*Tutorial on Modifying*

*EPRI OpenDSS Source Code*

*Alejandro Flores*

*Instructor: Dr. Olga Lavrova*

*ECE 595 Summer 2014*

# Abstract

In recent years, the increase in residential/commercial use of solar panels and the imminent increase in electrical vehicles hitting the roads are presenting new challenges to the current power distribution infrastructure. Voltage regulators and inverters, to mention a few, must evolve into devices that can adjust power in a matter of seconds.

To make these changes possible, one would want to run an OpenDSS simulation that can help analyze the system as a whole. However, some of the functionality needed for these simulations is still not available in OpenDSS. For this reason, this paper attempts to provide some guidance on how to edit and recompile the OpenDSS source code.

In this tutorial, we will go over some the history on how OpenDSS came about. We will then go over the essential tools needed to obtain and modify the source code. Finally, we will take a tour of the Delphi EX6 development solutions application.

The main idea behind this paper is to provide the essentials that can enable a developer to quickly get going with making changes to the OpenDSS source code.

# Background

In 1997, Roger Dugan and Tom McDemott, who worked at Electrotek Concepts, developed DSS, an electrical power systems simulation tool for utility power distribution systems (1). DSS was intended to provide a versatile research platform that could be used to analyze special distribution applications. The Electric Power Research Institute (EPRI) purchased the application in 2004 and in 2008 released and open source version called OpenDSS. Since then, it has become one of the most popular tools to model electric power distribution systems and perform a variety of distributed power and power quality analysis.

From the start, Roger Dugan has used Pascal programming language to develop DSS and now OpenDSS. In 1997, the Pascal programming language was used as part of an innovative move but with the years it turned more into a practical solution of programming (1). At any rate, know that basic knowledge of Delphi Pascal (aka Delphi) is a must.

# Getting Started

To begin the process of working with the OpenDSS source code, one must first have a subversion (SVN) client such as TortoiseSVN to download a copy of the source code. A quick search on Google should yield a result of a free SVN client.

Additionally, a Delphi compiler is perhaps the most essential and obvious requirement. The preferred development tool for this project is Delphi EX6 made by Embarcadero (<https://downloads.embarcadero.com/free/delphi>). However, if your project has budgetary constraints, you could alternatively use an open source options such as Lazarus Delphi compiler. Regardless of your selection, the compiling process should not be too involved. For this tutorial, I have decided to use the 30-day trial of Delphi EX6 because of its flexibility and ease of use.

## Source Code

EPRI makes the source code for OpenDSS available directly from their website but know that only key folders and files are needed to successfully compile the code.

To get the code, right click on your desktop and select TortoiseSVN then Repo-browser. Navigate to: <http://svn.code.sf.net/p/electricdss/code/trunk/Source> and simply drag and drop the folder you need onto a folder of your choice. The image below illustrates the main TortoiseSVN browser interface.

