

**Umeå University**  
Department of Applied Physics and Electronics

**Linux as Development Environment 7.5 ECTS**  
**5EL142 HT-16**

**Assignment 12 - Graphical Development Environments - Eclipse**

Submitted 2017-08-08  
Author: Lorenz Gerber, 20161202-2033 (lorenzottogerber@gmail.com)  
Instructor: Sven Rönnbäck, John Berge, Björne A Lindberg

## 1 Introduction

The aim of this assignment was to learn about graphical development environments such as Eclipse. To achieve certain exposure to Eclipse, two practical exercises had to be conducted. First, the GTK+ electrotest application from lab 11 had to be setup for development in Eclipse and extended with a button to run the given test library linumtest.h. Eventual bugs had to be found and corrected. For the second part, a C++/GTK+ 'Mandelbrot' application had to be set up in Eclipse and extended with zoom in/out functionality.

The general idea about 'Graphical Integrated Development Environments' is to provide language tailored, advanced tooling for developing larger software projects. Some examples of such tools are code completion, graphical debugging, graphical library management, dependency checking, versioning etc. Usually, such graphical IDE's such as eclipse simply integrate available console tools with a graphical possibility to choose the options.

## 2 Method

Here it was chosen to install 'eclipse neon' using the 'Oomph' installer. The 'Oomph' installer is a separate piece of software that provisions the installation of 'eclipse' development environments. After startup of 'oomph', 'Eclipse CDT' the eclipse version for C/C++ development is chosen, then the whole IDE is installed automatically.

### 2.1 electrotestgtk

Setup of the electrotestgtk project for development was conducted according to the instructions given on the course homepage in the following order:

1. 'Setting up projects in Eclipse'
2. 'Configuration of Eclipse Projects for GTK+-2.0'
3. 'Installation and Use of own Libraries in Eclipse'

The first step, setting up a project was followed as described. For the GTK+-2.0 configuration, some additional steps were needed that the libraries were found correctly. Namely, '/usr/include/gtk-2.0' and '/usr/include/glib.2.0' had to be added in 'Properties' -> 'C/C++ Build' -> 'Settings' -> 'Cross GCC compiler' -> 'includes'. Similar, '/usr/share/glib-2.0' was added in 'Properties' -> 'C/C++ Build' -> 'Settings' -> 'Cross GCC Linker' -> 'Libraries'.

Installation of own libraries worked fine according the instructions. Prior to installation, new library versions with debugger flag (-g) set were generated.

After including the linumtest library, it was found that the component library did not adhere to the given function names, hence, this was corrected. Then the button and a callback function was added to the the electrotestgtk application.

Debugging was conducted by graphically setting breakpoints, starting the debug process by clicking the respective button and then using 'step over' and/or 'step into' functionality while inspecting the values of variables.

### 2.2 Mandelbrot

Setting up of the Mandelbrot project was correspondingly to the 'electrotestgtk' project except for choosing a 'C++' template. All the 'include' and 'Library' settings were done for the 'g++' build toolchain.

- compiler flags for optimizing code

### **3 Results**

#### **3.1 electrotestgtk**

As mentioned earlier, prior to debugging, the component library had to be recompiled with the correct function names. Then the linumtest program was run. It run without crash, however the test results indicated an error in the component library. Now the debugger was used to step trough the code and analyze the it's function. For lab 6, I was in charge of another library, so I was not fully aware of how the component library works. The test failed in the 'bTestComponent' function. On stepping through, it was found that the second call to 'e\_resistance' with the input parameter '100' expects a return value of '2', however e\_resistance returns 1, hence the test fails. According to my understanding of how E12 resistances are calculated, 100 is part of the series, hence the result should indeed be 1. To amend the problem, the test function was modified to expect '1' as return value. Now all tests passed and the GTK button turns green.

Result part II

### **4 Discussion**

### **5 Conclusion**