

## Help Index



[Copyright Notice](#)

[What is VBRC?](#)

[Who needs VBRC?](#)

[What VBRC can do](#)

[What VBRC cannot do](#)

[What you need](#)

[What is included in the package](#)

[Installation](#)

[Using VBRC](#)

[The TEST.EXE program](#)

[Registration Information](#)

## Copyright Notice

VBRC is a shareware product. You are free to copy and evaluate VBRC, but are expected to register the product upon continued use. Registration entitles the user to certain benefits. See [Registration Information](#)



VBRC (c) Boru Engineering, Inc. 1992

## What is VBRC?

VBRC is a system which converts Visual Basic FORMs into compilable Windows dialog templates

## Who needs VBRC?

VBRC is essential for:

- 1. Developers who must port Visual Basic apps to other languages, and want to preserve as much of the screen information as possible.**
- 2. Designers who prototype in Visual Basic to establish the application's look and feel. VBRC saves the often substantial time invested in re-drawing the forms in a dialog editor.**
- 3. Anyone who would prefer to use Visual Basic as their primary dialog editor.**

## What VBRC can do

1. Preserve the precise coordinates of FORMS in the Windows Resource format.
2. Automatically identify most built-in Visual Basic control types.
3. Minimize the manual steps required to move from a Visual Basic FORM to Windows dialog template.

## What VBRC cannot do

A Visual Basic FORM represents the combination of resource and code elements normally found in procedural Windows development languages such as C and PASCAL. As such, any representation of dialog templates alone will not define attributes which are controlled through coding. These include but are not limited to:

- 1. Settings of individual text fonts for child controls in a FORM.**
- 2. The painting of bitmaps within a window area.**
- 3. Inclusion of special control types such as Drive or File Directory list boxes, which don't exist as Windows primitives.**

Other attributes which can be set within the \*.dlg file but are stored internally in Visual Basic and thus must be set by modifying the VBRC resource script:

- 1. Edit control attributes such as justification. Example: a Text Box has been set for Right Justification in Visual Basic. To duplicate this in the \*.dlg script, simply change the control attributes from WS\_CHILD to WS\_CHILD | ES\_RIGHT, and re-run TESTMAKE.BAT.**

## What you need

It is assumed you already have:

- 1. MS Windows version 3.0 or higher**
- 2. Visual Basic, or (\*.exe) files generated through VB which are executable with the VB runtime DLL.**
- 3. Any development environment that supports compilation of standard (\*.rc) resource files to (\*.res) binary files.**

## What is included in the package

### Application Files:

<b>vbrc.exe</b>	Application executable.
<b>vbrc.dll</b>	Application DLL.
<b>vbrc.hlp</b>	Hypertext help file for Windows.
<b>readme.txt</b>	A short summary of VBRC.
<b>register.txt</b>	Registration form.
<b>commdlg.dll</b>	Windows common dialog box DLL for < Win 3.1 users. See <a href="#">COMMDLG.DLL</a>

### TEST.EXE Files:

<b>test.exe</b>	The TEST executable. Displays a dialog box when run.
<b>test.rc</b>	The user edits this file to include a VBRC FORM.
<b>test.ico</b>	Icon file.
<b>testmake.bat</b>	Re-builds TEST.EXE based on users' *.dlg file.
<b>form1.dlg</b>	Sample output from VBRC, gets compiled into TEST.EXE.
<b>form1.mak</b>	Sample VB project. Has one form to practice on.
<b>form1.frm</b>	Sample VB FORM.
<b>form1.exe</b>	Sample VB executable, created from form1.mak.



