Circuit Tester Analysis

Rhys Adams

Contents

Stating the problem

The problem

GCSE physics students in DGSB require knowledge of circuits as part of their course, but unfortunately the physical equipment to teach many of these circuits is expensive, limited in the school, is easily broken, and home learning is not possible as equipment cannot be taken outside of school. Most students simply learn about these circuits from an image in a textbook.

Justification for computational solution

The physics calculations are an ideal fit for a computational solution. This is because the amount of maths required is significant, so would be time-consuming for anybody to do to just demonstrate the physics.

These calculations are not required by GCSE level physics, so would be difficult to be shown manually by a teacher who does not need to know or explain the equations. Showing the values of current and voltage at any point is sufficient at GCSE level.

There will be a high level of abstraction, inheritance and polymorphism, as this is needed for a hierarchy of components, with one base component extending off into a dozen other components.

Stakeholders

I will have two stakeholders, who will guide my progress throughout the project:

Persona	Name(s)	Requirements	Availability
Physics Teachers	*To Be Decided*	• Input circuits from textbook • Display on large whiteboard • Save circuits to send students to experiment with/fix at home • Show the current/voltage at each point of the circuit	Weekly
Physics Students	• My Sister (GCSE) • My school peers (A-Level)	• Load Circuits emailed by teachers • Diagnose problems with circuits	Daily

Existing Solutions To The Problem

Qucs

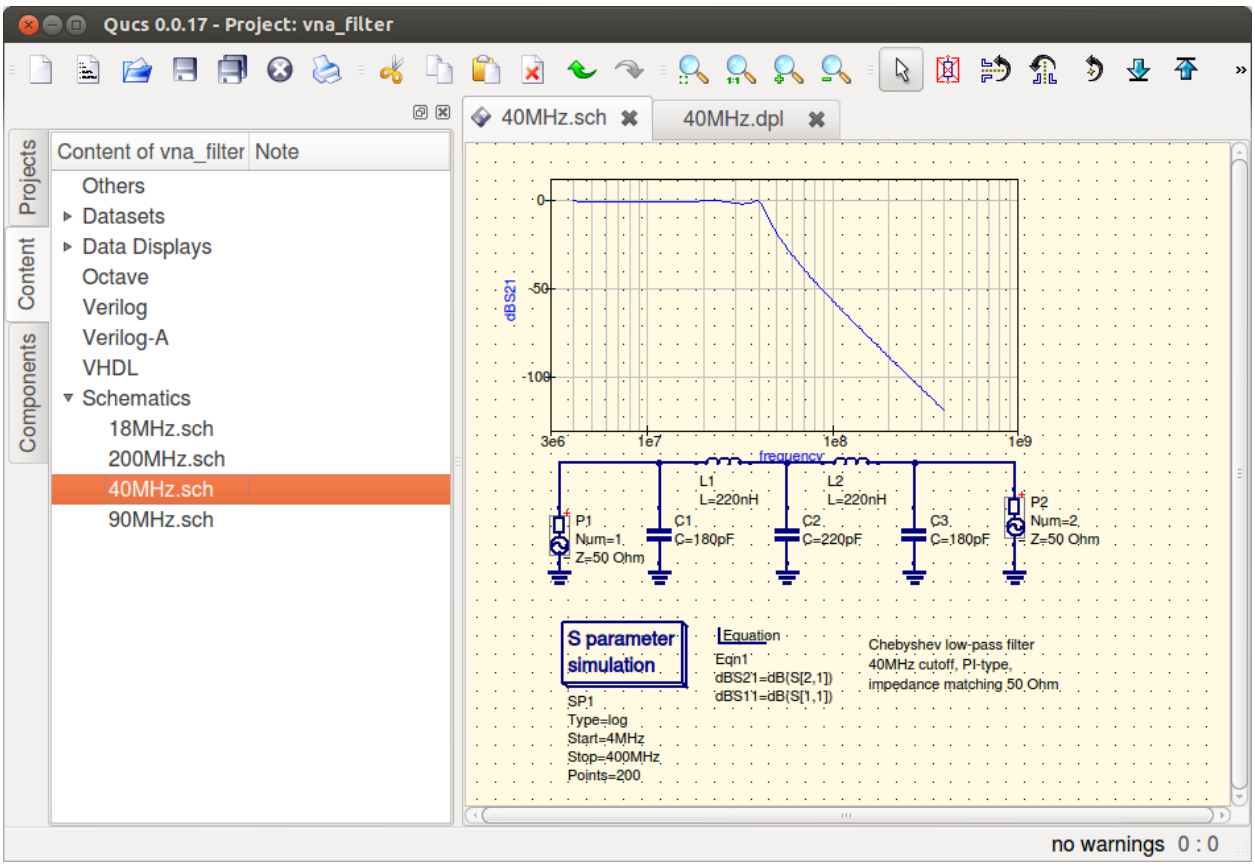

Figure 1: Qucs

What I Like:

- GUI program, not CLI.
- Drag & drop of components onto a canvas.
- Can import/export circuits.
- Can simulate circuits.
- Uses SPICE circuit format.

