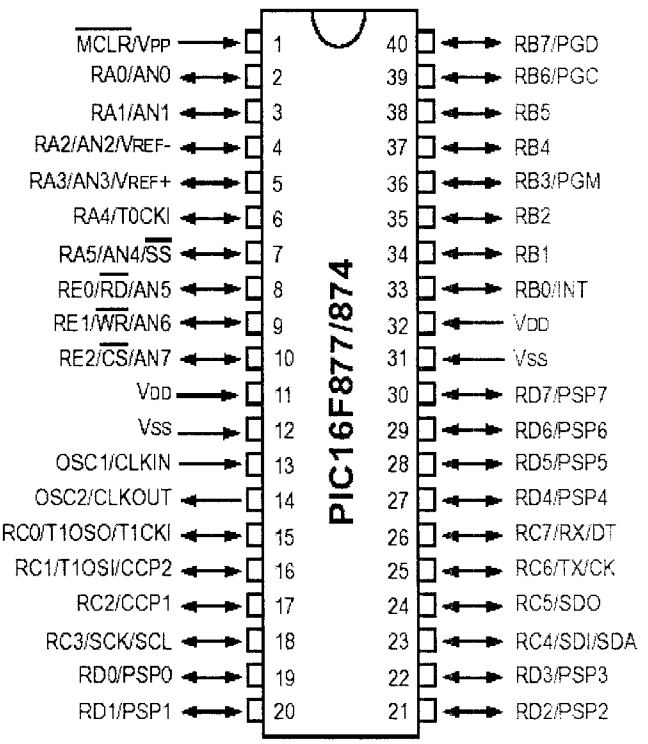
- Labels
- Instructions
- Operands
- Directives
- Comments

As a result of the process of translating a program written in assembler language we get files like:

- Program errors file (Program_Name.ERR)
- List file (Program_Name.LST)



PDIP



INSTRUCTION SET

Each midrange instruction is a 14-bit word divided into an OPCODE which specifies the instruction type and one or more operands which further specify the operation of the instruction.

The midrange instruction set summary lists the instructions recognized by the MPASM assembler. The instruction set is highly orthogonal and is grouped into three basic categories:

Byte-oriented operations
Bit-oriented operations
Literal and control operations

For byte-oriented instructions, 'f' represents a file designator and 'd' represents a destination designator. If 'd' is zero, the result is placed in the W register. If 'd' is one ,the result is placed in the file register.

For bit-oriented instructions, 'b' represents a bit field designator which selects the number of the bit affected by the operation, while 'f' represents the number of the file in which the bit is located.

For literal and control operations , 'k' represents an eight or eleven bit constant or literal value.

VISUAL BASIC 6.0 – THE FRONT END TOOL

Visual Basic has been chosen as it provides a GUI based environment for creating user-friendly forms. Visual Basic is an ideal programming language for developing sophisticated applications in windows platform. Rather than writing numerous lines of code to describe the appearance and location of interface elements, you simply add prebuilt objects into place on the screen. The 'Basic' part refers to the BASIC language.

It makes use of Graphical User Interface (GUI) for creating robust and

and easier way. In a GUI environment coding is simpler to linear programming methods and it is highly interactive and user-friendly. One of the interesting features of Visual Basic is the Integrated Development Environment (IDE). Another important feature of Visual Basic is that it has easy methods to allow users to control and access databases like MS Access, Oracle, etc., The front end can also be connected to the databases via ODBC, JDBC, etc.,

MICROSOFT ACCESS - THE BACK-END TOOL

MS ACCESS is a powerful mutli-user DBMS used to store and manipulate large amount of information and automate repetitive tasks, such as maintaining and generating invoices. Data in access is organized in the form of tables within a table, records are arranged according to a common reference value, known as the primary key field. The key value is unique and helps in identifying the records. A combination of two or more fields can also be used as primary key.

Databases in access have the default extension (.MDB). Access also maintains index files for tables. Index object provides access to data.

SOFTWARE REQUIREMENT SPECIFICATION

3. SOFTWARE REQUIREMENT SPECIFICATION

3.1. INTRODUCTION

3.1.1. Purpose

The purpose of this project 'Patient ID card' is to develop a card, which has the details of the patient stored in it. Each patient is given a card and a unique identification number is used to identify a patient. This card helps the medical practitioner in case of emergency treatment of the patient and hence helps in accessing the details of a patient without much stress.

3.1.2. Scope

The card is useful for doctors who can get a detailed idea of the previous medical history of a patient so that the future course of treatment can be decided by taking into consideration the previous medical background of the patient.

3.1.3. Definitions, Acronyms, Abbreviations

Definition

Smart Card : A smart card is a credit card sized plastic card with an integrated circuit (computer chip) embedded in it. This chip stores electronic data and programs that are protected by advanced security features providing data portability, security and convenience.

Microcontroller : A microcontroller is a general purpose device which has inbuilt CPU, memory and peripherals to make it as a mini computer.

The PIC Chip : Microchip Technology's series of microcontrollers is called PIC chips.

ACRONYMS AND ABBREVIATIONS

PIC	: Programmable Interface Controller
USART	: Universal Synchronous Asynchronous Receiver Transmitter
I2C	: Inter – Integrated Circuit

3.1.4. Overview

The 'Patient ID card' developed helps the doctor and the patient. This card is mainly used for storing the patient's medical history so that the patient can be given quick and better treatment in case of emergency. The card also helps in analysing the present medical investigations of the patient so that the doctor can decide on the treatment given to the patient. In the course of treatment, the doctor updates the data entries so that the card can supply the latest information everywhere and at all times.

3.2. GENERAL DESCRIPTION

3.2.1. Product Perspective

The main component, which is used in the development of the "Patient ID card" is the smart card with a circuit embedded in it,which contains data ,clock ,ground and positive pins , this chip will store all the data required. It will also ensure that the data in the chip is protected at

the same time providing access to data and also provides features for updating the content in the card.

3.2.2. Product Functions

The product developed will be useful in hospitals in which the patients in emergency will be handled with great care after referring the previous medical history, this benefits patients who are unable to convey their medical state by this product. This product will also contain a unique medical identification number, which will protect the information from being viewed by unauthorized people.

3.2.3. User Characteristics

The user need not be very well versed in any of the major computer languages. The user should be able to provide the unique medical identification number of the patient and also must be able to update the database by entering the required information in the forms provided.

3.2.4. General Constraints

The only constraint is that the user should enter the valid pin code and then only the user will be allowed to view or update the information in the database.

