ELEC-A7151 Object-oriented programming with C++

Circuit Simulator

Documentation

Team Members:

Henrik Toikka

Kwasi A. Boateng

Patrick Linnanen

Unna Arpiainen

1. Overview

Circuit simulation is a technique of modelling a circuit using mathematical expressions to check and verify the design of electrical and electronic circuits. Our main goal is to develop simulator software with an impressive GUI to aid researchers, students, and engineers understand the basic concepts of electronics.

Our circuit simulator software can load, save, edit and simulate electrical circuits. The program supports the Netlist format. The GUI cannot yet solve the sketched circuit; however, a circuit can be solved by writing a netlist file for the specified circuit which can be loaded in the main.cpp file.

2. Software structure

The software consists of two main components: The solver and the GUI

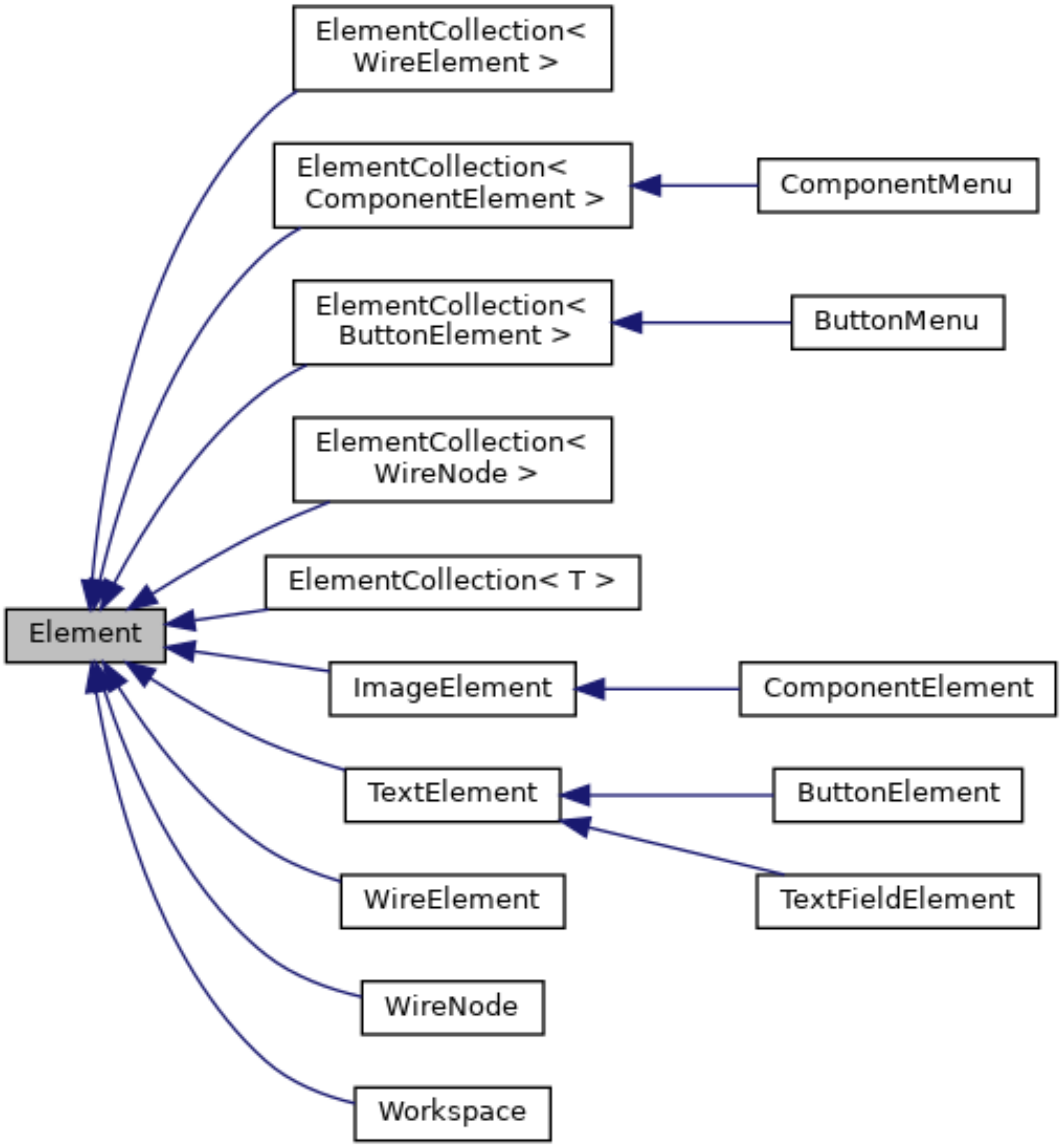
The solver is currently purely a command line interface. The solver was implemented to solve a circuit using the modified nodal analysis (MNA) method. Thus, the solver generates a matrix **A** and a vector **b** from a set of linear equations evolving from each node of the circuit. The solver is responsible to simply solve the matrix equation:

