COMPUTER TELEPHONY INTEGRATION - PHONE BANKING

PROJECT REPORT

P-942

SUBMITTED IN PARTIAL FULLFILMENT 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.R.Karunambikai
Reg No –9837S0052

Under The Guidance of
Ms.L.S.Jayashree M.E ,
Senior Lecturer
Kumaraguru College Of Technology
Chinnavadempatti
Coimbatore.

Mrs.Geetha Sekhar,
Technical Leader,
HumanWare Integrated Systems Ltd,
Chennai.

Department Of Computer Science And Engineering
Kumaraguru College Of Technology
Coimbatore –641006
April 2003

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

KUMARAGURU COLLEGE OF TECHNOLOGY

COIMBATORE - 641 006

CERTIFICATE

This is to certify that the project entitled

PHONE BANKING

has been submitted by

Ms. R.KARUNAMBIKAI

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

Guide 20/3/03

S JL 24/3/03
Head of the Department

Certified that the candidate was examined by us in the Project Work Viva Voce Examination held on 4-4-2003 and the University Register Number was 9837S0052

Internal Examiner

External Examiner

HUMANWARE

PROJECT CERTIFICATE

This is to certify that **Ms. R.Karunambikai,** has completed a project with us as per following details:

PROJECT NAME: **PHONE BANKING**

BRIEF CHARACTERISTICS: The Project is an Interactive Voice Response Application where the caller can call his bank and be updated on transactions pertaining to his account at any time of the day. It also enables the caller to know general information about the bank.

TECHNOLOGIES INVOLVED: **Computer Telephony Integration**

DATE OF COMMENCEMENT: **21.12.2002**

DATE OF COMPLETION: **17.03.03**

DURATION: 3 **Months**

We have found her work good, enthusiastic and in line with our expectations.

During the course of our association with Ms.Karunambikai, we also found that she had a good conduct and was capable of introspection and conceptual analysis, which we feel is necessary for a good programmer.

Mrs Geetha Sekhar
Project Coordinator.

ACKNOWLEDGEMENT

With profound gratitude I Praise the lord for his abundant blessings

With profound sincerity and gratitude I thank our Principal **Dr.K.K.Padmanabhan,BSc(Engg),MTech,Ph.D** and our HOD **Dr.Thangaswamy Ph.D.**

I express my sincere thanks to my guide **Ms.L.S.Jayashree, M.E**, Senior Lecturer/ CSE Kumaraguru college Of Technology, Coimbatore for being a constant source of support and encouragement.

I express my gratitude to the course Co-coordinator **Mrs.Devaki, B.E, M.S**, Kumaraguru college Of Technology, Coimbatore for her regular supervision in the course of completion of the project.

It gives me great Felicity to convey my deep & sincere thanks to **Mrs.Geetha Sekhar**, Technical Leader, HumanWare Integrated Systems Ltd, Chennai for permitting me to develop a Project and providing me the necessary guidance & help to complete the project.

I would like to thank **Mr.ManiKandan**, Head (Education Delivery), HumanWare Integrated Systems Ltd, Chennai for providing me an opportunity to do this project work.

I also take this opportunity to thank each & every staff of CSE Department of Kumaraguru College Of Technology, Coimbatore and my parents, relatives and friends who have helped me in making a successful completion of the project

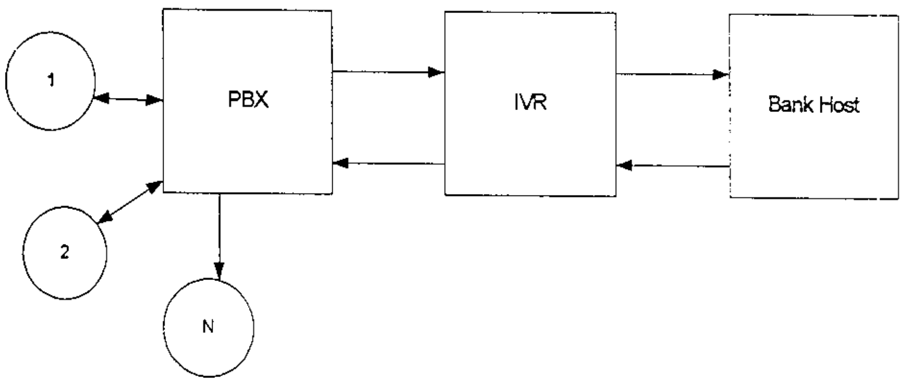
SYNOPSIS

Interactive Voice Response System (IVR)

Interactive Voice Response is a System of recorded menus and voice prompts for automated information retrieval. The applications that access and update databases while responding to caller queries are referred to as Interactive Voice Response System. The Computer Telephony (CT) components used in the IVR combines data and voice systems to provide a more efficient telephone services through telephone and perform transactions.

The prototype IVR application are banking by phone and automated order entry .We can query over bank balance from a 24 hour automated service line or dial an information number offering a "touch-tone" menu selection by utilizing a voice processing system at the service provider end.

P-942



Interactive voice Response Based Phone Banking System

The **Phone Banking** using Interactive Voice Response (IVR) is a novel idea that facilitates the clients to use the bank services effectively through telephone and is developed for Desi Bank, Mumbai at HumanWare Integrated Systems Ltd, Chennai.

The Phone banking scores over the manual banking system with its distinctive features. It offers high level of convenience to both customers and the bank. The round-the-clock availability of the service, ability to handle high volume of transactions and enhanced security are the highlighting features of this system.

Phone Banking provides scope for accounting and general information. The customers can access the accounting information by providing their account number and the pin number. The accounting information includes details regarding the balance status, cheque status, funds transfer, stop payment of cheque and chequebook request. The general information provides the users with instructions for opening new account, interest rates, exchange rates for foreign currencies and loan details.

CALL PROCESSING

Call processing is the combination of telecommunication and data processing allowing callers to use telephone almost like keyboard interacting with computer. The information transmitted over telephone lines can be divided into two categories, VOICE and DATA. Voice processing adds intelligence to the underlying process of transmitting sound from one place to another. A device with voice processing functions is called the voice response unit or VRU.

VOICE STORE AND FORWARD

This is simply the industry jargon for record and playback. A voice processing system with store and forward capabilities has the capability has the ability to record audio and play it back at a later time.

DIGIT CAPTURE

Every telephone allows the user to dial digits from zero to nine. Voice processing system will often be able to recognize and act on the digits dialed by the caller. DTMF (Dual Tone Multi Frequency) push button or touch-tone dialing can be detected by the voice system and results in a computer telephony network.

MODULES INVOLVED IN THE PROJECT:

Phone Banking is composed of two modules: IVRS and Admin module. IVRS is the telephony branch of the application. Admin is the administrator module of the application. A Recording tool that facilitates dynamic Recording of "Vox" files comes with this application. The application is developed using Dialogic SDK 5.01, 'C' in VC++ Environment , Visual Basic 6.0 – Front End and Oracle 8.0 - Back End.

COMPUTER TELEPHONY INTEGRATION

- *Phone Banking*

S.NO	CONTENTS	PAGE NO
1	**COMPANY PROFILE**	1
2	**SYSTEM REQUIREMENTS SPECIFICATION**	**3**
2.1	INTRODUCTION	3
2.2	GENERAL DESCRIPTION	5
2.3	SPECIFIC REQUIREMENTS	8
3	**SYSTEM ANALYSIS**	**22**
3.1	EXISTING SYSTEM	22
3.2	PROPOSED SYSTEM	22
3.3	REQUIREMENT ANALYSIS	23
3.4	FEASIBILITY STUDY	23
3.5	DATA FLOW DIAGRAM	25
3.6	CALL FLOW DIAGRAM	29
4	**SYSTEM DESIGN**	**35**
4.1	SYSTEM ARCHITECTURE	35
4.2	APPLICATION ARCHITECTURE	37
4.3	TABLE DESIGN	38
4.4	NORMALIZATION	40
4.5	MODULE DESCRIPTION	41
4.6	SAMPLE SCREENS	44
5	**CODING STANDARDS**	**47**
6	**TESTING AND CORRECTION**	**48**
7	**INSTALLATION OF THE SYSTEM**	**53**
8	**CONCLUSION**	**54**
8.1	BENEFITS	54
8.2	LIMITATIONS	54
8.3	ENHANCEMENTS	55

About the Company...

1. COMPANY PROFILE

Computer Telephony is the technique of coordinating the actions of computer systems and telephone. This technology has existed in commercial form since 1980's, but it has been exploited only in a few niche markets particularly in large call centers, where call volumes easily justified the cost of complex custom-built system.

Businesses have to proactively respond. Respond irrespective of fact that it may be a web/voice/email/acd or fax interaction. Respond irrespective of the time and location of request; respond irrespective of the type of information required.

Servion[t] Global Solutions is a company whose focus is to revolutionize the business response of every single organization through its contact center suite products and systems Integration capabilities.

With Business Response solutions from Servion[t], every company is empowered with an inequitable advantage of linking all customer interaction channels up into one unified customer response system that is quick responding and customer friendly.

With over 200 customers and 500 installations worldwide, the Servion[t] products and solutions handle approximately 10 million interactions (voice/fax/acd/web/email) per month.

Servion[t] has single largest Principal Tie-up with global giants like Avaya Communication, Dialogic Corporation, and Nortel Networks. Servion[t] has also established tie-ups with BEA Systems and Microsoft Inc.

Performance

Some of the trend setting achievements of Servion[t] during the last year that have been setting the standards for the IVR / Call center /CTI /Internet Solutions & Middleware industry have been listed below:

- Citibank's private equity division has identified Servion[t] as one of the key players for their investment plans of $5-10 million on IT enabled services.
- "CTI specialist of the year" was presented to Servion[t] presented as the editor's choice awarded by Telephony Middle East for the year 1999-2000.
- "Most Significant Sale" award across the Asia Pacific Region awarded by Lucent technologies for the Southern Railways turn key project.
- "Excellent Performance" as Value Added Reseller award of Dialogic Corporation
- The only company in India to have already implemented a multi-lingual world-class call center for a leading foreign bank in Middle East.
- The only company that is recognized by Lucent Technologies and Dialogic Corporation as having the necessary skill set to be a "Partner of Choice" to market and support business communication product across the Asia Pacific Region.
- The first company to develop and implement an "Extranet" solution for a leading manufacturing (Automobile) industry.

A Peep into the Project…

2. SOFTWARE REQUIREMENTS SPECIFICATION

2.1. INTRODUCTION

2.1.1 OBJECTIVE

To design an automated phone banking system using Interactive Voice Response System that facilitates the clients to use the banking services effectively through telephone and an Administrative System for controlling, monitoring and assessing the telephony application. The project aim is to eliminate the need of customers to be physically present for any transaction or enquiry and to provide a Centralized call flow administration.

2.1.2 PURPOSE

The purpose of Corporate Phone Banking application is to cater services to the corporate Customers of Desibank N.A, Mumbai.

The Automated Tele Banking System (ATBS) using Interactive Voice Response (IVR) is a novel idea that facilitates the clients to use the bank services effectively through telephone.

The ATBS scores over the manual banking system with its distinctive features. The ABTS offers high level of convenience to both customers and the bank. The round-the-clock availability of the service, ability to handle high volume of transactions and enhanced security are the highlighting features of this system.

The IVR-Tele Banking system, using INTEL dialogic card is one among the pioneering efforts to offer new level of performance using computer telephony networks. Text-to-Speech concept used here represents a price performance breakthrough, enabling telephony application developers to offers leading edge functionally such as unified messaging and text-based IVR to small system market at an attractive price. Speech technologies enhance the value of telephony boards, giving the developers more functionality at cost effective price points.

Using text to talk, the host computer can read and standard ASCII Text over telephone lines making Text to Speech a useful alternative to access large records such as database of names and addresses that are impractical to record as digitized sound files.

The IVR-Admin, which is an integral part of this application, gives an edge over other telephony applications, which are devoid of an administrator module. Administrator is vital for monitoring and reporting the day-to-day status of the application. Without Admin the management will find itself in no man's land as far as the application is concerned. Administration is the remote, which operates the IVR-Tele Banking at its will.

The Recording Tool, which enables the dynamic recording of "vox" files, is a blessing in disguise for the bank, which is going to incorporate this application.

2.1.3 SCOPE

IVR System provides scope for telephony services like,

- Tele Banking
- Railway Reservation and Enquiry System
- Airline Reservation and Enquiry System
- Theatre Reservation and Enquiry System
- Hotel Reservation and Enquiry System
- Hospital Enquiry
- Aids Awareness System
- Sun sign Characteristics System

IVR systems can be implemented

- In Windows environment.
- To handle many calls at the same time using Multiplexed channels.

Dialogic card provide synchronous and asynchronous modes to operate on receiving calls and handling his requests.

2.1.4 ABBREVIATION

IVR	-	Interactive Voice Response
PBX	-	Private Branch eXchange
ATBS	-	Automated Tele Banking System
PCI	-	Peripheral Component Interconnect
ISA	-	Industry Standard Architecture
CSO	-	Customer Support Organization
SDK	-	Software Development Kit
DCM	-	Dialogic Configuration Manager
TPT	-	Termination Parameter Table

DEFINITION:

DIALOGIC SOFTWARE:

The Dialogic Software, which automatically gets installed in the system along with the Dialogic card, provides Dialogic API functions of Intel, essential for the Dialogic card to accept and respond to calls.

DIALOGIC CONFIGURATION MANAGER:

Checks that all the hardware is properly connected & maintains the data used to configure Dialogic products.

INTERACTIVE VOICE RESPONSE SYSTEM:

Interactive Voice Response System facilitates the clients to use the banking services effectively through telephone and an Administrative System for controlling, monitoring and assessing the telephony application

2.2 GENERAL DESCRIPTION

2.2.1 OVERVIEW:

The automated phone banking system-using IVRS is a novel idea that facilitates the clients to use the bank services effectively. Telebanking is a service that makes banking easy from any touch tone telephone 24 hours a day by enabling customers to obtain various information about their account with the bank & also perform various transfers between their accounts. Standard menu options provide for checking balances inquire about specific cheque & obtain fact copy of last specified transaction, additional options & menus can be added on demand. Tele banking mode seamlessly integrates within the existing banking network.

2.2.2 PRODUCT FUNCTION

IVR-Call flow handles the customer's calls and provides a proactive customer service. It enables the channels to receive calls from the customers and responds to the callers by itself, without any manual assistance. It retrieves answers from the bank host, based on information each caller provides, the access number and T-pin number, etc, which the caller presses on his/her telephone.

The caller may be a customer who may have various queries regarding his account or a non-customer who may access general information about the products. The information is played back to the caller in the form of ". vox" files. Apart from retrieving information dynamically, the module also transfers call to the operator or to a desired extension number that is provided by the user.

