

基于 OPC 技术的 WinCC 实时数据归档

强明辉, 张琳琳, 马永伟

(兰州理工大学 电气工程与信息工程学院, 甘肃 兰州 730050)

摘要: 随着信息化步伐的加快, 企业对生产数据的实时性和完整性要求越来越高, 利用传统方式实现 WinCC 数据的转存已不能满足生产需要。利用 VB 编写 OPC 接口, 实现 WinCC 与数据库之间的动态数据交换, 解决了 WinCC 组态软件中实时数据转存到标准数据库的难题, 在实际应用中取得了良好的效果。

关键词: WinCC; 数据库; VB; OPC; 动态数据交换

DOI: 10.3969/j.issn.1000-3886.2015.03.007

[中图分类号] TP273 [文献标志码] A [文章编号] 1000-3886(2015)03-0021-03

OPC-based WinCC Real-time Data Archiving

QIANG Ming-hui, ZHANG Lin-lin, MA Yong-wei

(College of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou Gansu 730050, China)

Abstract: With accelerating informatization in recent years, enterprises pose more and more high demand on the instantaneity and integrity of their production data, and WinCC data dumping in the traditional way cannot satisfy the requirements in production. OPC interface is compiled through VB to realize data exchange between WinCC and the database, thus solving the problem of real-time data dumping to the standard database in WinCC configuration software. Excellent results are achieved in practical applications.

Keywords: WinCC; database; VB; OPC; dynamic data exchange

0 引言

SIMATIC WinCC 是由西门子公司和微软公司共同开发的一款功能强大的 HMI/SCADA 应用软件系统^[1]。WinCC 本身具有自己的归档数据库, 但其进行归档后的数据库为压缩格式, 需要通过 WinCC 专门集成的工具分析并显示数据, 这为第三方的访问带来了许多不便。由于 WinCC 集成了 OPC 技术, 所以可以对其进行二次开发。本文是利用 VB 编写 OPC 客户端, 以 WinCC 作为 OPC 的服务器, 利用 OPC 协议实现 VB 和 OPC 之间的动态数据交换, 然后通过 VB 的 ADO 数据接口, 把数据存储在数据库中。解决了 WinCC 实时数据转存到标准数据库的难题, 为企业管理和监控系统提供了标准的数据源。

1 OPC 技术

OPC 是针对不同供应厂商的设备和应用程序之间的软件接口标准化, 使其间的数据交换更加简单化的目的而提出的。OPC 技术将各个设备驱动程序和通讯程序封装成独立的 OPC 服务器, OPC 客户端无需知道系统的性能特点, 而只需通过标准的 OPC 接口访问 OPC 服务器^[2-5]。

OPC 的核心是 COM/DCOM 技术, COM 技术是一种软件组件间相互数据交换的有效方法, COM 对象分为客户端和服务端两类, 客户端通过 COM 接口访问服务器。

OPC 数据访问提供从数据源读取和写入特定数据的功能。OPC 客户程序通过接口从服务器读取数据或写入数据。在 OPC 客户机访问服务器时, 需要创建一个服务器对象, 调用这个服务器对象的接口, 服务器对象创建组对象并返回组对象的指针, 客

户程序获得组对象的指针后调用其接口。OPC 客户程序访问服务器的结构图如图 1。

2 WinCC 与数据库的数据交换

2.1 WinCC 与 VB 的数据交换

2.1.1 WinCC 的 OPC 服务器设置

WinCC 的 OPC 服务器设置需要分几步完成:

1) 在安装 WinCC 的时候需要安装 OPC Server, 具体方法是在安装选项的 Communication 组件中, 选择 OPC Server 项。

2) 对 WinCC 的 OPC 服务器进行 DCOM 设置。具体方法是单击“开始”菜单并选择“运行”, 输入 dcomcnfg, 打开“组件服务”对话框; 在此对话框中, 单击“组件服务”——“我的电脑”——“DCOM 配置”——选择“OPCServer. WinCC”——右键“属性”按钮, 打开“OPCServer. WinCC 属性”对话框; 在此对话框中, 单击“安全”选项中自定义启动和激活权限, 在“标识”标签中选择“交互式用户”选项。