What I Dislike:

- Massively complicated - menus inside menus with buried functionality.

- Unsuitable for students - many components irrelevant to the GCSE.
- Dated, rarely updated (last updated 2017), uses old tools (QT4)

What I Will Take Forward:

- GUI program.
- Drag & drop components.
- Circuit import/export.
- Circuit simulation ability.

Comments:

QUCS is a nice GUI program that can simulate circuits, although it is very difficult to use for a student, as there are many more components than any student could ever want.

Partsim

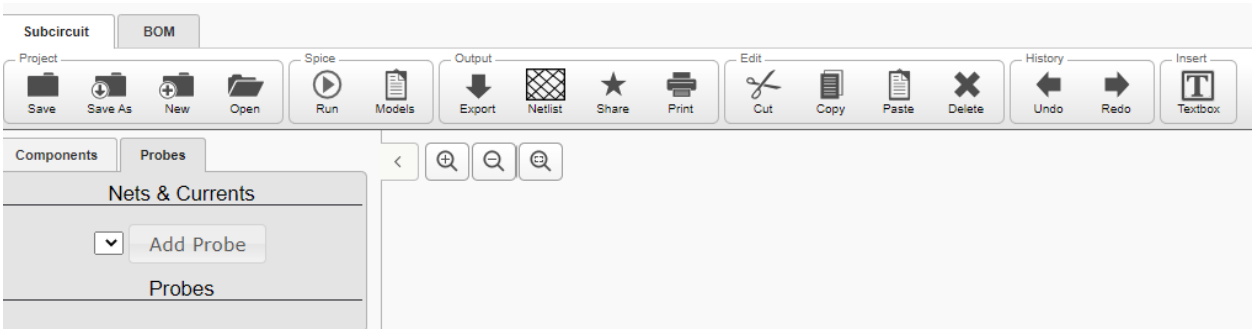

Figure 2: PartSim

What I Like:

- Has less menus than QUCS, is easier to navigate.
- Can import/export circuits.
- Can simulate circuits.

What I Dislike:

- It is a web app, so is not native and uses up a lot of computing power.
- It is made for companies, not students.
- Too many components irrelevant to the GCSE.

What I Will Take Forward:

- Fewer menu options.
- Circuit import/export.
- Circuit simulation ability.

Comments:

Partsim is a webapp which is meant to design and simulate circuits for businesses, so it obviously not meant for students to use, especially GCSE ones.

Webapps generally use a lot more processing power than desktop apps, so this makes the program less accessible to students who may have fewer computing resources.

CircuitLab

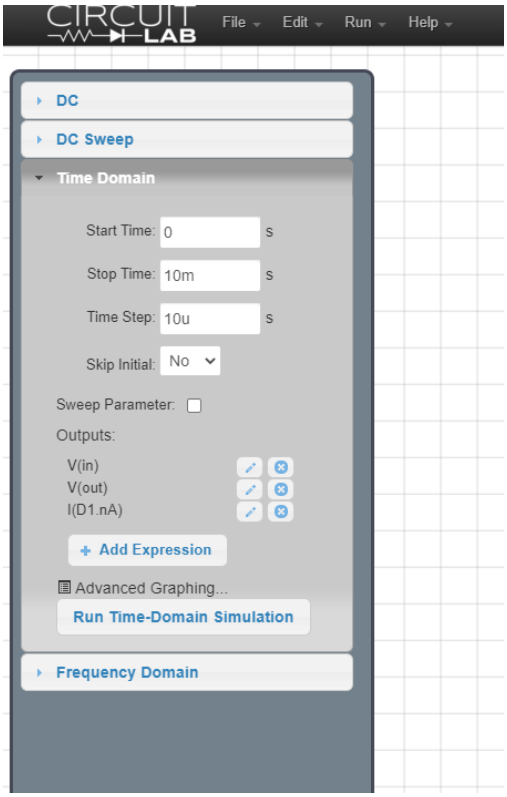

Figure 3: CircuitLab

What I Like:

- Has less menus than QUCS, but more than Partsim.
- Nice layout (scrollable sidebar).

What I Dislike:

- It is a web app like Partsim, so is not native and uses up a lot of computing power.
- It is made for companies, not students.
- Too many components irrelevant to the GCSE.
- Limited version, the full version is £24/person/year.
- Definitely more of a commercial product.
- Simulation button is very confusing, has way too many options.

What I Will Take Forward:

- Free software.
- Scrollable sidebar of components.