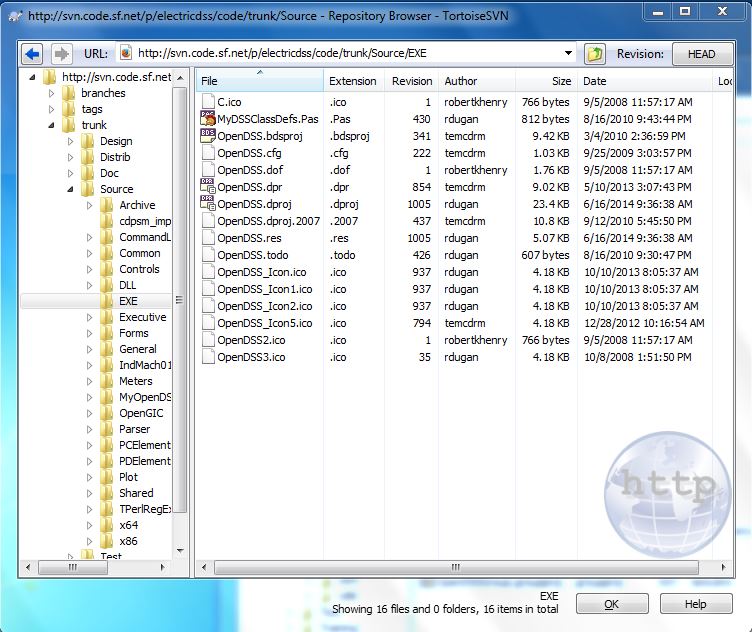


Figure - TortoiseSVN Browser

As previously mentioned, only certain folders are needed to recompile the source code. The table below lists the folders and files needed for this project.

|  |  |  |  |
| --- | --- | --- | --- |
| Common | General | PDElements | 64bit |
| Control | Meters | Plot | Meter General |
| Executive | Parser | Shared | OpenDSS3.ico |
| Forms | PCElements | 32bit | KLUSolver.dll |

Figure - List of Folder to Copy

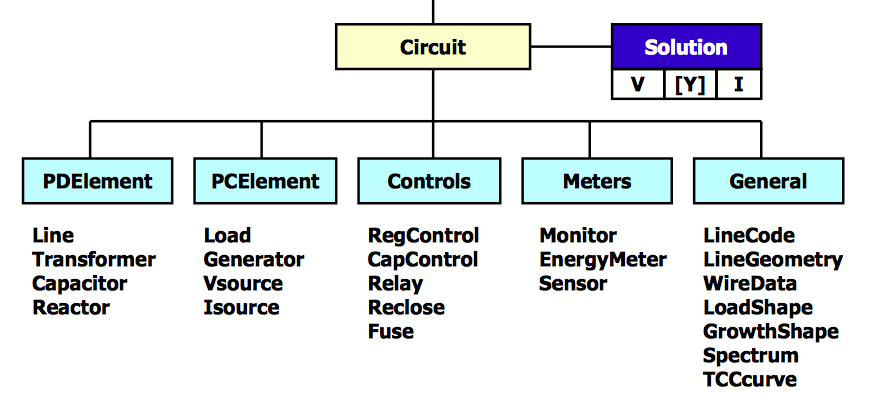
Furthermore, the diagram below helps locate some of the key files within each of the downloaded folders.

Figure – OpenDSS Key Files

Additional documentation related to the code can be found at the EPRI code repository site <http://svn.code.sf.net/p/electricdss/code/trunk/Distrib/Doc/>.

## Delphi EX6

With a good understanding of what the code structure is, we can move forward with opening the source code with Delphi EX6. To do this, open Delphi EX6 then go to “File” then “Open Project” and navigate to “EXE/OpenDSS.bdsproj”; this will open the entire project. At first, Delphi EX6 might seem a little overwhelming therefore I will attempt to provide a brief explanation of the different panels.