FEATURES:

- Complete customer profile maintenance
- Account transaction management
- Checking the status of the cheque issued and realized
- Transfer of funds between accounts
- Transfer of funds towards credit card payment
- Handling loans information
- Handling locker information
- Handling fixed and recurring deposit information
- Credit card blocking
- Account blocking providing general information about the bank's products
- Facilitating call transfer

Recording Tool:

The recording tool which comes along with this application opens the channel through which the recording is to be done, sets it off hook, waits for the ring, then sets the channel off hook and records the voice and saves it as a *.vox file. It allows the user to dynamically change the time of recording for each voice file.

2.2.3 USER CHARACTERISTICS

The customer must dial the valid customer identification number and telephone identification number and

* Retrieve Account related information

* Request for chequebook and drafts and Grievance registration and general information like Interest rates and Foreign Exchange Rates.

2.2.4 GENERAL CONSTRAINT

The dialogic card installed should be checked before and after installation.

BEFORE INSTALLATION:

- The operating system to be used should be Windows/Unix.
- Check the space for 600mb.
- Dialogic Driver 5.01 is to be used.
- Slot – PCI/ISA should be checked to be free.
- PBX extension should be 4 numbers

AFTER INSTALLATION:

- Dialogic Card status should be Start/Stop.
- Run sample application.
- Extension should be properly connected.
- Customer Support Organization (CSO) can be used to transmit extension.

2.3. SPECIFIC REQUIREMENTS

2.3.1 PERFORMANCE REQUIREMENTS

SECURITY

The customers with only valid customer identification number and telephone identification number could be allowed to perform Queries.

2.3.2 SYSTEM SPECIFICATION

SOFTWARE REQUIREMENTS

- VC++ Environment
- C
- Dialogic SDK 5.01
- Visual Basic 6.0
- Oracle 8.0

DIALOGIC SOFTWARE:

The Dialogic Software, which automatically gets installed in the system along with the Dialogic card, provides Dialogic API functions of Intel, essential for the Dialogic card to accept and respond to calls.

FEATURES:

- Single programming paradigm for all computer telephony technologies.
- Scalability allows an application to run on small or large system.
- Binary compatible API with Dialogic Development Kit for Windows 95; provides further scalability to a smaller, lower cost operating system.
- Multiple compiler supports including Microsoft Visual C++ 2.0 and higher, Borland C++ and higher.
- ". WAV" file plays and records.
- SMP support.

- Supports threads, optimizing system resource usage.
- Synchronous and Asynchronous programming models.
- Dynamic Link Library reduces version dependencies.
- Bulk data handled by driver/library leaving application free from real-time constraints.
- Interrupt driven architecture optimizes use of system resources.
- Downloadable firmware offers optimum performance, reliability and upgradability.
- Software architecture reduces memory use and lowers maintenance costs.
- State of art development support.

FUNCTIONS OFFERED:

Device Management Functions:

- Function to open a channel and making it available for callers.
- Function to close the channel and when it is no longer needed.
  - ✓ dx_open (voice device name, network device name)
  - ✓ dx_close (voice device name, network device name)

Configuration Functions:

- Functions to toggle with the hook state of the channel .The states available are On-hook and Off-hook.
- Function to make the channel wait for a substantial number of rings.
  - ✓ dx_sethook(device, Hook state , mode)
  - ✓ dx_wtring (device, No of rings, Hookstate, Timeout)

I/O Functions:

- Functions to transfer calls to any desired extension.
- Function to receive the digit pressed by the caller on the telephone pad.
  - ✓ dx_getdig (device, tpt, digitp, mode)

Play and Record Functions:

- Functions to record voice files on a channel.

- Functions to play the recorded voice files.
    - ✓ dx_playf (file name)
    - ✓ dx_recf (file name)

Structure and Clearance Functions:

- Functions available to clear the structure used to receive input and terminate I/O operations.
    - ✓ dx_clrdigbuf (device)
    - ✓ dx_clrpt (tpt, size)

The C Language

C is a very flexible and powerful programming language originally designed in the early 1970's.It is a professional language. C is often called a "Middle Level" programming language. This is not a reflection on its lack of programming power but more a reflection on its capability to access the system's low level functions. Most high-level languages (e.g.: Fortran) provide everything that the programmer wants to do already built into the language. A low level language (e.g.: Assembler) provides nothing other than access to machine basic instruction set. A middle level language, such as C, probably doesn't supply all the constructs found in high languages – but it provides with all the building blocks that is needed to produce the results.

The largest measure of C's success seems to be based on purely practical considerations:

- ✓ The portability of the compiler
- ✓ The standard library concept
- ✓ A powerful and varied repertoire of operators
- ✓ An elegant syntax
- ✓ Ready to access to the hardware when needed
- ✓ The ease with which the applications can be optimized by hand-coding isolated procedures

What comprises a C program?

- ✓ Instructions to the compiler

✓ Information about the structures, which will hold the data to be stored and manipulated

✓ Instructions, which manipulate the data

Controlling the Compiler

"The compiler directives are the same for all versions of C".

One of the very powerful features of C is the way in which the compiler processes the program by including directives. There are many directives available, in a C program a directive is always preceded by the # character and must appear right at the beginning of a line. Directives can be used to "build in" particular values, for example constants and also allow to change which parts of the program which the compiler works on, making it possible to use the same piece of program o several different types of computer.

Storing the Data

Programs work by processing data. The data has t be stored within the computer whilst the program processes. All the computer languages support variables of one form or another. A variable is simply named location in which a value is held whilst the program runs also let's to build up structures, which can hold more than one item.

Describing the Solution

The actual instructions, which describe the solution to the problem, must also be part of the program. In the case of C a lump of program, which does one particular thing, is called a "function".

A function can be very small, or very large. It can return a value, which may or may not be of interest. It can have any name and the program can contain as many functions .One function may refer to others. The c language also has a large number of function libraries. These save from "re-inventing the wheel" each time a program is written.

Within a function there will be a number of statements. A statement is an instruction to perform one particular operation. The really gripping part of the programs is that a statement can decide which statement is performed next, so that the program can look at thing and decide what to do.

The C language actually runs the program by looking for a function with a special name, main. This function is called when the program starts running, and when main finishes, the program ends.

Uses of C:

C was initially used for system development work, in particular the programs that make-up the operating system, mainly because it produces code that runs nearly as fast as code written in assembly language. Some examples of the use of C might be:

- Operating Systems
- Language Compilers
- Assemblers
- Text Editors
- Print Spoolers
- Network Drivers
- Modern Programs
- Data bases
- Language Interpreters
- Utilities

In recent years C has been used as a general-purpose language because of its popularity with programmers.

Visual Basic 6.0

Microsoft Visual Basic, the fastest and easiest way to create applications for Microsoft Windows. Whether you are an experienced professional or brand new to Windows programming, Visual Basic provides you with a complete set of tools to simplify rapid application development.

What is Visual Basic?

The "Visual" part refers to the method used to create the graphical user interface (GUI). Rather than writing numerous lines of code to describe the appearance and location of interface elements, you simply add pre-built objects into place on screen. If you've ever used a drawing program such as Paint, you already have most of the skills necessary to create an effective user interface.

The "Basic" part refers to the BASIC (Beginners All-Purpose Symbolic Instruction Code) language, a language used by more programmers than any other language in the history of computing. Visual Basic has evolved from the original BASIC language and now contains several hundred statements, functions, and keywords, many of which relate directly to the Windows GUI. Beginners can create useful applications by learning just a few of the keywords, yet the power of the language allows professionals to accomplish anything that can be accomplished using any other Windows programming language.

The Visual Basic programming language is not unique to Visual Basic. The Visual Basic programming system, Applications Edition included in Microsoft Excel, Microsoft Access, and many other Windows applications uses the same language. The Visual Basic Scripting Edition (VBScript) is a widely used scripting language and a subset of the Visual Basic language. The investment you make in learning Visual Basic will carry over to these other areas.

Whether your goal is to create a small utility for yourself or your work group, a large enterprise-wide system, or even distributed applications spanning the globe via the Internet, Visual Basic has the tools you need.

- Data access features allow you to create databases, front-end applications, and scalable server-side components for most popular database formats, including Microsoft SQL Server and other enterprise-level databases.

- ActiveX technologies allow you to use the functionality provided by other applications, such as MicrosoftWord Wordprocessor, Microsoft Excel spreadsheet, and other Windows applications. You can even automate applications and objects created using the Professional or Enterprise editions of Visual Basic.

- Internet capabilities make it easy to provide access to documents and applications across the Internet or intranet from within your application, or to create Internet server applications.

Your finished application is a true .exe file that uses a Visual Basic Virtual Machine that you can freely distribute.

Oracle 8.0

Oracle is the most powerful, enterprise – capable relational database management system.

Server Functions and features:

Work done by the database is handled by a complex set of memory structures and operating system processes called the instance. Every Oracle database has an instance associated with it. A database is mounted by only one instance.

Discrete processes perform specialized tasks within the RDBMS that work together to accomplish the goals. Each process has a separate memory block that it uses to store private variables, address stacks, and other runtime information. The processes use a common shared area in which to do its work-a section of memory that can be written to and read from at the same time by many different programs and processes. This memory block is System Global Area (SGA).

Standard Features

- Relational Architecture
- Administration tools
- Security management
- Backup and recovery tools
- SQL interface for Queries and updates
- Connection management for external applications
- Stored procedures and triggers

Advantages :

1. It provides minimum data redundancy and minimum data inconsistency
2. It eliminates the difficulty in accessing and manipulating data.
3. It offers uniform security and privacy control
4. Data sharing can be accomplished easily.
5. It provides data Integration.

Client Server concepts:

The User is called the Client. The Client and Server are said to follow a PROTOCOL. The server satisfies the protocol by providing the services it offers and the client satisfies the protocol by requesting only those services that have been offered. The terminology is, that "A Client messages the server, and in response the server executes a method". The server may execute a method by calling a member function or by retrieving an attribute.

Oracle8.0 strongly supports the OOPS concepts like Data Abstraction, Data hiding and Inheritance.

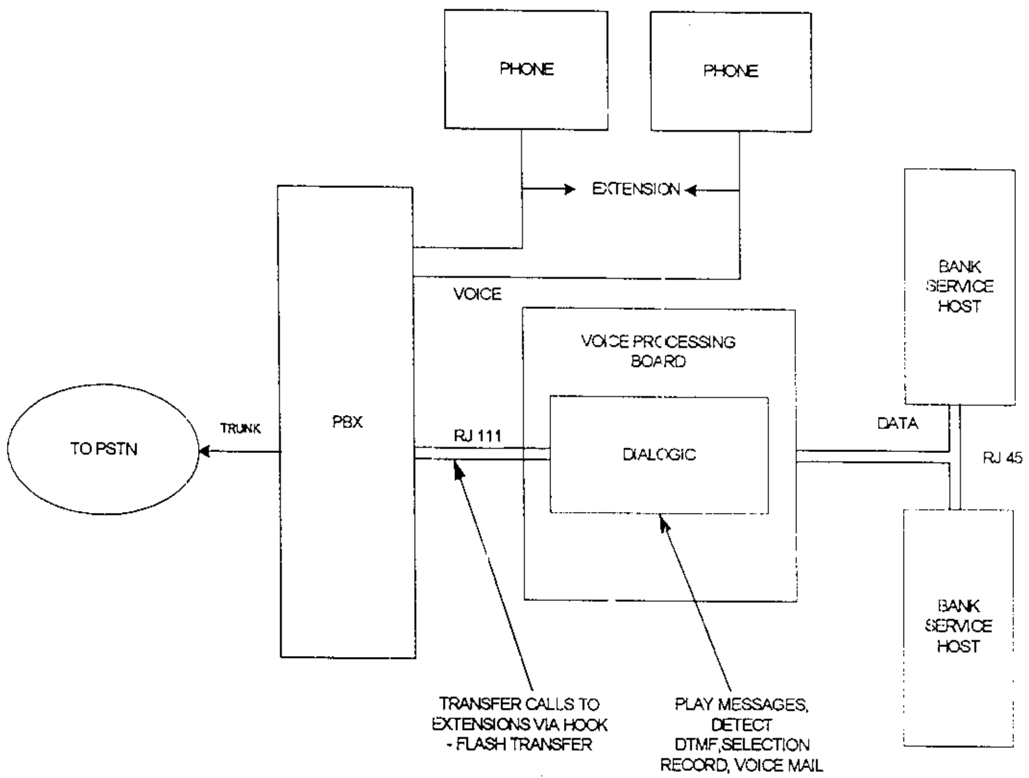
HARDWARE REQUIREMENTS

- Pentium 3 and above or IBM PC-AT bus or compatible computer.
- 20 MB Free Hard Disk space.
- 64MB RAM

The main components include:

- Private Branch eXchange (PBX).
- An Interactive Voice Response System (IVR).
- A Local Area Network of computers and a telephone connected to each computer.
- The communication Middle–Ware that fetches information from the main server of the bank (host) to the call center component that makes a request for such information.
- DM3 Boards
- Intel Dialogic D41/H Analog card

COMPONENTS OF IVR PHONE BANKING



Private Branch eXchange (PBX)

This is basically a switch that is used to connect the telephone lines of the call center to the IVR. The PBX by itself has an Analog to Digital converter thereby converting the analog signals from the telephone line to the digital signals for the IVR system. The PBX has a component called AUTOMATIC CHANNEL DISTRIBUTOR (ACD). This is used to multiplex many Analog lines to a single Digital line for efficiency reasons. Another application of the ACD is that it connects the caller to the entry point of the call center.

Interactive Voice Response System

It is the Backbone of the whole system. It acts as an interface between the information sources-the host had the information seeker. An IVR system is typically a separate server that contains digital signal processing (DSP) hardware that analyzes and reproduce voice patterns.

The IVR server interacts with a phone system through a dedicated connection. The way in which the IVR server is connected to the phone system depends on its capabilities and physical connection. An IVR can be connected to a key system, PBX, or other type of phone switch through analog ports, digital ports, and even a LAN or WAN connection.

It consists of a Voice Processing Board (VPB), which hosts a Dialogic Card. The Dialogic card is connected to the ISA/PCI slot of the computer. The PBX is connected to the VPB via RJ-11cables for voice transmission and the Dialogic card is connected to the Bank host through RJ-45 cables for data transmission.

The dialogic card is the most important component of the voice processing system that performs voice to data and data to voice conversion.

The four-line D/41H™ board and its two-line version, the D/21H™ board, are ideal for applications that need high-performance voice processing but don't require the large-scale system sophistication. The D/21H and D/41H boards use same dialogic Application Programming Interface (API). Even the weakest of telephone signals traveling over difficult telephone lines can be recorded and played back with complete clarity.

The D/21H and D/41H boards use the latest Digital Signal Processor (DSP) voice-processing technology, making them ideal for small and medium sized, server based computer telephony system – particular under Windows operating system.

International caller ID is supported on the d/21H and d/41 board, letting an application such as IVR calling party information via a telephone trunk line.