3.3. SPECIFIC REQUIREMENTS

3.3.1. Functional Requirements

3.3.1.1. Introduction

The basic use of this card is for quicker and easier retrieval of data and hence decreases the amount of paper work in hospitals. It also provides increased patient convenience and secure access to data.

3.3.1.2. List of Inputs

1) The user identification number
2) The medical information to be stored in the card

3.3.1.3. Information processing requirement

The card is interfaced with the PC using the USART protocol and I2C protocol is used to communicate between the integrated circuits.

The user on entering the unique identification number is checked for identity by verifying it with the identification number present on the card. If the number matches then the data on the card can be read, viewed or updated. Apart from storing the data on the card there is also a database which stores the details of the patient.

3.3.2. Performance Requirements

3.3.2.1. Security

The main advantage of using the smart card is that it is tamper - resistant. Information stored on the card can be PIN protected and/or read-write protected. Hence it is capable of performing data encryption. It is capable of processing (not just storing) the information. It is also possible for post-issuance updation of information.

3.3.3. Design Constraints

3.3.3.1. Hardware Limitations

Minimal requirements are:

- Pentium III processor and above.
- 64 MB RAM
- GB Hard Disk
- 104 Keys Keyboard
- PIC16F877 Microcontroller
- Transformer
- MAX 232 IC
- Seven segment LEDS
- Switch
- Transistors
- Crystal oscillator
- Active and passive components
- Smart card
- PICSTART Plus Programmer
- VDD(4.5V - 5.0V)

3.3.3.2. External Interface Requirement

The interface required to connect the PC to the microcontroller unit is the USART protocol and the I2C protocol is used for communication between the integrated circuits.

3.3.3.3. User Interfaces, Screen Formats

The user interfaces to be designed will be user-friendly so that no other professional training is required on the user part.

A user interface for prompting the patient id will be present and another form in which the details or various categories present in the medical history will be there.

3.3.3.4. Hardware Interfaces and Software Interfaces with other systems

The hardware interface to be designed provides communication between the PC and the PIC microcontroller.

The software interface provides a database which stores the required information of the patient.

The user - interface is to be designed in Visual Basic.

The MPLAB software is to be used for programming the PIC microcontroller in assembly language.

3.3.4. Other Requirements

3.3.4.1. Operations required by the user

Patient Identification number should be unique for every patient in order to maintain integrity.

DESIGN DOCUMENT

4. DESIGN DOCUMENT

4.1 INPUT DESIGN

Reliable input design ensures accurate output design from a system.The system to be designed is configured from the following data.

1. Insertion of the card in the card slot
2. Providing the correct password to gain access to the data
3. Signal from the MAX-232, the USART

4.2 PROCESS DESIGN

The user password is validated and the required processing is done for the transfer of patient details to the database as well as the smart card.

The user can also make changes in the existing information and this is once again updated in the database and card.

4.3 OUTPUT DESIGN

Outputs from the system are required to communicate the results of the processing to the users.

The abnormalities noted in the patient's health and the results of the various tests are reported as the output and any further changes are updated in the database and simultaneous in the microcontroller.

4.4 DATABASE DESIGN

The database is defined as an integral collection of data. The overall objective in the development of database technology has been to treat data as an organizational resource and as an integrated whole.
The database for the patient identification card contains:

1. Personal details of the patient
2. Medical history of the patient
3. Family history of the patient
4. Immunization and investigations

SYSTEM IMPLEMENTATION AND TESTING

5. SYSTEM IMPLEMENTATION AND TESTING

5.1 SYSTEM IMPLEMENTATION

Implementation is the stage of the project where the design is turned into a working system. If the implementation is not carefully planned and controlled, it can cause chaos.

Implementation involves careful planning analysing the current system and the constraints that might exist on the implementation, design of the procedures to achieve the changeover and training the staff to use the system.

Implementation involves the entire design of the project the circuit design initially involves the following procedures:

- The PIC Microcontroller P16F877 in which the VDD pin is connected to the positive supply of 5 Volts and the VSS pin is grounded.

- The RC7 and RC6 pins are required for achieving handshaking between the PC and the microcontroller.

- PIC chip uses TTL logic levels, but it must communicate with a PC that uses RS-232. The MAX232 chip is used to facilitate this conversion. It requires a +5V supply and four 1-uF electrolytic capacitors, and provides two channels of TTL-to-RS-232 conversion, and two channels of RS-232-to-TTL conversion.

- The PC acts as a Data Terminal Equipment (DTE) to which the Data Communications Equipment (DCE) is connected. DCE sends data on DB9 pin 2 and receives data on DB9 pin 3, while DTE receives on pin 2 and pin sends on pin 3.

- Pins 2 and 3 are the transmit and receive lines for the data, and pin 5 is the ground line.

- The pins SDA and SCL i.e., RC4 AND RC3 pins are used for providing communication between PIC microcontroller and the smart card.