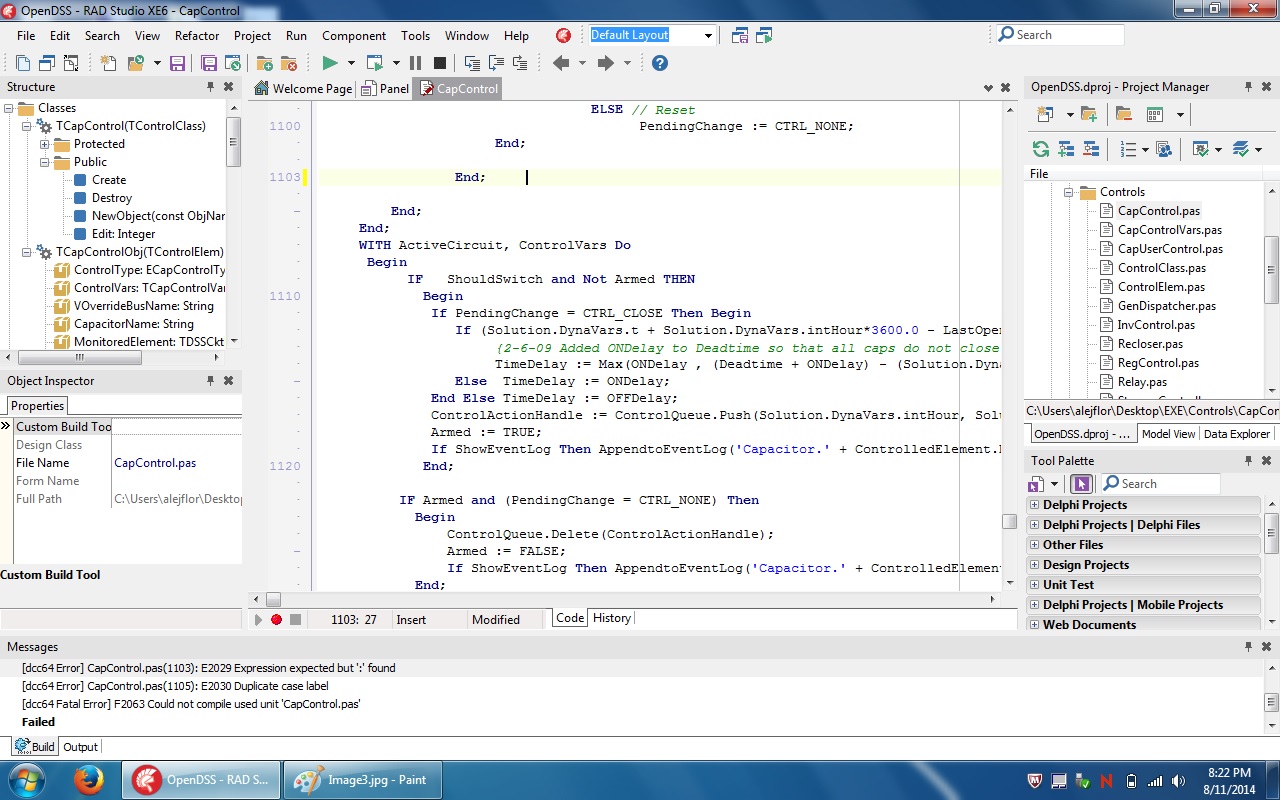


Figure - Delphi EX6 Main Window

The center panel displays the active file to be modified. Multiple files can be opened in different tabs but only the visible tab is considered the active file.

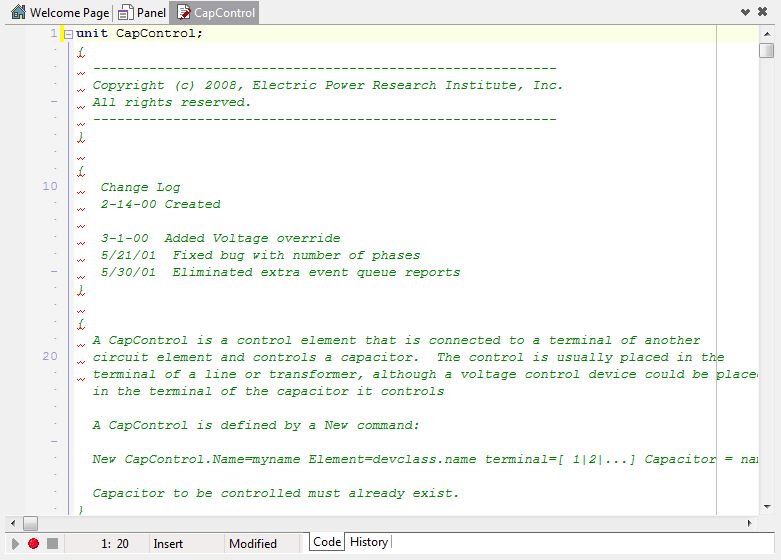


Figure - Delphi EX6 Main Panel

The panel on the top right is the Project Manager panel that displays the file structure within the OpenDss.exe executable. In this list, you can also find the required files by expanding the folder icon.

