

报警存储与历史报警查询

配置参考文档

北京亚控科技发展有限公司

技术部

目录

一、功能概述	4
一、功能概述	4
二、工程实例	4
三、操作步骤:	4
1、新建工程:	4
1.1) 定义设备:	4
1.2) 定义变量:	4
1.3) 定义报警:	6
1.4) 实时报警信息:	6
2、报警配置:	8
2.1) 建立报警数据库:	8
2.2) 设置 ODBC 数据源:	9
2.3) 报警配置:	11
3、进入运行系统:	14
4、历史报警查询:	15
4.1) 创建 KVADODBGrid 控件:	15
4.2) 创建日历控件:	18
4.3) 编辑其他画面:	20
5、进入运行系统:	21
四、注意事项	22

图表

图一 定义变量压力.....	5
图二 定义变量流量.....	5
图三 报警组定义.....	6
图四 压力变量报警定义.....	7
图五 流量变量报警定义.....	7
图六 ODBC 数据源管理器.....	9
图七 选择数据源的驱动程序.....	10
图八 数据源定义.....	10
图九 选择数据库.....	10
图十 ODBC 数据源定义	11
图十一 报警配置.....	11
图十二 数据库配置.....	12
图十三 数据库配置.....	13
图十四 数据库配置.....	13
图十五 实时报警.....	14
图十六 Alarm 表	14
图十七 插入通用控件.....	15
图十八 KV 控件属性.....	16

图十九 数据链接属性.....	17
图二十 KV 控件配置.....	18
图二十一 运行系统画面.....	18
图二十二 日历控件.....	19
图二十三 控件事件函数.....	20
图二十四 报警查询.....	21
图二十五 历史报警查询.....	22

一、功能概述

常规需求：很多工业现场要求将变量的报警信息进行存储，并且可以灵活的进行历史报警的查询、打印。

组态王中的实现方法：**组态王支持通过 ODBC 接口将数据存储到关系数据库中，并且提供 KVADODBGid 控件对存储的历史报警信息进行条件查询，并可以对查询结果进行打印。**关系数据库可以为 Access 数据库或者 SQLServer 数据库。

二、工程实例

我们通过一个简单的工程对报警信息的存储以及历史报警信息的查询进行讲解。历史报警的查询主要根据日期、报警组为条件进行查询。报警信息存储的数据库以 Access 数据库为例进行。下面就以此为例来演示完成这一现场需求的设置步骤。

三、操作步骤：

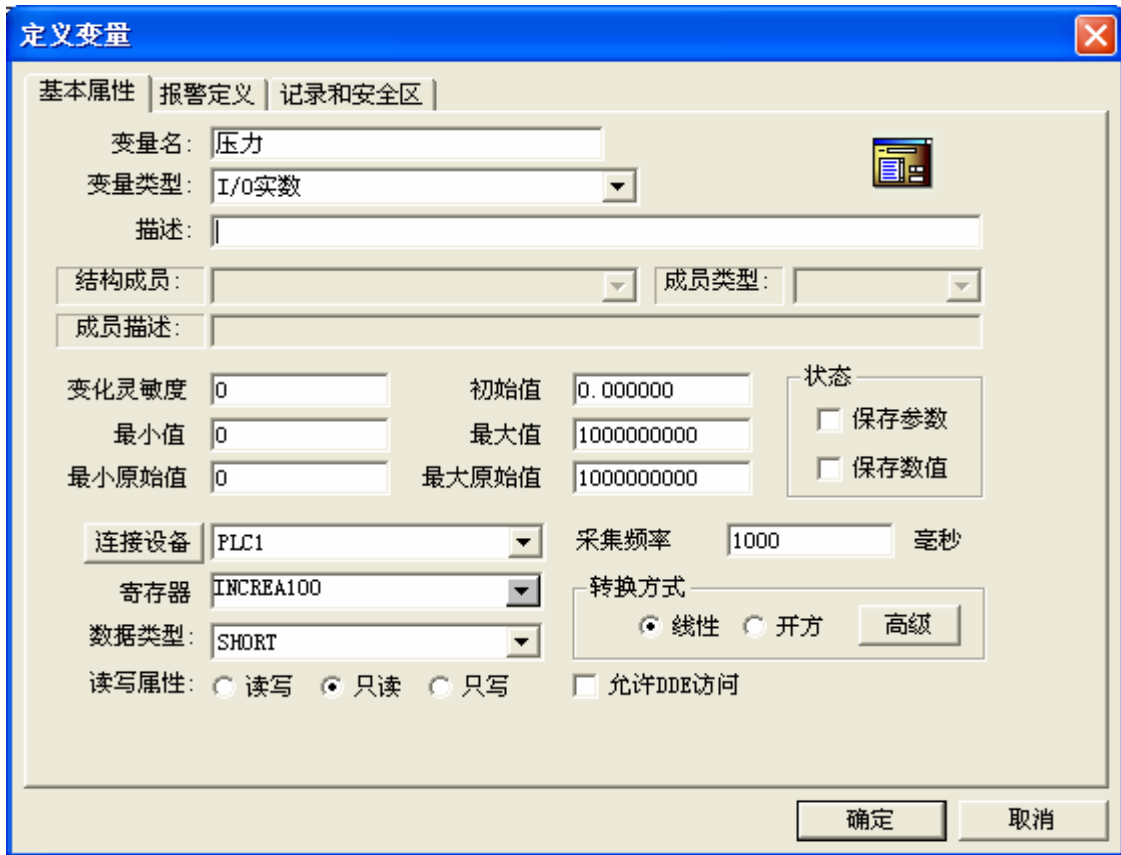
1、新建工程：

1.1) 定义设备：

新建一个工程，在设备处新建设备，我们在本例中定义一个仿真 PLC 的设备，设备列表为“PLC—亚控—仿真 PLC—串口”，设备名称为“PLC1”。此仿真 PLC 可以做为虚拟设备与组态王进行通讯。仿真 PLC 主要有如下的寄存器：自动加 1 寄存器 INCREA，自动减 1 寄存器 DECREA，随机寄存器 RADOM，常量寄存器 STATIC，常量字符串寄存器 STRING，CommErr 寄存器。具体的寄存器的使用请参考组态王 IO 驱动帮助。

