**ACD驱动开发**

# 文档适用范围

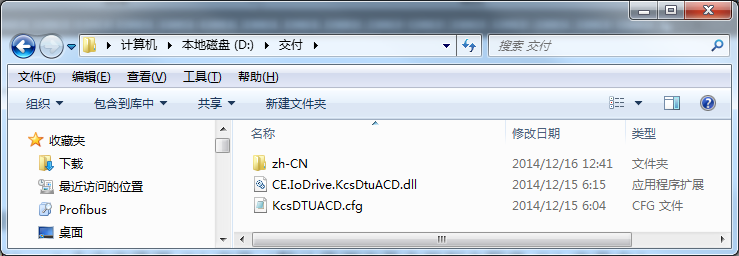
本规范适用于九思易驱动开发人员，开发NX ACD设备驱动的文档规范。

# 驱动概述

## 1.1 文件构成

驱动程序基本构成：一个主程序dll文件、2个cfg文件（中/英文驱动描述文件,后缀名为.cfg），1个中文资源dll文件（默认资源文件内嵌到主程序dll文件中）。

驱动开发提交的结果参照如下目录结构：\*.dll文件为主程序dll文件，\*.cfg文件为默认语言（英文）的驱动描述文件，zh-CN文件夹下为中文的驱动描述文件和中文的资源dll文件。



1. 图1-1交付根目录

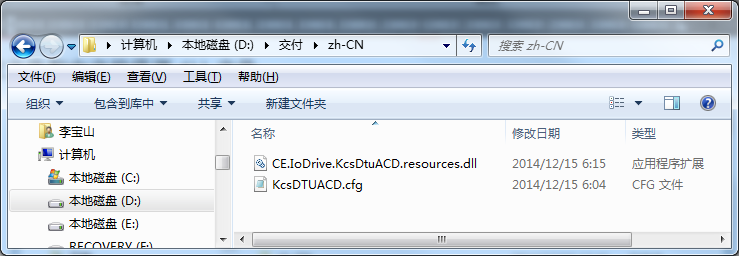


图1-2交付/zh-CN根目录

## 1.2 驱动名称

**代码命名规则：**

**命名空间**：ControlEase.IoDriver.XXX --- XXX为设备厂家名称

**开发类名称**：XXXDev（XXX代表设备型号或加上其它区分的信息，如 KcsW3100GDtuDev，表示凯诚晟型号为W3100的DTU设备驱动的开发类）

**运行类名称**：XXX（XXX代表设备型号或加上其它区分的信息，如 KcsW3100GDtu，表示凯诚晟型号为W3100的DTU设备驱动的运行类）

**资源命名规则**：Resources.resx（英文）， Resources.zh-CN.resx（中文）

**程序集名**：CE.IoDriver.Manufacture.XXX.dll（XXX含义：提供本驱动程序区别于其它驱动程序的名称，不局限于型号,接口类型或协议名称的组合等规则，只要能区别于其它驱动程序便可）

**Cfg命名规则：**

提供2个驱动描述XML文件，分别支持中文和英文。

CE.IoDriver.Manufacture.XXX.cfg(中文的配置文件,与主程序dll文件名称一致，只是后缀名不同而已)。示例如下：

<?xml version="1.0" encoding="utf-8" ?>

<ACDDriver Name="KcsDTU" Manufacture="凯诚晟" Model="W3100G" Category="DTU" Description="凯诚晟W3100 DTU设备驱动程序" PCSide="Ethernet" DeviceSide="Serial" Assembly="CE.IoDrive.KcsDtuACD.dll"

DevClass="ControlEase.IoDrive.KcsDtuDeviceDev" RunClass="ControlEase.IoDrive.KcsDtuDevice" >

</ACDDriver>

CE.IoDriver.Manufacture.XXX.cfg(英文的配置文件，与主程序dll文件名称一致，只是后缀名不同而已)。示例如下：

<?xml version="1.0" encoding="utf-8" ?>

<ACDDriver Name="KcsDTU" Manufacture="Kcs" Model="W3100G" Category="DTU" Description="KcsDtuACDDriver" PCSide="Ethernet" DeviceSide="Serial" Assembly="CE.IoDrive.KcsDtuACD.dll"

DevClass="ControlEase.IoDrive.KcsDtuDeviceDev" RunClass="ControlEase.IoDrive.KcsDtuDevice" >

</ACDDriver>

## 1.3 安装目录

主程序dll和默认的cfg文件位于FX软件安装目录下的*\Drivers\ACD\*文件夹下，中文cfg文件位于*\Drivers\ACD\zh-CN*文件夹下，中文的资源dll文件位于NX软件安装目录下的*zh-CN*文件夹下。

# 功能实现

具体ACD驱动实现主要根据ACD的通信协议和IO架构规范，完成用于IO系统中的运行的功能和开发的功能。

## 2.1 运行功能实现

主要是按照某种通信协议完成准备ACD、对Device Driver和Device之间往来数据的叠加和提取、检测ACD的状态以及通信完毕后停止与ACD设备的通信等功能。