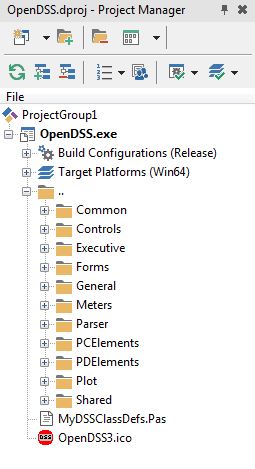


Figure - Delphi EX6 Project Manager

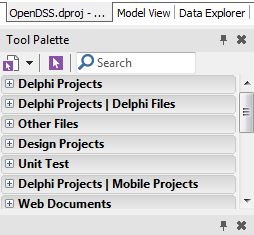


Figure - Delphi EX6 Tool Palette

The bottom right panel will contain the Tool Palette that contains the many tools used for the front-end development.

To the left, you will find the list of classes and variables used in the currently active file. Below that, is the Object Inspector in which you can configure and/or make changes to the properties of a particular tool inside a form.

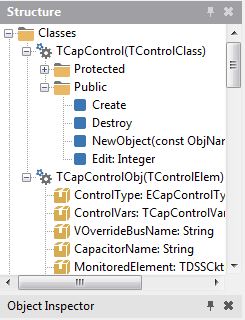


Figure - Delphi EX6 Structure

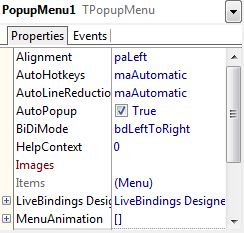


Figure - Delphi EX6 Object Inspector

Finally, the lower messages panel will become visible after you run the code for the first time. The messages panel will display any errors associated with the code. This panel will perhaps be the most used of all.

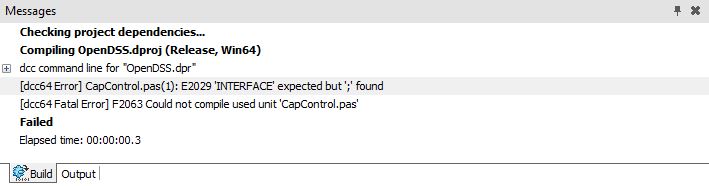


Figure - Delphi EX6 Message Panel

## Compiling

Now that we’ve covered the basics, use the Project Manager panel to double click on the file you wish to modify. This will cause the file to become active on the main panel. Make the desired modification to the code and compile the code by going to the Run menu then select Run. Alternatively, the code can be recompiled by using the F9 key.

If the code successfully compiles, the application will automatically execute and open. If errors are found, the Message panel will display a list of errors to correct. Conveniently, one can go to the section of code that contains the error by simply clicking on the error within the message panel.

Once a program has successfully compiled, the program can be manually executed by going either the x64 and or x86 folder. Please note that the KLUSolver.dll file must be in the same directory as your exe file, otherwise the program will fail. The exe and KLUSolver files can now be distributed to your testers.

There is obviously way more features available within Delphi EX6 that can assist in troubleshooting your code but covering those fall outside the scope of this tutorial.

## Commenting

As with any programing project, documenting your changes is extremely important. Throughout the code, you will find log comments at the beginning of each file. Take the time to create an entry and briefly explain your enhancement.

# Conclusion

Hopefully this tutorial helped in in gaining the basic understanding of how to modify and recompile the OpenDSS source code. One important aspect to keep in mind when working on enhancement is to maintain the solver separate from the rest of the OpenDSS code(2). As per EPRI, maintaining the solver separate from the code allows for a cleaner code.

# Works Cited

1. Dugan, Roger.”OpenDSS Introductory Training Level1.”April 2009. PowerPoint presentation
2. Dugan, Roger, EPRI Advisor , June 9, 2014
3. Lavrova, Olga, UNM ECE Advisor, June 2, 2014
4. “Electric Power Research Institute.” Smart Grid Resource Center, Web June. 2014.