APPLICATION:

- Voice messaging
- Automated attended
- Interactive voice response
- Audiotex
- Inbound and outbound telemarketing
- Small call centers

Functional Description:

The D/21H and D/41H voice processing board builds on the patented Intel Dialogic dual-processor architecture that combines the signal processing capabilities of a DSP with the decision-making and data movement functionality of a general-purpose control microprocessor by using faster processors and considerably more memory. This dual-processor approach offloads many low-level decision-making tasks from the host computer, thus enabling easier development of more powerful applications. This architecture handles real-time events, manages data flow to the host PC for faster system response time, reduces host PC processing demands, processes DTMF and telephony signaling, and frees the DSP to perform signal processing on the incoming call.

Each of the four loop start interfaces receives analog voice and telephony signaling information from the telephone network (see the block diagram). Each telephone line interface uses reliable, solid-state hook switches (no mechanical contacts) and FCC-part 68 class B ring detection circuitry. This FCC-approved ring detector is less susceptible to spurious rings created by random voltage fluctuations on the network. Each interface also incorporates circuitry that protects against high-voltage spikes and adverse network conditions and lets applications go off-hook any time during ring cadence without damaging the board.

Part of the telephone interface for the D/4PCI board includes an on-hook audio path that detects Caller ID information. Depending on the level of service offered by the local telephone provider, Caller ID can include the date, time, caller's telephone number, and in some enhanced Caller ID environments, the name of the person calling. The on-hook audio path can also detect touchtone while the line is on-hook. This capability lets the board operate behind PBXs that require on-hook touchtone detection for their signaling.

Inbound telephony signaling (ring detection and loop current detection) are conditioned by the line interface and routed via a control bus to the control processor. The control processor responds to these signals, informs the application of telephony signaling status, and instructs the line interface to transmit outbound signaling (on-hook/off-hook) to the telephone network.

The audio voice signal from the network is bandpass filtered and conditioned by the line interface and then applied to a COder/DECoder (codec) circuit. The codec filters, samples, and digitizes the inbound analog audio signal and passes this digitized audio signal to a Motorola DSP.

Based on Spring Ware firmware loaded in DSP RAM, the DSP performs the following signal analysis and operations on this incoming data:

- Uses automatic gain control (AGC) to compensate for variations in the level of the incoming audio signal. The D/4PCI board also includes special circuitry to detect and amplify extremely weak line signals due to harsh telephone line conditions or back-to-back local loops often found in toll-free service environments
- Applies an adaptive differential pulse code modulation (ADPCM) or pulse code modulation (PCM) algorithm to compress the digitized voice and save disk storage space
- Detects the presence of tones - DTMF, MF, or an application-defined single- or dual-frequency tone
- Uses silence detection to determine when the line is quiet and the caller is not responding

For outbound data, the DSP performs the following operations:

- Expands stored, compressed audio data for playback
- Adjusts the volume and rate of speed of playback upon application or user request

- Generates tones - DTMF, MF, or any application-defined general-purpose tone

The dual-processor combination also performs the following outbound dialing and call progress monitoring

- Transmits an off-hook signal to the telephone network
- Dials out (places an outbound call)
- Monitors and reports results: line busy or congested; operator intercept; ring, no answer; or if the call is answered, whether answered by a person, an answering machine, a facsimile machine or a modem

When recording speech, the DSP can use different digitizing rates from 24 to 64 Kb/s as selected by the application for the best speech quality and most efficient storage. The digitizing rate is selected on a channel-by-channel basis and can be changed each time a record or play function is initiated. The popular 11 kHz, 8-bit linear multimedia WAVE format is also supported on the D/4PCI voice board.

Outbound processing is the reverse of inbound processing. The DSP processed speech is transmitted by the control microprocessor to the host PC for disk storage. When replaying a stored file, the microprocessor receives the voice information from the host PC and passes it to the DSP, which converts the file into digitized voice. The DSP sends the digitized voice to the CODEC to be converted into analog voice and then to the line interface for transmission to the telephone network.

The on-board microprocessor controls all operations of the D/4PCI board via a local bus and interprets and executes commands from the host PC. This microprocessor handles real-time events, manages data flow to the host PC to provide faster system response time, reduces PC host processing demands, processes DTMF and telephony signaling before passing them to the application, and frees the DSP to perform signal processing. Communications between this microprocessor and the host PC is via the shared RAM that acts as an input/output buffer and thus increases the efficiency of disk file transfers. This RAM interfaces to the host PC via the PCI bus. All operations are interrupt-driven to meet the demands of real-time systems. All D/4PCI boards installed in the PC share the same interrupt line. When the system is initialized, SpringWare firmware is downloaded from the host PC to the on-board code/data RAM and DSP RAM to control all board operations. This downloadable firmware gives the board all of its intelligence and enables easy feature enhancement and upgrades.

TYPES OF DIALOGIC CARDS:

D21H	- 2 ports
D41H	- 4 ports
D41ESC	- 4 ports
D40VFX	- 4 ports (FAX)
D4PCI	- 4 ports
D160ESC	- 16 ports

A Detailed System Study…

3. SYSTEM ANALYSIS

3.1. EXISTING SYSTEM

The current system, which is the conventional banking system, requires the physical presence of the client at the bank during he transactions. This system has the following limitations:

- ✓ Manual assistance is needed
- ✓ Error-prone due to human interference
- ✓ Service restricted to bank hours
- ✓ Large volume of transactions cannot be handled as everything is done manually
- ✓ Continual expenditure in the form of compensation to human labour
- ✓ Red tapism is a serious problem encountered
- ✓ Work becomes stagnant in case of human unavailability

3.2. PROPOSED SYSTEM

The proposed system, IVR-TeleBanking is an automated telephony system, which overcomes the above limitations. It has an edge over the conventional banking system due to incorporation of the following features:

- ✓ Facilitates the clients to use the bank services effectively through telephone
- ✓ Eliminates the need for manual assistance
- ✓ Round the clock service provided to clients
- ✓ Ability to handle high volumes of transactions
- ✓ Requires only a Dialogic Card and a PBX, which amounts to a once in a life investment
- ✓ Work schedules not affected as everything is computerized
- ✓ It works on windows, which is a GUI platform
- ✓ It can be fixed easily in a PC slot
- ✓ Installing these systems is relatively cheaper

- ✓ Easy to upgrade
- ✓ Platform independent.

3.3. REQUIREMENT ANALYSIS

3.3.1 Problem Recognition

The current system, which is conventional banking system, requires the physical presence of the client at the bank during transactions. This system requires manual assistance for it's functioning. The error-rate is high due to human interference.

Service is restricted to bank hours. Large volume of transactions cannot be handled as everything is done manually. There is a continual expenditure in the form of compensation to human labour. Red-tapism is a serious problem encountered. Work becomes stagnant in case of human unavailability.

3.4 FEASIBILITY STUDY

Before any user request is to be accepted, it is mandatory to check whether the new system is feasible or not. The major purpose of the analysis is to see that the development is technically and operationally helpful to the organization or not. Basically, the feasibility analysis could be further subdivided as Technical, operational and financial.

Technical Feasibility

The proposed is introduced in a technically sound environment such that the necessary technology exists with adequate holding capacity and allows future expansions. The Dialogic card is easily upgradeable. The system environment is sophisticated, as it is GUI based and provides utmost convenience to the end user

Operational Feasibility

The request of system development is beneficial only when the new system is operationally feasible that is there should be sufficient support to the new system. As the proposed system will definitely help all the staff of the company, it is found to be feasible.

Financial Feasibility

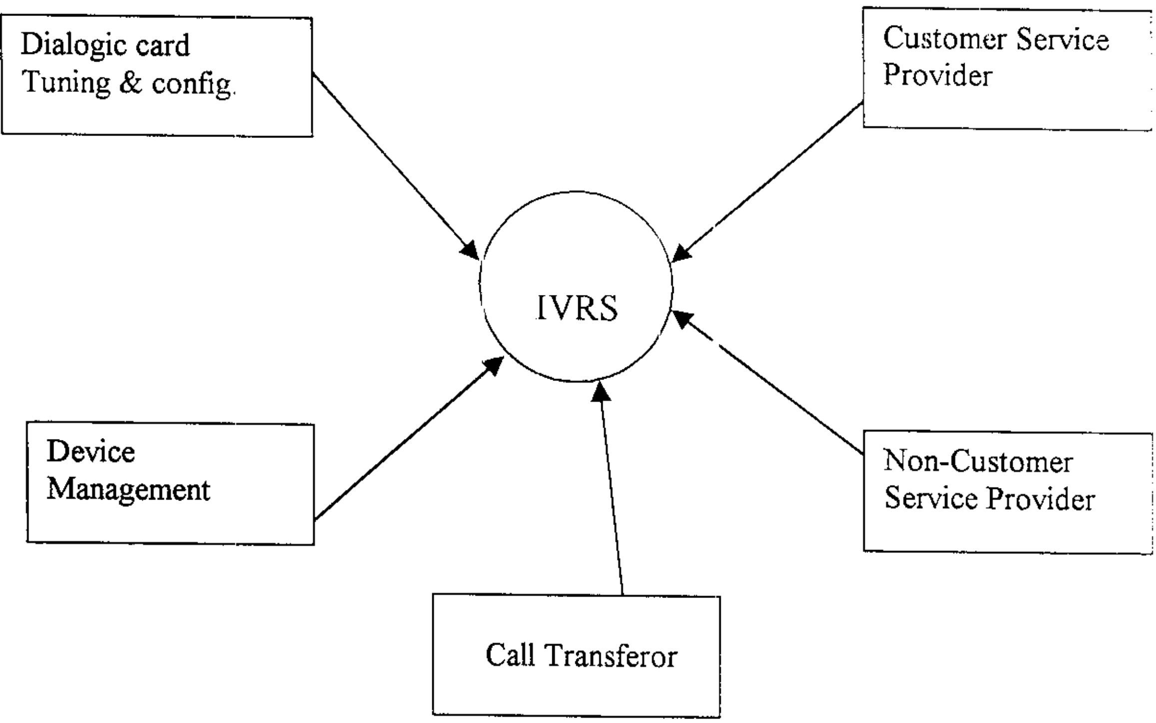
Any new development is feasible only when it is cost justified. Usually cost benefit analysis is made to find out the savings or the extra amount that would arise on accounting of the new development.

Every organization will choose a system only when there is some reduction in the cost for all times. The IVR system requires only a Dialogic card and a PBX, which amounts to a once in a life investment. It a highly beneficial to the Bank as it is cost reducing and revenue producing.

3.5. DATA FLOW DIAGRAM

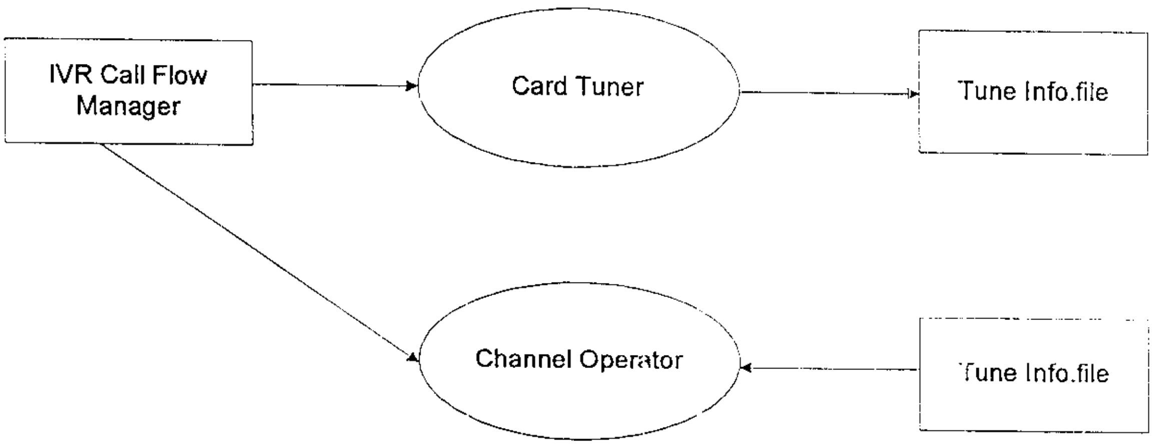
Level 0:

CONTEXT FLOW DIAGRAM

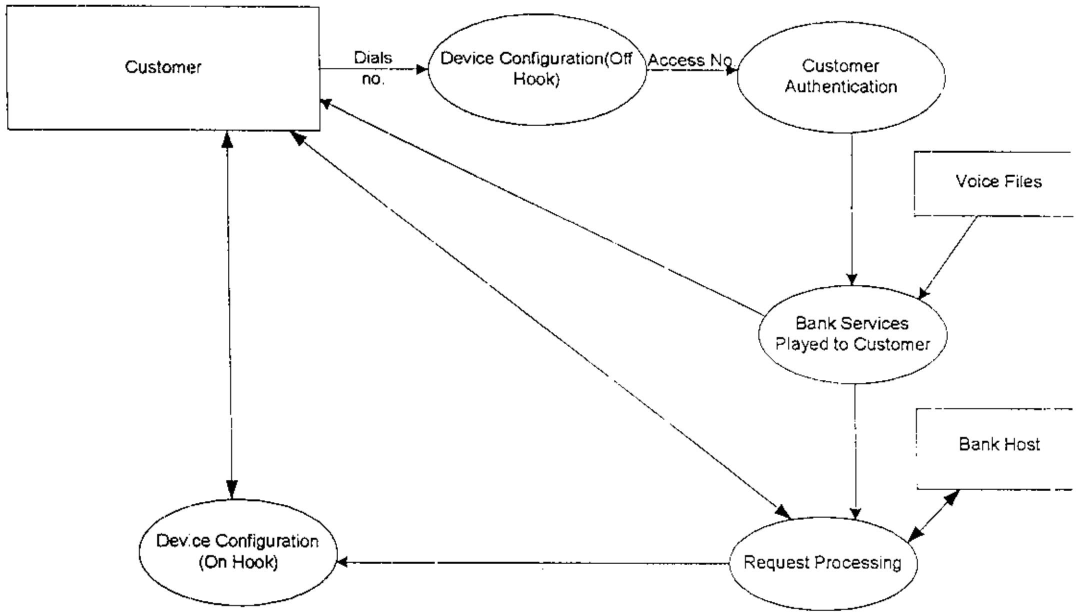


Level 1:

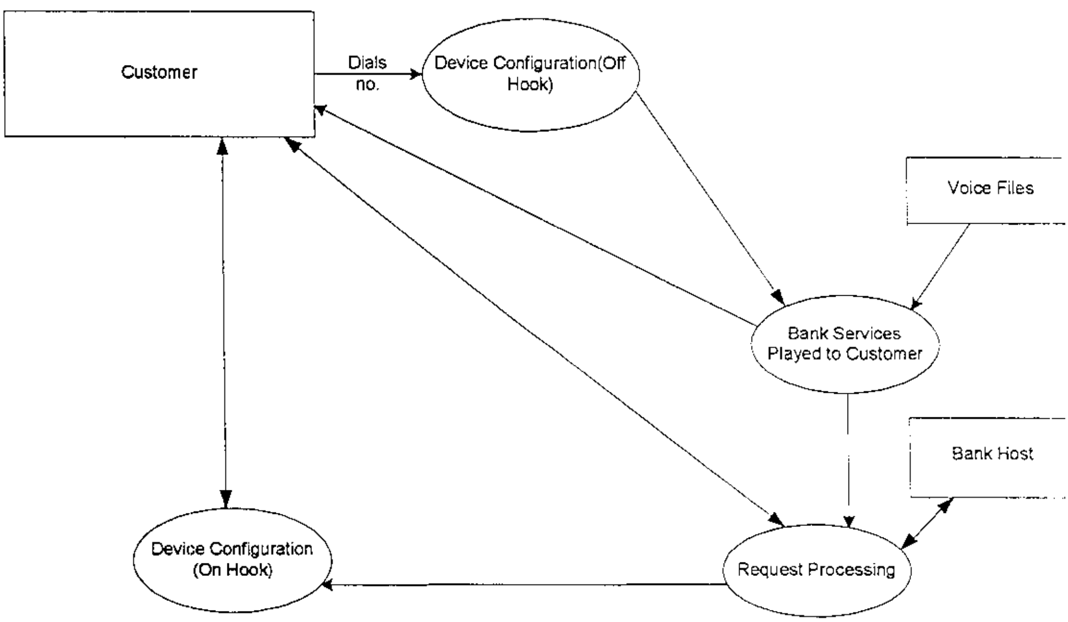
DFD FOR CARD AND DEVICE MANAGEMENT

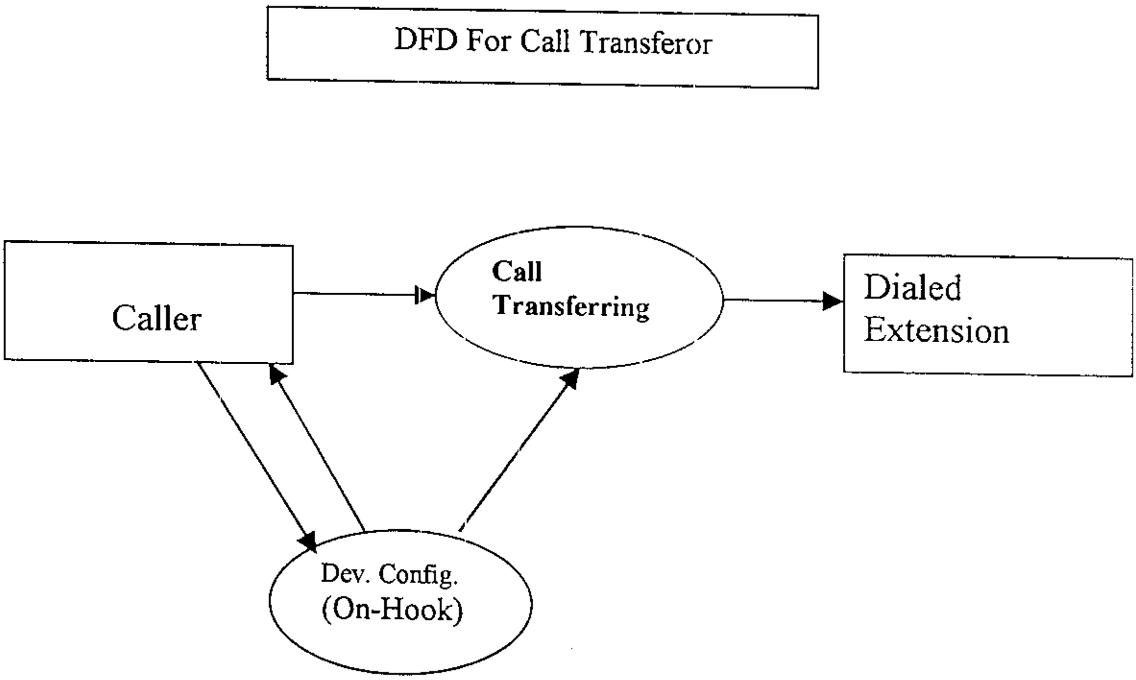


DFD for Customer Service Provider



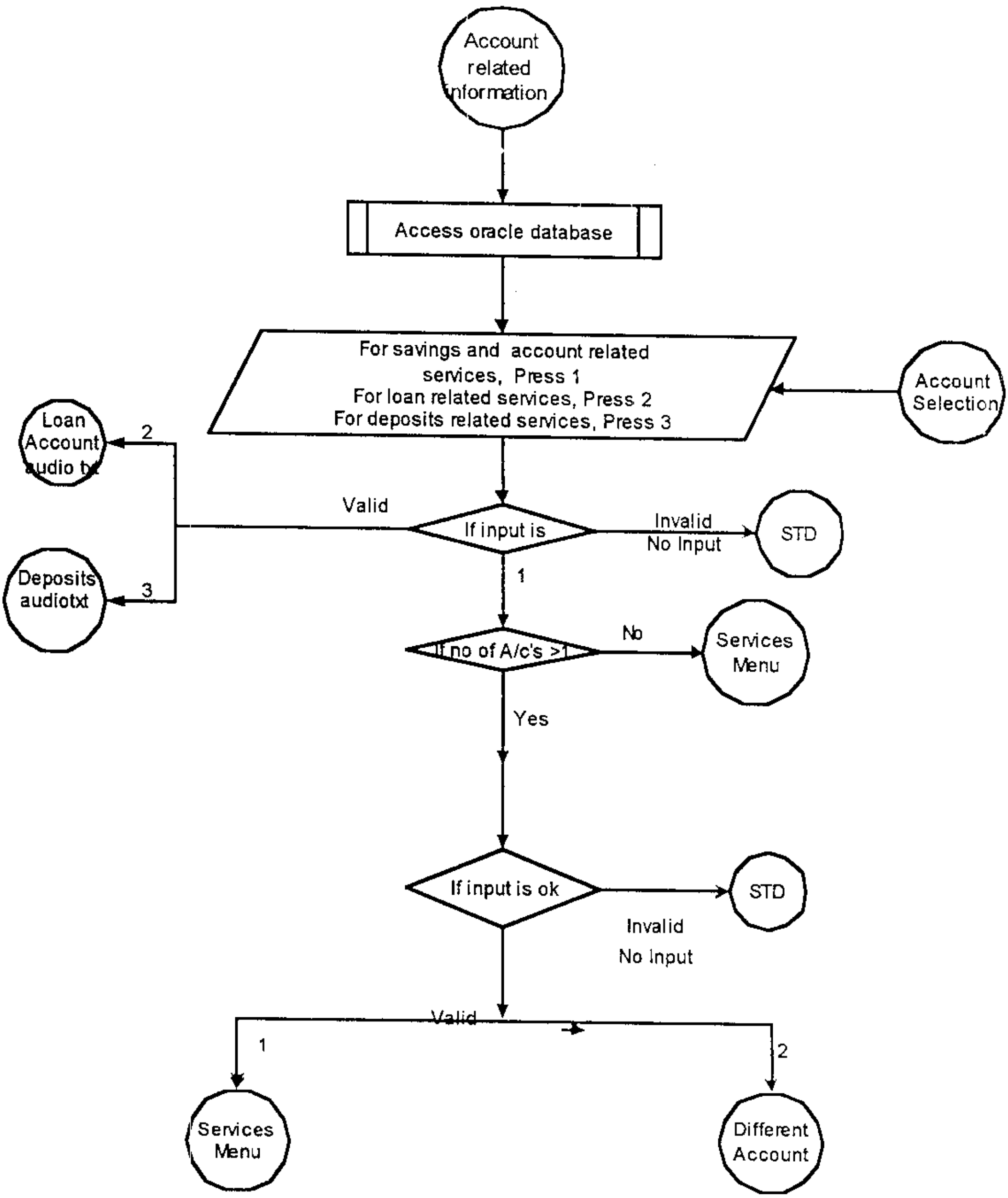
DFD for Non-Customer Service Provider



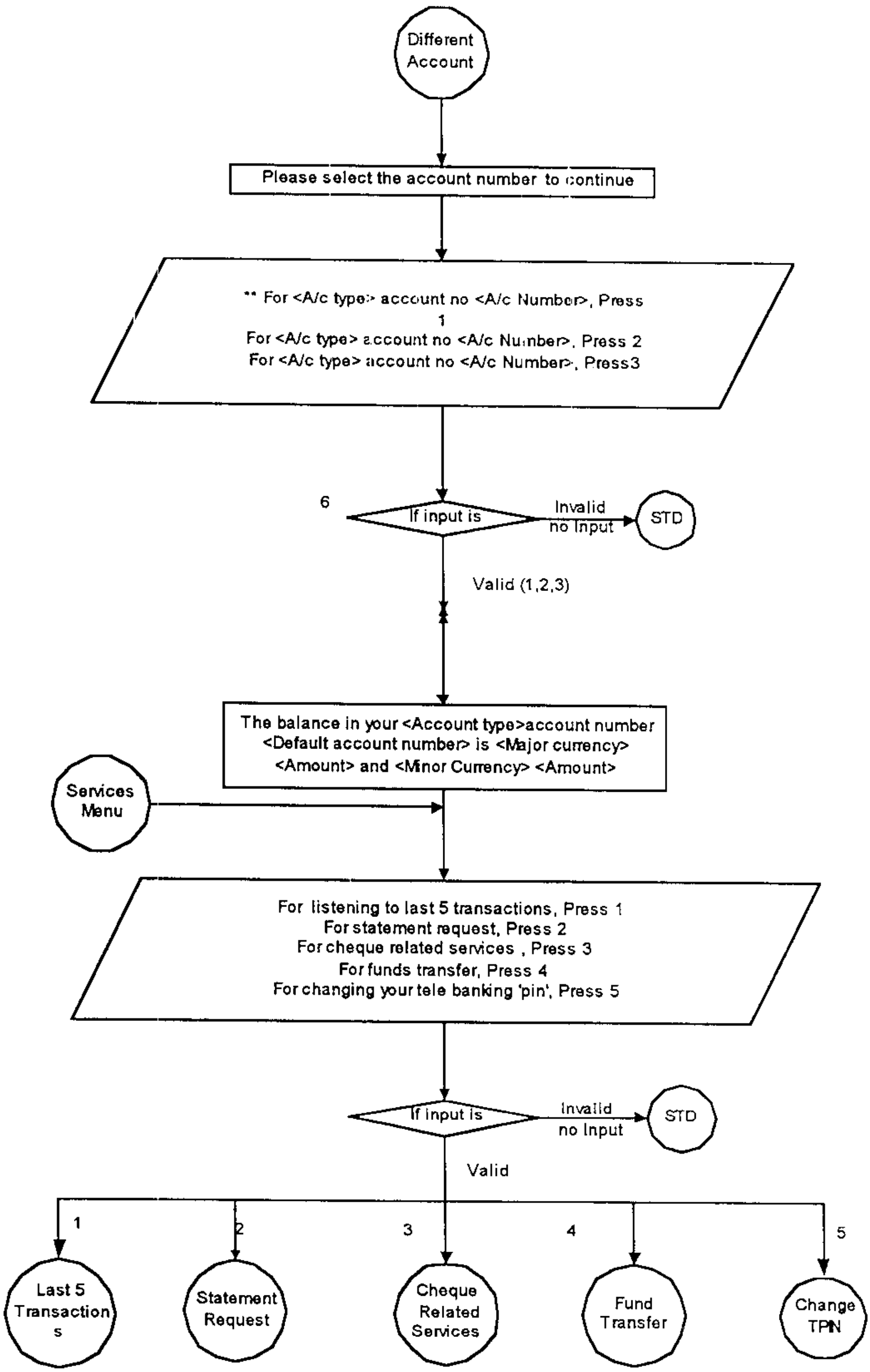

DFD For Call Transferor
Caller
Call
Transferring
Dialed
Extension
Dev. Config.
(On-Hook)

