**整个解决方案包括三个项目：**

* WinControler
* Configuration
* SRCRTL

1. **WinControler项目：(生成文件为WinControler.dll)**

WinControler项目是一个类库项目，是整个解决方案的核心类库，其中包括大部分的控制功能：

* 程序及程序功能的控制;
* 命令行功能控制
* 自定义功能控制
* 窗口控制。

**类库中包括的类型有：**

控制层类：

Controler

ApplicationControler

CustomedFuncControler

CommandLineControler

WindowControler

WinControler

数据控制层类：

DataAccess

实体层类：

App

AppFunction

CustomedFunc

CommandLine

VicCmd

详细类结构及说明见[类图](类图.png)。

Controler是功能控制的抽象基类，由其派生出的类有：

程序及程序功能控制器（ApplicationControler）;

自定义功能控制器(CustomedControler);

命令行功能控制器(CommandLineControler);

窗口功能控制器(WindowControler)。

利用继承结构的目的是利用其多态性,以在WinControler中进行一致性地调用各功能控制器。

DataAccess类负责数据库的操作，包括对数据库的增、删、改、查;

实体层的类对象对应数据库中表的记录:

App对应表App

AppFunc对应表Func

CustomedFunc对应表CustomedFunc

CommandLine对应表CommandLine

VicCmd对应表VicCmd

数据库表解析见数据库说明。

对应类按功能分组为：

**程序及程序功能控制：**

ApplicationControler

App

AppFunc

**自定义功能控制：**

CustomedControler

CustomedFunc

**命令行功能控制：**

CommandLineControler

CommandLine

**窗口功能控制：**

WindowControler

另外此类库项目中还包括配置功能的窗体，用于配置管理。即类：Forms/MainConfigForm

1. **SRCRTL项目：(生成文件为SRCTRL.exe)**

此项目包含：

* 语音控制程序的主界面
* 语音识别，识别后通过调用WinControler.dll中的功能类实现相应控制。
* 鼠标定位
* 鼠标按键控制

类：

frmMain:主界面类，其中包括语音识别和发送功能控制命令

GridLineForm：网格空格，显示网格、定位命令识别、鼠标控制。

1. **Configuration项目：(生成文件为Configuration.exe)**

配置管理程序项目，其中配置功能窗体的定义在类库项目WinControler中.

1. **数据库说明：**

表：

|  |  |  |
| --- | --- | --- |
| 表 | 字段 | 说明 |
| **App**  （存储应用程序配置信息） | Cmd | 命令代码 |
| Desc | 描述 |
| Path | 程序启动路径 |
| **Func**  （存储程序功能配置信息） | Cmd | 命令代码 |
| Keys | 功能快捷键 |
| App | 所属程序的命令代码 |
| Desc | 功能描述 |
| **VicCmd**  (语音命令配置信息，存储所有语音命令，语音识别引擎的语法) | Desc | 语音命令 |
| Cmd | 命令对应的命令代码 |
| **CommandLine**  (存储命令行配置信息) | Description | 命令功能描述 |
| CmdLine | 执行功能的命令行 |
| Command | 命令代码 |
| Confirm | 标志位，标志在执行命令前是否需要确认 |
| **CustomFunc**  (自定义功能，通过快捷键实现) | Keys | 快捷键 |
| Desc | 功能键描述 |
| Command | 命令代码 |

注意：[功能模块及整体结构图](功能模块及整体结构图.png) 和 [整体流程图](整体流程图.png) 为前期设计,与最后的作品不完全一致。

1. **参考资料：**

# Using Keyboard Input      <http://msdn.microsoft.com/en-us/library/ms646268(v=vs.85).aspx#processing_keystrokes>

# Virtual-Key Codes     <http://msdn.microsoft.com/en-us/library/dd375731(v=vs.85).aspx>

# About Messages and Message Queues     <http://msdn.microsoft.com/en-us/library/ms644927(v=vs.85).aspx#windows_messages>

# Keyboard Accelerators     <http://msdn.microsoft.com/en-us/library/ms645526(v=vs.85).aspx>

# MSG Structure      <http://msdn.microsoft.com/en-us/library/ms644958(v=vs.85).aspx>

# WM\_LBUTTONDOWN Message     <http://msdn.microsoft.com/en-us/library/ms645607(v=vs.85).aspx>

# Marshal 类 (托管与非托管代码交互的方法集)     <http://msdn.microsoft.com/zh-cn/library/system.runtime.interopservices.marshal(v=VS.100).aspx>

# INPUT Structure     <http://msdn.microsoft.com/en-us/library/ms646270(v=vs.85).aspx>

# MOUSEINPUT Structure     <http://msdn.microsoft.com/en-us/library/ms646273(v=vs.85).aspx>

# KEYBDINPUT Structure     <http://msdn.microsoft.com/en-us/library/ms646271(v=vs.85).aspx>

# Windows Data Types     <http://msdn.microsoft.com/en-us/library/aa383751(v=vs.85).aspx>

**PowerBASIC Information Center Tutorials**<http://www.garybeene.com/power/pb-tutor-msg-keyboard.htm>

mouse\_event          <http://msdn.microsoft.com/en-us/library/ms646260(v=vs.85).aspx>

<http://www.sqlite.org/>

<http://www.sqlite.com.cn/>