3) 在完成上述步骤以后, 打开组态好的 WinCC 监控界面。参数设置界面如图 2 所示。

2.1.2 OPC 客户端的设计及编程

利用 Visual Basic 开发 OPC 客户程序时, 要对 DLL 进行注

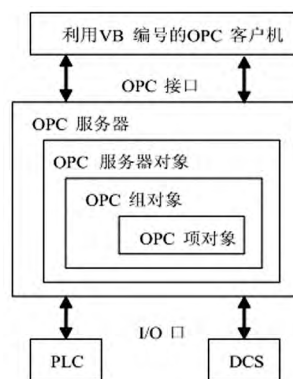


图 1 OPC 客户程序访问服务器结构图



图 2 WinCC 人机界面显示

册。具体的方法是从 VB 菜单中选择“工程”——“引用”，在可用的引用一览表中，选择 OPC Automation 2.0 后按确定。

1) VB 窗体设计

注册完 DLL 以后，就要设计 VB 的窗体。在窗体上添加两个 Command 按钮，分别命名为连接服务器和退出。根据项目要求，需同步传输 23 个数据，在窗体上布局 23 个 Text 控件会比较麻烦，本文采用编程的方式，在程序运行时，将 23 个 Text 控件同时显示在窗体上。部分代码如下：

```

Load Text1(1) //加载 Text 控件数组
Load Label1(1) //加载 Label 控件数组
Label1(1).Left = ((I - 1) \ 10) * (Label1(1).Width + 2000) + Label1(1).Left //Label 控件的横坐标
Label1(1).Top = ((I - 1) Mod 10) * (Label1(1).Height + 100) + Label1(1).Top //Label 控件的纵坐标
Text1(1).Left = ((I - 1) \ 10) * (Text1(1).Width + 2200) + Text1(1).Left //Text 控件的横坐标
Text1(1).Top = ((I - 1) Mod 10) * (Text1(1).Height + 100) + Text1(1).Top //Text 控件的纵坐标
Label1(1).Visible = True
Text1(1).Visible = True

```

VB 窗体如图 3。

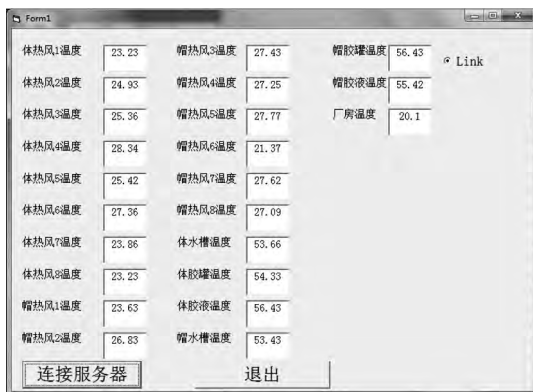


图 3 VB 窗体设计

2) OPC 客户端编程

VB 的 OPC 客户端和 OPC 服务器之间数据互相通讯的主要源代码如下：

变量声明

```

Dim WithEvents objServer As OPCServer //OPC 服务
Dim WithEvents objGroup As OPCGroup //OPC 组
Dim objGroups As OPCGroups
Dim objTestGrp As OPCGroup
Dim objItems As OPCItems
Dim IClientHandles(23) As Long //客户句柄
Dim ServerHandles() As Long //服务器句柄
Dim IErrors() As Long
Dim ItemIDs(23) As String //OPC 的标签
Dim vtItemValues() As Variant//用于存储 OPC 数据的地址
Dim a(1 To 23) As Variant