1.2) 定义变量：

在数据词典中新建两个 IO 变量，变量名称分别为 **压力**、**流量**，变量类型为 IO 实数，连接的设备为前面定义的仿真 PLC 设备“PLC1”，寄存器分别为 INCREA100，DECREA100，数据类型为 short，读写属性为“只读”。这样压力变量就会从 0 开始自动加 1，其变化范围是 0 到 100，流量变量从 100 开始自动减 1，其变化范围是 0 到 100。变量定义画面如下图一、图二所示：



图一 定义变量压力



图二 定义变量流量

1.3) 定义报警：

我们首先定义报警组，在报警组处双击进行报警组对话框，点击“增加”定义两个报警组，分别为“压力报警”、“流量报警”，确认完成报警组的定义，如图三所示：



图三 报警组定义

报警组定义完成后。重新编辑变量的定义，在变量定义的“报警定义”选项中我们对这两个变量进行报警的定义。压力变量定义报警组名为“压力报警”，设置报警限为高、高高限，限值分别为 70，95。流量变量定义报警组名为“流量报警”，设置报警限为低、高报警，报警限值分别为 10，90。定义报警画面如下图四、图五所示：

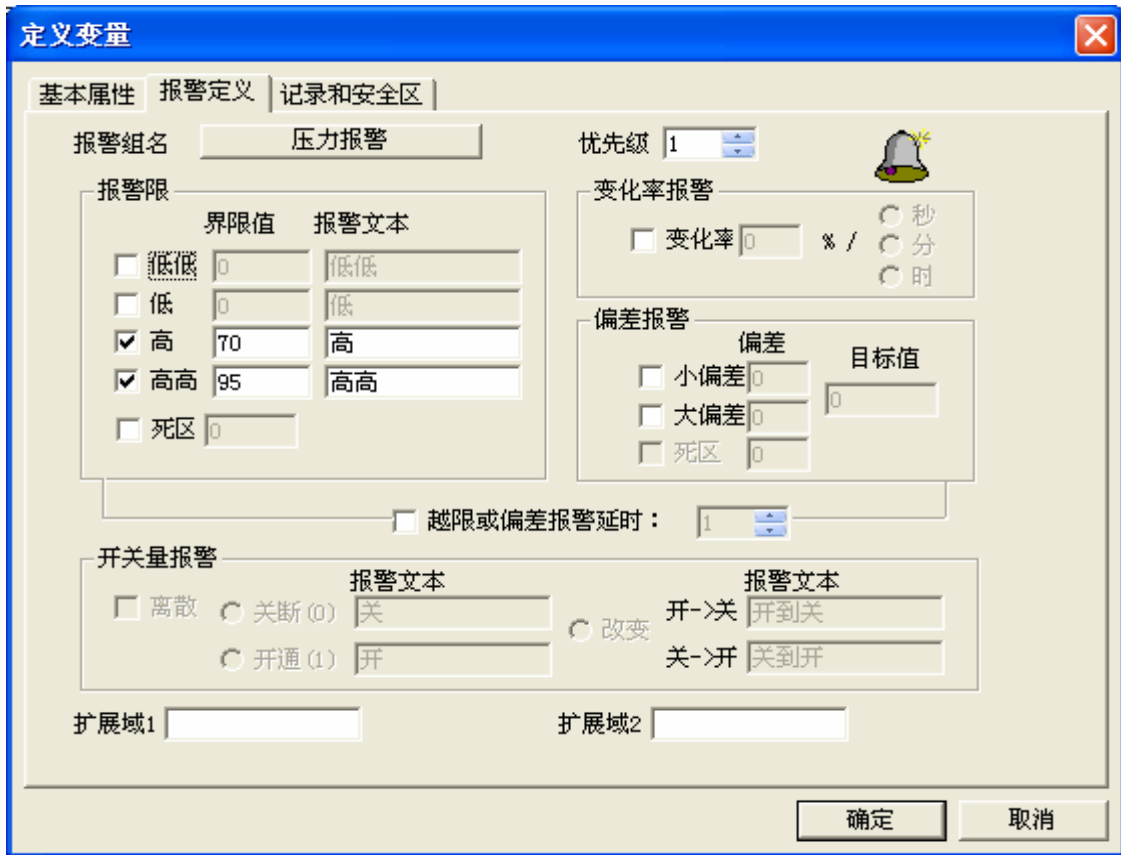
1.4) 实时报警信息：

变量的报警就定义完成后，我们新建一个“实时报警”画面，在工具箱中选择报警窗口，然后在画面上完成报警窗口的制作，双击画面上的报警窗口，为报警窗口命名为“报警”，根据需要可以对报警窗口进行灵活的配置，详细的配置可以参考组态王手册或者组态王帮助，但是必须注意报警窗口的名字一定要填写，如果报警窗口没有名字，则此报警窗口无效。

在画面上制作两个按钮，分别为“画面切换”、“退出系统”。命令语言如下：

画面切换按钮：ShowPicture("报警查询");

退出系统按钮：exit(0);



