Cukurova University

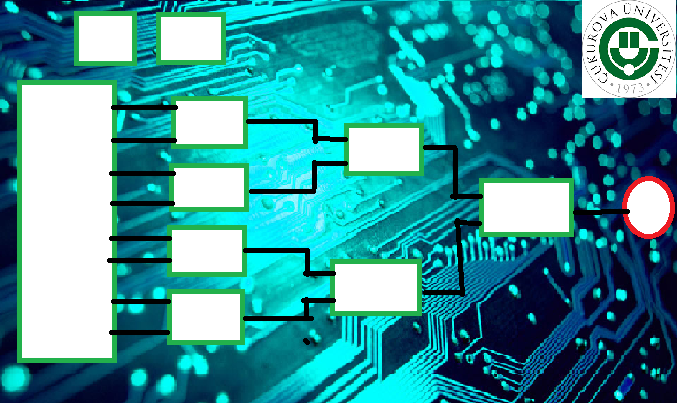
Fouad Asil | EEE425 | December 18, 2015

Super visor : prof. Turgay Ibrikci

LogiCAD V0.1

As an example for open source programs

Muh. Mim. Faculty

EEE department

# 

Contents

[Contents 0](#_Toc438223860)

[Introduction 1](#_Toc438223861)

[What is "Open Source" ? 1](#_Toc438223862)

[What is LogiCAD ? 1](#_Toc438223863)

[The general Algorithm 1](#_Toc438223864)

[The solver Array 2](#_Toc438223865)

[General Algorithm 4](#_Toc438223866)

[How can this program developed?? 6](#_Toc438223867)

[Conclusion 6](#_Toc438223868)

[References 6](#_Toc438223869)

# Introduction

This program was made for EEE425 Course in cukorova university . as a project .

The main Idea is develop an open source program by appling an algorithm starting form the base , with a great ability to

Develop it by any programmer , Using C# language .

# What is "Open Source" ?

It’s a computer software with its source code made available .

It’s the new era of developing and co-operating between all the programmers from every were .

The open source will make the programs develop very fast with respecting to the rights .

# What is LogiCAD ?

Logicad is an open software based on an algorithm which can solve any logic circuit using the a advantages of c# language with interface .

Its able to develop to cover the general ideas and any special circuit type .

The first open source version is V0 which can cover the series circuits with two gates (and , Or) . as an example of special propose using of this algorithm .

# The general Algorithm

Combining windows Interface benefits with Array’s Calculations .

Easy to understand and able to develop to specific circuits solver (As I showed in V0.1)

Can handle the most simple and most complex circuits .

The main Idea is to convert the circuit to an array , with ability to easy input using the interface .

SPICE Language Inspired Me .

# The solver Array

This Array is the Skeleton of this algorithm , it can fill by using programmed relations and formulas depending in the circuit or by the interface ! .

In LogiCAD , I used the both to show the ability's of this algorithm .

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** | **Level** | **1st inp** | **2nd inp** | **Type** | **Output** |
| **01** | n |  |  |  |  |
| **02** |  |  |  |  |
| **…** |  |  |  |  |
| **2^n** |  |  |  |  |
| **11** | n-1 |  |  |  |  |
| **12** |  |  |  |  |
| **13** |  |  |  |  |
| **…** |  |  |  |  |
| **10 + 2^(n-1)** |  |  |  |  |
| **21** | n-2 |  |  |  |  |
| **22** |  |  |  |  |
| **23** |  |  |  |  |
| **…** |  |  |  |  |
| **20 + 2^(n-2)** |  |  |  |  |
| **…** |  |  |  |  |  |
| **…** |  |  |  |  |  |
| **…** |  |  |  |  |  |
| **n\*10 + 2^0** | 1 |  |  |  |  |

Example for solver Array when N = 4 (series) after solution (LogiCAD v0)