## Installation

1. Make a new directory, preferably named VBRC.
2. Copy the VBRC distribution (\*.zip) file to the directory.
3. Unzip the file in place.(type pkunzip VBRC10)
4. If you have Windows 3.1 or higher, you can delete the file COMMDLG.DLL, provided there is already a copy in your \WINDOWS\SYSTEM directory

## Using VBRC

Starting VBRC

Loading a Form

Starting the conversion

File Save As dialog box

Inspecting the output file

Exiting VBRC

## Starting VBRC

The following are three possible ways to launch VBRC:

- **Double click on VBRC.EXE in File manager**
- **Select File-Run from the File Manager menu and type vbrc (Provided VBRC.EXE is in a directory listed in your DOS PATH statement)**
- **Create a program icon in Program Manager and double click it.**

## Loading a FORM

There are three ways to load a FORM into the VBRC window. All require that the target FORM is active on the desktop:

- 1. Start the Visual Basic environment and open the desired project file. Then select any of the FORMS listed in the project.**
- 2. From the Visual Basic environment, select RUN for the project, and bring up any FORM to the screen by running the project.**
- 3. Run an executable (\*.exe) file created with Visual Basic and bring up a FORM(s) as above.**

VBRC automatically detects FORMS which are activated in Windows by any of the above methods.

## Starting the conversion

FORMS which have been activated will be visible in the VBRC window. The FORM'S caption is used as its identifier. Highlight a the FORM using the mouse or the up/down arrow keys. Then, either:

- Press {ENTER}.
- Click the {OK} button.
- Double click on the highlighted FORM in the VBRC window.

This will bring up the File Save As dialog box which is used to name the destination file, it's location, and how the user wishes to number child controls in the dialog template.

## File Save As dialog box



This dialog box allows the user to configure:

- 1. The name of the destination file generated from the Visual Basic FORM.**
- 2. The location directory of the destination file.**
- 3. The starting offset of the Control ID's which are created to name the child controls.**

Either type in a new file name (the default extension is \*.dlg), or highlight an existing file to overwrite it. Once a valid file name is selected, it should appear in the File edit box at the top of the dialog.

To change the default starting Control ID value, type in an integer value  $\leq 64000$  in the edit field at the lower right of the dialog box. The default value is 100.

To initiate the conversion process, either:

- Click on the {OK} button.**
- Double click on the filename.**

The File Save As dialog box will then disappear, indicating file conversion is in progress. When the file is complete, the new dialog template will be loaded into Notepad for inspection

## Inspecting the output file

The output file is a standard Windows resource file containing a compilable dialog template.

Items to check:

- 1. The dialog name is the first word in the template preceding "DIALOG". It is generated from the FORM caption by uppercasing all letters and adding underscores wherever blanks are found.**
- 2. Each addressable child control in the dialog template requires an integer id to be addressed by Windows API calls. Serial values are thus assigned by VBRC, starting at a default value of 100, or the value entered in the File Save As dialog box. Your Windows code will need to use these values when making API calls to the controls. These values may later be changed into precompiler #defines depending on the needs and size of the project.**

## Exiting VBRC

Either of the following will close VBRC:

- Clicking the {Quit} button.
- Pulling down the system menu (top left corner) and selecting Close.
- Pressing {ALT-F4}



## The TEST.EXE program

Purpose of TEST.EXE

Making TEST.EXE

Running TEST.EXE

## Purpose of the TEST.EXE program

This is a simple Windows program which displays the contents of a dialog template (\*.dlg) such as might be created using VBRC and draws a single dialog box on screen.

TEST.EXE is included in VBRC to:

- 1. Allow a convenient means of viewing the compiled results of a VBRC output file (dialog template).**
- 2. Demonstrate some of the interaction required between Windows C code and the dialog resource to achieve the look and feel of a VB FORM. See [Benefits of Registration](#)**

## Making the TEST.EXE program

Before building the TEST.EXE executable, be sure to check two important parameters:

- 1. TEST.RC contains an rcinclude statement. The named file should point to your VBRC output file (\*.dlg). The default name is "form1.dlg". If the file name of the target template file is different, modify the line to reflect that name.**
- 2. TEST.RC references a name from the STRINGTABLE, which points to the DIALOG resource which has been created from your FORM. This must be the same as the name found prior to the DIALOG keyword at the beginning of the VBRC output file (\*.dlg). The default name is "FORM1".**

If either of the above are not correct, TEST.EXE will be unable to display the FORM onscreen.