$$Ax = b$$

Where **x** are the voltages from each node.

The GUI is written in a very versatile way. A GUI holds and draws Element types which have a lot of overridable functionality. Generally a Element would override three main functions: Update, Draw and HandleEvent. Many different elements can exist: buttons, menus among many others. Here is a graph of Elements that have currently been implemented.

More about the GUI can be found in **doc/html/classGUI.html**


ElementCollection< WireElement >
ElementCollection< ComponentElement >
ComponentMenu
ElementCollection< ButtonElement >
ButtonMenu
ElementCollection< WireNode >
ElementCollection< T >
Element
ImageElement
ComponentElement
TextElement
ButtonElement
WireElement
TextFieldElement
WireNode
Workspace

3. Build instructions

Note: Follow the linux steps if you are building on WSL. You need to install an X11 server to run the program, so we don't recommend using WSL.

Linux (recommended)

- Tested on Ubuntu 22.04

1. Clone repository

git clone https://version.aalto.fi/gitlab/toikkah2/circuit-simulator

cd circuit-simulator/

2. Install required packages

sudo apt-get update

sudo apt-get install libboost-all-dev libsfml-dev build-essential cmake

3. Compile and Run

./build.sh

Windows 10 (not recommended)

- Tested on Windows 10

1. Install vcpkg and boost (https://vcpkg.io/en/getting-started.html)

cd ~

git clone https://github.com/Microsoft/vcpkg.git

cd vcpkg

./bootstrap-vcpkg.bat

./vcpkg.exe install boost

2. Install CMake (https://cmake.org/download/)

Download "Windows x64 Installer" for example.

Remember to add it to PATH (selection box in installer)
3. Install MSYS2 and Mingw32 (https://www.msys2.org/)

a) Follow the installer and run MSYS2

b) Update packages

pacman -Syu

c) Re-open MSYS2 and update again. Install required packages

pacman -Su

pacman -S --needed base-devel mingw-w64-x86_64-toolchain

d) Add MinGW compiler to your PATH variable

- Navigate to "Environment Variables"
- Add "C:\msys64\mingw64\bin" to System PATH variable

e) Check installation

- Open cmd.exe and run:

g++ --version

4. Clone the repository

git clone https://version.aalto.fi/gitlab/toikkah2/circuit-simulator

cd circuit-simulator/

5. Compile and Run

./build.bat

Basic user guideline

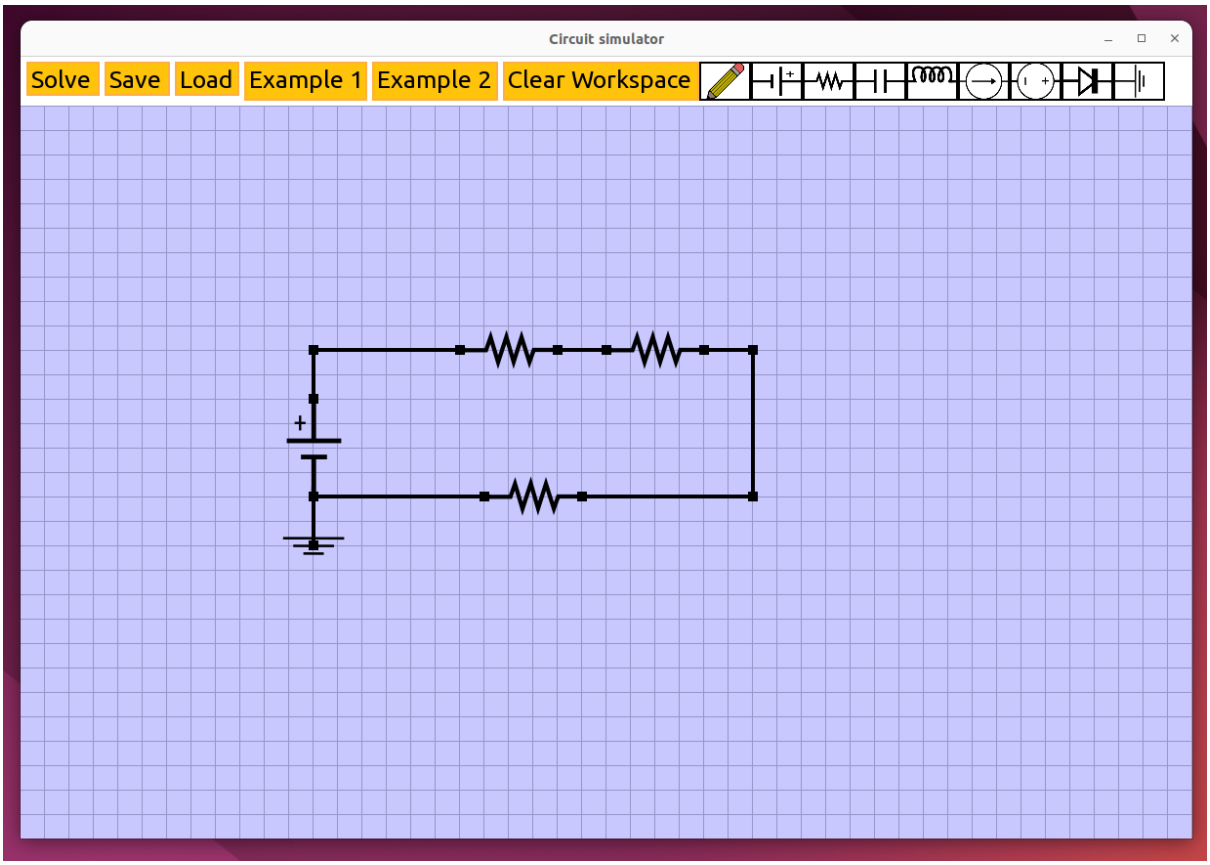
You can select example circuits from the top bar by pressing the button. If you want to create your own circuits you can clear the workspace and start using the components from the component menu.

You can create a component by clicking the component inside the menu and clicking on the workspace. You can move components by holding them with the left mouse button. While holding the components you can rotate them by hitting the ‘R’ key.

Deleting components can be done via clearing the workspace, or individually holding a component and pressing ‘DEL’ key.

Now that you have a circuit of your own you can save it by hitting the Save button on the top left and it will by default be saved in “savefiles/your_circuit.workspace”. You can clear the workspace and load it from the Load button later.

Solve button currently does nothing as the solver is not integrated to the workspace. It should give the node voltages and currents in the standard output.



4. Testing

Testing is done via automatic unit testing using Google's gtest library. Using the build.sh script it will automatically run the unit tests executable and give you the test results.

5. Work log

We communicated through telegram and Zoom meetings. Notes about our meetings can be found in the MeetingNotes.md file.