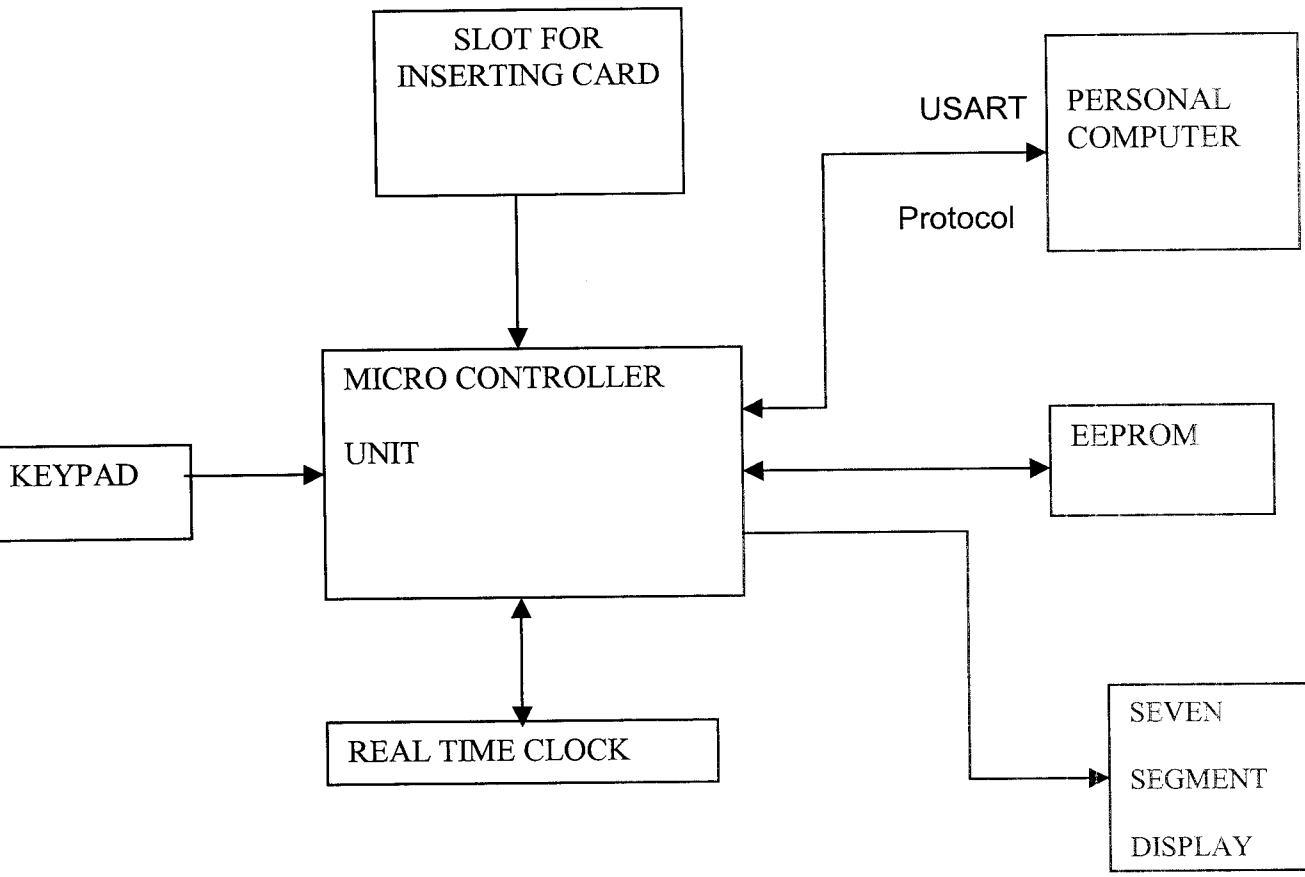
- The SDA pin is used for data transfer and SCL pin is used to provide synchronisation between PIC microcontroller and smart card.

- Once the circuit is setup the data entered is saved in the EEPROM memory of the microcontroller, which then transfers the data through the SDA pin to the smart card through the data pin.

- Seven segment displays are provided in order to provide display instructions for insertion of card, entering the password and hence right to access the reception of data and transfer of data.

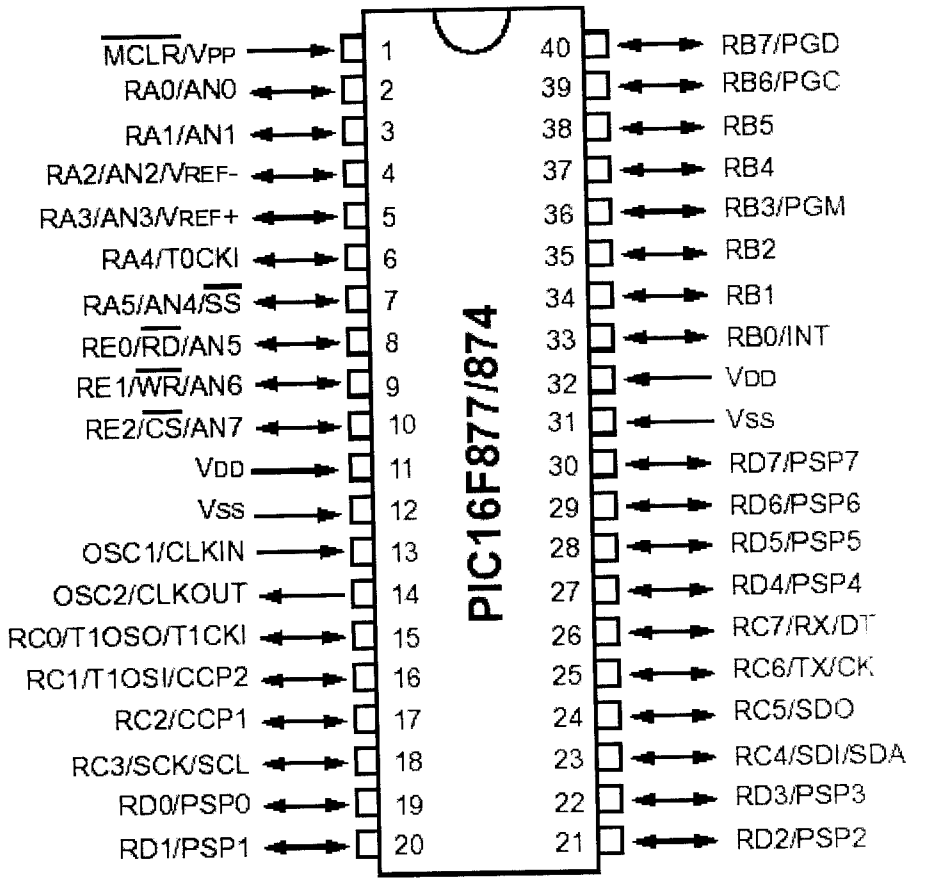
- The user interface is created using VISUAL BASIC as front end wherein the necessary data is entered and MSACCESS acts as back end. Features are provided wherein the user can enter new data, update existing data as well as view the existing data.