3.6 CALL FLOW DIAGRAM

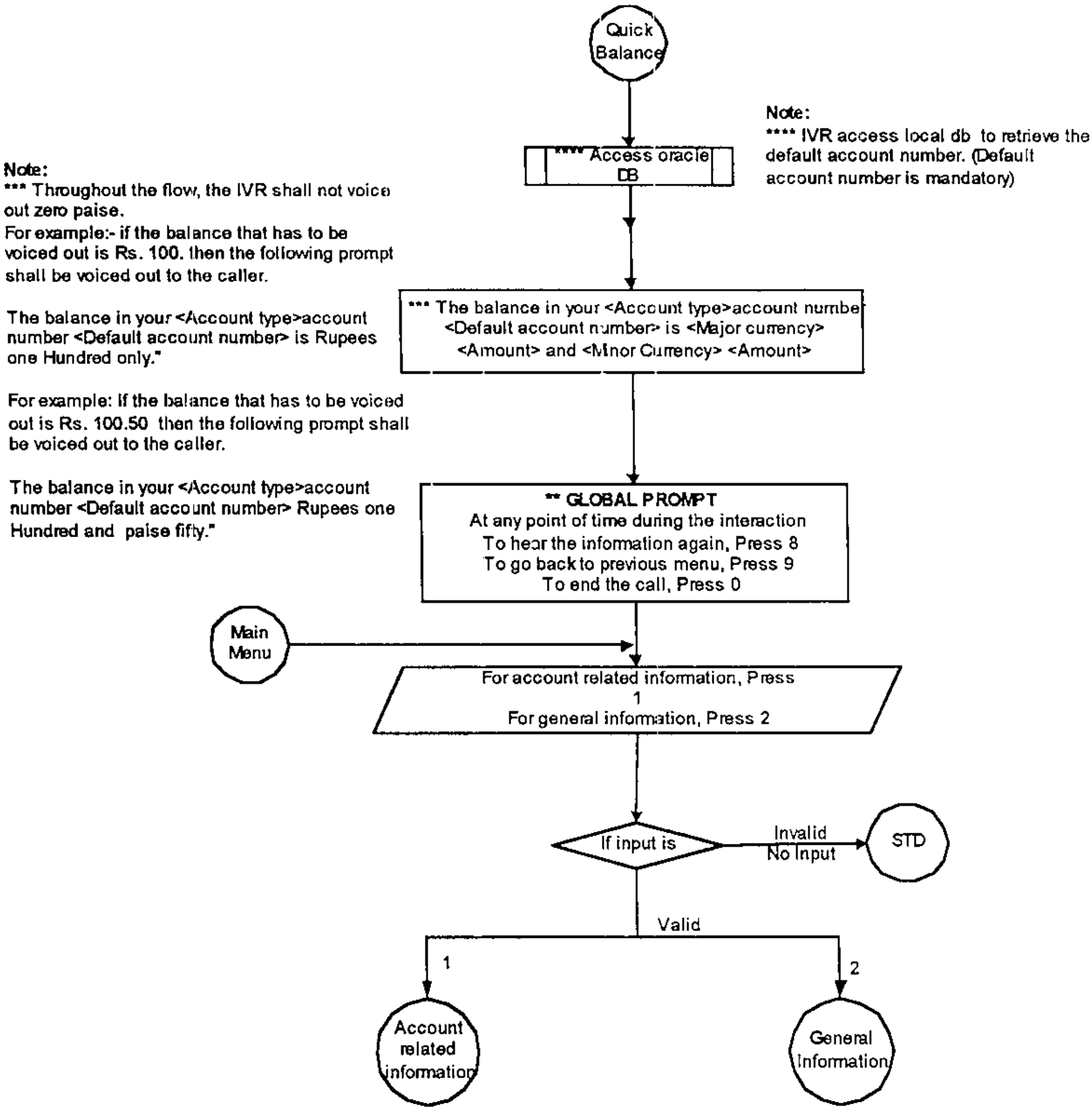
Account Related Information



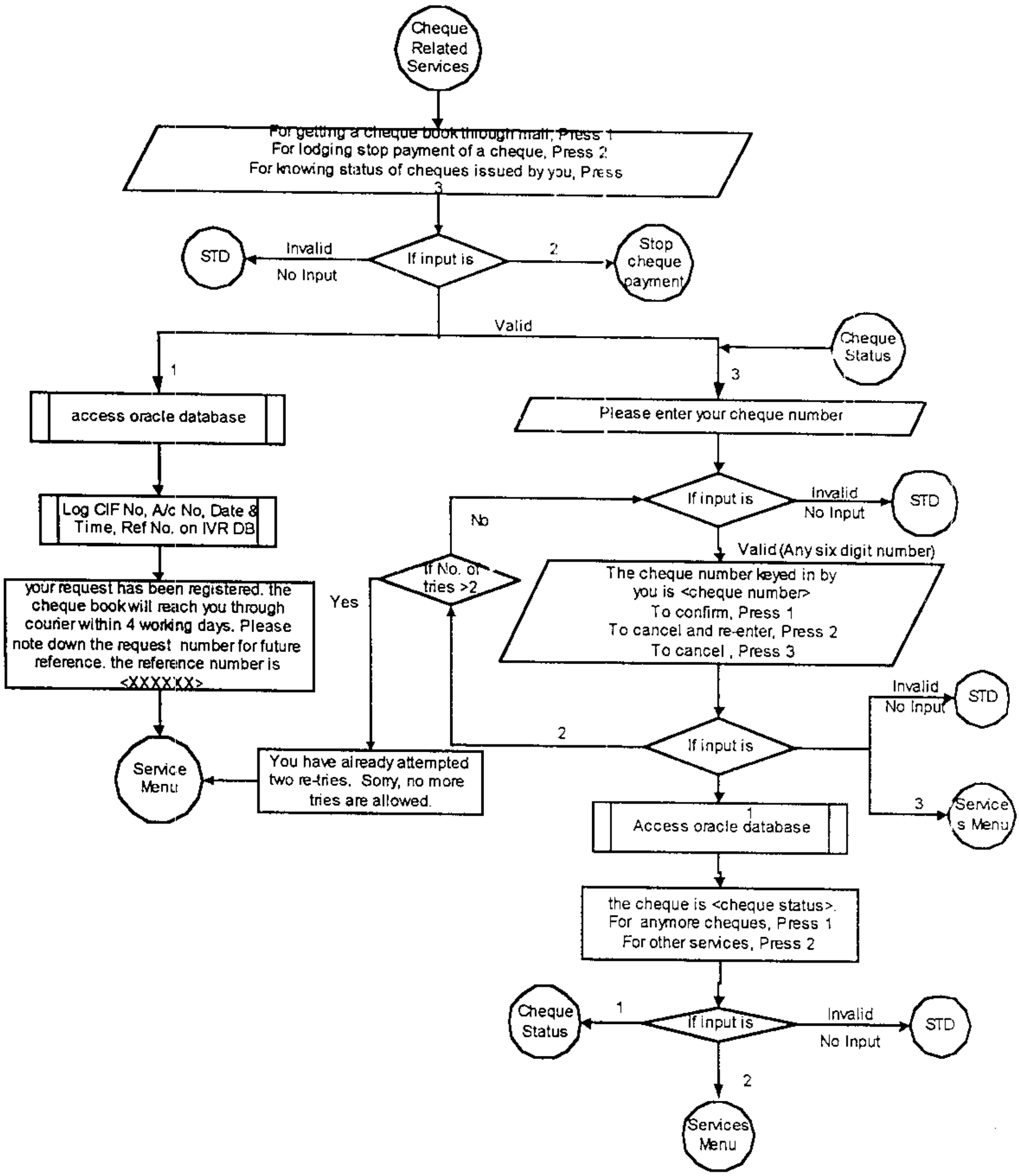
Different Account



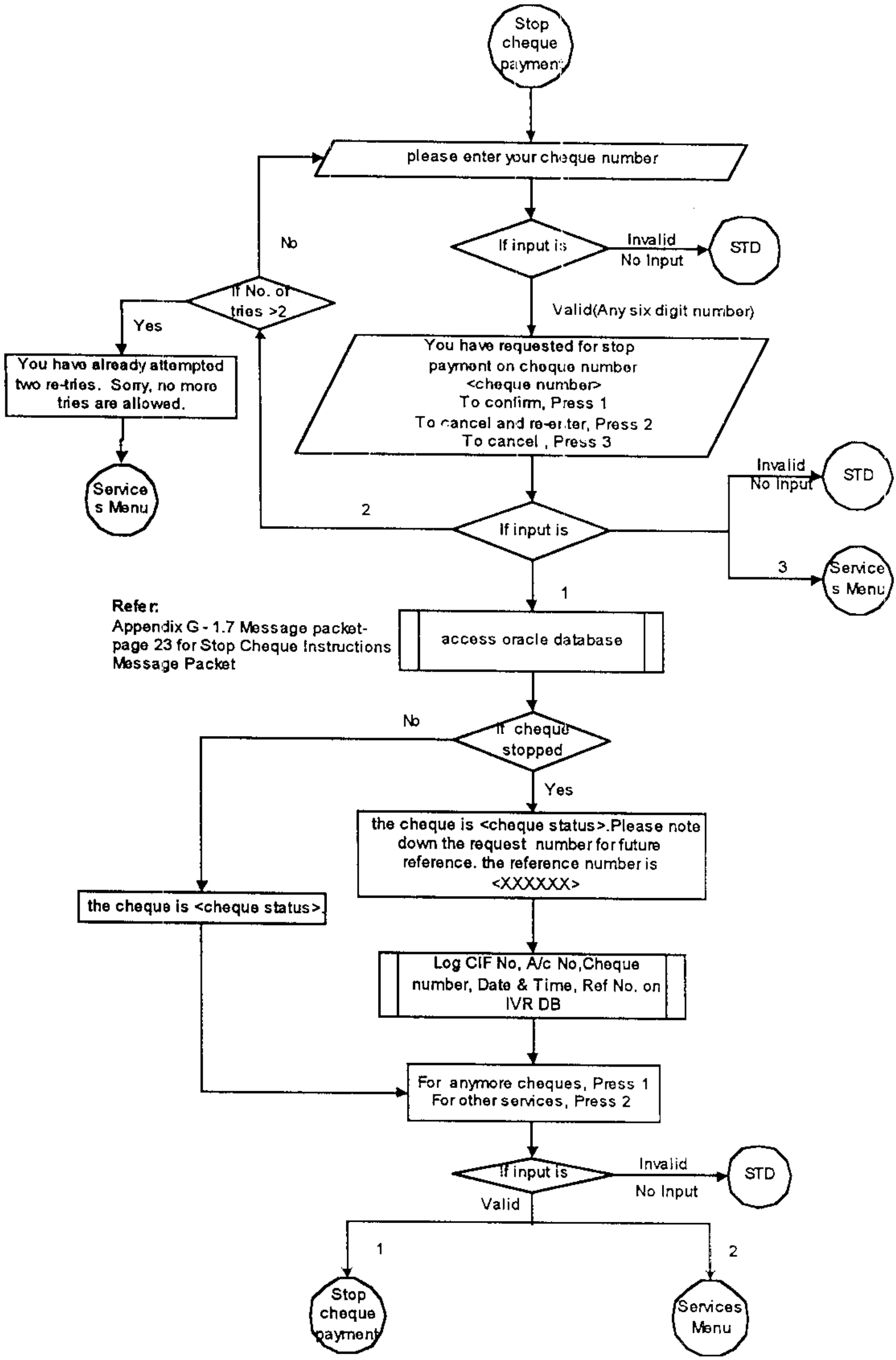
Quick Balance



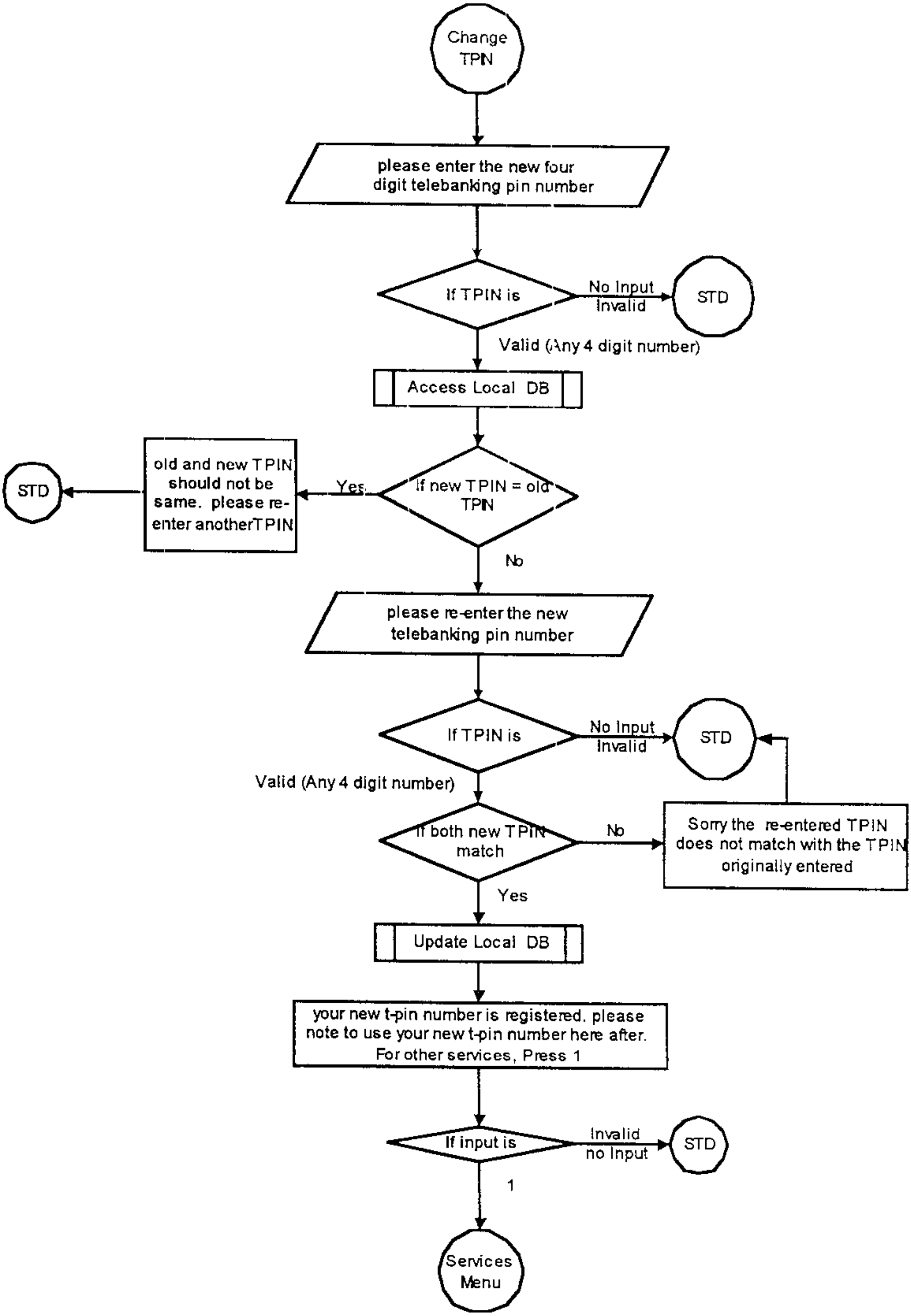
Cheque Related Services



Stop cheque payment



Change TPIN



Change TPN
please enter the new four digit telebanking pin number
If TPIN is
No Input
Invalid
STD
Valid (Any 4 digit number)
Access Local DB
If new TPIN = old TPN
Yes
old and new TPIN should not be same. please re-enter anotherTPIN
STD
No
please re-enter the new telebanking pin number
If TPIN is
No Input
Invalid
STD
Valid (Any 4 digit number)
If both new TPIN match
No
Sorry the re-entered TPIN does not match with the TPIN originally entered
Yes
Update Local DB
your new t-pin number is registered. please note to use your new t-pin number here after. For other services, Press 1
If input is
Invalid
no Input
STD
1
Services Menu

Design of the System…

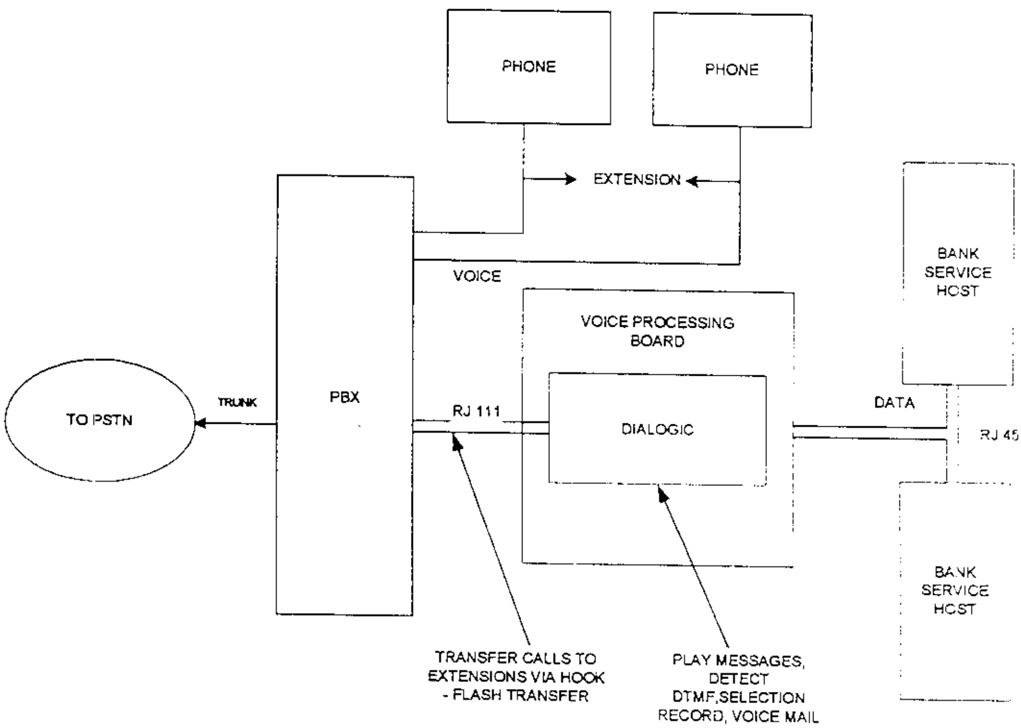
4. SYSTEM DESIGN

4.1 SYSTEM ARCHITRECTURE

The main components include:

- Private Branch eXchange (PBX).
- An Interactive Voice Response System (IVR).
- A Local Area Network of computers and a telephone connected to each computer.
- The communication Middle–Ware that fetches information from the main server of the bank (host) to the call center component that makes a request for such information.

