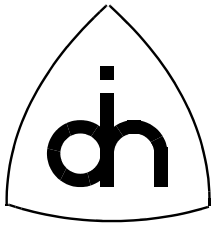
Odin TeleSystems Inc.



OTX ISDN API

Guide for Programmers

Doc. No. 1412-1-SCA-1003-1

Rev. 1.0-P2

January 16, 2004

Odin TeleSystems Inc.
http://www.OdinTS.com

This document is published by:
Odin TeleSystems Inc.
800 East Campbell Road, Suite 334
Richardson, Texas 75081-1873
U. S. A.

Printed in U. S. A.

1. Table of Content

2. Introduction

This document describes the structure of OTX (Odin Telecom FrameworX) ISDN API. Currently, Call Control for Primary Rate Interface (PRI: consisting 23 64kpbs B-Channel for service and one D-Channel for transmitting control information) is supported. The OTX ISDN Call Control (OtxCc) API provides high-level call control interface functions for service, for example, voice. This document introduces major functions for this C language API, for example, connecting a call and disconnecting a call.

To write a OTX ISDN application, OTX Hardware API is also needed, for OTX driver lever API functions, refer to <<Programmer's Guide for OTX Hardware API>> (Doc. No. 1111-1-SCA-100201).

3. Architecture

3.1 Overview

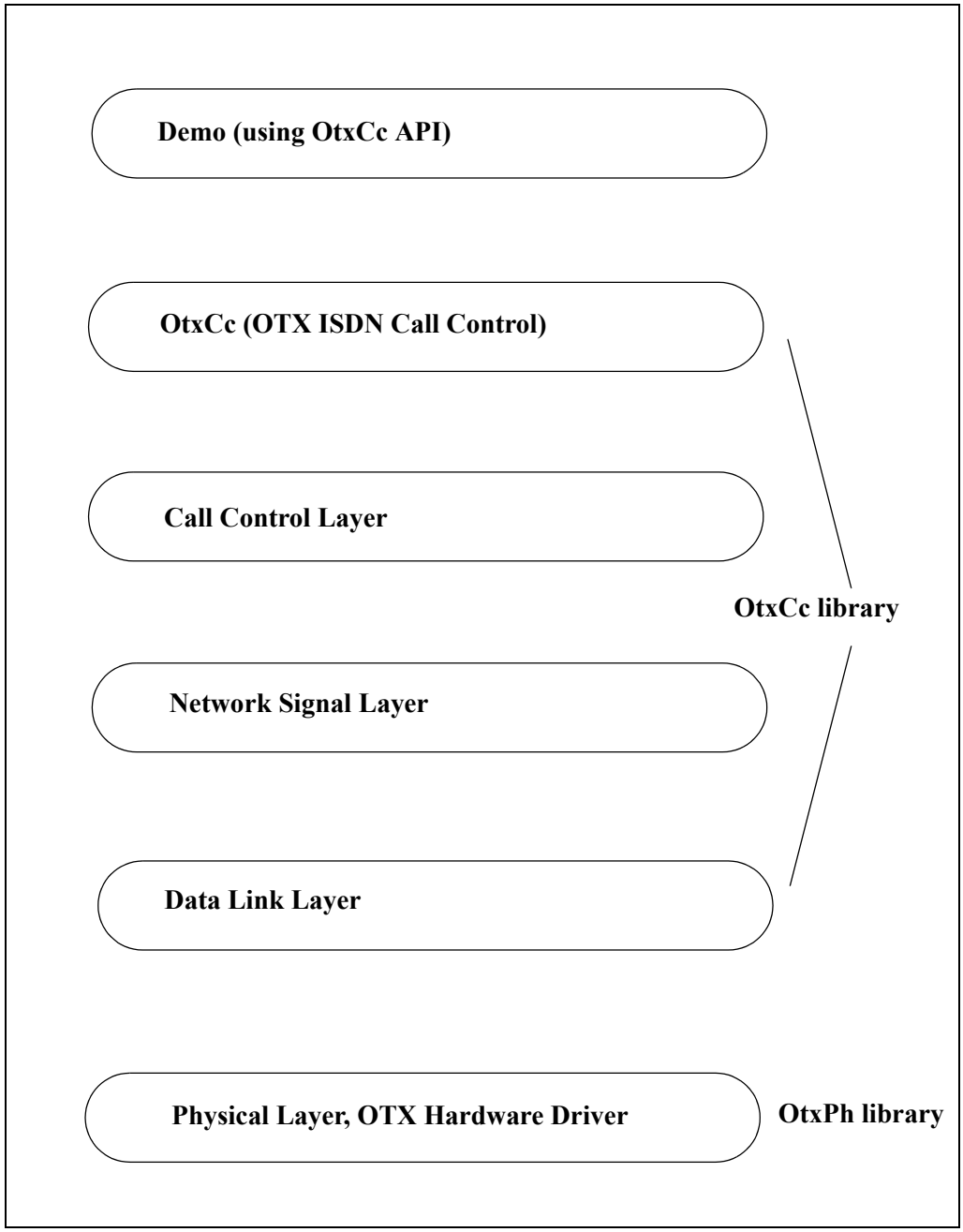


Figure 1. Architecture Overview

OTX ISDN structure is similar to the model of Open Systems Interconnection (OSI), this structure comprises physical layer, data link layer, network layer and call control layer, OtxCc provides interface for call control layer.

3.2 Structure of software:

Three layers of software structure are categorized: library OtxPh.lib, OtxCcPri.lib and application (demo). OtxCcPri.lib is the library which can be accessed by Call Control API, demo program shows how to use OtxCc API to develop applications.