GENERAL BLOCK DIAGRAM

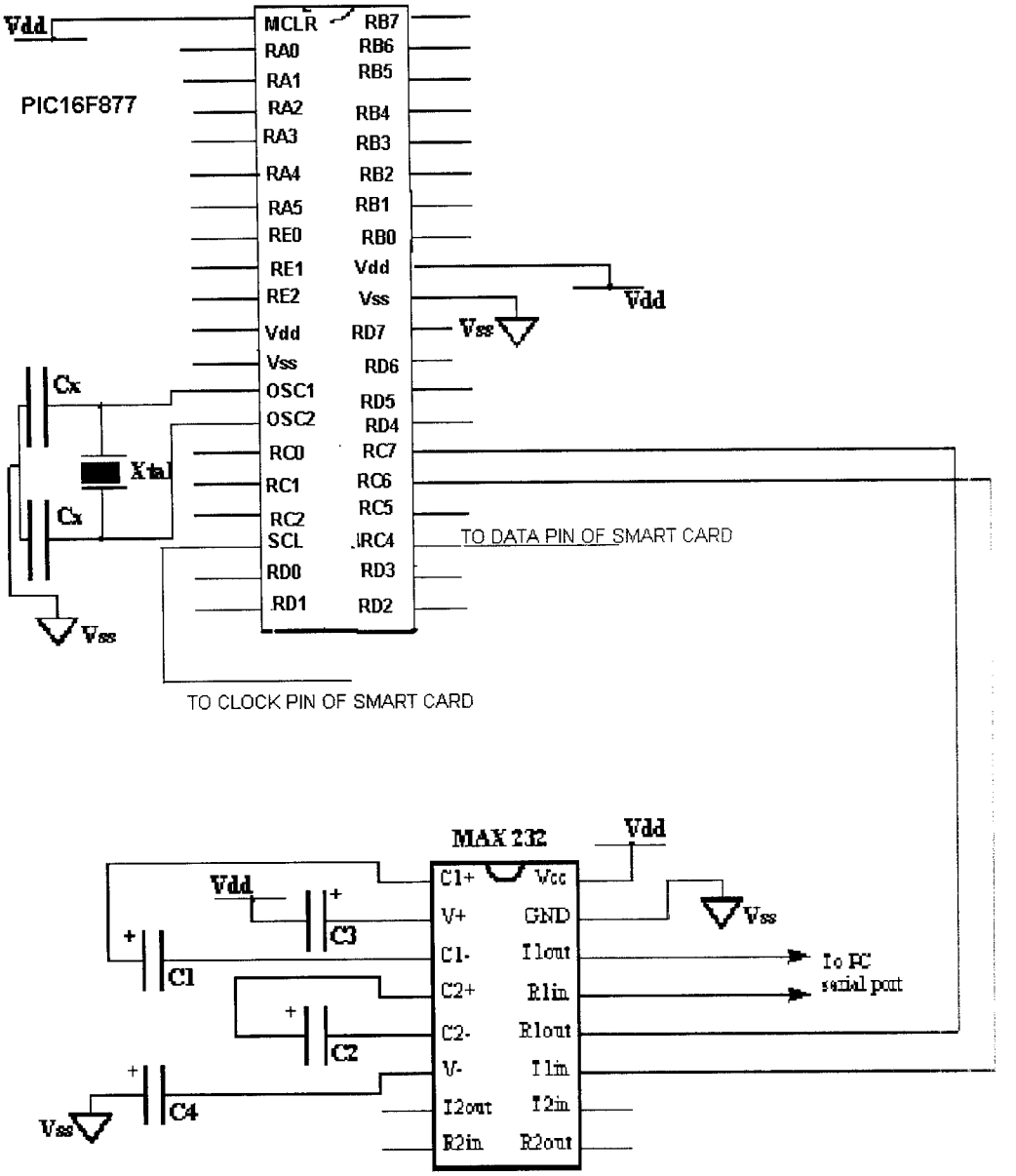


PIC 16F877 BLOCK DIAGRAM

PDIP



CIRCUIT DIAGRAM



Selecting a FLASH PICmicro Microcontroller from the mid-range family as the target MCU allows us to make the code simpler and cleaner. It also provides larger RAM memory space and a deeper hardware stack. Interrupts have been used to "virtualize" the receiving routine as a software peripheral and to free the design of the hard real time constraint that it usually poses.

The PIC16F877 Microcontroller provides amigration path fron OTP to FLASH in the 40-pin package, with a wide range of peripheral integration options. This unit features 14-bit instruction set, 5 to 8 channels of 10-bit Analog - to - Digital Converters, interrupt handling capability, various serial interface capabilities, Capture/Compare/PWM, Brown-out detection and an 8-level deep stack. The PIC16F877 provides performance and versatility to meet the most demanding requirements of today's cost-sensitive analog designs.Plus,with FLASH program memory,PIC16F877 device can be reprogrammed over the entire operating voltage range.

5.2 SYSTEM TESTING

System testing is the stage of implementation which is aimed at ensuring that the system works accurately and efficiently before the live operations commences, thus the system test in implementation should be a confirmation that all is correct and an opportunity to show the users that the system works.

BLACK BOX TESTING

The testing focuses the functional requirements of the software. It finds the errors of incorrect or missing functions, interface errors, errors in database access ,performance errors and initialization errors.

WHITE BOX TESTING

Using the white box testing all logical decisions on the true and false side of the product is tested. All the loops within the operational bunds are checked. Logical errors and incorrect assumptions are identified and rectified.

HARDWARE TESTING

This testing involves the check for the alignment and functioning of the various components being used. The alignment check involves whether correct connection is established and functioning check involves the check for the working of the different components.

FUTURE ENHANCEMENTS

6. FUTURE ENHANCEMENTS

This product can be further enhanced to contain various other details and can be used for several patients, using a smart card for each of the patients and hence helps in computerising the functioning of the entire hospital. Apart from the medical history the same card can be used for general purposes like online banking, shopping and can be made acceptable by the all people by making it more economical.

And security features can be further enhanced and the card can be made more tamper-resistant.

CONCLUSION

7. CONCLUSION

This project is a useful system for the patients which can ensure better facility and faster treatment to the patients and lesser strain to both the doctors and the patients. This also eliminates the manual work needed to record the patient health details and hence avoids the discrepancies involved in it.

The GUI based environment is aimed to provide ease of operability and interactive ness. Since the entire operation is Microcontroller based any medication is achieved through the software the hardware remains untouched.

REFERENCES

8. REFERENCES

BOOKS

- John Peatman, "Design with PIC microcontrollers",Prentice Hall of India Private Limited, 1998.
- Jan Axelson, "Serial Port Complete", Penarm International Public India Limited.
- Michael Halvorson, "Learn Microsoft Visual Basic 6.0 Now", Prentice Hall of India Private Limited, Second Edition, 2000.
- Michael Smith, "Visual Basic 5.0 Super Bible Volume –1", Prentice Hall of India Private Limited, Third Edition, 2001.
- PIC Microcontroller Reference Manual
- PIC16F877 datasheet

Websites

- www.microchip.com
- www.piclist.com
- www.embedded.com
- www.smartchip.com
- www.maxking.com