COMPONENTS OF IVR PHONE BANKING



Private Branch Exchange (PBX)

This is basically a switch that is used to connect the telephone lines of the call center to the IVR. The PBX by itself has an Analog to Digital converter thereby converting the analog signals from the telephone line to the digital signals for the IVR system. The PBX has a component called AUTOMATIC CHANNEL DISTRIBUTOR (ACD). This is used to multiplex many Analog lines to a single Digital line for efficiency reasons. Another application of the ACD is that it connects the caller to the entry point of the call center.

Interactive Voice Response System

It is the Backbone of the whole system. It acts as an interface between the information sources-the host had the information seeker. An IVR system is typically a separate server that contains digital signal processing (DSP) hardware that analyzes and reproduce voice patterns. The IVR server interacts with a phone system through a dedicated connection. The way in which the IVR server is connected to the phone system depends on its capabilities and physical connection. An IVR can be connected to a key system, PBX, or other type of phone switch through analog ports, digital ports, and even a LAN or WAN connection.

It consists of a Voice Processing Board (VPB), which hosts a Dialogic Card. The Dialogic card is connected to the ISA/PCI slot of the computer. The PBX is connected to the VPB via RJ-111cables for voice transmission and the Dialogic card is connected to the Bank host through RJ-45 cables for data transmission.

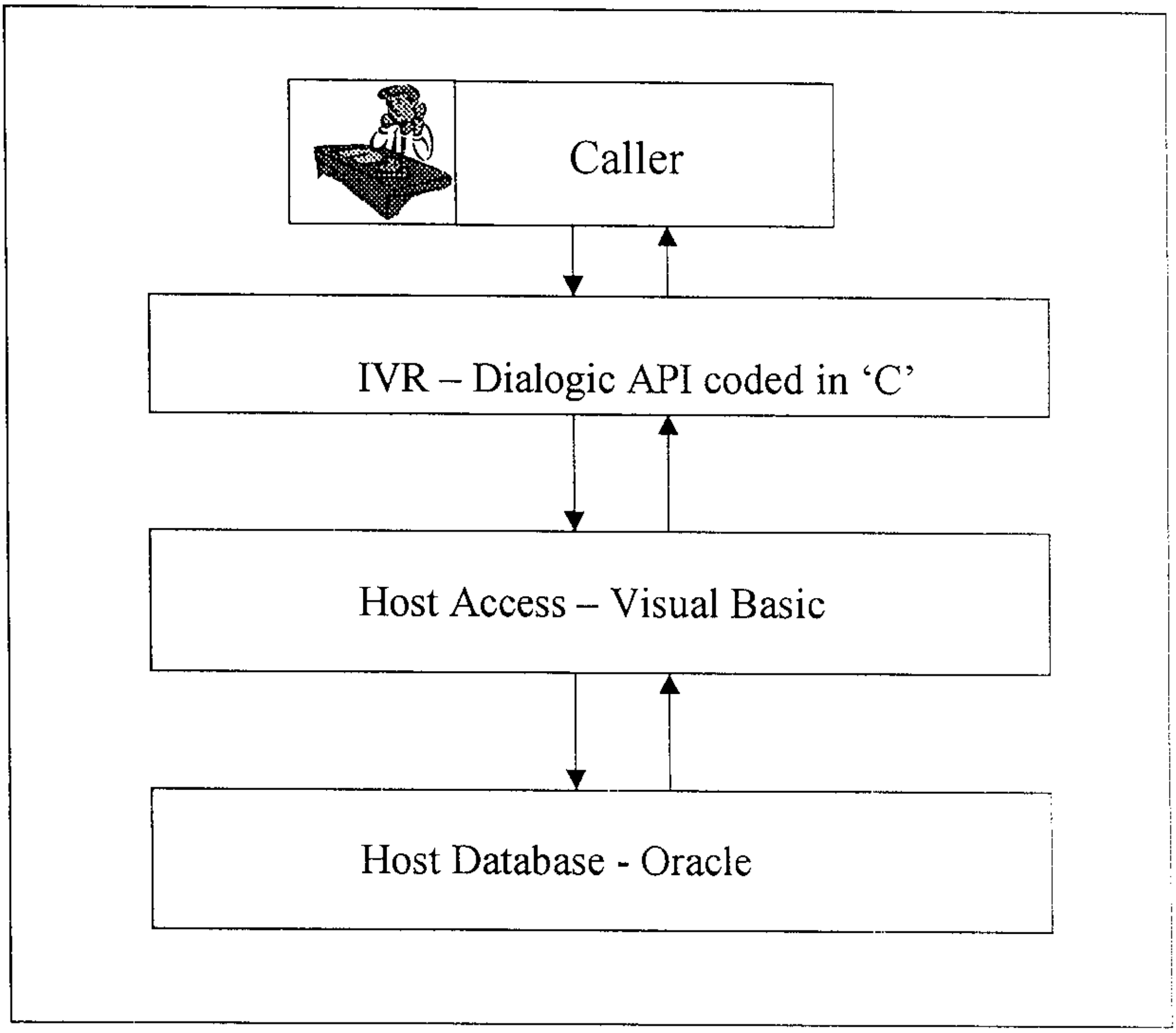
The dialogic card is the most important component of the voice processing system that performs voice to data and data to voice conversion.

The four-line D/41H^{TM} board and its two-line version, the D/21H^{TM} board, are ideal for applications that need high-performance voice processing but don't require the large-scale system sophistication. The D/21H and D/41H boards use same dialogic Application Programming Interface (API).Even the weakest of telephone signals traveling over difficult telephone lines can be recorded and played back with complete clarity.

The D/21H and D/41H boards use the latest Digital Signal Processor(DSP) voice processing technology, making them ideal for small and medium sized, server based computer telephony system – particular under Windows operating system.

International caller ID is supported on the d/21H and d/41 board, letting an application such as IVR calling party information via a telephone trunk line.

4.2. APPLICATION ARCITECTURE



User Categories

The Users can be categorized into:

- Customers
    - Account Holders

They can access general Information about the bank , loans, accounts and also make their account related enquiries and transactions.

- Non-Customers

They can access only the general Information part of the application.

- Bank Administrators

The Admin part of the project is handled by the Bank administrators.

4.3 TABLE DESIGN

Password Table:

Name	Type	Description
CUS_ID	NUMBER(9)	Unique Customer ID
TPIN	NUMBER(4)	TPIN ID

Customer Information Table :

Name	Data Type	Description
CUS_ID	NUMBER (9)	Unique Customer ID
CUS_NAME	VARCHAR2 (10)	Customer Name
CUS_ADD	VARCHAR2 (20)	Customer Address
CUS_PH	VARCHAR2 (10)	Customer Phone No
DOFB	DATE	Date of Birth
OCCU	VARCHAR2 (10)	Occupation
NOM_NAME	VARCHAR2 (10)	Nominee's Name
NOM_ADD	VARCHAR2 (20)	Nominee's Address

Transaction Table :

Name	Data Type	Description
CUS_ID	NUMBER (9)	Unique Customer ID
ACC_NO	NUMBER (10)	Account Number
DOFTRANS	DATE	Date of Transaction
TYPE_TRANS	VARCHAR2 (10)	Type of Transaction
AMOUNT	NUMBER (10,2)	Cash Amount

Cheque Transaction Table :

Name	Data Type	Description
ACC_NO	NUMBER (10)	Account Number
CHQ_NO	NUMBER (10)	Cheque Number
DOFISSUE	DATE	Date of Issue
AMOUNT	NUMBER (10,2)	Amount For

Cheque Book Request Table :

Name	Data Type	Description
ACC_NO	NUMBER (10)	Account Number
DOFREQ	DATE	Date of Request
STATUS	CHAR (1)	Issued Status
DOFISSUE	DATE	Date of Issue

Stop Cheque Payment Table :

Name	Data Type	Description
ACC_NO	NUMBER (10)	Account Number
CHQNO	NUMBER (10)	Cheque Number
DOFREQ	DATE	Date of Request

ACCOUNTS TABLES

Savings Account Table :

Name	Data Type	Description
CUS_ID	NUMBER (9)	Unique Customer ID
ACC_NO	NUMBER (10)	Account Number
CUR_BAL	NUMBER (10,2)	Current Balance

Recurring Account Tables :

Name	Data Type	Description
CUS_ID	NUMBER (9)	Unique Customer ID
ACC_NO	NUMBER (10)	Account Number
CUR_BAL	NUMBER (10,2)	Current Balance

Current Account Table :

Name	Data Type	Description
CUS_ID	NUMBER (9)	Unique Customer ID
ACC_NO	NUMBER (10)	Account Number
CUR_BAL	NUMBER (10,2)	Current Balance

NRI Account Table :

Name	Data Type	Description
CUS_ID	NUMBER (9)	Unique Customer ID
ACC_NO	NUMBER (10)	Account Number
CUR_BAL	NUMBER (10,2)	Current Balance

4.4 NORMALIZATION

The theory behind the arrangement of attributes into tables to ensure that basic manipulation operations on such tables do not cause data inconsistency problem is known as the normalization theory.

The normalization of data helps to ensure that a data design takes care of:

- ✓ Minimum of duplication of data.
- ✓ Elimination of data inconsistency problem.
- ✓ Providing flexibility to support different functional requirement.

✓ Enabling the model to be translated to database design.

The first stage in normalization is to reduce the data to its first normal form, by removing repeating items and showing them as separate record but including in them the key of the original record.

The next stage of reduction to the second normal form is to check that all item in each record are entirely dependent on the key of the record but on another data item, then it is removed with its key to form another record. This is done until each record contain data item, which are entirely dependent on the key of their record.

This final stage of the analysis, the reduction to third normal form, involves examining each record to see whether any item is mutually dependent. If there are any, then they are removed to a separate record leaving one of the items behind in the original record and using that as the key in the newly created record.

4.5 MODULE DESCRIPTION

The project involves two modules:

- ✓ ADMIN MODULE
- ✓ IVR MODULE

ADMIN MODULE:

The Admin module part is done with ***Visual Basic*** as the front-end and ***Oracle*** as the database back-end.

➢ This module is basically a portal, through which the Administrator can enter the requisite information, which can be stored in the database.

➢ The **Admin** module generates reports that are required by the administrator.

- The report generated uses the Oracle database as a "data mine", through which the administrator can mine data useful for the bank.

The following basic transactions are performed:

- Adding new customers and maintaining their information
- Creating accounts for customers
- Editing and deleting TPIN(Telephone Personnel Identification Number)

The following account transactions are also performed:

- Maintaining different accounts such as
  - Savings Account
  - Current Account
  - Recurring Account
  - NRI Account
- Deposit of amount in each account
- Withdrawal calculations and Balance Amount processing

The Admin module also provides cheque services:

- Cheque processing
- Issue of Cheque
- Stop payment of Cheque

The Reports are generated using Data Report which includes:

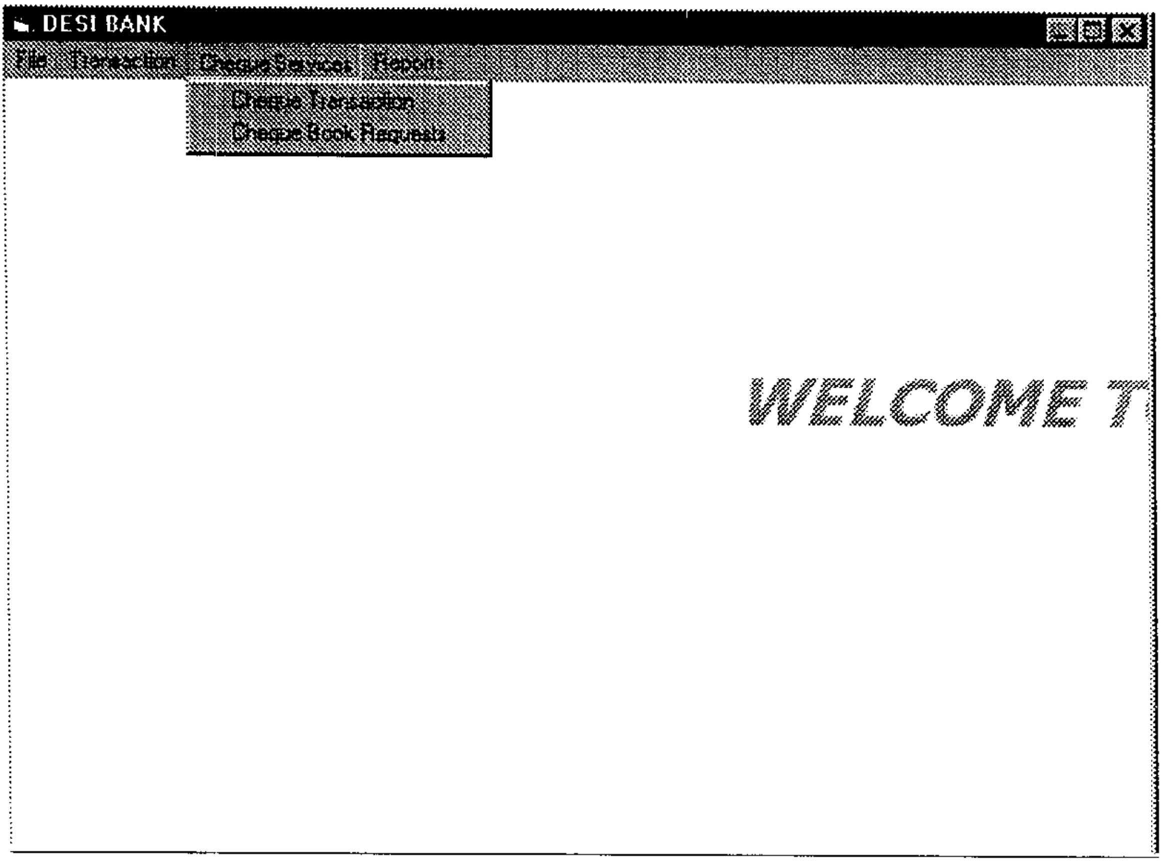
- Day to day transaction report
- Number of calls received profile
- Customer information
- Account details of customers
- Customer's Cheque details