实现这些功能时，需要从ACD驱动运行基类派生，完成基类中规定的各种功能的实现。依据ACD与PC之间的通信方式的不同要选择不同的ACD驱动基类，选择规则如下表：

|  |  |
| --- | --- |
| **ACD的通信方式** | **基类** |
| 串口 | SerialACD |
| 以太网 | EthernetACD |
| USB | USBACD |

### 描述ACD的基本特性

* **ACD是否主动建立通信连接(IsActiveConnect)**

在与设备通信前，必须先建立连接，连接成功后方可进行通信。建立通信连接可能是IO主动（如串口通信时，IO主动打开串口及建立了通信连接），也可能是ACD主动（如以太网通信时，ACD作为Client，主动与IO建立Socket通信连接）。IO系统中需要根据该信息决定是否启用通信连接监视功能，以使后续的数据采集正常进行。

* **ACD的传输快慢（TransmitType）**

ACD在硬件连接上位于PC和设备之间，主要起到PC和设备之间数据传输的作用，不同的ACD其数据传输性能差异较大，具体驱动开发要考虑ACD的这种特性，以便IO架构针对这种差异做不同的处理，以提高通信的效率。该特性有两种取值：

Normal：默认值，代表正常的传输速度

Slow：代表很慢的传输速度

* **ACD在通信网络中的唯一标识(**DeviceID)

ACD处在一个多设备组成的通信网络中，须指定该ACD在网络中区别于其它ACD的唯一标识，如485网络中的站号，以太网络中的IP地址等。

* **ACD的可配置参数（ConfigData）**

一种ACD驱动可驱动多个同类的ACD设备与IO系统交互，不同的ACD其配置不同，故具体ACD驱动要指定有哪些参数需要配置。

### 2.1.2 实现功能

##### 准备ACD

准备工作依ACD设备的不同而不同， 如电话Modem在工作之前需要完成拨号等准备工作。实现的方法是：

protected virtual ExecutionResult Start ( )；

##### 往来数据的叠加和提取

ACD驱动要对Device Driver发给Device的数据按照ACD的通信协议进行叠加等处理，以处理成ACD设备认识的数据，发送给ACD设备，实现方法：

protected virtual ExecutionResult SendDeviceData ( byte[] buffer )

ACD驱动要对Device发给Device Driver的数据按照ACD的通信协议进行提取等处理，以处理掉ACD设备的信息后返回Device的数据，实现方法：

protected virtual byte[] ReceiveAndAnalyzeData ( )

##### ACD状态

ACD状态的获得，可以是ACD Driver周期性地去读取反映ACD状态的通道状态，也可以是周期性的向ACD发出指令询问其工作状态，亦或不断接受ACD主动上发的表面自己工作状态的（心跳）信息，来判断其工作状态。

ACD驱动实现中，要在ACD发生故障时，使用基类提供的功能通知IO处理故障，调用的方法是：

Protected void OnFailured ( EventArgs e )；

##### 停止与ACD设备的通信

实现该方法，完成停止与ACD设备通信的逻辑：

protected virtual ExecutionResult Stop ( )；

##### 其他功能

###### 分析ACD标识数据

在描述ACD基本特性时，若设备特性满足如下条件：1.ACD主动发起通信连接；2.ACD主动上发其标识数据；需要重载实现如下方法：

string AnalyzeIdentityData ( byte[] idData, out byte[] backData )

其中参数“idData”是需要解析的设备标识数据；“backData”是ACD上发标识数据后给设备回馈的数据，若ACD上发标识数据后不需要给其回馈数据则置为空值；

备注：在解析ACD标识数据时，如果解析出来的标识数据同驱动配置数据相同则返回标识，否则返回字符串空值。

###### 构建获取标识命令

在描述ACD基本特性时，若ACD特性满足如下条件：1.ACD主动发起通信连接；2.ACD不主动上发其标识数据；3.需要给ACD发送命令来查询其标识；这时需要重载实现如下两个方法：

string AnalyzeIdentityData ( byte[] idData, out byte[] backData )；该方法实现参照“分析设备标识数据”所描述内容；

byte[] BuildGetIdentityCommand ( )，用来获取ACD标识的命令，根据ACD通信协议要求构建该命令的字节数组并返回。

## 2.2 开发功能实现

具体通道驱动实现开发类时从ACDDev类派生，主要提供如下功能：

需要用户配置的内容ConfigData，以及编辑配置内容的WPF控件。

# 常用术语

|  |  |  |
| --- | --- | --- |
| **序号** | **名词** | **名词定义** |
| 1 | 辅助通信设备  (ACD:Auxiliary Communication Device) | 在数据通信现场中，为扩大通信距离/完成协议类型转换/减少布线等目的，常需要在PC和设备间接入一些中间设备,如DTU、串口服务器、电话Modem等，我们把这些起到辅助通信作用的设备统称为“ACD”。 |
| 2 | ACD驱动  （ACD Driver） | IO系统中的一种软件程序，对ACD的抽象，负责描述ACD与通信有关的信息，以及按照ACD的通信协议完成对ACD的操控，实现与Device的通信，并获得ACD的状态。不同的ACD驱动与ACD的通信协议相关。 |
| 3 |  |  |