图四 压力变量报警定义



图五 流量变量报警定义

报警窗口定义完成后，如果此时进入运行系统，则当出现报警后，报警信息会在报警窗口中出现。需要注意的是，报警窗口显示的信息在计算机的内存中，如果组态王退出后再进入运行系统则原来的报警并不存在了，也就是说历史的报警信息并没有保存下来。下面我们会详细讲解一下如何将报警信息进行保存以方便以后的查询。

2、报警配置：

组态王报警配置主要分为三个配置选项：文件配置、数据库配置、打印配置。文件配置主要是将报警信息存储到文件中，文件格式为 *.al2，我们可以通过记事本打开此文件对存储的信息进行浏览，因为此存储格式浏览不是很方便，我们现在不推荐客户使用。数据库配置是将报警信息存储到关系数据库中，如 Access，SQLServer 等，此方式浏览、查询比较方便，本文就是以数据库配置作为讲解的重点。打印配置为报警信息的实时打印，需要注意的是打印配置选择的打印机必须为带字库的针式打印机。下面我们主要以 Access 数据库为例讲解报警存储到数据库的使用配置。

2.1) 建立报警数据库：

在 Access 中新建一个空数据库，例如建立路径为：D:\报警存储与查询\报警数据库.mdb。在此数据库中创建一个数据表：表的名称为：Alarm。表的字段名称如下表，字段类型为文本类型。

字段名称	说明
AlarmDate	报警日期
AlarmTime	报警时间
VarName	变量名
GroupName	报警组名
AlarmValue	报警值
LimitValue	限值
AlarmType	报警类型
Pri	优先级
Quality	质量位
AcrDate	事件日期
AcrTime	事件时间
OperatorName	操作员名
VarComment	变量描述
ResumeValue	恢复值
EventType	事件类型
MachineName	工作站名称
IOServerName	报警服务器名称

为了方便客户使用，在组态王的安装盘中已经有一个已经做好的一个数据库文件，我们可以直接使用。此文件在组态王的安装盘的路径为：Value Pack\报警窗数据库，文件名为：报

警窗数据库.mdb。我们可以直接拷贝此文件到计算机的硬盘中直接使用。需要注意的是光盘中的文件为“只读”属性，必须将“只读”属性去掉才可以。

2.2) 设置 ODBC 数据源：

组态王通过 ODBC 数据源将报警信息存储到数据库中，因此我们必须先建立 ODBC 数据源。没有Microsoft Access Driver, 打开目录：“C:\Windows\SysWOW64”，双击该目录下的“odbcad32.exe”文件，就进去ODBC数据源管理界面了

在“控制面板”-----“管理工具”----“ODBC 数据源”中建立 ODBC 数据源，点击“ODBC 数据源”弹出“ODBC 数据源管理器”，如下图六所示：

在“用户 DSN”中点击“添加”，弹出“选择数据源驱动程序”窗口，如下图七所示：选择“Microsoft Access Driver (*.mdb)”驱动，点击“完成”。弹出如图八所示窗口，填写 ODBC 数据源的名称，根据需要对数据源进行命名，如“报警”，点击“选择(S)”，如图九示，选择我们前面定义的数据库文件“D:\报警存储与查询\报警数据库.mdb”。点击“确定”完成 ODBC 数据源的定义，如图十所示。其他数据库如 SQLServer 的 ODBC 定义请参考相关文档。



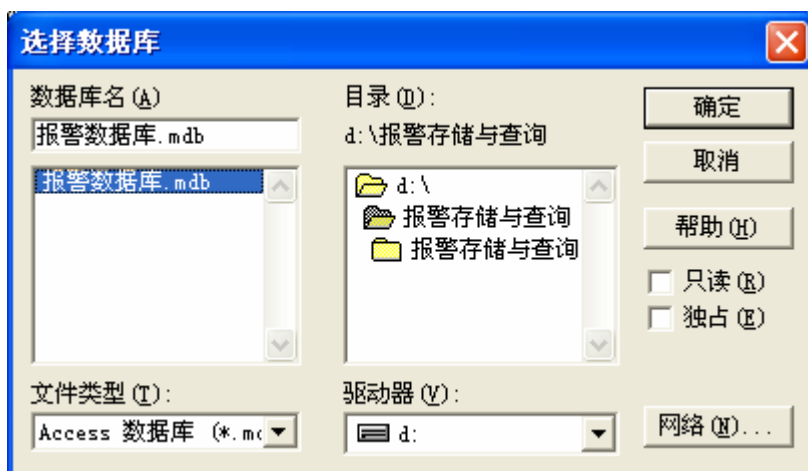
图六 ODBC 数据源管理器



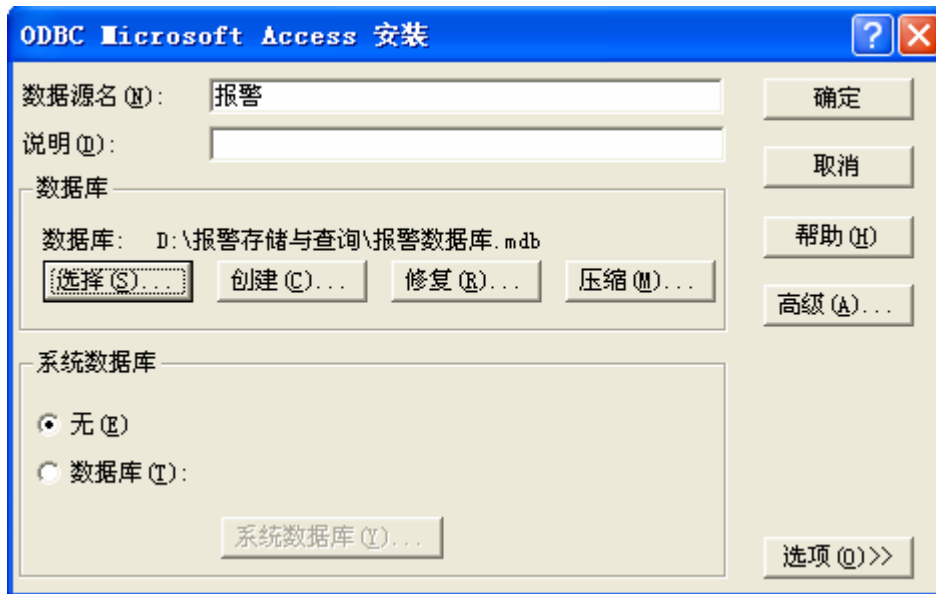
图七 选择数据源的驱动程序



图八 数据源定义



图九 选择数据库



图十 ODBC 数据源定义

2.3) 报警配置:

数据库以及 ODBC 数据源定义完成后，我们进行报警配置中的数据库配置。双击组态王工程浏览器的“系统配置”中的“报警配置”，弹出如图十一的“报警配置”对话框。选择“数据库配置”选项卡，如图十二所示：我们根据需要将“记录报警事件到数据库”打上勾，点击报警格式，根据实际情况对报警格式进行选择配置，需要注意的是默认的报警格式没有选择报警日期、事件日期，因此必须进行报警格式的配置。



图十一 报警配置

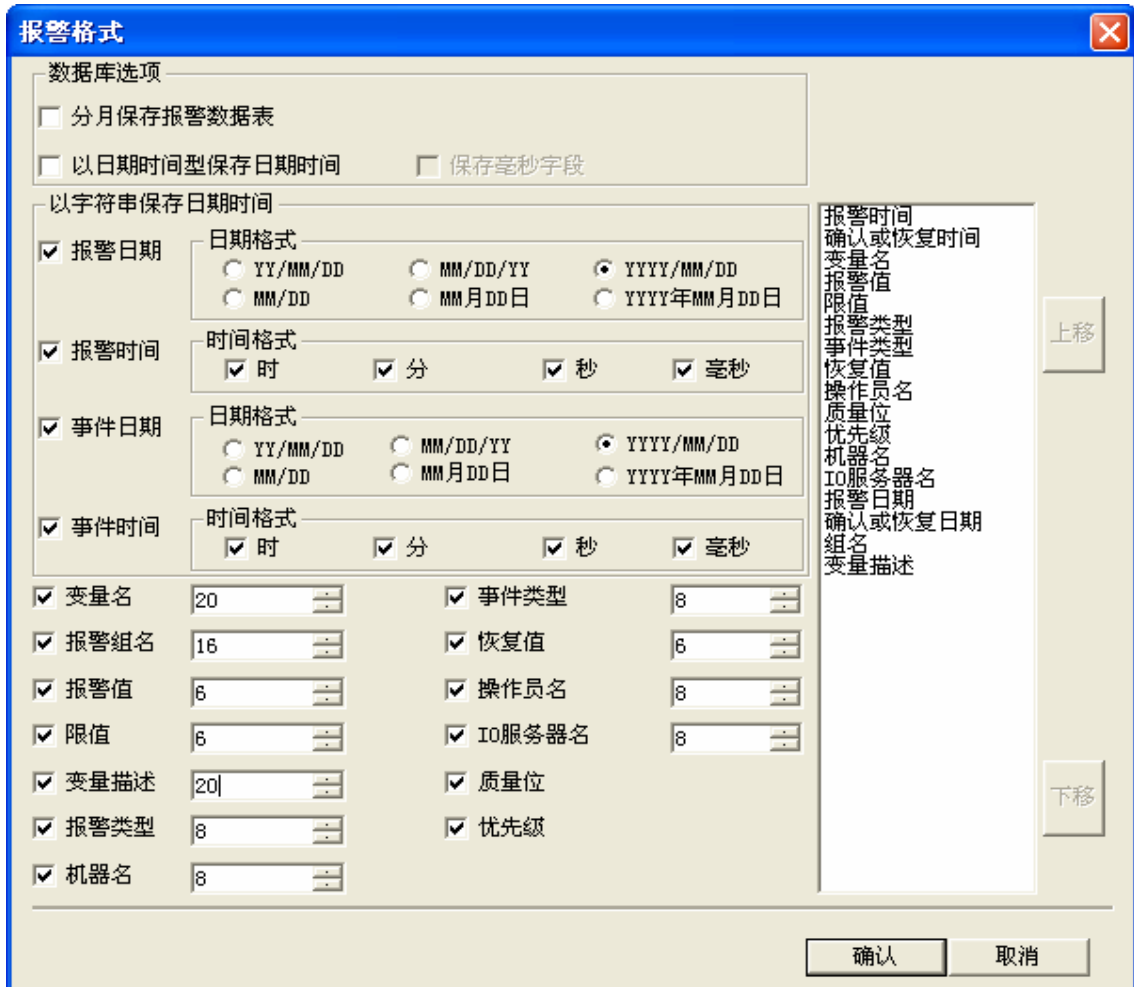


图十二 数据库配置

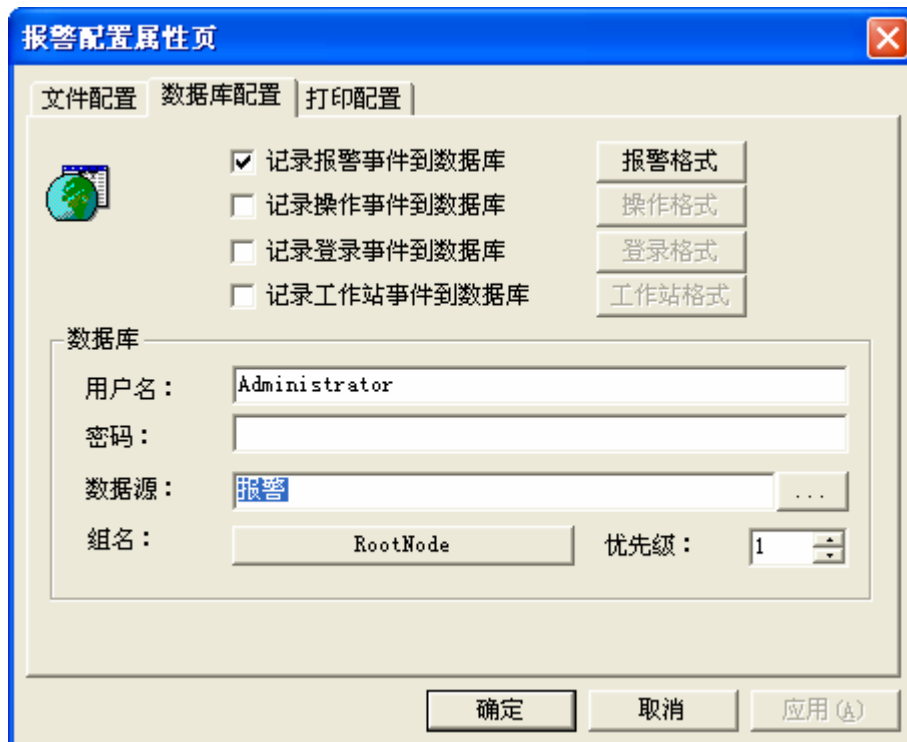
“报警格式”配置如图十三所示：需要注意的是：在 6.52 版本之前的报警格式配置中没有“数据库选项，分月保存报警数据表，以日期时间类型保存日期时间”选项。6.52 版本增加了此选项。其中“分月保存报警数据表”选项如果选中，则保存报警信息的数据库中的数据表每月生成一个，并且无需建表，只需要建一个空的数据库即可。采用分月保存的方式的优点在于：如果报警信息数据量比较大，分表存储可以提高查询的速度。缺点在于：无法进行跨月的查询，在编写脚本进行查询时需要考虑查询的是那一个数据表。