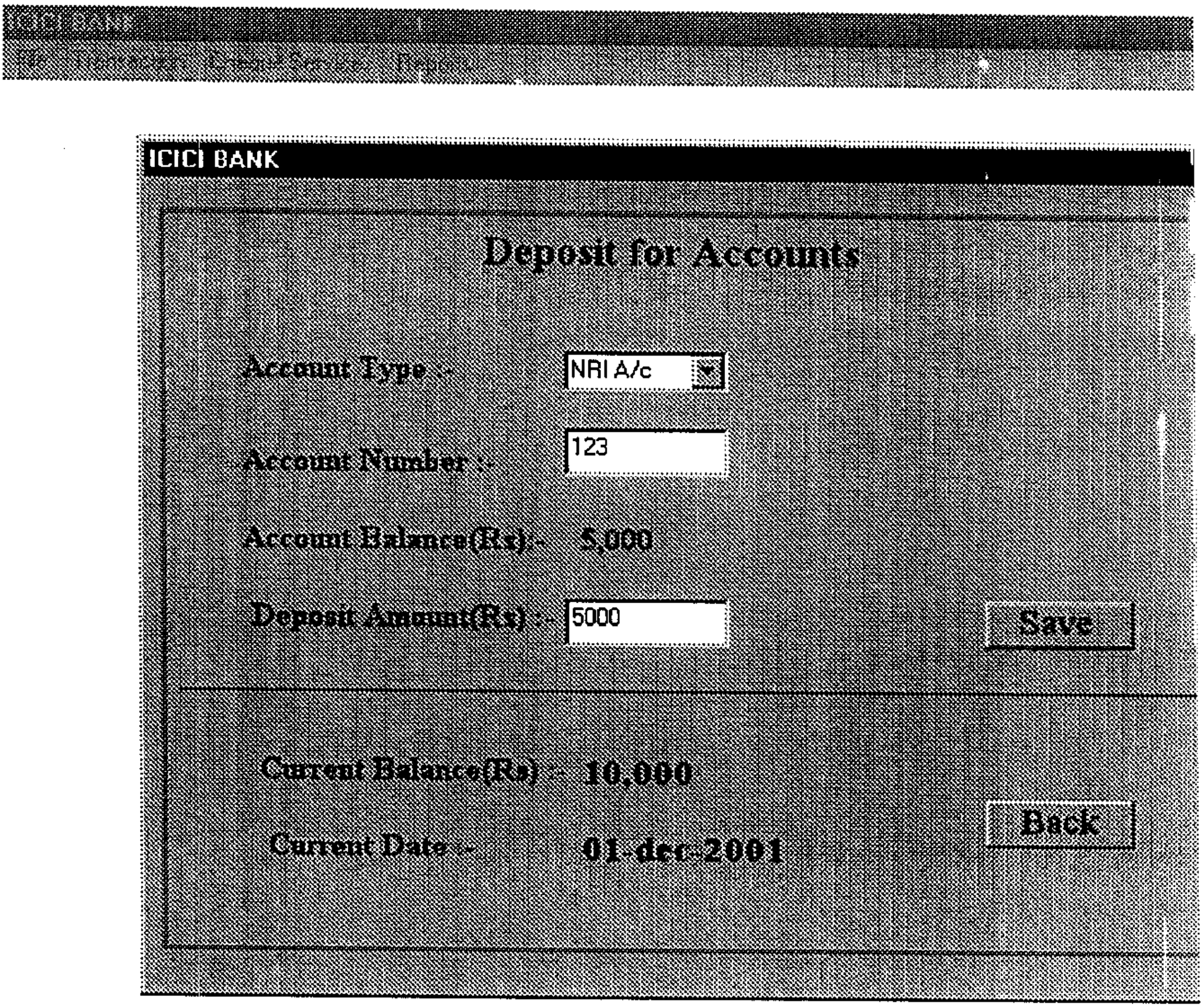
The **IVR** module is done using C in Visual C++ Environment, since it provides the flexibility of working in assembly level as well as high level. Because of its machine level interaction, C++ coding is still a force to reckon with. The **IVR** module is coded for interaction with incoming calls and how to go about them. The **IVR** module interacts with the database when the user specifies his account number and retrieves the account balance and gives it to the user in words (or as voice).

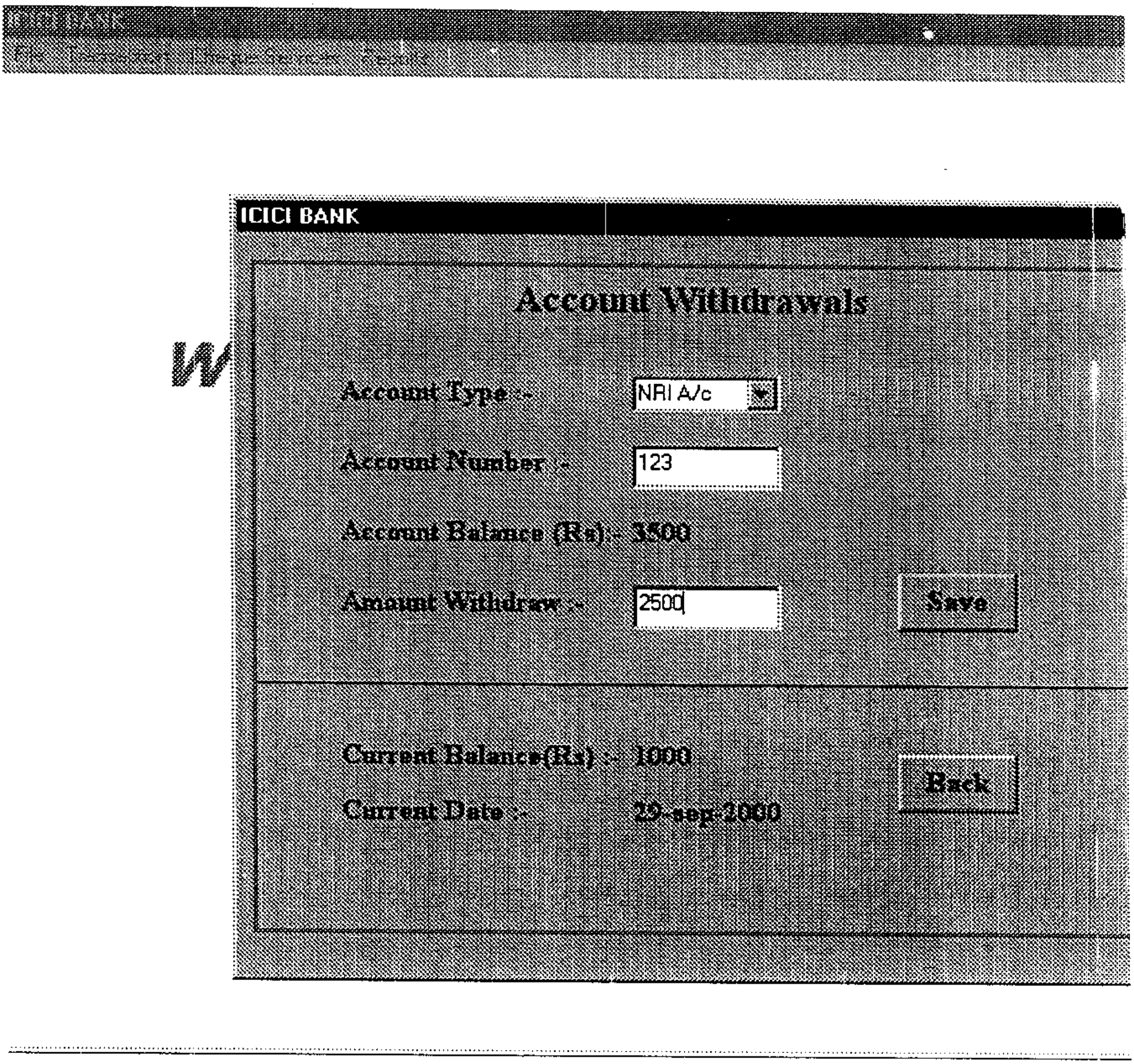
The IVR module provides the following:

- ✓ Services:
  - o Quick Balance
  - o Different Accounts Information
  - o Cheque related Services
- ✓ General Information about the Bank
- ✓ Product Information such as the loans Available, Exchange rates and interest rates details

4.6 SAMPLE SCREENS




ICICI BANK
Deposit for Accounts
Account Type :-
NRI A/c
Account Number :-
123
Account Balance(Rs):-
5,000
Deposit Amount(Rs) :-
5000
Save
Current Balance(Rs) :-
10,000
Current Date :-
01-dec-2001
Back


ICICI BANK
Account Withdrawals
Account Type :-
NRI A/c
Account Number :-
123
Account Balance (Rs):- 3500
Amount Withdraw :-
2500
Save
Current Balance(Rs) :- 1000
Back
Current Date :-
29-sep-2000

Design Translation Process…

5. CODING STANDARDS

Coding style encompasses a coding philosophy that stresses simplicity and clarity. It is an important attribute of source code and can determine the intelligible of a program. The elements of style include internal documentation, methods for data declaration, procedures for statement construction and I/O coding techniques.

Internal documentation of source code begins with the selection of identifier (variables and labels) names, continues with the placement and composition of commenting and concludes with the visual organization of the program.

The complexity and organization of data structure are defined during the design step. The style of data declaration is established when code is generated. The order of data declarations should be standardized, even if the programming language has no mandatory requirements. Ordering makes attributes easier to find, expanding testing, debugging and maintenance.

The construction of logical flow is established during design. Statement construction should abide by one overriding rule. Each statement should be simple and direct; code should not be convoluted to effect efficiency.

The style of input and output is established during software requirements analysis and design, not coding. The manner in which I/O is implemented can be the determining characteristic for system acceptance by the user community. Input and output style will vary with the degree of human interaction.

IVR- Call Flow has been designed in such a way that any future changes can be into the system with minimal changes in coding thereby maintaining the entire system in a easier manner.

System Testing…

Bugging and Error correction...

6. TESTING AND CORRECTION

Testing:

Testing is a process of executing a program with the intent of finding an error. A good case is one that has a high probability of finding an as yet undiscovered error. A successful test is one that uncovers an as yet undiscovered error. Testing is a set of activities that can be planned in advance and conducted systematically. Testing begins at the module level and works "outward" toward the integration of the entire computer based system. Testing and debugging are different activities, but debugging must be accommodated in any testing strategy.

Software testing is one element of a broader topic that is often referred to as verification and validation. Verification refers to the set of activities that ensure that software correctly implements a specific function. Validation refers to a different set of activities that ensure that the software that has been built is traceable to requirements. Testing provides the last bastion from which quality can be assessed and, more programmatically errors can be uncovered.

After the system is put together, system testing is performed. Here the system is tested against the system requirements to see if all the requirements are met and the system performs as specified by requirements. The entire software is tested and the goal is to se if the software meets the requirements.

Unit Testing

Unit testing concentrates on each unit of the software as implemented in the source code. Initially tests focus on each module individually, ensuring that it functions properly as a unit. Hence, the name UNIT TESTING. Unit testing makes heavy use of white box testing techniques, exercising specific paths in a module's control structure to ensure complete coverage and maximum error detection. Next, modules must be assembled or integrated to form the complete software package.

Unit testing focuses verification effort on the smallest unit of software design the module. The goal of unit testing is to test modules or "units" and not the entire software system. The programmer most often does unit testing, the programmer, after finishing the coding of a module, tests it with some data. The tested module is then delivered for system integration and further testing. Using the detail design description as a guide, important control paths are tested to uncover errors within the boundary of the module. The relative complexity of tests and the errors detected as a result is limited by the constrained scope established for unit testing. The unit test can be conducted in parallel for multiple modules.

The test that occur as part of unit testing are illustrated below:

- Interface
- Local Data Structures
- Boundary Conditions
- Independent paths
- Error-handling paths

The module interface is tested to ensure that information properly flows into and out of the program unit under test. The local data structure is examined to ensure that data stored temporarily maintains its integrity during all steps in an algorithm's execution. Boundary conditions are tested to ensure that the module operates properly at boundaries established to limit or restrict processing. All independent paths through the control structures are exercised to ensure that all statements in a module have been executed at least once. And finally, all error-handling paths are tested.

Integration Testing

When all the modules and sub-modules designed are integrated together to form the system in its entirety, integration testing is made to check the degree to which the software system is designed to satisfy the user's requirement specification.

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit-tested modules and build a program structure that has been dictated by the design.

The common types of integration problems looked for were:

- Version mismatches.
- I/o format or protocol mismatches.
- Conflicting data views or usage, especially global variables.
- Data integrity violations.
- Wrong call order or wrong type of parameters in the function calls.
- Missing or overlapping functions.
- Resource problems especially in memory handling.

Validation testing

This testing concentrates on confirming that the software is error-free in all respects. All the specified validations are verified and the software is subjected to hard-core testing. It also aims at determining the degree of deviation that exists in the software designed from the specification of the user. If there exists any deviation from the user's specification, they are listed out and are corrected.

System Testing

This testing is a series of different tests whose primary purpose is to fully exercise the computer-based system. This involves:

- ✓ Implementing the system in a stimulated production environment and testing it.
- ✓ Introducing errors and testing for error handling.

The series of tests here are as follows:

Recovery Testing

- Simulating disaster and testing if the system behaves gracefully.
- Testing for consistency of the database after recovery.

Stress Testing

- Testing for abnormal shutdown.
- Testing the effect of extended operation.

Volume Testing

- Applying large volumes of data and checking for correctness.

Security Testing

- Testing for security risk factors unauthorized access of system.

The IVR-Tele Banking (Call Flow) module was tested extensively in all these areas. After all the testing procedures are completed, the software becomes a perfect system free of logical errors and becomes capable of handling all the requirements specified by the user.

Installation ...

7. INSTALLATION OF THE SYSTEM

System Implementation

After the testing and debugging of the software we need to implement it. This will be found easier and more efficient because implementation has no obstacles since we have already undergone debugging. This involves user training system testing and successful running of the developed proposed system. Implementation is the process of converting a new or received system design into an operational one. It is the key stage in achieving a successful new system because; usually it involves a lot of upheaval in the user department. It must therefore be carefully planned and controlled. Apart from planning the two major tasks of preparing for implementation are education and training of users. Education of users should take place much earlier in the project. Training has to be given to the staff regarding the new system. Once staff has been trained the system can be tested.

User Training

After the system is implemented successfully, training of the user is one of the most important sub tasks of the developer. For this purpose user system manuals are prepared and handed over to the user to operate the developed system. Here the users are trained to operate the developed system.

Maintenance

Maintenance is about changes done to the existing software products. Most such changes are enhancements, improvements to the functionality of the software as opposed to repair work to fix errors.

8. CONCLUSION

8.1 BENEFITS

The IVR-Tele Banking system, using INTEL dialogic card is one among the pioneering efforts to offer new level of performances using computer telephony networks. Text-to-Speech concept used here represents a price – performance breakthrough, enabling telephony application developers to offer leading edge functionality such as unified messaging and text-based Interactive Voice Response (IVR) to the small system market at an attractive price. Speech technologies enhance the value of telephony boards, giving developers more functionality at cost effective price points.