```

连接服务器

```

Set objServer = New OPCServer//创建新的 OPC 服务器
objServer.Connect (" OPCServer. WinCC. 1")
//连接 OPC 服务器
If objServer.ServerState = OPCRunning Then
Option1.Value = True //连接服务器成功的标志
添加 OPC 组和 OPC 标签
Set objGroups = objServer.OPCGroups //添加一个 OPC 组集合
Set objTestGrp = objGroups.Add("TestGrp")
//添加一个 OPC 组
Set objItems = objTestGrp.OPCItems //建立 OPC 标签集合
For I = 1 To 23//生成 23 个项标识符
ItemIDs(I) = "bb" & I
IClientHandles(I) = I
objItems.AddItem 23, ItemIDs, IClientHandles, ServerHandles, IErrors//添加 OPC 标签
断开服务器
objItems.Remove 23, ServerHandles, IErrors
//清除 OPC 的标签
objGroups.Remove "TestGrp" //清除 OPC 组
objServer.Disconnect //断开 OPC 的服务器
Set objItems = Nothing
Set objTestGrp = Nothing
Set objServer = Nothing

```

本程序是利用 Timer1 的 Timer 事件来进行数据更新的显示，如果数据有新的更新就要导入到数据库中，有关定时器的代码将在下面介绍。

2.2 VB 与数据库的数据交换

当数据由 WinCC 传输到 VB 以后，由于 VB 不是一个存储单元，所以要将数据存储到指定的归档数据库中，以备报表查询。因为该项目的数据量不大，选用 Access 为归档数据库。

2.2.1 VB 访问数据库的 ADO 接口

在 VB 中，常用的数据访问接口有下列几种：Active 数据对象 ADO、开放式数据库连接 ODBC、远程数据库对象 RDO、数据库访问对象 DAO。其中 ADO 接口是 Microsoft 开发数据库应用程序面向对象的新接口。采用了 OLE DB 的数据访问模式，是数据库访问对象 DAO、远程数据对象 RDO 和开放数据库互联 ODBC 三种方式的扩展，具有更加简单、灵活的操作性能^[6]。

ADO 对象模型定义了一个可编程的分层对象集合，主要有三个对象成员 Connection、Command 和 Recordset 对象，以及几个集合对象 Errors、Parameters 和 Fileds 等组成。

2.2.2 VB 利用 ADO 接口访问数据库

在 VB 使用 ADO 控件访问 Access 前,先要在 Access 中进行归档数据库环境设置。在 Access 中建立一个名为温度的数据库,并在该数据库的子目录中新建一张表,命名为现场温度记录,这张表用于存储从 WinCC 服务器中获得的现场温度数据。接着,在 VB 的开发环境下开发与数据库连接的程序,本次使用的 VB 控件主要是 Timer 和 Adodc 相应的程序设计步骤如下。

1) 建立 VB 与归档数据库链接。在 VB 中使用 Adodc 控件前,必须先对其进行设置,具体方法是:

首先,通过“工程”——“部件”菜单命令选择“Microsoft ADO Data Control 6.0 (SP6) (OLEDB)”选项,将 ADO 数据控件添加到工具箱,点击 ADO 数据控件,将其添加到窗体上。

然后,设置 Adodc2 的 ConnectionString 属性,在 Adodc2 控件上右键“Adodc 属性”——使用连接字符串——单击“生成”按钮——选择 Microsoft Jet 4.0 OLE DB Provider——单击“下一步”按钮——选择或输入数据库名称——单击“测试连接”按钮——弹出测试连接成功对话框,点击“确定”按钮即可。