本文我们还是按照报警信息存储到一个数据表的方式为例进行介绍，也就是说我们不选中“分月保存报警数据表”。其他选项需要注意的就是数据长度要根据实际情况进行设置，并且选中“报警组名”，如果使用描述则“变量描述”也需要选中。

报警格式设置完成后，点击“确定”返回“数据库配置”画面，在数据源处选择我们前面定义的数据源“报警”。如图十四所示：点击“确定”完成报警的配置。



图十三 数据库配置



图十四 数据库配置

3、进入运行系统：

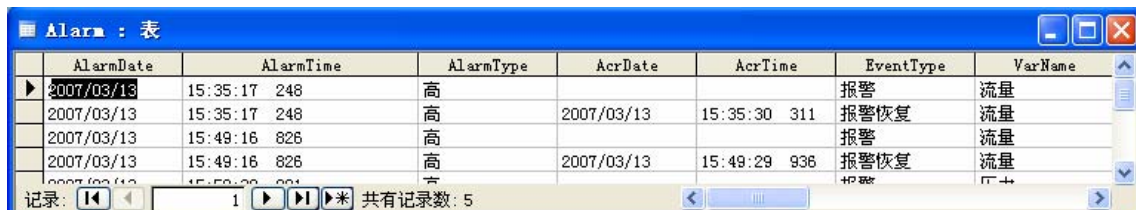
画面开发完成后保存画面，在工程浏览器的“系统设置”－“设置运行系统”－“主画面配置”中，将新建的画面设置为主画面。确认后点击工程浏览器的“View”按钮切换到运行系统。

系统运行后将“实时报警”画面打开，如下图十五所示：



图十五 实时报警

当有报警产生后，会在报警画面中显示当前的报警信息，同时也会将报警信息存储到 Access 数据库中。我们可以打开 D:\报警存储与查询\报警数据库.mdb 的数据库，打开“Alarm”表，如图十六所示：报警信息已经存储到数据库中。