The IVR-Admin, which is an integral part of this application, gives the application an edge over other telephony applications, which are devoid of an administrator module. Administrator is vital for monitoring and reporting the day-to-day status of the application. Without Admin the management will find itself in no man's land as far as the application is concerned. Administration is the remote, which remotes the IVR-Tele Banking at its will.

The Recording Tool, which enables the dynamic recording of "vox" files, is a blessing in disguise for the bank, which is going to incorporate this application.

8.2 LIMITATIONS

- This application is not meant for online transactions.
- The security aspect of this application does come under scrutiny.
- Exclusive training in Dialogic software must be given to the administrator before this application can be deployed in the bank.

8.3 ENHANCEMENT

The system has been designed in such a way that any future changes can be brought into the system with minimal changes in coding thereby maintaining the entire system in easier manner. This application can be extended for Internet Telephony. The admin module can be extended to have campaign management as one of its features.

BIBLIOGRAPHY

- Robert Lafore , *Programming In "C"* – Sams Publishing

- Noel Jerke , *Visual Basic 6.0*, The Complete Reference –, Tata McGraw-Hill Publishing company

- Building User-Friendly Voice Systems, a Computer Telephony and Telemedia Conference organized by Housley Communications in September 1995.

- Dialogic API toolkit

- www.dialogic.com

APPENDICES

• VOX Files

Dialogic has established a family of sound file formats for call processing hardware known as VOX (from the Latin word "vox", meaning "voice"). VOX is not compatible with the Windows .WAV format, however Parity Software offers a product called VOX Studio which can convert between VOX and Wave files and also record and play VOX files on Wave devices.

A digitized audio recording is characterized by a number of parameters:

- The number of bits used to store a *sample*. A sample is a binary number specifying the amplitude (loudness) of the sound at the instant a measurement was taken. Sample sizes will generally be 4-, 8- or 16-bit.

- The number of samples recorded per second, the *sampling rate*, which is usually measured in thousands of samples per second, kHz. Windows Wave files typically use 11 kHz or 22 kHz, VOX files use 6 kHz or 8 kHz.

- The compression scheme used, if any. There is a family of schemes known as ADPCM (Adaptive Differential Pulse Code Modulation), which is commonly used in voice processing hardware. ADPCM is designed especially for encoding the human voice. Files recorded with PCM (Pulse Code Modulation) are not compressed.

- *Companding* scale. Companding is a technique to squeeze a bigger range out of a given number of bits. The human ear is more sensitive to small changes in amplitude when the volume is low, companding exploits this effect by using a non-linear mapping of sample values (the binary number used to store the sample) to sample sizes (the absolute value of the amplitude specified by the sample). A file format is referred to as *linear* if no companding is used. There are two common companding scales in use in computer telephony: the mu-law (μ-law) scale which is used internally by the digital telephone network in North America, and the A-law scale, which is used in most other countries.

Different manufacturers have different flavors of ADPCM, so be aware that a file recorded using equipment from one vendor may not work on another vendor's board.

Dialogic boards offer the following VOX file format options:

- 4-bit ADPCM or 8-bit PCM sampling
- 6 KHz or 8 KHz sampling rate
- A-law or mu-law companding (applies to PCM only: ADPCM expands to linear PCM and is therefore not companded)

This gives a total of 6 different VOX file data types. The sampling rates of these types are as follows.

Bits/ sample	**Sample type**	**Sampling rate**	**Kilobits/ second**	**Kilobytes/ minute**	**Megabytes/ hour**
4	ADPCM	6 KHz	24 Kbps	180 Kb	10.8 Mb
4	ADPCM	8 KHz	32 Kbps	240 Kb	14.4 Mb
8	Alaw	6 KHz	48 Kbps	360 Kb	21.6 Mb
8	Mulaw	6 KHz	48 Kbps	360 Kb	21.6 Mb
8	Alaw	8 KHz	64 Kbps	480 Kb	28.8 Mb
8	Mulaw	8 KHz	64 Kbps	480 Kb	28.8 Mb

Note: the above table uses Kb and Mb in the old-fashioned metric sense of Kilo and Mega, i.e. 1,000 and 1,000,000 respectively. Disk storage is often measured in Kb=1024 bytes and Mb=1,048,576 bytes. The difference is a little more than 2% and usually not important. Note also that Kbps means Kilobits per second, not Kilobytes per second.

The public telephone network uses 8 kHz 8-bit PCM encoding identical to the 64 Kbps VOX format, so this can give quality as good as theoretically possible in digital transmission using the network. There is little point in providing a higher quality format today, although future applications using the SC bus will be able to combine several PCM channels into single, high-fidelity channels which could be used, for example, to carry CD-quality sound or video. This type of application will use hardware which has not progressed beyond the early development stages at the time of writing.

The default format is the most compressed, and therefore the lowest quality: 4-bit ADPCM sampled at 6 KHz (24 Kbps).

Note that VOX file compression acts by reducing to the number of bits required to store a sample value by a fixed amount. All VOX file formats have a fixed transfer rate, i.e. number of bits per second, as shown in the above table. If you know how many bytes of VOX data you have, you know exactly how long it will take to play that data. (Most compression algorithms do not have this feature: a program like PKZIP will compress at varying efficiency depending on the content of the file, newcomers to call processing therefore sometimes expect that the transfer rate of VOX files might vary, but this is not the case).

There is a trade-off between sound quality and the resources required to run your application. The best quality VOX format requires almost three times as much disk space and three times as much data throughput from disk storage as does the default format. Using the best quality format may limit the number of lines which can be run successfully, particularly if the VOX files are located on a network file server and must therefore be transported across a LAN.

A good compromise may be to record your system prompts in the best format (64 Kbps) and store them in a RAM drive so that they can be delivered very efficiently to the voice card. New messages from callers could be recorded in the most compressed format (24 Kbps) to save disk space and disk/network load.

There is a major weakness of VOX files. The only data in a VOX file is raw sample data; there is no file header to allow application software such as VOS to determine the exact flavor of VOX file, or indeed to allow an application to distinguish between a VOX file and any other type of file. This is in contrast to Wave files, which include a header that specifies the sample size and sampling rate of the file.

Dialogic boards are not designed for high-fidelity recording. However, you can use a multi-media PC with a sound card, microphone and Wave file editor to create high-quality Wave files, then convert them to VOX for use on a Dialogic. This can produce results that are noticeably better than recordings made directly on a Dialogic card. Parity Software's VOX Studio product offers all the software needed to do this plus a CD-ROM with more than 300 sound effects in 8-bit and 16-bit Wave format which can be converted to VOX. This will allow you to produce professional prompts for your system using desktop PC technology rather than an expensive digital recording studio.

You will probably use voice editor software if you need to manipulate VOX or Wave files. If for some reason you need to do your own cutting and pasting of VOX files you should be aware that there is a pitfall for the unwary. PCM data can be cut and pasted without problems, but ADPCM files are different because of the compression scheme. To interpret ADPCM data you must begin at the start of the file and keep track of a scale factor used to determine the sample value. If a section is cut from the middle of an ADPCM file and added to another file, the data must be adjusted for the current scale factor at the point where the data is added. VOX file editors, of course, know how to do this automatically for you.

Technical Specifications

Number of ports	4
Maximum boards/system	16
Analog network interface	On-board loop start interface circuits
Microprocessor	Intel® 80C188
Digital signal processor	Motorola* DSP56002
HOST INTERFACE	
Bus compatibility	PCI (complies with PCISIG Bus Specification, Rev. 2.1)
PCI bus speed	33 MHz
Shared memory	8 KB page, PnP selectable on 16 KB boundaries
Base addresses	Selected by PCI BIOS
Interrupt level	One IRQ (IntA) shared by all boards
TELEPHONE INTERFACE	
Trunk type	Loop start (or ground start for answer only)
Impedance	600 Ohm for D/4PCI. Matching complex impedance specified in CTR-21 for D/4PCI-EURO.
Ring detection	25 Vrms min., 15.3 Hz to 68 Hz, 150 Vrms max.
Loop current range	20 mA to 120 mA, DC (polarity insensitive), D/4PCI-EURO current limits at 60 mA per CTR-21 specifications
Crosstalk coupling	-80 dB at 3 kHz channel-to-channel
Frequency response	300 Hz to 3400 Hz ±3 dB (transmit and receive)
Connector	Four RJ-11

ENVIRONMENTAL REQUIREMENTS	
+5 VDC	650 mA
+12 VDC	55 mA
−12 VDC	53 mA
Operating temperature	0°C to +50°C
Storage temperature	−20°C to +70°C
Humidity	8% to 80% noncondensing
Form factor	PC AT (PCI); 6.9 in. long, 0.75 in. wide, 3.85 in. high (excluding edge connector)

AUDIO SIGNAL	
Receive range	-50 dBm to -13 dBm (nominal), for average speech signals‡ configurable by parameter†
Automatic gain control	Application can enable/disable. Above −30 dBm results in full-scale recording, configurable by parameter.†
Silence detection	−40 dBm nominal, software adjustable†
Transmit level (weighted average)	−9 dBm nominal, configurable by parameter†
Transmit volume control	40 dB adjustment range, with application-definable increments

FREQUENCY RESPONSE	
24 Kb/s	300 Hz to 2600 Hz ±3 dB
32 Kb/s	300 Hz to 3400 Hz ±3 dB
48 Kb/s	300 Hz to 2600 Hz ±3 dB
64 Kb/s	300 Hz to 3400 Hz ±3 dB

AUDIO DIGITIZING	
24 Kb/s	ADPCM @ 6 kHz sampling
32 Kb/s	ADPCM @ 8 kHz sampling

48 Kb/s	µ-law PCM @ 6 kHz sampling
64 Kb/s	µ-law PCM @ 8 kHz sampling
Digitization selection	Selectable by application on function call-by-call basis
Playback speed control	Pitch controlled Available for 24 and 32 Kb/s data rates Adjustment range: ±50% Adjustable through application or programmable DTMF control
WAVE AUDIO	
	Supports 11 kHz linear PCM, 8-bit mono mode (available only when running Windows)
DTMF TONE DETECTION	
DTMF digits	0 to 9, *, #, A, B, C, D per Bellcore LSSGR Sec 6
Dynamic range	Programmable, default set at -36 dBm to +0 dBm per tone
Minimum tone duration	40 ms, can be increased with software configuration
Interdigit timing	Detects like digits with a 40 ms interdigit delay Detects different digits with a 0 ms interdigit delay.
Acceptable twist	10 dB
Signal/noise ratio	10 dB (referenced to lowest amplitude tone)
Noise tolerance	Meets Bellcore LSSGR Sec 6 and EIA 464 requirements for Gaussian, impulse, and power line noise tolerance
Cut-through	Detects down to -36 dBm per tone into 600 Ohm load impedance
GLOBAL TONE DETECTION	
Tone type	Programmable for single or dual
Max. number of tones	Application-dependent
Frequency range	Programmable within 300 Hz to 3500 Hz
Max. frequency deviation	Programmable in 5 Hz increments

Frequency resolution	Less than 5 Hz. Note: certain limitations exist for dual tones closer than 60 Hz apart.
Timing	Programmable cadence qualifier, in 10 ms increments
Dynamic range	Programmable, default set at –36 dBm0 to 0 dBm per tone
GLOBAL TONE GENERATION	
Tone type	Generate single or dual tones
Frequency range	Programmable within 200 Hz to 4000 Hz
Frequency resolution	1 Hz
Duration	10 ms increments
Amplitude	-43 dBm to -3 dBm per tone, programmable
MF SIGNALING	
MF digits	0 to 9, KP, ST, ST1, ST2, ST3 per Bellcore LSSGR Sec 6, TR-NWT-000506 and CCITT Q.321
Transmit level	Complies with Bellcore LSSGR Sec 6, TR-NWT-000506
Signaling mechanism	Complies with Bellcore LSSGR Sec 6, TR-NWT-000506
Dynamic range for detection	–25 dBm to –1 dBm per tone
Acceptable twist	6 dB
Acceptable frequency variation	Less than ±1 Hz
CALL PROGRESS ANALYSIS	
Busy tone detection	Default setting designed to detect 74 out of 76 unique busy/congestion tones used in 97 countries as specified by CCITT Rec. E., Suppl. #2. Default uses both frequency and cadence detection. Application can select frequency only for faster detection in specific environments.
Ring back detection	Default setting designed to detect 83 out of 87 unique ring back tones used in 96 countries as specified by CCITT Rec. E., Suppl. #2. Uses both frequency and cadence detection.

Positive voice detection accuracy	>98% based on tests on a database of real world calls
Positive voice detection speed	Detects voice in as little as 1/10th of a second
Positive answering machine detection accuracy	80% to 90% based on application and environment
Fax/modem detection	Pre-programmed
Intercept detection	Detects entire sequence of the North American tri-tone. Other SIT sequences can be programmed.
Dial tone detection before dialing	Application enable/disable Supports up to three different user-definable dial tones Programmable dial tone drop out debouncing
TONE DIALING	
DTMF digits	0 to 9, *, #, A, B, C, D per Bellcore LSSGR Sec 6, TR-NWT-000506
MF digits	0 to 9, KP, ST, ST1, ST2, ST3
Frequency variation	±0.5% of nominal frequency
Rate	10 digits/s maximum, configurable by parameter†
Level	–5 dBm per tone, nominal, configurable by parameter†
PULSE DIALING	
10 digits	0 to 9
Pulsing rate	10 pulses/s, nominal, configurable by parameter†
Break ratio	60% nominal, configurable by parameter†
ANALOG CALLER IDENTIFICATION	
Applicable standards	Bellcore TR-TSY-000030 Bellcore TR-TSY-000031 TAS T5 PSTN1 ACLIP: 1994 (Singapore) British Telecom SIN 242 (Issue 01) British Telecom SIN 227 (Issue 01) Japan NTT CLIP

Modem standard	Bell 202 or V.23, serial 1200 b/s (simplex FSK signaling)
Receive sensitivity	-48 dBm to -1 dBm
Noise tolerance	Minimum 18 dB SNR over 0 dBm to -48 dBm dynamic range for error-free performance
Data formats	Single Data Message (SDM) and Multiple Data Message (MDM) formats via API calls and commands
Line impedance	600 Ohm for D/4PCI. Matching complex impedance specified in CTR-21 for D/4PCI-Euro.
Message formats	ASCII or binary SDM, MDM message content
ANALOG DISPLAY SERVICES INTERFACE (ADSI)	
	FSK generation per Bellcore TR-NWT-000030. CAS tone generation and DTMF detection per Bellcore TR-NWT-001273.

* All specifications are subject to change without notice.

† Analog levels: 0 dBm0 corresponds to a level of +3 dBm at tip-ring analog point. Values vary depending on country requirements; contact your account manager.

‡ Average speech mandates +16 dB peaks above average and preserves -13 dB valleys below average.