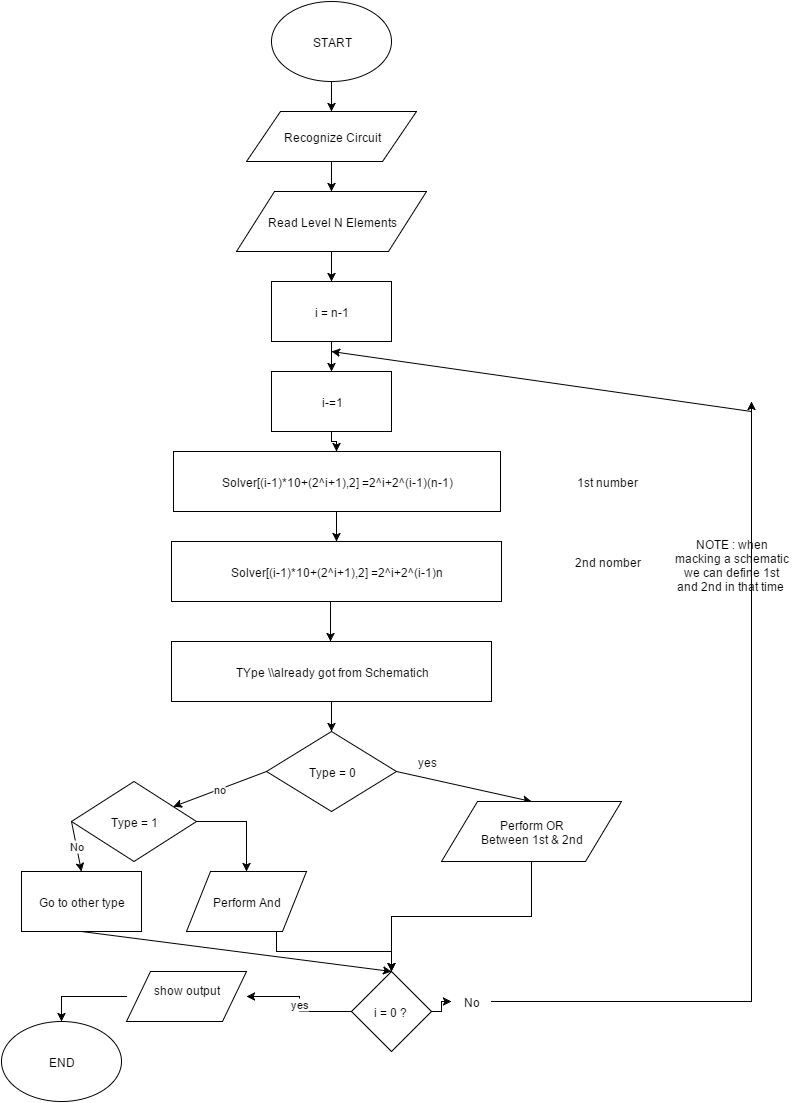
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Key** | **Level** | **1st** | **2nd** | **Type** | **output** |
| **1** | 4 | 0 | 0 | 0 | 1 |
| **2** | 4 | 0 | 0 | 0 | 0 |
| **3** | 4 | 0 | 0 | 0 | 1 |
| **4** | 4 | 0 | 0 | 0 | 0 |
| **5** | 4 | 0 | 0 | 0 | 1 |
| **6** | 4 | 0 | 0 | 0 | 0 |
| **7** | 4 | 0 | 0 | 0 | 1 |
| **8** | 4 | 0 | 0 | 0 | 0 |
| **11** | 3 | 1 | 0 | 1 | 0 |
| **12** | 3 | 1 | 0 | 0 | 1 |
| **13** | 3 | 1 | 0 | 0 | 1 |
| **14** | 3 | 1 | 0 | 1 | 0 |
| **21** | 2 | 0 | 1 | 0 | 1 |
| **22** | 2 | 1 | 0 | 1 | 0 |
| **31** | 1 | 1 | 0 | 0 | 1 |

# General Algorithm

Is defines as this :

1. Recognize inputs (code and interface)
2. Read Inputs
3. Make calculations for last level
4. Use it to solve n-i level …
5. Show the output

And can be described by the following flow chart :



# How can this program developed??

For the general case we can use The interface to determined the elements of the solver array .

Also we can apply next ideas :

* Add a special Gates The ability to Draw the schematic
* Connect the inputs and the output outside !
* Dealing with Errors .
* Active Helper !

And As an open-source program any one can do this !

# Conclusion

I made an open source program using an algorithm which I developed for this goal , and I applied a special purpose application to show the power of my algorithm in the simplest way , and I'm planning to publish this open-source , and share it with all the world !

# References

msdn.microsoft.com

stackoverflow.com

www.codeproject.com

Wilipedia.com

Wikipedia.com

Our Lectures Notes