3.2.1 OtxPh.lib

includes physical layer and OTX hardware driver API

3.2.2 OtxCcPri.lib

includes Call Control Layer, Networking Signal Layer and Data Link Layer

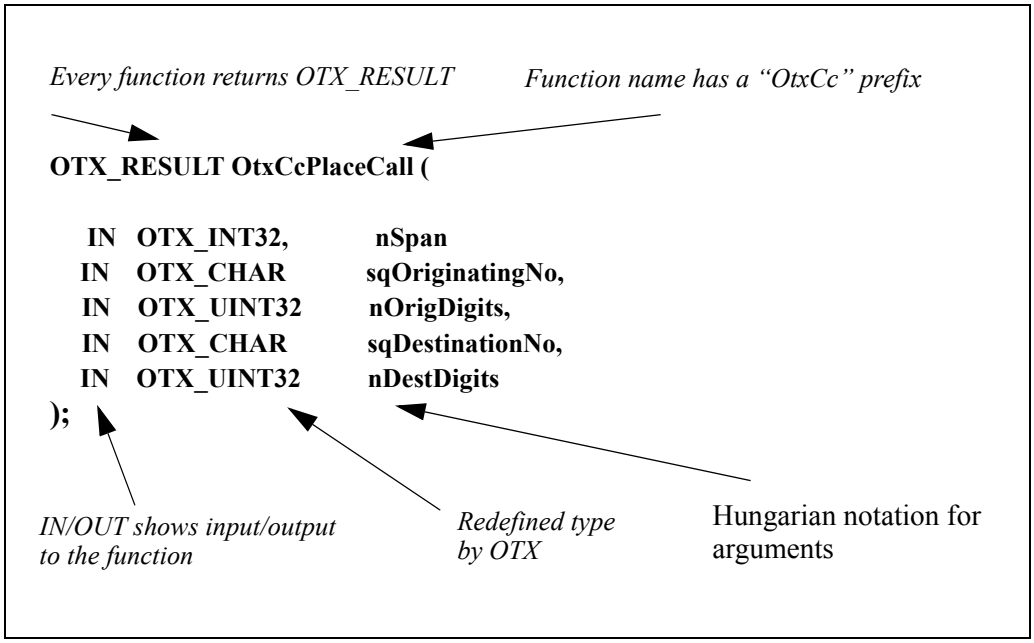
3.2.3 Application (Demo)

Demo program shows the flow of using OtxCc API, it is an event driven, multiple threads program.

4. Call Control API

4.1 API Coding Convention

All functions, data types and macros within OtxCc follow a naming convention. All names have a prefix “OtxCc’ for Odin Telecom frameworX Call Control. Example 1.



Example 1. OTX ISDN API Coding Convention.

As shown in Example 1, all types have been redefined to facilitate portability. For example, a 32-bit integer type is called *OTX_INT32* in the API. Hungarian notation is also used in the API, for detailed type redefinition and hungarian notation, refer to <<Programmer's Guide for OTX Hardware API>> (Doc. No. 1111-1-SCA-100201).

4.2 Data Structure

For each call, it has some elements, for example, Span ID, Connection ID, calling number, called number, the detail structure is in Example 2.

```
typedef struct OtxCcCallElement {

  OTX_HANDLE  hCallHandle;
  OTX_CHAR     *pszCallNo;

  OTX_INT32       nSpan;
  OTX_INT32      nChannel;

 OtxCcCallStateE eCallState;
  OtxCcInbandSignalE eReturnSignal;
} OtxCcCallElement;
```

Example 2. Call Element Structure

4.3 API Function

Most API calls have parameters nSpan and nConn_id, nSpan refers to the identification of span (T1/E1 line), nConn_id refers to the Connection ID of the call.

4.3.1 Open and Initialize Service

Following are OtxCc API functions to open and initialize OTX hardware driver, and ISDN call control service: OtxCcInit(), OtxCcConnect(), OtxCcOpenService() and OtxCcInitService().

Major functions to setup a connection between caller and callee are introduced as followings:

4.3.2 Make Call

OtxCcPlaceCall(): place a call with calling number and called number

4.3.3 Disconnect Call

OtxCcDisconnectCall(): release a call

4.3.4 Answer Call

OtxCcAnswerCall(): when an incoming call is detected, it answers the call

4.4 Message Description

Communication messages between callee and caller are provided to indicate the processing of a call, Span Identification Number (ID), B-channel ID and Connection ID are provided with the message.

Some messages have Connection ID, some don't. Followings are major messages, the whole set of messages, refer to OtxCcDef.h.

- OTX_CC_EC_CONN_IN: indicate an incoming call
- OTX_CC_EC_CONN_RS: answering an incoming call
- OTX_CC_EC_CONN_CO: call connection confirmed
- OTX_CC_EC_ALERT_IN: alert indication
- OTX_CC_EC_PROGRESS_IN: outgoing call progression
- OTX_CC_EC_CLEAR_IN: connection release indication
- OTX_CC_EC_CLEAR_CO: connection release confirmed

5. Directory Structure of API

At root directory, ReadMe gives some references of OTX ISDN product and what is added for each new release; OtxIsdnRev.h contains the release version.

For any library (*.lib) and executable (*.exe) files, only release version is provided.

5.1 Doc

Help file with *.hlp and *.html format

5.2 OtxApi

OTX hardware driver API header files and its library OtxDrv.lib

5.3 OtxCc

OTX ISDN Call Control API header files and library OtxCcPri.lib

5.4 OtxIsdnDemo

Demo programs for OTX ISDN are under specific network (for example: PRI)

5.5 OtxPh

OTX ISDN Physical Layer library OtxPh.lib