图十六 Alarm 表

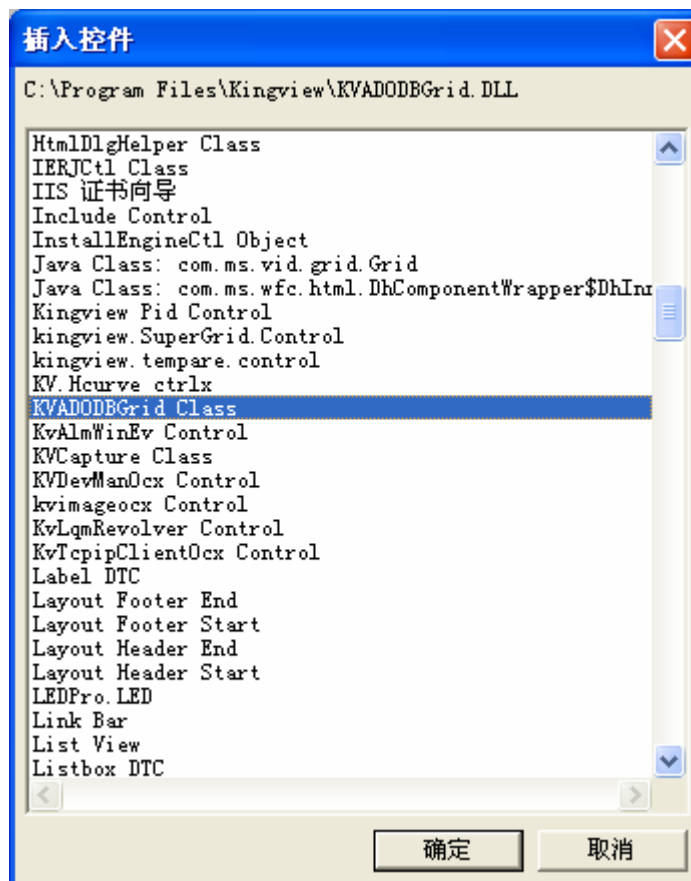
4、历史报警查询：

前面我们已经将报警存储到数据库中了，下面我们就介绍一下如何对存储到数据库中的报警进行查询。我们根据日期、报警组为例进行报警的查询。

历史报警的查询主要是利用 KVADODBGrid 控件进行查询。

4.1) 创建 KVADODBGrid 控件：

在工程中新建画面“报警查询”，单击工具箱中的“插入通用控件”按钮则弹出“插入控件”对话框。在“插入控件”对话框内选择“KVADODBGrid Class”控件，如图十七所示，在此画面中放入此控件。双击此控件，为控件命名，控件名称可以根据需要确定，我们命名为“KV”。



图十七 插入通用控件

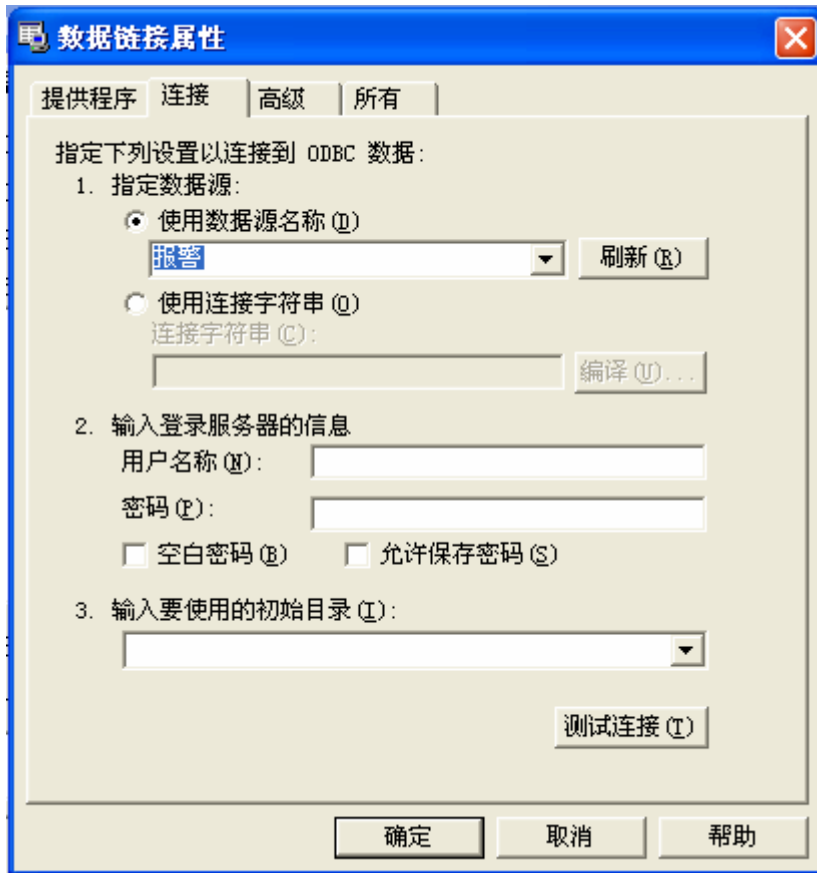
选择控件，单击右键，在弹出的菜单中选择“控件属性”。弹出控件固有属性对话框，如下图十八所示。

点击“浏览”按钮弹出“数据链接属性”如图十九所示，选择“连接”选项卡，在“指定数据源”处选择“使用数据源名称”选项，通过下拉列表选择我们前面所定义的 ODBC

数据源“报警”，点击“确定”，返回图十八画面，“数据源”与“数据库”连接完成，下面进行数据表的配置。



图十八 KV 控件属性



图十九 数据链接属性

在“表名称”处选择我们需要查询的数据表“Alarm”。选择完成后，数据表的字段会显示在“有效字段”栏，我们可以将需要的字段添加到右边，在添加过程中可以对标题以及格式等进行相应的修改，如图二十所示：点击“确定”完成对 KV 控件的配置。



图二十 KV 控件配置

配置完成后，同时按下键盘的“Ctrl”+“Alt”+“O”可以对控件的列宽进行设置，我们可以根据字段内容的多少设置合适的列宽以增加画面的美观程度。设置完成后的画面如图二十一所示：