Micro-Cap

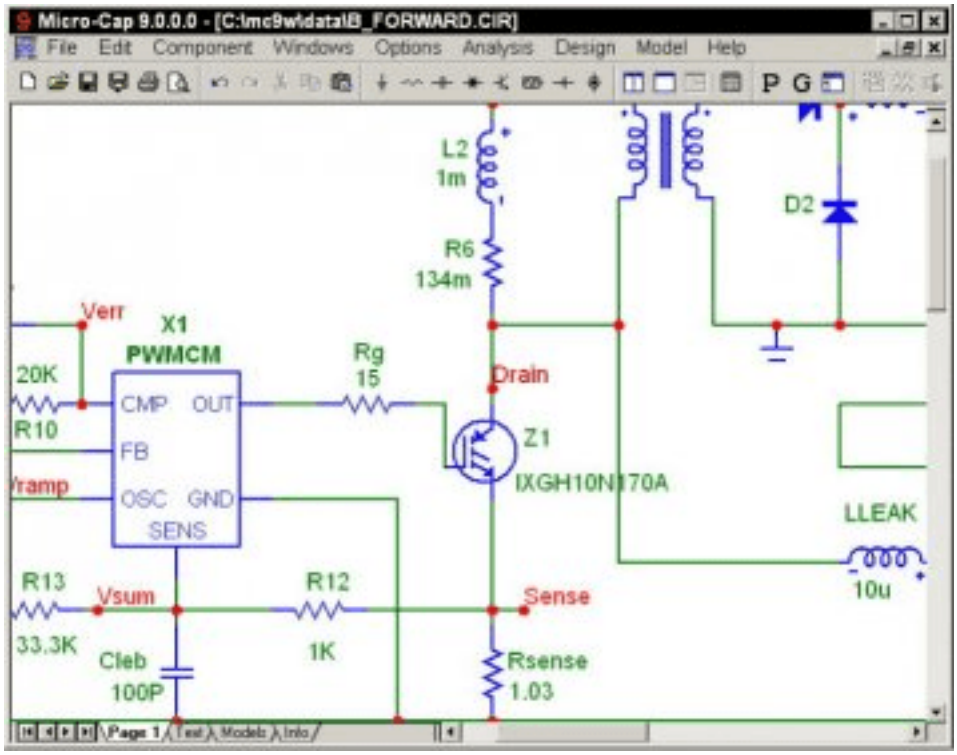

Figure 4: Micro-Cap

What I Like:

- Desktop GUI, not a webapp.
- Uses SPICE circuit format.

What I Dislike:

- Very old, boasts compatability with windows 98
- Very difficult to use, arguably more difficult than QUCS

What I Will Take Forward:

- GUI program

Comments:

This is a very old circuit simulator than until recently cost money to purchase on a CD. Now thought, it is free to use without a license key, so is seeing more adoption, although it is still not as widely used.

Physical Circuits

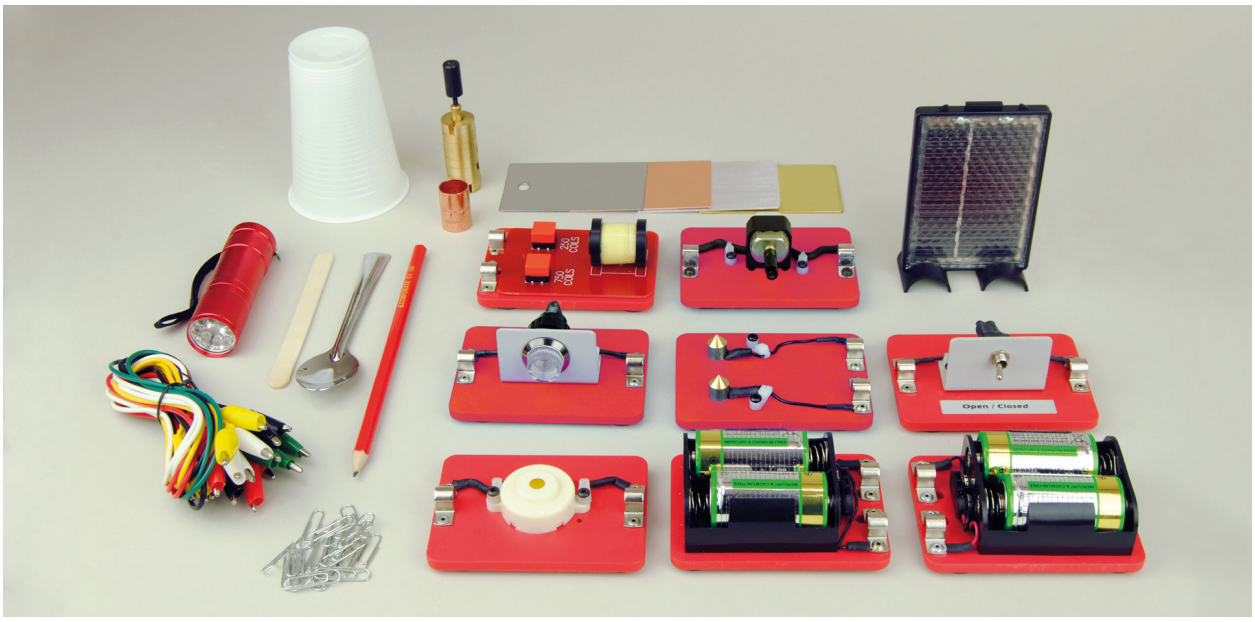

Figure 5: Physical Method

What I Like:

- The only true way to see the effects of circuits.
- Everything is physically correct.