To incorporate a newly created dialog template (\*.dlg):

**From the command line, type {TESTMAKE}.**

## Running the TEST.EXE program

If any compilation errors are encountered making TEST.EXE, double check items #1 and #2 under Making the TEST.EXE program.

Once compiled, run the TEST.EXE program using methods similar to those found in Starting VBRC, substituting the TEST.EXE name in those commands.

TEST.EXE has no main window and simply creates a dialog box using your FORM layout derived from VBRC. TEST.EXE assumes you have chosen the default Visual Basic Font (Helv08) for all child controls, and assigns that font to all controls in the FORM.

## **Registration Information**

**Benefits of Registration**  
**Registration Form**

## Benefits of Registration

Registering VBRC with Boru Engineering entitles you to:

- 1. The right to continue using VBRC.**
- 2. A copy of the latest version of VBRC on disk.**
- 3. The C source code to the TEST.EXE program (VRBC source is not currently available).**
- 4. Notification of future updates, along with reduced registration fees.**

To register, see [Registration Form](#), and select Print Topic from the Windows Help Menu

**REGISTRATION FORM  
VBRC ver. 1.0**

\*\*\*\*\*

Registering VBRC entitles you to:

- 1. The right to continue using VBRC.**
- 2. A copy of the latest version of VBRC on disk.**
- 3. The C source code to the TEST.EXE program  
(VBRC source is not currently available).**
- 4. Notification of future updates, along with reduced  
registration fees.**

\*\*\*\*\*

NAME \_\_\_\_\_

STREET \_\_\_\_\_

CITY \_\_\_\_\_

STATE \_\_\_\_\_ ZIP \_\_\_\_\_

TELEPHONE NUMBER \_\_\_\_\_

Qty	Unit Price	Total
_____	VBRC registration	\$45.00 _____

\*\*\*\*\*

Specify preferred media: 3.50" \_\_\_\_\_ 5.25" \_\_\_\_\_

Make checks payable to: Boru Engineering, Inc.

Mail to:

Boru Engineering, Inc.  
2150 Walnut Place  
Cinnaminson, N.J 08077

COMMDLG.DLL

Special Dynamic Link Library first included with Windows 3.1. Required to run VBRC, a copy of this file is distributed with the package for the benefit of Windows 3.0 users.



## Dialog Template

An ASCII text file which defines the characteristics of dialog box objects to the Windows Resource Compiler. A typical dialog template would appear as follows:

```
FORM1 DIALOG LOADONCALL MOVEABLE DISCARDABLE 71, 48, 146, 70
CAPTION "Form1"
STYLE WS_OVERLAPPED | WS_CAPTION | WS_SYSMENU | WS_THICKFRAME |
WS_MINIMIZEBOX | WS_MAXIMIZEBOX
BEGIN
    CONTROL "List1", 107, "listbox", WS_CHILD | LBS_STANDARD, 72, 28, 56, 7
    CONTROL "Combo1", 106, "combobox", WS_CHILD | CBS_HASSTRINGS |
CBS_AUTOHSCROLL | CBS_DROPDOWN, 12, 24, 32, 10
    CONTROL "Test FORM for use with VBRC", 101, "static", WS_CHILD, 28, 16, 92, 8
    CONTROL "Cancel", 102, "button", WS_CHILD | BS_PUSHBUTTON, 20, 44, 44, 12
    CONTROL "OK", 103, "button", WS_CHILD | BS_PUSHBUTTON, 80, 44, 44, 12
END
```

The CONTROL entries represent the size, positioning and type of child windows (controls) drawn within the dialog box. These correspond to the objects drawn on a Visual Basic FORM.