最后,设置 Adodc2 的 RecordSource 属性,选择命令类型为 2 - adCmdTable,在表名项选择现场温度记录。

根据工艺要求,本项目需要同步传输 23 个数据,所以需要 23 个 Text 控件,这些控件是通过编程来显现在 VB 的窗体上的,对于这些控件的 ADO 属性设置,我们还需编程实现,部分代码如下:

```
Set Text1(I).DataSource = Adodc2 将所有 Text 控件的 DataSource 属性设置为 Adodc2
```

```
设置所有 Text 控件的 DataField 属性
```

```
For I = 1 To 8
```

```
Text1(I).DataField = "体热风" & I & "温度"
```

```
Next
```

```
For I = 9 To 16
```

```
Text1(I).DataField = "帽热风" & I - 8 & "温度"
```

```
Next
```

```
Text1(17).DataField = "体水槽温度"
```

```
Text1(18).DataField = "体胶罐温度"
```

```
Text1(19).DataField = "体胶液温度"
```

```
Text1(20).DataField = "帽水槽温度"
```

```
Text1(21).DataField = "帽胶罐温度"
```

```
Text1(22).DataField = "帽胶液温度"
```

```
Text1(23).DataField = "厂房温度"
```

2) 实现实时数据的存储。在窗体上插入一个定时器控件,命名为 Timer1,将其 Interval 属性设置为 100,在 Timer1_Timer() 事件中编程。为了不产生冗余,建了一个数据缓冲数组,把每次提取出来的数据同之前的数据缓冲组中的值作对比,若不同就插入数据库,同时更新数据缓冲数组中的值。部分代码如下:

```
PrivateSub Timer1_Timer()
```

```
Dim Update As Boolean
```

```
Dim I As Integer
```

```
SyncRead OPCache, vtItemValues, lErrors
```

```
//同步读取 WinCC 的数据
```

```
For I = 1 To 23
```

```
If vtItemValues(I) <> a(I) Then //将 OPC 标签数组中的数和缓冲
```

数组中的数进行对比

```
Update = True
```

```
Exit For
```

```
End If
```

```
Next
```

```
If Update Then
```

```
Adodc2.Recordset.AddNew //向 Access 数据库中插入数据
```

```
Adodc2.Recordset.Fields("日期及时间") = Now //向 Access 数
```

据库中插入当前的日期和时间

```
For I = 1 To 23
```

```
Text1(I).Text = vtItemValues(I)
```

```
a(I) = vtItemValues(I)
```

```
Next
```

```
Adodc2.Recordset.Update //保存插入到数据库中的数据
```

```
End If
```

```
End Sub
```

部分实时数据进入表后如图 4。

	帽水槽温度	帽胶罐温度	帽胶液温度	厂房温度	日期及时间
	53.43	56.43	55.42	20.1	2014/7/11 11:33:14
	53.43	56.43	55.42	20.1	2014/7/11 11:34:14
	53.43	56.43	55.42	20.1	2014/7/11 11:34:27
	52.62	56.43	55.42	20.1	2014/7/11 11:35:14
	52.62	57.23	55.42	20.1	2014/7/11 11:35:28
	52.62	57.23	53.62	20.1	2014/7/11 11:35:34
	52.62	57.23	53.62	20	2014/7/11 11:35:43
	52.75	57.23	53.62	20	2014/7/11 11:36:04
	52.75	56.42	53.62	20	2014/7/11 11:36:14
	52.75	56.42	55.43	20	2014/7/11 11:36:25
	53.42	56.42	55.43	20	2014/7/11 11:37:09
*	0	0	0	0	

图 4 数据存入 Access 数据库

3 实时数据的查询

由图 4 可知,只要 WinCC 从下位机采集的这一组数据中的任何一个数据有变化,那么该组数据就会传入到数据库中。根据工艺要求可知,每个采集点的温度均有一定的范围,超过这个范围就会对设备或产品产生一定的损害,所以需建立一个查询,当温度高于该范围时,将其筛选出来,插入到该数据库的另外一个表中,供以后的分析。构造的 SQL 语句通式如下:

```
Insert into + "表名" + select + "选择内容" + from + "表名" + where + "条件"
```

建立查询的相关步骤如下:

1) 在数据库温度中再建立一张新表,命名为查询温度,用来存储查询出来的超出范围的温度。

2) 在 VB 窗体上再布局一个 ADO 控件,命名为 Adodc1,将其的 ConnectionString 设置的与 Adodc2 一样,RecordSource 属性设置为查询温度。然后编写代码: Adodc1.RecordSource = "SQL 语句" 即可以进行相关的查询。

4 结束语

利用 VB 设计的 OPC 客户端数据接口,传输数据稳定,实时性好,能同时传输大批量数据。ADO 数据库接口简单,传输稳定,实时性好,技术成熟。通过 OPC 技术和 ADO 接口有效地完成了 WinCC 到数据库转储功能。为企业进一步分析数据,利用数据,提供了可靠的数据源。

(下转第 39 页)

电流 I_b 的参考值(电流内环)再通过 PI 调节器产生导通占空比 D 触发双向 DC/DC 变换器的开关器件对电池进行充放电。

3 含蓄电池储能的直驱永磁风电系统仿真

为了验证所提控制策略的正确性,在 MATLAB/Simulink 中搭建系统模型并进行仿真。

仿真参数如下:(1)模拟风力机:空气密度 1.225 kg/m^3 ;风轮半径 40 m 桨距角 0° ;(2)永磁同步发电机:定子电感 0.47 mH ,定子电阻 0.019Ω 极对数 1;(3)网测:电网线电压 575 V ,滤波电感 2.6 mH ;(4)直流侧:电容 $9000 \mu\text{F}$,直流侧设定电压 1100 V 双向 DC/DC 换流器电感 0.5 mH 滤波电容 $760 \mu\text{F}$ 。(5)蓄电池参数:额定电压 800 V 额定容量 2 MAh 初始容量 50% 。

本文将功率值标么化(pu),标么化的功率基值取发电机额定功率。

当风速按阶跃变化时(0.1 s 由 10 m/s 降到 5 m/s ,在 0.4 s 由 5 m/s 上升到 15 m/s),电网的需求功率恒定不变时,对系统进行仿真。系统仿真波形图如图 5 所示,包括风速变化波形、发电机输出功率 P_s 、电网需求功率指令 P_{agc} 、电网侧实际输出功率 P_g 及电池充放电功率 P_b 波形。

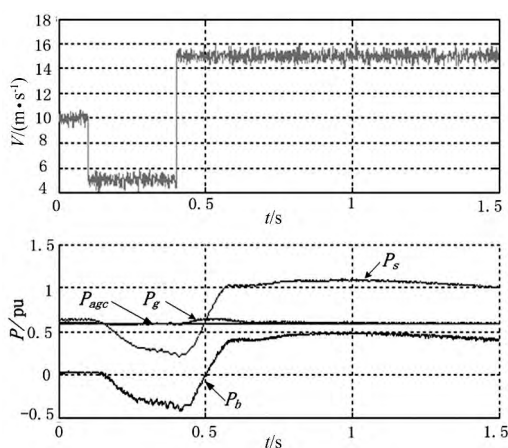
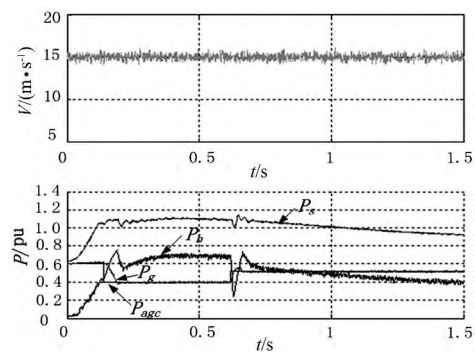


图 5 风速变化时系统仿真波形图

从图 5 可以看出,当风速在不断变化时,风电机输出功率会相应地波动,当风电机输出功率小于电网需求指令功率时,蓄电池能够快速放电;当风电机输出功率大于电网需求指令功率时,蓄电池能够快速充电,使电网侧实际输出功率很好地跟踪功率指令,达到较好的平抑效果。