APPENDIX

9. APPENDICES

9.1 Sample Code

```
Option Explicit
Dim med, med1, fam, imm, imm1 As String
Dim txt, a, b, b1, b2, b3, b4, c, temp1, c1, c2, c3, c4, c5, c6, c7
Dim d5, d6, d7, e1, e2, e3, e4, e5, e6, e7, e8
Dim f1, f2, f3, f4, f5, f6, f7, f8, g6, g7, g8
Dim k, k1 As Integer
   Dim data(40) As Byte
Dim timerflag As Boolean, SecTimerFlag As Boolean "**

Private Sub CmdClear_Click()
   CmbBlood.Text = ""
   CmbRh.Text = ""
   CmbHep.Text = "NO"
   CmbBcg.Text = "YES"
   CmbDpt.Text = "YES"
   CmbOpv.Text = "YES"
   CmbMeasles.Text = "YES"
   CmbMmr.Text = "YES"
   CmbHepa.Text = "YES"
   CmbHepb.Text = "YES"
   CmbInf.Text = "YES"
   CmbChick.Text = "YES"
   CmbThy.Text = "YES"
End Sub
Private Sub CmdExit_Click()
   End
End Sub
Private Sub CmdSave_Click()
```

```
'Patient details
rs.Fields(0) = PatientDet.TxtId.Text
rs.Fields(1) = PatientDet.TxtName.Text
rs.Fields(2) = PatientDet.TxtAddr.Text
rs.Fields(3) = PatientDet.TxtNationality.Text
rs.Fields(4) = PatientDet.TxtStd.Text + PatientDet.TxtNo.Text
rs.Fields(5) = PatientDet.Cbogen.Text
rs.Fields(6) = PatientDet.TxtDob.Text
rs.Fields(7) = PatientDet.Cbomart.Text
rs.Fields(8) = PatientDet.TxtOccupation.Text
rs.Fields(9) = PatientDet.TxtEmerPer.Text
rs.Fields(10) = PatientDet.TxtEmerCode.Text + PatientDet.TxtEmerNo.Text
' If MSComm1.PortOpen = True Then
    'temp = PatientDet.TxtId.Text
  ' MSComm1.Output = MSComm1.Output + temp
    'temp1 = PatientDet.TxtName.Text
'   MSComm1.Output = MSComm1.Output + temp1
' End If

 'Medical Info
 rs.Fields(11) = PersonalInfo.TxtPhysician.Text
 rs.Fields(12) = PersonalInfo.TxtPhyAddr.Text
 rs.Fields(13) = PersonalInfo.TxtCode.Text + PersonalInfo.TxtNumb.Text
 rs.Fields(14) = PersonalInfo.TxtHeight.Text
 rs.Fields(15) = PersonalInfo.TxtWeight.Text
 rs.Fields(16) = PersonalInfo.TxtBp.Text
 rs.Fields(17) = PersonalInfo.TxtMedication.Text
 rs.Fields(18) = PersonalInfo.TxtAllergies.Text
 rs.Fields(19) = PersonalInfo.TxtSurgery.Text
 rs.Fields(20) = PersonalInfo.TxtHabits.Text
 'If MSComm1.PortOpen = True Then
```

```
'End If

"""""Medical History

rs.Fields(21) = MedicalHistory.Cbodia.Text
rs.Fields(22) = MedicalHistory.Cbohyp.Text
rs.Fields(23) = MedicalHistory.Cboheart.Text
rs.Fields(24) = MedicalHistory.Cbotb.Text
rs.Fields(25) = MedicalHistory.Cboasth.Text
rs.Fields(26) = MedicalHistory.Cbobld.Text
rs.Fields(27) = MedicalHistory.Cbokid.Text
rs.Fields(28) = MedicalHistory.Cboneuro.Text
rs.Fields(29) = MedicalHistory.Cbopsy.Text
rs.Fields(30) = MedicalHistory.Cboliver.Text
rs.Fields(31) = MedicalHistory.Cbothyroid.Text
'If MSComm1.PortOpen = True Then


If MedicalHistory.Cbodia.Text = "ABNORMAL" Then
        med = "1"
Else
        med = "0"
End If
If MedicalHistory.Cbohyp.Text = "ABNORMAL" Then
        med = med & "1"
Else
        med = med & "0"
End If
If MedicalHistory.Cboheart.Text = "HIGH" Then
        med = med & "1"
Else
        med = med & "0"
```

```
        med = med & "1"
Else
        med = med & "0"
End If
If MedicalHistory.Cboasth.Text = "HIGH" Then
        med = med & "1"
Else
        med = med & "0"
End If
If MedicalHistory.Cbobld.Text = "HIGH" Then
        med = med & "1"
Else
        med = med & "0"
End If
If MedicalHistory.Cbokid.Text = "HIGH" Then
        med = med & "1"

Else
        med = med & "0"
End If
If MedicalHistory.Cboneuro.Text = "HIGH" Then
        med = med & "1"
Else
        med = med & "0"
End If

med = med
Text3.Text = med
If MedicalHistory.Cbopsy.Text = "HIGH" Then
        med1 = "1"
Else
        med1 = "0"
```

```
        med1 = med1 & "1"
Else
        med1 = med1 & "0"
End If
If MedicalHistory.Cbothyroid.Text = "HIGH" Then
        med1 = med1 & "1"
Else
        med1 = med1 & "0"
End If
med1 = "00000" & med1
Text4.Text = med1

"""Family medical History

rs.Fields(32) = FamilyHist.Cboasth.Text
rs.Fields(33) = FamilyHist.Cbobp.Text
rs.Fields(34) = FamilyHist.Cbodia.Text
rs.Fields(35) = FamilyHist.Cboheart.Text
rs.Fields(36) = FamilyHist.Cbotb.Text
rs.Fields(37) = FamilyHist.Cbocan.Text
rs.Fields(38) = FamilyHist.Cbostrok.Text
rs.Fields(39) = FamilyHist.Cboalergy.Text

If FamilyHist.Cboasth.Text = "YES" Then
   fam = "1"
Else
   fam = "0"
End If
If FamilyHist.Cbobp.Text = "YES" Then
        fam = fam & "1"
Else
        fam = fam & "0"
```

```
        fam = fam & "1"
    Else
        fam = fam & "0"
    End If
    If FamilyHist.Cboheart.Text = "YES" Then
        fam = fam & "1"
    Else
        fam = fam & "0"
    End If
    If FamilyHist.Cbotb.Text = "YES" Then
        fam = fam & "1"
    Else
        fam = fam & "0"
    End If
    If FamilyHist.Cbocan.Text = "YES" Then
        fam = fam & "1"
    Else
        fam = fam & "0"
    End If
    If FamilyHist.Cbostrok.Text = "YES" Then
        fam = fam & "1"
    Else
        fam = fam & "0"
    End If

    If FamilyHist.Cboalergy.Text = "YES" Then
        fam = fam & "1"
    Else
        fam = fam & "0"
    End If
    fam = fam
    Text1.Text = fam
```

```
'-------------------------------------------------'
 'Immunisation

  rs.Fields(40) = CmbBlood.Text
  rs.Fields(41) = CmbRh.Text
  rs.Fields(42) = CmbHep.Text
  rs.Fields(43) = CmbBcg.Text
  rs.Fields(44) = CmbDpt.Text
  rs.Fields(45) = CmbOpv.Text
  rs.Fields(46) = CmbMeasles.Text
  rs.Fields(47) = CmbMmr.Text
  rs.Fields(48) = CmbHepa.Text
  rs.Fields(49) = CmbHepb.Text
  rs.Fields(50) = CmbInf.Text
  rs.Fields(51) = CmbChick.Text
  rs.Fields(52) = CmbThy.Text
  rs.Update
If CmbHep.Text = "YES" Then
  imm = "1"
Else
  imm = "0"
End If
If CmbBcg.Text = "NO" Then
  imm = imm & "0"
Else
  imm = imm & "1"
End If
If CmbDpt.Text = "NO" Then
  imm = imm & "0"
Else
  imm = imm & "1"
End If
If CmbOpv.Text = "NO" Then
```

```
Else
   imm = imm & "1"
End If
If CmbMeasles.Text = "NO" Then
   imm = imm & "0"
Else
   imm = imm & "1"
End If
If CmbMmr.Text = "NO" Then
   imm = imm & "0"
Else
   imm = imm & "1"
End If
If CmbHepa.Text = "NO" Then
   imm = imm & "0"
Else
   imm = imm & "1"
End If
If CmbHepb.Text = "NO" Then
   imm = imm & "0"
Else
   imm = imm & "1"
End If
imm = imm
Text2.Text = imm

If CmbInf.Text = "NO" Then
   imm1 = imm1 & "0"
Else
   imm1 = imm1 & "1"
End If
If CmbChick.Text = "NO" Then
```

```
    imm1 = imm1 & "1"
End If
If CmbThy.Text = "NO" Then
    imm1 = imm1 & "0"
Else
    imm1 = imm1 & "1"
End If
imm1 = "11111" & imm1
Text5.Text = imm1
If MSComm1.PortOpen = False Then
    MSComm1.InputLen = 0
    MSComm1.PortOpen = True
    " Included on 26-feb
    " 4-Mar Start
       'MSComm1.Output = med '+ med1 + fam + imm + imm1 & Chr$(13)
    " 4-Mar end
       Timer1.Enabled = True
    ''''''''''''
End If
End Sub

Private Sub CmdView_Click()
   ' MSComm1.PortOpen = False
    'Timer3.Enabled = True
    Viewopt.Show
End Sub


Private Sub Form_Load()
    enter
    FamilyHist.Hide
    med = "0"
    med1 = "0"
```

```
" Included on 26-feb
    Timer1.Enabled = False
    Timer2.Enabled = False  "**
    Timer3.Enabled = False  "**
    Timer4.Enabled = False  "**
    timerflag = False       "**
    SecTimerFlag = False
 ''''''''''''''''''''
 " 4-Mar End

 'rs.Open "Immunization", cn, adOpenDynamic, adLockOptimistic,
adCmdTable
End Sub



Private Sub Timer1_Timer()
      Static i As Integer
      Static j As Integer
      Dim temp
      Dim x As Long

      If timerflag = False Then
         If j = 0 Then temp = med
         If j = 1 Then temp = med1
         If j = 2 Then temp = fam
         If j = 3 Then temp = imm
         If j = 4 Then temp = imm1


         i = i + 1

         Mid(temp, i, 1)
```

```
    If i = 8 Then
       i = 0
       j = j + 1


       If j = 5 Then
          Timer1.Enabled = False
          j = 0
          'MsgBox "Sent all the values"
         ' Exit Sub
       Else
          Timer1.Enabled = False
          Timer2.Enabled = True
          timerflag = True
          'Exit Sub
       End If

       "For x = 1 To 200   ' for delay
       "Next

    End If
  End If

  If timerflag = True Then
     Exit Sub
  End If
End
  Load Viewopt
  Timer3.Enabled = True
  Exit Sub
```

```
Private Sub Timer2_Timer()


   Static flag As Boolean
   If flag = True Then
      timerflag = False
      Timer2.Enabled = False
      Timer1.Enabled = True
   End If
   flag = Not flag
End Sub



 Private Sub Timer3_Timer()

    Static i As Integer

    If SecTimerFlag = False Then
       If MSComm1.PortOpen = False Then
          MSComm1.PortOpen = True
       End If

       MSComm1.InputLen = 1
       If MSComm1.InBufferCount <> 0 Then
          data(i) = Val(CStr(MSComm1.Input))


          Text6.Text = Text6.Text & data(i)
          i = i + 1
          If (i Mod 8) = 0 Then
```

```
        End If
        If i = 40 Then
            Timer3.Enabled = False
            Call ViewDaTA
            i = 0
            MsgBox "TRANSMISSION OVER"
          End If

      End If
   End If


End Sub

Private Sub Timer4_Timer()
   Static flag As Boolean
   If flag = True Then
      SecTimerFlag = False
      Timer4.Enabled = False
      Timer3.Enabled = True
   End If
   flag = Not flag
End Sub


Private Sub ViewDaTA()


"" ***************MEDICAL HISTORY***************""""
      If data(0) = 1 Then
         Viewopt.TxtData.Text = "DIABETES - ABNORMAL" + vbNewLine +
vbNewLine
      End If
```