What I Dislike:

- Takes a long time to set up
- Experiment short
- May not be worth the time to demonstrate a simple circuit, teachers will generally use the paper method instead as textbooks will show diagrams of circuits.

Comments:

Of course physical circuits are the only true way to show the effect of the real world on circuits, better than any calculations done by humans or computers.

The downside though is that it takes too long to set up these experiments, when the experiment itself could take just 5 minutes, and the setup could take half an hour. Many teachers will just not want to setup these experiments as it is just too much work for such a little reward.

Paper-based

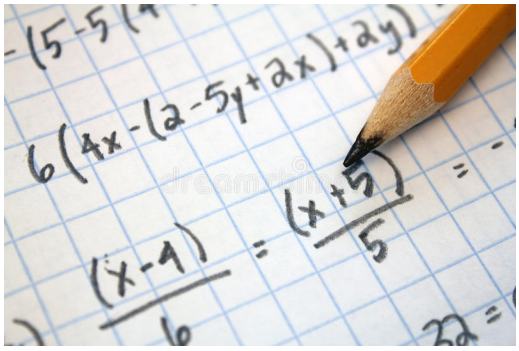

Figure 6: Paper Method

What I Like:

- Much better for teaching students
- Shows all the calculations.

What I Dislike:

- Some required calculations are not needed for the GCSE, so the students would need to learn lots of irrelevant equations.
- Takes a long time to work out larger circuits.
- Can have errors that will be taken forward.

What I Will Take Forward:

- Uses most of the same physics calculations as the ones in the course.

Comments:

Paper based calculations are an excellent way to learn, but students at GCSE level only need to learn a few equations, which are not enough to simulate entire circuits.

Students would need to learn difficult, advanced physics to calculate what my program will do, as this would be a waste of time for teachers and students, just to demonstrate a simple physical idea.

Research Questions

For physics teachers:

1. Are you currently satisfied with the electronics equipment in school? Why?
2. Would you feel comfortable teaching students electronics with a computer program?
3. Do you feel it would be beneficial to be able to send students assignments to make circuits at home on this program?
4. Would you agree that students would understand theories better if they were given an example of the theories working? E.g. seeing the current splitting between a parallel circuit.
5. Do you feel that a program that can be used on personal devices would be a helpful aid for home learning?
6. Have you ever used any kind of circuit planning tools before? If so, what was your opinion on it, and what would you like to see added?
7. Have you got anything else to add?

For physics students:

1. Are you currently satisfied with the electronics equipment in school? Why?
2. Do you think that being able to experiment with cicuits at home would aid in your learning?
3. Would you say that you understand theories better if they were given an example of the theories working?
4. What are the qualities of real circuits that you would like to be reflected in a piece of software?
5. Have you ever used any kind of circuit planning tools before? If so, what was your opinion on it, and what would you like to see added?
6. Have you got anything else to add?

Feature Requirements

Requirements	Explanation
Input circuits from textbook	The teacher will be able to show examples of circuits endorsed by the exam board.
Be able to be visible on a large screen	The teacher will probably use a projector to display the program, so the program should scale well on a computer.
Show information about the circuit at any point.	To show where current is diverted inside circuits, over distances, etc.
Change component settings during operation.	To show how different variables affect circuit operation, to aid in learning.
Work without noticeable lag on low-end hardware	Students and teachers may not have high-end hardware, so this program will need to run reasonably well on older and slower computers.

Requirements

General requirements:

- A screen - user needs to be able to see the program
- A keyboard & mouse - user needs these to navigate the program. A touchscreen is inadequate as the menu bar will be fairly small.
- Operating systems listed below with approximately 4GB RAM.
- This is because these systems are fairly up to date, and will be able to run compiled versions of my program.
- The RAM requirement is a common quantity of RAM. The program will probably run on less, but 4GB is a safe amount.
- No additional software dependencies are required as my output will be a self-contained executable.

Mac OSX:

- Mac OSX 10.10 or greater (x64).
- 4 GB RAM

Windows:

- Windows 7 or greater (x64).
- 4 GB RAM

Linux:

- RHEL 7 or equivalent (x64).
- 4 GB RAM