当风速恒定为 15 m/s ,电网需求功率为阶跃波形时(在 $t = 0.16 \text{ s}$ 从 0.6 变为 0.4 , 0.6 s 时从 0.4 变为 0.5),对系统进行了仿真。系统仿真波形图如图 6 所示,包括风速波形、发电机输出功率 P_s 、输出功率指令 P_{agc} 、电网实际输出功率 P_g 及蓄电池充放电功率 P_b 波形。

从图 6 可以看出,当风速保持恒定,网测需求功率变化时,由于输出功率指令始终小于发电机发出功率,电池一直在充电。当指令功率下降时,电池吸收的功率增加;当指令功率上升时,指令吸收的功率减少。由于蓄电池良好的响应特性,能够很好的跟踪功率指令,可以对电网实现削峰填谷。



4 结束语

本文在分析传统直驱永磁风力发电系统的基础上,在直流侧加入蓄电池储能系统,通过对储能系统的控制来调节发电机输送到电网的功率。当风速变化时,蓄电池储能环节能够平滑风电机输出功率波动;当风速不变时,蓄电池储能环节能够平衡电网需求功率。仿真证明,加入储能系统之后可以有效地改善永磁直驱风力发电系统并网特性。

参考文献:

- [1] 熊倩,廖勇,姚骏.含飞轮储能单元的直驱永磁风力发电系统有功功率平滑控制[J].电力自动化设备,2013,33(5):97-105.
- [2] 李蓓,郭剑波.平抑风电功率的电池储能系统控制策略[J].电网技术,2012,36(8):38-43.
- [3] 熊倩,廖勇.含飞轮储能的直驱风电系统功率平滑控制策略研究[J].华东电力,2012,40(9):1533-1539.
- [4] 陈星莺,刘孟觉,单渊达.超导储能单元在并网型风力发电系统的应用[J].中国电机工程学报,2001,21(12):63-66.
- [5] 王文亮,葛宝明,毕大强.储能型直驱永磁同步风力发电控制系统[J].电力系统保护与控制,2010,38(14):43-48.
- [6] 姚骏,廖勇,瞿兴鸿,等.直驱永磁同步风力发电机的最佳风能跟踪控制[J].电网技术,2008,32(10):11-16.
- [7] 孙春顺,王耀南,李欣然.飞轮辅助的风力发电系统功率和频率综合控制[J].中国电机工程学报,2008,28(29):111-116.
- [8] 张步涵,曾杰,毛承雄,等.电池储能系统在改善并网风电场电能质量和稳定性中的应用[J].电网技术,2006,30(15):54-58.
- [9] 胡书举,李建林,许洪华.直驱风电系统变流器建模和跌落特性仿真[J].高电压技术,2008,34(5):949-954.
- [10] 葛善海,衣宝廉,张华民.多硫化钠溴储能电池高效电极的研究[J].电源技术,2003,27(5):446-450.

【作者简介】徐小三(1990-)男,安徽人,硕士生,研究方向为风力发电。
张明(1965-)男,上海人,副教授,研究方向为风力发电、控制系统。

(上接第 23 页)

参考文献:

- [1] 甄立东.西门子 WinCC V7 基础与应用[M].北京:机械工业出版社,2011.
- [2] 日本 OPC 协会. OPC 应用程序入门[M].日本:2002.
- [3] 韦源,于平. Visual Basic 程序设计基础[M].北京:清华大学出版社,2001.
- [4] 朱翠苗,郑广成. VB 访问 SQL Server 数据库的几种方案[J].河南科

- [5] 林启宽.用 VB 开发 WinCC 的 OPC 客户机[J].工业控制计算机,2006,19(8):80-81.
- [6] 赵秀梅.基于 WinCC 工控组态软件的关系数据库的研究[J].微型机与应用,2010,29(6):1-2.

【作者简介】强明辉(1960-)男,陕西扶风人,教授级高级工程师,主要从事过程控制、检测技术方面的研究。