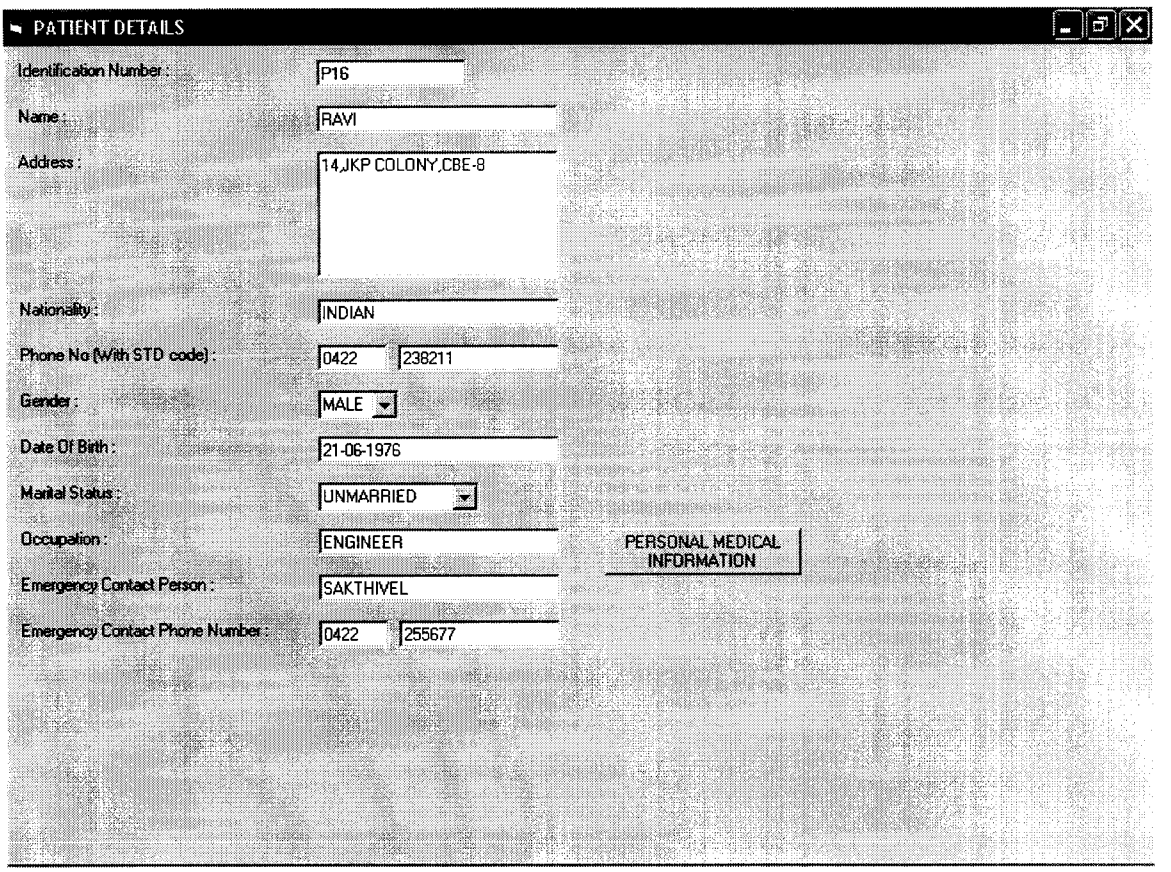
```
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "HYPERTENSION -
ABNORMAL" + vbNewLine + vbNewLine
      End If
      If data(2) = 1 Then
         Viewopt.TxtData.Text = Viewopt.TxtData.Text + "HEART DISEASE -
HIGH" + vbNewLine + vbNewLine
      End If
     If data(3) = 1 Then
         Viewopt.TxtData.Text = Viewopt.TxtData.Text + "TUBERCULOSIS -
HIGH" + vbNewLine + vbNewLine
      End If
      If data(4) = 1 Then
         Viewopt.TxtData.Text = Viewopt.TxtData.Text + "ASTHMA - HIGH" +
vbNewLine + vbNewLine + vbNewLine
      End If
      If data(5) = 1 Then
         Viewopt.TxtData.Text = Viewopt.TxtData.Text + "BLEEDING
DISORDERS - HIGH" + vbNewLine + vbNewLine
      End If
      If data(6) = 1 Then
         Viewopt.TxtData.Text = Viewopt.TxtData.Text + "KIDNEY
DISORDERS - HIGH" + vbNewLine + vbNewLine
      End If
      If data(7) = 1 Then
         Viewopt.TxtData.Text = Viewopt.TxtData.Text + "NEUROLOGIC
DISORDERS - HIGH" + vbNewLine + vbNewLine
      End If

      If data(13) = 1 Then
         Viewopt.TxtData.Text = Viewopt.TxtData.Text + "PSYCHIATRIC
DISORDER - HIGH" + vbNewLine + vbNewLine
      End If
```