报警日期	报警时间	变量名称	恢复日期	恢复时间	事件类型	报警类型	报警组

图二十一 运行系统画面

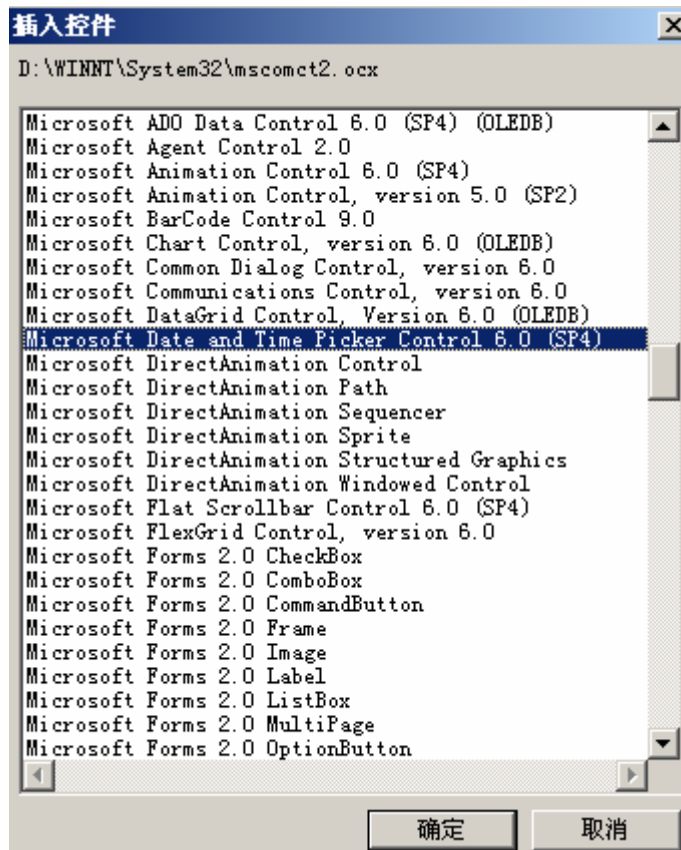
4.2) 创建日历控件：

我们按照日期进行历史报警的查询，使用微软提供的通用控件“Microsoft Date and Time Picker Control”，此控件在安装 VB 或者 VC 或者 Office2000 后会在通用控件中找到。

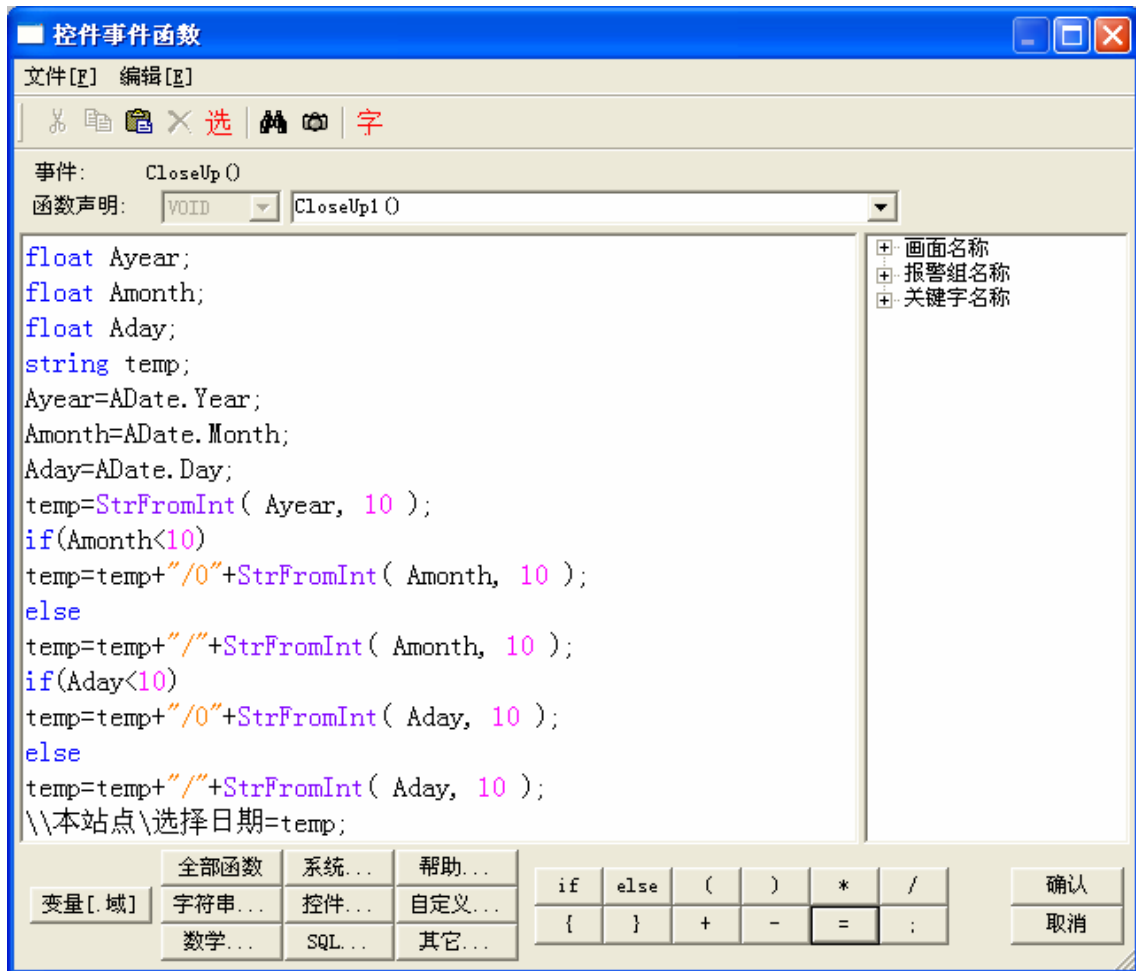
插入通过控件，如图二十二所示：选择后画到画面上，双击控件，在“常规”选项卡中为控件命名为“ADate”，点击“确定”，保存画面。再次双击日历控件，选择“事件”选项卡，在“事件”选项卡中点击 CloseUp 事件，弹出控件事件函数编辑窗口，在函数声明中为此函数

命名：CloseUp1()); 在编辑窗口中编写脚本程序，如图二十三所示：在编写脚本程序之前在数据词典中定义字符串变量“选择日期”，编写脚本程序如下：

```
float Ayear;  
float Amonth;  
float Aday;  
string temp;  
Ayear=ADate.Year;  
Amonth=ADate.Month;  
Aday=ADate.Day;  
temp=StrFromInt( Ayear, 10 );  
if(Amonth<10)  
temp=temp+"/0"+StrFromInt( Amonth, 10 );  
else  
temp=temp+"/"+StrFromInt( Amonth, 10 );  
if(Aday<10)  
temp=temp+"/0"+StrFromInt( Aday, 10 );  
else  
temp=temp+"/"+StrFromInt( Aday, 10 );  
\\本站点\选择日期=temp;
```