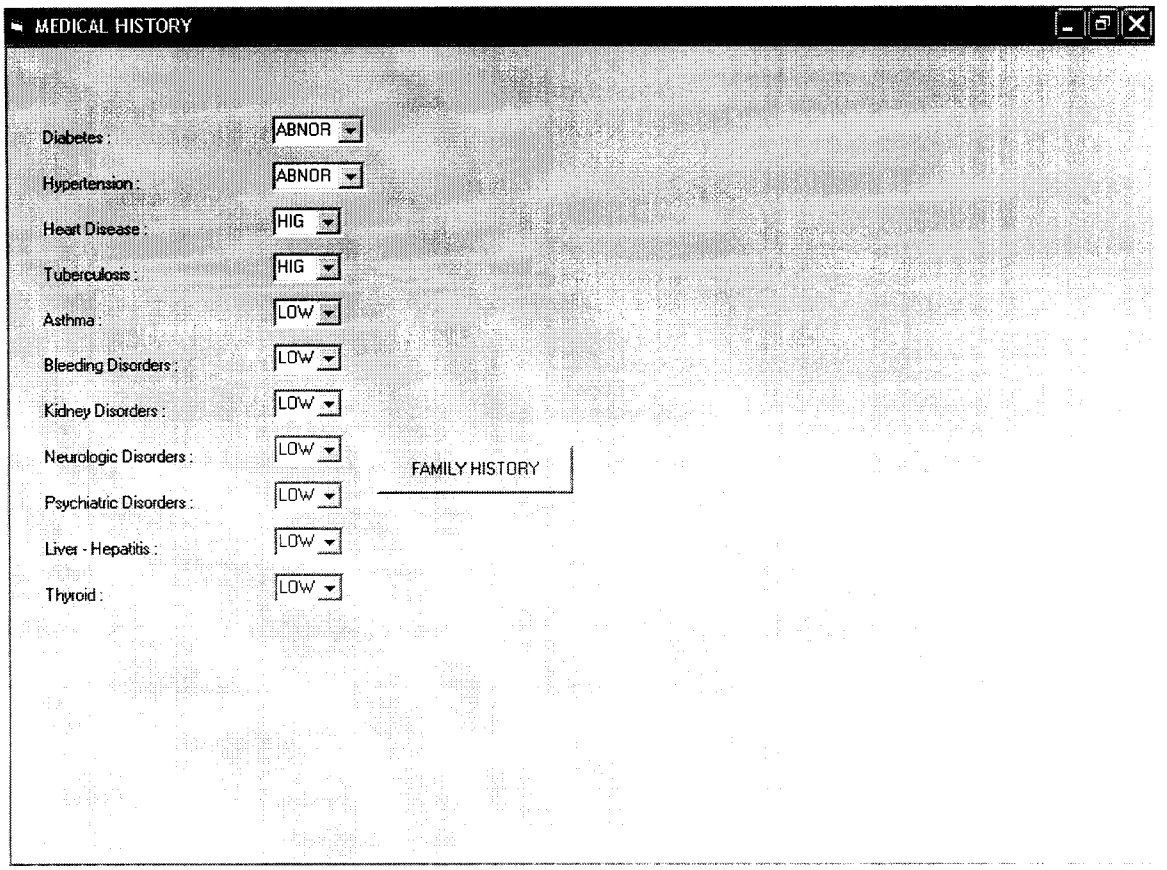
```
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "LIVER-HEPATITIS -
HIGH" + vbNewLine + vbNewLine
    End If
    If data(15) = 1 Then
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "THYROID
DISORDER - HIGH" + vbNewLine + vbNewLine
    End If

"""********************FAMILY HISTORY**********************"""

    If data(16) = 1 Then
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "ASTHMA - YES" +
vbNewLine + vbNewLine
    End If
    If data(17) = 1 Then
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "BLOODPRESSURE
- YES" + vbNewLine + vbNewLine
    End If
    If data(18) = 1 Then
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "DIABETES - YES" +
vbNewLine + vbNewLine
    End If
    If data(19) = 1 Then
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "HEART DISEASE -
YES" + vbNewLine + vbNewLine
    End If
    If data(20) = 1 Then
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "TUBERCULOSIS -
YES" + vbNewLine + vbNewLine
    End If
    If data(21) = 1 Then
        Viewopt.TxtData.Text = Viewopt.TxtData.Text + "CANCER - YES" +
vbNewLine + vbNewLine
```