图二十二 日历控件



图二十三 控件事件函数

编辑完成后点击“确认”，完成对日历控件的设置。

4.3) 编辑其他画面：

如图二十四所示：制作五个按钮，其中各个按钮的命令语言如下：

日期报警查询按钮：按照选择的日期对所选日期的所有的报警进行查询。

```
string whe;
whe="AlarmDate="+\\本站点\\选择日期+"";
```

```
KV.Where=whe;
```

```
KV.FetchData();
```

```
KV.FetchEnd();
```

压力报警查询按钮：按照选择的日期对所选日期的压力报警进行查询。

```
string whe;
```

```
whe="AlarmDate="+\\本站点\\选择日期+" and GroupName='压力报警'";
```

```
KV.Where=whe;
```

```
KV.FetchData();
```

```
KV.FetchEnd();
```

流量报警查询：按照选择的日期对所选日期的流量报警进行查询。

```
string whe;
```

```
whe="AlarmDate='"+\\本站点\选择日期+"' and GroupName='流量报警';
```

```
KV.Where=whe;
```

```
KV.FetchData();
```

```
KV.FetchEnd();
```

报警打印：对查询结果进行打印输出。

```
KV.Print();
```

画面切换：

```
ShowPicture("实时报警");
```

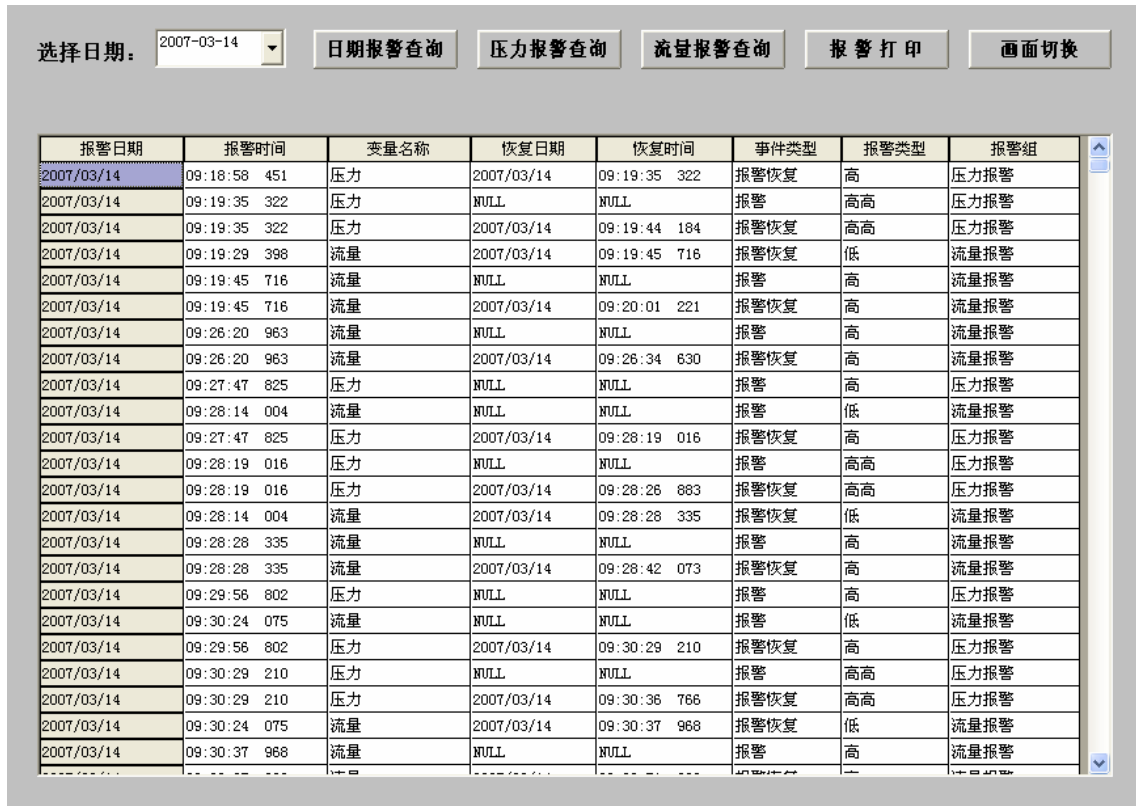


图二十四 报警查询

5、进入运行系统：

完成画面制作并保存后，切换到运行系统，显示如图十五所示的画面，通过“画面切换”

按钮进行“报警查询”画面，如图二十五所示：通过不同的按钮可以实现不同的报警的条件查询。



图二十五 历史报警查询

四、注意事项

1. 如果报警数据库是从光盘中拷贝的，一定要将只读属性去掉。
2. 报警配置中的数据库配置的报警格式设置时，各个字段的长度需要根据实际情况进行设置。
3. 报警配置中如果选择“分月保存报警数据表”，则数据库中的报警表会自动生成，如果不选择此项，则“Alarm”表需要手动建立。
4. KVADODBGrid 控件的详细使用方式请参考组态王手册或者帮助。
5. 报警数据库可以选择 SQLServer 或者其他关系数据库，其他数据库的 ODBC 数据源的定义请参考相关文档。
6. 本演示工程只是报警存储到数据库以及对存储到数据库的历史报警进行条件查询的简单例程，更深一步的学习使用请参考组态王使用手册以及函数手册，或者参加组态王培训班。如有问题请致电亚控公司技术部。