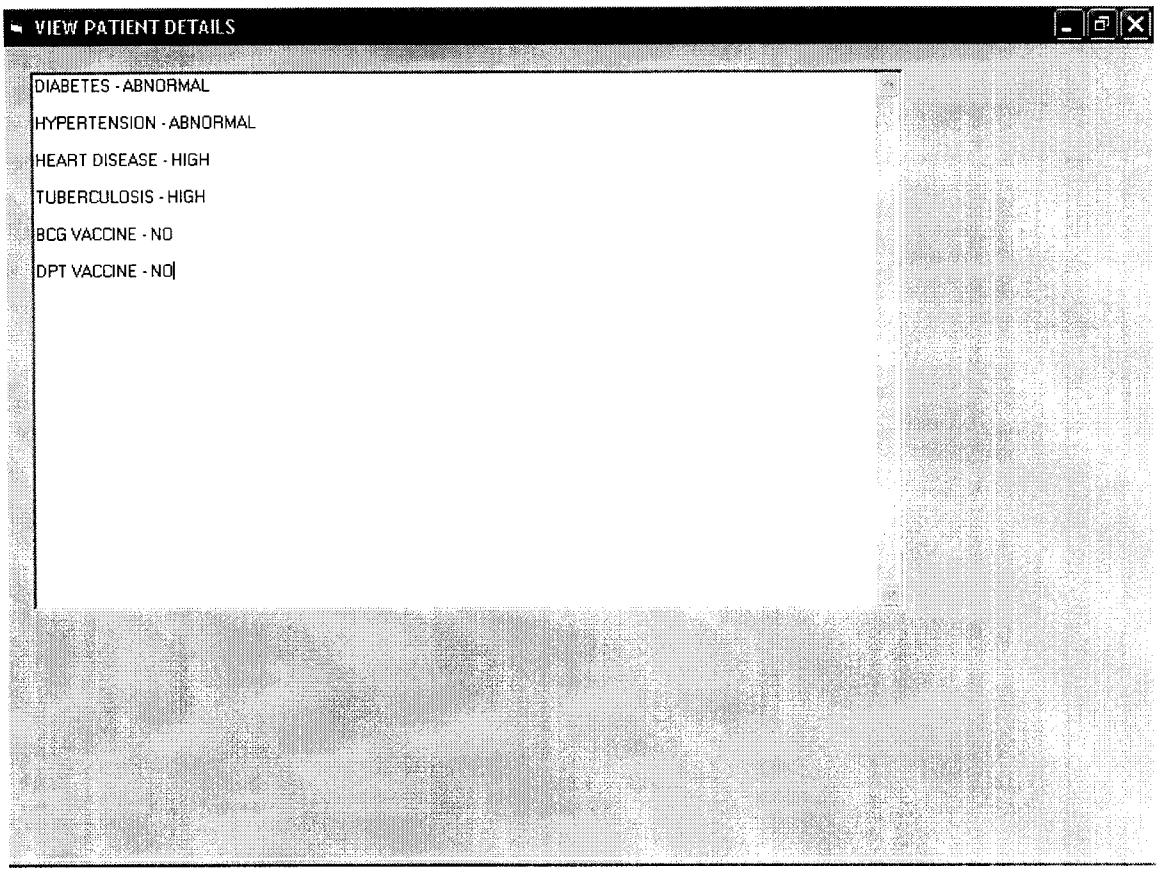
```
        If data(22) = 1 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "STROKE - YES" +
vbNewLine + vbNewLine
        End If
        If data(23) = 1 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "ALLERGIES - YES"
+ vbNewLine + vbNewLine
        End If

"""**************************IMMUNIZATION*****************"""
        If data(24) = 1 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "INVESTIGATION -
HEPATITIS-B - YES" + vbNewLine + vbNewLine
        End If
        If data(25) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "BCG VACCINE - NO
" + vbNewLine + vbNewLine
        End If
        If data(26) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "DPT VACCINE - NO"
+ vbNewLine + vbNewLine
        End If
        If data(27) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "OPV VACCINE -
NO" + vbNewLine + vbNewLine
        End If
        If data(28) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "MEASLES VACCINE
- NO" + vbNewLine + vbNewLine
        End If
        If data(29) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "MMR VACCINE -
NO" + vbNewLine + vbNewLine
```

```
        If data(30) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "HEPATITIS-A
VACCINE - NO" + vbNewLine + vbNewLine
        End If
        If data(31) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "HEPATITIS-B
VACCINE - NO" + vbNewLine + vbNewLine
        End If

        If data(37) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "H - INFLUENZA -
NO" + vbNewLine + vbNewLine
        End If
        If data(38) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "CHICKEN POX
VACCINE - NO" + vbNewLine + vbNewLine
        End If
        If data(39) = 0 Then
            Viewopt.TxtData.Text = Viewopt.TxtData.Text + "THYROID VACCINE
- NO" + vbNewLine + vbNewLine
        End If

End Sub
```

9.2 Sample Output




MEDICAL HISTORY
Diabetes :
ABNOR
Hypertension :
ABNOR
Heart Disease :
HIG
Tuberculosis :
HIG
Asthma :
LOW
Bleeding Disorders :
LOW
Kidney Disorders :
LOW
Neurologic Disorders :
LOW
Psychiatric Disorders :
LOW
Liver - Hepatitis :
LOW
Thyroid :
LOW
FAMILY HISTORY


VIEW PATIENT DETAILS
DIABETES - ABNORMAL
HYPERTENSION - ABNORMAL
HEART DISEASE - HIGH
TUBERCULOSIS - HIGH
BCG VACCINE - NO
DPT VACCINE - NO